'O' LEVEL

made simple

Based on NIELIT 'O' Level Revised Syllabus for the Year 2020 for Module 2(M2-R5)

Covers Syllabus for A2-R5 papers of NIELIT 'A' Level Examinations

WEB DESIGNING AND PUBLISHING

Included Sample Papers



- Web Design, HTML Basics
- Cascading Style Sheet
- CSS Framework
- JavaScript and Angular JS
- Photo Editor
- Web Publishing and Browsing
- January 2021 Exam Paper
- Solved Paper and Unsolved Paper to Practice

PROF. SATISH JAIN
M. GEETHA IYER

Scanned with OKEN Scanner

[Based on NIELIT 'O' Level Revised Syllabus for Year 2020 for Module 2(M2-R5)]

'O' Level

Made simple

WEB DESIGNING AND PUBLISHING

(Also Covers Syllabus for A2-R5 papers of NIELIT 'A' Level Examinations)

KGNT COMPUTER, SIKTOL, NIJAPADA
103/1
23

by

Satish Jain

B.Sc., B.E. (II Sc), M.E. (II Sc), M.Tech. (IIT Kanpur),
Wing Commander (Retd.)
Ex Professor of Information Technology,
Institute of Information Technology & Management,
GGS Indraprastha University, Delhi.

M. Geetha

B.Com., Diploma (IT)

bpb

Scanned with OKEN Scanner

FIRST EDITION 2020
SECOND REVISED AND UPDATED EDITION 2021
Reprinted 2022

Copyright © BPB Publications, INDIA

ISBN: 9789389845747

All Rights Reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise without the prior written permission of the publisher with the exception that the program listing may be entered, stored and executed in a computer system, but they may not be reproduced for publication.

LIMITS OF LIABILITY AND DISCLAIMER OF WARRANTY

This information contained in this book is true and correct and the best of Author's and publisher's knowledge. We are committed to serve students with best of our knowledge and resources. We have taken utmost care and attention while editing and printing of this book but Author and publisher should not be held responsible for errors, omission and any other unintentional mistake that might have crept in However, errors brought to our notice shall be gratefully acknowledged and attended to.

All trademarks referred to in the book are acknowledged as properties of their respective owners. This book is written solely for the benefit and guidance of the students.

Distributors:

BPB PUBLICATIONS
20, Ansari Road, Darya Ganj,
NEW DELHI-110002
Ph:23254990/23254991

BPB BOOK CENTRE
376 Old Lajpat Rai Market,
DELHI-110 006
Ph: 23861747

MICRO MEDIA
Shop No. 5, Mahendra Chambers, 150
DN Rd. Next to Capital Cinema, *V.T.*
(C.S.T.) Station, MUMBAI-400001
Ph: 22078296/22078297

DECCAN AGENCIES
4-3-329, Bank Street,
HYDERABAD-500195
Ph: 24756967/24756400

NIELIT is in no way connected with this book. This book is written solely for the benefit and guidance for students appearing for NIELIT Examination

Published by Manish Jain for BPB Publications, 20, Ansari Road, Darya Ganj, New Delhi- 110002 and
Printed at Akash Press, Delhi

Scanned with OKEN Scanner

Dedicated to

Sri Raja Rajeshwari and Sri Sai Baba

Whose blessings will overcome all the hurdles in life

Scanned with OKEN Scanner

PREFACE

This book is an up-to-date revised edition. It is based on NIELIT 'O' Level Module-2 syllabus for Web **Designing and Publishing**. Our objective in bringing out this book in this form is to explain to the students, technical concepts in an easy-to-understand language and in a simple manner. The purpose of this module is to provide skill to students in designing layouts of web sites by using a combination of HTML, CSS, and JavaScript to create web pages. It is observed that most of the books available in the market do not explain these tools and techniques in a step-by-step manner from the first time user's point of view. We have pressed into service all our Teaching and Professional experience in writing this book in an easy-to-understand Language along with use of examples and Figures. A very useful feature of this book is that different types of HTML tag are explained with practical examples, so as to enable a reader to understand them without any outside help.

Changes in the field of computer science including Web designing are taking place very fast and many new concepts/terminologies are added from time to time. Hence, the scope for the questions set in the examinations will be much wider. This has led us to insert latest topics and concepts. These new topics include HTML5 new elements, CSS Framework and Angular JS. This book includes current trends on Internet such as Client and Server side scripting languages, types of websites and AngularJS. It is a client side JavaScript MVC framework to develop a dynamic web application. It is hoped that these topics will also be useful for students taking any other professional examinations related to Web Designing.

We shall feel obliged to the readers if they send us their critical opinion regarding the presentation, readability and coverage in this book and give us suggestions, if any, to improve the subject matter.

15th April, 2020

Authors

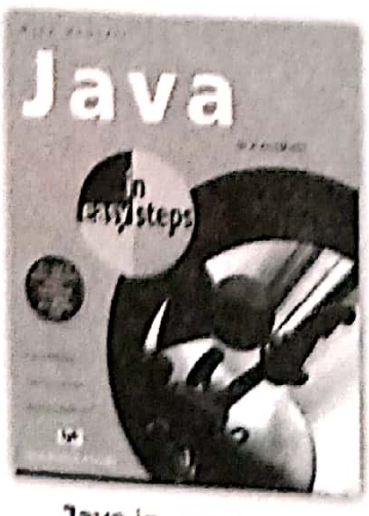
OTHER TITLES OF INTEREST



Blockchain for Business with Hyperledger Fabric
ISBN: 9789389328608



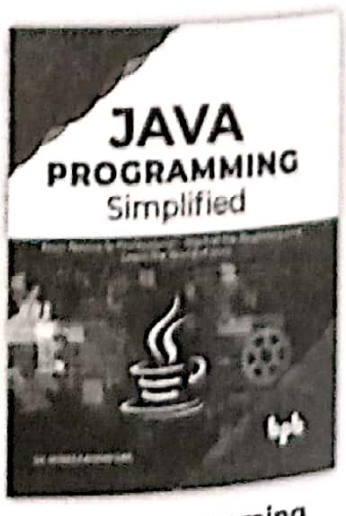
100+ Solutions in Java
ISBN: 9789389845600



Java in easy steps
ISBN: 9789387284210



Let us Java - 5th Edition
ISBN: 9789388176385



Java Programming Simplified
ISBN: 9789389845143

Scanned with OKEN Scanner

M2-R5: Web Designing and Publishing

Outline of Course

S.No	Topic	Written Marks (Max.)
1.	Introduction to Web Design and Editors, HTML Basics	25
2.	Cascading Style Sheets (CSS)	20
3.	CSS Framework	15
4.	JavaScript and Angular JS	20
5.	Photo Editor, Web Publishing and Browsing	20
6.	Total	100

Detailed Syllabus

1. Introduction to Web Design

Introduction of Internet, WWW, Website, Working of Websites, Webpages, Front End, Back End, Client and Server Scripting Languages, Responsive Web Designing, Types of Websites (Static and Dynamic Websites).

2. Editors

Downloading free Editors like Notepad++, Sublime Text Editor, making use of Editors, File creation and editing, saving.

3. HTML Basics

HTML: Introduction, Basic Structure of HTML, Head Section and Elements of Head Section, Formatting Tags: Bold, Italic, Underline, Strikethrough, Div, Pre Tag Anchor links and Named Anchors Image Tag, Paragraphs, Comments, Tables: Attributes –(Border, Cellpadding, Cellspacing , height , width), TR, TH, TD, Rowspan, Colspan Lists : Ordered List , Unordered List , Definition List, Forms, Form Elements, Input types, Input Attributes, Text Input Text Area, Dropdown, Radio buttons , Check boxes, Submit and Reset Buttons Frames: Frameset, nested Frames.

HTML 5 Introduction, HTML5 New Elements: Section, Nav, Article, Aside, Audio Tag, Video Tag, HTML5 Form Validations: Require Attribute, Pattern Attribute, Autofocus Attribute, email, number type, date type, Range type, HTML embed multimedia, HTML Layout, HTML Iframe.

4. CSS

Introduction to CSS, Types of CSS, CSS Selectors: Universal Selector, ID selector, Tag Selector, Class Selector, Sub Selector, Attribute Selector, Group Selector, CSS Properties: Back Ground properties, Block Properties, Box properties, List properties, Border Properties, Positioning Properties, CSS Lists CSS Tables, CSS Menu Design CSS Image Gallery.

Scanned with OKEN Scanner

5. CSS Framework

Web Site Development using W3.CSS Framework, W3.CSS Intro, W3.CSS Colors, W3.CSS Containers, W3.CSS Panels, W3.CSS Borders, W3.CSS Fonts, W3.CSS Text, W3.CSS Tables, W3.CSS List, W3.CSS Images, W3.CSS Grid.

6. JavaScript and Angular JS

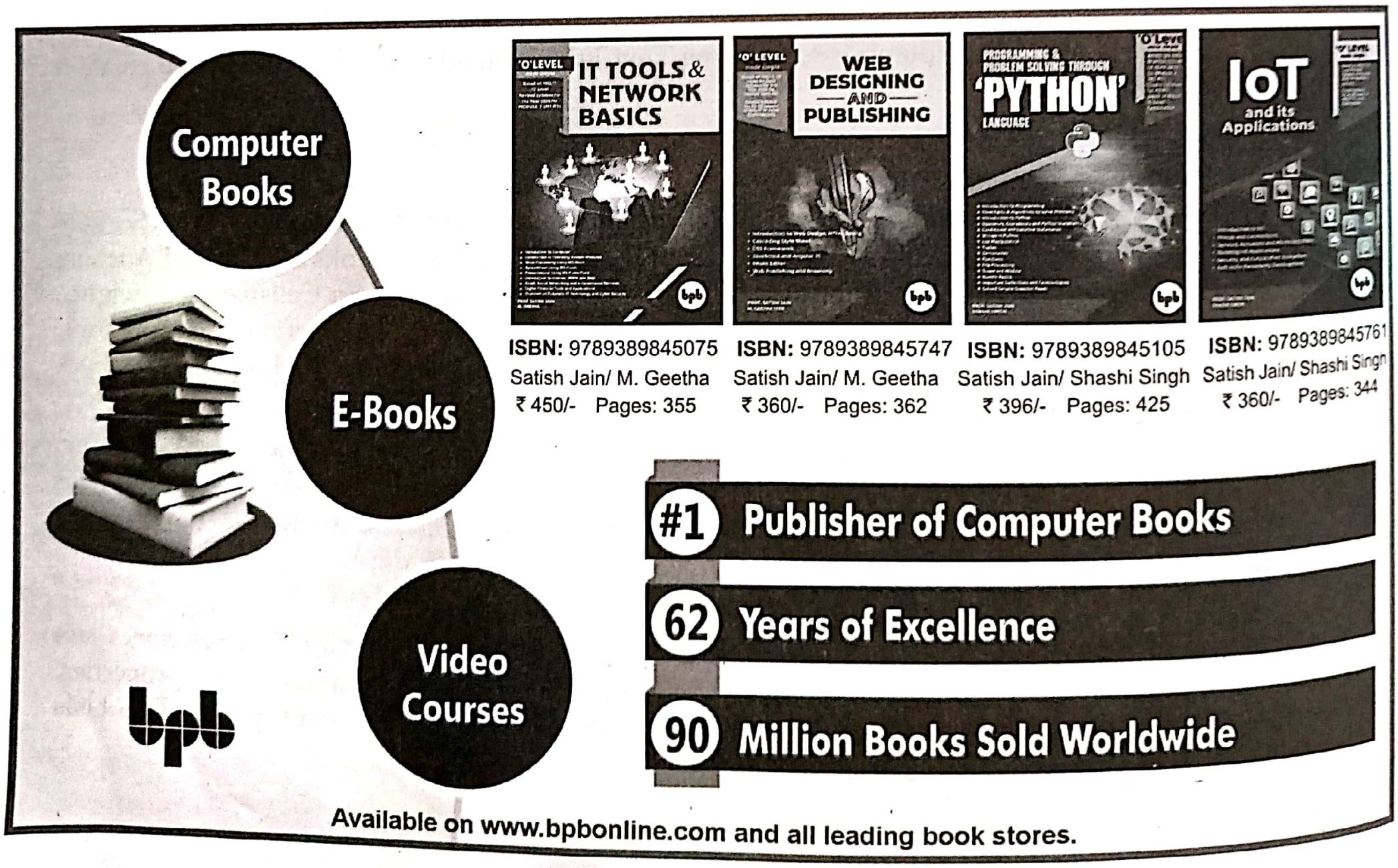
Introduction to Client Side Scripting Language, Variables in Java Script, Operators in JS, Conditions Statements, JS Popup Boxes, JS Events, Basic Form Validations in JavaScript. Introduction to Angular JS: Expressions, Modules and Directives.

7. Photo Editor

Features of Photo Editing: Tools: Selection Tools, Paint Tools, Transform Tools, Text Tool, Layers, Brightness/ Contrast, Improve Colors and tone, Filters.

8. Web Publishing and Browsing

Overview, SGML (Standard Generalized Markup Language), Web hosting Basics, Documents Interchange Standards, Components of Web Publishing, Document management, Web Page Design Considerations and Principles, Search and Meta Search Engines, WWW, Browser, HTTP, Publishing Tools.



Scanned with OKEN Scanner

USEFUL TIPS FOR THE EXAMINEE

Dear Reader,

Every examination has its peculiarity and the best results are often obtained by identifying that particular feature and working with the help of right strategies, which in turn can bead opted by correctly analyzing the question paper.

The 'O' Level Examination being no exception to this rule, the authors present an effective aid to guide you to analyze the question paper and thus help to chalk out your own effective plan of action.

The instructions given on the first page of the question paper should always be read very carefully. As the answer sheets of Part ONE will be evaluated with the help of a computer, you have to be extra cautious.

The answer sheet contains separate boxes for writing answers to each question in this part. You should read the following paragraphs to avoid mistakes and clear the first part without any difficulty.

How to avoid mistakes?

If you are not sure about a particular answer- do not fill the box in the first try itself, but make an attempt later. After having given an initial reading to all the questions, first attempt only those questions about which you are very sure regarding the correct answer. It is while taking a second go at the questions, that you answer the remaining ones, always making sure that you spend only a limited amount of time on each one of them. Finally, go through all the answers once decided and fill the boxes with black ink as per the instructions given in the question paper.

How to answer the questions of Part TWO?

Once you have done well in Part ONE, rest assured, passing Part TWO will not pose much of a problem, if you have effectively followed the instructions given below. While answering Part TWO, the time limit should be strictly kept in mind and every effort must be made to complete all the questions, by keeping the answers brief and to the point. Remember you can also resort to making a sketch/diagram in your answer wherever possible, instead of describing it in words, because "a thousand words of description can be effectively expressed by a small diagram." As the number of questions are normally more for an average student, this precious time saved, enables you to have a wider coverage of the paper.

While attempting the descriptive type of questions, keep a check on the length of your answers and keep in mind the time frame available for answering the remaining questions.

The last stage, being the revision stage, you should save at least 5 to 10 minutes to revise all your answers. Try to find the gross errors, like units of measurement, or wrong question number you might have put for a specific answer, etc. This does happen because of the inherent tension of the examination.

This book has covered the complete syllabus in an easy to understand manner. It is advised, however, that in addition to this book, you should also refer to a few books on 'Web Designing and Publishing'. This will enable you to broaden the horizon of your knowledge.

Make your own notes, by reading each topic from this book and other standard books. Once you are quite sure about the comprehension of the topic, try to answer the Exercises given at the end of each chapter of this book.

In the second reading, go through the sample papers given at the end of the book. Solve these papers without consulting any book. After you have answered them, compare your response with the relevant matter available in the text of this book. Where you find your answers differ, re-study that part again.

WISH YOU A GRAND SUCCESS in the 'O' Level Examination, and a very bright future in the field of computer science.

Authors

Scanned with OKEN Scanner

Contents

Chapter 1 Introduction to Web Design 1

1.1 Introduction 1
1.2 Introduction to Internet 1
1.2.1 History of Internet 2
1.2.2 Features of Internet 3
1.2.3 Applications of Internet 3
1.3 World Wide Web (WWW) 5
1.3.1 Evolution of WWW 5
1.4 What is Website? 6
1.4.1 How the website works? 6
1.5 Front End and Back End Application 9
1.5.1 Front End Development 9
1.5.2 Back End Development 10
1.6 Client and Server Scripting Languages 11
1.6.1 Client side Scripting 11
1.6.2 Server side Scripting 11
1.7 Responsive Web Designing 12
1.8 Downloading free Editors 13
1.8.1 Notepad++ 14
1.8.2 Sublime Text Editors 20
Review Questions with Answers 27

Chapter 2 HTML Basics 35

2.1 Introduction 35
2.2 Introduction to HTML Basics 35
2.2.1 Basic Structure of HTML 35
2.3 Formatting Tags 39
2.3.1 Using the HTML <Div> Tag 40
2.3.2 Absolute Links 43
2.3.3 IMG (Image Element) 50
2.3.4 Paragraph <P> Tag 50
2.3.5 Comments in Html Document (<!>) 53
2.4 HTML TABLES 53
2.4.1 Basic Table Tags 53
2.4.2 Creating a Table 54
2.4.3 Table Attributes 54
2.5 HTML Lists 65
2.5.1 Ordered List <Ol> 65
2.5.2 Unordered List <Ul> 66
2.5.3 Definition List 67
2.6 HTML Forms 68
2.7 Form's Elements 69
2.8 HTML Form Program 76
2.9 HTML FRAMES 78
2.10 HTML5 Introduction 83
2.10.1 HTML5 Form Validations 89
2.10.2 HTML5 Attributes 90
2.10.3 New Input Types in HTML5 92
2.10.4 HTML Form Program 96
2.11 HTML Embed Multimedia 98
2.12 HTML Layout 101
2.12.1 Layout using Tables 101
2.12.2 Div Based Layout 102
2.12.3 HTML Iframe 104
Review Questions with Answers 106

Chapter 3 Cascading Style Sheet 115

3.1 Introduction 115
3.2 Introduction to CSS 115
3.3 Types of CSS 117
3.3.1 CSS Selectors 122
3.3.2 CSS Universal Selector 126
3.3.3 CSS Grouping Selector 126
3.3.4 Attribute Selector 127
3.3.5 Sub Selector 135
3.3.6 CSS Tag Selector 138
3.4 CSS Properties 139
3.4.1 CSS Background Properties 141
3.4.2 CSS Background Image 142
3.4.3 CSS Background Repeat 143
3.4.3 CSS Background Position 145
3.4.3 CSS Background Attachment 148
3.5 Border Properties 153
3.5.1 Border-style 153
3.5.2 Border-width 156
3.5.3 Border-color 156

Scanned with OKEN Scanner

3.5.4 Border Properties using Shorthand 158
3.5.5 CSS Border Radius 158
3.6 Block Properties 160
3.7 CSS BOX Properties 164
3.7.1 Positioning Properties 166
3.8 CSS Lists 171
3.9 CSS Tables 176
3.9.1 Table Borders 178
3.9.2 Table Border Radius 179
3.9.3 Colors in Table 179
3.9.4 Collapse Table border 180
3.9.5 Table Stripped 183
3.9.6 Padding in Table 184
3.9.7 Border Image in Table 185
3.10 CSS Menu Design 193
3.11 CSS Image Gallery 204
Review Questions with Answers 206

Chapter 4 CSS Framework 213

4.1 Introduction to W3.CSS 213
4.2 Website Development using W3.CSS Framework 213
4.2.1 How to use W3.CSS? 213
4.3 W3.CSS Colors Basics 214
4.3.1 W3-Color 214
4.3.2 W3-Text-Color 216
4.3.3 W3-CSS Hover Colors 216
4.3.4 W3-hover-text-color 217
4.3.5 W3.CSS Containers 217
4.3.6 Articles and Sections 219
4.3.7 Container Padding 220
4.4 W3.CSS Panels 220
4.4.1 Using panels for creating notes 221
4.4.2 W3.CSS Rounded Panels 222
4.4.3 W3CSS Closing and Opening Panels 223
4.4.4 W3CSS Card 224
4.4.5 Hover Shadow Effect 224
4.5 W3.CSS Borders 225
4.5.1 W3.CSS Thick Borders 226
4.5.2 W3.CSS Rounded and Hoverable Borders 227
4.5.3 W3.CSS Fonts 228
4.5.4 W3.CSS External Fonts 229
4.6 W3.CSS Text 230
4.6.1 W3CSS Text Alignment 231
4.6.2 Wide and Slim Text 232
4.6.3 Shadow and Opacity 232
4.7 W3.CSS Tables 233
4.7.1 W3.CSS Table border Strips border 233
4.7.2 Table with a Colored Heading 234
4.7.3 Hoverable Tables 235
4.7.4 Table as Card 236
4.7.5 Responsive Table 237
4.7.6 W3.CSS XXLarge 238
4.8 W3.CSS Lists 239
4.8.1 Basics WS.CSS List 239
4.8.2 Lists as Cards 240
4.8.3 Hoverable Lists with Close Option 241
4.8.4 W3.CSS List Sizes 243
4.9 W3.CSS Images 243
4.9.1 Images as Cards 244
4.10 W3.CSS Image Effects 245
4.10.1 Adding Opacity to Images 245
4.10.2 Creating Grayscale Images 246
4.10.3 Creating Sepia Image Effect 246
4.10.4 Creating Hover Effects 247
4.11 W3.CSS Grid 249
4.11.1 Two equal and unequal columns 249
4.11.2 W3.CSS Mixed Grid Layouts 252
4.11.3 Mixed Layout for Mobile, Tablet and Desktop 253
Review Questions with Answers 254

Chapter 5 JavaScript and Angular JS 263

5.1 Introduction 263
5.2 Introduction to Client Side Scripting Language 263
5.2.1 Advantages and disadvantages of client-side scripting 264

Scanned with OKEN Scanner

5.3 Variables in JavaScript 265
5.4 Operators in JavaScript 276
5.5 Conditional Statements in JavaScript 286
5.5.1 Different Types of Conditional Statements 286
5.6 Display Popup Message Box 290
5.6.1 Alert Box 290
5.6.2 Confirm Box 290
5.6.3 Prompt Box 291
5.7 JavaScript Events 292
5.7.1 HTML Events 293
5.8 Form Events 295
5.8.1 Focus Event (onfocus) 296
5.9 Basic Form Validations in JavaScript 299
5.9.1 Creating the HTML Form 299
5.10 Introduction to Angular JS 302
5.11 Angular JS Expression 310
5.11.1 AngularJS Numbers 312
5.11.2 Angular JS Strings 312
5.11.3 Angular JS Arrays 313
5.11.4 What is AngularJS Modules? 314
5.11.5 Modules and Controllers in Files 315
Review Questions with Answers 316

Chapter 6 Photo Editor **322**

6.1 Introduction 322
6.1.1 Main Features of Photo Editing 322
6.1.2 Photo Editor Tools and Controls 322
6.1.3 Using Selections 322
6.1.4 Selection Tools 323
6.1.5 Combining the Tools 325
6.1.6 Using the Edit Brush 326
6.1.7 Text Tool 326
6.1.8 Layers 328
6.1.9 Brightness/Contrast 329
6.1.10 Using the Standard Tab 330
6.1.11 Improve Colors and Tone 331
6.1.12 Filters 331
Review Questions with Answers 332

Chapter 7 Web Publishing and Browsing **334**

7.1 Introduction 334
7.2 Overview of World Wide Web 334
7.3 SGML (Standard Generalized Markup Language) 334
7.4 Web Hosting Basics 335
7.4.1 Hosting Types 336
7.4.2 Basic Features of a Web Hosting 336
7.5 Document Interchange Standards 338
7.5.1 Connectionless and Connection -Oriented Protocols 338
7.5.2 Stateless and Stateful Protocols 339
7.6 Components of Web Publishing 339
7.7 Document Management 340
7.8 Web Page Design Consideration and Principles 340
7.8.1 Planning your Website Design 340
7.8.2 Creating Goals for your Site 341
7.8.3 The Website Design and Development Process 344
7.8.4 How to Develop a Website? 346
7.9 Search engines 347
7.10 Meta-Search Engines 347
7.11 World Wide Web (WWW) 348
7.12 Web Browser 348
7.13 HTTP 348
7.14 Publishing Tools 349
Review Questions with Answers 352
Model Questions 360

Paper January 2021 **363**
Solved Sample Paper 1 **366**
Unsolved Sample Paper 1 **374**

Scanned with OKEN Scanner

CHAPTER 1

Introduction to Web Design

1.1 Introduction

Web Design is the process of publishing content on the Internet. It includes creating and uploading Websites, updating the associated Web Pages, and posting content to these Web pages online. The content meant for Web publishing can include text, videos, digital images, artwork, and other forms of media.

Publishers must possess a Web server, a Web publishing software, and the Internet connection to carry out Web publishing. Web publishing is also known as **online publishing.**

Structure

- Introduction to Internet and WWW
- Working of Websites
- Web pages and Front and Back End
- Client and Server Side Scripting Languages
- Responsive Web Designing
- Types of Websites such as Static and Dynamic
- Downloading Editors like Notepad++, Sublime Text Editor
- Make use of Editors
- File creation editing and saving

Objectives

- The reader will be able to understand the following:
- Types of website
- Understand the role of front end and back end application
- Understand the concept of client side scripting and server side scripting
- Difference editors available for writing code
- Understand working of editors

1.2 Introduction to Internet

Internet is an interconnection between several computers of different types belonging to various networks across the world. It is a network of networks. Millions of people use the Internet to search and share information and ideas, etc. It has grown into an important infrastructure supporting a widespread, multi-disciplinary community. This community now consists of students, scientists and researchers, large corporations, non-profit organizations, government agencies, and individual consumers.

The Internet is an immensely complex combination of thousands of technologies and dozens of services used by millions of people around the world each day. *Figure 1.1* shows a Web of computers which is laid around the globe. The Internet links thousands of computer networks. Every network and every computer in these networks exchange information according to certain rules called **protocols.**

Scanned with OKEN Scanner

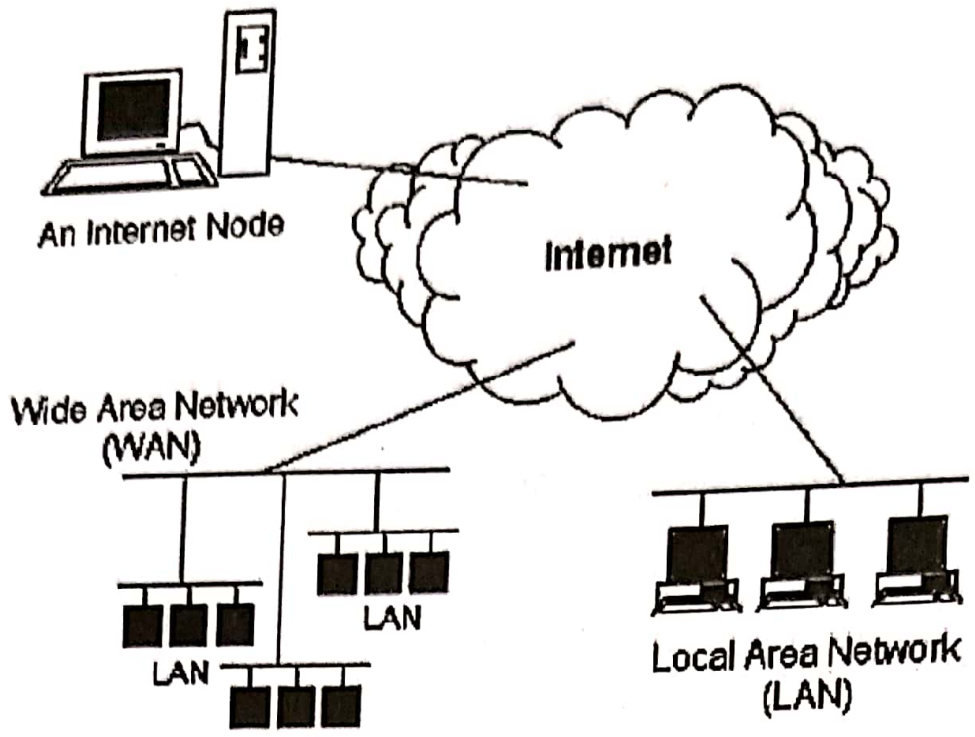


Figure 1.1: Interne is the network of computer connected with one another all over the world

These different computers and networks are united with the common thread of two protocols, i.e., Internet **Protocol (IP)** and **Transmission Control Protocol (TCP).**

☞ *Internet is a computer-based worldwide information network. It is the network of computers connected with one another all over the world.*

1.2.1 History of Internet

The Internet started as a U.S. government project in the year 1969 called the ARPANET (**A**dvanced **R**esearch **P**rojects **A**dministration Network) which supervised it in the beginning. The ARPANET reached universities, research laboratories and some military labs.

In the late 1980s, the National Science Foundation of U.S funded the development of a network (using the Internet protocols), named **NSFNET**, to connect super computer centers in the U.S. Many colleges and universities were encouraged to connect to that network. The number of sites increased rapidly.

There were more than 10,000 sites in 1987 and more than 100,000 in 1989. Similar activity, although not on such a large scale, was taking place in other countries as well. This large worldwide collection of networks and computer systems communicating according to the same protocols has come to be what is called the **Internet.**

In 1990, ARPANET was dismantled, and the public network in the U.S. was turned over to **NSFNET**. In the early 1990s, commercial networks with their own Internet exchanges or gateways were allowed to conduct business on the Internet, and in 1993 the NSF created the InterNIC to provide the Internet services.

The exponential growth of the Internet along with the inclusion of commercial networks and services has been accompanied by an astounding increase in the population of Internet users. The Internet is reaching the size and importance of an infra-structure. From a mere research project, the Internet has grown rapidly into something that involves millions of people worldwide.

The backbone of the Internet is made up of highspeed communication links owned by companies like **AT&T (American Telephone and Telegraph Company)**, Sprint or **VSNL (Videsh Sanchar Nigam Limited)** in India. Connections to the backbone are made by *Internet* **Service Providers (ISPs)** who then sell access bandwidth to individuals or commercial organizations.

Scanned with OKEN Scanner

1.2.2 Features of Internet

World Wide Web: It is a part of the internet,

- E-mail
- News
- Telnet
- FTP

E-mail: Electronic mail is the most reason to use the Internet. You need an e-mail program and an account on an Internet mail server with a domain name to create, send and receive e-mail messages.

News: One Internet based service called **news**, includes tens of thousands of newsgroups. Each newsgroup hosts discussion on a specific topic. It is a system of special interest discussion groups, called **newsgroups**, to which readers can send, or Post messages which are then distributed to other computer in the network.

Telnet: It is a specialized service that lets you use one computer to access the contents of another computer a telnet host. A telnet program creates a Window into the host, so you can access files, issue commands and exchange data.

File Transfer Protocol: It is the internet tool used to copy files from one computer to another. Using a FTP program or a web browser, you can log into an ETP host computer over the internet and copy files on to your computer.

Internet Relay Chat: It is a service that allows users to communicate in real time by typing text in a special window. You can use a special IRC program to participate in chat room discussion but many chat rooms are set up in websites, to chat directly in their browser Window.

Internet has the following characteristics:

a. **A Complex Network:** With the simplified definition as a "**network of networks,**" that comprises over 150 million computers.
b. **Disorganized:** The Internet can be cumbersome and confusing, even for experienced users.
c. **A Decentralized System:** Millions of individual networks and over 140 million individual computers connected throughout the world.
d. **Composed of billions of Files:** Files pertaining to thousands of subjects, disciplines and professions are available in different file formats.
e. **Widely Used:** More than 147 million people use the Internet, over 40 million of whom use it daily.
f. **International in Scope:** This global network is accessed by people in approximately 140 countries; people in over 155 countries use Internet for electronic mail purpose.
g. **Expanding Exponentially:** The Internet is growing at the rate of 12% per month.

1.2.3 Applications of Internet

With the help of Internet, you can:

- Exchange messages using e-mail (Electronic mail).
- Transfer files as well as software.
- Browse through information on any topic on web.
- Communicate in real time (chat) with others connected to the Internet.
- Search databases of government, individuals and organizations.
- Read news available from leading news groups.
- Send or receive animation and picture files from distant places.

Scanned with OKEN Scanner

- Set up a site with information about your company's products and services.

Advantages of Internet

The Internet provides opportunities and can be used for a variety of things. Some of the things that you can do via the Internet are:

a. **E-mail:** E-mail is an online correspondence system. With e-mail, you can send and receive instant electronic messages, which works like writing letters. Your messages are delivered instantly to people anywhere in the world.

b. **Access Information:** The Internet is a virtual treasure trove of information. Any kind of information on any topic under the sun is available on the Internet. The `**search engines**' on the Internet can help you to find data on any subject that you need.

c. **Shopping:** Along with getting information on the Internet, you can also shop online. There are many online stores and sites that can be used to look for products as well as buy them using your credit card. You do not need to leave your house and can do all your shopping from the convenience of your home.

d. **Online Chat:** There are many chat rooms on the Web that can be accessed to meet new people, make new friends, as well as to stay in touch with old friends.

e. **Downloading Software:** This is one of the most happening and fun things to do via the Internet. You can download innumerable, games, music, videos, movies, and a host of other entertainment software from the Internet.

Disadvantages of Internet

There are certain dangers related to use of the Internet. These are as follows:

a. **Theft of Personal Information:** If you use the Internet, you may be facing grave danger as your personal information, such as name, address, credit card number, etc., can be accessed by other culprits to make your problems worse.

b. **Spamming:** Spamming refers to sending unwanted e-mails in bulk, which provide no purpose and needlessly obstruct the entire system.

c. **Virus Threat:** Virus is nothing but a program which disrupts the normal functioning of your computer systems. Computers attached to the Internet are more prone to virus attacks and they can end up into crashing your whole hard disk, causing you considerable headache.

d. **Pornography:** This is perhaps the biggest threat related to your children's healthy mental life. A very serious issue concerning the Internet. There are thousands of pornographic sites on the Internet, which can be easily found and can be a detrimental factor to letting children use the Internet.

How the Internet Works?

Computers need specific software to use TCP/IP. This software, often called the **TCP/IP stack**, is included in Windows Operating system. The following points see how the internet works.

a. All computers on the internet use a protocol called **TCP/IP (Transmission Control Protocol/Internet Protocol)**. A protocol is a set of rules that describes how computers communicate.

b. When you send a message across the Internet to another use, TCP divides the data into units called packets and attaches to each packet. The information is necessary to reassemble the data and check for errors. IP then labels all the packets with a header containing the address of the destination and sends them on their way.

c. Once the packets reach their destination, the computer on the receiving end removes the IP header and uses the data that TCP attached to each packet to make use none of the packets have been lost or damaged. The packets are then reassembled into the original message.

Scanned with OKEN Scanner

d. Computers need specific software to use TCP/IP.
e. The hardware used to connect computers to the internet varies. If you access the Internet from home, you most likely use a modem.
f. If you browser the Internet from a computer, you are probably connecting through a company LAN with an Internet gateway. A gateway is a special computer on a LAN that communicates with the Internet.
g. The speed at which you can send and receive information depends on both the speed on modem and the available bandwidth. Bandwidth is the capacity of data lines to carry Internet traffic.

1.3 World Wide Web (WWW)

The World Wide Web, or simply the "Web" is a global information system that is the fastest-growing area of the Internet. In other words World Wide Web) is a series of servers that are interconnected through hypertext. Hypertext is a method of presenting information in which certain text are highlighted, that, when selected, displays more information on that particular topic (see *Figure 1.2*).

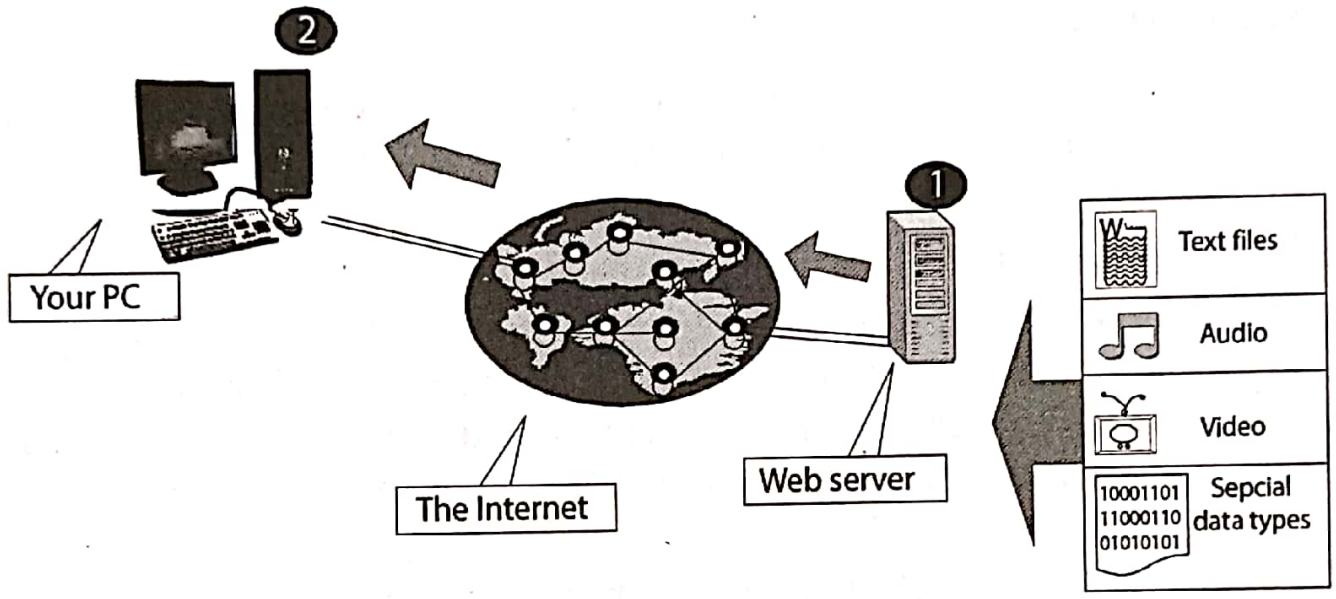


Figure 1.2: Some data types available on the Web

The WWW is an Internet-based global information system. It makes available multimedia information from over 4 million computers around the world. The Web offers video, interactive multimedia and live audio, in addition to more basic data types, such as text documents and still photographs (see *Figure 1.2*).

1.3.1 Evolution of the WWW

The WWW is a huge collection of hypertext pages on the Internet. The concept of WWW was developed in **Switzerland** at the **European Particle Research Centre (known as CERN)**, in the year 1989. The first text-based prototype was operational in 1991. In the month of December 1991, a public demonstration was given at Hypertext '91 conference in **San Antonio, Texas (USA)**. In the year 1993, the first graphical interface software package called **Mosaic** was released.

Hypertext enables you to read and navigate text and visual information in a non-linear way based on what you want to know next, unlike a textbook where the subject is described continuously or linearly.

The Mosaic became so popular that a year later, the author of Mosaic, namely, Marc Andressen left the National Center for Supercomputing Applications, where Mosaic was developed, forming a company called **Netscape Communications Corporation**. This company developed the clients, servers and other Web software.

Scanned with OKEN Scanner

In the year **1994, CERN and MIT of USA** signed an agreement setting up the World Wide Web Consortium, an organization devoted to further developing the Web, standardizing protocols, and interoperability between sites. Since then time, hundreds of universities and companies have joined the Consortium.

In the first year after Mosaic was released, the number of WWW servers grew from 100 to 7000. The growth is expected to be exponential in the years to come and will probably be the force driving the technology and use of the Internet into every walk of life.

Web servers on the Internet are collectively referred to as the World Wide Web. The W3C is the closest anyone gets to set the standards for and enforcing rules about the WWW. You can visit the Consortium's home page at http://www.w3c.org/. The second group of organizations that influences the Web is the browser developers, most notably Netscape Communications Corporation and Microsoft Corporation of USA.

To access the Web server, the user use a client software called a **browser program.** With a browser, you can choose an element on the web page, which can then cross-link us to a computer animation, or play sound, or show another web page. Browsers can even contact another Web server located across the world.

1.4 What is Website?

A Web site is a collection of Web pages associated with a particular person, business, government, school and organization. Website are stored on a Web server, a special computer that makes Web pages available for people to browser. Websites can consist of only a few Web pages or many hundreds of Web pages.

1.4.1 How the Website Works?

A website is a collection of web pages. These web pages are usually located on a web server that is connected to the Internet. The web server is a computer that has been built specifically to host websites, and contains web server software. The web server is usually located with a web hosting provider. The web server is usually located with a web hosting provider (i.e. a company that provides web hosting to its customers).

Before you being creating your own website and launch it to the Internet, it is important to know how websites work.

- A website is simply a collection of web pages of code that describes the layout, format and content on a page.
- The web server is an internet-connected computer that receives the request for a web page sent by your browser.
- The browser connects the computer to the server through an IP address. The IP address is obtained by translating the domain name.

In order to display your website on the Internet, you will need:

- A Website
- A domain name
- A server

Website

A website is typically a collection of web pages, images and other elements that are linked together to form a larger, structured document. A website can be made up of a single page or it could have thousands of pages. Each page will have its own text, images and other elements. All web pages and elements are then placed in a folder and stored on your web host server.

Each web page is written in codes and these codes describe the layout, format and content on the page. The most common coding language used to create web page is HTML.

Steps for connecting to Website are explained below:

Scanned with OKEN Scanner

1. Type the URL for a Web site say www.yahoo.com into your Web browser.
2. Your browser attempts to make a connection to the Web server.
3. The Web server receives the request.
4. The Web site home page is downloaded from the Web server to your PC.
5. The Web page is displayed by your Web browser and the connection between the server and your browser is closed.

Domain name

A domain name is the address that you type into your web browser address bar to get to a website. An example of a domain name is **www.Google.com**. A domain name is unique to a website. In other words, no two websites can have the same domain name.

Web server

Web servers are computers whose job is to respond to a browser's request for a web page and deliver it through the Internet. Pages hosted on a web server can be displayed to anyone all over the world. It is like a hard drive that stores your website files and images. In order to host your page on a web server, you need to pay a hosting charge.

First open your web browser and type in a domain name, your browser will display the web pages of the domain name you have entered. How your web browser know what information to display?

Each website will have a website address, or a domain name and each domain name is tied to the IP address of the web server it resides on. When you type in a domain name in your web browser, your web browser is actually conducting a series of inquiries that include looking up the IP address of the domain name, locating the web server that hosts the web pages of the domain name, submitting a request to that server for a copy of the web page(s), receiving the web page(s) from the server and finally translating the codes on the web page to present the information on your screen.

Types of Website

A website can be of two types:

Static Website

Static website is the basic type of website that is easy to create. Static website has webpages with constant or fixed content which cannot be changed by the user who visits the site. Static webpages are built using the HTML code and the content is static, it does not change automatically until the webmaster makes changes. These websites are the fundamental or primary websites built using the HTML coding and connected to a web server.

A static website looks and feels very basic and fundamental. These websites are cost efficient to develop and host which is a perfect fit for small companies. The changes or updates are done by the designer and the changes are replaced with the original site on the Internet. (*See Figure 1.3*)

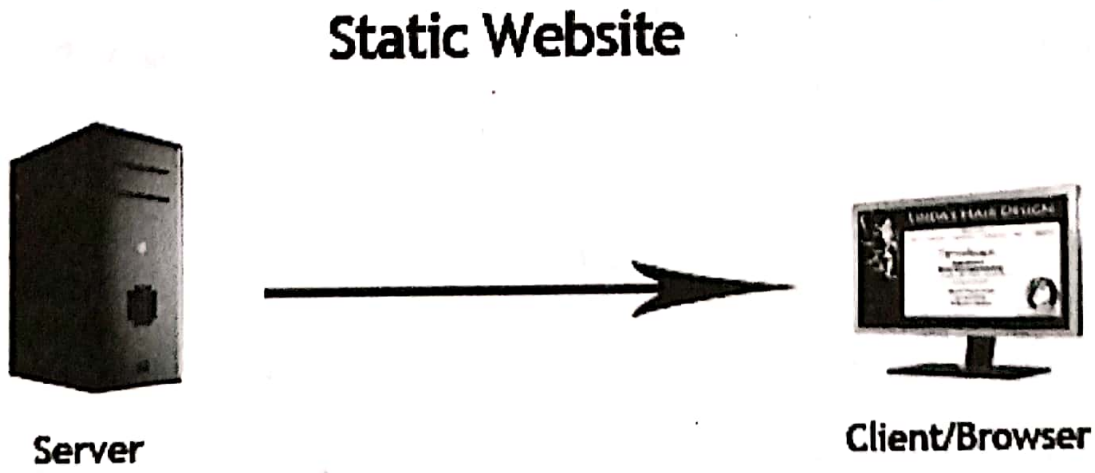


Figure 1.3: Static Website

Scanned with OKEN Scanner

Advantages

Some of the advantages of a static website are:

1. **Time Saving:** The big advantage of a static website is that it is quick to develop website much faster than a dynamic by web developer.
2. **Cost Effective:** Cheaper to develop.
3. **Inexpensive hosting:** It can get dedicated servers at a cheaper price
4. **Easy Indexing:** Search engines like **Google, Bing** etc. can easily index a static website as they are just a series of coded HTML or CSS files
5. **Fast Transferring:** Static websites do not have complex structures like a dynamic website and can be easily and quickly transferred from server to client without much processing time.

Disadvantages

Some of the disadvantages of a static website are:

1. **Difficult to Change:** The big disadvantage of a static website is that its content cannot be changed easily. Any novice cannot update the content. It requires the expertise of a web developer to update, add or change any content on a static website. All the HTML files would need to individually change even for a slight change made to the website.
2. **Not Good for the Long Run:** A static website is not apt for the long run as any business would need to make a number of updates to be in sync with the latest trends.
3. **Limited Functionality:** It does not offer all the functionalities that a dynamic website can. One can add text, images, videos, and hyperlinks in the content but there are no other special functions that a static website can perform.

Dynamic Website

Dynamic website is a collection of dynamic web pages whose content changes dynamically. It accesses content from a database or Content Management system (CMS). Therefore, when you alter or update the content of the database, the content of the website is also altered or updated. It is built and developed using server-side coding language such as **PHP, ASP, JSP, Ruby** etc. In a dynamic website the content is taken from other databases as and when required by the users. (*See Figure 1.4*)

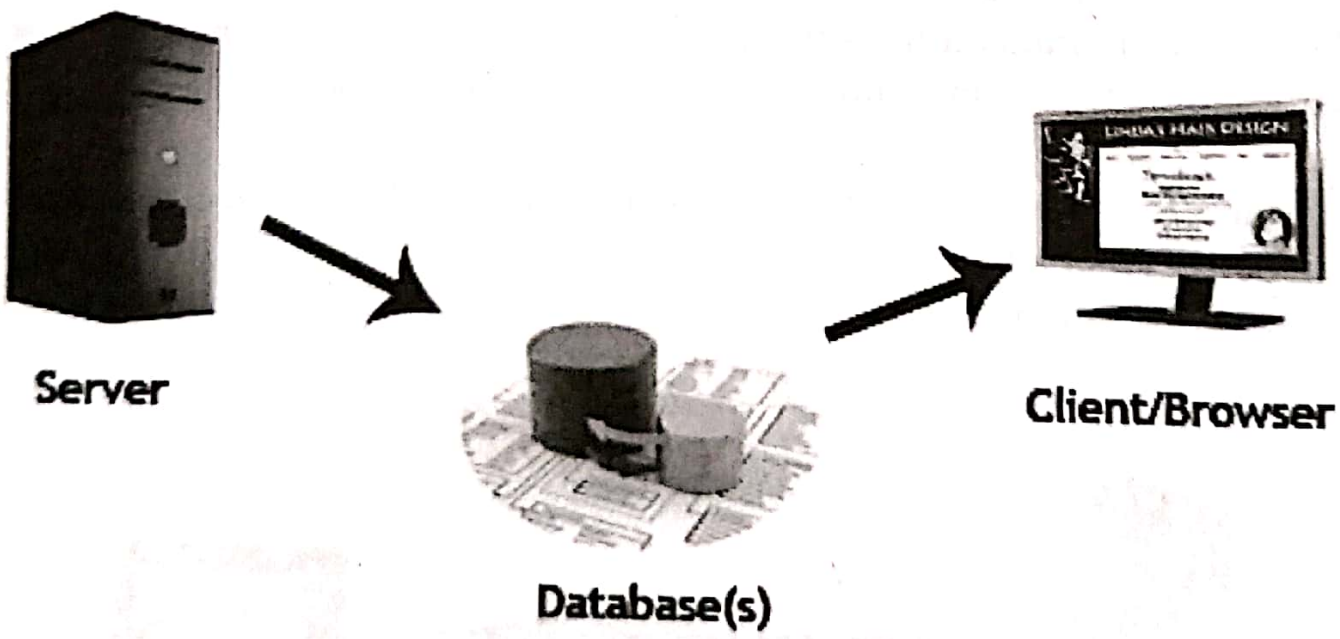


Figure 1.4: Dynamic Website

It uses client-side scripting or server-side scripting. Client side scripting generates content at the client computer based on user input. The web browser downloads the web page from the server and processes the

Scanned with OKEN Scanner

code within the page to render information to the user. In server side scripting, the software runs on the server and processing is completed in the server then plain pages are sent to the user.

Advantages of Dynamic Websites

1. **Easy to update:** The biggest advantage of a dynamic website is that it can be easily updated.
2. **Interactive:** It interact with the users and changes according to their behavior.
3. **Smooth Navigation:** It offers a smoother navigation and lets the user jump from one page to the other without any problem.

Disadvantages of Dynamic website:

1. **Higher cost:** It can cost big cash in their development and even the hosting cost is high.
2. **Slow Processing:** Having a number of functions to perform with complex technology, the dynamic websites becomes slower to process and load.

Difference between Static website and Dynamic website

Static Website	Dynamic Website
Prebuilt content is same every time the page is loaded.	Content is generated quickly and changes regularly.
It uses the HTML code for developing a website.	It uses the server side languages such as PHP, Servlet, JSP, and ASP.net etc. for developing website.
It sends exactly the same response for every request.	It may generate different HTML for each of the request.
Flexibility is the main advantage of static website	Content Management System is the main advantage of dynamic website.
Information are change rarely.	Information are change frequently.
In static website, database is not used	In dynamic website, database is used.

Web Pages

World Wide Web information is presented on Web pages, which you download to your computer using a Web browser such as Internet Explorer. Each Web page can combine text with images, sound, music and video.

A web page is an electronic document written in computer language called **HTML (Hypertext Markup Language)**. Each web page has a unique address, called a **URL (Uniform Resource Locator)** that identifies its location on the network.

A website has one or more related web pages. Web pages on a site are linked together through a system of hyperlinks to jump between them by clicking on a link. On the web, you navigate through pages of information according to your interests.

The first page of the web site is called the **Home page**. The home page provides an overview of what you will find at the website. A site can have one page or many pages. If there is a lot of information, the home page may be the only page. But usually, you will find at least a few other pages.

1.5 Front End and Back End Application

Frontend and backend are the two most popular terms used in web development. These are very crucial for web development but are quite different from each other.

1.5.1 Front End Development:

Front end development manages everything that users visually see first in their browser or application. It is a part of website that users interacts with directly is termed as **front end**. It is also referred to as the `**client**

Scanned with OKEN Scanner

side' of the application. It is built using a combination of technologies such as **Hypertext Markup Language (HTML)**, **JavaScript** and **Cascading Style sheets (CSS)**. Front-end Developers design and construct users experience elements on the web page including buttons, menus, pages, links, graphics and more.

- **HTML:** Hypertext Markup Language is the core of a website, providing the overall design and functionality. It is the combination of Hypertext of Hypertext and Markup language. Hypertext defines the link between the web pages. The markup language is used to define the text documentation within tag, which defines the structure of web pages.
- **CSS:** Cascading Style sheets referred to, as CSS is simply designed language intended to simply the proves of making web pages presentable. It allow you to apply styles to web pages. More importantly, CSS enables you to do this independent of the HTML that makes up each web page.
- **JavaScript:** This event-based language is useful for creating dynamic elements on static HTML web pages.
- It allows developers to access elements separate from the main HTML page, as well as respond to server-side events.

Front End Frameworks and Libraries

Frond-end frameworks such as **Augular, Ember, Backbone** and **React** are also popular.

- **Angular JS:** It is a JavaScript opens source front-end framework that is mainly used to develop single page web applications. It is continuously growing and expanding framework which provides better ways for developing web applications. It changes the static HTML to dynamic HTML. It is an open source project, which can be freely used and changed by anyone.
- **React JS:** It is a declarative, efficient, and flexible JavaScript library for building user interfaces. It is an open-source component based front end library responsible for the view layer of the application and it maintained by **Facebook. Bootstrap** is a free and open-source for creating responsive websites and web application. It is the most popular **HTML, CSS** and **JavaScript framework** for developing mobile websites.
- **JQyery:** jQuery is an open source JavaScript library that simplifies the interactions between **HTML/CSS** document.

1.5.2 Backend Development

Back end development refers to the server side of an application and everything that communicates between the database and the browser. It stores and arranges data, and makes everything on the client-side of the website and it is the part of the website, which you cannot see and interact. Back-end developers are responsible for creating the logic on which an app or a website works. It should have certain skills in-depth knowledge of the back-end programming language or framework, accessibility and security compliance, a fair understanding of front-end technologies like HTML, CSS and the ability to manage a hosting.

- **Back end Languages:** The back end portion is built by using some languages which are discussed below:
- **PHP:** It is a **server-side scripting language** designed specifically for web development. PHP code executed on the server side so, it is called server-side scripting language.
- **C++:** It is a general purpose programming language and widely used for competitive programming. It is also used as **backend language.**
- **Java:** It is one of the most popular and widely used programming language and platform.
- **Python:** It is programming language that lets you work quickly and integrate systems more efficiently.
- **JavaScript:** It can be used as both (front end and back end) programming languages.

Scanned with OKEN Scanner

These languages are used to create dynamic sites which are different from static sites in that these types of websites store database information. Content on the site is constantly changing and updating. *Examples* of dynamic sites include **Facebook**, **Twitter** and **Google Maps**.

1.6 Client and Server Scripting Languages

Scripting languages, which can be embedded within HTML, commonly are used to add functionality to a Web page, such as different menu styles or graphic displays. These types of languages are client-side scripting languages; affecting the data, that end user seems in a browser window. Other scripting languages are server-side scripting languages that manipulate the data, usually in a database on the server.

1.6.1 Client-Side Scripting

It is performed to generate a code that can run on the client end (browser) without needing the server side processing. These types of scripts are placed inside an HTML document. The client-side scripting can be used to examine the user's form for the errors before submitting it and for changing the content according to the user. The web requires three elements for its functioning which are, client, database and server. *For example*, when a user makes a request via browser for a webpage to the server, it just sent the HTML and CSS, and the browser interprets and renders the web content in the client end.

1.6.2 Server-side Scripting

It is a technique of programming for producing the code which can run software on the server side. Any scripting or programming that can run on the web server is known as server-side scripting. The server-side scripting constructs a communication link between a server and client (user). Earlier the server side scripting is implemented by the *CGI scripts*. The CGI was created and execute the scripts from programming languages such as **C++** or **Perl** on the websites.

Difference between Client-side scripting and server-side scripting

For web pages they are instructions either to the web server (server-side scripting) or to the Web browser(client-side scripting). Let us look how they differ from each other. (*See Figure 1.5*)

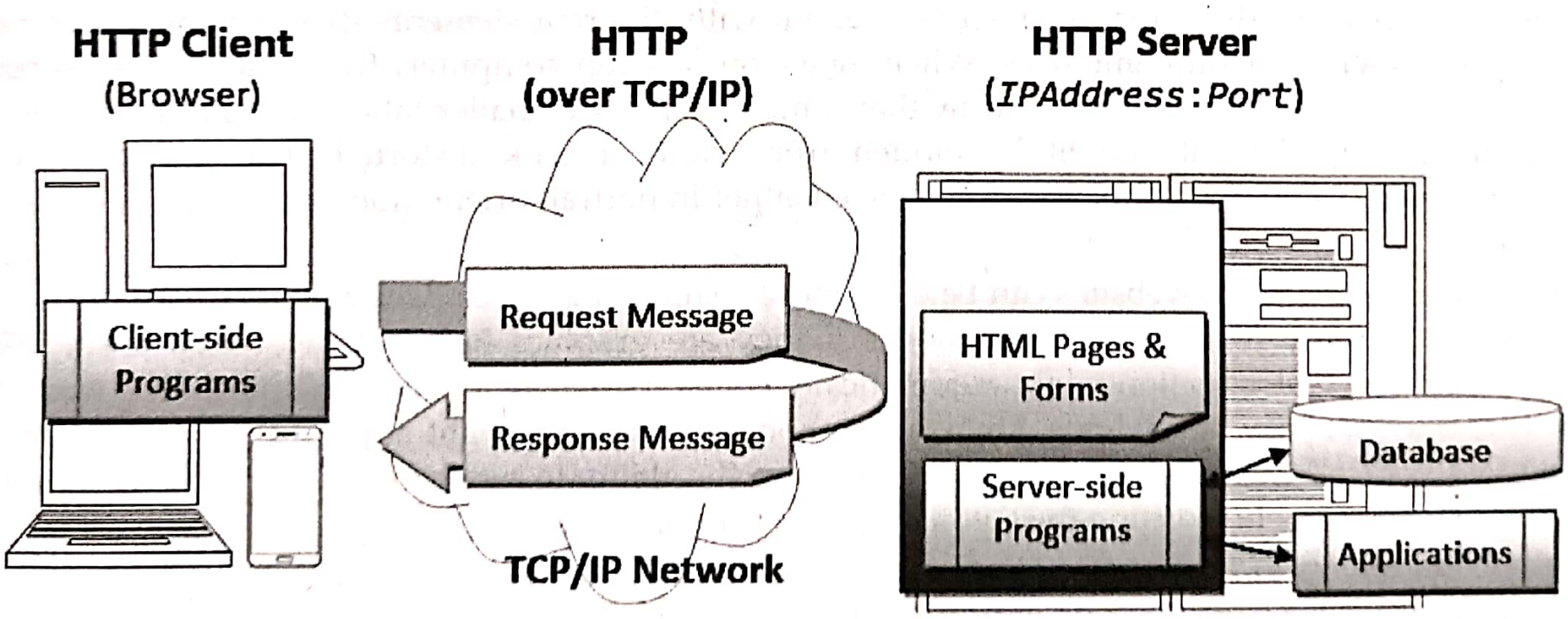


Figure 1.5: Client-side and Server-side Scripting

Server Side Scripting

a. It is used to create dynamic pages based a number of conditions when the users browser makes a request to the server.
b. The Web server executes the server side scripting that produces the page to be sent to the browser.
c. Server executes server-side scripts to send out a page but it does not execute client-side scripts.

Scanned with OKEN Scanner

d. It is used to connect to the databases that reside on the web server.
e. It can access the file system residing at the web server.
f. It cannot be blocked by the user.
g. Relatively secure.
h. Examples of Server side scripting languages are: PHP, JSP, ASP, ASP.Net, Ruby, Perl etc.

Client-side scripting

a. It is used when the user's browser already has all the code and the page is altered on the basis of the users input.
b. The Web browser executes the client side scripting that resides at the user's computer.
c. The browser receives the page sent by the server and executes the client-side scripts.
d. It cannot be used to connect to the databases on the web server.
e. It cannot access the file system that resides at the web server.
f. It is possible to be blocked by the user.
g. It is insecure.
h. Examples of Client side scripting languages are: JavaScript, CSS, HTML, etc.

1.7 Responsive Web Designing

Responsive web design is an approach whereby a designer creates a web page that responds to or resizes itself depending on the type of device. That could be an oversized desktop computer monitor, a laptop or devices with small screens such as smartphones and tablets.

Responsive Web design has become an essential tool for anyone with a digital presence. With the growth of smartphones, tablets and other mobile computing devices, more people are using small-screens to view web pages.

The purpose of responsive design is to have one site, but with different elements that respond differently when viewed on devices of different sizes. When views on desktop computer, for instance, the website might show three columns. But when you view that same layout on a smaller tablet, it might force you to scroll horizontally. Or elements might be hidden from view or look distorted. The impact is also complicated by the fact that many tablets can be viewed either in portrait orientation or turned sideways for landscape view.

On a small smartphone screen, websites can be even more challenging to see. Large images may break the layout. Sites can be slow to load on smartphones if they are graphics heavy. However, if a site uses responsive design, the tablet version might automatically adjust to display just two columns. That way, the content is readable and easy to navigate. On a smartphone, the content might appear as a single column, perhaps stacked vertically. Or possibly the user would have the ability to swipe over to view other columns. Images will resize instead of distorting the layout or getting cut off.

Three Main Components of Responsive Design

Responsive website design consists of the following three main components:

- **Flexible layouts:** Using a flexible grid to create the website layout that will dynamically resize to any width.
- **Media queries:** An extension to media types when targeting and including styles. It allow designers to specify different styles for specific browser.
- **Flexible media:** Makes media (images, video and other formats) scalable, by changing the size of the media as the size of the viewport changes.

Scanned with OKEN Scanner

Responsive Web Design vs. Traditional Web Design

An example of a non-responsive web design page is one that reads well on desktop browsers but has very small, unreadable text on smartphones, often due to having too many columns or images that are too large to fit within a smartphone's limited viewport display width.

How Does Responsive Web Design Work?

Responsive sites use liquid grids. All page elements are sized by proportion, rather than pixels. *For example,* if you have three columns, you would not say exactly how wide each should be in relation to the other columns. Column 1 should take up half the page, column 2 should take up 30%, and column 3 should take up 20%. Media such as images is also resized relatively. That way an image can stay within its column or relative design element.

The goal of responsive design is to avoid the unnecessary resizing, scrolling, zooming, or panning that occurs with sites that have not been optimized for different devices. It is often very difficult to navigate these sites, and it may even cost you potential customers who become frustrates to figure out. It also need to design a dedicated mobile website for smartphone users. Instead of designing multiple websites for different screen sizes, you can design just one website that scales up or down automatically to match the device it is being viewed on.

The benefits of choosing responsive design for your website include:

- Cost effectiveness
- Flexibility
- Improved user experience
- Search engine optimization gains
- Ease of management

Cost effectiveness: Maintaining separate sites for your mobile and non-mobile audiences can get expensive. By using responsive design, you can save money by eliminating the cost of paying for a mobile site. You will need to invest in a single site design to appeal to all visitors and all devices.

Flexibility: When you have a website with responsive design, you can make changes quickly and easily. You do not need to worry about making changes on websites.

Improved use experience: User experience is crucial to website owners. You want people to like your site, and you want it to be easy to use to convince them to come back. If someone visits your website on a mobile device, and it takes forever to load or your pictures do not have the proper resolution, it can make your company appear unprofessional. No one want to do business with a place that is unprofessional. But responsive design, which offers a much better user experience, can help convince people to give your company a chance. Because zooming and scrolling will be eliminated, content can be viewed quicker, and the overall impression that visitors have will be much more positive.

Search engine optimization: Search engine optimization is a strategy used by many companies to help boost themselves in Google's search page ranking. Responsive design can help with SEO because Google, gives preference to websites that are mobile-friendly and also give you a big boost in search engine results.

Ease of Management: Most businesses, especially smaller ones, do not have a lot of time to update or refresh the way their website looks. But rather than having to hire a designer to handle every aspect of your website, responsive design allows you to make the changes yourself, quickly and easily.

1.8 Downloading Free Editors

Microsoft introduced Multi-Tool Notepad, a mouse-based text editor written by **Richard Brodie**, in May **1983** at the Spring **COMDEX** computer expo in Atlanta.

Scanned with OKEN Scanner

Notepad is a simple text editor for Microsoft Windows and a basic text-editing program which enables computer users to create documents. It was first released as a mouse-based **MS-DOS** program in **1983**, and has been included in all versions of Windows since **Windows 1.0** in **1985**. Notepad allows you create, open and read plaintext files. If the file contains special formatting or is not a plain text file, it will not be able to be read in Notepad.

Notepad Features

Notepad in Windows 10 have the following features:

- Shortcut to delete previously typed word
- Status bar
- Wrap-Around
- Text zooming
- Search text with Bing in Notepad

1. **Shortcut to delete previously typed word:** Firstly, you can use a simple shortcut **Ctrl + Backspace** to delete a previous word you typed or entered.
2. **Status bar:** It is visible at the bottom of the Notepad screen and displays the ending line of the text file in use. Alongside it, you can find information related to column and line number with word-wrapping and current zoom level. The Status bar itself is enabled by default.
3. **Wrap-Around:** Earlier, Notepad had certain limitations. For instance, you could only search or replace text using a single word. However, now you can find a new wraparound option visible under **Find** tool to search a fragment of a text. Notepad will save previously typed values and checkboxes and automatically apply them when you re-open the **Find** dialog box.
4. **Text zooming:** It has also added options to quickly zoom text in Notepad. So, when you attempt to change the text layout, you can notice the zoom percentage on its status bar.
5. **Search with Bing in Notepad:** The new Notepad enables Windows 10 users quickly find the meaning or spelling of a word within the app.

1.8.1 Notepad++

Notepad++ is a text editor and source code editor for use with MS Windows. Its supports tabbed editing which allows working with multiple open files in a single window. Running in the MS Windows environment, its use is governed by **GPL License**. Based on the powerful editing component **Scintilla**. The Notepad++ is one of the best editing desktop application written in C++ and officially uses the pure **Win32 API & STL** which ultimately make sure that the users experiences the best execution speed and ultimate small size of the users program. This all has been done by optimizing daily routines of the application without compromising the user-friendly nature of the application. It is just like reducing pollution of the environment. It uses fewer resources like low of CPU usage, reduction of CPU power consumption, and ultimately contributing better towards the greener environment.

Undoubtedly, it is one of the best and powerful tools, equipped with the best features which can easily replace the Notepad. It also features the 27 programming languages, support searching of regular expressions, folding, syntax highlighting, edits, views of code, etc.

Notepad++ also support a number of languages. You can easily translate the user interface of the application into your native language. You can also add Translation Binary in the Translation Binary page in order to transform the user interface of Notepad++ in your native language.

Features of Notepad++ version

Let us have a look at the features of Notepad++.

a. Notepad++ supports syntax highlighting and syntax folding.

Scanned with OKEN Scanner

b. The user can also customize the syntax highlighting feature.
c. It also supports auto-completion.
d. You can easily open multiple documents in different tabs.
e. You can easily see the multiple views of multiple sites.
f. The latest version of the application also comes up with the Drag and Drop functional.
g. It also equipped with the file status auto detection.
h. You can easily zoom in and zoom out the document.
i. You can also do macro recordings & playback in the Notepad++.

Notepad++ is a very well-known **desktop-based text editor**. The authorities are regularly working on fixing bugs and introducing the best features to increase the user experience. You can easily download the version of the Notepad++ from their website.

1. Open the download page of the official website in your browser.
2. Download the latest Notepad++ version to your desktop i.e. https://notepad-plus-plus.org/.
3. Open the folder with the downloaded package on your computer. Click the **.exe** file to run the Installer.
4. Follow all installation steps:
   a. Select installation language that you want to use as shown in *Figure 1.6*. After clicking **OK**, Welcome to Notepad++ version Setup page appears. Click **Next**.

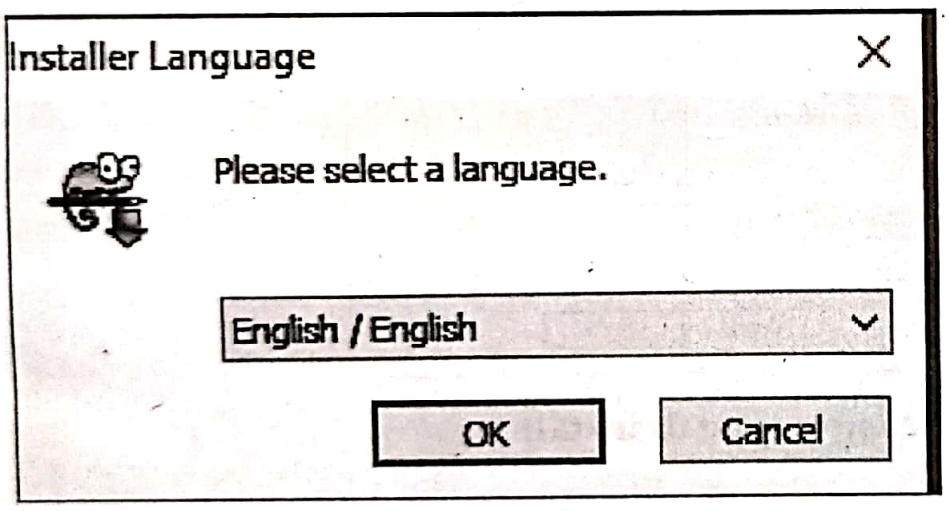


Figure 1.6: Select language

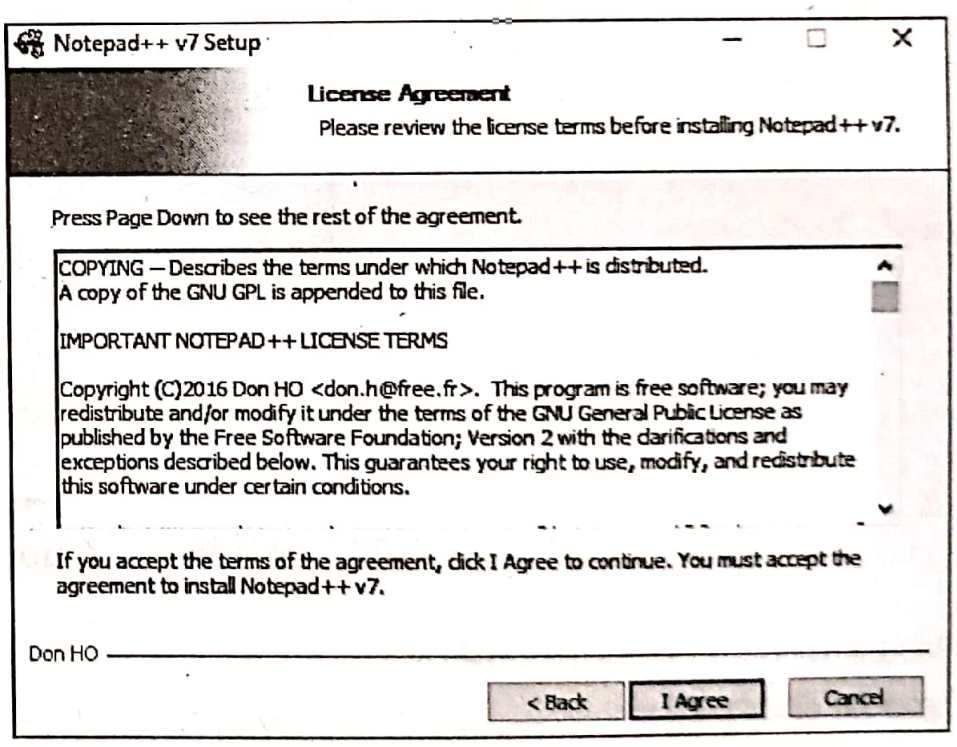


Figure 1.7: License Agreement

b. Agree to the terms of the License Agreement as shown in *Figure 1.7*. Click **Next**.
c. Click **Browser...** button to choose the Location of installation on your computer as shown in *Figure 1.8*. Click **Next**.
d. Select the type of installation. To recommend to use a Custom one as shown in *Figure 1.9*. Click **Next**.
e. Choose the components for your Notepad++ installation and click install to confirm as shown in *Figure 1.10*.
f. Please wait when the system finish installation as shown in *Figure 1.11*. Now, you are ready to use Notepad++ software for editing codes.

Scanned with OKEN Scanner

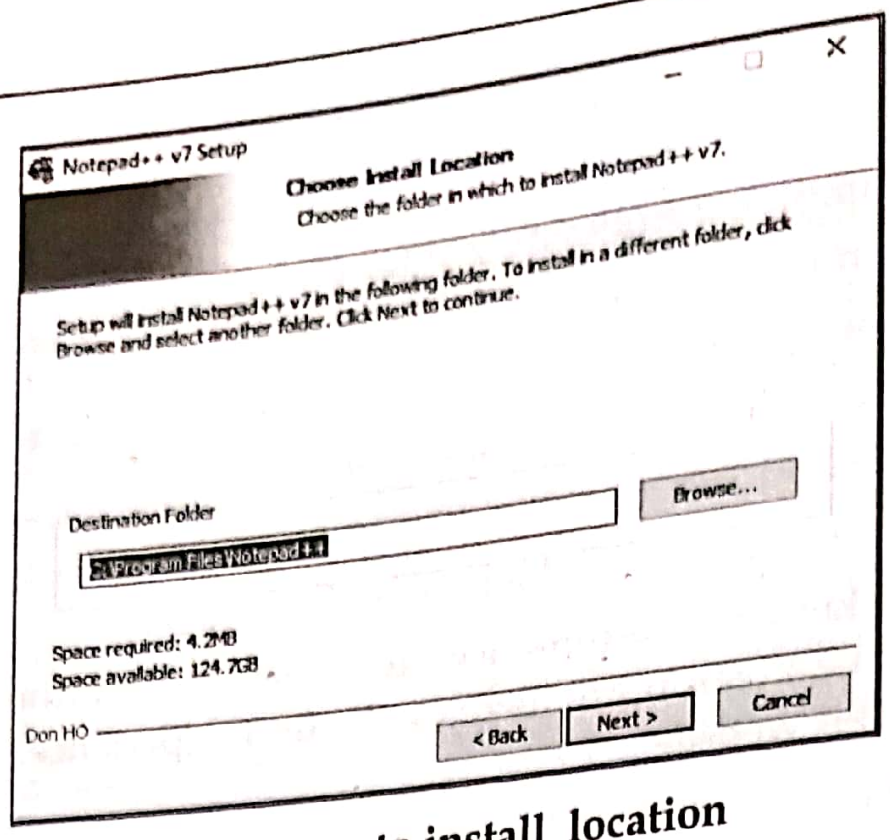


Figure 1.8: Choose to install location

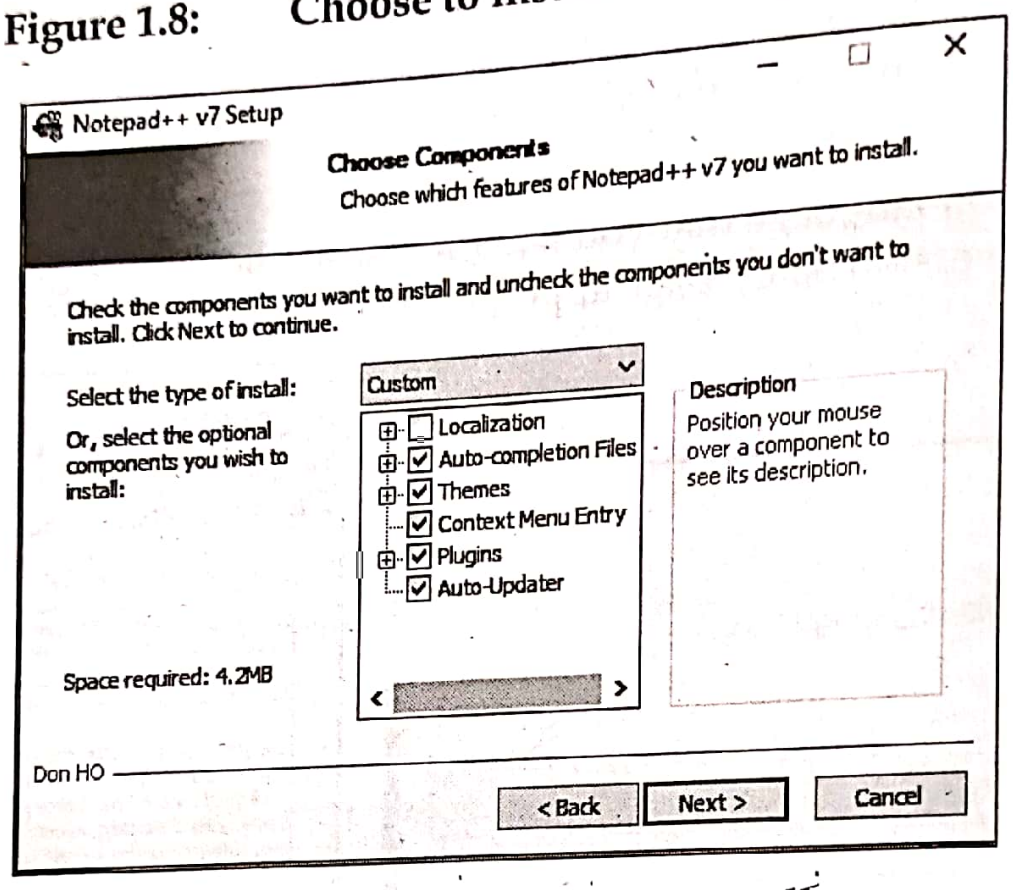


Figure 1.9: Choose customtype to install

Pros and Cons of Notepad++

Pros of Notepad++

1. Being built for the Windows platform, it can also run on Linux, Unix, Mac OS X.
2. It is a lightweight application and launches quickly.
3. It supports external plugins including macros.
4. The interface supports tabbed editing.
5. Add bookmarks for future concerns.
6. Drag and Drop option for beginners.
7. Option to find and replace text
8. Code and text folding
9. Smart syntax highlighting
10. Auto indentation and auto complete
11. FTP support.

Scanned with OKEN Scanner

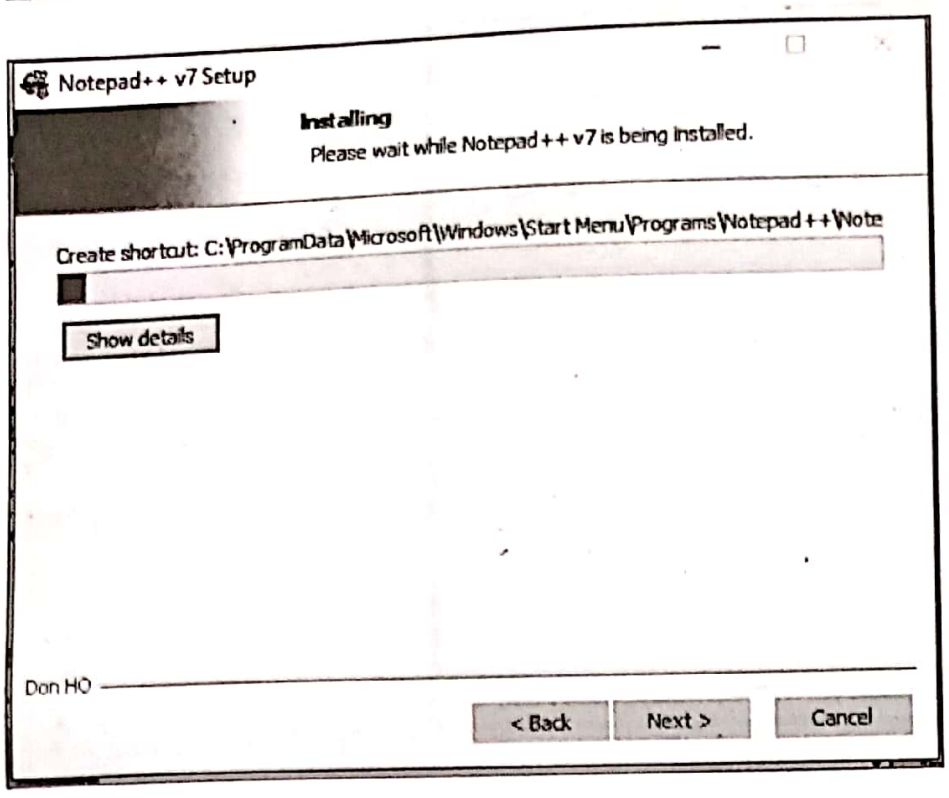


Figure 1.10: Installing Notepad++ is being installed

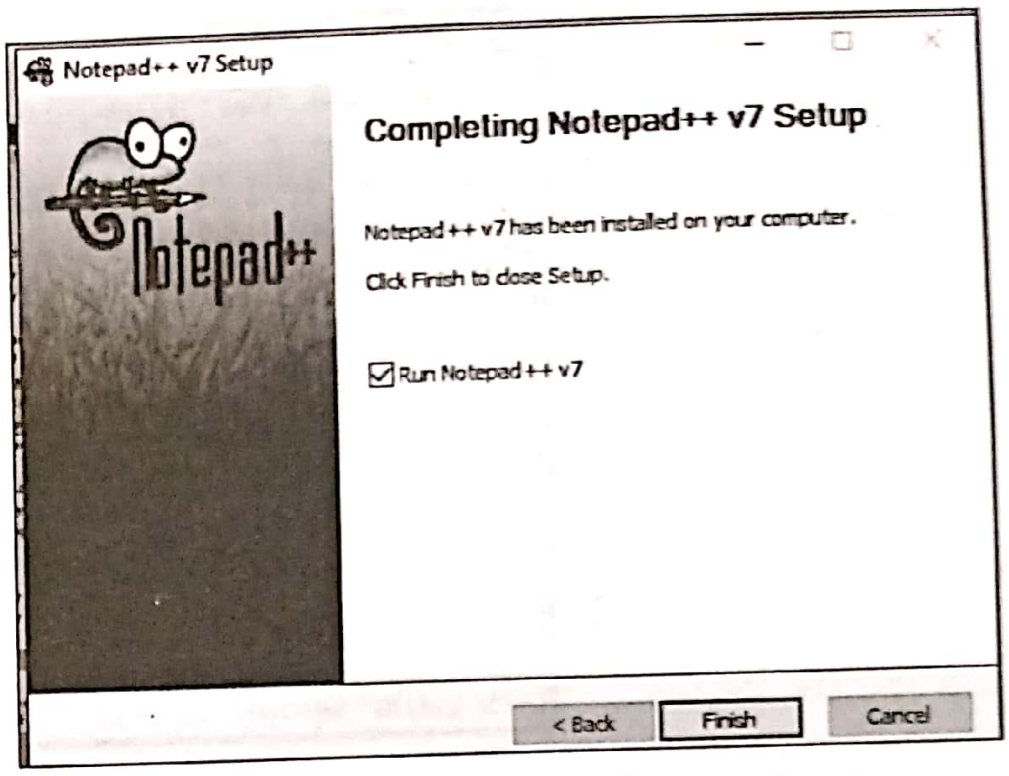


Figure 1.11: Completing Notepad++ Setup

Cons of Notepad++

1. Remote file editing does not support HTTP, SSH or WebDAV
2. Does not support large files.

Creating a simple C++ Program

1. Click the **Language** tab, a drop-down menu appears. Highlight **C**. A sub menu appears select **C++.**
2. Add a title to your program. Type in //followed by your program's title *for example*, "**My First program**", then press **Enter** as shown in *Figure 1.12*.

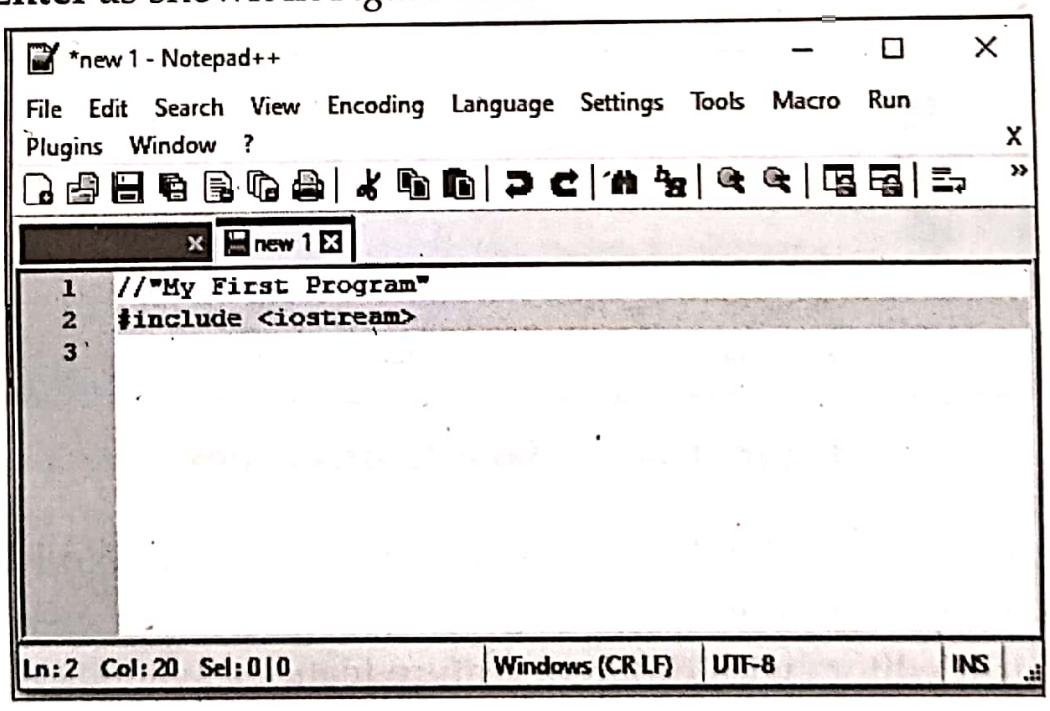


Figure 1.12: Type a prgram in Notepad++

3. Type **#include <iostream>** into Notepad++, the press Enter. This command instructs C++ to run the following lines of code as a program as shown in *Figure 1.12*.
4. Type **int main()** into Notepad++, to declare the program function then press **Enter.**
5. Type { to add an open bracket, then press Enter. Your program main code will go between this open bracket and a closed bracket }.
6. Type **std::count << "Hello World!";** to enter execution code and press **Enter** as shown in *Figure 1.13*.

Scanned with OKEN Scanner

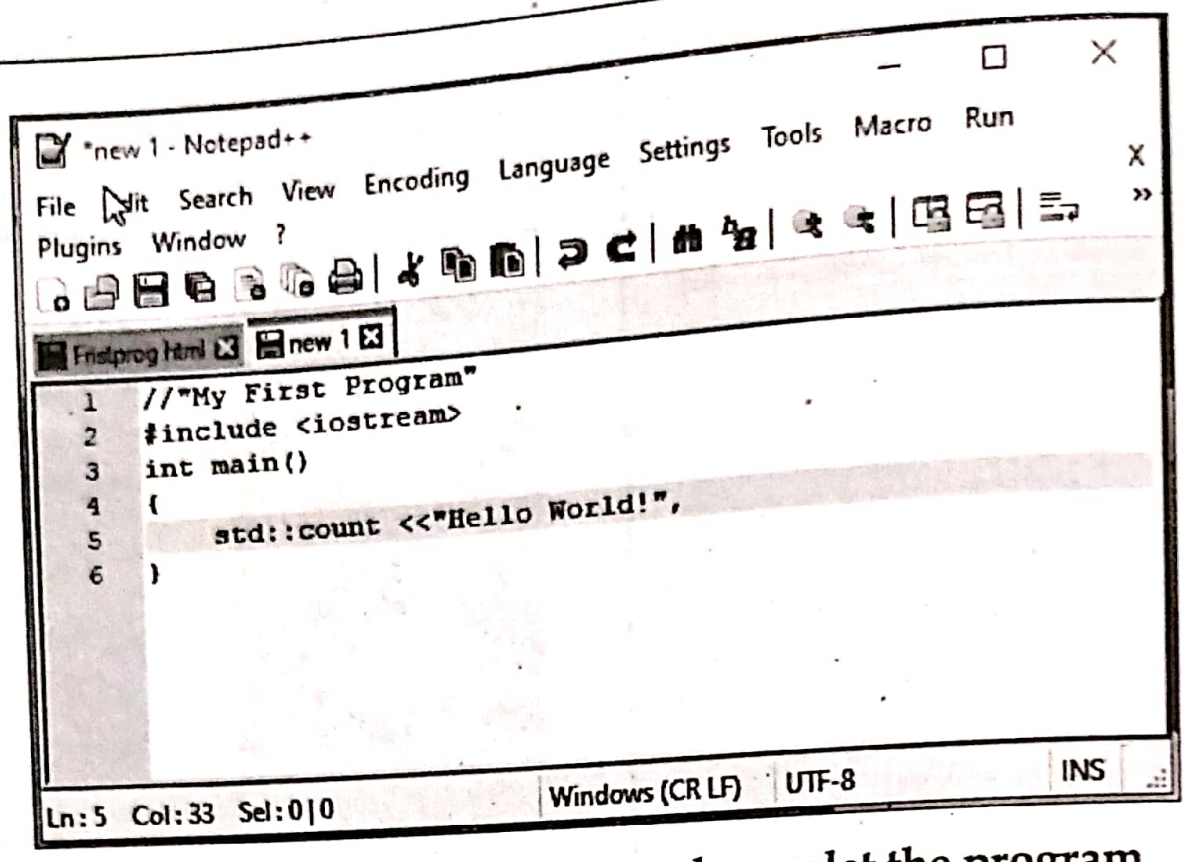


Figure 1.13: Enter function and complet the program

7. Click **File** menu and choose **Save As...** The **Save As** dialog box appears as shown in *Figure 1.14*. Enter name for the program, and then select a save location and click **Save.**

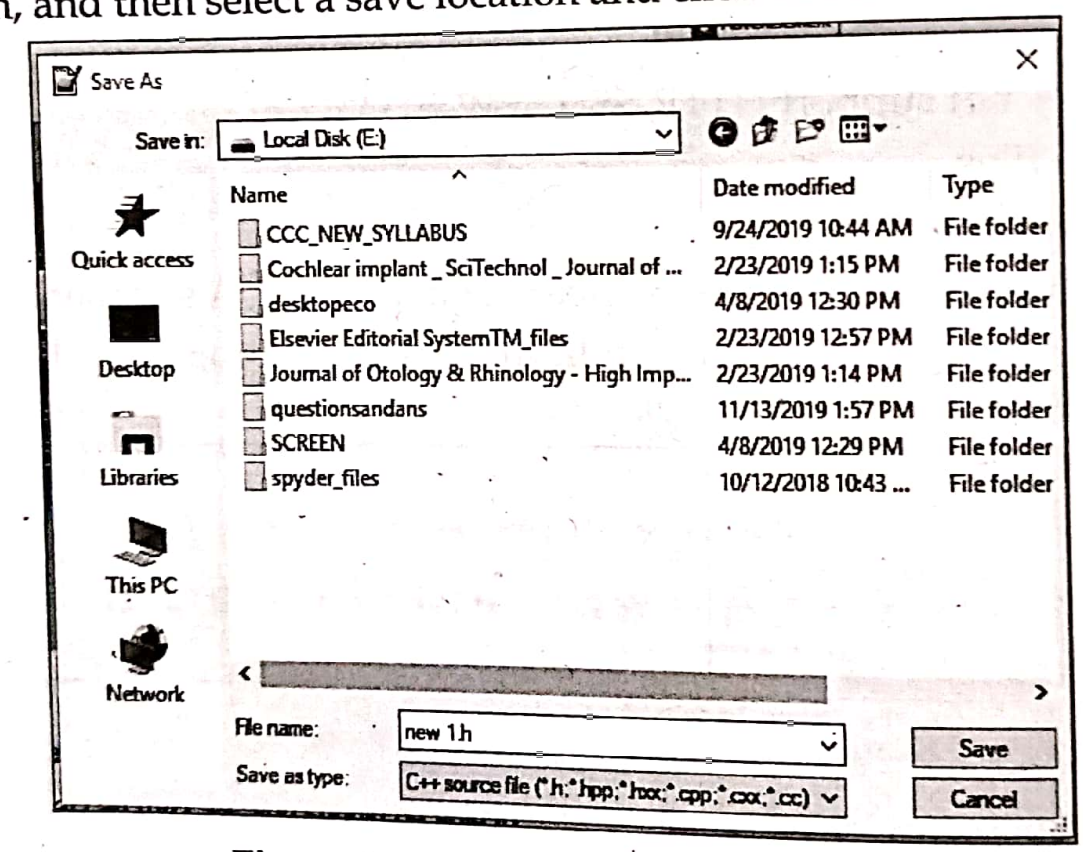


Figure 1.14: Save As dialog box

8. After saving that can run C++, you should be able to open your "Hello World!" program in it.

Different Editors available for writing Code

There are number of good text editors with different feature features available, they all have the same goal to make writing code easier for you.

1. **Atom:** It is useful code editor for programmers due to its simple interface compared to the other editors The features of Atom are:
   - Package manger integrated for Plugins support.
   - The feature of smart auto completion.
   - Supports Command Palette.
   - Allow cross-platform editing.

Scanned with OKEN Scanner

2. **TextWrangler:** TextWrangler is a powerful and feature rich text editor. This powerful and feature rich text editor is a **freeware.**

   The features of TextWrangler are:

   - It comes with a very intuitive user interface that makes it super easy to browser files and work on projects.
   - It has advances search and replace tools, and it can directly edit and save files to FTP, SFTP serves.
   - It offers syntax highlighting for many programing languages, supports UTF-8, and allows you to convert character encoding of text documents.

3. **Bluefish:** It is a cross-platform editor is a speedy tool, which can handle dozens of files simultaneously. The tool allows developers to conduct remote editing. This code editor tool offers many options to programmers and web developers, to write websites, scripts, and programming code.

   The Features of Bluefish are:

   - Load hundreds of files within seconds.
   - Auto-recovery of changes in modified documents after a crash, or shutdown.
   - Project support feature helps you to work efficiently on multiple projects.
   - Unlimited undo/redo functionality.

4. **Brackets:** It is a lightweight tool developed by Adobe. It is an open source text editor which is free to download. It allows you to a toggle between your source code and the browser view.

   The Features of Brackets are:

   - Quick Edit UIfeature puts context-specific code and tools inline.
   - It offers live preview, preprocessor support, and inline editors.
   - It comes with the inbuilt extension manager for fast and effective extension management.

5. **Light Table:** It is an IDE and text editor for software development. The tool offers fast feedback and allowing instant execution, debugging and access to documentation.

   The features Light Table are:

   - Inline evaluation.
   - Light Table is a lightweight, clean and sleek interface.
   - Powerful editing and plugin manager.

6. **Visual Studio Code:** It is an open source editor software. IT offers built-in support for Typescript, JavaScript, and Node.js. It is autocompleted with IntelliSense features provides smart completions based on variable types, essential modules, and function definitions.

   The Features of Visual Studio Code are:

   - Easy working with Git and other SCM providers.
   - Code refactoring & debugging.
   - Easily extensible and customizable.
   - It provides an enriching programming experience to developers without letting them download the massive visual studio editions file, which is approximately 3GB in size.

7. UltraEdit: It is a powerful HTML and Code editor available for **Mac, Windows**, and **Linux**.

   The Features of UltraEdit are:

   - It comes with a built-in file comparison utility, auto-completion, advanced layout, multi-tab, multi-pane editors, and syntax highlighting for the most popular programming languages.

Scanned with OKEN Scanner

1.8.2 Sublime Text Editors

Sublime Text is a cross-platform source code editor with a Python application programming interface. It natively supports many programming languages and markup languages, and functions can be added by users with plugins, typically community-built and under free-software licenses.

Various features that are supported by Sublime are as follow:

- Syntax Highlight
- Auto indentation
- File Type Recognition
- Macros
- Plug-in and Packages

Sublime Text editor is used as an Integrated Development Editor like Visual Studio code and NetBenas. The current version of Sublime Text editor is 3.0 and is compatible with various operating systems like Windows, Linux and MacOS.

Why Use Sublime Text?

When you use suitable text editor, you can enjoy its rich beneficial features. It offers its users the following benefits:

- Ability to solve linker errors.
- Keeping track of all files and folders to work.
- Problem solving capabilities.
- Keeping color combination for syntax combination.

Downloading Sublime Text Editor

Sublime Text Editor is supported by major operating system:

- Window
- Linux and its distributions
- OS X

You can download Sublime Text from its official website **www.sublimetext.com.**

Installation on Windows

You will have to follow the steps to install sublime Text on Windows:

1. Download the **.exe** package from the official website i.e. **https://www.sublimetext.com/3** as shown in *Figure 1.15.*

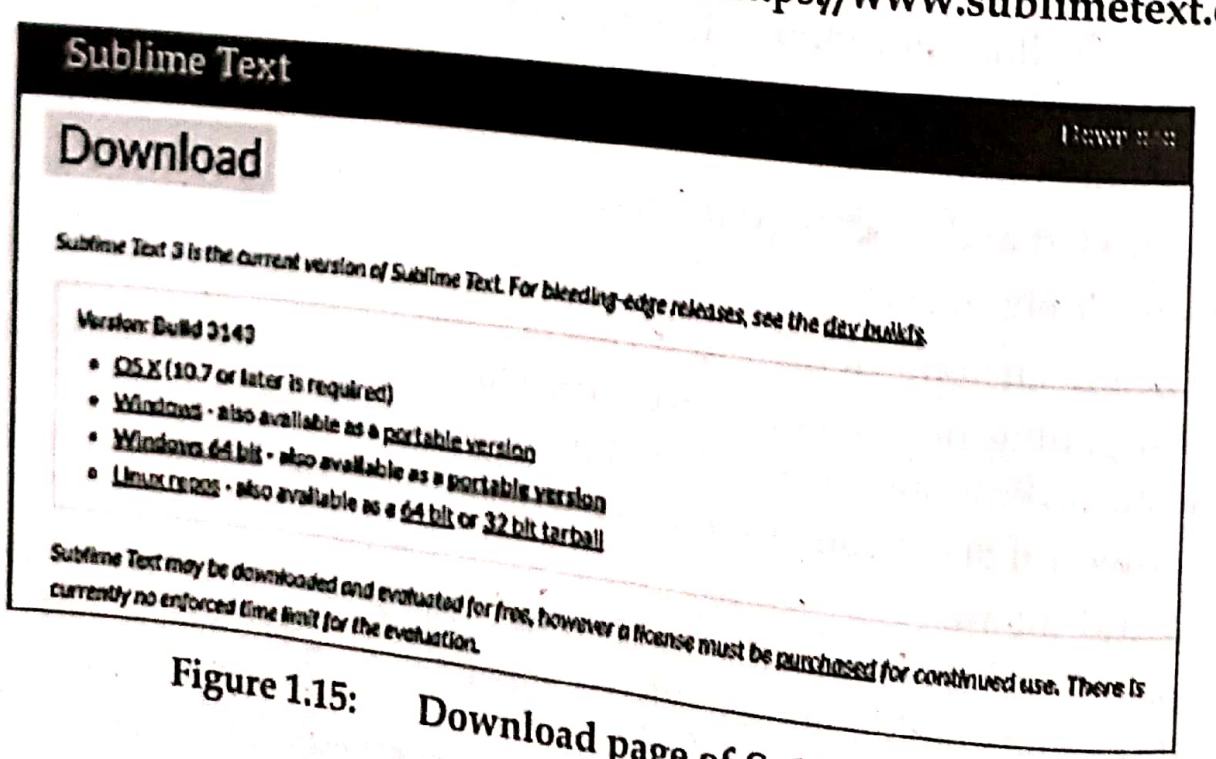


Figure 1.15: Download page of Sublime text

2. Now, run the executable file. This defines the environment variables. When you run the executable file, you can observe the following window on your screen. Click **Next**. (See *Figure 1.16*)

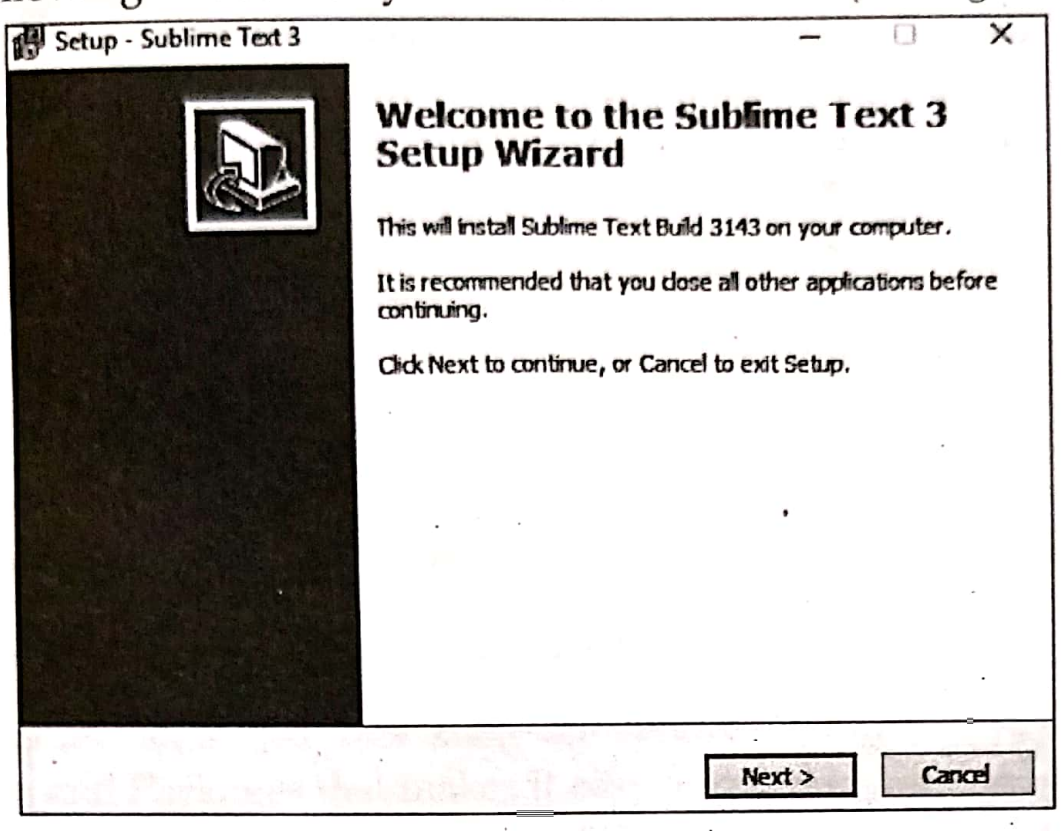


Figure 1.16: Welcome of Sublime Text setup wizard page

3. Now, choose a destination location to install **Sublime Text3** and click **Next.** (See *Figure 1.17*)
4. Verify the destination folder and click **Install**. (See *Figure 1.18*)
5. Now, click **Finish** to complete the installation. (See *Figure 1.19*)
6. Now, upon a successful installation, your editor will appear as shown in *Figure 1.20*.

Pros and Cons of Sublime Text Editor

The Pros of Sublime text editor are as follow:

1. Multi-Line select and editing.
2. It offers a cross-platform support.
3. Instant file switching.
4. Black screen support distraction free editing mode.

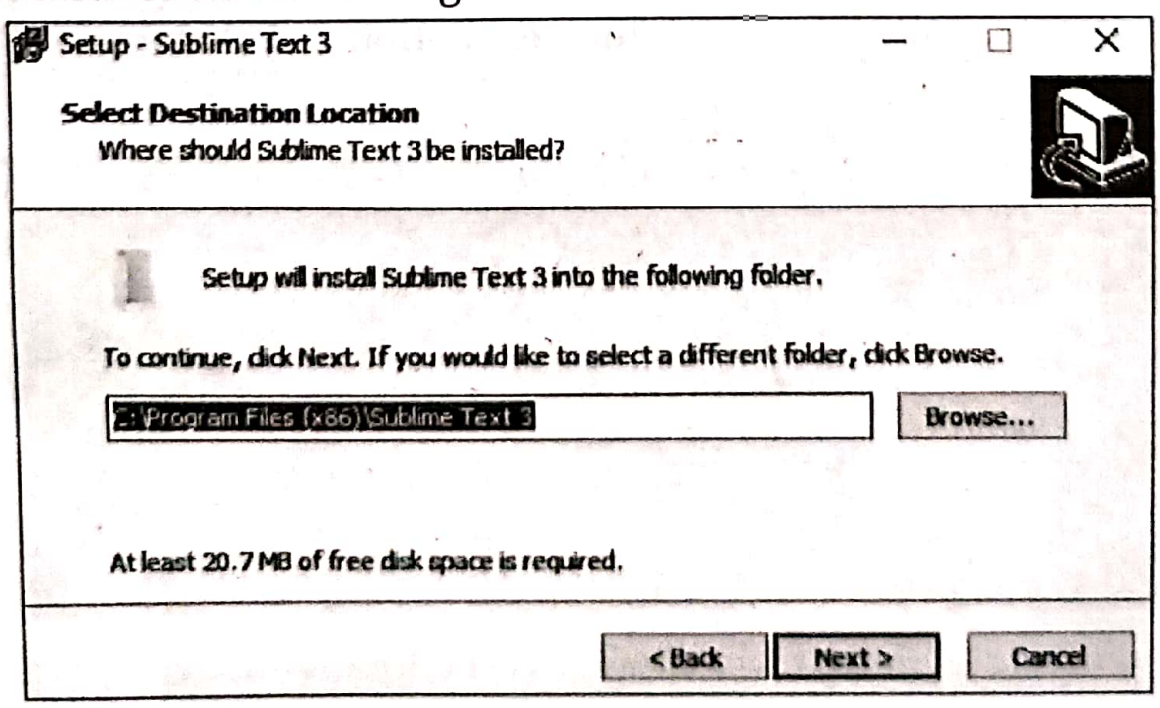


Figure 1.17: Select Destination Location

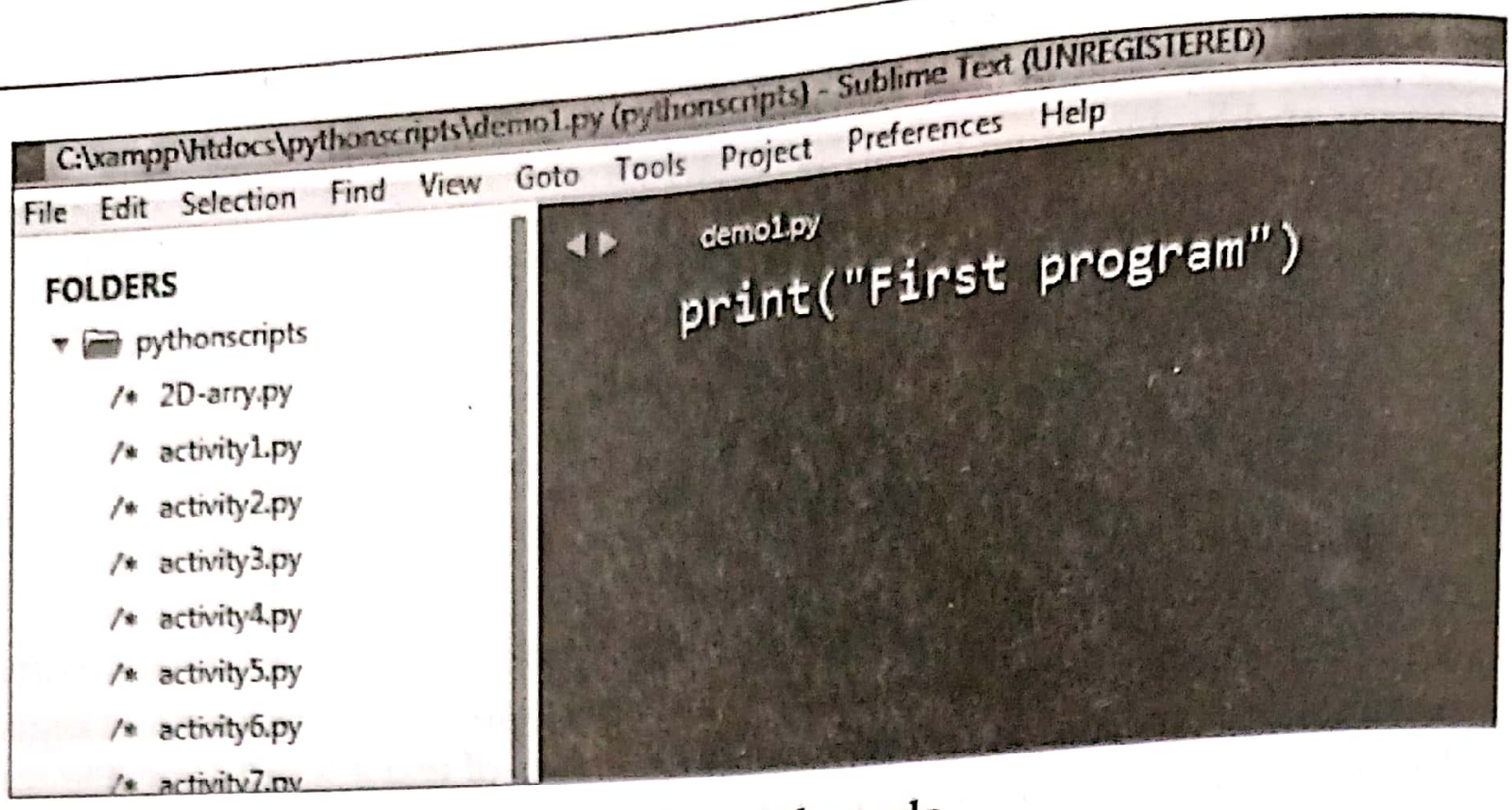


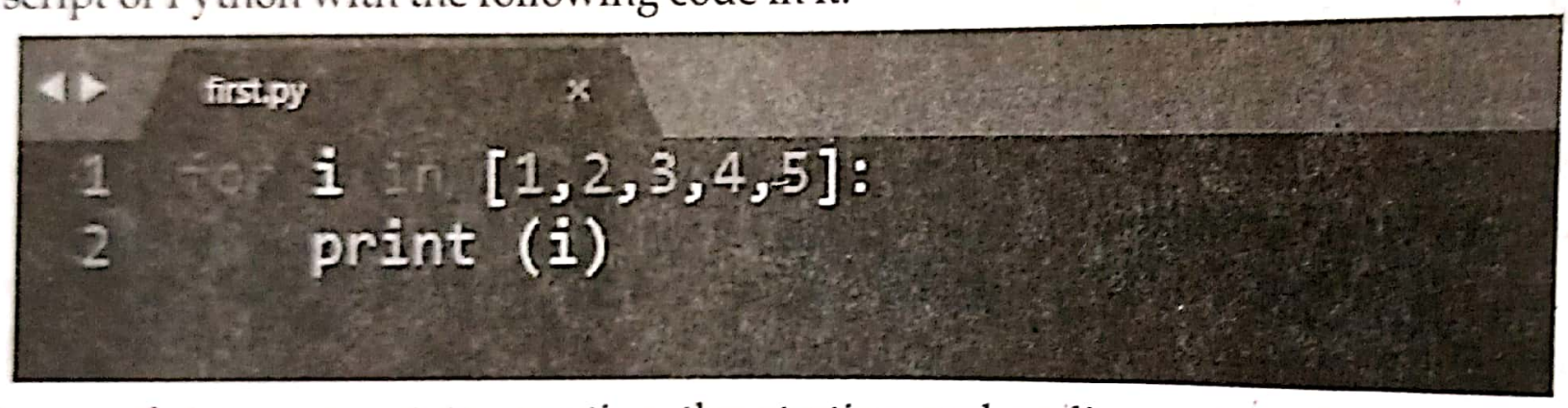
Figure 1.22: Insert the code

4. As your code is on Python project, we will save the file as **demo1.py** as shown in *Figure 1.22.*

Editing Text Document

Code editors allow the users to edit the code scripts and text documents with various shortcut keys. In this example, you will see how to add print statements at two locations.

1. The first script of Python with the following code in it.



In the given script, you want to mention the starting and ending point of the loop, using simple statements. You can write the print statement wherever required. In the Sublime Text editor, you can simultaneously add comments and statements in two places with a shortcut of **Ctrl+Cursor** for Windows. *(See Figure 1.23)*

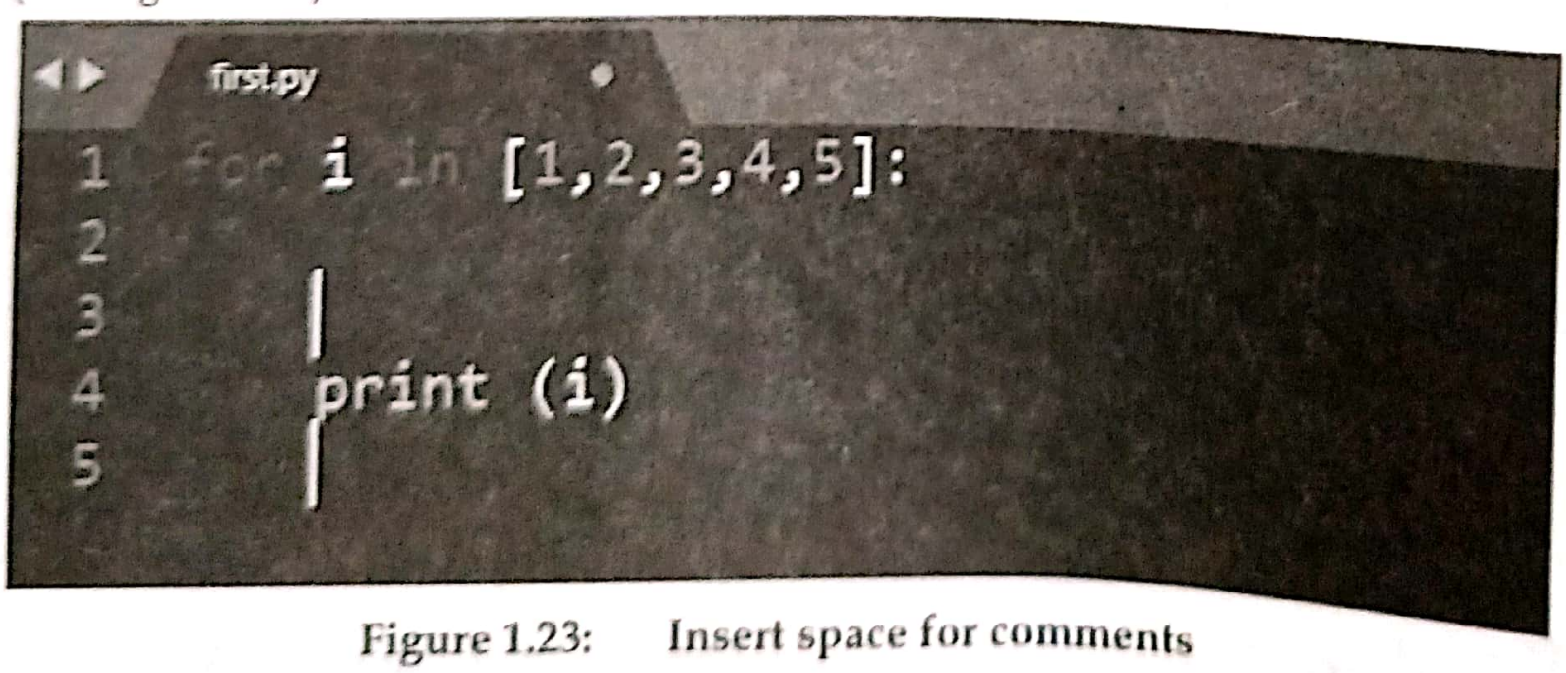


Figure 1.23: Insert space for comments

Now, you can insert the print statements at both the locations of the mentioned cursor points, as shown in Figure 1.24.

Scanned with OKEN Scanner

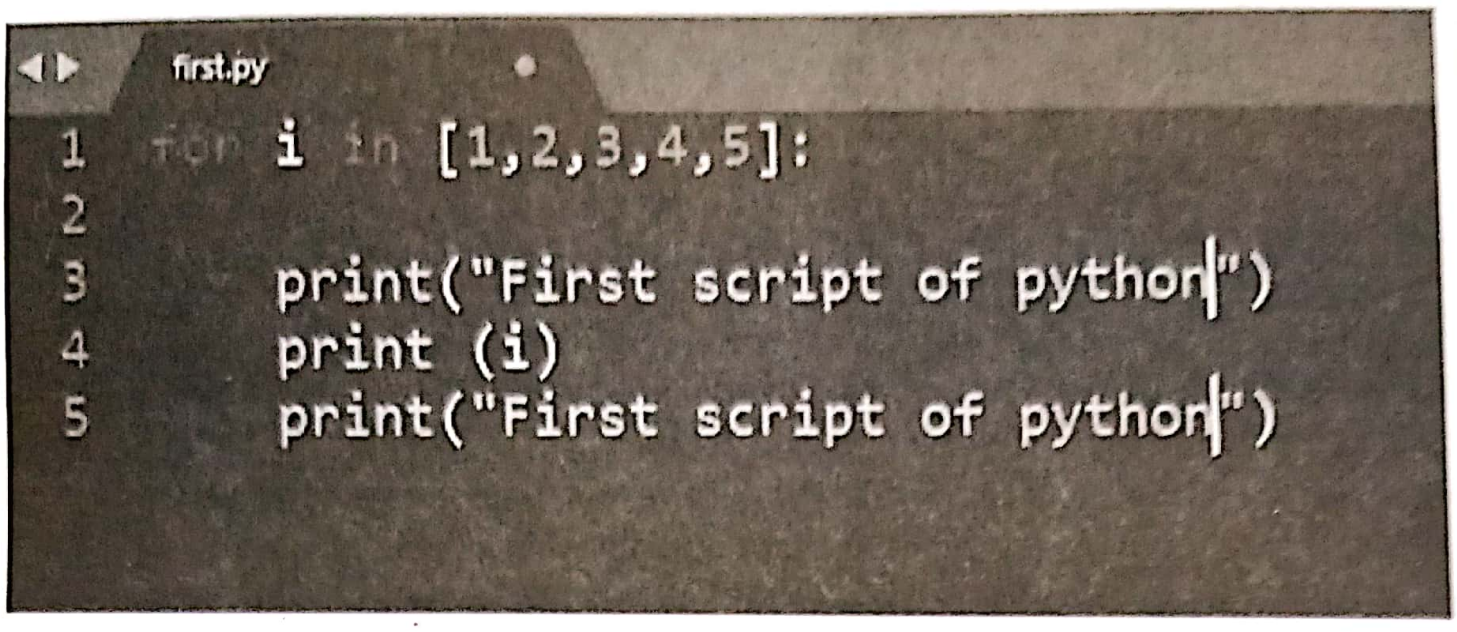


Figure 1.24: Insert comments

Finding Occurrences

Sublime Text editor includes a feature to find the occurrences of the keywords included in the scripts. If you want search for a keyword, say print from the given code, you can use **Ctrl+D** as shown in *Figure 1.25*.

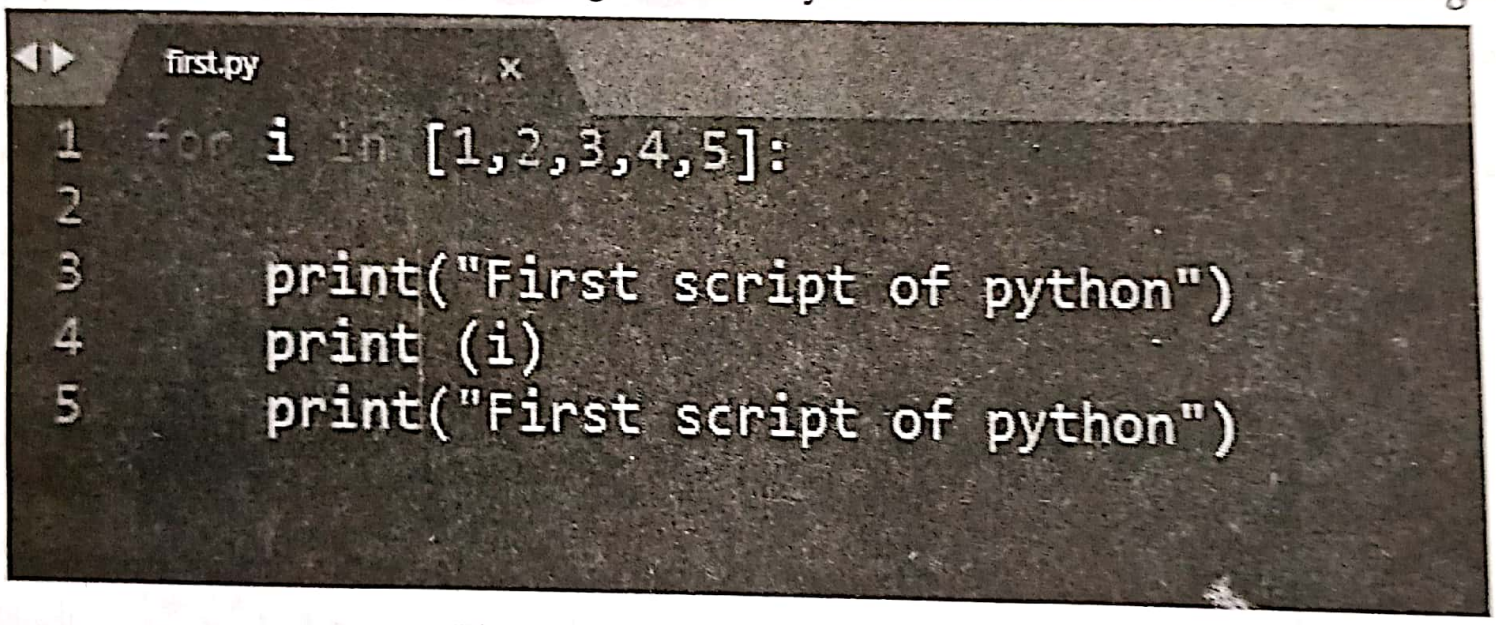
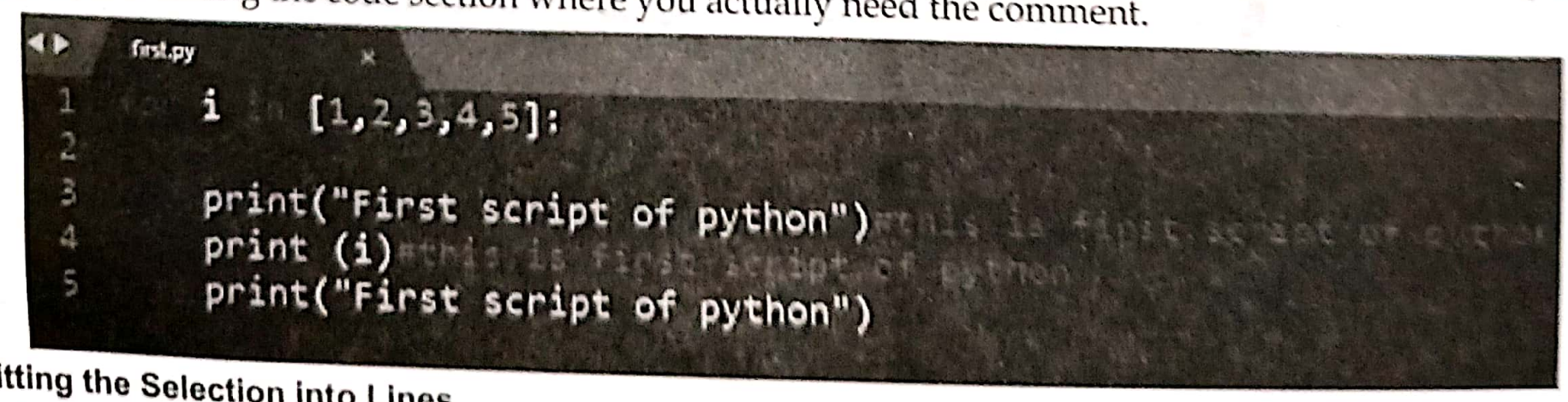


Figure 1.25: Find keywords

Appending Comments in Line

We can append comments at the end of line using the shortcut key **Ctrl+Shift+L** for Windows operating system after selecting the code section where you actually need the comment.



Splitting the Selection into Lines

The key combination **Ctrl+Shift+L** allows you to select the block of lines and split them on Windows as shown below:

Scanned with OKEN Scanner

```
first.py
1   i   [1,2,3,4,5]:
2
3       print("First script of python")
4       print (i)
5       print("First script of python")
```

Wrapping Complete Paragraph in HTML File

The key combination **Alt+Shift+w** for Windows, allows the user to spit the section of code and c
various paragraphs. It is known as a wrap selection with html tag. This also allows you to include HT
tag using Sublime editor. After wrapping the text, the text wrapped with a HTML tag as shown below.

```
first.html
1  <p>This is demonstration of first html file</p>
```

Finding all Occurrences

The key pattern **Ctrl+Shift+F** allows you to find a keyword in a given project as shown in Figure 1.26.
The shortcut key opens a window with **three** options included in it, namely, **find, where** and replace.

- Find refers to the search of the particular keyword.
- Where mentions the section as to which repository is searched.
- Replace mentions the keyword, which is replaced instead of find keyword.

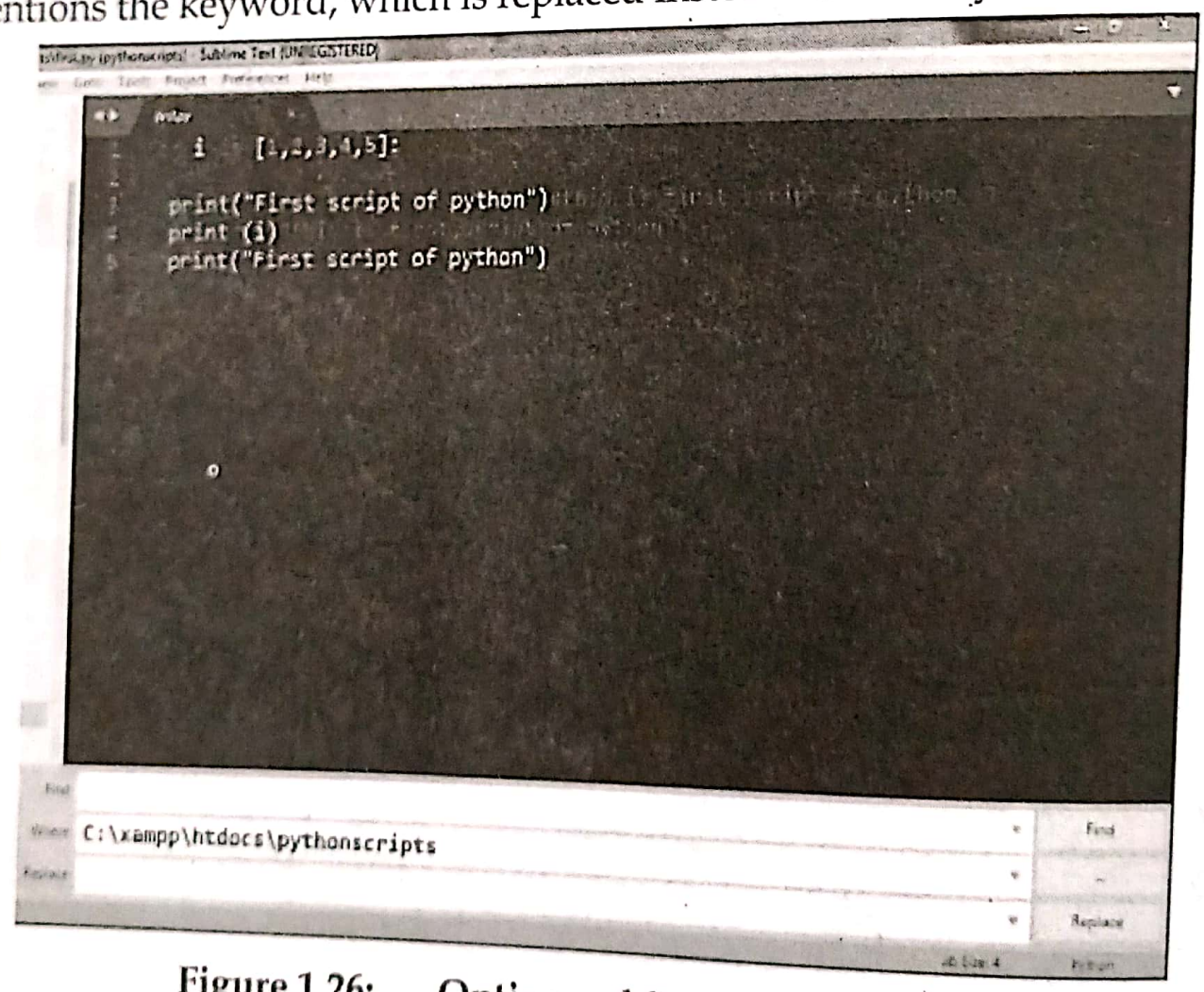


Figure 1.26: Options of finding keywords

Conclusion

This chapter begins; with Internet is a network of global exchanges - including private, public, business, academic and government networks - connected by guided, wireless and fiber-optic technologies. Next, we discusses, the

Scanned with OKEN Scanner

World Wide Web is a collection of various pages written in HTML markup language. Each page available on the website is called a web page and the first page of any website is called home page for that site. Next, we discusses the two most popular terms used in web development Front end and back end. Next, we discusses about client and server side scripting languages. Next, you will learn Responsive Web designing is an approach whereby a designer creates a web page that responds to or resizes itself depending on the type of device. Then, we will learn types of Websites Static and Dynamic websites. Finally, you will learn, basic text-editing program such as Notepad++ and Sublime Text Editor to make writing code easier and how to create and editing the file and save the file.

REVIEW QUESTIONS WITH ANSWERS

1. Multiple Choice

1.1 To make your website mobile friendly, you can make your website.
a. Responsive
b. Reactive
c. Fast Loading
d. Light

1.2 Which of the following the use of small programs that are executed based on user interaction with a web page is referred to as:
a. Web scripting
b. Client-side scripting
c. Server-side scripting
d. HTML

1.3 Server side scripting websites consists of two main parts, one is scripting languages and other is:
a. Server interaction mode
b. User interaction mode
c. Scripting engine
d. None of these

1.4 Connecting websites to backend servers, processing data and controlling behavior of higher layers, all these things are done by:
a. HTML language
b. JavaScripts
c. CSS scripts
d. None of the above

1.5 Server side scripts is:
a. A visible to the end user
b. Defined to user
c. Invisible to the end user
d. Both (b) and (c)

1.6 A series of specially designed documents, which are all liked to viewed on Internet is called:
a. World Wide Web
b. Web server
c. Search engine
d. Both (b) and (c)

1.7 This is an Internet forum to discuss wide range of topics like current news, articles and current affairs, etc.
a. E-Discussion
b. Newsgroups
c. Chatting
d. E-Learning

1.8 A browser is a:
a. Line that delineates the edge of a programmer window
b. Device used for modifying the window
c. Software programme especially designed for viewing web pages on the Internet
d. None of the above

1.9 A website address is a unique name that identifies a specific _____ on the web.
a. Web browser
b. PDA

Scanned with OKEN Scanner

c. Website
d. None of the above

1.10 Internet's initial development was supported by:

a. Communication
b. Information Retrieval
c. Presentation of Information
d. All of the above

2. State whether the following statements are True or False

2.1 When a client browser requests a web page and the page is sent back, a direct physical connection between client and server is formed while the information is being sent.

2.2 It is extremely expensive and difficult for private individuals to post information to the World Wide Web.

2.3 The Internet is considered a very dynamic network.

2.4 TextMate is the name of the application that allows us to work in both code and design views

2.5 Creating a website is a complex process.

2.6 Once you complete the planning phase you need to set up the structure of the site by creating a storyboard.

2.7 The first webpage that appears when the viewers go to a website is called the default page.

2.8 Static websites are basic pages that require simple code and design elements to create.

2.9 Back end development refers to the server side of an application and everything that communicates between the database and the browser.

2.10 A web browser can display any ASCII text file has a file name with the file extension txt.

3. Match the following:

3.1	An Internet address is	a.	WAP
3.2	The process of trading goods over the Internet is known as	b.	HTML, Javascript an CSS
3.3	Most world wide web pages contain commands in the language	c.	32 bits in length
3.4	World Wide Web was proposed by	d.	HTTP
3.5	Front-end is built using a combination of	e.	E-commerce
3.6	It is an application that stores the Web pages requested by user	f.	Internet. infrastructure and Service providers
3.7	Notepad++	g.	ARPANET
3.8	The first network that planted the seeds of Internet was	h.	TCP/IP
3.9	Protocols is used for WWWW	i.	It supports tabbed editing, which allows working with multiple open files in a single window
3.10	Main protocol used in Internet	j.	Web serve
		k.	NSFnet

4. Fill in the blanks

a. Sublime Text b. Internet c. Applications /Web server

Scanned with OKEN Scanner

d. ISP e. Static website f. Client
g. Hyperlink h. Software i. IP address 4.6
j. Dynamic k. Angular JS l. Dreamviewer

4.1 The vast network of computers that connects millions of people all over the world is called _____.

4.2 A word in a web page that, when clicked, opens another document is called _____.

4.3. A website main page is called its _____.

4.4 _____ is a cross-platform source code editor with a Python application programming interface.

4.5 For a small website, one needs to by space from the _____.

4.6 A computer on Internet are identified by _____.

4.7 Computer that requests the resources or data from other computer is called as _____ computer.

4.8 Static websites are generally designed using _____.

4.9 Student A used combination of PHP, MYSQL and HTML for his project. What is the type of _____ website developed by Student?

4.10 _____ is a JavaScript opens source front-end framework that is mainly used to develop single page web applications.

Descriptive Questions with Answer

5.(a)What is Sublime and why use Sublime Text?

Answer

Sublime Text editor is a full highlighted Text editor for altering documents or a code base. It incorporated different features for altering code base which causes developers to monitor changes. Different features that are supported by sublime are as follows:

- Syntax highlight
- Auto indentation
- File type recognition
- Plug-in and Packages
- Macros
- It is utilized as an Integrated Development Editor like Visual Studio code and NetBeans.

When you utilize a reasonable Text editor, you can make the most of its helpful features. It offers its users the below advantages:

- Ability to solve linker errors.
- Keeping track of all files and folders to work with.
- Connectivity with version control systems like Git.
- Problem solving capabilities.
- Keeping color combination for syntax combination.

5.(b) What is Internet and mention some uses of Internet?

Answer

Internet, computer-based global information system. The Internet is composed of many interconnected computer networks. Each network may link tens, hundreds, or even thousands of computers, enabling them to share information and processing power. The Internet has made it possible for people all over the world to communicate with one another effectively and inexpensively.

Scanned with OKEN Scanner

Companies, individuals, and institutions use the Internet in many ways.

1. Companies use the Internet for electronic commerce, also called e-commerce, including advertising, selling, buying, distributing products, and providing customer service. In addition, companies use the Internet for business-to-business transactions, such as exchanging financial information and accessing complex databases.
2. Businesses and institutions use the Internet for voice and video conferencing and other forms of communication that enable people to telecommute (work away from the office using a computer).
3. The use of e-mail on internet has greatly speeds communication between companies, and among other individuals.
4. Scientists and scholars use the Internet to communicate with colleagues, perform research, distribute lecture notes and course materials to students, and publish papers and articles.
5. Individuals use the Internet for communication, entertainment, finding information, and buying and selling goods and services.

5.(c) Define Website and Web-Server?

Answer

A web site is a related collection of World Wide Web files that includes a beginning file called a home page. *For example*, the website Yahoo has the home page address of http://www.yahoo.com. A website may consist of thousands of files spread out over many servers in worldwide locations.

A Web server is simply a computer with an Internet connection that runs software designed to send out HTML pages and other file formats.

6.(a) A What are different types of Website?

Answer

A website is a collection of linked web pages that share a unique domain name. Each web page of a given website provides links in the form of clickable portion of text that allows the user to move from one page of the website to another.

To access a website, type its domain name in your browser address bar, and the browser will display the website's main web page or home page.

There are main types of websites are as follows:

1. Brochure: A brochure website is the type of websites typically have a few pages, and will be used by small businesses. For example, a small company would only need a brochure website with a homepage displaying contact information and have a couple of photos of their work. It is like an online business card for potential customers.
2. e-Commerce: It is a website through which users are able to pay for a product or service online. This will normally involve one company selling to multiple users, but can also take the form of a multi-vendor ecommerce website also known as marketplace websites. Examples of eBay and Etsy.
3. Portal: A portal website brings together information from lots of different sources on the web. Most of the portals to offer these free services such as search engine, news, sports and weather. Web publishing reference tools such as yellow pages, stock quotes, and maps, shopping and e-mail and other communications services.
4. Social media: Social media websites are platforms which allows the sharing of images or ideas. They encourage online interaction and sharing. The most popular social media website is Facebook. Other social media sites include YouTube, Twitter, Instagram and LinkedIn.
5. Blog: A blog short for Weblog, is an informal Web site consisting of timestamp, or post articles, or journal format, usually listed in reverse chronological order. A blog that contains video clips called a

Scanned with OKEN Scanner

video blog. A microblog allows users to posrt short messages, usually between 100 and 200 characters, for others to read. Twitter is a popular microblog.

6.(b) What is a Responsive website?

A responsive website is based on the technology of computing device used by the visitor to display the site. It is basically website design that will looks good at any size – from a large desktop LCD monitor to the smaller screens and also used on smartphones and tablets.

Responsive website design (RWD) enables site designers to create once and publish the same content for all devices. It is a website development of rendering web pages in an efficient, optimized and easy-to-read format across a variety of devices and web browsers on different platforms.

Rather than the old-style approach of designing web pages for viewing on just desktop or laptop PCs, responsive design utilizes a variety of newer web development features and functionality to deliver an optimal view to users on mobile devices like smartphones and tablets as well as on traditional PCs and other electronic devices.

Because of their smaller display size, mobile devices typically require a modified layout for content to help users navigate through the website more efficiently, and with responsive design, web developers are able to code their web pages with the flexibility to render readable content at a variety of display sizes. This is due largely in part to websites that can continually and fluidly change, based on factors such as the viewport width.

6.(c) What is Internet? Explain evolution of Internet.

The Internet is a global system of interconnected computer networks that use the standard Internet protocol suite(TCP/IP) to link several billion devices worldwide. It is a *network of networks* that consists of millions of private, public, academic, business, and government networks, of local to global scope, that are linked by a broad array of electronic, wireless, and optical networking technologies. The Internet carries an extensive range of information resources and services, such as the inter-linked hypertext documents and applications of the World Wide Web (WWW), the infrastructure to support email, and peer-to-peer networks for file sharing and telephony. The origins of the Internet date back to research commissioned by the United States government in the 1960s to build robust, fault-tolerant communication via computer networks. While this work, together with work in the United Kingdom and France, led to important precursor networks, they were not the Internet. There is no consensus on the exact date when the modern Internet came into being, but sometime in the early to mid-1980s is considered reasonable. From that point, the network experienced decades of sustained exponential growth as generations of institutional, personal, and mobile computers were connected to it.

Most communications media including telephone, music, film, and television are being reshaped by the Internet, giving birth to new services such as voice over Internet Protocol (VoIP) and Internet Protocol television (IPTV). Newspaper, book, and other print publishing are adapting to website technology, or are reshaped into blogging and web feeds. The Internet has enabled and accelerated new forms of human interactions through instant messaging, and social networking. Online shopping has boomed both for major retail outlets and small artisans and traders. Business-to-business and financial services on the Internet affect supply chains across entire industries.

The Internet has no centralized governance in either technological implementation or policies for access and usage; each constituent network sets its own policies. Only the overreaching definitions of the two principal name spaces in the Internet, the Internet Protocol address space and the Domain Name System, are directed by a maintainer organization, the Internet Corporation for Assigned Names and Numbers (ICANN). The technical underpinning and standardization of the core protocols (IPv4 and IPv6) is an activity of the Internet Engineering Task Force (IETF), a non-profit organization of loosely affiliated international participants that anyone may associate with by contributing technical expertise.

Scanned with OKEN Scanner

7.(a)What are the disadvantages of the Internet?

1. There is a lot of wrong information on the Internet. Important files can be easily taken by the hackers.
2. The unnecessary e-mails, advertisements etc. sometimes are said to be spam because they have the ability to show down the system and makes the users to face lots of problems.
3. Pornography that can get in the hands of young children too easily.
4. Easy to waste a lot of time on the Internet. You can start surfing, and then realize far more time has passed than your realized.
5. There are a lot of unscrupulous businesses that have sprung up on the Internet to take advantage of people.
6. The malware or virus threats are so deadly that affects the system to a greater extend. It immediately deletes all important files and the system ends up being crashed.
7. 8. The people using internet often will forget the difference between virtual and real world. This causes the people to get depressed quickly and it leads to social isolation and obesity problem due to lack of any physical exercises. So its better to play outdoor rather in the Internet.

7.(b) Difference between Frontend and Backend

Frontend	Backend
It refers to the client-side of the application.	It refers to the server-side of the application.
Users can see and interact with a part of website.	Everything that happens behind the scenes.
It includes everything that attributes to the visual aspects of websites.	It includes a web server that communicates with a database to serve request that the frontend presents.
It forms the basis of what users can touch and experience on their web browsers.	The website that is never visible to the end users.
The essentials of frontend web development include HTML, CSS and JavaScript	The essentials of backend development include Ruby, Python, Java, Net etc.

8.(a) What is Web Browser? Name any two type of web browser.

Web browser is a software application that allows us to see the information on the web. When a user requests a particular website, the web browser retrieves the necessary content from a web server and then display the resulting web page.

Web browser can show text, audio, video, animation etc. It is the responsibility of a web browser to interpret text and commands contained in the web page.

j. **Internet Explorer:** This browser was made by Microsoft Corporation and bundled together into Windows operating system. The latest operating system, Windows comes with Microsoft latest web browser named Internet Explorer 11. Internet Explorer supports HTML 4.01, CSS Level 1, XML 1.0, and DOM Level. Microsoft Edge replaced IE as the Windows default browser beginning with Windows 10, but IE is still a popular browser for users running older versions of windows.

k. **Microsoft Edge:** Microsoft Edge is the new browser built into Windows 10. Internet Explorer 11 remains available alongside Edge on Windows 10 for compatibility. The browser includes an integrated Adobe Flash player and a PDF reader. It also integrates with online platforms in order to provide voice control, search functionality, and dynamic information related to searches within the address bar. Users can make annotations to web pages that can be stored with One Drive but cannot save HTML pages and does not support ActiveX.

Scanned with OKEN Scanner

8.(b) Write down the difference between Static and Dynamic web site.

The difference between of static and dynamic website are as follows:

a. Dynamic websites can be easily upgraded by end users. Static websites is difficult to upgrade.
b. The theme of website and content of website are fixed. Website designs and content can be changed on runtime.
c. With the help of markup language, static websites load quickly whereas dynamic websites delays to load because of more processing time.
d. Static website provides information to their clients like an organization website. Dynamic website is where content changes frequently, like an Electronic-commerce site etc.
e. Static website directly run on browser and does not require other internet application language. This website can be created from HTML and CSS. Dynamic website run the application on server and the result will be display on webpage. So, this requires server application language like PHP, JSP etc.
f. Static websites are easy to develop and an experienced person can develop it. But in case of Dynamic websites, it cannot be developed easily because it requires qualify developers to create it, manage it, test it and maintain security of application and database.
g. In Static website, if you want to change the page content then we have to upload that page on server many times. Dynamic sites provide some facilities that it possible to change the page content using server application. There is no need to upload the page on server.

9.(a) What are Web publishing tools? What is the meaning of document management in context of Web publishing?

Answer

Web publishing tools consist of tools and resources like programming languages, designing software, animation software etc. which help to create and develop web pages. There are a number of Web Publishing Tools developed by various corporations.

A web content management system (WCMS) is a software system that provides website authoring, collaboration, and administration tools designed to allow users with little knowledge of web programming languages or markup languages to create and manage website content with relative ease.

Most systems use a content repository or a database to store page content, metadata, and other information assets that might be needed by the system. Most systems use server side scripting to improve performance.

A web content management system is used to control a dynamic collection of web material, including HTML documents, images, and other forms of media. A CMS facilitates document control, auditing, editing, and timeline management. A WCMS typically has the following features:

Automated templates: Create standard output templates (usually HTML and XML) that can be automatically applied to new and existing content, allowing the appearance of all content to be changed from one central place.

Access control: Some WCMS systems support user groups. User groups allow you to control how registered users interact with the site. A page on the site can be restricted to one or more groups. This means an anonymous user (someone not logged on), or a logged on user who is not a member of the group a page is restricted to, will be denied access to the page.

Easily editable content: Once content is separated from the visual presentation of a site, it usually becomes much easier and quicker to edit and manipulate. Most WCMS software includes WYSIWYG editing tools allowing non-technical users to create and edit content.

Scalable feature sets: Most WCMS software includes plug-ins or modules that can be easily installed to extend an existing site's functionality.

Scanned with OKEN Scanner

Document management: CMS software may provide a means of collaboratively managing the life cycle of a document from initial creation time, through revisions, publication, archive, and document destruction.

Versioning: Like document management systems, CMS software may allow the process of versioning by which pages are checked in or out of the WCMS, allowing authorized editors to retrieve previous versions and to continue work from a selected point. Versioning is useful for content that changes over time and requires updating, but it may be necessary to go back to or reference a previous copy.

9.(b) List various features of web browser.

There are many web browsers available having different features. Some of the basic features mentioned below:

It handles requests for HTML files, interprets links, and deals with embedded images, audio, and video elements.

It keeps the history of the websites visited.

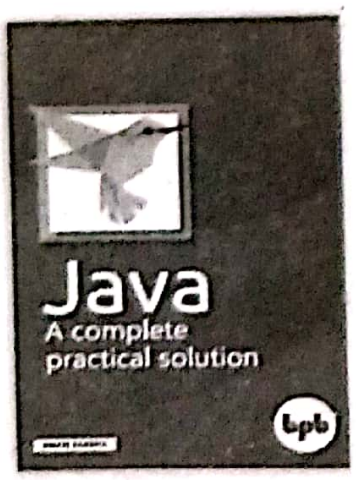
It provides a row of buttons at the top of the browser window for browsing the Internet conveniently.

It supports web standards currently in use such as HTML, HTTP, JavaScript, and Unicode.

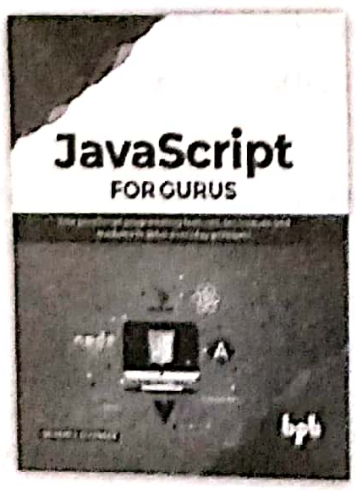
It connects to an e-mail program for importing favorites/bookmarks and sending and receiving e-mails from the most commonly used applications and formats.

Answers

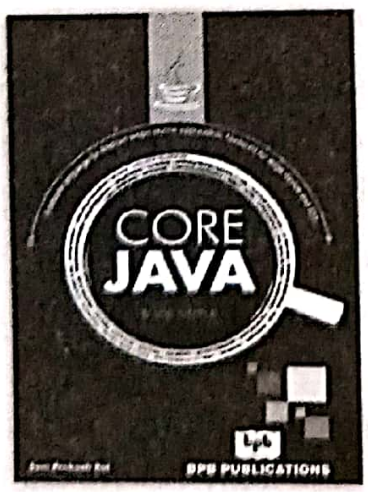
1.1	a	1.2	a	1.3	c	1.4	b	1.5	c	1.6	a
1.7	b	1.8	c	1.9	c	1.10	d				
2.1	F	2.2	F	2.3	T	2.4	F	2.5	T	2.6	T
2.7	F	2.8	T	2.9	T	2.10	F				
3.1	c	3.2	e	3.3	a	3.4	f	3.5	b	3.6	j
3.7	i	3.8	g	3.9	d	3.10	h				
4.1	b	4.2	g	4.3	c	4.4	a.	4.5	d	4.6	i
4.7	f	4.8	l	4.9	j	4.10	k				



Java
A Complete Practical Solution
ISBN: 9789388176507



JavaScript for Gurus
ISBN: 9789389423655



CORE JAVA Made Simple
ISBN: 9789386551979



Java in Depth
ISBN: 9789386551573



Mastering HTML, CSS & Java Script Web Publishing
ISBN: 9788183335157

OTHER TITLES OF INTEREST

Scanned with OKEN Scanner

CHAPTER2

HTML Basics

2.1 Introduction

HTML is the standard markup language for creating web pages developed by **Tim Berners-Lee** in **1991**. It used by the browser to manipulate text, images and other content to display it in required format. Each page contains a series of connections to other pages called **hyperlinks**. The HTML standard and all other Web-related standards are developed under the authority of the **World Wide Web Consortium** (W3C). Standards, specifications, and drafts of new proposals found at **http://www.w3.org/**. The most recent work is the HTML 4.0 Specification, which is growing in support by major browsers.

Structure

Basic Structure of HTML

- Formatting Tags
- Tables
- Lists
- Frames
- HTML 5 Introduction and New Elements
- HTML embed multimedia
- HTML Layout and Iframe

Objectives

The reader will be able to understand to develop static website using different HTML Controls.

- Basic structure of HTML
- Formatting Tags
- Tables
- Lists
- Frames
- HTML 5 Elements
- HTML Form Validations

2.2 Introduction to HTML Basics

Hypertext Markup Language (HTML) is a language understood by your web browser like Internet Explorer or Netscape Navigator. All the documents available on the World Wide Web have written in HTML, which run on any type of computer having a Web browser. It has a collection of styles (indicated by tags) that define the various components of a World Wide Web document. An HTML document can be opened in any operating system or any hardware such as PC, Macintosh or UNIX operated machine.

2.2.1 Basic Structure of HTML

HTML stands for Hyper Text Markup Language and is the basic structural element that is used to create web pages.

Scanned with OKEN Scanner

HTML Page Structure

An HTML page consists of the following tags:

<!DOCTYPE>: A DOCTYPE declaration must be specified in the first line of each web document.

<HTML>: This tag specifies the document as an HTML document.

<HEAD>: This tag contains information about the document, including its title, scripts used, style definitions and document descriptions.

<TITLE>: This tag contains the document title. The title specified inside **<Title>** tag appears on the browser's title bar. You should be very careful while choosing the title because a title is also used to identify a Web page for search engines.

<BODY>: This forms the largest part of the HTML document. The **<Body>** tag encloses all the tags, attributes and information to be displayed in the Web page. The information written within the body tag is displayed in the browser Window.

Different HTML Tags

Different HTML tags are explained in detail in the following sections.

The following example shows the tags that make up the standard skeletal structure of an HTML document:

```
<!DOCTYPE>
<HTML>
<HEAD>
<TITLE>Document Title</TITLE>
</HEAD>
<BODY>
Contents of Document
</BODY>
</HTML>
```

<!DOCTYPE> Type

The DOCTYPE tells the browser which version of HTML the page is written.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.o Transitional//EN"
      http://www.w3.org/TR/xhtml1-transitional.dtd>
```

HTML Element <Html>

The HTML element tells the web browser that the file contains HTML-coded information. It is also called **HTML root element** and used to wrap all the code. Most elements are written with a start tag and an end tags with content in between. HTML documents are structured into two parts, the Head and Body. These are contained within the HTML element.

```
<HTML>
..
</HTML>
```

HEAD Element <Head>

The <head> element is the container for all the head elements, which provide information about the page document such as meta-information and document type information. It is place before the <body> tag. The contents of the head are not displayed and a <body> element surrounds all the content (text, images, videos, link etc.) that will be displayed in a web browser.

Scanned with OKEN Scanner

```
<HTML>
    <HEAD>
      <TITLE>Document Title </TITLE>
   </HEAD>
</HTML>
```

TITLE Element <Title>

Title element contains the document's title and identifies its contents in a global context. The title is displayed somewhere in the browser window (usually at the top), but not within the text area. You should choose title head somewhat descriptive, unique and relatively short.

```
<TITLE>Document Title</TITLE>
```

BODY <BODY>

The second and largest part of HTML document is the body, which contains the contents of the document (displayed within the text area of a browser window). The tags explained below are used within the body of the HTML document.

The tags used to indicate the start and the end of the main body of textual information are:

```
<BODY>
...
</BODY>
```

The default characteristics of a page, such as background color, text color, font size and font width, etc., can be specified as attributes of the **<BODY>** tag. The attributes for the <BODY> tag are as follows:

a. BACKGROUND
b. BGCOLOR
c. TEXT
d. LINK, ALINK AND VLINK
e. LEFT MARGIN
f. TOP MARGIN

Head Section and Elements of Head Section

The <head> element is a container for metadata. It contains other head elements such as <title>, <meta>, <link>, <style>, <script> and <base>.

HTML<title> element

The <title> element is displayed in a bowser tab. An example of title element.

```
<html>
<head>
    <title>A simple HTML document </title>
</head>
<body>
    <!—Document body..>
</body>
</html>
```

HTML <meta> Element

It is used to specify page description, author and keywords etc. It can also be used to refresh the page or set

Scanned with OKEN Scanner

cookies. An example of meta element.

```
<head>
   <title>Specifying Metadata</title>
     <meta charset="utf-8">
   < meta name="Author" content="Web Designing">
    <meta name="keywords" content="JavaScript, CSS, HTML5">
   <meta name="description" content="HTML BASICS">
</head>
```

HTML <link> Element

The <link> element is used to load an external style sheet into HTML document, but it can also be used for search engines by providing links to relevant resources on the website's navigational structure. An example of <link> element.

```
<head>
    <title>Linking StyleSheet</title>
    <link rel="CSS Stylesheet" href="Style.CSS">
</head>
```

The syntax is <link rel=" " href="">. The rel attributes specifies the relationship between the document containing the hyperlink and the destination resources. The href attributes specifies the address of the hyperlink.

HTML <style> Element

The <style> element is used to define embedded style information for an HTML document. The syntax is written as <style> </style>. An example of <style> element.

```
<head>
    <title>Embedding Style Sheets</title>
    <style type="text/css">
        H1 { font -size: 15px;
             color: blue; }
      P { font -weight: normal;
          color: green;}
     </style>
    </head>
```

HTML <script> Element

The <script> element is used to define client-side script such as JavaScript within the document or embedded external JavaScript file through the SRC attribute. The syntax is written as <script> </script> with the script inserted between the start and end tags.

```
<html>
<head>
    <title>Adding JavaScript</title>
    <script type="text/javascript" scr="script.js"> </script>
</head>
<body>
   <!—Document body>
</body>
</html>
```

Scanned with OKEN Scanner

HTML <base> Element

HTML <base> element specifies a base URL for all relative links contained in the document. The Syntax <base> is written as **<base href="" target="">**. It is used either a **href** attribute or target.

```
<html>
<head>
    <title>Define a base URL</title>
    <base href="https://abc.tutorial.com/" target="_blank">
</head>
<body>
    <p>Go to <a href="html-Elements/">HTML Elements</a></p>
</body>
</html>
```

The <base> tag is inserted between the <head> tags, and there must be enter more than one <base> element in the document.

2.3 Formatting Tags

HTML Formatting tags such as <b>, <i>, <u>, <strike> etc., are used to make some text on your web pages to appear differently than normal text.

Boldface: The boldface tag **<B> </B>** is used to make the text you want to show in Boldface style.

Italics: The italics tag **<I> </I>** is used to mark the text you want to display in *Italics* face style.

Underline: The tag **<U> </U>** is used to make the text you want to display the text as **Underlined**.

Strikethrough: The tag <s> </s> is used to make the text to be strike out. The content of strikethrough tags would appear with a horizontal line over the text.

Example 1:

The following Example 1 shows the text style formatting tags as applied to the HTML document and the output as shown in *Figure 2.1*.

```
<HTML>
<HEAD>
<TITLE>A Simple HTML Styles tags</TITLE>
</HEAD>
<body BGCOLOR=BLUE TEXT=WHITE>
<H1>Welcome to the World of HTML</H1>
<P><b>This text is displayed in Boldface</b></P>
<P><I>This text is displayed in Italics</I></P>
<P><U>This text is displayed in Underlined</U></P>
<P>This text is displayed in between <s>strikethrough </s> typeface</p>
</body>
<address>Created by Jain</address>
</HTML>
```

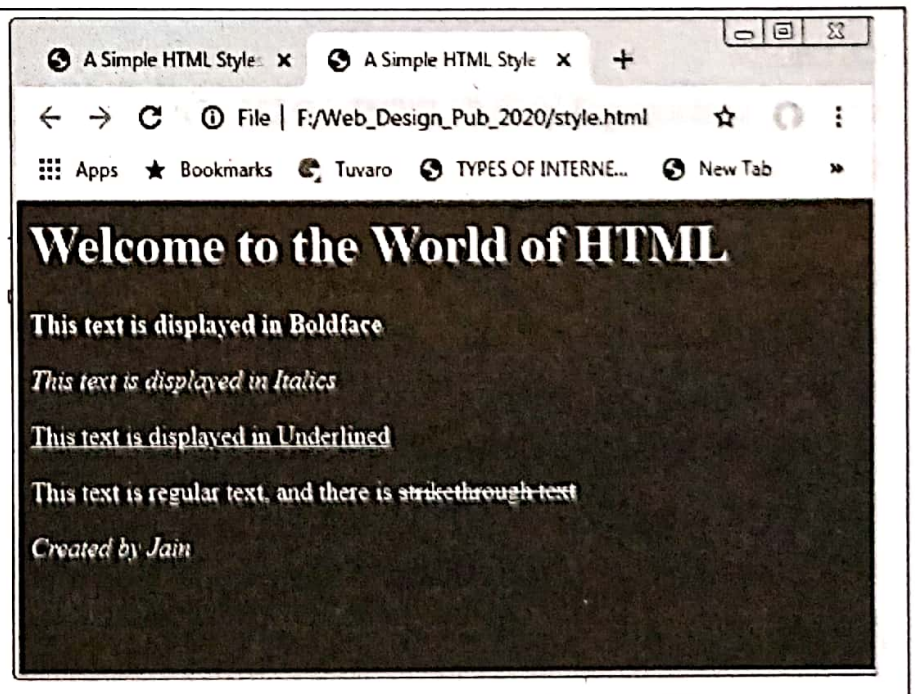


Figure 2.1: Viewing in the Chrome

Scanned with OKEN Scanner

HTML DIV TAG

The HTML element (or **HTML Document Division Element**) element is a block level element browsers will render a line break before and after its content. It can be used to group elements for styling purposes (using the **class** or **id** attributes), or because they share attribute values, such as **lang**.

The **<DIV>** tag using Styles and different attributes these are:

- The <DIV> tag is for Division, as in dividing or sub-dividing a Page into many Sections or functional blocks, for example: Header, Footer, Navigation pane, Main Content pane, Right Pane etc.
- This HTML <DIV> tag is a container for other HTML items or elements. This HTML-Div Tag is used to layout HTML elements on a page, and can be used to group HTML Elements together. Ultimately, you can say that <DIV> tag is container unit that condenses other page elements and divides the HTML document into sections.
- You can use the <DIV> tag to layout your page, the other way to layout your HTML web page is table. I would like to suggest you to use Div tag for page payout.
- Web designers use <DIV> elements to combine together HTML elements and implement CSS styles to many elements at once. For instance, by covering a set of paragraph elements into a element, the designer can take advantage of CSS styles.
- Designer can apply a design to all paragraphs at once by applying a design style to the <DIV> tag instead of coding the same style for each paragraph element.
- Within a <DIV>, you can you can nest and align Paragraphs, Headings Lists, Definitions, Pre-formatted text, Images and Tables etc. You can use the **ALIGN** attribute for **Horizontal Text Alignment**.

2.3.1 Using the HTML <Div> Tag

The following section demonstrates, how to use div tag to layout your page. Simple script to display the content of tag.

Example 2:

```
<!DOCTYPE html>
<html>
<html>
<title>Example of HTML DIV TAG</title>
<body>
<div style="border:1px; background-color:black; color: white; padding:5px">
<p>Welcome to BPB Publications, here you get on latest books</p>
<p>All kinds IT related book</p>
</div>
</body>
</html>
```

The output of given coding in Example 2 as shown in Figure 2.2.

Attributes

id (name): The **id** attribute assigns an identifier to the related element. This identifier must be unique in the document and can be used to refer to that element in other instances (e.g., from client-side scripts).

Class (cdata): The **class** attribute assigns a class name (or a list of class names separated by spaces) to the container element. It is used together with style sheets and tells the browser the class (or classes) to which the element is related.

Scanned with OKEN Scanner

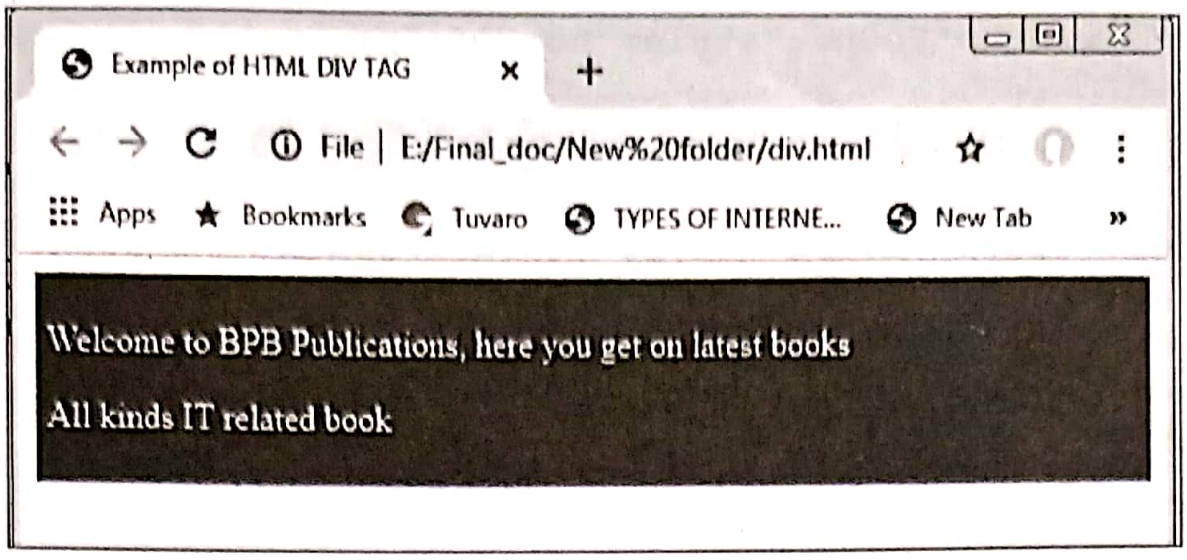


Figure 2.2: Previewing in chrome browser

Style (style): This attribute is used to define presentational attributes for the containing element, and its value should be composed by style sheets properties. Although in some cases it is to place presentational attributes in external files, relating them to elements with the "class" attribute.

Align

The align attribute specifies the horizontal alignment of the content inside a element.

The Syntax of align

```
<div align="left|right|center|justify">
```

The attribute value is:

- **Left:** Text is aligned to the left margin.
- **Right:** Text is aligned to the right margin.
- **Center:** Text is centered.
- **Justify:** Text is justified to both margins.

HTML Div Tag - Alignment and Style

First of all, you will add border in each Div tag. You can do this by using in-line CSS. In-line CSS is CSS inside HTML which is used for adding style to page.

- This is done by adding style="border: 1px solid red;" to each the div's tag. That will add a red border to one html div.
- It should be kept in mind while designing website that as per requirement, add the borders to HTML tags. You should also take care: what each div contains, where it starts and ends.
- HTML div's alignment can be changed by using the align attribute. To align a div using the align attribute.
- HTML Div tag can also be used to set layout width and Height, *for example,* **<div width:100px; Height:100px;">**

Example 3:

Script of HTML Div Tag Layout using the Align Attribute and adding Style.

```
<html>
<body>
<div style="width:400px;">
<div align="center" width="200px;"style="border:5px solid green;">This is first
block</div>
```

Scanned with OKEN Scanner

```
<div align="left" width="200px;"style="border:5px solid red;">This is second
block</div>
<div align="right" width="200px;"style="border:5px solid blue;">This is third
block</div>
<div align="center" width="200px;"style="border:5px solid pink;">This is fourth
block</div>
<div align="center" bgcolor="green" width="200px;align="center"
<div align="center" red;background:black;color:white;">Last block !
style="border:5px solid
Background is black</div>
</div>
</body>
</html>
```

The output of given coding in Example 3 as shown in Figure 2.3.

HTML-Div Inside of Div

Placing ** ** elements inside of other ** ** elements allows these elements can been further subdivide in Example 4.

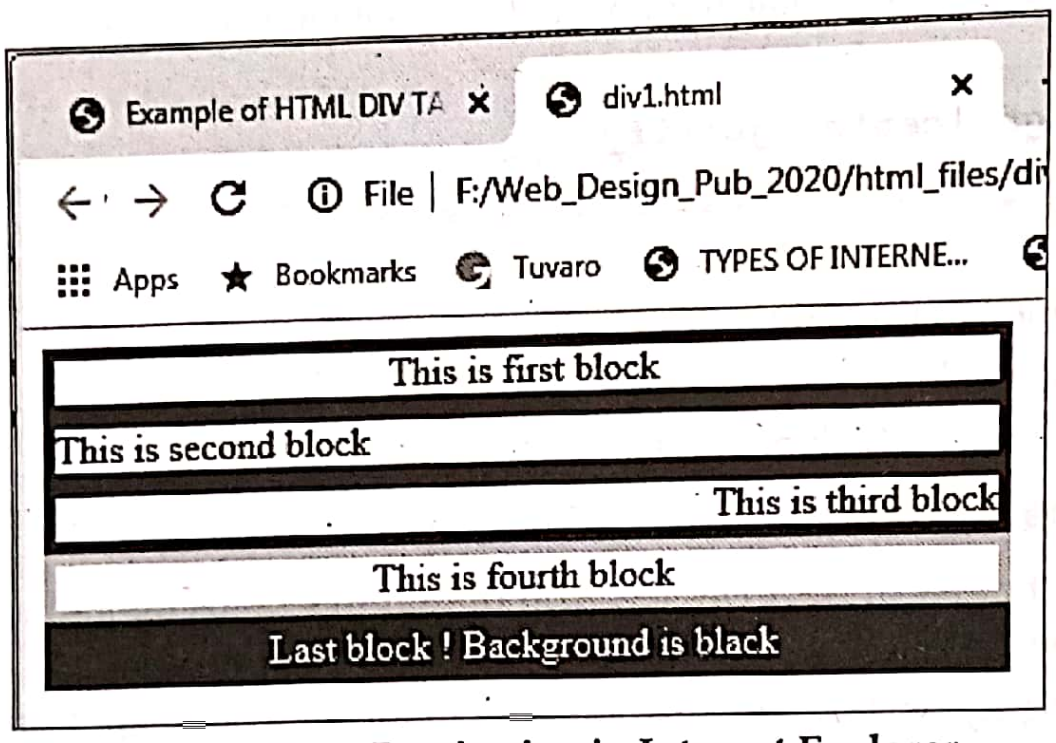


Figure 2.3: Previewing in Internet Explorer

Example 4:

```
<html>
</body>
<div id="myDiv" name="myDiv" title="Example Div Element" style="font-family:
Helvetica; font-size: 12pt; border: 2px solid black;">
  <div id="subDiv1" name="subDiv1" title="Subdivision Div Element"
style="color: #FF0000; border: 1px dotted green;">
    <h5>Paragraph 1</h5>
    <p>This is block of first paragraph</p>
    <p>Here is another paragraph.</p>
  </div>
  <br/>
  <div id="subDiv2" name="subDiv2" title="Subdivision Div Element"
style="color: #FF00FF;border: 1px dashed green;">
```

(Contd...)

Scanned with OKEN Scanner

```
    <h5>Paragraph 2</h5>
    <p>This is block of second paragraph</p>
    <p>Here is another paragraph.</p>
  </div>
</div>
</body>
</html>
```

The output of given coding in Example 4 as shown in Figure 2.4.

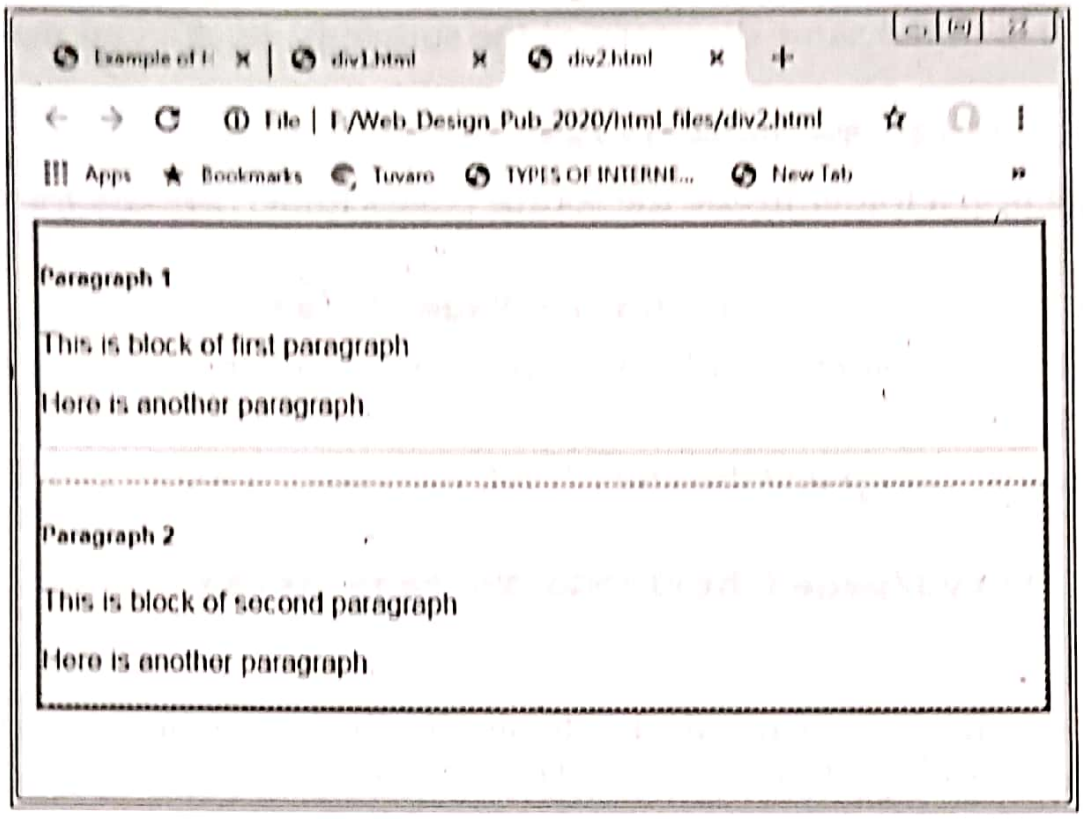


Figure 2.4: Previewing in chrome browser

Preformat Element <PRE> </PRE>

The **<PRE>** tag stands for **preformatted text.** It allows you to include preformatted text in HTML document (including spaces and lines). The text contained in this tag is rendered in a fixed pitch font like the **Courier font**. Most browsers follow the HTML standard for block level elements and create a paragraph break before the **<PRE>** start tag and after the **</PRE>** closing tag.

Anchor links and Named Anchors Image Tag

The HTML anchor tag defines a hyperlink that links one page to another page. It can create hyperlink to other web page as well as files, location, or any URL. The attribute **href** links to destination page or URL.

Hyperlinks are created with an **href** tag (**hyperlink reference**). The tag looks like this:

```
<a href="page1.html">Go To Page 1</a>
```

In this example, the text **Go To Page 1** becomes a hyperlink to a page called **page1.html**. The link looks like this: **Go To Page 1**.

There are two distinct types of hyperlink: **absolute** and **relative.** Absolute links are quite simple, but relative links are actually divided into two further categories: **document-relative** and **site-root-relative**.

2.3.2 Absolute Links

An absolute link (or **absolute URL link**) contains a complete internet address, just the same as if you typed the address into your browser's address bar. It does not matter where the source page is in relation to the

Scanned with OKEN Scanner

destination page, the link will always work as long as the destination page exists. Most absolute links begin with "http://" and follow this simple format:

```
<a href="http://www.mywebsite.com/index.html">Go To My Site</a>
```

Document-Relative Links

Document-relative links use directions from the source page to the destination page. There are three ways to do this - the method you use will depend on the location of the source and destination pages in relation to each other. To create hyperlinks from the page called **sourcepage.html** to other pages in the site.

1. If the destination page is in the same directory as the source page. All you need to specify is the source file name:
   ```
   <a href="page1.html">Go To Page 1</a>
   ```
2. If the destination page is in a folder inside the source page's folder. Specify the folder name and then the file name:
   ```
   <a href="directory2/page2.html">Go To Page 2</a>
   ```
3. If the destination page is in a folder outside the source page's folder. Use the special instruction ../ which signifies **one directory higher.**
4. The following link means "Go up one directory level, then go to a folder called directory3, then to a file called page3.html":
   ```
   <a href="../directory3/page3.html">Go To Page 3</a>
   ```

Site-Root-Relative Links

Site-root-relative links use a **single forward-slash** / to signify the instruction: "Start at the document root of the site and go down the directory path from there." The format is:

```
<a href="/main-directory/directory4/page4.html">Go To Page 4</a>
```

Despite the cumbersome name, this method is quite simple. The link begins at the same level as your domain and works down from there. The link above is the same as:

```
<a href="http://www.mysite.com/main-directory/
    directory4/page4.html">Go To Page 4</a>
```

This method has the significant advantage of staying intact if you move the source page. The exact same link will work from any page anywhere in the site.

A link is specified using HTML tag **<a>**. This tag is called **anchor tag** and anything between the opening **<a>** tag and the closing **** tag becomes part of the link and a user can click that part to reach to the linked document. Following is the simple syntax to use <a> tag.

```
<a href="Document URL"…. attributes- list>Link Text</a>
```

Example 5:

```
<!DOCTYPE html>
<html>
<head>
<title>Hyperlink Example</title>
</head>
<body>
<p>Click following link</p>
<a href=" http: //www.bpbonline.com" target="_self">Books Stores</a>
</body>
</html>
```

Scanned with OKEN Scanner

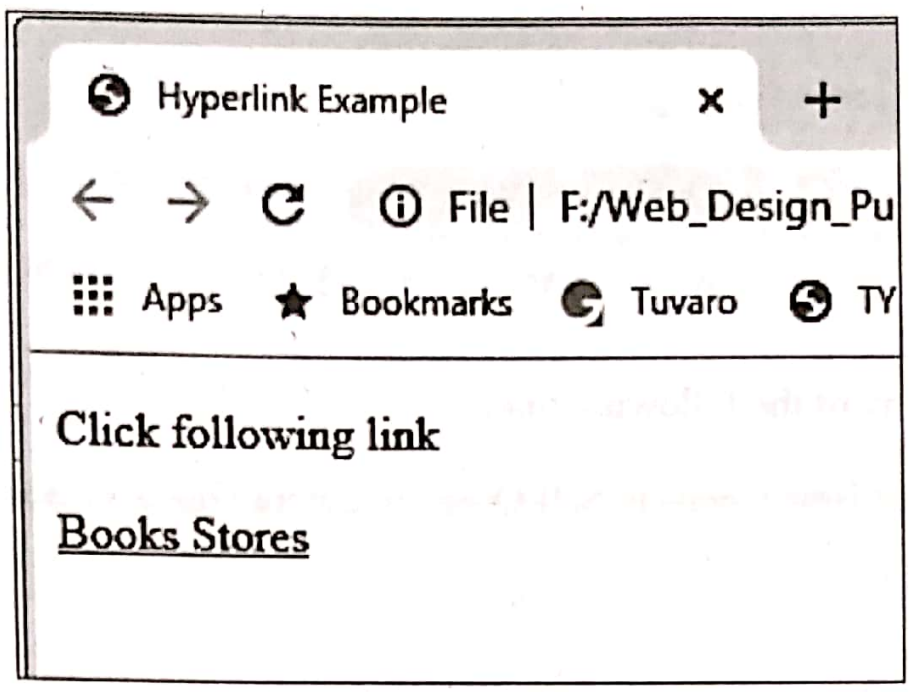


Figure 2.5: Viewing the file in Chrome of Example 5

This will produce following result, when you can click on the link generated Book Stores to reach to the home page of BPB Publications.

The Target Attribute

The target attribute specifies where to open the linked document. This attribute is used to specify the location where linked document is opened.

This example will open the linked document in a new browser Window or in a new tab. The target window options are:

Option	Description
_blank	Opens the linked document in a new Window or tab.
_self	Opens the linked document in the same frame.
_parent	Opens the linked document in the parent frame.
_top	Opens the linked document in the full body of the window.
targetframe	Opens the linked document in a named targetframe.

In example to understand basic difference in few options given for target, attribute.

Example: 6

```
<!DOCTYPE html>
<html>
<head>
<title>Hyperlink Example</title>
</head>
<body>
<p>Click any of the following links</p>
<a href="Rose Flowers.html" target=" _blank" >Opens in New</a>
<a href="Rose Flowers.html" target=" _self" >Opens in Self</a>
<a href="Rose Flowers.html" target=" _parent" >Opens in Parent</a>
<a href="Rose Flowers.html" target=" _top" >Opens in Body</a>
</body>
</html>
```

Scanned with OKEN Scanner

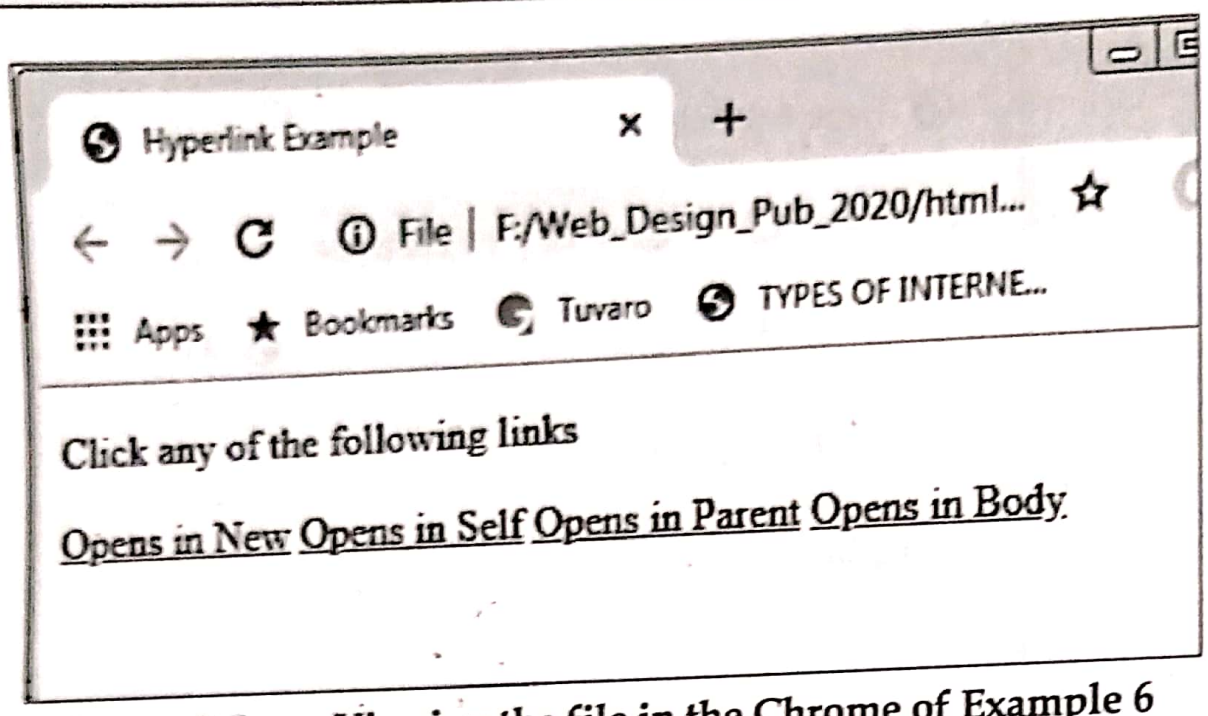


Figure 2.6: Viewing the file in the Chrome of Example 6

The output of given coding in Example 6 as shown in Figure 2.6.

Internal Hyperlink

You can create a link to a particular section of a given webpage by using **name** attribute. An internal hyperlink is a link, which points to a particular part of a page. This can be useful in long pages with lots of sub-sections. *For example*, the links at the top of this page are internal links pointing to each sub-heading.

Internal hyperlinks require an anchor tag with the name attribute like so:

```
<a name="top"> </a>
```

Create an anchor like this at the place in the page you want to link to. It does not matter if there is anything between the open and close tags.

Then create a hyperlink, which refers to this anchor with a **hash (#)** like so:

```
<a href="#top">Go To Top of Page</a>
```

This assumes that the name anchor is on the same page as the hyperlink. To link to an anchor on a separate destination page, simply create a normal link with the anchor name appended to the file name like so:

```
<a href="page1.html#part2">Go To Page 1, Part 2</a>
```

Example 7:

```
<! DOCTYPE html>
<html>
<head>
<title>Internal Link</title>
</head>
<body>
<H1>Anchors Tags</H1>
<p>To link to a specific spot on a web page, you need to use a pair of
anchors. One where you are linking from, and one where you would like to
link to. The first one, is where you are linking from. It is very similar to
the normal link tag. </P
<a name="top"></a>
<a href="#top">Go To Top of Page</a>
</body>
</html>
```

The output of the given code in Example 7 as shown in Figure 2.7.

Scanned with OKEN Scanner

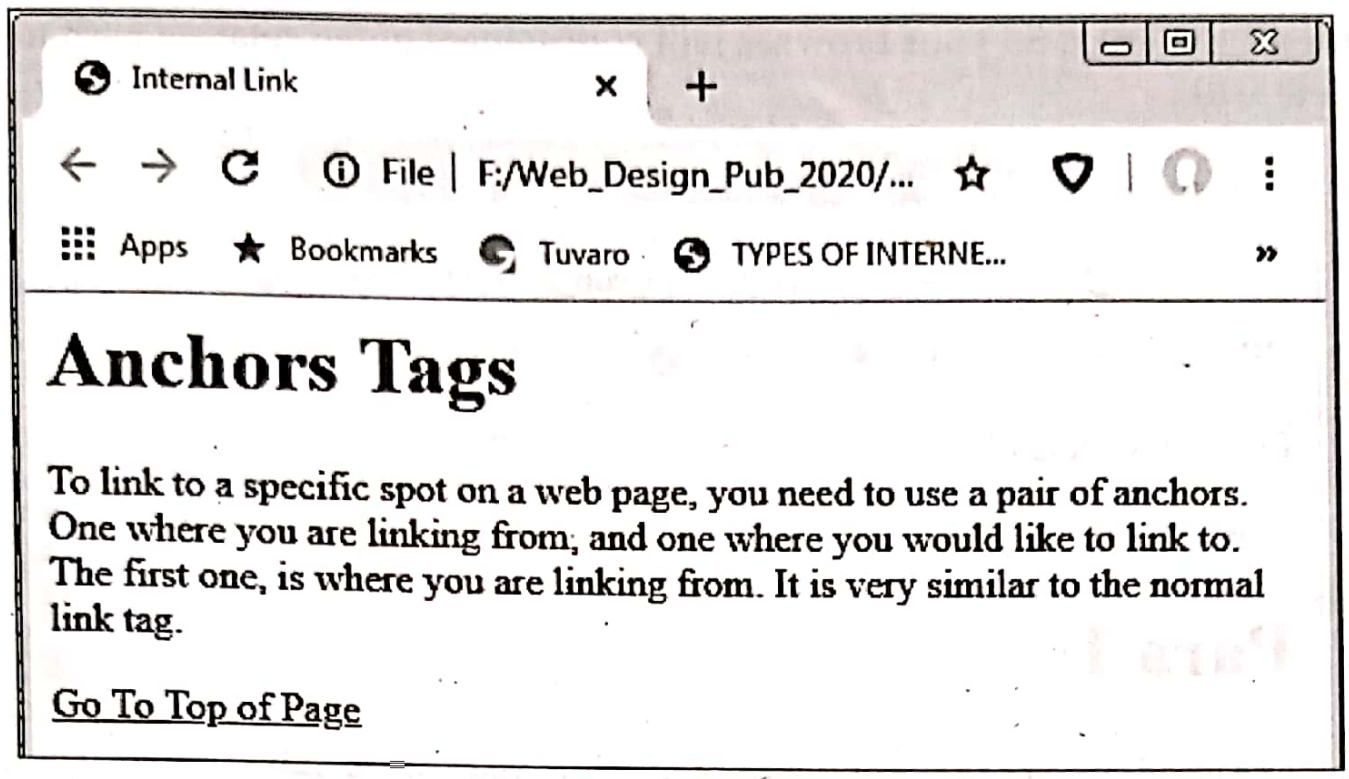


Figure 2.7: Viewing the file which is the result of Example 7

Naming Sections with id

A slightly more modern way of allowing links to be pointed at arbitrary parts of your page is to use the id attribute, which can be applied to any HTML element.

This means you do not need to keep setting up new empty elements throughout your page; you can simply add a unique **id** value to an existing element.

Use the **id** attribute to mark the element to which you want to link. For example:

```
<h1 id="heading1">heading 1</h1>
```

You can now create a link to that element by using # in the link attribute. The # must be followed by the **id** of the tag you want to link to. *For example*:

```
<a href="#heading1">Link to heading 1</a>
```

Example 8:

```
<html>
<head>
</head>
<body>
<p><a href="#Paragraph1">Link to Paragraph 1</a></P>
<p><a href="#paragraph2">Link to paragraph 2</a></p>
<h1 id="Paragraph1">Para 1</h1>
<p>Use the id attribute to mark the element to which you want to link.</p>
<h1 id="Paragraph2">Para 2</h1>
<p>Use the id attribute to mark the element to which you want to link. </p>
</body>
</html>
```

The output of the given code in Example 8 as shown in Figure 2.8.

Use of Base Path

When you link HTML documents related to the same website, it is not required to give a complete URL for every link. You can get rid of it if you use **<base>** tag in your HTML document header. This tag is used to

Scanned with OKEN Scanner

give a base path for all the links. So your browser will concatenate given relative path to this base path and will make a complete URL.

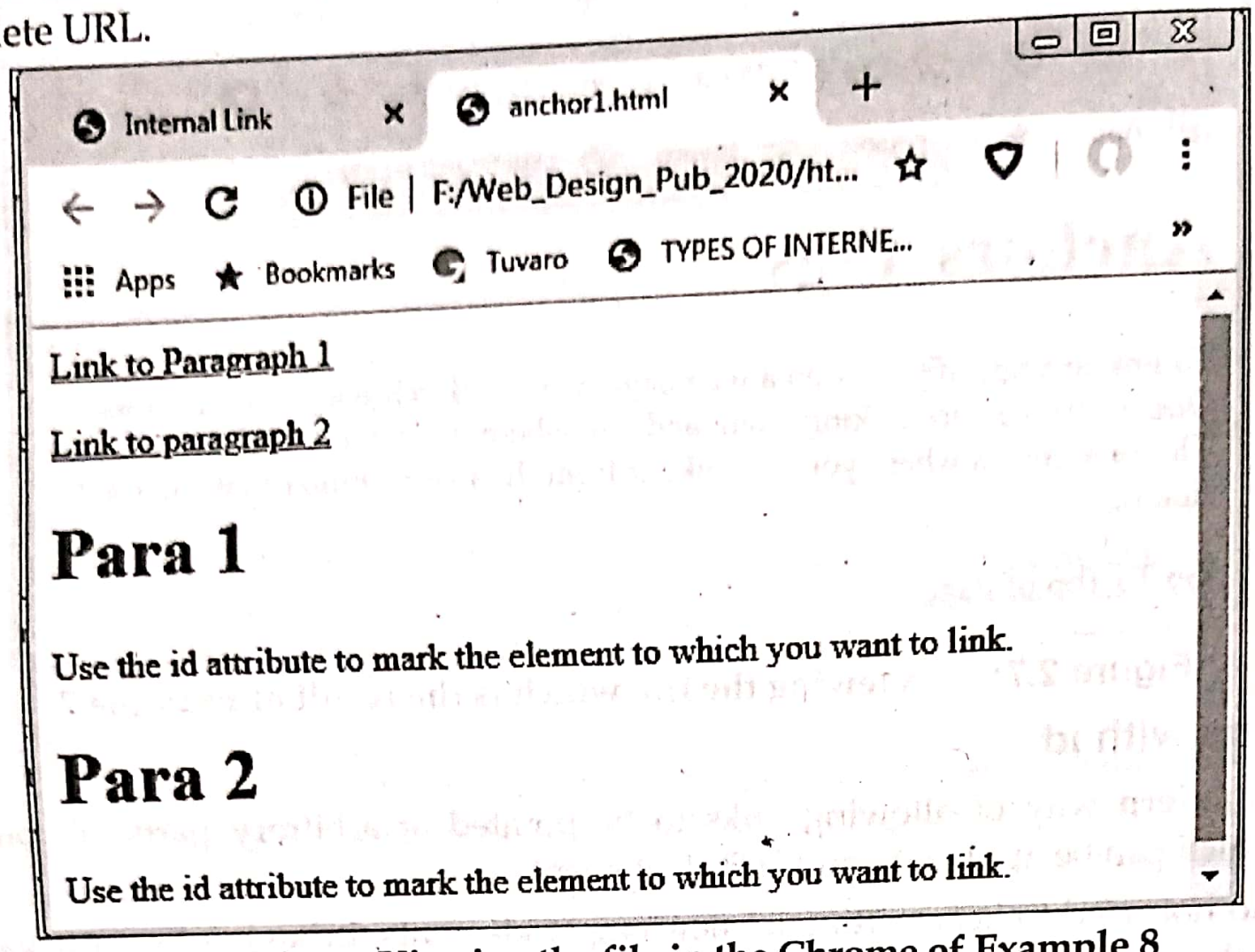


Figure 2.8: Viewing the file in the Chrome of Example 8

For *example*, makes use of **<base>** tag to specify base URL and later we can use relative path to all the links instead of giving complete URL for every link.

Example 9:

```
<! DOCTYPE html>
<html>
<head>
<title>Hyperlink Example</title>
<base href=" http: //www.onlinetutorial.com/" >
</head>
<body>
<p>Click following link</p>
<a href="/html/index.html" target=" _blank" >HTML Tutorial</a>
</body>
</html>
```

In *Example 9*, when you can click on the link generated HTML tutorial to reach to the HTML tutorial.

Now given URL **<a href="/html/index.htm">** is being considered as

<a href="http://www.onlinetutorial.com/html/index.htm".

Setting Link Colors

You can set colors of your links, active links and visited links using link, alink and vlink attributes of **<body>** tag.

To see how link, alink and vlink attributes work in Example 10.

Scanned with OKEN Scanner

Example 10:

```
<!DOCTYPE html>
<html>
<head>
<style>
a:link {
    color: green;
    background-color: transparent;
    text-decoration: none;
}
a:visited {
    color: pink;
    background-color: transparent;
    text-decoration: none;
}
a:hover {
    color: red;
    background-color: transparent;
    text-decoration: underline;
}
a:active {
    color: yellow;
    background-color: transparent;
    text-decoration: underline;
}
</style>
</head>
<body>
<p>You can change the default colors of links</p>
<a href="rose.jpg" target="_blank">HTML Images</a>
</body>
</html>
```

The output of the Example 10 as shown in Figure 2.9.

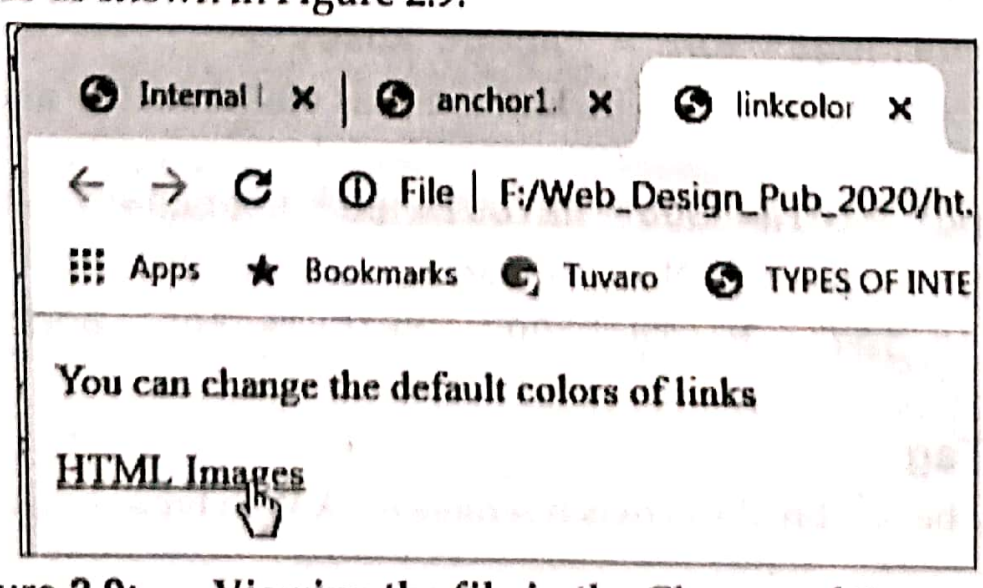


Figure 2.9: Viewing the file in the Chrome of Example 10

Scanned with OKEN Scanner

2.3.3 IMG (Image Element)

An image can be inserted into a Web page using the tag **<IMG> </IMG>**. This tag takes the name of the image file (filename.gif) as an attribute. In addition, HTML also allows control of the height, width, border, etc., of every image placed on Web page. The attributes are associated with **<IMG>** tag.

The IMG element is empty, meaning there is no closing **</IMG>**. It has several attributes:

SRC = "image_url"

SRC gives the URL of the image document. The URL points the location where the image is stored.

Aligning Image in Web Page

ALIGN tells the browser how to align the image with the text. There are different types of alignments these are:

- **Top align:** It aligns the Top of the image with the top margin of the item in the line **Align="TOP"**.
- **Bottom align:** It aligns the Bottom of the image with the baseline of the text **Align="BOTTOM"**.
- **Middle align:** It aligns the Middle of the image with the baseline of the text **Align="MIDDLE"**.
- **Left and Right align:** It supports left and right-aligned images. The image floats to the left or right margins, with the text following the image element flowing around the image, for example, **Align="Left"** and **Align="Right"**.
- **Border:** Specifies the size of the border to place around the image.
- **Width:** Specifies the width of the image in pixels.
- **Height:** Specifies the height of the image in pixels.
- **HSPACE:** Indicates the amount of space to the left and right of the image.
- **VSPACE:** Indicates the amount of space to the top and bottom of the image.
- **ALT:** Indicates the text to be displayed in case the browser unable to display the image specified in the SRC attribute.

Creating Image Links

To add an image as a link, start by adding an **image** tag. The following steps need to be followed to add an image as a link in the HTML file program.

a. Add an image tag and SRC=attribute to the document as shown below:

```
<IMG SRC="Roses.jpg">
```

b. Note that Roses.jpg is the file which will be used for linking.

c. Include an alternative text using the ALT=attribute as shown below:

```
<IMG SRC = "Roses.jpgf" ALT = "About Roses">
```

d. Add the remaining attributes you want to include, such as the HEIGHT= and WIDTH = and BORDER = attributes as follows:

```
<IMG SRC="Roses.jpg" WIDTH="300" HEIGHT="80" BORDER="2" ALT="About Roses">
```

e. Add the link anchor tag, <A> before and after the image as shown below:

```
<A><IMG SRC="Roses.jpg" WIDTH="300" HEIGHT="80" BORDER="2" ALT ="About
Roses"></A>
```

2.3.4 Paragraph <P> Tag

In the source file, there may be line breaks between sentences. A Web browser ignores the line breaks and

Scanned with OKEN Scanner

starts a new paragraph only when it encounters another **<P>** tag. The Paragraph can be aligned **left, right** or **centre** by using the **ALIGN** attribute.

Example 11:

```
<!DOCTYPE html>
<html>
<body>
<p>The image is a link. You can click on
it.</p>
<a href="Rose Flowers.html">
<img      src="Nature.jpg"      alt="HTML
tutorial"

style="width:42px;height:42px;border:1">
</a>
</body>
</html>
```

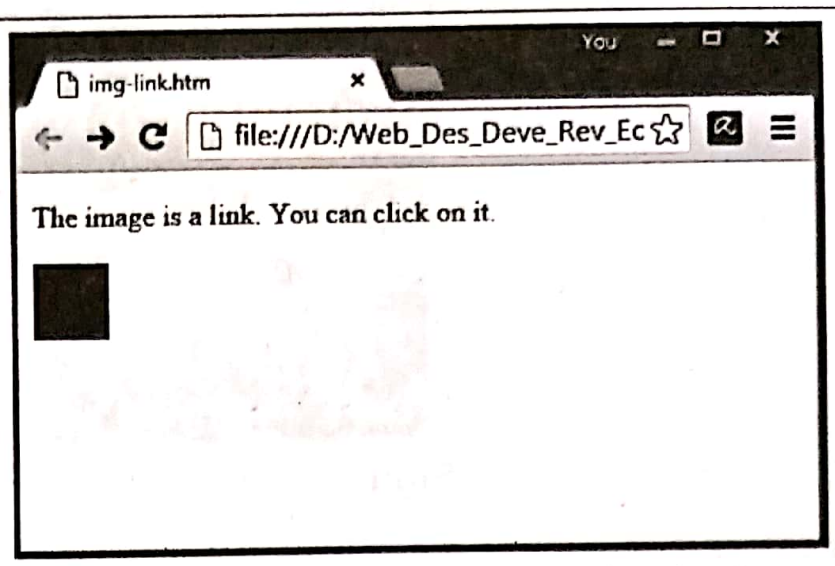


Figure 2.10: Viewing the file in the Chrome Example 11

The output of the given code in Example 11 as shown in Figure 2.10. If you click the image, the link of the image opened.

You must indicate paragraphs with ** ** elements. The browser ignores any indentations or blank lines in the source text. Without ** ** elements, the document becomes one large paragraph. The ** ** closing tag is optional. This is because browser understands that when it encounters a tag, it implies that there is an end to the paragraph.

A sample HTML document contains the **left alignment**. Here, the line Welcome to the World of HTML would be shown to the left of the page when browsed by IE.

To preserve readability in HTML files, you should put headings on separate lines. Use a blank line to identify the start of a new section and separate paragraphs with blank lines (in addition to the <P> tags). These extra spaces are needed to help you when you edit files. Note that the browser ignores the extra spaces because it has its own set of rules on spacing that do not depend on the spaces you put in your source file.

Using the **<P>** and **</P>** as a paragraph container means that you can centre a paragraph by including the ALIGN = alignment attribute in source file.

Example 12:

```
<html>
<head>
<title>a simple html example of paragraph</title>
</head>
<body bgcolor=blue text=white>
<h1>welcome to the world of html</h1>
<p>this is the first paragraph and it is left aligned</p>
<p align=right>and this is the second paragraph and it is right aligned.</p>
<p align=center>And this is the Third paragraph and it is center aligned</P>
</body>
<address>Created by Jain</address>
</html>
```

The output of the given code as shown in Figure 2.11.

Scanned with OKEN Scanner

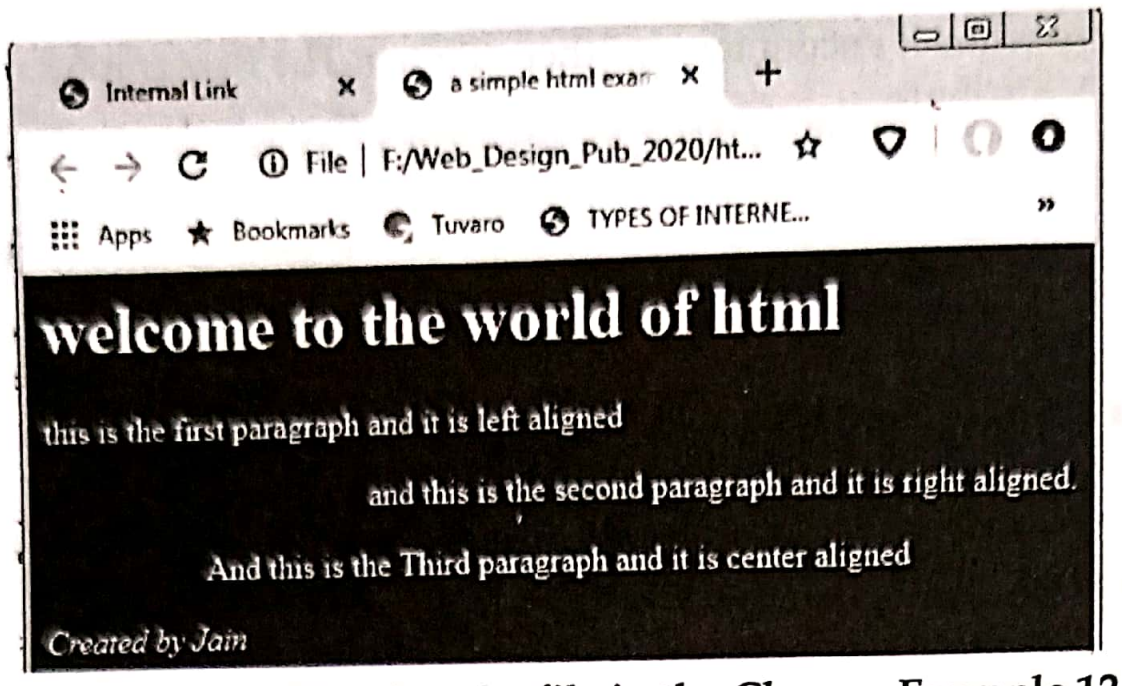


Figure 2.11: Viewing the file in the Chrome Example 12

Line Break Tag

The **
** tag has a space between the characters **br** and the forward slash. This tag is an example of an empty element, where you do not need opening and closing tags, as there is nothing to go in between them.

Horizontal Lines

The **<hr>** tag creates a line from the current position in the document to the right margin and breaks the line accordingly. *For example*, **
 breaks a line** and **<hr> inserts a horizontal rule.** These elements are called **empty elements.** Empty elements carry out their specific jobs and do not act on blocks of text.

Preserve Formatting

Sometimes, you want your text to follow the exact format of how it is written in the HTML document. In those cases, you can use the preformatted tag <pre>. Any text between the opening **<pre>** tag and the closing **</pre>** tag will preserve the formatting of the source document.

Example 13:

```
<html>
<head>
<title>An example of Line, HR Line and Pre tag</title>
</head>
<body bgcolor=blue text=white>
<h1>welcome to the world of html</h1>
<p>Hello<br />
I have received your e-mail<br />
Thanks<br />
John</p>
<p>This is first Line paragraph </p>
<hr />
<p>This is second line paragraph </p>
<p><b>Preserve Formatting</b></P><pre>
function testFunction( strText ){
   alert (strText)
}
```

Scanned with OKEN Scanner

```
</pre>
</body>
<address>Created by Jain</address>
</html>
```

The above example will display the output as shown in Figure 2.12.

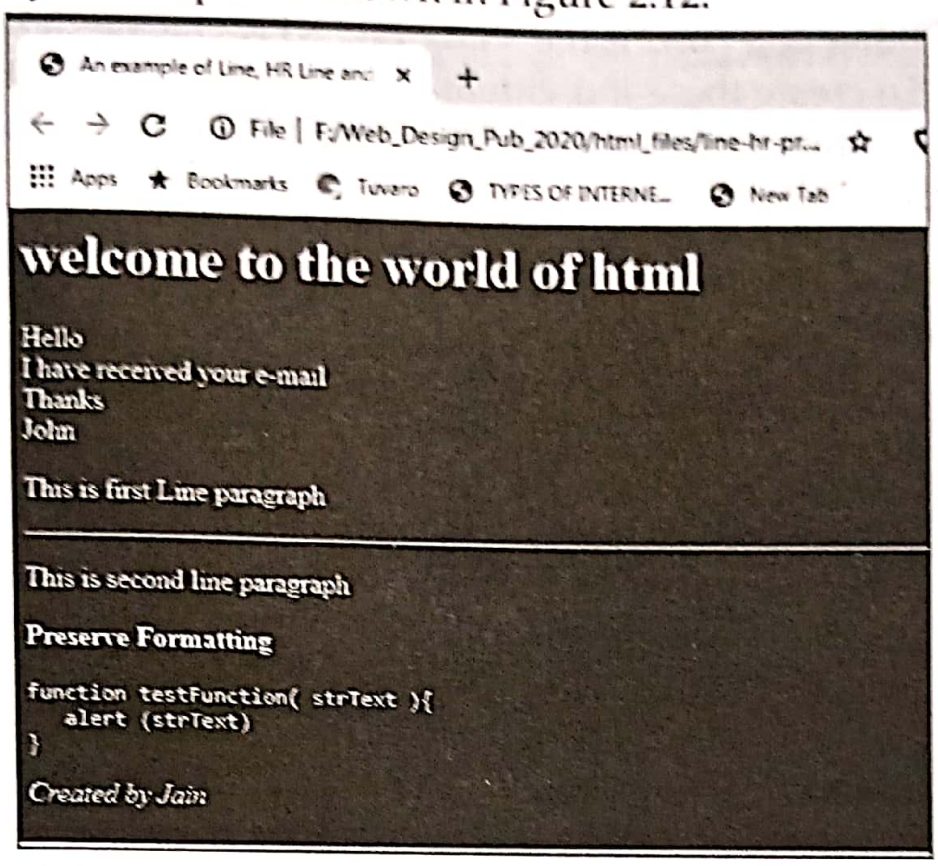


Figure 2.12: Viewing the file in the Chrome of Example 13

2.3.5 Comments in Html Document (<!>)

Whenever you type an HTML page, keep in mind that you or someone else will almost certainly need to make changes to it someday. Simple text pages are easy to read and revise, but complex Web pages with graphics, tables, and other layout tricks can be quite difficult to understand.

Those pages, which contains comment are likely to be very helpful to anyone who might need to make changes to this page in the future. Anything you type in an HTML file between the **<COMMENT>** and **</COMMENT>** tags will not appear on the actual Web page. Only when someone selects Source from View menu or edits the HTML file with a text editor will he/she will see the comments written in this page. For example,

```
<COMMENT><!- This is the Comment. -></COMMENT>
```

Every opening **<COMMENT>** tag is followed by <!- and every closing </COMMENT> tag is preceded by ->. Comments will be hidden by both past and future Web browsers. This is because previous versions of HTML used <!- and -> as comment tags, and older browsers will not recognize <COMMENT>.

2.4 HTML TABLES

HTML Tables are contained within a TABLE element. The TABLE element denotes the range of the table, and uses attribute to define properties of it. *For example*, the BORDER attribute indicates the size of the border to draw around the table and between each of the table's cells. In the tables, each box within the table is called a **cell**. There are several other attributes besides BORDER.

2.4.1 Basic Table Tags

The table element in HTML is <Table> </Table>. Everything within a table is enclosed between these two tag which are both required.

Scanned with OKEN Scanner

```
<Table>
.......
</Table>
```

There are a number of optional attributes to specify in the **<TABLE>** tag, but these are discusses after the creation of as simple table.

After the **<TABLE>**, the next thing you need is the **<TR>** tag. **<TR>** creates a **table row**, which contains one or more cells of information. To create these individual cells, you use the **<TD>** tag. **<TD>** stands for table data. You place the table information within the **<TD>** and **</TD>** tags. Within the table element, you use **<TH>** for **table headers** and the CAPTION element for the **table's caption.**

2.4.2 Creating a Table

To create a simple Table in the HTML document.

Example 14:

```
<HTML>
<HEAD>
<Title>An Example of Table Format</Title>
</head>
<body>
<table>
<caption>This is the caption of Table</caption>
<TR> <TH> </TH>
<TH>Column 1</TH> <TH>Column 2</TH>
<TH>Column 3</TH> </TR>
<TR> <TH>ROW 1</TH>
<TD>row1 column1</TD> <TD>row1 column2</TD>
<TD>row1 column3</TD> </TR>
<TR> <TH>ROW 2</TH>
<TD>row2 column1</TD><TD>row2 column2</TD>
<TD>row2 column3</TD> </TR>
<TR> <TH>ROW 3</TH>
<TD>row 1 column1</TD>
<TD>row 2 column2</TD>
<TD>row 3 column3</TD></TR>
</TABLE>
</Body>
</HTML>
```

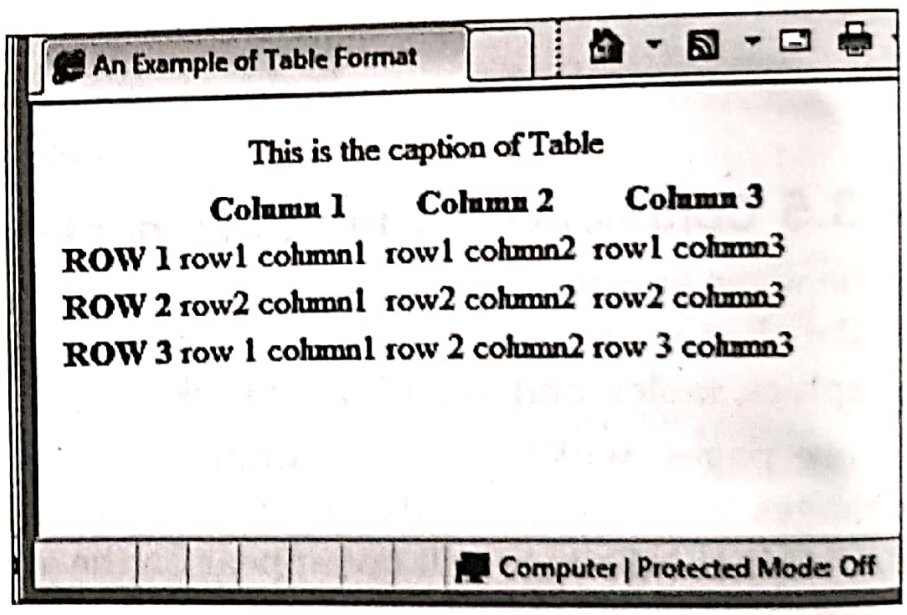


Figure 2.13: Viewing the file in the Chrome of Example 14

The above example will display the output as shown in Figure 2.13.

2.4.3 Table Attributes

Align

To align table, **ALIGN** attribute of **<TABLE>** tag may be used. This can have values **left, right** and **center** indicating that the table should be placed.

- **Left align:** Aligned at the left window border.

Scanned with OKEN Scanner

- **Right align:** Left-align table.
- **Center align:** Center-align table.
- **Right align:** Right-align table.

Example 15:

```
<Html>
<Head>
<Title>Table Alignment</Title>
</head>
<body>
<p> This is Left Alignment</p>
<Table Border="4" Align = "left">
<TR>
<TD>Data 1</TD>
<TD>Data 2</TD></TR>
<TR>
<TD>Data 1</TD>
<TD>Data 2</TD></TR>
</table>
<HR>
<p Align="Center">This is Center Alignment</p>
<Table Border="4" Align = "center">
<TR>
<TD>Data 1</TD>
<TD>Data 2</TD></TR>
<TR>
<TD>Data 1</TD>
<TD>Data 2</TD> </TR>
</table>
<HR>
<p>This is Right Alignment</p>
<Table Border="4" Align = "right">
<TR>
<TD>Data 1</TD>
<TD>Data 2</TD> </TR>
<TR>
<TD>Data 1</TD>
<TD>Data 2</TD>
</TR>
</table>
</Body>
</Html>
```

The output of Example 15 as shown in Figure 2.14.

Scanned with OKEN Scanner

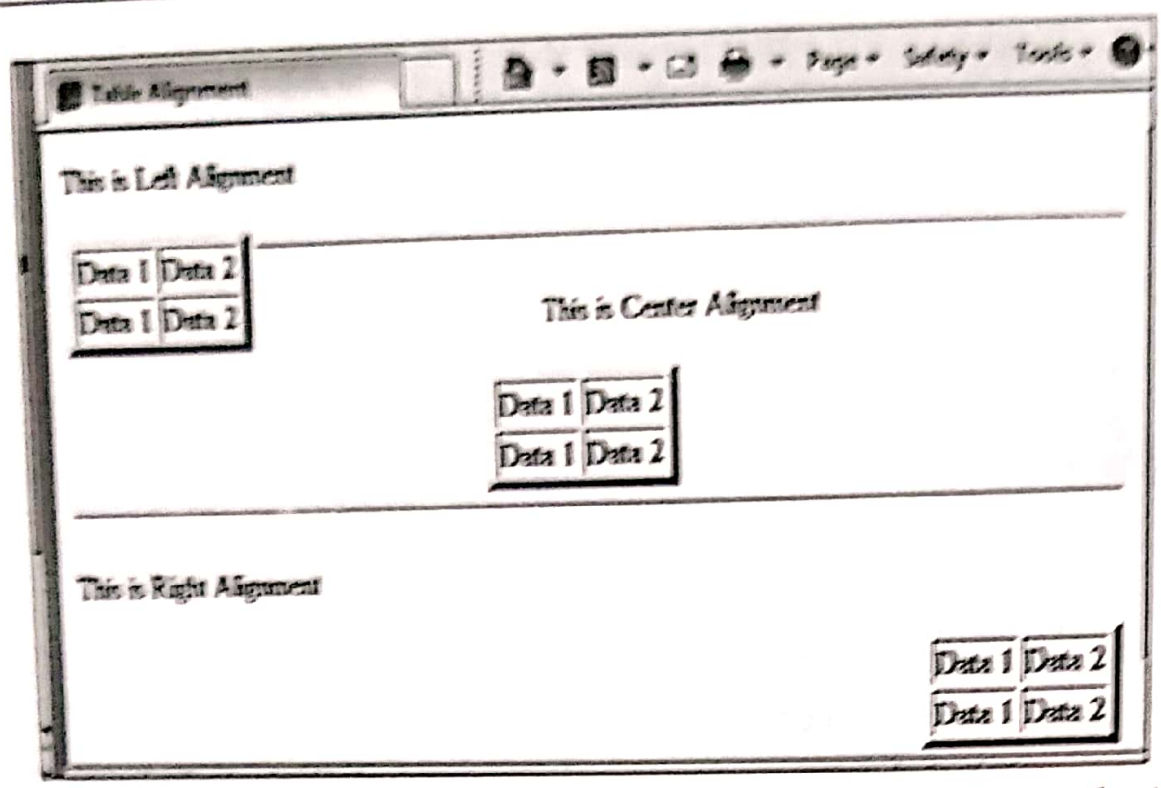


Figure 2.14: Viewing the file in the Chrome of Example 15

Table Border

BORDER is an attribute of the table element, which specifies how wide a table border shall be. BORDER="0" specifies no border at all. The default value for the BORDER attribute is 1. If you do not specify the value, the border will be one pixel wide. The attribute of BORDER="value".

Example 16:

```
<Html>
<Head>
<Title>Add Border to the table</Title>
</head>
<body> <table BORDER="4">
<TR> <TD>Data 1</TD>
<TD>Data 2</TD>
<TD>Data 3</TD>
<TD>Data 4</TD>
</TR>
<TR>
<TD>Data 1</TD>
<TD>Data 2</TD>
<TD>Data 3</TD>
<TD>Data 4</TD>
<TR>
</table>
</body>
</Html>
```

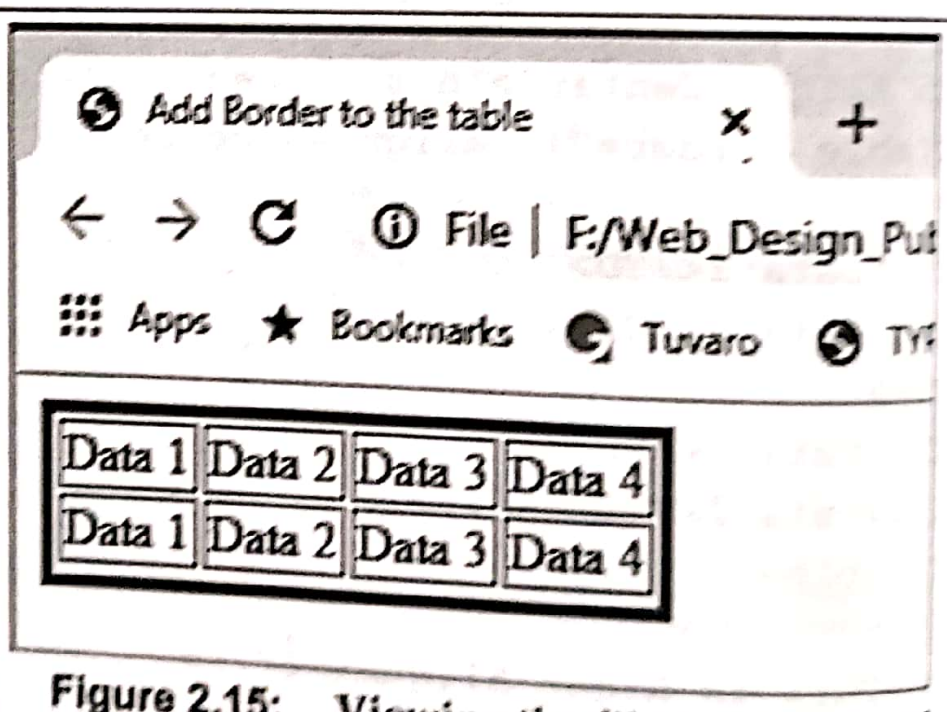


Figure 2.15: Viewing the file in Chrome of Example 16

The output of Example 16 as shown in Figure 2.15.

Table's Border Color

BORDERCOLOR attribute is used to specify the color of table's border. The attribute of BORDERCOLOR is BORDERCOLOR="colorname".

Example 17:

```
<Html>
<Head>
<Title>Add Border and Border color to the table</Title>
</head>
<body>
<table BORDER="3" BORDERCOLOR="Brown">
<TR>
<TD>Data 1</TD> <TD>Data 2</TD> <TD>Data 3</TD> <TD>Data 4</TD> </TR>
<TR>
<TD>Data 1</TD> <TD>Data 2</TD> <TD>Data 3</TD> <TD>Data 4</TD> </TR> </table>
</body>
</Html>
```

The output of Example 17 as shown in Figure 2.16.

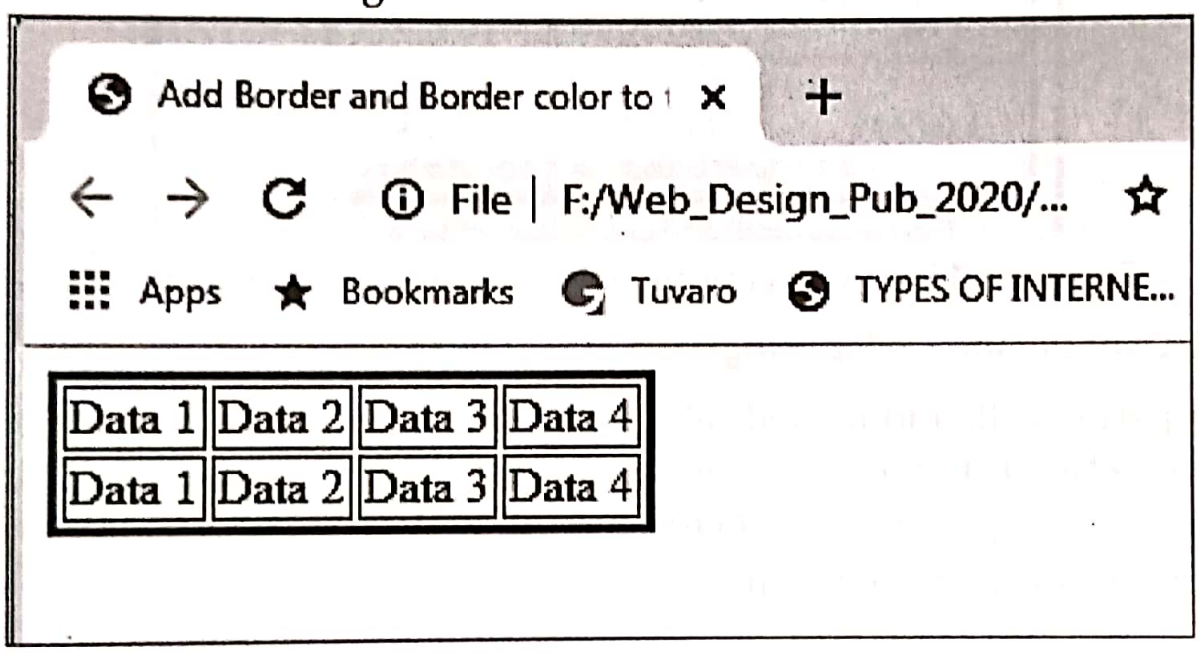


Figure 2.16: Viewing the file in Chrome of Example 17

Cellspacing and Cellpadding

To control the spacing in cells, use CELLSPACING AND CELLPADDING attributes. Cellpadding refers, to the inner margins of free space that each cell of a table has as its padded border. On the other hand, cellspacing refers, to how space is left between each cell wall in a table.

Example 18:

```
<!DOCTYPE html>
<Html>
<Head>
<Title>Table using border and cellpadding</Title>
</head>
<body>
<table BORDER="3" cellpadding="20">
<caption>Good Food Timings</caption>
<TR>
<TH>Break Fast</TH>
<TH>Lunch </TH> <TH>Dinner</TH> </TR>
```

(Contd...)

Scanned with OKEN Scanner

```
<TR>
<TD>8.00 AM</TD> <TD>1.00 PM</TD>
<TD>8.00 PM</TD> </TR>
</table>
</body>
</html>
```

The output of Example 18 as shown in Figure 2.17.

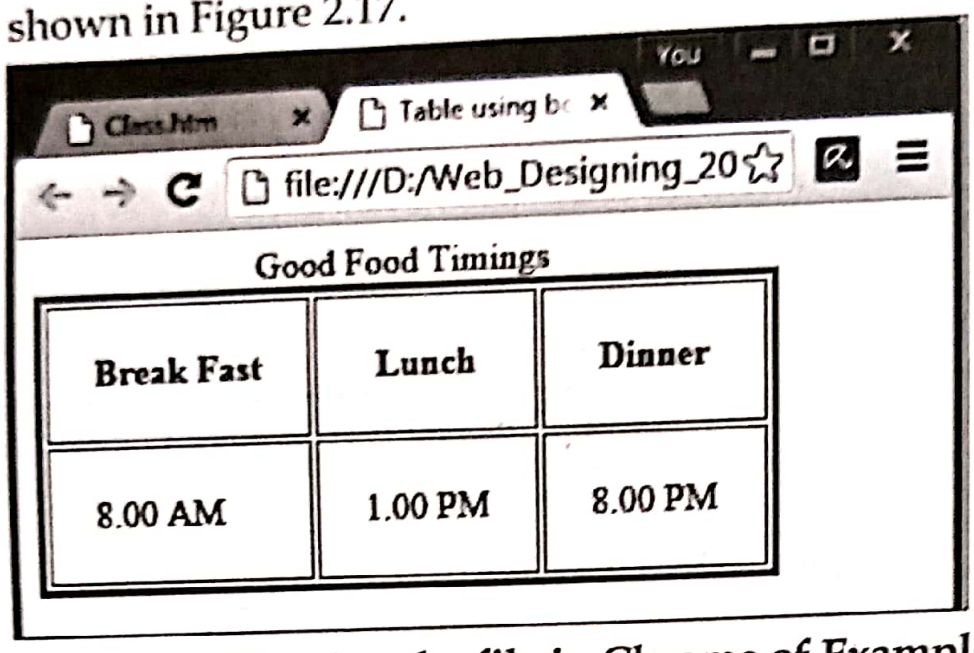


Figure 2.17: Viewing the file in Chrome of Example 18

An HTML Table with a Caption and Cellspacing

Table contains two properties cellpadding and cellspacing. Padding is always space inside the cell while the cellspacing is the space between two cells. Table caption must place above the tables. This is because we generally read tables from the top down, and therefore want to see the caption at the top.

An HTML Table with a Caption and Cellspacing.

Example 19:

```
<Html>
<Head>
<Title>Table using border and cellpadding</Title>
</head>
<body>
<table BORDER="3" cellpadding="20" cellspacing="20">
<caption>Good Food Timings</caption>
<TR>
<TH>Break Fast</TH> <TH>Lunch </TH> <TH>Dinner</TH> </TR>
<TR>
<TD>8.00 AM</TD> <TD>1.00 PM</TD> <TD>8.00 PM</TD> </TR>
</table>
</body>
</html>
```

The output of Example 19 as shown in Figure 2.18.

Table Height and Width

You can also set the table's height and width in terms of **pixels** and **percentage** of browser Window. For instance, to set the width to 50% and height to 40% of the margins.

Scanned with OKEN Scanner

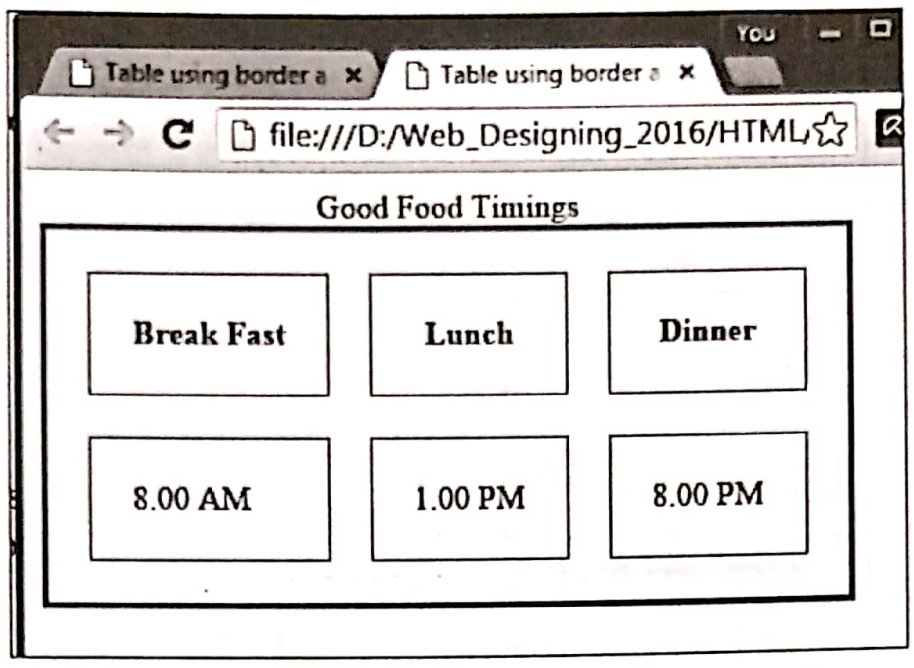


Figure 2.18: Viewing the file in the Chrome of Example 19

Example 20:

```
<Html>
<Head>
<Title>To set the height and width of a table</Title>
</head>
<body>
<Table BORDER="3" Height=120, Width=150>
<TR>
<TD>Data 1</TD>
<TD>Data 2</TD> </TR>
<TR>
<TD>Data 1</TD>
<TD>Data 2</TD> </TR></Table>
<Table BORDER="3" Height=50%, Width=50%>
<TR>
<TD>Data 1</TD>
<TD>Data 2</TD> </TR>
<TR>
<TD>Data 1</TD>
<TD>Data 2</TD> </TR></table>
</body>
</Html>
```

The output of Example 20 as shown in Figure 2.19.

Vertical Alignment of cell Contents

VALIGN-"Top" | "Center" | "Bottom"

It positions the data in each cell of a row at a specific location vertically within the cell.

The syntax of **<td valign="value">**

- **Top Align:** Top-align content.
- **Center Align:** Center-align center.
- **Bottom Align:** Bottom-align center.

Scanned with OKEN Scanner

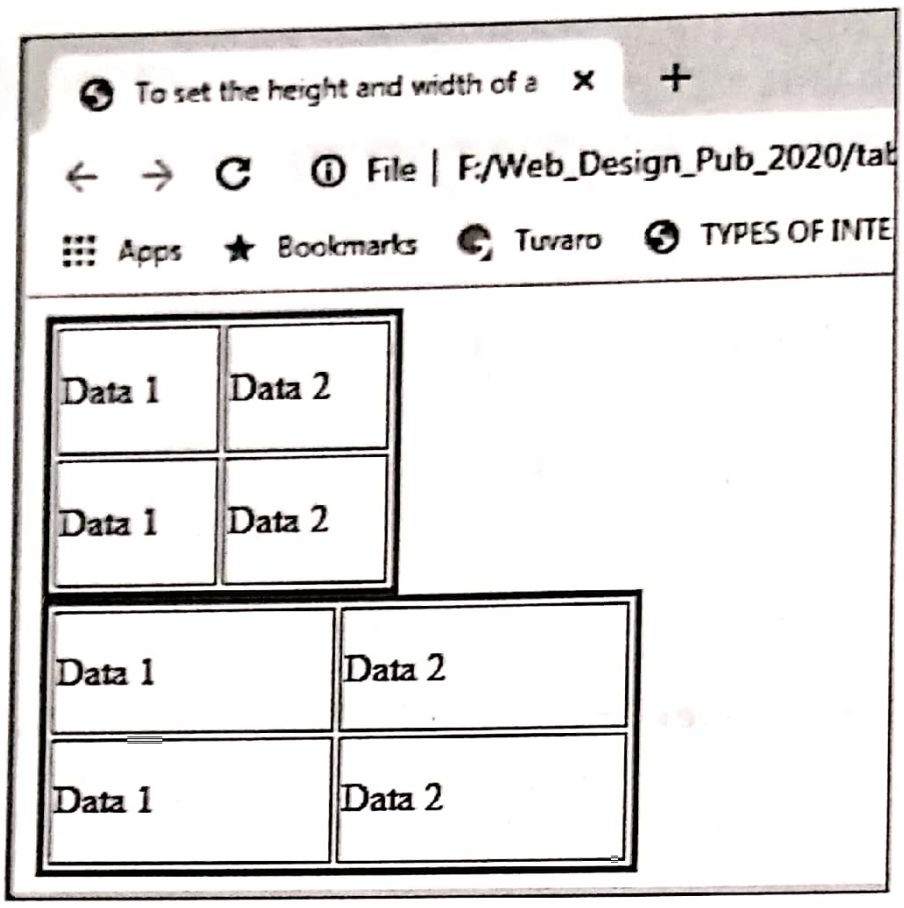


Figure 2.19: Viewing the file in the Chrome of Example 20

Example 21:

```
<Html>
<Head>
<Title>To set the VALIGN of a table</Title>
</head>
<body>
<Table BORDER="3" height="100%">
<TR>
<TH>Heading 1</TH>
<TH>Heading 2</TH>
<TH>Heading 3</TH> </TR>
<TR>
<TDValign="top">Data 1</TD>
<TDValign="bottom">Data 2</TD>
<TDValign="center">Data 3</TD> </TR>
<TR>
<TDValign="center">Data 1</TD>
<TD>Data 2</TD>
<TD>Data 3</TD> </TR>
<TR>
<TDValign="bottom">Data 1</TD>
<TDValign="bottom">Data 2</TD>
<TDValign="bottom">Data 3</TD> </TR>
</table>
</body>
</Html>
```

The output of the Example 21 as shown in Figure 2.20.

Scanned with OKEN Scanner

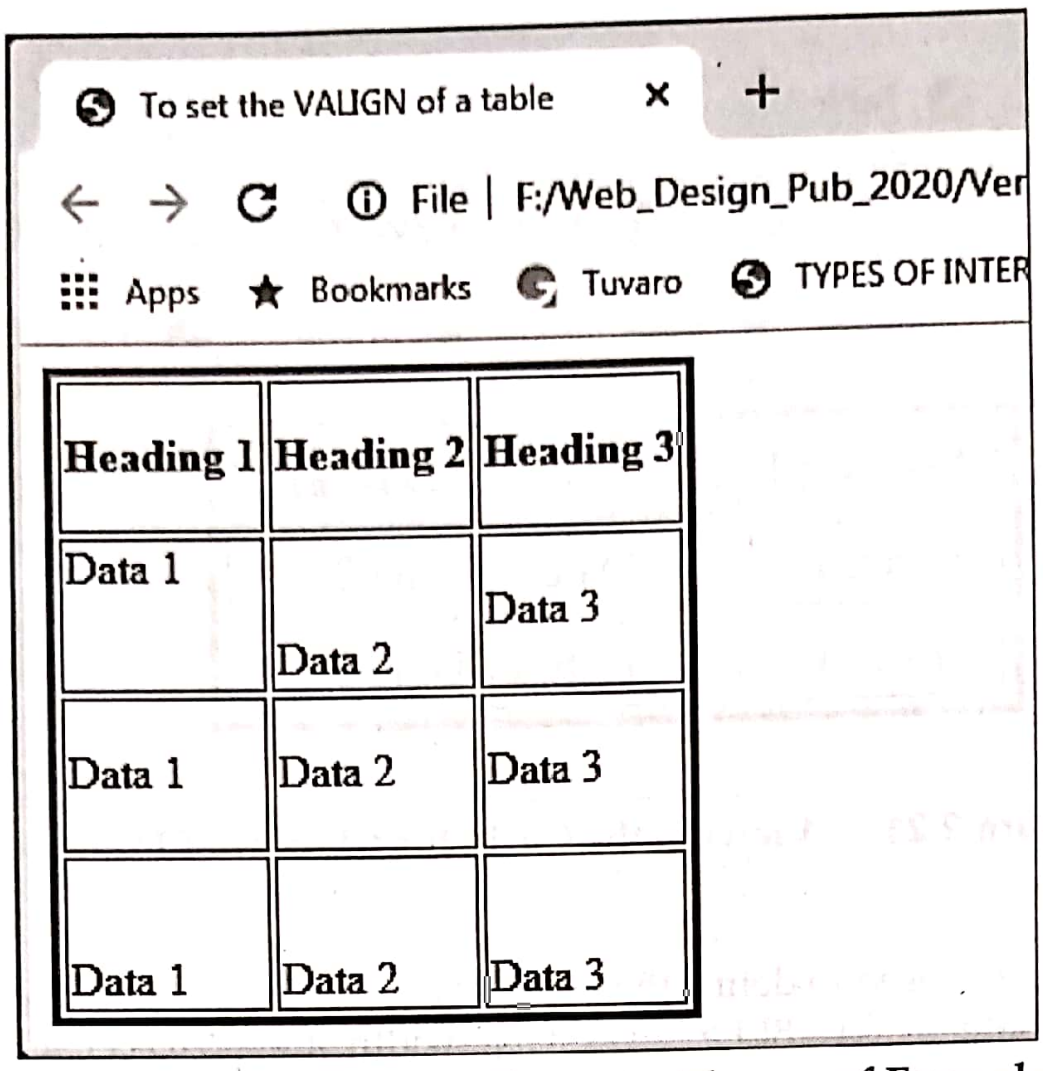


Figure 2.20: Viewing the file in the Chrome of Example 21

TD and TH Attributes

- **Align:** It is used to align the data within the cell horizontally, according values of left, right and center.
- **Left:** Align content to the left of the cell.
- **Center:** Align content to the center of the cell.
- **Right:** Align content to the Right of the cell.

Example 22:

```
<Html>
<Head>
<Title>Table Data Alignment</Title>
</head>
<body>
<table BORDER="3" cellspacing="3" cellpadding="3">
<TR>
<TH>Heading 1</TH> <TH>Heading 2</TH> <TH>Heading 3</TH> </TR>
<TR>
<TD Align="center">Data 1</TD> <TD Align="right">Data 2</TD> <TD>Data 3</TD>
</TR>
<TR>
<TD Align="center">Data 1</TD> <TD Align="right">Data 2</TD> <TD>Data 3</TD>
</TR>
</table>
</body>
</html>
```

The output of the Example 22 as shown in Figure 2.21.

Scanned with OKEN Scanner

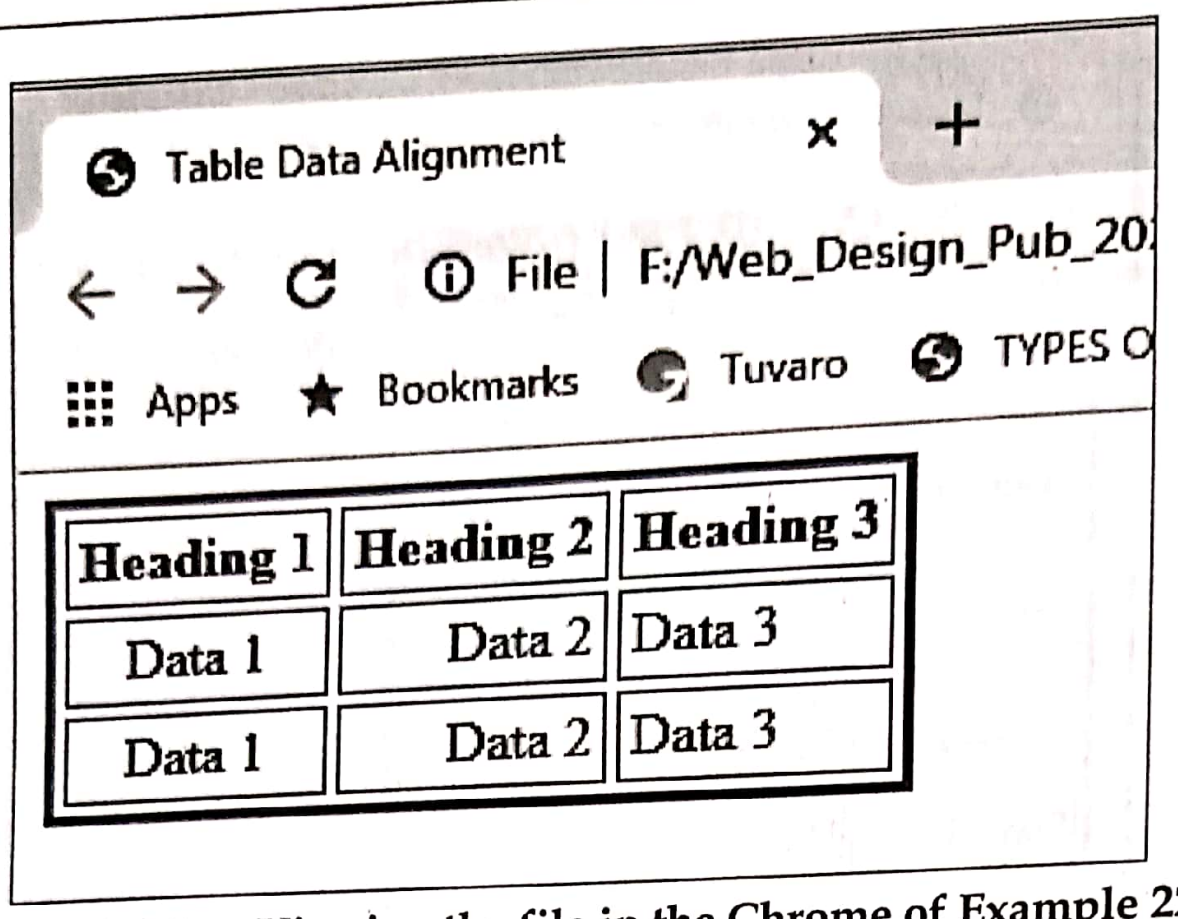


Figure 2.21: Viewing the file in the Chrome of Example 22

Setting the Cell Width

The Width attribute of <TD> is used to define the width of a cell either in **pixels** or **percentage** value of table width. The WIDTH attribute of <TABLE> gives table-width and WIDTH attribute of <TD> specifies individual cell width.

Example 23:

```
<Html>
<Head>
<Title>Setting the Cell Width</Title>
</head>
<body>
<table BORDER="3" width=”500”>
<TR>
<TH>Heading 1</TH> <TH>Heading 2</TH> <TH>Heading 3</TH> </TR>
<TR>
<TD Align="center">Data 1</TD> <TD Align="right">Data 2</TD> <TD>Data 3</TD>
</TR>
<TR>
<TD Align="center">Data 1</TD> <TD Align="right">Data 2</TD> <TD>Data 3</TD>
</TR>
</table>
</body>
</Html>
```

The output of Example 23 as shown in Figure 2.22.

Changing the Cell Span ROWSPAN and COLSPAN Attributes

The rowspan and colspan attributes are used to specify the span of cell. COLSAPN is an attribute of the **<TABLE>** tab but is also an attribute of the **<TH>** and the **<TD>** tags where you have used it more often

Scanned with OKEN Scanner

than with the **<TABLE>** tag. The following example displays COLSPAN and background color for individual cells in table.

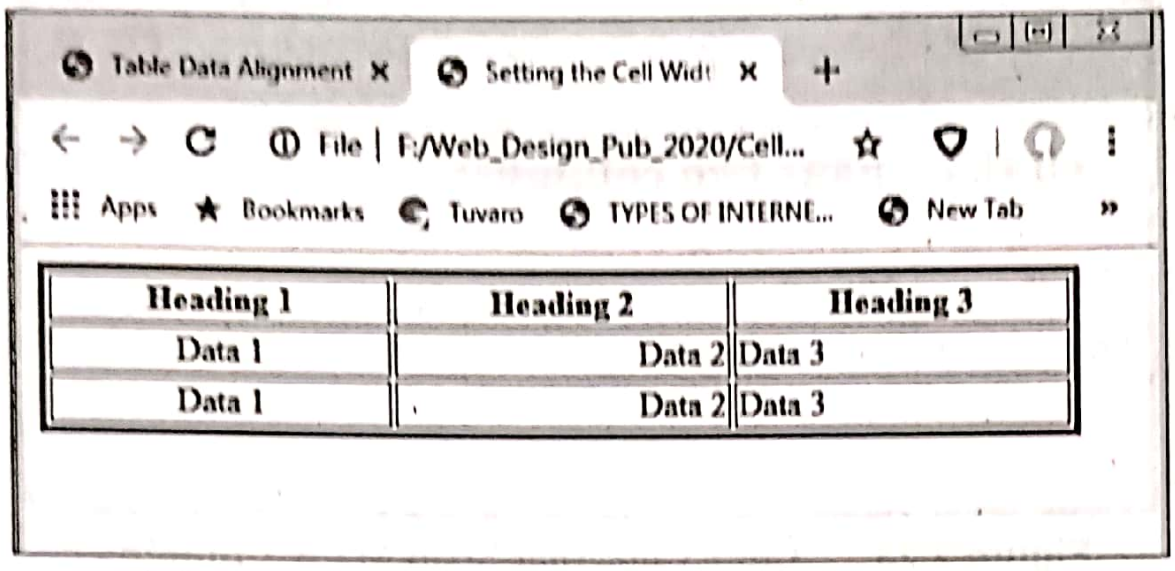


Figure 2.22: Viewing the file in the Chrome

Table Cells that Span Many Columns.

Example 24:

```
<!DOCTYPE html>
<html>
<head>
<style>
table, th, td {
    border: 1px solid black;
    border-collapse: collapse;
}
th, td {
    padding: 5px;
    text-align: left;
}
</style>
</head>
<body>
<h2>Cell that spans two columns:</h2>
<table style="width:100%">
  <tr>
    <th>Name</th>
    <th colspan="2">Address</th>   </tr>
  <tr>
    <td>Ahsok</td>
   <td>Dwarka Phase I</td>
   <td>Rohini Sec 10</td>  </tr>
</table>
</body>
</html>
```

The output of the Example 24 as shown in Figure 2.23.

Scanned with OKEN Scanner

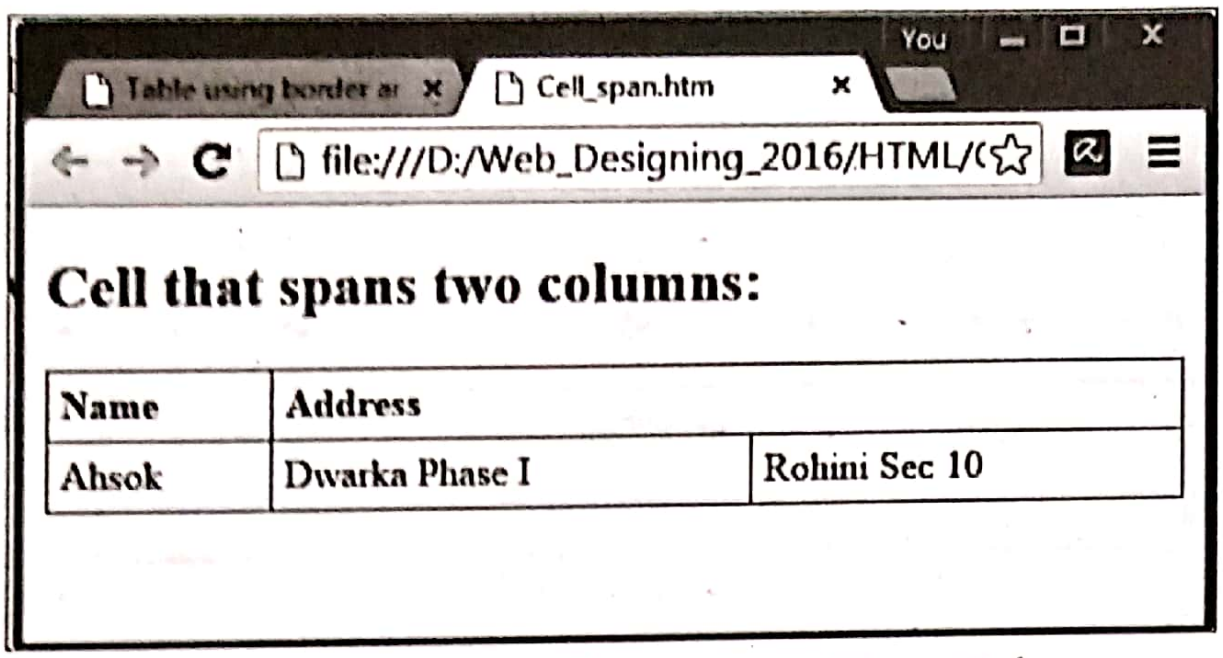


Figure 2.23: Table Cells that Span Many Columns

Table Cells that Span Many Rows.

Example 25:

```
<!DOCTYPE html>
<html>
<head>
<style>
(Contd…)
table, th, td {
    border: 1px solid black;
    border-collapse: collapse;
}
th, td {
    padding: 5px;
    text-align: left;
}
</style>
</head>
<body>
<h2>Cell that spans two Rows:</h2>
<table style="width:100%">
  <tr>
    <th>Name</th> <td>Ashok</td> </tr>
  <tr>
    <th rowspan="2">Address</th> <td>Dwarka Phase I</td> </tr>
<tr>
    <td>Rohini Sec 10</td>  </tr>
</table>
</body>
</html>
```

The output of the Example 25 as shown in Figure 2.24.

Scanned with OKEN Scanner

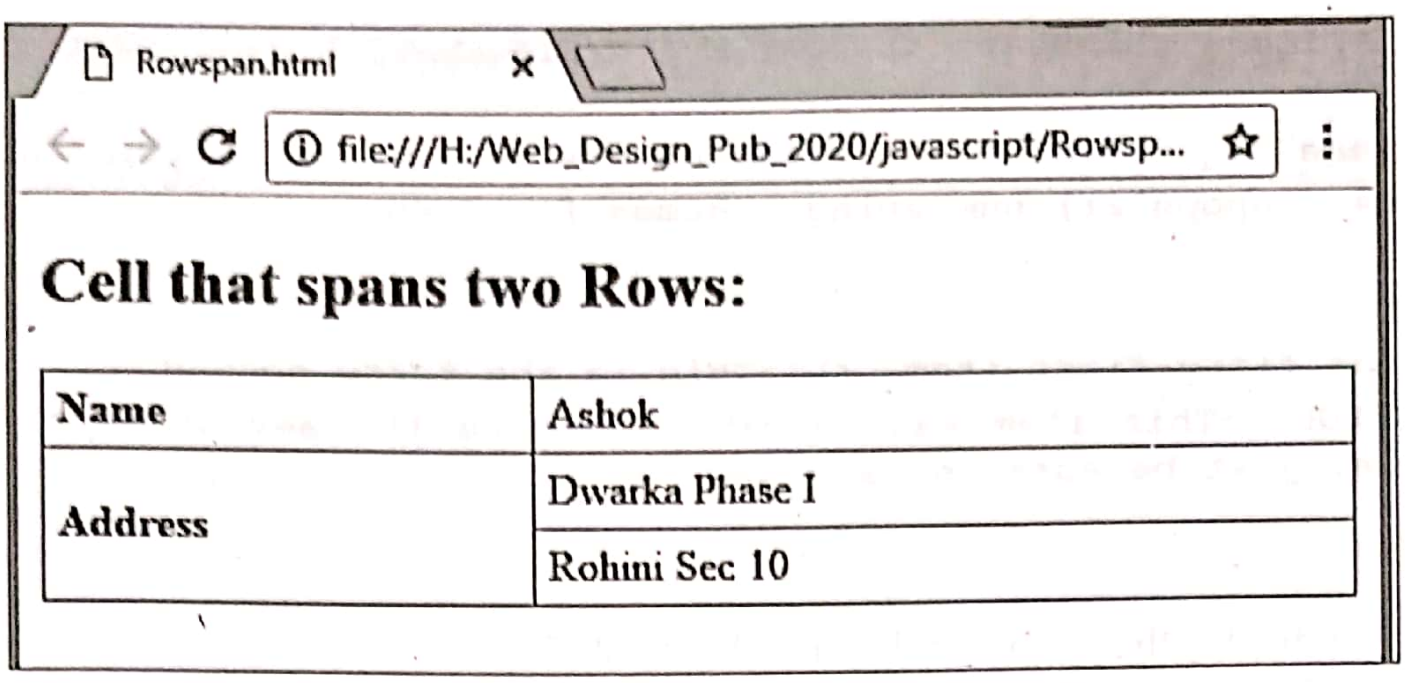


Figure 2.24: Table Cells that Span Many Rows

2.5 HTML Lists

A list is a record of information, such as names of students, usually written on each line and ordered in a way that makes a particular thing easy to find.

There are three different types of HTML lists:

- **Ordered List or Numbered List (ol):** Used to create a list of related items, in a specific order.
- **Unordered List or Bulleted List (ul):** Used to create a list of related items, in no particular order.
- **Description List or Definition List (dl):** Used to create a list of related items, in a specific order.

2.5.1 Ordered List <OI>

Ordered List is a list with numbers in an order **ascending** or **descending**. In this, each item (paragraph) is prefaced with a number. The tags used for this are <OL> </OL>. The browser applies the appropriate number to each line when it opens the page.

Ordered lists have additional attributes that you can use to specify the first number of the list, as well as to create hierarchical information. First, you can start a numbered list with a value other than 1 (or **A, a, I** or **i**).

- A: Specifies a sequence of upppercase letters.
- a: Specifies a sequence of lowercase letters.
- I: Specifies a sequence of uppercase Roman numerals.
- i: Specifies a sequence of lowercase Roman numerals.

Simply include the **START = attribute** in the initial **<OL>** tag, as in **<OL START = 51>**. Or, you can even change specific numbers within a list by using the **VALUE = attribute** in the **<L1>** tag, as in **<L1 VALUE = 7>**. To use these attributes, include them in the **<OL>** tag.

Example 26:

```
<html>
<head>
<TITLE> A Simple HTML Example</TITLE>
</head>
<body BGCOLOR = YELLOW>
<H1 ALIGN=LEFT>Sample Lists</H1>
<OL>
```

Scanned with OKEN Scanner

```
<LI>Lions <LI>Tigers <LI>Bears <LI>Oh, My! </OL>
<OL TYPE=A>
<LI>Outlines use sequential lists with letters.
<LI>So do some (unpopular) numbering schemes for documentation.
</OL>
<OL START=51>
<LI>This is the fifty-first item. <LI>This is the fifty second.
<LI TYPE=i VALUE=7>This item was renumbered to be the seventh, using lowercase
roman numerals, just because we can. </OL>
</body>
</html>
```

The Example 26 will display the output as shown in Figure 2.25.

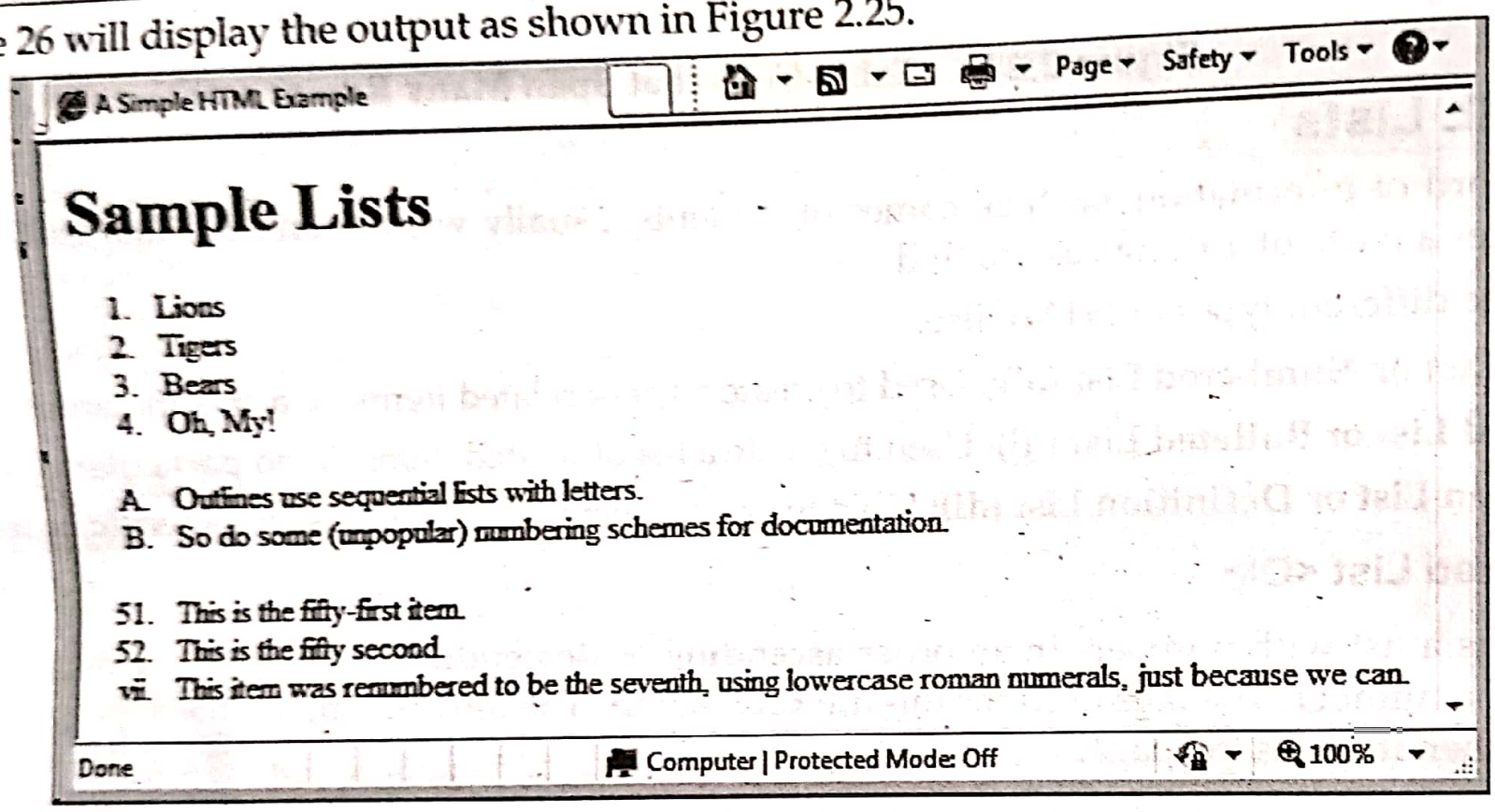


Figure 2.25: Viewing a file in the chrome of Example 26

2.5.2 Unordered List <UL>

This is an unordered list. In this, each item (paragraph) is prefaced with a bullet. The tags used for this are <UL> and </UL>.

The basic bulleted list has a default progression of bullet types that changes as you move through indented levels. The TYPE attribute can be used in the <UL> element so that no matter what the indent level the bullet type can be specified thus:

- TYPE = disc
- TYPE = circle
- TYPE = square

Example 27:

```
<!DOCTYPE html>
<head>
<TITLE> A Simple HTML Example</TITLE>
</head>
<body BGCOLOR = YELLOW>
```

Scanned with OKEN Scanner

```
<H3>Unordered list with different Styles</H3>
<P><ul>
<ul style="list-style-type:disc">
<li>Section 1 </li>
<li>Section 2 </li>
<li>Section 3 </li>
</ul></P>
<p><ul>
<ul style="list-style-type:circle">
<li>Coffee</li>
<li>Tea</li>
<li>Milk</li>
</ul></p>
<p><ul>
<ul style="list-style-type:square">
<li>Orange</li>
<li>Grapes</li>
<li>Mango</li>
</ul> </p>
</body>
</html>
```

The Example 27 will display the output as shown in Figure 2.26.

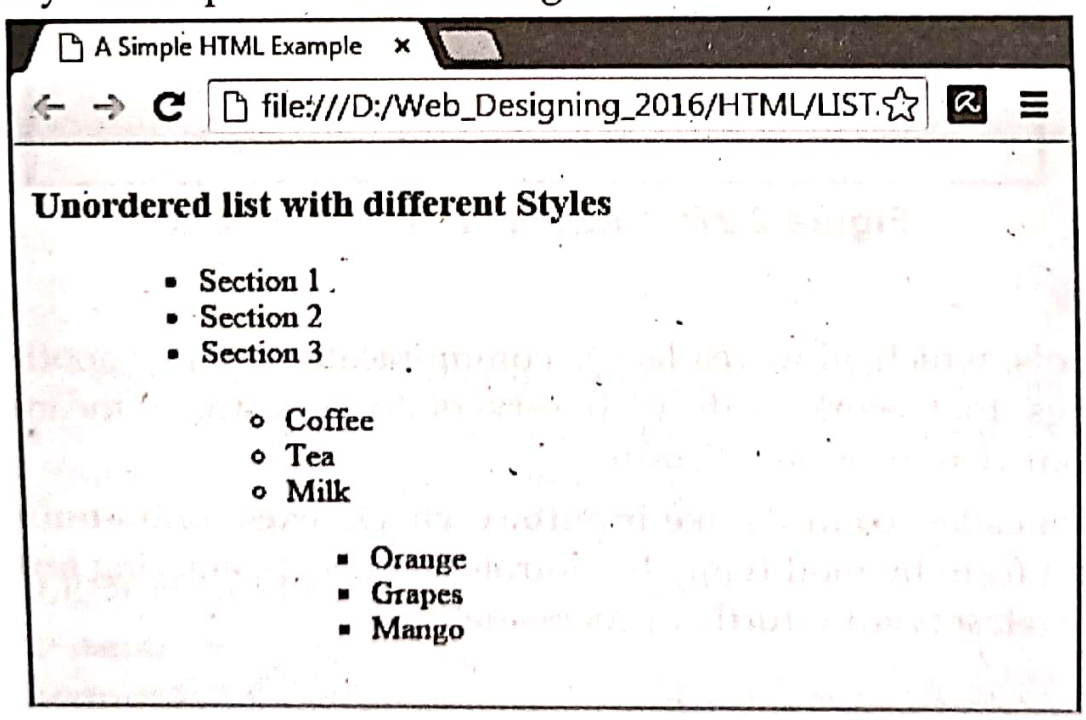


Figure 2.26: Result of Example 27

2.5.3 Definition List

The definition list is created with the **<dl>** element and usually consists of a series of terms and their definitions. Inside the **<dl>** element, you will usually see pairs **of <dt>** and **<dd>** elements.

Example 28:

```
<!DOCTYPE html>
<head>
<TITLE> A Simple HTML Example</TITLE>
```

Scanned with OKEN Scanner

```
</head>
<body BGCOLOR = YELLOW>
<H3>Definition List</H3>
<dl>
<dt>HTML</dt>
<dd>Hypertext Markup Language (HTML) document are ASCII (American Standard
Code for Information Interchánge) or plain text documents that drive the World
Wide Web.</dd>
<dt>Internet</dt>
<dd>The Internet is a combination of many of networks, and a large number of
databases and other services.</dd>
</body>
</html>
```

The Example 28 will display the output as shown in Figure 2.27.

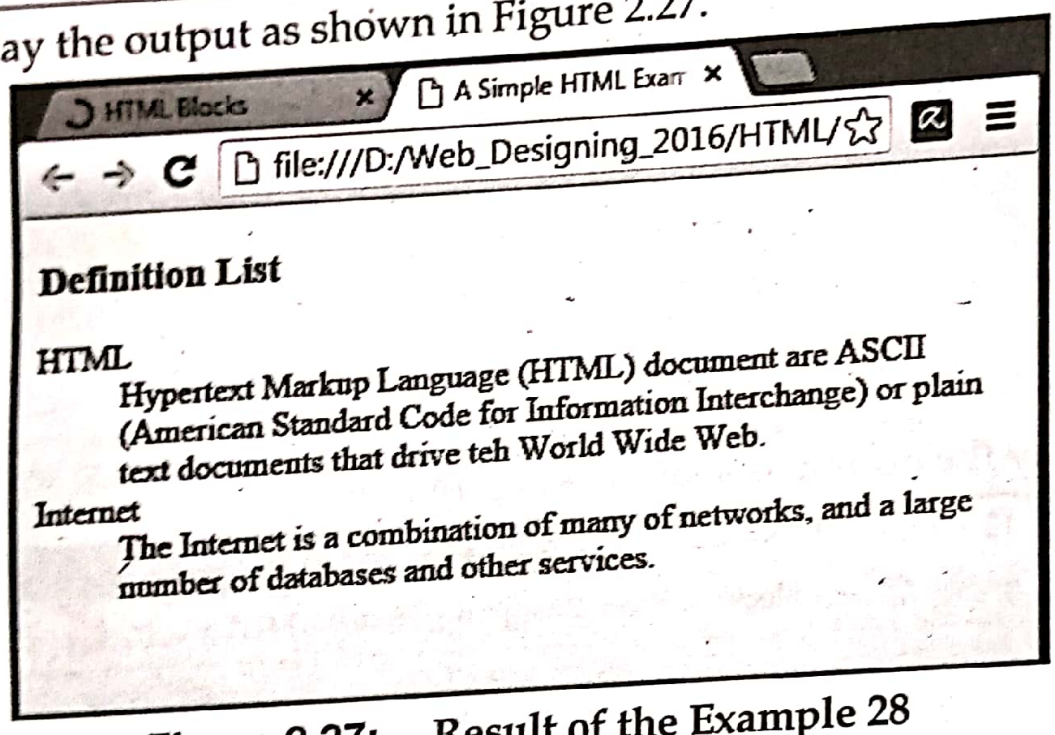


Figure 2.27: Result of the Example 28

2.6 HTML Forms

Forms are such HTML tools, which allow readers to communicate with one another via the Web. Form is a collection of markup tags that work with Web servers to produce a means of obtaining whatever information; you need from visitors of our Website.

It contain special elements called **controls** like **inputbox, checkboxes, radio-buttons, submit buttons**, etc. Users generally complete a form by modifying its controls *for example*. entering text, selecting items, etc. and submitting this form to a web server for further processing.

<FORM> Tag

A FORM tag is used in an HTML document just like any other element. It has a start tag, **<FORM>** and an end tag **</FORM>**. The form itself contains regular text, other HTML elements, such as tables and form elements, such as check boxes, pull-down menus and text fields. Forms cannot be nested, i.e., you cannot put one into another. Forms can include other elements, such as list or **<PRE>**. FORM attributes are mentioned in the following paragraphs. For example,

```
<FORM ACTION = "HTTP://WWW.GOOGLE.COM\HTML\
FORMS.HTML"METHOD = "POST">
```

Different attributes of the forms are as follows:

(a) Action attribute

Scanned with OKEN Scanner

(b) Method attribute

(c) Enctype attribute

Action Attribute: It specifies the program or script that accepts the contents of the form for processing. If this attribute is absent, the base URL (URL where the form is located on the server) of the form is automatically used.

Method Attribute: Method attribute specifies which method type will be used to submit the form data. The method we use depends on how our particular server works. The possible values are:

(a) get

(b) post

These are the HTTP methods that a browser uses to talk to a server. **GET** is the default method for browsers to submit information. HTML pages are generally retrieved by requesting a single URL from a Web server using the GET method. *For example,* when we write

```
HTTP://WWW.GOOGLE.COM\HTML\FORMS.HTML
```

The above URL is translated into a valid HTTP **GET** request like the following:

```
GET HTML/FORMS.HTML
```

The GET method is not very secure, since that data input appears in the URL. Furthermore, it is impossible to append a large amount of data with the GET method. Another problem is that it is not suitable for foreign languages other than English.

Enctype Attribute: ENCTYPE specifies the encoding for the fill-out form contents. It only applies if METHOD is set to POST. It identifies the media type. This media type is used for encoding the name/value pairs of the form's data.

2.7 Form's Elements

The Form element consists of the following parts:

(a) SELECT

(b) OPTION

(c) TEXTAREA

(d) INPUT Option

SELECT: Inside **<FORM> ... </FORM>** tag, any number of **SELECT** tags are permitted freely intermixed with other HTML elements and text. This tag allows a user to choose one of the set of alternatives. **SELECT** element is used for query. The SELECT element is specialized for creating drop-down or scrollable lists of menus only.

The basic structure of a **SELECT** element is mentioned as follows:

```
<SELECT Name="select-menu">
<OPTION> View the product?
<OPTION> Call for help?
<OPTION> Request a catalogue?
<OPTION> Exit from this form?
</SELECT>
```

The attributes to **SELECT** tags are as follows:

Name: This attribute is used to create a label for the SELECT data structure like ("select-menu") in the above example. Functionally, this is equivalent to the radio button possibility with the **INPUT** element of **Type = "radio"**, where one, and only one, of a set of options can be made.

Scanned with OKEN Scanner

Size: It determines that how many options are visible at a time. The default value of size is 1. If SIZE is 2 or more, the SELECT will be represented as a scrolled list.

Multiple/Single: Because the default value of choosing options is SINGLE, therefore, user can choose only SINGLE. In both of the above examples of **SELECT** statements, the user can make one, and only one, choice. By adding the attribute **MULTIPLE**, the user will be able to choose more than one among several options.

A select box, also called drop down box which provides option to list down various options in the form of drop down list, from where a user can select one or more options.

Here is example HTML code for a form with one drop down box.

Example 29:

```
<!DOCTYPE html>
<head>
<title>HTML Select Box</title>
</head>
<body>
<form>
<select>
  <optgroup label="Sports">
    <option value="baseball">Baseball</option>
    <option value="basketball">Basketball</option>
  </optgroup>
  <optgroup label="Shopping">
    <option value="food">Food</option>
    <option value="gift">Gift</option>
  </optgroup>
</select>
</form>
</body>
</html>
```

The Example 29 will display the output as shown in Figure 2.28.

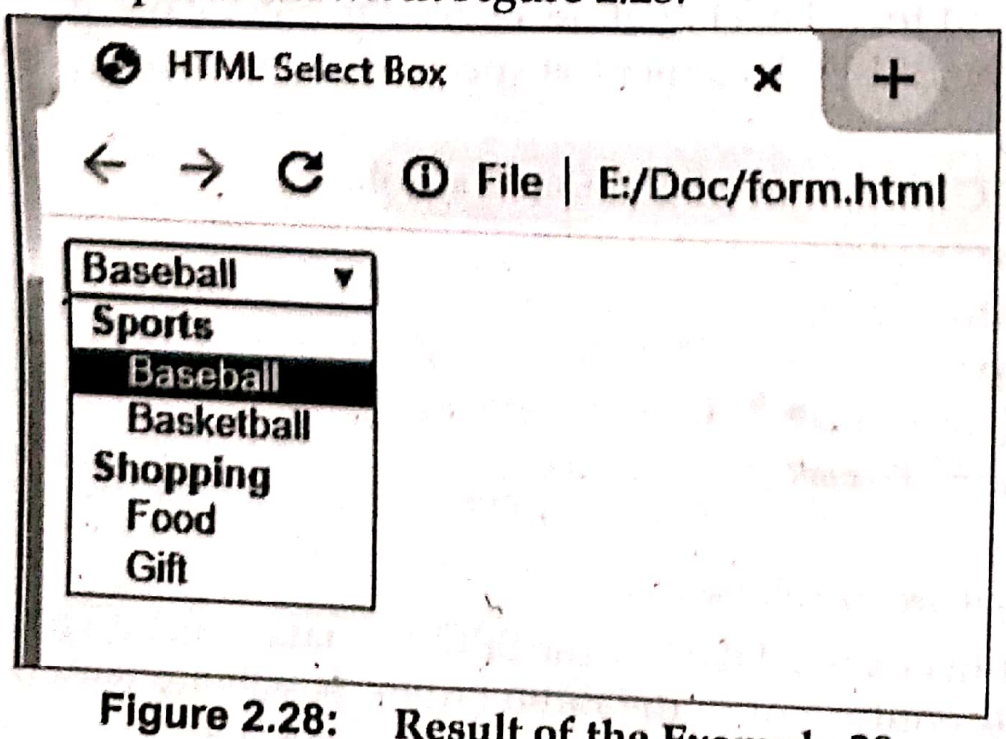


Figure 2.28: Result of the Example 29

Scanned with OKEN Scanner

OPTION: This tag occurs only within the **SELECT** element (described previously) and is used to represent each choice of the **SELECT**. The attributes of **OPTION** are:

- **Selected:** It indicates that this option initially is selected.
- **Value:** If present, this is the value returned by SELECT if this option is chosen; otherwise, the value returned is that set by the OPTION element.

The **<option>** tag comes in pairs; the closing tag (**</option>**) is required.

The **<option>** tag is generally used with the value attribute to specify which value should be sent to the server for the further processing.

Example 30:

```
<!DOCTYPE html>
<head>
<title>HTML option</title>
</head>
<body>
<form>
<div> Enter your favorite cricket player: Press any character</div>
 <input type="text" id="favCktPlayer" list="CktPlayers">
 <datalist id="CktPlayers">
 <option value="Sachin Tendulkar">
 <option value="Rahul Dravid">
 <option value="Rohit Sharma">
 <option value="Saurav Ganguly ">
 <option value="Mahendra Singh Dhoni">
 </select>
</datalist>
</form>
</body>
</html>
```

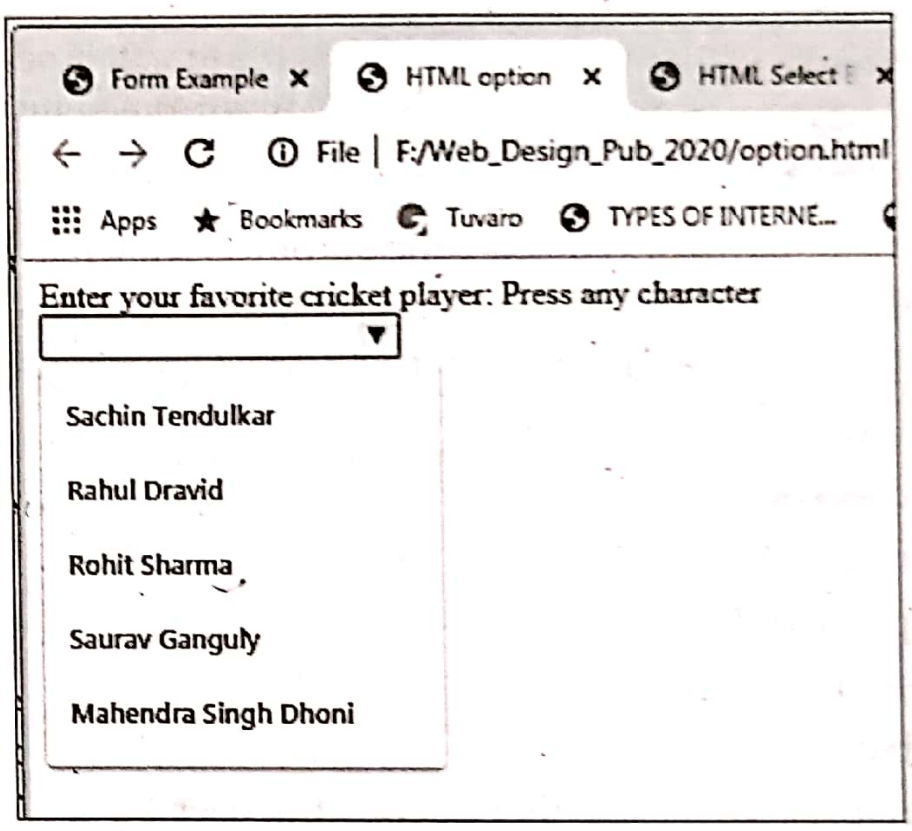


Figure 2.29: Result of the Example 30

The Example 30 will display the output as shown in Figure 2.29.

TEXTAREA: TEXTAREA element is used to ask the user to enter several lines of text (as opposed to the single line with an INPUT element of Type="text"). A sample TEXTAREA tag is as follows:

```
<TEXTAREA NAME="comments" ROWS=4 COLS=30></TEXTAREA>
```

This allows the user to enter any amount of text, with only **four rows** and **30 columns** visible. The attributes associated with **TEXTAREA** are as follows:

- **Cols:** The number of columns that will be displayed (the user can use the scrollbar to move through more columns if necessary).
- **Name:** The logical name associated with the returned text.
- **Rows:** The number of rows that will be displayed (note that the user can use more rows and scroll down to them).
- **Maxlentgh:** The total number of characters allowed in the text area.
- **Placeholder:** Specifies a short hint that describes the expected value of a text area.

Scanned with OKEN Scanner

Example 31:

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Textarea</title>
<body>
<form>
<label
for="Textarea">Textarea:</label>
<textarea      rows="4"      cols="30"
placeholder="Type text here...">
</textarea>
</form>
</head>
</body>
</html>
```

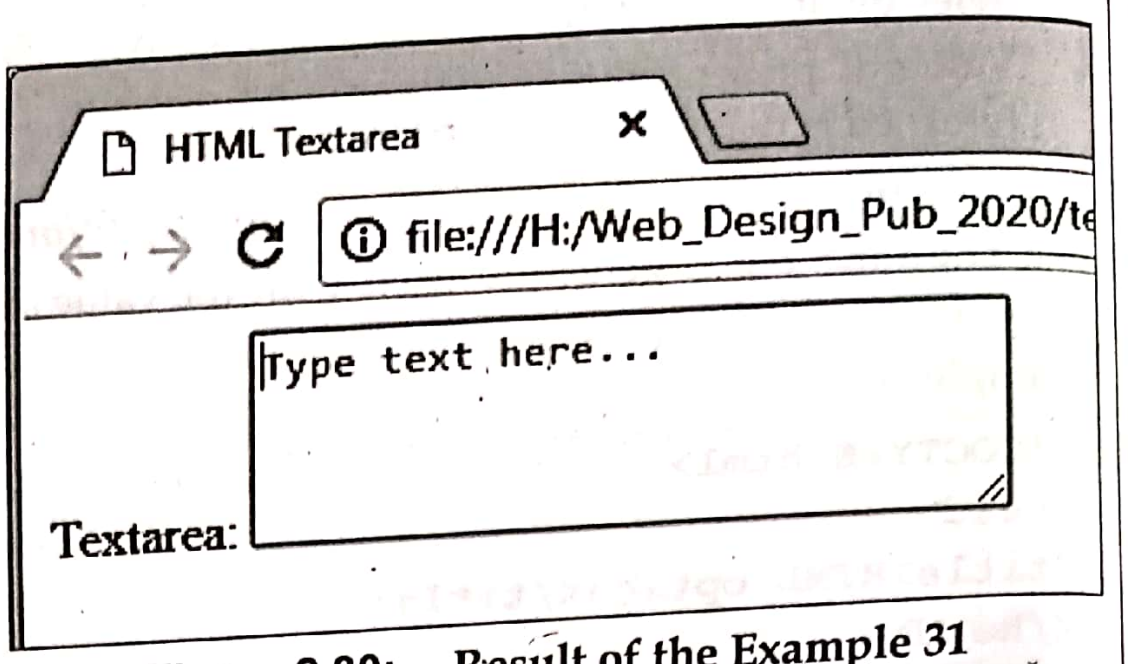


Figure 2.30: Result of the Example 31

The Example 31 will display the output as shown in Figure 2.30.

INPUT: INPUT element is the basic way to get input from the user in a variety of situations. The reason is that an INPUT element can define many types of graphical user interface features. These are as follows:

- text
- checkbox
- radio
- password
- submit
- button

A sample **INPUT** tag that queries the user for a name (using "text" as the value of the attribute Type) is given below:

```
<INPUT Type = "text" Size = 40 Name = "user-name">
```

Type identifies the text value of the attribute that will be displayed as 40 characters.

The Name attribute of the **INPUT** element designates the variable name that will be used in the data structure sent to the query program. This name is used to identify the value of the user's response from this **INPUT** element.

The **INPUT** tag is very versatile and can be used to create many kinds of user interface features.

The **INPUT** element has the following attributes:

- **Align:** Used only with the image Type. Possible values are **top, middle,** and **bottom,** and they define the relationship of the image to the text following it.
- **Accept:** It specifies a filter for what file types the user can pick from the file input dialog box.
- **Checked:** CHECKED (no value needed) specifies that this checkbox or radio button is checked by default. This is only appropriate for checkboxes and radio buttons.
- **Maxlength:** MAXLENGTH is the maximum number of characters that are accepted as input. This is only appropriate for text entry fields and password entry fields (and only for single-line text entry fields at that). If this is not present, the default will be unlimited.

Scanned with OKEN Scanner

- **Name:** Specifies a symbolic name used in transferring and identifying the output from this element of the form.
- **Size**: SIZE is the physical size of the input field in characters. This is only appropriate for text entry fields and password entry fields. If this is not present, the default is 20. Multiline text entry fields can be specified as follows:
  `SIZE = WIDTH, HEIGHT`
- **Src**: Defines the source file for the image when the attribute **Type** is set to image.
- **Text:** A single-line, text-entry area (the **TEXTAREA** element is used for multiline text input). If the only element in the form has attribute **Text**, the user can submit the form by pressing key on the keyboard.
- **Type:** Defines what kind of input field is presented to the user.
- **Value:** Value can be used to specify the default contents of the field. For a checkbox or a radio button, **VALUE** specifies the value of the button when it is checked.
- **Text:** Text is an input element which is used to give the single line input of text and is defined by **<input type="text">** tag. The code below demonstrates the single line input text.

Example 32:

```
<!DOCTYPE html>
<html>
<body>
<form action="action_page.php">
Name:<br>
<input type="text" name="Name">
<br>Father name:<br>
<input type="text" name="Father name">
<br><br>
<input type="submit">
</form>
<p>Deafult width of the form is 20
characters.</p>
</body>
</html>
```

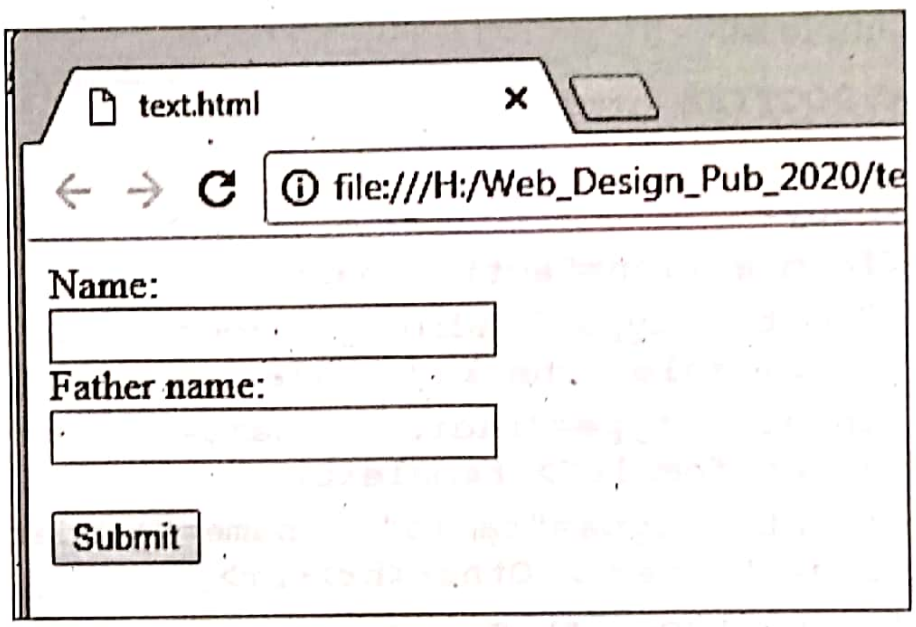


Figure 2.31: Result of the Example 32

The Example 32 will display the output as shown in Figure 2.31.

Checkbox: Check boxes are utilized when more than one option is required to be chosen and are defined by **<input type="check box">**. The code below demonstrates the use of check box.

Example 33:

```
<!DOCTYPE html>
<html>
<head>
<title>Checkbox Control</title>
</head>
<body>
<form>
<input type="checkbox" name="HTML" value="on"> HTML
```

(contd...)

Scanned with OKEN Scanner

```
<input type="checkbox" name="HTML5" value="on"> XHTML5
</form>
</body>
</html>
```

The Example 33 will display the output as shown in Figure 2.32.

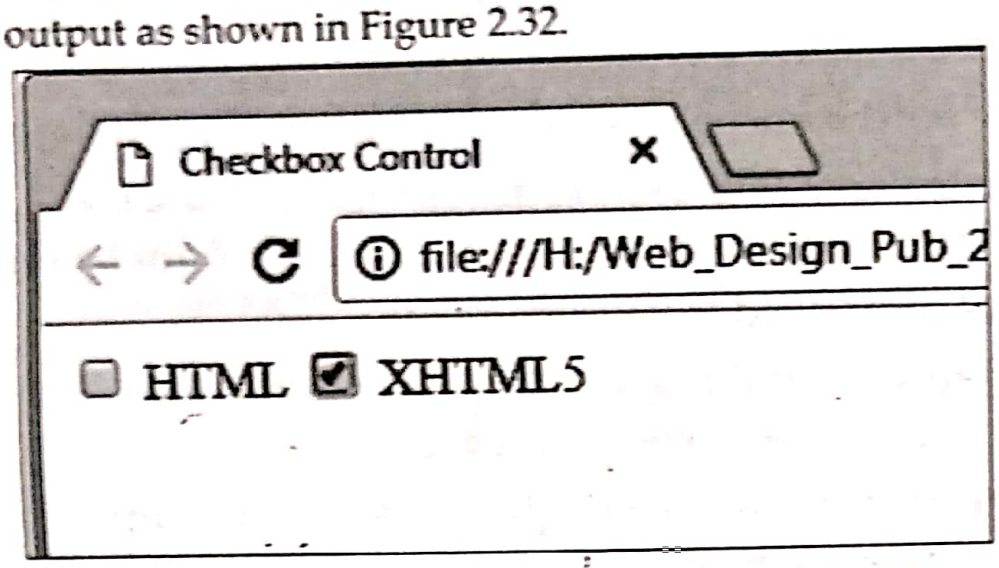


Figure 2.32: Result of the Example 33

Radio: Radio button is an input element which is used to select only one option from the given options and defined by **<input type="radio">**. The code below demonstrates the use of radio button.

Example 34:

```
<!DOCTYPE html>
<html>
<body>
<form action="action_page.php">
<input type="radio" name="gender"
value="male" checked> Male<br>
<input type="radio" name="gender"
value="female"> Female<br>
<input type="radio" name="gender"
value="other"> Other<br><br>
<input type="submit">
</form>
</body>
</html>
```

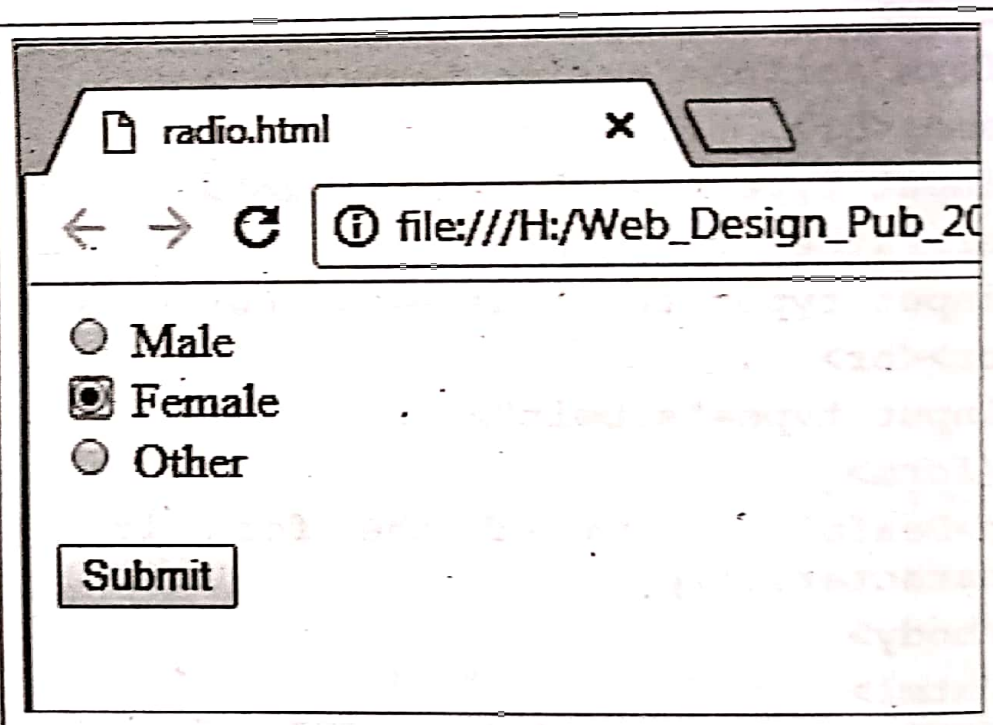


Figure 2.33: Result of the Example 34

The Example 34 will display the output as shown in Figure 2.33.

Password: Password is a single lined text box which display the characters as a dotted lines and defined by **<input type="password">**. The code below demonstrates the use of password field.

Example 35:

```
<!DOCTYPE html>
<html>
<body>
<form action="password">
Name:<br>
<input type="text" name="userid">
```

(Contd...)

Scanned with OKEN Scanner

```
<br>
Password:<br>
<input type="password" name="psw">
</form>
<p>The characters in a password field are masked (shown as asterisks or
circles).</p>
</body>
</html>
```

The Example 35 will display the output as shown in Figure 2.34.

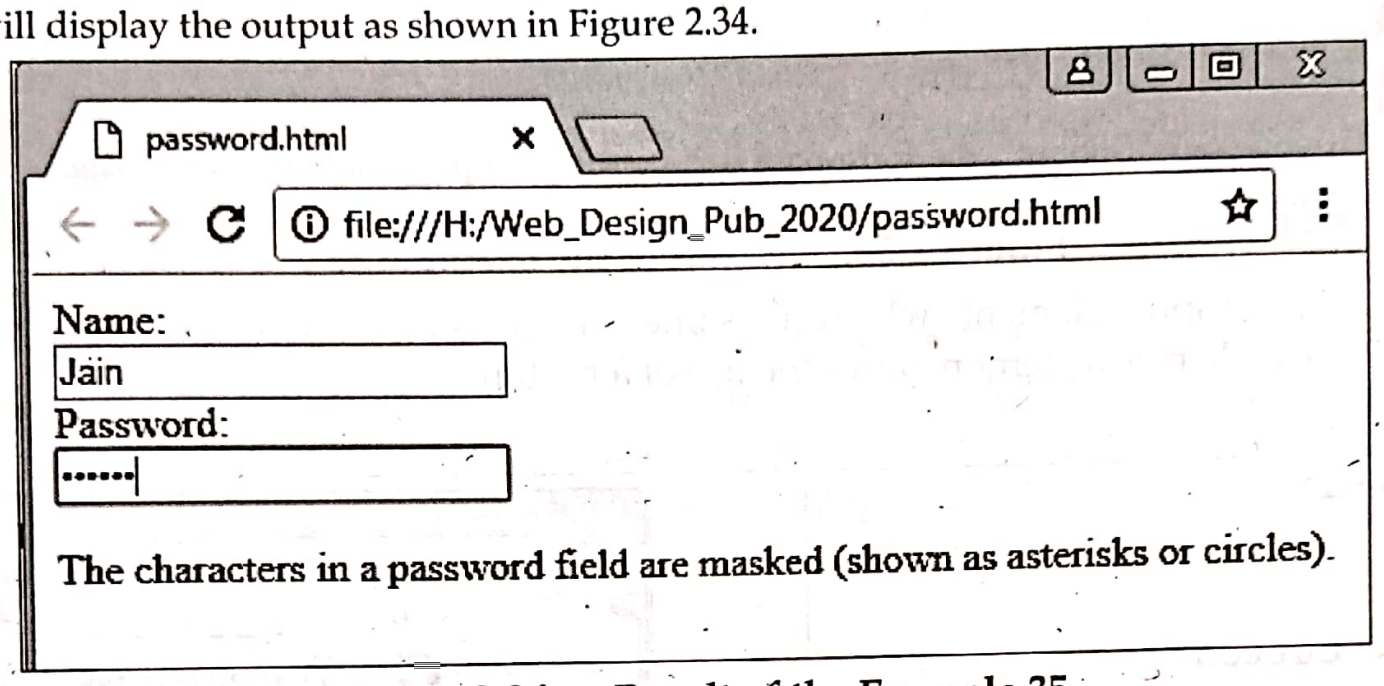


Figure 2.34: Result of the Example 35

Submit: Submit is an input element and is used to submit the form to the form-handler which is a server page and process the input data. Submit input type is defined by **<input type="submit">**. The code below demonstrates the use of submit button.

Example 36:

```
<!DOCTYPE html>
<html>
 <body>
 <form action="action_page.php">
Name:<br>
<input type="text" name="Name" value="Mickey">
<br>
Father name:<br>
<input type="text" name="Fathername" value="Dany">
<br><br>
 <input type="submit" value="Submit">
</form>
 <p>If you click "Submit", the form-data will be sent to a page called
"action_page.php".</p>
</body>
</html>
```

The Example 36 will display the output as shown in Figure 2.35.

Scanned with OKEN Scanner

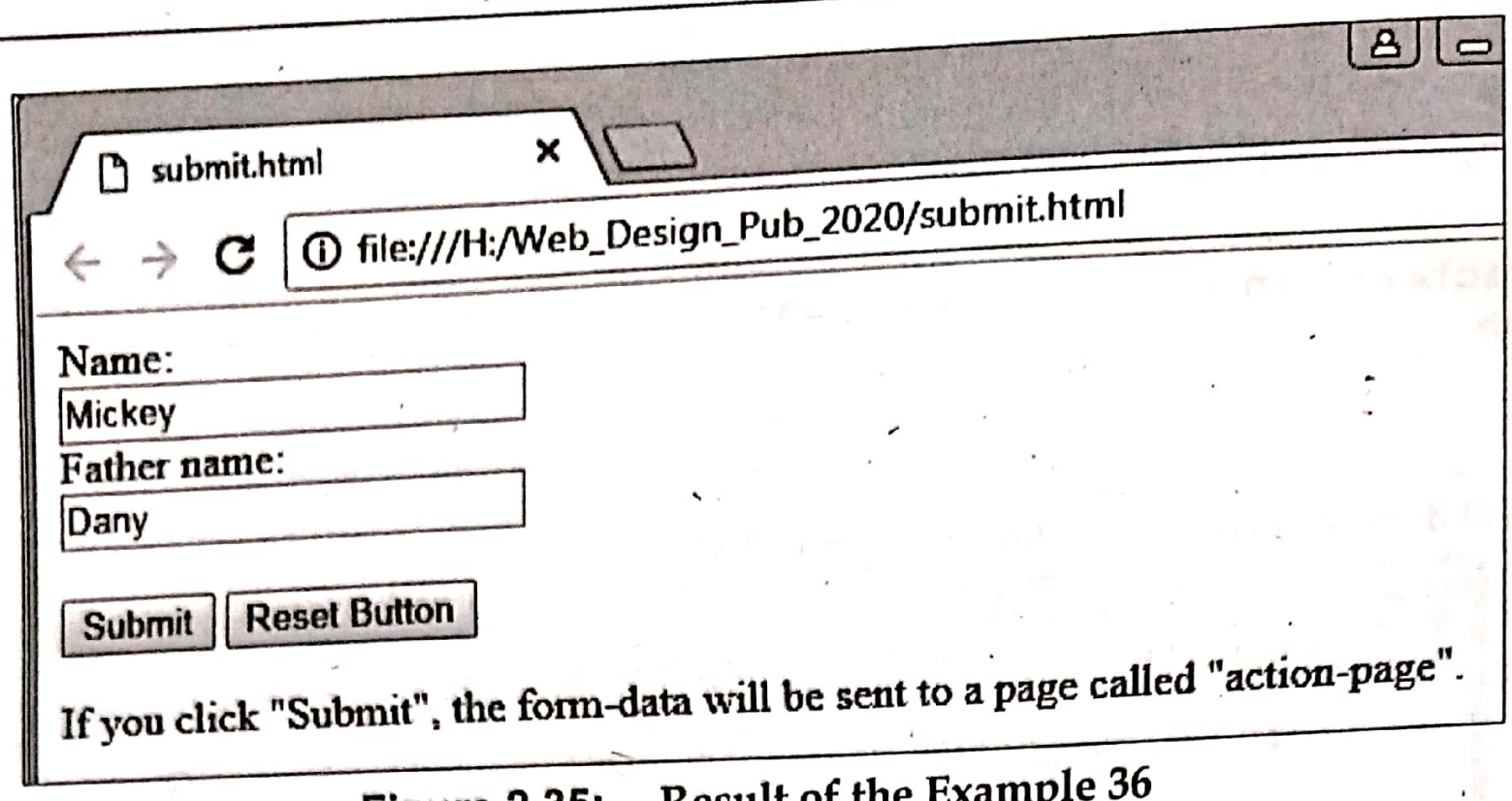


Figure 2.35: Result of the Example 36

Button: Button is an input element which is same like a submit button and defined by <input type="button">. The code below demonstrates the use of a button.

Example 37:

```
<!DOCTYPE html>
<html>
<body>
<input type="button"
   onclick="alert('Splessons!')"
       value="Click Me!">
</body>
</html>
```

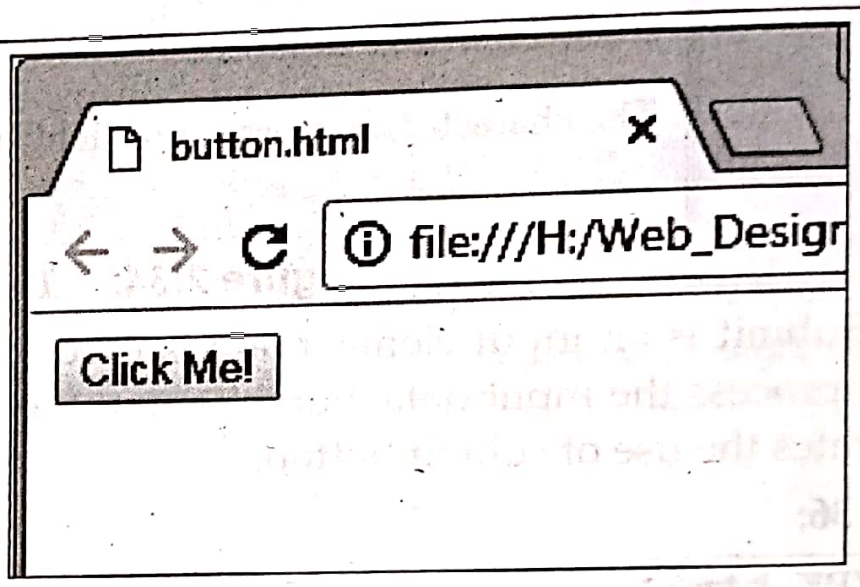


Figure 2.36: Result of the Example 37

2.8 HTML Form Program

The HTML <form> tag is used for creating a form for user input. Example 38 gives an HTML program for a form design. The corresponding output as shown in Figure 2.37.

Syntax :

```
<form>
    form content goes here..
</form>
```

A Form can contain textfields, checkboxes, radio-buttons and more. Text field creates a single line text.

Example 38:

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>Form Example</title>
</head>
```

(Contd...)

Scanned with OKEN Scanner

```
 <body bgcolor="#CCFF66">
<form align="center">
<FORM METHOD="POST" ACTION="http://www.mcp.com
/cgi-bin/post-query">
<h3 style="color:#F00">Personal Details</h3>
<strong>Name:</strong>
<input type="text" name="name" id="name"><br><br>
<strong>Password:<strong>
<input type="password" name="password" id="password"><br><br>
 <strong>E-mail id:</strong><input type="text" name="name" id="name"><br><br>
<strong>Gender:</strong> <input type="radio" name="radiogroup1" value="radio"
id="radiogroup1"> Male
<input type="radio" name="radiogroup1" value="radio" id="radiogroup2">
Female<br><br>
<strong>Contact#:</strong> <input type="text" name="mobile" id="mobile">
 <h3 style="color:#0000ff">Educational Qualification</h3>
 <strong>Degree:</strong> <select name="degree" id="degree">
        <option selected>-- Select Group --</option>
        <option>B.Com</option>
        <option>B.sc</option>
        <option>B.com Computers</option>
        <option>B.A</option>
      </select><br>
      <br>
<strong>Engineering:</strong> <select name="eng" id="eng">
        <option selected>-- Select Group --</option>
        <option>CSE</option>
        <option>ECE</option>
        <option>CIVIL</option>
        <option>EEE</option>
      </select><br><br>
<strong>Hobbies:</strong> <input type="checkbox" name="CheckboxGroup1"
value="checkbox" id="CheckboxGroup1">Crosswords Puzzles
<input type="checkbox" name="CheckboxGroup1" value="checkbox"
id="CheckboxGroup2">Reading Books<br><br>
<h3 style="color:#F00">Address</h3>
<textarea name="textarea" cols="35" rows="5" id="textarea"></textarea><br>
<br>
Attch Resume: <input type="file" name="fileField" id="fileField"><br><br>
 <input type="submit" value="Submit">
<input type = "reset" name = "reset"  value = "Reset" />
</form>
</body>
</html>
```

The Example 38 will display the output as shown in Figure 2.37.

Scanned with OKEN Scanner

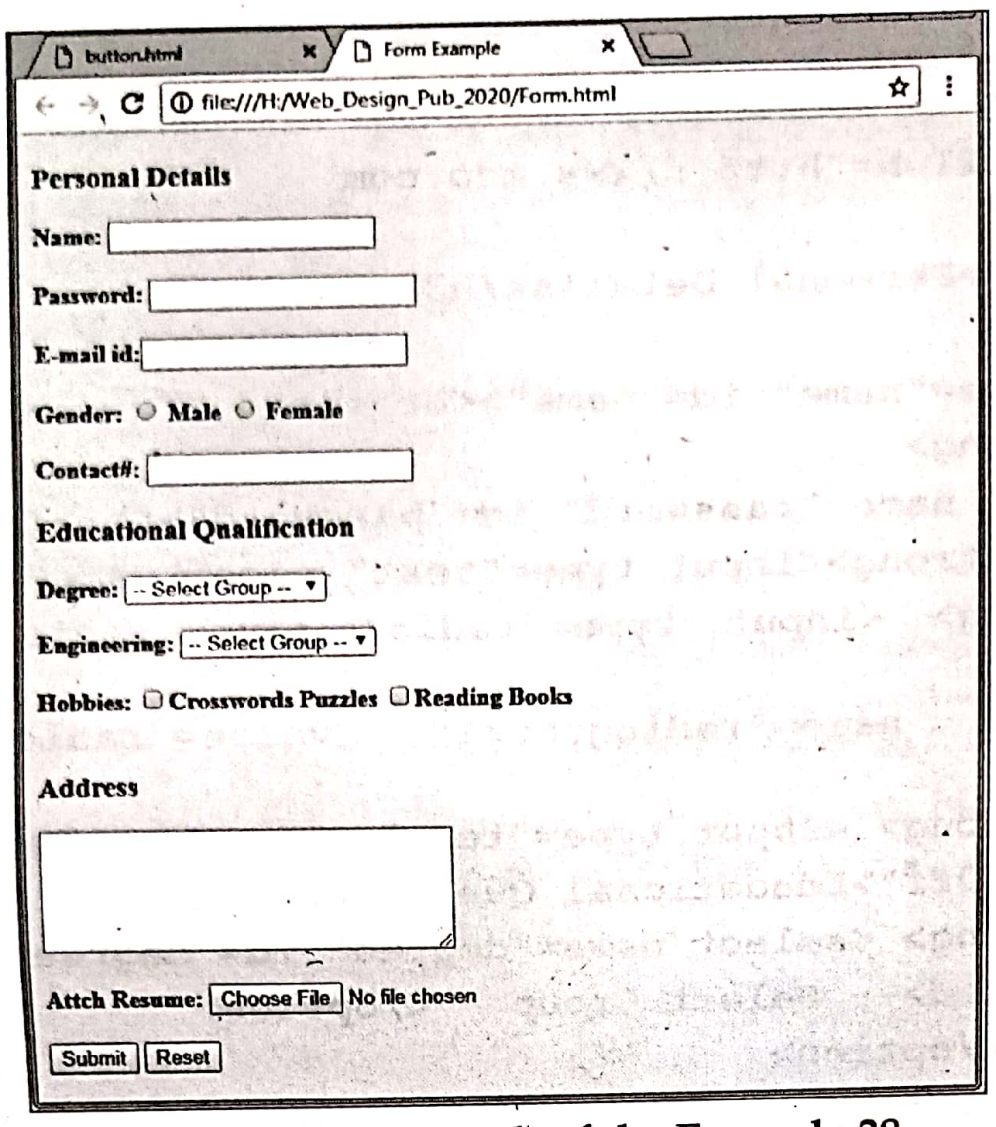


Figure 2.37: Result of the Example 38

2.9 HTML FRAMES

Frames divide a single browser window into multiple, independent regions, each of which can display a different HTML document. You begin by created FRAMESET document, which sets up the layout by establishing the number of frames, their position and other characteristics.

This frameset document is structural only and does not contain any of the content that will appear on the final web page. It does however, specify a URL reference for each frame, which will be the HTML document, or normal web page that will be displayed within it.

To use frames on a page, you use **<frameset>** tag instead of **<body>** tag. The **<frameset>** tag defines, how to divide the Window into frames. The **rows** attribute of **<frameset>** tag defines **horizontal frames** and **cols** attribute defines **vertical frames**. Each frame is indicated by **<frame>** tag and it defines which HTML document shall open into the frame.

The main advantage of using frames becomes apparent when a user clicks on one of the links of any frame, the other frames on the same Web page do not change at all, but a new page is loaded and displayed in the same frame.

<FRAMESET> TAG

To create a frame definition document, you use the **<FRAMESET>** tag. When you use **<FRAMESET>** tag in an HTML document, it is replacement for the <BODY> tag, as shown below:

```
<HTML>
<HEAD>
</HEAD>
<FRAMESET rows = "33%, 33%, 33%">
<FRAMESET cols = "50%, 50%">
<FRAME name = "frame1">
```

Scanned with OKEN Scanner

```
<FRAME name = "frame2">
</FRAMESET>
<FRAME name = "frame3">
<FRAME name = "frame4">
</FRAMESET>
<BODY>
 ..contents to display in non-frame-capable
user agent..
</BODY>
</HTML>
```

If you include a <FRAMESET> tag in an HTML document, you cannot include a <BODY> tag. Basically, the two tags are mutually exclusive. In addition, no other formatting tags, hyperlinks, or document text should be included in a frame definition document, except in one special case the **<NOFRAME>** tag. <FRAMESET> tags contain only the definitions for the frames in this document.

The <frameset> Tag Attributes

The attributes of the <frameset> tag:

- **Cols:** The cols attribute is used to create vertical frames in web browser. This attribute is basically used to define the number of columns and its size inside the frameset tag. The size or width of the column is set in the frameset in the following ways:
- *For example*, to create three vertical frames, use cols = "100, 500, 100".
- **Rows:** The rows attribute is used to create horizontal frames in web browser. This attribute is used to define no of rows and its size inside the frameset tag. The size of rows or height of each row use the following ways:
- *For example*, to create two horizontal frames, use rows = "10%, 90%".
- Border: This attribute of frameset tag defines the width of border of each frames in pixels. For example, **<frameset border="4" frameset>**
- **Frame border:** This attribute of frameset tag is used to specify whether the three-dimensional border should be displayed between the frames or not for this use two values 0 and 1, where 0 defines for no border and value 1 signifies for yes there will be border.
- **framespacing:** This attribute of frameset tag is used to specify the amount of spacing between the frames in a frameset. This can take any integer value as an parameter which basically denotes the value in **pixel.**

For example, framespacing = "10" means there should be 10 pixels spacing between each frames.

To illustrate this, the following example will define a frameset with two columns and one row.

First, define a frameset with two columns.

Example 39:

```
<html>
<head>
<title>Frame Tag Example</title>
</head>
<frameset rows="100,*">
<frame src="FirstFrame.html">
<frameset cols="150,*">
<frame src="twocolums.html">
```

Scanned with OKEN Scanner

```
<frame src="twocolums1.html">
</frameset>
<noframes>
<body>
<p>Alternate content</p>
</body>
</noframes>
</frameset>
</html>
```

The Example 39 will display the output as shown in Figure 2.38.

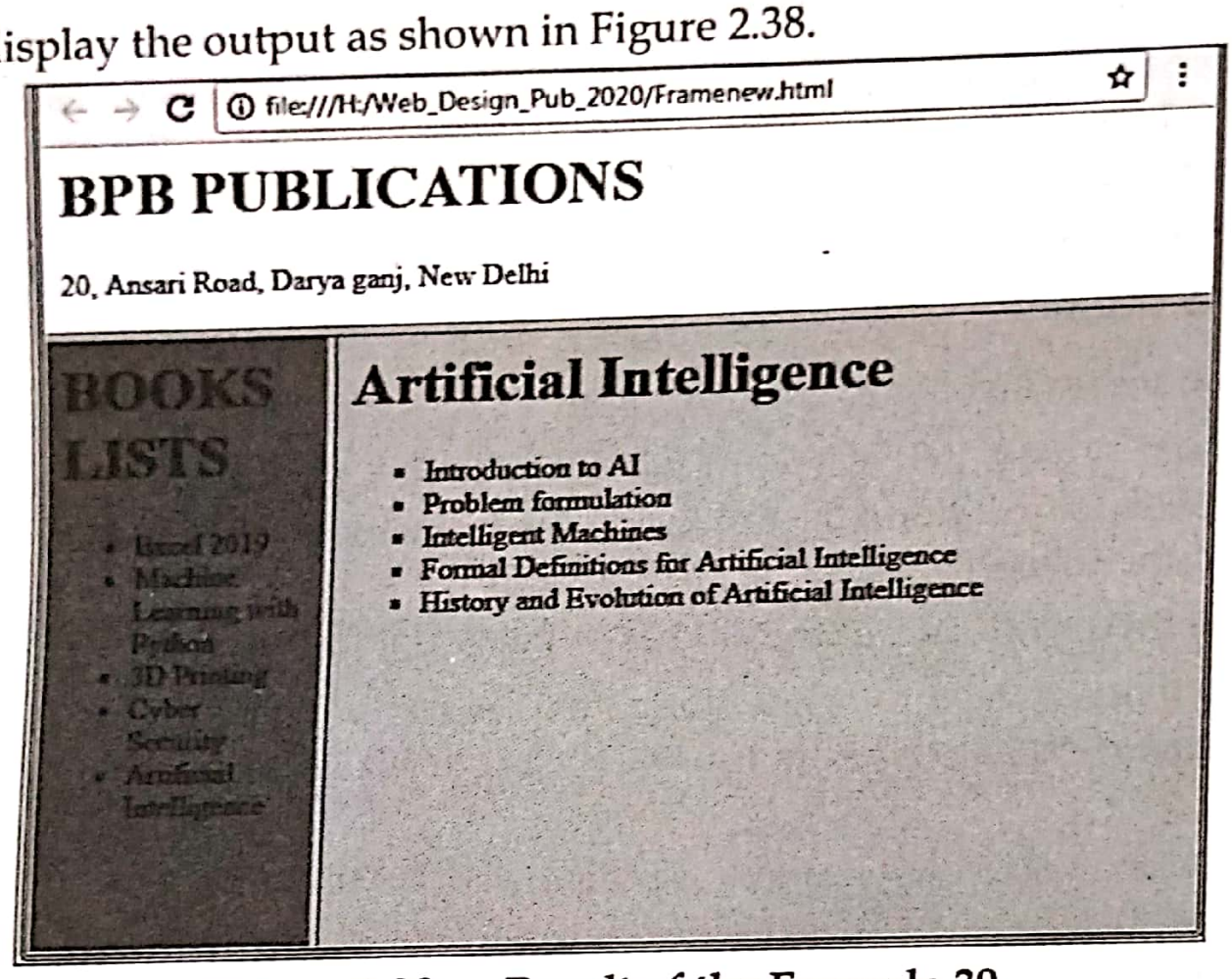


Figure 2.38: Result of the Example 39

Creating Two or More Columns (<FRAMESET COLS>)

When you define a **<FRAMESET>** tag, you must include one of the two attributes as part of the tag definition. The first of these attributes is the COLS attribute. The syntax of the COLS attribute is as follows:

```
<FRAMESET COLS = "column width, column width, ...">
```

The **COLS** attribute informs the browser to split the screen into a number of vertical frames whose widths are defined by column width values separated by commas. We define the width of each frame in three different ways:

a. Pixels

b. Percentage of the total width of the <FRAMESET>

c. An asterisk (*)

When you use *, the browser uses as much space as possible for the specified frame. The following **<FRAMESET>** tag creates a screen with three vertical frames. The first frame is 100 pixels wide, the second is 50 percent of the width of the screen, and the third uses all the remaining space.

```
<FRAMESET COLS = "100, 50%, *">
```

The COLS attribute defines the number of vertical frames or columns in a frameset.

To define a frameset with three equal-width columns, use COLS="*, *, *".

Scanned with OKEN Scanner

Example 40:

```
<html>
<head>
<title>TAG index</title>
</head>
<frameset cols="130,*,130">
<frame src="linkcolor.html">
<frame src="section.html">
<frame src="paragraph.html">
<noframes>
<body>
<p>Alternate content</p>
</body>
</noframes>
</frameset>
</html>
```

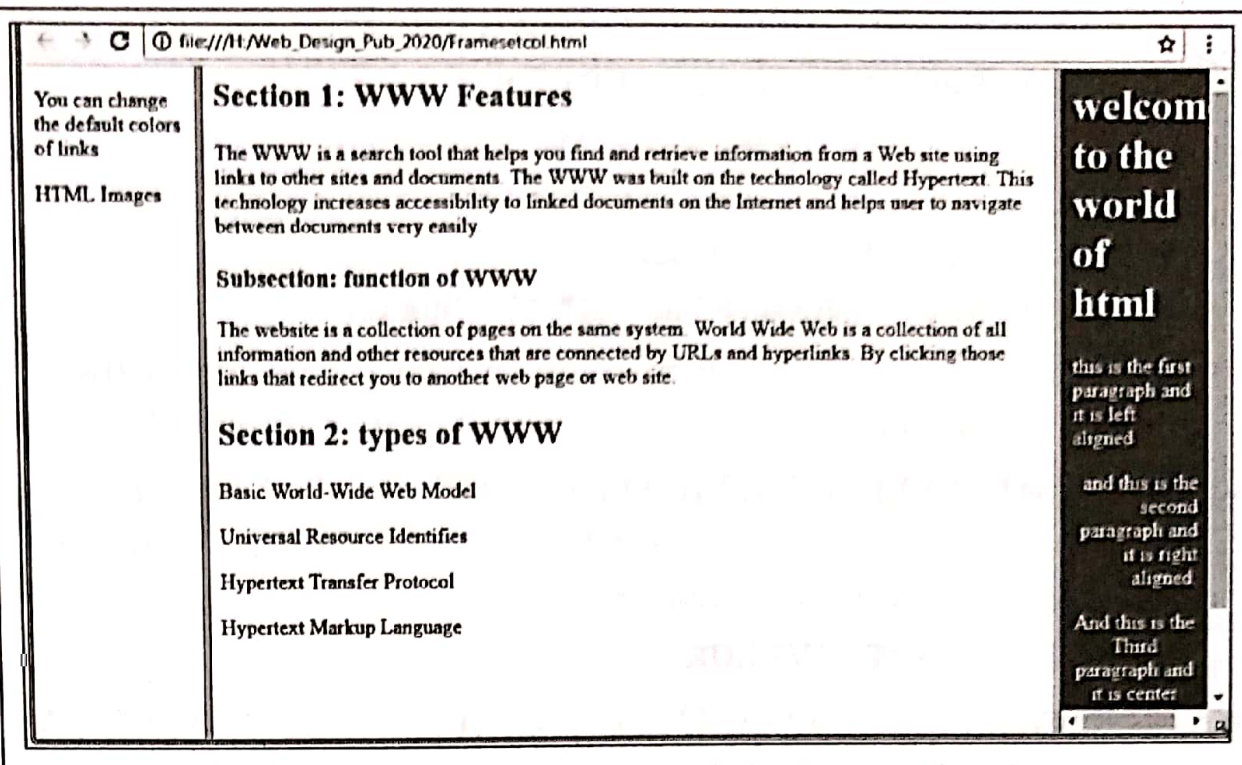


Figure 2.39: Result of the Example 40

The Example 40 will display the output as shown in Figure 2.39.

Creating Two or More Rows (<FRAMESET ROWS>)

The ROWS attribute works in a similar way as the COLS attribute, except that it splits the screen into horizontal frames rather than vertical ones. *For example*, to split the screen into two equal-height frames, you write either of the following:

<FRAMESET ROWS = "50%, 50%"> or <FRAMESET ROWS = "*, *">

The ROWS attribute defines the number of horizontal frames or rows in a frameset.

Example 41:

```
<html>
<head>
<title>TAG index</title>
</head>
<frameset rows="20%,80%">
<frame src="topfirst.html">
<frame src="bottom.html">
<noframes>
<body>
<p>Alternate content</p>
</body>
</noframes>
</frameset>
</html>
```

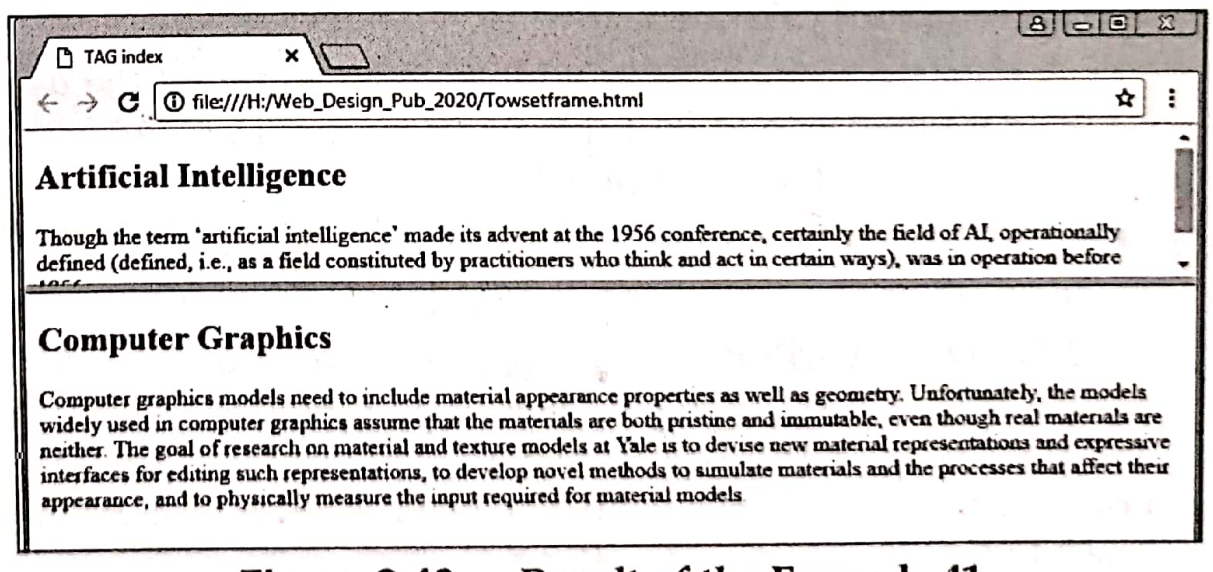


Figure 2.40: Result of the Example 41

The Example 41 will display the output as shown in Figure 2.40.

Attributes of Frame Tag:

- **<FRAME SRC>** Attribute: The **<FRAME SRC>** attribute in each **<FRAME>** tag specifies the address of

Scanned with OKEN Scanner

the document to be displayed in the frame. When omitted, the frame is displayed as a blank space. The syntax of the SRC attribute is shown below:

```
<FRAME SRC="document URL">
```

- **<FRAME NAME> Attribute:** The NAME attribute is used to assign a name to a frame so that it can be targeted by links in other documents. The NAME attribute is optional. By default, all windows are unnamed. Names must begin with an alphabetic character.

  ```
  <FRAME SRC="main.htm" NAME="main">
  ```

  This displays the main.htm page in that frame when the page loads, and names the frame main. (The frame name need not match the name of its contents. It is given so in this example).

- **<FRAME MARGINHEIGHT> Attribute:** To adjust the margin that appears above and below a document within a frame, set the MARGINHEIGHT to a number indicated by pixels. The general form of MARGINHEIGHT attribute is as follows:

  ```
  MARGINHEIGHT="VALUE"
  ```

- **<FRAME MARGINWIDTH> Attribute:** The MARGINWIDTH attribute enables you to adjust the margin on the left and right side of a frame to the number indicated by pixels. Margins cannot be less than one. This ensures that frame objects will not touch frame edges, and cannot be specified in a way that leaves no space for the document contents. **MARGINWIDTH** attribute is optional. The general form of **MARGINWIDTH** attribute is as follows:

  MARGINWIDTH="VALUE"

- **<FRAME SCROLLING> Attribute:** The **<FRAME SCROLLING>** attribute allows you to control scrollbar as and when the scrollbars appear in the frame. Normally, scrollbars are added only if there is not enough room to display all the contents in a frame. The general form of SCROLLING attribute is given below:

  ```
  SCROLLING="YES|NO|AUTO"
  ```

The SCROLLING attribute is optional. The default value is **auto**. Different parts of the SCROLLING attribute are as follows:

- **AUTO (default)**: By default, if the contents of a frame take up more space than the area available, the browser automatically adds scrollbars to either the side or the bottom of the frame so that the user can scroll through the document.
- **NO**: Setting the value of SCROLLING to NO disables the use of scrollbars for the current frame. If you set SCROLLING = "NO" but there is more text in the document than can fit inside the frame, you will not be able to scroll the additional text into the frame view.
- **YES**: If you set SCROLLING to YES, the scrollbars are included in the frame regardless of whether they are required or not.
- **<NORESIZE> Attribute**: By default, users can move the position of borders around each frame on the current screen by grabbing the border and moving it with their mouse. To lock the borders of a frame and prevent them from being moved, you use the NORESIZE attribute. The NORESIZE attribute does not require a value. It can simply be used as a flag to indicate that the frame is not resizable by the user. The NORESIZE attribute is optional.

  However, all frames are resizable by default.

- **<NOFRAMES> Tag:** If you load a frame definition document into a Web browser that does not support frames, you get only a blank page. To get around this problem, the <NOFRAMES> tag enables you to include body text as part of the document. Frame-capable browsers ignore all tags and data contained in a NOFRAME element.

Scanned with OKEN Scanner

Nested Framesets

You can achieve more complex layouts by using nested tags. Any frame within a frameset can contain another frameset.

For example, shows a layout of two columns, the first with two rows and the second with three rows. This is created by nesting two tags with row specifications within a top-level that specifies the columns.

Example 42:

```
<html>
<head>
<title>TAG index</title>
</head>
<frameset cols="50%,*">
<frameset rows="50%,*">
<frame src="H:/Web_Design_Pub_2020/Delhi.htm">
<frame src="H:/Web_Design_Pub_2020/global.htm">
</frameset>
<frameset rows="33%,33%,*">
<frame src="H:/Web_Design_Pub_2020/natural.htm">
<frame src="H:/Web_Design_Pub_2020/kerala.htm">
<frame src="H:/Web_Design_Pub_2020/punjab.htm">
</frameset>
</frameset>
</html>
```

The Example 42 will display the output as shown in Figure 2.41.

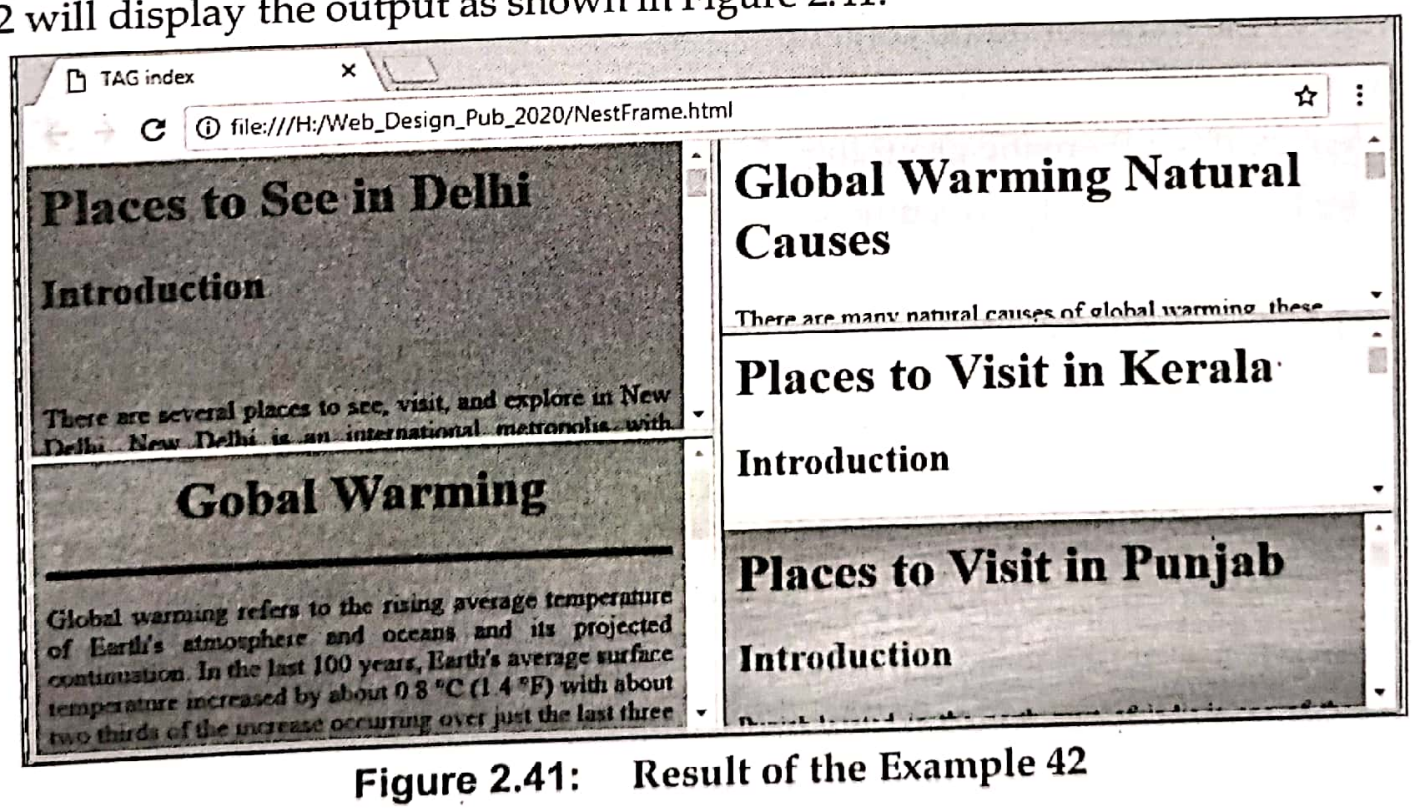


Figure 2.41: Result of the Example 42

2.10 HTML5 Introduction

HTML5 is the latest version of HTML. It is not a programming language, but relatively a markup language. The combination of hypertext and markup language used to design web pages. Hypertext defines the link between the web pages. Markup language is used to define the text document within tag which defines the

Scanned with OKEN Scanner

structure of web pages. It has improved the markup available for documents and has introduced **Application Programming Interfaces(API)** and **Document Object Model(DOM)**.

HTML Features

HTML 5 adds a lot of new features to the HTML specification, and it is easy to implement. These are:

- It supports multimedia features controls by using <audio> and <video> tags.
- The new vector graphics elements tags.
- Improve semantic content by including <header> <footer>, <article>, <section> and <figure> are added.
- Drag and Drop- The user can hold or drag an object it on a new location.
- It helps to locate the geographical location of a client.
- It provides web application methods to store data on web browser.
- It uses SQL database to store data.
- To draw various shapes like triangle, rectangle, circle, etc.
- Handling incorrect syntax.
- DOCTYPE declaration i.e. <!doctype html>
- Character encoding i.e. <meta charset="UTF-8">

HTML5 New Elements

- **<article>:** The <article> tag is used to represent an independent piece of content of a document, such as a blog entry or newspaper article.
- **<aside>:** The <aside> tag is used to describe the main object of the web page. It basically identifies the content that is related to the primary content of the web page. It used mainly author information, links, related content and so on.
- **<audio>:** It defines the music or audio content.
- **<video>:** It defines the video content.
- **<section>:** It demarcates a thematic grouping of content.
- **<Nav>:** Represents a section of the document intended for navigation.

Advantages

The advantages of HTML5 are as follows:

- All browsers supported.
- More device friendly.
- Easy to use and implement.
- HTML 5 in integration with CSS, JavaScript, etc. can help build beautiful websites.

Disadvantages

The disadvantages of HTML5 are as follows:

- Long codes have to be written which is time consuming.
- Only modern browsers support it.

The DOCTYPE

HTML 5 use simple syntax to specify DOCTYPE as follows:

```
<!DOCTYPE html>
```

The above syntax is case-insensitive.

Scanned with OKEN Scanner

The character encoding declaration is:

```
<meta charset="UTF-8">
```

The markup for an HTML 5 document would look like the following:

```
<!DOCTYPE html>
<html>
   <head>
      <meta charset = "utf-8">
      <title>...</title>
   </head>
     <body>
      <header>...</header>
      <nav>...</nav>
      <article>
         <section>
            ...
          </section>
      </article>
      <aside>...</aside>
        <footer>...</footer>
     </body>
</html>
```

HTML 5 <article> Tag: The HTML <article> tag is an HTML5 element that defines a self-contained composition in the HTML document. This tag is also commonly referred to as the <article> element. This tag is generally used on **Forum post, Blog post, News story, comment** etc.

Example 43:

```
<!doctype html>
<html>
<head>
<meta charset="UTF-8">
<title>HTML5 ARTICLE TAG</title>
</head>
<body>
<article>
  <h1>Heading for Article</h1>
  <p>Text that appears under
article</p>
</article>
</body>
</html>
```

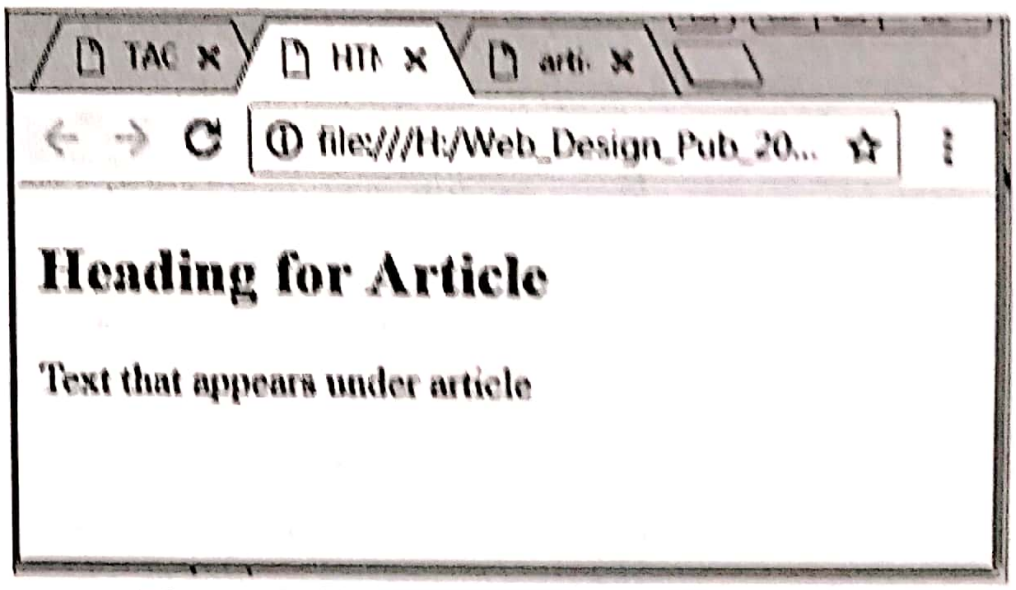


Figure 2.42: Result of the Example 43

The Example 43 will display the output as shown in Figure 2.42.

In this HTML5 Document example, we have created an **<article>** tag displays 2 tags - <h1> tag and <p> tag.

Scanned with OKEN Scanner

<aside> tag: The HTML **<aside>** tag provides information about the main content. The aside element is new to HTML5 and it can be used in two different contexts. The context of the **<aside>** element is based on whether or not it inside or outside the article element.

Example 44:

```
<html>
<head>
<meta charset="UTF-8">
<title>HTML NEW ELEMENTS </title>
</head>
<body>
<article>
  <h1>Expression of vascular endothelial growth factor in patients with
premalignant lesions and squamous cell carcinoma of oral cavity.</h1>
  <p>AIMS:To evaluate and compare expression of VEGF in patients of
premalignant lesions and squamous cell carcinoma of oral cavity.</p>
</article>
<aside                                          style="font-size:larger;font-
style:italic;color:orange;float:right;width:30%;padding-left:7px;">
In this study we concluded that VEGF expression increases as the lesion
progresses from premalignant lesions to oral squamous cell carcinoma and is
strongly associated with lymph node status (N staging).
</aside>
<aside>
  <p>Text that appears under aside</p>
</aside>
</body>
</html>
```

The Example 44 will display the output as shown in Figure 2.43. In this example, you have created an **<article>** tag and following that an **<aside>** tag.

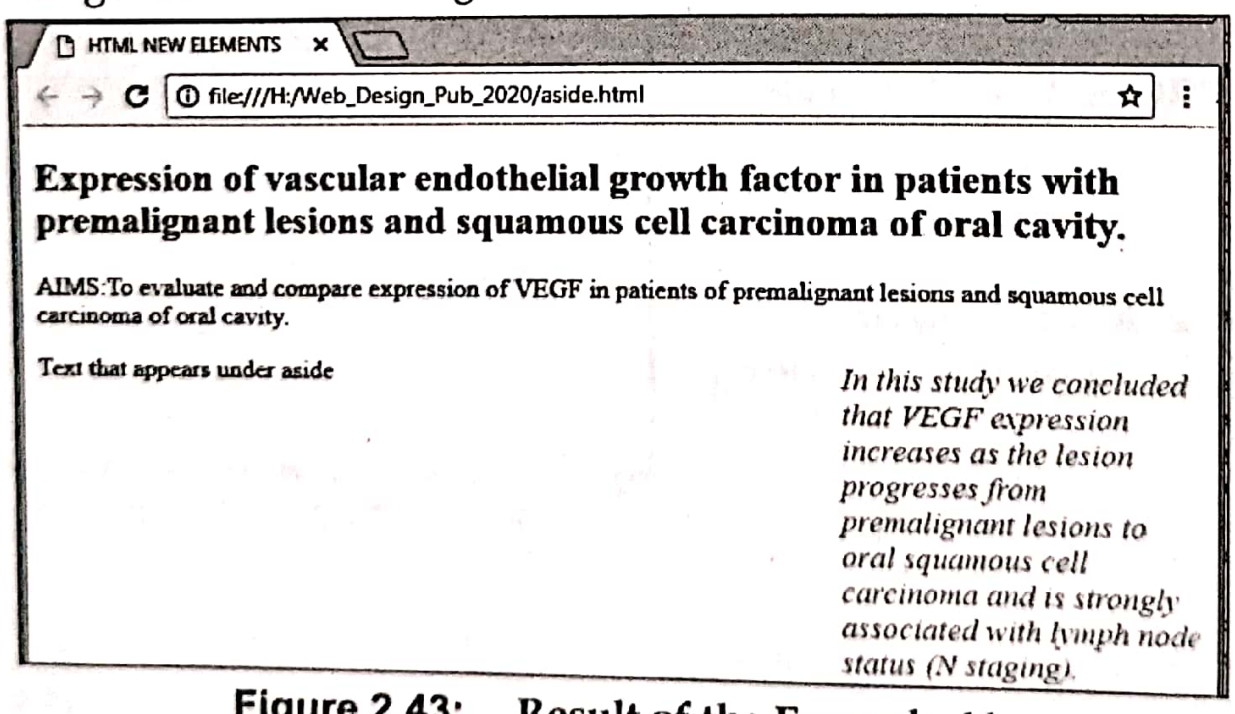


Figure 2.43: Result of the Example 44

<section> tag: The HTML **<section>** tag is an HTML5 element that defines a generic section in the HTML document. This tag is also commonly referred to as the <section> element.

Scanned with OKEN Scanner

Example 45:

```
<html>
    <head>
        <title>Example of section tag</title>
    </head>
    <body>
        <section>
            <h1>Section 1: WWW Features</h1>
            <p>The WWW is a search tool that helps you find and retrieve
information from a Web site using links to other sites and documents. The
WWW was built on the technology called Hypertext. This technology increases
accessibility to linked documents on the Internet and helps user to navigate
between documents very easily.</p>    <section>
            <h1>Subsection: function of WWW </h1>
            <p>The website is a collection of pages on the same system. World
Wide Web is a collection of all information and other resources that are
connected by URLs and hyperlinks. By clicking those links that redirect you
to another web page or web site.</p> </section>
</section>
 <section>
 <h1>Section 2: types of WWW</h1>
<p> Basic World-Wide Web Model</p>
<p>Universal Resource Identifies</p>
<p>Hypertext Transfer Protocol</p>
<p>Hypertext Markup Language</p> </section>
</body>
</html>
```

The Example 45 will display the output as shown in Figure 2.44. In this HTML5 Document example, you have created a **<section>** tag that displays 2 tag - <h1> tag and <p> tag.

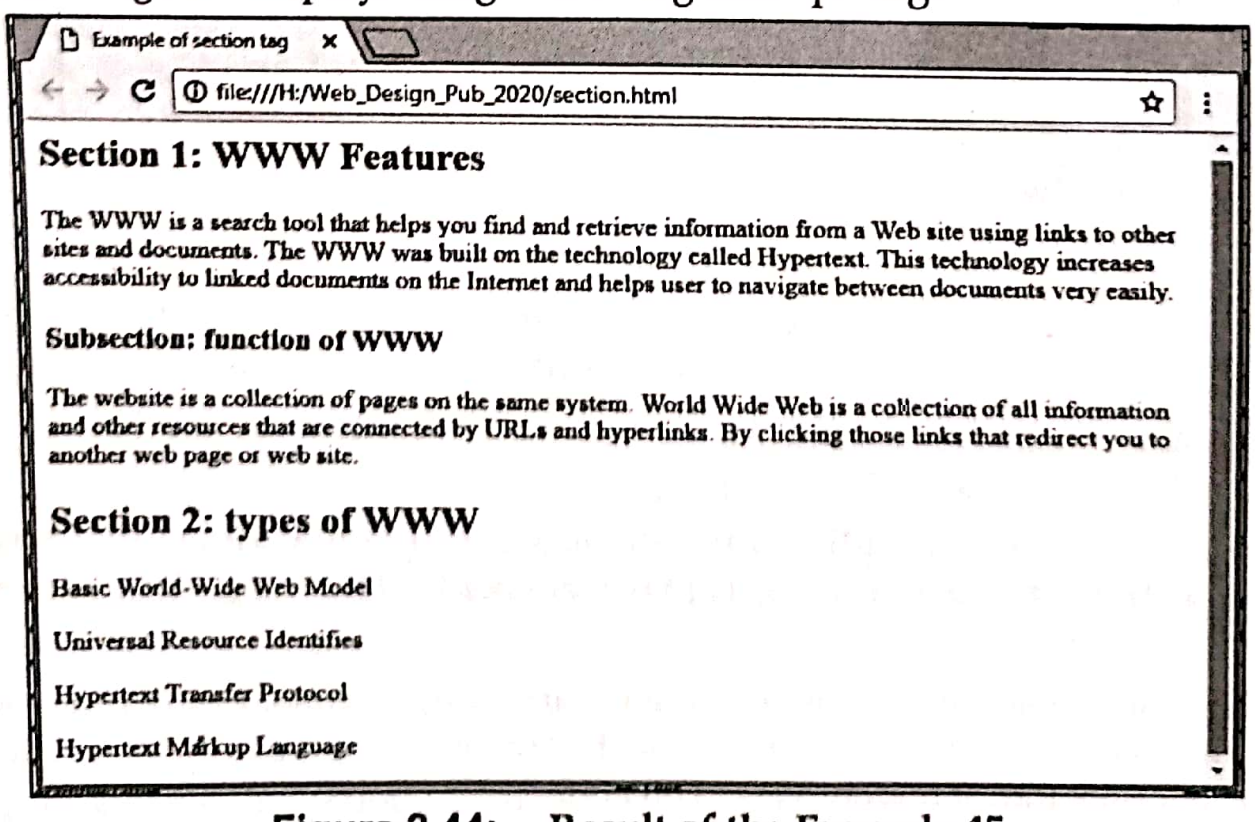


Figure 2.44: Result of the Example 45

Scanned with OKEN Scanner

<nav> Tag: The <nav> element defines a section of navigation links (i.e. links to other pages or to parts within the page itself) in a document. *Examples* of some navigation links are **menu, table of contents, indexes,** etc.

Example 46:

```
<!doctype html>
<html>
<head>
<meta charset="UTF-8">
<title>Navigation Tag </title>
</head>
<body>
<p>Tourism is one of the fastest growing industries of the world. It plays vital role in the economic development of a country. India is one of the popular tourist destination in Asia.</p>
<nav>
<ul>
<li><a href="E:/html_files/Delhi.htm" target="_top">Delhi</a></li>
<li><a href=":/html_files/global.htm" target="_top">Tamilnadu</a></li>
<li><a href="E:/html_files/kerala.htm" target="_top">Kerala</a></li>
<li><a href="E:/html_files/punjab.htm" target="_top">Punjab</a></li>
</ul>
</nav>
</html>
```

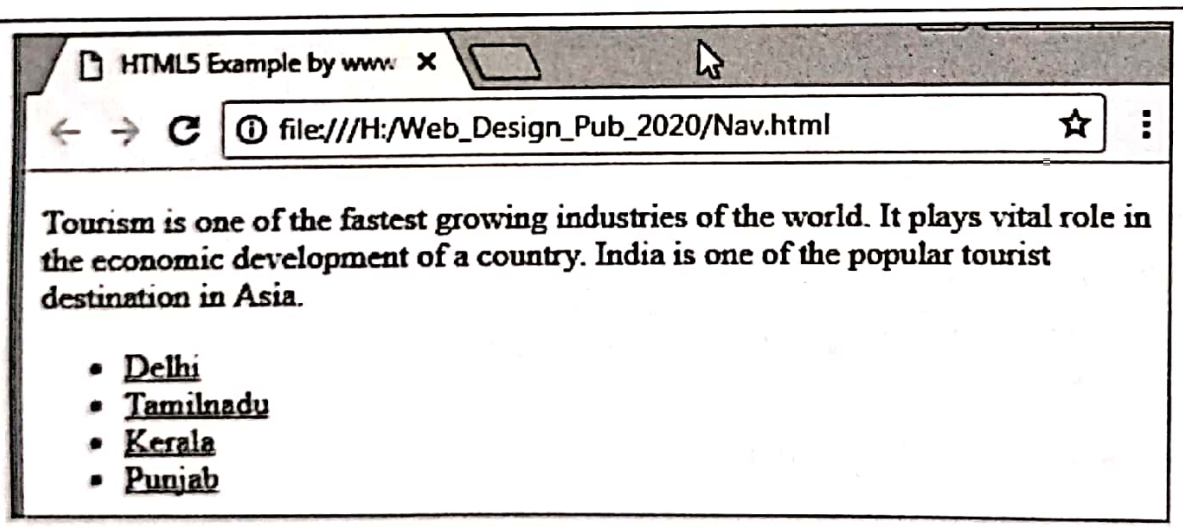


Figure 2.45: Result of the Example 46

The Example 46 will display the output as shown in Figure 2.45. In this HTML5 Document example, you have created a **<nav>** tag that has a **<ul>** tag with three **<li>** navigation links.

<audio> Tag: The **<audio>** element is used to embed audio content in an HTML document without requiring any additional plug-in like Flash player.

With the use of HTML5, user can play audio, video and also play graphic files without using JavaScript. The below examples demonstrate the use of some HTML5 elements. The image below demonstrates HTML5 media controls.

<video> Tag: HTML5 <video> element provides a standard way to embed video in web pages. However, the video element is relatively new, but it works in most of the modern web browsers. By using, the HTML5 user can insert the video files into the web page. The code below demonstrates how to insert the video file into the HTML document.

Scanned with OKEN Scanner

Example 47:

```
<html>
    <head>
        <title>HTML5 Examples</title>
  </head>
    <body>
     <audio controls>

   <source
src="F:/Web_Design_Pub_2020/Shree.ogg
" type="audio/ogg">
    </audio>
    <body>
</html>
```

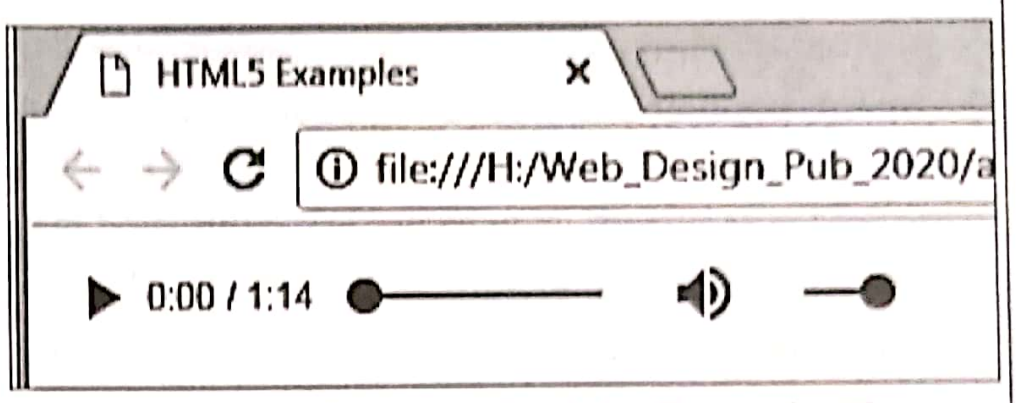


Figure 2.46: Result of the Example 47

By running the above code in a preferred browser, user can get the following output as shown in below audio file.

There are **three** video formats supported for HTML video tag:

- mp4
- webM
- ogg

Example 48:

```
<html>
    <head>
        <title> HTML5 Exampless</title>
    </head>
    <body>
        <video width="400" controls>
            <source
src="H:/Web_Design_Pub_2020/video1.mp4"
type="video/mp4">

        </video>
    <body>
</html>
```



Figure 2.47: Result of the Example 48

By running the above code in a preferred browse you can get the following output as shown below.

2.10.1 HTML5 Form Validations

HTML5 form controls is the ability to validate most user data without relying on scripts. This is done by using validation attributes on form elements. Validation attributes allow you to specify rules for a form

Scanned with OKEN Scanner

input, such as minimum and maximum length of the data. Then the minimum and maximum values (the min and max attributes); whether the data needs to be a number, an e-mail address, or something else (the type attribute); and a regular expression pattern that the data must match (the pattern attribute). In supporting browsers, if the entered data follows all of the specified rules, it is considered valid; if not, it is considered invalid.

Go to any popular site with a registration form, and you will notice that they provide feedback when you don't enter your data in the format they are expecting. You will get messages such as:

"This field is required" (You ca not leave this field blank.)

"Please enter your phone number in the format xxx-xxxx".

"Please enter a valid e-mail address" (Used in the format of "somebody@example.com")

"Your password needs to be between 8 and 30 characters long and contain one uppercase letter, one symbol and a number."

This is called **form validation**. When you enter data, the browser check to see that the data is in the correct format and within the constraints set by the various validation-related attributes. If the information is correctly formatted, the user agent (and application) allows the data to be submitted to the server and (usually) saved in a database; if the information is not correctly formatted, it gives the user an error message explaining what needs to be corrected.

You want to make filling out web forms as easy as possible. So why do you insist on validating our forms? There are three main reasons:

- **You want to get the right data, in the right format:** Your applications would not work properly if user's data is stored in the incorrect format.
- **You want to protect your user's accounts:** Forcing our users to enter secure passwords makes it easier to protect their account information.
- **You want to protect yurselves:** There are many ways that malicious users can misuse unprotected forms to damage the application.

2.10.2 HTML5 Attributes

HTML5 has provides such attributes that can minimize the script code. Let us discuss here some attributes like autofocus, placeholder and required. Just use these attributes with input tag and get the functionality. You will see the example and functionality explanation.

The Required Attribute

The required attribute specifies an input field must have be filled out before submitting the form. It works with the following input types: text, search, url, e-mail, password, date pickers, number, checkbox, radio and file.

The HTML5 attribute required is a type of validation. If an input field is using required attribute then that field must have some value on submission. The blank page will not submitted and you will get one pop up showing message that this must have some value. Find the example.

Syntax

<input required>

Example 49:

```
<!DOCTYPE html>
<html>
<body>
```

Scanned with OKEN Scanner

```
<h4>HTML 5 Required Attribute</h4>
<p>The required attribute specifies that an input field must be filled out
before submitting the form.</p>
<form action="/action_page.php" >
    <b>Your Name:</b> <input type="text" name="name" required><br>
  <br/><input type="submit" value="Submit">
</form>
</body>
</html>
```

The Example 49 will display the output as shown in Figure 2.48.

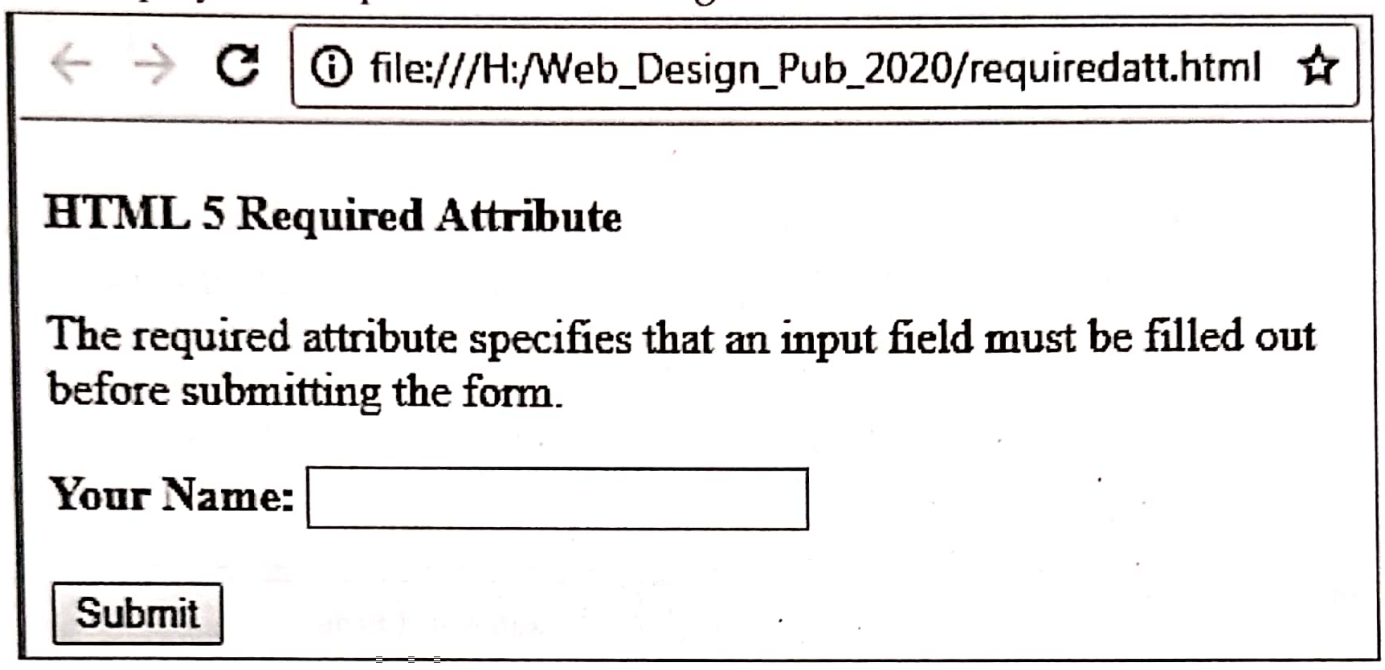


Figure 2.48: Result of the Example 49

The pattern Attribute

This attribute is used to specify the regular expression on which the input element value is checked. This attribute works with the following input types: text, password, date, search, e-mail, etc. Use the Global title attribute to describe the pattern for helping the user.

Syntax:

```
<input pattern ="regular_exp">
```

Example 50:

```
<!DOCTYPE html>
<html>
<head>
<title>Pattern attribute</title>
</head>
<body>
<p>The pattern attribute specifies a regular expression</p>
<form action="/action_page.php">
<b>Password: </b> <input type="text" name="Password"
pattern="[A-Za-z]{1,5}" title="five letter Password">
<input type="submit">
</form>
</body>
</html>
```

The Example 50 will display the output as shown in Figure 2.49.

Scanned with OKEN Scanner

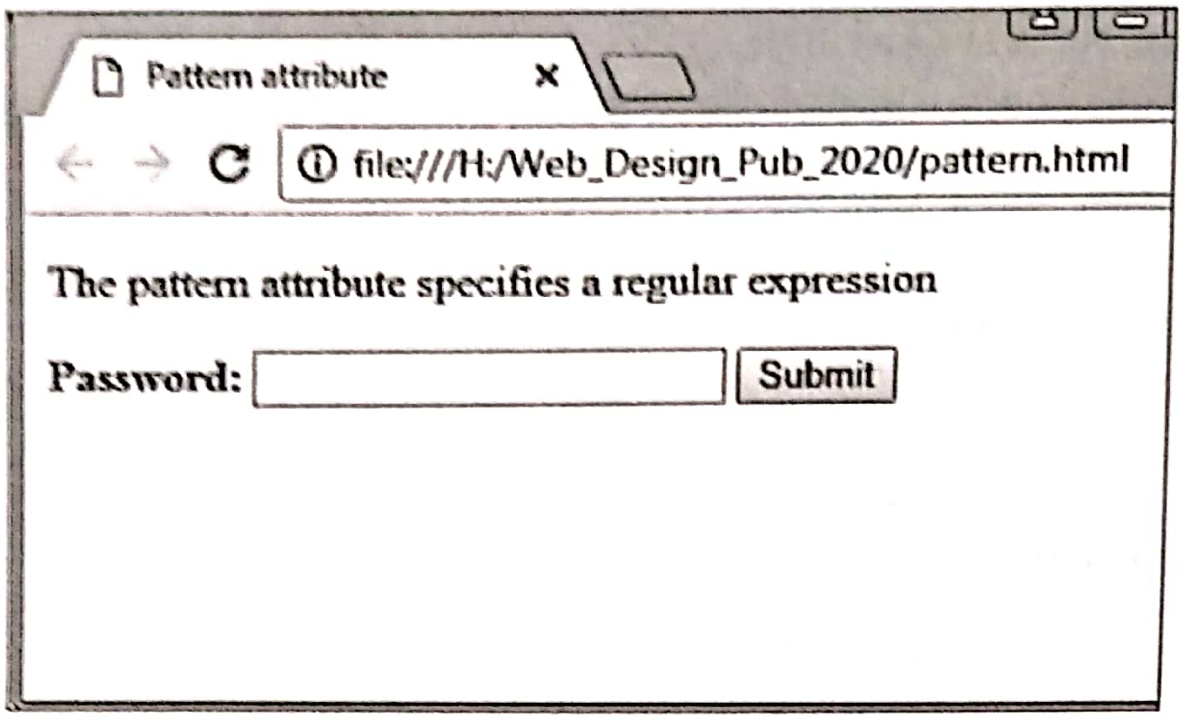


Figure 2.49: Result of the Example 50

The autofocus Attribute: The autofocus attributes focus the input element automatically when user reloa the page, autofocus is a boolean attribute. The snippet code below demonstrate the use of autofous.

It supports only the following elements: **<button>, <input>, <select>** and **<textarea>**

Syntax:

<elementName autofocus> .

Example 51:

```
<!DOCTYPE html>
<html>
<body>
<h2>The autofocus Attribute</h2>
<p></p>
<form action="/action_page.php">
  First name:<input type="text" name="fname" autofocus><br>
  Last name: <input type="text" name="lname"><br>
  <input type="submit">
</form>
</body>
</html>
```

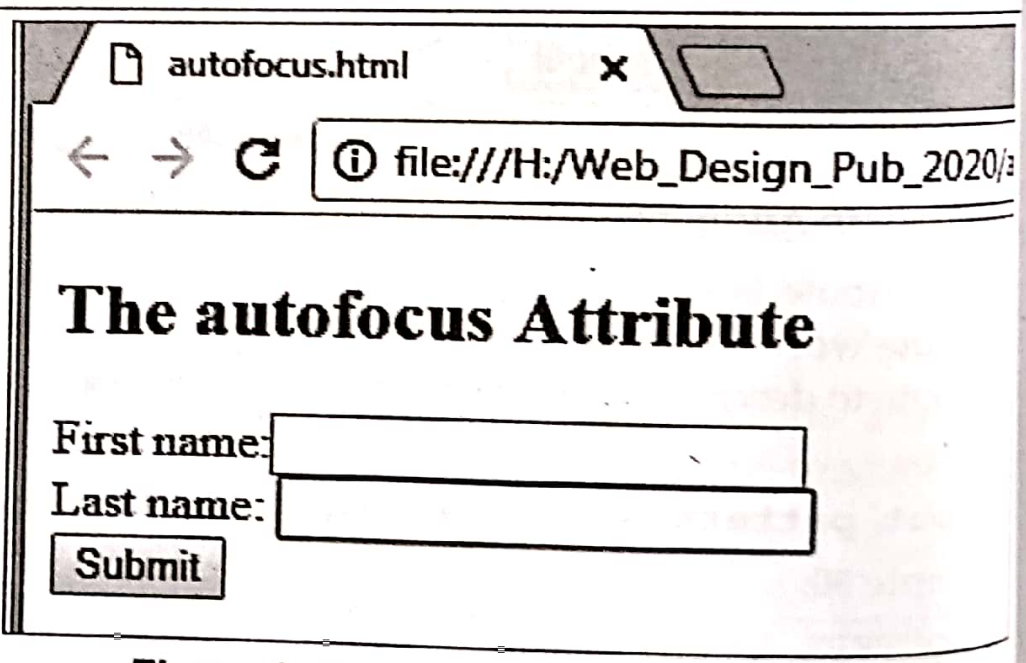


Figure 2.50: Result of the Example 51

The Example 51 will display the output as shown in Figure 2.50.

By running the above code in a preferred browser user can get the following output as shown in abov image.

2.10.3 New Input Types in HTML5

HTML5 introduce several new <input> types like email, date, time, color, range, and so on. Let us take brief look at each of the following new input types:

- date, e-mail, number, range, time, url

Input Type Date: The <input type="date"> is used for input fields that should contain a date. The user t select a date from a drop-down calendar. The date value includes the **year, month,** and **day.**

Scanned with OKEN Scanner

Example 52:

```
<!DOCTYPE html>
<html>
<body>
<h2>INPUT DATE FIELD</h2>
<p>the user to select a date from a drop-down</p>
<form action="/action_page.php">
  Birthday:
  <input type="date" name="bday">
  <input type="submit">
</form>
</body>
</html>
```

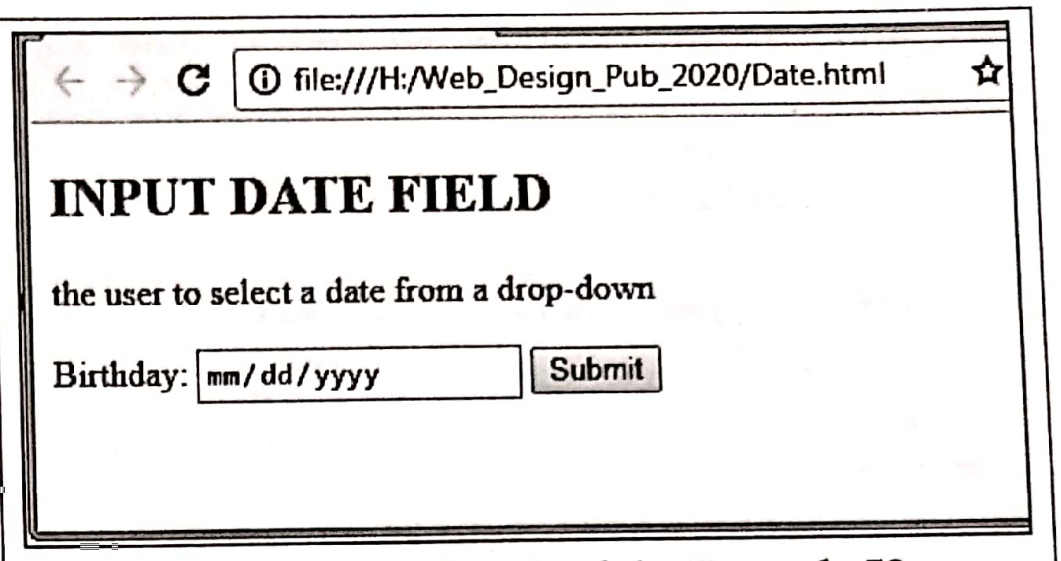


Figure 2.51: Result of the Example 52

By running the above code in a preferred browser user can get the following output as shown in above image.

Input Type E-mail: The e-mail input type allows the user to enter e-mail address. It is used in combination with the required attribute, the browser may look for the patterns to ensure a properly-formatted e-mail address should be entered.

Example 53:

```
<!DOCTYPE html>
<html>
<body>
<h2>Email Input type</h2>
<p></p>
<form action="/action_page.php">
   <b>Enter E-mail Address:</b>
  <input type="email" name="email">
  <input type="submit">
</form>
</body>
</html>
```

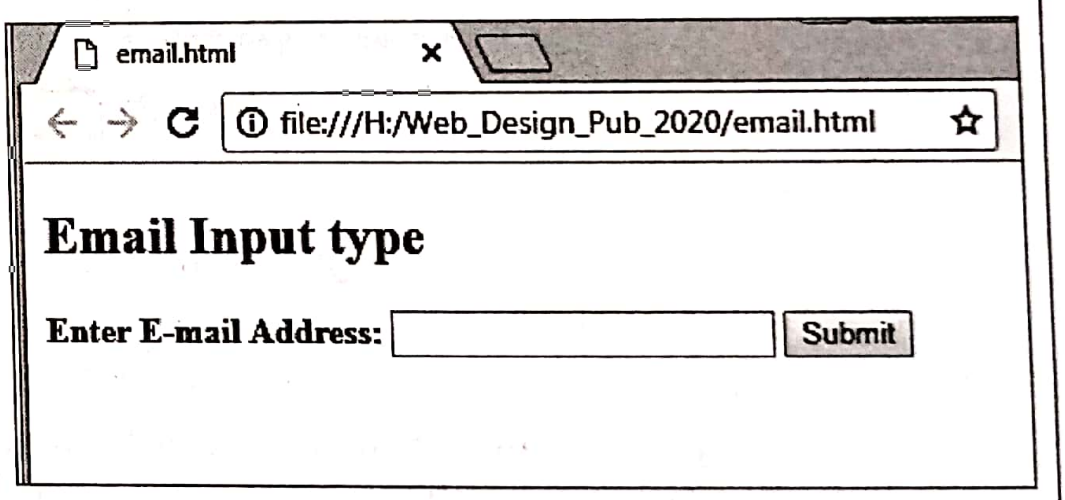


Figure 2.52: Result of the Example 53

By running the above code in a preferred browser user can get the following output as shown in above image.

Input Type Number: The input type number can be used for entering a numerical value. You can also restrict the user to enter only acceptable values using the additional attributes **min, max,** and **step.**

The following example will allow you to enter a numeric value between 1 to 10.

Example 54:

```
<!DOCTYPE html>
<html>
<body>
<h2>Input Type Number Element</h2>
```

(Contd...)

Scanned with OKEN Scanner

```
<p>You can also set restrictions on what numbers are accepted and can use the
min and max attributes to add in the input field</p>
<style>
    input[type="number"]:valid{
        outline: 2px solid green;
    }
    input[type="number"]:invalid{
        outline: 2px solid red;
    }
</style>
<form action="/action_page.php">
  Quantity (between 1 and 10):
  <input type="number" name="quantity" min="1" max="10">
  <input type="submit">
</form>
</body>
</html>
```

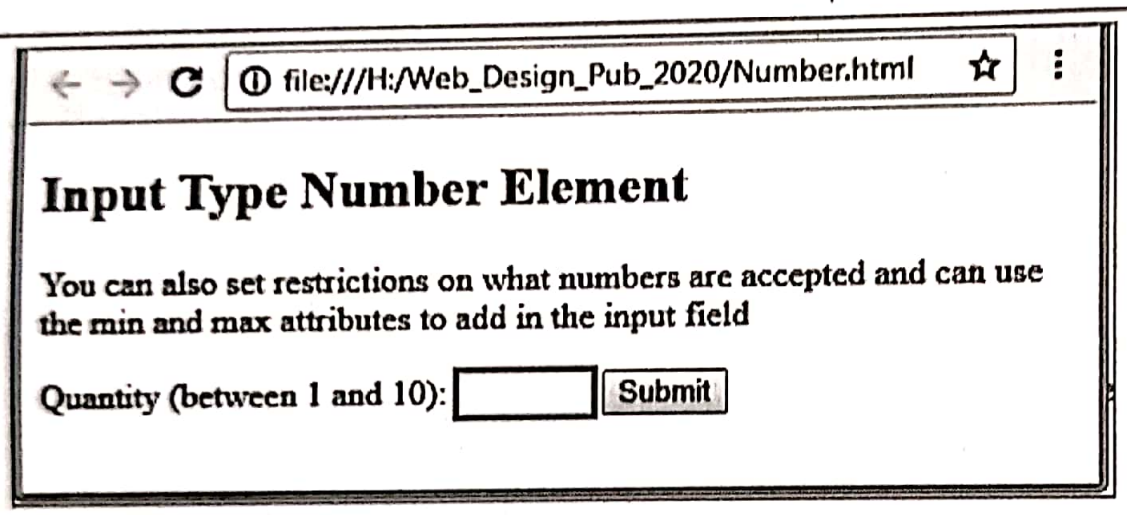


Figure 2.53: Result of the Example 54

By running the above code in a preferred browser user can get the following output as shown in above image.

Input Type Range: The range input type can be used for entering a numerical value within a specified range. Default range is **0 to 100**. However, you can set restrictions on what numbers are accepted with the **min**, **max**, and **step** attributes:

Example 55:

```
<!DOCTYPE html>
<html>
<body>
<h2>Input type Range Element</h2>
<p>The input element, having the "range" value in its type attribute,
represents a field for the input of a number inside a range<p>
<form action="/action_page.php" method="get">
  Age:
  <input type="range" name="Age" min="0" max="10">
  <input type="submit">
</form>
</body>
</html>
```

Scanned with OKEN Scanner

By running the above code in a preferred browser user can get the following output as shown in below image.

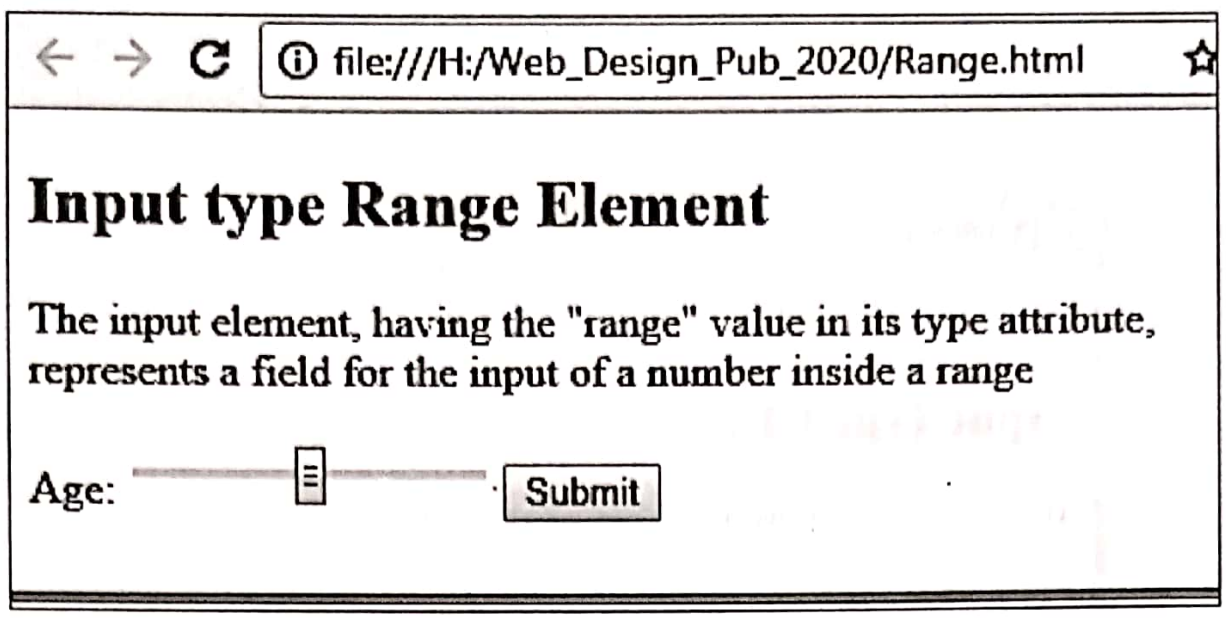


Figure 2.54: Result of the Example 55

Input Type Time: The **<input type="time">** allows the user to select a time.

Example 56:

```
<!DOCTYPE html>
<html>
<body>
<h2>Input Type Time Element</h2>
<p>In the html 5
 <strong><time></strong> tag is used to
add time or date or both date and time
in HTML page. </p>
<form action="/action_page.php">
  Select a time:
  <input type="time" name="usr_time">
  <input type="submit">
</form>
</body>
</html>
```

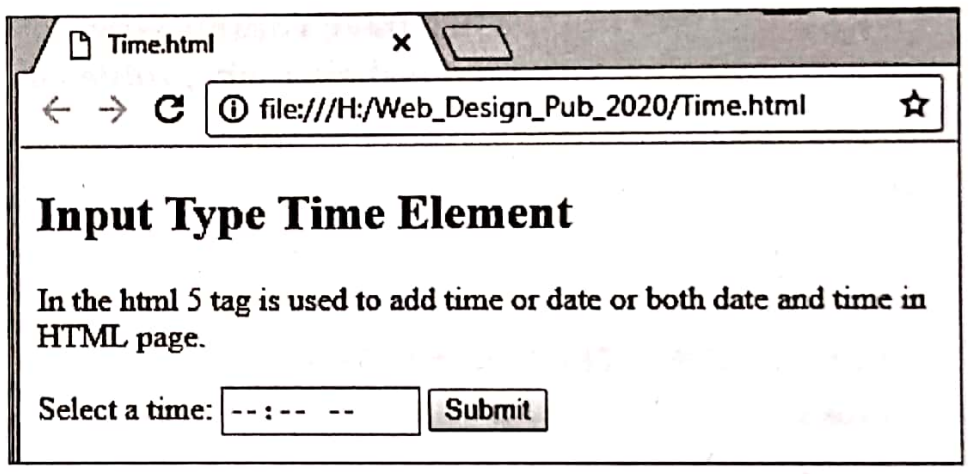


Figure 2.55: Result of the Example 56

By running the above code in a preferred browser user can get the following output as shown in above image.

Input Type Url: The **<input type="url">** is used for input fields that should contain a URL address. You can use the multiple attribute to enter more than one URL.

Example 57:

```
<!DOCTYPE html>
<html>
<body>
<h2>Input Type URL Element</h2>
<p>The input type url is used for input files contains a URL address:</p>
<form action="/action_page.php">
 <b>Enter URL Address: </b>
```

Scanned with OKEN Scanner

```
  <input type="url" name="homepage">
  <input type="submit">
</form>
</body>
</html>
```

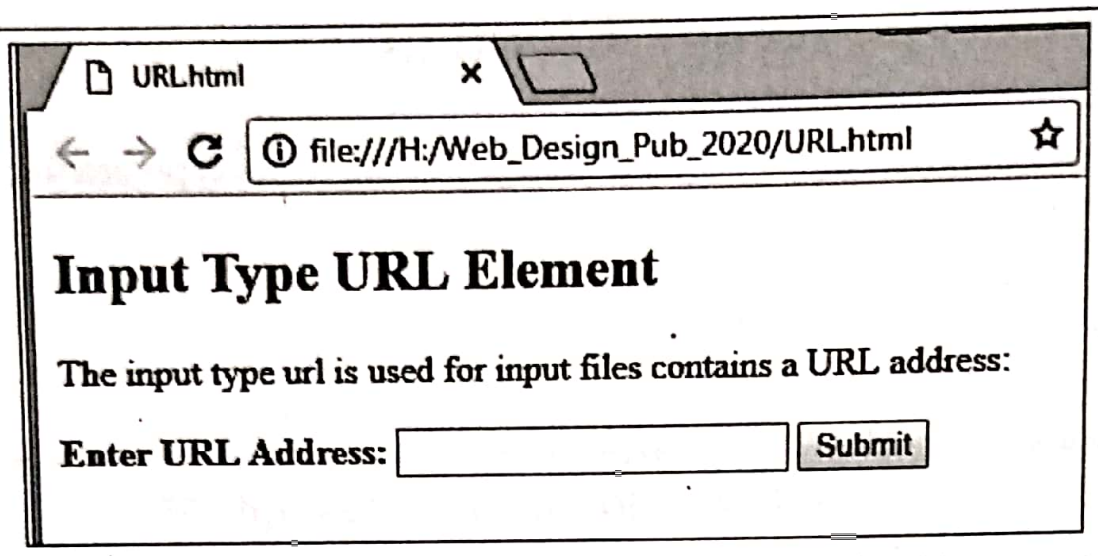


Figure 2.56: Result of the Example 57

By running the above code in a preferred browser user can get the following output as shown in above image.

2.10.4 HTML Form Program

Registration form in HTML 5 have various types are used intensively on pre-registration of website to allow for user or visitor can create their own profile on your website. The corresponding output as shown in Figure 2.57.

Example 58:

```
<html>
<head>
<title>HTML Table</title>
</head>
<body>
<form method="" action="">
<table border="1" align="left" width="400" bgcolor="yellow" >
<caption><b>Registration form</b></caption>
<tr>
<th>Enter Frst name</th>
<td><input type="text" name="fn" id="fn1" maxlength="10" title="enter first
name"
       placeholder="enter first name" required/></td>
</tr>
<tr>
<th>Enter last name</th>
<td><input type="text"/></td>
</tr>
<tr>
<th>Enter password</th>
```

(Contd...)

Scanned with OKEN Scanner

```
<td><input type="password"/></td>
</tr>
<tr>
<th>Re-enter password</th> <td><input type="password"/></td> </tr>
<tr>
<th>Enter e-mail</th> <td><input type="e-mail"/></td> </tr>
<tr>
<th>Enter mobile</th> <td><input type="number"/></td> </tr>
<tr>
<th>Enter address</th> <td><textarea rows="8" cols="20"></textarea></td> </tr>
<tr>
<th>Select your gender</th> <td>
male<input type="radio" name="g" value="m"/>
female<input type="radio" name="g" value="f"/> </td>
</tr>
<tr>
<th>Select your hobbies</th> <td>
hobby1<input type="checkbox" name="x[]" value="h"/>
hobby2<input type="checkbox" name="x[]" value="h2"/>
hobby3<input type="checkbox" name="x[]" value="h3"/> </td>
</tr>
<tr>
<th>Select your DOB</th> <td><input type="date"/></td> </tr>
<tr>
<th>Select Country</th> <td>
<select name="country">
<option value="" selected="selected" disabled="disabled">Select your
country</option>
<option value="1">India</option>
<option value="2">Pakistan</option>
</select>
</td>
</tr>
<tr>
<th>Upload your pic</th> <td><input type="file"/></td> </tr>
<tr>
<td colspan="2" align="center"><input type="submit" value="Save My Data"/>
<input type="reset" value="Reset Data"/> </td> </tr>
</table>
</form>
</body>
</html>
```

By running the above code in a preferred browser user can get the following output as shown in below image.

Scanned with OKEN Scanner

Registration form

Enter Frst name	enter first name	
Enter last name		
Enter password		
Re-enter password		
Enter e-mail		
Enter mobile		
Enter address		
Select your gender	male ○ female ○	
Select your hobbies	hobby1 ☐ hobby2 ☐ hobby3 ☐	
Select your DOB	mm/dd/yyyy	
Select Country	Select your country ▾	
Upload your pic	Choose File No file chosen	
	Save My Data Reset Data	

Figure 2.57: Result of the Example 58

2.11 HTML Embed Multimedia

Multimedia is content that uses a combination of different content forms such as text, images, animations and interactive content. Multimedia can be added to web pages using the **<object>** element or the **<embed>** element. You can add music or video into your web page. The easiest way to add video or sound to your web site is to include the special HTML tag called **<embed>**. You do not need to have any ActiveX, Java VM, VBscript or JavaScript to support this **<embed>** tag. *For example*, use <embed> to display a movie of your choice, and **<noembed>** to display a single JPG image.

Here is a simple example to play embed a midi file:

Example 59:

```
<!DOCTYPE html>
<html>
<head>
<title>HTML embed Tag</title>
</head>
<body>
<embed
src="h:/web_design_pub_2020/am1.mp3"
width="100%" height="60" >
<noembed><img        src="yourimage.gif"
></noembed>
</embed>
</body>
</html>
```

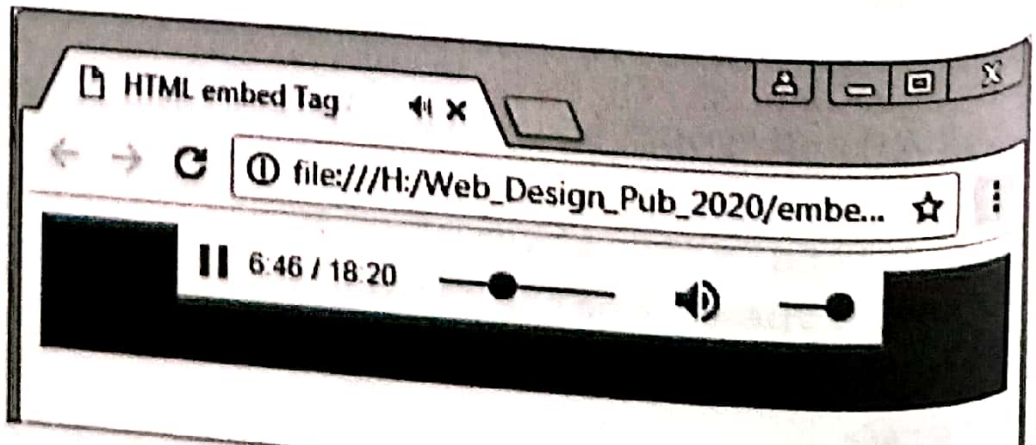


Figure 2.58: Result of the Example 59

Scanned with OKEN Scanner

By running the above code in a preferred browser user can get the following output as shown in above image.

Attributes:

Following is the list of important attributes for <embed> element.

- **Align:** Determines how to align the object. It takes either center, left or right.
- **Autostart:** Indicates if the media should start automatically. Netscape default is true, Internet Explorer is false.
- **Loop:** Specifies if the sound should be played continuously (set loop to true), a certain number of times (a positive value) or not at all (false). This is supported by Netscape only.
- **Playcount:** Specifies the number of times to play the sound. This is alternate option for loop if you are usiong IE.
- **Hidden:** Defines if the object shows on the page. A false value means no and true means yes.
- **Height:** Height of the object in pixels or en.
- **Width:** Width of the object in pixels or en.
- **Pluginspage:** Specifies the URL to get the plugin software.
- **Name:** A name used to reference the object.
- **Src:** URL of the object to be embedded. This can be any recognizable by the user's browser. It could be .mid, .wav, .mp3, .avi and so on).
- **Volume:** Controls volume of the sound. Can be from 0 (off) to 100 (full volume). This attribute is supported by Netscape only.

You can use various media types like Flash movies (.swf), AVI's (.avi), and MOV's (.mov) file types inside embed tag.

- **.swf files**– are the file types created by Macromedia's Flash program.
- **.wmv files**– are Microsoft's Window's Media Video file types.
- **.mov files**– are Apple's Quick Time Movie format.
- **.mpeg files**– are movie files created by the Moving Pictures Expert Group.

Here is a simple example to play an animated file.

Example 60:

```
<!DOCTYPE html>
<html>
<head>
<title>HTML embed Tag</title>
</head>
<body>
<embed src = "e:/sparkles.gif" width
= "220" height = "220" >
<noembed><img src = "yourimage.gif"
alt = "Alternative Media" ></noembed>
</embed>
</body>
</html>
```

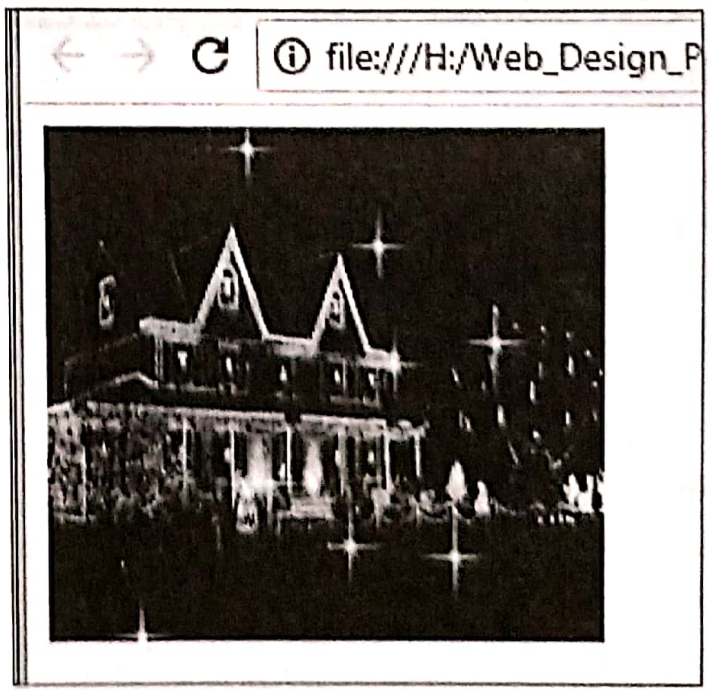


Figure 2.59: Result of the Example 60

Scanned with OKEN Scanner

By running the above code in a preferred browser user can get the following output as shown in below image.

Background Audio

You can use **<bgsound>** tag to play a soundtrack in the background of your webpage. This tag is having only two attributes **loop** and **src**. Both these attributes have same meaning as explained above.

Here is a simple example to play a small midi file:

Example 61:

```
<!DOCTYPE html>
<html>
<head>
<title>HTML embed Tag</title>
</head>
<body>
<bgsound src="/html/sound.mp3" >
<noembed><img src = "yourimage.gif" ></noembed>
</bgsound>
</body>
</html>
```

This will produce blank screen. This tag does not display any component and remains hidden.

HTML Object tag:

HTML 4 introduces the **<object>** element, which offers an all-purpose solution to generic object inclusion. The **<object>** element allows HTML authors to specify everything required by an object for its presentation by a user agent.

For example, you can embed a PDF document in an HTML document as follows:

Example 62:

```
<!DOCTYPE html>
<html>
   <head>
      <title>HTML embed Tag</title>
   </head>
   <body>
      <object
data="H:/Web_Design_Pub_2020/digital.pdf"
type="application/pdf"
  width="300" height="200">
  alt:<a href="H:Web_Design_Pub_2020/
         digital.pdf">test.pdf</a>
</object>
   </body>
</html>
```

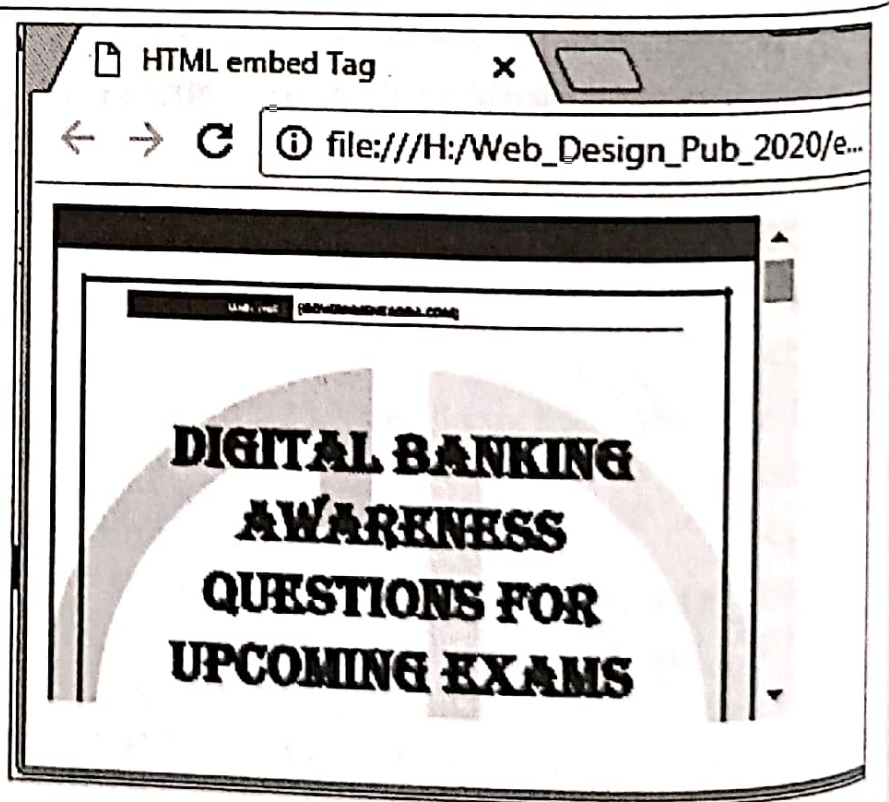


Figure 2.60: Result of the Example 62

By running the above code in a preferred browser user can get the following output as shown in above image.

Scanned with OKEN Scanner

2.12 HTML Layout

To create a website layout to position the various elements that make a web page in a well-structured manner and give attractive look to the website.

In most websites on the internet it displays their content in multiple rows and columns, formatted like a magazine or newspaper to provide the users a better reading and writing. This can be easily achieved by using the HTML tags, such as **<table>**, ** **, **<header>**, **<footer>**, **<section>**, etc. and adding some CSS styles to them.

HTML offers several semantic elements that define the different parts of a web page:

- **<header>:** Defines a header for a document or a section.
- **<nav>:** Defines a container for navigation links.
- **<section>:** Defines a section in a document.
- **<article>:** Defines an independent self-contained article.
- **<aside>:** Defines content aside from the content (like a sidebar).
- **<footer>:** Defines a footer for a document or a section.
- **<details>:** Defines additional details.
- **<summary>:** Defines a heading for the <details> element.

2.12.1 Layout-using Tables

Table provides the simplest way for creating layouts in HTML using HTML **<table>** tag. This process of putting the contents such as text, images, and so on into rows and columns.

The following layout is created using an HTML table with 3 rows and 2 columns – the first and last row spans both columns using the table's colspan attribute:

Example 63:

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Layout using Tables</title>
</head>
<body>
<table width = "90%" border = "2">
<tr>
<td colspan = "2" bgcolor = "#b5dcb3">
 <h1>Medical Online Journal</h1>
 </td>
 </tr>
<tr valign = "top">
<td bgcolor = "#aaa" width = "50">
<b>Useful Links</b><br />
Author Role<br />
Editor Role<br />
Reviewer Role </br>
Publisher Role</td>
```

(Contd...)

Scanned with OKEN Scanner

```
<td bgcolor = "#eee" width = "100" height = "200">
<h3>Indian Journal of Medical Research</h3>
<h3>Submit Manuscript</h3>
<h3>About the Journal</h3>
</td>
</tr>
<tr>
<td colspan = "2" bgcolor = "#b5dcb3">
<left>
Copyright © 20019 Medical Journal.com
</left>
</td>
</tr>
</table>
</body>
</html>
```

By running the above code in a preferred browser user can get the following output as shown in below image.

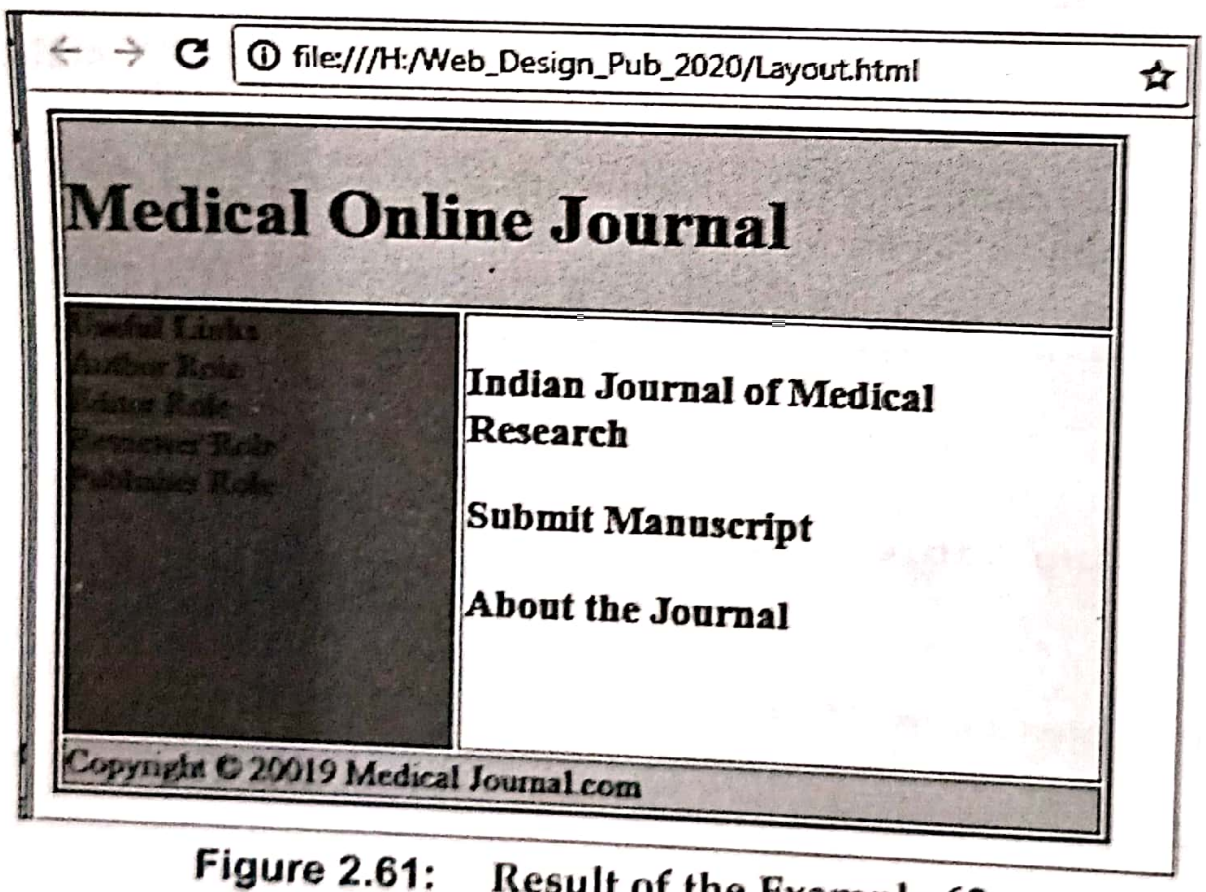


Figure 2.61: Result of the Example 63

2.12.2 Div Based Layout

The (stands for division) element used for marking out a block of content, or set of other elements inside an HTML document. The following example uses the div elements to create a multiple column layout. It will produce the same result as in the previous example that uses table element:

Example 64:

```
<body>
<body>
<div style = "width:90%">
```

(Contd...)

```
<div style = "background-color:#b5dcb3; width:100%;">
<h1>Medical Online Journal</h1>
</div>
<div style = "background-color:#aaa; height:200px; width:150px; float:left;">
<div><b>Useful Links</b></div>
Author Role<br />
Editor Role<br />
Reviewer Role </br>
Publisher Role
</div>
<div style = "background-color:#eee; border: 2px; height:200px; width:300px;
float:left;" >
<h3>Indian Journal of Medical Research</h3>
</div>
<div style = "background-color:#aaa; height:200px; width:95px; float:right;">
<div><b>Journals Topics </b></div>
Cardiology<br />
Endocrinology<br />
Genetics<br/>
Neurology
</div>
<div style = "background-color:#b5dcb3; clear:both">
<left>
Copyright © 2019 Medical Journal.com
</left>
</div>
</div>
</body>
</html>
```

By running the above code in a preferred browser user can get the following output as shown in below image.

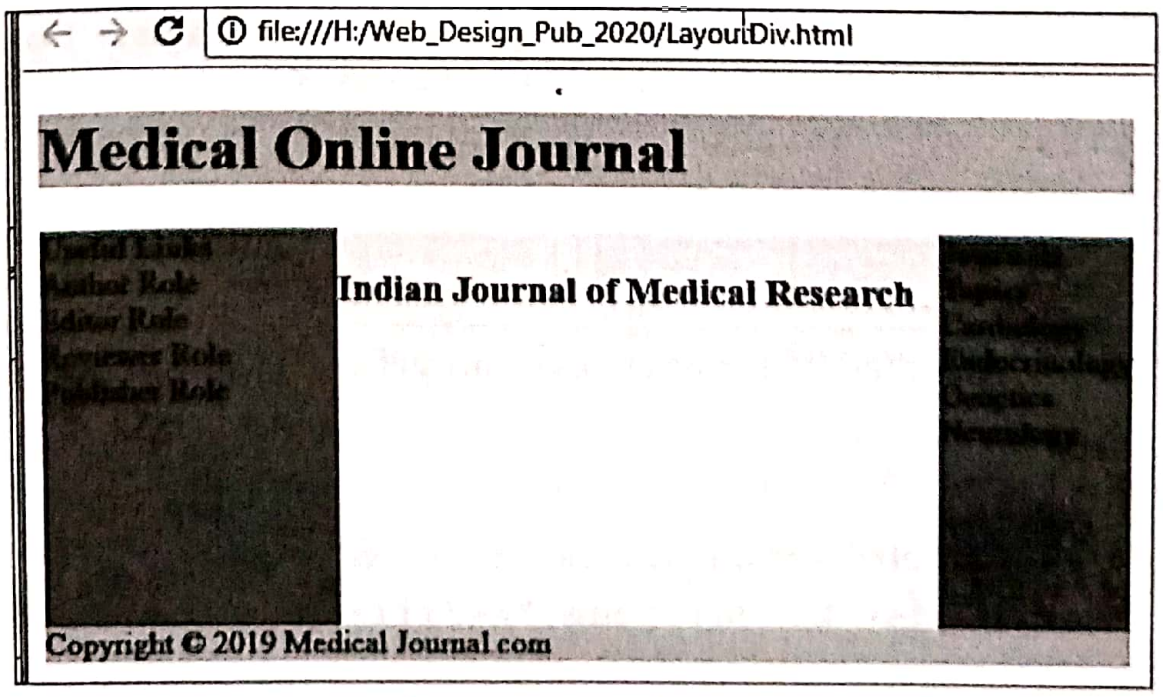


Figure 2.62: Result of the Example 64

Scanned with OKEN Scanner

2.12.3 HTML Iframe

An iframe is used to display a web page within another web page. An HTML iframe are defined with the **<iframe>** tag.

To develop the Iframes user can make use of **<iframe>** tag, inside the **<iframe>** tag use **<frameset>** instead of the **<body>** tag. **<frameset>** tag can be use any were in the document, by using **<iframe>** tag then it will create the a rectangular region for the document.

```
<iframe src="URL"></iframe>
```

Here, **src** attribute specifies the web address (URL) of the inline frame page.

Following are some of the attributes of iframes in HTML.

Attribute	Description
name	Used to give names to the frames, which are useful while loading one frame into another frame.
src	Indicated the loaded file name inside a frame.
frame border	Used to give the frame border and overrides the frame border value.
margin height	Defines the space between the two frames.
margin Width	Specify the space or width between the two frames.
noresize	User can resize the frame by clicking and dragging border of the frames.

Set Height and Width

Use the height and width attributes to specify the size of the iframe. The height and width are been specified in pixels by default:

```
<iframe src="file.htm" height="200" width="300"></iframe>
```

Example 65:

```
<!DOCTYPE html>
<html>
<body>
<h2>Example of HTML Iframes</h2>
<p>Iframes are used to display a web pages in web page. Use the height and
width attributes to specify the size of the iframe:</p>
<iframe src="E:/Digital.pdf" height="350" width="450"></iframe>
<p>Document content also go here...</p>
</body>
</html>
```

By running the above code in a preferred browser user can get the following output as shown in below image.

Remove the Border

To remove the border, add the style attribute and use the CSS border property:

```
<iframe src="file.htm" style="border:none;"></iframe>
```

Scanned with OKEN Scanner

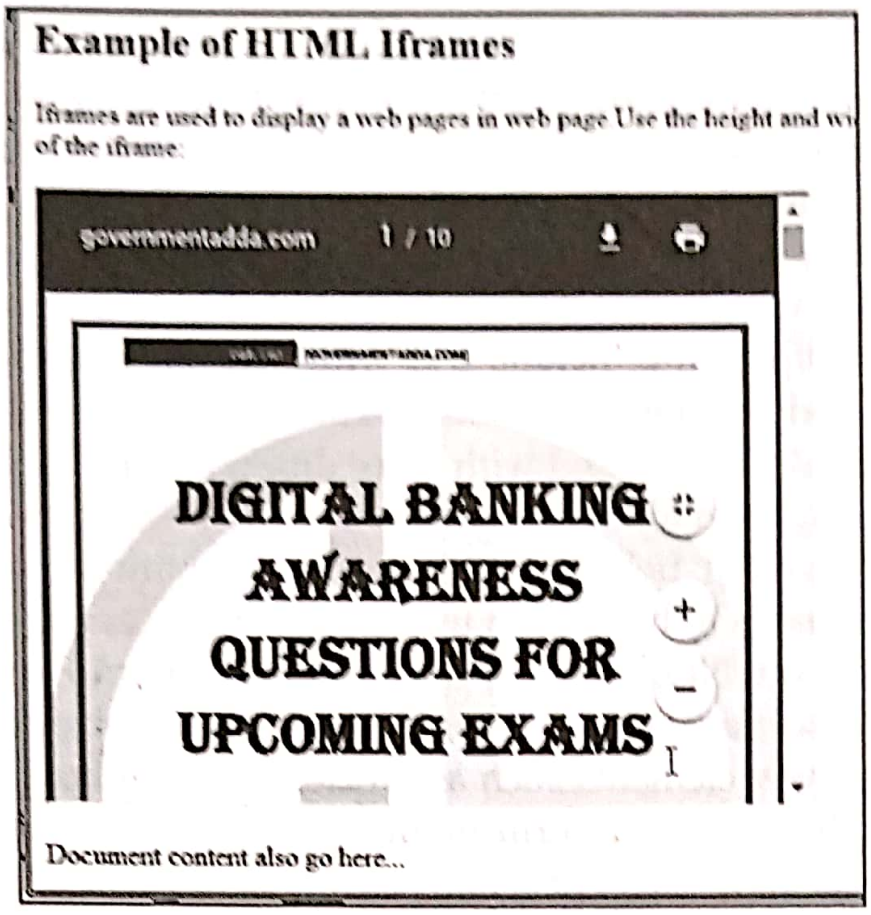


Figure 2.63: Result of the Example 65

Target for a link

You can set a target frame for a link by using **iframe**. Your specified target attribute of the link must refer to the name attribute of the iframe. An iframe can be named using the name attribute. This implies that when a link with a target attribute with that name as value is clicked, the linked resource will open in that iframe.

An iframe are be used as the target frame for a link.

The target attribute of the link must refer to the name attribute of the iframe:

```
<iframe src="file1.html" name="myFrame"></iframe>
```

Example 66:

```
<!DOCTYPE html>
<html>
<body>
<h2>Iframe Link</h2>
<iframe height="300px" width="100%"
src="d:/indian
   journal.html"
name="iframe_a"></iframe>
<p><a                                href="
http://www.ijmr.org.in/
   text.asp?2019/150/5/425/275152"
      target="iframe_a">Medical
           journal</a></p>
</body>
</html>
```

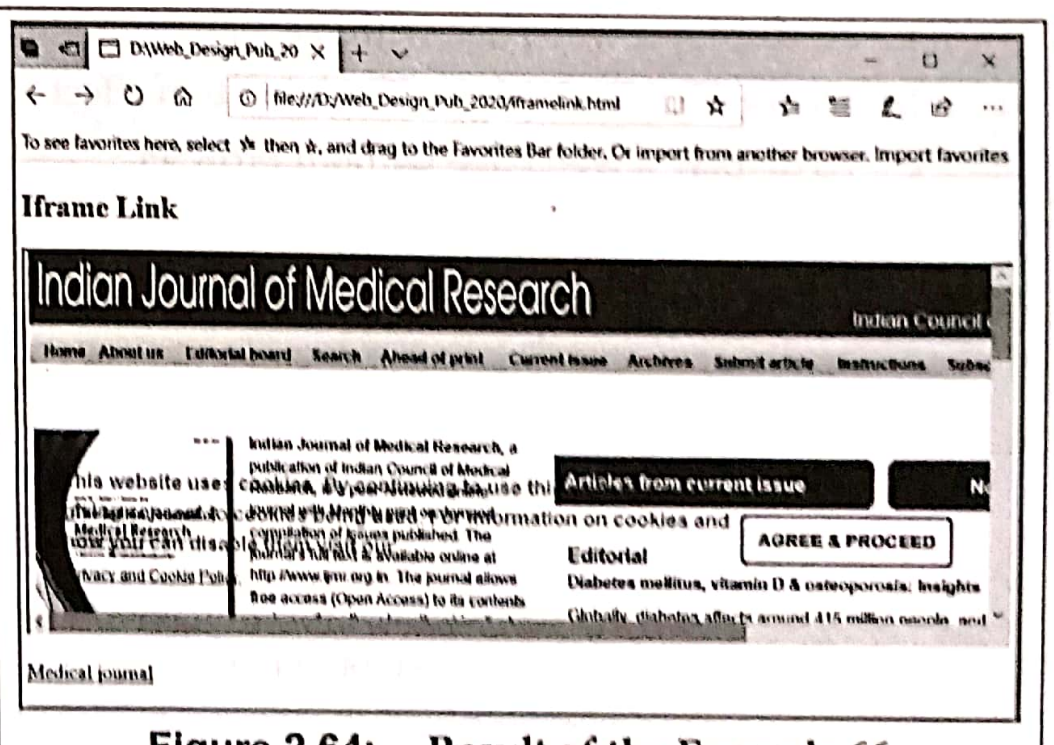


Figure 2.64: Result of the Example 66

Conclusion

The <HTML> is a markup language, which is used by the browser to manipulate text, images and other content to display it in required format. The chapter begins with basic structure of an HTML document. It

Scanned with OKEN Scanner

consists of 5 elements: <!DOCTYPE>, <html>,<head>,<title> and <body> and followed by Elements of Head section, which provide extra information in the document. Next, we will discuss formatting tags that you can use to make some text on the web pages to appear differently than normal text, for example, you can use the tag <b> to make the text bold, <i> to make the text italic. Next, you learn, anchor tag such as href, name and the target attribute are three parts of anchor tag, which is used to define the beginning and end of a hyperlink link. Next, we will discuss Tables in html is used to display data in tabular form and its attributes border, cellspacing, cellpadding, height and width to manage the layout of the page such as header section, navigation bar, body content and footer section. Next, you will learn HTML lists are used to specify lists of information in the document and with three different types of html lists such as ordered list, unordered list and description list. Then followed by HTML form is a section of a document which contains controls such as text fields, password fields, checkboxes, radio buttons, submit button and son on. It facilitates the user to enter data that is to be send to the server for processing such as name, e-mail address, password, phone number and so on. Next, we will have discussed html Frames are used to divided your browser window into multiple sections and can load a separate html document. Next, you will learn introduction to HTML 5 and its new elements such as article, aside, audio and video tag and followed by HTML 5 form validation for validating html form input elements before submitting them to a ser. It has built-in validation rules such as new elements, new attributes and new form controls for better and interactive form. Lastly, html layout specifies a way in which the web pages can be arrange using CSS and JavaScript for creating layouts for responsive and dynamic website designing. The next chapter, you will learn Cascading Style Sheet is use to define styles for your web pages.

REVIEW QUESTIONS WITH ANSWERS

Multiple Choice

1.1 Which from the following is not a type of screen frames in HTML?

a. iframe
b. noframe
b. frameset
d. uframe

1.2 What is the use of iframe in HTML?

a. To display a web page within a web page
b. To display a web page with animation effect
c. To display a web page without browser
d. All of the above

1.3 What an article element contains?

a. Audio
b. Text or embedded content
c. Image
d. Video

1.4 Which of the following HTML 4 attributes is removed in HTML5?

a. Text
b. Rules
c. Link
d. All of the mentioned

1.5 Which of these tags are all <table> tags?

a. <table><head><tfoot>
b. <table><tr><td>
c. <table><tr><tt>
d. <thead><body><tr>

1.6 How can you open a link in a new browser Window?

a. <a href="url" new>
b. <a href="url" target="new">
c. <a href="url" target="_blank">
d. <a href="url" target="">

Scanned with OKEN Scanner

1.7 <b> tag makes the enclosed text bold. What is other tag to make text bold?

a.
b. <dar>
c. <block>
d. <emp>

1.8 What should be the first tag in any HTML document?

a. <head>
b. <title>
c. <html>
d. <document>

1.9 Which tag allows you to add a row in a table?

a. <td> and </td>
b. <dr> and </dr>
c. <th> and </th>
d. <tr> and </tf>

1.10 How can you make a bulleted list?

a. <dl>
b. <nl>
c.
d.

1.11 How can you make a numbered list?

a. <dl>
b.
c. <list>
d.

1.12 What is the correct html for making a hyperlink?

a. Book Store
b. Book Store
c. https://bpbonline.com</a
d. url=https://bpbonline.com>Book Store

1.13 Choose the correct HTML tag to make a text italic.

a.
b. <italics>
c. <italic>
d. <i>

1.14 What does vlink attribute mean?

a. Visited link
b. Virtual link
c. Very good link
d. Active link

1.15 Which tag creates a check box for a form in HTML?

a. <checkbox>
b. <input type="checkboc">
c. <input=checkbox>
d. <input checkbox>

1.16 Which of the following is an attribute related to font tag?

a. size
b. face
c. color
d. All of the above

1.17 Which tag is used to list individual items of an ordered list?

a. LI
b. OL
c. UL
d. None of the above

1.18 Which of the following is not a type of attribute for input tag?

a. day
b. week
c. month
d. time

1.19 Which of the following defines a group of related options in a drop-down list.

Scanned with OKEN Scanner

a. <form>
b. <optgroup>
c. <output>
d. <option>

1.20 Which element does not support autofocus attribute?

a. <input>
b. <select>
c. <textarea>
d. <base>

State whether the following statements are True or False

2.1 To insert a video, you use a tag and set its src attribute to a local or remote URL containing a playable movie.
2.2 <select> tag creates a check box for a form in HTML.
2.3 The cellpadding attribute is used with <td> tag.
2.4 The default value of border attribute is 1.
2.5 The cellspacing attributes tells the browser how much space to include between the walls of the table.
2.6 <!doctype html> is the correct syntax of doctype in HTML5.
2.7 The align and valign are the attributes uses with <td> tag to set the alignment of data in a cell.
2.8 To insert a video, you use a video tag and set its scr attribute to a local or remote URL containing a playable movie.
2.9 Radio buttons are small, round buttons that enable users to select a single option from a list of choices.
2.10 The base tag and frame tags can be used to embed an inline frame within a web page.

3. Match the following:

3.1	The acronym w3c stands for	a.	Table header
3.2	In the <th> tag, th stands for	b.	Combining
3.3	Which attribute of <td> tag is used to merge two or more columns to form a single column	c.	Wide World Web
3.4	Two or more cells in a table on a web page is called	d.	<table> tag
3.5	The correct html code to right align the content inside a table cell	e.	Colspan
3.6	Border, frame, cellspacing, cellpadding, align are the attributes of	f.	Font and style
3.7	HTTP	g.	<td alignt="right">
3.8	The attribute in an input element inserts a button that, when click will clear the contents	h.	Scr
3.9	_____ attributes specify the layout of frames within the browser window.	i.	type="reset"
3.10	The attribute of <iframe> tag used to point the location of a page is	j.	<body> tab
		k.	Reuest-Responser protocol

Scanned with OKEN Scanner

4. Fill in the blanks

a. Base and target	b. colspan	c. Nav
d. Cells	e. Header	f. src
g. Rowspan	h. Underline the text	i. Pixels
j. iframe	k. Frameset	l. Display the bulleted list

4.1 HTML 5 documents may contain a _____ element, which is used to set the header section of a document.

4.2 _____ contain the navigation menu, or other navigation functionality for the page.

4.3 Use the _____ tag and _____ attribute to force all the links on a page to load in a particular frame.

4.4 The _____ attribute of the frame tag tells the browser which html page to load into that frame.

4.5 The _____ attribute is used to span two or more columns in a table.

4.6 The _____ attribute is used to combine the cells vertically.

4.7 The height and width attributes of <table> tag are specified in terms of _____ or percentage.

4.8 Cellspacing is used to create space between different _____ in a table.

4.9 <UL>...</UL> tag is used to _____.

4.10 A group of frames is called a _____.

Descriptive Questions with Answer

5.(a) What is HTML?

Answer

HTML is short for HyperText Markup Language and is the language of the World Wide Web. It is the standard text formatting language used for creating and displaying pages on the Web. This means that an HTML document, written in plain text, is used to describe the structure and content of web pages, with links to other pages.

5.(b) What is the difference between HTML elements and tags?

Answer

Elements: Each part of a web page, such as a paragraph, an image, a link or anything else you can interact with, is an element. Each type of element has its own behavior. *For example*, you can click on links, or type in text boxes.

Tags: An HTML document is a simple, plain text document, which we are able to open with any text editor on our computer. When we open one, we will see the document is made up of tags, which are keywords surrounded by angled brackets, each of which describes an HTML element. Here, we can see HTML tags telling the browser how to render the text element inside:

This text is surrounded by HTML tags!

Most tags have opening and closing tags. The opening tag is written with the tag name in angled brackets, like <tagname> whereas the closing tag adds a forward slash: </tagname>. Anything between these opening and closing tags is considered to be contents of that tag.

5.(c) What are attributes and how do you use them?

Answer

Each tag can also have additional attributes, which change the way the tag is display. For example, an <input> tag has a type attribute, which you can use to specify whether it's a text field, checkbox, radio button or one of many more options.

Attributes are specified directly after the name of the tag, inside the two angled brackets. They should only ever appear in opening tags or in self-closing tags - they can never be in closing tags. They are followed by

Scanned with OKEN Scanner

an equals sign = and the chosen value in double quotes ". Do not include any spaces before and after the equals sign. Multiple attributes can be defined on a single tag, separated by a space.

```
<!-- Text field -->
<input type="text" />
<!-- Checkbox -->
<input type="checkbox" />
<!-- Radio button -->
<input type="radio" value="on" />
```

6.(a) What new features were added to HTML5?

Answer

It introduced a number of semantic elements, which is to say elements that convey meaning. Some of the new semantic elements are **<header>**, **<footer>**, **<section>**, and **<article>**. They are semantic in that they are not just simple containers, but they tell the browser more about their contents.

There are additional form element types, like "number", "date", "calendar" and "range". Video and audio elements have also been added, as well as new graphic elements, such as <svg> and <canvas>.

6.(b) What is Cell Spacing and Cell Padding?

Cell Spacing is referred to space/gap between the two cells of the same table.

Cell Padding is referred to the gap/space between the content of the cell and cell wall or Cell border.

Example:

```
<table border cellspacing="3">
<table border cellpadding="3">
<table border cellspacing="3" cellpadding="3">
```

6.(c) Write the HTML code to create the table which is given below:

IDNO	Name	Class	DOB
110	Manju	VI Std	4-Nov-1993

Answer

The HTML code is

```
<html>
<body>
<table border="3">
<tr align="center" valign="middle">
<th>IDNO</th>
<th>Name</th>
<th>Class</th>
<th>DOB</th>
</tr>
<tr align="left" valign="bottom">
<td>110</td>
<td>Manju</td>
```

Scanned with OKEN Scanner

```
<td>VI Std</td>
<td>4-Nov-1993</td>
</tr>
</table>
</html>
```

7.(a) Which HTML tag is used to display the data in the tabular form?

Answer

The HTML table tag is used to display data in tabular form (row and column). It also manages the layout of the page, e.g., header section, navigation bar, body content, footer section. Here is the list of tags used while displaying the data in the tabular form:

Tag	Description
<table>	It defines a table.
<tr>	It defines a row in a table.
<th>	It defines a header cell in a table.
<td>	It defines a cell in a table.
<caption>	It defines the table caption.
<colgroup>	It specifies a group of one or more columns in a table for formatting.
<col>	It is used with <colgroup> element to specify column properties for each column.
<tbody>	It is used to group the body content in a table.
<thead>	It is used to group the header content in a table.
<tfooter>	It is used to group the footer content in a table.

7.(b) Explain list elements in HTML.

Answer

The list elements in HTML are:

Ordered List (<ol>): An Ordered List or ol tag are the list which will list the items in an ordered way, i.e. numbered or alphabetically.

Unordered List (<ul>): An Unordered List or ul tag is the list which will list the items in an unordered way, i.e. in bulleted format or in any other format.

Definition List (<dl>): A definition List or dl tag arranges the items in the way in which they are arranged in a dictionary.

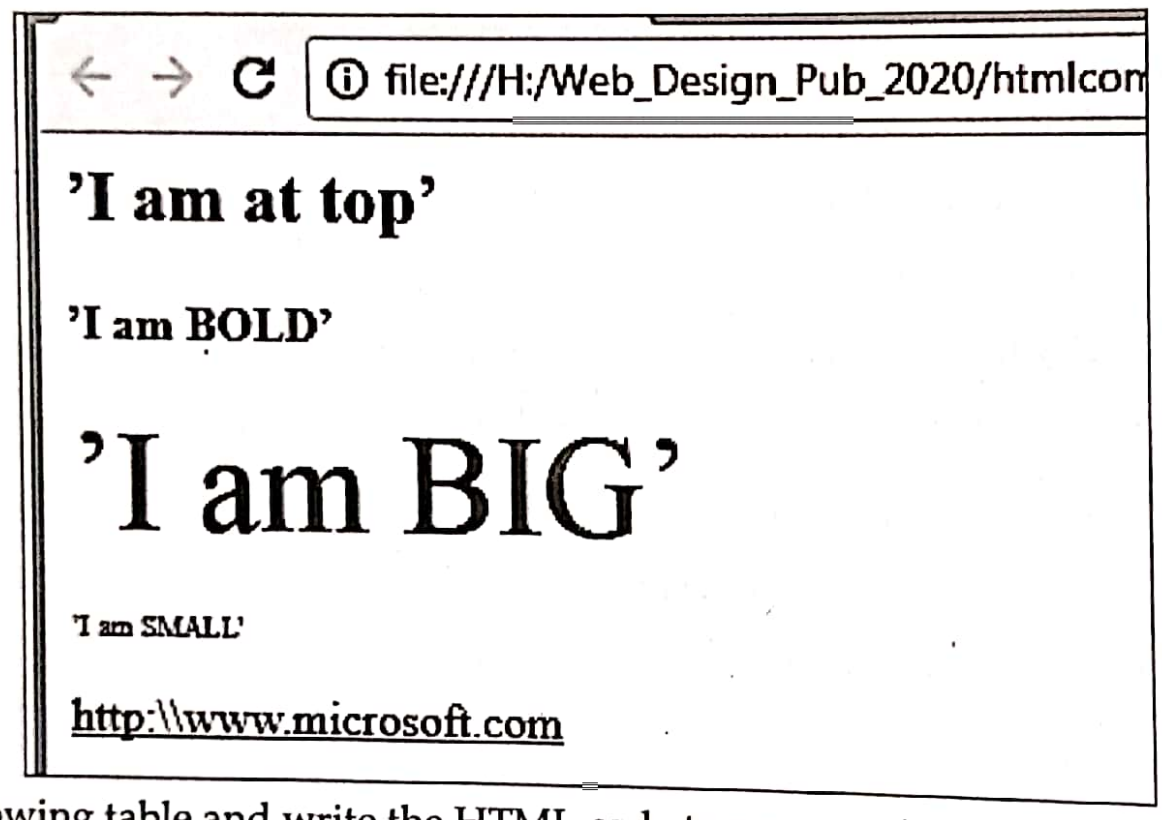
8.(a) Write HTML commands to achieve the following:

1. Put a Level two heading 'I am at top'.
2. Put 'I am BOLD' in bold face
3. Render 'I am BIG' in a larger font.
4. Render 'I am SMALL' in a smaller font.
5. Provide a link to http://www.microsoft.com

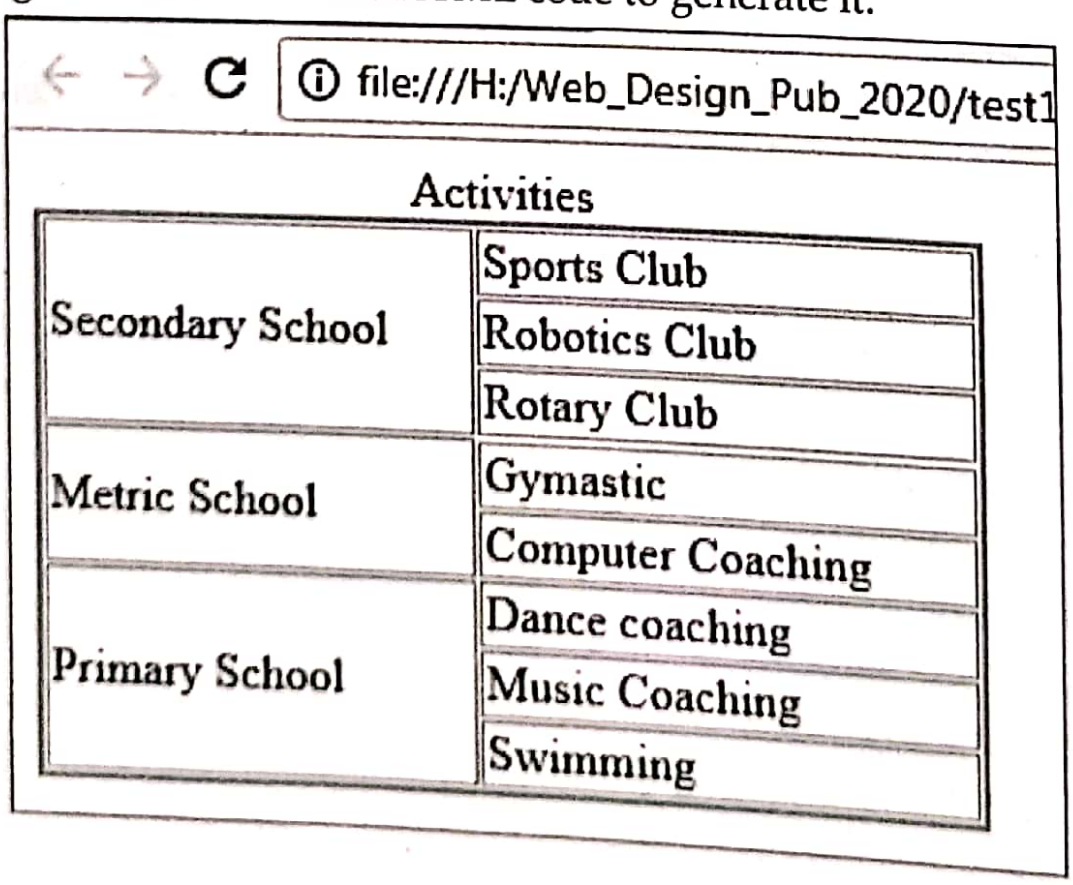
Scanned with OKEN Scanner

Answer

```
<html>
<head>
<TITLE> HTML Commands</TITLE>
</head>
<body BGCOLOR = "WHITE">
<H2>'I am at top'</H2>
<P><b>'I am BOLD'</b></p>
<p><Font size ="+7">'I am BIG'</Font></P> <p><Font size ="1">'I am
SMALL'</Font></P> <A HREF=" ">http:\\www.microsoft.com</A>
</body>
</html>
```



8.(b) Observe the following table and write the HTML code to generate it.



Activities

Secondary School	Sports Club
	Robotics Club
	Rotary Club
Metric School	Gymastic
	Computer Coaching
Primary School	Dance coaching
	Music Coaching
	Swimming

Scanned with OKEN Scanner

Answer

```
<html>
<body><basefont face="Ariel">
<table border ="2" bordercolor="organge" cellspacing="1" width="320">
<caption>Activities</caption>
<tr>
<td rowspan="3">Secondary School</td> <td>Sports Club</td> </tr>
<tr>
<td>Robotics Club</td> </tr>
<tr>
<td>Rotary Club</td> </tr>
<tr>
<td rowspan="3">Metric School</td> <tr>
<td>Gymastic</td> </tr>
<tr>
<td>Computer Coaching</td> </tr>
<tr>
<td rowspan="3">Primary School</td> <td>Dance coaching</td> </tr>
<tr>
<td>Music Coaching</td> </tr>
<tr>
<td>Swimming</td> </tr>
</table>
</body>
</html>
```

9.(a) What is the user of <Body> tag in HTML code? Write any two attribute used with this tab.

Answer

The **<body>** tag contains the contents of our document including background color, text, color, link color and page margins. Also, various mark-up elements are allowed within the body to indicate headings, paragrphs, list, hypertext and so on. These are described in the <body> documents. The two attributes are: background and bgcolor.

9.(b) (i) Name the most important element in HTML form tag. Explain.

(ii) What is the use of the Submit button with HTML form Input tag?

(iii) What is the use of HTML frames?

Answer

1. The most important form element is the input element. The input element is used to select user information. An input element can vary in many ways, depending on the type attributed. An input element can be of type text field, checkbox, password, radio button, submit button and so on.
2. A submit button is used to send form data to a server. The data is sent to the page specified in the form's action attribute.
3. With frames, we can display more than one HTML document in the same browser Window. Each HTML document is called a frame, and each frame is independent of the others.

Scanned with OKEN Scanner

Answers

1.1	d	1.2	a	1.3	b	1.4	d	1.5	b	1.6	c
1.7	a	1.8	c	1.9	d	1.10	c	1.11	b	1.12	a
1.13	d	1.14	a	1.15	b	1.16	d	1.17	a	1.18	a
1.19	b	1.20	d								
2.1	T	2.2	T	2.3	F	2.4	F	2.5	F	2.6	T
2.7	T	2.8	T	2.9	T	2.10	F				
3.1	c	3.2	a	3.3	e	3.4	b	3.5	g	3.6	d
3.7	k	3.8	i	3.9	f	3.10	h				
4.1	e	4.2	c	4.3	a	4.4	f	4.5	b	4.6	g
4.7	i	4.8	d	4.9	l	4.10	k				

OTHER TITLES OF INTEREST



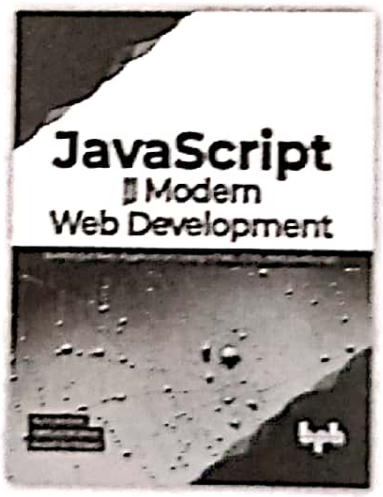
Exploring **Web Components**
ISBN: 9789389423976



HTML, JavaScript, DHTML and PHP
ISBN: 9788183330084



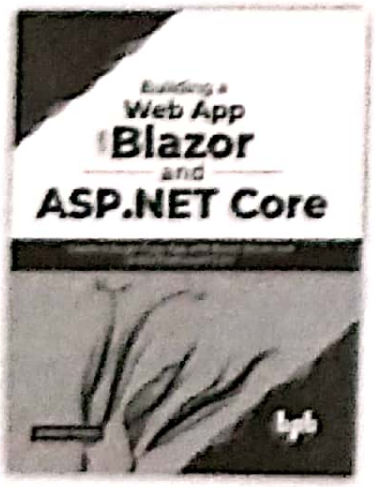
Building Modern Web Applications With **Jakarta EE, NoSQL Databases** and **Microservices**
ISBN: 9789389423341



JavaScript for Modern **Web Development**
ISBN: 9789389328721



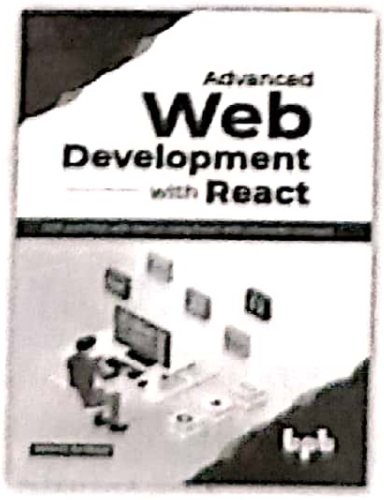
Web Designing and **Development Training** Cuide
ISBN: 9788183335799



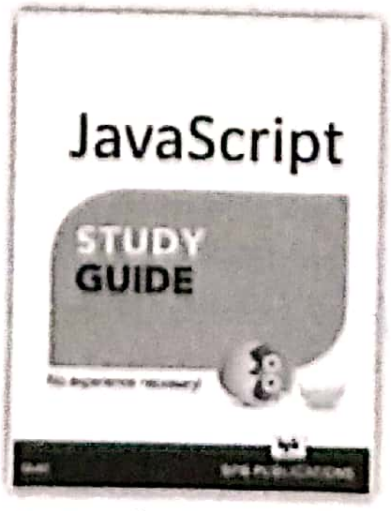
Building a Web App with **Blazor** and **ASP .Net Core**
ISBN: 9789389845457



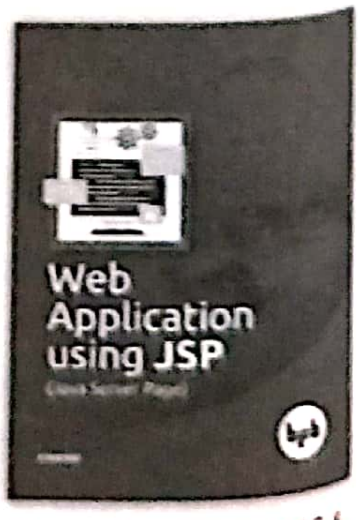
Web Technologies
ISBN: 9788183335751



Advanced **Web Development** with **React**
ISBN: 9789389423594



JavaScript Study Guide
ISBN: 9789386551016



Mastering HTML, CSS & Java Script Web Publishing
ISBN: 9789388176200

Scanned with OKEN Scanner

CHAPTER 3

Cascading Style Sheet

3.1 Introduction

Cascading Style Sheets is a style sheet language is use for describing the presentation of a document written in a markup language in HTML. Using CSS, you can control the color of the text, the style of fonts, spacing between paragraphs, how columns are sized and what background images or colors are used. The major purpose of CSS is for describing the presentation of Web pages, including colors, layout, and fonts that allows to adapt the presentation to different types of devices, such as large screens and so on. CSS is independent of HTML and can be used with any XML-based markup language. It look and formatting properties for how content should be display on web-page and give you control to make global change and apply all web page.

Structure

- Introduction to CSS and its types
- CSS Selectors
- CSS Properties
- CSS Lists and Tables

Objectives

The reader will be able to understand the following:

- Explain the purpose of CSS.
- Explain the role of CSS in websites.
- Discuss the roles of effects in website.

3.2 Introduction to CSS

CSS is an abbreviation **for Cascading Style Sheets**. CSS works with HTML and other Markup Languages (such as XHTML and XML) to control the way the content is presented. Cascading Style Sheets is a means to separate the appearance of a webpage from the content of a webpage. CSS is a recommendation of the **World Wide Web Consortium** (the **W3C**).

The concept of style sheet is very simple. First, the designer creates a standard Web page, using standard HTML tags. It is design so that it can be displayed properly in browsers that do not support style sheets. Now, suppose designer wants the **heading** (H1) should be display in **red color** and the **paragraph** (P) should be display in **blue** and **Helvetica font**. To accomplish this, the author creates style rules that format the content on the Web page in the manner that he or she chooses.

Style sheets would thus combine HTML tags (H1) with properties (such as color: red) to format each HTML tag. In the case of style sheets, an HTML tag is used as a selector. The property and value of the selector are combined into what is called a **declaration**. Following are some examples of style rules:

```
H1 {COLOUR: RED}
P {FONT-FAMILY: HELVETICA; COLOUR: BLUE}
```

In the above stylesheet tag H1 and P is used as selector and their properties such as colour and font-family is used as a declaration part.

Scanned with OKEN Scanner

The Web designer designs the page content and the style rules that define its appearance. The designer also attaches the style rules to the standard HTML document using methods through the use of external style sheet, an embedded style sheet or an inlinestyle sheet.

Cascading Style Sheet Level 1

The first implementation of Cascading Style Sheets known as CSS1 (Cascading Style Sheet level 1), enables you to specify everything from typefaces for different HTML element to font colors, background colors and graphics, margins, spacing, type style, etc. Browsers that support this type of style sheet, which include Netscape Navigator 4 and later and IE 4 and later, apply the style definition to the final appearance of the document.

Cascading Style Sheet Level 2

The next generation of style sheets, CSS2, became a formal recommendation in **May 1998**. Many CSS2 tags are supported in the latest versions of both Internet Explorer and Netscape Navigator. In the very near future, style sheets will enable you to accomplish even more exciting thing on the Web. With CSS2, XHTML and compliant browsers, you will be able to generate Web pages that target different types of media. *For example*, you will be able to design aural style sheets that speak page elements to a user while using spatial audio and surround-sound properties. You will also be able to split Web pages into multiple pages, much as you do in a word processor or page layout program.

Define CSS Comments

You can define in to your <style> section or Style Sheet using /* and */ (following example line 1 and line 3 comment define).

Example 1:

```
/* Header style start from here */
      #header {
            Color: blue; ……../* Default header color */
            Margin-left: 20px;
      }
      #header .logo {
            Font-size: 12px;
            Background-color: #FF6633;
      }
```

Advantage of CSS

The advantages of CSS are as follows:

1. **Easy Manage:** CSS with you can better manage whole webpage. CSS allow managing entire site elements looks and formatting in single style sheet file.
2. **Global Change:** CSS Style Sheet changes apply to global change for all webpage. When Style Sheet appear with webpage it is known as Cascading Style Sheet.
3. **Save time:** When you create HTML document, you define separate set attributes value in each element, but, it is limited use. But, CSS give a lot of flexibility to set properties and values either group of element or single element. So, it is benefit to avoid same code write again and again.
4. **Easy Maintain/Update:** CSS style sheet maintain easier and anytime you can edit elements properties and values.

Scanned with OKEN Scanner

5. **Three ways to write CSS:** You can write CSS styles 3 way **Inline element** line, **Internal style** write in header section, or **External style sheets** write in external **.css** extension files. External style sheets enable you to change the elements and layout style of all the pages in a Website, just by editing one single file.
6. **Page Load Faster:** Webpage with stylesheets take a bit of second to load very fastly.

3.3 Types of CSS

There are three ways to adding style to a document. These are as follows:

1. External style sheet can be linked to one or more pages, applying the same style rules to all.
2. Embed a document-wide style sheet in the <HEAD> element of the document.
3. Use Inline style exactly where the styles need to be applied.

Each of these style sheet linking is explained in further sections.

Linking External Style Sheet

External style sheets keep the style rules in a separate file, apart from the HTML Web documents. The advantage of using external style sheets is that you can apply the same style rules to more than one document in you Website. This enables you to create pages that have a consistent appearance. External style sheets also provide the advantage of enabling you to quickly change the appearance of your Web page in the future. By defining your styles in a single document and linking them to multiple pages, you only need to edit the style sheet to change the presentation style to all pages linked to it.

External style sheet consists set of rules in four part:

1. External Source link
2. Selector (element, class, id)
3. Property and
4. Value

How to write External Stylesheet

External style sheet linked to a webpage. Selector is normally HTML element (or class, id) to assign CSS properties and set suitable values.

External Style Sheet Source Link

```
<head>
<link rel="stylesheet" type="text/css" href="style.css" />
</head>
```

Property value

```
Body {
        font: 12pt; font-family: serif;
        color: black; background: white
      }
```

Although a Web page is saved with an .htm or .html extension, an external style sheet is saved with a .css extension. It defines all the common style rules that are shared in your Web documents. Then, using the <LINK> tag, you link the external style sheet to each HTML Web page.

Now suppose you link a file STYLE.CSS to the HTML document. For this, first of all you have to create a .css file for style sheet. Following is the style for text you have to write in your notepad file and then save it as STYLE.CSS.

Scanned with OKEN Scanner

```
Body {font: 12pt; font-family: serif; color:
        black; background: white}
H3 {font: 24pt; font-family: comic sans ms; color: black}
H4 {font: 16pt; font-family: sans-serif; color: black}
P {text-indent: 0.5 in}
A:link {color: yellow; text-decoration: none}
A:visited: {color: red; text-decoration: 'u'}
A:active: {color: blue; text-decoration: none}
```

After that you link this .css file to the HTML tags as shown in Example 2, using the <LINK> tag.

Various attributes are explained here with: REL attribute controls the persistent style, default styles and alternate styles. Here, you set

```
Rel = "stylesheet"
```

It forces the use of persistent styles and applies the styles in your style sheet, regardless the users' local selection.

HREF attributes is used to indicate the URL of the style sheet to use. In this example, the style sheets resides in the same directory in which HTML file is saved.

MEDIA attribute is used to provide an indication of which media the style sheet should apply to. This attribute enables the page designer to define one style for computer screens and another for printer.

TYPE attribute when set

```
TYPE = "TEXT/CSS"
```

It indicates that the linked style sheet is a cascading style sheet, but another form of style sheet certainly could be linked.

The file given below contains CSS property. This file save in **.CSS** extension for e.g. style.css. The output as seen on Internet Explorer of the coding in Example 2 as shown in Figure 9.1.

CSS Style

```
body {
  background-color: #C0C0C0;
}

.main {
     text-align:center;
 }
.BPB  {
  color:#808080;
  font-size:40px;
  font-weight:Bold;
}

#books {
 font-style:bold;
 font-size:20px;
}
```

Now, is the html file that is making use of the created external style sheet.

Scanned with OKEN Scanner

Example 2:

```
<!DOCTYPE>
<html>
<head>
<link rel="stylesheet" href="style.css"/>
</head>
<body>
<div class="main">
<div class="BPB">BPB Publications</div>
<div id="books">Largest selling Computer Science books in Asia
</div>
</div>
</body>
</html>
```

The output of given coding in Example 2 as shown in Figure 3.1.

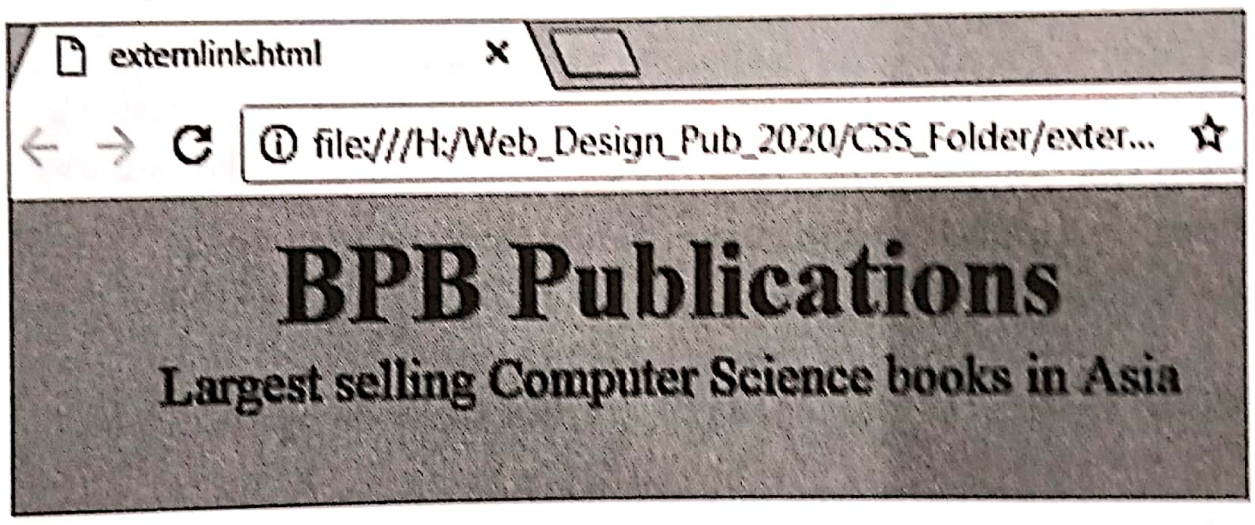


Figure 3.1: Output of Example 2

Embedding Style Sheets or Internal Style Sheet

Another way to include an external style sheet is to embed it. When you embed a style sheet, you write the style rules directly within the HTML document. You have to enclose the style rules within the <STYLE> and </STYLE> tag pairs and place this pair within the head section of the HTML document. Because multiple forms of style sheet may be included (beyond the standard css format), therefore, you should still include the TYPE attribute to indicate which format of style sheet you are using. This is regardless of the browser' support for other style sheet technologies.

One concern when including style sheets within an HTML document is that not all browsers understand style information. To avoid problem, comment out the style information by using an HTML comment tag <!---> so that the comments are not displayed onscreen by older browsers. A complete example of Embedded style sheet is given in Example 3.

Example 3:

```
<html>
<head>
<title>cascading style sheets</title>
```

(Contd...)

Scanned with OKEN Scanner

```
<style type = "text/css">
body {font: 12pt; font-family: serif; color: black;
        background: white}
H3 {font: 24pt; font-family: comic sans ms; color: black}
h4 {font: 16pt; font-family: sans-serif; color: black}
p {text-indent: 0.5in}
a:link {color: yellow; text-decoration: none}
a:visited: {color: red; text-decoration: 'u'}
a:active: {color: blue; text-decoration: none}
</style>
</head>
<body>
<h3>chapter content</h3>
<h4><a href="#dsl" >digital subscriber line</a></h4>
<h4><a href="#www" >world wide web</a></h4>
<h4><a href="#inet" >internet</a></h4>
<h4><a href="#lan" >local area network</a></h4>
<h4><a href="#wan" >wide area network</a></h4>
<h4><a href="#hlink">hyperlink</a></h4>
<h4><a name="dsl">digital subscriber line</a></h4>
<p> here we have to write the details of digital subscriber line </p>
<h4><a name="#www" >world wide web</a></h4>
<p> here we have to write the details of world wide web
</p>
<h4><a name="#inet" >internet</a></h4>
<p> here we have to write the details of internet </p>
<h4><a name="#lan" >local area network</a></h4>
<p> here we have to write the details of local area
network </p>
<h4><a name="#wan" >wide area network</a></h4>
<p> here we have to write the details of wide area network.</p>
<h4><a name="#hlink">hyperlink</a></h4>
<p>here we have to write the details of hyperlink.</p>
</body>
</html>
```

The output of Example 3 as shown in Figure 3.2.

Inline Style Sheets

External and Embedded style sheets create a style for the whole document. However, you can add style information right down to the single element. The simplest way to add style information, but not necessarily the best is to add style rules to the particular HTML element.

Scanned with OKEN Scanner

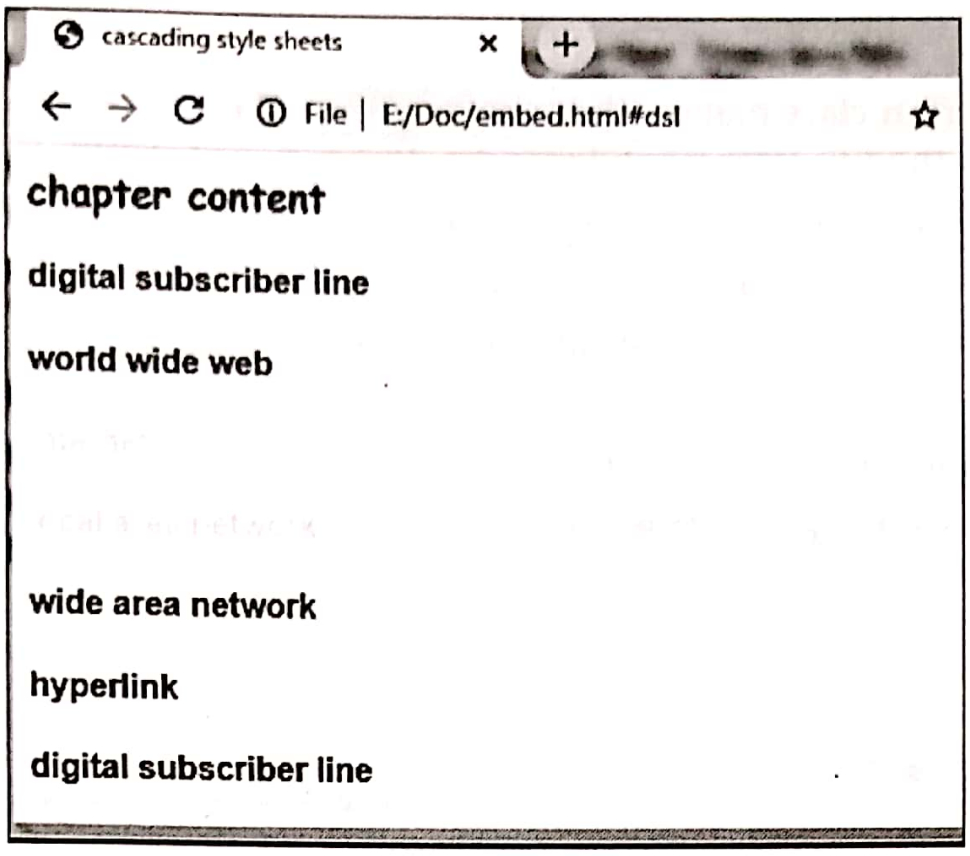


Figure 3.2: Output of the Example 3

Suppose, you want to set one particular <H1> tag to render in 22 pt, red, comic sans ms font. You could apply that style to all <H1> elements, or to a class of them by applying a embedded style. On the other hand, you could apply the style to only the tag in question. This is done with the STYLE attribute, which can be used within HTML element.

```
<P STYLE = "FONT-SIZE: 12PT; FONT-FAMILY: COMIC SANS MS; COLOR: RED">Get whatever you love otherwise you will be forced to love whatever you get.</p>
```

This sort of style information need not be hidden from a browser that may not be style-sheet-aware. This is so, because browsers ignore any attributes that they do not understand.

By using **inline style**, you can easily apply a style to a certain section or division of a document by using the **<DIV>** element. The tag **<DIV>** works well to surround block elements, such as paragraph. Setting a two paragraph portion of a document to have a yellow background with red text is shown below:

```
<DIV STYLE = "FONT-SIZE: 12PT; FONT-FAMILY: COMIC SANS MS; COLOR: RED; FONT-WEIGHT=: BOLD">
<P>Get whatever you love otherwise you will be forced to love whatever you get. This is the rule of the nature, we can't help this.</P>
<p>This is another paragraph describing the philosophy of spending happy life.</P>
</DIV>
```

<DIV> element should be used with a CLASS attribute, to name a section of text as being of a certain style as specified in a style sheet. Essentially, text surrounded by the **<DIV>...</DIV>** elements will be formatted according to the description attached to the ALIGN attribute within the **<DIV>** elements. *For example,*

```
<DIV ALIGN="left">This text will be displayed left aligned in the browser window.</DIV>
```

If you want to provide style information only for a few words, then you use <SPAN> tag. This tag is used to apply style locally to a particular paragraph. *For example*:

```
<P>Get <SPAN STYLE = "FONT-WEIGHT: BOLD; COLOR: BLACK; BACKGROUND: YELLOW"> whatever you love </SPAN> otherwise you will be forced to love whatever you get.</P>
```

Scanned with OKEN Scanner

CSS Class

CSS class is a selector to assign class name either one or group of element and apply specific styles. Using class selector you can easily modify element styles.

CSS class selector define using **class attribute** with **value** (user define class name).

CSS class selector identifies using (".") **dot sign** concat with user define class name.

CSS Class use for when you using JavaScript its identify of element using **getElementByIClass()** function.

Syntax

If you assign same class name multiple element syntax is:

```
.class_name {property: value; }
```

Example 4:

```
<html>
<head>
<style type="text/css">
.line {
 color:purple;
 font-size:25px;
 background-color:#CCFF99;
}
</style>
</head>
<body>
<p class="line">This is a First paragraph.</p>
<div class="line">This is a Second paragraph.</div>
</body>
</html>
```

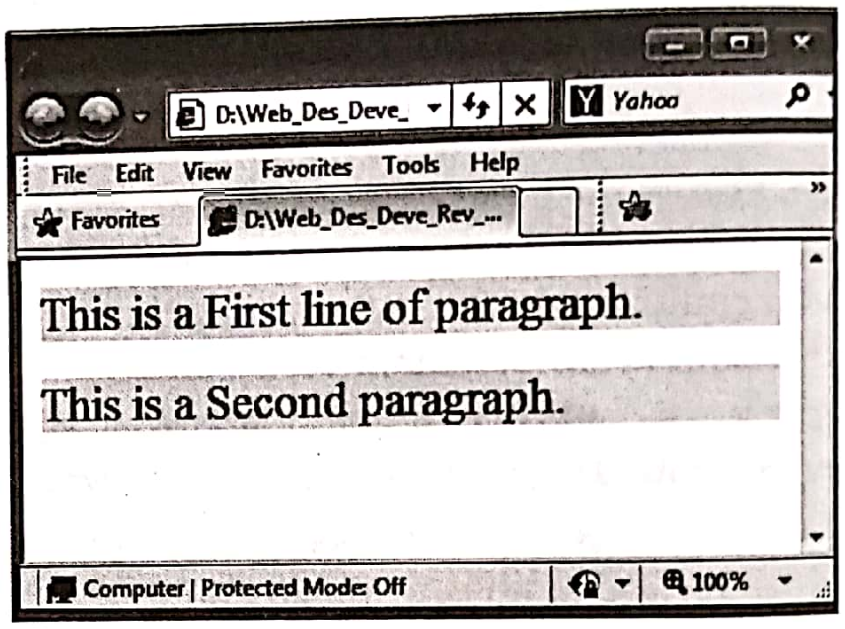


Figure 3.3: The output of the code Example 4

The output of the code given in Example 4 as shown in Figure 3.3.

3.3.1 CSS Selectors

A CSS selector is the most important part of CSS. They are used to select elements on a HTML page. The selected elements thus can be style according to the need and demand of the users. CSS selectors are of various types.

- CSS Element Selector
- CSS ID Selector
- CSS Class Selector
- CSS Universal Selector
- CSS Group Selector
- Tag Selector
- Attribute Selector
- Sub Selector

Scanned with OKEN Scanner

CSS Element Selector: The element selector selects HTML elements based on the element name.

Syntax:

```
Element_name {
      //CSS Property
}
```

Here, all <p> elements on the page will be center-aligned, with the green text color:

Example 5:

```
<!DOCTYPE html>
<html>
<head>
<style>
P {
   text-align: center;
   color: Green;
   }
</style>
<body>
<p>Every paragraph have the same alignment and same color</P>
<p id="para1">First paragraph</p>
<p>Second paragraph</p>
</body>
</html>
```

The output of the code given in Example 5 as shown in Figure 3.4

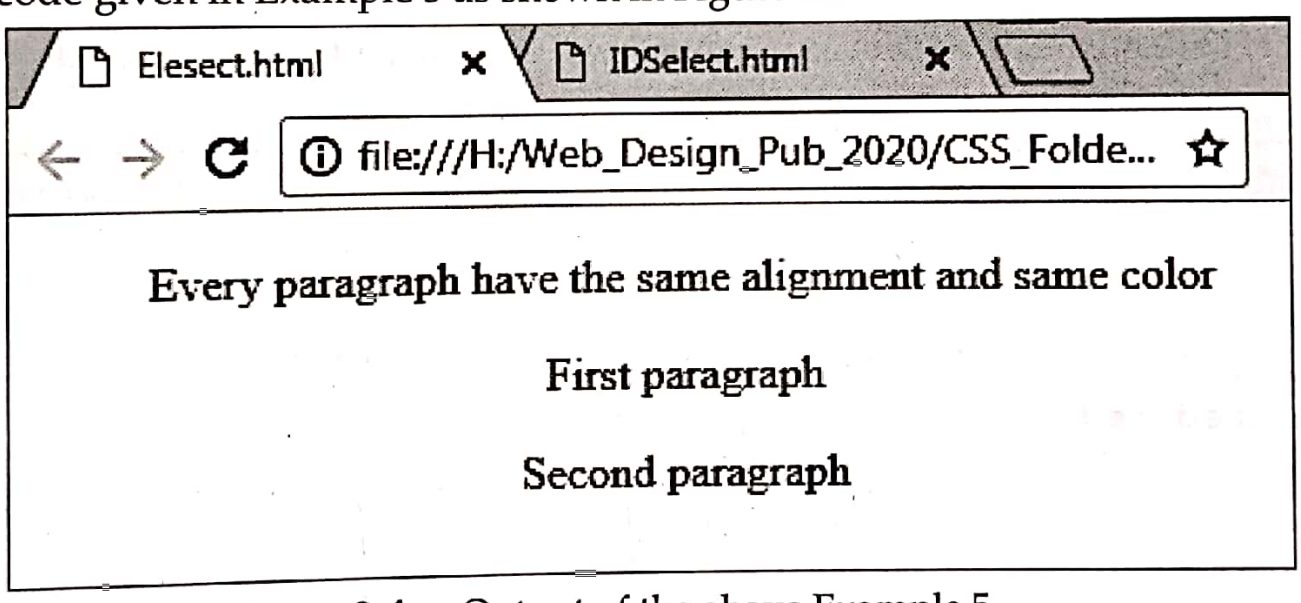


Figure 3.4: Output of the above Example 5

Class ID Selector: The id selector uses the id attribute of an HTML element to select a specific element. To select an element with a specific id, write a **hash (#)** character, followed by the id of the element. You should use IDs if only one element on the page should have the style applied, and/or you need a unique identifier for that element. *For example*, you might assign an ID to a **<nav>** tag which contains left menu. The style for this ID could contain the position, background-color, size and so on. You need not use any other element on the page to use the particular style.

CSS Class Selector: The class selector selects html elements with a specific class attribute. You declare a CSS class by using a dot **(.)** followed by the class name. After class name you simply enter the properties/value that you want to assign to your class. You should use classes when your style needs to be applied multiple times on the same page.

Scanned with OKEN Scanner

Example 6:

```
<!DOCTYPE>
<html>
<head>
<style>
#para {
  text-align: center;
  color: blue;
  text-size:30px;
  font-weight:bold;
}
</style>
<body>
<p id="para">CSS ID Selector</p>
<p>This sentence is not affected
by the style</p>
</body>
</html>
```

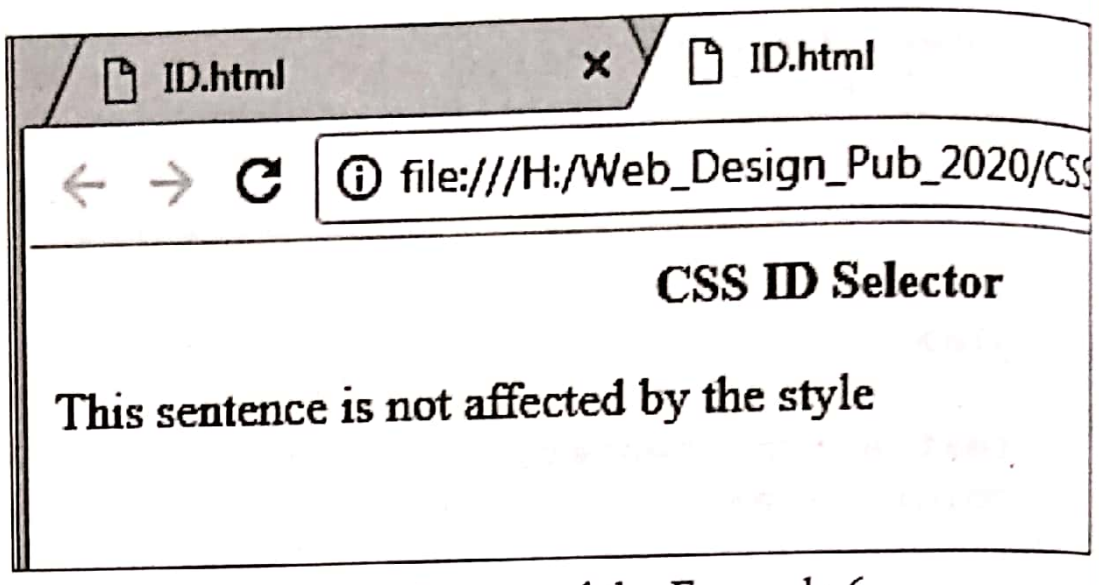


Figure 3.5: Output of the Example 6

The output of the code given in Example 6 as shown in Figure 3.5

For example, you might have many <h1> elements that need the same style applied.

In this example, all HTML elements with class="center" will be blue and center-aligned.

Example 7:

```
<!DOCTYPE>
<html>
<head>
<style>
div.css-class {
border:2px dotted red;
padding:20px;
}
p.css-class {
text-align:center;
color: blue;
}
</style>
<div class="css-class">CSS Class</div>
<p class="css-class">CSS class to apply a style to a given class of an
element.</p>
</body
</head>
</html>
```

The output of the code given in Example 7 as shown in Figure 3.6

Scanned with OKEN Scanner

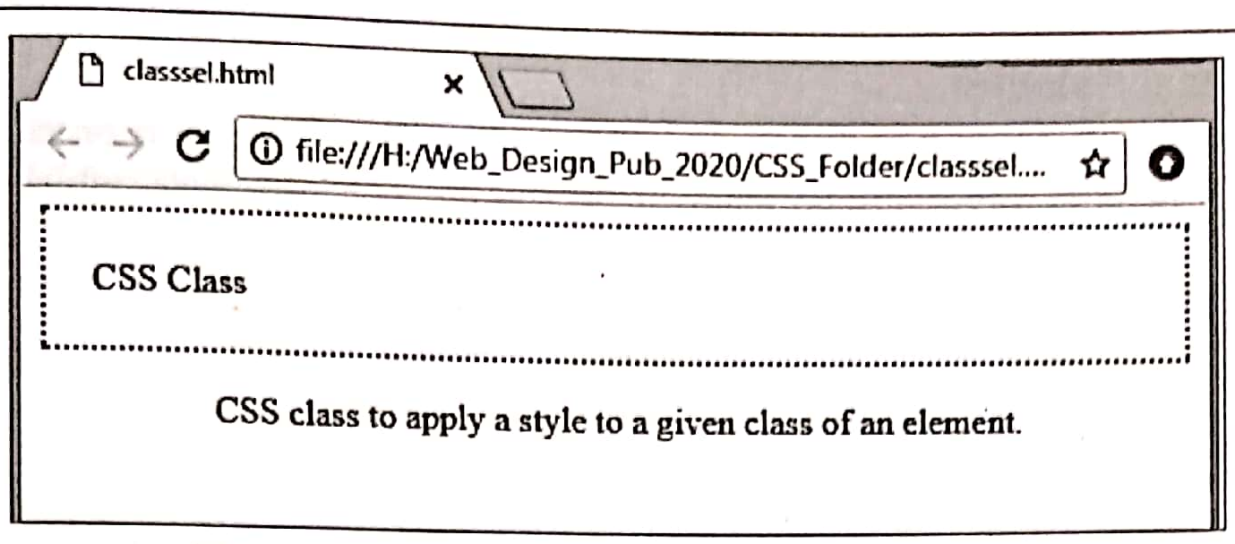


Figure 3.6: Output of the above Example 7

HTML elements can also refer to more than one class.

Example 8:

```
<!DOCTYPE>
<html>
<head>
<style>
div.css-class {
border:1px dotted red;
padding:10px;
}
div.css-class {
text-align:center;
color: blue;
}
</style>
<div class="css-class">CSS Class
<p>CSS class to apply a style to a given class of an element.</p>
</div>
</body>
</head>
</html>
```

The output of the code given in Example 8 as shown in Figure 3.7

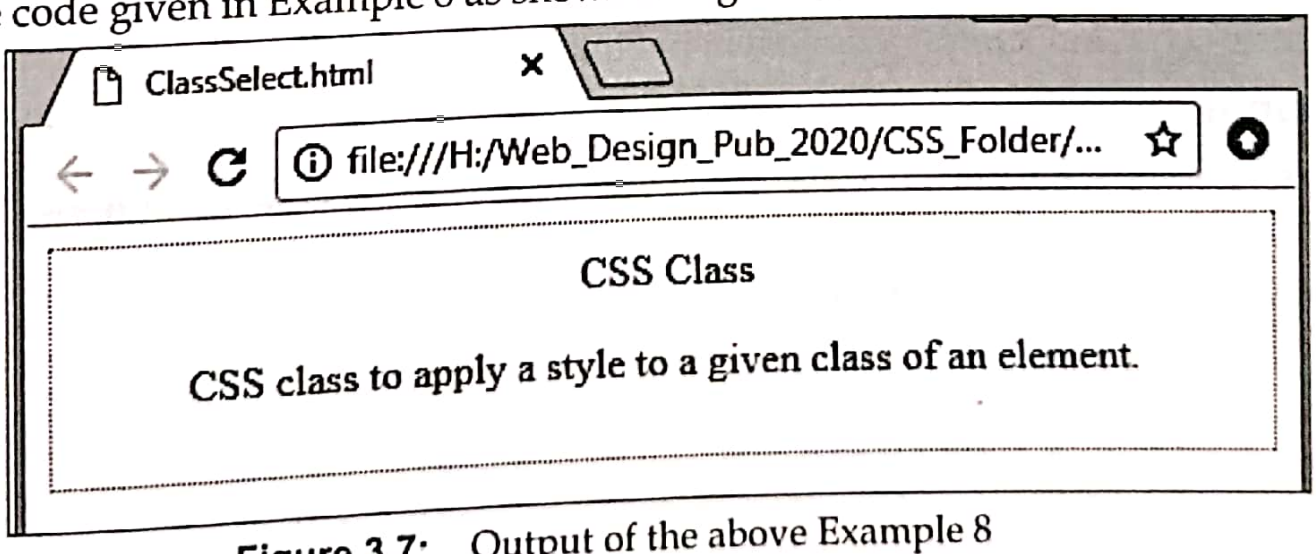


Figure 3.7: Output of the above Example 8

Scanned with OKEN Scanner

3.3.2 CSS Universal Selector

The universal selector selecting all elements on a page. Every HTML page built on content placed within HTML tags. Each set of tags represents an element on the page. *For example*, which uses the universal selector.

```
* {
 Color: green;
Font-size: 20px;
Line-height: 25px;
}
```

The code inside the curly braced (color, font-size, and line-height) will apply to all elements on the HTML page. The universal selector as declared using an **asterisk.**

Example 9:

```
<!DOCTYPE>
<html>
<head>
<style>
div.css-class {
border:1px dotted red;
padding:10px;
}
div.css-class {
text-align:center;
color: blue;
}
</style>
<div class="css-class">CSS Class
<p>CSS class to apply a style to a
given class of an element.</p>
</div>
</body>
</head>
</html>
```

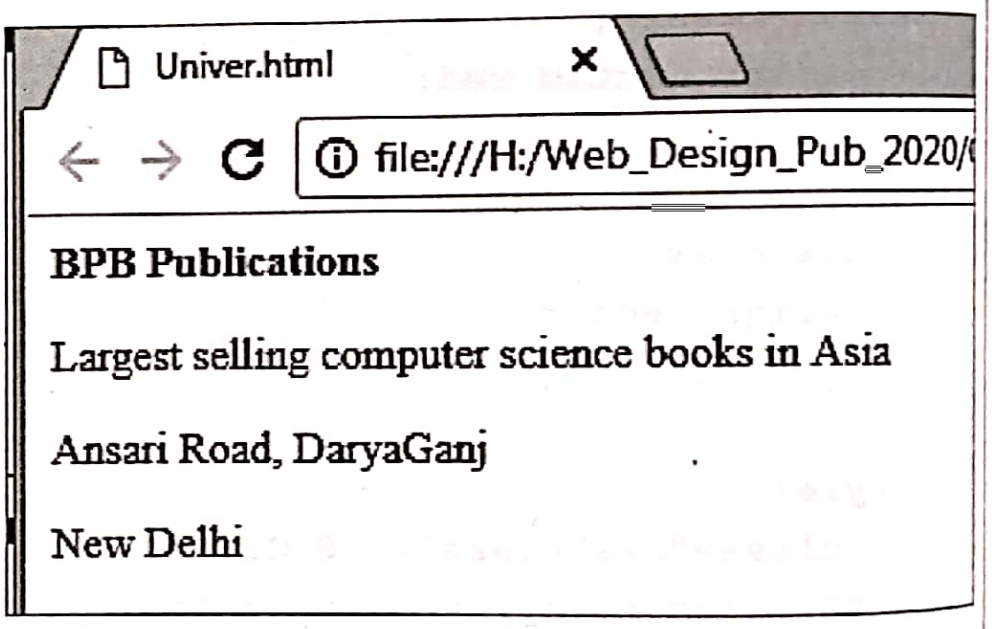


Figure 3.8: Output of the Example 9

The output of the code given in Example 9 as shown in Figure 3.5

3.3.3 CSS Grouping Selector

The grouping selector selects all the HTML elements with the same style definitions. It means that all elements targeted by any of the selectors will be affect by the CSS rule.

Let us see the CSS code without group selector.

```
h1 {
    text-align: center;
    color: blue;
}
```

Scanned with OKEN Scanner

```
h2 {
  text-align: center;
  color: blue;
}
p {
  text-align: center;
  color: blue;
}
```

Now, you can written better to group the selectors, to minimize the code. To group selectors, separate each selector with a comma. The following example, you have grouped the selectors from the code above.

Example 10:

```
<!DOCTYPE>
<html>
<head>
<style>
h1,h2,p {
 text-align: left;
 color: blue;
}
</style>
</head>
<body>
<h1>Introduction to CSS</h1>
<h2>Types of CSS</h2>
<p>CSS Selectors</p>
</body>
</html>
```

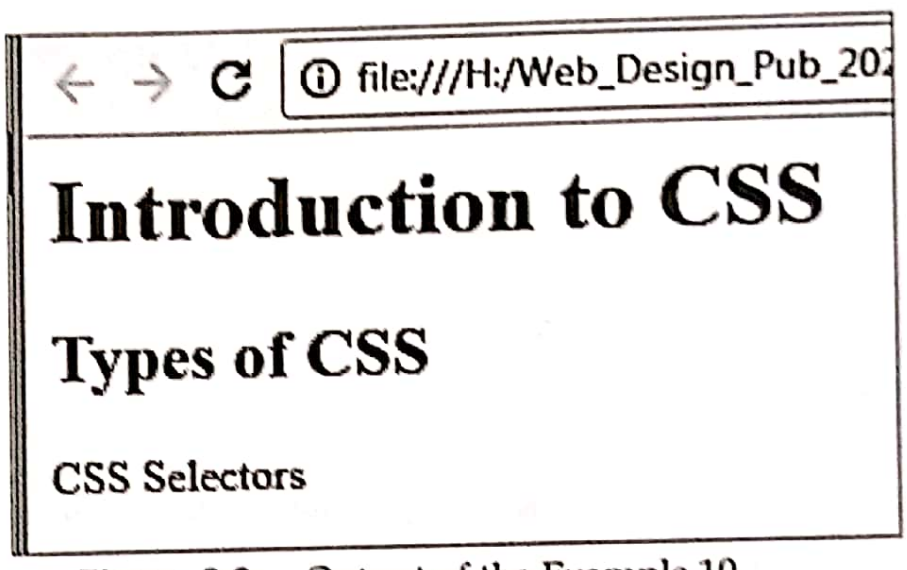


Figure 3.9: Output of the Example 10

The output of the code given in Example 9 as shown in Figure 3.8

3.3.4 Attribute Selector

Attribute selector means the property, character or behavior of the selector. If you want to select an element with a particular attribute, then you can use Attribute Selector. It is one of the ways to select a particular element with a particular type of attribute.

There is multiple way to use CSS Attribute Selector. Let us discuss all one by one:

Attribute Selector	Selectors Example
[attribute=name]	a[target] [color: blue, background-color: gray;}
[attribute-name="value"]	div[lang="en-US"] {background-color: blue;}
[attribute-name|="value"]	p[lang|en] {background: green;}
[attribute-name~="value"]	image[title~="border"] {broder: 1px solid green;}
[attribute-name$="value"]	img[src$=".jpg"] {border: 3px blue;}
[attribute*="value"]	a[href*="tutorial"] {color: red}
[attribute-name^="value"] underline;}	a[href^=https://] {color: pint; text-decoration:

Scanned with OKEN Scanner

Let us select a particular element with a particular attribute (without any value).

Syntax:

```
[attribute-name] {
   Properties:values;
};
```

The following example selects all <a> elements with a target attribute:

Example 11:

```
<!DOCTYPE>
<html>
<head>
<style>
a[title] {
color:red;
}
a[target] {
 text-decoration:underline;
 background-color: yellow;
}
</style>
</head>
<body>
<p>The links with a target attribute gets a yellow background</p>
<a href="https://www.wikipedia.org" target="_top">wikipeida.org</a>
<a href="https://www.bseindia.com" target="_blank">BSEINDIA</a>
</body>
</html>
```

The output of the code given in Example 11 as shown in Figure 3.10

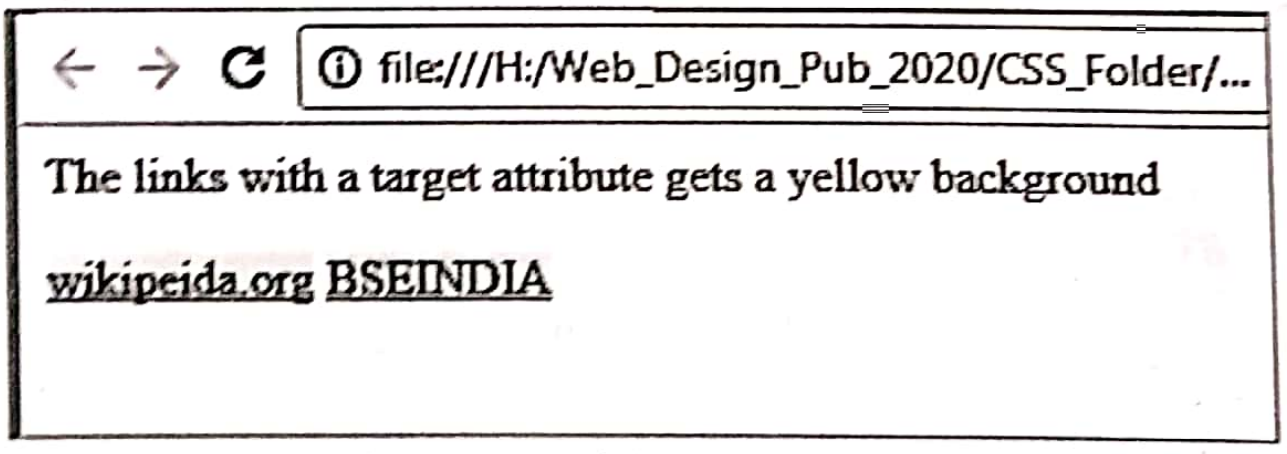


Figure 3.10: Output of the above Example 11

CSS [attribute="value"] Selector

Let us select a particular element with a particular attribute with a particular fixed value.

```
[attribute-name="value"] {
 Properties:values;
};
```

Scanned with OKEN Scanner

The following example, select a tag with the href attribute having the value of hyperlink as **http://bpbonline.com** & will set the text color to green.

Example 12:

```
<!DOCTYPE>
<html>
<head>
<style>
a[href="http://www.bpbonline.com"] {
color: green;
}
div[lang="en"] {
background-color: pink;
padding:4px;
}
</style>
</head>
<body>
<div lang="en">
  <a  href="http://www.bpbonline.com"  target="_blank">Example  of  CSS  div
attribute
        Selector using [attribute="value"]</a><br>
<a href="https://www.wikipedia.org" target="_blank">This is not using the CSS
      Selector which is defined in the head section here</a>
</div>
</body>
</html>
```

The output of the code given in Example 12 as shown in Figure 3.11

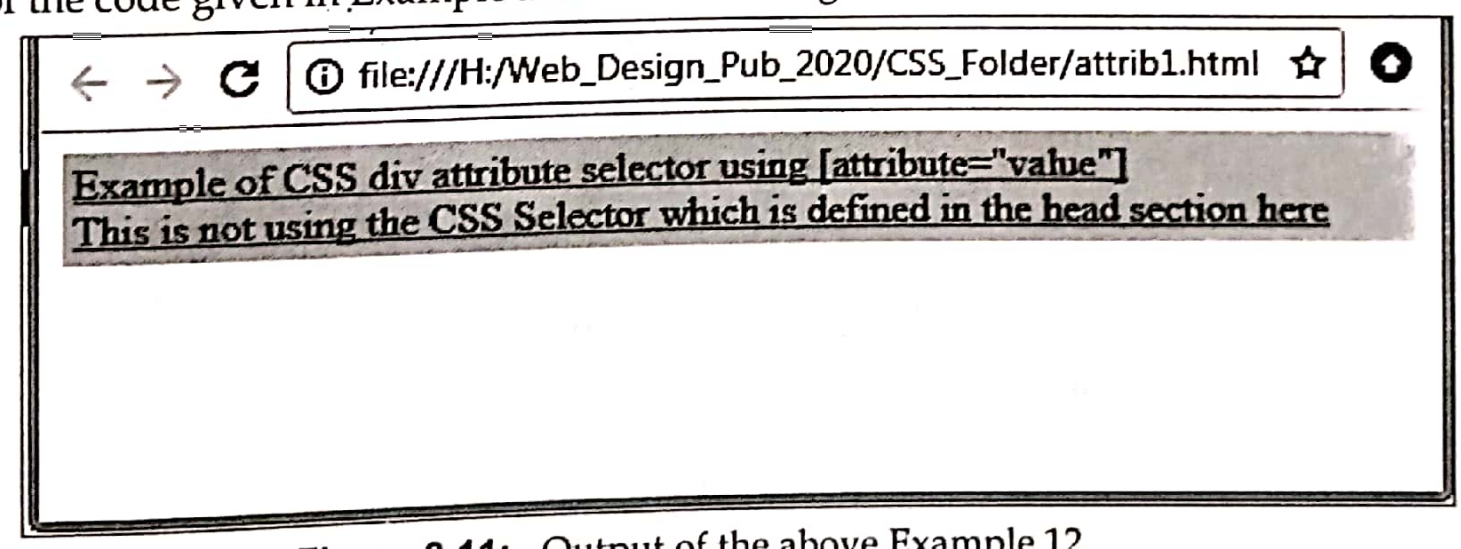


Figure 3.11: Output of the above Example 12

CSS [attribute~="value"] Selector

The [attribute~="value"] selector is used to select elements with an attribute value containing a specified word. The following example selects all elements with title attribute that contains a space-separated list of words, one of which is Mount.

Scanned with OKEN Scanner

Example 13:

```
<!DOCTYPE>
<html>
<head>
<style>
[title~=Mount] {
border: 5px solid orange;
}
</style>
</head>
<body>
<p>All images with the title attribute containing the word "Mount" get orange order.</p>
<img src="H:/Mount.jpg" title="Mount Water" width="150" height="125">
<img src="H:/Mount1.jpg" title="Mount" width="225" height="155">
<img src="H:/island.jpg" title="sea_iland" width="200" height="275">
</body>
</html>
```

The output of the code given in Example 13 as shown in Figure 3.12

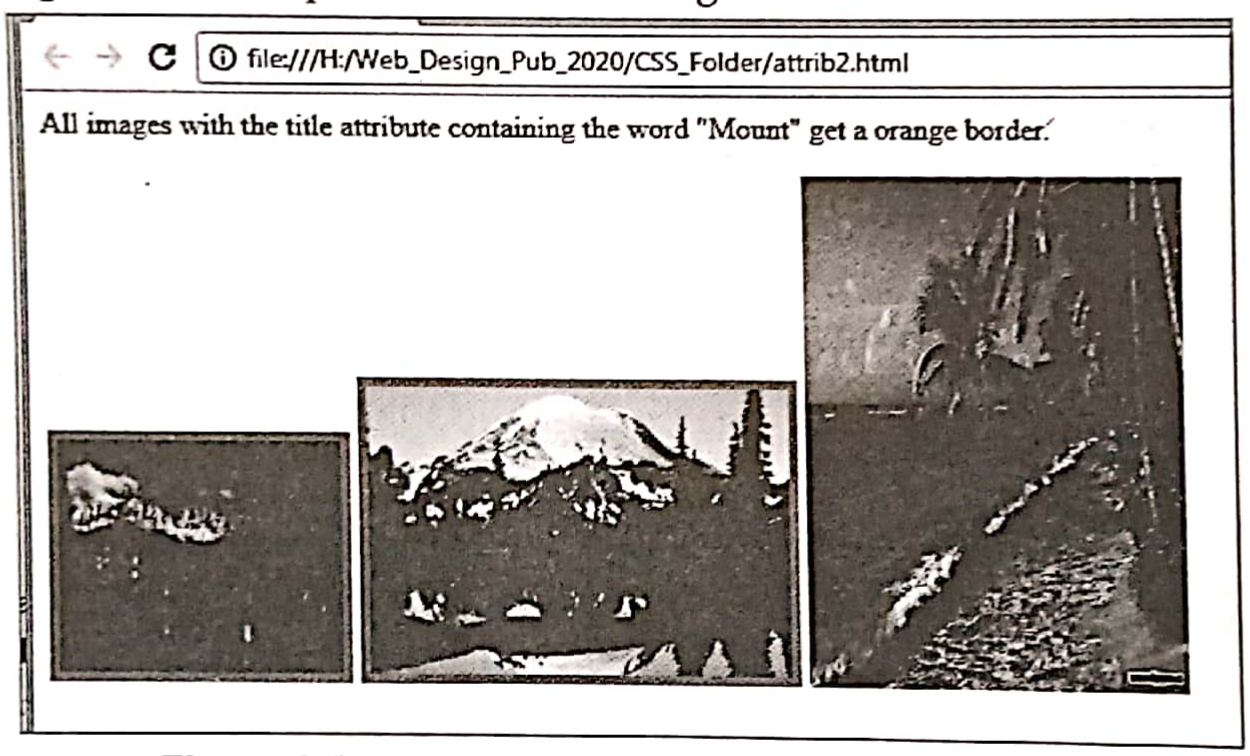


Figure 3.12: Output of the above Example 13

CSS [attribute|="value"} Class Selector

The [attribute|="value"] Selector is used to select elements with the specified attribute starting with the specified value. The following example selects all elements with a class attribute value that being with top.

Example 14:

```
<!DOCTYPE>
<html>
<head>
```

(Contd...)

Scanned with OKEN Scanner

```
<style>
[class|="top"] {
  background-color: yellow;
  border:4px groove tomato;
}
</style>
<body>
<h2>Example of CSS attribute Class Selector</h2>
<p class="top-header">CSS Attribute Selector with the class as it is a fixed a match. [attribute|="value"] is applied </p>
<p class="top-text">CSS Attribute Selector is applied as it "top" matches one of the value [attribute|="value"] is applied </p>
<p class="topcontent">Topcontent is not same as top so [attribute |=top="value"] is not applied.</p>
</body>
</html>
```

The output of the code given in Example 14 as shown in Figure 3.12

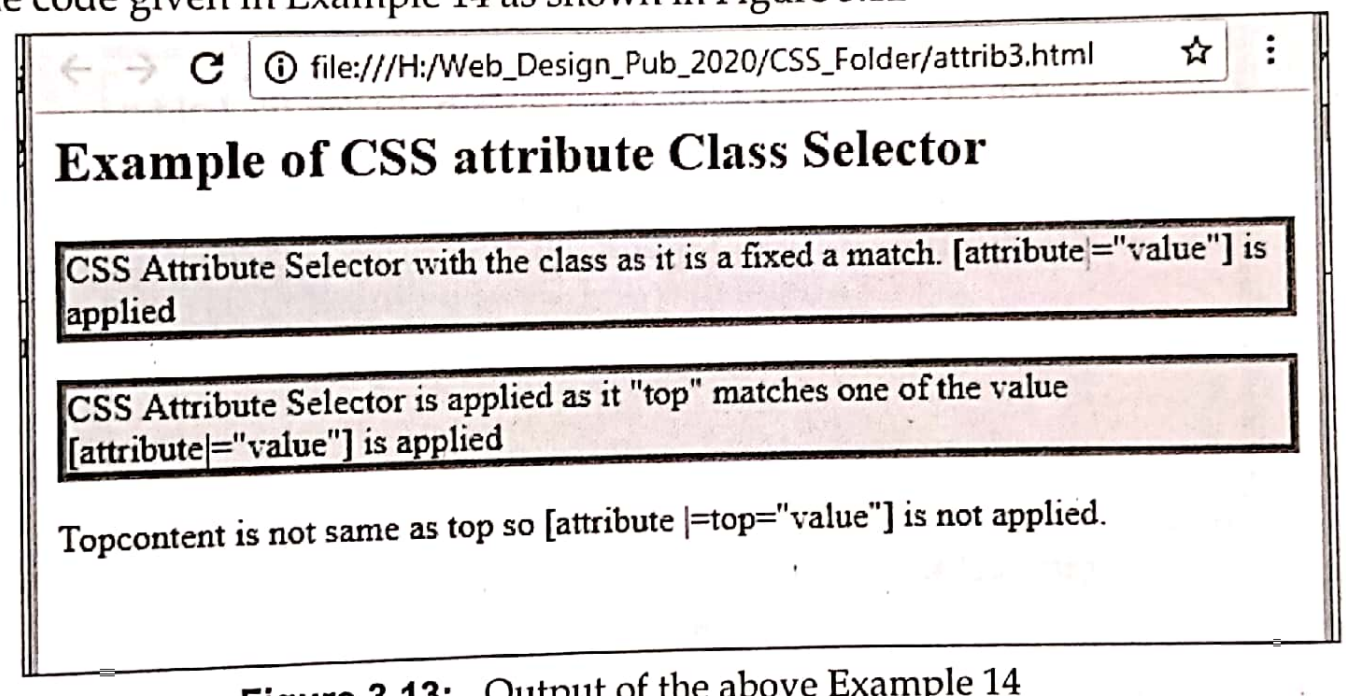


Figure 3.13: Output of the above Example 14

CSS [attribute^="value"] Selector

The [attribute^="value"] selector is used to select elements whose attribute value begins with a specified value. To select all the href which are secured i.e. which starts with "https://" , then you can use this example. Here, you are not validating the selected links, rather just styling these with color as blue and underlining them.

Example 15:

```
<!DOCTYPE>
<html>
<head>
<style>
a[href^="https://"] {
color: orange;
```

(Contd...)

Scanned with OKEN Scanner

```
text-decoration:none;
}
a[href^="http://"] {
background-color: yellow;
text-decoraton:underline;
}
</style>
<body>
<ul>
<li><a  href="http://Google.com"  target="_blank">The  Link  starts  with
http://</a></li>
<li><a  href="https://Yahoo.com"  target=_"blank">The  Link  starts  with
https://</a></li>
</ul>
</body>
</html>
```

The output of the code given in Example 15 as shown in Figure 3.14

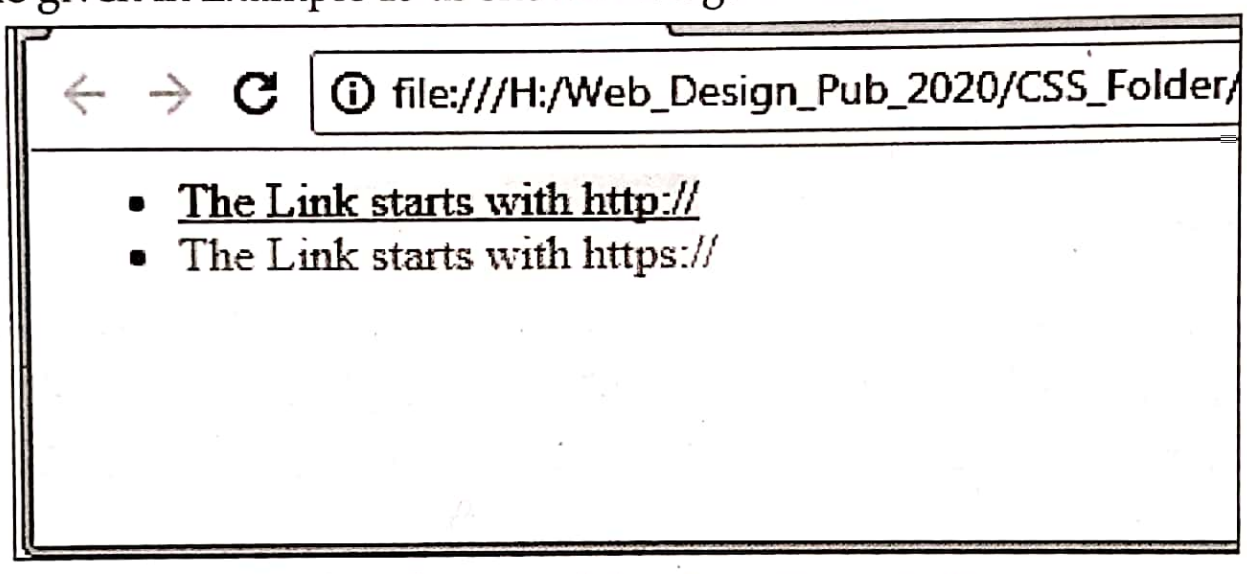


Figure 3.14: Output of the above Example 15

CSS [attribute $="value"] Selector

The [attribute$="value"] Selector is used to select elements whose attribute value ends with a specified value.

Example 16:

```
<!DOCTYPE>
<html>
<head>
<style>
a[href$="bpbonline.com"] {
background-color: cyan;
color: red;
}
div[id$="auto"] {
```

(Contd...)

Scanned with OKEN Scanner

```
max-width:500px;
height:auto;
background:yellow;
}
img[src$=".jpg"] {
border:5px ridge blue;
width:50%;
height:30%;
}
</style>
</head>
<body>
<h5>The below div ID ends with "auto"</h5>
<div id="text-auto">
 <a href="https://www.bpbonline.com" target"_blank">
 <img src="H:/Mount.jpg" alt="Books store">
<br>Image ends with ".jpg" and href ends with "bpbonline.com"
</a>
</div>
</body>
</html>
```

The output of the code given in Example 16 as shown in Figure 3.15

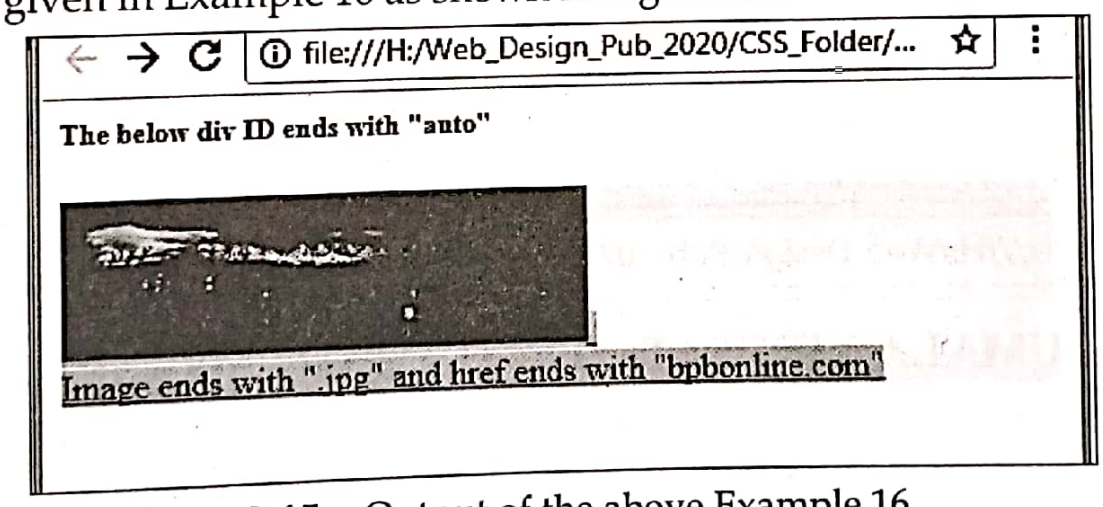


Figure 3.15: Output of the above Example 16

```
div[id$="auto"] {
width: 400px
height:auto;
background:yellow;
}
```

This will select all the div with ids which end with auto & will set the width, height and background color.

[attribute*="value"] Selector

This selector selects all the elements whose attribute value contain the specified value present anywhere. The value does not match the whole word. The following example selects all elements with a class attribute value that contains th.

Scanned with OKEN Scanner

Example 17:

```
<!DOCTYPE>
<html>
<head>
<style>
[class*="ti"] {
  color:green;
  font-size:20px;
  font-weight:bold;
  text-align:center;
}
[class*="tr"] {
  font-size:18px;
  text-align:center;
  margin-top:0px;
}
</style>
</head>
<body>
<div class="tiru">TIRUMALA TIRUPATI DEVASTHANAMS TIRUPATI</div>
<div class="trth">Venkatadri samam sthanam brahmande nasti kinchana</div>
<div class="the tr">Venkatesa samo devo na bhuto na bhavishyati.</div>
</body>
</html>
```

The output of the code given in Example 17 as shown in Figure 3.16

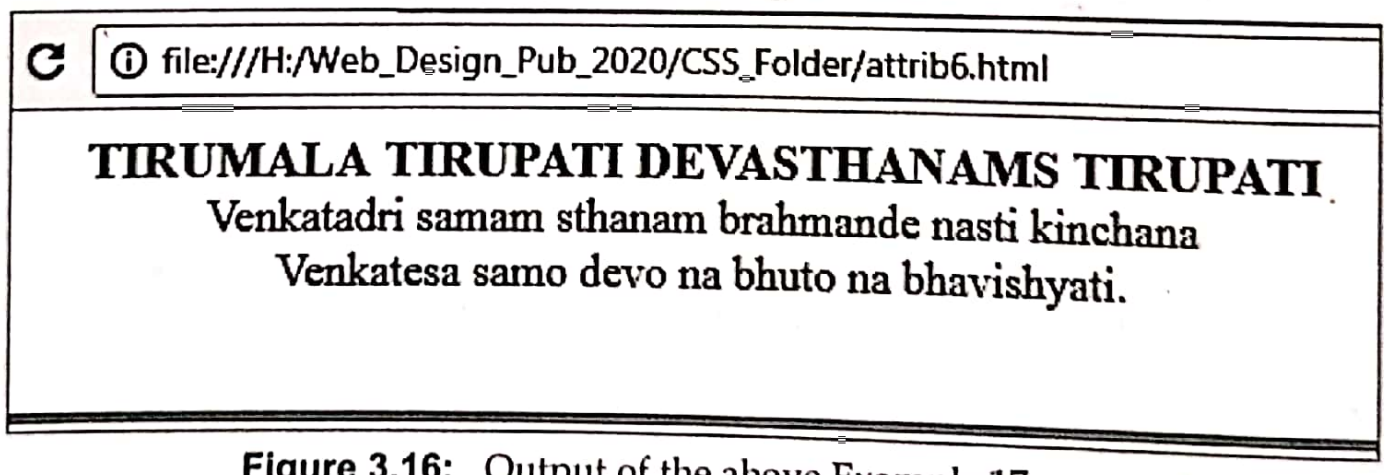


Figure 3.16: Output of the above Example 17

The attribute selectors can be useful for styling forms without **Class** or **ID**.

Example 18:

```
<!DOCTYPE>
<html>
<head>
<style>
input[type="text"] {
```

(Contd...)

Scanned with OKEN Scanner

```
width: 150px;
 display: block;
 margin-bottom: 10px;
 background-color: yellow;
}
input[type="button"] {
 width: 120px;
 margin-left:25px;
 display: block;
}
</style>
</head>
<body>
<form name="input" action="" method="get">
 FirstName:<input type="text" name="Name" value="Satish" size="15">
 LatName:<input type="text" name="Name" value="Jain" size="10">
<input type="button" value="Submit Button">
</form>
</body>
</html>
```

The output of the code given in Example 18 as shown in Figure 3.17

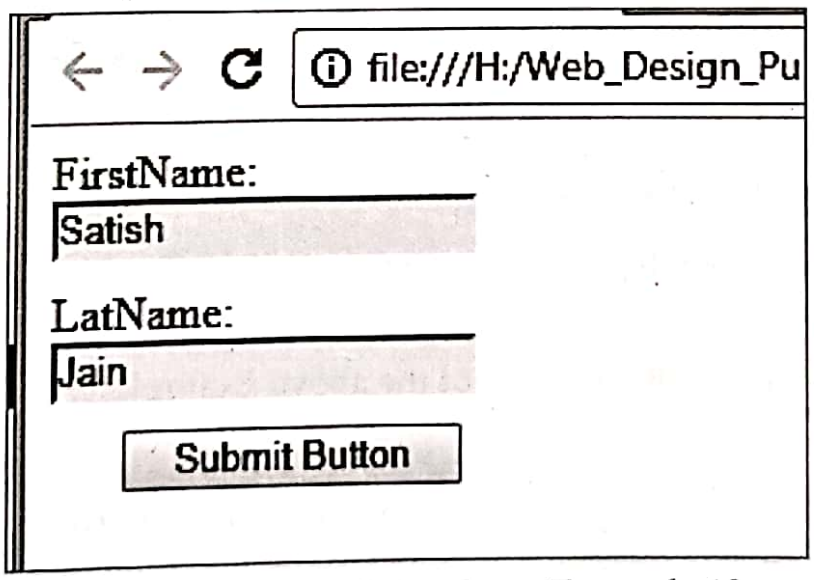


Figure 3.17: Output of the above Example 18

3.3.5 Sub Selector

A CSS selector can contain more than one simple selector. There are four different combinators in CSS:

Descendant Selector

The descendant CSS selector is used to select elements that are descendants of other elements. The selected elements do not have to be immediate children of the specified ancestor, they just have to be nested inside the ancestor somewhere.

Syntax

This syntax apply to match all **<p>** element that are descendants of specified element. All **<p>** element that are descendants of ** ** to apply CSS.

Scanned with OKEN Scanner

Example 19:

```
<!DOCTYPE>
<html>
<head>
<title>CSS Descendant Selector </title>
<style>
div p {
  font-size;24px;
  background-color: yellow;
 }
</style>
</head>
<body>
<div>
<p> This is first paragraph text</p>
<h4>Fourth heading text</h4>
<p> This is another paragraph text</p>
</div>
<p>This is third paragraph text and not in div</p>
</body>
</html>
```

The output of the code given in Example 19 as shown in Figure 3.18

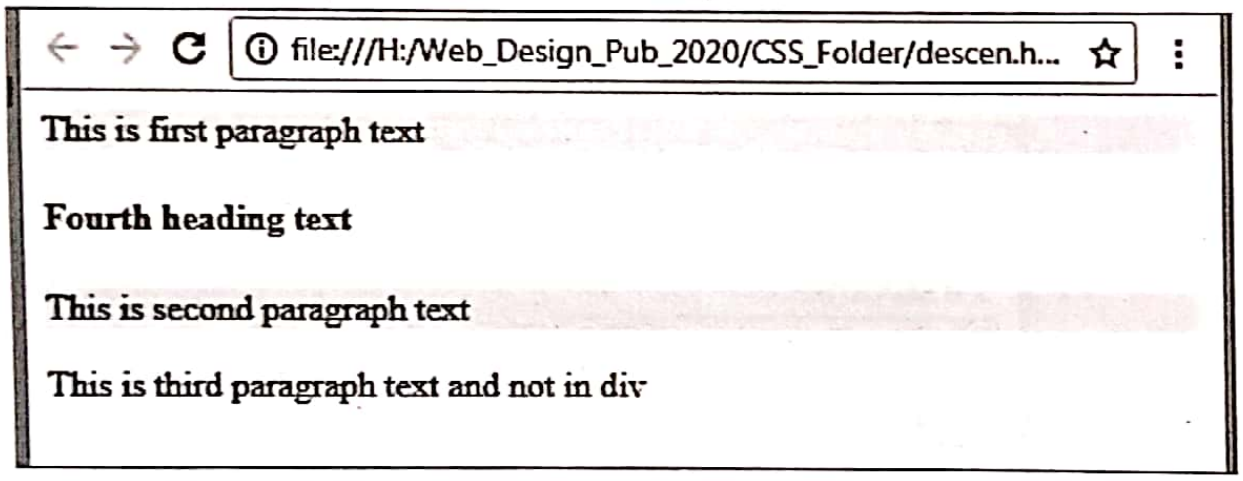


Figure 3.18: Output of the above Example 19

Child Selector

It is use to match all the elements, which are child of a specified element. It gives the relation between two elements. The element selector selects elements which are the children of specific parent. The child selector selects all elements that are the children of a specified element. *For example*, selects all **<p>** elements that are children of a ** ** element.

Example 20:

```
<!DOCTYPE>
<html>
<head>
<title>CSS Descendant Selector </title>
<style>
```

(Contd...)

Scanned with OKEN Scanner

```
<div> p {
  font-size;24px;
  background-color: yellow;
  padding:2px;
 }
</style>
</head>
<body>
<div>
<p> This is first paragraph text</p>
<section>This is second paragraph text</p></section>
<h4>Fourth heading text</h2>
<p> This is second paragraph text</p>
</div>
<p>This is third paragraph text</p>
</body>
</html>
```

The output of the code given in Example 20 as shown in Figure 3.20

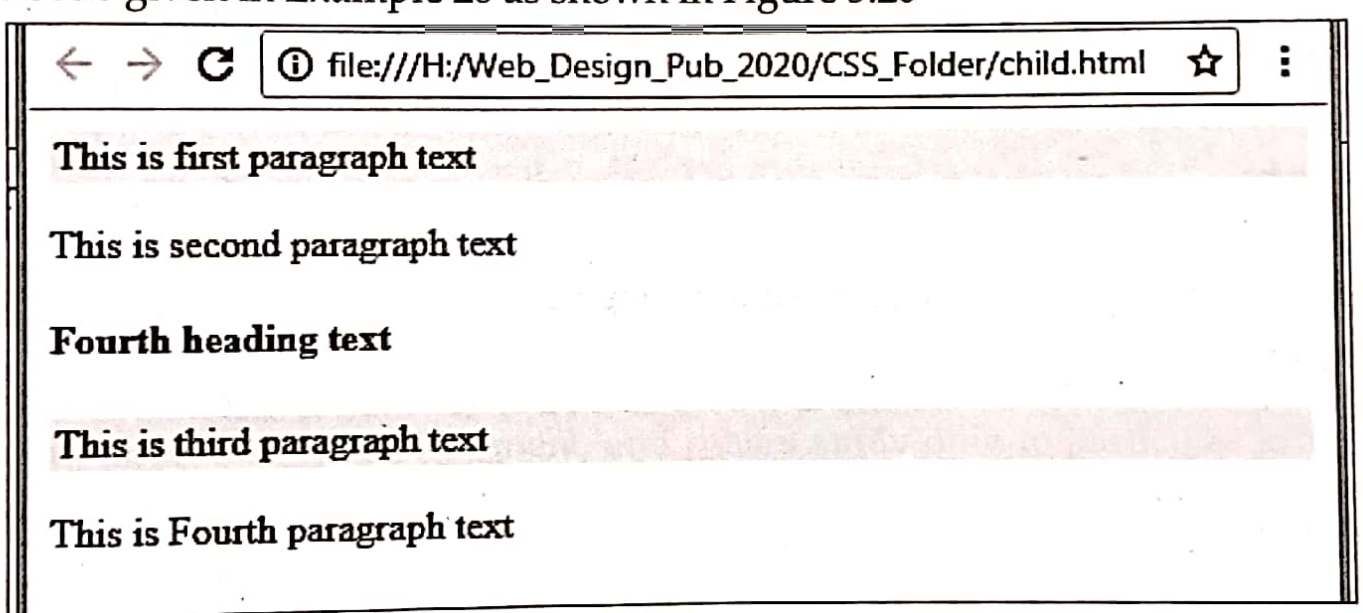


Figure 3.19: Output of the above Example 20

Adjacent Sibling Selector

The adjacent sibling CSS selector is use to select HTML elements that are adjacent siblings of some other HTML element. *For example,* all <h> elements that are placed immediately after elements.

Example 21:

```
<!DOCTYPE>
<html>
<head>
<title>CSS Descendant Selector </title>
<style>
div + h {
```

(Contd...)

Scanned with OKEN Scanner

```
  font-size;24px;
  background-color: yellow;
  padding:2px;
 }
</style>
</head>
<body>
<div>
<p> This is first paragraph text</p>
<p> This is second paragraph text</p>
</div>
<h3>This is third paragraph text</h3>
</body>
</html>
```

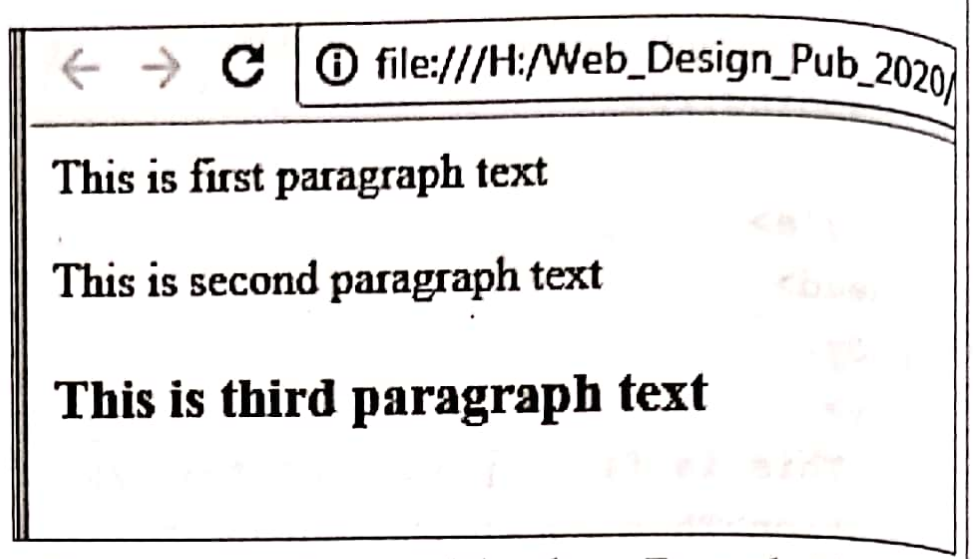


Figure 3.20: Output of the above Example 21

The output of the code given in Example 21 as shown in Figure 3.19

3.3.6 CSS Tag Selector

To select tags by their name and apply styles on them you use tag selector.

Syntax of CSS rule-set

```
Selector
{
Declaration list;
}
```

To implement tag selector, in place of selector, we write tag name.

Syntax of tag selector:

```
Tagname
{
Declaration list;
}
```

It selects every html tag with the specified tag name and applies styles on them.

Example 22:

```
<!DOCTYPE html>
<html>
<head>
<title>Tag Selector</title>
<style>
 P {
  color: green;
  font-size:20px;
  border: 2px solid blue;
}
```

Scanned with OKEN Scanner

```
</style>
</head>
<body>
<p>This the first paragraph and the style mentioned in the above code, the text color in green and font size is 12 pt and the border of the paragraph around is color solid blue.</p>
<h3>This is h3 heading</h3>
<p>This the another paragraph group with the same style applied.</p>
</body>
</html>
```

The output of the code given in Example 22 as shown in Figure 3.21.

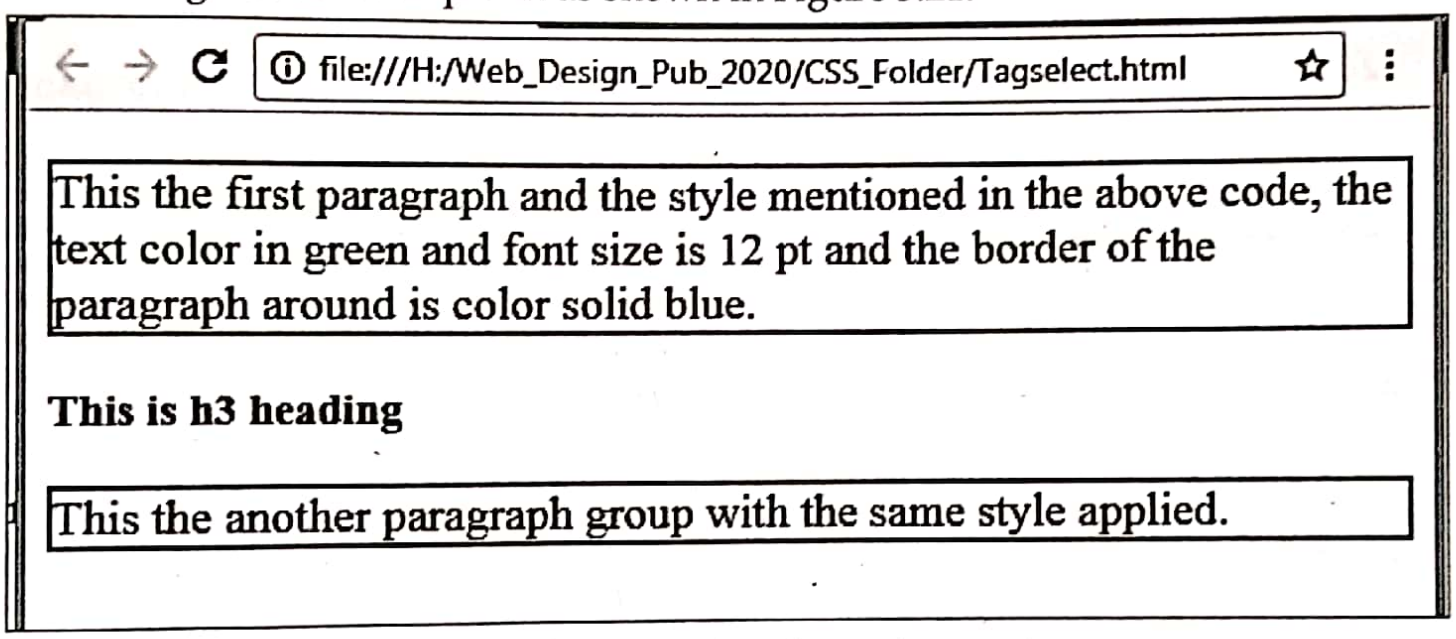


Figure 3.21: Output of the above Example 22

3.4 CSS Properties

CSS properties are used to apply styles of structured document. Each property has a set of possible values. Some properties can affect any type of element, and others apply only to particular groups of elements. A style sheet consists of a list of rules. Each rule set consists of one or more selectors, and a declaration block.

In the following example, you set the background color of the <body> element to yellow.

```
Body {
Background-color: yellow;
}
```

In this case, background-color is the property, and yellow is its value.

And you can apply multiple properties against any selector.

```
Body {
 Background-color: yellow;
Color : green;
Font-size: 16px;
}
```

For example, where styles can fit into an HTML document:

Scanned with OKEN Scanner

Example 23:

```
<!DOCTYPE>
<html>
<head>
<title>CSS Properties Example</title>
<style>
body {
  background: lightsalmon;
  font-family: Book Antiqua;
  font-size: 12px;
  }
section {
  padding: 20px;
  margin: 20px;
  background color: gray;
  border: 5px solid gold;
}
h1 {
 color: red;
}
p {
 color: organge;
}
a:link,
a:visited {
 color: deeppink;
}
a:hover {
 color: yellow;
 }
a:active {
 background: blue;
 color: white;
}
</style>
<section>
<h1>Basic CSS Properties Example</h1>
<p>This is a basic example to use CSS properties can be applied to a web
page.</p>
<p><a href="H:/html5_tutorial.pdf/" target="_blank">HTML with CSS
tutorial</a></p>
</section>
</html>
```

The output of the code given in Example 23 as shown in Figure 3.22.

Scanned with OKEN Scanner

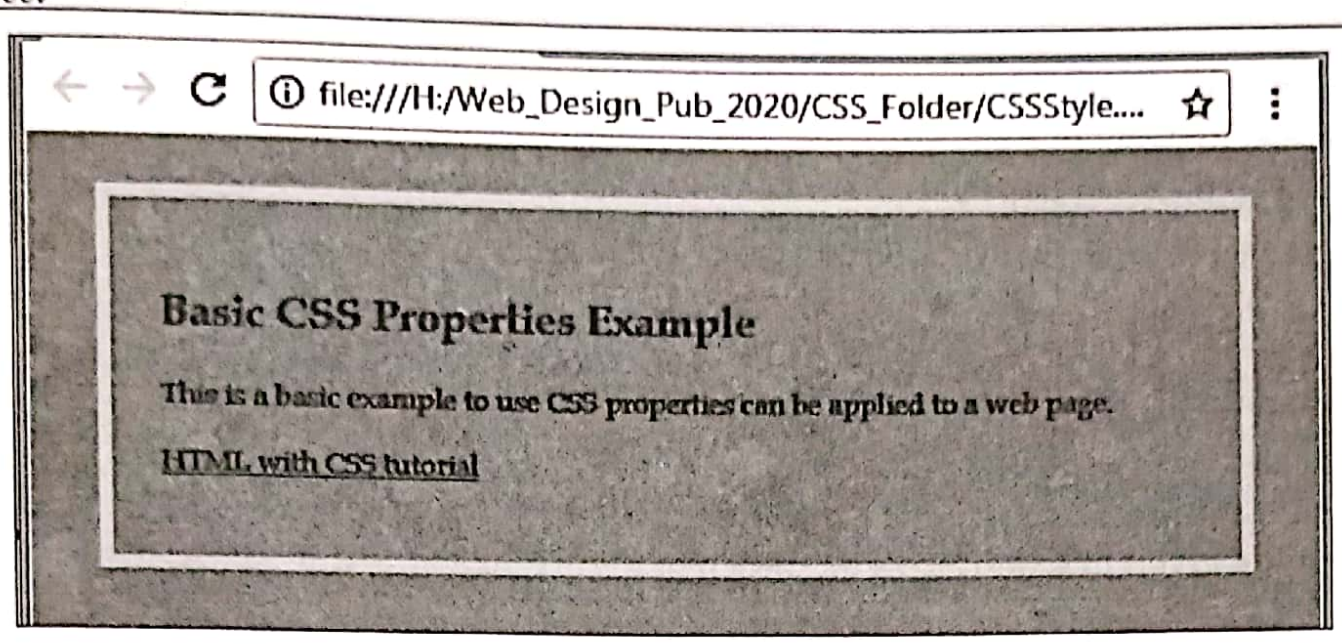


Figure 3.22: Output of the above Example 23

3.4.1 CSS Background Properties

The CSS background properties are used to define the background effects for elements. There are five types of CSS background properties that affects the HTML elements:

1. Background-color
2. Background-image
3. Background-repeat
4. Background-attachment
5. Background-position

CSS Background-color: The background-color property specifies the background color of an element.

Syntax

```
Background-color:<color>
```

Example 24:

```
<!DOCTYPE>
<html>
<head>
<title>CSS Properties Example</title>
<style>
h1 {
 background-color: red;
}
div {
background-color: gold;
}
p {
 background-color:orange;
}
</style>
</head>
```

(Contd...)

Scanned with OKEN Scanner

```
<body>
<h1>Background Properties</h1>
<div>
This has a background color of gold inside the div element
<p>This sentence has its own background and still in the div element.</p>
</div>
</body>
</html>
```

The output of the code given in Example 24 as shown in Figure 3.22.

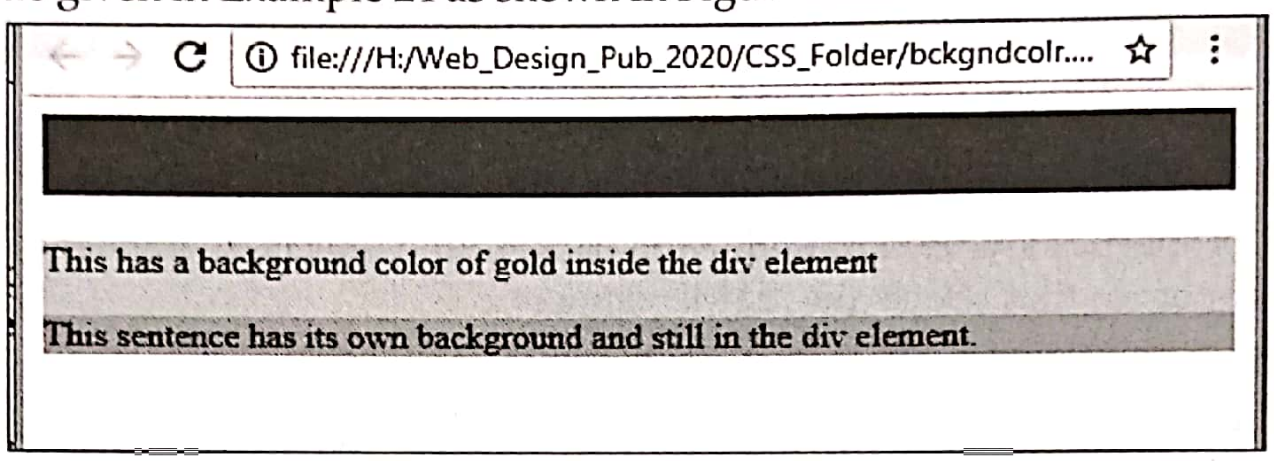


Figure 3.23: Output of the above Example 24

3.4.2 CSS Background-image

The background-image property specifies an image to use a background of an element. You can also set height and width of the background image.

Example 25:

```
<!DOCTYPE>
<html>
<head>
<title>CSS Properties Example</title>
<style>
body {
 background-image: url("H:/Mount.jpg")
 }
</style>
</head>
<body>
<h1>Image background</h1>
<p>This has a background color of gold inside the div element
<p>This sentence has its own background and still in the div element.</p>
</body>
</html>
```

The output of the code given in Example 25 as shown in Figure 3.24.

Scanned with OKEN Scanner

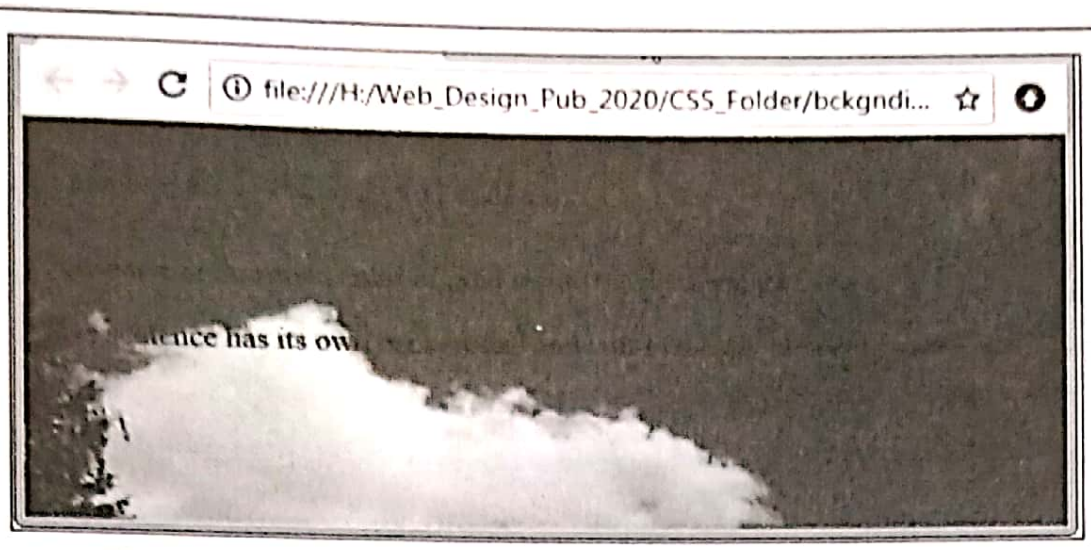


Figure 3.24: Output of the above Example 25

3.4.3 CSS background-repeat

By default, the background-image property repeats an image both horizontally (x-axis) and vertically (y-axis). To control whether the background image should repeat or not, you can use the background-repeat property.

a. Background-repeat: Repeat means the background image should repeat in the horizontal and vertical direction. This is the default value.
b. Background-repeat: No-repeat means the background image should not repeat in any direction.
c. Background-repeat: Repeat-x means the background image should repeats in horizontal direction (x-axis).
d. Background-repeat: Repeat-y means the background image should repeats in vertical direction (y-axis).
e. Background-repeat: Space means the background image should repeat with even whitespaces around between the repeated images.
f. Background-repeat: Round means the background image should repeat and the background image will either squeeze or stretch around each other.

Example 26:

```
<!DOCTYPE>
<html>
<head>
<title>CSS Properties Example</title>
<style>
body {
 background-image:url(H:/Blur.jpg);
 background-repeat: repeat-x;
 }
</style>
</head>
<body>
<h1>Image background</h1>
<p>This has a background image is repeat in horizontal direction.</p>
</body>
</html>
```

Scanned with OKEN Scanner

The output of the code given in Example 26 as shown in Figure 3.25.

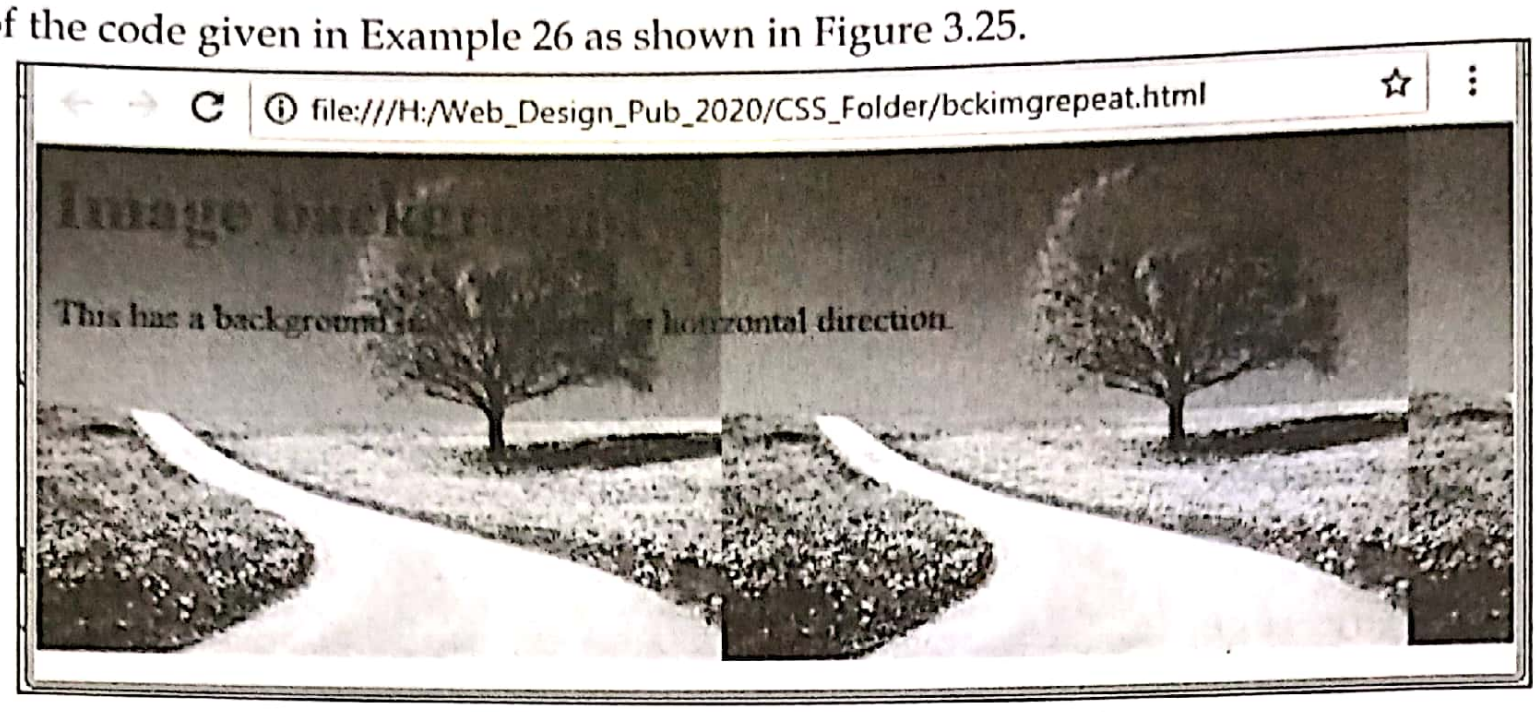


Figure 3.25: Output of the above Example 26

Background image Blur

To add a visual effect like a blur, it is better to use the filter property and set the filer as blur. This helps in setting a blur effect. If you want the background image more blurry, then set a higher value of blur else set a lower value. The default value of blur is 0.

Example 27:

```
<!DOCTYPE>
<html>
<head>
<title>CSS Properties Example</title>
<style>
 .img-blur{
 background-image:url(/blur.jpg);
 background-repeat:no-repeat;
 filter:blur(4px);
 height:900px;
  }
</style>
</head>
<body>
<h3>Example of Image Blur</h1>
<div class="img-blur"></div>
<p>If you want the background image more blurry, then set a higher value</P>
</body>
</html>
```

The output of the code given in Example 27 as shown in Figure 3.26.

Scanned with OKEN Scanner

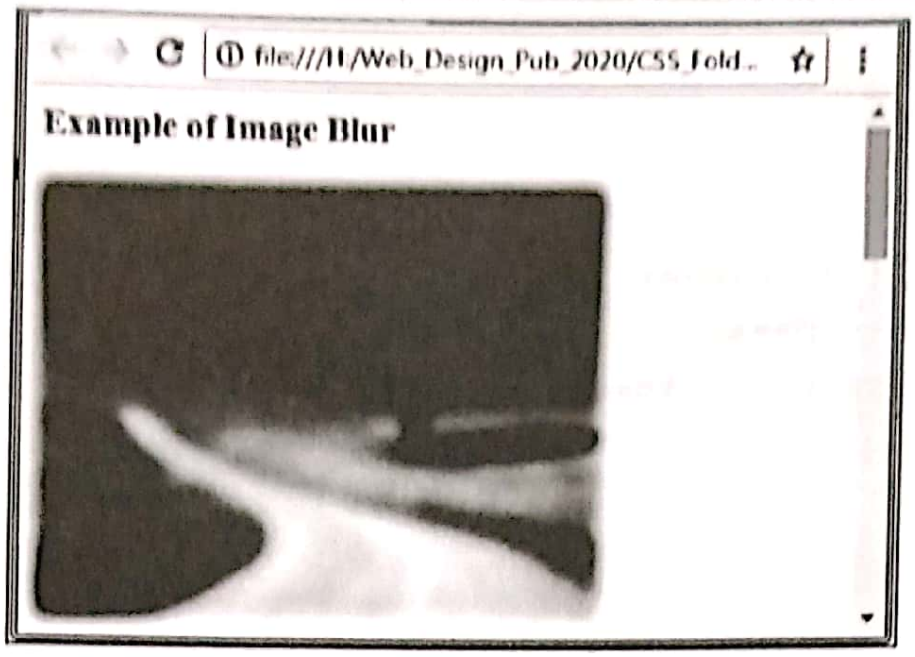


Figure 3.26: Output of the above Example 27

3.4.4 CSS Background Position

The background-position property is used to control the position of the background images. It is easy to use the positioning keywords such as left, right, top, and bottom and center. It define the vertical position of the background-image. It can be one of the following:

a. **Fixed:** Fixed value expressed in px, em.
b. **Percentage:** Percentage value.
c. **Top:** Image is vertically align to the top equivalent to 0% for the vertical position.
d. **Center:** Image is vertically center. Equivalent to 50% for the vertical position.
e. **Bottom:** Image is vertically align to the bottom. Equivalent to 100% for the vertical position.

The following example demonstrates to set the background image position in different ways:

Example 28:

```
<!DOCTYPE>
<html>
<head>
<title>CSS Properties Example</title>
<style>
 .imgpos1 {
background-image:url("H:/Flower.jpg");
background-repeat:no-repeat;
background-position:left;
width:100%;
height:250px;
}
.imgpos2 {
background-image:url("H:/Flower.jpg");
background-repeat:no-repeat;
background-position: 200px 50px;
```

(Contd...)

Scanned with OKEN Scanner

```
width:100%;
height:250px;
}
.imgpos3 {
background-image:url("H:/Flower.jpg");
background-repeat:no-repeat;
background-position:right bottom;
width:100%;
height:450px;
}
</style>
</head>
<body>
<h3>Example of CSS background Image Position</h1>
<b>background-position:left;</b>
<div class="imgpos1"> </div>
<b>background-position: 200 pixels and 50 pixels </b>
<div class="imgpos2"> </div>
<b>background-position:right bottom</b>
<div class="imgpos3"> </div>
<p>We have set the image position in different ways</P>
</body>
</html>
```

The output of the code given in Example 28 as shown in Figure 3.27.

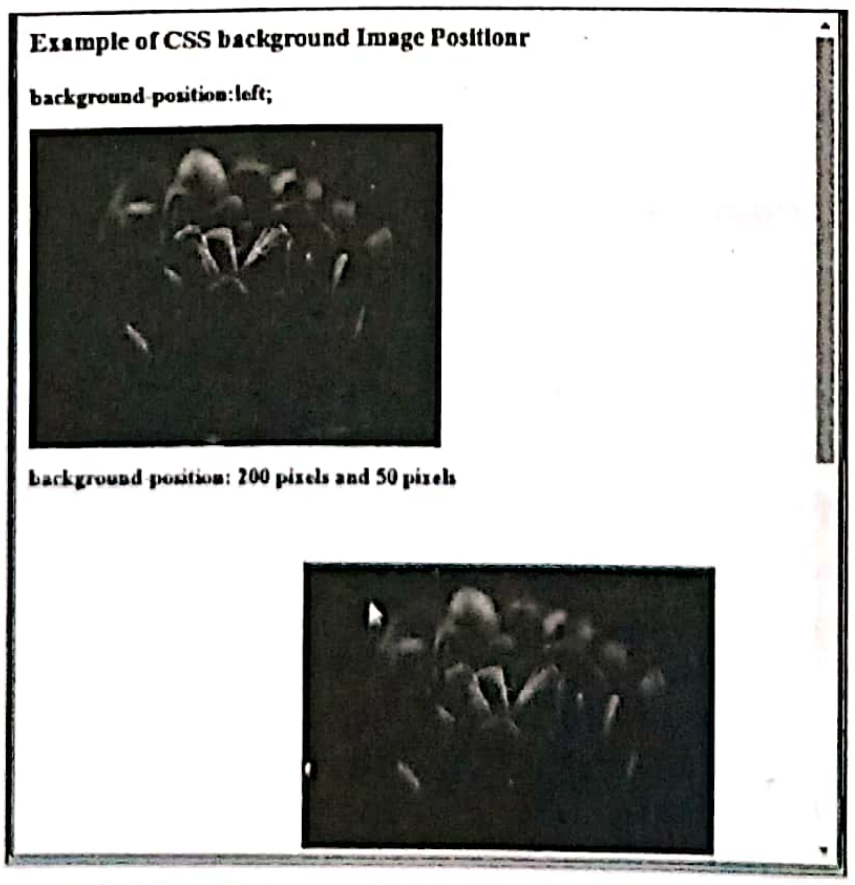


Figure 3.27: Output of the above Example 28

Scanned with OKEN Scanner

CSS Align Background Image Position X or Y

The background-position-y helps to set the background position along vertical direction only and position-x to set horizontal direction.

Example of background image position y

```
<style>
.img-pos-y1 {
background-image:url(img/flower.jpg);
background-repeat:no-repeat;
background-position-y:right;
background-size:300px;
height:250px;
}
.img-pos-y2 {
background-image:url(img/flower.jpg);
background-repeat:no-repeat;
background-position-y:bottom;
background-size:300px;
height:250px;
}
.img-pos-y3 {
background-image:url(img/flower.jpg);
background-repeat:no-repeat;
background-position-y:70%;
background-size:300px;
height:250px;
}
</style>
```

Let us test this above code and run it in the browser.

Background size

You will notice that we have already used the background-size in the previous examples so let us understand it exclusively. The background-size controls the size of the background images.

The Syntax of Background Size:

background-size:length(value) X%, auto, cover, contain

This is possible by providing 2 values for the background-size as below:

background-size:300px 70%;

/*Width of image is 300px & height is 70% of the parent element */

background-size: auto 20%;

/*Width of image will be automatically assigned by browser & height is 20% of the parent element*/

Note

- background-size: Auto is the default value.
- The length can be any valid length values like px, em, vw, etc.

Scanned with OKEN Scanner

- background-size: Cover sets the background image to the entire container by resizing it.
- background-size: Contain enables the image to be fully visible by resizing it.

Sometimes, it is difficult to say whether you should use **background-size:** Cover or background-size: contain. In that case, remember that –

background-size: cover makes the image as large as possible to cover the full area even if it has to stretch the image or cut the image at edges.

Just try switching from **background-size:** Ccover to background-size: contain and vice-versa and see which works best for you.

Example 29:

```
<!DOCTYPE html>
<html>
 <head>
<style>
     body, html {
      height: 100%;
      margin: 0;
     }
 div {
     background-image:
url("H:/cat1.jpg");
     height: 100%;
     background-position: center;
     background-repeat:        no-
repeat;
     background-size: cover;
     }
   </style>
 </head>
 <body>
   <div>
    <h2>Example    of    Full    Page
Cover Image</h2>
   </div>
 </body>
</html>
```

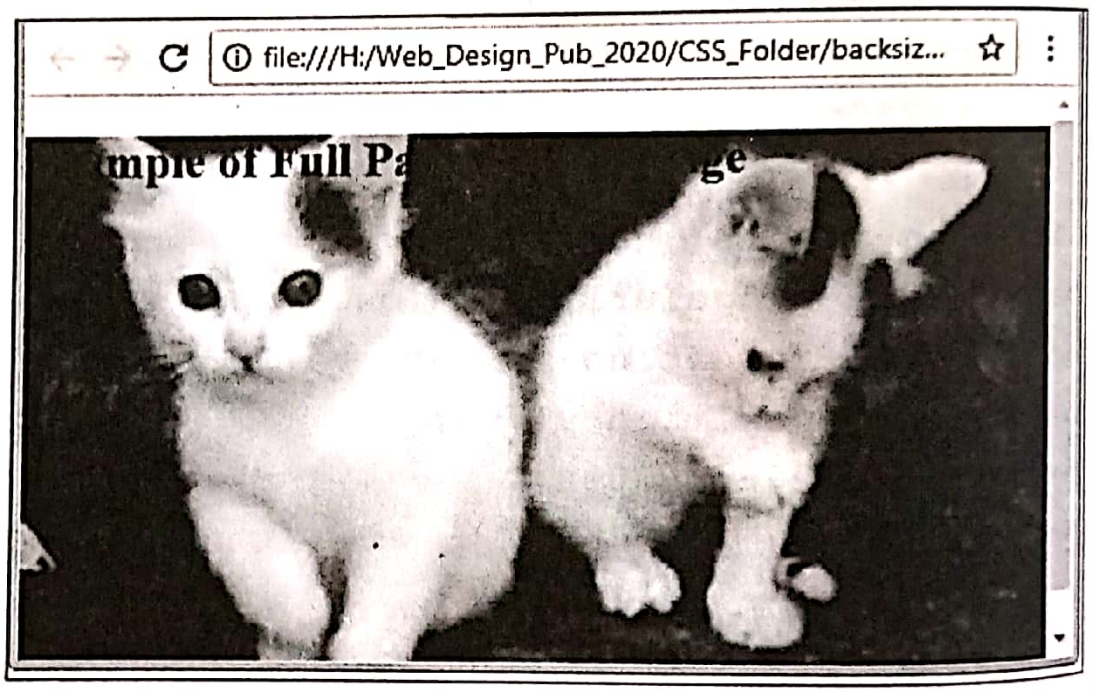


Figure 3.28: Output of the Example 29

The output of the code given in Example 29 as shown in Figure 3.28.

CSS Moving background image

To create a moving background image, you can use @keyframes and the animation property.

3.4.5 CSS Background-Attachment

The background-attachment sets whether the Background Image will be fixed, scroll when the page will scroll or scroll when the content is scrolled.

Scanned with OKEN Scanner

Example 30:

```
<!DOCTYPE>
<html>
<head>
<title>CSS Properties Example</title>
<style>
@keyframes moveimage {
from {
background-position: 0 100;
}
to {
background-position: -300px 0px;
}
}
@-webkit-keyframes moveimage {
from {
background-position: 0 100;
}
to {
background-position: -300px 0px;
}
}
.imgmoveval {
 text-align:center;
 background-image: url("H:/Flower.jpg");
 background-size: cover;
 height:400px;
 -webkit-animation: moveimage 20s linear 2;
 -webkit-animation-fill-mode: forwards;
animation: moveimage 20s linear 2; /*Set image moving seconds an how many
times it should move*/
animation-fill-mode: forwards; /*Set whether image should move forward or
backwards */
}
</style>
</head>
<body>
<div class="imgmoveval">
<h3>Moving background image</h3> </div>
</body>
</html>
```

Scanned with OKEN Scanner

Let us test the above code in the web browser.

a. **Fixed**—The background is fixed relative to the viewport (browser Window). Scrolling the contents of the element or scrolling the contents of the entire page will not cause the image to move.

b. **Scroll**—The background is fixed relative to the boundaries of the element's box. Scrolling the entire page (moving the placement of the element itself) will cause the image to move with the element.

c. **Local**—The background is fixed relative to the contents of the image. If the contents of the image are scrolled, the background will scroll with them. If the document as a whole is scrolled (causing the element content to be moved), the image will move with the element.

Example 31:

```
<!DOCTYPE html>
<html>
  <head>
    <style>
     .fixed-bg {
        top:10px;
        background-image:url("H:/col_range.jpg");
      width: 250px;
      height: 200px;
      background-repeat: no-repeat;
        background-position:0px 50px;
        background-color: lightsalmon;
        text-align:left;
        overflow:hidden;
      background-attachment: fixed;
      color:crimson;
      font-size:20px;
        padding-left:205px;
        border:2px solid black;
     }
    .scroll-bg {
      background-image:url("H:/col_range.jpg");
    width: 250px;
    height: 340px;
      overflow:auto;
    background-repeat: no-repeat;
      background-color:green;
      text-align:left;
    background-attachment: scroll;
    color:white;
    font-size:20px;
```

(Contd...)

```
    padding-left:205px;
border:2px solid black;
   }
   .local-bg {
     background-image:url("H:/col_range.jpg");
    width: 250px;
    height: 200px;
    background-repeat: no-repeat;
     background-color:yellow;
     overflow:auto;
     text-align:left;
    background-attachment: local;
     color:black;
     font-size:20px;
       padding-left:205px;
       border:2px solid black;
    }
   </style>
  </head>
  <body>
    <p>The background attachment sets whether the background image will be
fixes, scroll when the page will scroll or scroll when the content is
scrolled </p>
    <div class="main">
     <div class="fixed-bg">
      <b><u>background-attachment:fixed</u></b>: The background image will
be fixed even on scroll.
       The background-attachment property specifies whether the background
image should scroll or be fixed (will not scroll with the rest of the page):
Try to scroll and you will notice that the image will be fixed while the
content will scroll.
     </div>
     <div class="scroll-bg">
      <b><u>background-attachment:scroll</u></b>: The background-attachment
property specifies whether the background image should scroll or be fixed
(will not scroll with the rest of the page)  Try to scroll and you will
notice that the image will be fixed while the content will scroll. The
background-attachment property specifies whether the background image should
scroll or be fixed (will not scroll with the rest of the page)  Try to
scroll and you will notice that the image will be fixed while the content
will scroll.
       </div>
       <div class="local-bg">
```

(Contd...)

Scanned with OKEN Scanner

```
      <b><u>background-attachment:local</u></b>: The background image will
also scroll with the content in case you scroll up or down.

      The background-attachment property specifies whether the background
image should scroll or be fixed (will not scroll with the rest of the page):
Try to scroll and you will notice that the image will be fixed while the
content will scroll.
    </div>
   </div>
  </body>
</html>
```

The output of the code given in Example 31 as shown in Figure 3.28.

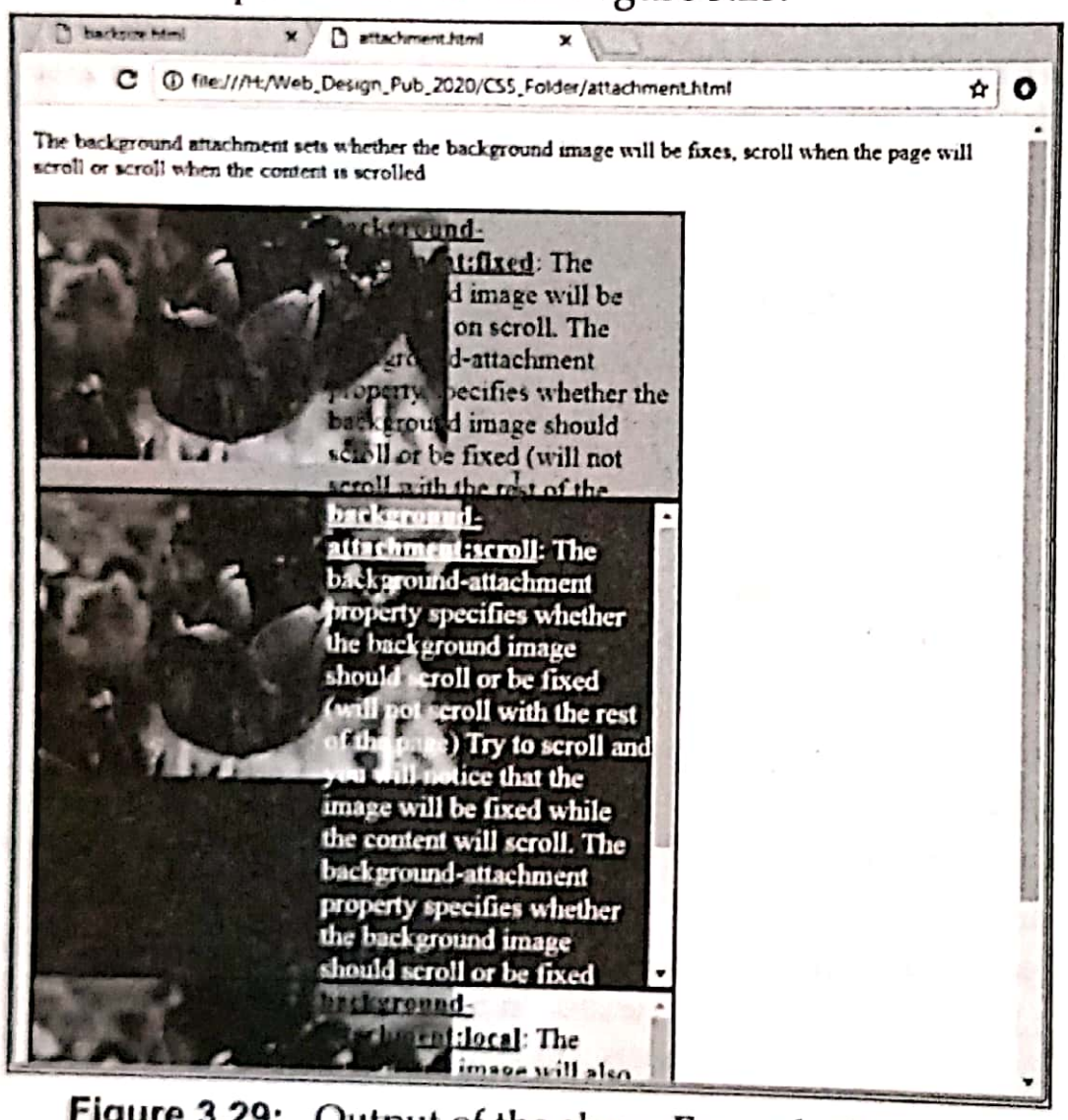


Figure 3.29: Output of the above Example 31

The background shorthand property

You can combine all of these rules into a single line labelled background.

body { background: #efefef url(/path/to/light-texture.png) repeat fixed top left; }

When using the shorthand property, you can omit any of the components just you use the default value.

When using the shorthand property the order of the property values is:

- background-color
- background-image
- background-repeat
- background-attachment
- background-position

Scanned with OKEN Scanner

Example 32:

```
<!DOCTYPE>
<html>
<head>
<title>CSS background shorthand property</title>
<style>
body {
 background: #f0e68c url("H:/Lotus.jpg") no-repeat right top; margin-right:
200px; width: 300px;
}
</style>
</head>
<body>
<h3>Lotus Temple</h3>
<p>Lotus is the symbol common to Hinduism, Buddhism, Jainism and Islam.
Around the blooming petals, there are nine pools of water, which light up,
in natural light. It looks spectacular at dusk when it is flood it.</p>
</body>
</html>
```

The output of the code given in Example 32 as shown in Figure 3.30.

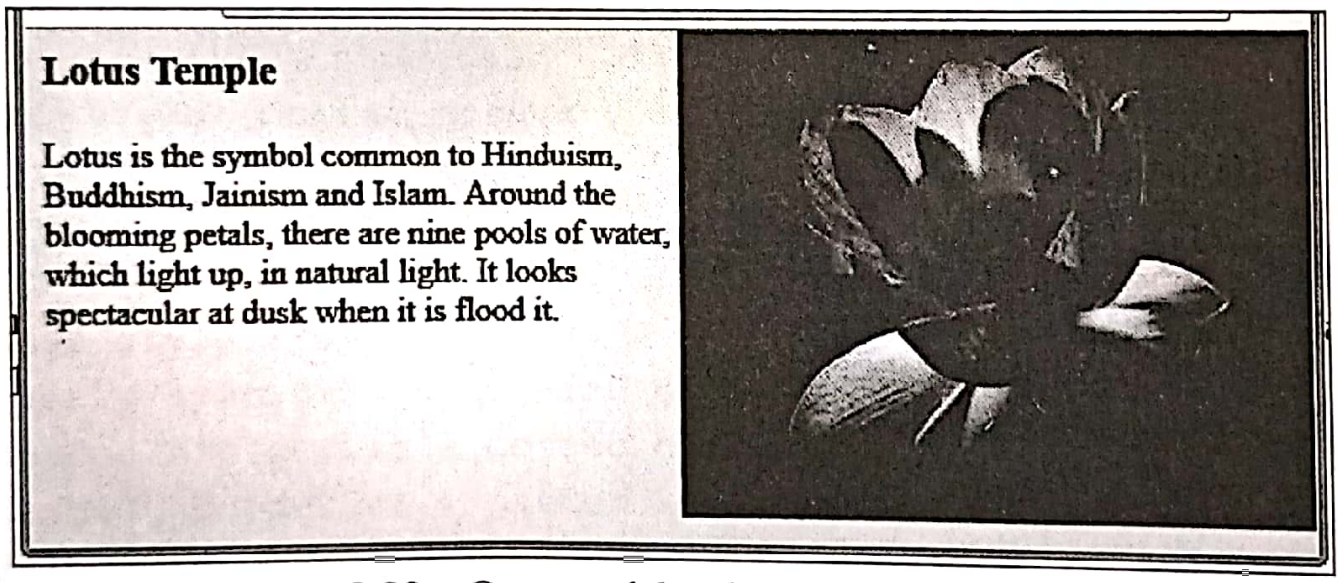


Figure 3.30: Output of the above Example 32

3.5 Border Properties

Use a CSS border to set up a border around the HTML elements.

Using CSS border, you can specify types of borders for your elements using border-style, border-color, border-width, border-image, and border-radius.

`Syntax of CSS-Border: border: border-width border-style border-color;`

This is a shorthand syntax of CSS border and there is another syntax for CSS borders as well as you will see on this web page. border-style property is **mandatory** and cannot be removed.

3.5.1 Border-style

This border-style property is used to set the appearance (style) of the borders. The default value is **none**. It includes the following style for the border:

Scanned with OKEN Scanner

- **DOTTED:** Dotted border
- **DASHED:** dash-line border.
- **SOLID:** Normal thickness of border.
- **DOUBLE:** Thickness of the border twice the normal thickness.
- **INSET:** Border look like that it is merged into the page.
- **OUTSET:** Creating an illusion that the inside of the element is raised above the page.
- **GROOVE**: The border style gets doubled.
- **RIDGE**: The border style gets fold.
- **HIDDEN:** The border style gets hidden.

If browser does not support all the border style such as inset, outset, then it will substitute a solid line in border-style.

For example,

```
P {border: border-width border-style border-color;}
```

The border-style property can have from one to four values (for the top border, right border, bottom border, and the left border).

The following examples shows different types of border style.

Example 33:

```
<!DOCTYPE html>
<html>
<head>
<style>
p.dotted {border-style: dotted;}
p.dashed {border-style: dashed;}
p.solid {border-style: solid;}
p.double {border-style: double;}
p.groove {border-style: groove;}
p.ridge {border-style: ridge;}
p.inset {border-style: inset;}
p.outset {border-style: outset;}
p.none {border-style: none;}
p.hidden {border-style: hidden;}
p.mix {border-style: dotted dashed solid double;}
</style>
</head>
<body>
<h2>The border-style Property</h2>
<p>This property specifies what kind of border to display:</p>
<p class="dotted">To define a dotted border.</p>
<p class="dashed">To define a dashed border.</p>
<p class="solid">To define a solid border.</p>
```

(Contd...)

Scanned with OKEN Scanner

```
<p class="double">To define a double border.</p>
<p class="groove">To define a groove border.</p>
<p class="ridge">To define a ridge border.</p>
<p class="inset">To define an inset border.</p>
<p class="outset">To define an outset border.</p>
<p class="none">This No border.</p>
<p class="hidden">To define a hidden border.</p>
<p class="mix">To define a mixed border.</p>
</body>
</html> <!DOCTYPE html>
<html>
<head>
<style>
p.dotted {border-style: dotted;}
p.dashed {border-style: dashed;}
p.solid {border-style: solid;}
p.double {border-style: double;}
p.groove {border-style: groove;}
p.ridge {border-style: ridge;}
p.inset {border-style: inset;}
p.outset {border-style: outset;}
p.none {border-style: none;}
p.hidden {border-style: hidden;}
p.mix {border-style: dotted dashed solid double;}
</style>
</head>
<body>
<h2>The border-style Property</h2>
<p>This property specifies what kind of border to display:</p>
<p class="dotted">To define a dotted border.</p>
<p class="dashed">To define a dashed border.</p>
<p class="solid">To define a solid border.</p>
<p class="double">To define a double border.</p>
<p class="groove">To define a groove border.</p>
<p class="ridge">To define a ridge border.</p>
<p class="inset">To define an inset border.</p>
<p class="outset">To define an outset border.</p>
<p class="none">This No border.</p>
<p class="hidden">To define a hidden border.</p>
<p class="mix">To define a mixed border.</p>
</body>
</html>
```

The output of the code given in Example 33 as shown in Figure 3.30.

Scanned with OKEN Scanner

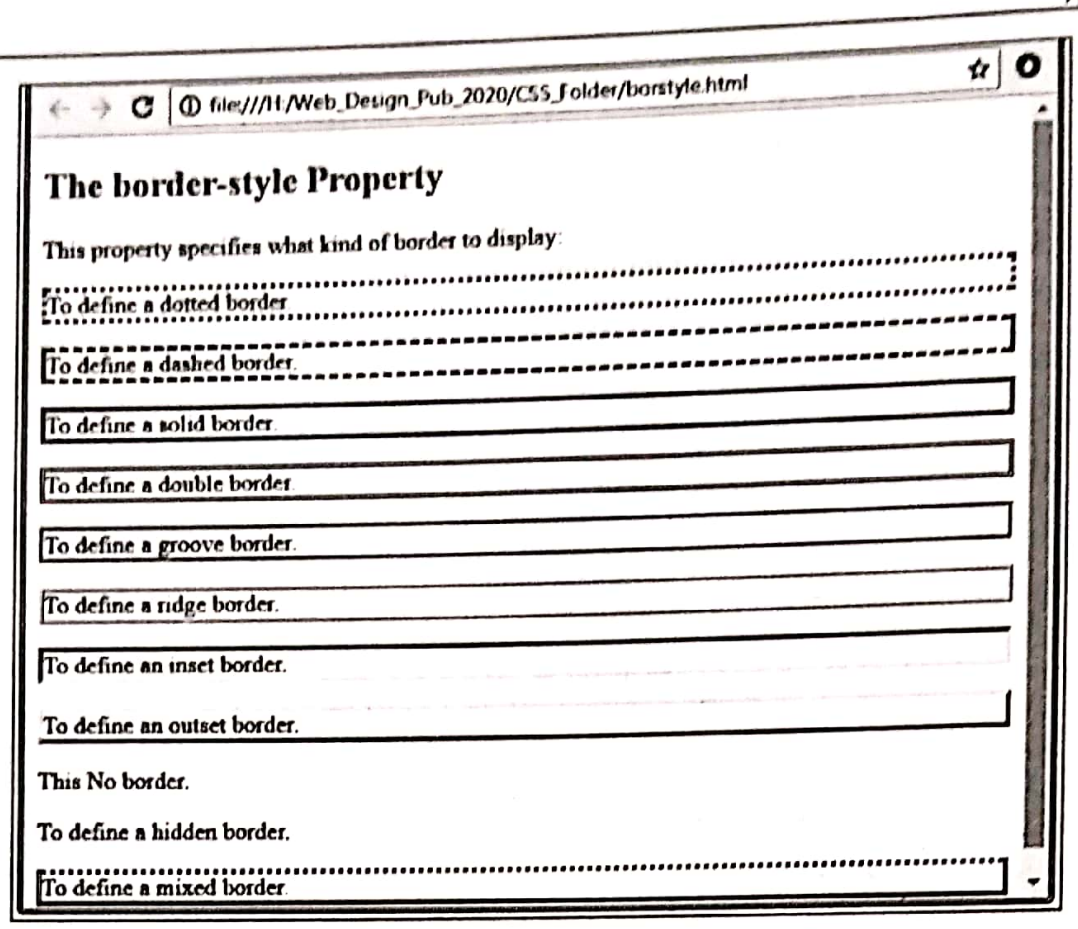


Figure 3.31: Output of the above Example 33

3.5.2 Border-width

Numerous properties are used to set the width of borders. Individual width of the border can be controlled by **border-top-width, border-right-width, border-left-width** and **border-bottom-width.** You can also set all four borders at once with border-width property. Width of the border can be specified by using thin, medium and thick (by absolute or relative measurement). For example:

```
P {border-style: solid; border-width: 10px}
P.double {border-style: dotted; border-width: thick}
```

You can individually change the width of the bottom, top, left, and right borders of an element using the following properties:

- border-bottom-width changes the width of bottom border.
- border-top-width changes the width of top border.
- border-left-width changes the width of left border.
- border-right-width changes the width of right border.

CSS border color for specifies sides:

There are four sides of borders that is left the border, top, right, and bottom border. You can also set different colors for specific borer. You can also set different colors for specific borders using different values of border property.

3.5.3 Border-color

This property is used to create a border using color that is different from the foreground color of the element. The border color uses the foreground color of the element as a default setting. Color values are specified by using either a supported color name or a numeric **RGB specification**. With this property, you can also provide a individual color for every side of the border. *For example*:

```
P {Border-Style: Solid; Border-Color: Green}
```

The border-color property can have from **one** to **four** values (for the **top border, right border, bottom border,** and the **left border**).

The following example shows all these border width and color:

Scanned with OKEN Scanner

Example 34:

```
<!DOCTYPE>
<html>
 <head>
 </head>
<body>
<p style = "border-width:5px; border-style:solid; border-color: red;">
 Solid border whose width is 5px and red color.
</p>
<p style = "border-width:5pt; border-style:solid; border-color: green;">
 Solid border whose width is 5pt and color is green.
</p>
<p style = "border-width:thin; border-style:solid; border-color:yellow;">
Solid border whose width is thin and color is yellow.
</p>
<p style = "border-width:medium; border-style:solid; border-color: brown;">
This is a solid border whose width is medium and color is brown.
</p>
<p style = "border-width:thick; border-style:solid; border-color: blue;">
This is a solid border whose width is thick and color is blue.
</p>
<p style = "border-bottom-width:5px;border-top-width:10px;
border-left-width: 2px;border-right-width:15px;border-style:solid;">
This is a border with four different width.
</p>
</body>
</html>
```

The output of the code given in Example 34 as shown in Figure 3.32.

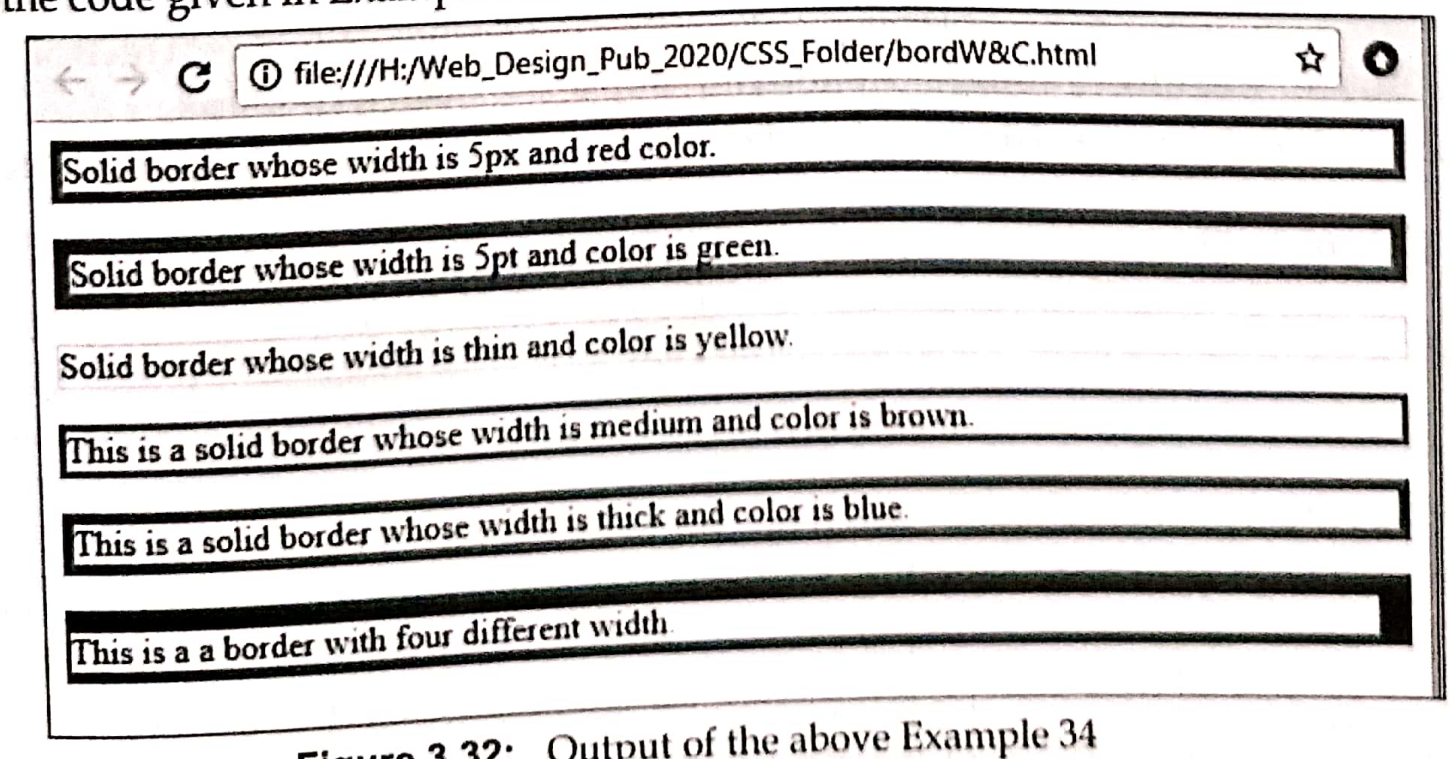


Figure 3.32: Output of the above Example 34

Scanned with OKEN Scanner

3.5.4 Border Properties Using Shorthand

The border property allows you to specify color, style, and width of lines in one property.

The following example shows how to use all the three properties into a single property.

Example 35:

```
<!DOCTYPE html>
<html>
<head>
<style>
p {
border: 5px dashed red;
}
</style>
</head>
<body>
 <p>This example is showing shorthand property for border width, style and color</p>
</body>
</html>
```

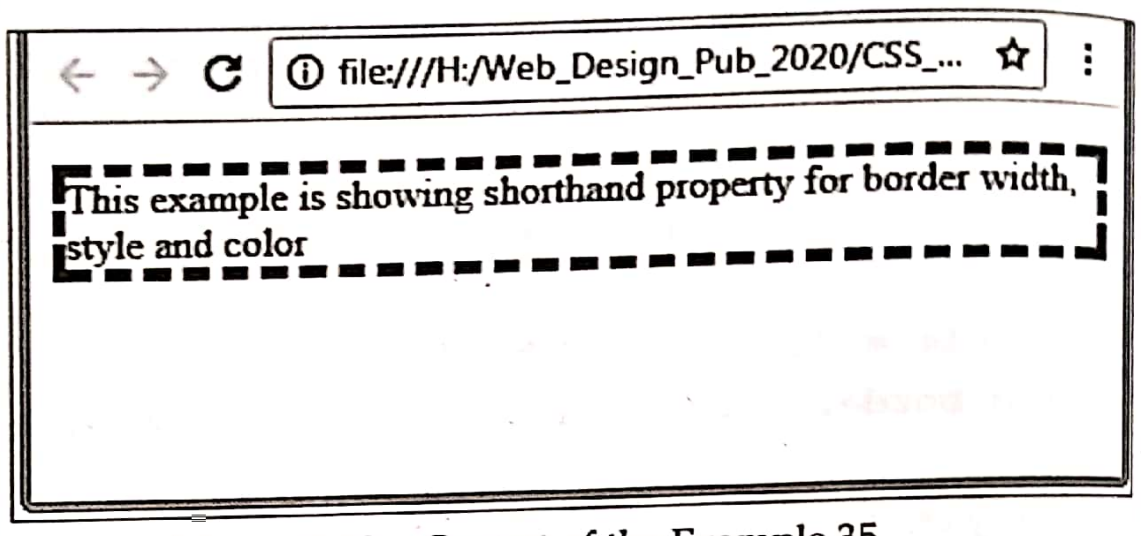


Figure 3.33: Output of the Example 35

The output of the code given in Example 35 as shown in Figure 3.33.

3.5.5 CSS border radius

If you want to set the round borders, then you should use the border-radius property of CSS. You can also set the shadow for the borders using a box-shadow property of CSS.

Example 36:

```
<!DOCTYPE html>
 <html>
 <head>
   <style>
   #Div1 {
     width:50%;  height:260px;   padding:50px;    margin:20px;
     background-color:  gold;    border-radius:50%;      border:  20px  solid
blue;
     text-align:left;
   }
   </style>
 </head>
 <body>
  <div id="Div1">
   <p style=font-size:12px;>Border Radius</p>    </div>
 </body>
</html>
```

The output of the code given in Example 36 as shown in Figure 3.34.

Scanned with OKEN Scanner

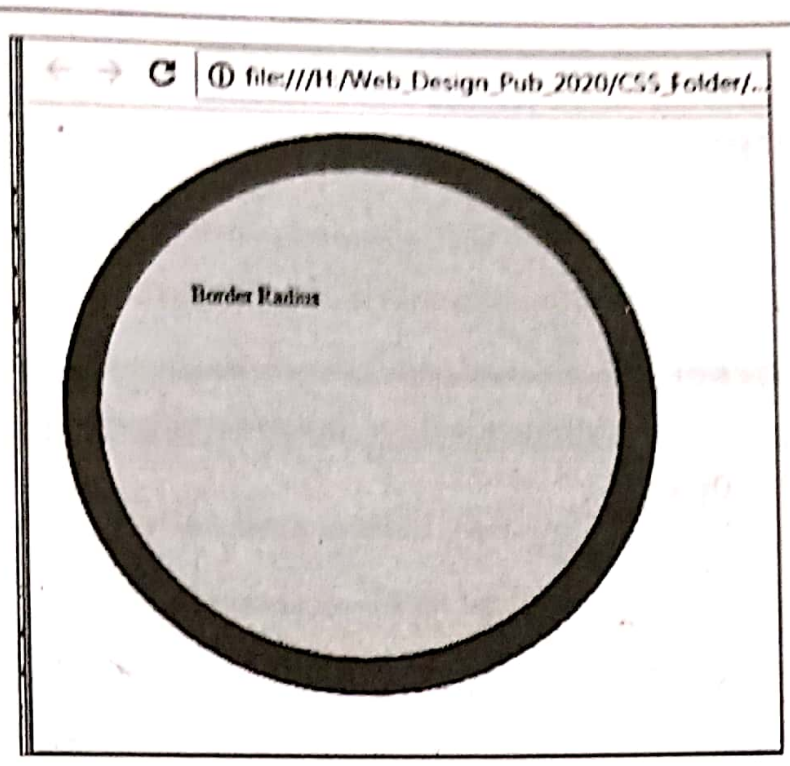


Figure 3.34: Output of the above Example 36

You can also set the shadow for the borders using a box-shadow property of CSS and it adds shade around the borders.

It uses these following attributes:

X-offset: X-offset is the horizontal displacement of the shadow of the box.

Y-offset: Y-offset is the vertical displacement(offset along the Y-Axis) of the shadow of the box.

Color: Color is always optional in box-shadow. color defines the color of the shadow and by default, it takes the color of the text.

- blur-radius: It makes the shadow little dim and hazy.
- spread-radius: It makes the shadow bigger or smaller.
- Inset: It will create a shade from the outer side towards the inner side
- Initial: It will acquire the attributes and properties from its parent.
- Inherit: It always optional.

Note:

If x-offset > 0, then the shadow will be towards the right-hand side of the box.

If the x-offset< 0, then the shadow will be towards the left-hand side of the box.

If y-offset > 0, then the shadow will be towards the bottom side of the box which means the shade will be below the box.

y-offset<0, then the shadow will be towards the top side of the box which means the shade will be at the top of the box.

Example 37:

```
<!DOCTYPE html>
<html>
 <head>
  <style>
  #borderl {
    border: 2px solid red;
```

(Contd...)

Scanned with OKEN Scanner

```
    padding: 10px;
    box-shadow: 20px 05px;
  }

  #border2 {
    border: 4px solid green;
    padding: 10px;
    box-shadow: -20px -10px;
  }
  </style>
</head>
<body>
 <div id="border1">
   <h4>Horizontal offset is 20px and vertical offset is 5px.</h4>
   <p>Positive values of an x-offset shadow towards the right side along the
x-axis</p>
   <p>Positive values of a y-offset shadow towards the bottom side along the
y-axis</p>
 </div>
 <br>
 <br>
 <div id="border2">
   <h4>Horizontal offset is -20px and vertical offset is -10px.</h4>
   <p>Negative values of an x-offset shadow towards the left side along the
x-axis</p>
   <p>Negative values of a y-offset shadow towards the top side along the y-
axis</p>
 </div>
 </body>
</html>
```

The output of the code given in Example 37 as shown in Figure 3.35.

3.6 Block Properties

A block element always starts on a new line, and fills up the horizontal space left and right on the web page. The block value of the display property forces an element to behave like block-level element, like a or <p> element.

The display property decides how the browser will display an element.

There are important display properties. These are:

1. Inline
2. Block
3. Inline-block
4. None and table

Scanned with OKEN Scanner

Horizontal offset is 20px and vertical offset is 5px.

Positive values of an x-offset shadow towards the right side along the x-axis

Positive values of a y-offset shadow towards the bottom side along the y-axis

Horizontal offset is -20px and vertical offset is -10px.

Negative values of an x-offset shadow towards the left side along the x-axis

Negative values of a y-offset shadow towards the top side along the y-axis

Figure 3.35: Output of the above Example 37

Inline elements are continues with the flow of the text and does not start in the new line until you are forcing it to start in a new line. It does not accept the width and height values. Some of the inline elements are:

, <a>, <b>, , and . The style rules in the following example displays the <p> and <li> elements as inline-level elements:

Example 38:

```
<!DOCTYPE html>
<head>
<title>Example of CSS display</title>
<style>
 p {
 display: inline;
 background: #8FBC8F;
 padding: 10px;
}
ul li {
 display: inline;
 margin:10px;
}
</style>
</head>
<body>
<p>Display a list of links as a horizontal menu</p>
<ul>
<li>HTML</li> <li>JAVA</li> <li>CSS</li> <li>JavaSCRIPT</li> </ul>
</body>
</html>
```

The output of the code given in Example 38 as shown in Figure 3.36.

Scanned with OKEN Scanner

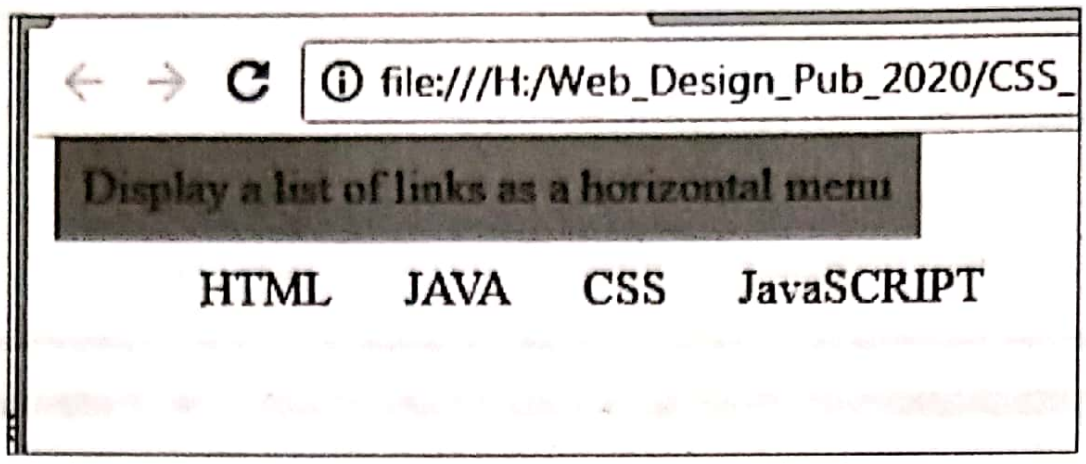


Figure 3.36: Output of the above Example 38

Display: Block

The block elements are those which breaks the normal flow of the text and starts in new line. This property is used as the default property of div. This property places the div one after another vertically. Height and width of the div can be changed using the block property if the width is not mentioned, then div under block property will take up the width of the container.

Example 39:

```
<!DOCTYPE html>
<html>
<head>
<style>
 div {
  width:70px;
  marign:5px;
  height:50px;
  display:block;
  text-align:left;
  background:gold;
  border:2px solid black;
  }
</style>
</head>
<body>
<h3>Example of display: Block property</h3>
<div>Block 1</div>
<div>Block 2</div>
<div>Block 3</div>
<div>Block 4</div>
<div>Block 5</div>
</body>
</html>
```

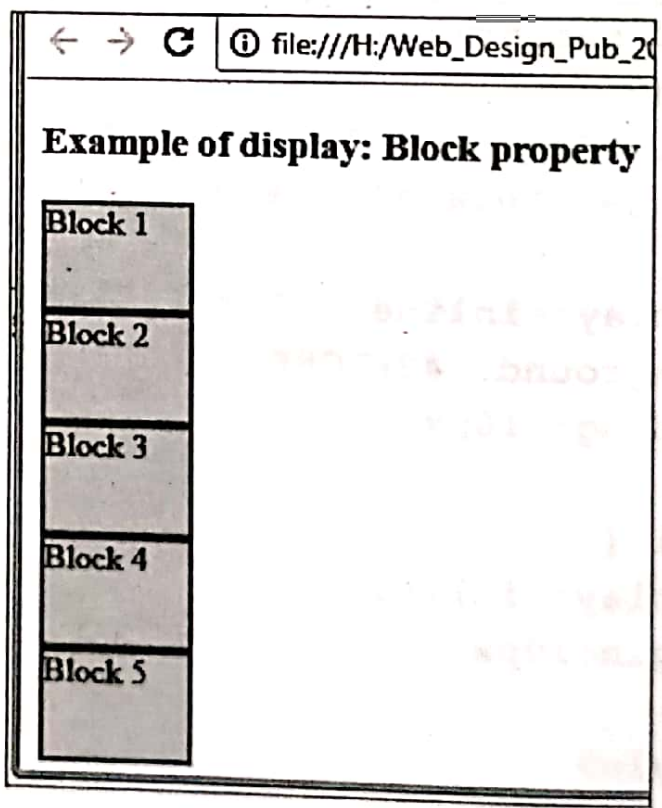


Figure 3.37: Output of the Example 39

The output of the code given in Example 39 as shown in Figure 3.37.

Scanned with OKEN Scanner

Inline-Block Element

The inline-block elements have some behavior similar to inline elements in a way that it continues with the flow of the text. The inline-block value of the display property causes an element to generate a block box that will be flowed with surrounding content that is the same line as adjacent content. The following example shows style rules displays the and elements as inline-block.

Example 40:

```
<!DOCTYPE html>
<html>
<head>
<style>
img {
 width:200px;
 height:200px;
 margin:5px;
 display:inline-block;
 }
</style>
</head>
<body>
<h3>Example of Inline-Block property</h2>
<img src="h:/Greenapple.jpg">
<img src="h:/Apple1.jpg">
</body>
</html>
```

The output of the code given in Example 40 as shown in Figure 3.38.

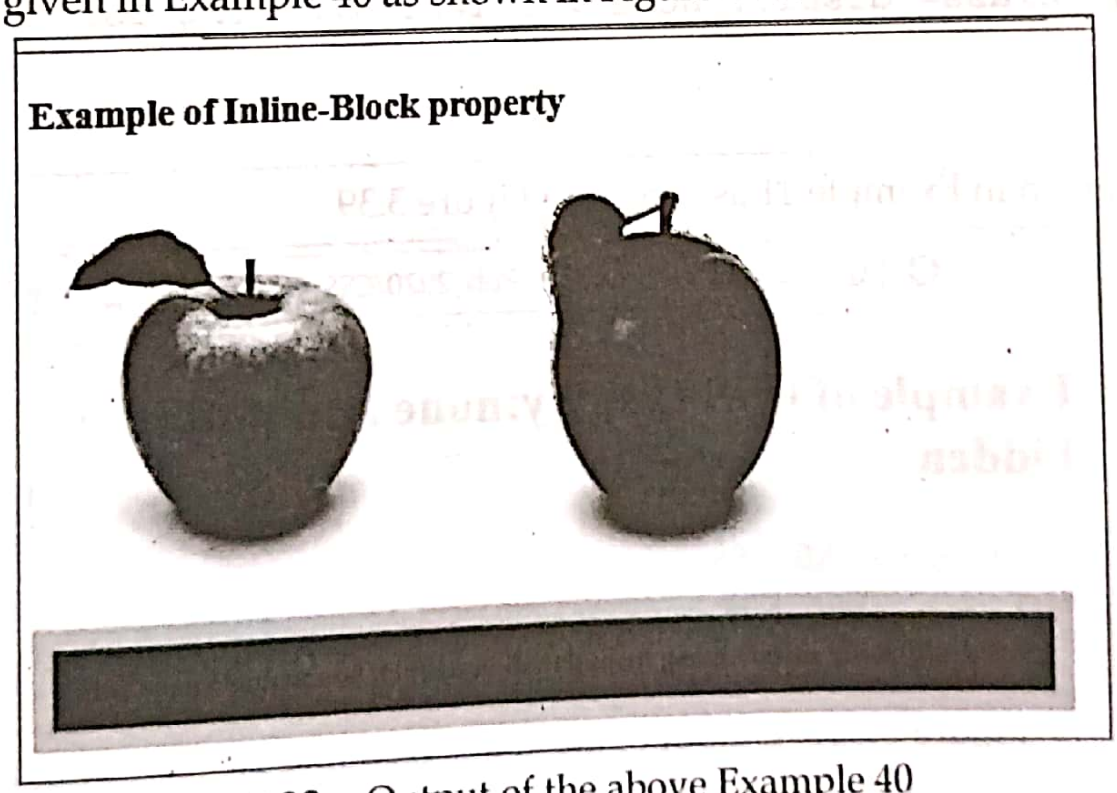


Figure 3.38: Output of the above Example 40

CSS display: none style

CSS display type none means element is not display, element is no longer displayed.

Display: None, Visibility and Hidden

Display: None makes sure that the element is not display. It also removes the allocated space for the element.

Scanned with OKEN Scanner

Visibility: Hidden hides the element but it does not remove the extra space of the element which is hidden. For example, if you want to keep the extra space, then use visibility: hidden else use display: none property.

Example 41:

```
<!DOCTYPE html>
<html>
<head>
<style>
 .display-hidden {
  visibility: hidden;
 }
.display-none {
 font-size: 20px;
 display: none;
}
</style>
</head>
<body>
<h2>Example of CSS Display:none and visibility: hidden</h2>
<div>
Easy <a href="#,">to learn HTML</a> CSS </div>
<div>
Easy <a href="#" class="display-hidden">to learn HTML</a>CSS </div>
<div>
Easy <a href="#" class="display-none">to learn</a>HTML CSS </div>
</body>
</html>
```

The output of the code given in Example 41 as shown in Figure 3.39.

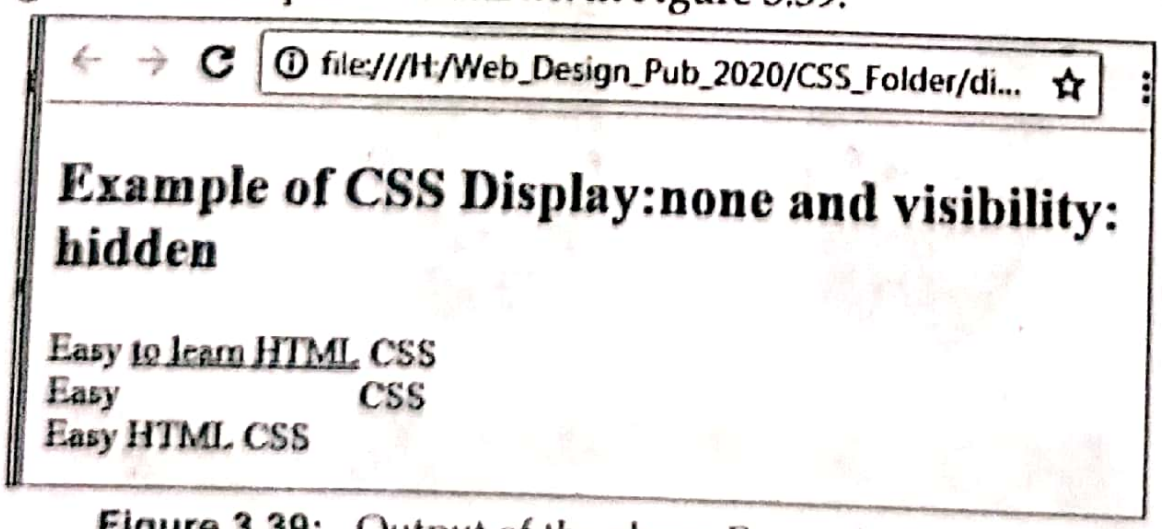


Figure 3.39: Output of the above Example 41

3.7 CSS BOX Properties

CSS box model is a container that contains multiple properties including borders, margin, padding and the content. It is used to create and layout of web pages. It can be used for customizing the layout of different elements. There are four sides of the area, so each side of the box model has-

Scanned with OKEN Scanner

- Top-margin, right margin, a bottom margin and left margin.
- Top-border, right border, bottom border and left border.
- Top-padding, right padding, bottom padding and left padding.
- Width and height of the content area.

Example of CSS Box model

For example, the element is surrounded with these values,

Width: 400px; /* width of the content area */

Height: 50px; /* height of the content area */

Padding: 5px 10px; /* space between outer edge of the content to inner edge of border */

Border: 4px blue; /* space between outer edge of padding to inner edge of margin */

Margin: 5px 8px; /* space between outer edge of border to inner edge of margin itself */

To calculate the total length:

Total width of element = width of the content + left padding + right padding + left border + right border + left margin + right margin

= 400px + 10px + 10px + 4px + 4px + 8px + 8px = 444px

Total height of element = height of the content + top padding + bottom padding + top border + bottom border + top margin + bottom margin

= 50px + 5px + 5px + 4px + 4px + 5px + 5px =78px

Example 42:

```
<!DOCTYPE html>
<html>
<head>
<style>
 .box_model {
  width: 600px;
  height: 70px;
  padding: 10px 15px;
  border:10px solid pink;
  margin: 15px 10px;
}
 .box_model1 {
  width: 620px;
  height: 70px;
  padding:10px 10px;
  border:10px solid powderblue;
  margin: 05px 05px;
}
</style>
</head>
```

(Contd...)

Scanned with OKEN Scanner

```
<body>
<h2>There is two boxes with different width of the content but same total
length of the element</h2>
<div class="box_model">
<p  style=background-color:lightgreen;text-align:center;>The  width  of  the
area is 700px but the total width of the element is 660px.</p>
</div>
<div class="box_model1">
<p style=background-color:yellow;text-align:center;>The width of the area is
720px but the total width of the element is 660px.</p>
</div>
</body>
</html>
```

The output of the code given in Example 42 as shown in Figure 3.40.

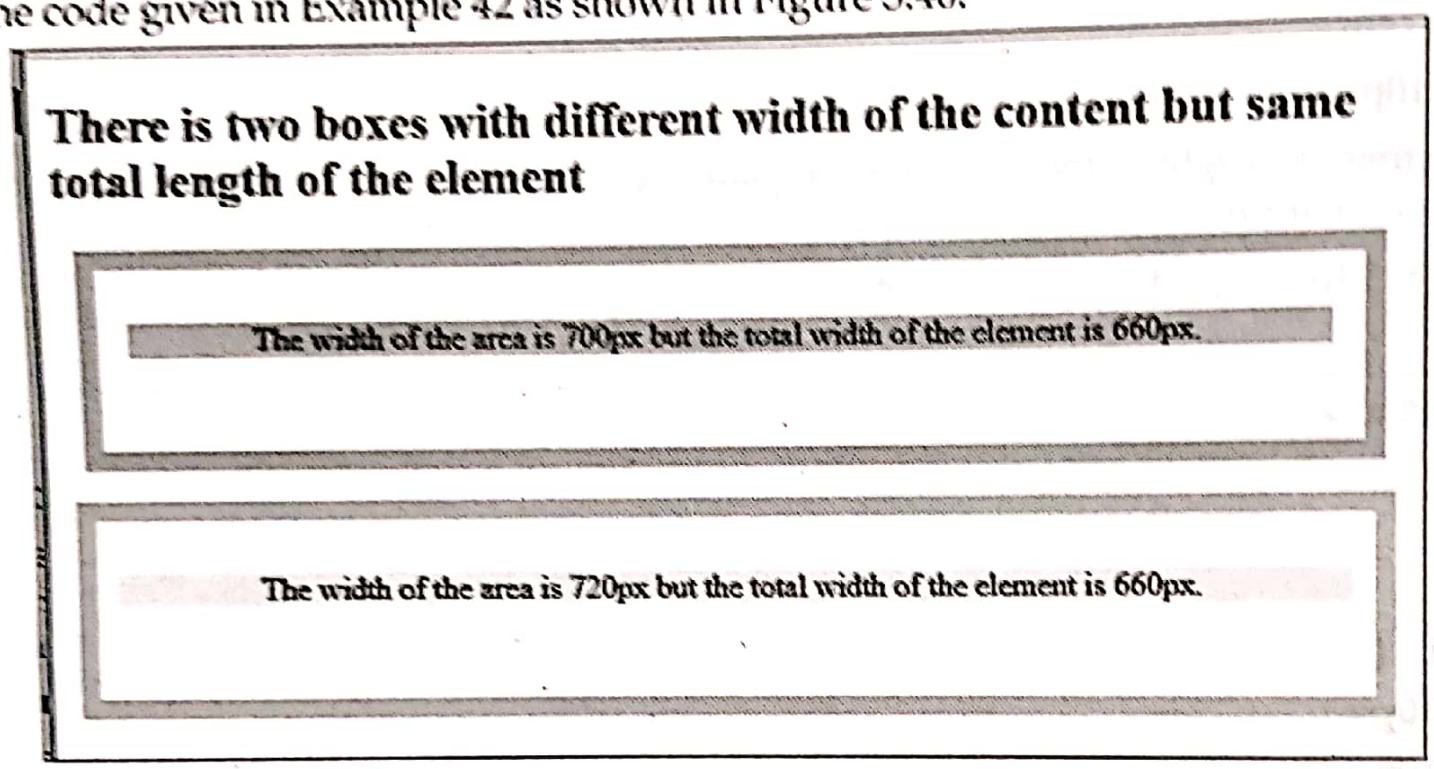


Figure 3.40: Output of the above Example 42

3.7.1 Positioning Properties

The position property set an element positioning to display in web page. This position property always defines what type of position method will be used an element. It is used to define how an element is positioned.

There are **five** important types of positioning method used for an element. These are:

a. Static
b. Fixed
c. Relative
d. Absolute and
e. Sticky

Elements have positioned using the **top, bottom, left,** and **right** properties. However, these properties will not work unless the position property is set. It also work differently depending on the position value.

CSS Static Position: A static position is the default position. Static positioned elements are not affect by the top, bottom, left, and right and z-index properties.

Scanned with OKEN Scanner

Example 43:

```
<!DOCTYPE>
<html>
<head>
<style>
 .default {
  border-width: 5px;
  border-style:ridged;
  background-color: gray;
  border: dotted;
 }
.css-position-static {
 border: 2px dashed red;
 padding:10px;
 position:static;
}
</style>
</head>
<body>
<h3>CSS Position Static</h3>
<div class="default">
<p>The div is a default position</p>
</div>
<br>
<div class="CSS-position-static">
<p>An element with position: static. It is not positioned according to the
normal flow of the web page</p>
</div>
</body>
</html>
```

The output of the code given in Example 43 as shown in Figure 3.41.

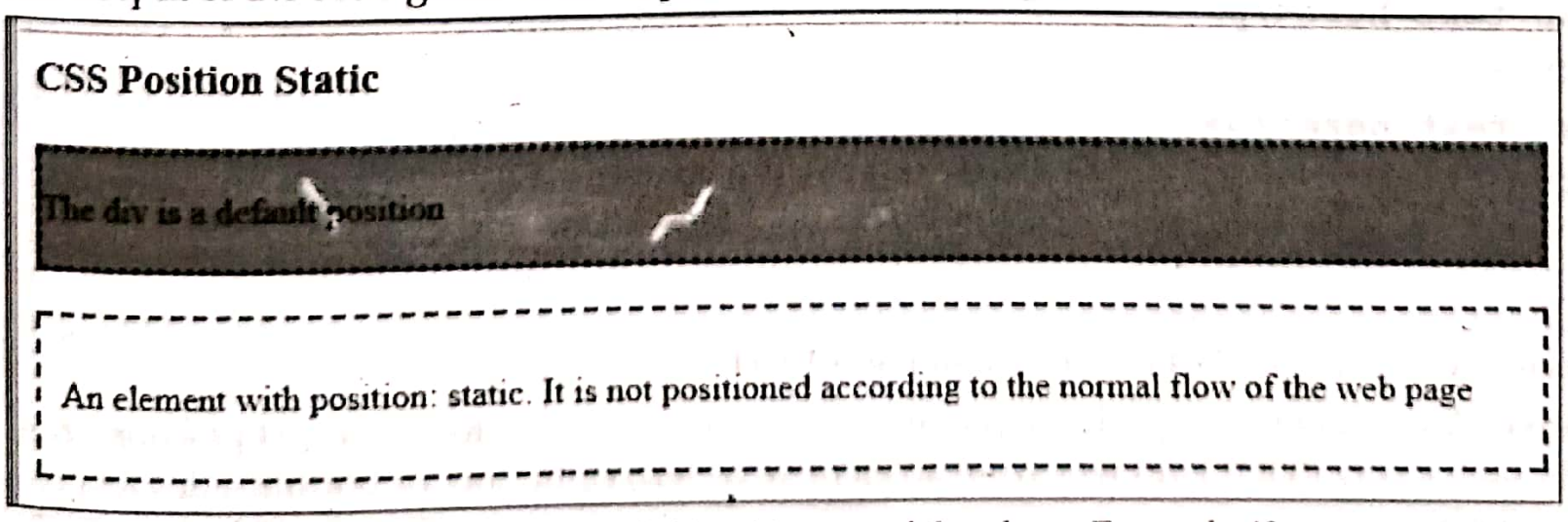


Figure 3.41: Output of the above Example 43

Scanned with OKEN Scanner

CSS Position: fixed property: To set element is fixed even if Window scroll vertically or horizontally element is fixed place. CSS position fixed followed by the Window offset **top, right, bottom** and **left.**

Example 44:

```
<!DOCTYPE>
<html>
<head>
<style>
 .css-position-fixed {
 border: 2px dashed red;
 background-color:gold;
 position:fixed;
 top: 90px;
 left: 45px;
 text-align:center;
 padding:5px;
 }
</style>
</head>
<body>
<h3>CSS Position fixed</h3>
<div class="CSS-position-fixed">
<p>This is the fix positioned text</p>
</div>
<br>
<p>Scroll down text here</p>
<br><br><br><br>
<p>Scroll down text here</p>
 <br><br><br><br>
<p>Scroll down text here</p>
 <br><br><br><br>
<p>Scroll down text here</p>
<br><br><br><br>
<p>Scroll down text here</p>
<br><br><br><br>
</body>
</html>
```

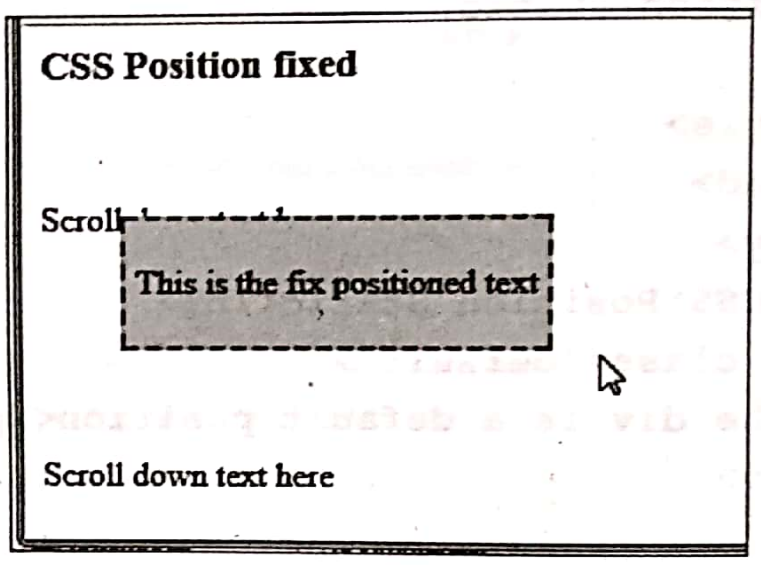


Figure 3.42: Output of the Example 44

The output of the code given in Example 44 as shown in Figure 3.41.

CSS Position: Relative: An element with position: relative is positioned relative to its normal position. You can set an element in such a way that it goes way from an element in **left, right, top,** or **bottom** sides relative to the default normal position.

Scanned with OKEN Scanner

Example 45:

```
<!DOCTYPE>
<html>
<head>
<style>
  .default {
  background-color: lightpink;
  border: solid;
 }
 .css-position-relative_top {
  border: 2px dashed gold;
  position: relative;
  top: 20px;
  padding:10px;
  background-color:orange;
}
 .css-position-relative_right {
  border: 2px dotted blue;
  position:relative;
  right: 25px;
  padding-left: 30px;
  background-color:lightgreen;
}
 .css-position-relative_bottom {
  border: 2px double black;
  position: 1.5em;
  background-color:yellow;
 }
 .css-position-relative_left {
  border: 2px ridge brown;
  position:relative;
  left: -18px;
  padding-left:30px;
  background-color: lightyellow;
}
</style>
</head>
<body>
<div class="default">
<p>This is the default position</p>
```

(Contd...)

Scanned with OKEN Scanner

```
</div>
<br>
<div class="css-position-relative_top">
<p>This moves the div 20px away from top relative to default position so this will shift the div towards the bottom of the page relative to its default position</p>
</div>
<br><br><br>
<div class="css-position-relative_right">
<p>This moves the div 25px away from right relative to default position so this will shift the div towards the left of the page relative to its default position</p>
</div>
<br><br><br>
<div class="css-position-relative_bottom">
<p>This moves the div 1.5em away from bottom relative to default position so this will shift the div towards the top of the page relative to its default position</p>
</div>
<br><br><br>
<div class="css-position-relative_left">
<p>This moves the div px away from top relative to default position so this will shift the div towards the bottom of the page relative to its default position</p>
</div>
</body>
</html>
```

The output of the code given in Example 45 as shown in Figure 3.42.

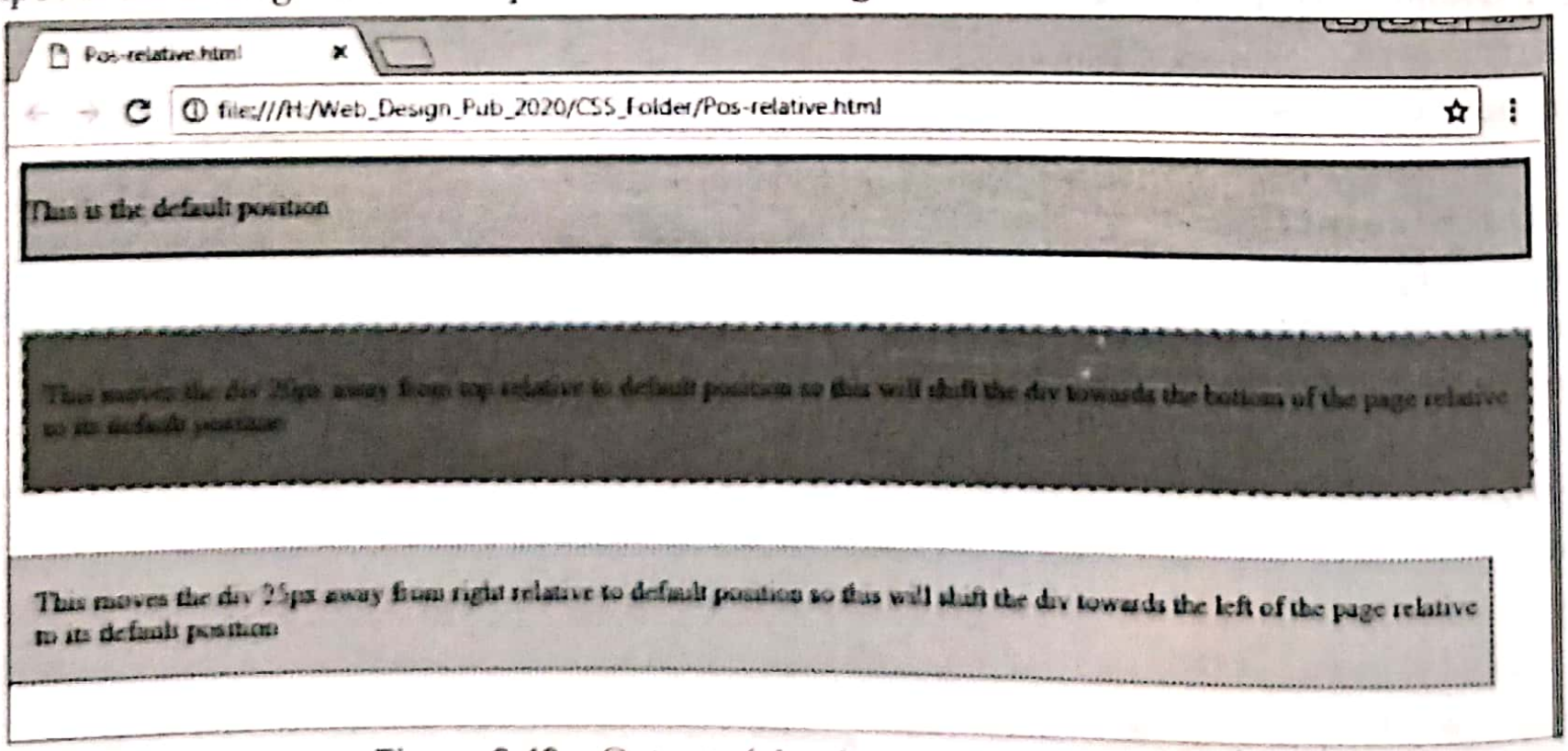


Figure 3.43: Output of the above Example 45

Scanned with OKEN Scanner

CSS Position: Absolute: An element with position: absolute; is used to position an element relative to the first parent element that has a position other than static. If no such element is found, it will have positioned on a page relative to the top-left corner of the browser Window.

Example 46:

```
<!DOCTYPE html>
<html>
<head>
<style>
 div.relative {
  position: relative;
  width: 400px;
  height:200px;
  border: 3px solid #73AD21;
 }
 div.absolute {
  position: absolute;
  top: 80px;
  right: 0;
  width: 200px;
  height: 100px;
  border: 3px solid #73AD21;
}
</style>
</head>
<body>
<h2>Position: absolute</h2>
<p>An element with position: absolute is positioned to relative to the nearest positioned ancestor</p>
<div class="relative">This div element has position: relative
<div class="absolute">This div element has position: absolute</div>
</div>
</body>
</html>
```

The output of the code given in Example 46 as shown in Figure 3.44.

CSS Position: Sticky: It enables an element to scroll down(if the page can be scrolled) up to a fixed point like **position:relative** and then it gets fixed at a certain position after you have scrolled till a fixed area stars behaving like **position:fixed**. It is a mixture of relative and fixed position but you have to keep in mind about the offset position compared to the nearest element in the scrolling area.

3.8 CSS Lists

There are two types of CSS lists.

- **Ol:** Ordered list which follows a sequence/order like numbers, alphabets.

Scanned with OKEN Scanner

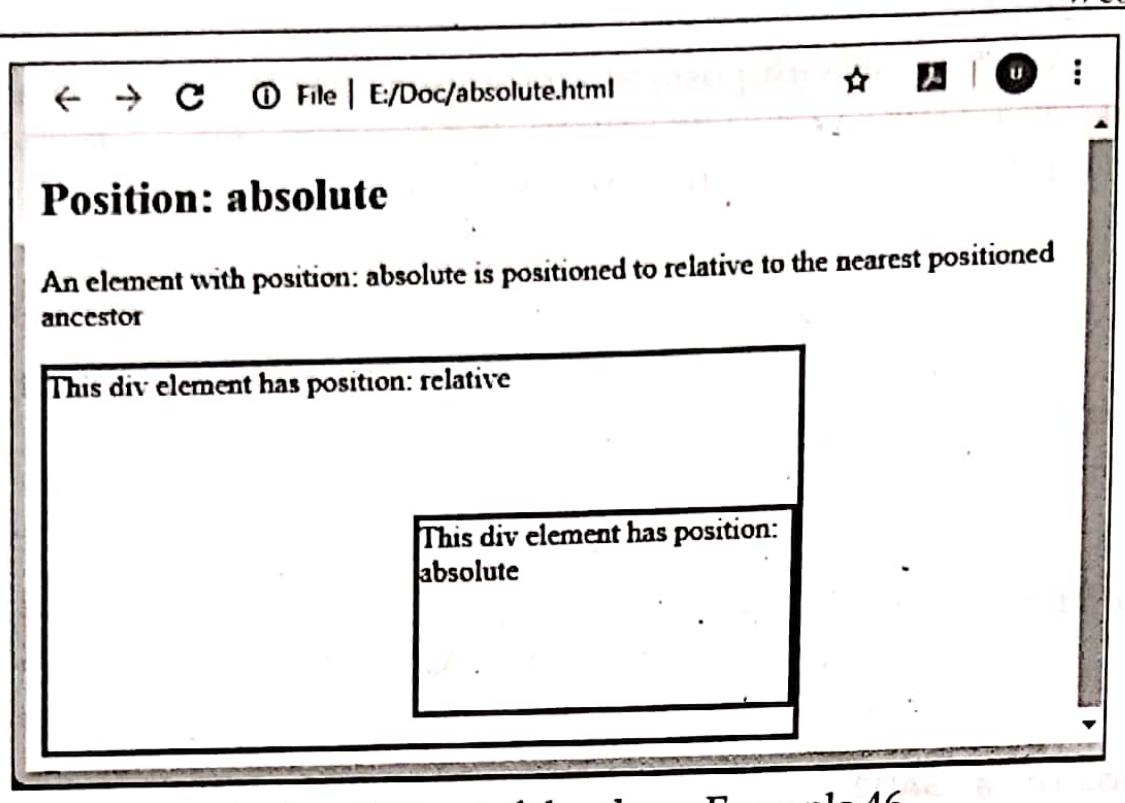


Figure 3.44: Output of the above Example 46

- **Ul:** Unordered list items are marked with round bullets.

There are three important style properties for **ordered** and **unordered** list.

- **List-style-type:** type of bullet points.
- **List-style-position:** To set the position of the bullet points.
- **List-style-image:** to set a background image for the bullet list type.

Syntax of list style type is

List-style-type: decimal | decimal-leading-zero | lower-roman | upper-roman | lower-alpha | upper-alpha. For ordered list, the default style type is **decimal**. For example, you have different type of list style type.

Example 47:

```
<!DOCTYPE html>
<html>
<head>
<title>List Style type for ordered list</title>
<style>
 #ol-d {
list-style-type: decimal;
}
#ol-d-lz {
list-style-type: decimal-leading-zero;
 }
#ol-lr {
list-style-type: lower-roman;
}
#ol-ur {
list-style-type: upper-roman;  }
```

(Contd...)

Scanned with OKEN Scanner

```
#ol-la {
 list-style-type: lower-alpha;
 }
 #ol-ua {
 list-style-type: upper-alpha;
 }
</style>
</head>
<body>
<ul id="ol-d">
 <li>English</li>  <li>Maths</li>  <li>Science</li>  <li>Computer</li>
 <li>History</li> </ul>
<hr>
<ul id="ol-d-lz">
 <li>Apple</li>  <li>Orange</li>  <li>Banana</li>  <li>Grapes</li>
 <li>Pineapple</li> </ul>
<hr>
<ul id="ol-lr">
 <li>Laptop</li>  <li>Mobile</li>  <li>Smartwatch</li>  <li>TV</li>
 <li>Washing Machine</li> </ul>
<hr>
<ul id="ol-ur">
 <li>Pen</li>  <li>Pencil</li>  <li>Mouse</li> <li>Keyboard</li>
<li>Harddisk</li> </ul>
<hr>
<ul id="ol-la">
 <li>Pendrive</li>  <li>LCD</li>  <li>LED</li>  <li>Monitor</li>
 <li>Mouse</li> </ul>
<hr>
<ul id="ol-ua">
<li>Peacock</li> <li>Crown</li> <li>Rabbit</li> <li>Snake</li> <li>Deer</li>
</ul> <hr>
</body>
</html>
```

The output of the code given in Example 47 as shown in Figure 3.45.

For unordered list, the default style type is **disc**. The value is **circle, square, disk** and **none.**

Example 48: list style type for unordered list:

```
<!DOCTYPE html>
<html>
<head>
<title>List Style type for ordered list</title>
<style>
```

(Contd...)

Scanned with OKEN Scanner

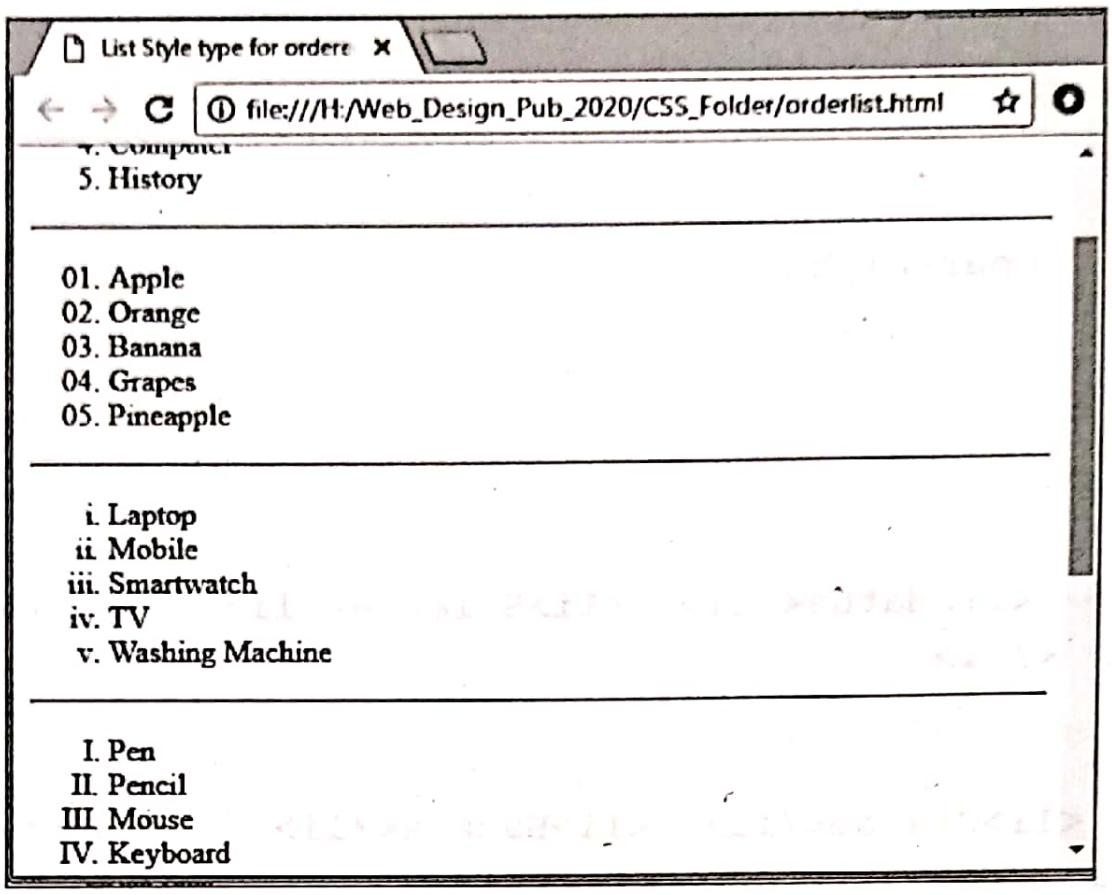


Figure 3.45: Output of the above Example 46

```
#ul-disc {
 list-style-type: disc;
 }
 #ul-square {
 list-style-type: square;
  }
 #ul-circle {
 list-style-type: circle;
 }
</style>
</head>
<body>
<ul id="ul-disc">
 <li>English</li>  <li>Maths</li>  <li>Science</li>  <li>Computer</li>
 <li>History</li> </ul>
<hr>
<ul id="ul-square">
 <li>Apple</li>  <li>Orange</li>  <li>Banana</li>  <li>Grapes</li>
 <li>Pineapple</li> </ul>
<hr>
<ul id="ul-circle">
 <li>Laptop</li>  <li>Mobile</li>  <li>Smartwatch</li>  <li>TV</li>
 <li>Washing Machine</li> </ul> <hr>
</body>
</html>
```

The output of the code given in Example 48 as shown in Figure 3.46.

Scanned with OKEN Scanner

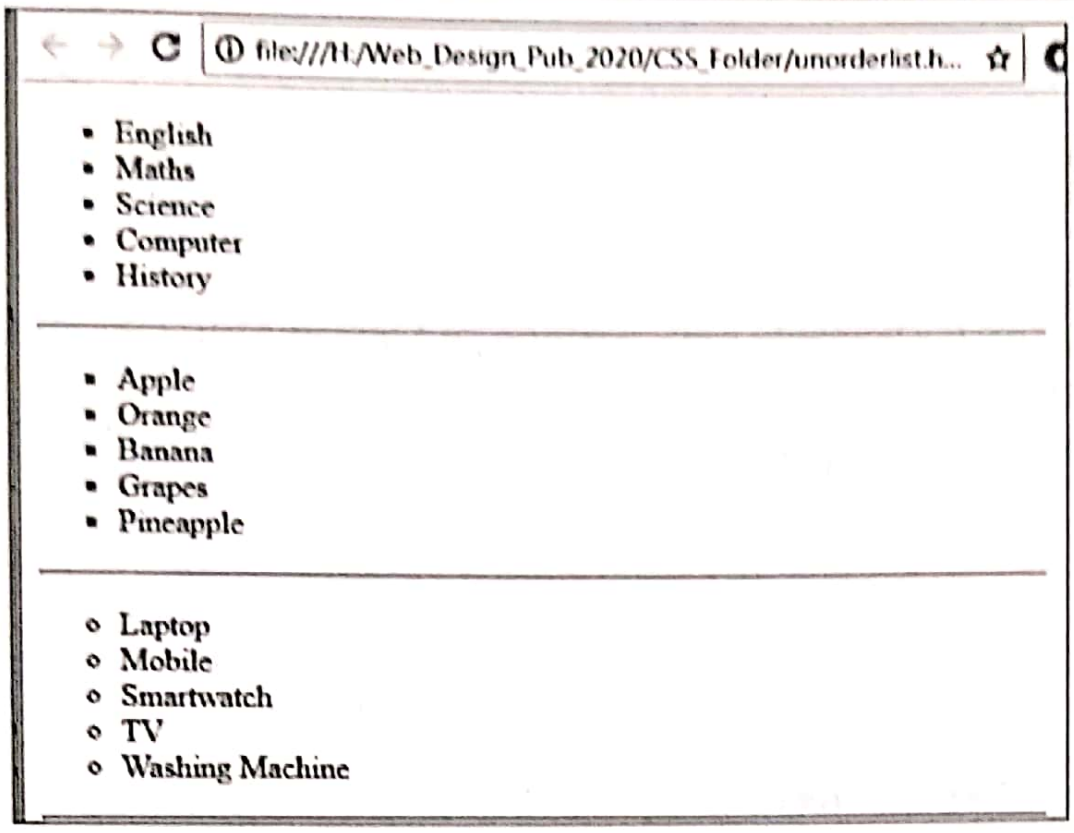


Figure 3.46: Output of the above Example 48

List Style Position: The line is position outside of the list item boxes, by default. However, you can use the list-style-position property to specify whether the bullet points appear inside or outside of the list item block boxes.

Example 49: list style position

```
<html>
<head>
<title>List-style-Position</title>
<body>
 <ul    style="list-style-type:circle;
list-style-position:outside;">
 <li>Maths</li>          <li>Social</li>
<li>English</li> </ul>
<ul     style="list-style-type:Square;
list-style-position:inside;">
 <li>Maths</li>          <li>Social</li>
<li>English</li></ul>
<ol    style="list-style-type:decimal;
list-style-position:outside;">
 <li>Maths</li>          <li>Social</li>
<li>English</li>
</ol>
<ol      style="list-style-type:lower-
alpha; list-style-position:inside;">
 <li>Maths</li>          <li>Social</li>
<li>English</li> </ol>
</body>
</html>
```

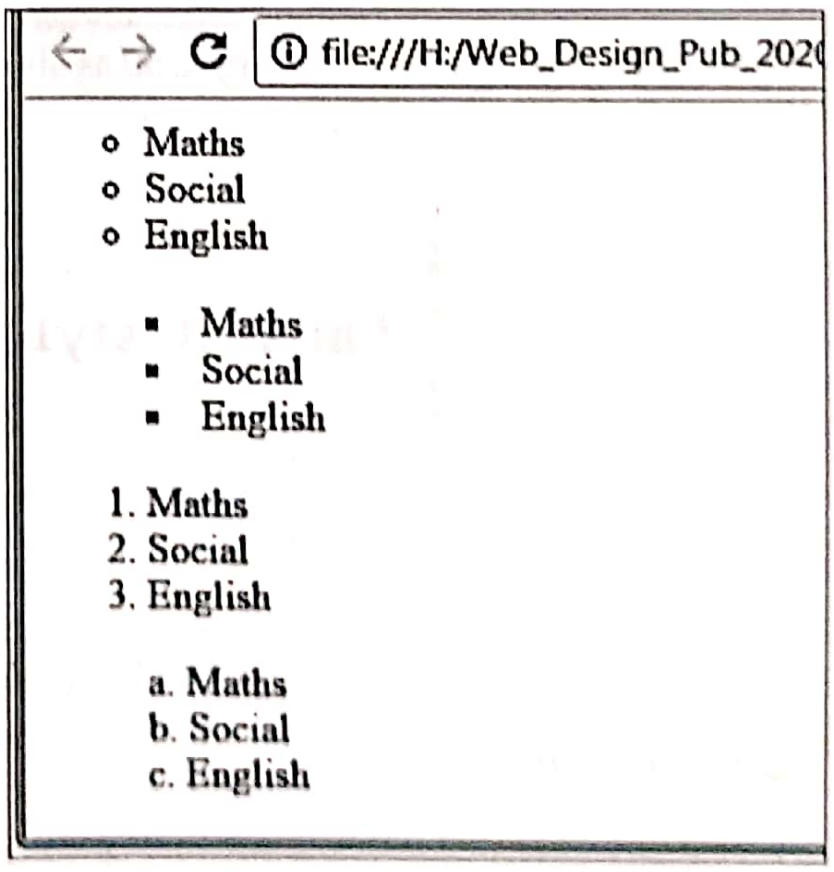


Figure 3.47: Output of the above Example 49

The output of the code given in Example 49 as shown in Figure 3.47.

Scanned with OKEN Scanner

Style List Image

You can set a background image(icons) in place of the item marker using the list-style-image property. When using an image as bullet using the list-style-image property, the bullets does not line up to the text accurately across the browser.

Shorthand list property

The list-style is the shorthand for list-style-type, position and image. Using the shorthand property, you can combine the individual properties.

Example 50: of the shorthand list property

```
<!DOCTYPE>
<html>
<head>
<style>
ol {
 list-style: upper-latin inside;
}
</style>
</head>
<body>
<h2>The <code>list-style</code> Shorthand property</h2>
<ol>
<li>List 1</li> <li>List 2</li> <li>List 3</li> </ol>
</body>
</html>
```

The output of the code given in Example 50 as shown in Figure 3.48.

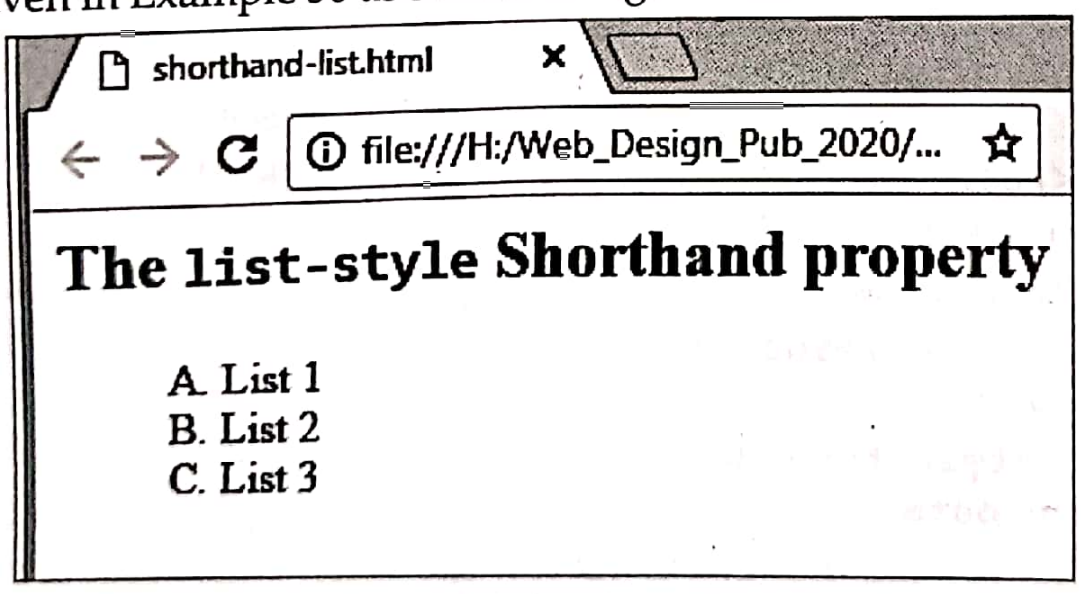


Figure 3.48: Output of the above Example 50

3.9 CSS Tables

When you build an HTML table without any styles or attributes, browser display them without any border. Styling a table with CSS can greatly improve its appearance.

```
Syntax of column width: td {width: value (width values); }
Column height: td {height: value (height values); }
Row height: tr{ line-height: value(height values); }
```

Example for column width and height and Row height

Scanned with OKEN Scanner

Example 51:

```
<!DOCTYPE>
<html>
<head>
<style>
table, tr, td
 {
border: 2px solid blue;
}
td {
 width:150px;
}
tr {
 line-height: 45px;
}
</style>
</head>
<body>
<h3>CSS Column width and Row height</h3>
<table>
<tr>
<th>FirstName</th> <th>LastName</th> <th>Mobile No</th> </tr>
<tr>
<td>Ashok</td> <td>Singal</td> <td>98234928389</td> </tr>
<tr>
<td>John</td> <td>Carter</td> <td>873938439</td> </tr>
<tr>
<td>Ritika</td> <td>Wason</td> <td>8928394389</td> </tr>
</table>
</body>
</html>
```

The output of the code given in Example 51 as shown in Figure 3.49.

CSS Column width and Row height

FirstName	LastName	Mobile No
Ashok	Singal	98234928389
John	Carter	873938439
Ritika	Wason	8928394389

Figure 3.49: Output of the above Example 51

Scanned with OKEN Scanner

3.9.1 Table borders

To specify table borders in CSS, you can set the table border, head border, table description borders etc.

Example 52: Table Borders

```
<!DOCTYPE>
<html>
<head>
<style>
 table, th, td {
  border:3px solid black;
 }
</style>
</head>
<body>
<h3>CSS Table borders</h3>
<table>
 <tr>
<th>FirstName</th> <th>LastName</th> <th>Mobile No</th> </tr>
<tr>
<td>Ashok</td> <td>Singal</td> <td>8739384393</td> </tr>
<tr>
<td>John</td> <td>Carter</td> <td>8739384394</td> </tr>
<tr>
<td>Harry</td> <td>Potter</td> <td>8739384399</td> </tr>
<tr>
<td>Ritika</td> <td>Wason</td> <td>8928394389</td> </tr>
</table>
</body>
</html>
```

The output of the code given in Example 52 as shown in Figure 3.48.

CSS Table borders

FirstName	LastName	Mobile No
Ashok	Singal	8739384393
John	Carter	8739384394
Harry	Potter	8739384399
Ritika	Wason	8928394389

Figure 3.50: Output of the above Example 52

Scanned with OKEN Scanner

3.9.2 Table Border Radius

If you add a border-radius property, then you can make the corners of the Table borders as rounded.

For example, of table border radius property is:

```
<style>
  #table-border-radius, th, td {
  Border: 6px solid black;
  Border-radius: 10px;
 Padding: 6px;
}
</style>
```

You can add and change the style and the result in the browser Window.

3.9.3 Colors in Table

The color or background-color property changes the color of the table. The heading of the table, row or columns can have different colors.

Example 53: of colors in Table

```
<!DOCTYPE>
<html>
<head>
<style>
 table, th, td {
  border:3px solid aqua;
 }
tr {
 color:palegreen;
 background: green;
 }
th {
 color: fuchsia;
 background: plum;
 }
</style>
</head>
<body>
<h3>CSS Table borders</h3>
<table>
 <tr>
<th>FirstName</th> <th>LastName</th> <th>Mobile No</th> </tr>
<tr>
<td>Ashok</td> <td>Singal</td> <td>8739384393</td> </tr>
<tr>
```

(Contd...)

Scanned with OKEN Scanner

```
<td>Ashok</td> <td>Singal</td> <td>8739384393</td> </tr>
<tr>
<td>John</td> <td>Carter</td> <td>8739384394</td> </tr>
<tr>
<td>Harry</td> <td>Potter</td> <td>8739384399</td> </tr>
<tr>
<td>Ritika</td> <td>Wason</td> <td>8928394389</td> </tr>
</table>
</body>
</html>
```

Run the above code in the browser.

3.9.4 Collapse table border

Every table cell has separate borders as well as there is some space between the borders of adjacent table cells. It happens because table cell borders are separate by default. But you, can collapse the separate table borders into one by using the border-collapse property on the <table> element.

Example 54: of the collapse table border

```
<!DOCTYPE>
<html>
<head>
<style>
 table, th, td {
  border:3px solid aqua;
  border-collapse:collapse;
  text-align:left;
  width:70%;
  }
</style>
</head>
<body>
<h3>CSS Table with border-collapse</h3>
<table>
<tr>
<th>FirstName</th> <th>LastName</th> <th>Mobile No</th> </tr>
<tr>
<td>Ashok</td> <td>Singal</td> <td>8739384393</td> </tr>
<tr>
<td>John</td> <td>Carter</td> <td>8739384394</td> </tr>
<tr>
<td>Harry</td> <td>Potter</td> <td>8739384399</td> </tr>
```

(Contd...)

Scanned with OKEN Scanner

```
<tr>
<td>Ritika</td>
<td>Wason</td>
<td>8928394389</td>
</tr>
</table>
</body>
</html>
```

The output of the code given in Example 54 as shown in Figure 3.51.

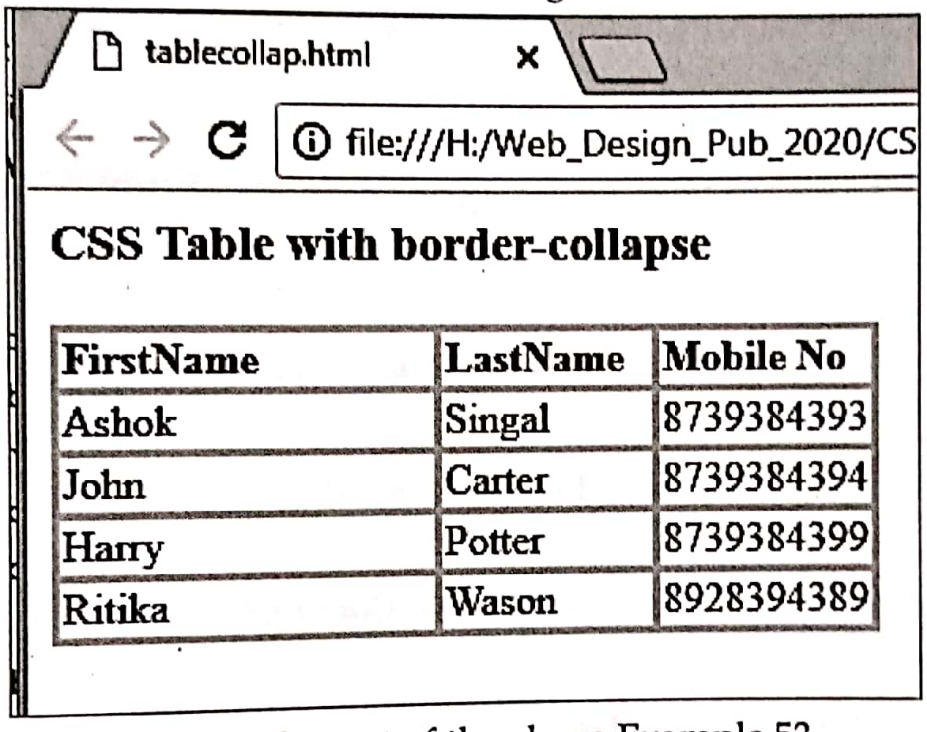


Figure 3.51: Output of the above Example 53

Colors specific rows in the table

Syntax:

```
tr:nth-child(m)
 {
 Background-color: blue;
Color:white;
}
```

This applies the color and background color property to n^{th} row from top.

Example 55: Colors specific rows in the table

```
<!DOCTYPE>
<html>
<head>
<style>
 table, tr, td {
  border:3px solid aqua;
  border-collapse:collapse;
  width:80%;
  }
```

(Contd...)

Scanned with OKEN Scanner

```
.th {
 color: fuchsia;
 background: plum;
 }
tr:nth-child(5)
 {
 background-color: green;
 color:white;
}
</style>
</head>
<body>
<h3>CSS Table particulare color to a specific row</h3>
<table>
 <tr>
<th>FirstName</th> <th>LastName</th> <th>Mobile No</th> </tr>
<tr>
<td>Ashok</td> <td>Singal</td> <td>8739384393</td> </tr>
<tr>
<td>John</td> <td>Carter</td> <td>8739384394</td> </tr>
<tr>
<td>Harry</td> <td>Potter</td> <td>8739384399</td> </tr>
<tr>
<td>Ritika</td> <td>Wason</td> <td>8928394389</td> </tr>
</table>
</body>
</html>
```

The output of the code given in Example 55 as shown in Figure 3.52.

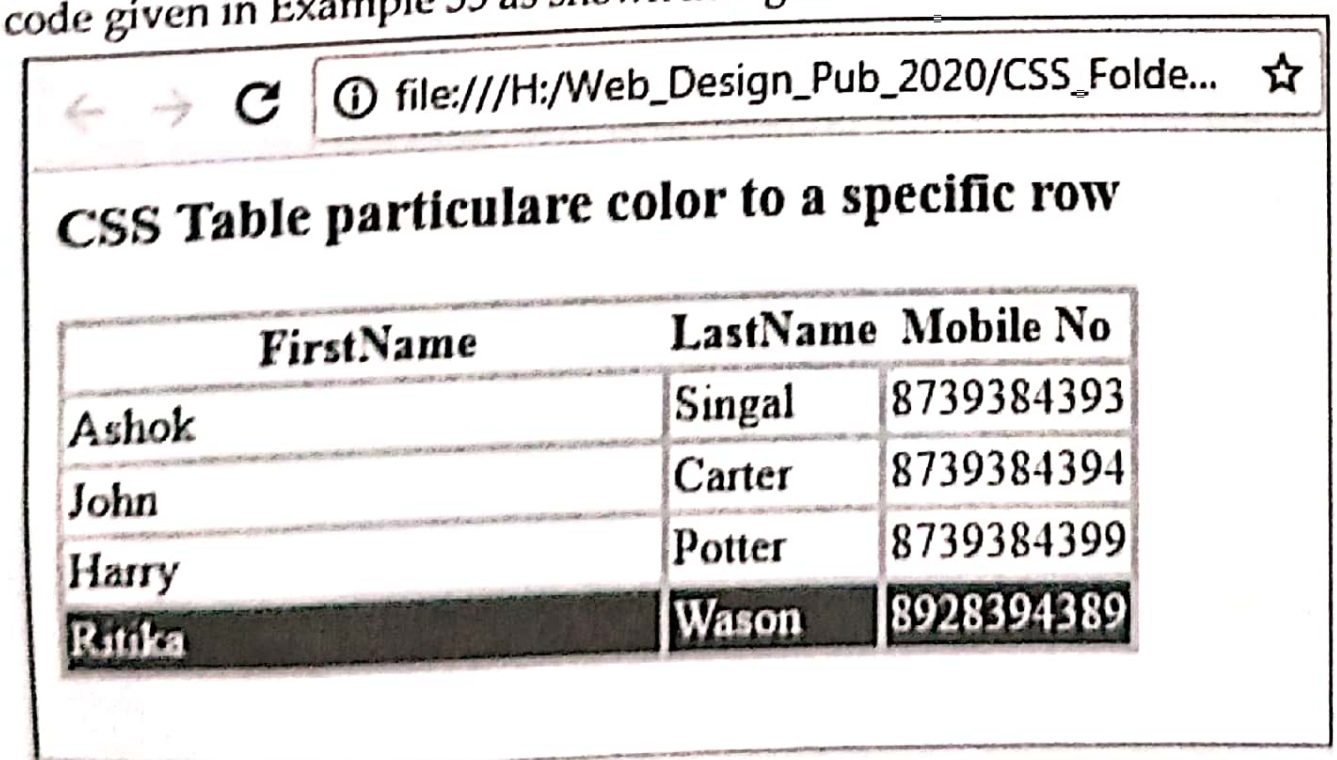


Figure 3.52: Output of the above Example 55

Scanned with OKEN Scanner

In the same way, you can add a particular color to a specific column.

Syntax to add color specific column in table

```
td:nth-child(k)
 {
 background-color: orange;
 color:white;
}
```

This specifies the color & background color property to k^{th} row from top.

3.9.5 Table Stripped

The table stripped have odd rows such as (1,3,5,7) and so on have a particular color & the even rows (2,4,6) and so on have a particular color. This is way of creating apply the same color to alternate rows.

Syntax of odd table strip

```
tr:nth-child(odd) {
 background-color: gold;
 color:white;
}
```

It will set all the odd rows(including the head row) with background as gold color and font color as white.

Syntax of even table strip

```
tr:nth-child(even) {
background-color: blue;
color: green;
}
```

It will set al the even rows (2,4,6) with background as blue color and font color as green.

Example 56: **Add strip on Table**

```
<!DOCTYPE>
<html>
<head>
<style>
 table, th, td {
  border:3px solid green;
  }
  table
  {
    border-collapse:collapse;
    width:80%;
    text-align:left;
   }
tr:nth-child(even)
 {
 background-color:gold;
 }
</style>
```

(Contd...)

Scanned with OKEN Scanner

```
</head>
<body>
<h3>CSS Table Striped</h3>
<table>
<tr>
<th>FirstName</th> <th>LastName</th> <th>Mobile No</th> </tr>
<tr>
<td>Ashok</td> <td>Singal</td> <td>8739384393</td> </tr>
<tr>
<td>John</td> <td>Carter</td> <td>8739384394</td> </tr>
<tr>
<td>Harry</td> <td>Potter</td> <td>8739384399</td> </tr>
<tr>
<td>Ritika</td> <td>Wason</td> <td>8928394389</td> </tr>
</table>
</body>
</html>
```

The output of the code given in Example 56 as shown in Figure 3.53.

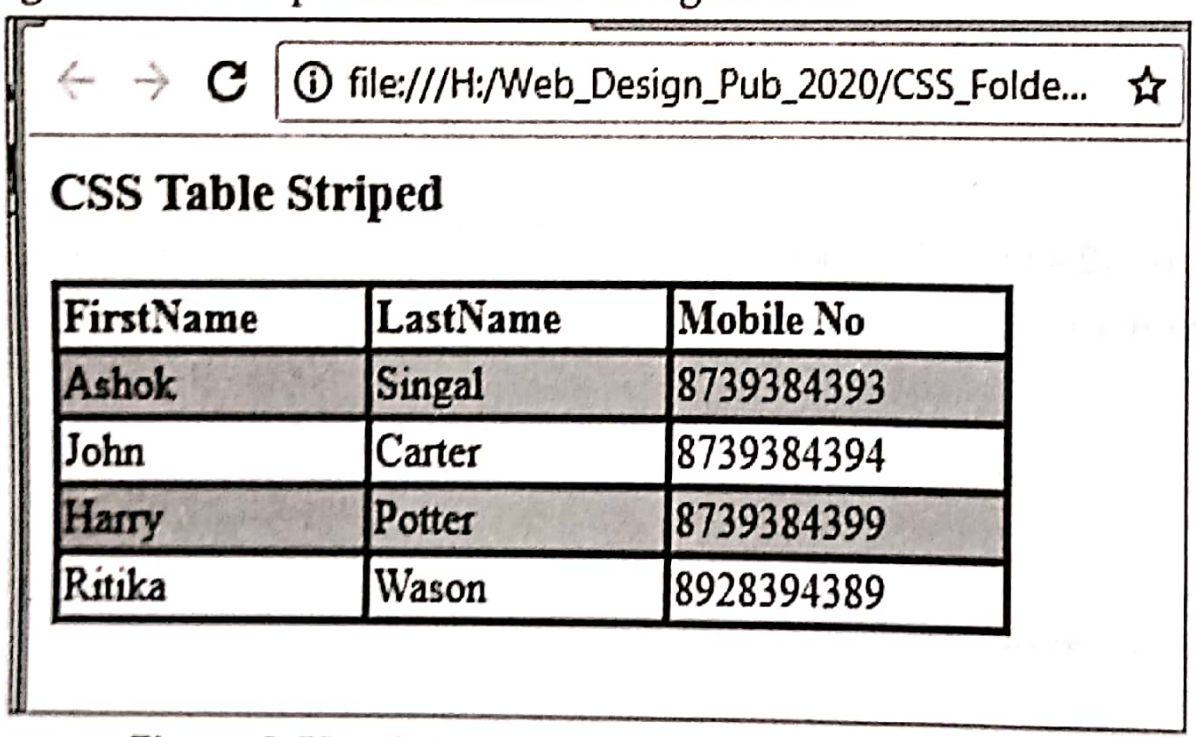


Figure 3.53: Output of the above Example 56

3.9.6 Padding in Table

We cans specify padding for table header and table data using the CSS padding property.

Example 57: Padding in Table

```
<!DOCTYPE>
<html>
<head>
<style>
 table, th, td {
```

(Contd...)

Scanned with OKEN Scanner

```
  border:5px solid black;
  border-collapse:collapse;
  }
th, td {
  padding:15px;
 }
</style>
</head>
<body>
<h3>CSS Table Padding</h3>
<table>
 <tr>
<th>FirstName</th> <th>LastName</th> <th>Marks</th> </tr>
<td>Ashok</td> <td>Singal</td> <td>93</td> </tr>
<tr>
<td>John</td> <td>William</td> <td>87</td> </tr>
<tr>
<td>Harish</td> <td>Jamesh</td> <td>43</td> </tr>
<tr>
<td>Ritika</td> <td>Wason</td> <td>83</td> </tr>
</table>
</body>
</html>
```

The output of the code given in Example 57 as shown in Figure 3.54.

CSS Table Padding

FirstName	LastName	Marks
Ashok	Singal	93
John	William	87
Harish	Jamesh	43
Ritika	Wason	83

Figure 3.54: Output of the above Example 57

3.9.7 Border Image in Table

The border-image property sets image in the form of borders.

Scanned with OKEN Scanner

For example,

```
Border-image: url("H:/gradient.jpg") 30% stretch;
```

You can also use other border-image properties like a repeat, round etc., in place of stretch.

Background Images in Table

To set a background image for the table, we can use background-image property.

Syntax:

```
Background-image: url(image url file name.jpg);
```

Table hover

Use the pseudo class selector :hover selector on <tr> to highlights table rows, table head or a particular cell.

Example 58: Table Hover

```
<!DOCTYPE>
<html>
<head>
<style>
 table, th, td {
  border: 20px solid black;
  border-collapse: collapse;
 }
tr:hover {
background-color: #dfd9d9;
color:red;
}
</style>
</head>
<body>
<h3>CSS Table Padding</h3>
<table>
 <tr>
<th>FirstName</th> <th>LastName</th> <th>Marks</th> </tr>
<td>Ashok</td> <td>Singal</td> <td>93</td> </tr>
<tr>
<td>John</td> <td>William</td> <td>87</td> </tr>
<tr>
<td>Harish</td> <td>Jamesh</td> <td>43</td> </tr>
<tr>
<td>Ritika</td> <td>Wason</td> <td>83</td> </tr>
</table>
</body>
</html>
```

The output of the code given in Example 58 as shown in Figure 3.55.

Scanned with OKEN Scanner

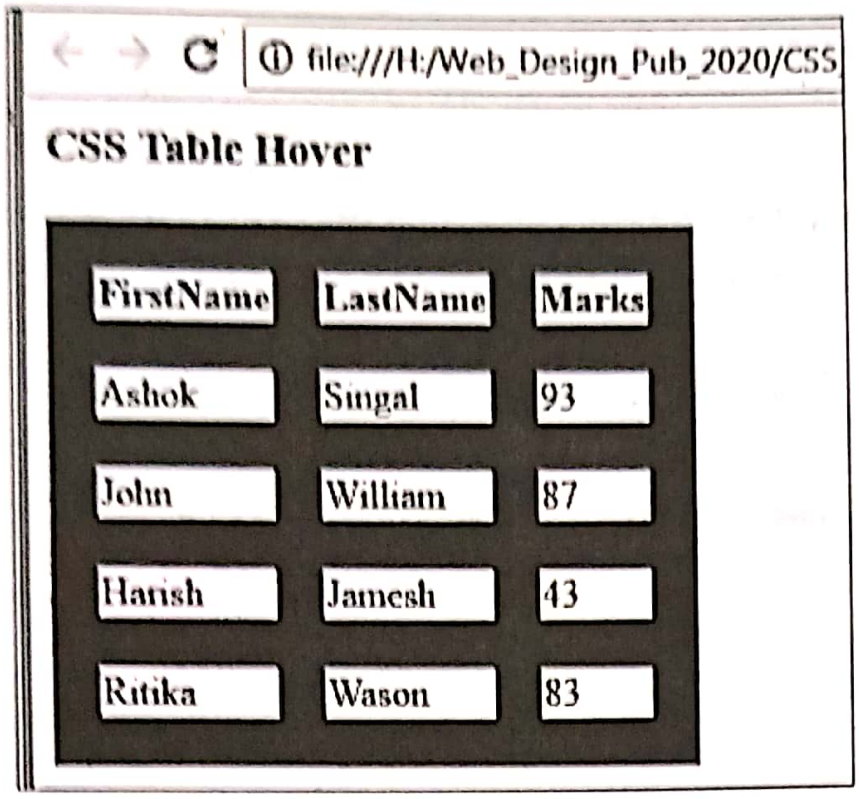


Figure 3.55: Output of the above Example 58

Horizontal and vertical Alignment

To align the content in the table, you can use CSS align property.

Syntax to align:

- Text-align: left; /*Align to the left of the cell */
- Text-align: right; /*Align to the Right of the cell */
- Text-align: center; /*Align to the center of the cell */
- Text-align: justify; /* To justify the content */

Syntax of vertical alignment

```
Vertical-align: bottom /*Align to the bottom of the cell */
```

Example 59: Horizontal and vertical alignment in table

```
<!DOCTYPE>
<html>
<head>
<style>
 table, th, td {
  border: 10px solid black;
 }
table {
border-collapse: collapse;
width:100%
}
td {
height: 50px; vertical-align:bottom;
</style>
</head>
<body>
```

(Contd)

Scanned with OKEN Scanner

```
<h3>Vertical-align property</h3>
<table>
 <tr>
<th>FirstName</th> <th>LastName</th> <th>Marks</th> </tr>
<td>Ashok</td> <td>Singal</td> <td>93</td> </tr>
<tr>
<td>John</td> <td>William</td> <td>87</td> </tr>
<tr>
<td>Harish</td> <td>Jamesh</td> <td>43</td> </tr>
<tr>
<td>Ritika</td> <td>Wason</td> <td>83</td> </tr>
</table>
</body>
</html>
```

The output of the code given in Example 59 as shown in Figure 3.56.

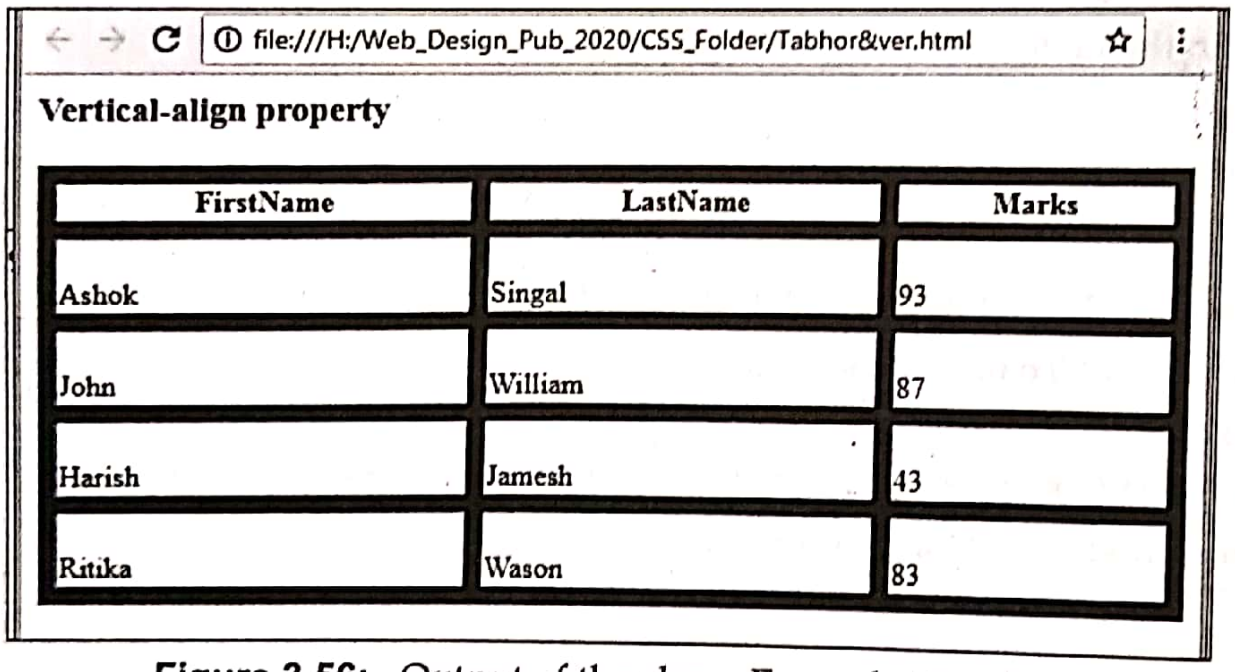


Figure 3.56: Output of the above Example 59

Controlling the Table Layout

By default, a table expands and indentures to accommodate the data contained inside it. As data fills inside the table, it continues to expand as long as there is space. However, it is necessary to set a fixed width for the table in order to manage the layout.

The CSS table-layout defines the algorithm to be used to layout the table cells, rows and columns. This property takes one of **two** values:

Auto: This algorithm, the width of the table and it cells are adjusted to fit the content. This is default.

Fixed: This algorithm, the horizontal layout of the table does not depend on the contents of the cells, it depends on the table's width, the width of the columns, and borders or cell spacing.

Example 60:

```
<!DOCTYPE>
<html>
```

(Contd...)

Scanned with OKEN Scanner

```
<head>
<style>
 table {
  width: 250px;
  border-collapse: separate;
}
table, tr, th, td {
 border: 1px solid black;
}
.auto {
  table-layout: auto;
 }
.fixed {
 table-layout: fixed;
 }
td {
 width: 50%
}
</style>
</head>
<body>
<h3>Different Table Layout</h3>
<table class="auto">
<caption>Auto Layout</caption>
<tr>
<th>Name</th> <td>John Peter</th> </tr>
<tr>
<th>E-mail Address</th> <td>johpet123@yahoo.co.in </tr>  </table>
<br>
<table class="fixed">
<caption>Fixed</caption>
<tr>
<th>Name</th> <td>John Peter</th> </tr>
<tr>
<th>E-mail Address</th> <td>johpet123@yahoo.co.in </tr>
</body>
</html>
```

The output of the code given in Example 60 as shown in Figure 3.55.

Scanned with OKEN Scanner

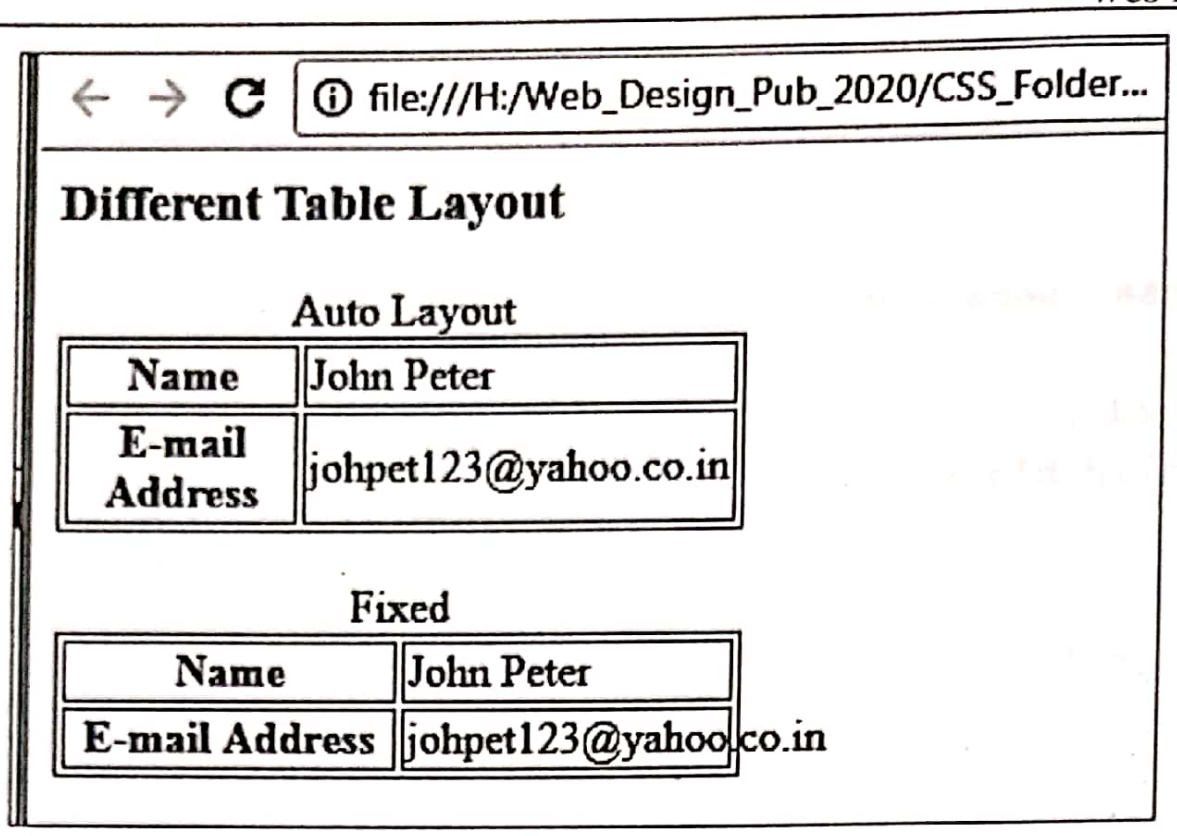


Figure 3.57: Output of the above Example 60

Empty Cells

The empty-cells CSS property controls the borders and background of cells that have no visible content in a table using separate borders model. The empty cells property contains three values: **show, hide** or **inherit.**

Example 61: handling empty cells

```
<!DOCTYPE>
<html>
<head>
<style>
 table {
  width: 350px;
  border-collapse: separate;
}
table, th, td {
 border: 1px solid black;
}
table.empty-show {
 empty-cells: show;
}
table.empty-hide {
 empty-cells: hide;
}
</style>
</head>
<body>
<h3>Table with Empty Cells</h3>
```

(Contd...)

Scanned with OKEN Scanner

```
<table class="empty-show">
<tr>
<th>Name</th>
<td>John Peter</th>
</tr>
<tr>
<th>E-mail Address</th>
<td></td>
</tr>
</table>
<br>
<h3>Table with Hidden Empty cells</h3>
<table class="empty-hide">
<tr>
<th>Name</th>
<td>John Peter</th>
</tr>
<tr>
<th>E-mail Address</th>
<td></td>
</tr>
</body>
</html>
```

The output of the code given in Example 61 as shown in Figure 3.56.

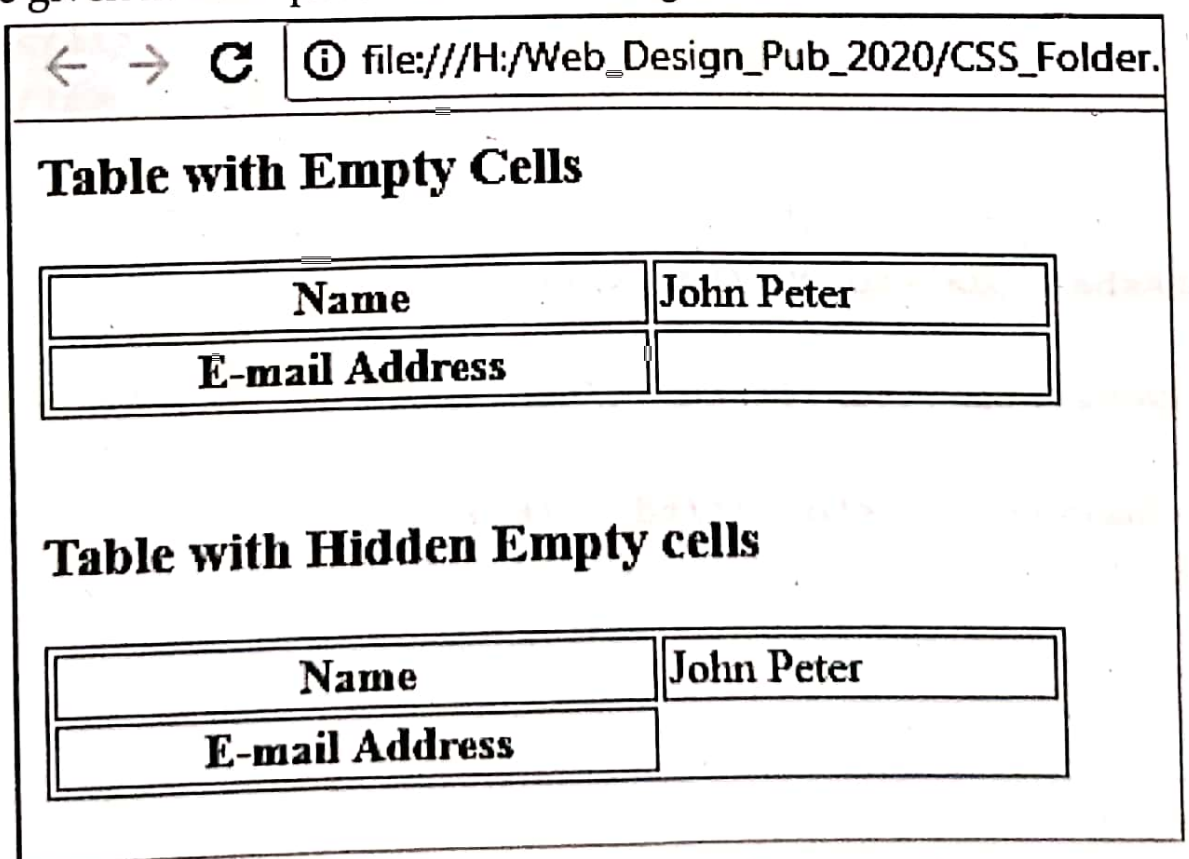


Figure 3.58: Output of the above Example 61

Scanned with OKEN Scanner

Captions for Table

The caption property sets the caption for the table. It is the name of the table.

Syntax to set caption

Caption-side: Bottom: To set the caption at the bottom of the table.

Caption-side: Top: To set the caption at the top of the table. This is default.

Caption-side: Inherit:

Example 62:

```
<!DOCTYPE>
<html>
<head>
<style>
table, td, th {
 border: 2px solid gray;
}
caption {
 caption-side: bottom;
}
</style>
</head>
<body>
<table>
<caption><Strong>Table 1:</strong>Students Marks List</caption>
<thead>
<tr>
<th>No</th> <th>Name</th> <th>Marks</th> </tr>
</thead>
<tbody>
<tr>
<td>1</td> <td>Asha</td> <td>90</td> </tr>
<tr>
<td>2</td> <td>Neha</td> <td>74</td> </tr>
<tr>
<td>3</td> <td>Sharvi</td> <td>48</td> </tr>
</tbody>
</table>
</body>
</html>
```

The output of the code given in Example 62 as shown in Figure 3.59.

Scanned with OKEN Scanner

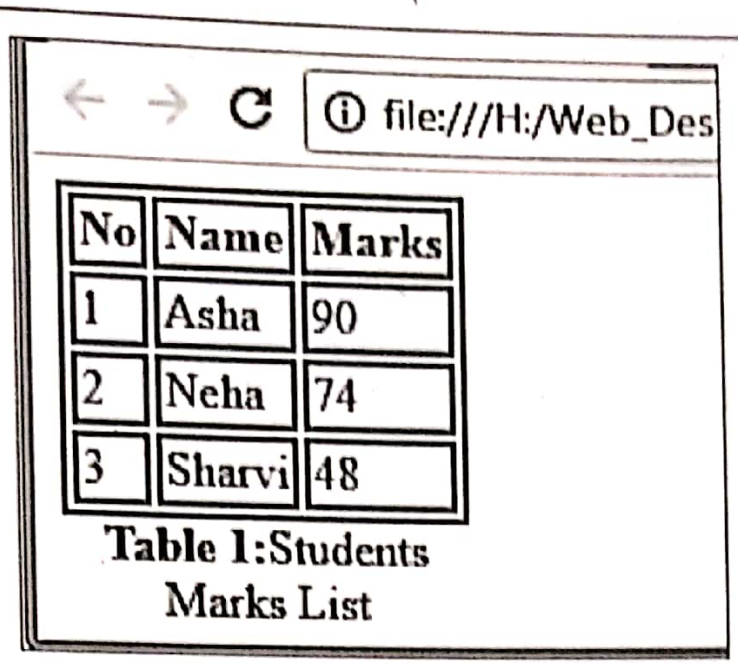


Figure 3.59: Output of the above Example 62

3.10 CSS Menu Design

With menu, designing in CSS it is one of the most important parts of website is navigation menu, which is very decorative and interactive section of page.

Navbar in CSS refers to a group of links that lead to different pages of a website.

Navbars are either **vertical** or **horizontal.**

The <nav> element should wrap bars such as the main navigation links of websites.

The navigation bar is a list of links, so using the <ul> and <li> elements makes perfect sense:

To select creating Navigation bars

Step1: Create an unordered list like below:

```
<ul>
<li>News</li>
<li>Markets</li>
<li>Mutual Funds</li>
<li>Commodities</li>
</ul>
```

Step2: Add links to the list items using <a> tag so that the list items becomes clickable.

```
<ul id="#ul-nb">
<li><a href="#">News</a></li>
<li><a href="#">Markets</a></li>
<li><a href="#">Mutual Funds</li>
<li><a href="#">Commodities</a></li>
```

Step 3: Let us remove the bullets and the margins and padding from the list:

```
#ul-nb {
List-style-type: none;
Margin: 0;
Padding: 0;
}
```

Step 4: You can set other style related properties on the list items such as:

```
Float properties like float: Left
```

Scanned with OKEN Scanner

Background color of the link list

Margin, padding properties: You might require setting margin and padding values as 0 if required.

Width & height related properties.

Text-decoration: None is often used to remove the underline from the link.

Text-align property and so on.

Example 63: creating Navigation bars

```
<!DOCTYPE>
<html>
<head>
<style>
#ul-nb {
list-style:none;
margin: 0;
padding: 0;
}
#ul-nb li {
 float:left;
 padding:10px;
 background:orange;
 text-align:center;
 margin-left: 5px;
}
#ul-nb li:hover {
 background:red;
 opacity:0.8;
 color:white;
}
</style>
</head>
<body>
<ul id="ul-nb">
<li><a href="#News">News</a></li>
<li><a href="#Markets">Markets</a></li>
<li><a href="#Mutual Funds">Mutual Funds</a></li>
<li><a href="#Commodities">Commodities</a></li>
</ul>
<br><br>
<p>A background color is added to the links to show the link area</p>
</body>
</html>
```

The output of the code given in Example 63 as shown in Figure 3.58.

Scanned with OKEN Scanner

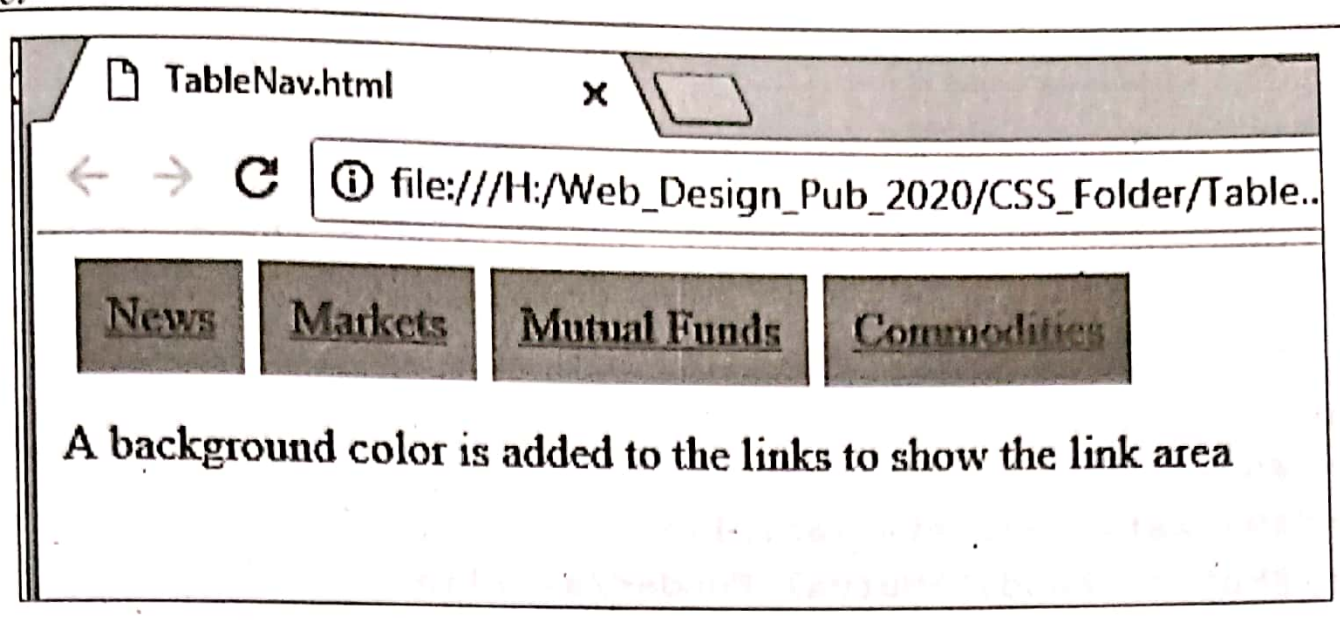


Figure 3.60: Output of the above Example 63

Horizontal Navigation bar

The horizontal bar which is horizontal direction along x-axis. There are three ways to create horizontal navigation bars.

1. To make list items as float without using the display as block.
2. To make list items as float with display as block.
3. Using list items as inline.

Example 64: Horizontal Navigation using Float & display as block

```
<!DOCTYPE>
<html>
<head>
<style>
#ul-nb {
list-style:none;
overflow:hidden;
background:orange;
margin: 0px;
padding: 0px;
}
#ul-nb li a {
 text-align:center;
 padding:10px;
 width: 100px;
}
#ul-nb li {
 float:left;
}
#ul-nb li:hover {
background: red;
 }
```

(Contd..)

Scanned with OKEN Scanner

```
a {
 display:block;
 }
</style>
</head>
<body>
<ul id="ul-nb">
<li><a href="#News">News</a></li>
<li><a href="#Markets">Markets</a></li>
<li><a href="#Mutual Funds">Mutual Funds</a></li>
<li><a href="#Commodities">Commodities</a></li>
</ul>
</body>
</html>
```

The output of the code given in Example 64 as shown in Figure 3.59.

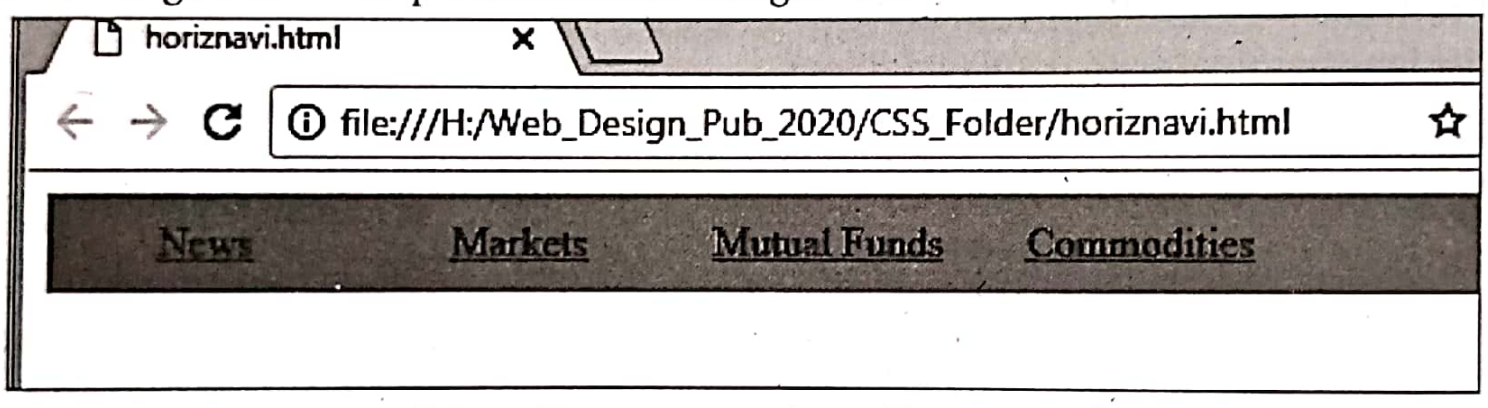


Figure 3.61: Output of the above Example 64

Horizontal Navigation using display as Inline

If you create horizontal bar using display as Inline then you will not be able to apply the width and height properties and does not any effect on width & height properties.

Horizontal Navigation bars by creating dividers

You can create dividers between menu.

Example 65:

```
<!DOCTYPE>
<html>
<head>
<style>
.navigation-menu ul {
  margin:0px;
  padding:0px;
}
#ul-nb li a {
 text-align:center;
 padding:10px;
 width: 100px;
}
```

(Contd...)

Scanned with OKEN Scanner

```
ul {
 padding:0;
 overflow:hidden;
 background-color: skyblue;
}
 li {
 float: left;
 }
li a {
 display: inline-block;
 color:black;
 text-align:center;
 padding: 10px 20px;
 text-decoration: none;
 }
li a:hover {
 background-color: slateblue;
 }
#navigation ul
{
 font-size: 1.5em;
 }
#navigation ul li
 {
display:inline;
color:white;
}
#navigation li:not(:first-child):before {
content: " | ";
}
</style>
</head>
<body>
<div class="navigation-menu">
<div id="navigation">
<ul>
<li><a href="#News">News</a></li>
<li><a href="#Markets">Markets</a></li>
<li><a href="#Mutual Funds">Mutual Funds</a></li>
<li><a href="#Commodities">Commodities</a></li> </ul>
</body>
</html>
```

The output of the code given in Example 65 as shown in Figure 3.62.

Scanned with OKEN Scanner

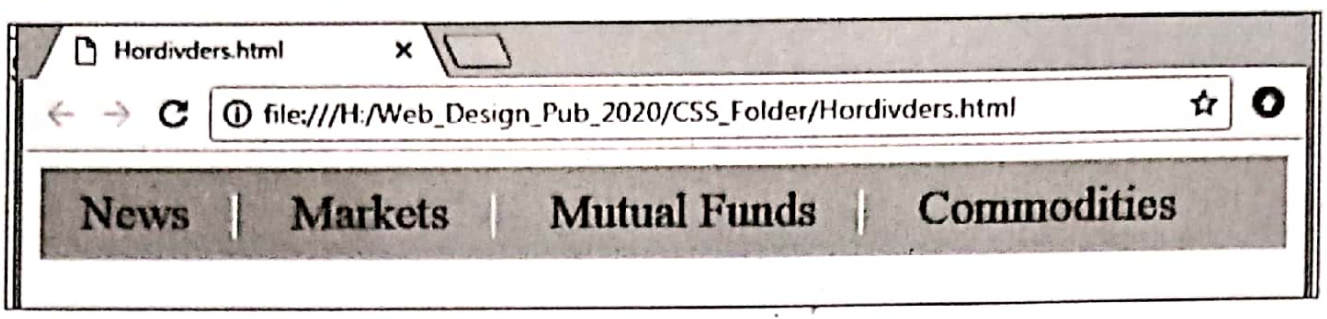


Figure 3.62: Output of the above Example 65

Vertical Navigation Bar

It create a basic vertical navigation bar with a gray background color and change the background color of the links when the user moves the mouse over them:

Example 66:

```
<!DOCTYPE>
<html>
<head>
<style>
ul {
 line-style-type:none;
 margin:0px;
 padding:0px;
 width:200px;
 background-color:yellow;
 border: 1px solid black;
 }
li a {
 display:block;
 color: red;
 padding: 8px 16px;
 text-decoration: none;
 }
li {
 text-align:center;
 border-bottom: 1px solid black;
}
li:last-child {
 border-bottom: none;
}
li.active {
 background-color: yellow;
 color:white;
}
```

(Contd...)

Scanned with OKEN Scanner

```
li a:hover:not(.active) {
 background-color: black;
 color:white;
}
</style>
</head>
<body>
<h3>Vertical Navigation bar</h3>
<ul>
<li><a class="active" href="#News">News</a></li>
<li><a href="#Markets">Markets</a></li>
<li><a href="#Mutual Funds">Mutual Funds</a></li>
<li><a href="#Commodities">Commodities</a></li>
</ul>
</body>
</html>
```

The output of the code given in Example 66 as shown in Figure 3.63.

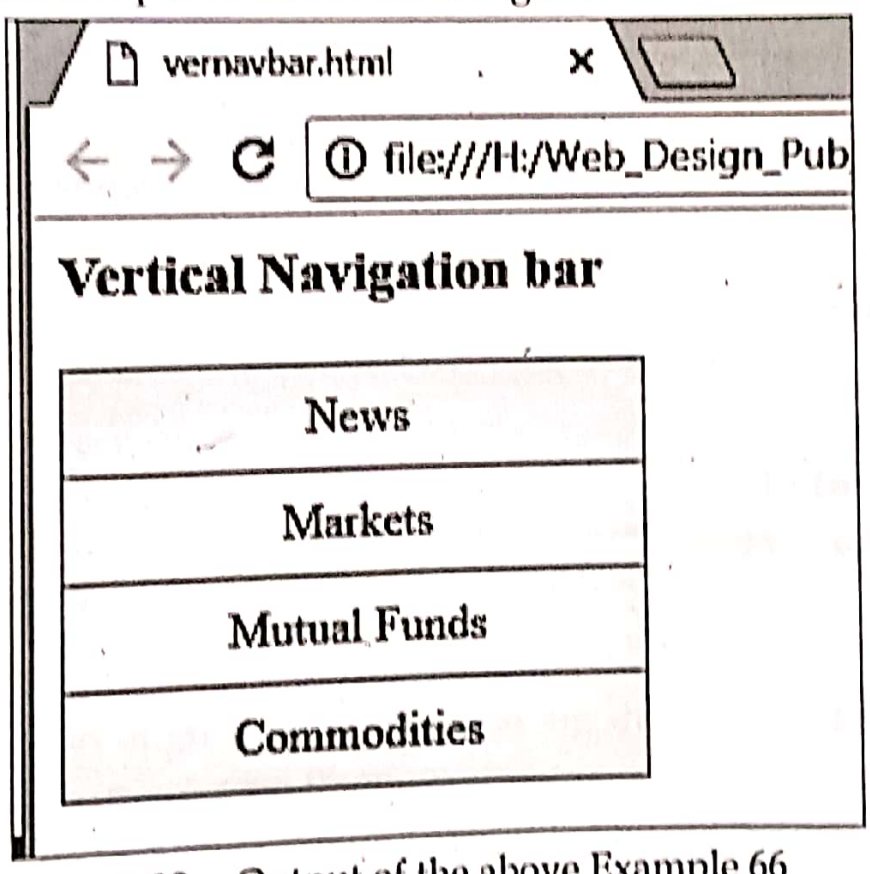


Figure 3.63: Output of the above Example 66

Fixed Navigation Bar

You can create navigation bar stay at the top, bottom or side of the page, even when the user scrolls the page.

Example 67:

```
<!DOCTYPE>
<html>
<head>
```

(Contd...)

Scanned with OKEN Scanner

```
<style>
body {margin:0;}

ul {
 list-style-type:none;
 margin: 0;
 padding: 0;
 overflow: hidden;
 background-color:purple;
 position: fixed;
 top:0;
 width: 100%;
}

li {
 float: left;
}
li a {
 display: block;
 color: white;
 text-align: center;
 padding: 14px 16px;
 text-decoration: none;
 }

li a:hover:not(.active) {
 background-color: Tea green;
}
.active {
 background-color: red;
}
</style>
</head>
<body>
<h3>Fixed Top Navigation bar</h3>
<ul>
<li><a class="active" href="#News">News</a></li>
<li><a href="#Markets">Markets</a></li>
<li><a href="#Mutual Funds">Mutual Funds</a></li>
```

(Contd...)

Scanned with OKEN Scanner

```
<li><a href="#Commodities">Commodities</a></li>
</ul>
<div                                style="padding:20px;margin-top:20px;background-
color:yellow;height:1500px;">
<p>Text goes here.....</p>
<p>Text goes here.....</p>
<p>Text goes here.....</p>
<p>Text goes here.....</p>
<p>Text goes here.....</p>
<p>Text goes here.....</p>
</div>
</body>
</html>
```

The output of the code given in Example 67 as shown in Figure 3.61.

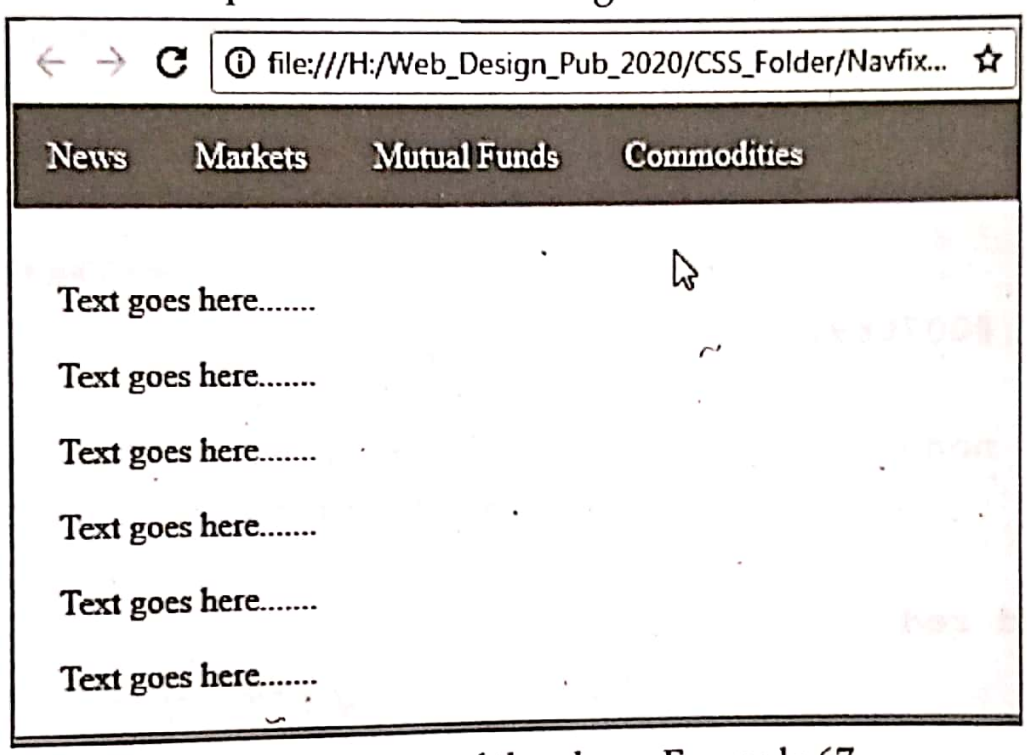


Figure 3.64: Output of the above Example 67

Drop down Navigation Menu

The justify: content: space-between align the flex items along the main axis in such a way that each two successive items have equal spaces in between them.

Note: Hide the Menu Items at start by hiding the unordered list inside the main menus by using-

```
li ul {
 display: none;
}
```

Display the items inside each menu vertically and the position of each element must be relative to other elements like this:

```
li:hover ul, {
 display: block;
 position: absolute;
 }
```

Do not float the list to left or right. Add the float: none.

Scanned with OKEN Scanner

Example 68: Drop Down Navigation Menu

```
<!DOCTYPE>
<html>
<head>
<style>
ul {
 list-style-type:none;
 padding:2px;
 }
ul li {
 display: block;
 float: left;
 }
li ul {
 display: none;
 }

ul li a {
 display:block;
 background-color:#007689;
 padding: 10px;
 text-decoration: none;
 color:white;
 width:100px;
 border: 2px solid red;
 }
 ul li a:hover {
  background-color: #007689;
  color:white;
  }
li:hover ul {
 display: block;
 position:absolute;
}
li:hover li {
 float:none;
 }
li:hover a{
 background-color:#94C548;
```

(Contd...)

Scanned with OKEN Scanner

```
color:black;
 }
li:hover li a:hover {
 background: black;
 width:100px;
}
</style>
</head>
<body>
<h3>Drop Down Navigation Menu</h3>
<ul>
<li>
 <a href="#">News</a>
<ul>
 <li><a href="#">Latest News</a></li>  <li><a href="#">Top News</a></li>
 <li><a href="#">Hindi News</a></li> </ul>
</li>
<li>
 <a href="#">Markets</a>
<ul>
 <li><a href="#">Pre-markets</a></li>
 <li><a href="#">Global Markets</a></li>
 <li><a href="#">Stock Screener</a></li> </ul>
</li>
<li>
 <a href="#">Mutual Funds</a>
<ul>
 <li><a href="#">Axis Bluechip Fund</a></li>
 <li><a href="#">LIC MF Large Cap</a></li>
 <li><a href="#"> CR Bluechip Equity</a></li> </ul>
</li>
<li>
 <a href="#">Commodities</a>
<ul>
 <li><a href="#">Gold Price</a></li>  <li><a href="#">Silver Price</a></li>
 <li><a href="#">Crudeoil Price</a></li> </ul>
</li>
</ul>
</body>
</html>
```

The output of the code given in Example 68 as shown in Figure 3.65.

Scanned with OKEN Scanner

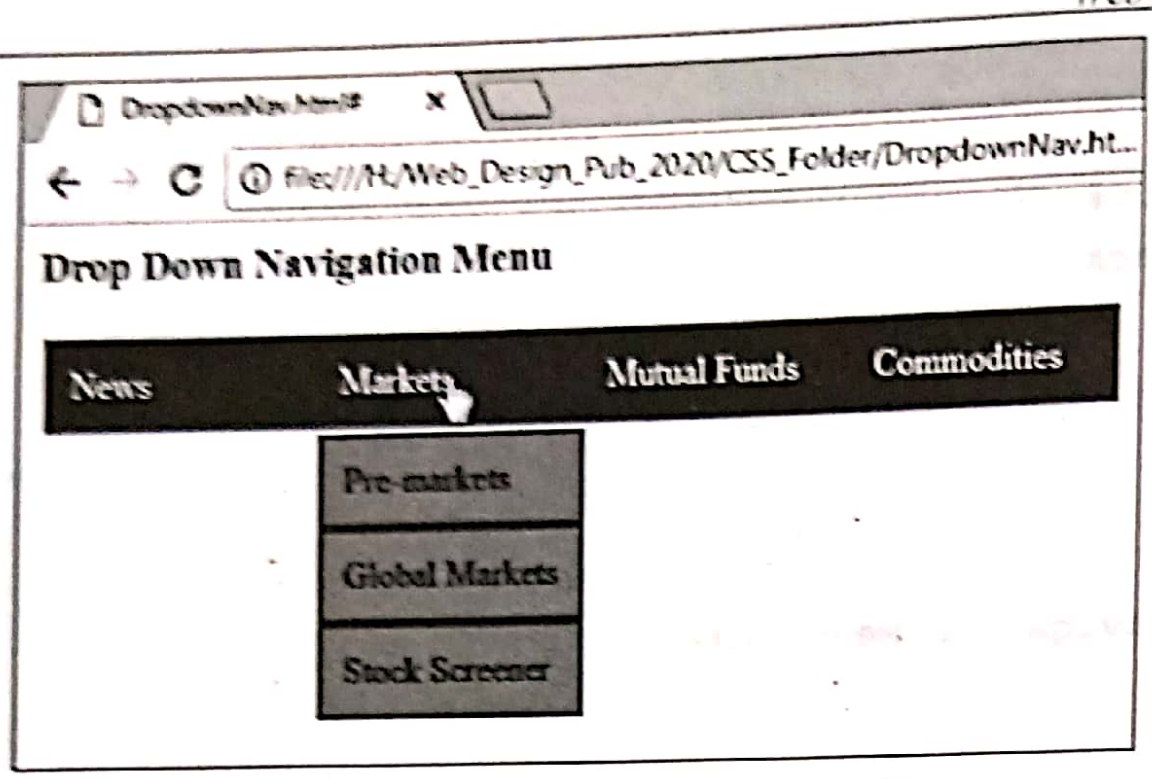


Figure 3.65: Output of the above Example 68

3.11 CSS Image Gallery

Image gallery is used to store and display collection of pictures.

Creating a basic gallery structure

Each gallery contains number of div section.

Each div section contains an image and its description.

Example 69:

```
<!DOCTYPE html>
<html>
<head>
<style>
div.gallery {
  margin: 5px;
  border: 1px solid #ccc;
  float: left;
  width: 280px;
}
div.gallery:hover {
  border: 1px solid #777;
}
div.gallery img {
  width: 100%;
  height: auto;
}
div.desc {
  padding: 15px;
  text-align: center;
}
</style>
```

(Contd...)

Scanned with OKEN Scanner

```
</head>
<body>
<div class="gallery">
  <a target="_blank" href="h:/flower.jpg">
    <img src="h:/flower.jpg" alt="Cinque Terre" width="600" height="400">
  </a>
  <div class="desc">Add a description of the image here</div>
</div>
<div class="gallery">
  <a target="_blank" href="h:/col_range.jpg">
    <img src="h:/col_range.jpg" alt="Forest" width="600" height="400">
  </a>
  <div class="desc">Add a description of the image here</div>
</div>
<div class="gallery">
  <a target="_blank" href="h:/Mount.jpg">
    <img src="h:/Mount.jpg" alt="Northern Lights" width="600" height="400">
  </a>
  <div class="desc">Add a description of the image here</div>
</div>
<div class="gallery">
  <a target="_blank" href="h:/Mount1.jpg">
    <img src="h:/Mount1.jpg" alt="Mountains" width="600" height="400">
  </a>
  <div class="desc">Add a description of the image here</div>
</div>
</body>
</html>
```

The output of the code given in Example 68 as shown in Figure 3.63.

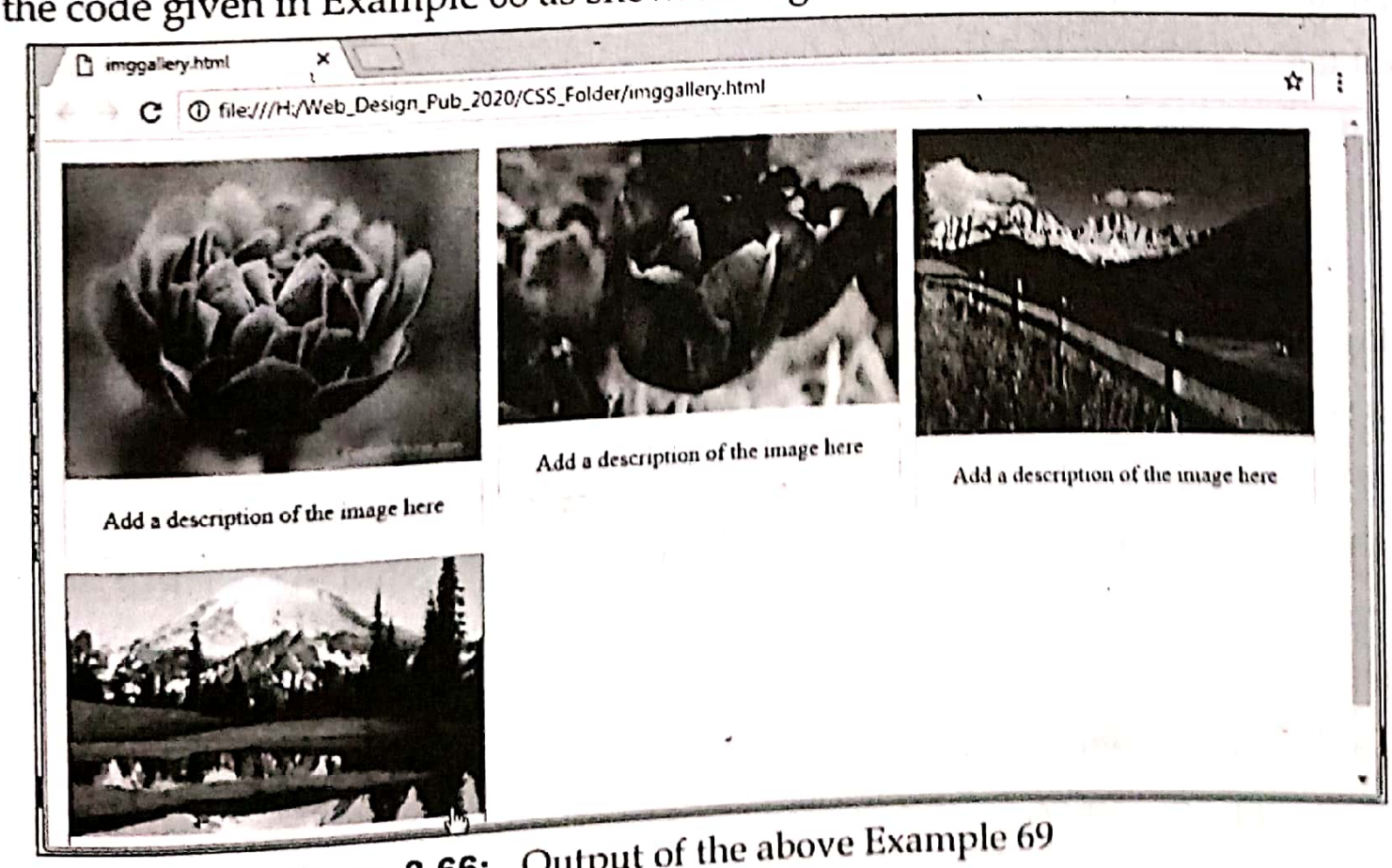


Figure 3.66: Output of the above Example 69

Conclusion

Cascading stylesheets is an integral part of the web development process. A highly effective HTML tool provides easy control over layout and presentation of website pages. Next, we have discussed types of stylesheets to set the style in web pages, which contain HTML elements such as background color, font-size, font-family, color and so on. Next, we have discussed CSS selectors are the part of CSS rule set and used to select html elements according to its Id, Class, Type and Attribute etc. Then, we have discusses CSS basic properties such as background, Block, Box and Border properties. After, we have discusses CSS List and its properties such as specify bullets and numbering in the lists, sets the position of the list-item, sets an image as list item and setting all list properties in a single declaration that is, shorthand property. Next, we have discussed CSS Tables to apply style on HTML tables for better look and feel using CSS properties. Finally, we have discussed Menu design easy to navigate the bar with the combination of lists and links.

REVIEW QUESTIONS WITH ANSWERS

1. Multiple Choice

1.1 CSS is a _____ language.

a. Structural
b. Presentational
b. Markup
d. Behavioral

1.2 Name the syntax of declaring a CSS rule is:

a. property-name = value;
b. property-name : value
c. property-name = value
d. property-name:value;

1.3 Styles that are placed within style tag are called as:

a. External styles
b. Inline styles
c. Embedded styles
d. Html styles

1.4 Which tag is used to link an external style sheet to a webpage?

a. Link tag
b. Style tag
c. Script tag
d. @media

1.5 Which symbol indicates class selector in CSS.

a. Hash
b. Plus
c. Dot
d. None

1.6 To select tags by their id attribute value which selector is used:

a. Class selector
b. ID selector
c. Tag selector
d. Sibling selector

1.7 Which of the following selector selects the element that is the targeted in the URL?

a. :selected
b. :link
c. :target
d. :enabled

1.8 What this code will happen:

```
<p style="color:blue">text</p>
```

a. to make that specific paragraph blue
b. Error

Scanned with OKEN Scanner

c. nothing happen
d. None of the above

1.9 Which of the following CSS list properties is/are correct?

a. set different list item markers for ordered lists
b. Set different list item markers for unordered lists
c. Set an image as the list item marker
d. All of the mentioned

1.10 Which of the following selector selects the element that is the first child of its parent that is its type?

a. :first-of-type
b. :last-child
c. ::first-line
d. ::first-letter

1.11 If we do not want to allow a floating div to the left side of an element, which css property will you use?

a. Margin
b. Clear
c. Float
d. Padding

1.12 Which selector is being used to specify a group of elements

a. Name
b. ID
c. Class
d. Tag

1.13 Which property is used in CSS to change the background color?

a. bgcolor
b. background-color
c. color
d. back-color

1.14 Which of the following selector matches a element based on its id?

a. The universal selector
b. The Descendant Selector
c. The Class selector
d. The id selector

1.15 Which of the following properties will we use to display border around cell without any content?

a. Empty-cell
b. Blank-cell
c. Noncontent-cell
d. no-cell

2. State whether the following statements are True or False

2.1 Inline element has no line break and you cannot set height and width of the element.
2.2 Box-sizing allows you to include the padding and border in element total width and height.
2.3 Linking to an external style sheet allows you to have hyperlinks from your page to the World Wide Web.
2.4 The link element should be place at the top of the body section.
2.5 Specifying an element's position as absolute removes it from the normal flow of elements on the page.
2.6 <a href="mailto:sales124@gmail.com"> is the correct e-mail link.
2.7 Relative positioning removes elements from the general flow of elements on the page.
2.8 You can only use absolute lengths to specify the height and width of an element.
2.9 The box model applies to the span element.

Scanned with OKEN Scanner

2.10 You can write the CSS code once and then you can reuse same sheet in multiple HTML pages.

3. Match the following

3.1	The default value of "position" attribute is	a.	Body {color: black}
3.2	CSS	b.	The property sets the shadow for a box element
3.3	Syntax of background color	c.	Static
3.4	When we write <img src="img.png">, what "img.png" Inside double quote implies.	d.	It defines the different properties of all four sides of an element's border in a single declaration
3.5	Box-shadow	e.	#id
3.6	Background-attachment	f.	p strong {background-color:yellow;}
3.7	Border-width	g.	value
3.8	P{line-height: 150%;} is type of selector	h.	Cascading Style Sheets
3.9	_____ selectors selects any tag with an id attribute set	i.	Element selectors
3.10	Strong element within a p element have a yellow background	j.	It sets the background image to scroll or not to scroll with its associated element's content
		k.	{body;color:black}

4. Fill in the blanks

a. CSS
b. background-attachment
c. margin
d. a {text-decoration:none;}
e. Absolute
f. class
g. Float-align
h. list-style-image
i. border-collapse
j. Display
k. float-left

4.1 _____ property that is used for controlling image-scroll.

4.2 The _____ attribute is used to define the name(s) of the class(es) to which a particular tag belongs.

4.3 _____ property sets a consistent margin on all four sides of the affected element.

4.4 By default, hyperlinks are displayed with an underline. _____ you remove the underline from all hyperlinks by using CSS code.

4.5 _____ has introduced text, list, box, margin, border, color and background properties.

4.6 _____ property assigns a graphic image to a list item.

4.7 If the element has position as _____ that will place the element depends upon its nearest parent element.

Scanned with OKEN Scanner

4.8 _____ property influences the horizontal alignment of an element, making it "float" toward the left or right margin of its containing element.

4.9 Which _____ CSS property is used for controlling the layout.

4.10 _____ CSS property that can be used for collapsing the borders between table cells.

3.7.2 Descriptive Questions with Answer

5.(a) What is CSS?

Answer: It stands for Cascading Style Sheets. It is use for applying the styles for the HTML elements, which is use to describe the look and formatting of a document written in markup language. It can also be use with any kind of XML documents including plain XML and SVG. CSS is designed to separate the presentation and content, including layout and colors, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics. It enables multiple web pages to share formatting by specifying the relevant CSS in a separate .css file.

5.(b) How can you integrate CSS on a web page?

Answer

a. There are three methods to integrate CSS on web pages.
b. Inline method: It is use to insert style sheets in html document.
c. Embedded method: It is use to add a unique style to a single document.
d. External method: CSS can be place in an external file and linked via link element.

5.(c) What are the advantages of CSS?

Answer

The advantages of CSS are:

a. Accessibility
b. Page will load fast
c. Offline browsing
d. Maintenance is easy

6.(a) List out the components of CSS style?

Answer: The different components of CSS style are:

Property

Selector and Value

6.(b) What is the usage of Class selector and name differentiate Class selector from ID selector?

Answers: The .class selector selects elements with a specific class attribute. To select elements with a specific class, write a period (.) character, followed by the name of the class. With ID selector, we can select an element by pointing out the unique ID name set with the id attribute.

If we use that selector can be used only once in html document. If we use that selector on more than one element, our code will not pass validation.

The difference between the Class and ID selector is:

The ID selector consists of a hash (#), followed by the unique ID name of a referring HTML element. ID selector cannot start with a number and must be at least one character long. They also case-sensitive, and must exactly match across HTML, CSS and JavaScript.

The ID selector in CSS document:

```
#para1 {
```

Scanned with OKEN Scanner

```
  text-align:center;
 color: blue;
}
```

Class selector

When we want to style name more than one object on a web page, we should be using a class selector. To use multiple elements in an HTML document can have the same class value. In addition, a single element can have multiple class name separated by white space.

A Class selector can have any name that starts with a letter, hyphen or underscore.

A class selector in html document:

```
<style>
 .center {
  Text-align: center;
 Color: blue;
}
</style>
In CSS a class selector starts with a dot(.), like this:
.bank { /* this is the class selector */
 Text-color: red;
}
```

The only difference between them is the "id" is unique in a page can only apply at one element, while "class" selector can apply to multiple elements.

6.(c) What is the difference between padding and margin?

Answer: In CSS, the margin is the property by which we can create space around elements. We can even create space to the exterior define borders.

In CSS have margin property as follows?

Margin-top, Margin-right, Margin-bottom and margin-left

Margin property has some defined values as shown below:

a. Auto: This property browser calculates the margin.
b. Length: It sets the margin value in px, pt, cm etc.
c. %: It sets the width % of the element.
d. Inherit: This property we can inherit the margin property from the parent element.

In CSS, padding is the property by which we can generate space around an element's content as well as inside any known border;

CSS padding also has properties like:

Padding-top, Padding-right, padding-bottom and padding-left. Negative values are not allowed in padding.

7.(a) What are the difference between relative and absolute in CSS?

Answer: The main difference between relative and absolute is that "relative" is used for the same tag in CSS and it means that if we write the left:10px then the padding will shift to 10px in the left while absolute is totally relative to the non-static parent. It means if we write left:10px then the result will be 10px far from the left edge of the parent element.

Scanned with OKEN Scanner

7.(b) Differentiate between Inline and block element.

Answer: Inline element does not have an element to set width and height and also it does not have the line break.

Example: links <a>, emphasized words , important words , and abbreviations <abbr>

Block elements are meant to structure the main parts of our page, by dividing our content in coherent block.

Example: width and height, paragraphs <p>, lists: unordered <ul> and ordered lists

Headings: from 1st level <h1> to 6th level headings <h6>, articles <article> and sections <section>

8.(a) How do we make a rounded corner by using CSS?

Answer: We can make a rounded corner by using the property "border-radius". We can apply this property to any element.

```
<html>
<head>
<style>
 #rcorner {
 Border-radius: 25px;
 Background: gray;
Padding: 20px;
Width: 200px;
Height: 150px;
}
</style>
</head>
<body>
<h3>The border-radius property</h3>
<p id-"rcorner">Rounded corners</P>
</body>
</html>
```

Run the above code in the browser.

8.(b) What is the use of the Box Model in CSS and what are its elements?

Answer: In CSS, the box model is a box that binds all the html elements and it includes features like margins, border, padding and the content.

Margin: the top most layer, the overall structure showed.

Border: the padding and content option with a border around is shown. Background color affects the border.

Padding: space showed. Background color affects the border.

Content: Actual content is showed.

By using a box model we will get the authority to add the borders all around the elements and we can also define the space between the elements.

9. (a) Write the following background element in CSS?

(i) Which property is use to set the background color of an element?

(ii) Which property is use to set the background image of an element?

Scanned with OKEN Scanner

(iii) Which property is use to control the position of an image in the background?

(iv) Which property is used to control the scrolling of an image in the background?

Answer

(i)The background-color property is use to set the background color of an element.

(ii)The background-image property is use to set the background image of an element.

(iii)The background-position property is use to control the position of an image in the background.

(iv)The background-attachment property is used to control the scrolling of an image in the background.

(b) What is the difference between "display:none" and "visibility:hidden" in CSS?

Answer

Display:none: This will just hide the element and does not take any space of the element.

Display:hidden: This also hides the element and will take space for the element and this will affect the entire layout of the document.

(c) List out the possible values for attribute – position in CSS?

Answers: The list of possible values for attribute –"position"

Static, Inherit, Fixed, Absolute and relative.

(d) Which property is used for underlying the link in CSS?

Answers: Property – "text decoration" is used to for underlining the link.

(e) what is the difference between "cell-padding and "cell-spacing"?

Answers: Cell-padding: It is used to leaver the space between the content of cell and wall/border of the cell.

Cell-spacing: It is used to specify the space between the cells.

Answers

1.1	b	1.2	d	1.3	c	1.4	a	1.5	c	1.6	b
1.7	c	1.8	a	1.9	d	1.10	a	1.11	b	1.12	c
1.13	b	1.14	d	1.15	a						
2.1	T	2.2	T	2.3	F	2.4	F	2.5	T	2.6	T
2.7	F	2.8	F	2.9	F	2.10	T				
3.1	c	3.2	h	3.3	a	3.4	g	3.5	b	3.6	j
3.7	d	3.8	i	3.9	e	3.10	f				
4.1	b	4.2	f	4.3	c	4.4	d	4.5	a	4.6	h
4.7	e	4.8	g	4.9	j	4.10	i				

Scanned with OKEN Scanner

CHAPTER 4

CSS Framework

4.1 Introduction to W3.CSS

W3CSS is one of the most famous used CSS framework for in-built responsive websites. It is based on plain CSS that makes it fast. It does not use any additional library that slows a website such as JavaScript or JQuery. It is compatible with all the browsers including **Chrome, Firefox, Opera** and so on. It is also compatible with most of the devices currently use in mobile phones to desktops.

Structure

- Introduction to CSS and its types
- CSS Selectors
- CSS Properties
- CSS Lists and Tables

Objectives

- The reader will be able to understand the following:
- Purpose of CSS
- Role of CSS in websites
- Roles of effects in website

4.2 Web site Development using W3.CSS Framework

W3.CSS is a CSS developed by w3schools.com. It is a modern CSS frame with built-in responsiveness. It supports responsive mobile design by default, and it is smaller and faster than similar CSS frameworks. It also speed up and simplify web development because it is easier to learn to use than other CSS framework. The main advantage is that it is fee to use without any licensing required.

4.2.1 How to Use W3.CSS?

There are two ways to use W3.CSS:

Local Installation: You can download the W3.CSS file on your local machine and include it in your HTML code.

CDN Based Version: You can include the W3.CSS file into your HTML code directly from the **Content Delivery Network (CDN).**

Local installation go to https://www.w3schools.com/w3css_download.asp to download the latest version.

To use W3.css, just add a link to "w3.css" in your web pages.

For example,

```
<!DOCTYPE>
<html>
<title>First program in w3.css</title>
<meta name="viewpoint" content="width-device-width, initial-scale=1">
<lin rel="stylesheet" type="text/css" href="//foldername/w3.css">
Basic Usage
```

Scanned with OKEN Scanner

All of the CSS classes start with a w3 prefix and you can add it inline inside the class of the tag.

```
<div class="w3-container"></div>
```

Now, let us take your first simple example to see how W3.CSS can actually be:

Example 1:

```
<!DOCTYPE html>
<html>
<head>
<title>My First Program in W3.CSS</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet" type="text/css" href="w3.css">
</head>
<body>
<div class="w3-container w3-blue">
<h1>Nature</h1>
</div>
<img src="h:/Flower.jpg" alt="Falls" style="Width:60%">
<div class="W3-container">
<p>Close your eyes and imagine the sunrise over a wide-open country field, morning rain glistening on forest trees, and snow-covered mountaintop peaks. Yes, nature is absolutely breathtaking, which is why we rounded up these nature quotes that capture the earth's stunning beauty, even when it almost seems indescribable. If you don't have an outdoor weekend getaway in the works, these outdoor quotes will encourage you to schedule a family road trip as soon as possible. And if you're planning your own hiking trip or camping adventure, you might become stressed with your to-do list. To make your worries melt away, simply scroll through these quotes about the wonderful wilderness out there. Doing so will instantly relax you and make you feel unbelievably grateful for this beautiful planet we get to call home.</p>
</div>
</body>
</html>
```

The output of the above code as shown in Figure 4.1.

4.3 W3CSS Colors Basics

Color is important aspect of visual attractive of a website. Pick the right color combination can make a site more appealing for the visitor. W3.CSS framework comes with a variety of colors that can be used to perform different tasks. There are four different classes perform these tasks. These are:

1. It is used to sets background color of any html element.
2. It is used to sets the color of the text of an element.
3. It is used to sets the background color when mouse is hovered over an element.
4. It is used to changes the color of the text when mouse is hovered over it.

4.3.1 W3-Color

The w3-color classes set the background color for any html element.

Scanned with OKEN Scanner

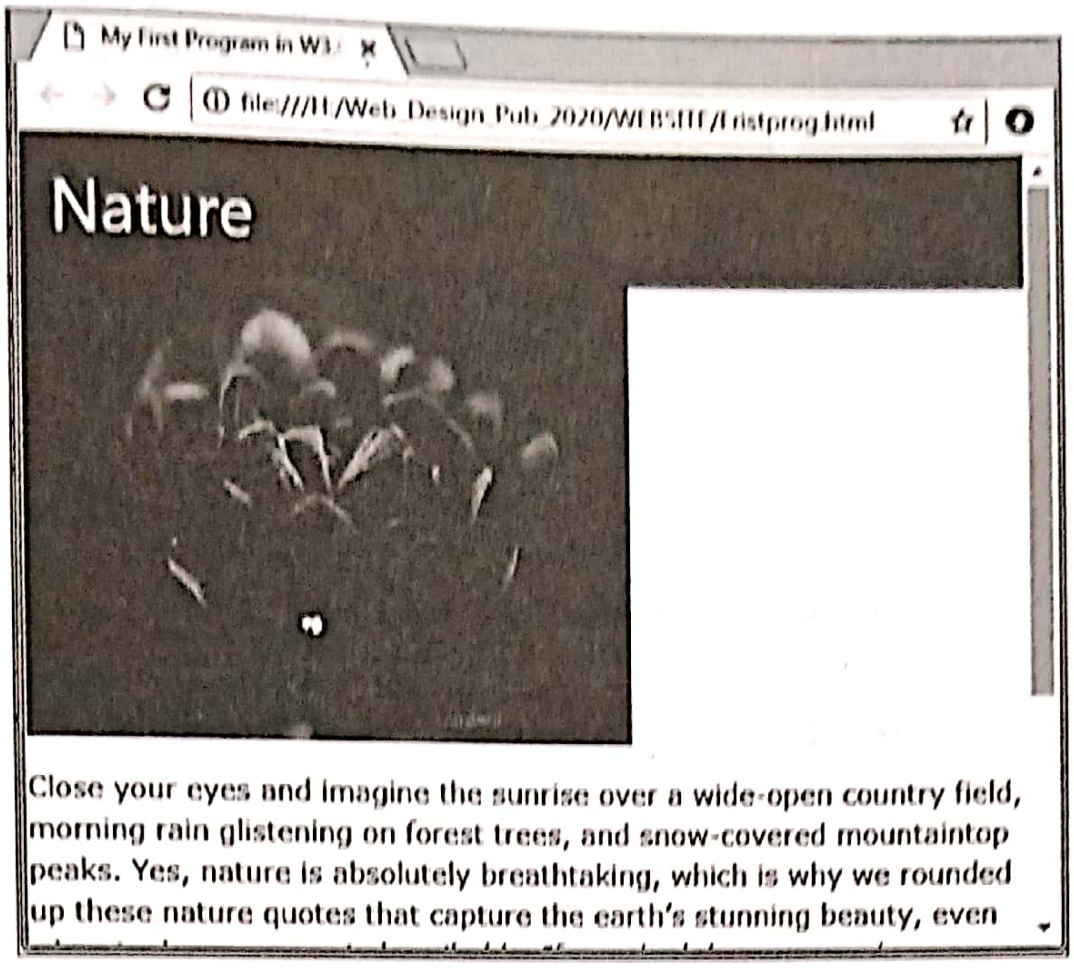


Figure 4.1: Output of the above Example 1

Example 2:

```
<!DOCTYPE html>
<html>
<title>W3.CSS Colors</title>
<head>
<meta name="viewpoint" content="width=device-width, initial-scale=1">
<link rel="stylesheet" type="text/css" href="w3.css">
</head>
<body>
<div class="w3-container w3-red">
<h3>This is with dartk red background</h3>
</div>
<div class="w3-container w3-blue">
<h3>This is with blue background</h3>
</div>
<div class="w3-container w3-teal">
<h3>This is with Teal background</h3>
</div>
<div class="w3-container w3-yellow">
<h3>This is with pale yellow background</h3>
</div>
</body>
</html>
```

The output of the code given in Example 2 as shown in Figure 4.2.

You can see in the above code that it contains four divs. All the three divs have a class of w3-container which adds properties of a container to these classes.

Scanned with OKEN Scanner

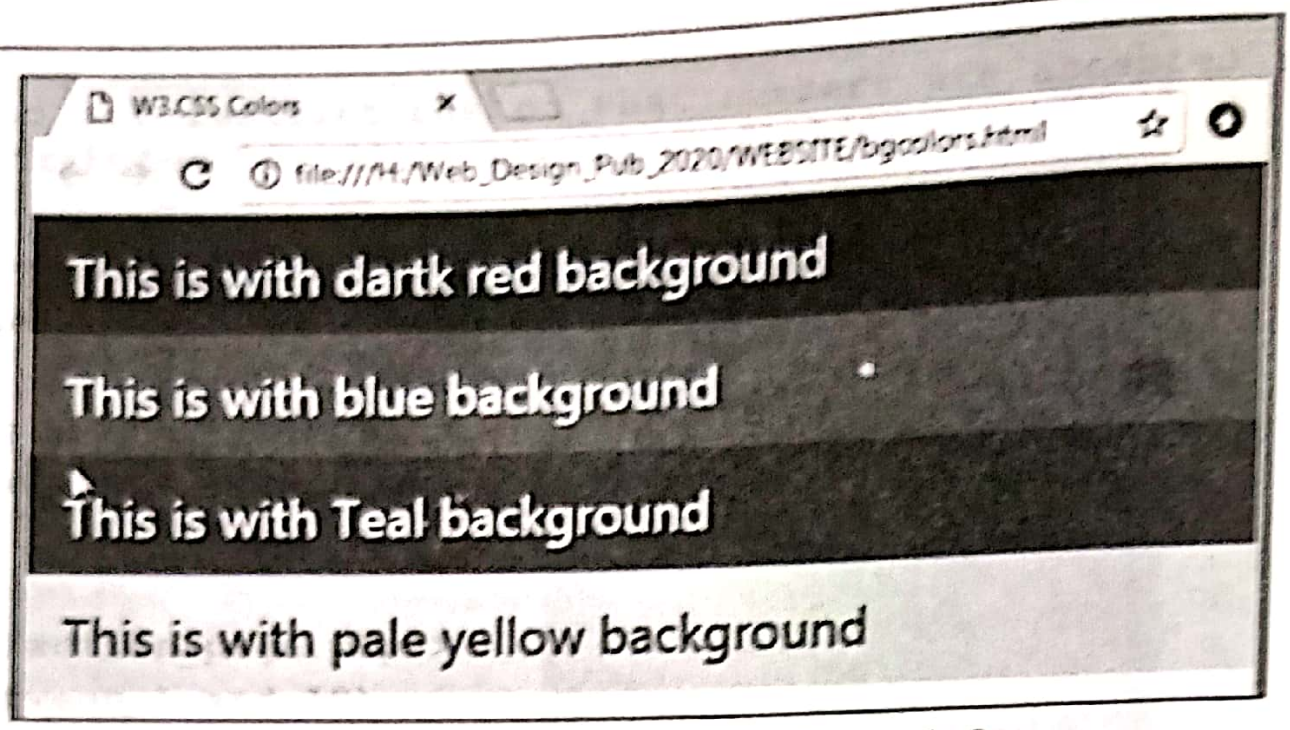


Figure 4.2: Output of the above Example 2

4.3.2 W3-Text-color

The w3-text color class is used to set the text color of an element.

Example 3:

```
<!DOCTYPE html>
<html>
<title>W3.CSS Text Colors</title>
<head>
<meta name="viewpoint" content="width=device-width, initial-scale=1">
<link rel="stylesheet" type="text/css" href="w3.css">
</head>
<body>
<div class="w3-container w3-text-lime">
<h3>This text color is Lime</h3>
</div>
<div class="w3-container w3-text-purple">
<h3>This text color is deep purple</h3>
</div>
<div class="w3-container w3-text-gray">
<h3>This text color is gray</h3>
</div>
<div class="w3-container w3-text-cyan">
<h3>This text color is cyan</h3>
</div>
</body>
</html>
```

The output of the code given in Example 3 as shown in Figure 4.3.

4.3.3 W3CSS Hover Colors

There are two classes change the color upon mouse hover. The first one is the w3-hover-color class. This class sets the background color of an element when mouse hovers over it. The second class is W3-hover-text color which sets the color of the text upon mouse hover. Let us have a look at each of them in example 4.

Scanned with OKEN Scanner

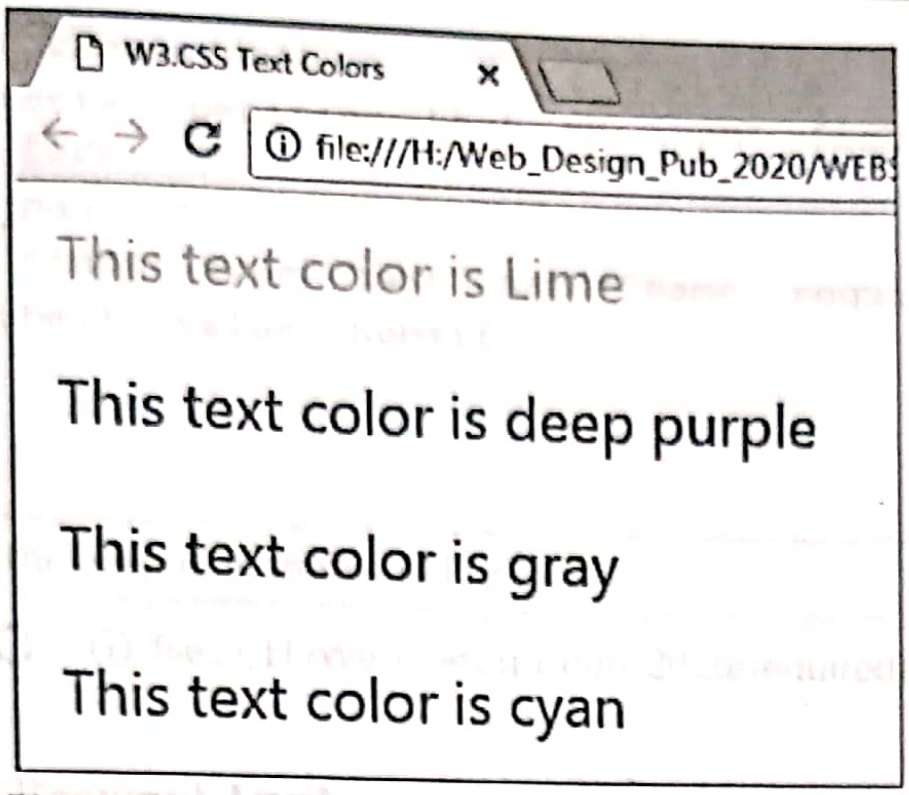


Figure 4.3: Output of the above Example 3

Example 4:

```
<!DOCTYPE html>
<html>
<title>W3.CSS Hover Colors</title>
<head>
<meta name="viewpoint" content="width=device-width, initial-scale=1">
<link rel="stylesheet" type="text/css" href="w3.css">
</head>
<body>
<div class="w3-container w3-Indigo">
<h3>Hover Colors</h3>
</div>
<div class="w3-container w3-teal w3-hover-green">
<h3>This is with teal background</h3>
</div>
<div class="w3-container w3-gray w3-hover-blue">
<h3>This is with gray background and hovering with blue</h3>
</div>
<div class="w3-container w3-orange w3-hover-deep-orange">
<h3>This is with orange background and hovering with deep orange</h3>
</div>
</body>
</html>
```

The output of the code given in Example 4 as shown in Figure 4.4.

4.3.4 W3-hover-text-color

If you want to change the color of text upon mouse hover, you can use w3-hover-text color class.

4.3.5 W3.CSS Containers

Like colors classes, W3.CSS frameworks contains a class that can be used to style container type elements

Scanned with OKEN Scanner

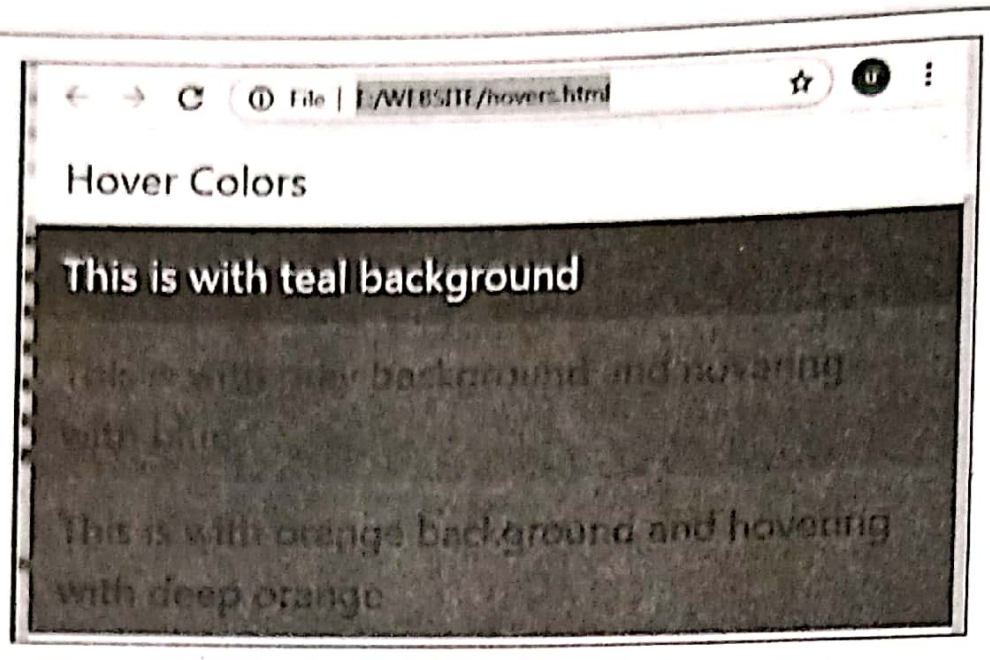


Figure 4.4: Output of the above Example 4

such as , <article>, <section>, <header>, <footer>, <form> etc. The class used for W3.CSS containers is W3-container. This class add 16px to the left and right of an HTML element. It also used to style multiple container type elements by adding uniform padding, margins, fonts, colors and alignments.

To use a container, just add a w3-container class can be used to style header and footer elements.

Example 5:

```
<!DOCTYPE html>
<html>
<title>W3.CSS Hover Colors</title>
<head>
<meta name="viewpoint" content="width=device-width, initial-scale=1">
<link rel="stylesheet" type="text/css" href="w3.css">
</head>
<body>
<header class="w3-container w3-purple w3-text-blue">
<h2>BPB Publications</h2>
</header>
<footer class="W3-container w3-lime w3-text-cyan">
<h4>Ansari Road, Daryaganj</h4>
</footer>
</body>
</html>
```

The output of the code given in Example 5 as shown in Figure 4.5.

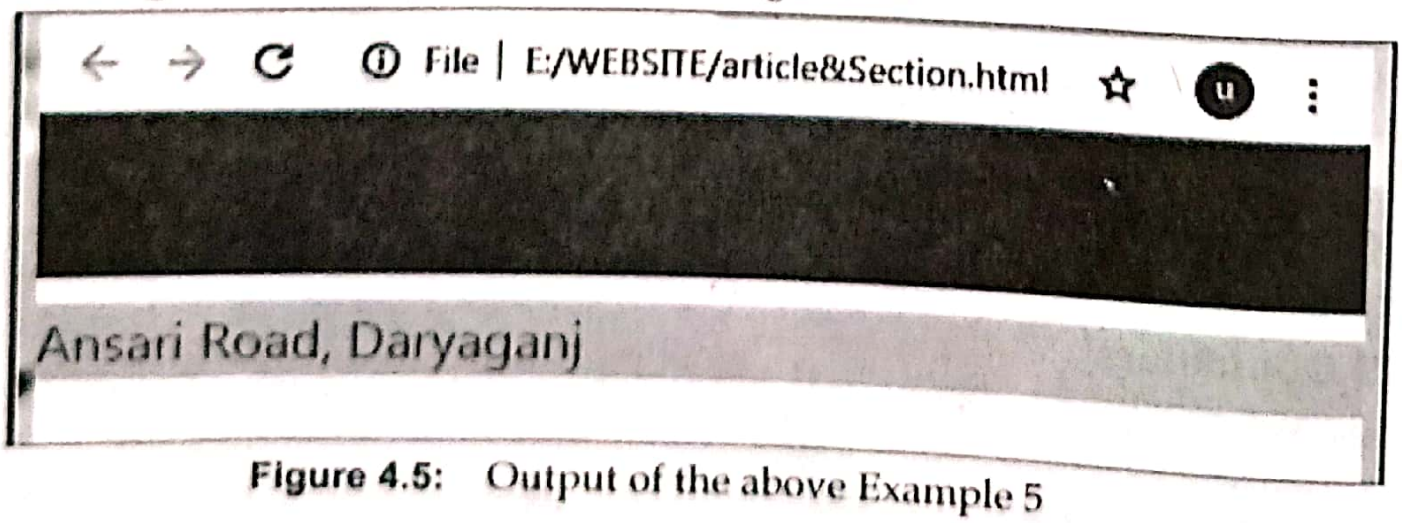


Figure 4.5: Output of the above Example 5

Scanned with OKEN Scanner

In the above code, you have a header and footer. The w3-container class has been added to both of them. Inside the header, you have h2 heading while inside the footer, you have h4 heading. The background colors and text colors of both the header and footer have same fonts, common margin and padding.

4.3.6 Articles and Sections

W3-container class can also add predefined styles to <article> and <section> elements.

Example 6:

```
<!DOCTYPE html>
<html>
<title>W3.CSS </title>
<head>
<meta name="viewpoint" content="width=device-width, initial-scale=1">
<link rel="stylesheet" type="text/css" href="w3.css">
</head>
<body>
<div class="W3-container w3-teal">
  <h2>Medical Journals</h2>
  <p>The journal will cover technical and clinical studies related to health,
ethical and social issues in field of health policy submitted to biomedical
journal website.</p>
<hr>
</div>
<article class="w3-container w3-yellow">
<h2>Expression of vascular endothelial growth factor.</h2>
<p>The oral cavity is the upper most part of the digestive tract.</p>
<hr>
</article>
<section class="w3-container w3-brown">
<h2>Medical Ethics</h2>
<p>Articles dealing with ethical aspects of practice of medicine or medical
research in India.</p>
<hr>
</section>
</body>
</html>
```

The output of the code given in Example 6 as shown in Figure 4.6.

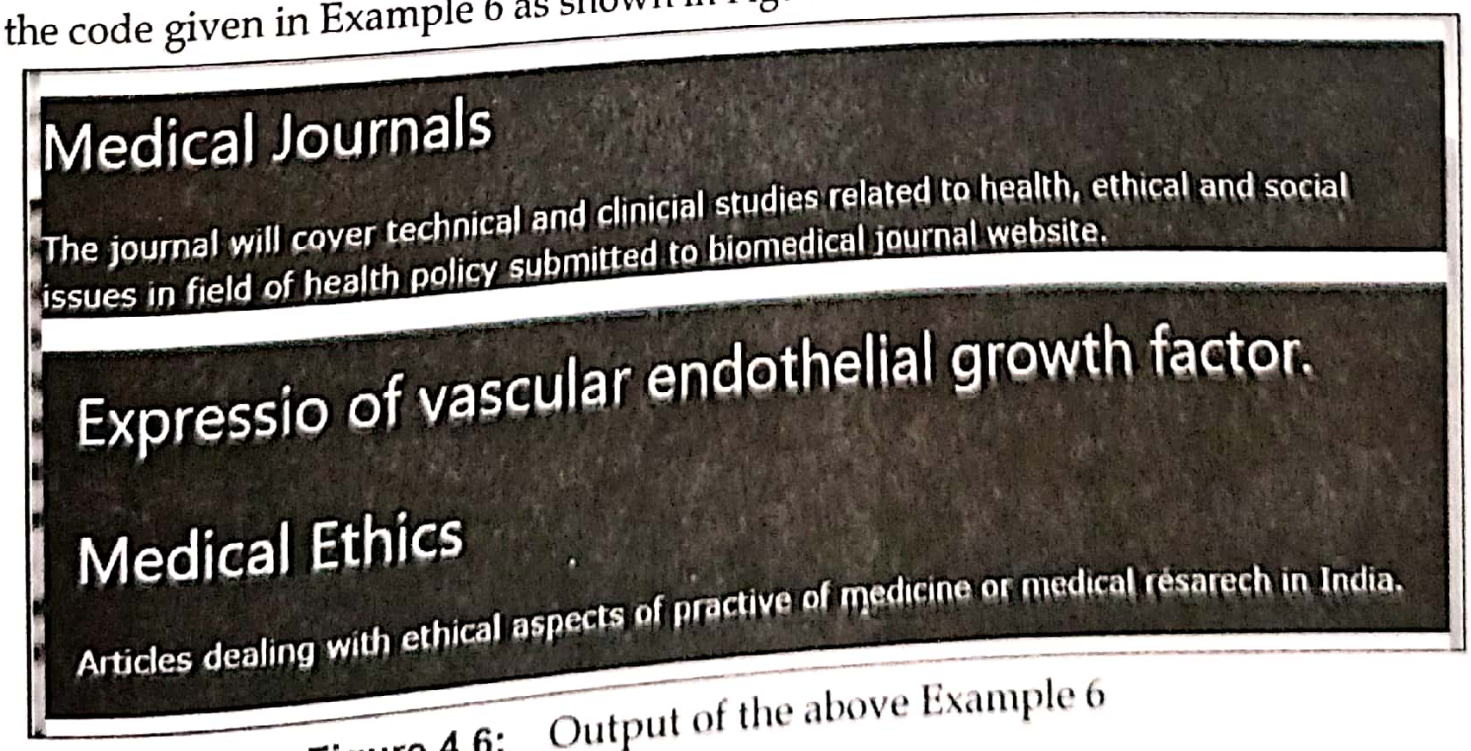


Figure 4.6: Output of the above Example 6

Scanned with OKEN Scanner

4.3.7 Container Padding

The w3-container class has a default 16px left and right padding, and no top or bottom padding.

Example 7:

```
<!DOCTYPE html>
<html>
<title>W3.CSS </title>
<head>
<meta name="viewpoint" content="width=device-width, initial-scale=1">
<link rel="stylesheet" type="text/css" href="w3.css">
</head>
<body>
<div class="w3-container w3-blue">
The w3-container class has a default 16px left and right padding, and no top
or bottom padding.
</div>
</body>
</html>
```

The output of the code given in Example 7 as shown in Figure 4.7.

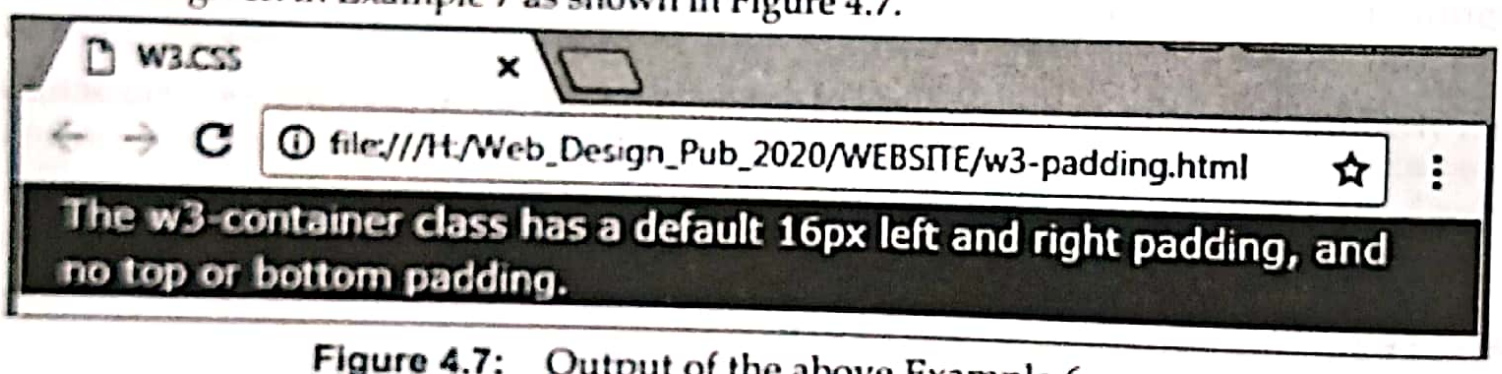


Figure 4.7: Output of the above Example 6

4.4 W3.CSS Panels

W3.CSS panels are a type of container; however, panels add top and bottom margin of 16px in addition to left and right padding of 16px. It is use for displaying notes, quotes, alerts, cards etc. You can also create closeable and showable panels. Let us create these different types of panels.

Example 8:

```
<!DOCTYPE html>
<html>
<title>W3.CSS </title>
<head>
<meta name="viewpoint" content="width=device-width, initial-scale=1">
<link rel='stylesheet" href="w3.css">
</head>
<body>
<div class="w3-container">
 <h3>Displaying Panels</h3>
 <p>Panels are the same as containers except for adding top and bottom margin.</p>
```

Scanned with OKEN Scanner

```
<div class="w3-panel w3-blue-gray">
  <h2>Panels are the type of container</h2>
  <p>This is a panel and it has blue gray background.</p>
</div>
<div class="w3-panel w3-green">
  <p>This is a panel and it has green background.</p>
</div>
<div class="w3-container w3-blue">
 <p>This is container</p>
</div>
<div class="w3-container w3-yellow">
 <p>This is container</p>
</div>
</div>
</body>
</html>
```

The output of the code given in Example 8 as shown in Figure 4.8.

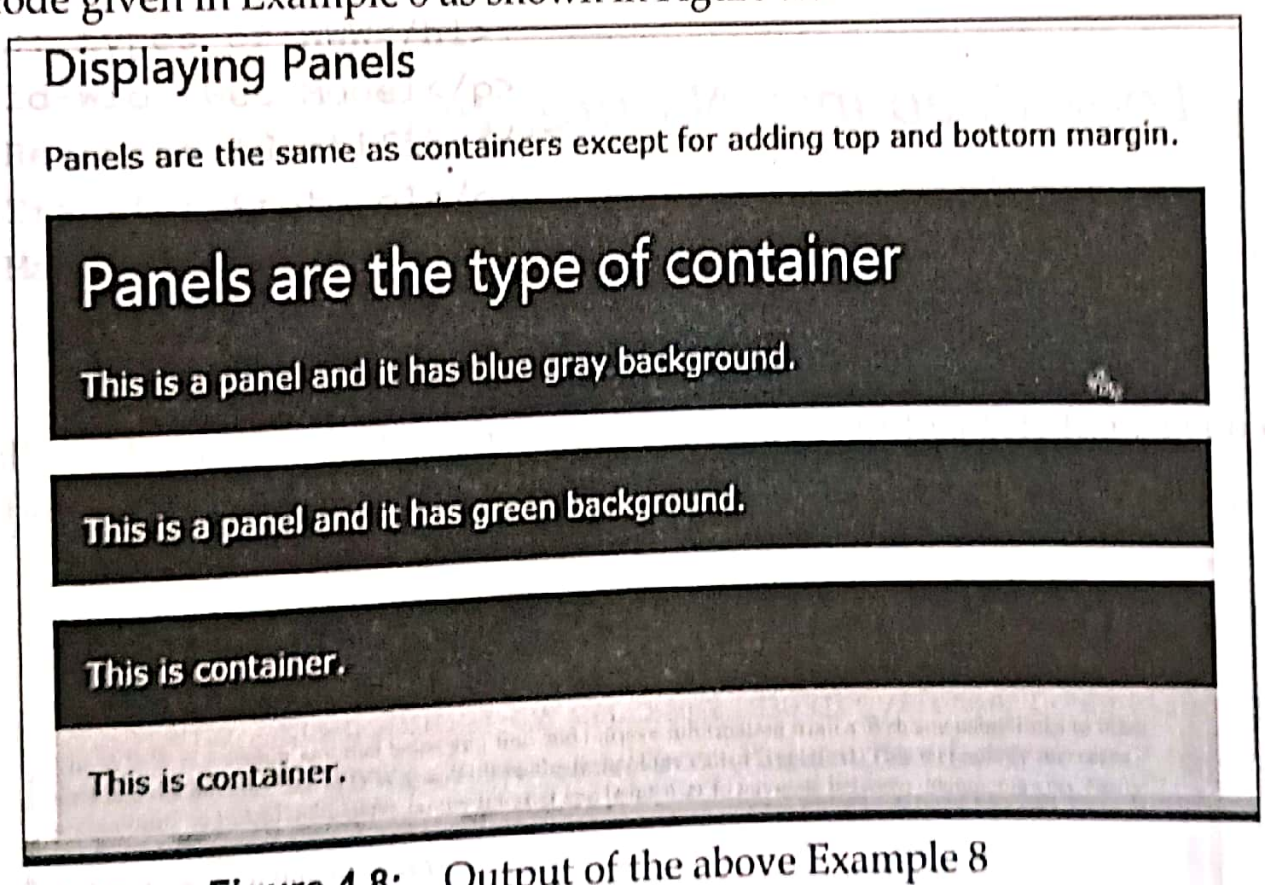


Figure 4.8: Output of the above Example 8

4.4.1 Using Panel for Creating Notes

W3.CSS panel can also be used to create quotes types effect.

Example 9:

```
<!DOCTYPE html>
<html>
<title>W3.CSS </title>
<head>
<meta name="viewpoint" content="width=device-width, initial-scale=1">
```

Scanned with OKEN Scanner

```
<link rel="stylesheet" href="w3.css">
</head>
<body>
<div class="w3-container">
 <h3>Panles for Quotes</h3>
 <p>The w3-panel class can be used for display Quotes</p>
<div class="w3-panel w3-pale-yellow w3-leftbar w3-xxlarge w3-serif">
     <p>Look deep into Nature, and then you will understand everything
better.</p>
</div>
</div>
</body>
</html>
```

The output of the code given in Example 9 as shown in Figure 4.9.

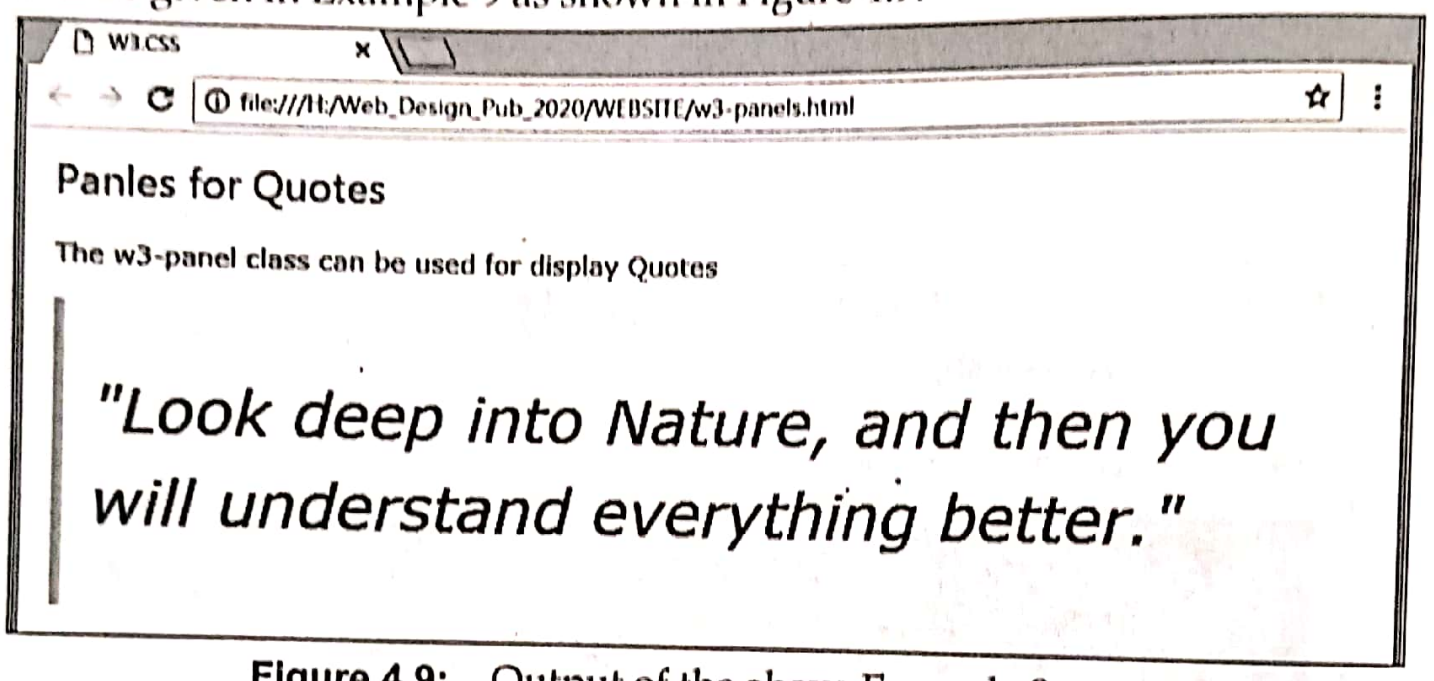


Figure 4.9: Output of the above Example 9

In the above example, you have added pale yellow and left bar, you have added w3 xxlarge w3-sans-serif classes to create a quotes effect. Inside the panel div, you we have italicized the text to create real quote effect.

4.4.2 W3.CSS Rounded Panels

Creating rounded panel is easy; you have to add the class w3-rounded-size, where you have to replace size with the roundness size you want. In *Example 9*, you have use to large rounded border with round-xlarge size.

Example 10:

```
<!DOCTYPE html>
<html>
<title>W3.CSS </title>
<head>
<meta name="viewpoint" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="w3.css">
</head>
```

(Contd...)

Scanned with OKEN Scanner

```
<body>
<div class="w3-panel w3-blue-grey w3-round-xlarge">
<h3>Rounded Panels</h3>
 <p>Look deep into Nature, and then you will understand everything
better.</p>
 </div>
</body>
</html>
```

The output of the code given in Example 10 as shown in Figure 4.10.

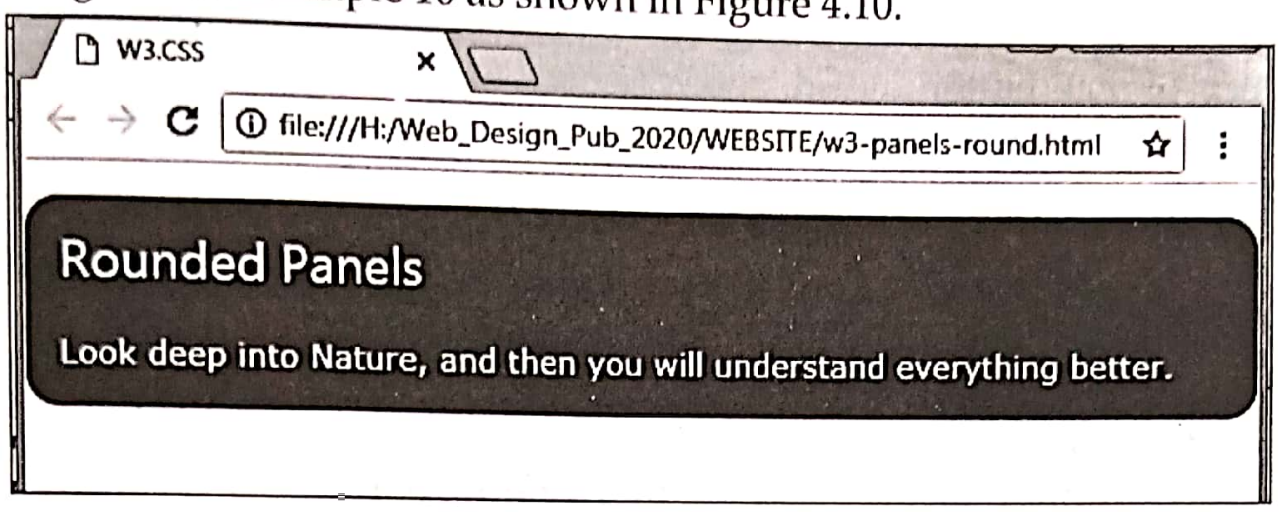


Figure 4.10: Output of the above Example 10

4.4.3 W3CSS Closing and Opening Panels

Here, you have a button whose onclick event opens a div with id "paneldiv". This button opens a panel. Similarly, if you look at the panel div, here first element is the span tag with class "w3-closebtn", the onclick even of this span tag sets the display property of the parent div that is panel div to none, which hides the panel.

Example 11:

```
<!DOCTYPE html>
<html>
<title>W3.CSS </title>
<head>
<meta name="viewpoint" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="w3.css">
</head>
<body>
  <button class="w3-btn w3-deep-orange" onclick="document.getElementById
     ('paneldiv').style.display='block'">Show Panel</button>
    <div id="paneldiv" class="w3-panel w3-blue" style="display:none">
    <span  onclick="this.parentElement.style.display='none'"  class="w3-button
w3-large
         w3-display-topright">X</span>
  <h2>This is Panel</h2>
  <p>Click the X button to close the panel</p>
 </div>
</body>
</html>
```

Scanned with OKEN Scanner

The output of the code given in Example 11 as shown in Figure 4.11.

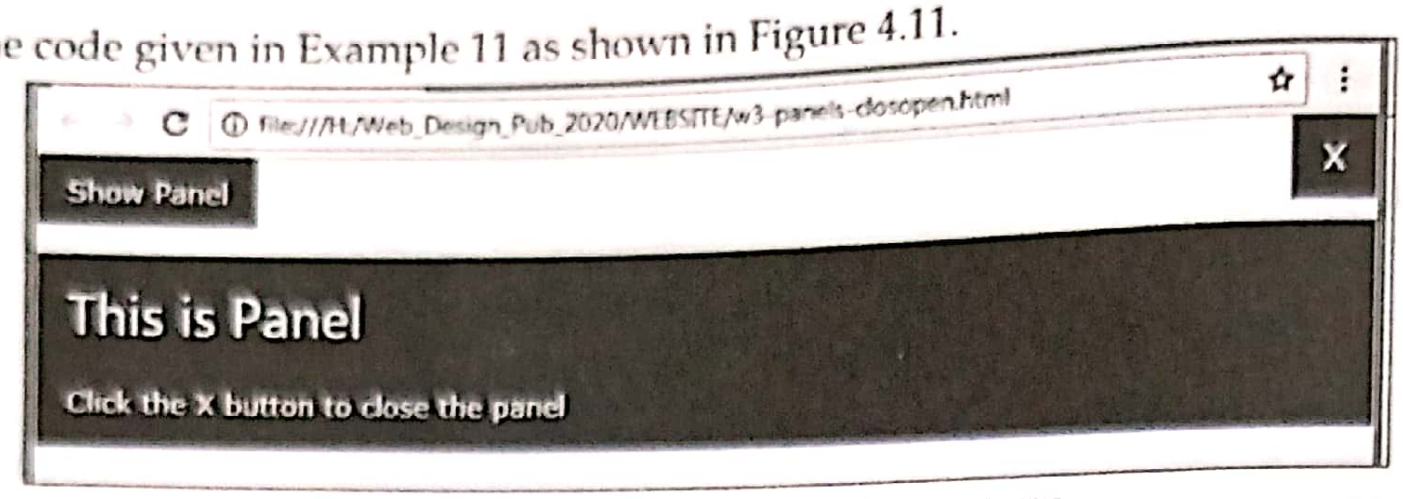


Figure 4.11: Output of the above Example 11

You should see a deep orange button, which read "Show Panel." If you click the button, the blue panel appears. Notice that blue panel has a cross button at top right corner, clicking the cross closes the panel.

4.4.4 W3.CSS Card

When you are talk about cards that you came in your minds are greeting cards or business. W3.CSS cards are used to create effects like these. These are containers with shadowed border. To create these cards it used w3-card-x where x is the number of pixels of shadowed border. It can be 2,4,8,12 etc. You can add containers within a card div. To create multiple sections within a card. For example, you can create header, content and footer section using container classes within a card. Let us create a card with header and footer in the following example.

Example 12:

```
<!DOCTYPE html>
<html>
<title>W3.CSS </title>
<head>
<meta name="viewpoint" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="w3.css">
</head>
<body>
   <div class="w3-card-4">
  <header class="w3-container w3-teal">
  <h2>BPB Publications</h2>
  </header>
    <div class="w3-container">
  <h5>Largest selling Computer Science Books</h5>
  </div>
  <footer class="w3-container w3-teal">
  <h5>Copyright @ Bpbonline.com</h5>
  </footer>
</div>
</body>
</html>
```

The output of the code given in Example 12 as shown in Figure 4.12.

4.4.5 Hover Shadow Effect

You can add shadow effect to any div when you move mouse over it by adding w3-hover-shadow class to the div.

Scanned with OKEN Scanner

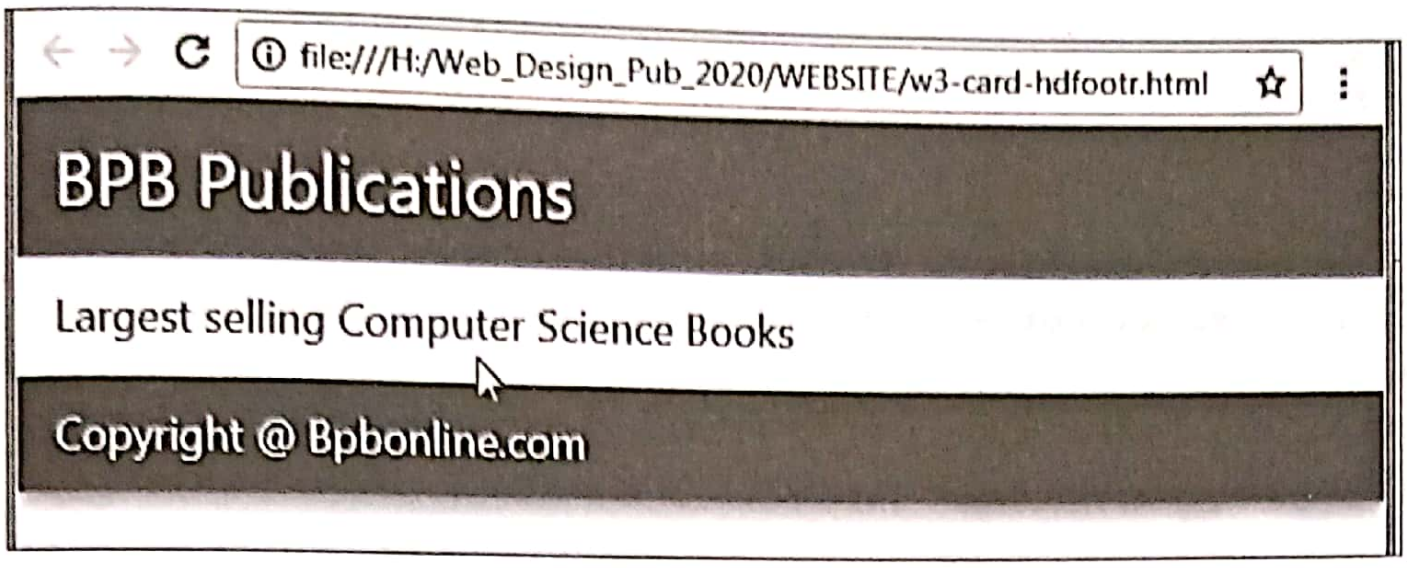


Figure 4.12: Output of the above Example 12

For example,

```
<div class="w3-blue w3-hover-shadow w3-center">
<h2>Move Mouse Over Me</h2>
</div>
```

When you hover your mouse over that div, you shall see a shadow appear below the div.

4.5 W3.CSS Borders

You have covered containers in the last article. However, W3.CSS allows you to different types of borders around html elements. W3.CSS borders are implement via twelve different border classes. These are classes are as follows:

- w3-border: This class adds border on all sides of elements.
- w3-border-right: It adds border on the right side of the element.
- w3-border-left: It adds border on the left side of the element.
- w3-border-top: It adds border at the top of the element.
- w3-border-bottom: It adds border at the bottom of the element.
- w3-border-0: It removes border from the element.
- w3-rightbar: It adds a thick border to the right of the element.
- w3-leftbar: It adds a thick border to the left of the element.
- w3-bottombar: It adds a thick border to the bottom of the element.
- w3-topbar: It adds a thick border to the top of the element.
- w3-border-color: It adds specified colored border.
- w3-hover-border-color: It change the color of the border upon mouse hover.

Example 13:

```
<!DOCTYPE html>
<html>
<title>W3.CSS </title>
<head>
<meta name="viewpoint" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="w3.css">
</head>
```

(Contd...)

Scanned with OKEN Scanner

```
<body>
   <div class="w3-container">
    <h2>Different Borders</h2>
   </div>
   <div class="w3-container w3-border w3-border-yellow">
    <p>This is overall border.</p>
    </div>
   <div class="w3-container w3-border-red w3-border-left">
   <p>This is left border.</p>
   </div>
   <div class="w3-container w3-border-purple w3-border-right">
    <p>This is right border.</p>
   </div>
   <div class="w3-container w3-border-bottom w3-border-top">
   <p>This is top and bottom border.</p>
   </div>
</body>
</html>
```

The output of the code given in Example 13 as shown in Figure 4.13.

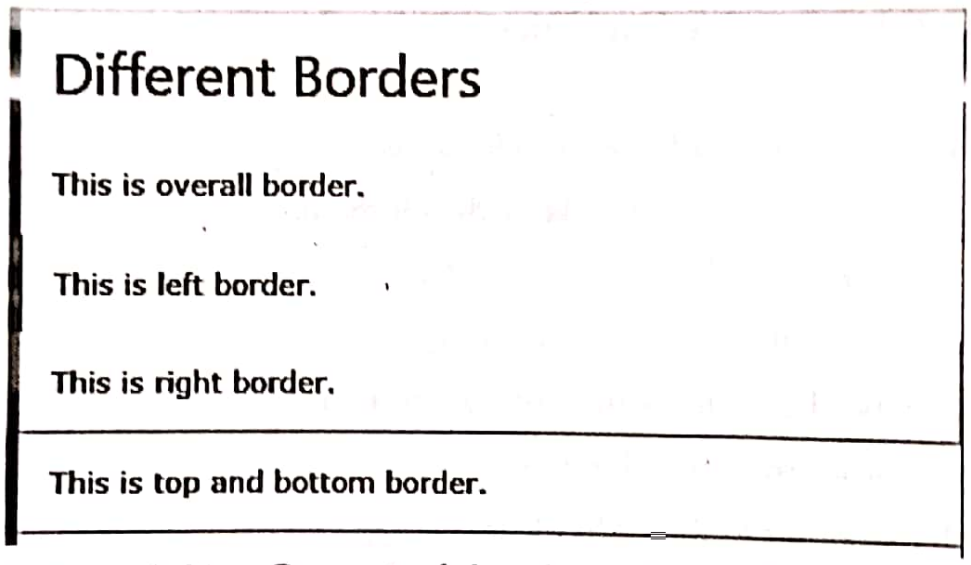


Figure 4.13: Output of the above Example 13

4.5.1 W3.CSS Thick Borders

To add thick borders around an html element, to use w3-rightbar, left-bar, top-bar and bottom-bar classes. Let us look at the following example.

Example 14:

```
<!DOCTYPE html>
<html>
<title>W3.CSS </title>
<head>
<meta name="viewpoint" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="w3.css">
</head>
<body>
```

(Contd...)

Scanned with OKEN Scanner

```
<div class="w3-container">
 <h2>Different Borders thickness</h2>
</div>
<div class="w3-panel w3-leftbar w3-border-blue">
<p>This is thick blue left broder</p>
</div>
<div class="w3-panel w3-rightbar w3-border-green">
<p>This is thick green right border</p>
</div>
<div class="w3-panel w3-bottombar w3-topbar w3-border-red">
 <p>This is thick red top and bottom bar border</p>
</div>
<div class="w3-panel w3-topbar w3-orange w3-border-indigo">
<p>This is top border with background color orange</p>
</div>
</body>
</html>
```

The output of the code given in Example 14 as shown in Figure 4.14.

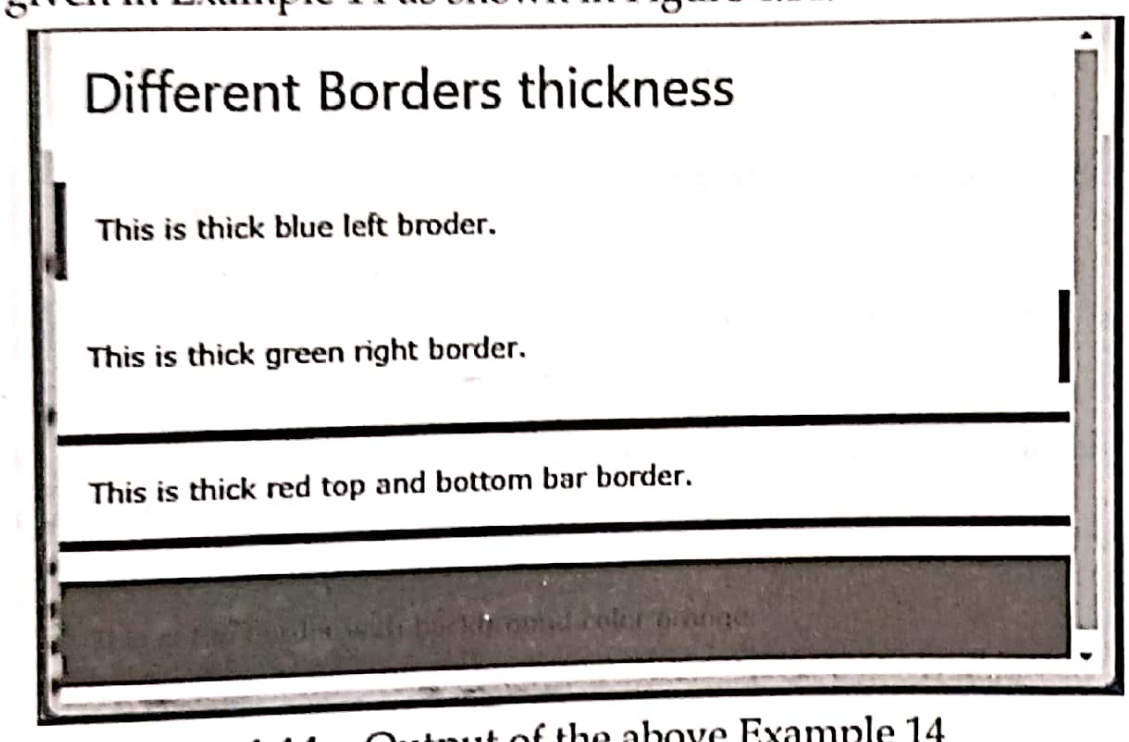


Figure 4.14: Output of the above Example 14

4.5.2 W3.CSS Rounded and Hoverable Borders

It very easy to create rounded border around html element. To use w3-round-size class along with w3-border class. Similarly, to create hoverable border effect, use w3-hover-border-color class with the element. Let us look at the following example.

Example 15:

```
<!DOCTYPE html>
<html>
<title>W3.CSS </title>
<head>
<meta name="viewpoint" content="width=device-width, initial-scale=1">
```

(Contd...)

Scanned with OKEN Scanner

```
<link rel="stylesheet" href="w3.css">
</head>
<body>
   <div class="w3-container">
    <h2>Hoverable Borders</h2>
   </div>
   <div class="w3-panel w3-border w3-border-purple w3-hover-border-green">
   <p>Border that turns red on hover</p>
   </div>
   <div  class="w3-panel  w3-border  w3-border-light-grey  w3-hover-border-
yellow">
   <p>Border that turns yellow on hover</p>
   </div>
<div class="w3-panel w3-orange w3-border w3-round-xlarge">
   <p>This is rounded border with orange</p>
   </div>
</body>
</html>
```

The output of the code given in Example 15 as shown in Figure 4.15.

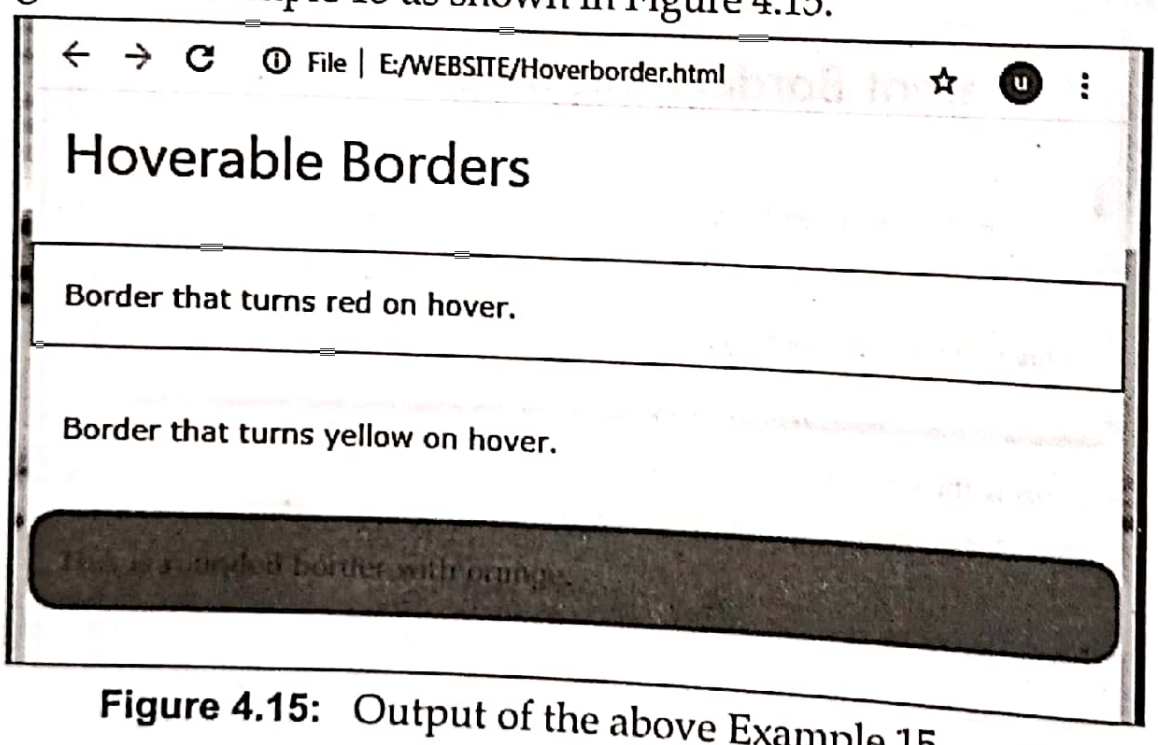


Figure 4.15: Output of the above Example 15

When a user hovers on the purple panel, the border color changes to green. Similarly, the second panel contains the rounded border effect where the border of the panel has been rounded via w3-border-xlarge class. You can also use small, or extra large classes as well to style to the rounded border.

You can use div style class of border thickness size. For example,

** **
<p>Thick white invisible border that hover the mouse turns green.

4.5.3 W3.CSS Fonts

W3.CSS fonts have different classes that are used to style font size and font styles. There are eight classes that can be used to style w3.CSS fonts size. There are w3-tiny, small, medium, large, xlarge, xxlarge, xxxlarge and jumbo. The default class is **w3-medium** that is **15px** and default font is **Verdana** and font spacing is **1.5**.

Scanned with OKEN Scanner

Example 16:

```
<!DOCTYPE html>
<html>
<title>W3.CSS </title>
<head>
<meta                    name="viewpoint"
content="width=device-width,  initial-
scale=1">
<link rel="stylesheet" href="w3.css">
</head>
<body>
  <div class="w3-container">
   <h2>Font-Size Classes</h2>
     <p class="w3-tiny">Tiny text</p>
   <p class="w3-small">Small text</p>
   <p          class="w3-Medium">Medium
text</p>
   <p class="w3-large">Large text</p>
   <p     class="w3-xlarge">X     Large
text</p>
   <p    class="w3-xxlarge">XX    Large
text</p>
   <p   class="w3-xxxlarge">xxx   Large
text</p>
   <p class="w3-jumbo">Jumbo text</p>
</div>
</body>
</html>
```

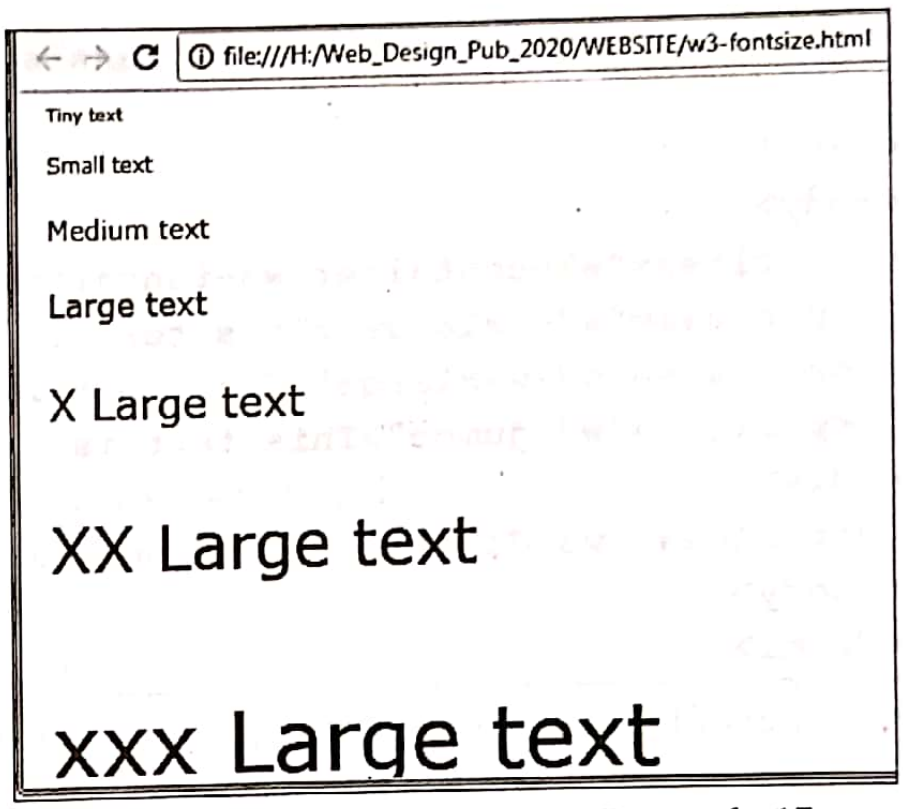


Figure 4.16: Output of the above Example 15

The output of the code given in Example 16 as shown in Figure 4.16.

4.5.4 W3.CSS External Fonts

Apart from available fonts, you can also add fonts from external source such as Google fonts to webpage. To add link to the external stylesheet for the font that you want to add. Give the font name in the header section and use it in the program.

Example 17:

```
<!DOCTYPE html>
<html>
<head>
<title>My First Program in W3.CSS</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet" type="text/css" href="/website/Nature/w3.css">
                                                    rel="stylesheet"
<link
href="https://fonts.googleapis.com/css?family=Tangerine">
```

(Contd...)

Scanned with OKEN Scanner

```
<link     rel="stylesheet"     href="https://fonts.googleapis.com/css?family-
Josefin+sans">
<style>
.w3-tangerine {
  font-family: "Tangerine", serif;
}
.w3-josefin {
 font-family: 'Josefin Sans', sans-serif;
 }
</style>
<body>
<div class="w3-container w3-tangerine">
  <p class="w3-xxlarge">This text is tangerine font with xxlarge size.</p>
  <p class="w3-xxxlarge">This text is tangerine font with xxxlarge size.</p>
  <p class="w3-jumbo">This text is tangerine font with jumbo size.</p>
</div>
<div class-"w3-Josefin w3-xlarage">This is text with Josefin font.</div>
</body>
</html>
```

The output of the code given in Example 17 as shown in Figure 4.17.

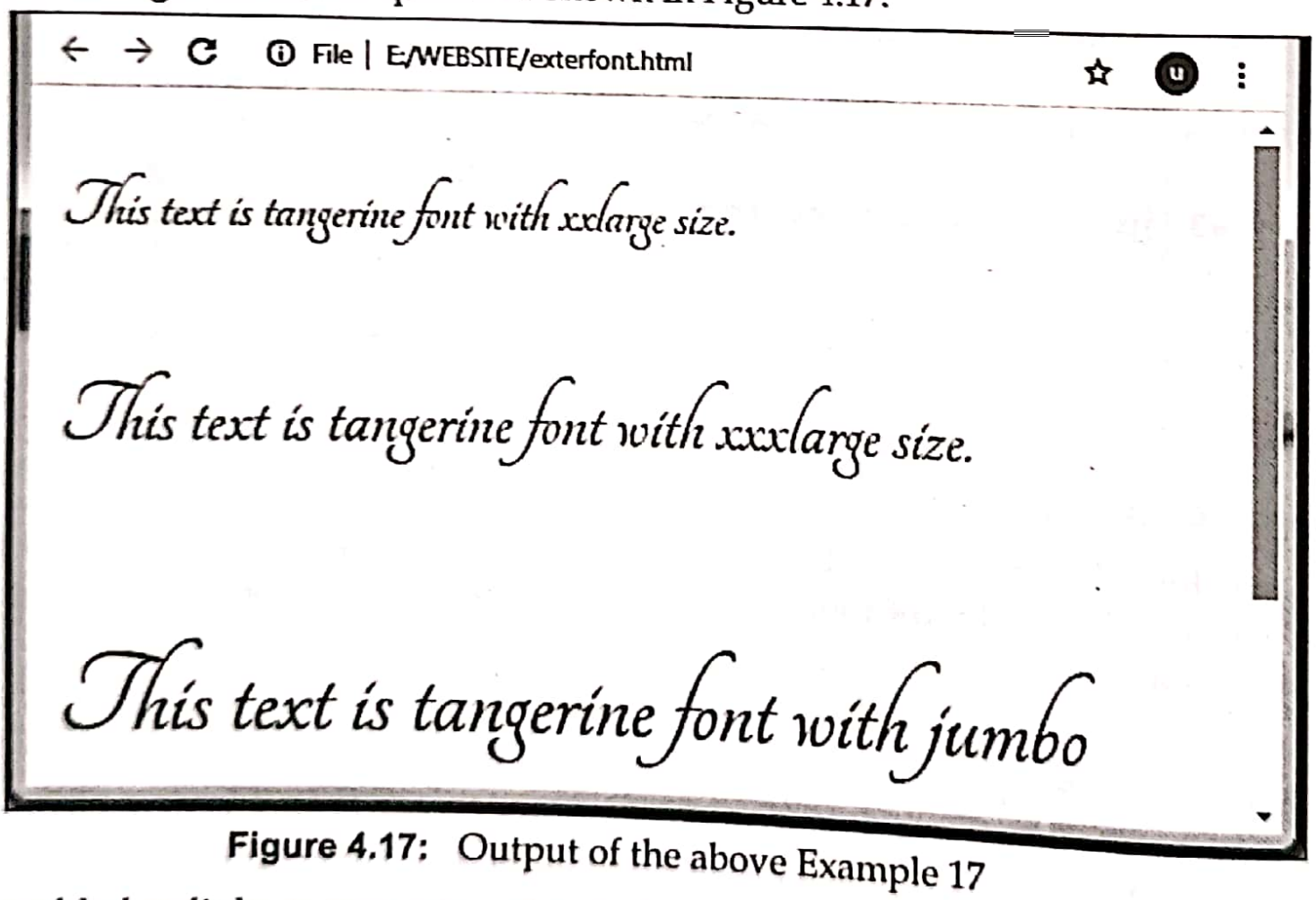


Figure 4.17: Output of the above Example 17

Here, you have added to links to two external stylesheets for two different fonts. First is the **tangerine** and **josefin**. The first class sets the font family to **tangerine** font while the second class sets the font family to **josefin sans**. Inside the body, you created two divs.

4.6 W3.CSS Text

W3.CSS text framework contains many classes that are used to style text and add text effects. These classes can align text, center text, add shadows to the text and change its opacity. Let us discuss these classes.

Scanned with OKEN Scanner

4.6.1 W3.CSS Text Alignment

Three classes are used to align text within a container element. The w3-left align aligns the content to left, right aligns the content to right and center aligns centers the content.

Example 18:

```
<!DOCTYPE html>
<html>
<title>W3.CSS</title>
<head>
<meta name="viewpoint" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="w3.css">
</head>
<body>
  <div class="w3-container w3-center">
    <h2>Centered Content</h2>
     <img src="H:/Blur.jpg" alt="Hills" style="width:80%,max-width:320px">
     <p>The w3-center class centers an element</p>
</div>
   <div class="w3-container">
   <h2>Text Alignment</h2>
   <div class="w3-container w3-border w3-large">
   <div class="w3-left-align">This is left aligned.</div>
   <div class="w3-center">This is centered.</div>
   <div class="w3-right-align">This is right aligned.</div>
</body>
</html>
```

The output of the code given in Example 18 as shown in Figure 4.18.

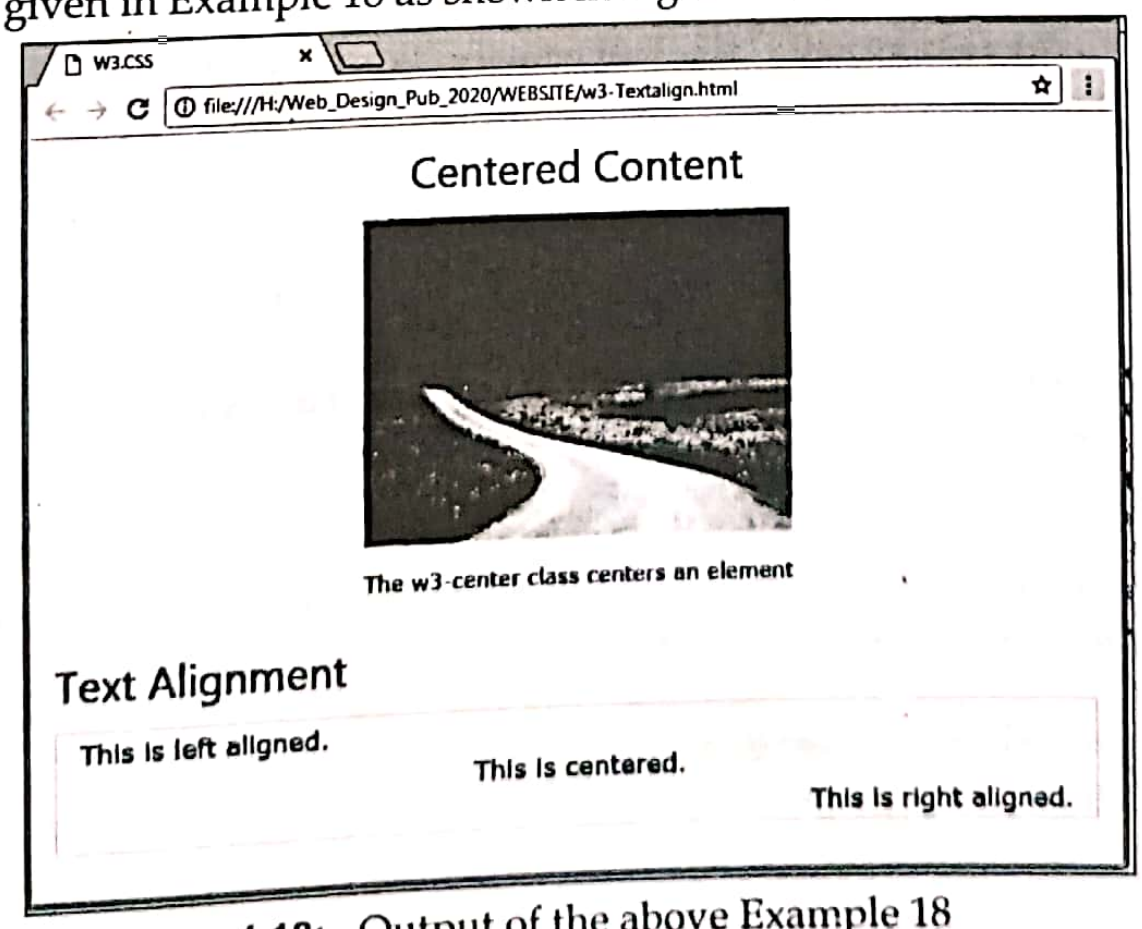


Figure 4.18: Output of the above Example 18

Scanned with OKEN Scanner

4.6.2 Wide and Slim Text

Wide text has additional spaces between each character while slim text has compressed space between each character. To create wide classes, use w3-wide and slim classes use w3-slim.

Example 19:

```
<!DOCTYPE html>
<html>
<title>W3.CSS</title>
<head>
<meta name="viewpoint" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="w3.css">
</head>
<body>
  <div class="w3-wide">Wide class between space each character</div>
  <div class="w3-slim">Slim class specifies compressed space</div>
</body>
</html>
```

The output of the code given in Example 19 as shown in Figure 4.19.

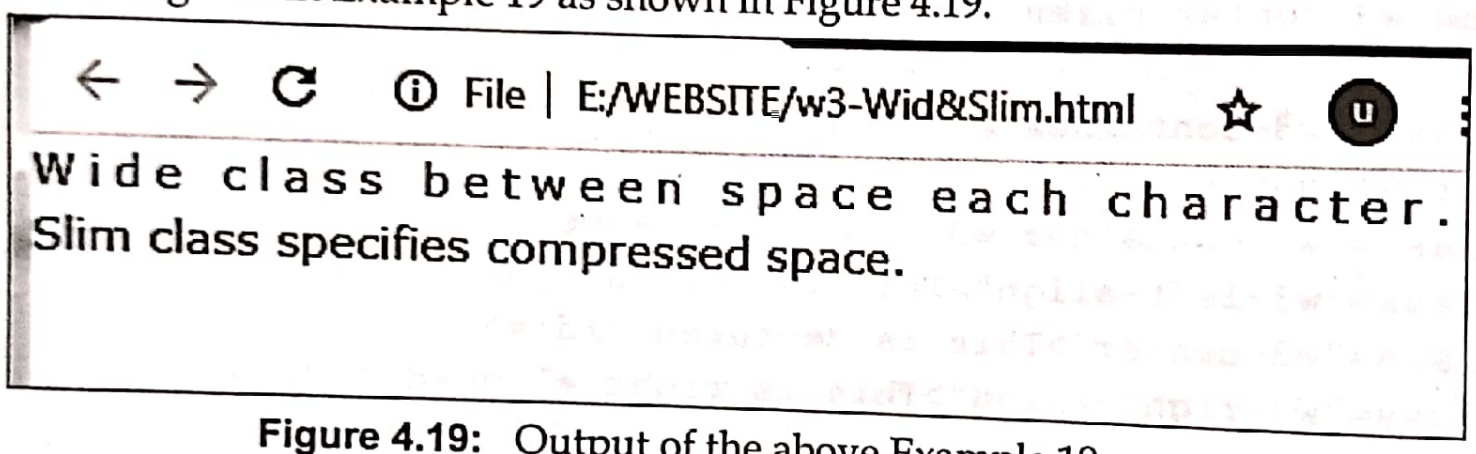


Figure 4.19: Output of the above Example 19

4.6.3 Shadow and Opacity

You can also add shadow to the text through w3-text-shadow class or blur effects to text. However, you can add opacity to the text through w3-opacity class. Text opacity class is design to work with all colors.

Example 20:

```
<!DOCTYPE html>
<html>
<title>W3.CSS</title>
<head>
<meta name="viewpoint" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="w3.css">
</head>
<body>
  <div class="w3-panel w3-deep-orange w3-xxlarge">
  <h2 class="w3-text-shadow">Text Shadow</h2>
  </div>
  <div class="w3-panel w3-blue-grey w3-xxlarge">
   <h2 class="w3-opacity">Text opacity</h2>
</div>
</body>
</html>
```

Scanned with OKEN Scanner

The output of the code given in Example 20 as shown in Figure 4.20.

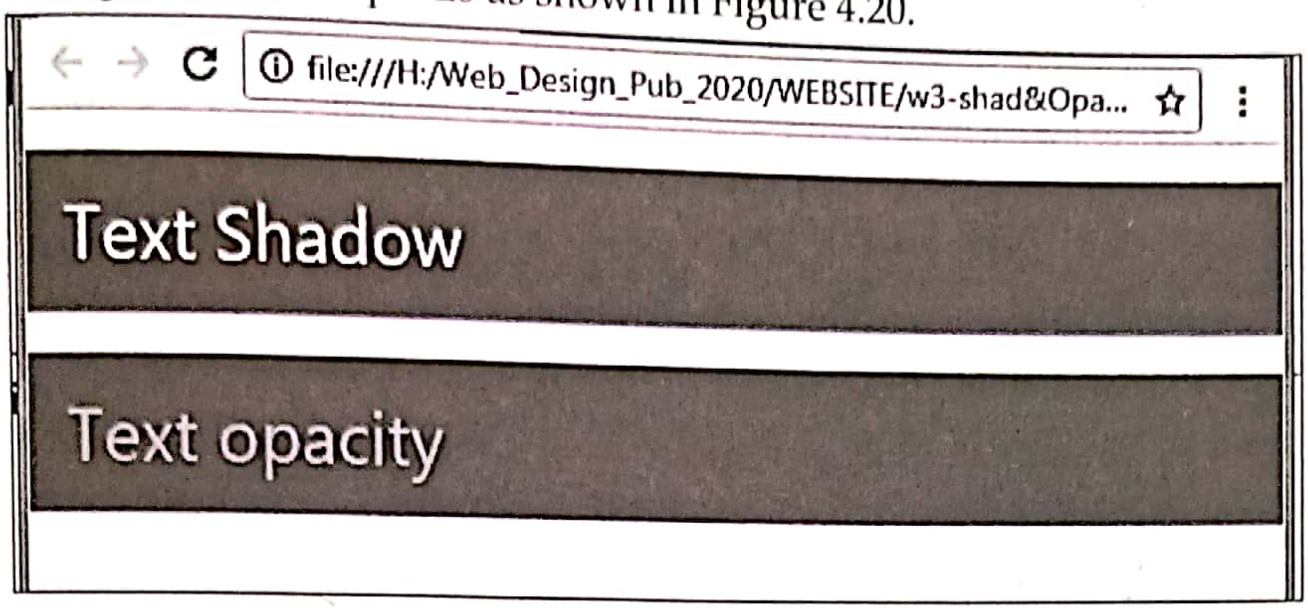


Figure 4.20: Output of the above Example 20

In the first panel, you have added a shadow effect to the text while in the second panel, you have added text with opacity.

4.7 W3.CSS Tables

Table was the most fundamental layout element. With the advent of divs, the role of tables as the fundamental layout elements. It can be used to display different types of tables using various styles over w3-table.

4.7.1 W3.CSS Table border Strips border

To create table aesthetics, you can use w3-striped class along with w3-table class to create a table whose alternate rows have given background color. You can also add w3-border class around the entire table. The w3-bordered class adds border across rows. Let us see the example of these classes have given below.

Example 21:

```
<!DOCTYPE html>
<html>
<title>W3.CSS</title>
<head>
<meta name="viewpoint" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="w3.css">
</head>
<h2>Bordered Striped Table</h2>
<body>
  <table class="w3-table w3-border w3-bordered w3-striped">
<tr>
<th>FirstName</th>  <th>LastName</th>  <th>Marks</th> </tr>
<td>Ashok</td> <td>Singal</td> <td>93</td> </tr>
<tr> <td>John</td> <td>William</td> <td>87</td>  </tr>
<tr>
<td>Harish</td> <td>Jamesh</td> <td>43</td> </tr>
<tr>
<td>Ritika</td> <td>Wason</td> <td>83</td> </tr>
</table>
</body>
</html>
```

Scanned with OKEN Scanner

The output of the code given in Example 21 as shown in Figure 4.21.

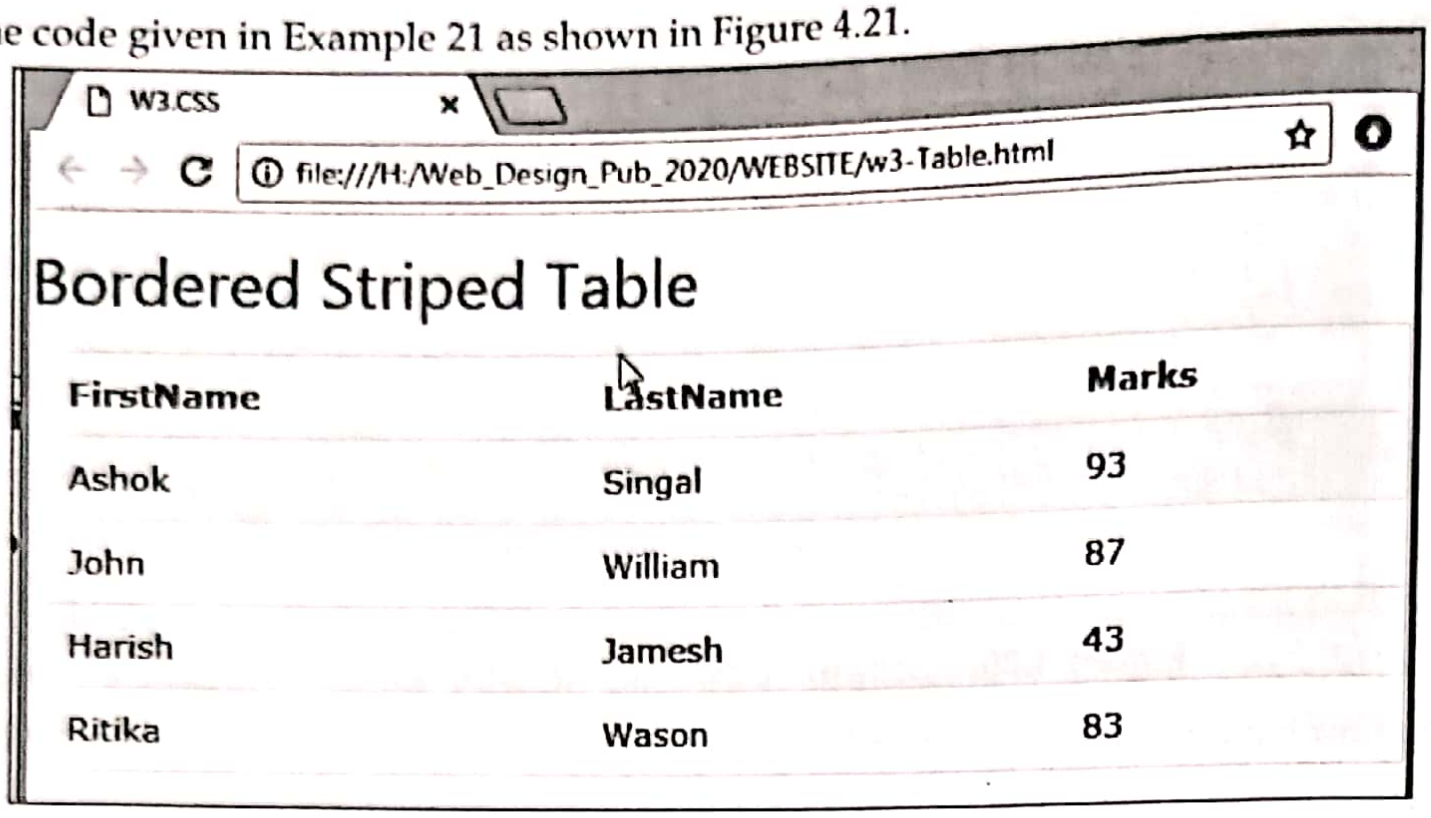


Figure 4.21: Output of the above Example 21

4.7.2 Table with a Colored Heading

To add color in the heading, use w3-color classes to the table. The following examples shows to add stripped with light-blue border with alternative rows in the table.

Example 22:

```
<!DOCTYPE html>
<html>
<title>W3.CSS</title>
<head>
<meta name="viewpoint" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="w3.css">
</head>
<h2>Colored Table heading</h2>
<body>
   <table class="w3-table w3-light-blue w3-border w3-striped">
<thead>
<tr class="w3-blue">
<th>FirstName</th> <th>LastName</th> <th>Marks</th> </tr>
</thead>
<td>Ashok</td> <td>Singal</td> <td>93</td> </tr>
<tr>
<td>John</td> <td>William</td> <td>87</td> </tr>
<tr>
<td>Harish</td> <td>Jamesh</td> <td>43</td> </tr>
<tr>
<td>Ritika</td> <td>Wason</td> <td>83</td> </tr>
</table>
</body>
</html>
```

Scanned with OKEN Scanner

The output of the code given in Example 22 as shown in Figure 4.22.

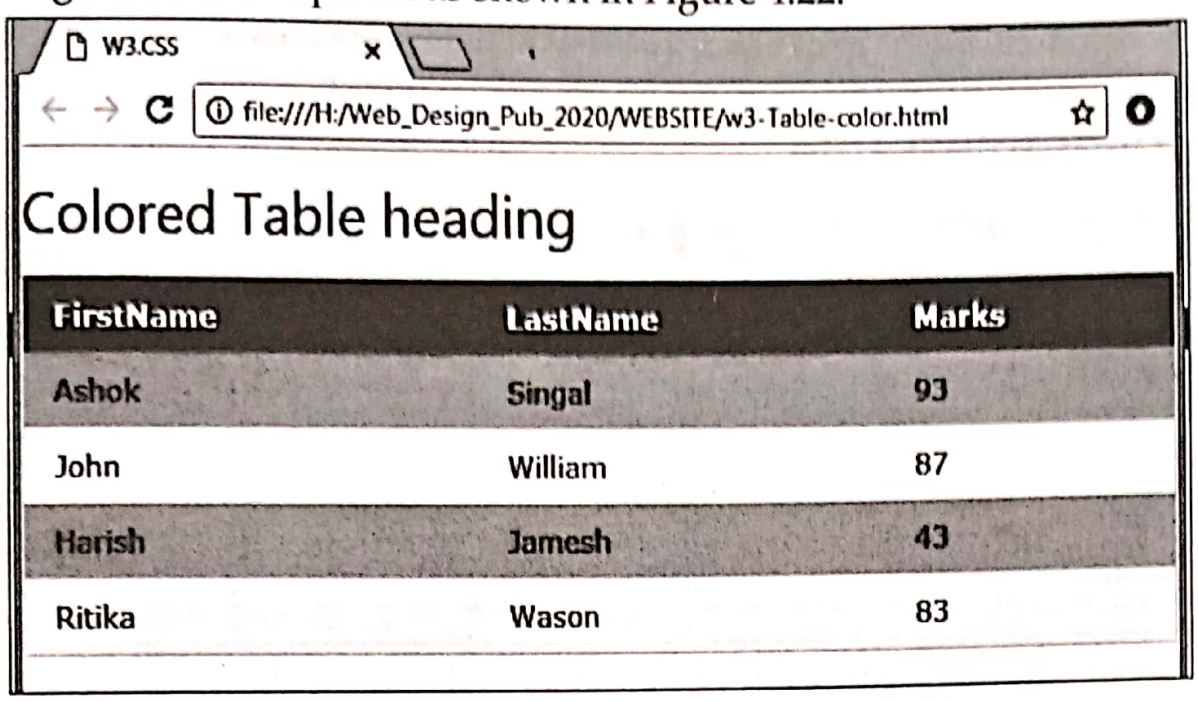


Figure 4.22: Output of the above Example 22

4.7.3 Hoverable Tables

To enhance visual aesthetics of a table, use w3-hoverable class. This class adds an effect to the table, which changes the table row color upon mouse hover. Let us look the following code:

Example 23:

```
<!DOCTYPE html>
<html>
<title>W3.CSS</title>
<head>
<meta name="viewpoint" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="w3.css">
</head>
<h2>Hoverable Table</h2>
<body>
   <table class="w3-table w3-bordered w3-border w3-striped">
<thead>
<tr class="w3-blue w3-hover-green">
<th>FirstName</th> <th>LastName</th><th>Marks</th> </tr>
</thead>
<tr class="w3-hover-red">
<td>Ashok</td> <td>Singal</td> <td>93</td> </tr>
<tr class="w3-hover-yellow">
<td>John</td> <td>William</td> <td>87</td> </tr>
<tr class="w3-hover-purple">
<td>Harish</td> <td>Jamesh</td> <td>43</td> </tr>
<tr class="w3-hover-orange">
<td>Ritika</td> <td>Wason</td> <td>83</td> </tr>
</table>
</body>
</html>
```

Scanned with OKEN Scanner

The output of the code given in Example 23 as shown in Figure 4.23.

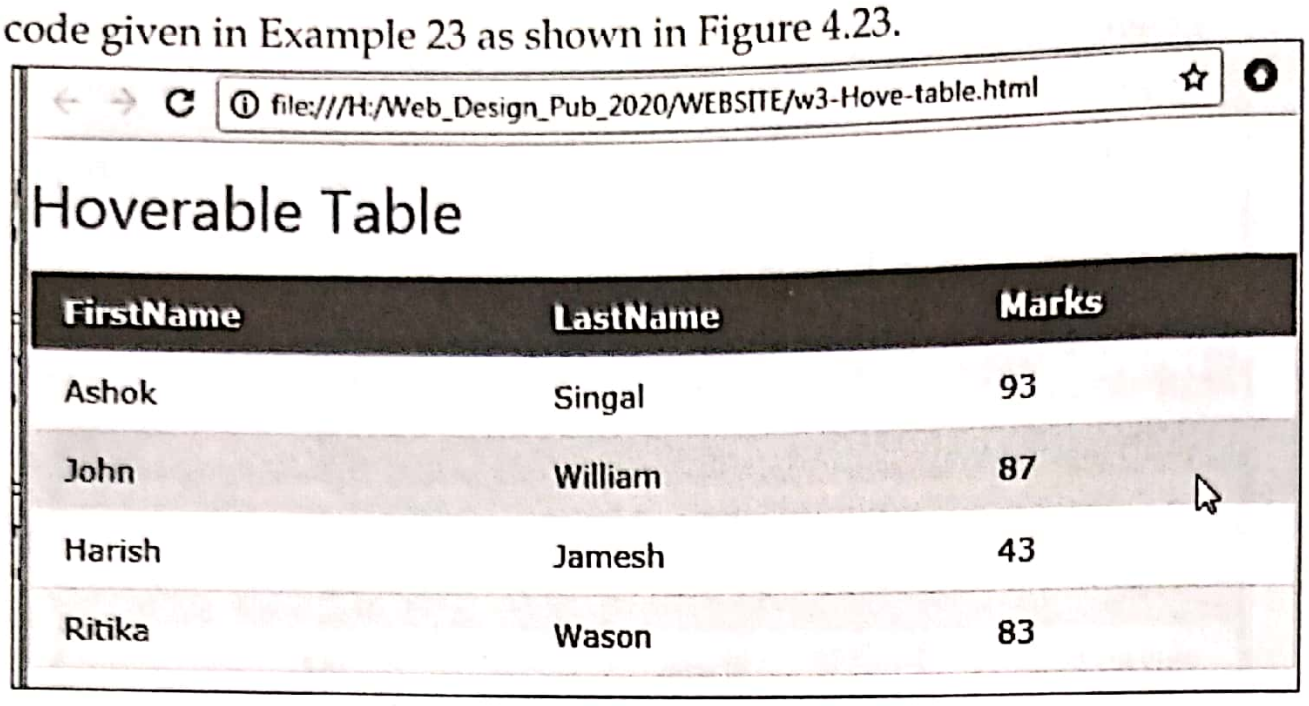


Figure 4.23: Output of the above Example 23

When you move the mouse over the rows, you can see the rows have different hover color to that rows.

4.7.4 Table as Card

The w-card classes make the table look like card.

Example 24:

```
<!DOCTYPE html>
<html>
<title>W3.CSS</title>
<head>
<meta name="viewpoint" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="w3.css">
</head>
<h2>Table as a Card</h2>
<body>
   <table class="w3-table w3-striped w3-bordered w3-card-4">
<thead>
<tr class="w3-orange">
<th>FirstName</th> <th>LastName</th> <th>Marks</th> </tr>
</thead>
<tr> <td>Ashok</td> <td>Singal</td> <td>93</td> </tr>
<tr> <td>John</td> <td>William</td> <td>87</td> </tr>
<tr> <td>Harish</td> <td>Jamesh</td> <td>43</td> </tr>
<tr> <td>Ritika</td> <td>Wason</td> <td>83</td> </tr>
</table>
</body>
</html>
```

The output of the code given in Example 24 as shown in Figure 4.24.

Scanned with OKEN Scanner

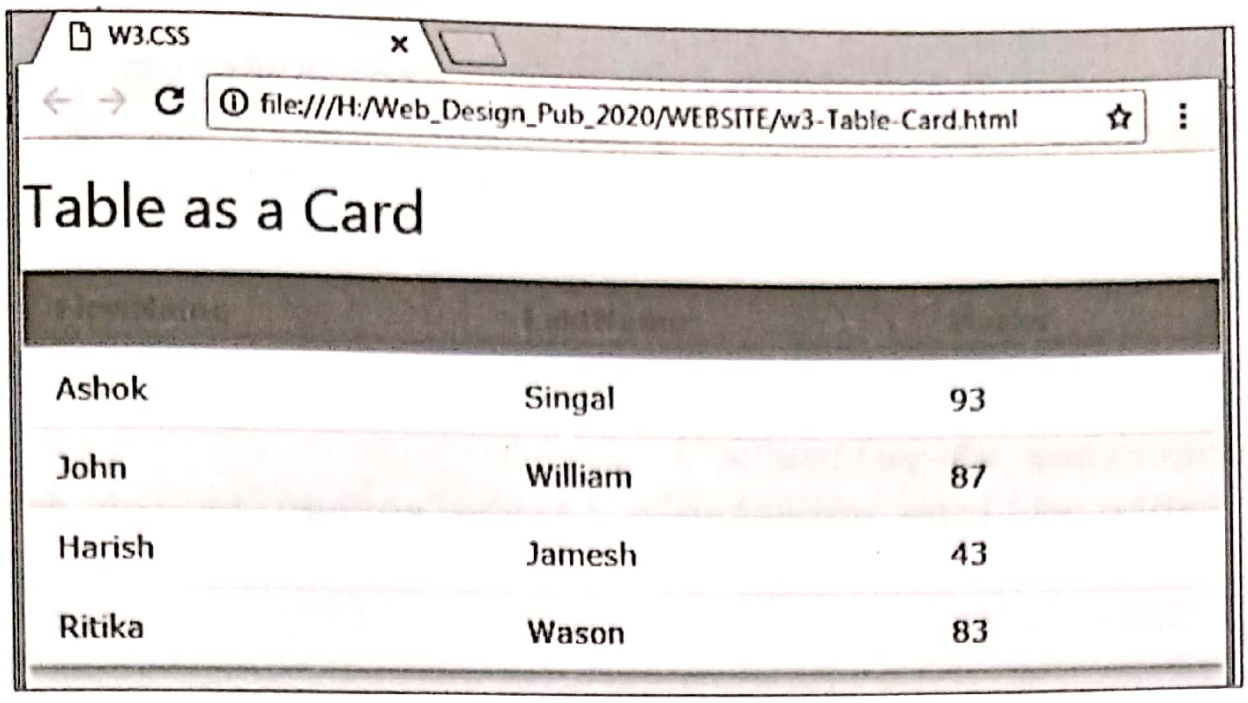


Figure 4.24: Output of the above Example 24

4.7.5 Responsive Table

A responsive table will display a horizontal scrollbar if the screen in too small to display the full content.

Example 25:

```
<!DOCTYPE html>
<html>
<title>W3.CSS</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="w3.css">
<body class="w3-container">
<h2>Responsive Table</h2>
<div class="w3-responsive">
<table class="w3-table w3-striped w3-bordered w3-border">
<tr>
  <th>Company  Name</th>       <th>quarterly  Sales</th>       <th>Sales</th>
<th>PBDIT</th>
  <th>NetProfit</th>       <th>NetWorth</th>       <th>EPS</th>       <th>Total
Debt</th>
  <th>Total share cap</th>   <th>Total Assets</th>   <th>Dividend</th> </tr>
<tr>
  <td>ABC</td>       <td>Year  2017</td>       <td>2978</td>       <td>2.64</td>
<td>377</td>
  <td>17,257</td>       <td>29</td>       <td>23482</td>       <td>85.48</td>
<td>1138999</td>
  <td>200%</td>   </tr>
<tr>
  <td>XYZ</td>       <td>Year  2018</td>       <td>4799</td>       <td>4.25</td>
<td>508</td>
  <td>16,898</td>       <td>15</td>       <td>20894</td>       <td>73.23</td>
<td>1899898</td>
  <td>30%</td> </tr>
```

(Contd...)

Scanned with OKEN Scanner

```
<tr>
  <td>YYY</td>        <td>Year  2019</td>        <td>5898</td>        <td>1.90</td>
<td>277</td>   <td>6000</td>
  <td>25</td>         <td>1500</td>        <td>90.23</td>        <td>2898349</td>
<td>50%</td> </tr>
</table>
</div>
<br>
<div class="w3-container w3-yellow">
<p>A responsive table will be scrollable if the screen is too small</p>
</div>
</body>
</html>
```

The output of the code given in Example 25 as shown in Figure 4.25.

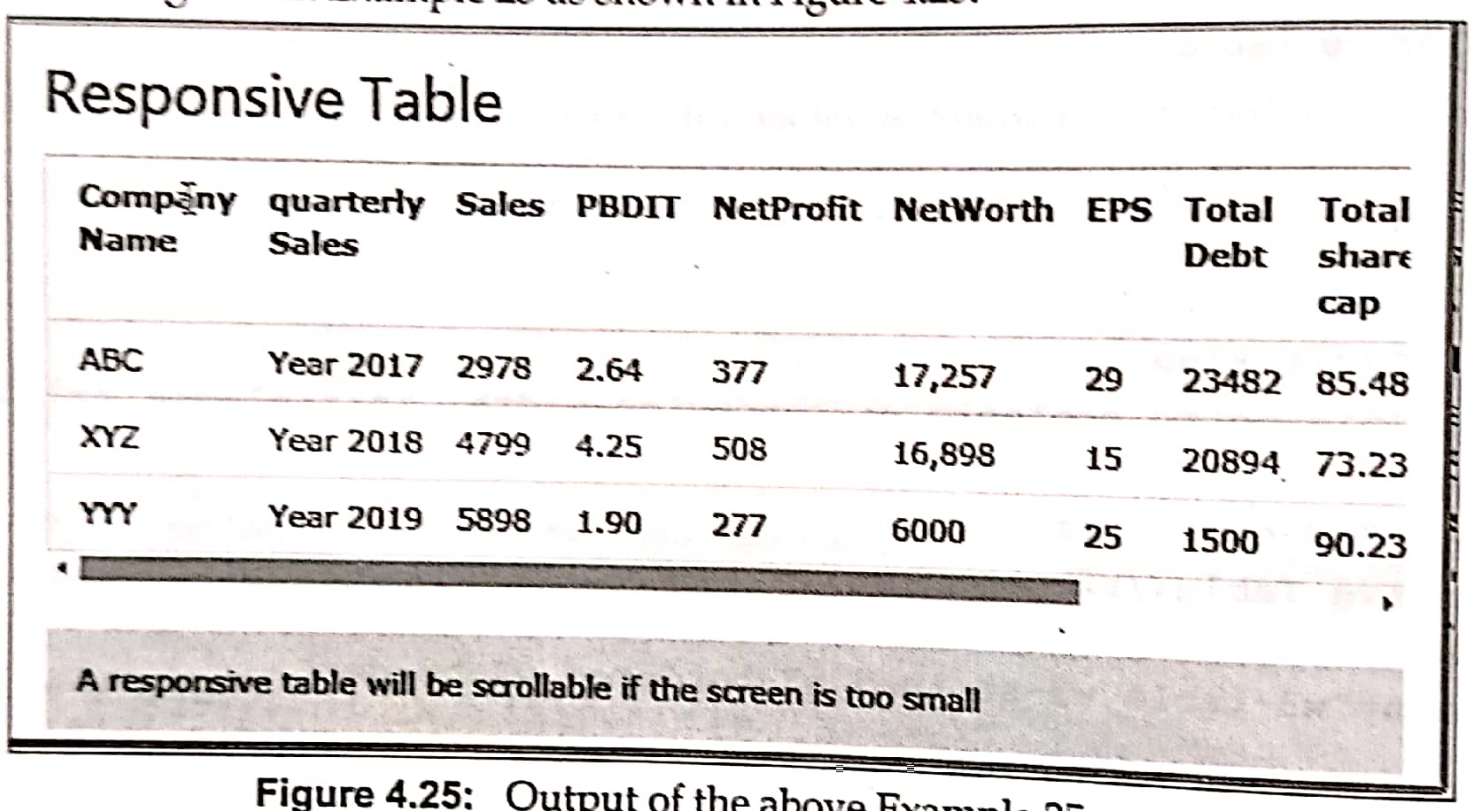


Figure 4.25: Output of the above Example 25

4.7.6 W3.CSS XXLarge

To draw a table with extra-large font use w3-xxlarge class.

Example 26:

```
<!DOCTYPE html>
<html>
<title>W3.CSS</title>
<head>
<meta name="viewpoint" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="w3.css">
</head>
<body class="w3-container">
<br>
<table class="w3-table w3-striped w3-bordered w3-border w3-xxlarge">
```

(Contd...)

Scanned with OKEN Scanner

```
<tr>
<th>FirstName</th> <th>LastName</th> <th>Marks</th> </tr>
<td>Ashok</td> <td>Singal</td> <td>93</td> </tr>
<tr>
<td>John</td> <td>William</td> <td>87</td> </tr>
<tr>
<td>Harish</td> <td>Jamesh</td> <td>43</td> </tr>
<tr>
<td>Ritika</td> <td>Wason</td> <td>83</td> </tr>
</table>
</body>
</html>
```

The output of the code given in Example 26 as shown in Figure 4.26.

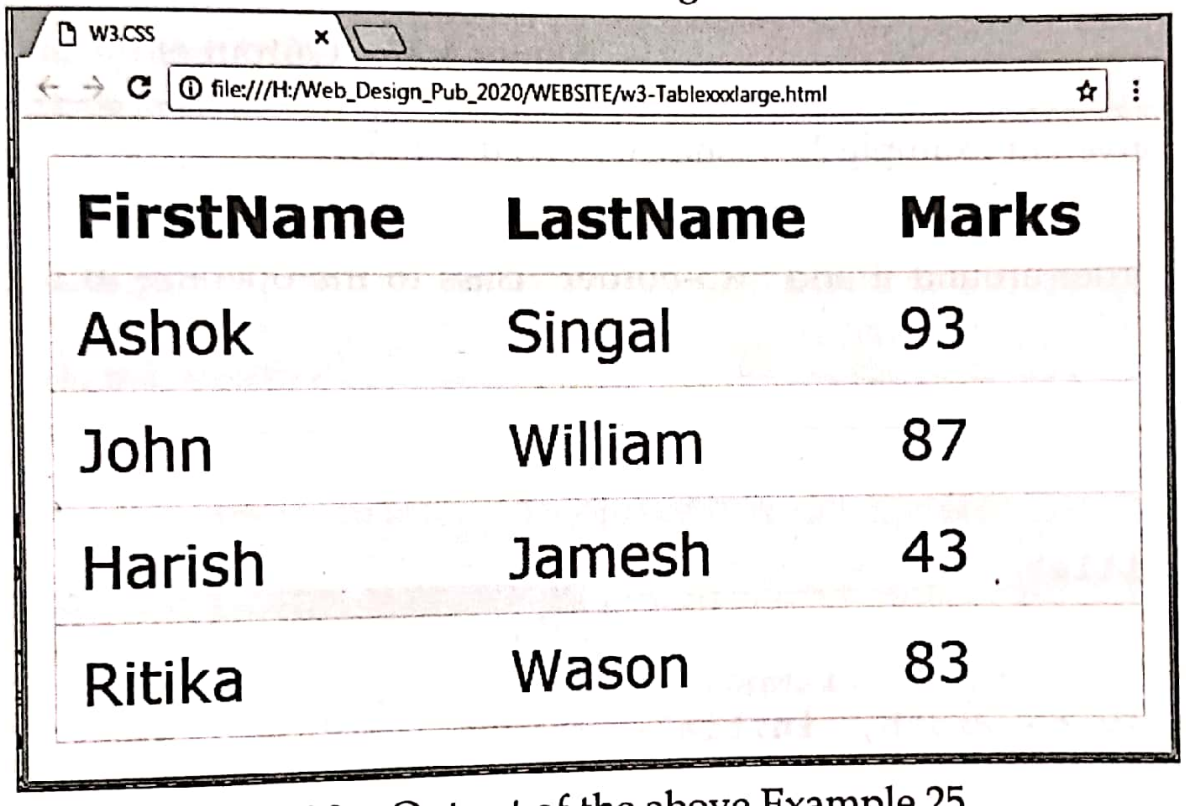


Figure 4.26: Output of the above Example 25

4.8 W3.CSS Lists

In the previous section, you saw how to create tables through W3.CSS and how to add different effects to these tables. Now, you shall cover another topic i.e. W3.CSS lists. Lists in html used to display data in the form of lists. There are used to create menu items and that is to be displayed in list form. There are two types of lists in html **ordered** and **unordered** lists. Let us see example an unordered list.

4.8.1 Basics WS.CSS List

To create W3CSS lists is by creating an ordered lists and then adding w3-list class to the opening "**ul**" tag. This creates a list where each item is stacked vertically upon each other with a lower border.

Example 27:

```
<!DOCTYPE html>
<html>
<title>W3.CSS</title>
<head>
```

(Contd...)

Scanned with OKEN Scanner

```
<meta                          name="viewpoint"
content="width=device-width, initial-
scale=1">
<link rel="stylesheet" href="w3.css">
</head>
<div class="w3-container">
<h2>Basics lists</h2>
<ul class="w3-ul">
 <li>English</li>          <li>Maths</li>
<li>Science</li>      <li>Computer</li>
<li>History</li> </ul>
</body>
</html>
```

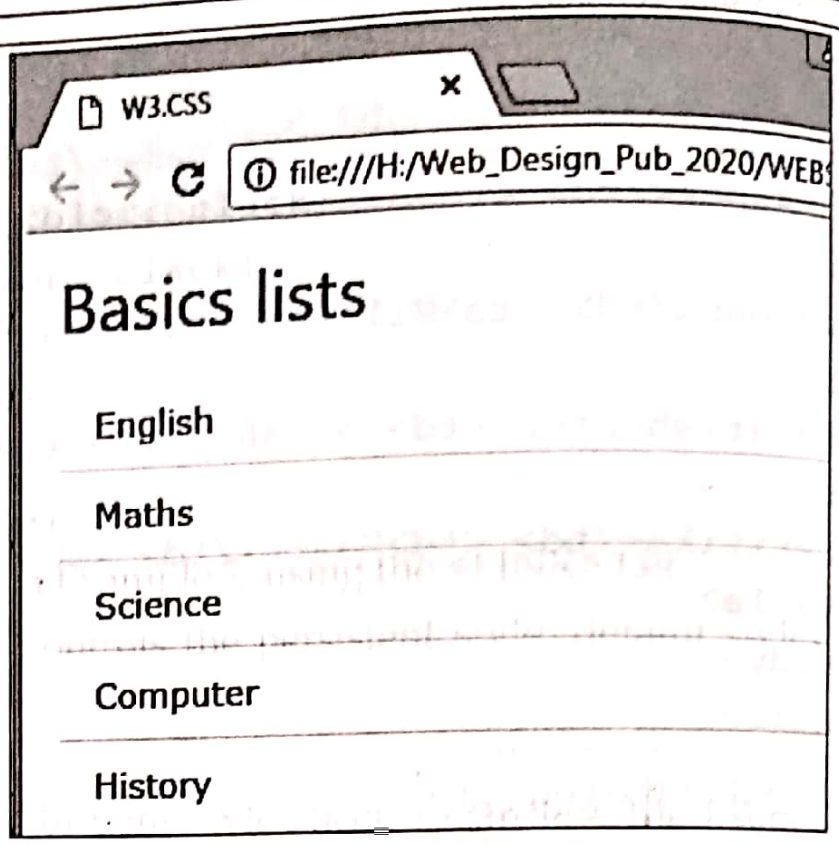


Figure 4.27: Output of the above Example 27

The output of the code given in Example 27 as shown in Figure 4.27.

Bordered List

To create a list with border around it add "**w3-border**" class to the opening **ul** tag. It creates an overall border around the list.

Example 28:

```
<!DOCTYPE html>
<html>
<title>W3.CSS</title>
<head>
<meta                          name="viewpoint"
content="width=device-width, initial-
scale=1">
<link rel="stylesheet" href="w3.css">
</head>
<div class="w3-container">
<h2>Bordered list</h2>
<ul class="w3-ul w3-border">
 <li>English</li>
 <li>Maths</li>
 <li>Science</li>
 <li>Computer</li>
 <li>History</li>
</ul>
</body>
</html>
```

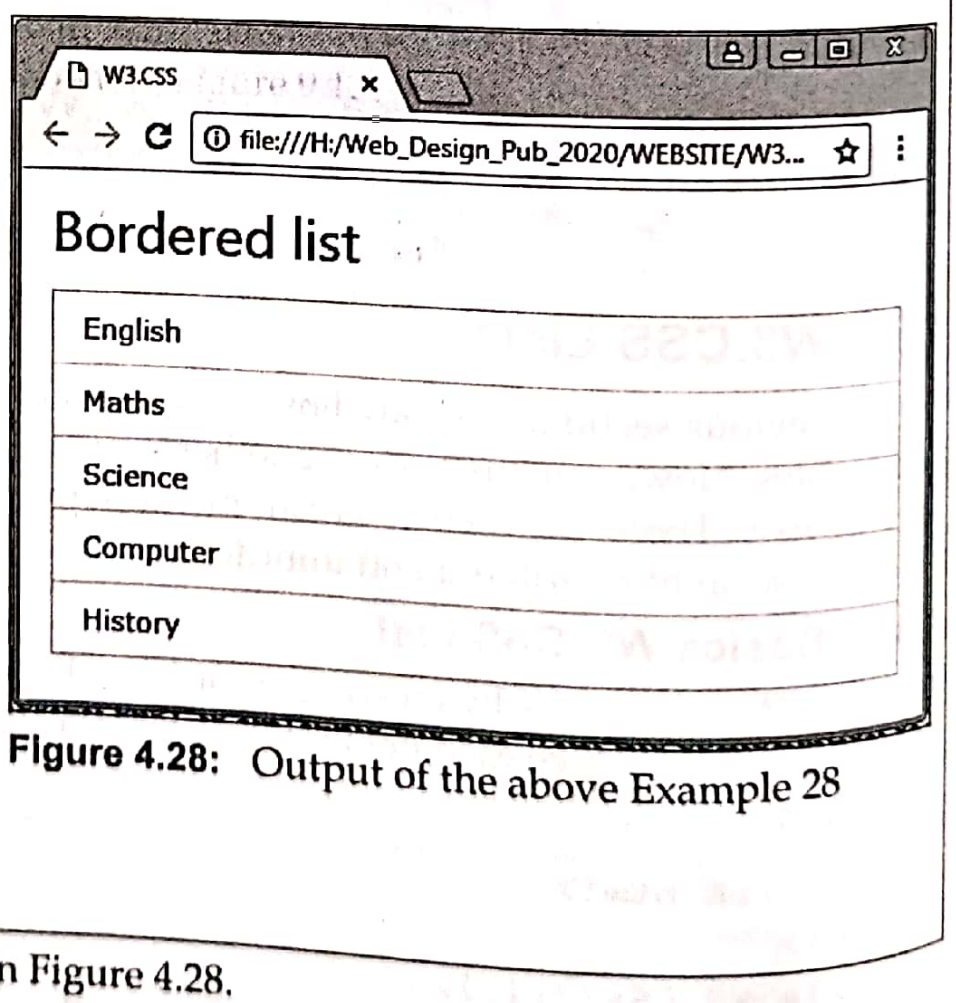


Figure 4.28: Output of the above Example 28

The output of the code given in Example 28 as shown in Figure 4.28.

4.8.2 Lists as Cards

To create lists as cards add "w3-cards-size" class to the opening "ul" tag.

Scanned with OKEN Scanner

Colored Lists

You can create colored lists by changing the background color of the whole table or individual item. To create background color of the complete list, add **w3-color** class to the opening "**ul**" tag. On the other hand, if you want to change color of the individual list item add w3-color class to the opening "li" tag of the individual item.

Example 29:

```
<!DOCTYPE html>
<html>
<title>W3.CSS</title>
<head>
<meta name="viewpoint" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="w3.css">
</head>
<div class="w3-container">
<h2>Colored List Item</h2>
<ul class="w3-ul w3-blue-grey">
 <li>Name</li>  <li>Address</li>  <li>PhoneNo</li> </ul>
<ul class="w3-ul w3-blue-grey">
 <li>Usha</li>  <li>Janakpuri west</li>  <li class="w3-pink">01124839898</li>
</ul>
<ul class="w3-ul w3-blue">
 <li>Rekha</li>          <li>Vikaspuri     west</li>          <li     class="w3-
pink">01145787878</li> </ul>
</body>
</html>
```

The output of the code given in Example 29 as shown in Figure 4.29.

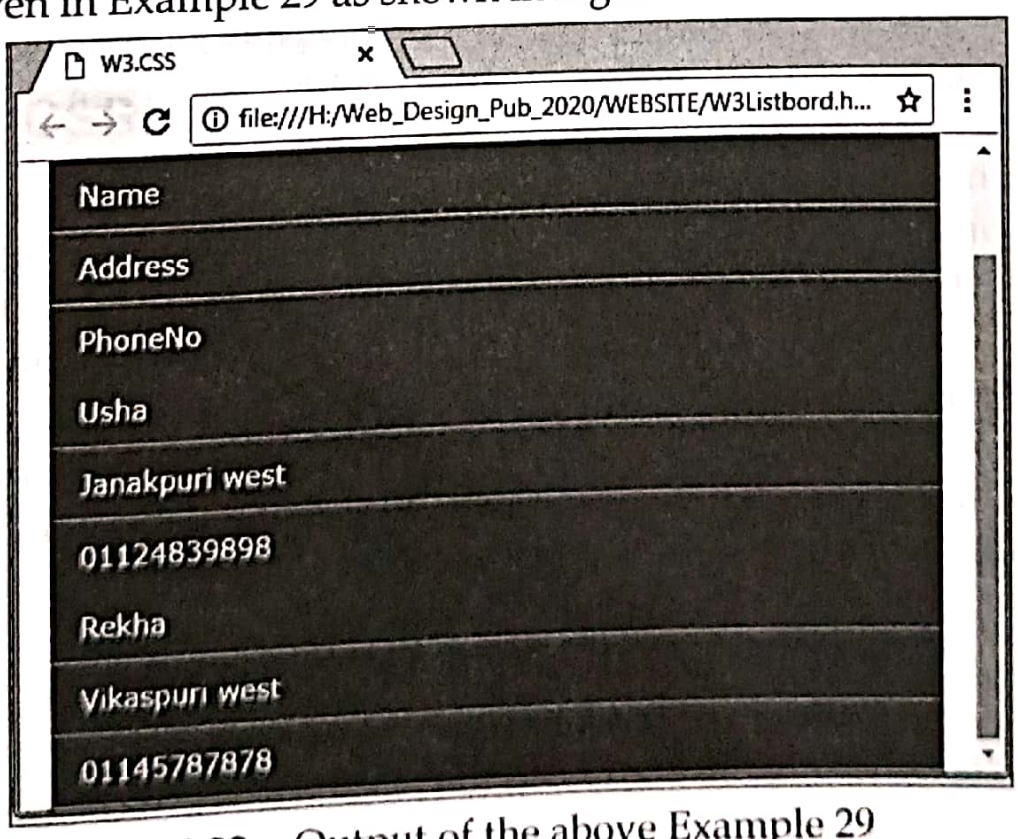


Figure 4.29: Output of the above Example 29

4.8.3 Hoverable Lists with Close Option

You can also create hoverable list that change color when hovered the list item. To add "**w3-hoverable**" class to the opening "**ul**" tag where as to add hover color to individual list item, add "**w3-hover-color**" class

to opening "li" tag. You need to create lists with close option against each item. It makes it simple for us. To add "**w3-display-container**" class to each opening "li" tag. After the list item, add span tag with a button that display a cross. Also, set the onclick property of the span tag so that it sets the display property of the parent tag as none. Let us create a code to understand this concept.

Example 30:

```
<!DOCTYPE html>
<html>
<title>W3.CSS</title>
<head>
<meta name="viewpoint" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="w3.css">
</head>
<div class="w3-container">
<h2>Closing List</h2>
<p>To close/hide the list item, click on the x icon</p>
<ul class="w3-ul w3-card-4" style="width:60%">
 <li>Name <span onclick="this.parentElement.style.display='none'" class="w3-
button w3-transparent w3-dispaly-right">X</span></li>
 <li>Address<span onclick="this.parentElement.style.display='none'" class="w3-
button w3-transparent w3-dispaly-right">X</span></li>
 <li>PhoneNo<span onclick="this.parentElement.style.display='none'" class="w3-
button w3-transparent w3-dispaly-right">X</span></li>
 <li>Usha <span onclick="this.parentElement.style.display='none'" class="w3-
button w3-transparent w3-dispaly-right">X</span></li>
 <li>Janakpuri west <span onclick="this.parentElement.style.display='none'"
class="w3-button w3-transparent w3-dispaly-right">X</span></li>
 <li>01124839898 <span onclick="this.parentElement.style.display='none'"
class="w3-button w3-transparent w3-dispaly-right">X</span></li>
</ul>
</body>
</html>
```

The output of the code given in Example 30 as shown in Figure 4.30.

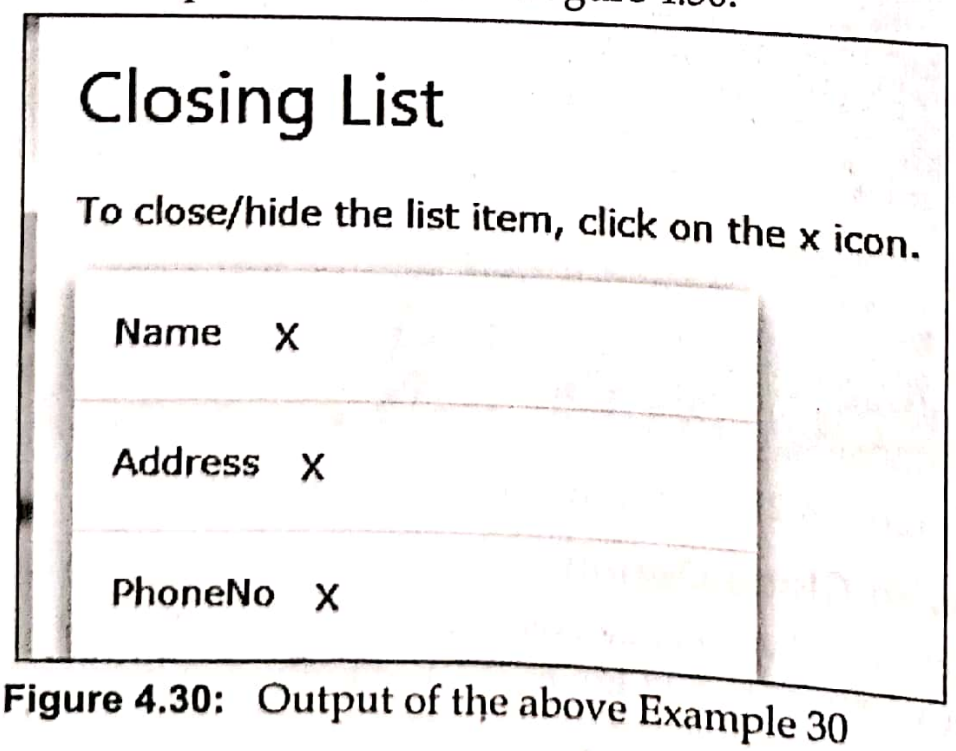


Figure 4.30: Output of the above Example 30

Scanned with OKEN Scanner

Padded List

To create padding list use "w3-padding" class to increase or decrease the space between rows list item. To add "w3-padding" class to each opening "li" tag with padding size.

For example,

```
<ul class="w3-ul w3-border" style="width:50%">
 <li class="w3-padding-8">Name</li>
</ul>
```

4.8.4 W3.CSS List Sizes

You saw list of card types, colored lists, hoverable lists and so on. You can also change list sizes. There are multiple classes, which can be used to create lists of different sizes such as tiny, small, large lists, extra-large lists and jumbo lists.

For example, you have create different types of large lists. To use simply append, "**w3-large**" class to the opening "**ul**" tag of the list.

Example 31:

```
<!DOCTYPE html>
<html>
<title>W3.CSS</title>
<head>
<meta name="viewpoint" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="w3.css">
</head>
<h2>Different types of Large Lists</h2>
<ul class="w3-ul w3-xlarge">
 <li>English</li>   </ul>
<ul class="w3-ul w3-xxlarge">
 <li>English</li>   </ul>
<ul class="w3-ul w3-xxxlarge">
 <li>English</li>   </ul>
<ul class="w3-ul w3-jumbo">
 <li>English</li>   </ul>
</body>
</html>
```

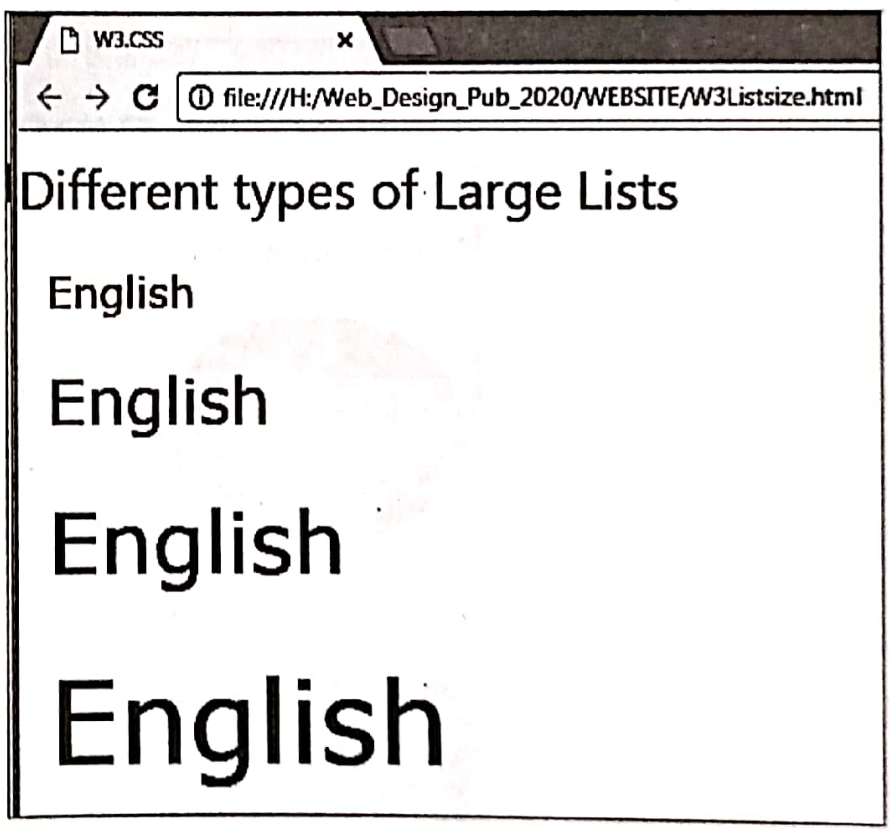


Figure 4.31: Output of the above Example 31

The output of the code given in Example 30 as shown in Figure 4.29.

4.9 W3.CSS Images

Images are one of the most important buildings blocks of websites. Today, you would rarely find a website without an image. To add images to a website "**img**" tag is used. However, W3.CSS images come with classes that can easily change the image looks. You can create images with rounded bordered, circle images, bordered images and images within a card etc. For example, you have created different types of images in the following code.

Scanned with OKEN Scanner

Example 32:

```
<!DOCTYPE html>
<html>
<title>W3.CSS</title>
<head>
<meta name="viewpoint" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="w3.css">
</head>
<h2>Different types of Image</h2>
<p>Image in a Circle</p>
<img src="h:/Blur.jpg" class="w3-circle" alt="Hills" style="width:30%">
<p>Image in a Rounded</p>
<img src="h:/Blur.jpg" class="w3-round" alt="Hills" style="width:30%">
<p>Bordered Image</p>
<img src="h:/Blur.jpg" class="w3-border w3-padding w3-padding-16" alt="Hills"
style="width:30%">
</body>
</html>
```

The output of the code given in Example 32 as shown in Figure 4.32.

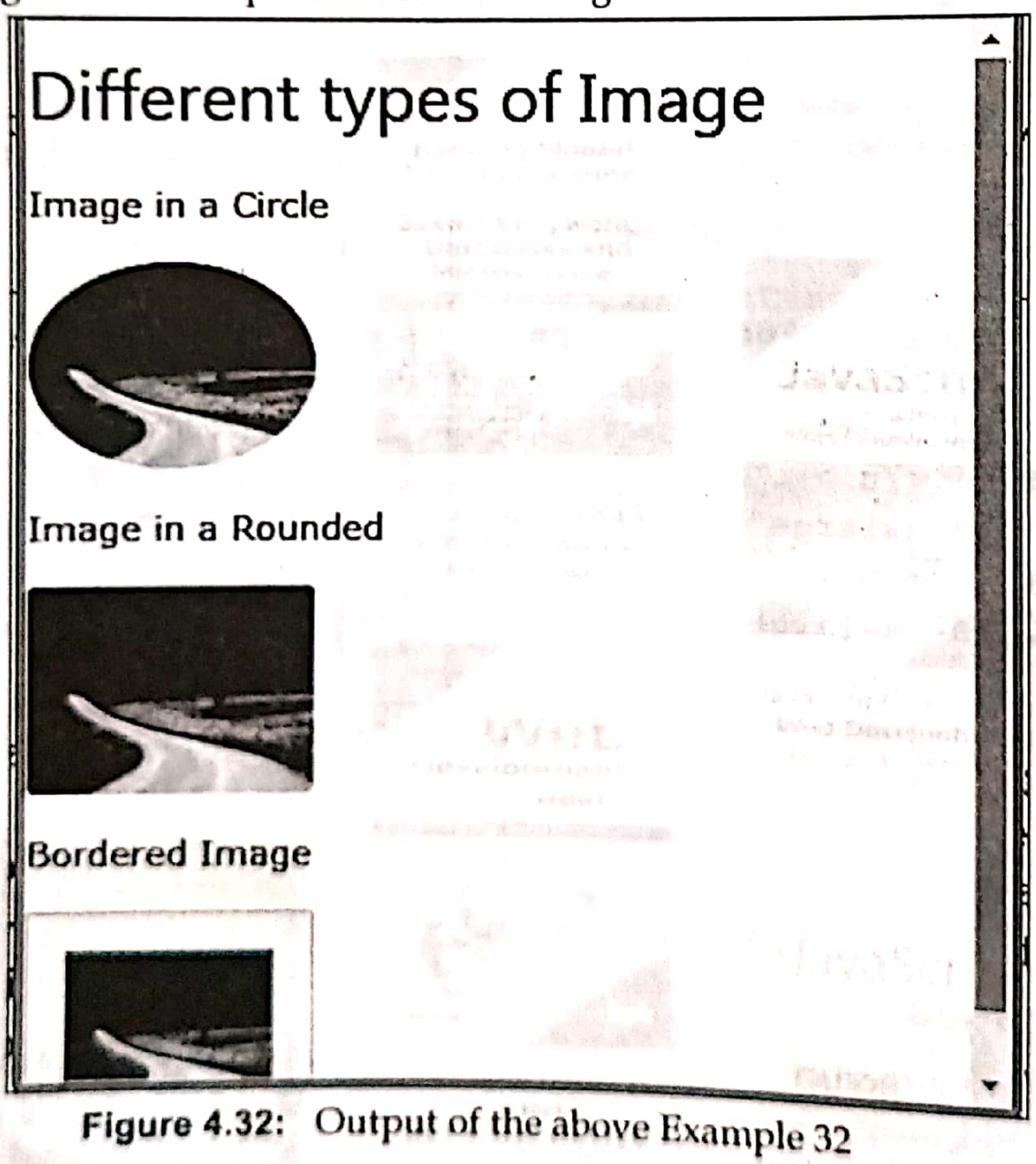


Figure 4.32: Output of the above Example 32

4.9.1 Images as Cards

You can also create images as card with a drop shadow at the bottom. To create a div and add class "w3-card-number" to the div, inside that div, add the markup for your image.

Scanned with OKEN Scanner

Example 33:

```
<!DOCTYPE html>
<html>
<title>W3.CSS</title>
<head>
<meta name="viewpoint" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="w3.css">
</head>
<h2>Image as a Card</h2>
<div class="w3-card-4" style="width:50%">
<img scr="h:/ducky1.jpg" alt="Duck" style="width:100%">
<div class="w3-container">
<h4><b>Duck</b></h4>
</div>
</div>
</body>
</html>
```

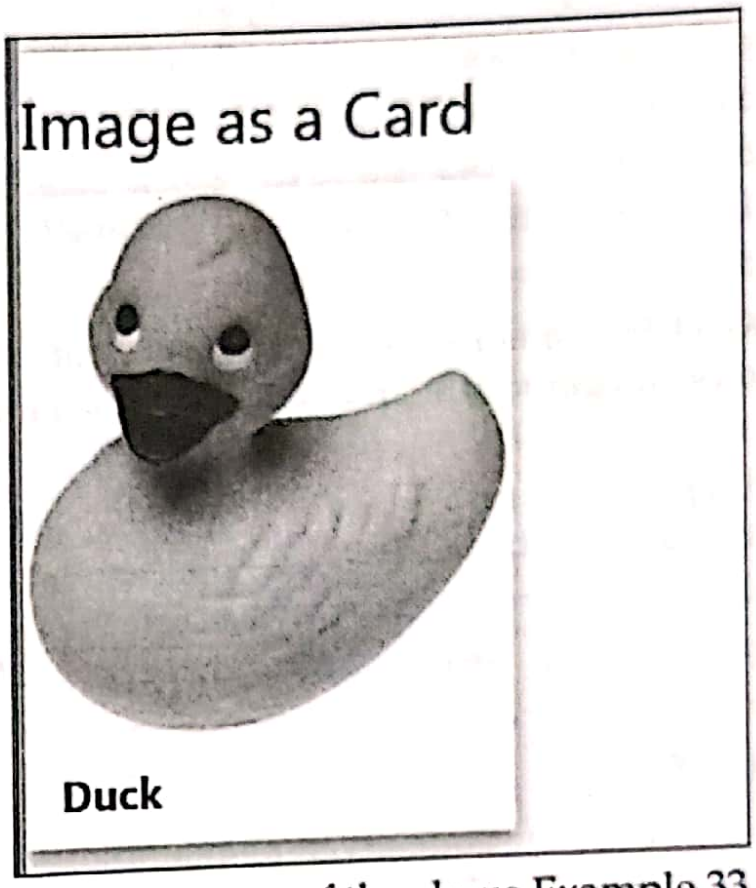


Figure 4.33: Output of the above Example 33

The output of the code given in Example 33 as shown in Figure 4.33.

4.10 W3.CSS Image Effects

You saw how to add rounded corners to image, how to circle an image, how to border an image and how to create card like images. Now, you are going to create some of the fancy image effects that you can add to your images. These effects include opacity effects, grayscale effects sepia effects and hover effects. It is difficult to create these effects without W3.CSS and it require lots of extensive JavaScript coding.

4.10.1 Adding Opacity to Images

To add opacity to the image to create a certain effect. W3.CSS uses w3-opacity, w3-opactiy-min and max classes to add opacity to an image. Let us create the following example where these three opacities have been added to an image.

Example 34:

```
<!DOCTYPE html>
<html>
<title>W3.CSS</title>
<head>
<meta name="viewpoint" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="w3.css">
</head>
<h2>Different types of opactiy</h2>
<div class="w3-row-padding s3-margin-top">
<div class="w3-third">
<div class-"w3-card-2">
```

(Contd....)

Scanned with OKEN Scanner

```
<img      src="h:/flower.jpg"          alt="Duck"      class="w3-opacity-min"
style="width:60%">
<div class="w3-container">
<p>Opacity set to minimum</p> </div> </div> </div>
<div class="w3-third">
<div class-"w3-card-2">
<img src="h:/flower.jpg"  alt="Duck" class="w3-opacity" style="width:60%">
<div class="w3-container">
<p>Opacity</p> </div> </div> </div>
<div class="w3-third">
<div class-"w3-card-2">
<img      src="h:/flower.jpg"          alt="Duck"      class="w3-opacity-max"
style="width:60%">
<div class="w3-container">
<p>Opacity set to Maximum</p> </div> </div> </div> </div>
</body>
</html>
```

The output of the code given in Example 34 as shown in Figure 4.34.

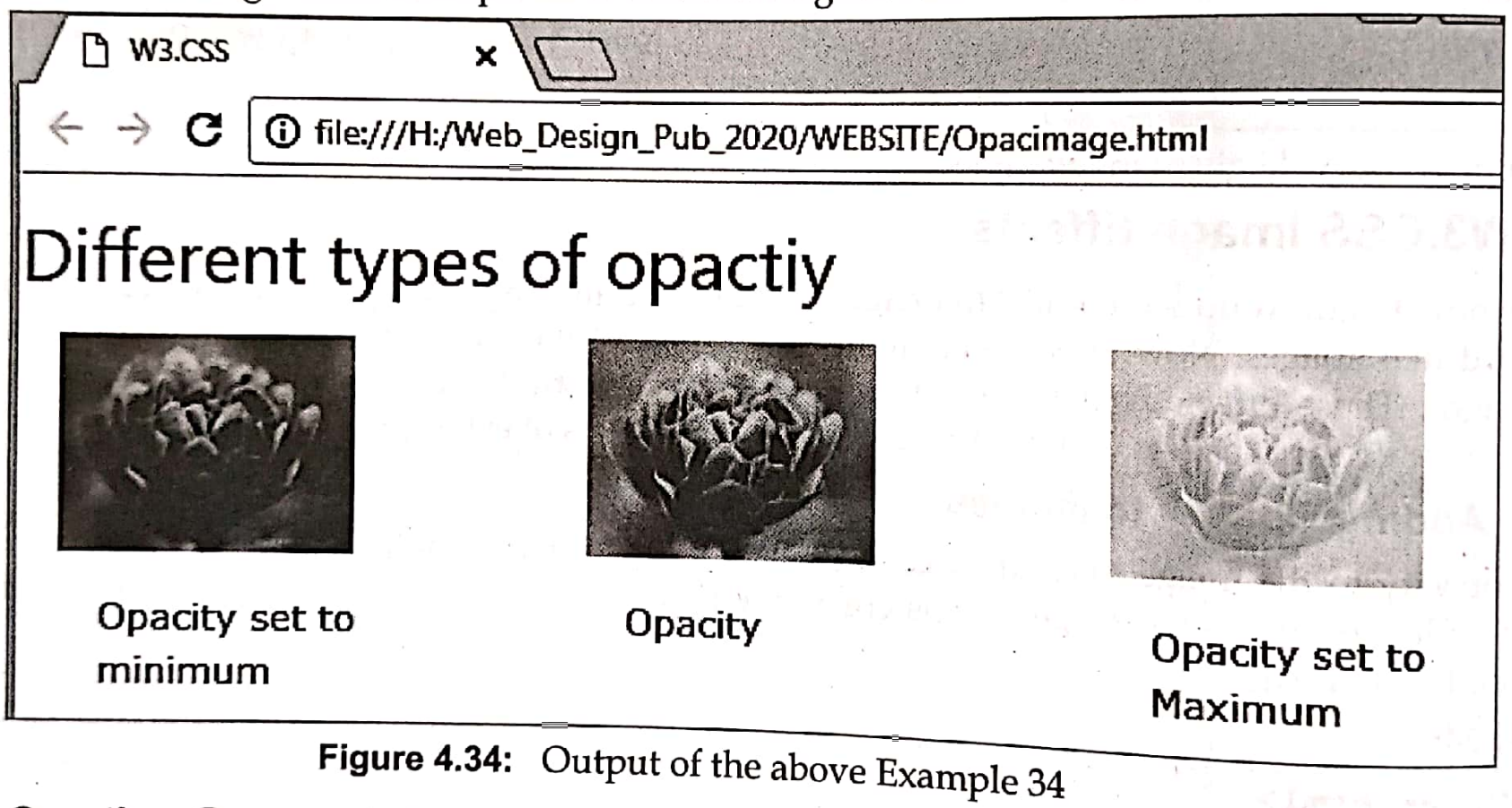


Figure 4.34: Output of the above Example 34

4.10.2 Creating Grayscale Images

To create grayscale images use w3-grayscale, w3-grayscale-min and max classes to create grayscale images. Let us try yourself as grayscale image.

For example,

```
<img src="filename path/filename with extension" class="w3-grayscale-max">
```

4.10.3 Creating Sepia Image Effect

W3.CSS uses w3-sepia, w3-sepia-min and max classes to create sepia image effect. Let us create the following example where these three types of sepia effects to been created.

Scanned with OKEN Scanner

Example 35:

```
<!DOCTYPE html>
<html>
<title>W3.CSS</title>
<head>
<meta name="viewpoint" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="w3.css">
</head>
<h2>Different types of Sepia</h2>
<div class="w3-row-padding s3-margin-top">
<div class="w3-third">
<div class-"w3-card-2">
<img src="h:/flower.jpg"  alt="Duck" class="w3-sepia-min" style="width:60%">
<div class="w3-container">
<p>Sepia effect set to minimum</p> </div> </div> </div>
<div class="w3-third">
<div class-"w3-card-2">
<img src="h:/flower.jpg"  alt="Duck" class="w3-sepia" style="width:60%">
<div class="w3-container">
<p>Sepia effect</p> </div> </div> </div>
<div class="w3-third">
<div class-"w3-card-2">
<img src="h:/flower.jpg"  alt="Duck" class="w3-sepia-max" style="width:60%">
<div class="w3-container">
<p>Sepia effect set to Maximum</p> </div> </div> </div> </div>
</body>
</html>
```

The output of the code given in Example 35 as shown in Figure 4.35.

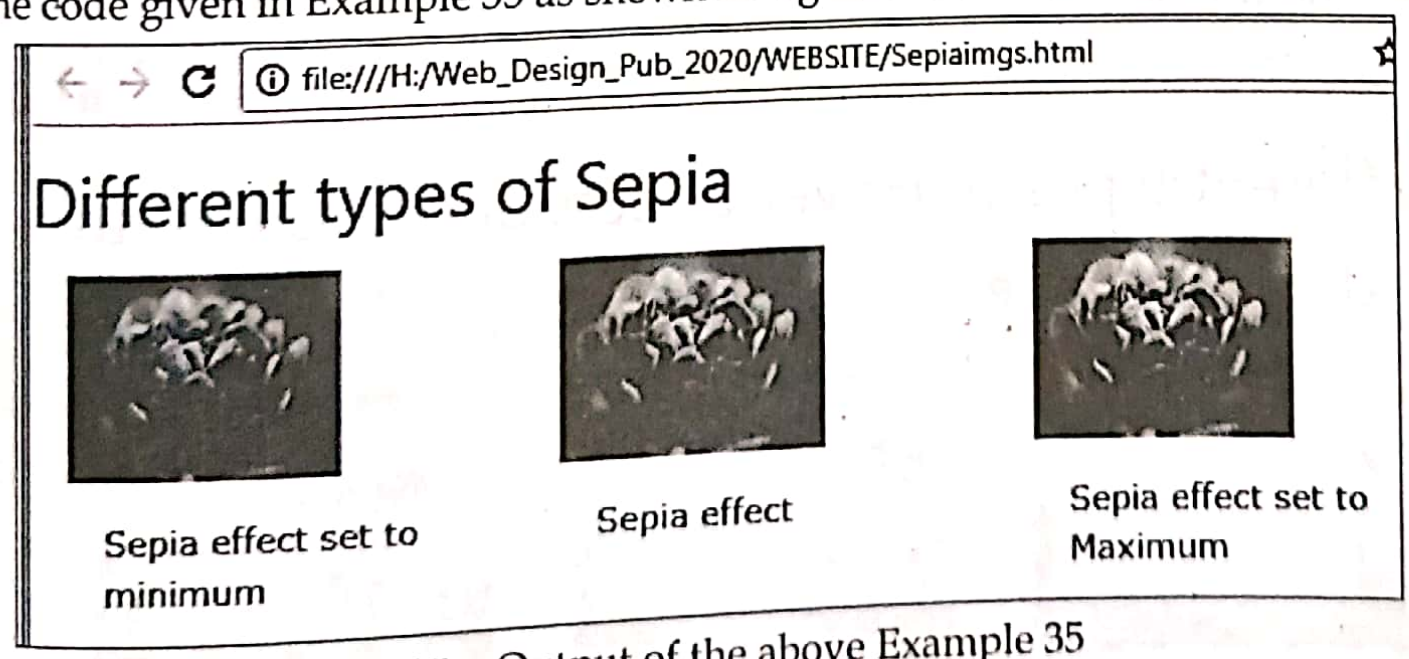


Figure 4.35: Output of the above Example 35

4.10.4 Creating Hover Effects

You can also create opacity, grayscale and sepia effect on images when mouse is hovered over the image. To do so, add w3-hover-grayscale, w3-hover-opacity, or w3-hover-sepia classes respectively to the images and also position the text in an image with the w3-display-classes.

Scanned with OKEN Scanner

Example 36:

```
<!DOCTYPE html>
<html>
<title>W3.CSS</title>
<head>
<meta name="viewpoint" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="w3.css">
</head>
<h2>Different types of Hover Effects and position the text in an image</h2>
<div class="w3-row-padding s3-margin-top">
<div class="w3-third">
<div class-"w3-card-2">
<div class="w3-display-container w3-text-red">
<img src="h:/Tree.jpg" alt="Duck" class="w3-hover-opacity" style="width:90%">
<div  class="w3-display-topleft  w3-container"><p>Top  Left</P></div>  </div>
</div> </div>
<div class="w3-third">
<div class-"w3-card-2">
<img src="h:/Tree.jpg"  alt="Duck" class="w3-hover-sepia" style="width:90%">
<div class="w3-display-container w3-text-blue">
<div  class="w3-display-bottomright  w3-container"><p>Bottom  right</P></div>
</div> </div> </div>
<div class="w3-third">
<div class-"w3-card-2">
<img src="h:/Tree.jpg" alt="Duck" class="w3-hover-grayscale"
     style="width:90%">
<div class="w3-container">
<p>Hover effect set to grayscale</p> </div> </div> </div> </div>
</body>
</html>
```

The output of the code given in Example 36 as shown in Figure 4.36.

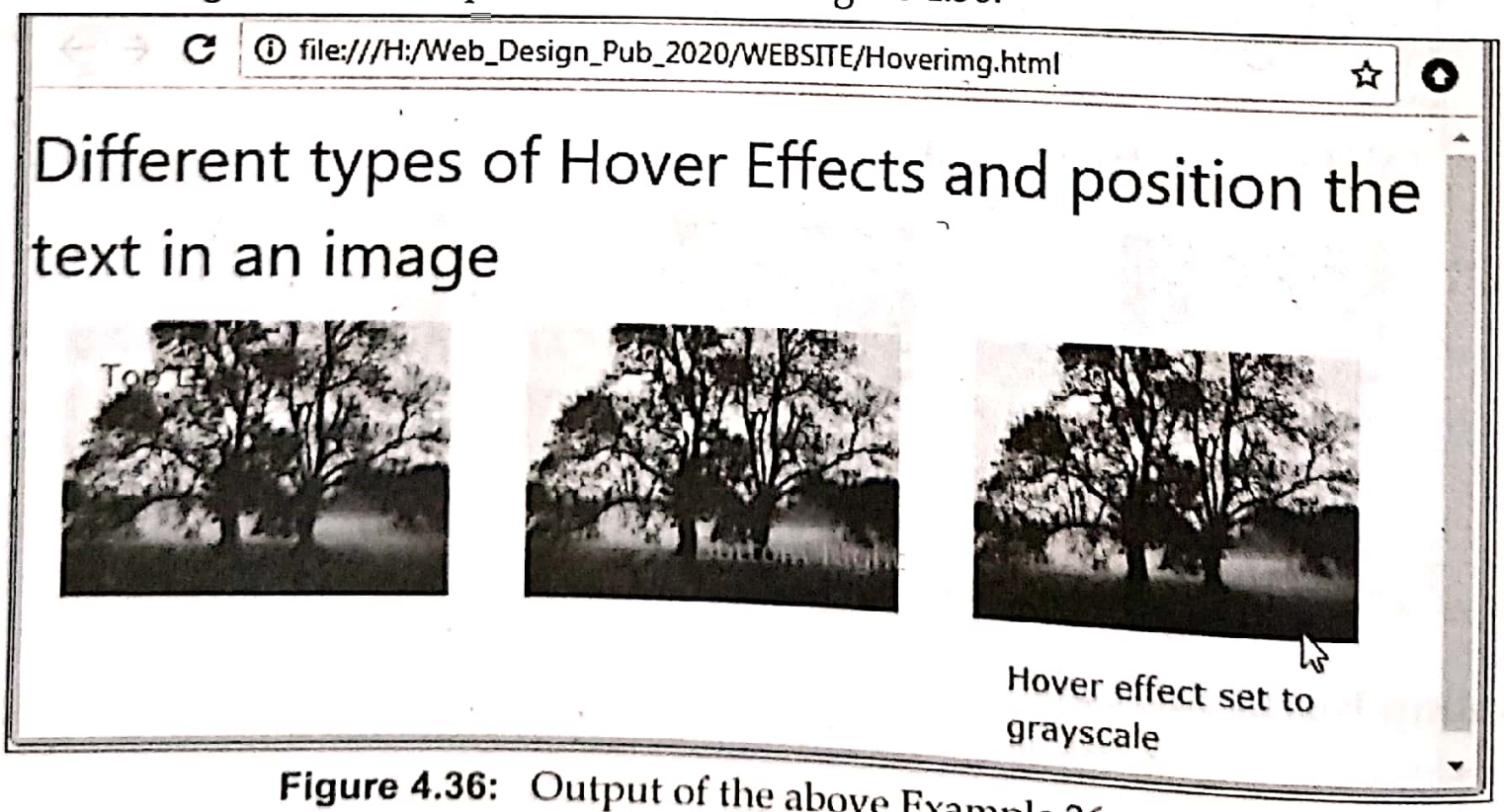


Figure 4.36: Output of the above Example 36

Scanned with OKEN Scanner

4.11 W3.CSS Grid

W3.CSS grid system targets three screen sizes: **small screens** such as mobiles, **medium screens** such as tablets and **large screens** such as desktop. A responsive layout adjusts according to changing screen size itself. A row is created using w3-row class. Inside each row, you can create maximum of 12 columns. A column is specified by w3-col class.

Column must reside inside a row to be fully responsive:

w3-row: It defines a padding-less container for responsive columns.

w3-row-padding: It defines a padded container for responsive columns.

w3-col: It defines a column with sub classes.

Columns for small screen (smart phones):

s1: It defines 1 of 12 columns for small screens that is width: 8.33%

s2: It defines 2 of 12 columns for small screens that is width: 16.66%

s3: It defines 3 of 12 columns for small screens that is width: 25.00%

s12: It define 12 of 12 columns for small screens that is width: 100%

Columns for medium screens (tablets):

m1: It defines 1 of 12 columns for medium screens that is width: 8.33%

m2: It defines 2 of 12 columns for medium screens that is width: 16.66%

m3: It defines 3 of 12 columns for medium screens that is width: 25.00%

m12: It defines 12 of 12 columns for medium screens.

Columns for Large screens (laptops):

l1: It defines 1 of 12 columns for large screens that is width: 8.33%

l2: It defines 2 of 12 columns for large screens that is width: 16.66%

l3: It defines 3 of 12 columns for large screens that is width: 25.00%

l12: It defines 12 of 12 columns for large screens.

If you need to specify the small class, it scales up. So if you wish to set the same width for small, medium and large screens. If you want the same width, you only need to specify the medium class for medium and large screens.

However, if you only use medium or large classes, the width will always transform to 100% on small screens.

To create a responsive grid, you need to create a div with class "**w3-row**". Inside the div, to create columns you need to create divs with class **w3-col**. In addition, for each individual screen size you use classes such as sx, mx and lx where s is small, m is for medium, l is for large an x is the number of columns. For example, to create two equal and unequal column grid.

4.11.1 Two equal and unequal columns

In this example, you have a row that has two further divs. The first div takes 6 columns on the small screen and the second div shall also take 6 columns on the small screen. Since you have not specified the column size for medium and large screen sizes. The column size for small screen will be applicable to them. Next, you again create a row that has two further div. The first div takes 4 columns on the small screen and the second div takes 8 columns on the small screen inside a container.

Scanned with OKEN Scanner

Example 37:

```
<!DOCTYPE html>
<html>
<title>W3.CSS</title>
<head>
<meta name="viewpoint" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="w3.css">
</head>
<div class="w3-row">
  <div class="w3-col s6 w3-blue">
 <p>This div covers 6 columns on all screens</p> </div>
<div class="w3-col s6 w3-green">
  <p>This div covers 6 columns on all screens</p> </div> br
<div class="w3-row w3-container">
 <div class="w3-col s8 w3-blue">
  <p>This div covers 8 columns on all screens</p>  </div>
 <div class="w3-col s4 w3-green">
  <p>This div covers 4 columns on all screens</p>  </div>
</body>
</html><!DOCTYPE html>
<html>
<title>W3.CSS</title>
<head>
<meta name="viewpoint" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="w3.css">
</head>
<div class="w3-row">
  <div class="w3-col s6 w3-blue">
 <p>This div covers 6 columns on all screens</p> </div>
<div class="w3-col s6 w3-green">
  <p>This div covers 6 columns on all screens</p> </div> br
<div class="w3-row w3-container">
 <div class="w3-col s8 w3-blue">
  <p>This div covers 8 columns on all screens</p>  </div>
 <div class="w3-col s4 w3-green">
  <p>This div covers 4 columns on all screens</p>  </div>
</body>
</html>
```

The output of the code given in Example 37 as shown in Figure 4.37.

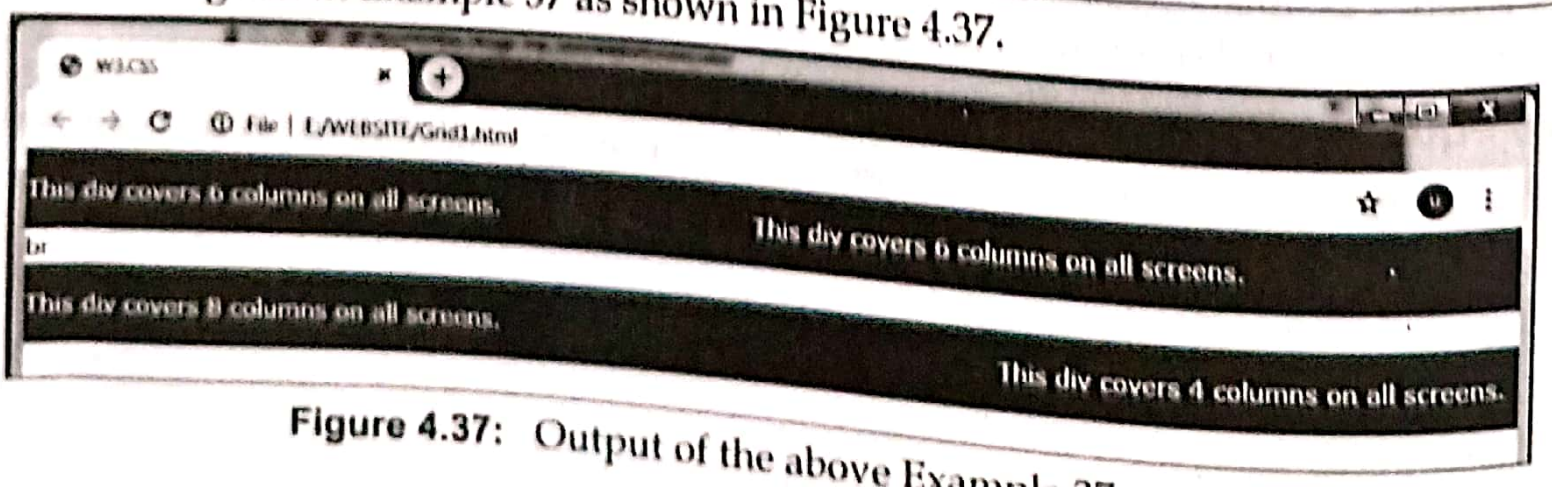


Figure 4.37: Output of the above Example 37

Scanned with OKEN Scanner

Six Columns

Six equal-width columns (16% each) on tablets, laptops and desktops. On mobile phones, the columns will automatically stack (100% width).

Example 38:

```
<!DOCTYPE html>
<html>
<title>W3.CSS</title>
<head>
<meta name="viewpoint" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="w3.css">
</head>
<div class="w3-row">
 <div class="w3-col m2 w3-blue">
 <p>Six equal width columns</p> </div>
<div class="w3-col m2 w3-green">
 <p>Six equal width columns</p> </div>
 <div class="w3-col m2 w3-red">
 <p>Six equal width columns</p>  </div>
 <div class="w3-col m2 w3-dark-grey">
 <p>Six equal width columns</p>  </div>
 <div class="w3-col m2 w3-purple">
 <p>Six equal width columns</p>  </div>
 <div class="w3-col m2 w3-brown">
 <p>Six equal width columns</p>  </div>
<div class="w3-row w3-container">
 <p>Inside a container</p>
 <div class="w3-col m2 w3-blue">
 <p>Six equal width columns</p> </div>
<div class="w3-col m2 w3-green">
 <p>Six equal width columns</p> </div>
 <div class="w3-col m2 w3-red">
 <p>Six equal width columns</p>  </div>
 <div class="w3-col m2 w3-dark-grey">
 <p>Six equal width columns</p>  </div>
 <div class="w3-col m2 w3-purple">
 <p>Six equal width columns</p>  </div>
 <div class="w3-col m2 w3-brown">
 <p>Six equal width columns</p>  </div>
</body>
</html>
```

The output of the code given in Example 38 as shown in Figure 4.38.

Scanned with OKEN Scanner

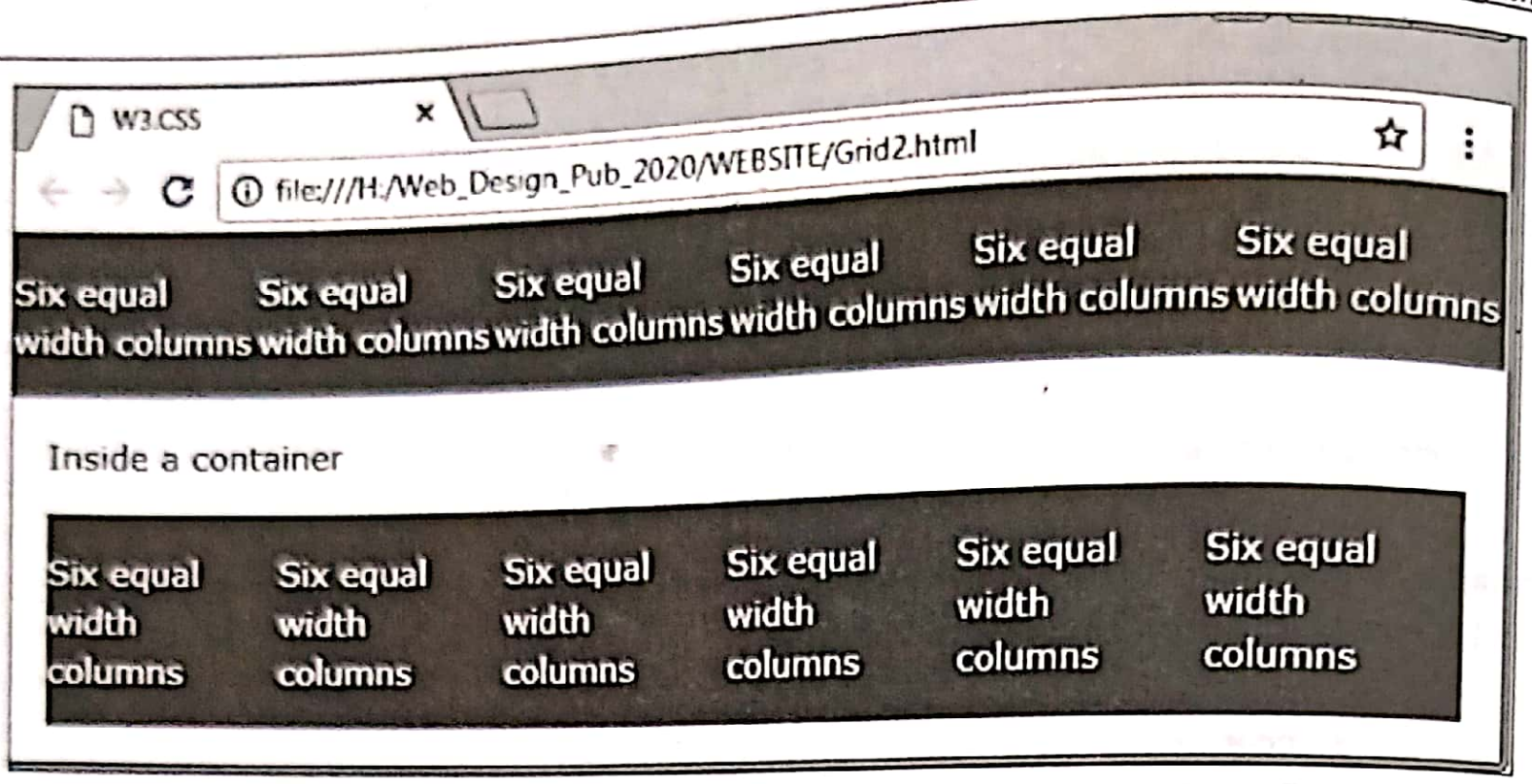


Figure 4.38: Output of the above Example 38

4.11.2 W3.CSS Mixed Grid Layouts

By mixed grid layouts you refer to layouts that look different on different screens. To create mixed layout for mobile and desktops and then for mobile, tablets and desktop screen, you have to use two "s" and "l". It is used to specify column size for small screens or mobiles while the latter specified size for large screens or desktop.

Example 39:

```
<!DOCTYPE html>
<html>
<title>W3.CSS</title>
<head>
<meta name="viewpoint" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="w3.css">
</head>
 <div class="w3-row">
  <div class="w3-col s5 17 w3-blue">
 <p>Div covers 5 column on small and 7 columns on large screens.</p> </div>
<div class="w3-col s5 18 w3-green">
  <p>Div covers 5 column on small and 8 columns on large screens.</p> </div>
<div class="w3-row w3-container">
 <p>Inside a container</p>
  <div class="w3-col s6 17  w3-blue">
 <p>Div covers six column on small and 7 columns on large screens.</p>
</div>
<div class="w3-col s6 18 w3-green">
  <p>Div covers six column on small and 8 columns on large screens.</p>
</div>
</body>
</html>
```

The output of the code given in Example 39 as shown in Figure 4.39.

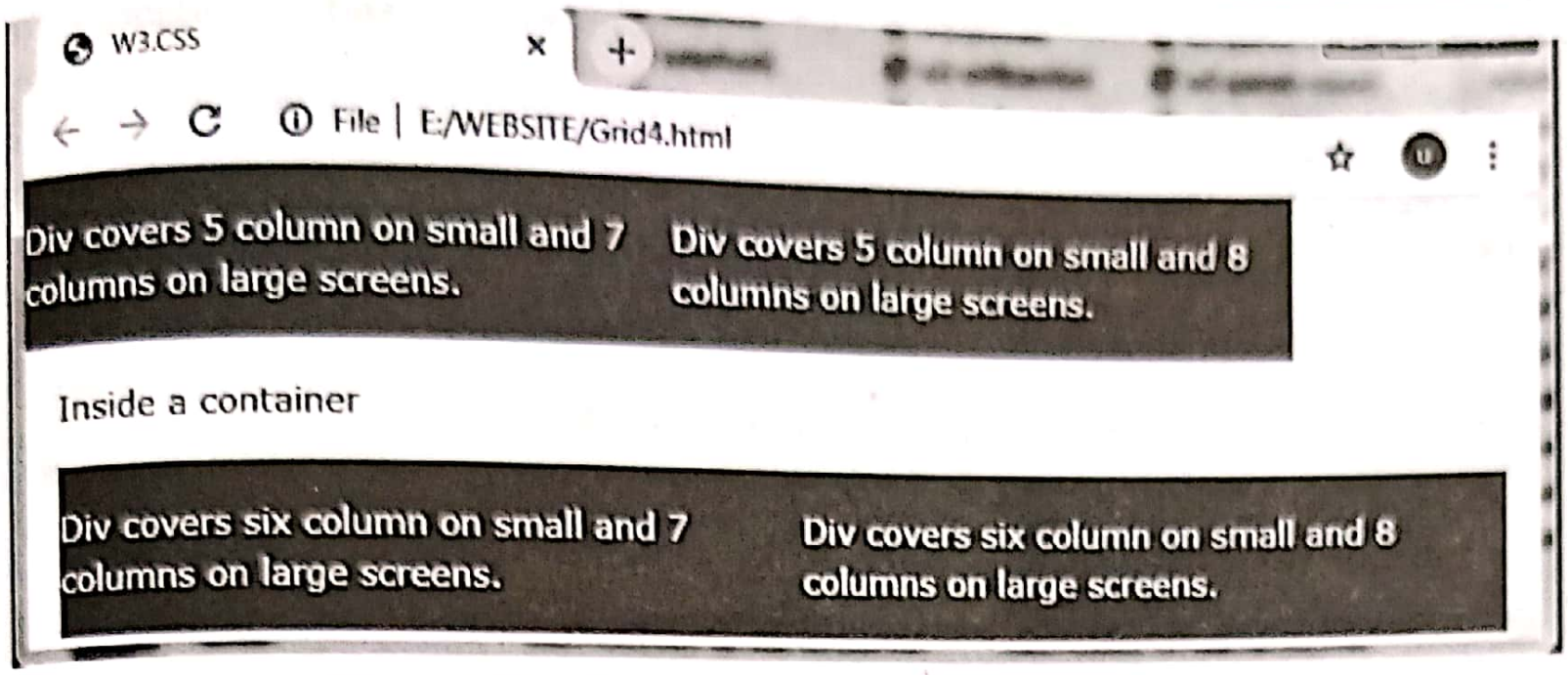


Figure 4.39: Output of the above Example 39

In the following example, you have a row that has two divs. The first div takes 5 columns on the small screen and the second div also takes 5 columns on the small screen. For large screen, the first div will take 7 columns i.e. 58.31% of the width and the second column will takes eight columns i.e. 66.64% of the width.

4.11.3 Mixed Layout for Mobile, Tablet and Desktop

To create mixed layout for mobile, tablet and desktop screens, you have to use two "s", "m", "l". It is used to specify column size for small screen or mobile while the latter specifies screen size for large screens or desktop.

Example 40:

```
<!DOCTYPE html>
<html>
<title>W3.CSS</title>
<head>
<meta name="viewpoint" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="w3.css">
</head>
 <div class="w3-row">
  <div class="w3-col s4 m6 13 w3-blue-grey">
 <p>Div covers 4 columns on small 6 columns on medium and 3 columns on large
screens.</p> </div>
<div class="w3-col s4 m3 17 w3-dark-grey">
  <p>Div covers 4 columns on small 3 columns on medium and 7 columns on
large screens.</p> </div>
<div class="w3-col s4 m3 12 w3-light-grey">
  <p>Div covers 4 columns on small 3 columns on medium and 2 columns on
large screens.</p> </div>
<div class="w3-row w3-container">
 <p>Inside a container</p>  <div class="w3-col s4 m6 13 w3-blue-grey">
 <p>Div covers 4 columns on small 6 columns on medium and 3 columns on large
screens.</p> </div>
<div class="w3-col s4 m3 17 w3-dark-grey">
```

(Contd...)

Scanned with OKEN Scanner

```
<p>Div covers 4 columns on small 3 columns on medium and 7 columns on large
screens.</p> </div>
<div class="w3-col s4 m3 l2 w3-light-grey">
  <p>Div covers 4 columns on small 3 columns on medium and 2 columns on
large screens.</p> </div>
</body>
</html>
```

The output of the code given in Example 40 as shown in Figure 4.40.

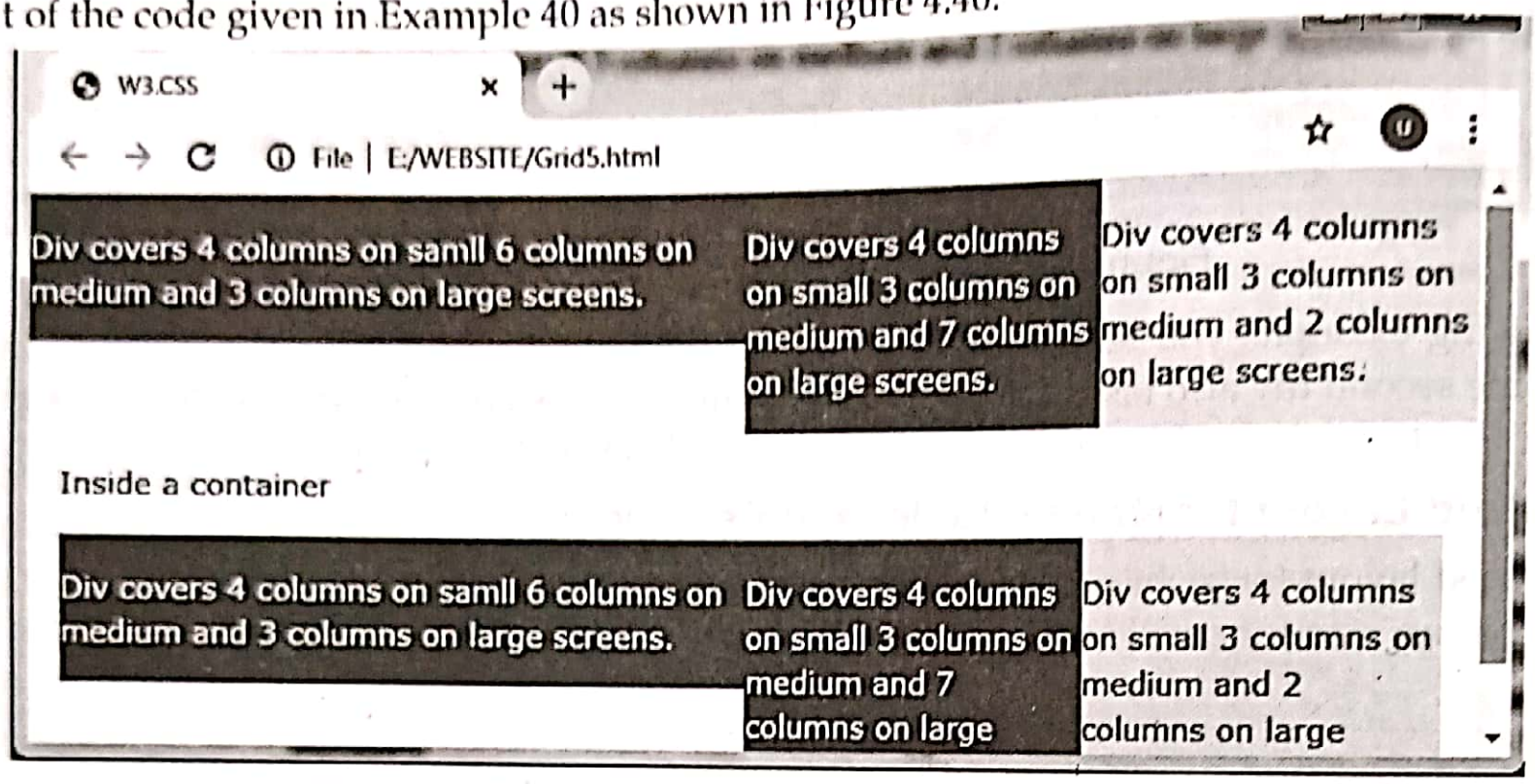


Figure 4.40: Output of the above Example 40

This example will produce a three column layout with 33.33%, 33.3%,33.3% split on small screen, 50%, 25%, 25% split on medium screens and a 25%, 58.34% and 16.66% split on large screens.

Conclusion

W3.CSS is a modern CSS framework for building responsive websites. W3.CSS is based on plain CSS which makes it extremely fast. Next, we have discussed on w3css colors. It supports a rich set of color classes. These color classes are used in marketing and sticky notes. Next, we have discussed W3CSS frameworks contains a class that can be used to style container type elements such as divs, sections, headers, footers, articles etc. Then, we have discussed W3.CSS panels used for displaying notes, quotes, alerts, card and so on. After, we have discussed. W3.CSS borders with different types of borders around html elements. Next, we have discussed W3.CSS fonts and text with different classes that are used to style font size and style text and add text effects. Next, you will learn tables and lists. Tables is the most fundamental layout element, which you want to display data in tabular form whereas lists are used to display data in the form of lists. Next, you will learn W3.CSS images are one of the most important building blocks of website to display images in beautiful and interesting way using an "img" tag. Finally, we have discussed W3.CSS grid system has a 12 column that supports responsive classes for small, large and medium screen sizes that are fully compatible with PC, tablets, and mobile devices.

REVIEW QUESTIONS WITH ANSWERS

1. Multiple Choice

1.1 To use w3.css in your program _____ add stylesheet document in the web page.

a. href="https://www.w3school.com/lib/.css"

Scanned with OKEN Scanner

b. href="http://www.w3schools.com/lib/w3.css"
c. hrefs="http"//www.w3school.com/...lib/w3.css"
d. href="https://www.w3schools.com/...lib/w3.css"

1.2 Which one of the following class to set background color performed by w3.css?

a. w3.color
b. w3-background-color
c. w3-bgcolor
d. w3.color

1.3 Which of the class to set the element of text color?

a. w3.text.color
b. w3.css-text-color
c. w3-text-color
d. w3.text.color

1.4 To set border on all slides which class to add borders style.

a. w3_border
b. w3-border
c. w3-borders
d. w3-bordered

1.5 w3-container class used to style multiple container type elements in a similar style by adding:

a. Uniform
b. Padding
c. Fonts
d. All of the above

1.6 In which classes to give hoverable background colors to the text and buttons by using:

a. w3-hover-color
b. w3.hover-color
c. w3:hover-color
d. w3-hoverable-color

1.7 Which class to set background color of any html element in w3.css?

a. w3.color
b. w3_color
c. w3-color
d. w3:color

1.8 Which of these the w3-container class is not used for HTML container elements like?

a.
b.
c. <footer>
d. <article>

1.9 Choose the correct class in W3.CSS to sets the background color when mouse is hovered over an element.

a. w3hover:color
b. w3-color-hover
c. w3hover:color
d. w3-hover-color

1.10 Which class to write bordered shadow for displaying paper-like cards?

a. w3-card
b. w3:card
c. w3.card
d. None of the above

2. State whether the following statements are True or False

2.1 The w3-container class has a default 16px left and right padding and no top or bottom padding.
2.2 W3.CSS default font size is 20px.
2.3 The default font is verdana, which has a good letter spacing in W3.CSS.

Scanned with OKEN Scanner

2.4 The w3-li class is used to display a basic list.
2.5 W3-bottombar is not a type of bar classes.
2.6 The w3-color-opacity classes make images transparent.
2.7 w3-col defines the number of columns occupied in the 12 column system.
2.8 The w3-hover opacity class adds add transparency to the image when mouse over.
2.9 Add w3-padding class to the image it adds a further padding between the image and its border.
2.10 W3.CSS is not compatible with opera and safari browsers.

3. Match the following

3.1	It is used to add transparency to the image when mouse over	a.	S4
3.2	It is the rounded element with border radius 2px	b.	Which padded container for responsive columns
3.3	The rounded element with border radius 4px:	c.	w3-hover-opacity
3.4	It specify the hoverable border col:	d.	w3-bordered
3.5	Column-level styles for small screen devices defines as	e.	w3-round
3.6	w3-row-padding	f.	ul opening tag
3.7	To add bottom border to each row of the table by using	g.	w3-round-small
3.8	To add background color of the complete list add w3-color with	h.	Is rounded element with border radius 64px
3.9	To add hover color to individual list item add w3-hover-color before	i.	w3-opacity-min
3.10	w3-round-jumbo	j.	li opening tag
		k.	w3-hover-border-color

4. Fill in the blanks

a. w3-table-all
b. w3-card
c. w3-table
d. w3-round
e. w3-hoverable
f. w3-display-classes
g. w3-panel
h. w3-border, w3-bordered
i. w3-container
j. <blockquote>
k. w3-card-9

4.1 To create a simple table, with no affects and striping, simple use _____ class inside the table tag of the table.

Scanned with OKEN Scanner

4.2 There are two classes that add border in the W3.CSS tables. The first class is _____ class that adds border around the whole table. The second class is the _____ class adds border around each strip.

4.3 To create a table having strips, border around the strips and borders around the whole table, create a table and add _____ class to the opening table tag.

4.4 By using visual aesthetics of a table, use _____ class adds an effect to the table which changes the table row color upon mouse hover.

4.5 To create a card with W3.CSS, use the _____ class with div.

4.6 _____ class is the perfect class to display notes and information.

4.7 The _____ class adds rounded corners to an image.

4.8 To position the text in an image _____ class is used in w3.css style.

4.9 User can give padding to the text by using _____ () class which gives 16px left and right padding by default and which does not give top and bottom padding.

4.10 To use html _____ in w3-container class can be used to display quotes to add an extra left margin.

Descriptive Questions with Answer

5.(a) What is W3.CSS?

Answer

W3.CSS stands for Cascading Style Sheet. It is developed by w3schools.com. It is used to create faster, beautiful and responsive websites.

5.(b) What are the features of W3.CSS?

Answer

The features of W3.CSS are as follows:

a. It is in-built responsive web designing.
b. It provides standard CSS.
c. It is free to use.
d. It is inspired by Google Material Design.

5.(c) What are the techniques to use W3.CSS?

Answer

There are two techniques to use W3.CSS.

Local installation: You can download the W3.CSS file on our local machine and include it in to html.

CDN based: You can include the W3.CSS file into our html code directly from the CDN.

6.(a) What are the classes of W3.CSS container?

Answer

- w3-container
- w3-red
- w3-border
- w3-teal

6.(b) How can you use subclasses in W3.CSS?

Answers

Scanned with OKEN Scanner

```
<div class="w3-col s2 m4 l3">
 <p>The text use 2 columns on small screen, 4 on a medium screen and 3 on a large screen</p>
</div>
</div>
```

6.(c) What are the classes of W3.CSS card?

Answer

The various classes of w3.card:

- w3-card
- w3-card-2
- w3-card-4
- w3-card-8
- w3-card-12

7.(a) How to include the external font styles and give the external fonts in the head section.

Answer

```
<!DOCTYPE html>
<html>
<head>
<title>W3.CSS</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet" type="text/css" href="w3.css">
 <link rel="stylesheet" href="https:     //fonts.googleapis.com/
                        css?family=Tangerine">
    <style>
        .w3-tangerine {
            font-family: "Tangerine", serif;
        }
    </style>
    <body>
        <div class="w3-container w3-tangerine">
            <p class="w3-xxlarge">HELLO!</p>
            <p class="w3-xxxlarge">HELLO!</p>
            <p class="w3-jumbo">HELLO!</p>
        </div>
    </body>
</html>
```

The output of the above code as shown in Figure 4.41.

7.(b) Write a program by using w3.css class i.e. w3-text-shadow to make the text shadow by using all colors to designed.

Scanned with OKEN Scanner

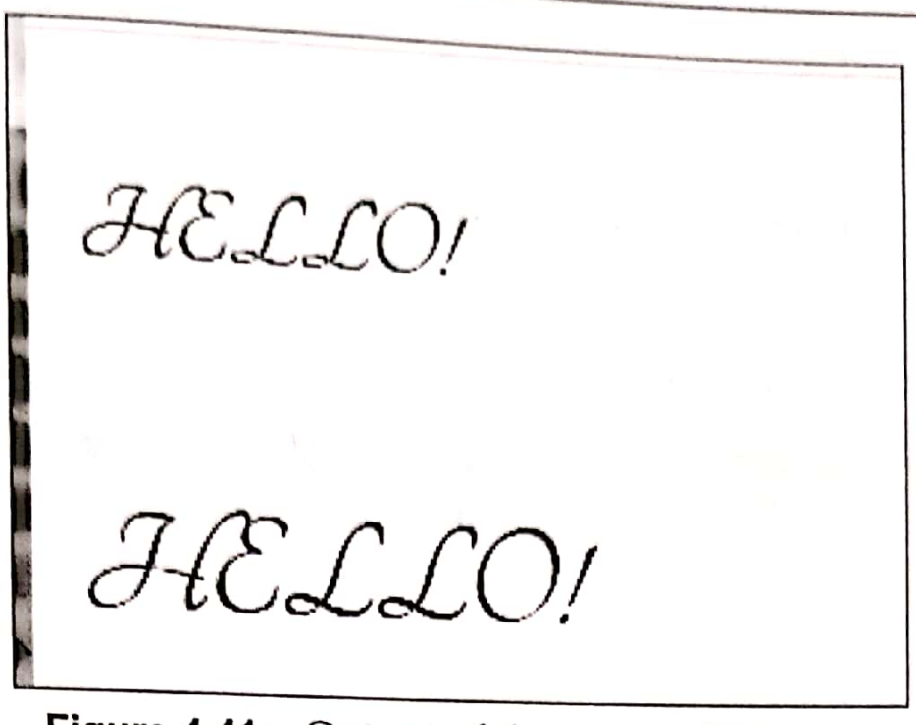


Figure 4.41: Output of the answer 7(a)

Answer

```
<!DOCTYPE html>
<html>
<head>
<title>W3.CSS</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet" type="text/css" href="w3.css">
 <body>
        <div class="w3-container">
          <h2 class="w3-text-shadow">TEXT SHADOW!</h2>
        </div>
        <div class="w3-container w3-light-grey">
          <h2 class="w3-text-shadow">TEXT SHADOW!</h2>
        </div>
        <div class="w3-container w3-blue">
          <h2 class="w3-text-shadow">TEXT SHADOW!</h2>
        </div>
        <div class="w3-container w3-lime">
          <h2 class="w3-text-shadow">TEXT SHADOW!</h2>
        </div>
    </body>
</html>
```

The output of the above code as shown in Figure 4.42

Scanned with OKEN Scanner

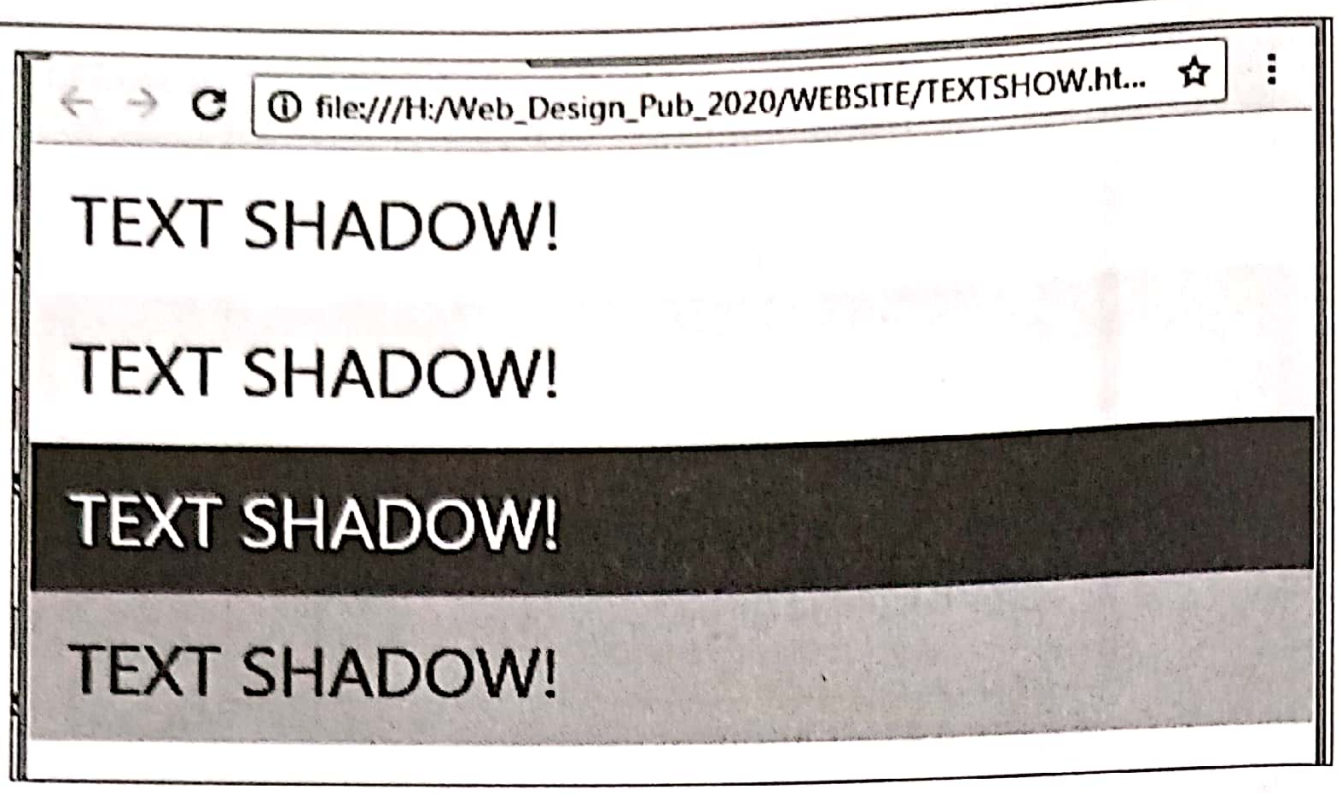


Figure 4.42: Output of the answer 7(b)

8.(a) Create Photo card by using the w3-container card classes.

Answer

```
<!DOCTYPE html>
<html>
<head>
<title>W3.CSS</title>
<meta                    name="viewport"
content="width=device-width, initial-
scale=1">
<link                    rel="stylesheet"
type="text/css" href="w3.css">
 <body>
        <div class="w3-container">
            <h2>Photo Card</h2>
            <div    class="w3-card-12"
style="width:50%">
                <img        src=".jpg"
alt="splessons" style="width:100%">
                <div        class="w3-
container w3-center">
                    <p>.</p>
                </div>
            </div>
        </div>
    </body>
</html>
```



Figure 4.43: Output of the answer 8(a)

The output of the above code as shown in Figure 4.43.

8.(b) Write a program contains header and paragraphs container.

Scanned with OKEN Scanner

Answer

```
<!DOCTYPE html>
<html>
<head>
<title>W3.CSS</title>
<meta                    name="viewport"
content="width=device-width, initial-
scale=1">
<link                   rel="stylesheet"
type="text/css" href="w3.css">
 <body>
     <div   class="w3-container   w3-
blue">
      <H1>Heading Section</h1>
            <p>This                  is
paragraph</p>
             <p>Simple   programming
Tutorials</p>
          </div>
    </body>
</html>
```

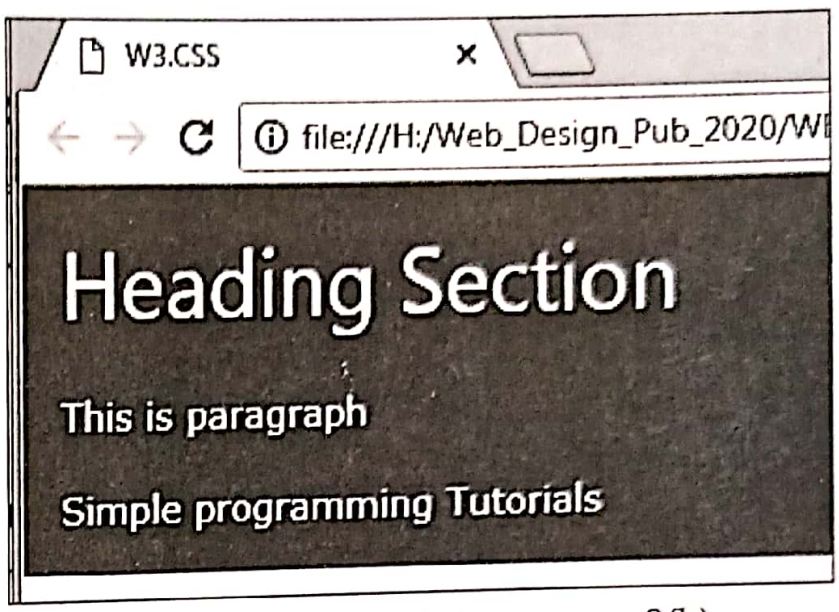


Figure 4.44: Output of the answer 8(b)

The output of the above code as shown in Figure 4.44.

9. (a) What are the various table classes in W3.CSS?

Answer

There are different table classes in W3.CSS:

a. w3-table: It represents a basic table without any border.
b. w3-striped: It is used to display a stripped table.
c. w3-bordered: It is used to draw a table with a border across rows.
d. w3-card: It is used to draws a table as a card.

(b) What are the uses of W3.CSS image classes?

Answer

a. w3-image: It is used to represent a basic style image without any border.
b. w3-circle: It is used to create circle image.
c. w3-title: It is used to display the text over an image
d. w3-card: It is used to create an image as a card.

(c) what is the difference between W3.CSS and CSS?

Answers

CSS: It is cascading style sheet used for style the web pages, with help of css, you can style the web pages, websites, etc. CSS is a modern framework with built in responsiveness.

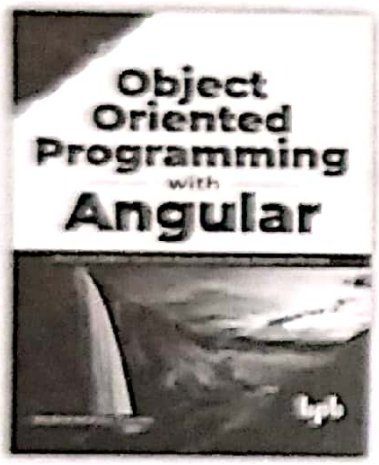
W3.CSS: It is CSS framework, build for responsive design, like bootstrap, that means you create responsive websites with help of W3.CSS. It is very easy to learn and quite fast for mobile application. It has in-built mobile responsive feature which helps you to make mobile responsive.

Scanned with OKEN Scanner

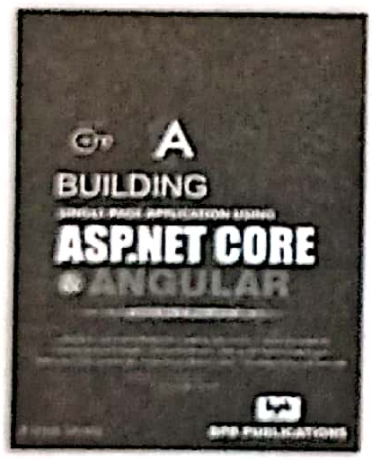
Answers

1.1	b	1.2	a	1.3	c	1.4	b	1.5	d	1.6	a
1.7	c	1.8	b	1.9	d	1.10	a				
2.1	T	2.2	F	2.3	T	2.4	F	2.5	F	2.6	F
2.7	T	2.8	T	2.9	T	2.10	F				
3.1	c	3.2	g	3.3	e	3.4	k	3.5	a	3.6	b
3.7	d	3.8	f	3.9	j	3.10	h				
4.1	c	4.2	h	4.3	a	4.4	e	4.5	b	4.6	g
4.7	d	4.8	f	4.9	i	4.10	j				

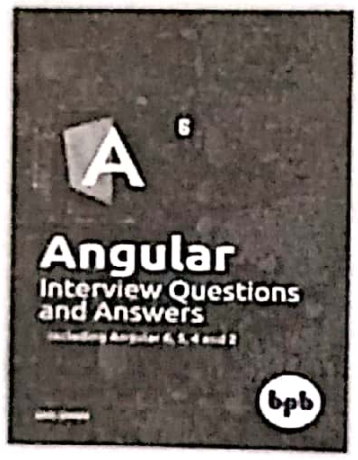
OTHER TITLES OF INTEREST



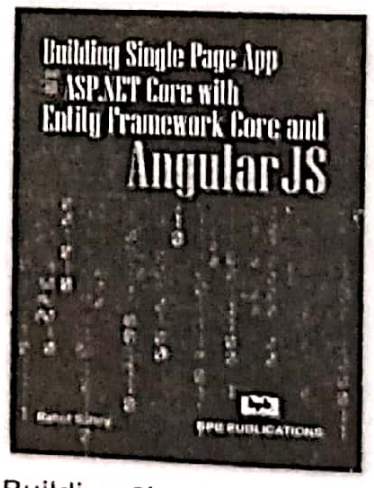
Object Oriented Programming with **Angular**
ISBN: 9789389328363



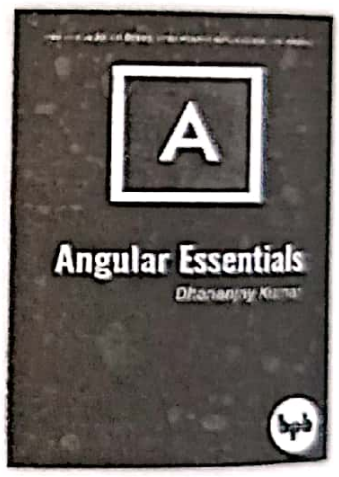
Building Single Page App using **ASP.NET Core** and **Angular**
ISBN: 9789386551900



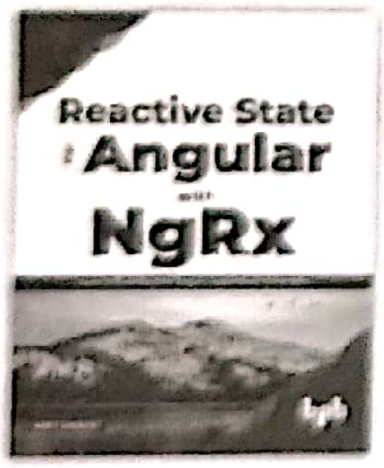
Angular Interview Questions and Answers
Including Angular 6, 5, 4 and 2
ISBN: 9789388176699



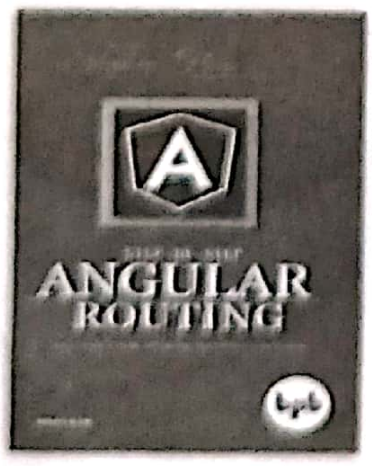
Building Single Page App Using ASP .NET Core With Entity Framework Core and **Angular JS**
ISBN: 9788183335966



Angular Essentials
ISBN: 9789388511247



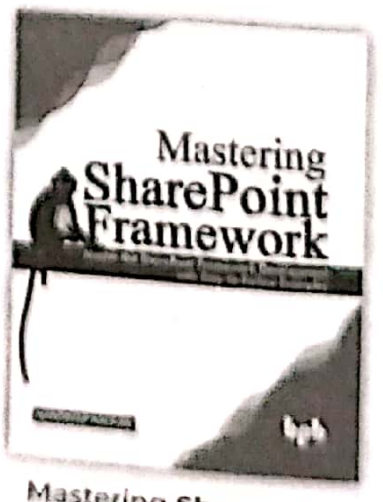
Reactive State for **Angular** with **NgRx**
ISBN: 9789389898248



Step-by-Step **Angular Routing**
ISBN: 9789388511667



Microprocessor & **Microcontroller** Interview Questions
ISBN: 9789389845112



Mastering **Sharepoint Framework**
ISBN: 9789389328875



Oracle GoldenGate With **Microservices**
ISBN: 9789389328486

Scanned with OKEN Scanner

CHAPTER 5

JavaScript and Angular JS

5.1 Introduction

JavaScript is a programming language written for websites to run in browser. It is used to validate the user input at browser before submitting the page to server for further processing. This reduces load on the server. It helps to provide immediate feedback to user as it runs on browser whereas Angular is a framework written using JavaScript and is made for building single-page web applications. It is a continuously growing and expanding framework which provides better ways for developing web application. It changes the static html to dynamic html. Angular's data binding and dependency injection eliminate much of the code writing. It happens within the browser, making it an ideal partner with server technology.

Structure

- Introduction to Client-Side Scripting Language
- Variables in JavaScript
- Operators in JavaScript
- Conditions Statements
- JS Popup Boxes
- JS Events
- Basic Form Validations in JavaScript
- Introduction to Angular JS include Expression, Modules and Directives

Objectives

The reader will be able to understand the following:

- Apply client side scripting
- Adding validations and checks on forms
- Introduction to AngularJS is front-end framework that is used to develop single page web application

5.2 Introduction to Client Side Scripting Language

Scripting is called a program that means some code that runs on computer to perform any task. Client side scripting generates a code that executes in the client side web browser. Therefore, it does not require any server side processing. It is usually, client side scripts are place inside an HTML document. This type of scripting can reduce the server load. When a user requires a specific web page through the browser, the server sends an HTML file, in which the browser can interpret and render the content in the client side. Examples of client side scripting languages are HTML, CSS, and JavaScript. The client-side content is generated on the client's computer. The web browser retrieves a page from the server, then processes the code embedded in the page (typically written in JavaScript) and displays the retrieved page's content to the user.

Client side scripting is fast, because user think that client side scripting is more secure than server side scripting. By using Form, validation is done by client side scripting. The script can draw types of animations and different functions like those that displaying system clock is done easily with client slide scripting.

Scanned with OKEN Scanner

5.2.1 Advantages and Disadvantages of Client side Scripting

Developers might consider using client-side scripts for many reasons, and they might have some valid reasons not to use them.

Advantages

1. Client-side scripting offers various advantages, including:
2. It allows for more interactivity by immediately responding to user's actions.
3. Client-side scripting is used to send users requests to the server, and also retrieve data from the server.
4. Client side scripting is used to provide remote service for applications, such as content delivery, software registration.
5. It can be substituted with alternative for example, HTML if user's browsers do not support scripts.

Disadvantages

Client-side also create additional uncertainties, which include:

1. Not all browsers support scripts, so, users may experience errors if no alternatives have been provided.
2. Different versions of browser support different type support scripts differently.
3. More development time and effort might be required if the scripts are not already available through other resources.

How to write a JavaScript into an HTML page.

The example below shows how to use JavaScript to write text on a web page.

Example

```
<html>
<body>
 <script type="text/javascript">
 document.write("hello world!");
</script>
</body>
</html>
```

The example below shows how to add HTML tags to the JavaScript:

```
<html>
<body>
 <script type="text/javascript">
 document.write("<h1>hello world!</h1>");
</script>
</body>
</html>
```

The output of the code shown in browser as hello world! in the web page browser.

To insert a JavaScript into an HTML page, you use the <script> tag, inside the <script> tag and you use the type attribute to define the scripting language.

So, the <script type="text/javascript"> and </script> tells where the JavaScript starts and ends.

```
<html>
 <body>
 <script type="text/javascript">
...
```

Scanned with OKEN Scanner

```
</script>
</body>
</html>
```

The document.write command is a standard JavaScript command for writing output to a page.

By entering the document.write command between the <script> and </script> tags, the browser will recognize it as a JavaScript command and executes the code line.

5.3 Variables in JavaScript

Variable means anything that can vary. It includes variables, which hold the data value, and it can be changed anytime.

1. While declaring a JavaScript variable there are some rules;
2. Name must start with a letter (a to z or A to Z), underscore(_) or dollar ($) sign.
3. After first letter you can use digits (0 to 9).
4. JavaScript variables are case-sensitive, for example x and X are different variables.

For example,

```
var x = 100;
```

Here's what is happening in the example above:

var is the keyword that tells JavaScript you are declaring a variable.

X is the name of that variable.

= is the operator that tells JavaScript a value is coming up next.

100 is the value for the variable to store.

Using variables

After you declare a variable, you can reference it by name elsewhere in your code.

Example,

```
var x =100;
x + 102;
```

Output

202

You can even use a variable when declaring other variables.

Example,

```
var x = 100;
var y = x+102;
y;
```

Output

202

There are two types of variables in JavaScript: **local variable** and **global variable.**

Local variable

A variable that is declared inside a function definition is called **local** and has scope to that function only. It does not support block scope in which a set of curly braces {...} defines a new scope.

Scanned with OKEN Scanner

For example,

```
<!DOCTYPE html>
<html>
<body>
<script>
  var a = "Welcome to Javascript";
   function show() {
 //A local variable is declard in this function
 var a = "Hello World !";
alert("Value of 'a' inside the function " + a); //Hello world!
}
alert("value of 'a' outside the function: " + a);  //Welcome to Javascript !
</script>
</body>
</html>
```

Global variable

A variable that is declared outside of a function definition is called a global variable and its scope is through your program means its value is accessible and modifiable throughout your program. *For example,*

```
<!DOCTYPE html>
<html>
<body>
<script>
  var a = "Welcome to Javascript";
   function show() {
 //A local variable is declard in this function
 var a = "Hello World !";
alert("Value of 'a' inside the function: " + a); //Hello world!
// b will have global scope
 b = "Hello World !";
 display();
alert("value of 'a' outside the function: " + a);  //Welcome to Javascript !
function display() {
//since b variable has global scope
alert("value of 'b' outside the function: " + b); //Hello World!
}
</script>
</body>
</html>
```

Declare Multiple Variables in a Single Line

You can declare multiple variables and set their initial values in a single statement. Each variable is separated by **commas**, as shown in the following example.

Scanned with OKEN Scanner

Example 1:

```
<!DOCTYPE html>
<html>
<body>
<h2>Declaring mulitple variables in a Single line</h2>
 <p id="d1"></p>
 <p id="d2"></p>
 <p id="d3"></p>
<script>
  var name = "Govind", MobName = "Volvo", Price = "200000";
document.getElementById("d1").innerHTML = name;
document.getElementById("d2").innerHTML = MobName;
document.getElementById("d3").innerHTML = Price;
</script>
</body>
</html>
```

The output of the above code as shown in Figure 5.1.

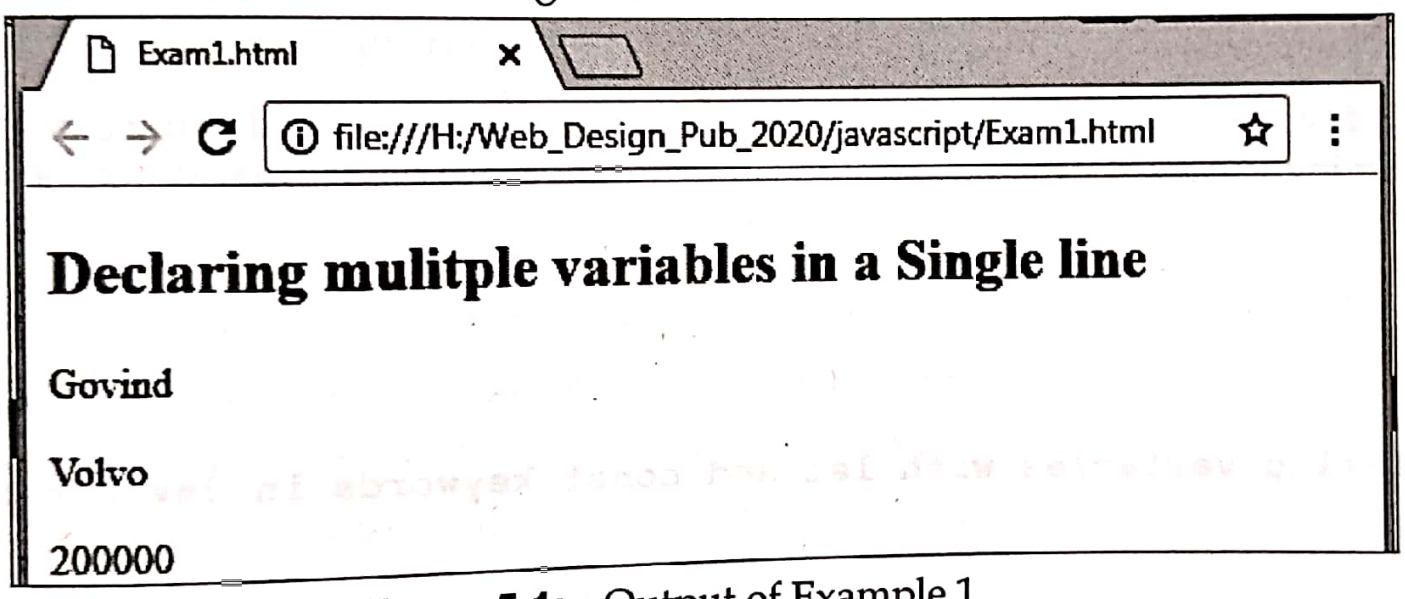


Figure 5.1: Output of Example 1

Loosely-typed Variables

JavaScript variables are loosely-typed which means it does not require a data type to de declared. You can assign any type of literal values to a variable e.g. string, integer, float, Boolean etc.

Example 2:

```
<!DOCTYPE html>
<html>
<body>
<h2>JavaScript Loosely-typed variables</h2>
 <p id="d1"></p>
 <script>
  var two = 1;   //number value
    two = 2.1;   // decimal value
    two = true; //boolean value
```

(Contd...)

Scanned with OKEN Scanner

```
    two = null; //null value
    two = 'two'; //string value
document.getElementById("d1").innerHTML = two;
</script>
</body>
</html>
```

The output of the above code as shown in Figure 5.2.

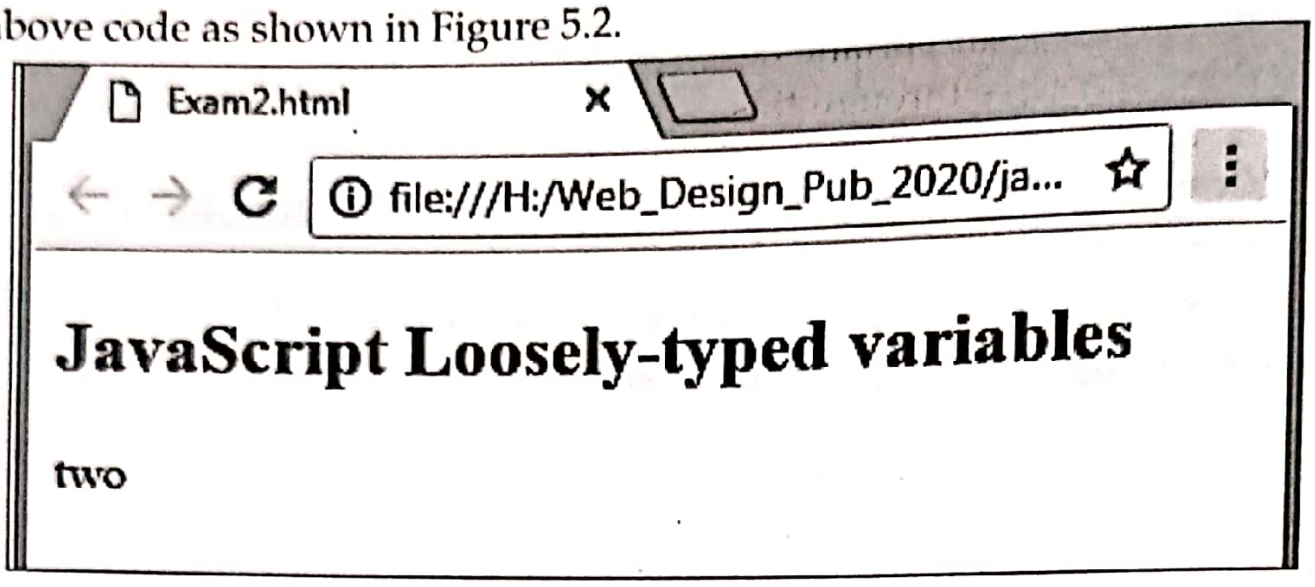


Figure 5.2: Output of Example 2

The let and const keywords

The const keyword works exactly the same as let, except that variables declared using const keyword cannot be reassigned later in the code.

Example 3:

```
<!DOCTYPE html>
<html>
<head>
<title>Declaring variables with let and const keywords in Javascript</title>
</head>
<body>
  <script>
   //Declaring variables
  let name = "Harrish";
   let age = 25;
   let Married = false;
   //Printing variable values
  document.write(name + "<br>");
  document.write(age + "<br>");
  document.write(Married + "<br>");
  //Declaring constant
   const PI=3.14;
//pringint constant value
document.write(PI); //3.14
//trying to reassing
PI = 10; //error
```

Scanned with OKEN Scanner

```
</script>
</body>
</head>
</html>
```

The output of the above code as shown in Figure 5.3.

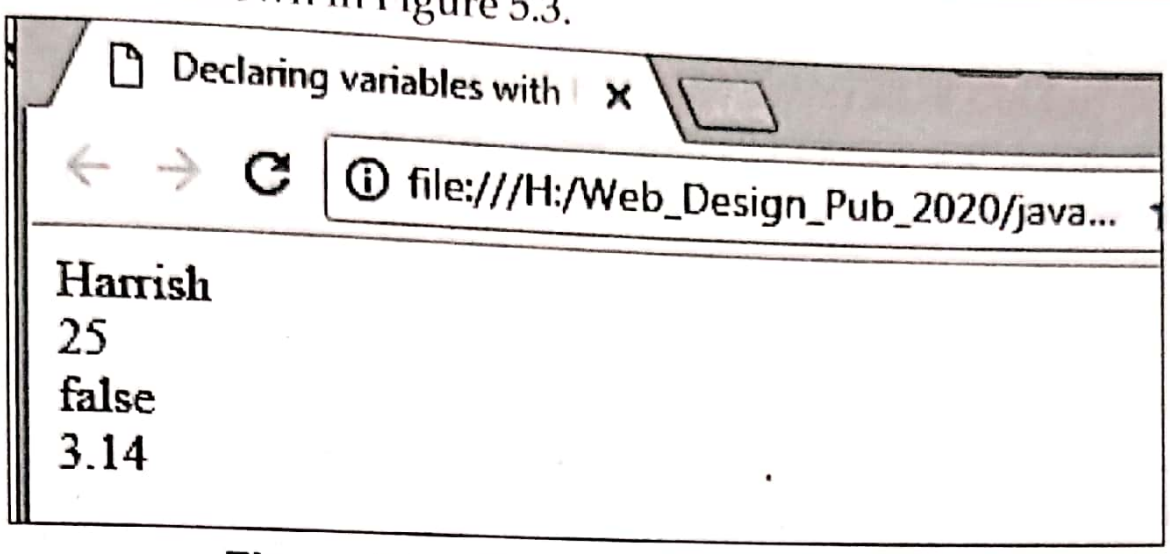


Figure 5.3: Output of Example 3

Data types in JavaScript

Data types basically specify what kind of data can be stored and manipulated within a program. There are two types of data types in JavaScript. These are:

1. **Primitive data type:** String, Number, Boolean and Null and Undefined
2. **Non-primitive (or reference) data type:** Object, Date and Array

Primitive data types

There are five types of primitive data types. These are as follows:

- **String Data Type:** It represents sequence of characters e.g. "string"
- **Number Data Type:** It represents numeric values e.g. 100
- **Boolean:** It represents Boolean value either true or false
- **Undefined:** It represents undefined value
- **Null:** It represents null i.e. no value at all

Non-primitive data types

The non-primitive data types are as follows:

- **Object:** It represents instance through which we can access members
- **Array:** It represents group of similar values
- **RegExp:** It represents regular expression

Sting Data Type

The string data type is used to represent textual data (i.e. sequence of characters). It had been created using single or **double quotes** surrounding one or more characters are shown below.

Example 4:

```
<!DOCTYPE html>
<html>
<head>
```

Scanned with OKEN Scanner

```
<title>Javascript string data types</title>
</head>
<body>
 <p id="Str"></p>
  <script>
   //Creating variables
    var a = "Well done is better than well said.";  //double quotes
    var b = 'Well done is better than well said.';  //single quotes
  document.getElementById("Str").innerHTML = a + "<br>" + b ;
  </script>
</body>
</head>
</html>
```

The output of the above code as shown in Figure 5.4.

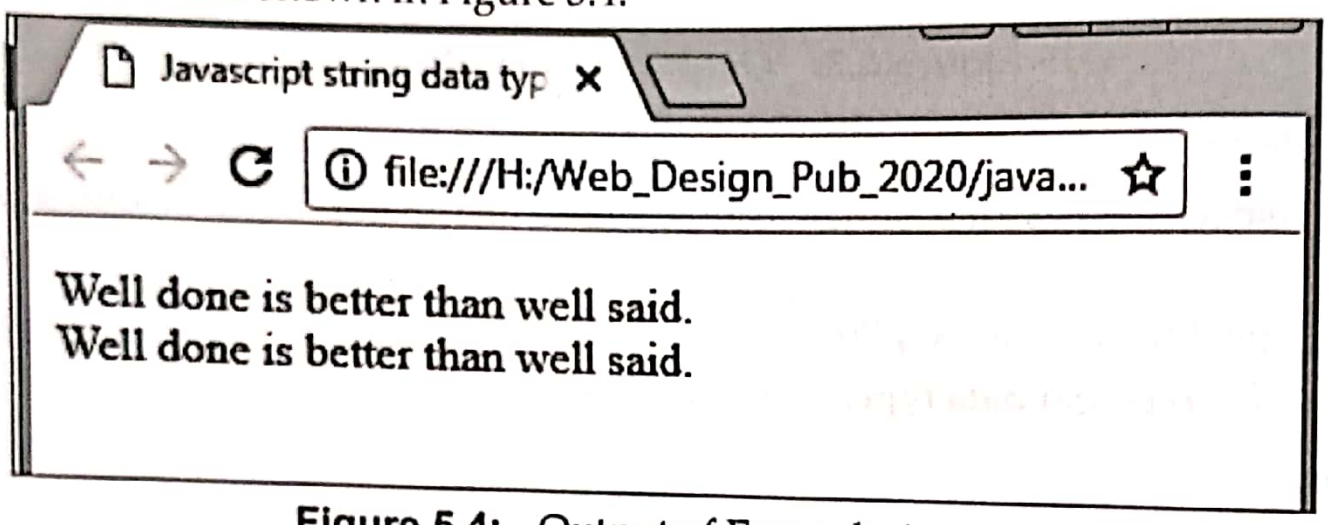


Figure 5.4: Output of Example 4

String object comparison

While working with String object because comparison of string objects using **= = operator** compares string objects and not the values.

Example 5:

```
<!DOCTYPE html>
<html>
<head>
<title>String Object comparison</title>
</head>
<body>
 <p id="s1"></p>
 <p id="s2"></p>
 <p id="s3"></p>
 <p id="s4"></p>
 <p id="s5"></p>
 <script>
    var str1 = new String('Welcome to Javascript');
    var str2 = new String('Welcome to Javascript');
```

(Contd..)

Scanned with OKEN Scanner

```
    var str3 = 'Welcome to Javascript';
    var str4 = str1;
  document.getElementById("s1").innerHTML = str1 == str2;
  document.getElementById("s2").innerHTML = str1 == str3;
  document.getElementById("s3").innerHTML = str1 === str4;
  document.getElementById("s4").innerHTML = typeof(str1);
  document.getElementById("s5").innerHTML = typeof(str3);
  </script>
</body>
</head>
</html>
```

The output of the above code as shown in Figure 5.5

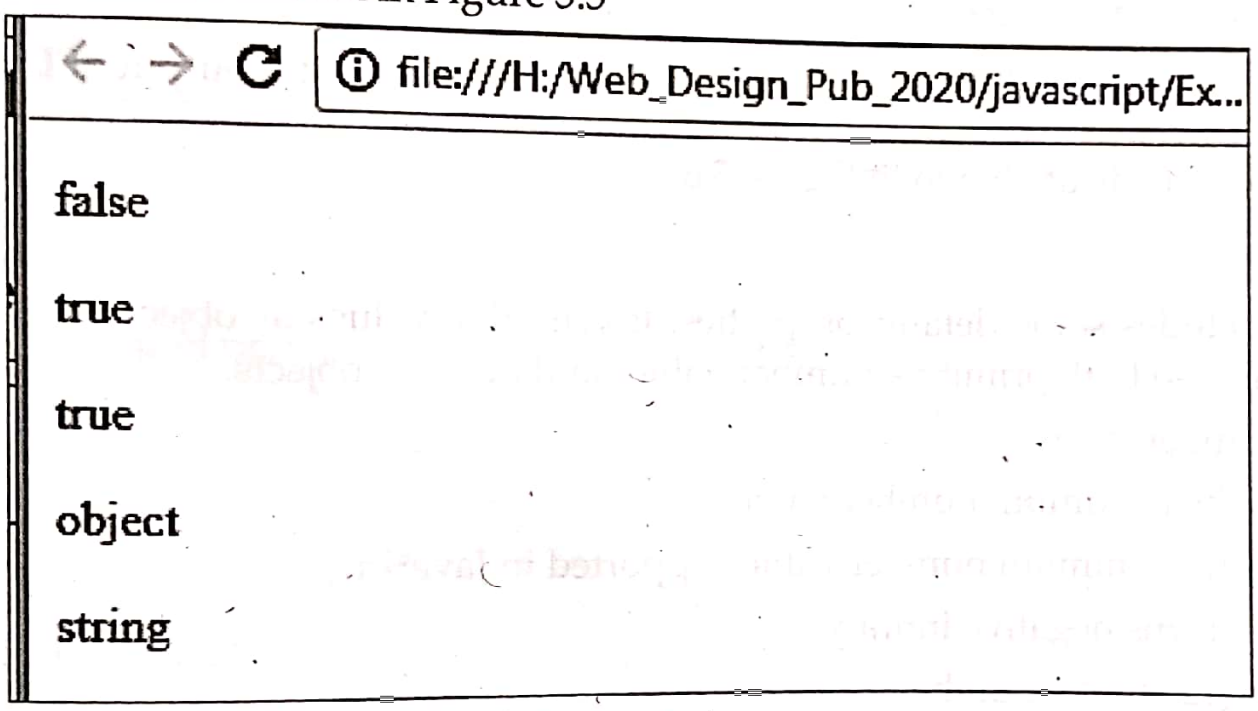


Figure 5.5: Output of Example 5

Number Data Type

The Number is a primitive data type. It represents **integer, float, hexadecimal, octal** and **exponential** value.

Example 6:

```
<!DOCTYPE html>
<html>
<head>
<title>JavaScript</title>
</head>
<body>
<h1>Number Data Type</h1>
  <script>
      var a = 25;  //integer
      var b = 90.5;  //float
      var c = 0xfff;  //hexadecimal
      var d = 3.56e4;  //exponential
      var e = 050; //octal
```

(Contd...)

Scanned with OKEN Scanner

```
        var  f  =  3.56e-4;  //negative
exponential
    document.write(a + "<br>");
    document.write(b + "<br>");
    document.write(c + "<br>");
    document.write(d + "<br>");
    document.write(e + "<br>");
    document.write(f);
  </script>
</body>
</head>
</html>
```

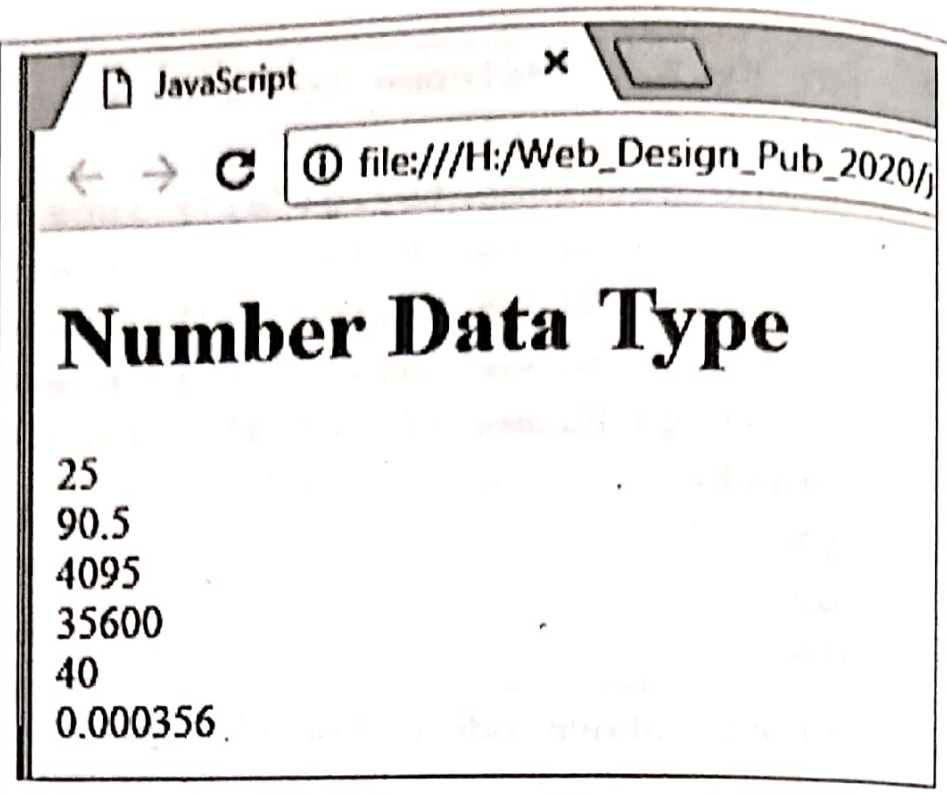


Figure 5.6: Output of Example 6

The output of the above code as shown in Figure 5.6

Number Properties

The Number type includes some default properties. It primitive values as object, so all the properties and methods are applicable to both primitive number values and number objects.

The properties of Number type

Max_value: Returns the maximum number value.

Min_value: Returns the minimum number value supported in JavaScript.

Negative_infinity: Returns negative infinity.

NaN: Returns NaN value (not a number).

Positive_infinity: It represents positive infinity.

Example 7:

```
<!DOCTYPE html>
<html>
<head>
<title>JavaScript</title>
</head>
<body>
<h1>Number Data Type</h1>
   <script>
     alert(' Max Value: ' + Number.MAX_VALUE +
   '\n Min Value: ' + Number.MIN_VALUE +
   '\n Negative Infinity:' + Number.NEGATIVE_INFINITY +
   '\n Positive Infinity:' + Number.POSITIVE_INFINITY +
   '\n Nan:' + Number.NAN
);
</script>
</body>
</head>
</html>
```

Scanned with OKEN Scanner

The output of the above code as shown in Figure 5.7.

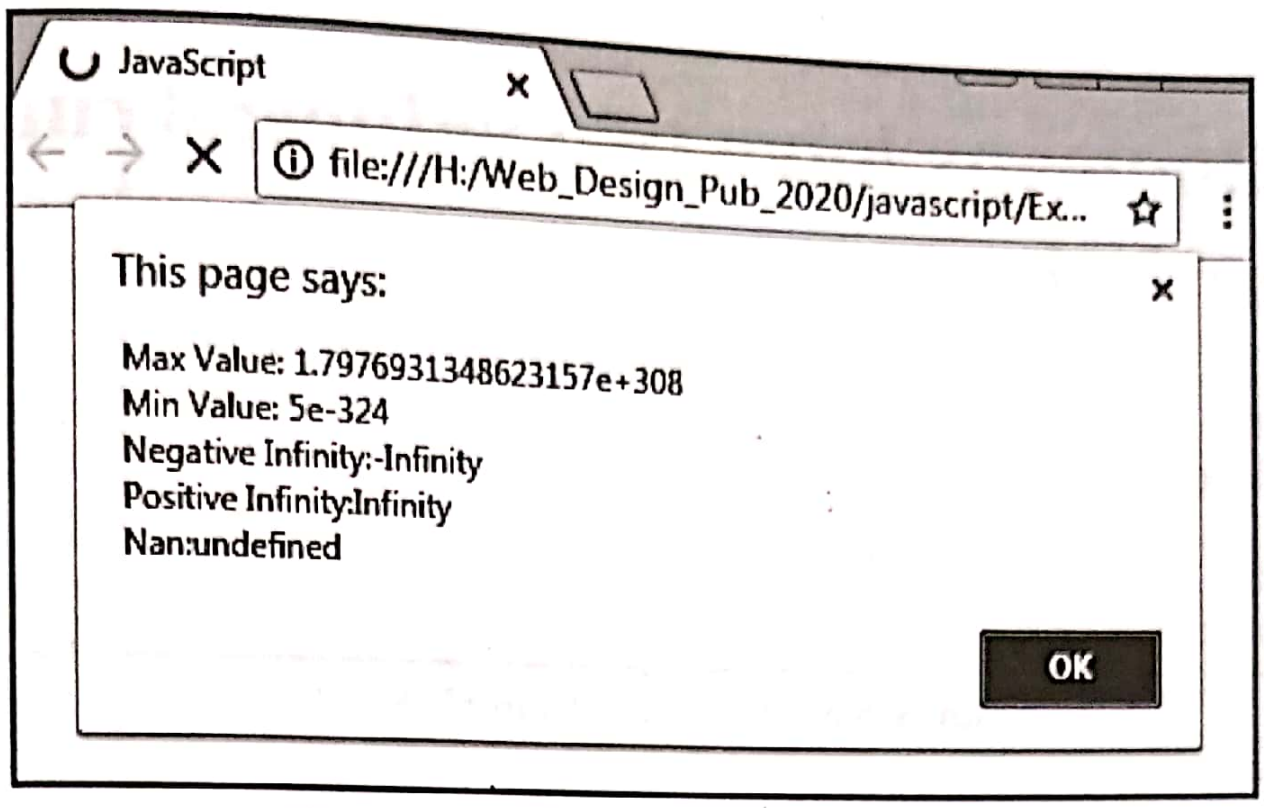


Figure 5.7: Output of Example 7

The Boolean Data Type

The Boolean data type can hold only two values i.e. True or False.

Example 8:

```
<!DOCTYPE html>
<html>
<head>
<title>JavaScript</title>
</head>
<body>
<h1>Booleans have two values: True or False</h1>
<p id="s1"></p>
  <script>
        var x = 5;
        var y = 5;
        var z = 6;
document.getElementById("s1").innerHTML =
 (x == y) + "<br>" + (x == z);
 var a = 2, b = 5, c = 10;
   document.write(b > a); //output: true
   document.write("<br>");
   document.write(b > c) //output: false
 </script>
</body>
</head>
</html>
```

The output of the above code as shown in Figure 5.8.

Scanned with OKEN Scanner

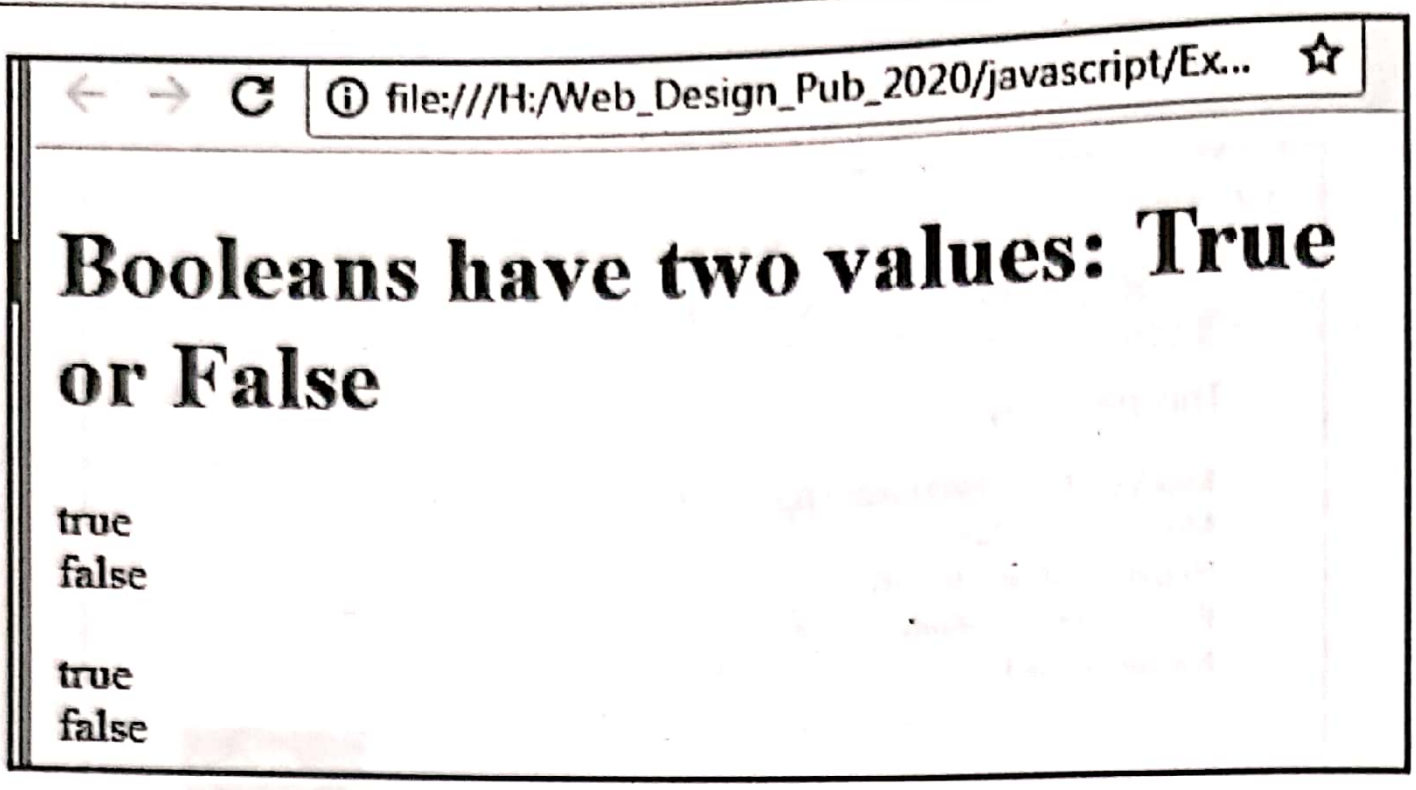


Figure 5.8: Output of Example 8

Null Data Type

This is another special data type that can have only one value the null value. A null value means that there is no value.

Example 9:

```
<!DOCTYPE html>
<html>
<head>
<title>JavaScript</title>
</head>
<body>
<h1>Booleans have two values: True or False</h1>
<p id="s1"></p>
   <script>
    var person = {firstname:"James", lastname:"Bond", age:50};
    person = null;
  document.getElementById("s1").innerHTML = typeof person;
 document.getElementById("s1").innerHTML =  typeof undefined + "<br>" +
 typeof null + "<br><br>" +
 (null === undefined) + "<br>" +
 (null == undefined);
 </script>
</body>
</head>
</html>
```

The output of the above code as shown in Figure 5.9.

Object Data type

The object is a complex data type, which allows you to store collection of data. It contains properties, defined as a key-value pair in { }, . A property key (name) is always a string, but the value can be any data type, such as strings, numbers, Booleans or complex data types like arrays, function and other objects.

Scanned with OKEN Scanner

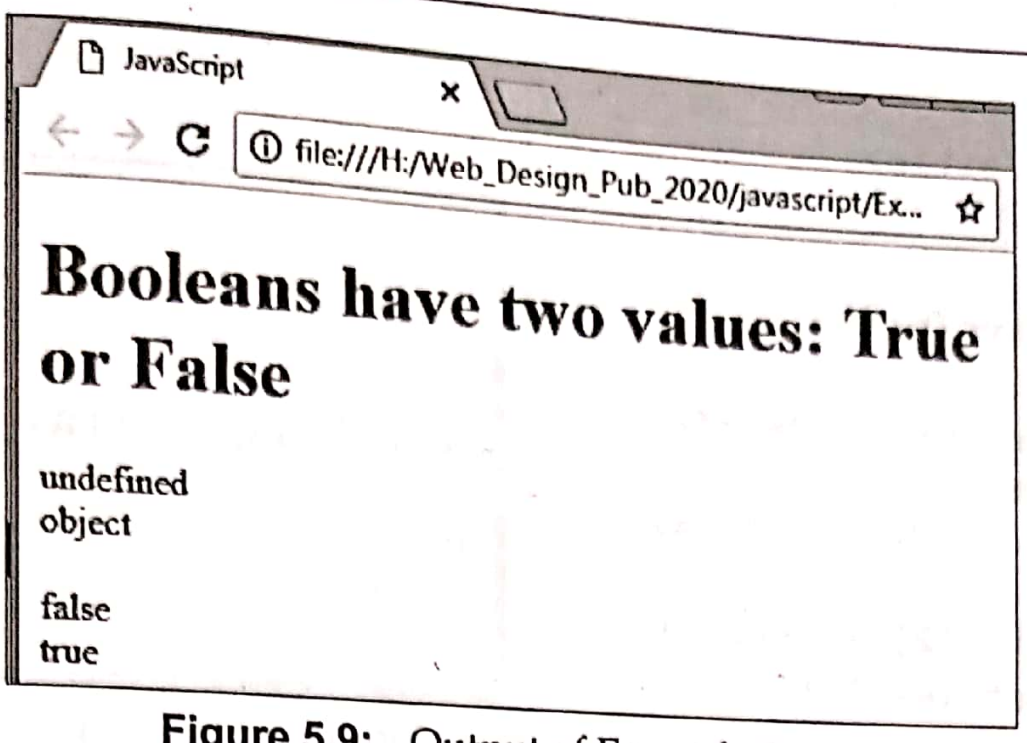


Figure 5.9: Output of Example 9

The following example will show you the simple way to create an object in JavaScript.

Example 10:

```
<!DOCTYPE html>
<html>
<head>
<title>JavaScript</title>
</head>
<body>
<h2>JavaScript Objects</h2>
  <script>
    emp =
{id:102,name:"Shyam",salary: 44000}
document.write(emp.id+" " +emp.name +
" " +emp.salary);
    </script>
</body>
</head>
</html>
```

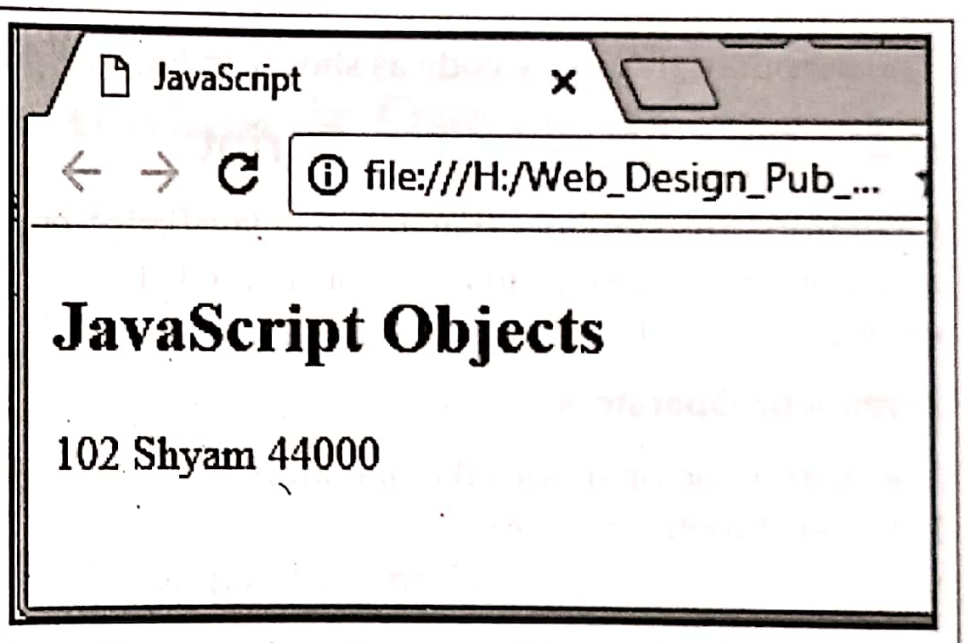


Figure 5.10: Output of Example 10

The output of the above code as shown in Figure 5.10.

Non-Primitive Data type

Array Data Type

An array is a type of object used for storing multiple values in single variable. Each value in an array has a numeric positon, known as its **index**, and it may contain data of any data type-number, strings, objects and even other arrays. The array index starts from 0, so that the first array element is arr[0], arr[1].

The simple way to create an array is by specifying the array elements as a **comma**-separated list enclosed by **square brackets**, as shown in example below.

Example 11:

```
<!DOCTYPE html>
<html>
```

Scanned with OKEN Scanner

```
<head>
<title>JavaScript</title>
</head>
<body>
<h2>JavaScript Array</h2>
   <script>
var fruits = ["Apricot", "Jackfruit",
"Kiwifruit", "Apple"];
var countries = ["US", "Brazil",
"England"];
document.write(fruits[2] + "<br>");
document.write(countries[1]);
</script>
</body>
</head>
</html>
```

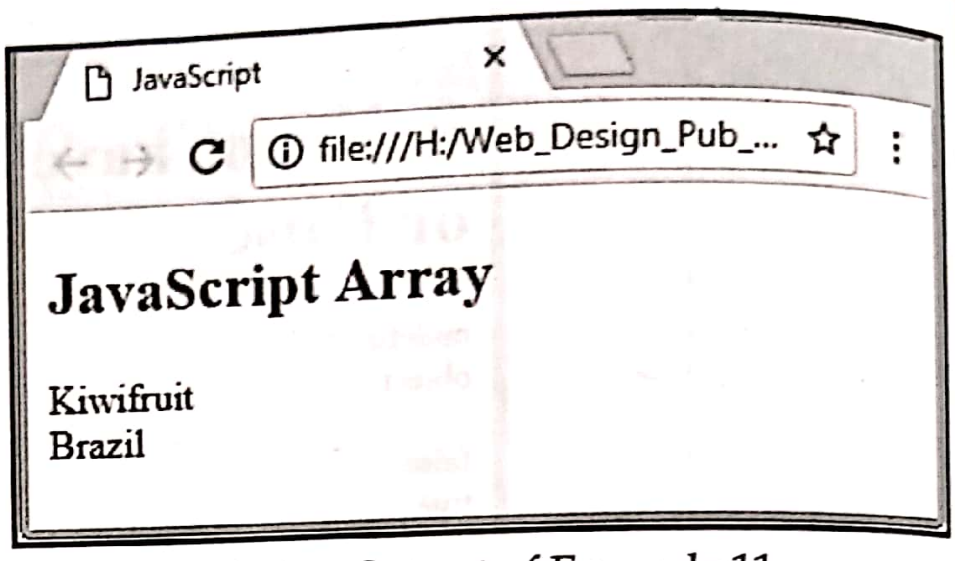


Figure 5.11: Output of Example 11

The output of the above code as shown in Figure 5.11.

5.4 Operators in JavaScript

Operators are symbols that tell the JavaScript engine to perform some sort of actions. *For example,* the addition (+) symbol is an operator that tells JavaScript to add two variables, while the equal to greater than or less than symbols are the operator that tells the JavaScript to compare two variables.

Arithmetic Operators

The arithmetic operator take operand (as a values or variable) and return the single value. Let us complete list of arithmetic operators.

You have numeric value: x =10, y = 5 and result.

Operator	Description	Example	Results
+	Addition	x+y	15
-	Subtraction	x-y	5
*	Multiplication	x*y	50
/	Division	x/y	2
%	Moduls	x%y	0
++	Increment	x++	11 (Increase operand value by one).
--	Decrement	x--	9

The following example will show you these arithmetic operators.

Example 12:

```
<!DOCTYPE html>
<html>
<head>
```

Scanned with OKEN Scanner

```
<title>JavaScript</title>
</head>
<body>
<h2>Arithmetic Operators</h2>
   <script>
    var x = 10;
    var y = 5;
    var z = "Test";
    var linebreak = "<br/>";
document.write("x + y = ");
result = x + y;
document.write(result);
document.write(linebreak);
document.write("x - y = ");
result = x - y;
document.write(result);
document.write(linebreak);
document.write("x / y = ");
result = x + y;
document.write(result);
document.write(linebreak);
document.write("x % y = ");
result = x + y;
document.write(result);
document.write(linebreak);
document.write("x + y + z = ");
result = x + y + z;
document.write(result);
document.write(linebreak);
x = ++x;
document.write("++x = ");
result = ++x;
document.write(result);
document.write(linebreak);
y = --y;
document.write("--y = ");
result = --y;
document.write(result);
document.write(linebreak);
</script>
</body>
</head>
</html>
```

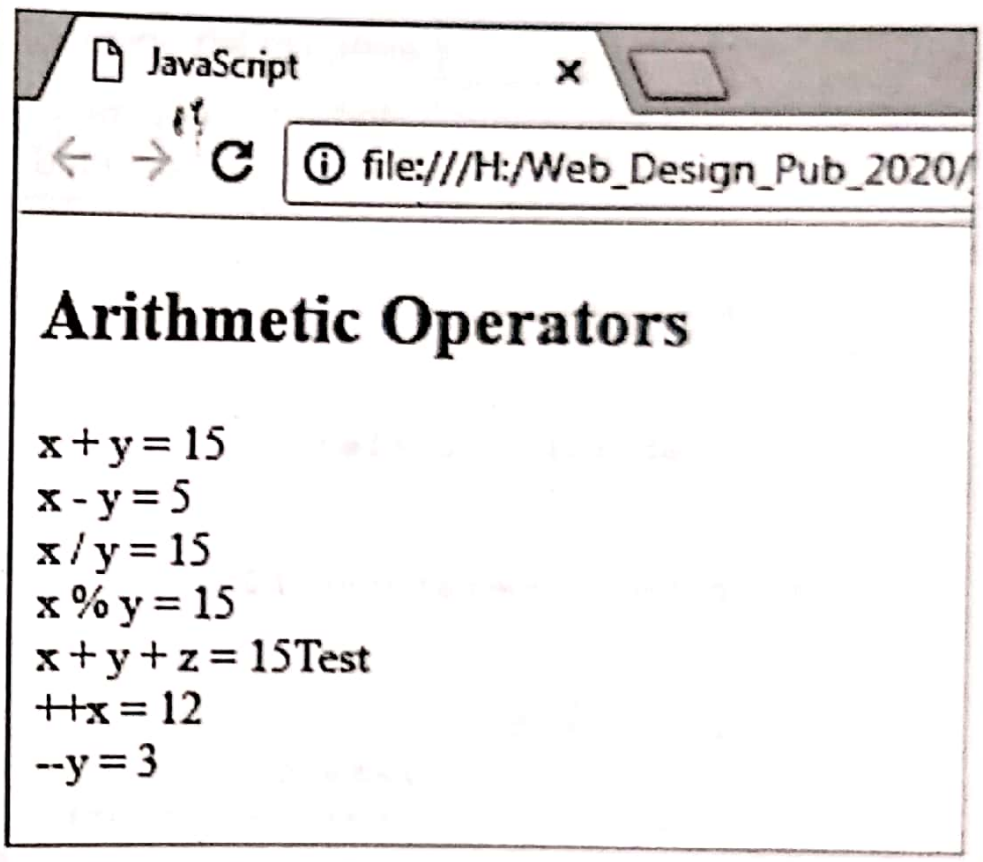


Figure 5.12: Output of Example 12

The output of the above code as shown in Figure 5.12.

Assignment Operators

Assignment operators assign values to left operand based on right operand, equal (=) operates is used to assign a values.

Scanned with OKEN Scanner

You have numeric values x = 20, y = 5 and result.

Operator	Description	Example	Results
=	Assign value from one operand to another operand value.	x = y	x =y
+=	Addition of operand and finally assign to left operand.	x += y	X = x + y
-=	Subtraction of operands and finally assign to left operand.	x -=y	x = x -y
*=	Multiplication of operand and finally assign to left operand.	x *=y	x = x* y
/=	Division of operand and finally assign to left operand.	x/=y	x = x/y
%=	Modulus of operand and finally assign to left operand.	x%=y	x = x%y

Example 13:

```
<!DOCTYPE html>
<html>
<head>
<title>JavaScript</title>
</head>
<body>
<h2>Assignment Operators</h2>
<pre>
<script>
var x = 20; y = 5;
var result = x; //Assignment to
left operand(result) base on right
operand(y).
document.writeln("Assignment:     ",
result);
document.writeln("Addition:       ",
result += y);
document.writeln("Subtraction:    ",
result -= y);
document.writeln("Multiplication:
", result *= y);
document.writeln("Division:       ",
result /=y);
dcoument.writeln("Modulus:        ",
result %= y);
</script>
</pre>
</body>
</head>
</html>
```

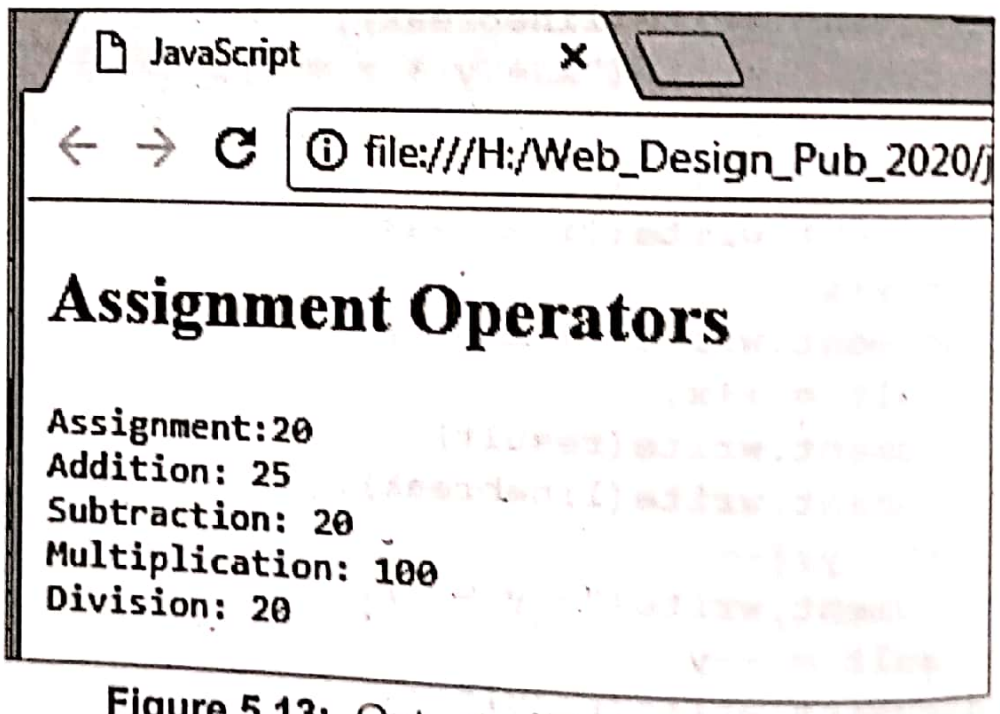


Figure 5.13: Output of Example 13

The output of the above code as shown in Figure 5.13.

Scanned with OKEN Scanner

Comparison Operators

Comparison operator determine the two operands satisfied the given condition. It returns either true or false.

Operator	Name	Example	Results
==	Equal	x ==y	True if x is equal to y.
===	Identical	x ===y	True if x is equal to y, and they are of the same type.
!=	Not equal	x != y	True if x is not equal to y.
!==	Not identical	x !==y	True if x is not equal to y, or the y are not of the same type.
<	Less than	x < y	True if x is less than y.
>	Greater than	x > y	True if x is greater than y.
>=	Greater than or equal to	x >= y	True if x is greater than or equal to y.
<=	Less than or equal to	x <= y	True if x is less than or equal to y.

Example 14:

```
<!DOCTYPE html>
<html>
<head>
<title>JavaScript</title>
</head>
<body>
<h2>Comparison Operators</h2>
<script>
var a = 30;
var b = 40;
var linebreak = "<br/>";
document.write("(a == b) => ");
result = (a == b);
document.write(result);
document.write(linebreak);
document.write("(a === b) => ");
result = (a === b);
document.write(result);
document.write(linebreak);
document.write("(a != b) => ");
result = (a != b);
document.write(result);
document.write(linebreak);
```

(Contd...)

Scanned with OKEN Scanner

```
document.write("(a !== b) => ");
result = (a !== b);
document.write(result);
document.write(linebreak);
document.write("(a < b) => ");
result = (a < b);
document.write(result);
document.write(linebreak);
document.write("(a > b) => ");
result = (a > b);
document.write(result);
document.write(linebreak);
document.write("(a <= b) => ");
result = (a <= b);
document.write(result);
document.write(linebreak);
document.write("(a >= b) => ");
result = (a >= b);
document.write(result);
document.write(linebreak);
</script>
</pre>
</body>
</head>
</html>
```

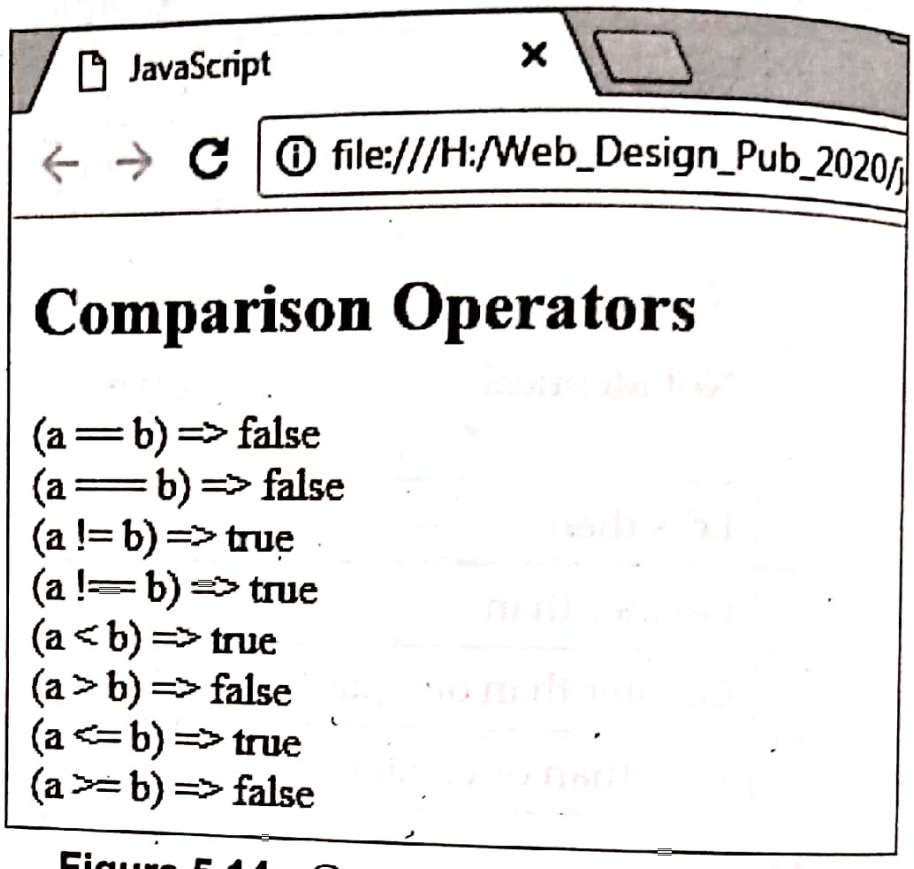


Figure 5.14: Output of Example 14

The output of the above code as shown in Figure 5.14.

Logical Operator

Logical operators are typically used to combine conditional statements.

Operator	Description	Example	Results
&&	And	x && y	True if both x and y are true.
||	Or	x || y	True if either x or y is true.
!	Not	!x	True if x is not true.

The following example will show you how these local operators work.

Example 15:

```
<!DOCTYPE html>
<html>
<head>
<title>JavaScript</title>
</head>
<body>
```

(Contd...)

Scanned with OKEN Scanner

```
<pre>
<script>
  document.writeln("AND:  ", (5 == 5)
&& (10 == 10));
  document.writeln("AND:  ", true  &&
false);
  document.writeln("OR:  ", (5 == 5)
|| (5 == 10));
  document.writeln("OR:  ", true  ||
false);
  document.writeln("AND: ", 5 && 10);
  document.writeln("OR: ", 5 || 10);
  document.writeln("NOT: ", !5)
  document.writeln("NOT: ", !true);
  document.writeln("NOT: ", !false);
</script>
</pre>
</body>
</head>
</html>
```

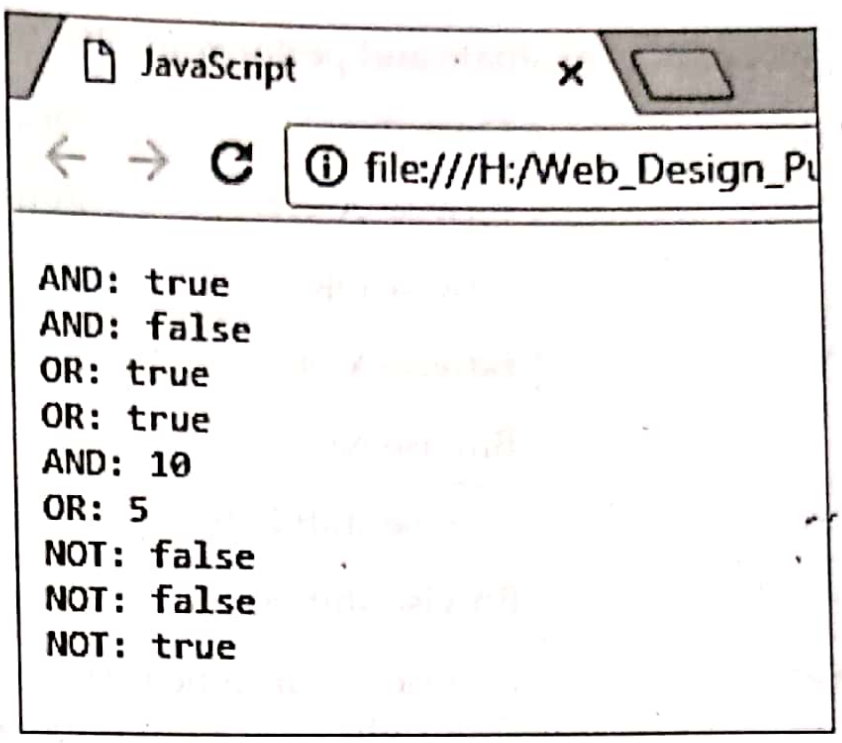


Figure 5.15: Output of Example 15

The output of the above code as shown in Figure 5.15.

String Operators

There are two operators which can also be used be for strings.

Operator	Description	Example	Results
+	Concatenation	str1 + str2	Concatenation of str1 and str2.
+=	Concatenation assignment	str1 += str2	Appends the str2 to the str1.

The following example will show you these string operators in actions.

Example 16:

```
<!DOCTYPE html>
<html>
<head>
<title>JavaScript</title>
</head>
<body>
<h2>String Operators</h2>
<p id="s1"></p>
<script>
  var txt1 = "Lakshmi" ;
  var txt2 = "Sahithi" ;
document.getElementById("s1").innerHTML
= txt1 + " " + txt2;
</script>
</body>
</head>
</html>
```

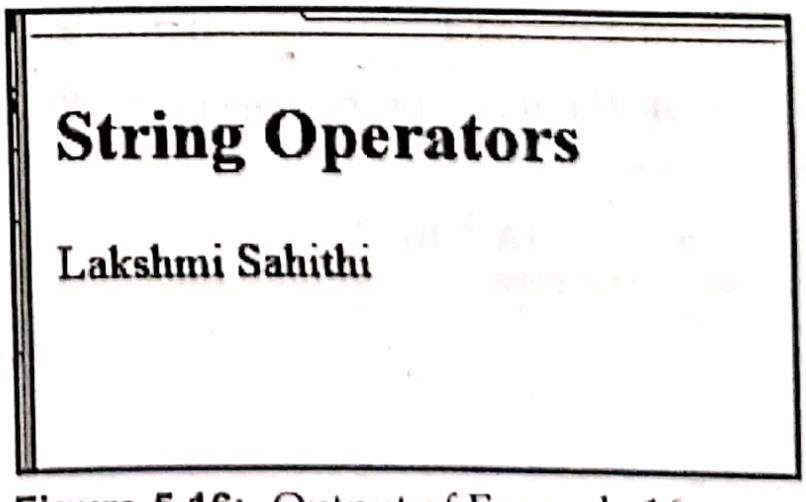


Figure 5.16: Output of Example 16

Scanned with OKEN Scanner

The output of the above code as shown in Figure 5.16.

Bitwise Operators

Bitwise operators evaluate and perform specific bitwise expression.

Sign	Operator	Description
&	Bitwise AND	Return bitwise AND operation for given two operands.
|	Bitwise OR	Return bitwise OR operation for given two operands.
^	Bitwise XOR	Return bitwise XOR operation for given two operands.
~	Bitwise NOT	Return bitwise NOT operation for given operand.
<<	Bitwise shift Left	Return left shift of given operands.
>>	Bitwise shift Right	Return right shift of given operands.
>>>	Bitwise unsigned shift right	Return right shift without consider sign of given operands.

Bitwise AND (&): It accepts two operands. Bitwise (&) return 1 if both the bits are set to 1.

A	B	(A & B) output
0	0	0
0	1	0
1	0	0
1	1	1

Bitwise OR (|): It accepts two operands. Bitwise OR (|) return 1 if any of the operand is set 1 and 0 in any other case.

A	B	(A | B)
0	0	0
0	1	1
1	0	1
1	1	1

Bitwise XOR (^): It accepts two operands. Bitwise XOR (^) returns 1 if both the operand is different and 0 in any other case.

A	B	(A ^ B)
0	0	0
0	1	1
1	0	1
1	1	0

Scanned with OKEN Scanner

Bitwise NOT (~): It is unary operator i.e. accepts single operands. Bit-wise (~) flips the bits i.e. 0 becomes 1 and 1 becomes 0.

A	(~A)
0	1
1	0

Left Shift (<<): It accepts two operands. The first operator specified the number and the second operator specified the number of bits to shift. Each bits are shifted towards left and 0 bits are added from the right. The excess bits from the left are discarded.

Shift Propagating Right Shift (>>): It accepts two operands. The first operand specifies the number and the second operand specifies the number of bits to shift. Each bit shifted towards the right, the overflowing bits are discarded. Right shift bits which are added from the left depends upon the sign of the number (i.e. 0 if positive and 1 if negative).

Unsigned Right Shift (>>>): It accepts two operands. The first operand specified the number and the second operand specifies the number of bits to shift. Each bit is shifted towards right, the overflowing bits are discarded. 0 bit is added from the left so its zero fill right shift.

The following code to implementation of the bitwise operator.

Example 17:

```
<!DOCTYPE html>
<html>
<head>
<title>JavaScript</title>
</head>
<body>
<h2>Bitwise Operators</h2>
<script>
   var a = 6;
   var b = 1;
  document.write("A & B = " + (a & b)
+ "<br>");
  document.write("A | B = " + (a | b)
+ "<br>");
  document.write("~A = " + (~a) +
"<br>");
  document.write("A >> B = " + (a >>
b) + "<br>");
  document.write("A >>> B = " + (a
>>> b) + "<br>");
  document.write("A << B = " + (a <<
b) + "<br>");
</script>
</body>
</head>
</html>
```

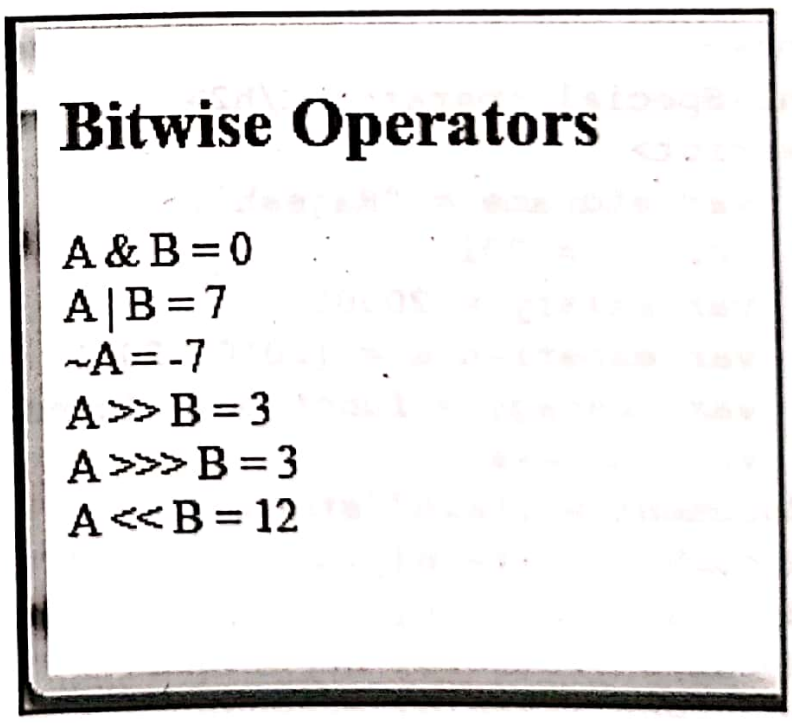


Figure 5.17: Output of Example 17

Scanned with OKEN Scanner

Special Operators

The following operators are known as JavaScript Special operators.

Operator	Description
delete	Delete operator deleted a property from the object.
instanceof	Checks if the object is an instance of given type.
typeof	Checks the type of object.
void	It discards the expression return value.
new	It creates an instance (object).
yield	Check what is returned in a generator by the generator's iterator.
(?)	Comma operator allows multiple expression to be evaluated as single statement.

typeof

The typeof operator return valid data type identifiers as a string of given expression, typeof operator return six possible values: "string", "number", "Boolean", "object", "function" and "undefined".

Example 18:

```
<!DOCTYPE html>
<html>
<head>
<title>JavaScript</title>
</head>
<body>
<pre>
<h2>Special operators</h2>
<script>
   var stdname = "Rajesh";
   var id = 201;
   var salary = 20000;
   var experience = [2010, 2011, 2012, 2013, 2014];
   var message = function() {console.log("Salary hiked by 20%"); }
   var address;
 document.writeln("stdname    : ", typeof(stdname));
 document.writeln("id         : ", typeof id);
 document.writeln("salary     : ", typeof salary);
 document.writeln("experience : ", typeof experience);
 document.writeln("message    : ", typeof message);
 document.writeln("address    : ", typeof address);
</script>
</pre>
</body>
</head>
</html>
```

The output of the above code as shown in Figure 5.17.

Scanned with OKEN Scanner

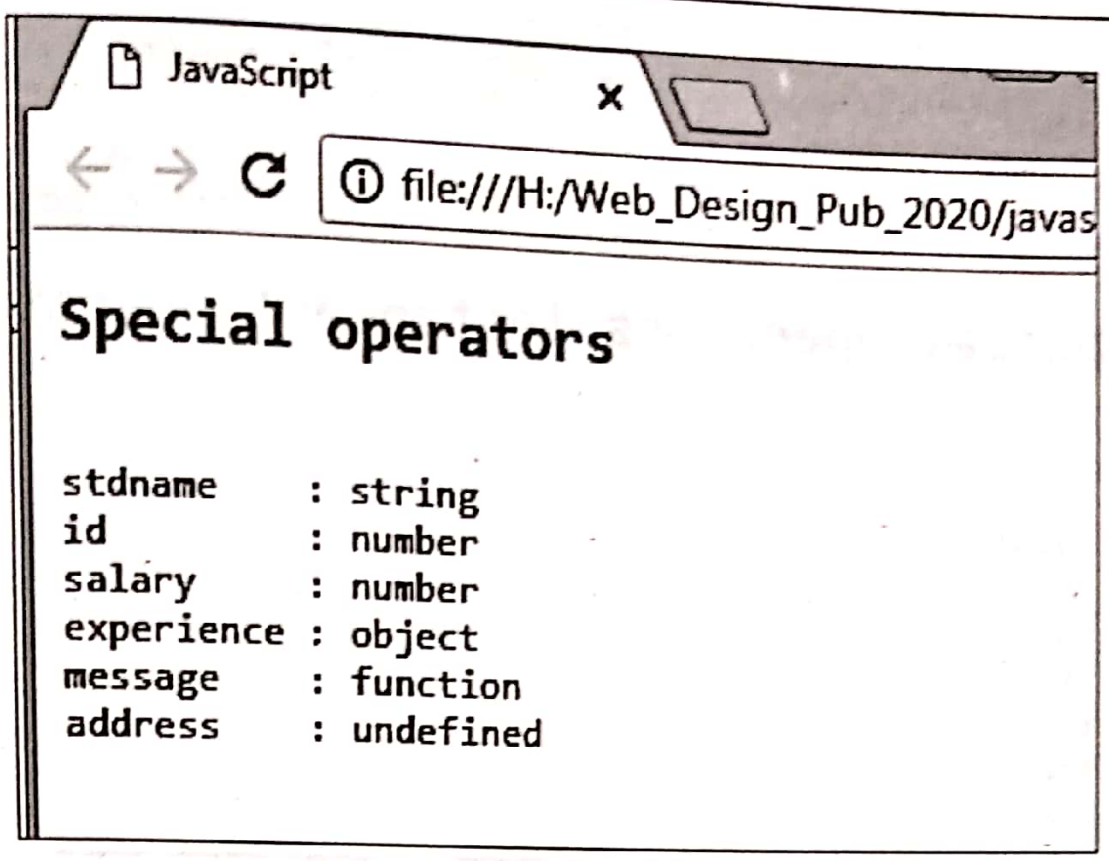


Figure 5.18: Output of Example 18

instanceof

The instanceof indicate Boolean result, return true, if object is an instance of specific class.

Example 19:

```
<!DOCTYPE html>
<html>
<head>
<title>JavaScript</title>
</head>
<body>
<pre>
<h2>Special operators instanceof operator</h2>
<script>
   var num1 = new Number(10);
   document.writeln("1:", num1 instanceof Number);
   var num2 = 5;
   document.writeln("2:", num2 instanceof Number);
   document.writeln("3:", true instanceof Boolean);
   document.writeln("4:", 0 instanceof Number);
   document.writeln("5:", "" instanceof String);
   document.writeln("6:", new Boolean(true) instanceof Boolean);
   document.writeln("7:", new Number(0) instanceof Number);
   document.writeln("8:", new String("") instanceof String);
</script>
</pre>
</body>
</head>
</html>
```

The output of the above code as shown in Figure 5.19.

Scanned with OKEN Scanner

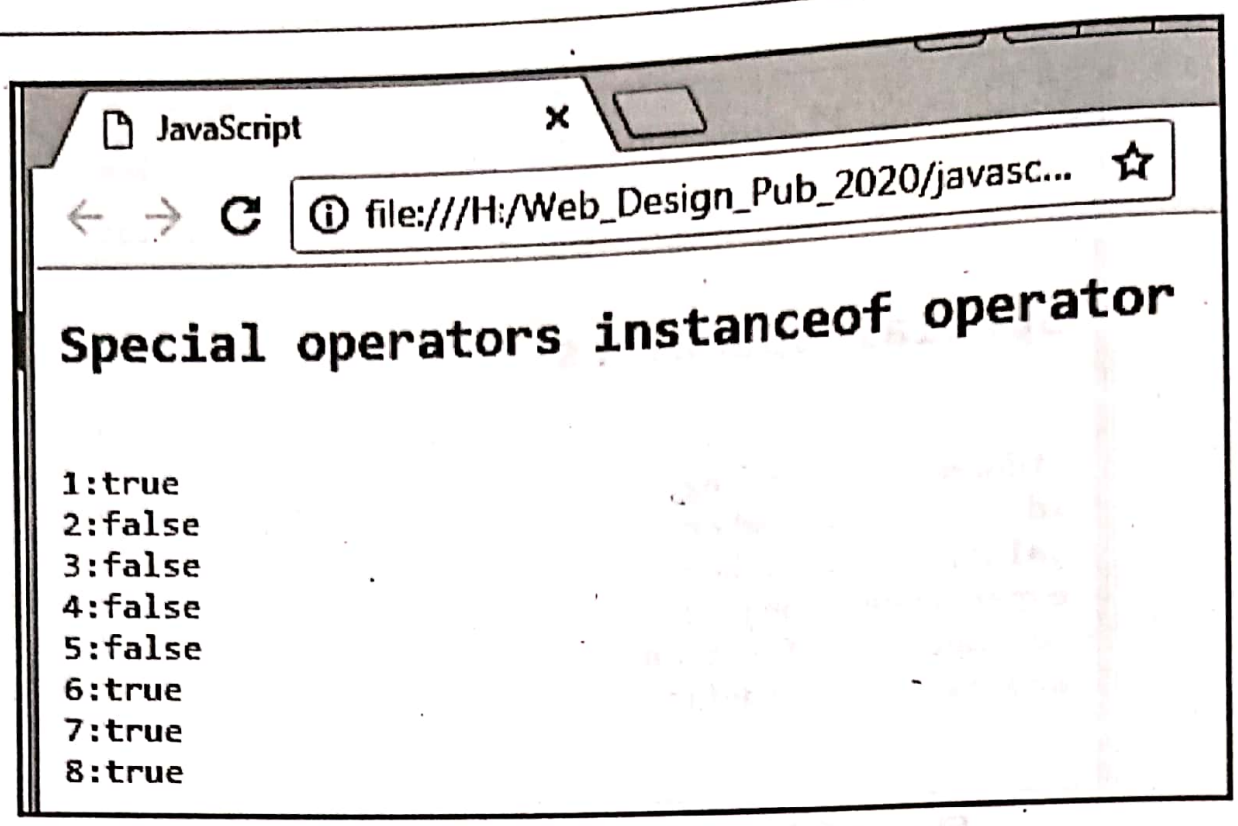


Figure 5.19: Output of Example 19

5.5 Conditional Statements in JavaScript

Conditional statements are used to decide the flow of execution based on different conditions. If a condition is true, you can perform one action and if the condition is false, you can perform another action.

5.5.1 Different Types of Conditional Statements

There are mainly four types of conditional statements in JavaScript.

a. **if statement:** Use this statement if you want to execute a set of code when a condition is true.
b. **if...else statement:** Use this statement if you want to select one of two sets of lines to execute.
c. **If...Else If...Else statement:** To specify a new condition to test, if the first condition is false.
d. **switch statement:** Use this statement if you want to select one of many sets of lines to execute.

if statement

The if statement is the fundamental control statement that allows JavaScript to make decisions and execute statements conditionally.

Syntax:

```
if (condition)
{
code to be executed if condition is true
}
```

You can use "If statement" if you want to check only a specific condition.

Example 20:

```
<!DOCTYPE html>
<html>
<head
<title>If statements</title>
<script type="text/javascript">
```

(Contd...)

Scanned with OKEN Scanner

```
var age = prompt("Please enter your
age");
      if(age<5)
      document.write("You are child
<br />");
      if(age>=5)
      document.write("You are NOT
child <br />");
      </script>
</head>
<body>
</body>
</html>
```

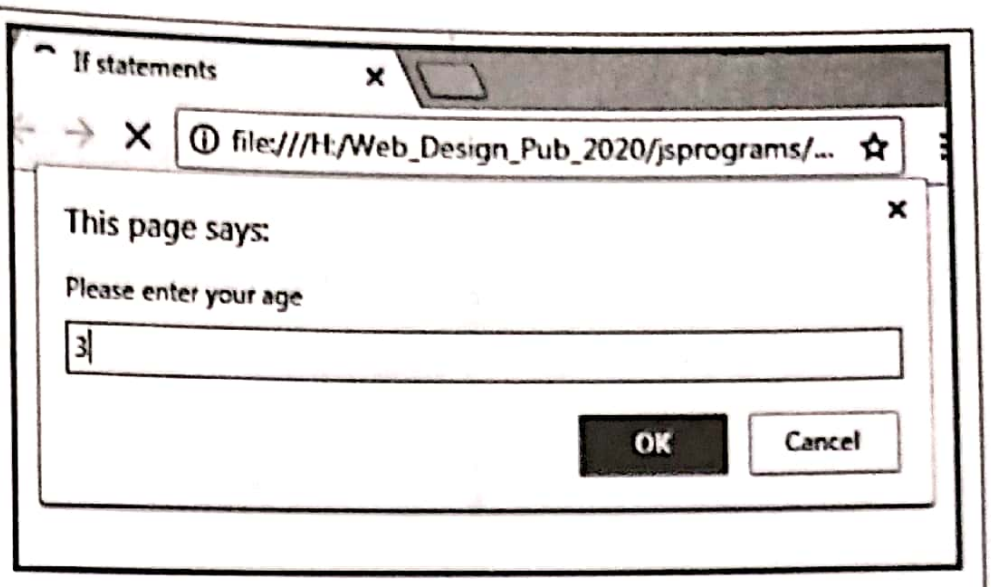


Figure 5.20: Output of Example 20

The output of the above code as shown in Figure 5.20.

The else Statement

Use the else statement to specify a block of code to be executed if the condition is false.

Syntax:-

```
if (condition) {
      //  block of code to be executed if the condition is true
} else {
  //  block of code to be executed if the condition is false
}
```

If the hour is less than 18, create a "Good day" greeting, otherwise "Good evening".

Example 21:

```
<!DOCTYPE html>
<html>
<body>
<p>Click the button to display a time-based greeting:</p>
<button onclick="myFunction()">Click it</button>
<p id="d1"></p>
<script>
function myFunction() {
  var hour = new Date().getHours();
  var greeting;
  if (hour > 18) {
    greeting = "Good day";
  } else {
    greeting = "Good evening";
  }
  document.getElementById("d1").innerHTML = greeting;
}
</script>
</body>
</html>
```

The output of the above code as shown in Figure 5.21.

Scanned with OKEN Scanner

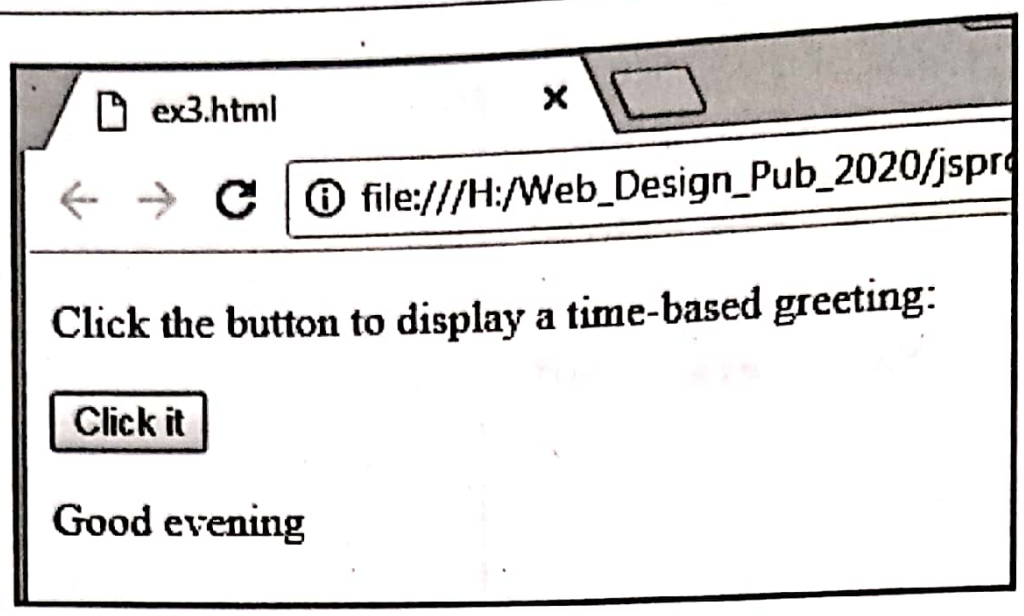


Figure 5.21: Output of Example 21

If...Else If...Else statement

The if...else if... statement is an advanced form of if...else that allows JavaScript to make a correct decision out of several conditions.

Syntax

```
The syntax of an if-else-if statement is as follows -
if (condition 1) {
   lines of code  to be executed if condition 1 is true
} else if (condition 2) {
   Lines of code to be executed if condition 2 is true
}
else {
   lines of code to be executed if condition 1 is false and condition 2 is
false
}
```

The following example 22 to learn how to implement an if-else-if statement in JavaScript.

Example 22:

```
<!DOCTYPE html>
<html>
<head>
<script type="text/javascript">
 var one = prompt("Enter the first number");
   var two = prompt("Enter the second number");
      one = parseInt(one);
      two = parseInt(two);
         if (one == two)
           document.write(one + " is equal to " + two + ".");
        else if (one<two)
            document.write(one + " is less than " + two + ".");
        else
            document.write(one + " is greater than " + two + ".");
</script>
</head>
</html>
```

The output of the above code as shown in Figure 5.22.

Scanned with OKEN Scanner

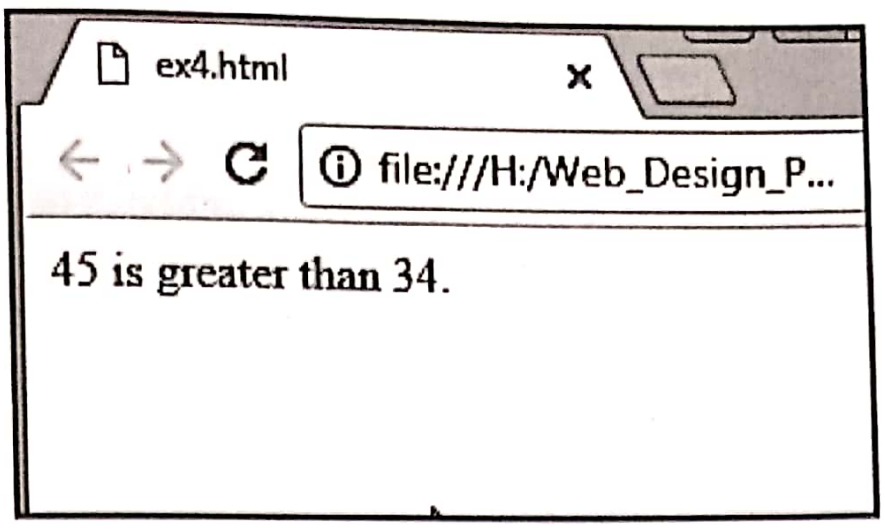


Figure 5.22: Output of Example 22

JavaScript Switch

The JavaScript switch statement is used to execute one code from multiple expressions. It is just like else if statement that you have learned in previous page. But it is convenient than if..else..if because it can be used with numbers, characters etc.

Syntax:

```
switch(expression) {
case value1:
 code to be executed;
 break;
case value2:
 code to be executed;
 break;
......
default:
 code to be executed if above values are not matched;
}
```

Let us see the simple example of switch statement in JavaScript.

Example 23:

```
<!DOCTYPE html>
<html>
<body>
<script>
var grade='B';
var result;
switch(grade){
case 'A':
result="A Grade";
break;
case 'B':
result="B Grade";
break;
```

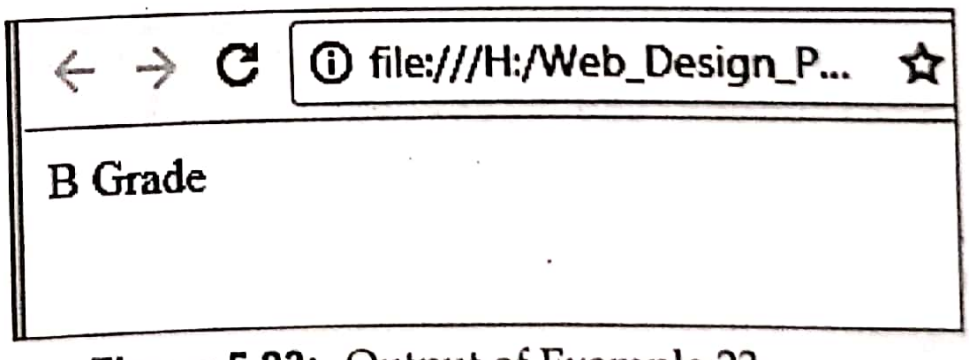


Figure 5.23: Output of Example 23

Scanned with OKEN Scanner

```
case 'C':
result="C Grade";
break;
default:
result="No Grade";
}
document.write(result);
</script>
</body>
</html>
```

The output of the above code as shown in Figure 5.23.

5.6 Display Popup Message Box

JavaScript popup provides different built-in functions to display popup messages for different purposes e.g. to display a simple message or display a message and take user's confirmation on it or display a popup to take a user's input value.

5.6.1 Alert Box

To use alert() function it display a popup message to the user. This popup will have **OK** button to close the popup.

Syntax

```
Window.alert("sometext goes here");
```

Example 24:

```
<!DOCTYPE html>
<html>
<head>
<title>JavaScript</title>
</head>
<body>
<h2>Display Popup Message Box</h2>
  <button  onclick="myFunction()">Open
Alert Box</button>
<script>
   function myFunction() {
 alert("Check the Password");
}
</script>
</body>
</head>
</html>
```

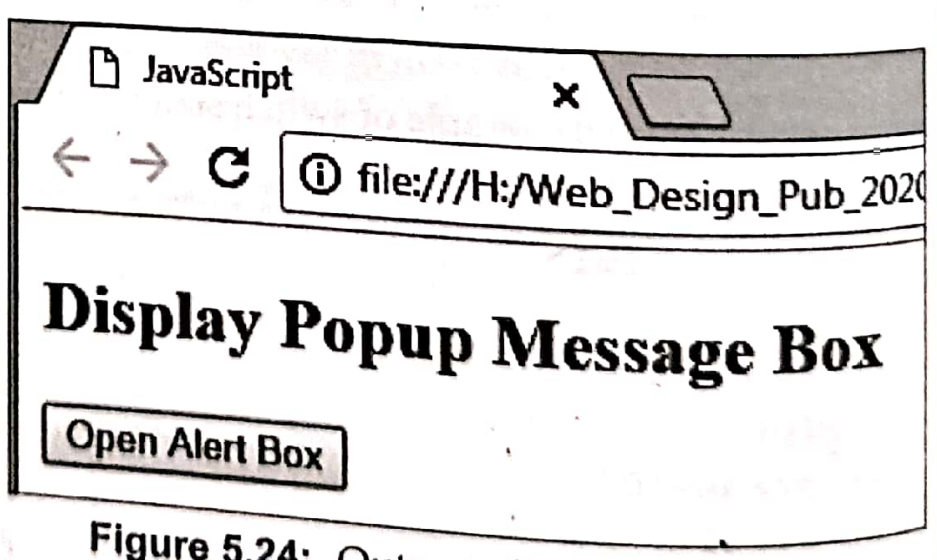


Figure 5.24: Output of Example 24

The output of the above code as shown in Figure 5.24.

5.6.2 Confirm Box

A confirm box is often used if you want the use to verify or accept something. When a confirm box pops up,

Scanned with OKEN Scanner

the user will have to click either OK or Cancel to proceed. If the user click OK, the box returns true. If the user click Cancel, the box returns false.

Syntax:

```
window.confirm("your query here")
```

Example 25:

```
<!DOCTYPE html>
<html>
<head>
<title>JavaScript</title>
</head>
<h2>JavaScript Confirm Box</h2>
  <style>
    h1 {
      color:green;
    }
</style>
<body>
<h3>Confirm Box</h3>
 <button onclick="Confirm()">Do you want to save changes</button>
 <p id="s1"</p>
 <script>
    function Confirm() {
      var x;
        if (confirm("Saved Data successfully!") == ture) {
             x = "OK pressed!";
        }  else {
             x = "Cancel!";
        }
        document.getElementById("s1").innerHTML = x;
      }
</script>
</body>
</head>
</html>
```

The output of the above code as shown in Figure 5.25.

5.6.3 Prompt Box

JavaScript Prompt Box is often use to get a value from user (means input to the value from user) and after value using the specify purpose can be process. *For example*, you want to calculate EMI based on users' preferred tenure of loan. For this kind of scenario, use JavaScript built-in function prompt().

Prompt function takes two string parameters. **First parameter** is the message to be displayed and **second parameter** is the default value, which will be in input text when the message is displayed.

Scanned with OKEN Scanner

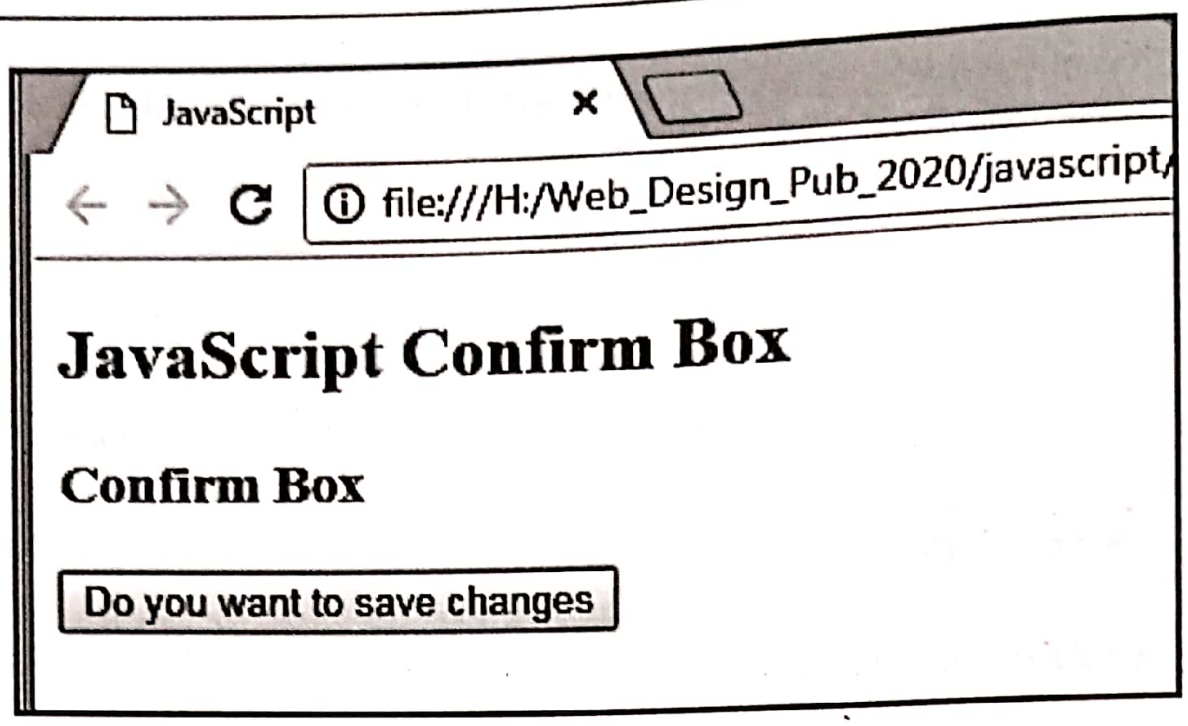


Figure 5.25: Output of Example 25

Syntax:

```
prompt("your Prompt here")
```

Example 26:

```
<!DOCTYPE html>
<html>
<body>
      <h1>JavaScript Pop messages</h1>
      <p id="msg"></p>
       <script>
          var freehold = prompt("Please enter preferred freehold in years",
"15");
            document.getElementById("msg").innerHTML = "You have entered " +
freehold + " years";
 </script>
</body>
</html>
```

The output of the above code as shown in Figure 5.26.

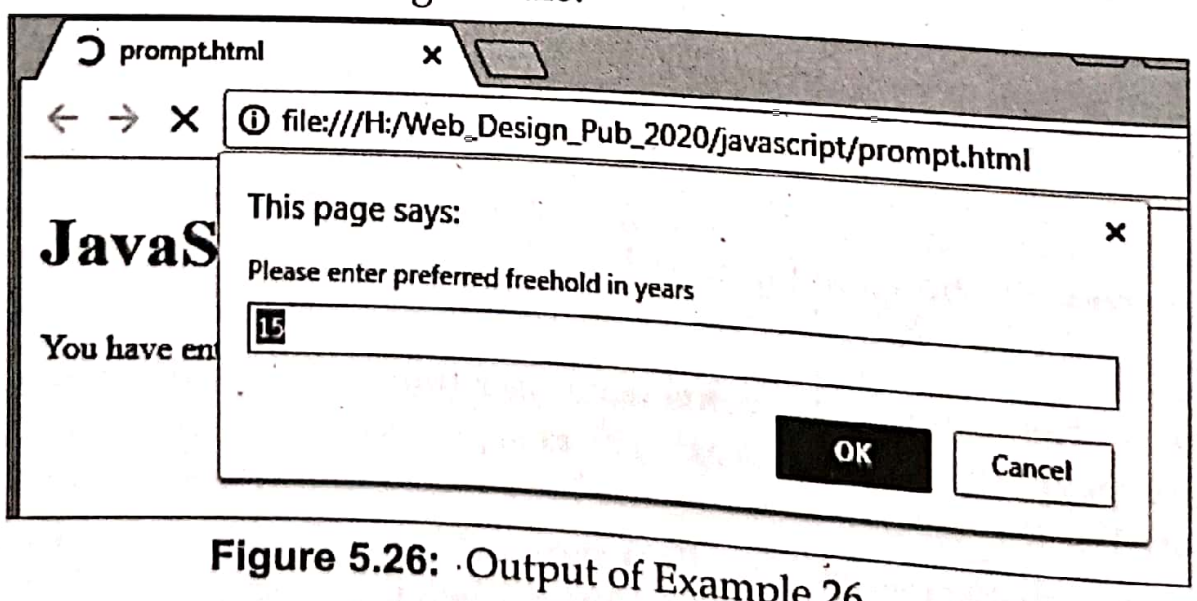


Figure 5.26: Output of Example 26

5.7 JavaScript Events

An event is something that happens when user interact with the web page, such as when he clicked a link or

button, entered text into an input box or **textarea**, made selection in a select box, pressed key on the keyboard, moved the mouse pointer, submits a form, etc. In some cases, the browser itself can trigger the events, such as the page load and unload events.

When an event occurs, you can use a JavaScript event handler (or an event listener) to detect them and perform specific task or set of tasks. By convention, the names for event handlers always begin with the word on, so an event handler for the click event is called **onclick**, similarly an event handler for the load event is called **onload**, event handler for the blur event is called **onblur**, and so on.

There are several ways to assign an event handler. The simplest way is to add them directly to the start tag of the HTML elements using the special event-handler attributes. *For example*, to assign a click handler for a button element, you can use **onclick** attribute.

5.7.1 HTML Events

Javascript has events to provide a dynamic interface to a webpage. These events are hooked to elements in the **Document Object Model(DOM)**.

Here is a list of some common HTML events:

Event	Description
onchange	An HTML element has been changed.
onclick	The user clicks an HTML element.
onmouseover	The user moves the mouse over an HTML element.
onmouseout	The user moves the mouse away from and HTML element.
onkeydown	The user pushes a keyboard key.
Onload	The browser has finished loading the page.

There are some JavaScript events:

onclick events: This is a mouse event and provokes any logic defined if the user clicks on the element it is bound to.

Example 27:

```
<!doctype html>
<html>
<head>
<script>
function hiThere() {
alert('Hi there!');
  }
</script>
</head>
<body>
<button type="button" onclick="hiThere()">Click me event</button>
</body>
</html>
```

The output of the above code as shown in Figure 5.27.

Scanned with OKEN Scanner

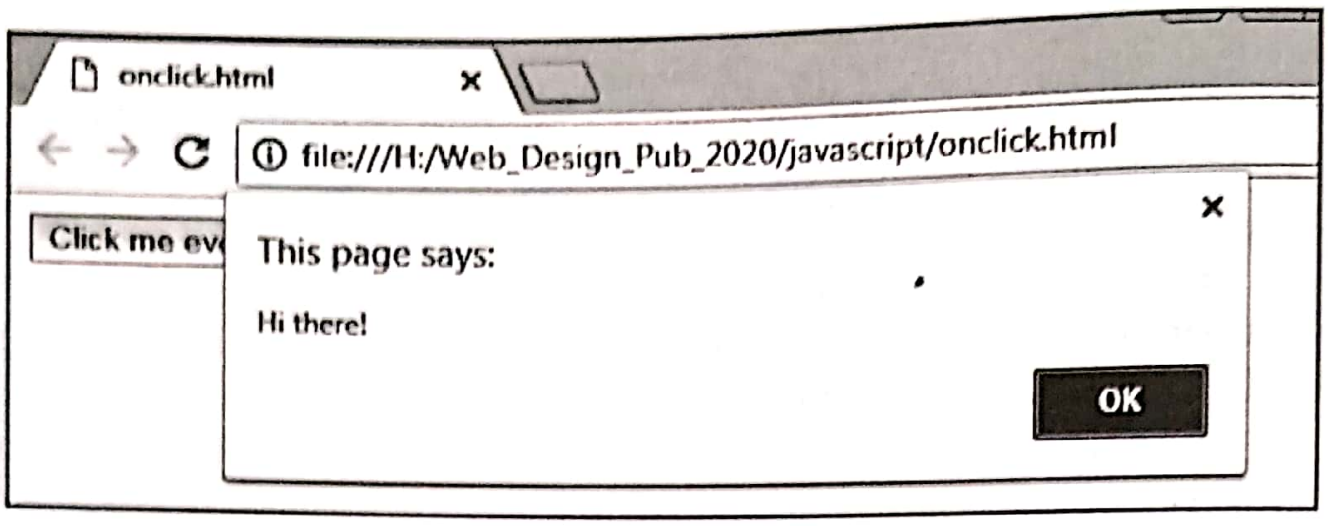


Figure 5.27: Output of Example 27

Onkeyup and onkeydown event: This event is a keyboard event and executes instructions whenever a key is released after pressing.

Example 28:

```
<!DOCTYPE html>
<html>
<head>
<script>
function color(color) {

document.forms[0].myInput.style.background
= color;
}
</script>
</head>
<body>
<form>
Write a message: <br>
<input                                    type="text"
Onkeydown="color('yellow')"
  Onkeyup="color('white')" name="myInput">
</form>
</body>
</html>
```

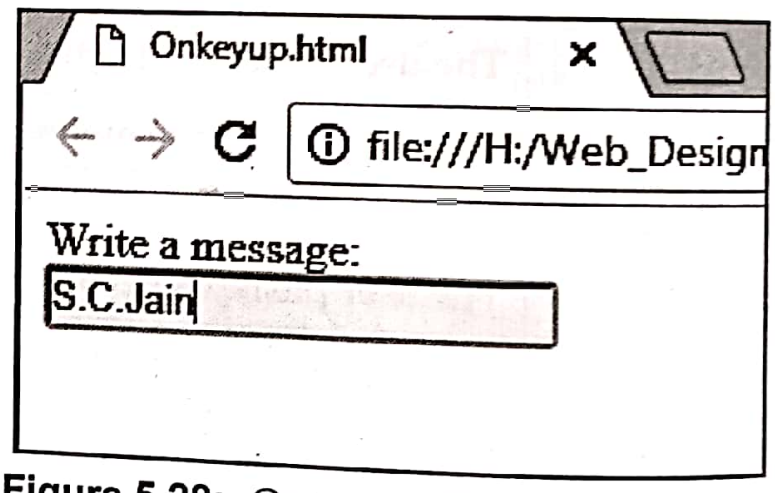


Figure 5.28: Output of Example 28

The output of the above code as shown in Figure 5.28.

onmouseover event: This event corresponds to hovering the mouse pointer over the element and its children, to which it is bound to.

Example 29:

```
<!DOCTYPE html>
<html>
<head>
  <h1  onmouseover="style.color='Blue'"  onmouseout="style.color='red'">Mouse
over the text</h1>
  </body>
</html>
```

The output of the above code as shown in Figure 5.29.

Scanned with OKEN Scanner

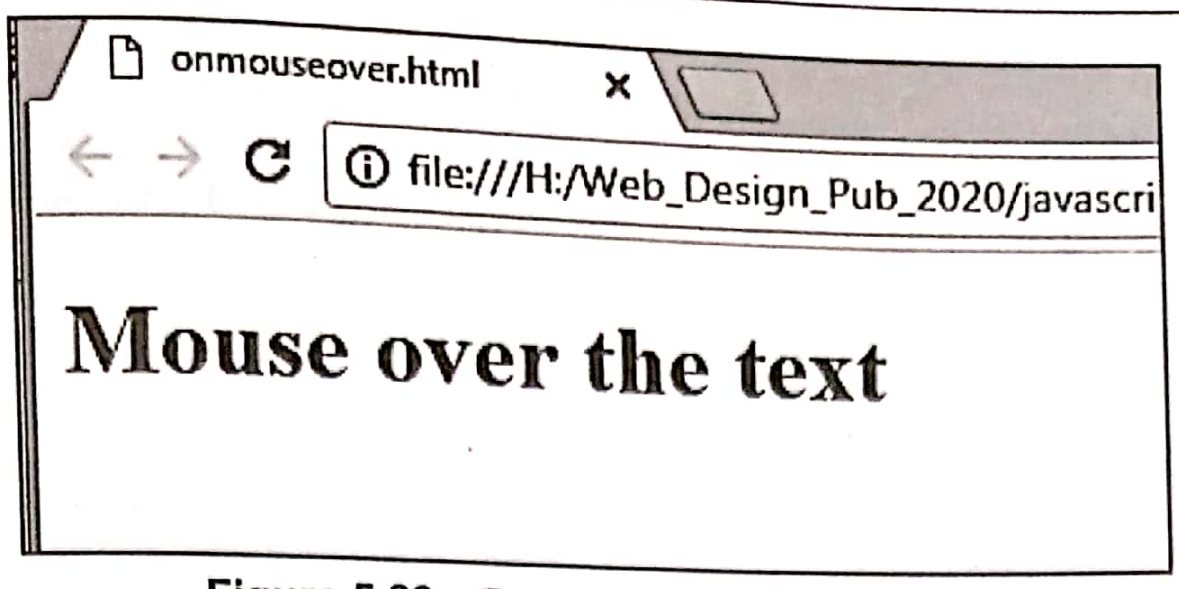


Figure 5.29: Output of Example 29

onmouseout event: Whenever the mouse cursor leaves the element which handles a **mouseout** event, a function associated with it is executed.

Contextmenu Event (oncontextmenu)

The **contextmenu** event occurs when a user clicks the right mouse button on an element to open a context menu. You can handle **a contextmenu** event with an **oncontextmenu** event handler.

The following example will show an alert message when you right-click on the elements.

Example 30:

```
<!DOCTYPE html>
<html>
  <head></head>
  <body>
<button   type="button"   oncontextmenu="alert('You   have   right-clicked   a
button!');">
          Right Click on Me</button>
    <a href="#" oncontextmenu="alert('You have right-clicked a link!');">Right
Click on Me</a>
</body>
</html>
```

The output of the above code as shown in Figure 5.30.

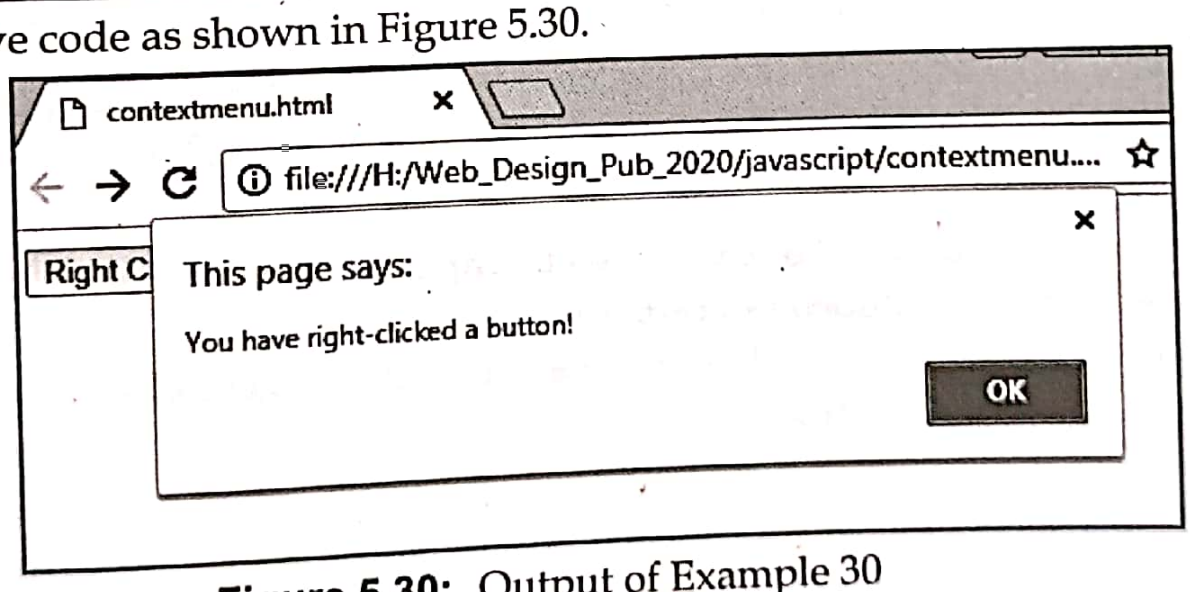


Figure 5.30: Output of Example 30

5.8 Form Events

A form event is fired when a form control receives or loses focus or when the user modifies a form control value such as by typing text in a text input, select any option in a select box etc. Here, there some most important form events and their event handler.

Scanned with OKEN Scanner

5.8.1 Focus Event (onfocus)

The focus event occurs when the user gives focus to an element on a web page. You can handle the focus event with the **onfocus** event handler. The following example will highlight the background of text input in yellow color when it receives the focus.

Example 31:

```
<!doctype html>
<html>
  <head>
  <body>
<script>
        function highlightInput(elm){
            elm.style.background     =
"orange";
        }
    </script>
    <input                          type="text"
onfocus="highlightInput(this)">
    <button
type="button">Button</button>
</body>
</head>
</html>
```

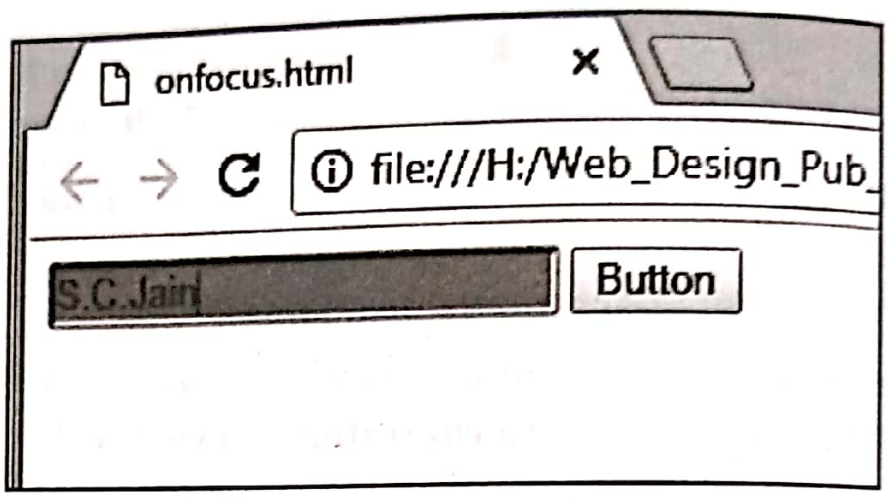


Figure 5.31: Output of Example 31

The output of the above code as shown in Figure 5.31.

Blur Event (onblur)

The blur event occurs when the user takes the focus away from a form element or a window. You can handle the blur event with the **onblur** event handler. The following example will show you an alert message when the text input element loses focus.

Example 32:

```
<!doctype html>
<html>
  <head>
  <body>
   <input type="text" onblur="alert('Text input loses focus!')">
    <button type="button">Submit</button>
      <p>Write something in the input box and then click elsewhere
       in the document body</p>
</body>
</head>
</html>
```

The output of the above code as shown in Figure 5.32

Scanned with OKEN Scanner

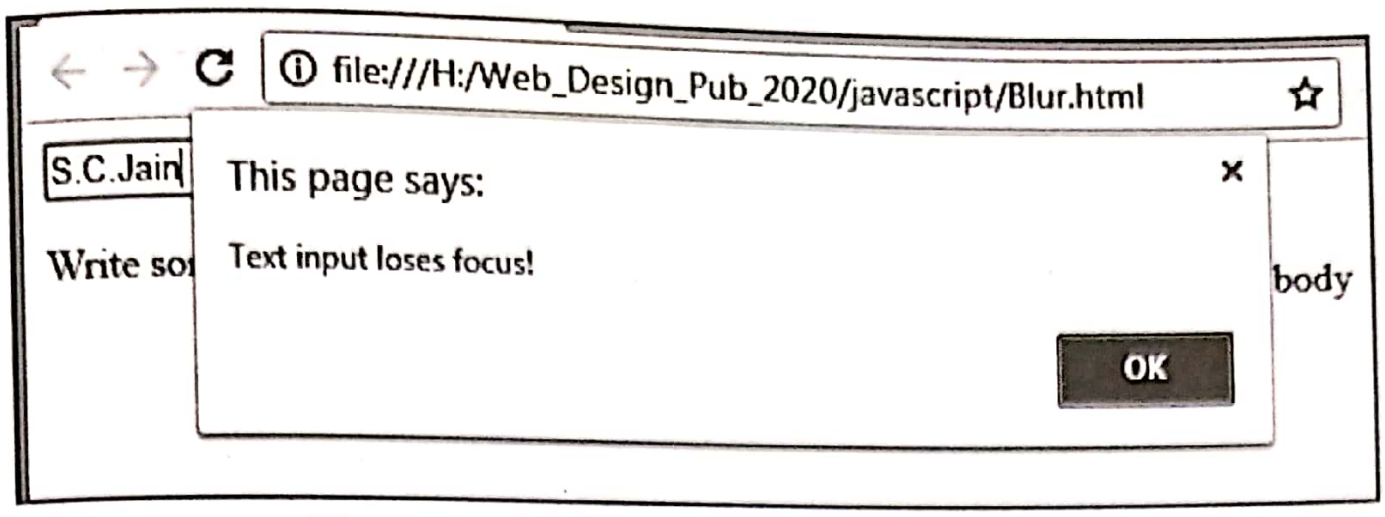


Figure 5.32: Output of Example 32

Change Event (onchange)

The change event occurs when a user changes the value of a form element.

You can handle the change event with the **onchange** event handler. The following example will show you an alert message when you change the option in the select box.

onchange event: This event detects the change in value of any element listing to this event.

Example 33:

```
<!doctype html>
<html>
  <head></head>
  <body>
    <input onchange="alert(this.value)" type="number">
  </body>
</html>
```

The output of the above code as shown in Figure 5.33.

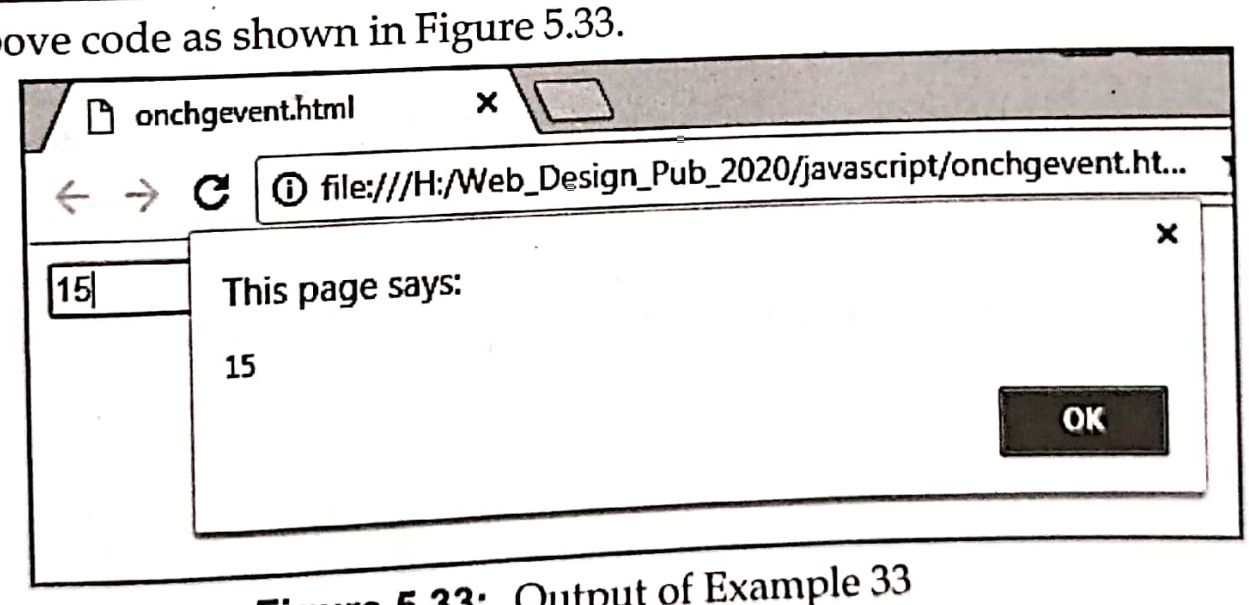


Figure 5.33: Output of Example 33

Submit Event (onsubmit)

The submit event only occurs when the user submits a form on a web page.

You can handle the submit event with the **onsubmit** event handler. The following example will show you an alert message while submitting the form to the server.

Example 34:

```
<!DOCTYPE html>
<html>
  <head>
```

Scanned with OKEN Scanner

```
  <body>
      <form action="/examples/html/action.php" method="post"
          onsubmit="alert('Form data will be submitted to the server!');">
        <label>First Name:</label>
        <input type="text" name="first-name" required>
        <input type="submit" value="Submit">
    </form>
</body>
</html
</body>
</head>
</html>
```

The output of the above code as shown in Figure 5.34.

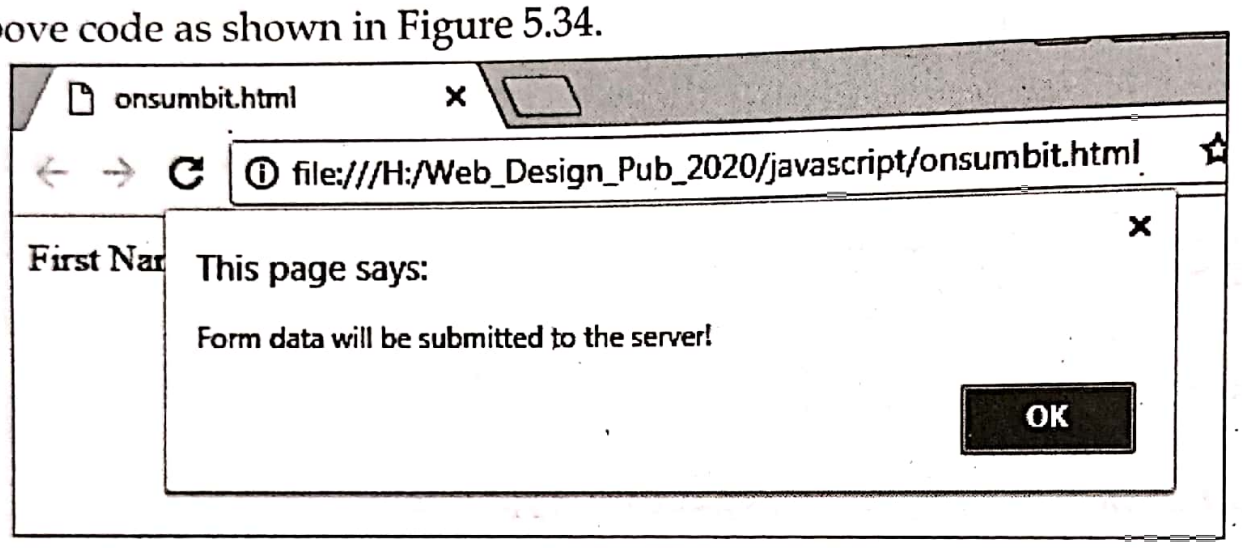


Figure 5.34: Output of Example 34

Window Events

Events are also triggered in situations when the page has loaded or when user resize the browser window, etc. Here are some most important document/Window events and their event handler.

Load Event (onload)

The load event occurs when a web page has finished loading in the web browser.

You can handle the load event with the **onload** event handler. The following example will show you an alert message as soon as the page finishes loading.

Example 35:

```
<!doctype html>
<html>
  <head>
     <body onload="window.alert('Page
is loaded successfully!');">
</body>
  <h3>Event fires after the page
loading finished</h3>
</head>
</html>
```

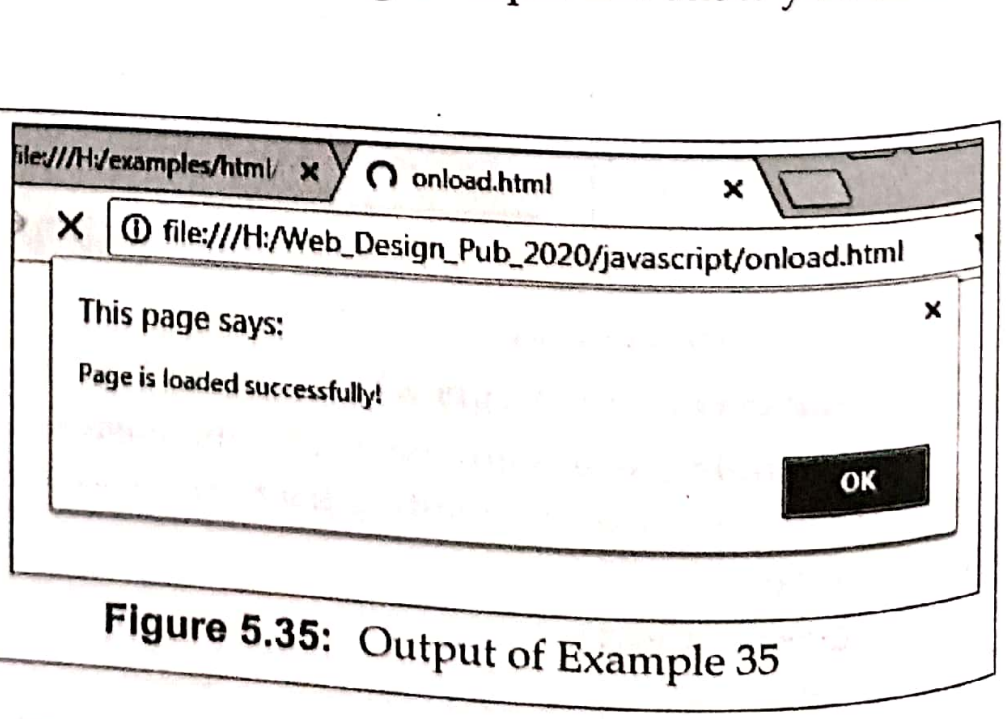


Figure 5.35: Output of Example 35

The output of the above code as shown in Figure 5.35.

5.9 Basic Form Validations in JavaScript

Forms are used in webpages for the user to enter their required details that are further send it to the server for processing. A form is also known as **web form** or HTML form. Web forms have become an essential part of web applications. It is often used to collect user's information such as name, e-mail address, location, age, and so on.

Form Validation with JavaScript

The form validation process typically consists of two parts– the required fields validation which is performed to make sure that all the mandatory fields are filled in, and the data format validation which is performed to ensure that the type and format of the data entered in the form is valid.

5.9.1 Creating the HTML Form

Let us first create a simple HTML form that you will validate on client-side using JavaScript when the user clicks on the submit button. Let us create an HTML file named "application-form.html" and place the following code in it, then save it in your system.

Example 36:

```
<!DOCTYPE html>
<head>
    <title>Simple HTML Form</title>
    <link rel="stylesheet" href="style.css">
    <script src="h:/Web_Design_Pub_2020/javascript/validator.js"></script>
</head>
<body>
<h1 style="text-align:center"> REGISTRATION FORM </h1>
<form name="RegForm" action="/submit.php" onsubmit="return validateForm()"
method="post">
    <p>Name: </b><input type="text" size=65 name="Name"> </p>
    <p>Address:</b> <input type="text" size=65 name="Address"></p>
    <p>E-mail Address:</b> <input type="text" size=65 name="EMail"></p>
     <p>Password:</b><input type="text" size=65 name="Password"> </p>
    <p>Telephone:</b> <input type="text" size=65 name="Telephone"> </p>
    <p>Select your Course :</b>
        <select type="text" value="" name="Subject">
            <option>BTECH</option>
            <option>BBA</option>
            <option>BCA</option>
            <option>B.COM</option>
            </select></p><br>
    <p><b>Comments:</b> <textarea cols="40" name="Comment"> </textarea></p>
    <p><input type="submit" value="send" name="Submit">
        <input type="reset" value="Reset" name="Reset">
    </p>
</form>
</body>
```

The output of the above code as shown in Figure 5.36.

Scanned with OKEN Scanner

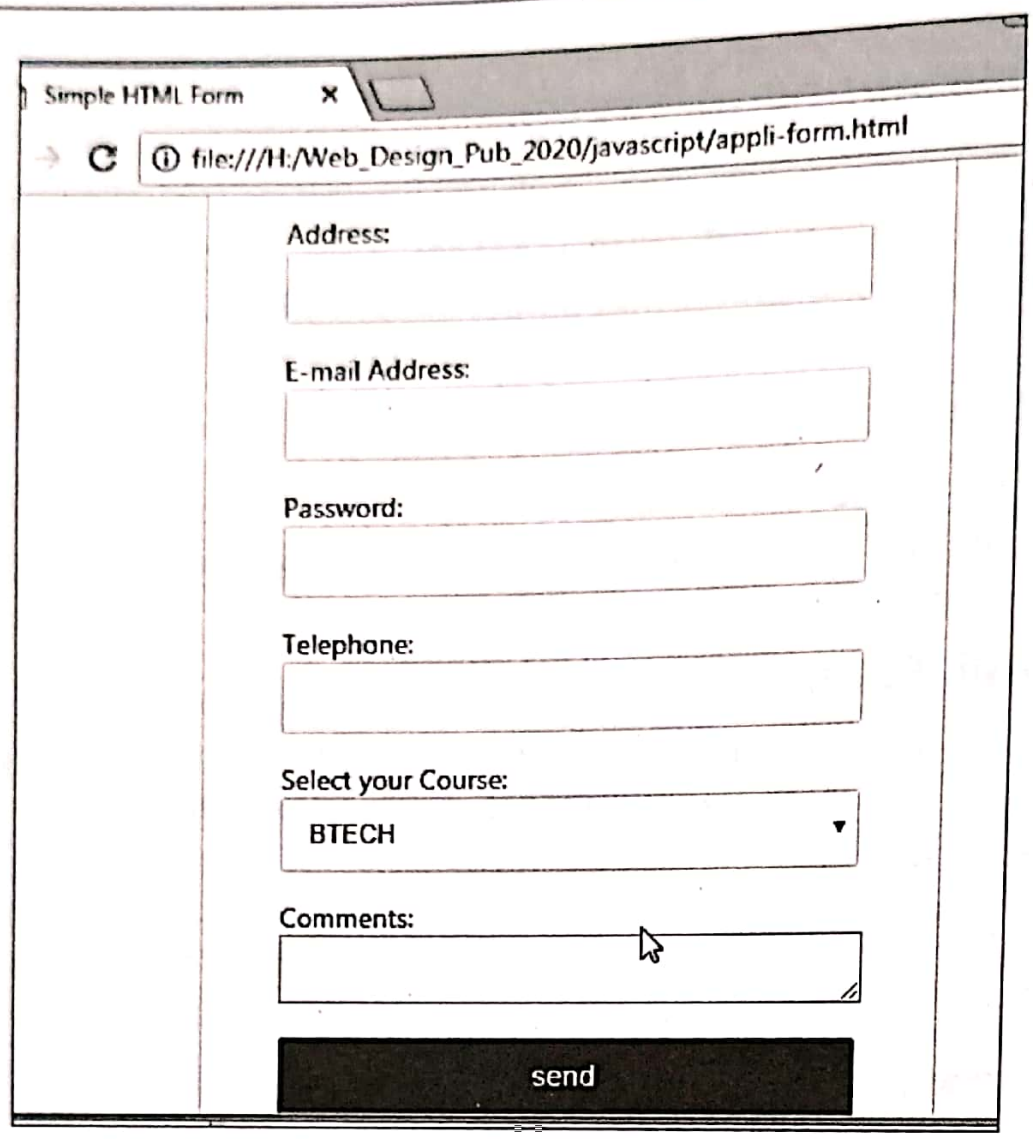


Figure 5.36: Output of Example 36

Form Validation Script

Let us create a JavaScript file name validator.js and place the code inside it, then save it at the same location where you have save the previous html file.

```
<script>
function validateForm()
    {
      var name = document.forms["RegForm"]["Name"];
      var email = document.forms["RegForm"]["EMail"];
      var phone = document.forms["RegForm"]["Telephone"];
      var what =  document.forms["RegForm"]["Subject"];
      var password = document.forms["RegForm"]["Password"];
      var address = document.forms["RegForm"]["Address"];
   if (name.value == "")
   {
       window.alert("Please enter your name.");
       name.focus();
       return false;
   }
   if (address.value == "")
   {
       window.alert("Please enter your address.");
```

(Contd...)

Scanned with OKEN Scanner

```
        address.focus();
         return false;

    }
    if (email.value == "")
    {
        window.alert("Please enter a valid e-mail address.");
        email.focus();
        return false;
    }
    if (phone.value == "")
    {
        window.alert("Please enter your telephone number.");
        phone.focus();
        return false;
    }
    if (password.value == "")
    {
        window.alert("Please enter your password");
        password.focus();
        return false;
    }
    if (what.selectedIndex < 1)
    {
        alert("Please enter your course.");
        what.focus();
        return false;
    }
    return true;
}</script>
```

The value of an individual form filed can been accessed and retrieved using the syntax document.formName.fieldName.value.

Adding style sheet to the Form

Finally, create the file named "style.css" and place the following code in it, then save it also at the same location where you have saved the previous two files.

```
body {
    font-size: 14px;
    color: bold
    background: #f9f9f9;
    font-family: "Segoe UI", "Helvetica Neue", Arial, sans-serif;
}
div {
    box-sizing: border-box;
    width: 100%;
```

(Contd...)

Scanned with OKEN Scanner

```
    border: 100px solid black;
    float: left;
    align-content: center;
    align-items: center;
}
input, select {
    border: 1px solid #ccc;
    padding: 10px;
    display: block;
    width: 100%;
    box-sizing: border-box;
    border-radius: 2px;
}
form {
    width: 300px;
    background: #fff;
    padding: 15px 40px 40px;
    border: 1px solid #ccc;
    margin: 50px auto 0;
    border-radius: 5px;
}
input[type="submit"] {
    font-size: 110%;
    font-weight: 100;
    background: #006dcc;
    border-color: #016BC1;
    box-shadow: 0 3px 0 #0165b6;
    color: #fff;
    margin-top: 10px;
    cursor: pointer;
}
input[type="submit"]:hover {
    background: #0165b6;
}
input[type="Reset"]:hover {
    background: blue;
```

Now, open the "application-form.html" file in a web browser.

5.10 Introduction to Angular JS

AngularJs is a Javscript open source front-end MVC framework that is mainly used to develop Single Page Web Applications(SPA). It was originally developed in **2008-2009** by **Misko hevery** and **Adam abrons**, and is now maintained by **Google**. The Agular JS changes static HTML to dynamic HTML. It extends the ability of HTML by adding built-in attributes and components and provides an ability to create custom attributes using simple JavaScript.

Scanned with OKEN Scanner

Why use it?

a. **Easy to work with:** It is basics of HTML, CSS and JavaScript, not necessary to be an expert in these technologies.

b. **Time saving:** It allows us to work with components and hence you can use them again which saves time and unnecessary code.

c. **Ready to use template:** It is mainly plain HTML, and it mainly makes use of the plain HTML template and passes it to the DOM and then the Angular JS compiler. It traverses the templates and then they are ready to use.

Concepts of AngularJS

Few things you need to know to before starting with AngularJS:

a. **Modules:** A module can be defined as a container that consists of various application parts. It is a set of functions defined in a JavaScript file and module divides an application into small and reusable components.

b. **Directives:** Directives indicate the compiler to associate a behavior to the DOM element or modify it. Angular JS contains several directives such as ng-app, ng-controller, ng-view, ng-if, etc.

c. **Expressions:** Angular JS expressions are expressed with {{ }} which indicate a data binding in HTML. These expressions can be added into the HTML templates.

d. **Controller:** A JavaScript object constructor function controls the Angular JS applications.

e. **Scope:** A JavaScript object acts as a link between the controller and the View. Each data manipulation and assignment takes place with the help of scope object.

f. **Data Binding:** It coordinates model and view about any changes in either of these two.

g. **Validations:** It take place with the help of Angular JS forms and controls.

h. **Filters:** It display the formatting of data on DOM and extend the behavior of directives and binding expressions.

i. **Services:** These are singletons that are used by directives, controllers, or other services.

j. **Testing:** The codes developed by **Dependency Injections** are tested. Some of the popular testing frameworks like Jasmine and Karma are two widely used technologies.

Advantages of Angular JS

The advantages of angular JS are as follows:

a. It is Open source JavaScript MVC framework.

b. It is supported by Google

c. There is no need to learn another scripting language. It is pure JavaScript and HTML.

d. It supports separation of concerns by using MVC design pattern.

e. Built-in attributes (directives) makes HTML dynamic.

f. Easy to extend and customize.

g. It supports Single Page Application.

h. Uses Dependency Injection.

i. Easy to Unit test.

Disadvantages of AngularJS

The disadvantages of Angular JS are as follows:

Scanned with OKEN Scanner

a. **Not Secure:** Its applications are not safe.
b. **Not Degradable:** If user of your application disables the JavaScript then it displays nothing except basic page.
c. **Complex at times:** It becomes complex to handles as there are multiple ways to do the same thing and it creates confusion and requires considerable efforts.

Model View Controller(MVC)

It is a software design pattern used to develop a web application. It consists of three components. These are:

a. **Model:** It used to manage the application data. It responds to the request from view and to the instructions from controller to update itself
b. **View:** It is responsible for displaying the application data. A presentation of data in a particular format, triggered by the controller's decision to present the data. They are script-based systems such as JSP,ASP, PHP and very easy to integrate with AJAX technology.
c. **Controller:** The main job is to connect the model and the view component. The controller responds to user input and performs interactions on the data model objects.

Normally, the MVC architecture have to split applications into these three components and then write the code to connect them. However, in Angular JS all you have to do is split the application into MVC and it does the rest by itself. It saves a lot of time and allows finishing the job with less code.

Combine model view controller in one application like as follows.

```
<html ng-app>
<body ng-controller="TextController">
<p>{{msg}}</p>
<script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.3.14/angular.min.js">
</script>
<script>
function TextController($scope) {
$scope.msg = 'Hello AngularJS;
}
</script>
</body>
</html>
```

How to create first application in AngularJS

An AngularJS application consists of following three main parts:

a. **ng-app**– Defines and links application to HTML.
b. **ng-model**– Binds the values of application data to HTML input controls.
c. **ng-bind**– Binds the Application data to HTML tags.

Steps to create AngularJS Application

1. **Load framework it is done by using <Script> tag.**

   <script src = "http://ajax.googleapis.com/ajax/libs/angularjs/1.3.14/angular.min.js">
   </script>

Scanned with OKEN Scanner

2. **Define application using ng-app directive**

```
<div ng-app = "">
//code
</div>
```

3. **Define a model name using ng-model directive.**

```
<p>Enter Text: <input type = "text"  ng-model = "name"></p>
```

4. Bind the value of above model defined using ng-bind directive.

```
<p>Hello<span ng-bind = "name"></span></p>
```

Example 37: Simple Program in AngularJS

```
<html>
<head>
<script src = "http://ajax.googleapis.com/
    ajax/libs/angularjs/1.3.14/angular.min.js"></script>
<title>Welcome AngularJS </title>
</head>
<body>
<h1> Hello Program in AngularJS </h1>
<div ng-app = "">
<p>Enter Text:<input type = "text" ng-model = "name"></p>
<p>Hello <span ng-bind = "name"></span></p>
</div
</body>
</html>
```

The output of the above code as shown in Figure 5.37.

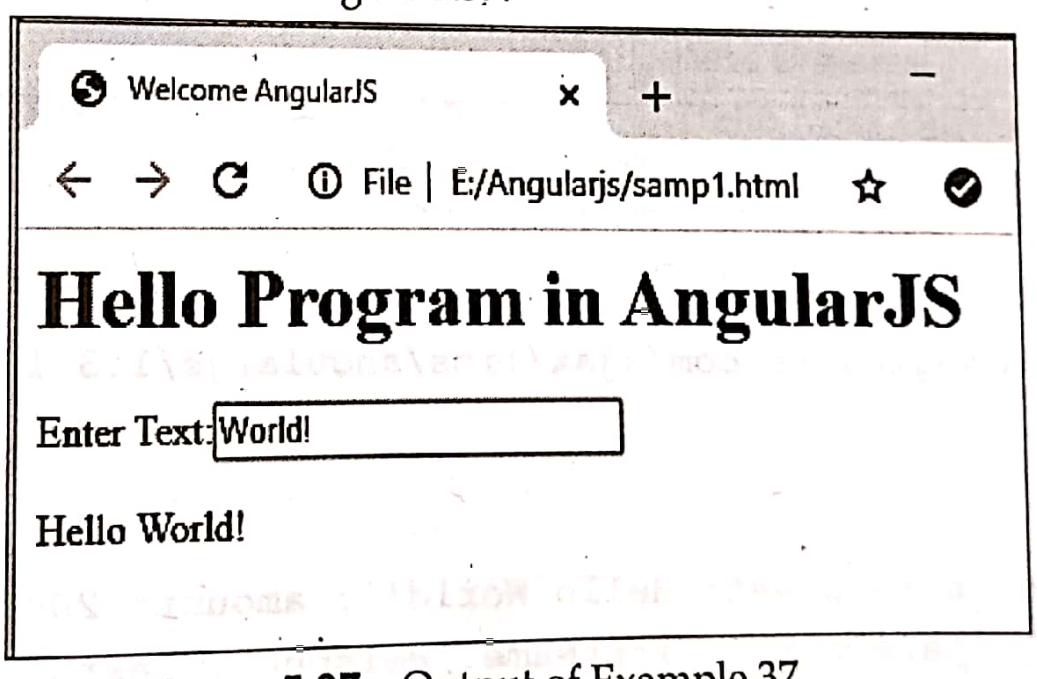


Figure 5.37: Output of Example 37

What is Angular JS Directives?

Directives are used to extend HTML. It is a started with **ng-prefix**. Angular JS contains following directives:

The **ng-app Directive:** It is used to initialize an Angular JS application. The ng-app directive is a starting point.

Example,

```
<div ng-app = "">
```

Scanned with OKEN Scanner

```
...
</div
```

The **ng-init Directive:** It is used to initialize variables in Angular JS application.

Example,

```
<div ng-app="" ng-init="name=['abc']">
//code
</div>
```

The **ng-model directive:** It associates the value of HTML controls with the application data.

Example,

```
<div ng-app = "">
//code
<p>Enter Text:<input type = "text" ng-model = "name"></p>
</div>
```

The **.ng-repeat directive:** It is used to repeat HTML elements for every item in a collection.

Example,

```
<div ng-app="" ng-init="names=['abc','xyz']">
<ul>
<li ng-repeat="n in names">
{{ n }}
</li>
</ul>
</div>
```

ng-init

The following example demonstrates **ng-init** directive can be used to initialize variables of string, number, array and object.

Example 38:

```
<!DOCTYPE html>
<html >
<head>
    <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.3.16/angular.min.js"></
script>
</head>
<body >
    <div ng-app ng-init="greet='Hello World!'; amount= 200;
myArr = [200, 300]; person = { firstName:'Welspun', lastName :'India'}">
        {{amount}}      <br />
        {{myArr[2]}}    <br />
        {{person.firstName}}
    </div>
</body>
</html>
```

The output of the above code as shown in Figure 5.38.

Scanned with OKEN Scanner

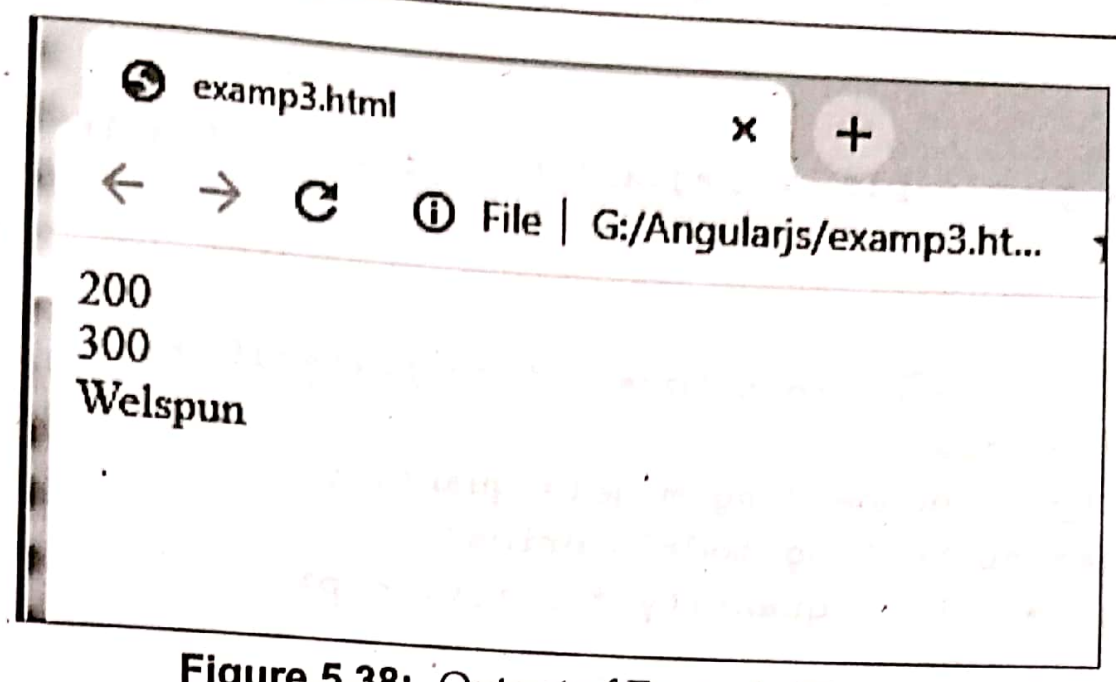


Figure 5.38: Output of Example 38

The above example, you initialized variables of string, number, array and object.

ng-model

The ng-model directive is used for two-way data binding. It binds <input>, <select> or <textarea> elements to a specified property on the scope object.

Example 39:

```
<!DOCTYPE html>
<html >
<head>
    <script
src="https://ajax.googleapis.com/ajax/
 libs/angularjs/1.3.16/angular.min.js">
    </script>
</head>
<body ng-app>
    <input  type="text"  ng-model="name"
/>
    <div>
       Hello {{name}}
    </div>
</body>
</html>
```

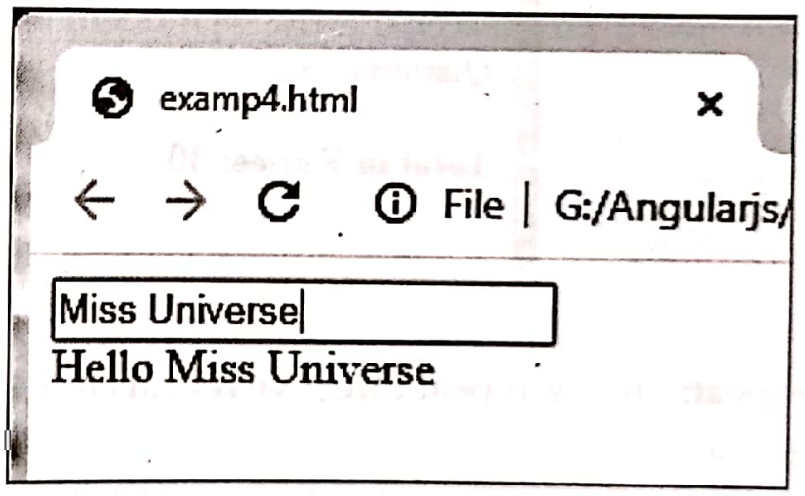


Figure 5.39: Output of Example 39

The output of the above code as shown in Figure 5.39.

☞ *Variables initialized in ng-init are different from the properties defined using ng-model directive. The variables initialized in ng-init are not attached to $scope object whereas ng-model properties are attached to scope object.*

ng-bing

The ng-bind directive binds the model property declared through $scope or ng-model directive. It also updates an element if the value of an expression changes.

Example 40:

```
<!DOCTYPE html>
<html>
```

Scanned with OKEN Scanner

```
<head>
    <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.3.16/angular.min.js"></
script>
</head>
<body>
<div data-ng-app="" data-ng-init="quantity=3;price=10">
<h2>Cost Calculator</h2>
Quantity: <input type="number" ng-model="quantity">
Price: <input type="number" ng-model="price">
<p><b>Total in Rupee:</b> {{quantity * price}}</p>
</div>
</body>
</html>
```

The output of the above code as shown in Figure 5.40.

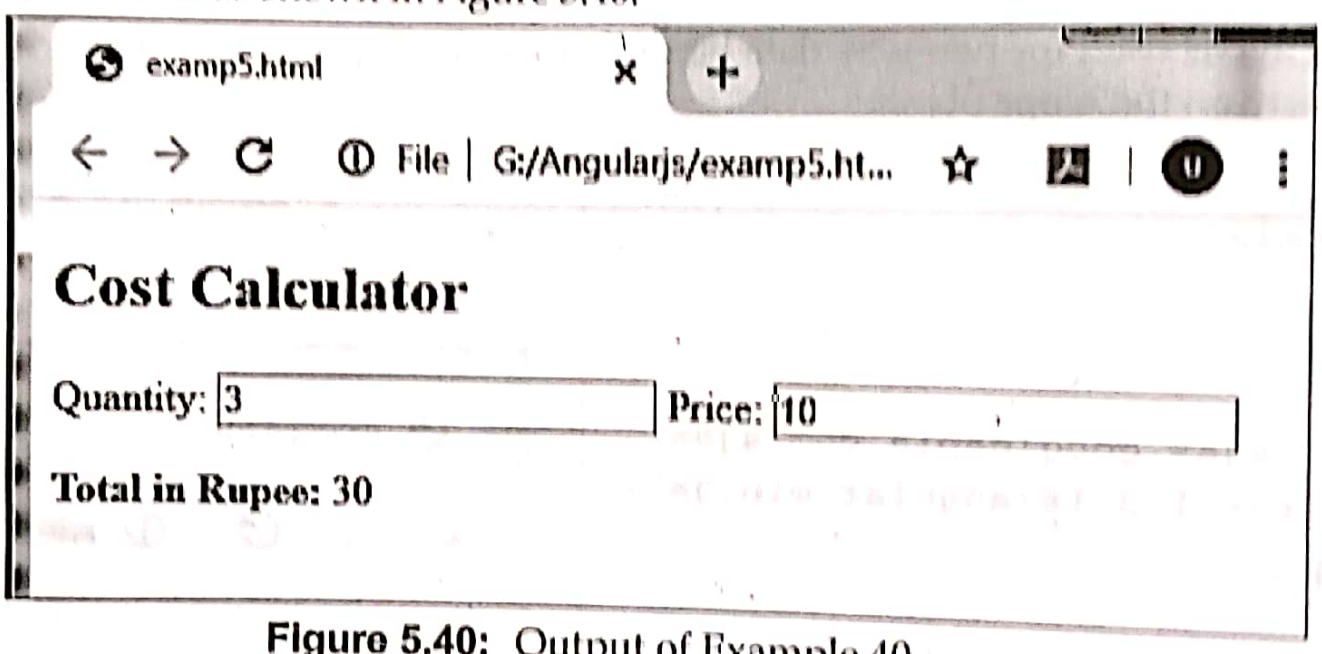


Figure 5.40: Output of Example 40

ng-repeat: The ng-repeat directive repeats html once per each item in the specified array collection.

Example 41:

```
<!DOCTYPE html>
<html>
<script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.6.9/angular.min.js"></s
cript>
<body>
<div ng-app="" ng-init="names=[
{name:'Janani',country:'India'},
{name:'Hemalatha',country:'Singapore'},
{name:'Kumar',country:'Pakaistan'}]">
<p>Looping with objects:</p>
<ul>
  <li ng-repeat="x in names">
  {{ x.name + ', ' + x.country }}</li>
```

Scanned with OKEN Scanner

```
</ul>
</div>
</body>
</html>
```

The output of the above code as shown in Figure 5.41.

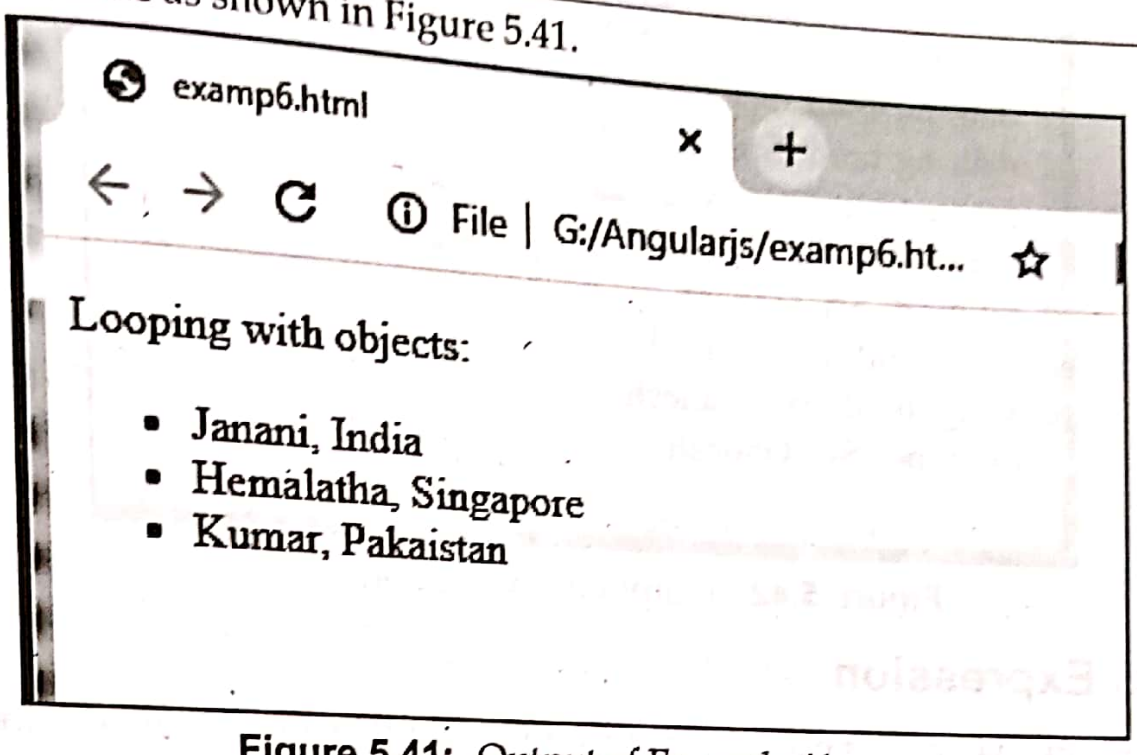


Figure 5.41: Output of Example 41

Directive

Directive can start with **x-** or **data-**, for example ng-model directive can be written as **data-ng-model** or **x-ng-model**.

Also, the **-** in the directive can be replaced with **:** or **–** or both. For example, ng-model can be written as **ng-model** or **ng:model**. It can be a mix with **data-** or **x-**.

Example 42:

```
<!DOCTYPE html>
<html >
<head>
    <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.3.16/angular.min.js"></
script>
</head>
<body ng-app>
    Enter Name:  <input type="text" ng-model="name" /> <br />
    data-ng-bind: <span data-ng-bind="name"></span><br />
    data-ng:bind: <span data-ng:bind="name"></span><br />
    data:ng:bind: <span data:ng:bind="name"></span><br />
    x:ng:bind:    <span x:ng:bind="name"></span><br />
    ng:bind:      <span ng:bind="name"></span><br />
    x-ng-bind:    <span x-ng-bind="name"></span><br />
    x_ng_bind:    <span x_ng_bind="name"></span><br />
    ng_bind:      <span ng_bind="name"></span>
</body>
</html>
```

The output of the above code as shown in Figure 5.42.

Scanned with OKEN Scanner

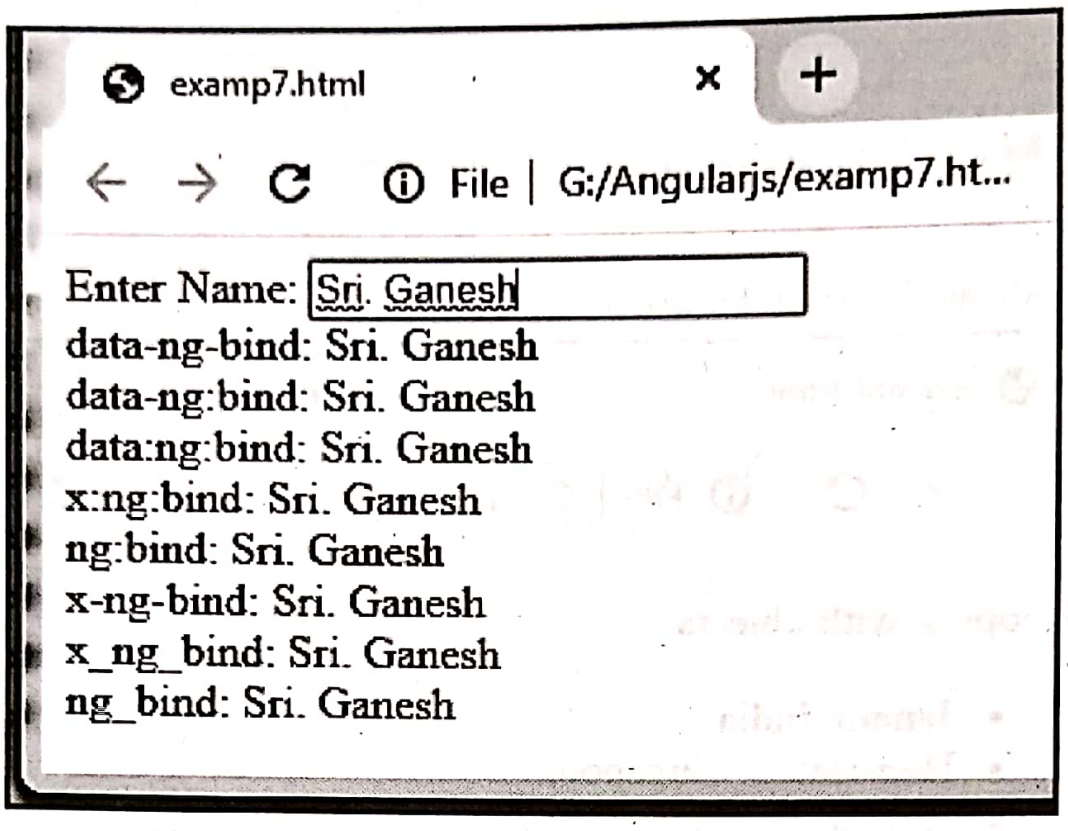


Figure 5.42: Output of Example 42

5.11 AngularJS Expression

Angular JS expression is like JavaScript expression surrounded with braces – {{ expression }}. It evaluates the specified expression and binds the result data to HTML. Angular JS expression can contain literals, operators and variables like JavaScript expression. *For example,* an expression {{2 + 2}} will produced the result 1 and will be bound to HTML.

Angular JS expressions are much like JavaScript expressions. They can contain literals, operators, and variables.

Example 43:

```
<!DOCTYPE html>
<html >
<head>
    <script
src="https://ajax.googleapis.com/ajax/
  libs/angularjs/1.3.16/angular.min.js"
></script>
</head>
<body >
    <h1>AngularJS Expression</h1>
<div ng-app>
<p>Addition: 5 + 5 = {{ 5 + 5 }}</p><br />
<p>Subtraction: 5 - 5 = {{5 - 5}}</p> <br />
<p>Multiplication: 5 * 5 = {{5 * 5}}</p> <br/>
<p>Division: 5/5 = {{5 / 5}}
    </div>
</body>
</html>
```

The output of the above code as shown in Figure 5.43.

Scanned with OKEN Scanner

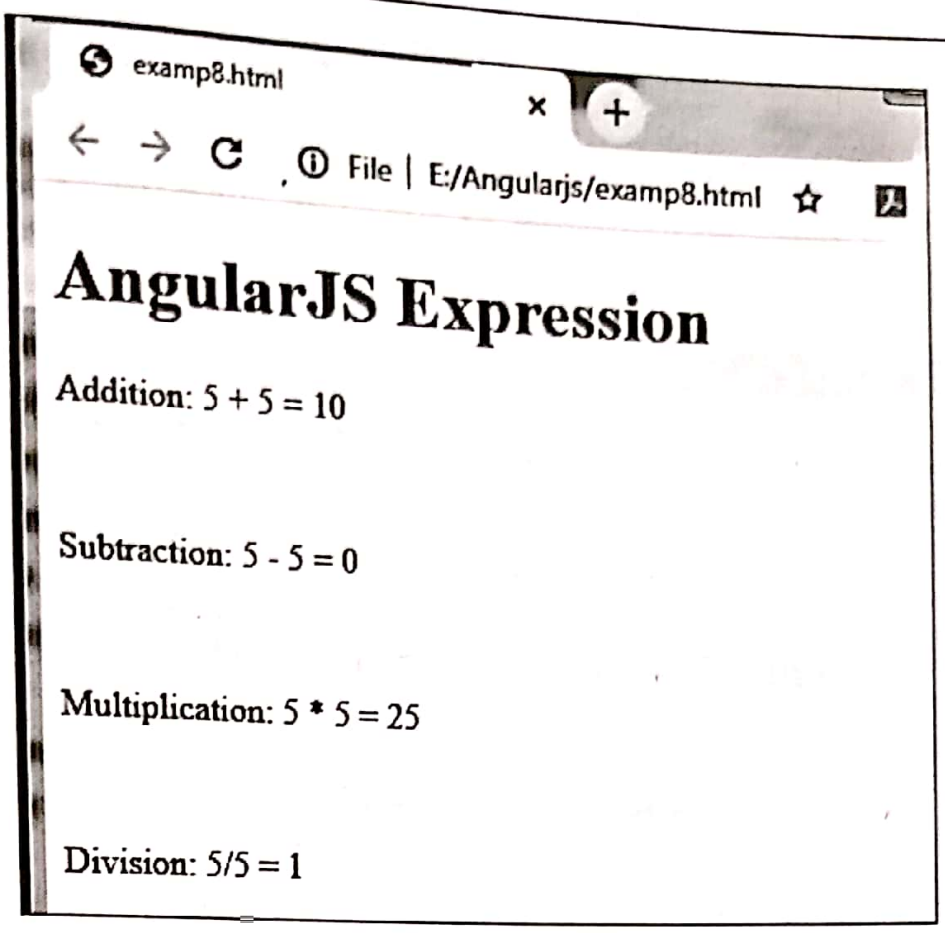


Figure 5.43: Output of Example 43

AngularJS expression is like JavaScript code expression except for the following differences.

AngularJS expression cannot contain conditions, loops, exceptions or regular expressions e.g. if-else, ternary, for loop, while loop etc.

- It cannot declare functions.
- It cannot contain comma or void.
- It cannot contain return keyword.
- You can write expressions wherever you like, Angular JS will simply resolve the expression and return the result.

Let Angular JS change the value of CSS properties. Change the color of the input box below, by changing its value:

Example 44:

```
<!DOCTYPE html>
<html>
<script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.6.9/angular.min.js">
</script>
<body>
<p>Change the value of the input field:</p>
<div ng-app="" ng-init="myCol='orange'">
<input style="background-color:{{myCol}}" ng-model="myCol">
</div>
<p>The background color of the input box will be whatever you write in the
input field.</p>
</body>
</html>
```

The output of the above code as shown in Figure 5.44.

Scanned with OKEN Scanner

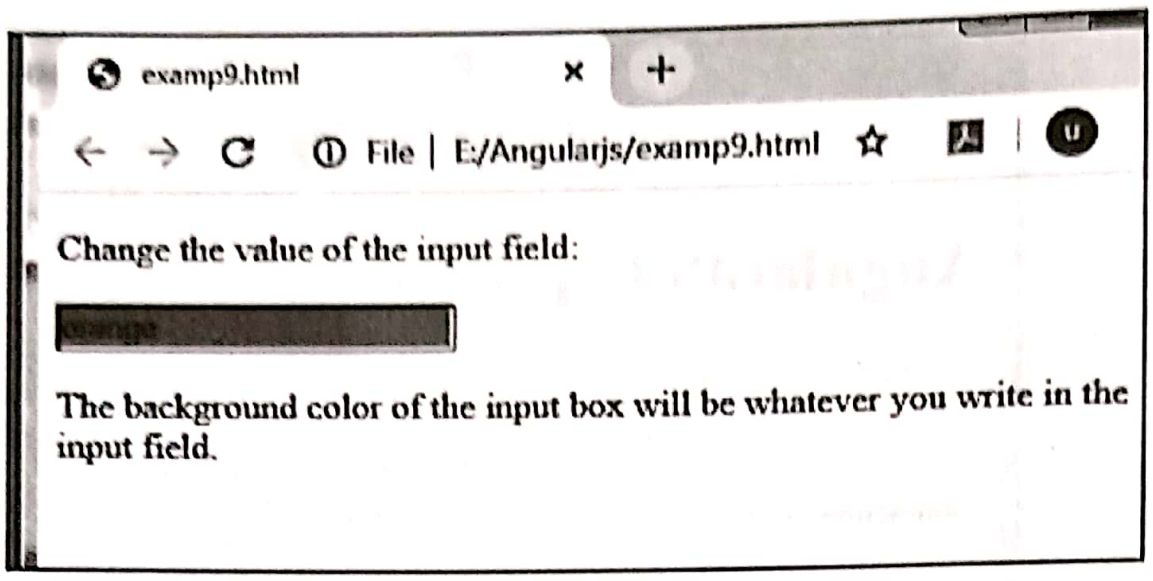


Figure 5.44: Output of Example 44

5.11.1 AngularJS Numbers

Angular JS numbers are like JavaScript numbers using ng-bind.

Example 45:

```
<!DOCTYPE html>
<html>
<script
src="https://ajax.googleapis.com/ajax/
libs/angularjs/1.6.9/angular.min.js"></script>
<body>
<div ng-app="" ng-init="quantity=5;cost=25">
<p>Total in Rupee: <span ng-bind="quantity *
cost"></span></p>
</div>
</body>
</html>
```

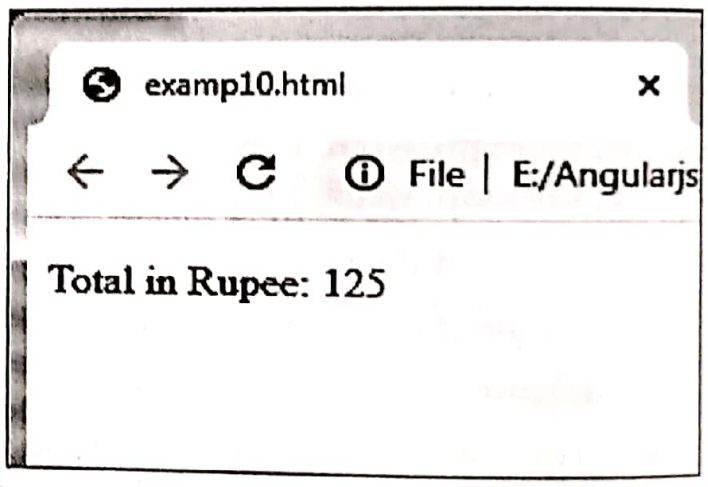


Figure 5.45: Output of Example 45

The output of the above code as shown in Figure 5.45

5.11.2 Angular JS Strings

Angular JS strings are like JavaScript strings.

Example 46:

```
<!DOCTYPE html>
<html>
<script src=
"http://ajax.googleapis.com/ajax/libs/angularjs/1.3.14/angular.min.js"></scri
pt>
<body>
<div ng-app="" ng-init="firstName='Neha';lastName='Singhvi'">
<p>Full name: <span ng-bind="firstName + ' ' + lastName"></span></p>
</div>
</body>
</html>
```

The output of the above code as shown in Figure 5.46.

Scanned with OKEN Scanner

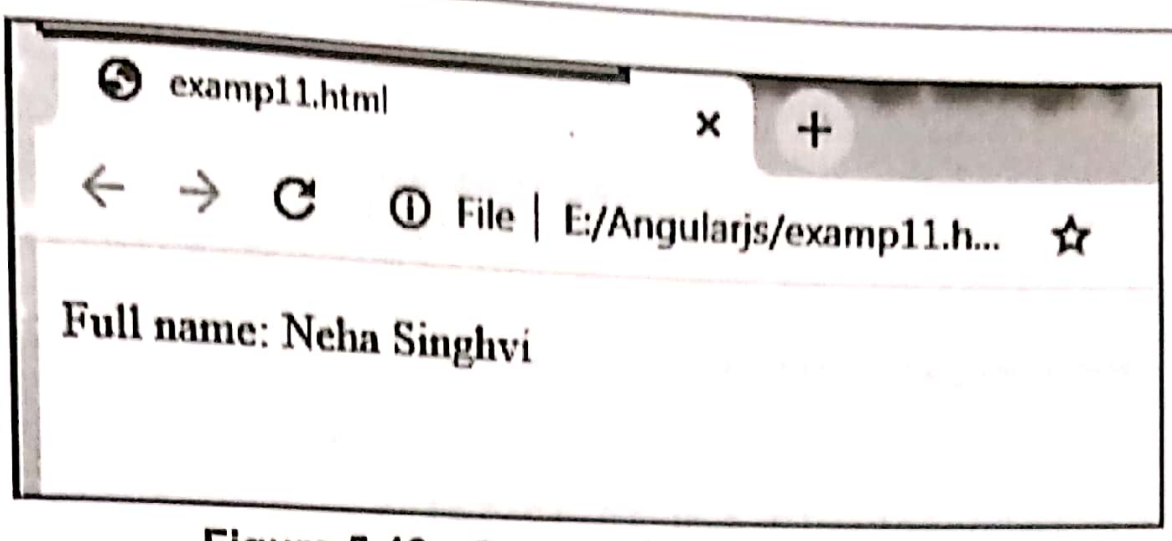


Figure 5.46: Output of Example 46

Angular JS Objects

Angular JS objects are like JavaScript objects.

Example 47:

```
<!DOCTYPE html>
<html>
<script
src="https://ajax.googleapis.com/ajax/

libs/angularjs/1.6.9/angular.min.js">
</script>
<body>
<div ng-app="" ng-init="person=
 {firstName:'Aditi',lastName:'Jain'}">
<p>The name is {{ person.lastName
}}</p>
</div>
</body>
</html>
```

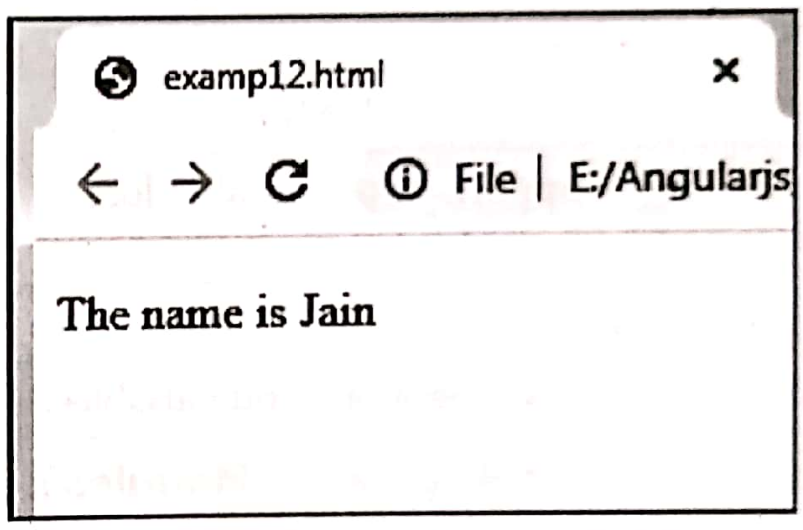


Figure 5.47: Output of Example 47

The output of the above code as shown in Figure 5.47.

5.11.3 Angular JS Arrays

Angular JS arrays are like JavaScript arrays.

Example 48:

```
<!DOCTYPE html>
<html>
<script
src="https://ajax.googleapis.com/ajax/
libs/angularjs/1.6.9/angular.min.js">
</script>
<body>
<div ng-app=""
 ng-init="points=[2,4,8,10,12,14]">
<p>The third result is {{points[4]
}}</p>
</div>
</body>
</html>
```

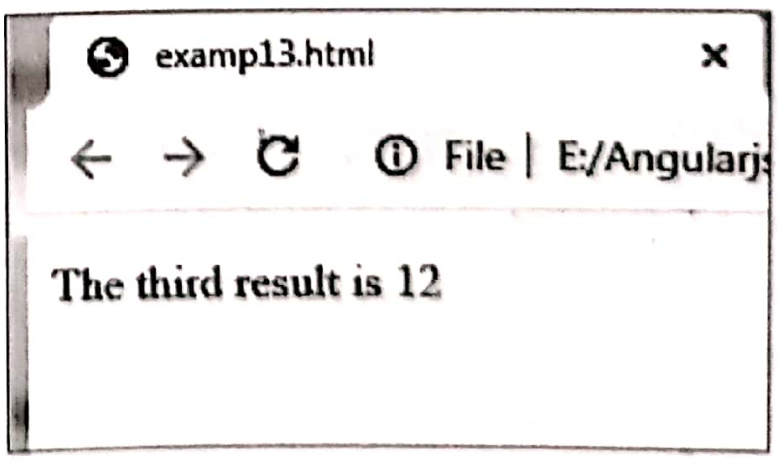


Figure 5.48: Output of Example 48

The output of the above code as shown in Figure 5.48.

Scanned with OKEN Scanner

Same example using ng-bind.

Example 49:

```
<!DOCTYPE html>
<html>
<script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.6.9/angular.min.js">
</script>
<body>
<div ng-app="" ng-init="points=[2,4,8,10,12,14]">
<p>The third result is <span ng-bind="points[4]"></span></p>
</div>
</body>
</html>
```

Difference between Angular JS Expressions and JavaScript Expressions

Angular JS Expressions	JavaScript Expressions
It can be written inside HTML.	It cannot be written inside HTML.
It do not support conditionals, loops, and exceptions.	It support conditionals, loops, and exceptions.
It support filters.	It does not support filters.
It contain literals, operators, and variables.	Contain literals, operators, and variables.

5.11.4 What is AngularJS Modules?

A module defines the application functionality that is applied to the entire HTML page using the ng-app directive. It defines functionality, such as services, directives, and filters, in a way that makes it easy to reuse it in different applications.

Creating a Module

A module is created by using the AngularJS function **angular.module**.

```
<div ng-app="myApp">...</div>
<script>
var app = angular.module("myApp", []);
</script>
```

The "myApp" parameter refers to an HTML element in which the application will run. Now you can add controllers, directives, filters, and more, to your Angular JS application.

Adding a Controller

Add a controller to your application, and refer to the controller with the **ng-controller** directive.

Example 50:

```
<!DOCTYPE html>
<html>
<script src="https://ajax.googleapis.com/ajax/libs/
            angularjs/1.6.9/angular.min.js"> </script>
```

(Contd...)

Scanned with OKEN Scanner

```
<body>
<h2>Angular Module</h2>
<div ng-app="myApp"
  ng-controller="myCtrl">
   {{ message}}
</div>
<script>
var   app   =   angular.module("myApp",
[]);
app.controller("myCtrl",
function($scope)  {
    $scope.message = "AngularJS
    Modules!";
});
</script>
</body>
</html>
```

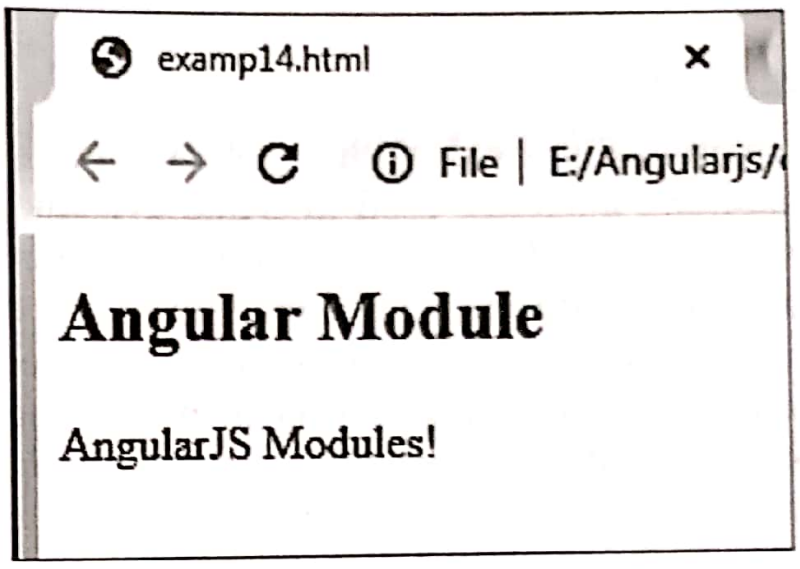


Figure 5.49: Output of Example 50

The output of the above code as shown in Figure 5.49.

5.11.5 Modules and Controllers in Files

It is common in Angular JS applications to put the module and the controllers in JavaScript files. In this example, "**myApp.js**" contains an application module definition, while "**myCtrl.js**" contains the controller.

Example 51:

```
<!DOCTYPE html>
<html>
<script
src="https://ajax.googleapis.com/ajax/
libs/angularjs/1.6.9/angular.min.js">
</script>
<body>
<div          ng-app="myApp"          ng-
controller="myCtrl">
{{ firstName + " " + lastName }}
</div>
<script src="myApp.js"></script>
<script src="myCtrl.js"></script>
</body>
</html>
```

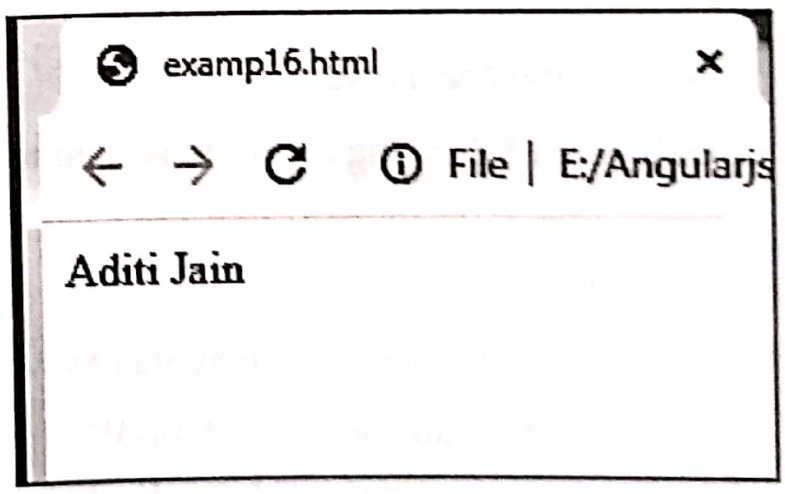


Figure 5.50: Output of Example 51

The output of the above code as shown in Figure 5.50.

myApp.js

```
var app = angular.module("myApp", []);
```

The [] parameter in the module definition can be used to define dependent modules. Without the [] parameter, you are not creating a new module, but retrieving an existing one.

myCtrl.js

Scanned with OKEN Scanner

```
app.controller("myCtrl", function($scope) {
 $scope.firstName = "John";
$scope.lastName= "Doe";
});
```

Conclusion

This chapter discussed with JavaScript and Angular JS. JavaScript is a general-purpose high-level programming language used to create dynamic websites and web applications to run in the client's web browser. Angular JS is a JavaScript-based framework, which extends html with new attributes and is specially designed to build dynamic single-based web applications. The next topics, beings with client-side scripting is source code that is, executed on the client's browser instead of the web-server. Next, we have discussed variables are a key element in programming languages, an understanding of the basics of JavaScript variables is key to learning scripting language. Then, we have discussed different types of operators in JavaScript. Next, we have discussed conditional statements are used to decide the flow of execution based on different conditions. Next, you will learn different types of alert messages such as alert, confirmation and prompt. Next, you will learn form validation in JavaScript provides a way to validate form's data on the client's computer before sending it to the web server. Finally, we have discussed AngularJS with Expression, Modules and Directives.

REVIEW QUESTIONS WITH ANSWERS

1. Multiple Choice

1.1 Which of the following type of variable is visible everywhere in your JavaScript code?

a. Global variable
b. Local variable
c. Both (a) and (b)
d. None of the above

1.2 The script tag must be placed in

a. Head
b. Head and body
c. Title and Head
d. All of the above

1.3 Which of the following attribute is used to include external js code inside your html document?

a. Src
b. Ext
c. Script
d. Link

1.4 What is the correct JavaScript syntax to write "Hello World"?

a. System.out.println("Hello World")
b. println ("Hello World")
c. document.write("Hello World")
d. response.write("Hello World")

1.5 What is the correct syntax for referring to an external script called " abc.js"?

a. <script href=" abc.js">
b. <script scr="abc.js">
c. <script name=" abc.js">
d. <script src=" abc.js">

1.6 Multiple declarations of variables are separated by which symbol.

a. Semicolon
b. Colon
c. Comma
d. Asterisk

Scanned with OKEN Scanner

1.7 Which of the following is the correct syntax to display "enter password" in an alert box using JavaScript?

a. alertbox("enter password");
b. msg("enter password");
c. msgbox("enter password");
d. alert("enter password");

1.8 Predict the output of the following JavaScript code.

```
<script type="text/javascript" language="javascript">
var x=5;
var y=6;
var res=eval("x*y");
document.write(res);
</script>
```

a. "30"
b. 30
c. 5*6
d. "5*6"

1.9 Which of these is not a logical operator?

a. !
b. &
c. &&
d. ||

1.10 What will be the output of the following JavaScript code?

```
int x=0;
for(x;x<10;x++);
console.log(x);
```

a. 0
b. 9
c. 10
d. Error

2. State whether the following statements are True or False

2.1 The external JavaScript file must contain <script> tag.
2.2 Javascript string using double quotes is exactly the same as a string using single quotes.
2.3 Multiple variables can be created and initialized in a single JavaScript statement.
2.4 In JavaScript, Window.prompt() method return true or false value.
2.5 JavaScript is server Side Scripting Language.
2.6 We can embed JS code inside HTML directly.
2.7 We cannot Place JS Code in the body tag.
2.8 Underscore can be used as first letter while declaring variable in JavaScript.
2.9 Do AngularJS provide reusable components?
2.10 ng-model binds the values of AngularJS application data to HTML input controls.

3. Match the following

3.1	JavaScript is a	a.	angular.module();
3.2	It is a controller in MVC	b.	It refers current object
3.3	AngularJS module can be created using	c.	Controlling the data
3.4	The structure of an if statement	d.	high-level, interpreted scripting language.

Scanned with OKEN Scanner

3.5	"this" keyword in JavaScript mean by	e.	High level programming language
3.6	Angular controllers are responsible for	f.	Synchronization between model and view part
3.7	Misko Hevery	g.	JavaScript is also called server-side JavaScript
3.8	Livewire	h.	A software Code that controls the interactions between the Model and View
3.9	A proper scripting language is a	i.	if (conditional expression is true) {then execute this code>->}
3.10	Data binding in Angular JS	j.	Father of Angularjs
		k.	Interface

4. Fill in the blanks

a.	Client	b.	Model view controller	c.	Declaration
d.	Local variable	e.	Html and JavaScript	f.	Binary
g.	Unary	h.	ng-model	i.	ng-app
j.	Charset	k.	ng-start		

4.1 var num; Statement is called as _____ Statement.

4.2 Angular JS applications are a mix of _____.

4.3 Angular JS is based on the _____ pattern.

4.4 JavaScript is interpreted by _____.

4.5 Variable declared inside JavaScript function will be called as _____.

4.6 Assignment operators is following type of operator _____.

4.7 "++" is _____ type of operator.

4.8 _____ Angular directive is used to binds the value of HTML controls (input, select, textarea) to application data?

4.9 _____ attribute is used to specify the character encoding used in an external script file.

4.10 _____ directive is used to start an angular JS application?

Descriptive Questions with Answer

5.(a) What is JavaScript?

Answer

JavaScript is an object-oriented computer programming language commonly used to create interactive effects within web browsers. It is first used by the Netscape browser, that provides access to the HTML **Document Object Model (DOM)**, provides access to the **Browser Object Model (BOM)**. It allows pages to respond to events, display special effects, accept variable text, validate data, make cookies, detect user's browser.

5.(b) Explain what is pop()method in JavaScript?

Answer

The pop() method is similar as the shift() method but the difference is that the Shift method works at the

Scanned with OKEN Scanner

start of the array. Also the pop() method take the last element off of the given array and returns it. The array on which is called is then altered.

For example, var cloths = ["Shirt", "Pant", "TShirt"];

cloths.pop();

5.(c) What are the features of JavaScript?

Answer

Following are the features of JavaScript:

a. It is a lightweight, interpreted programming language.
b. It is designed for creating network-centric applications.
c. It is complementary to and integrated with Java.
d. It is an open and cross-platform scripting language.

6.(a) How can you create an Array in JavaScript?

Answer

You can define arrays using the array literal as follows:

```
var x = [];
var y = [1, 2, 3, 4, 5];
```

6.(b) What is an operator and what are its types?

Answers

Operators are used for comparing values, perform arithmetic operations, etc. For example, if we take a simple expression, 4 + 5 is equal to 9. Here 4 and 5 are called operands and '+' is called the operator. JavaScript consists of different types of operators that are used to perform different operations.

Types of JavaScript Operators

There are different types of operators in JavaScript that are used for performing different operations. Some of the JavaScript Operators include:

a. Arithmetic Operators
b. Comparison Operators
c. Bitwise Operators
d. Logical Operators
e. Assignment Operators

6.(c) What are events in JavaScript and its types?

Answer

JavaScript has events that provide a dynamic interface to a webpage. These events are connected to elements in the Document Object Model(DOM).

Also, these events by default use the bubbling propagation i.e., upwards in the DOM from children to parent. We can bind events either as inline or in an external script. With the help of JavaScript, you can detect when certain events happen, and cause things to occur in response to those events.

Types of Events in JavaScript

There are different types of events in JavaScript that are used to react to events. Some of the famous or most commonly used events such as:

a. Onclick

Scanned with OKEN Scanner

b. Onkeyup
c. Onmouseover
d. Onload
e. Onfocus

7.(a) Explain what are directives? Mention some of the most commonly used directives in AngularJS application?

Answer

A directive is something that introduces new syntax, they are like markers on DOM element which attaches a special behavior to it. In any AngularJS application, directives are the most important components.

Some of the commonly used directives are ng-model, ng-App, ng-bind, ng-repeat, ng-show etc.

7.(b) what are the advantages of using Angular JS?

Answer

a. Angular JS has several advantages in web development.
b. Angular JS supports MVC pattern.
c. Can do two ways data binding using Angular JS.
d. It has per-defined form validations.
e. It supports both client server communication.
f. It supports animations.

8.(a) What do you understand by Angular JS?

Answer

Angular JS is a JavaScript framework that is used for making rich and extensible web applications. It runs on plain JavaScript and HTML; hence you don't need any other dependencies to make it work. Angular JS is perfect for **Single Page Applications (SPA)**. It is basically used for binding JavaScript objects with HTML UI elements.

8.(b) Distinguish between Angular JS and JavaScript expressions.

Answer

There are several differences between Angular JS and JavaScript expressions:

a. We can write Angular JS expressions in HTML, but we cannot write JavaScript expressions in HTML.
b. We cannot use conditional iterations, loops, and exceptions in Angular JS, but we can use all of these conditional properties in JavaScript expressions.
c. Filters are supported in Angular JS whereas filters are not supported in JavaScript.

9. (a) What are the Angular Modules?

Answer

The angular modules collectively define an angular application where we can write the angular code. It contains the different parts of an angular application. A module is created by angular module function in angular.

(b) Explain the ng-model directive in Angular JS.

Answer

This can be a leap hop with the custom HTML input form control (like input, textarea and select) to the

Scanned with OKEN Scanner

application data. It provides form validation behavior with a two-way binding.
<input ng-bind="expression"/>

(c) What is a controller in Angular JS?

Answers

A controller is a JavaScript function which is bound to the specified scope. Angular instantiates the new controller object and injects the new scope as a dependency. It can be used to set up the initial state of the scope object and to add behavior to the object. It cannot be used to share code or state across controllers, but instead of that Angular service can be used.

Example,

<Any ng-Controller=" expression">
</Any>

(d) What is ng-App directive in Angular JS?

Answers

It is used to define the Angular JS Application. It appoints the root element of an Angular JS application and it is kept near the <body> or <html> tag.

We can define any number of ng-app directives inside the HTML document, but only one AngularJS application can be bootstrapped automatically (auto-bootstrapped) and the other applications need to be bootstrapped manually.

Example:

```
<div ng-app="">
<p>My first expression: {{157 + 122}} </p>
</div>
```

Answers

1.1	a	1.2	b	1.3	a	1.4	c	1.5	d	1.6	c
1.7	d	1.8	b	1.9	b	1.10	c				
2.1	F	2.2	T	2.3	T	2.4	F	2.5	F	2.6	T
2.7	F	2.8	T	2.9	T	2.10	T				
3.1	d	3.2	h	3.3	a	3.4	i	3.5	b	3.6	c
3.7	j	3.8	g	3.9	e	3.10	f				
4.1	c	4.2	e	4.3	b	4.4	a	4.5	d	4.6	f
4.7	g	4.8	h	4.9	j	4.10	i				

Scanned with OKEN Scanner

CHAPTER 6

Photo Editor

6.1 Introduction

The Photo Editor is designed to be easy to use and user friendly. It is the fast, powerful, and easy-to-use image editor that everyone from beginners to professionals can use and enjoy. It includes a wide variety of image processing tools you can use to create, edit, and touch-up your digital images. This highly intuitive piece of software supports a high variety of formats which allows you to edit a vast number of images and save it in any format you want. It is equipped with the usual set of file operations together with some of the slightly higher level editing options & adjust colors in photos; and apply filters if you ever want to do a hands on job in customizing your images.

6.1.1 Main Features of Photo Editing

The main features of Photo editing are as follows:

- Import and export various graphic formats, including BMP, GIF, JPG, TIFF, PCX, PNG, TGA, J2K, WMF, EMF, and RAS.
- Apply 15 unique filters such as blur, sharp, Add noise, diffuse, mosaic, twirl, and emboss etc.
- Alignment and straightening of the image including rotate 90°, 180° and arbitrary, flip horizontal and vertical, perspective, and skew, etc.
- Adjusting image' colors easily with 11 adjustment commands such as Auto levels, auto contrast, brightness, contrast, gamma channel, hue, saturation, RGB channel, invert, and colorize, etc.
- Edit images directly from scanner.
- Print images with full functionality and keep them for your archive, showing to friends or colleagues.
- Also ability to increase or decrease the image dimension, which is called *image resizing* with lots of pre-defined and standard sizes, or have your own custom size.
- EPC image editor is provided as FREEWARE for private (non-commercial) or educational (including non-profit organization) use. In these cases, you are granted the right to use and to make an unlimited number of copies of this program.

6.1.2 Photo Editor Tools and Controls

You can use the pixel-based editing tools, such as red eye removal, special effects and many other tools, and combine them with the selections tool to fine-tune your image.

6.1.3 Using Selections

Use Selections to isolate an area of a photo in order to apply edits or filters to only that area, rather than the whole photo. There are five different selection tools and each selects an area of a photo in a unique way. The controls available change depending on which tool you select.

To Use a Selection Tool:

1. At the top of the toolbar on the left, select the button for one of the following selection tools:
   Brush selection tool, Rectangular selection tool, Elliptical selection tool, Lasso selection tool, Magic wand selection tool.
2. You can use Selections with the following tools:

Scanned with OKEN Scanner

Repair: Skin Tune.

Add: Vignette, special effects, tilt-shift.

Exposure/Lighting: Exposure, levels, auto levels, tone curves, light EQ™, dehaze, dodge and burn.

Color: White balance, color EQ, color balance, convert to black and white, split tone.

Detail: Sharpen, blur, noise, clarity, detail brush.

6.1.4 Selection Tools

Freehand Lasso: Click the left mouse button and drag the cursor to draw around the area that you wish to select. As you draw, a line appears showing where you have drawn. When you release the mouse, the end of the line joins to the start automatically to complete the selection. The whole selection animates and looks like marching ants.

Magic Wand: Click on any area of the image and all the pixels the same color are included in the selection. You can choose whether to select only the same color pixels that are actually touching the one you clicked, or all the pixels in the photo that are similar. You can increase the threshold to include more pixels in the selection. The lower the threshold, the more similar a pixel has to be to the one clicked in order to be included. The greater the threshold, the more different a pixel can be and still be in the selection.

Rectangular or Elliptical: Click and drag either a rectangle or an ellipse that begins where you first click and ends when you release the mouse.

Brush: You can use the brush selection tool to target your selection based on color, brightness, or a combination of both color and brightness. Select color, brightness, or magic from the drop-down menu in the context bar to define your selection. Then align the center of the brush with the color, brightness, or combination that you wish to target and begin brushing. You can increase the tolerance to include a wider range of pixels in your selection. The lower the tolerance, the more similar a pixel has to be to the one clicked in order to be included. You can erase the selection by right-clicking.

Using the Selection Tools

To Use the Lasso:

1. Select the **Lasso** button.
2. On the image, click and hold down the left mouse button as you draw around the area you want to select. (*See Figure 6.1*)

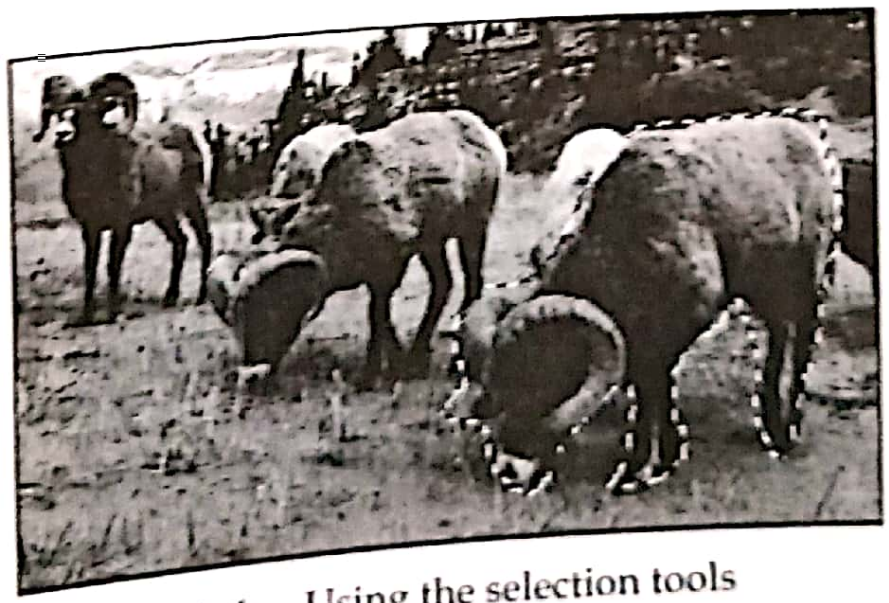


Figure 6.1: Using the selection tools

3. Release the mouse button to complete the selection.
4. To clear the selection, click anywhere on the image (for the Lasso, Rectangular, or Elliptical selection).

Scanned with OKEN Scanner

5. To add to an existing selection, hold down **SHIFT**, or press the **Add to selection** button in the context bar, and draw a shape that includes any part of the original selection line. . When you release the mouse, the original selection expands to include the outer edge of your addition. (*See Figure 6.2*)

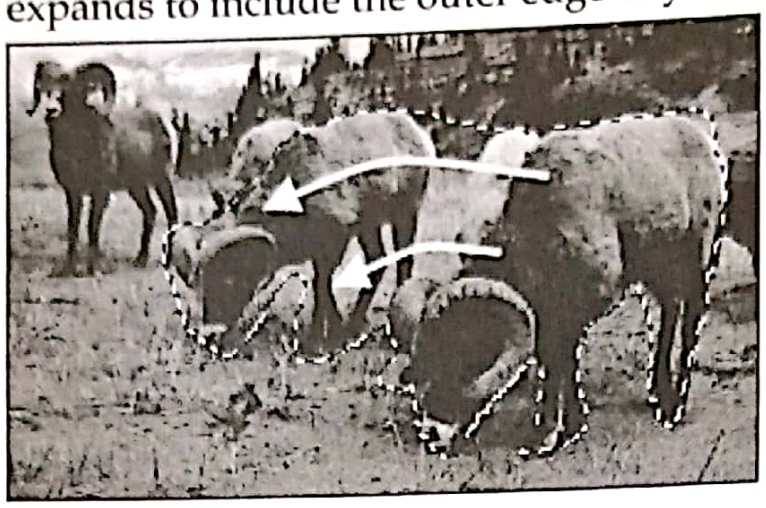


Figure 6.2: Add to selection

6. To add multiple areas into your selection, hold down SHIFT, or press the Add to selection button in the context bar and draw around any part of the image . (*See Figure 6.3*)
   As long as you do not touch the line of any other selection, you can add several separate areas to a selection.

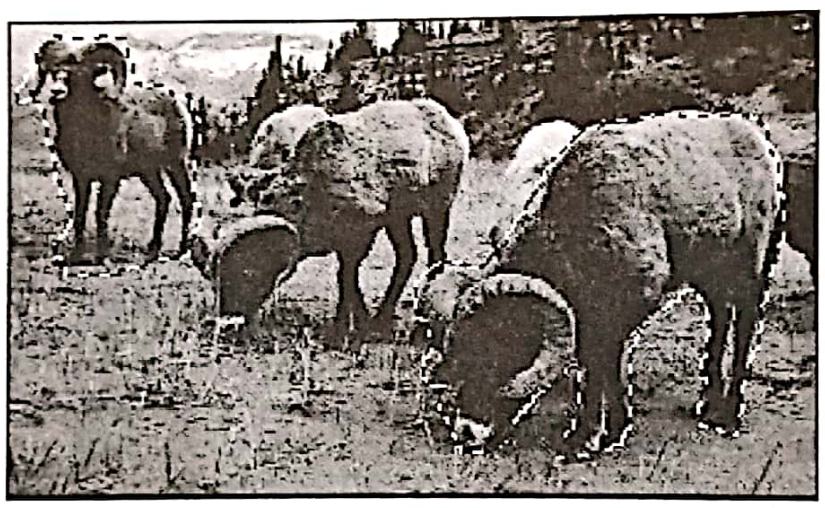


Figure 6.3: Add multiple areas selection

7. To subtract from a selection, hold down **CTRL**, or press the subtract from selection button in the context bar, and draw a shape that includes part of the original selection, or encircles it completely. (*See Figure 6.4*)

8. If you wish to apply edits or effects to the outside areas of the image, you need to click **Select | Inverse** to reverse the selected area.



Figure 6.4: Subtract from selection

Scanned with OKEN Scanner

Now the selection includes all the outside areas of the image with the central area outside the selection.

To Use the Magic Wand:

1. Select the **Magic Wand** button.
2. Click any color in the image.
3. Do any of the following:
   - Slide the **Threshold** slider to adjust the number of pixels to be included or excluded. The Threshold slider sets how similar a pixel needs to be to the one you clicked in order to be included in the selection.
   - Click the **Connected** checkbox if you want to include only pixels that are touching the one you clicked.
   - Hold down **SHIFT** and click around the selection to include more pixels.
   - Hold down **CTRL** and click around the selection to remove pixels from the selection.
4. From the **Wand type** drop-down list, select one of the following:
   - **Brightness**: Selects pixels similar in brightness to the area of the image you clicked.
   - **Color**: Selects pixels similar in hue and brightness to the area of the image you clicked.
   - **RGB:** Selects pixels with red, green, and blue combination similar to the area of the image you clicked.

To Use the Rectangular or Elliptical Selection Tool:

1. Select either **Rectangular** or **Elliptical** button.
2. Click and drag to draw the selection on the image.
3. Do any of the following:
   - To add to an existing selection, hold down **SHIFT** and drag another selection that includes any part of the original selection line. Release the mouse and the selection now includes the additional area.
   - To add multiple areas to your selection, hold down **SHIFT** and drag as many as you want. As long as none of the edges touch, you can continue adding areas to the selection.
   - To subtract from an existing selection, hold down **CTRL** and drag another selection that intersects with the one you want to reduce. Release the mouse and any part that was inside the new selection is removed.

6.1.5 Combining the Tools

You can use the Selection tools separately or in combination. For example, you can use the Magic Wand to select most of a particular color, then switch to the Freehand Lasso to add to, or subtract from, the selection. Or you could use the Rectangular selection to select a large general area, then refine it using the Freehand Lasso.

To Combine Selection Tools:

1. Choose a tool and make a selection.
2. Switch to another tool.
3. Press the **Add to selection** button.
4. Make another selection.

Scanned with OKEN Scanner

6.1.6 Using the Edit Brush

While the tools allow you to make a variety of global adjustments, sometimes you may only want to adjust specific areas of your photos. *For example*, instead of blurring an entire photo, you may want to blur certain areas, such as the background. To do this, and other local adjustments, you can use the *edit brush*.

The edit brush allows you to selectively edit your photo by simply brushing on the effects. The edit brush is essentially creating a mask that is used to determine which pixels will be affected by the current settings of the chosen operation.

To use the edit brush:

1. Select the tool you want to use.
2. Click the **edit brush** button to enter **Brushing mode** and open the Brush controls or, toggle
3. the Brush panel open and closed with the **B** key.
4. Specify the brush settings in the panel or on the fly as described in the Edit brush options.
5. Start painting the effects on your photo.
6. Adjust the sliders to get the effect you want.
7. Press done to apply your brush strokes.

Edit Brush Options

Add brush stroke Brush: While holding the left mouse button down.

Erase brush stroke Brush: While holding the right mouse button down.

Adjusts brush size: You can use the mouse wheel to adjust nib width or adjust the Nib Width slider.

Adjust Feathering: Use SHIFT + mouse wheel to adjust the amount of feathering or adjust the Feathering slider.

Pressure: Adjust the slider to control the strength of the brush.

Alternating between brush mode and cursor-applied tools: For filters that involve drawing or clicking on the image, and White Balance, hold down the **ALT** key to switch from Brush mode to using the tool.

Clear all brush: Removes all brush strokes from the image.

Invert all brush strokes: Toggle this option to invert brushed strokes. This makes brushed areas no longer brushed and untouched areas brushed. This is useful if you want the majority of the image brushed and a small section untouched. Simply brush only the small area you don't want affected and invert the brush strokes.

Load last applied brush strokes: Applies effects to the area most recently brushed and applied.

Show brush strokes: When this option is checked, your brush strokes will be displayed in the color shown. This is helpful when brushing a complex area with a subtle effect, as it can be difficult to tell if you have missed a spot

6.1.7 Text Tool

You can use the add text tool to add formatted text to a photo. You can also adjust the opacity of the text to create a watermark effect, which is useful for putting copyright information onto your photos.

To add text to an image:

1. Do one of the following:

   Click **Filter** menu highlight **Add** and choose **Text**.

   On the Toolbar, click the **Add** icon and choose Text from the menu.

Scanned with OKEN Scanner

2. Type the text you want to add in the **Text** field.
3. In the Font area, specify the font you want to use, the formatting options, such as italics or justification, and the color of the text. Drag the **size** slider to specify a point size, and then drag the **Opacity** slider to specify the transparency of the text.
4. Click and drag the text marquee to reposition it anywhere on your image, or drag the marquee's handles to resize it.
5. Select an option from the **Blend Mode** drop-down list to specify how you want the text to blend into the underlying image.
6. Choose one of the following options:
   - Select the **Bubble Text** checkbox, and set the bubble text settings as explained in the Table 6.1.
   - Note that other text effects cannot be applied to bubble text.
   - Select one or more of the effects, drop shadow, and bevel checkboxes to customize your text.
   - Click the settings button to adjust the settings for each of these options as explained in the table below.

Table 6.1: Add Text Options

Bubble Text Settings:	**Talk**	Encloses your text in a comic-book text bubble with a pointed stem.
	Thought	Encloses your text in a comic-book text bubble with a dotted stem.
	Stem Angle	Specifies the direction of the text bubble stem. Drag the arrow to adjust the angle
	Thickness	Specifies the thickness of the text bubble border.
	Length	Specifies the length of the text bubble stem.
	Fill	Specifies the background color of the text bubble. Click the arrow to select a color.
	Border	Specifies the color of the text bubble border. Click the arrow to select a color.
Effect Settings	**Effect**	Specifies the effect you want to apply to your text. Each effect has different options that control its appearance. Drag the sliders to adjust the appearance of the text.
Drop Shadow Settings	**Distance**	Specifies the distance between the drop shadow and the text. The higher the value, the farther the shadow and the higher the text appears.
	Blur	Specifies the amount of blur applied to the drop shadow. Sharp shadows tend to seem less realistic than those with a slight blur.
	Opacity	Specifies the darkness of the drop shadow where it falls on the image.
	Angle	Specifies the direction of the imaginary light source that creates the drop shadow. Drag the arrow to adjust the angle – note that this also adjusts the shading of the text bevel.

Scanned with OKEN Scanner

7. Do one of the following:
   - Click **Apply** to add the text to your image, and keep the Text tool open so you can add more text. The Text tool creates a new, empty marquee, and you can return to step 3 in the list above.
   - Click **Done** to add the text to your image, and leave the tool. If you still have an empty marquee open, it will be discarded.
   - Click **Cancel** to discard your changes and leave the tool.

6.1.8 Layers

You can use the layers pane to add effects and shapes to your photos in layers, perform photo manipulations, create composite images, and much more. Layers allow you to work on a single element of an image without disturbing others. Layers can be as transparent or opaque as you desire. This allows you to stack layers to achieve the level of visibility you want for the elements you add on each layer. You can also hide layers. You can use layers in conjunction with image effects and adjustment tools, drawing tools, text, and more. Effects and adjustments will be applied to the layer selected in the Layers pane. You can also create a blank image and layer other elements on top of it.

To Open the Layers Pane:

1. With an image open, go to View menu and choose Layers.
2. In the Layers pane, you will see Layer 1 with your image on it. At this point, you have the option of editing the photo, or adding additional layers containing other photos or drawing elements, such as shapes. (*See Figure 6.5*)

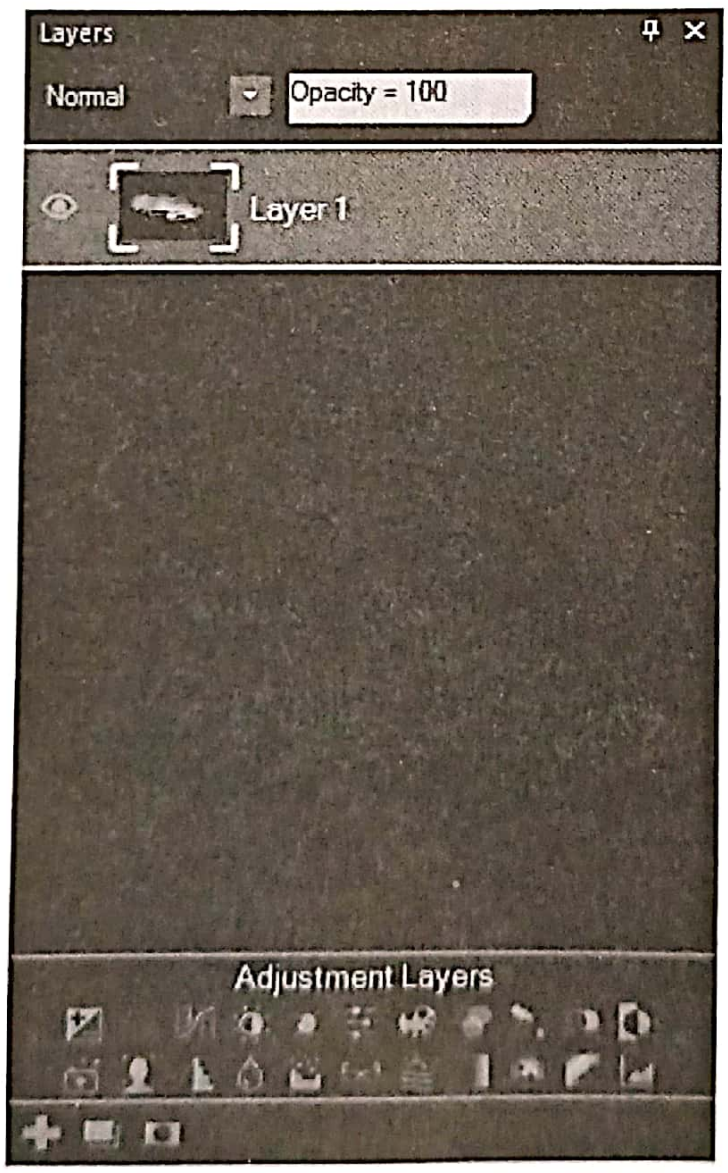


Figure 6.5: Layer pane

Scanned with OKEN Scanner

To Add a Layer:

1. Do one of the following:
   - Press the **Add Layer** button at the bottom of the **Layers** pane.
   - Go to **Layer** and choose **Add New Layer.**
2. To edit a layer, select the layer in the *Layers pane.*

To Set Layer Visibility:

1. Select the Layer and press the **Hide/Show Layer** button.

To Delete a Layer:

1. Select the layer and press the **Delete Layer** button.

To Duplicate a Layer:

1. Select the layer you would like to duplicate and press the **Duplicate** button.

To Move a Layer:

1. In the **Layers** pane, select the layer and drag it up or down.

To Name a Layer:

1. To rename a layer from "Layer 1", etc., select the layer and right-click.
2. Choose **Rename** Layer.
3. In the Rename Layer dialog, enter a name for your layer and press **OK.**

6.1.9 Brightness/ Contrast

You can use the Light EQ™ tool to adjust areas in an image that are too dark or too light, without affecting other areas of the photo. You can also simultaneously lighten dark areas that are too dark, and darken areas that are too bright. Examples would be a photo of a person silhouetted against a bright background like the sea, or a Window. In fact, most photos taken on a dull day, or with a flash, can be improved in various ways with fine adjustments. You can save your settings as a preset for future use.

There are three tabs each allows you to achieve results in different ways. In addition, there are several keyboard and mouse shortcuts that allow you to make adjustments on specific areas of the photo itself.

Basic tab: Is for very quick and easy adjustments using three sliders. Photo Editor analyzes the photo and varies the adjustment throughout the image automatically. For example, darker images are brightened more. By clicking the *Auto* button, you can have Photo Editor fully analyze the photo and set the slider positions automatically. You can also click directly on an area of the image to generate automatic settings optimal for that area, (usually the subject of the photo).

Standard tab: Works like a sound equalizer but with light. You can adjust the brightness and contrast of different tone bands (areas of relative brightness or darkness) of the image independently using a slider for each tone band. A graph shows the amount of brightening or darkening applied throughout the tonal range. The gray areas in the graph are suggested boundaries for adjustment to avoid clipping and loss of detail, and turn red to indicate where you have adjusted the sliders far enough to cause clipping.

Advanced tab: Gives you ultimate control of the brightness and contrast in an image. Basic adjustment curves can be constructed using four sliders, and you can manually adjust the curves by clicking and dragging within the graph area or on the image itself.

Using the Basic Tab

To Add Light to Dark Areas:

1. Drag the *Highlights slider* to the right to add light to the dark parts of the photo.

Scanned with OKEN Scanner

2. Adjust the settings using any of the controls in the Table 6.2.

To Darken Areas That Are Too Bright:

1. Drag the *Shadows slider* to the right to lighten and return detail to areas that are too dark.
2. Adjust the settings using any of the controls in the Table 6.2.

To Adjust Midtones That Are Too Light or Dark:

1. Drag the *Midtones slider* to the right to lighten and return detail to areas that are too dark.
2. Adjust the settings using any of the controls in the Table 6.2.

Table 6.2: Adjust the controls settings

Right-click on a slider	Right-click on a slider to reset it.
Exposure Warning	Click the Exposure Warning to make any overexposed or underexposed areas visible. Overexposed pixels turn red, underexposed pixels turn green. You can also toggle Exposure Warning on and off momentarily by holding down the E key.
Auto	Click to allow the software to analyze the photo and apply optimal settings based on the amount of light and dark pixels in the photo. Darker photos are brightened more than photos that are already bright.
Done	Click to apply your changes and close the tool.
Cancel	Click to discard all changes and close the tool.

6.1.10 Using the Standard Tab

The Standard tab has two sets of sliders. The top set is for brightening. The lower set is for darkening. The dark-to-light gradient on each slider indicates that you brighten that tone band when you move the slider up (top row of sliders) or darken that tone band when you move the slider down (bottom row).

Between the two sets of sliders are two horizontal bars graded from black to white. These bars indicate that sliders on the far left affect the dark tonal bands. The sliders on the right affect light tonal bands. Moving a slider changes the light in that particular tonal band. In the middle of the bands are two graphs that change as you adjust the sliders to represent the amount of darkening and lightening throughout the tonal range of the image. The two gray graphs show the maximum amount of brightening or darkening that you can apply before clipping begins. If you push a slider too far, the clipping is shown in bright pink.

To Use the Standard Tab:

1. Do one of the following:
   - Drag the tone band sliders up or down depending on which light bands you want to adjust.
   - Type the exact number into the number boxes or increment the numbers to make precise adjustments.
   - Click the # *Tone* drop-down list to increase or decrease the number of tone band sliders.
   - Click (or press E) to see any overexposed or underexposed pixels.
   - Click the **Auto** button to apply settings automatically.
   - Drag the **Darkening slider** to darken the light areas of the image.
   - Drag the **Contrast slider** to increase the contrast in the image.
   - Drag the **Fill Light slider** to lighten dark areas of the image.

Scanned with OKEN Scanner

- Left-click and drag upwards to brighten the dark areas of the photo. Try to prevent clipping
- Right-click and drag downwards to darken the bright areas of the photo. Try to prevent clipping

2. Click **Done** to apply your settings and close the tool. or click **Cancel** to discard your changes and close the tool.

6.1.11 Improve Colors and Tone

You can use the *Auto Levels* tool to automatically correct the exposure levels of your images. Auto Levels makes the darkest image pixels darker, and the brightest pixels brighter.

To Automatically Correct an Image's Levels:

1. Click **Filter** menu highlight **Exposure/Lighting** and select **Auto Levels.**
2. On the **Toolbar**, click the **Exposure/Lighting** icon and choose **Auto Levels** from the menu.
3. Select one of the following options:
   - **Auto Contrast and Color:** Adjusts color differences, brightness, and balances RGB channels.
   - **Auto Contrast:** Adjusts the color differences and brightness only.
   - **Auto Color:** Balances the RGB channels in the image, without changing the brightness or contrast.
4. Use the Strength slider to fine tune the amount of exposure you want to apply.
5. Click **Exposure Warning** to make overexposed and underexposed pixels visible. Overexposed pixels turn red, underexposed pixels turn green.
6. Do one of the following:
   - Click **Done** to accept your changes and close the panel.
   - Click **Cancel** to discard your changes and close the panel.
   - You can right-click a slider to reset to the default value.

Adjusting Tone Curves

You can use the Curves tool to change the tonal range of an image. Select the RGB color channel to adjust the entire range of the image, or select a specific color.

To Adjust the Brightness Curves in an Image:

1. Do one of the following:
   - Click Filter menu highlight Exposure/Lighting and choose Tone Curves.
   - On the Toolbar, click the Exposure/Lighting icon and choose Tone Curves from the menu.
2. Set the options as described below.
   - Click **Apply** to apply your changes.
   - Click **Done** to apply your changes and close the *Curves* tool.
   - Click **Cancel** to discard all changes and close the tool.

6.1.12 Filters

You can use filters to clean up or retouch image or photos. To apply special art effects that gives an image the appearance of a sketch or impressionistic painting, or creates unique transformations using distortions and lighting effects. The filters provided by Photoshop CS6 appear in the Filter menu.

To use a filter, choose the appropriate submenu command from the Filter menu. These guidelines can help in choosing filters:

Scanned with OKEN Scanner

- Filters are applied to the active, visible layer or a selection.
- For 8 bits per channel images, most filters can be applied through the *Filter Gallery*. All filters can be applied individually.
- Filters cannot be applied to **Bitmap-mode** or **indexed-color images.**
- Some filters work only on **RGB images**.
- All filters can be applied to **8-bit images.**
- The filters that can be applied to 16-bit images are: Liquify, Vanishing Point, Average Blur, Blur, Blur More, Box Blur, Gaussian Blur, Lens Blur, Motion Blur, Radial Blur, Surface Blur, Shape Blur, Lens Correction, Add Noise etc.
- The following filters can be applied to 32-bit images: Average Blur, Box Blur, Gaussian Blur, Shape Blur, Surface Blur, Add Noise, Cloud 1, Smart Sharpen and Unsharp Mask.

Conclusion

This chapter discussed with Photo Editor with different features of editing tools such as selection tools, text tools, layers, brightness/contrast, Improve colors and tone and filters.

REVIEW QUESTIONS WITH ANSWERS

1. Multiple choice

1.1 Which of are following is the latest version of Photo editor?

a. Version 7 b. Version 8

c. Version 9 d. Version 10

1.2 How many selection tools are available in photo editing?

a. Two b. Three

c. Four d. Five

1.3 Which tool is used to trace a freehand selection like a drawing with a pencil?

a. Lasso tool b. Free hand

c. Drawing tool d. Edit brush

1.4 Which tool is used to add formatted text to a Photo?

a. Text tool b. Edit tool

c. Image tool d. None of these

1.5 Standard tab has _____ sets of sliders.

a. Two b. Three

c. Four d. Five

2. State whether the following statements are True or False

2.1 Adjust color difference, brightness and balances RGB channels through auto contrast.

2.2 Basic adjustment curves can be constructed using four slides.

2.3 Rectangular selection Tools is a type of lasso tool.

2.4 Magic hand tool is also known as Magic Wand.

2.5 The shortcut key of magic wand tool is W.

Scanned with OKEN Scanner

3. Fill in the blanks

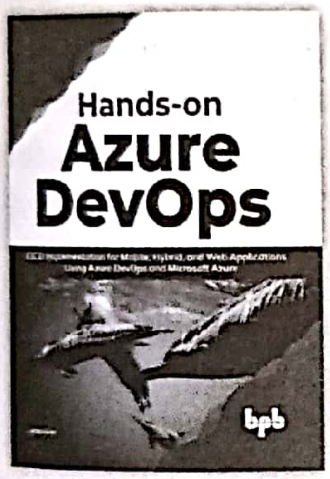
a. Rectangular b. Brush c. Elliptical
d. Nib width e. Clear all brush

3.1 _____ is a type of selection tool.
3.2 _____ is a basic photo editing tool.
3.3 _____ tool creates __shapes and paths.
3.4 _____ strokes remove all brush strokes from the image
3.5 _____ adjust the size of the brush

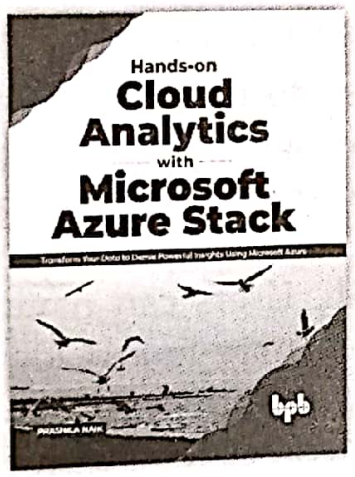
Answers

1.1	d	1.2	d	1.3	b.	1.4	a	1.5	a
2.1	F	2.2	T	2.3	F	2.4	T	2.5	T
3.1	c	3.2	b	3.3	a	3.4	e	3.5	d

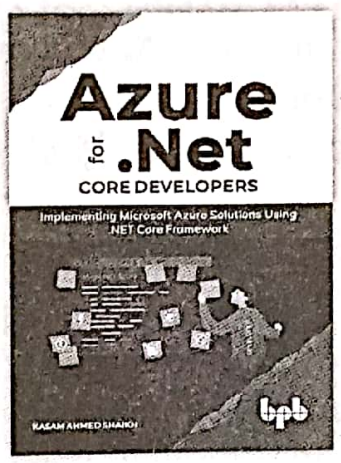
OTHER TITLES OF INTEREST



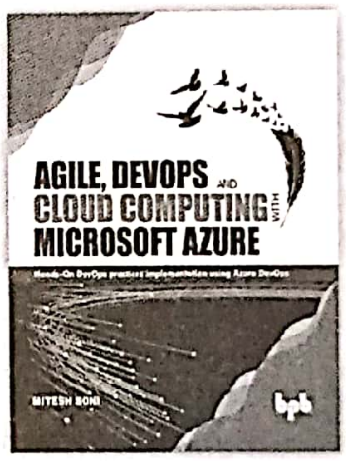
Hands on **Azure DevOps**
ISBN: 9789389845341



Hands-on **Cloud Analytics** with **Microsoft Azure Stack**
ISBN: 9789389898149



Azure for **.NET**
Core Developers
ISBN: 9789389328448



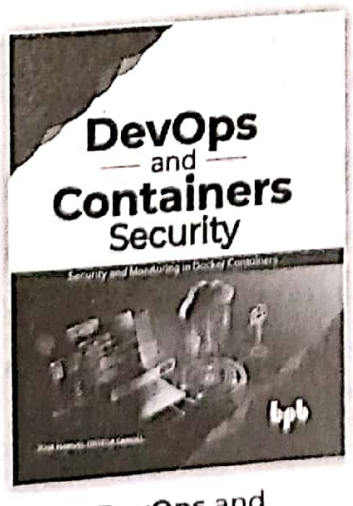
Agile, DevOps and **Cloud Computing** with **Microsoft Azure**
ISBN: 9789388511902



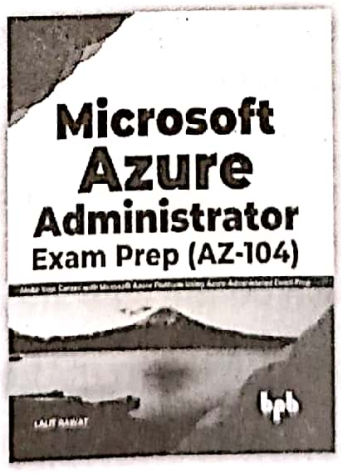
Mastering **Azure** Serverless Computing
ISBN: 9789389898163



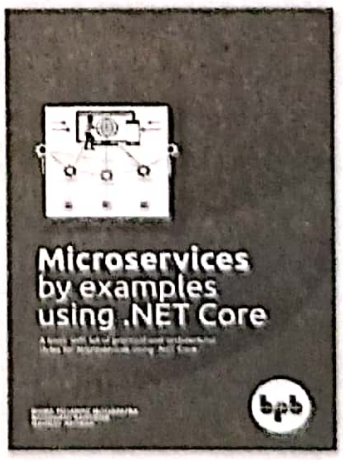
Learn Microsoft **Azure** Step by Step in 7 days for .NET Developers
ISBN: 9789388511384



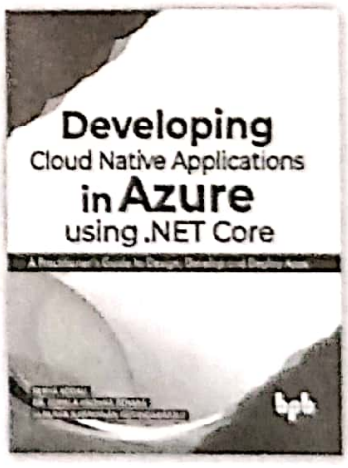
DevOps and **Containers** Security
ISBN: 9789389423532



Microsoft Azure Administrator Exam Prep (AZ-104)
ISBN: 9789389898767



Microservices by Examples Using .NET Core
ISBN: 9789387284586



Developing Cloud Native Applications in **Azure** using .NET Core
ISBN: 9789389328745

Scanned with OKEN Scanner

CHAPTER 7

Web Publishing and Browsing

7.1 Introduction

Web Publishing is the process of publishing content on the internet. It includes creating and uploading websites, updating the associated web pages, and posting content to these web pages online. The content meant for web publishing can include text, videos, digital images, artwork and other forms of media. Publishers must possess a web server, a web publishing software, and the internet connection to carry out web publishing. And the web browser is a software application that allows people to access, retrieved and view information on the Internet.

7.2 Overview of World Wide Web

The World Wide Web (WWW) is the most popular and promising method of accessing the Internet. The WWW is an information system designed specifically for posting useful information which can be used with ease and with document interchange in mind. **HTML (HyperText Markup Language)** is developed for this purpose. HTML was based on a subset of the **Standard Generalized Markup Language (SGML)**. To transfer HTML documents to remote sites, **HTTP (HyperText Transfer Protocol)** was devised. HTTP is used for interaction between computers on the internet. Any computer on the Internet that uses the HTTP protocol is called a **web server**, and any computer that can access that server is called a **web client**. The use of the client/server model and the HTTP allows different kinds of computers on the Internet to interact with each other. *For example*, a UNIX workstation may be the web server and a Windows PC (Personal Computer) may be the web client if both of them use the HTTP protocol for transmitting and receiving information.

The purpose of web browser is to connect to web servers, request documents, and then properly display those documents. Web browsers can allow you to upload/download files that are not meant to be displayed, and in some cases, it can allow you to send and retrieve e-mails. The main purpose at which web browsers are best is formatting and displaying the web pages. Each web pages is written in a language called HyperText Markup Language, popularly known as HTML. Users can navigate through a global network of resources at the click of a button. HyperText documents are linked together by keywords or graphical icons in the document. When a new word or idea is introduced, you can jump to another document containing complete information on the new topic. Readers see links as highlighted keywords or images and use these links to access additional documents or resources. A link is a special type of item in a hypertext document that connects one document to another and provides more information about the linked item.

Using the web's hypertext facilities, you have the freedom to supply information to readers in powerfully innovative ways. Web publications, these days, use many of the web's graphical, interactive, and multimedia features. However, ways to publish on the web are changing continuously.

7.3 SGML (Standard Generalized Markup Language)

HTML is developed as a subset of SGML, which is a high-level markup language. Like HTML, it defines different components of a document and describes formatting and hypertext links. HTML is much simpler than SGML and best suited for transmission over the Internet. SGML is not ideal for transmission across the internet for many types of computers, users and browser applications.

Scanned with OKEN Scanner

Web publishing technologies have influenced how documents are produced, stored and distributed. Documents are now increasingly available in electronic formats but the excess of formats has itself become a problem. The Standard Generalized Markup Language (SGML) was developed to provide a single interchange format and a sophisticated structuring format that would provide great capability for document processing. It is a technology that allows additional information to be added in electronic documents so that the value of documents is maximized. This includes the ability to manage, access, automate, and detect structural errors.

SGML uses a structural markup which enables the description of structured information independent of how the information is processed. Every SGML-based document requires descriptions or a set of rules about structured information in a document. This description is called Document Type Definition (DTD) and the SGML language provides a standard syntax for expressing DTDs. Any information that are marked up or added into an SGML document must follow the descriptions in a DTD. In other words, DTDs defines the specifications for valid components/parts within documents and rules about how subparts are organized.

A DTD is the core element of SGML documents. The DTDs consist of the definitions about SGML markups (tags) allowed in a document. The definitions also include a formal description of the document and the relationship of the elements, such as chapters or footnotes within the document. A marked-up element, an "element", has a start tag <Q> , content, and an end tag </Q>. For example, a title might be marked with the title tag as <TITLE> Document Processing </TITLE>. Markups can be for graphics, images, and other entities as well as text.

While the construction of a DTD and the markups of a document involves many considerations and can be extraordinarily complex, it is not unfair to think of an SGML document as consisting of three things. We believe this is a simple way to think about the SGML:

a. **An SGML declaration** that specifies parameters that can be used to determine whether an SGML parser can interpret a document. For most users, SGML declarations are invisible.

b. **A DTD** contains definitions for allowable markups. In practice, it may be a referent to a location where a DTD source is located rather than prepended entire definitions. A DTD specifies a general model for document elements.

c. **A document instance** consists of structured marked-up text, elements. It is easier to imagine that the contents of a document instance may change. A DTD is written such that an infinite number of document instances can be generated. In other words, one DTD could control a thousand documents that look different.

7.4 Web Hosting Basics

Hosting is also known as Web hosting or Website hosting. Web hosting is a service that allows individuals and organizations to have a Website made available via the World Wide Web (WWW). Web hosts are usually based of a data center that contains servers used to store the data that makes up the client Websites.

At the most basic level, clients upload their website files to the host server via File Transfer Protocol (FTP) or a web interface that mimics FTP functions. Web hosts may own their own servers or may simply rent servers from other web hosts and resell the space. Most web-hosting services are shared, meaning that multiple websites owned by different clients share space on the same server. Dedicated hosting is also available, in which the client rents an entire server for itself that it has full remote control over, but this is generally much more expensive than shared hosting. Managed hosting is a slightly less expensive service that gives a client a dedicated server but only limited control over its function.

A lot of people want to know how to host their website so here is the step-by-step tutorial for hosting your website. There are three basic requirements to host your website these are listed below.

Scanned with OKEN Scanner

a. Your website files
b. A domain name
c. Hosting space to store your files
d. A FTP client (FileZilla)

Now let us discuss step-by-step for the tutorial.

Step 1: You need to buy a suitable domain name for your website which will be the primary URL of your website. You can buy a domain name from GoDaddy. GoDaddy is the internet domain name registrar and web hosting company which manages the reservation of the internet domain names. You also need a hosting space to store your website files and for that you can buy a paid hosting space or you can use some free Web hosting providers to host your files. To buy a paid hosting space for your Website you should get better support and flexibility. A paid web hosting space may cost you around $5 per month.

Step 2: After completing step one you have your website hosting space and your domain name. So you need to print your domain name to hosting space. For that you need know the hosting accounts name servers. You can ask it from your website hosting provider. After getting the name servers you need to go to your domain control panel. There you will get an option to manage DNS. If you are unable to find the DNS manager you can ask your domain providers. So, under DNS manager you need to submit the name servers provided by your website hosting provider. It may take up to 48 hours to point your domain name to your hosting space.

Step 3: You need to prepare your website files at this step. If you know how to create a website then fine otherwise you can hire a website designer to create and design the website. If you already have the files, then you need to set up a FTP user in your hosting panel. You also need a FTP client to upload your files however most web hosting providers have their own program to upload files but it will be a lot easier to transfer your website files using a FTP client. The most popular FTP client is FileZilla. You can get it free from their website. Install FileZilla and login to your hosting space using your FTP login information. Using FileZilla you can drag and drop your file to your hosting space.

7.4.1 Hosting Types

Generally, there are three different types of web hosting: Shared hosting, Virtual Private Server (VPS), and Dedicated. While all types of hosting servers will act as a storage center for your website, they differ in the amount of storage capacity, control, technical knowledge requirement, server speed, and reliability.

Shared hosting: With shared hosting, several web hosting customers share the same computer. All of the websites of all the different accounts are stored in the same drive, processed by the same CPU, and delivered by the same web server.

Virtual Dedicated Hosting: This type of hosting is better for medium size business. With virtual dedicated hosting, you will have a dedicated bandwidth and dedicated RAM for your site. You will be given a root ID and password to maintain your web server. This type of hosting is created on a single server, but it is managed in such a way that every user will have dedicated speed and bandwidth. This is bit more expensive but really good one for medium size business.

Dedicated Server Hosting: With dedicated server hosting, you have complete control over an entire server. This has many advantages, they are more expensive than virtual dedicated hosting and should be considered when you have a very high traffic requirement.

7.4.2 Basic Features of a Web Hosting

The basic features that most of the website hosting provide:

Scanned with OKEN Scanner

Disc Space: Disk space means the amount of storage space provided to you by your web hosting provider. You need disk space to store your web files composed of text, images, video, audio, etc. disk space is measured in megabytes and a good hosting company will provide packages with differing amounts of space to cater to the needs of all sizes of websites- from smaller personal websites to larger company websites.

Monthly Traffic: A small or medium website will need between 1GB and 10GB of data transfer on a monthly basis. If you plan to keep a lot of audio and video on your website, then you need a plan with more data transfer capacity. Check different options based on your requirements. What are the other options available in case you cross the given data transfer limit. Your site should not be stopped in case you exceed given limit.

Bandwidth: Bandwidth means the amount of data that a website can transfer over a period of time. It determines the speed of your website. More bandwidth means more speed. The less bandwidth your site has, the slower it takes for it to load. Some hosting service providers provide unlimited bandwidth whereas some others have different prices based upon the amount you use. If your website has heavier pages like those with lots of images, videos etc. then you will need higher storage along with greater bandwidth.

Connection Speed: Nowadays, most service providers allow very fast connection speed. So choose a service provider who is giving better connection speed in terms of bits per second. You can have a connection speed ranging from **64KB** per second to **2.488GB** per second.

Email Accounts: Make sure you are going to get sufficient number of e-mail accounts. There are many other options available which come along with your e-mail account. Like, will you get IMAP, POP and E mail forwarding options available along with your e-mail facilities.

E-mailing Support: Apart from having e-mail accounts, it is also very important that your web server should have a facility to send e-mails from back-end. In case your site visitors want to contact to you using a form, then you will be able to use that e-mailing facility to send e-mails to your designated account. In simple terms, you should make sure that the SMTP Server is setup and working on your web server.

Programming Services: The website hosting packages also let you create web pages with programming languages including **HTML, PHP, ASP** as well as databases. The best offers can be called those providing **PHP** language and **MySQL** database.

Databases: There are many databases available **MySQL, Oracle, SQL Server,** etc. You should choose your server based on your database requirement. If you are buying space on a shared server, then you need to verify how much space will be allocated for your database. Many ISPs do not give more than a limited space for databases. If your site needs a lot of database size, then you should go for a virtual dedicated server.

Uptime: Uptime means the percentage of time that a hosting server stays up or running. 99.99% uptime would mean that your website will go down only for about 8 hours in a year while 98% uptime would mean that your website may remain down for about 7.3 days in a year.

Backup & FTP: Make sure your Service Provider is giving you more ways of taking regular backup of your website. If your site is changing every day, then it becomes very important that you should take regular backup of your website.

Control Panel: Make sure what type of facilities you will get to maintain your hosting account. Check if your service provider is providing you an easy-to-use Control Panel or some other similar tool. Using a Control Panel, you should be able to maintain basic operations related to your website such as logging your service request, your reboot request, or any other problem.

Customer Service: This is one of the basic and most essential features that one should look for while selecting website hosting service. Customer support options like those provided through chat, phone and e-mail etc. are extremely useful in the event of any technical or other problems related to your website.

Scanned with OKEN Scanner

ISP: To get your site, it is necessary to sign up with an **Internet Service Provider (ISP)**. The Internet Service Provider is a company that offers internet access to individuals and businesses for a monthly or yearly fee. ISPs support one or more forms of internet access, ranging from traditional modem dial-up to DSL and cable modem broadband service to dedicated T1/T3 lines. In addition to internet connection, ISPs may also provide related services like website hosting and development, e-mail hosting, domain name registration, etc.

ISPs, offer have different types of Internet connections namely, dial up, cable and DSL (Digital Subscriber Line) broadband. Hardware, such as dial up modem or a wireless modem and router are usually provided by the company. When you register with an ISP for its services, an account is created and you are provided with the login details –username and password. You connect to the Internet via your account and this way the company keeps a watch on your online activities.

ISPs can be both regional, and national, servicing the entire country and they are connected with each other through Network Access Points (NAPs). ISPs are also referred to as Internet Access Providers.

Guide to Web Publishing on the Internet

One of the methods that you may use is CuteFTP from: **ftp://ftp.cuteftp.com/pub/cuteftep/**

First, you create the pages for the website, then choose **File ->** Save As HTML, enter the necessary folder name and click Save. Your web page designer such as FrontPage 2000, saves it as an HTML document in the folder you specified.

Now start CuteFTP software, that is easy to use and reliable. You then upload all the files from the folder to the folder on the internet web server that has reserved space for your site. It is a simple process that may take about 10 minutes.

7.5 Document Interchange Standards

Every network and every computer in the networks exchange information according to certain rules called protocols. These different computers and different networks are united with the common thread of protocols to exchange their information over the network.

7.5.1 Connectionless and Connection-Oriented Protocols

With a connectionless protocol, clients connect to the server, make a request, get a response, and then disconnect. For the internet, **HTTP** is a **connectionless** protocol.

With a **connection-oriented protocol**, clients connect to the server, make a request, get a response, and then maintain the connection to handle future requests. An example of a connection-oriented protocol is **File Transfer Protocol (FTP)**. When you connect to an FTP server, the connection remains open even after you download a file. The maintenance of this connection consumes system resources. A server with too many open connections quickly gets overloaded. Consequently, many FTP servers are configured to allow only 250 open connections at one time, so only 250 users can access the FTP server at a time.

Difference between Connection-Oriented and Connectionless Protocol

Connectionless protocols differ from connection oriented protocols in the way requests and responses to requests are handled. HTTP is a connectionless protocol. When clients connect to the HTTP server being a connectionless protocol, they make a request, get a response, and then disconnect. Because the connection is not maintained, no system resources are used after the transaction is over.

Consequently, HTTP servers are limited only by active connections and can generally handle thousands of transactions with low system overhead. The drawback of connectionless protocols is that when the same client requests more data, the connection must be re-established. To web users, this means a delay whenever they request more information.

Scanned with OKEN Scanner

7.5.2 Stateless and Stateful Protocols

There are two kinds of protocols for the processed transactions on the internet. These are stateless and stateful protocols. In a **stateless protocol**, no information about a transaction is maintained after a transaction has been processed. In a **stateful protocol**, state information is kept even after a transaction has been processed.

Servers using stateful protocols maintain the following information:

- Status of the connection
- Processes running
- Status of the processes running

Generally, this state information resides in the memory and consumes system resources. When a client breaks a connection with a server running a stateful protocol, the state information has to be cleaned up and is often logged as well.

Difference between Stateless and Stateful Protocols

Stateless protocols differ from stateful protocols in the way information about requests is maintained.

HTTP is a stateless protocol. Stateless protocols are light because servers using them keep no information about completed transactions and processes. When a client breaks a connection with a server running a stateless protocol, no data has to be cleaned up or logged off. By not tracking state information, there is less overhead on the server, so it can generally handle transactions speedily.

With HTTP, you can transfer full-motion video sequences, stereo sound tracks, high-resolution images, and any other type of media. The standard that makes this possible is **MIME (Multipurpose Internet Mail Extensions)**. HTTP uses MIME to identify the type of object being transferred across the internet.

Object types are identified in a header field that comes before the actual data for the object. In HTTP, this header field is the content-type. By identifying the type of object in a header field, the client receiving the object can handle it appropriately.

7.6 Components of Web Publishing

After creating a website, to make your website available to everyone on the web, you need to publish it on the the web server. You can either your own server to send your files to someone or use the web server of the ISP.

Web Browser

A web browser is a software package supplied free by Microsoft Corporation and many other companies. It is used to view pages as well as navigate the WWW. Browsers are also referred to as web clients.

Web Cookies

Web cookies refers to a piece of information sent by a web server to a web browser that the browser software is expected to save and to send back to the server whenever the browser makes additional requests from the server. Web cookies (technically called Persistent Client State HTTP Cookies) allow a website to store information about you as well as to track your visits to it over a prolonged period of time. To accomplish this, the website stores information in a file in your PCs web browser. This information can then be checked by the website when you make future visits to the site.

In this manner, cookies provide a level of audience tracking to the Web.

Web Server

A Web server is a program that runs on a website and is responsible for replying to web browser requests

Scanned with OKEN Scanner

for files. You need a web server to publish documents on the web.

7.7 Document Management

The action of identifying, acquiring, processing, storing, updating and distributing the document is called **document management**. It may include file organization, uploading, etc. These are described in the following sections:

File Organization: When organizing the files on your site, keep files in separate folders. That way, the site is easier to update. The site maintains separate folders for images, HTML files, video, sound and **CGI (Common Gateway Interface)** scripts. The site has different folders name too; so, files related to the jobs section are arranged and put in a folder called jobs.

Uploading: Uploading means transferring files from your computer to the internet server. When you are uploading files from your site, do not forget that you need to upload them in the correct file format. HTML and text files should be uploaded in **ASCII (American Standard Code for Information Interchange)** format, while images, CGI scripts, Flash files, sounds and video should all be uploaded in the binary format.

File Names: When you are naming the files for your site, stick to what is known as 8.3 file names, i.e., 8 characters for file name and 3 characters for extensions. Some people may still be looking at your site on older machines. Use the *.htm ending rather than *.html. You would like as many people as possible to see your website, so do not exclude those who cannot see long file names.

Backups: Whatever you do, be sure to back up your files on your website. Put them on to a **Zip disk** or **CD (Compact Disk)** and then store it away in a safe place. If you lose everything, at least you have a backup ready to upload the site again.

7.8 Web Page Design Consideration and Principles

The following are some principles about web page design which will help you in designing an effective web page:

- Present your web page in a way that users are impressed within the first few seconds after they arrive at the website.
- The title of the page should be short, fascinating and descriptive.
- Keep web pages fresh and updated. Remove any outdated information and add new information from time to time.
- Keep your Web pages focused. Don't show everything on one page. Use separate pages for separate topics.
- Make sure buttons and other links behave as the viewers expect them to do.
- Make sure that the website avoids such features that are not supported by all the browsers.
- Use compressed images so that a web page gets quickly loaded.
- Test your web page before publishing to make sure that every link and other features work properly.

7.8.1 Planning your Website Design

When you decide to have a website for your business or personal interest there are a number of things you have to consider before you start actually building your website. Website planning has various steps:

- Creating goals for your site
- Organizing the site structure
- Creating your design look

Scanned with OKEN Scanner

- Designing the navigation scheme
- Planning and gathering your assets

7.8.2 Creating Goals for your Site

The first step is determining the goals of your website. Before even considering the practical aspects of creating a website, there are some important fundamental questions you need to ask yourself first. *For example,* what do you want to achieve through this website, and who exactly is this website for? Who's your target audience, and how do you best communicate with that audience? What information needs to be included, and how should that information be structured.

Your website plan outlines the purpose and goals of your website. It should address the following issues.

Things to Consider:

- Purpose of Website
- Target audience
- Website content
- Ways to bring in new visitors, repeat visitors
- Budget
- Other costs such as ISP, domain name and registration, database information to make available online and search engine registration and optimization.

Organizing the site structure

The basic aims of your website, you need now to consider how the content (text, images, etc.) should be organized and structured. Probably the most important aspect of website **structural design** is how you breakdown the content into logical sections (Main Areas, Pages, Headers, Sub-Headers, Lists, etc. as shown in Figure 7.1). As a general rule, things should be kept short. You need to create a strong hierarchy for the site and breakdown content into small units.

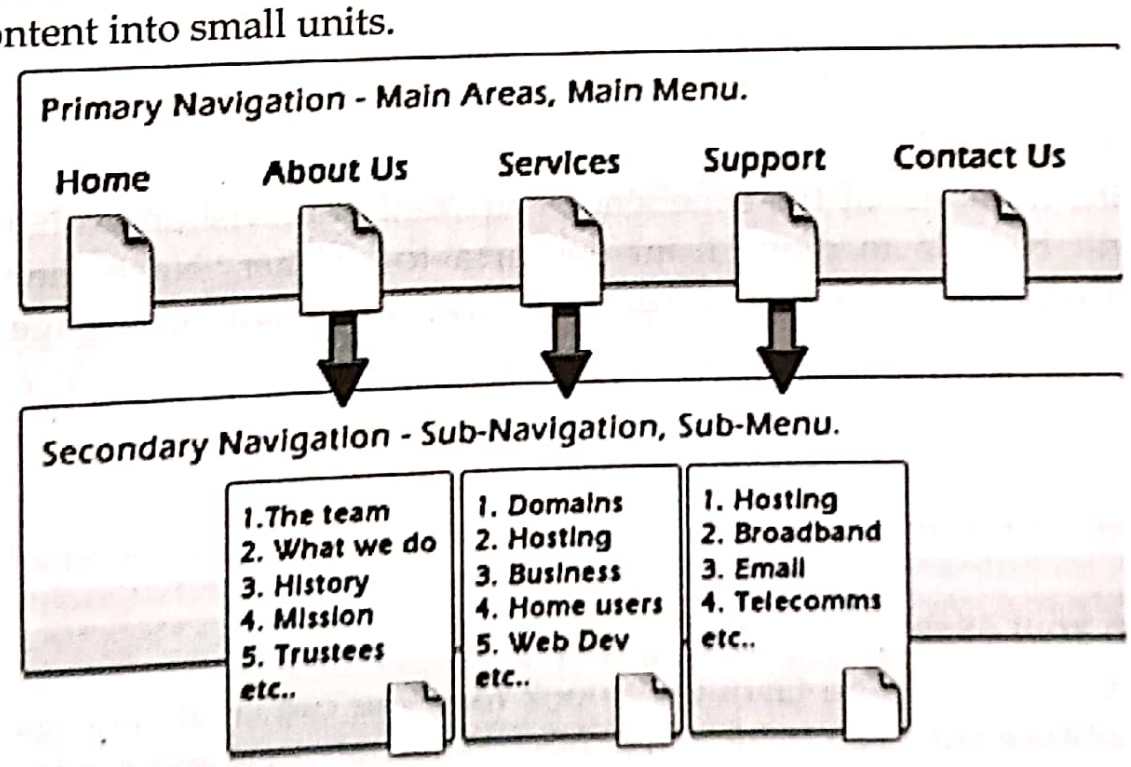


Figure 7.1: Organize the site structure

It is a good idea to create a graphical flowchart/sitemap for the site. This can help you visualize a logical hierarchy, and to see how easily information will be accessed. It will also help others to understand how your website is structured.

Scanned with OKEN Scanner

Creating your design look

You can save a lot of time later if you plan your design and layout before you begin working in Dreamweaver. You can simply create a mock-up drawing on paper of how you want the site layout to look and then follow it as you build your site; or you can create a composite drawing of your site using software, such as **Macromedia FreeHand** or **Macromedia Fireworks**.

Maintaining consistency in your page layout and design helps to ensure a good user experience. The user should be able to click through the pages in your site without getting confused. If all the pages have a different look, or the navigation is in a different place on each page, it might frustrate the user. Make sure your site provides a consistent look for your user.

Some of the factors you can take into consideration to improve the look and feel of the website are:

a. **Space and Balance:** Everything should look proportionate and proper.

b. **Colour:** The colours should look pleasant to the eyes.

c. **Font Type and Size:** Typeface of the text and size of the text in the web pages should be comfortable to read and make the matter easy to understand.

d. **Textures:** Background graphics or textures can be annoying while reading text. So be very careful and selective.

e. **Shapes:** Shapes make a site distinct. Therefore, include pleasant shapes.

f. **Special Effects:** Animation can add to the personation and you may like to include them in the website design. But they add to the size of the site and, therefore, may cause delay while accessing through a browser with low-speed modems.

g. **Consistency:** Using a single colour scheme throughout is a good way to achieve consistency, like the **Red Fort** in Delhi has consistency of the colour of the stone used throughout the outer boundary wall of the fort.

h. **Variation:** When you visit Red Fort from inside, you would find many variations depending on the building you are visiting. The same holds true for website designing. Different pages may appear exclusive based on the type of message you plan to convey.

Designing the navigation scheme

As you design your site, think about the experience you want your visitors to have. Think about how a visitor to your site will be able to move from one area to another. Navigation should be consistent throughout your site. If you place a navigation bar across the top of your home page, try to keep it there for all the linked pages.

A major aspect of usability for a website is the need for people to be able to find what they want, and quickly. Logical, practical and sensible navigation is essential if your site is to succeed.

Present clear, consistent, well-ordered navigation (main menus, primary links to pages), and include Sub-navigation if necessary. Include search facilities too if you have a large site that include forums, blogs etc.

Planning and gathering your assets

Once you know what your design and layout will look like, you can create and gather the assets, such as images, text, or media. Make sure you have all of these items gathered and ready to go before you begin developing your site.

If you are using images and graphics from a ClipArt site or someone else is creating them, make sure you collect them and put them in a folder on your site. If you are creating the assets yourself, make sure you create them all before you start development, including any images you need if you are using roll-overs. Then organize your assets so you can access them easily while creating your site.

Scanned with OKEN Scanner

Basic Website layout and structure

Your basic website structure should follow a logical pattern of organization. It easy for visitors to search for the information they are seeking and for search engines to index your website. Also, the basic layout of a website that is **logically organized** will be much easier to maintain and edit.

The basic structure and layout of a Website

The layout of your website should look as shown in Figure 7.2. The website basic divisions shown below reflect the linking network must use by web masters.

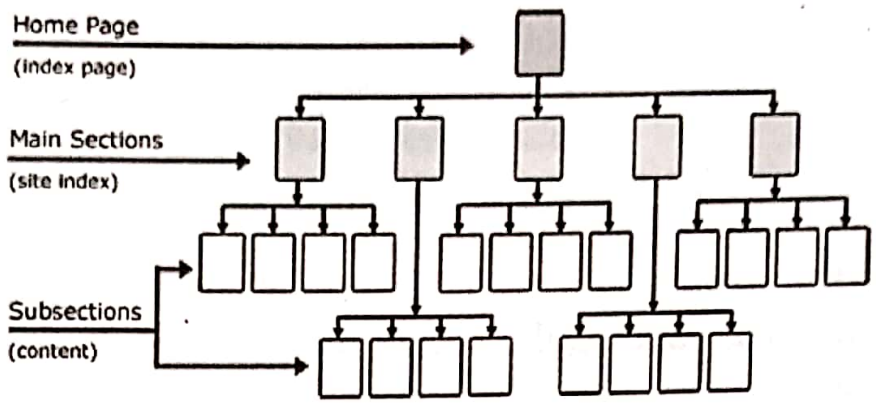


Figure 7.2: Basic of a Website layout

The basic divisions of a **Website layout** are as follows:

Home page: The home page is the first tier of website content and leads into the interior of the website. It links directly to the main sections of a site. In other words, from the home page links are introduced that access the internal website sections or main subject areas of a website.

The homepage is first introduced to your website (See Figure 7.3). As you go about designing your website it put a large part of your efforts into perfecting this home page. It is important that the home page clearly conveys the essence of what your website is all about.

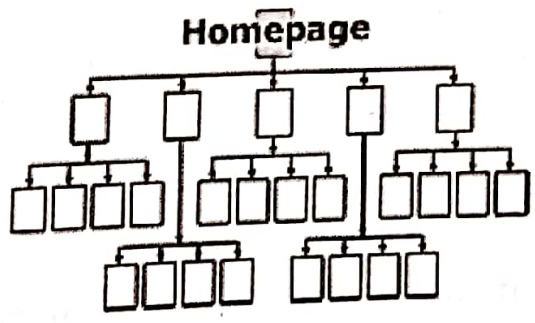


Figure 7.3: Home page

Main Sections: The main sections or subject areas of a website should be identified in this second layer of content (See Figure 7.4). Here is where the people visiting your website should be able to access more detailed information on the topic in which they are interested. This collection of documents should begin a logical progression of thought that leads deeper into the website content. From this level of data links should be provided that interface with the main body of the website's material or subsections.

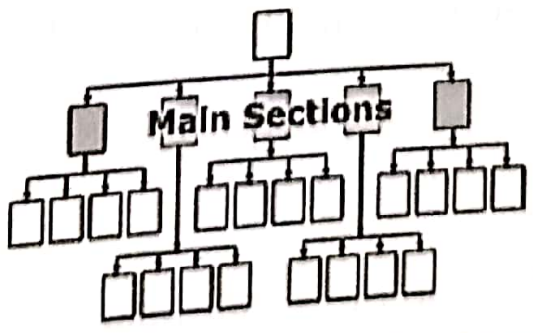


Figure 7.4: Main Sections

Scanned with OKEN Scanner

Links coming from the home page should lead to the main sections of your website. In other words, your index page should provide links that interface with specific subject areas of website content.

Subsections: Subsections should be placed in their proper categories and the content tied to the main sections of the website (See Figure 7.5). This is where the main body or database of your website content resides.

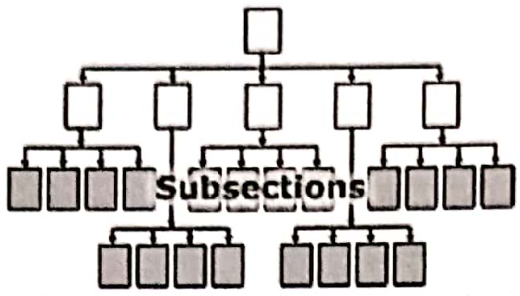


Figure 7.5: Subsections

The subsections of a website are the areas in which a collection of supporting material for the website's main sections is accumulated. This is the heart and soul of your website. These basic divisions of a website should also be in a constant state of expansion.

7.8.3 The Website Design and Development Process

There are numerous steps in the website design and development process. From gathering initial information, to the creation of your website, and maintenance, you need to keep your website up-to-date and current.

The exact process will vary slightly from designer to designer, but the basics are the same, which are discussed below.

- Information gathering
- Planning
- Design
- Development
- Testing and delivery
- Maintenance

Information Gathering: The first step in designing a successful website is to gather information. It is actually the most important one, as it involves a solid understanding of the company it is created for. It involves a good understanding of you – what your business goals and dreams are, and how the web can be utilized to help you achieve those goals.

Certain things to consider are:

Phase One

Purpose: What is the purpose of the site?

Goals: What do you hope to accomplish by building this website? Two of the more common goals are either to make money or share information.

Target Audience: Is there a specific group of people that will help you reach your goals? It is helpful to picture the "ideal" person you want to visit your website.

Content: What kind of information will the target audience be looking for on your site? Are they looking for specific information, a particular product or service, online ordering, etc.?

Scanned with OKEN Scanner

Phase Two: Planning

Using the information gathered from Phase one, it is time to put together a plan for your website. This is the point where a site map is developed.

The site map is a list of all main topic areas of the site, as well as sub-topics, if applicable. This serves as a guide as to what content will be on the site, and is essential to developing a consistent, easy to understand navigational system. A good user interface creates an easy way to navigate website, and is the basis for this.

During the planning phase, your web designer will also help you decide what technologies should be implemented. Elements such as what CMS (Content Management System), such as WordPress.

Phase Three: Design

Drawing from the information gathered up to this point, it is time to determine the look and feel of your site. As part of the design phase, it is also important to incorporate elements, such as the company logo or colors to help strengthen the identity of your company on the website.

Your web designer will create one or more prototype designs for your website. This is typically a .jpg image of what the final design will look like. Either way, your designer should allow you to view your project throughout the design and development stages. The most important reason for this is that it gives you the opportunity to express your likes and dislikes on the site design.

In this phase, communication between both you and your designer is crucial to ensure that the final website will match your needs and requirement. It is important that you work closely with your designer, exchanging ideas, until you arrive at the final design for your website.

Phase Four: Development

The developmental stage is the point where the website itself is created. At this time, your web designer will take all of the individual graphic elements from the prototype and use them to create the actual, functional site.

This is typically done by first developing the home page, followed by a "shell" for the interior pages. The shell serves as a template for the content pages of your site, as it contains the main navigational structure for the website. Once the shell has been created, your designer will take your content and distribute it throughout the site, in the appropriate areas.

Elements, such as CMS like, interactive contact forms, or e-commerce shopping carts are implemented and made functional during this phase, as well. This entire time, your designer should continue to make your in-progress website available to you for viewing, so that you can suggest any additional changes or corrections you would like to have done.

Phase Five: Testing and Delivery

Your web designer will attend to the final details and test your website. They will test things, such as the complete functionality of forms or other scripts, as well last testing for last minute compatibility issues (viewing differences between different web browsers), ensuring that your website is optimized to be viewed properly in the most recent browser versions.

A good web designer is one who is well-versed in current standards for website design and development. The basic technologies currently used are HTML and **CSS (Cascading Style Sheets)**. As part of testing, your designer should check to be sure that all of the code written for your website validates. Valid code means that your site meets the current web development standards – this is helpful when checking for issues, such as cross-browser compatibility.

Once you give your web designer final approval, it is time to deliver the site. An **FTP (File Transfer Protocol)** program is used to upload the website files to your server. Other final details include plug-in installation (for WordPress or other CMS-driven websites and SEO (**Search Engine Optimization**). SEO is

Scanned with OKEN Scanner

the optimization of your website with elements, such as title, description and keyword tags which help your Website achieve higher rankings in the search engines. The previously mentioned code validation is something that plays a vital role in SEO, as well. There are many WordPress plug-ins available that further enhance the default WordPress functionality – many of them are directly related to improving your SEO, as well.

Phase Six: Maintenance: Many designers offer maintenance packages at reduced rates, based on how often you anticipate making changes or additions to your website. If you prefer to be more update your own content, there is something called a CMS, such as can be implemented to your website. This is something that would be decided upon during the planning stage.

A website determined by a CMS gives you the ability to edit the content areas of the website. You are given access to a back-end administrative area, where you can use an online text editor. You will be able to edit existing content and you can even add new pages and content yourself. Other maintenance type items include regular site backups, WordPress upgrades, additional plug-in installation, etc.

7.8.4 How to Develop a Website?

The design and development of a website should be a thorough going and planned process. One of the most effective ways of doing this is to generate a website outline. The construction and maintenance of a website outline is imperative for the proper design and development of an efficient website. In this way, you can keep track of the physical data that comprises your site. In other words, a website outline helps you keep control of the growing database of files that accumulate as you develop your site.

1. **Organize your files:** Keep your website files organized on your hard drive just as you are presenting it on the internet. This simple step will save you hours of wasted effort as your website takes shape.
2. **Create a root directory:** Keep files that link directly to your index page and files that are used repeatedly in your root directory. But do not overload this directory. Make use of sub folders, of content, that link to these main pages. Place only prime pages in your root directory.
3. **Develop subdirectories:** Create a subdirectory of related pages or website sections. Keep all relevant information together in individual sub folders. This will help you and the search engines follow threads of thought and provide quick access to information for your visitors.
4. **Plan for the expansion of your Website:** If you think a page on a particular theme will expand into a number of pages, create a subdirectory for it during the development phase.
5. **Organize your directory links:** Keep files that link to individual pages, of your root directory, in unique subdirectories. Do not put files from your root directory into subdirectories, keep them separate and link them to folders that contain related content.
6. **Use simple file names:** Name your subdirectories and files in an easy to identify manner. Make good use of your keywords when naming sub folders and files. This will unite the content of your website. All of your website's file names should tie together in a logical way.
7. **Develop and maintain permanent file addresses:** Do not change your subdirectory or file names, or HTML page addresses. Name and address are indexed and arranged in a manner that these can easily be maintained in the future. This will help to prevent broken links as your website grows.
8. If you do have to change an address after your website, then place a redirect page or link in the old location. This will help your visitors find the information they are seeking with a minimum of effort and direct new visitors and search engines to the modified address.

Some Important Points

You should remember that you have about fifty seconds to grab the attention of your website viewers.

If you follow the under-mentioned tips, you will be able to grab the attention of your viewers or you may miss the opportunity.

Scanned with OKEN Scanner

1. Make the title of the website catchy, descriptive and accurate.
2. Provide clues at the top of the home page about what the page contains. Do not force your viewer to scroll it down to find the clues.
3. If your web page is more than three "screen fulls", break it into two pages.
4. Maintain a sense of balance. Do not let the images be small and large, widely varying in shapes.
5. Use text and link colors that complement and do not clash.
6. If you have used a shape, like a button, it must act like a button to open and close.
7. Make links descriptive but not like "Click Here!"
8. Keep file names short and consistent.
9. Create a link to the e-mail address of the web Master.
10. Tell your viewer about the size of any down loadable file you include.

7.9 Search Engines

Since the Internet is a vast collection of information, it is tedious to find specific information you actually need. Therefore, the search feature in a web browser, such as the Internet explorer provides an easy access to a special facility called **search engine**. Search engines scan the Internet for the words or topics you are looking for. A search engine is a program that searches through a database of web pages for particular information.

Find Information using a Search Engine

There are many search engines available on the web. The searching process is the same for all these search engines. Now you will describe the information searching process in Yahoo!.

1. Go to the Yahoo! Home page by logging on to **www.yahoo.com** as shown in Figure 7.6.



Figure 7.6: Yahoo! Home page and Query typed in the box

2. Type, the information to be searched for in the box provided for it and then click the search web button as shown in Figure 7.6
3. Within a few seconds, the search engine will display the available links to the web page as shown in Figure 7.7.

7.10 Meta-Search Engines

Meta-search software sits on your computer and enables you to search through many internet search engines simultaneously and to view and use the results. You can browse through the results on your computer and click those pages that you want to visit.

Scanned with OKEN Scanner

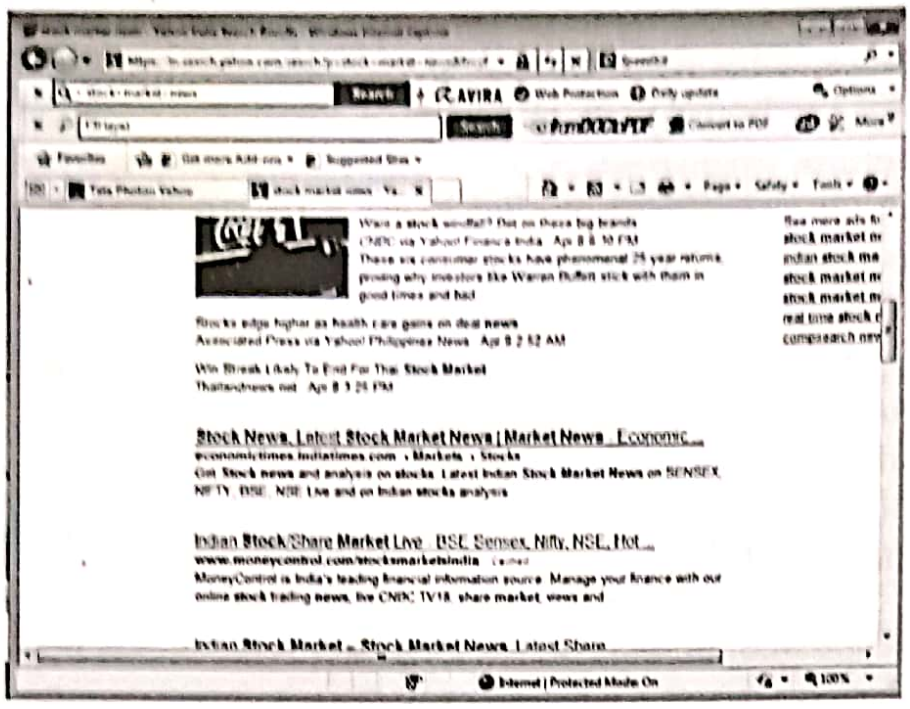


Figure 7.7: Result of Query

Meta-search engines do not create their own databases but make use of the databases of other search engines. They quickly process the user's search by invoking several individual search engines at once and return the results compiled into a convenient format.

7.11 World Wide Web(WWW)

The World Wide Web (WWW) is a network of online content that is formatted in HTML and accessed via HTTP. The term refers to all the interlinked HTML pages that can be accessed over the internet. The World Wide Web was originally designed in **1991** by **Tim Berners Lee** while he was a contractor at CERN.

The World Wide Web is most often referred to as the web. The World Wide Web is what most people think of as the internet. It is all the web pages, pictures, videos and other online content that can be accessed via a web browser. The Internet is the underlying network connection that allows us to send e-mail and access the World Wide Web. The web was a collection of text-based sites hosted by organizations that were technically set up a web server. It has continued to evolve since the original design, and it now includes interactive (social) media and user-generated content that requires little but no technical skills.

7.12 Web Browser

A browser or web browser is a program you use to view pages on net and navigate the World Wide Web. Browsers are sometimes referred to as web clients. A wide range of web browsers are available for every type of systems you can imagine, including graphical-user-interface, and text-only for dial-up UNIX connections. Most browsers are freeware. All you have to do is to get a browser downloaded from the internet.

7.13 HTTP

The **Hypertext Transfer Protocol (HTTP)** is a set of rules for exchanging files (text, graphics, images, sound, video and other multimedia files) on the WWW. It is an access method to access web pages used on the internet. The HTTP generally works in combination with WWW.

Your web browser is an HTTP client, sending requests to server machines. When the user enters a file request in the browser program by either opening a web file (typing in a URL) or clicking on a hypertext link, the browser builds an HTTP request and sends it to an HTTP program in the destination server machine, which receives the request and after any necessary processing returns the requested file.

Scanned with OKEN Scanner

7.14 Publishing Tools

There are a number of web publishing tools developed by various corporations. Some of the more popular ones are:

- Netscape Navigator Gold
- Macromedia's Backstage Desktop Studio
- Adobe's PageMill and SiteMill
- Other Web Publishing tools, such as HotDog, HoTMetal and WebEdit.

Netscape Navigator Gold: Netscape Gold offers a visual editing environment that closely approximates that of a web page and at the same time adopts many of the concepts used by word processors and document publishing tools.

Netscape Gold provides a web publishing environment that looks and feels like a web page.

Netscape Gold generates true HTML code, but during the editing phase, this code is hidden from the user by the graphical interface. When creating complex web pages, you will need to understand the principles behind the HTML language and its various constructs. Using Templates and the Page Wizard, Netscape has reduced the amount of work you need to do when designing a web page.

Macromedia's Backstage Desktop Studio: Backstage Desktop Studio consists of a suite of tools to assemble and manage an end-to-end website. The advantage of Macromedia is its excellence in multimedia web authoring. By combining that experience in imaginary with the simplicity of a point-and-click, drag-and-drop, **WYSIWYG (What You See Is What You Get)** strategy for the implementation tools, Macromedia has one of the best tools for designing web pages. Macromedia's Backstage Desktop Studio is one of the new generation of tools that combine dynamic web page authoring facilities with website management features.

Adobe PageMill and SiteMill: PageMill and SiteMill from Adobe feature a WYSIWYG interface, which makes operating them easy. At the same time, it incorporates typical Macintosh ease of use.

SiteMill: SiteMill is the perfect choice for both managing and editing web pages on the server. It catalogues a website, checks each link, and displays all errors. Currently, both programs support HTML as well as Netscape plug-ins (a software component that adds a specific feature to an existing software application).

Using PageMill: PageMill is an excellent choice for creating web pages. It allows even a novice to create a web page complete with forms, graphics and formatting. PageMill can be operated in two separate modes: **Browse** and **Edit**. The **Browse mode** is the default mode when you are opening a saved page. It shows a view of the page as though it were being served over the internet to your browser. The Edit mode is for creating and editing your page. You can toggle between the two modes by simply clicking the Browse/Edit icon at the screen's upper right corner. PageMill lets you easily create web pages that have forms, backgrounds, internal links, external links, images, image maps, and detailed text formatting.

You can accomplish all of this without any knowledge of HTML. PageMill supports normal cut-and-paste operation. If the dragged or copied text is (**Rich Text Format**) RTF-formatted, the attributes and styles are preserved; otherwise, you have to reformat the text.

Other Web Publishing Tools

There is a wide variety of web publishing tools. Many of these tools offer value for low price. These are described in the following subsections:

GoLive: Adobe GoLive boasts drag-and-drop authoring features that will have you building your own site in no time. Although Adobe was one of the first big software developers to release a drag-and-drop web authoring package in the shape of PageMill, it did not offer a real contender until the release of GoLive. GoLive incorporates cutting-edge features.

Scanned with OKEN Scanner

HotDog and HotDog Pro: HotDog supports HTML, Netscape extensions to HTML, and internet explorer extensions to HTML. When you have multiple documents open in the editor, you can switch between them quickly using tabs at the bottom of the editor Window. HotDog has a friendly interface.

HotDog lets you preview your pages without having to save the page. Other features include an advanced correction system that automatically replaces text and file paths when you publish your pages. HotDog is designed for beginners and experts alike. You will know this immediately when you start HotDog. In addition to being able to insert many common tags into a page directly from the toolbar, HotDog has a dialog box for adding tags.

Creating forms with HotDog is easy just click the Forms icon on the toolbar and HotDog displays a form builder. The form builder provides a friendly interface for selecting the type of element you want to add to a form.

HotDog has other friendly tools for managing links, building tables and creating frames. With HotDog Pro, not only you can customize the toolbar, but also you can select the icons and labels for toolbar buttons. Another nice feature of HotDog Pro is the ability to define your own tags or tag sets that can be accessed via the menus.

HoTMetaL and HoTMetaL Pro: HoTMetaL is an editor of SoftQuad Software, the specialized (Standard Generalized Markup Language) SGML editor. There are two editions of HoTMetaL. HoTMetaL is available free! What is even better is that SoftQuad has made a serious commitment to continue to develop and improve this freeware editor.

HotMetaL Pro supports features that put it in the same league as **Microsoft FrontPage** and **Macromedia Backstage**. Still, **SoftQuad** has the market experience to back its offerings with solid support and continual upgrades. Versions of this editor are available for **Windows, Macintosh, Power Macintosh,** and **UNIX.**

HoTMetaL Pro is an advanced authoring system that supports the most current HTML specification and extensions. It supports HTML, Java, ActiveX, objects, scripts and even Shockwave. HoTMetaL Pro follows an integrated WYSIWYG and tag marker approach.

Dreamweaver: In the space of two short years, Macromedia Dreamweaver has become the standard web authoring tool for creative designers. Now in its latest version, the package is still in a position to take advantage of future developments on the web. In the industry, e-business and e-commerce have become buzzwords and Dreamweaver acknowledges this with Roundtrip coded data import. The program can be customized for specific projects so that team members and clients alike can use the same package in the development cycle. Dreamweaver also prepares us for Web development with a core built around **XML (eXtensible Markup Language)**, CSS (**Cascading Style Sheet**), the eXtensible Scripting Language (XSL) expected to go mainstream as browser support comes online. Dreamweaver lets you keep one eye on the future without forgetting what is important now.

Choosing the right software to build your own Website

You have taken the decision to build your own website, but the next step is which software should you use to build and design website?

XsitePro is one of the most popular pieces of Web building software available on the market. Launched in 2004 it quickly became a firm favourite, with critical acclaim from hundreds of delighted customers. XsitePro recently released an upgrade to the original software, **XsitePro Version 2.5.**

XsitePro Features

XsitePro has a wealth of features, with too many to list here. However, some of the best features are as follows:

Scanned with OKEN Scanner

a. **Simple and straight forward design interface:** The user has complete control over the design of the page so you can create a site which looks and functions just has you want it to.
b. **Hundreds of website templates available:** If you want to create a professional website in a hurry, then you can choose from a catalog of website templates, all of which have been professionally created by XsitePro designers.
c. **A well thought through, interactive layout:** You can create your website easily thanks to the simple and easy to follow layout.
d. **Create unlimited Websites:** There is no limit to the number of websites you can create with XsitePro.
e. **Scripts:** Scripts can be run on a number of pages. This allows you to track events on your pages and report factors, such as on the return, on investment, on your Web pages.
f. **No HTML coding required:** Perfect for the novice, XsitePro delivers a built-in WYSIWYG designer, meaning no coding is ever required.
g. **Audio and Video in a few simple clicks:** Set your site apart from your competitors by adding an audio or video feature (or both), which can be added with a few clicks of your mouse.
h. **Easy linking:** Websites become more powerful when they have relevant links to other sites. With XsitePro you can create multipage directories of your links, with categories, sub categories, automatically generated screen shots and more.
i. **Use host and domain name of your choice:** XsitePro will work with any hosting account and domain name.
j. **Ideal for sites of any size:** XsitePro can handle sites with small amounts of content, right up to sites with thousands of web pages.
k. **Windows compatibility:** XsitePro works well with all versions of Windows 2000, plus Windows XP and Windows Vista (note, it is not Mac compatible).

Adobe Dreamweaver

The newest version of Dreamweaver is **Adobe Dreamweaver CS5**, and there are four different solutions to choose from so you can buy the package that best meets your needs.

Features of Dreamweaver

Live View: You can design your web pages under real world browser conditions with this new function. Simple code editing is required, and the effects of changes can be seen in real time.

3D Transformation Tools: Bring your websites to life using the Flash Professional software.

Faster design tools: The Photoshop extended software allows you to exercise greater precision with images whilst making changes quickly.

Compatible with Windows and Mac operating systems, unlike XsitePro, which is only Windows compatible.

Microsoft Expression Web (formerly FrontPage)

Microsoft Expression Web, formerly **Microsoft FrontPage**, is another leading and well-respected software package which allows the user to create professional websites quickly and effectively. The most current version is **Expression Web 3**, and as with XsitePro and Dreamweaver, Microsoft claims that Expression Web 3 is suitable for all types of user. However, in reality Expression Web 3 is aimed at professional Web designers and developers and the user; must have an understanding of coding in order to have any degree of success.

Microsoft Expression Web 3 is packed full of useful features, such as:

Scanned with OKEN Scanner

Professional Code Editor: This allows the user to debug effectively, as coding is color coded for HTML, JavaScript, CSS and PHP **(Hypertext Preprocessor)**.

Professional Design Surface: Users can create an impressive design layout with a design surface which allows for easy manipulation of images, copy positioning, sizing and margins.

See effects of changes in real time: Any changes can be viewed in a real time web browser, which can be viewed alongside the screen you are making the changes in.

Ideal for building standards-based Websites: With Expression Web 3, professional developers and designers can easily take advantage of the best of dynamic web technologies to quickly and efficiently design, develop, and maintain standards-based websites with seamless integration with ASP.NET 3.5 (server-side Web application framework), ASP.NET AJAX (a set of extensions to ASP.NET developed by Microsoft for implementing Ajax) and Visual Studio 2008 (used to develop Websites, Web applications and Web services).

Drag-and-Drop Controls: These controls simplify the process of adding rich media. Combined with advanced Photoshop the user has complete control over imported file layers.

Conclusion

This chapter discussed with web Publishing and browsing. It is the process of publishing content on the internet. It includes creating and uploading websites, updating the associated web pages, and posting content to these web pages online. The next topics, beings with **Standard Generalized Markup Language** (SGML, is a language for defining markup languages. Next, we have discussed Web hosting basics. It is a service that allows organizations and individuals to post a website or web page onto the Internet. Then, we have documents interchange standards and components of Web Publishing. Next, we have discussed document management, web page design considerations and principles. Finally, we have discussed search and meta search engines, WWW, browser, HTTP and publishing tools.

REVIEW QUESTIONS WITH ANSWERS

1. Multiple choice questions

1.1 You should consider the following point seriously while designing a Web page:

a. Identifying banners, logos, the desktop or client computer or similar devices
b. Navigational elements, such as Nav bar, buttons or any text links to other pages
c. Text of all types
d. All of the above

1.2 The factors you can take into consideration to improve the look and feel of the website are:

a. Space and balance
b. Colour
c. Consistency
d. All of the above

1.3 In order to establish a site, the following should be considered:

a. Provide new information regularly
b. Let your visitors register for e-mail notification of changes to the site
c. Update existing information
d. All of the above

1.4 Host differs from desktop computers in that it can handle:

Scanned with OKEN Scanner

a. Multiple connections
b. Single connection
c. No connection
d. All of the above

1.5 Acronym for SGML is

a. Standalone Generalized Markup Language
b. Standard Generalized Markup Language
c. Standard Generalized Markup Language
d. Standalone Generalized Markup Language

1.6 Netscape Navigator and other browsers, such as the following, are available free on the Internet:

a. Internet Explorer
b. HotDog
c. Adobe PageMaker
d. None of the above

1.7 Which of the following statements is false

a. You can make a website without using HTML
b. You can make a website without using PHP
c. You can make a website without using CSS
d. You can make a website without using Javascript

1.8 Which of the following is true

a. You need a server to host your website files
b. You don't need a server to host your website files
c. You can create a website without using HTML
d. You can't create a website without a CMS

1.9 Which of the following statements is false about hosting?

a. Shared hosting is cheaper than dedicated hosting
b. Shared hosting is safer than dedicated hosting
c. Dedicated hosting is safer than shared hosting
d. Though more expensive than shared hosting, Dedicated hosting is more secure

1.10 What is a cookie?

a. Computer code used to show animations and play sound
b. Information sent to your computer that is used later to give information back to visited website.
c. Computer code that automatically records all purchases and payments you make from your computer
d. An advertisement that will at some time be displayed on your computer screen

2. State whether the following statements are True or False

2.1 The Web pages of your website cannot be connected by hyperlinks.

2.2 You should include a lot of fancy graphics and sound clippings on your website so that it takes a lot of time for the visitor to access it.

Scanned with OKEN Scanner

2.3 Your ISP does not play any role when you want to transfer files to a Web server.

2.4 UNIX has Daemon program that is usually invisible to the user, providing important system services.

2.5 A method to decrease the download time is to use a cable modem or take ISDN connection.

2.6 The power of hypertext is its simplicity and transparency.

2.7 In HTTP, by identifying the type of object in a header field, the client receiving the object can handle it appropriately.

2.8 In website designing, you should make the title catchy, descriptive and accurate.

2.9 Netscape messenger is the e-mail client packaged with Netscape Communicator.

2.10 Software that may be used to help you locate web pages on topics that you do not know the addresses for is called search engines.

3. Match the following

3.1	It is stateless protocol:	a.	Web designer
3.2	Adobe SiteMill:	b.	Meta-search engine
3.3	The tools search information on the internet:	c.	TLS/SSL certificate
3.4	Example of meta-search engine:	d.	FTP
3.5	The action of identifying, acquiring, processing, storing, updating and distributing the document is called:	e.	Dedicated servers
3.6	Where a server shares many websites all placed at one place having the range from few to hundred:	f.	Stateful
3.7	HTTPS is a secure protocol which uses:	g.	HTTP
3.8	The files are uploaded using:	h.	Document management
3.9	Less costly and they allow better services than the shared ones due to the reliability:	i.	Dogpile
3.10	Protocol require server to save the status and session information:	j.	Shared web hosting service
		k.	TCP protocol network

4. Fill in the blanks

a.	Web browser	b.	Website	c.	Server
d.	Web search engine	e.	Different platforms	f.	Plug-ins
g.	Web server	h.	Web space	i.	Web host
j.	HotDog	k.	Web publishing		

4.1 The data storage space accessed via the internet, usually used to host websites and data files is called_____.

Scanned with OKEN Scanner

4.2 The software which helps to view the websites is called ----------.

4.3 A ---------- is a computer that performs actions for another computer.

4.4 A program that runs on a web server to process forms is called _____.

4.5 A _____ is a web-based tool that enables users to locate information on the World Wide Web.

4.6 _____ is an HTML editor developed by Sausage Software in the mid-1990s.

4.7 Web-applications were developed in order to interact _____.

4.8 _____ programs are automatically loaded and operates as a part of browser.

4.9 Computer which stores the different web pages is called as _____.

4.10 _____ is the process of publishing content on the Internet.

Descriptive Questions with Answer

5.(a) What are the features provided in web hosting?

Answer

1. It provides either the single page hosting or there are other hosting methods that can be used for the premium sites.
2. The web hosting provides database support and provides the use of the on-growing language.
3. It provides the platform for application development that can use the languages from PHP to Ruby, etc.
4. The web hosting platform allows the users to write and install the scripts that can allow them to use a particular feature of that platform.
5. Layers such as SSL (Secure Sockets Layer) or TLS (Transport layer security) used to provide more options of security for the websites.

5.(b) What is Web Hosting?

Answer: Web hosting is the act of renting space and bandwidth through a company so that we may publish our website online.

5.(c) What are the different types of hosting?

Answer: There are different types of hosting provided to upload and share their files using the web.

a. Free web hosting service: It provides limited services and offers it to advertisements and other services with limitations.
b. Shared web hosting service: where a server shares many websites all placed at one place having the range. The domains are shared with a common resource that is based on the server like RAM and CPU.
c. Reseller web hosting: This is the web hosting that allows clients to provide web hosting to others.
d. Virtual Dedicated Server: This is also known as also known as a Virtual Private Server (VPS). This provided the divided server resources in the virtual servers and the method that doesn't relate directly to the hardware.
e. Home server: This is a single machine server that can be thought as a personal server that is used to host one or more websites using the connection.

6.(a) Difference between HTTP and HTML.

Answer

HTML stands for HyperText Markup Language. It is a well-known markup language used to develop web pages. It has been around for a long time and is commonly used in webpage design. HTTP, on the other hand, stands for Hypertext Transfer Protocol. It is a means of data communication for the World Wide Web. It is an application protocol for distributed, collaborative, hypermedia information systems. HTTP is a

Scanned with OKEN Scanner

TCP/IP-based protocol used for transferring webpages from servers to clients. HyperText Markup Language - the formatting language used to design webpages. HTTP just is a protocol use to transfer data on internet. HTML is a language to create a web page. HTTP is the code in a web address URL that specifies that you are looking for a common website. Basically this protocol establishes the standard that computers on the internet use to communicate and transfer data through web browsers.

HTML is the language of web pages. A Word document contains text and formatting code for example. The text code is the content and the formatting code tells word how to display the text on the screen / page. HTML is very similar. It contains the content (which is easily edited). It's the formatting code that determines how a web page's content will look in a browser.

HTTP is the protocol that is used to request and deliver web page content (as files). When your browser requests a web page from a web server, it uses HTTP. But the owner of the web page uses FTP protocol to actually put the files on the web server (using the same address), or delete them from the server because there will be new files, or uploading new files. But both protocols use the same link (the internet).

6.(b) Difference between stateless protocol and stateful protocol

Answers: **Stateless protocol:** Stateless Protocol is a network protocol in which client send request to the server and server response back as per the given state. Examples of stateless protocol are HTTP (Hypertext Transfer Protocol), UDP (User Datagram Protocol) & DNS (Domain Name System). In Stateless, server restriction is not needed to keep the server information. In stateless, server and client are loosely coupled and can act independently. Server design is simple to implement. Server handles transactions in a very quicker way.

Stateful protocol: Stateful Protocol is a network protocol in which if client send a request to the server then it expects some kind of response, in case of no response then it resend the request. Examples of stateful protocol is FTP and Telnet. In stateful, a server is required to maintain the current state and session information. Server design is comparatively complex and difficult to implement. Server is slow comparatively to stateless.

6.(c) What are the advantages and disadvantages of connectionless protocol?

Answer

Advantages of connectionless protocol are as follows:

a. It sends the packet without handshaking.
b. It is faster than connection-oriented protocol.
c. The header size of the packet is smaller as compared to the packets in connection-oriented services.

Disadvantages of connectionless protocol are as follows:

a. It is not reliable and cannot ensure the data transmission to the destination.
b. Packets decide the route while transmission based on the network congestion.
c. It does not have a fixed path.
d. Different packets do not necessarily follow the same path.

7.(a) Where is the dedicated hosting service used?

Answer

Dedicated hosting services allow the users to have their own space or their own web server.

a. It can be handled by the user having the full control over the operating system and the resources it uses.
b. It does not allow the user to own the server but it is just another type of self-managed server.
c. They are less costly and they allow better services than the shared ones due to the reliability.

Scanned with OKEN Scanner

d. It requires lots of responsibility and maintenance to manage the resources and to provide the services to the dedicated servers.

e. It provides higher security as it is better than the other servers like shared servers that compromise on security as it is shared between many.

7.(b) What is the scope of web hosting services?

Answer

Web hosting scope is higher due to the increase of the traffic when the web page of the website is accessed. The files are uploaded using the FTP (File Transfer Protocol) using the web hosting and its services. Web interface is being used for the deployment of the website using the web hosting services and techniques. ISPs (Internet Service Providers) provide the services to the clients that are from the organizations or on the individual basis and provide the hosting solutions. It provides personal website hosting free to deploy and use. It also allows the accessing to be performed of the websites that is having the higher expense.

8.(a) Explain web Publishing. Write about any publishing tool.

Answer: Web publishing tools are applications that are used to design and build websites for either the internet or an intranet. They range from simple text editors that can generate web application code through to highly sophisticated and feature-rich web authoring packages. Some web publishing tools must be installed on a workstation. Others are server-based and are accessed via a web-interface. Web publishing tools are also called web authoring tools.

This website provides an overview of current web publishing application software. It is intended for individuals and small business owners that wish to establish a new website or want to upgrade an existing site to one that is more attractive, functional and maintainable.

We will need the following software before we begin creating our web pages.

FrontPage 2003 from Microsoft is one of the best web publishing tools on the market today. This sophisticated web authoring application provides an intuitive user interface. Although this web publishing tool is suitable for new web designers, it also provides advanced features that only professional web publishers would want to use. Because it is so simple to switch between the WYSIWYG (What You See Is What You Get) user interface and the "HTML Code" interface, one can easily view the HTML Code as pages are created. FrontPage also provides functionality to insert user-defined code into a page. This code is protected when changes are subsequently made using the WYSIWYG user interface.

It has numerous features that enable software objects to be reused, thereby saving much development and maintenance effort. These features include support for shared borders, website "themes", "included" pages and system generated navigation bars and titles.

It also provides XML capabilities that allow websites to interact with other websites over the internet. In addition to integrating with MS Access to provide data-driven functionality, FrontPage 2003 websites can also interface with SQL Server and XML databases and other databases using ODBC.

8.(b) Difference between connection oriented and connectionless protocol

Answer: With a connectionless protocol, clients connect to the server, make a request, get a response, and then disconnect. For the Internet, HTTP is a connectionless protocol.

With a connection-oriented protocol, clients connect to the server, make a request, get a response, and then maintain the connection to handle future requests. An example of a connection-oriented protocol is File Transfer Protocol (FTP). When we connect to an FTP server, the connection remains open after you download a file. The maintenance of this connection consumes system resources. A server with too many open connections quickly gets overloaded. Consequently, many FTP servers are configured to allow only 250 open connections at one time, so only 250 users can access the FTP server at once.

Scanned with OKEN Scanner

Connectionless protocols differ from connection-oriented protocols in the way requests and responses to requests are handled. HTTP is a connectionless protocol. When clients connect to the HTTP server being a connectionless protocol, they make a request, get a response, and then disconnect. Because the connection is not maintained, no system resources are used after the transaction is finished. Consequently, HTTP servers are limited only by active connections and can generally handle thousands of transactions with low system overhead.

The drawback to connectionless protocols is that when the same client requests more data, the connection must be re-established. To web users, this means a delay whenever they request more information.

9. (a) What is a cookie? How are they used on Internet?

Answer: A cookie is information that a web site puts on our hard disk so that is can remember something about us at a later time. Typically, a cookie records your preferences when using a particular site. Using the web's preferences when using a particular site. Using the Web's Hypertext Transfer Protocol (HTTP), each request for a web page is independent of all other requests. For this reasons, the web page server has no memory of what pages it has send to a user previously or anything about our previous visits. A cookie is a mechanism that allows the server to store its own information about a user on the user's own computer. We can view the cookies that have been stored on our hard disk. The location of the cookies depends on the browser. Internet explorer stores each cookie as a separate file under a Windows subdirectory.

Cookies have been of concern for internet privacy, since they can be used for tracking the browsing of a user. HTTP cookies are used for user authentication, user tracking, and maintaining user-specific information (preferences, electronic shopping cart, ect). Cookies are also used to track users across a website. Third-party cookies also allows for tracking across multiple sites.

(b) What is CGI? Explain.

Answer: The Common Gateway Interface (CGI) is a standard way for a web server to pass a web user's request to an application program and to receive data back to forward to the user. When the user requests a web page (for example, by clicking on a highlighted word or entering a website address), the server sends back the requested page. However, when a user fills out a form on a web page and sends it in, it usually needs to be processed by an application program. The web server typically passes the form information to a small application program that processes the data and may send back a confirmation message. This method or convention for passing data back and forth between the server and the application is called the Common Gateway Interface (CGI). It is part of the web's Hypertext Transfer Protocol (HTTP).

If we are creating a website and want a CGI application to get control, you specify the name of the application in the Uniform Resource Locator (URL) that we code in an HTML file.

A programmer can write a CGI application in a number of different languages can write such as C, C++, Java, and PERL.

(c) What is web hosting and web publishing?

Answers

Creating a website and placing it on the web server is called web publishing. A website is a collection of HTML pages with the home page typically named INDEX.HTML. websites are designed using web authoring software which provides a graphical layout capability or by hand coding in HTML or both. It includes creating and uploading websites, updating webpages, and posting blogs online. The published contents may include text, images, videos, and other types of media.

Web hosting is the service that makes our website available to be viewed by others on the internet. A web host provides space on its server, so that other computers around the world can access our website by means of a network or modem. The users want to view our website, all they need to do is type our website

Scanned with OKEN Scanner

address into their browser. Their computer will then connect to our server and our webpages will be delivered to them through the browser.

Most web hosting companies require that we own our domain name in order to host with them. If you do not have a domain name, the web hosting companies will help us purchase one.

The basic features of a web hosting plan include:

- Disk space
- Bandwidth or Data transfer
- E-mail accounts
- FTP access

(d) What is WWW? Explain briefly with the help of WWW architecture.

Answers

The World Wide Web is the part of the internet that contains websites and webpages. It was invented in 1989 by Tim Berners-Lee at CERN, Geneva, Switzerland. It is basically a system of internet servers that support specially formatted documents. The documents are formatted in a markup language called HTML (HyperText Markup Language) that supports links to other documents as well as graphics, audio and video files.

Architecture of WWW

WWW is basically a distributed client-server service. It this, a client can access the services from a server using a browser. These services are usually distributed over many locations called sites or websites. From the user's point of view web consists of a vast worldwide collection of documents called web pages. Each web page can contain link to other pages anywhere in the world. By clicking on such link user can access another web page. Such a text or image that enables the user to link to another web page is the string of text that points to another web page is called hypertext. Hypermedia is enhanced form of a hyperlink which not only links to the other pages or other sections within the same page but can also link with various medium like sound, animation, movie clip etc. Hypermedia is grouping of different media like sound, graphics, animations and text in a single file. These hyperlinks are created with the help of specialized language called Hypertext Mark up Language (HTML). In order to access these web pages on different sites, each of these pages has a specific address called Uniform Resource Locator (URL).

Answers

1.1	a	1.2	a	1.3	d	1.4	a	1.5	c	1.6	a
1.7	a	1.8	a	1.9	b	1.10	b				
2.1	F	2.2	F	2.3	F	2.4	T	2.5	T	2.6	T
2.7	T	2.8	T	2.9	T	2.10	T				
3.1	g	3.2	a	3.3	b	3.4	i	3.5	h	3.6	j
3.7	c	3.8	d	3.9	e	3.10	f				
4.1	h	4.2	a	4.3	c	4.4	b	4.5	d	4.6	j
4.7	e	4.8	f	4.9	g	4.10	k				

Scanned with OKEN Scanner

Model Test Paper

1. Multiple choice questions

1.1 _____ tag is used to add columns to a table.

a. Definition list
b. Definition list term
c. Definition list description
d. None of the above

1.2 Which of these tags are all <table> tags?

a. <table><head><tfoot>
b. <table><tr><td>
c. <table><tr><tt>
d. <thead><body><tr>

1.3 Which of the following is legal HTML syntax?

a. <FRAMESET COLS= "50%, 50%">
b. <FRAMESET ROW= "50%, 50%">
c. <FRAMESET COLUMNS= "50, 50">
d. <FRAME COLS= "50%, 50%">

1.4 A _____ can be inserted in HTML document which can act as a container for all the input elements.

a. Text field
b. Text area
c. Form
d. Command button

1.5 How will you make all paragraph elements 'RED' in color?

a. p.all {color: red;}
b. p.all {color: #990000;}
c. all.p {color: #998877;}
d. p {color: red;}

1.6 Which CSS property you will use if you want to add some margin between a DIV's border and its inner text?

a. Spacing
b. Margin
c. Padding
d. Inner-margin

1.7 Which are the different value types in JavaScript?

a. boolean, integer, float, string, array, object and null
b. boolean, number, string, function, object and undefined
c. boolean, number, date, regexp, array and object
d. boolean, number, string, function, object, null and undefined

1.8 What is the type of `null`, according to the `typeof` operator?

a. "undefined"
b. "object"
c. "array"
d. "null"

1.9 Which of the following types does NOT exist in javascript?

a. number
b. Boolean
c. object
d. Integer

1.10 Which of the following is a way to add a new value to the end of an array?

a. arr[value] = length;
b. arr.length = value;

Scanned with OKEN Scanner

c. arr[arr.length] = value;
d. arr[arr.length()] = value;

2. State whether the following statements are True or False

2.1 Src attibute of <frame> provides the URL to be displayed in the frame.
2.2 <iframe> tag is used to create floating frames.
2.3 Target attribute is used to display link in another window.
2.4 ng-app directive initializes an angular js application.
2.5 Is Angular JS supports two-way binding?
2.6 Can we extend Angular JS?
2.7 The border property sets the width of an element's four borders. This property can have from one to four values.
2.8 The border-image-width CSS property defines the offset to use for dividing the border image in nine parts.
2.9 Cols is an attribute of the <frameset> tag.
2.10 No resize attribute will you use if you do not want frame windows to be resizable.

3. Match the following

3.1	Removing unwanted image areas	a.	brightness
3.2	You can make an image appear brighter or darker by adjusting its	b.	Client-side scripting
3.3	To make Web pages more dynamic and more responsive	c.	ng-controller
3.4	Attaches the controller of MVC to the view	d.	Increase the distance between cell and content
3.5	HTML is a subset of	e.	Cropping
3.6	Add background color to your web page	f.	.class
3.7	All normal webpages consists of	g.	<fieldse>
3.8	Increasing the cellpadding means	h.	SGML
3.9	Which selectors selects any tag with an id attribute set	i.	Head and body
3.10	For grouping form controls we can use	j.	<body bgcolor="#FF000">
		k.	<legend>

4. Fill in the blanks

a. typeof
b. HTTP
c. Align
d. layout
e. Common gateway interface
f. option
g. Selector
h. Server
i. box-sizing
j. border-collapse
k. border-spacing

4.1 Styles specify formatting and page _____.

Scanned with OKEN Scanner

4.2 Web pages run server programs through a server feature called the _____.

4.3 _____ attribute of <Table> specifies the horizontal alignment of row or column.

4.4 Web page documents are stored on a _____ computer with special software. server,

4.5 A Web browser uses _____ to request a Web page from a Web server.

4.6 The options of a select list are specified with the _____ tag.

4.7 The _____ operator returns a string that identifies the type of its operand.

4.8 The _____ is the link between the html document and the style. It specifies what elements are affected by the declaration.

4.9 _____ property defines whether table cell borders are connected or separate.

4.10 The _____ property allows us to include the padding and border in an element's total width and height.

Descriptive Questions with Answer

Q5. a. What is Website? What are different types of Websites?

b. What is a Responsive Website?

c. What is browser? Name any three browser.

Q6. a. What is the new Structural Elements in HTML5?

b. What is local storage in HTML5? How it is used?

c. What is the use of <fieldset>?

Q7. a. What is the use of providing web hosting services?

b. Explain the different popup boxes JavaScript provides and state the major difference between them.

c. Differentiate between<TH> and <Caption> tags.

Q8. a Why forms are used in Webpages>

b. Differentiate between Get & Post methods of Form Tag.

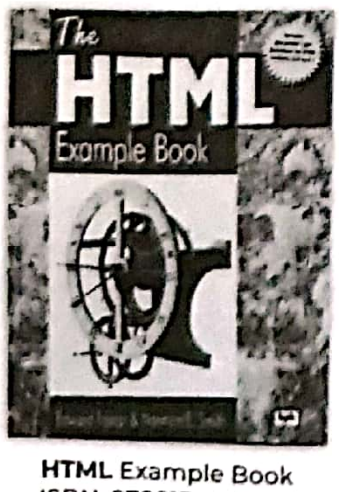
c. What are the different types of hosting?

Q9. a. What are the techniques to use W3.CSS Framework?

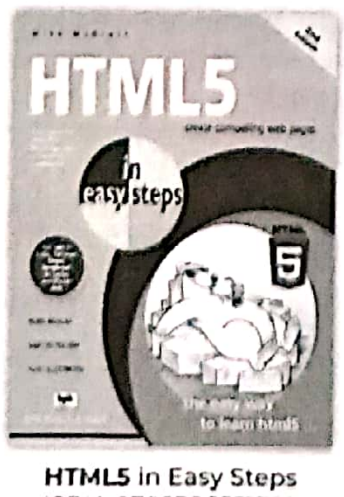
b. What is Angular JS? How does it work with HTML?

c. Write Html code to divide browser window into two horizontal sections. Divide bottom section into two vertical sections.

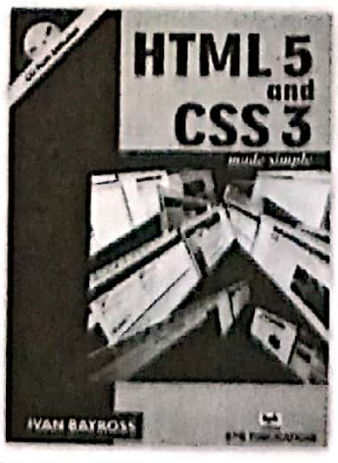
OTHER TITLES OF INTEREST



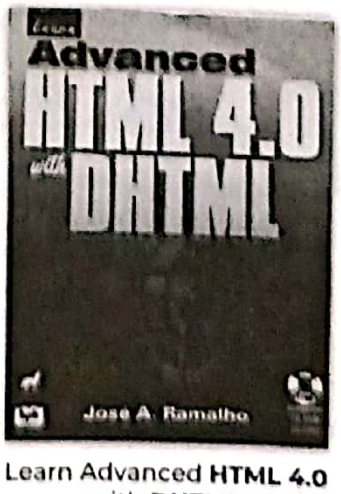
HTML Example Book
ISBN: 9788170299394



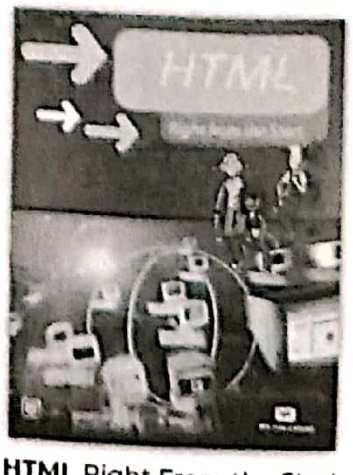
HTML5 in Easy Steps
ISBN: 9789386551382



HTML 5 & **CSS 3** made simple
ISBN: 9788183334419



Learn Advanced **HTML 4.0** with **DHTML**
ISBN: 9788176563062



HTML Right From the Start
ISBN: 9788176567633

Scanned with OKEN Scanner

Paper January 2021

भाग एक (PART ONE) सभी प्रश्नों के उत्तर दें (Answer all the Questions)

1. नीचे प्रत्येक प्रश्न के उत्तर के कई विकल्प दिए गए है। एक सबसे उपयुक्त विक्लप चुने और प्रश्न पत्र के साथ उपलब्ध कराए गए "ओएमआर" उत्तर पुस्तिका में दिए गए निर्देसज के अनुसार दर्ज करें। (Each question below gives a multiple choice of answers. Choose he most appropriate one and enter in the "OMR" answer sheet supplied with the question paper, following instruction therein. (1 x 10)

1.1 _____ टैग का उपयोग पैराग्राफ के आरभं को चिह्नित किया जाता है। (_____ tags is used to mark a beginning of paragraph.)

a. <TD> b.
 c. <p> d. <TR>

1.2 सबसे बड़ा शीर्षक के लिए सही एचटीएमएल टैग है: (Correct HTML tag for the largest heading is:)

a. <head> b. <h6>

c. <heading> d. <h1>

1.3 निम्नलिखित में से कौन सा एक कंटेनर है? (Which of the following is a container?)

a. <SELECT> b. <BODY>

c. <INPUT> d. (a) और (b) दोनों (Both (a) and (b))

1.4 वह विशेषता, जो वर्तमान दस्तावेज और HREF और URL के बीच संबंध को परिभाषित करती है: (The attribute, which define the relationship between current document and HREF and URL is:)

a. REL b. URL

c. REV d. उपरोक्त सभी (All of these)

1.5 <DT> टैग हमारे वेब पेज की एक पंक्ति को शामिल करने के लिए डिजाइन किया गया है लेकिन <DD> टैग एक _____ स्वीकार करेगा: (<DT> tag is designed to fit a single line of our web page but <DD> tag will accept a:)

a. टेक्स्ट की पंक्ति (line of text)

b. पूर्ण पैराग्राफ (full paragraph)

c. शब्द (word) d. अनुरोध (request)

1.6 करैक्टर एन्कोडिंग है: (Character encoding is:)

a. एक वर्ण में संख्याओं को दर्शाने के लिए प्रयोग की जाने वाली विधि (method used to represent numbers in a character)

b. एक संख्या में वर्ण का प्रतिनिधित्व करने के लिए प्रयोग की जाने वाली विधि (method used to represent character in a number)

c. एक प्रणाली जिसमें एक कोड होता है जो प्रत्येक वर्ण को एक पैटर्न या नेचुरल नंबर की सीक्वेंस या इलेक्ट्रिक पल्स डाटा ट्रांसमिट करने के लिए पेअर करता है (a system that consist of a code which pairs each character with a pattern, sequence of natural numbers or electrical pulse in order to transmit the data)

d. इनमे से कोई नहीं (None of these)

1.7 एच.टी.एम.एल. का पूर्ण रूप क्या है? (What is the full form of HTML?)

a. हाइफनेशन टेक्स्ट मार्कअप लैंग्वेज (Hyphenation text markup language)

b. हाइपर टेक्स्ट मार्कअप लैंग्वेज (Hyper text markup language)

c. हाइपर टेक्स्ट मार्किंग लैंग्वेज (Hyper text marking language)

d. हाइफनेशन टेक्स्ट मार्किंग लैंग्वेज (Hyphenation test marking language)

1.8 बेस URL को स्थापित करने के लिए एक बेहतर अप्रोच _____ उपयोग करना है: (A much better approach to establish the base URL is to use:)

a. बेस एलिमेंट (BASE element)

b. हेड एलिमेंट (HEAD element)

c. (a) और (b) दोनों (Both (a) and (b))

d. इनमे से कोई नहीं (None of these)

1.9 एक नई लिस्ट आइटम बनाने के लिए उपयोग किया जाने वाला टैग और इसमें एक हाइपरलिंक भी शामिल है: (The tag used to create a new list item and also include a hyperlink is:)

a. <LI> b. <DL> c. <DD> d. <UL>

1.10 ग्राफिक का कोई भी भाग जो दूसरे हॉट जोन में शामिल नहीं है, _____ का भाग माना जाता है: (Any part of the graphic that is not included in another hot zone is considered to be part of:)

a. rect b. point c. default d. polygon

2. नीचे दिया गया प्रत्येक विवरण या तो सही है अथवा गलत। एक सबसे उपयुक्त विक्लप चुने और उसे प्रश्न पत्र के साथ उपलब्ध कराए गए "ओएमआर" उतर पत्रिका में, उसमे दिए गए निर्देशों के अनुसार दर्ज करे। (Each statement below is either TRUE or False, Choose the most appropriate one and ENTER in the "OMR"

Scanned with OKEN Scanner

answer sheet supplied with the question paper, following instructions therein.) **(1 x 10)**

2.1 CSS का पूर्ण रूप कैस्केडिंग स्टाइल शीट्स है। (CSS stands for Cascading Style Sheets.)

2.2 CSS का उपयोग किसी भी प्रकार के XML दस्तावेज के साथ भी किया जा सकता है। (CSS can also be used with any kind of XML documents.)

2.3 एक CSS रूल सेट में एक सिलेक्टर और एक डिक्लेरेशन ब्लॉक होता है। (A CSS rule set contains a selector and a declaration block.)

2.4 CSS एलिमेंट सिलेक्टर में सी.इस.इस. के प्रकार हैं। (CSS Element Selector is types of selectors in CSS.)

2.5 यूनिवर्सल सिलेक्टर को वाइल्ड कार्ड करैक्टर के रूप में कभी नहीं किया जाता है। (The universal selector is never used as a wildcard character.)

2.6 विभिन्न शैली परिभाषाओं के साथ सभी तत्वों का चयन करने के लिए ग्रुपिंग सिलेक्टर का उपयोग किया जाता है। (The grouping selector is used to select all the elements with the different style definitions.)

2.7 आईडी सिलेक्टर एक विशिष्ट तत्व का चयन करने के लिए एक एच.टी.एम.एल. एलिमेंट की आईडी विशेषता का चयन करता है। (ID selector selects the id attribute of an HTML element to select a specific element.)

2.8 इनलाइन सीएसएस का उपयोग सीएसएस को कई लाइनों या एलिमेंट पर लागू करने के लिए किया जाता है। (Inline CSS is used to apply CSS on multiple lines or elements.)

2.9 इंटरनल स्टाइल शीट का उपयोग एक से अधिक दस्तावेजों के लिए कई स्टाइल को जोड़ने के लिए किया जाता है। (The internal style sheet is used to add multiple styles for more than one document.)

2.10 एक्सटर्नल स्टाइल शीट एक पूरी फाइल को बदलकर संपूर्ण वेब साइट के रूप को बदलने की सुविधा प्रदान करती है। (The external style sheet facilitates one to change the look of the entire web site by changing just one file.)

3. कॉलम X में दिए गए शब्दों और वाक्यांशों का मिलान निकटतम अर्थ/ शब्द/ फ्रांसेस कॉलम Y के साथ करें। अपना चयन, प्रश्नपत्र के साथ दी गई "ओएमआर" उत्तर पुस्तिका में दर्ज करें। (Match words and phrases in column X with the closest related meaning/word(s)/phrases in column Y. Enter your selection in the "OMR" answer sheet supplied with the question paper, following instructions therein.) **(1 x 10)**

कॉलम X (Column X)		कॉलम Y (Column Y)	
3.1	H1... H6	a.	.js extension
3.2	एक्सटर्नल जावास्क्रिप्ट फाइल (External JavaScript file)	b.	हेडिंग्स (Headings)
3.3	HTTP	c.	डबल फॉरवर्ड स्लेशेस (Double forward slashes)
3.4	जावास्क्रिप्ट सिंगल लाइन कमेंट (JavaScript Single line Comment)	d.	हाइपर टेक्स्ट ट्रांसफर प्रोटोकॉल (Hyper Text Transfer Protocol)
3.5	CSS	e.	कैस्केडिंग स्टाइल शीट (Cascading Style Sheet)
3.6	जावास्क्रिप्ट वेरिएबल (JavaScript variable)	f.	डायनामिक टाइप लैंग्वेज (Dynamic type language)
3.7	जावास्क्रिप्ट ग्लोबल वेरिएबल (JavaScript global variable)	g.	लोकल वेरिएबल एंड ग्लोबल वेरिएबल (Local Variable and Global Variable)
3.8	XML	h.	मल्टीप्ल एक्सप्रेशन से एक कोड निष्पषिद करना (To execute one code from multiple expressions)
3.9	जावास्क्रिप्ट (JavaScript)	i.	फंक्शन के बाहर घोषित करना (Declared outside the function)
3.10	जावास्क्रिप्ट स्विच स्टेटमेंट (JavaScript switch statement)	j.	एक्सटेंसिबल मार्कअप लैंग्वेज (Extensible Markup Language)
		k.	HTML
		l.	XML
		m.	डाटा डिपेंडेंसी (Data Dependency)

Scanned with OKEN Scanner

4. नीचे दिए गए सूची में किसी एक शब्द या वाक्यांश में से एक को भरने करने के लिए एक खाली स्थान है। निर्देश के अनुसार, प्रश्न पत्र से जुड़ी "ओएमआर" उत्तर पुस्तिका में अपनी पसंद दर्ज करें। **(Each statement below has a blank space to fit one of the word(s) or phrase(s) in the list below. Enter your choice in the "OMR" answer sheet attached to the question paper, following instruction therein.)** **(1 x 10)**

a. <PRE> फॉर्मेटेड टेक्स्ट (<PRE> formatted text)
b. <TABLE>
c. <HEAD>
d. इंटरफेसेस (Interfaces)
e. Out does type
f. <OL>
g. HTML एक्सटेंशन (HTML extension)
h. छः (six)
i. हैडर (Header)
j. <CAPTION>
k. पांच (Five)
l. <Body>
m. <Input>

4.1 _____ टैग एक संख्या/ क्रम लिस्ट बनाता है। (_____ Tag creates a number/order list.)

4.2 <TR>, <TD> और <TH> के लिए मुख्य कंटेनर _____ है। (Main container for <TR>, <TD> and <TH> is _____.)

4.3 <TEXT AREA> टैग के अंदर का टेक्स्ट _____ के रूप में कार्य करता है। (The text inside the <TEXT AREA> tag works like _____.)

4.4 <BASE> टैग केवल _____ के बीच प्रदर्शित होने के लिए डिजाइन किया गया है। (<BASE> tag is designed to appear only between _____.)

4.5 HTML में हमारे पास _____ शीर्षक लिखने के लिए हैडिंग टैग्स है। (In HTML we have _____ heading Tags for writing heading.)

4.6 ग्राफिक का कोई भी हिस्सा जो दूसरे गर्म क्षेत्र में शामिल नहीं है, उसे _____ का हिस्सा माना जाता है। (Any part of the graphic that is not included in another hot zone is considered to be part of _____.)

4.7 वह टैग जो आपको विवरण के भीतर अन्य एच.टी.एम.एल. टैग को रखने की अनुमति देता है, वह है _____। (The tag which allows you to rest other HTML tags within the description is _____.)

4.8 शब्द डॉक्यूमेंट नोड, नोडेलिस्ट, तत्व नोड आदि _____ हैं। (The word document Node, Nodelist, Element node etc. are _____.)

4.9 DTD के _____ के लिए टेम्पलेट बनाने के तरीके हैं। (DTD's are ways to create templates for _____.)

4.10 MIME टेक्स्ट फाइल को _____ के साथ सेव किया जाता है। (The MIME text file is saved with _____.)

भाग दो (PART TWO)

(किसी भी चार प्रश्नो के जवाब दें) (Answer any four questions)

5. a. एच टी एम एल क्या है? टैग से आपका क्या मतलब है? (What is HTML? What do you mean by tags?)
   b. एप्लेट्स क्या हैं? (What are applets?)
   c. XML दस्तावेज क्या है? XML का उपयोग विकास के लिए क्यों किया गया है? (What is well formed XML Document? Why XML has been used for development?) **(5 + 5 + 5)**

6. a. आप जावास्क्रिप्ट से आप क्या समझते हैं? जावास्क्रिप्ट के कुछ अनुप्रयोग दें। (What do you understand by JavaScript? Give some applications of JavaScript.)
   b. जावास्क्रिप्ट ऑब्जेक्ट से आप क्या समझते हैं? जावास्क्रिप्ट ऑब्जेक्ट बनाने के तरीके बताएं? (What do you understand by JavaScript object? Explain the ways of creating JavaScript object?)
   c. XML में नेस्टेड एलिमेंट क्या हैं? XQuery क्या है? (What are nested elements in XML? What is XQuery?) **(4 + 5 + 6)**

7. a. सी एस एस क्या है? सी एस एस के क्या फायदे हैं? (What is CSS. What are the advantages of CSS.)
   b. सी एस एस सिलेक्टर क्या है? किसी भी तीन प्रकार की व्याख्या करें। (What is a CSS Selector? Explain any three types.) **(7 + 8)**

8. a. एच टी एम एल फ्रेज टैग क्या हैं? स्पष्ट कीजिए। (What are HTML Phrase tags? Explain.)
   b. एच टी एम एल एलिमेंट को प्रभावित करने वाले 5 सीएसएस बैकग्राउंड प्रॉपर्टीज की व्याख्या करें। (Explain 5 CSS background properties that affect the HTML elements.) **(7 + 8)**

9. a. इनलाइन सीएसएस क्या है? उससे क्या असुविधाए होती हैं? (What is Inline CSS? What are the disadvantages?)
   b. मार्की क्या है? टेक्स्ट के एक खंड को अलग करने के लिए कितने टैग का उपयोग किया जा सकता है? स्पैन टैग का उपयोग क्या है? (What is a marquee? How many tags can be used to separate a section of texts? What is the use of a span tag?) **(8 + 7)**

Scanned with OKEN Scanner

Solved Sample Paper 1

भाग एक (PART ONE)

1. नीचे प्रत्येक प्रश्न के उत्तर के कई विकल्प दिए गए है। एक सबसे उपयुक्त विक्लप चुने और प्रश्न पत्र के साथ उपलब्ध कराए गए "ओएमआर" उत्तर पुस्तिका में दिए गए निर्देसज के अनुसार दर्ज करें। (Each question below gives a multiple choice of answers. Choose he most appropriate one and enter in the "OMR" answer sheet supplied with the question paper, following instruction therein. **(1 x 10)**

1.1 सीएसएस में "फॉन्ट–आकार" के रूप को कहा जा सकता है: (In CSS what does "font-size" can be called as:_

a. सिलेक्टर (Selector)

b. रूल (Rule)

c. प्रॉपर्टी (Property)

d. प्रॉपर्टी–नाम (Property-Name)

1.2 टेबल में रौ को जोड़ने के लिए निम्न में से किस टैग का उपयोग किया जाता है? (Which of the following tag is used to add rows in the table?)

a. <td> और </td> (<td> and </td>)

b. <th> और </th> (<th> and </th>)

c. <tr> और </tr> (<tr> and </tr>)

d. उपर्युक्त कोई नहीं (None of the above)

1.3 HTML में मेल भेजने के लिए सही तरीका कौन सा है? (Which of the following is the correct way to send mail in HTML?)

a. <a href = "mailto: xy@y">

b. <a href = "xy@y">

c. <mail xy@y</mail>

d. <href=mail "xy@Y">

1.4 दि गई छवि के चारों ओर टेक्स्ट का एक ब्लॉक ब्रैप करने के लिए, हम किस सीएसएस प्रॉपर्टी का उपयोग करेंगे? (If we want to wrap a block of text around an image, which CSS property will we use?)

a. ब्रैप (wrap) b. पुश (push)

c. फ्लोट (float) d. अलाइन (align)

1.5 HTML टैग क्या है जिसके तहत कोई जावास्क्रिप्ट कोड लिख सकता है? (What is the HTML tag under which one can write the JavaScript code?)

a. <JavaScript> b. <script>

c. <scripted> d. <Js>

1.6 जावास्क्रिप्ट निम्नलिखित उद्देश्य के लिए बनाया गया है: (JavaScript is designed for following purpose:)

a. HTML पेज को स्टाइल करने के लिए (To style HTML pages)

b. सर्वर पर DB से संबंधित क्वेरी निष्पादित करने के लिए (To execute Query related to DB on server)

c. HTML पृष्ठों में इंट्रक्टिवित्य जोड़ने के लिए (To add interactivity to HTML pages)

d. ऊपर के सभी (All of the above)

1.7 _____ कथन का उपयोग करके आप विशिष्ट कंडीशन के लिए परीक्षण करते हैं। (Using _____ statement is how you test for specific condition.)

a. सेलेक्ट (Select) b. If

c. स्विच (Switch) d. For

1.8 सीएसएस में, स्टाइलिंग एलिमेंट _____ द्वारा किया जा सकता है? (In CSS, styling elements can be done by?)

a. id b. क्लास (class)

c. tag d. ऊपर के सभी (All of the above)

1.9 HTML में अन–ओदेरेद या बुलेटेड लिस्ट के प्रकार क्या हैं? (What are the types of unordered or bulleted list in HTML?

a. डिस्क, स्क्वायर, ट्रायंगल (disc, square, triangle)

b. पोलीगोन, ट्रायंगल,सर्किल (polygon, triangle, circle)

c. डिस्क, सर्किल, स्क्वायर (disc, circle, square)

d. ऊपर के सभी (All of the above)

1.10 निम्नलिखित में से कौन सी सीएसएस प्रॉपर्टी बैकग्राउंड–पोजीशन प्रॉपर्टी के एक्स–कोआर्डिनेट को परिभाषित करती है? (Which of the following CSS property defines the x-coordinate of the background-position property?)

a. background-position-x

b. background-position-y

c. background-x-axis

d. background-y-axis

2. नीचे दिया गया प्रत्येक विवरण या तो सही है अथवा गलत। एक सबसे उपयुक्त विक्लप चुने और उसे प्रश्न पत्र के साथ उपलब्ध कराए गए "ओएमआर" उतर पत्रिका में, उसमे दिए गए निर्देशों के अनुसार दर्ज करे। (Each statement below is either TRUE or False, Choose the most appropriate one and ENTER in the "OMR" answer sheet supplied with the question paper, following instructions therein.) **(1 x 10)**

2.1 एक्सटर्नल जावास्क्रिप्ट फाइल में <script> टैग होना चाहिए। (The external JavaScript file must contain <script> tag.)

Scanned with OKEN Scanner

2.2 सीएसएस फाइल्स में एक्सटर्नल स्टाइलशीट संग्रहीत हैं। (External stylesheets are stored in CSS files.)

2.3 हम JS कोड को सीधे HTML के अंदर एम्बेड कर सकते हैं। (We can embed JS code inside HTML directly.)

2.4 सही स्थान निर्धारित करने के लिए शीर्ष, नीचे, बाएं और दाएं की स्थिति का उपयोग किया जाता है। (Positioning attributes top, bottom, left and right are used to determine the exact location.)

2.5 MAP टैग के भीतर, आप छवि के क्षेत्रों को निर्दिष्ट करने के लिए AREA टैग का उपयोग करते हैं जो हॉटस्पॉट के रूप में कार्य करेगा। (Within the MAP tag, you use the AREA tag to specify the areas of the image that will act as a hotspot.)

2.6 लिंक एलिमेंट को बॉडी सेक्शन के शीर्ष पर रखा जाना चाहिए। (The link element should be placed at the top of the body section.)

2.7 बॉक्स मॉडल केवल फ्लोटेड एलिमेंट पर लागू होता है। (The box model only applies to floated elements.)

2.8 स्विच के डिफॉल्ट स्टेटमेंट को हमेशा निष्पादित किया जाता है। (The default statement of a switch is always executed.)

2.9 एक एंकर के लिए एक लिंक बनाने के लिए, आप A टैग में HREF प्रॉपर्टी का उपयोग करते हैं। (To create a link to an anchor, you use the HREF property in A tag)

2.10 HTML में एक प्रोग्राम इंटरप्रेटर द्वारा प्रस्तुत और पढ़ा जा सकता है। (A program in HTML can be rendered and read by Interpreter.)

3. कॉलम X में दिए गए शब्दों और वाक्यांशों का मिलान निकटतम अर्थ/ शब्द/ फ्रांसेस कॉलम Y के साथ करें। अपना चयन, प्रश्नपत्र के साथ दी गई "ओएमआर" उत्तर पुस्तिका में दर्ज करें। (Match words and phrases in column X with the closest related meaning/word(s)/phrases in column Y. Enter your selection in the "OMR" answer sheet supplied with the question paper, following instructions therein.) **(1 x 10)**

कॉलम X (Column X)		कॉलम Y (Column Y)	
3.1	इसका उपयोग कार्ड के रूप में छवि बनाने के लिए किया जाता है। (It is used to create an image as a card.)	a.	एंगल (Angle)
3.2	HTML टैग्स का उपयोग दस्तावेज का वर्णन करने के लिए किया जाता है (HTML tags are used to describe document)	b.	सेल पैडिंग (Cell padding)
3.3	HTML में डिजाइन किए गए पेज को कहा जाता है: (Page designed in HTML is called as:)	c.	एक वेब पेज के भीतर एक वेब पेज प्रदर्शित करने के लिए। (To display a web page within a web page.)
3.4	HTML टैग को कोष्ठक से घिरा हुआ है (HTML tags are surrounded by brackets)	d.	हैडर (Header)
3.5	HTML टैग के ओपनिंग टैग को कहा जाता है (Opening tag of HTML tag is called as)	e.	रिक्वेस्ट –रिस्पांस प्रोटोकॉल (Request-Response Protocol)
3.6	एक एट्रिब्यूट प्रत्येक सेल के भीतर स्थान जोड़ती है। (An attribute add space within each cell.)	f.	W3-कार्ड (W3-card)
3.7	HTML में iframe का उपयोग (The use of iframe in HTML)	g.	रुट एलिमेंट (Root element)
3.8	टैग भीतर होना चाहिए (Tag must be within)	h.	वेब पेज (Web page)
3.9	HTTP	i.	कंटेंट (Content)
3.10	सभी XML दस्तावेज में बिल्कुल एक होना चाहिए (All XML document must contain exactly one)	j.	स्टार्टिंग टैग (Starting tag)
		k.	टाइटल (Title)

Scanned with OKEN Scanner

4. नीचे दिए गए सूची में किसी एक शब्द या वाक्यांश में से एक को भरने करने के लिए एक खाली स्थान है। निर्देश के अनुसार, प्रश्न पत्र से जुड़ी "ओएमआर" उत्तर पुस्तिका में अपनी पसंद दर्ज करें। (Each statement below has a blank space to fit one of the word(s) or phrase(s) in the list below. Enter your choice in the "OMR" answer sheet attached to the question paper, following instruction therein.) (1 x 10)

a. मार्जिन हाइट (Margin height)
b. CSS
c. फ्रेमसेट (Frameset)
d. बेस, टारगेट (Base, Target)
e. ऐंपरसेंड, सेमीकोलन (Ampersand, semicolon)
f. Var
g. <Script>
h. mailto
i. img और scr (Img and scr)
j. क्लाइंट–साइड (Client-side)
k. Src और alt (Src and alt)

4.1 _____ ने टेक्स्ट, लिस्ट, बॉक्स, मार्जिन, बॉर्डर, रंग और बैकग्राउंड प्रॉपर्टी प्रस्तुत किए हैं। (_____ has introduced text, list, box, margin, border, color, and background properties.)

4.2 हम कीवर्ड के साथ जावास्क्रिप्ट में सभी प्रकार के _____ घोषित कर सकते हैं। (We can declare all type of _____ in JavaScript with the keyword.)

4.3 जावास्क्रिप्ट एन्टीटीएस _____ से शुरू होती हैं और _____ से समाप्त होती हैं। (JavaScript entities start with _____ and end with _____.

4.4 फ्रेम के एक समूह को _____ कहा जाता है। (A group of frames is called a _____.)

4.5 _____ एट्रिब्यूट एक फ्रेम की कंटेंट और उस फ्रेम के ऊपरी और निचले किनारों के बीच की जगह को समायोजित करती है। (The _____ attribute adjusts the space between the content of a frame and the top and bottom edges of that frame.)

4.6 किसी विशेष फ्रेम में लोड करने के लिए पेज पर सभी लिंक को बाध्य करने के लिए _____ टैग और _____ एट्रीब्यूटर का उपयोग करें। (Use the _____ tag and _____ attribute to force all the links on a page to load in a particular frame.)

4.7 img टैग के लिए _____ एट्रिब्यूट पेअर आवश्यक है। (_____ pair of attribute is required for img tag.)

4.8 HTML में, अगर हमें ईमेल भेजने की आवश्यकता है, तो एंकर टैग की href एट्रिब्यूट _____ कीवर्ड से शुरू होती है। (In HTML, if we need to send the email then href attribute of anchor tag starts with _____ keyword.)

4.9 _____ HTML टैग का एक विस्तार है जो किसी भी प्रकार के जावास्क्रिप्ट स्टेटमेंट को संलग्न कर सकता है। (_____ tag is an extension to HTML that can enclose any number of JavaScript statements.)

4.10 _____ एक HTML पृष्ठ में एम्बेड किए गए जावास्क्रिप्ट स्टेटमेंट यूजर इवेंट जैसे कि माउस–क्लिक, इनपुट और पेज नेविगेशन से प्रतिक्रिया दे सकते हैं।(_____ JavaScript statements embedded in an HTML page can respond to user events such as mouse-click, from input, and page navigation.)

भाग दो (PART TWO)

(किसी भी चार प्रश्नो के जवाब दें) (Answer any four questions)

5. a. HTML एलिमेंट , एट्रिब्यूट और टैग में क्या अंतर है? (What is the difference between HTML elements, Attributes and tags?)

b. ब्लॉक–लेवल एलिमेंट और इनलाइन एलिमेंट के बीच अंतर क्या है? (What is the difference between a block-level element and an inline element?)

c. उल्लेख करें कि वेब पेज पर रेस्पॉन्सिव डिजाइन से आपका क्या अभिप्राय है? (Mention what do you mean by responsive design on a web page?)

6. a. निम्न टेबल लेआउट बनाने के लिए HTML कोड लिखें? (Write the HTML code for creating the following table layout?)

Serial No.	Product Name	Quantity	Total Price
1	Rice	2 Kg	500
2	Corn Flour	2 Kg	300
3	Marie Biscuits	2 pkt	20
4	Chilli Powder	200 gm	100
5	Mangoes	4 Kg	700

b. W3-CSS में विभिन्न टेबल क्लासेस क्या हैं? (What are the various table classes in W3.CSS?)

c. एंगुलर JS में कंट्रोलर क्या है? (What is a controller in Angular JS?)

7. a. कुछ सामान्य लिस्ट को नाम दें जो पृष्ठ को डिजाइन करते समय उपयोग की जाती हैं। (Name some common lists that are used when designing a page.)

b. क्लाइंट साइड स्क्रिप्टिंग के फायदे और नुकसान बताए। (Advantages and Disadvantages of Client Side Scripting.)

c. कैस्केडिंग का अर्थ क्या है? स्टाइल शीट कैस्केड कैसे करें? (What is the meaning of Cascading? How do style sheets cascade?)

Scanned with OKEN Scanner

8.

a. निम्नलिखित टर्म्स को समझाइए: (Explain the following terms:)

i. "==" और "===" में क्या अंतर है? (What is the difference between "==" and "==="?)

ii. जावास्क्रिप्ट में एक अन डिफाइंड वैल्यू क्या है? (What is an undefined value in JavaScript?)

iii. जावास्क्रिप्ट में विभिन्न प्रकार के एरर क्या हैं? (What are the different types of errors in JavaScript?)

b. i. CSS में Box Model का उपयोग क्या है? (What is the use of the Box Model in CSS?)

ii. CSS ओपेसिटी क्या है? (What is CSS Opacity?)

iii. छवि मैप क्या है? (What is an image map?)

c. i. CSS में नेविगेशन बार क्या हैं? (What are navigation bars in CSS?)

ii. CSS में रिलेटिव और अब्सोल्युट लिंक के बीच अंतर क्या हैं? (What are the differences between relative and absolute in CSS?)

9. किन्हीं तीन प्रश्नों के उत्तर। (Answers to any three questions.)

a. HTML फॉर्म क्या है? इसका सिंटेक्स लिखिए। (What is HTML Form? Write its syntax.)

b. HTML फॉर्म इनपुट टैग के साथ सबमिट बटन का उपयोग क्या है? (What is the use of the Submit button with HTML Form Input tag?)

c. HTML फ्रेमसेट एलिमेंट क्या है? What is HTML Frameset element?)

d.
 टैग <P> टैग से कैसे भिन्न है? (How is
 Tag different from <P> tag?)

e. विभिन्न फॉन्ट ऐट्रिब्यूट्स क्या उपलब्ध हैं? (What are the different font attributes available?)

ANSWERS TO PART ONE

A.	1.1 d	1.2 c	1.3 a	1.4 c
	1.5. b	1.6 d	1.7 b	1.8 d
	1.9 c	1.10 a		
B.	2.1 F	2.2 T	2.3 T	2.4 T
	2.5 T	2.6 F	2.7 F	2.8 F
	2.9 T	2.10 F		
C.	3.1 f	3.2 i	3.3 h	3.4 a
	3.5 j	3.6 b	3.7 c	3.8 d
	3.9 e	3.10 g		
D.	4.1 b	4.2 f	4.3 e	4.4 c
	4.5 a	4.6 d	4.7 k	4.8 h
	4.9 g	4.10 j		

ANSWERS TO PART TWO

5.

a. एचटीएमएल टैग

HTML एलिमेंट की शुरुआत और अंत को परिभाषित करने के लिए टैग का उपयोग किया जाता है। वे एक प्रारंभिक ब्रैकेट (<) से मिलकर बने होते हैं, इसके बाद एलिमेंट का नाम और फिर एक कोष्ठक (>) होता है। यदि टैग में एक एट्रिब्यूट का उपयोग किया जा रहा है, तो यह एलिमेंट के बाद शामिल किया जाएगा। HTML टाइटल टैग का एक उदाहरण <title> <title>।

HTML Tags

Tags are used to define the beginning and end of an HTML element. They consist of an opening bracket (<), followed by the name of the element and then a closing bracket (>). If an attribute is being used in the tag, it will be included after the element. An example of the HTML title tag <title> <title>.

HTML एलिमेंट

तत्व में ओपनिंग और क्लोजिंग दोनों टैग्स के साथ–साथ उन टैग्स के अंदर क्या है। इसमें आम तौर पर कुछ संरचनाएं होती हैं जिनका उपयोग संबंधित टैग को परिभाषित करने के लिए किया जाता है। HTML टाइटल एलिमेंट का एक उदाहरण।

<title>My Webpage</title>। इस उदाहरण में, HTML एलिमेंट "My Webpage" शीर्षक है जो ओपनिंग और क्लोजिंग टाइटल टैग के साथ पूरा होता है।

HTML Elements

The element consists of both the opening and closing tags as well as what's inside those tags. It normally consists of some structure that is used to define the respective tags. An example of the HTML title element.

<title>My Webpage</title>. In this example, the HTML element is a title of "My Webpage" complete with the opening and closing title tags

HTML ऐट्रिब्यूट्स

प्रॉपर्टी एक या अधिक HTML एलिमेंट के लिए एक प्रॉपर्टी को परिभाषित करने के लिए उपयोग किया जाता है। वे एलिमेंट के ओपनिंग टैग के भीतर पाए जाते हैं, जिसमें अक्सर ऐसे स्थान होते हैं जो मूल्य जोड़े द्वारा अलग किए जाते हैं। HTML Alt टेक्स्ट एट्रिब्यूट का एक उदाहरण:

<img src="Peacock-dancing.gif" alt="dancing outside the park">

HTML Attributes

Attributes are used to define a property for one or more HTML elements. They are found within the element's opening tag, often containing spaces that are separated by value pairs. An example, of the the HTML alt text attribute:

Scanned with OKEN Scanner

```
<img src="Peacock-dancing.gif" alt="dancing outside the park">
```

b. एक ब्लॉक–लेवल एलिमेंट को एक ब्लॉक के रूप में तैयार किया गया है, जो इसे उपलब्ध पूर्ण चौड़ाई को भरने के लिए फैला है, अर्थात इसके कंटेनर की चौड़ाई और हमेशा एक नई लाइन पर शुरू होगी। डिफॉल्ट रूप से ब्लॉक–स्तर वाले तत्व: , <img>, <section>, <form>, <nav> है।

इनलाइन एलिमेंट ड्रा किये जाते हैं जहां वे डिफाइन होते हैं और केवल वही स्थान लेते हैं जिसकी पूरी जरूरत होती है। यह समझने का सबसे आसान तरीका है कि वे कैसे काम करते हैं, यह देखना है कि पृष्ठ पर पाठ कैसे फ्लो है। उन तत्वों के उदाहरण जो डिफॉल्ट रूप से इनलाइन हैं: , <b>, , <a>, <input>।

A block-level element is drawn as a block that stretches to fill the full width available to it i.e, the width of its container and will always start on a new line. Elements that are block-level by default: , <img>, <section>, <form>, <nav>.

Inline elements are drawn where they are defined and only take up space that is absolutely needed. The easiest way to understand how they work is to look at how text flows on a page. Examples of elements that are inline by default: , <b>, , <a>, <input>.

c. रेस्पॉन्सिव डिजाइन एक इष्टतम देखने और बातचीत का अनुभव प्रदान करने के लिए साइट्स के निर्माण के लिए एक दृष्टिकोण है। यह सभी उपकरणों पर न्यूनतम स्क्रॉलिंग, पैनिंग और आकार बदलने के साथ साइट के आसान नेविगेशन पर ध्यान केंद्रित करता है। बूटस्ट्रैप रेस्पॉन्सिव वेब डिजाइन को विकसित करने के लिए उपयोग किया जाने वाला सबसे लोकप्रिय CSS, HTML और JS फ्रेमवर्क है।

Responsive design is an approach to building sites to provide an optimal viewing and interaction experience. It focuses on easy navigation of site with a minimum of scrolling, panning and resizing across all devices. Bootstrap is the most popular CSS, HTML and JS framework used for developing responsive web design.

6.

a.

```
<!DOCTYPE html>
<html>
<head>
<style>
table, th, td {
border: 2px solid black;
}
</style>
</head>
<body>
<table >
<tr>
<th> Serial No. </th> <th> Product Name </th> <th> Quantity </th> <th> Total Price </th> </tr>
<tr>
<td> 1 </td> <td>Rice</ td> <td> 2 Kg </td> <td> 500 </td> </tr>
<tr> <td> 2 </td> <td> Corn Flour </td> <td> 2 Kg </td> <td> 300 </td> </tr>
<tr> <td> 3 </td> <td> Marie Biscuits </td> <td> 2 pkt </ td> <td> 20 </td> </tr>
<tr><td>4</td> <td>Chilli Powder </td> <td> 200 gm </td> <td> 100 </td> </tr>
<tr> <td> 5 </td> <td>Mangoes</ td> <td> 4 Kg</td> <td> 700 </ td> </tr>
</table>
</body>
</html>
```

b. W3.CSS में विभिन्न टेबल क्लासेज हैं।

w3-table: यह किसी भी बॉर्डर के बिना एक मूल टेबल का प्रतिनिधित्व करता है।

w3- striped: इसका उपयोग स्ट्रिपपड़. टेबल को प्रदर्शित करने के लिए किया जाता है।

w3-bordered: इसका उपयोग पंक्तियों में एक सीमा के साथ टेबल बनाने के लिए किया जाता है।

w3-card: इसका उपयोग कार्ड के रूप में टेबल बनाने के लिए किया जाता है।

There are various table classes in W3.CSS.

w3-table: It represents a basic table without any border.

w3-striped: It is used to display a stripped table.

w3-bordered: It is used to draw table with a border across rows.

w3-card: It is used to draws a table as a card.

c. कंट्रोलर एक जावास्क्रिप्ट फंक्शन है जो निर्दिष्ट दायरे से जुड़ा हुआ है। एंगुलर नए कंट्रोलर ऑब्जेक्ट को इंस्टेंट करता है और एक निर्भरता के रूप में नए दायरे को इंजेक्ट करता है। कंट्रोलर का उपयोग स्कोप ऑब्जेक्ट की प्रारंभिक अवस्था को सेट करने और ऑब्जेक्ट में बेहेवियर जोड़ने के लिए किया जा सकता है। कंट्रोलर

Scanned with OKEN Scanner

में कोड या स्थिति साझा करने के लिए एक कंट्रोलर का उपयोग नहीं किया जा सकता है, लेकिन इसके बजाय एंगुलर सेवा का उपयोग किया जा सकता है।

A controller is a JavaScript function that is bound to the specified scope. Angular instantiates the new controller object and injects the new scope as a dependency. A controller can be used to set up the initial state of the scope object and to add behavior to the object. A controller cannot be used to share code or state across controllers, but instead of that Angular service can be used.

7.

a. पेज डिजाइन करने के लिए कई सामान्य लिस्ट का उपयोग किया जाता है। आप निम्न लिस्ट प्रकारों में से किसी एक या कॉम्बिनेशन को चुन सकते हैं:

- **ओर्देरेद लिस्टः** ओर्देरेद लिस्ट एक क्रमांकित प्रारूप में एलिमेंट्स को प्रदर्शित करती है। इसका प्रतिनिधित्व <ol> टैग द्वारा किया जाता है।
- **अन–ओर्देरेद लिस्टः** अन–ओर्देरेद लिस्ट बुलेटेड प्रारूप में एलिमेंट को प्रदर्शित करती है। इसका प्रतिनिधित्व <ul> टैग द्वारा किया जाता है।
- **डेफिनिशन लिस्टः** डेफिनेशन लिस्ट डिक्शनरी में एलिमेंट को डिक्शनरी फॉर्म की तरह प्रदर्शित करती है। वर्णन सूची को परिभाषित करने के लिए <dl>, <dt> और <dd> टैग का उपयोग किया जाता है।

There are many common lists used for design a page. You can choose any or a combination of the following list types:

- **Ordered list:** The ordered list displays elements in a numbered format. It is represented by <ol> tag.
- **Unordered list:** The unordered list displays elements in a bulleted format. It is represented by <ul> tag.
- **Definition list:** The definition list displays elements in definition form like in a dictionary. The <dl>, <dt> and <dd> tags are used to define description list.

b. डेवलपर्स कई कारणों से क्लाइंट–साइड स्क्रिप्ट का उपयोग करने पर विचार कर सकते हैं, और उनका उपयोग न करने के कुछ वैध कारण हैं।

Developers might consider using client-side scripts for many reasons, and they migh have some valid reasons not to use them.

लाभ

क्लाइंट–साइड स्क्रिप्टिंग कई लाभ प्रदान करता है:

1. यह उपयोगकर्ता के कार्यों का तुरंत जवाब देकर अधिक अन्तरक्रियाशीलता की अनुमति देता है।
2. क्लाइंट–साइड स्क्रिप्टिंग का उपयोग उपयोगकर्ताओं को सर्वर से अनुरोध भेजने के लिए किया जाता है, और सर्वर से डेटा को पुनः प्राप्त करने के लिए भी किया जाता है।
3. क्लाइंट साइड स्क्रिप्टिंग का उपयोग अनुप्रयोगों के लिए दूरस्थ सेवा प्रदान करने के लिए किया जाता है, जैसे कंटेंट वितरण, सॉफ्टवेयर रेगिस्ट्रशन ।
4. यदि उपयोगकर्ता के ब्राउजर स्क्रिप्ट का समर्थन नहीं करते हैं, तो इसे उदाहरण के लिए, भ्ज्डस् के साथ प्रतिस्थापित किया जा सकता है।

Advantages

Client-side scripting offers various advantages, including:

1. It allows for more interactivity by immediately responding to user's actions.
2. Client-side scripting is used to send users requests to the server, and also retrieve data from the server.
3. Client side scripting is used to provide remote service for applications, such as content delivery, software registration.
4. It can be substituted with alternative for example, HTML if user's browsers do not support scripts.

नुकसान

1. सभी ब्राउजर स्क्रिप्ट का समर्थन नहीं करते हैं, इसलिए, यदि कोई विकल्प प्रदान नहीं किया गया है, तो उपयोगकर्ता एरर का अनुभव कर सकते हैं।
2. ब्राउजर के विभिन्न संस्करण अलग–अलग प्रकार की स्क्रिप्ट का समर्थन करते हैं।
3. यदि अन्य संसाधन के माध्यम से स्क्रिप्ट पहले से उपलब्ध नहीं हैं, तो अधिक विकास समय और प्रयास की आवश्यकता हो सकती है।

Disadvantages

1. Not all browsers support scripts, so, users may experience errors if no alternatives have been provided.
2. Different versions of browser support different type support scripts differently.
3. More development time and effort might be required if the scripts are not already available through other resource.

c. CSS ने वेब–डेवलपमेंट में क्रांति ला दी और लोगों को वेबसाइट बनाने की प्रक्रिया का अनुभव हुआ। सीएसएस के अस्तित्व से पहले, तत्वों को एक इन–लाइन फैशन में स्टाइल करना पड़ता था या स्टाइल एक HTML पृष्ठ के प्रमुख भाग में लागू की जाती थी। सीएसएस की कैस्केडिंग प्रकृति के कारण इसे बदल दिया गया था। यहां सीएसएस कैस्केड के तीन प्रमुख तरीके दिए गए हैं:

Scanned with OKEN Scanner

एलिमेंटः एक ही सीएसएस स्टाइल को एक ही स्टाइल को प्राप्त करने के लिए कई तत्वों पर लागू किया जा सकता है।

मल्टीपल स्टाइल वन एलीमेंटः एक अनूठी शैली को प्राप्त करने के लिए कई शैलियों को एक विशेष एचटीएमएल तत्व पर लागू किया जा सकता है।

एकाधिक पृष्ठों के साथ एक ही स्टाइलः एक ही स्टाइल को विभिन्न HTML पृष्ठों पर पूरी तरह से लागू किया जा सकता है ताकि टेम्पलेट स्टाइल को बहुत जल्दी प्राप्त किया जा सके।

CSS brought about a revolution in web-development and how people perceive the process of building a website. Prior to the existence of CSS, elements had to be styled in an in-line fashion or the style were implemented in the head section of an HTML page. This was changed due to the cascading nature of CSS. Here are the three major ways CSS cascades:

Elements: The same CSS style can be applied to multiple elements to achieve the same style.

Multiple Style one Element: Multiple styles can be applied to a particular HTML element to achieve a unique style.

Same style with Multiple pages: The same styles can be applied to different HTML pages altogether to achieve a template styling very quickly.

8.

a. **i.** "==" केवल वैल्यू में समानता के लिए जाँच करता है जबकि "===" एक सख्त समानता परीक्षण है और यदि दोनों वेरिएबल के मान या प्रकार भिन्न हैं तो फालस रिटर्न देता है। तो, दूसरे विकल्प को ऑपरेंड्स के लिए समान होने के लिए मूल्य और प्रकार दोनों की आवश्यकता होती है।

"==" checks only for equality in value whereas "===" is a stricter equality test and returns false if either the value or the type of the two variables are different. So, the second option needs both the value and the type to be the same for the operands.

ii. अन–डिफाइंड वैल्यू का अर्थ है:

कोड में प्रयुक्त वेरिएबल मौजूद नहीं है।

वेरिएबल को किसी भी मूल्य पर नहीं सौंपा गया है

प्रॉपर्टी मौजूद नहीं है

Undefined value means the:

Variable used in the code does not exist.

Variable is not assigned to any value

Property does not exist

iii. एरर के तीन प्रकार हैं:

लोड टाइम एररः गलत सिंटैक्स एरर जैसे वेब पेज को लोड करते समय आने वाली एरर लोड टाइम एरर के रूप में जाना जाता है और यह गतिशील रूप से एरर को उत्पन्न करता है।

रन टाइम एररः एरर जो HTML लैंग्वेज के अंदर कमांड के गलत इस्तेमाल के कारण आती हैं।

लॉजिकल एररः ये एरर हैं जो किसी फंक्शन पर किए गए गलत तर्क के कारण होती हैं।

There are three types of errors:

Load time errors: Errors which come up when loading a web page like improper syntax errors are known as Load time errors and it generates the errors dynamically.

Run time errors: Errors that come due to misuse of the command inside the HTML language.

Logical errors: These are errors that occur due to the wrong logic performed on a function.

b. **i.** CSS बॉक्स मॉडल एक बॉक्स है जो सभी HTML एलिमेंट को बांधता है और इसमें मार्जिन, बॉर्डर, पैडिंग और वास्तविक कंटेंट जैसी विशेषताएं शामिल हैं।

In CSS, the box model is a box that binds all the HTML elements and it includes features like margins, border, padding and the actual content.

ii. यह वह प्रॉपर्टी है जो किसी एलिमेंट की पारदर्शिता पर विस्तृत होता है। इस प्रॉपर्टी के द्वारा, हम उस छवि को पारदर्शी कर सकते हैं जो 0.0–1.0 से मान ले सकती है। यदि मान कम है, तो छवि अधिक पारदर्शी है। IE8 और ब्राउजर के पुराने संस्करण 0–100 से मान ले सकते हैं।

It is the property that elaborates on the transparency of an element. By this property, we can transparent the image that can take the values from 0.0-1.0. If the value is lower, then the image is more transparent. IE8 and earlier versions of the browser can take the value from 0-100.

iii. एक इमेज मैप का उपयोग एक ही इमेज का उपयोग करके कई अलग–अलग वेब पृष्ठों को जोड़ने के लिए किया जाता है। इसे <map> टैग द्वारा दर्शाया गया है। आप उन छवियों में आकृतियों को परिभाषित कर सकते हैं जिन्हें आप एक छवि मानचित्रण के भाग के रूप में शामिल करना चाहते हैं।

An image map is used for linking many different web pages using a single image. It is represented by <map> tag. You can define shapes in images that you want to include as part of an image mapping.

c. **i.** नेविगेशन बार का उपयोग करके हम एक साधारण HTML पेज को उपयोगकर्ता–विशिष्ट और अधिक गतिशील वेब पेज में बना सकते हैं। मूल रूप से, यह लिंक की एक सूची है, इसलिए <ul> और <li> एलिमेंट का उपयोग पूर्ण समझ में आता है।

उदहारण

ul{

Scanned with OKEN Scanner

```
list-style-type: none;
margins: 0;
padding: 0;
}
```

By using navigation bars we can make an ordinary HTML page into a user-specific and more dynamic web page. Basically, it is a list of links, hence use of <ul> and <li> elements makes the perfect sense.

For example,

```
ul{
list-style-type: none;
margins: 0;
padding: 0;
}
```

ii. रिलेटिव और अब्सॉल्यूटे के बीच मुख्य अंतर यह है कि "रिलेटिव" का उपयोग CSS में एक ही टैग के लिए किया जाता है और इसका मतलब है कि अगर हम बाईं ओर लिखते हैं: 10px तो बाईं ओर पैडिंग 10px में बदल जाएगी जबकि रिलेटिव पूरी तरह से नॉन स्थैतिक पैरेंट है ।

The main differences between relative and absolute is that "relative" is used for the same tag in CSS and it means that if we write the left:10px then the padding will shift to 10px in the left while absolute is totally relative to the non-static parent.

9.

a. किसी सर्वर पर डेटा पास करने के लिए HTML फॉर्म का उपयोग किया जाता है। एक फॉर्म में टेक्स्ट फील्ड, चेकबॉक्स, रेडियो–बटन, सबमिट बटन और बहुत कुछ जैसे इनपुट एलिमेंट हो सकते हैं। एक फॉर्म में चुनिंदा लिस्ट, टेक्स्ट एरिया, फील्डसेट, लेजेंड और लेबल एलिमेंट को भी शामिल किया जा सकता है।

HTML फॉर्म बनाने के लिए <फॉर्म> टैग का उपयोग किया जाता है:

```
. input elements
.
</form>
```

HTML forms are used to pass data to a server. A form can contain input elements like text fields, checkboxes, radio-buttons, submit buttons and more. A form can also contain selects lists, textarea, fieldset, legend, and label elements.

The <form> tag is used to create an HTML form:

```
<form>
. input elements
.
</form>
```

b. फ्रेमसेट एलिमेंट में एक या अधिक फ्रेम एलिमेंट होते हैं। प्रत्येक फ्रेम तत्व एक अलग दस्तावेज ले सकता है। फ्रेमसेट तत्व बताता है कि कितने कॉलम या रौस फ्रेमसेट में होंगी और उनमें से प्रत्येक पर कितने प्रतिशत/ पिक्सेल का स्थान होगा।

The frameset element holds one or more frame elements. Each frame element can hold a separate document. The frameset element states HOW MANY columns or rows there will be in the frameset, and how much percentage/pixels of space will occupy each of them.

c. लाइन ब्रेक टैग, <BR>, एक पैरा टैग के समान है, लेकिन यह थोड़ा अलग तरीके से व्यवहार करता है। यह वर्तमान पैराग्राफ के भीतर एक नई लाइन शुरू करता है, लेकिन यह एक नया पैराग्राफ शुरू नहीं करता है।

A line break tag, <BR>, is similar to a paragraph tag, but it behaves in a slightly different way. It starts a new line within the current paragraph, but it does not start a new paragraph.

d. उपलब्ध विभिन्न फॉन्ट एट्रिब्यूट में फॉन्ट–वैरिएंट, फॉन्ट–वेट, फॉन्ट–स्टाइल, फॉन्ट–फॅमिली, फॉन्ट आकार की रेखा–ऊंचाई, कैप्शन, आइकन और फॉन्ट–परिवार शामिल हैं।

The various font attributes available include font-variant, font-weight, font-style, font-family, line-height of font size, caption, icon and font-family.

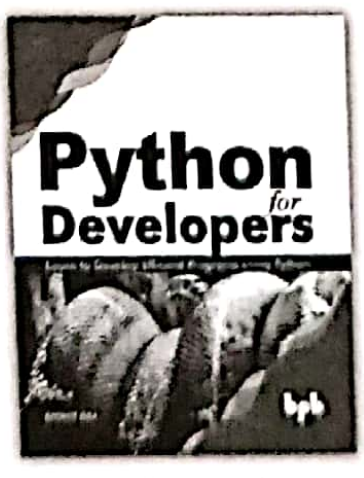
OTHER TITLES OF INTEREST



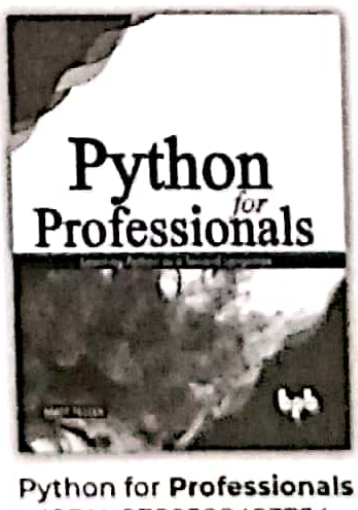
Python Made Simple
ISBN: 9789388511025



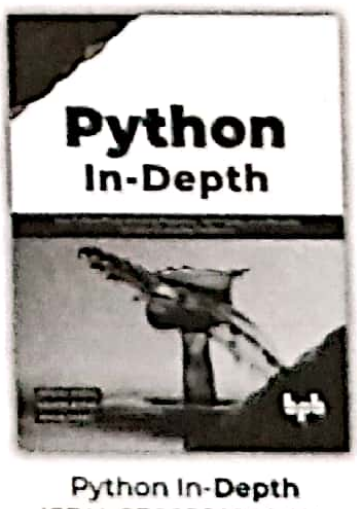
Programming in Python
ISBN: 9789386551276



Python for **Developers**
ISBN: 9788194401872



Python for **Professionals**
ISBN: 9789389423754



Python In-**Depth**
ISBN: 9789389328424

Scanned with OKEN Scanner

Unsolved Sample Paper 1

भाग एक (PART ONE)

1. नीचे प्रत्येक प्रश्न के उत्तर के कई विकल्प दिए गए है। एक सबसे उपयुक्त विक्लप चुने और प्रश्न पत्र के साथ उपलब्ध कराए गए "ओएमआर" उत्तर पुस्तिका में दिए गए निर्देसज के अनुसार दर्ज करें। (Each question below gives a multiple choice of answers. Choose he most appropriate one and enter in the "OMR" answer sheet supplied with the question paper, following instruction therein. (1 x 10)

1.1 सीएसएस में "रंगः लाल" को क्या कहा जा सकता हैः (In CSS what does "color:red" can be called as:

a. सिलेक्टर (Selector) **b.** रूल (Rule)
c. डिक्लेरेशन (Declaration)
d. उपर्युक्त कोई नहीं (None of the above)

1.2 निम्नलिखित में से किस एलिमेंट का उपयोग किसी एलिमेंट को विशिष्ट नाम प्रदान करने के लिए किया जाता है? (Which of the following attribute is used to provide a unique name to an element?)

a. क्लास (Class) b. ID
c. टाइप (Type) d. उपर्युक्त कोई नहीं (None of the above)

1.3 यदि हम किसी छवि के चारों ओर एक अच्छी दिखने वाली हरी बिंदीदार सीमा का उपयोग करना चाहते हैं, तो हम किस सीएसएस प्रॉपर्टी का उपयोग करेंगे? (If we want to use a nice looking green dotted border around an image, which CSS property will we use?)

a. border-color **b.** border-decoration
c. border-style **d.** border-line

1.4 निम्नलिखित में से कौन सा जावास्क्रिप्ट का उपयोग करके वार्निंग बॉक्स में "हैलो वर्ल्ड" प्रदर्शित करने के लिए सही वाक्यविन्यास है? (Which of the following is the correct syntax to display "Hello World" in an alert box using JavaScript?)

a. Alert-box("Hello World");
b. Confirm("Hello World");
c. Msgbox("Hello World");
d. alert("Hello World");

1.5 निम्नलिखित में से कौन सा जावास्क्रिप्ट का सबसे अच्छा वर्णन करता है? (Which of the following best describes JavaScript?)

a. लौ लेवल प्रोग्रामिंग लैंग्वेज (A low-level programming Language)
b. एक स्क्रिप्टिंग लैंग्वेज ब्राउजर में प्री–कंम्पाइलेड (A scripting language precompiled in the browser)
c. कंम्पाइलेड स्क्रिप्टिंग लैंग्वेज (A compiled scripting Language)
d. ऑब्जेक्ट ओरिएंटेड स्क्रिप्टिंग लैंग्वेज (An object-oriented scripting Language)

1.6 किसी भी पृष्ठ तत्व के लिए डिफॉल्ट है? (The default for any page element is?)

a. फिक्स्ड (Fixed) **b.** रिलेटिव (Relative)
c. अब्सोल्युट (Absolute) **d.** स्थैतिक (Static)

1.7 HTML टैग _____ एक HTML दस्तावेज के बारे में मेटाडेटा को परिभाषित करता है। (HTML tag _____ defines metadata about an HTML document.)

a. <meta> b. <link> c. <met> d. <data>

1.8 निम्नलिखित में से कौन सी सीएसएस प्रॉपर्टी में तैनात तत्वों का स्टैकिंग क्रम सेट है? (Which of the following CSS property sets the stacking order of positioned elements?)

a. x-index b. y-index
c. z-index d. ऊपर के सभी (All of the above)

1.9 बॉक्स मॉडल का कौन सा हिस्सा कंटेंट के आसपास के क्षेत्र को साफ करता है? (Which part of box model clears an area around the content?)

a. पैडिंग (Padding) b. बॉर्डर (Border)
c. मार्जिन (Margin) d. उपर्युक्त कोई नहीं (None of the above)

1.10 निम्नलिखित में से कौन सा ट्रू है? (Which of the following is true?)

a. आपको अपनी वेबसाइट फाइल्स को होस्ट करने के लिए एक सर्वर की आवश्यकता है। (You need a server to host your website files)
b. आपको अपनी वेबसाइट फाइल्स को होस्ट करने के लिए सर्वर की आवश्यकता नहीं है। (You don't need a server to host your website files.)
c. आप HTML का उपयोग किए बिना एक वेबसाइट बना सकते हैं। (You can create a website without using HTML)
d. आप बिना CMS के वेबसाइट नहीं बना सकते। (You cannot create a website without a CMS.)

2. नीचे दिया गया प्रत्येक विवरण या तो सही है अथवा गलत। एक सबसे उपयुक्त विक्लप चुने और उसे प्रश्न पत्र के साथ उपलब्ध कराए गए "ओएमआर" उतर पत्रिका में, उसमे दिए गए निर्देशों के अनुसार दर्ज करे। (Each statement below is either TRUE or False, Choose the most appropriate one and ENTER in the "OMR" answer sheet supplied with the question paper, following instructions therein.) (1 x 10)

Scanned with OKEN Scanner

2.1 आपको अपने कंप्यूटर पर वेब पेज में बदलावों को सत्यापित करने के लिए इंटरनेट से कनेक्ट होने की आवश्यकता नहीं है। (You do not have to connect to the internet to verify changes to a Web pages on your computer.)

2.2 आप इनलाइन छवि को हाइपरटेक्स्ट लिंक के रूप में नामित नहीं कर सकते। (You cannot designate an inline image as a hypertext link.)

2.3 क्या केवल एक सीएसएस प्रॉपर्टी का उपयोग करके फॉन्ट–वेट , फॉन्ट–फेस और फॉन्ट–आकार की घोषणा करना संभव है? (Is it possible to declare font-weight, font-face & font-size by using only one CSS property?)

2.4 HTML <list> टैग का उपयोग कई विकल्पों के साथ ड्रॉप डाउन सूची बनाने के लिए किया जाता है। (HTML <list> tag is used to create a drop down list with multiple options.)

2.5 स्पैन एलिमेंट एक ब्लॉक लेवल एलिमेंट है। (The span element is a block-level element.)

2.6 HTML टैग केस सेंसिटिव हैं। (HTML tags are case sensitive.)

2.7 HTML के सभी संस्करणों द्वारा मान्यता प्राप्त 256 रंग नाम हैं। There are 256 color names recognized by all versions of HTML.)

2.8 इनलाइन एलिमेंट को आम तौर पर एक नई लाइन शुरू किए बिना प्रदर्शित किया जाता है। (Inline elements are normally displayed without starting a new line.)

2.9 फ्रेम बॉर्डर को पूरी तरह से ब्राउजर के बहुमत से बंद करने के लिए, फ्रेमबॉर्डर = "0" को ओपनिंग फ्रेमसेट टैग में जोड़ें। (To turn the frame borders completely off in the majority of browsers, add frameborder="0" to the opening frameset tag.)

2.10 टेबल में कैप्शन जोड़ने के लिए, आप ओपनिंग टेबल टैग में कैप्शन एट्रिब्यूट का उपयोग करते हैं। (To add a caption to a table, you use the caption attribute in the opening table tag.)

3. कॉलम X में दिए गए शब्दों और वाक्यांशों का मिलान निकटतम अर्थ/ शब्द/ फ्रांसेस कॉलम Y के साथ करें। अपना चयन, प्रश्नपत्र के साथ दी गई "ओएमआर" उत्तर पुस्तिका में दर्ज करें। (Match words and phrases in column X with the closest related meaning/word(s)/phrases in column Y. Enter your selection in the "OMR" answer sheet supplied with the question paper, following instructions therein.) **(1 x 10)**

कॉलम X (Column X)		कॉलम Y (Column Y)	
3.1	वेरिएबल की घोषणा को प्रतीक द्वारा अलग किया जाता है (Declaration of variables are separated by symbol)	a.	पैडिंग (Padding)
3.2	यह टेबल को बिना बॉर्डर का प्रतिनिधित्व करता है (It represents a basic table without and Border)	b.	इनेबल और डिसएबल ऑप्शन (To enable or disable option)
3.3	हॉटस्पॉट है: (Hotspot refers to)	c.	होस्टिंग (Hosting)
3.4	प्रॉपर्टी निर्दिष्ट करने की अनुमति देती है कि किसी एलिमेंट और उसके बॉर्डर के बीच कितनी जगह दिखाई देती है (The property allows specifying how much space appears between the content of an element and its Border)	d.	टेबल हैडर (Table header)
3.5	वेब पेज एडिटर एक सिद्धांत पर काम करता है (Web page editors works on a principle)	e.	टेक्स्ट–डेकोरेशन:ओवेर्लिने (Text-decoration:overline)
3.6	चेक बॉक्स का उपयोग किया जाता है (Check boxes are used)	f.	मेटा–सर्च इंजन (Meta-search engine)
3.7	अपने html कोड को एक मशीन से सर्वर पर स्थानांतरित करने के रूप में जाना जाता है (Transferring your html code from one machine to server is known as)	g.	कॉमा (Comma)
3.8	<Th> टैग में, th का अर्थ है (In the <th> tag, th stands for)	h.	Dogpile
3.9	सीएसएस प्रॉपर्टी टेक्स्ट पर एक पंक्ति देने के लिए उपयोग किया जाता है। (CSS property used to give a line over the TEXT)	i.	WYSIWYG
3.10	मेटा–सर्च इंजन का उदाहरण: (Example of meta-search engine)	j.	वाई–फाई के उपयोग से इंटरनेट का उपयोग (Region with internet access using Wi-Fi)
		k.	W3-टेबल (W3-table)

Scanned with OKEN Scanner

4. नीचे दिए गए सूची में किसी एक शब्द या वाक्यांश में से एक को भरने करने के लिए एक खाली स्थान है। निर्देश के अनुसार, प्रश्न पत्र से जुड़ी "ओएमआर" उत्तर पुस्तिका में अपनी पसंद दर्ज करें। (Each statement below has a blank space to fit one of the word(s) or phrase(s) in the list below. Enter your choice in the "OMR" answer sheet attached to the question paper, following instruction therein.) **(1 x 10)**

a. Cols b. टैग्स (Tags)
c. Alt एट्रिब्यूट (Alt attribute)
d. एलिमेंट सिलेक्टर (Element selector)
e. पोर्टेबल (Portable)
f. बॉर्डर –रेडियस(border-radius)
g. सबलाइम टेक्स्ट (Sublime Text)
h. बूटस्ट्रैप (Bootstrap) i. Src
j. एम्बेड (Embed)

4.1 फ्रेम टैग की _____ विशेषता ब्राउजर को उस फ्रेम में लोड करने के लिए HTML पेज बताती है। (The _____ attribute of the frame tag tells the browser which HTML page to load into that frame.)

4.2 _____ विशेषता औसत चरित्र चौड़ाई के आधार पर टेक्स्ट एरिया की दृश्यमान चौड़ाई की पहचान करती है। (The _____ attribute identifies the visible width of a text area based on an average character width.)

4.3 रिलेटिव पथ आपके हाइपरटेक्स्ट लिंक को _____ बनाता है। (Relative path makes your hypertext links _____.)

4.4 _____ HTML कोड हैं जो दस्तावेज कंटेंट की उपस्थिति को नियंत्रित करते हैं। (_____ are the HTML codes that control the appearance of the document contents.)

4.5 _____ नाम से HTML एलिमेंट का चयन करता है। (_____ selects the HTML element by name.)

4.6 यदि चित्र प्रदर्शित नहीं किया जा सकता है, तो _____ एक छवि के लिए एक वैकल्पिक पाठ निर्दिष्ट करता है। (If the image cannot be displayed, then _____ specifies an alternate text for an image.)

4.7 _____ रेस्पॉन्सिव और मोबाइल–प्रथम वेबसाइट को विकसित करने के लिए सबसे लोकप्रिय सीएसएस फ्रेमवर्क है। (_____ is the most popular CSS framework for developing responsive and mobile-first websites.)

4.8 छवि में गोल कोनों को जोड़ने के लिए _____ प्रॉपर्टी का उपयोग करें। (Use the _____ property to add rounded corners to an image.)

4.9 _____ टैग का उपयोग एक वीडियो या ऑडियो का HTML दस्तावेज में शामिल करने के लिए किया जाता है। (An _____ tag is used for including a video or audio in an HTML document.)

4.10 _____ पायथन एप्लिकेशन प्रोग्रामिंग इंटरफेस के साथ एक क्रॉस–प्लेटफॉर्म स्रोत कोड एडिटर है। (_____ is a cross-platform source code editor with a Python application programming interface.)

भाग दो (PART TWO)

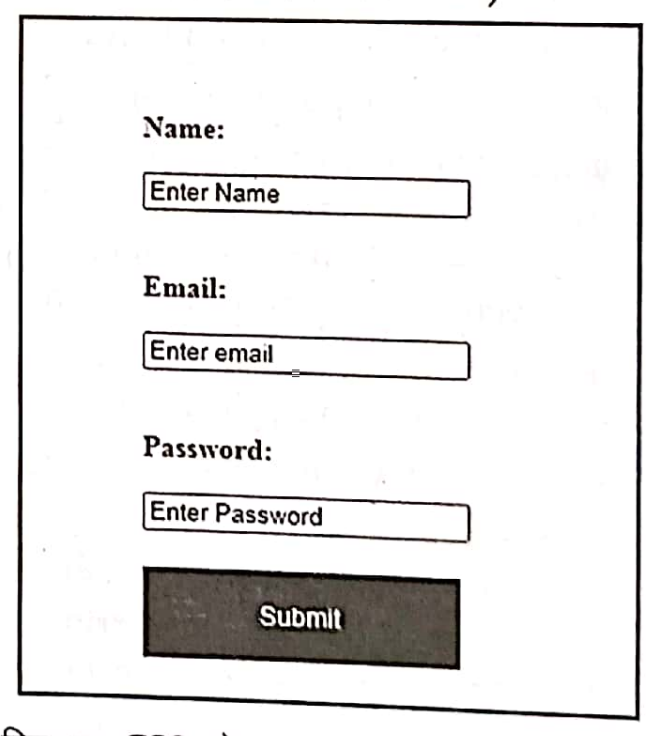
(किसी भी चार प्रश्नो के जवाब दें) (Answer any four questions)

5. a. स्थैतिक पेज और डायनामिक वेब पेज के बीच अंतर करें। (Differentiate between static page and dynamic web page.)

b. इमेज में टेक्स्ट का उपयोग करने का उद्देश्य क्या है? (What is the purpose of using alternative texts in images?)

c. एक्सटर्नल स्टाइल शीट्स के क्या लाभ और नुकसान हैं? (What benefits and demerits do External Style Sheets have?)

6. a. HTML में रूपों को परिभाषित करें। फॉर्म प्रदर्शित करने के लिए, फॉर्म बनाने के लिए HTML कोड लिखें। (Define forms in HTML. To Display the form, write the HTML code to create a form.)



b. दिए गए CSS कोड का क्या प्रभाव होगा? {box-sizing: border-box;} (What effect would this piece of CSS code have? {box-sizing: border-box;})

c. एंगुलर एक्सप्रेशन और जावास्क्रिप्ट एक्सप्रेशन के बीच मुख्य अंतर को स्पष्ट करें (Explain the key difference between angular expressions and JavaScript expressions)

Scanned with OKEN Scanner

7. **a.** टेबल क्या है? टेबल बनाने के लिए मूल आदेश क्या हैं? (What is table? What are the basic commands for creating a table?)

   b. आप एक ही पृष्ठ के साथ सेक्शंस के लिंक कैसे बनाते हैं? (How do you create links to sections with the same page?)

   c. W3.CSS इमेज क्लास के उपयोग क्या हैं? (What are the uses of W3.CSS image classes?)

8. **a.** निम्नलिखित टर्म्स को समझाइए: (Explain the following terms:)

   i. अलर्ट बॉक्स और कन्फर्मेशन बॉक्स में क्या अंतर है? (What is the difference between an alert box and a confirmation box?)

   ii. BGCOLOR को NAVY, हेडर 1 <H1> टेक्स्ट "टेक्स्ट उदाहरण के साथ आकार, रंग और फॉन्ट फेस" को व्हाइट, टेक्स्ट फॉन्ट साइज 20, कलर "पर्पल" और फॉन्ट फेस "एरियल" के रूप में सेट करने के लिए html कोड लिखें। (Write the html codes to set the BGCOLOR as NAVY, header 1 <H1> text "Text Example with size, color and font face" as White, text font size as 20, color "purple" and font face "Arial".)

   iii. एक वेब पेज में इमेज कैसे इन्सर्ट की जा सकती है? (How can an image be inserted in a Web Page?)

   b. i. <A> टैग में विशेषता HREF का उपयोग करने का उद्देश्य क्या है? (What is the purpose of using the attribute HREF in tag <A>?)

   ii. किसी लिस्ट में TYPE एट्रिब्यूट का उपयोग क्या है? (What is the use of TYPE attribute in a list?)

   iii. HTML पेज के तीन संरचनात्मक टैग का नाम और उनमें से किसी एक के उपयोग की व्याख्या करें। (Name the three structural tags of HTML page and explain the usage of any one of them.)

   c. i. आईडी और क्लास के उपयोग में क्या अंतर है? (What is the difference between the usage of an ID and a Class?)

   ii. सेक्शंस के एक अनुभाग को अलग करने के लिए उपयोग किए जाने वाले टैग क्या हैं? What are the tags used to separate a section of texts?)

   iii. CSS में बॉक्स–शैडो का उपयोग क्या है? (What is the use of box-shadow in CSS?)

9. किन्हीं तीन प्रश्नों के उत्तर दीजिए। (Answers to any three questions.)

   a. <? DOCTYPE HTML> टैग को HTML टैग माना जाता है? (Is the <!DOCTYPE html> tag considered as a HTML tag?)

   b. ALINK से आप क्या समझते हैं? एक उदाहरण से समझाएं। (What do you understand by ALINK? Explain with an example.)

   c. SELECT टैग का उद्देश्य क्या है? (What is the purpose of SELECT tag?)

   d. HTML में कौन सी इकाइयाँ हैं? (What are the entities in HTML?)

   c. निम्नलिखित के लिए HTML कोड विकसित करें: (Develop HTML code for the following:)

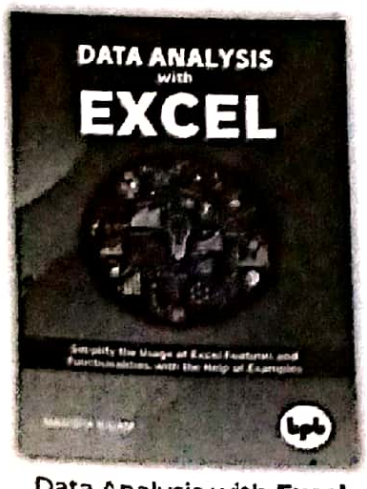
   A. पिक्चर पर हाइपरलिंक कैसे बनाएं? (How to make a hyperlink on a picture?)

   B. दूसरे वेबपेज का लिंक कैसे बनायें? (How to make a link to another Webpage?)

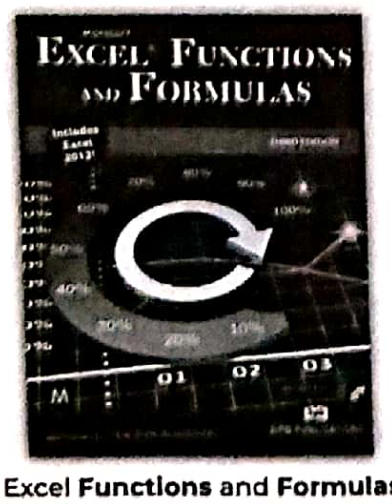
   C. एक वेबपेज पर कई कलर टेक्स्ट कैसे जोड़ें? How to add multiple color text to a webpage?)

   D. एक छवि के लिए HTML में एक नई विंडो बनाना। Creating a new window in HTML for a single image.)

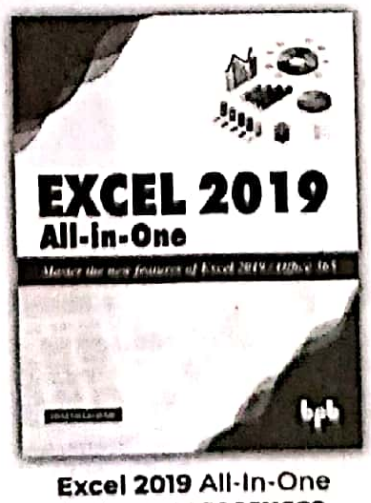
OTHER TITLES OF INTEREST



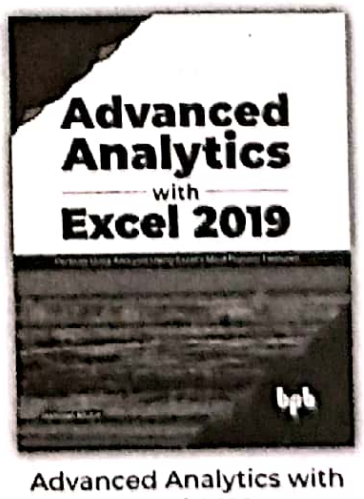
Data Analysis with **Excel**
ISBN: 9789388176675



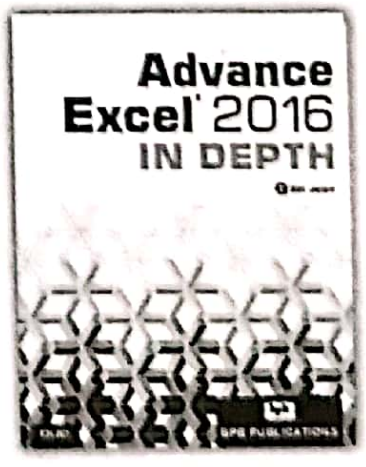
Excel **Functions** and **Formulas**
ISBN: 9788183331821



Excel 2019 All-In-One
ISBN: 9789388511582



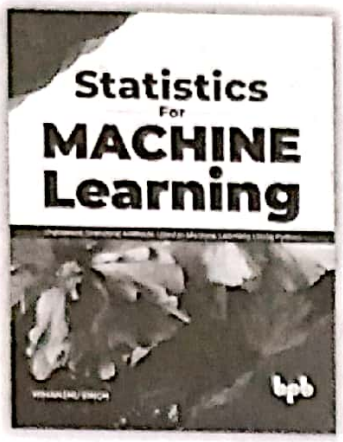
Advanced Analytics with
Excel 2019
ISBN: 9789389845808



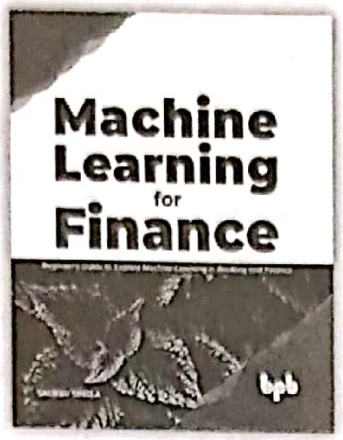
Advance **Excel 2016**
IN DEPTH
ISBN: 9788183335980

Scanned with OKEN Scanner

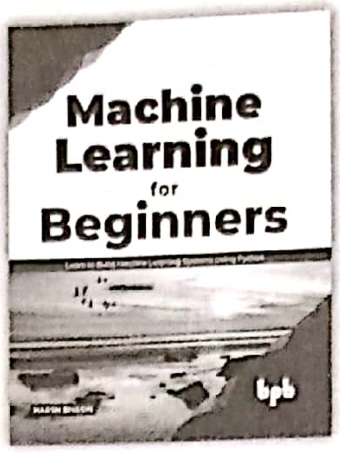
OTHER TITLES OF INTEREST



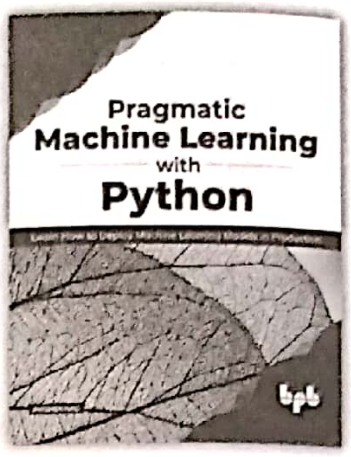
Statistics For **Machine Learning**
ISBN: 9789388511971



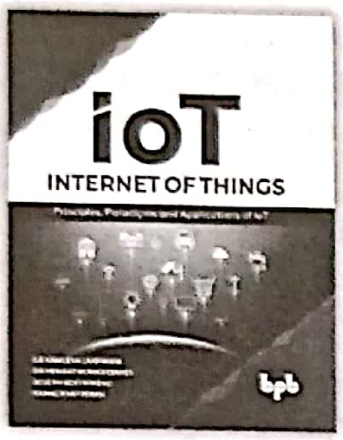
Machine Learning for **Finance**
ISBN: 9789389328622



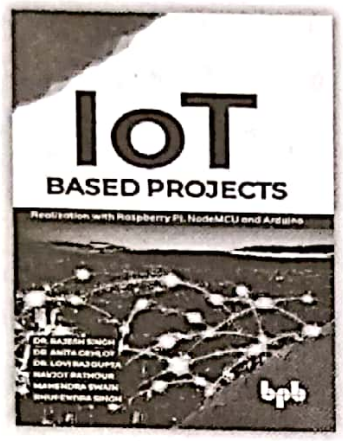
Machine Learning for **Beginners**
ISBN: 9789389845426



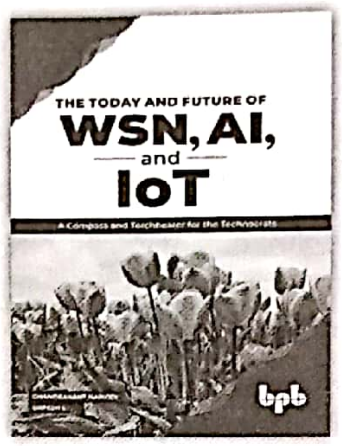
Pragmatic **Machine Learning** with **Python**
ISBN: 9789389845365



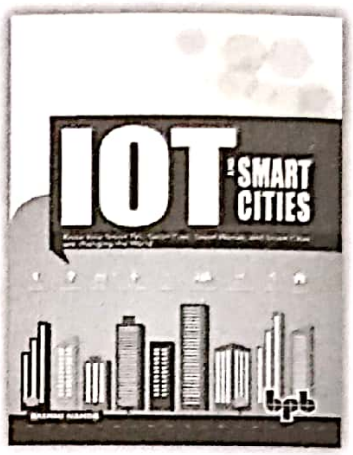
Internet of Things (IoT)
ISBN: 9789389423365



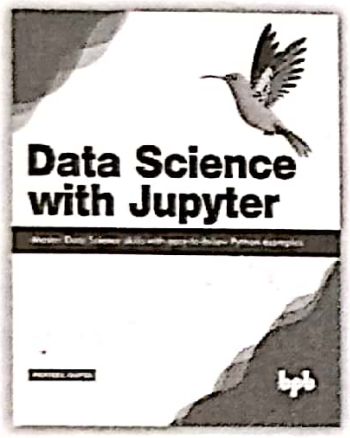
IoT Based Projects
ISBN: 9789389328523



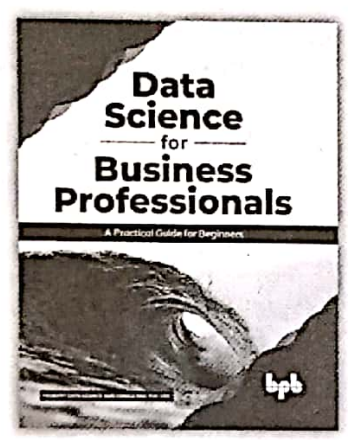
The Today & Future of **WSN, AI** and **IOT**
ISBN: 9789389845167



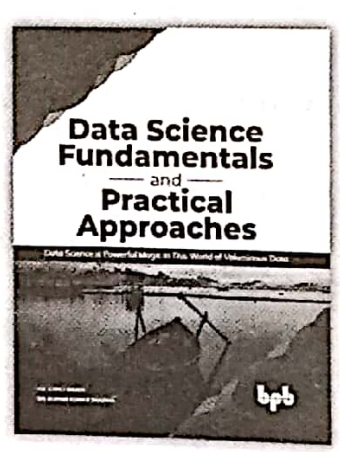
IOT and **Smart Cities**
ISBN: 9789388511322



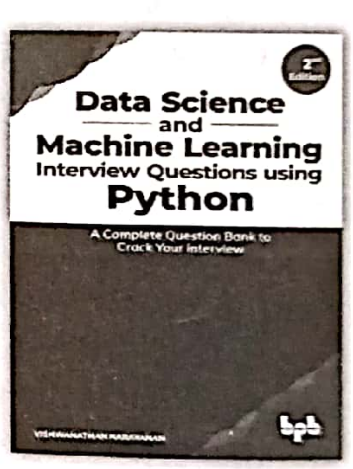
Data Science with **Jupyter**
ISBN: 9789388511377



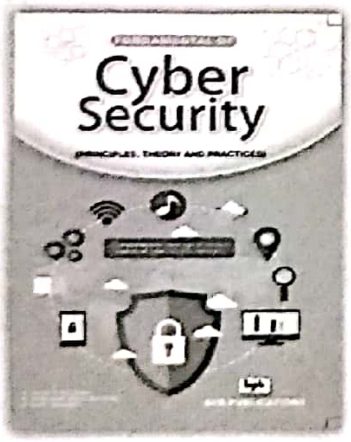
Data Science for **Business Professionals**
ISBN: 9789389423280



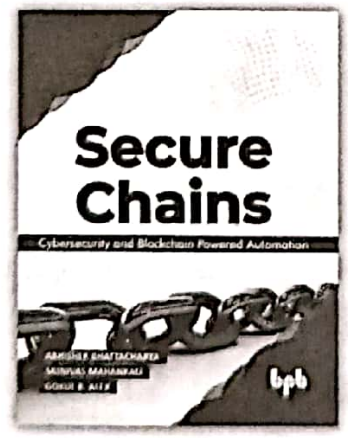
Data Science Fundamentals & **Practical Approaches**
ISBN: 9789389845662



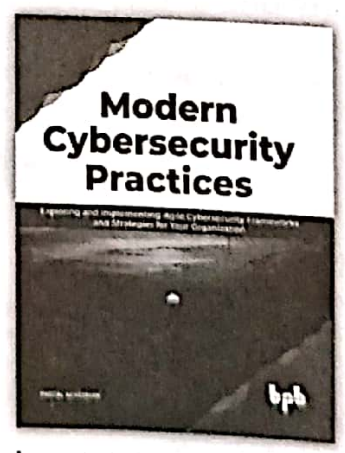
Data Science & **Machine Learning** Interview Questions using **Python**
ISBN: 9789389845785



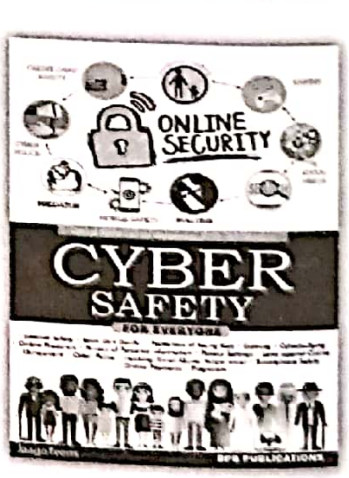
Fundamentals of **Cyber Security**
ISBN: 9789386551559



Secure Chains
ISBN: 9789389328653



Modern Cybersecurity Practices
ISBN: 9789389328257



Cyber Safety For Everyone
ISBN: 9789386551528

Available on **www.bpbonline.com** and all leading book stores.

#1 PUBLISHER OF COMPUTER BOOKS

63 YEARS OF EXCELLENCE

90 MILLION BOOKS SOLD WORLDWIDE

Scanned with OKEN Scanner

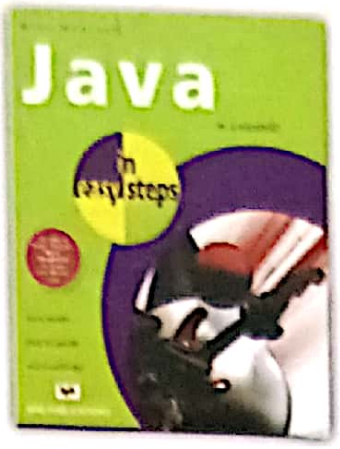
OTHER TITLES OF INTEREST



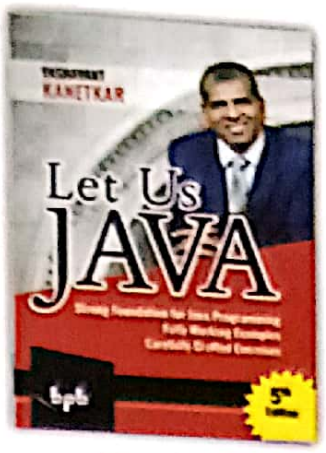
Coding for Beginners in Easy Steps
ISBN: 9789386551337



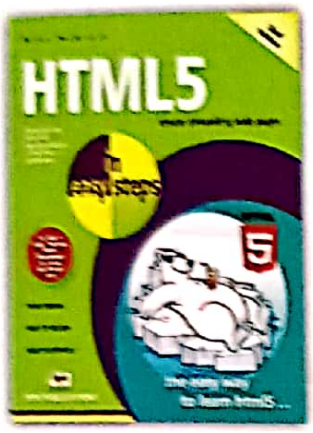
MASTERING HTML CSS & JavaScript Web Publishing
ISBN: 9788183335157



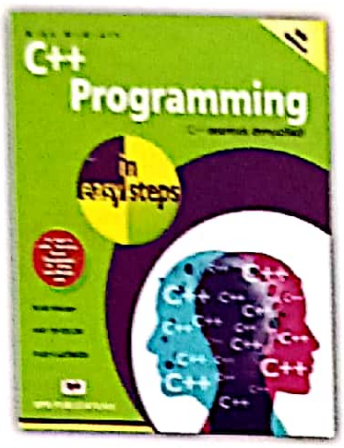
Java in Easy Steps
ISBN: 9789387284210



Let us Java
ISBN: 9789388176385



HTML 5 in Easy Steps
ISBN: 9789386551382

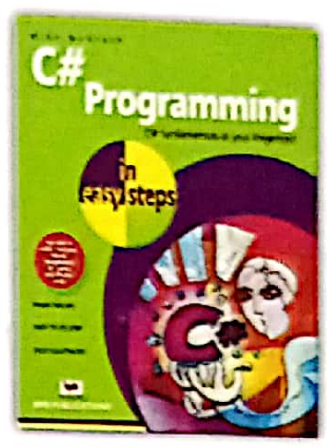


C++ Programming in Easy Steps
ISBN: 9789386551344



Let Us C
ISBN: 9789391392994

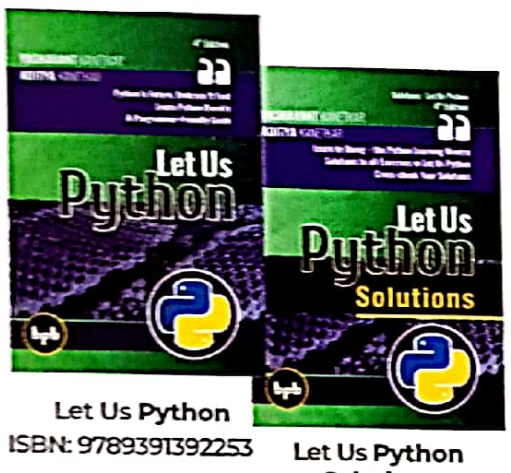
Let Us C Solutions
ISBN: 9789355510730



C# Programming in Easy Steps
ISBN: 9789386551320

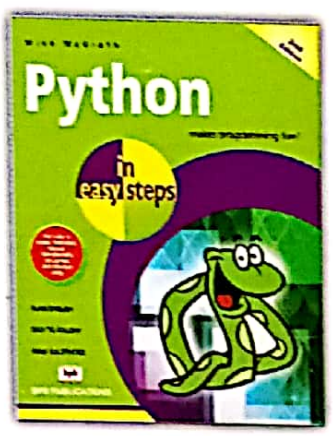


Python In-Depth
ISBN: 9789389328424

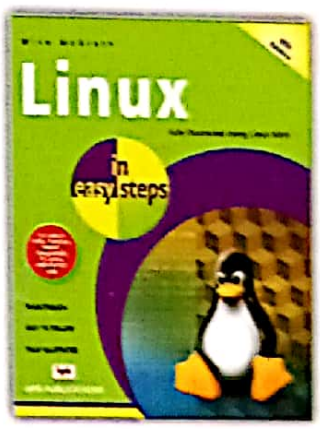


Let Us Python
ISBN: 9789391392253

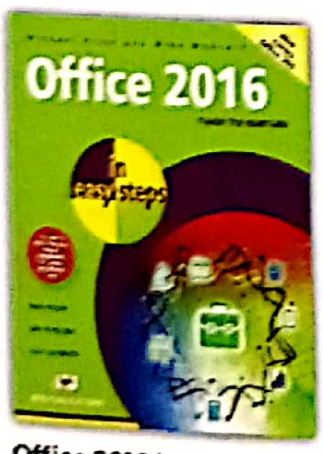
Let Us Python Solutions
ISBN: 9789391392376



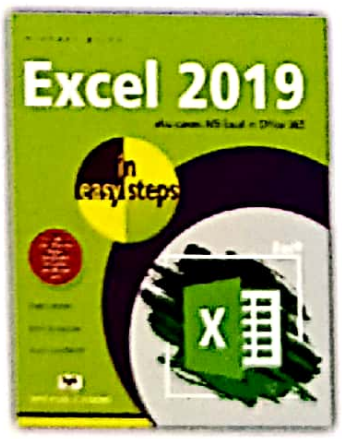
Python in Easy Steps
ISBN: 9789388176361



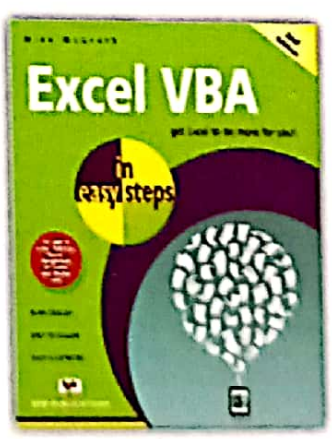
Linux in Easy Steps
ISBN: 9789388176729



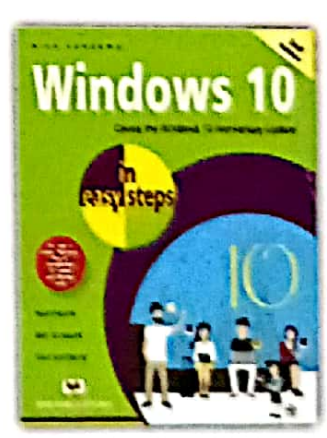
Office 2016 in Easy Steps
ISBN: 9789386551399



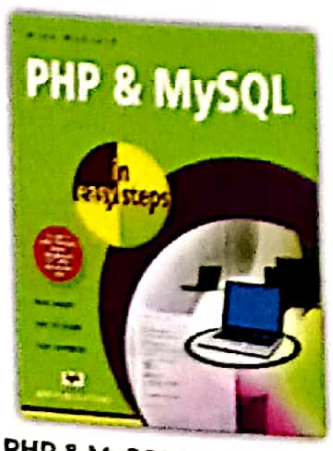
Excel 2019 in Easy Steps
ISBN: 9789388511261



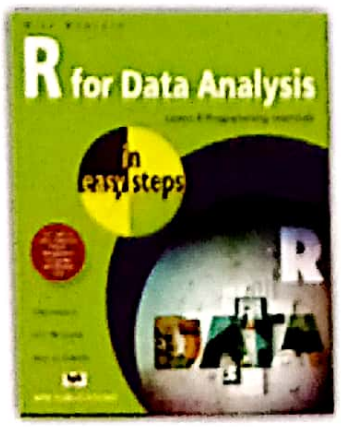
Excel VBA in Easy Steps
ISBN: 9789386551375



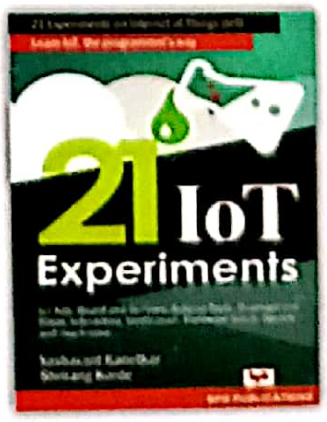
Windows 10 in Easy Steps
ISBN: 9789386551351



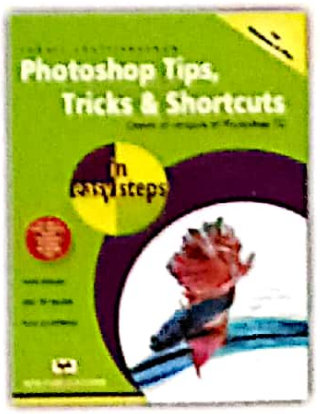
PHP & MySQL in Easy Steps
ISBN: 9789386551405



R for Data Analysis in Easy Steps
ISBN: 9781840787955



21 Internet Of Things (IOT) Experiments
ISBN: 9789386551832



Photoshop Tips, Tricks & Shortcuts in Easy Steps
ISBN: 9789386551580

Scanned with OKEN Scanner

WEB DESIGNING AND PUBLISHING

DESCRIPTION

Book provides the technological development in the field of Web Designing and Publishing. It emphasize on comprehensive knowledge of publishing content on Internet. The book covers the basics process of planning and maintenance of Web pages and Websites in Chapter 1.
Chapter 2 covers using HTML basics to create Web pages using HTML tags to display information in a Web browser. Chapter 3 covers Cascading Style Sheets and it is used by web pages to help keep information in the proper display format and it also help to define font, size, color, spacing, border and location of HTML information on a web page, to create a continuous look throughout multiple pages of a website. It also covers different types of Style Sheets and various CSS properties.

Chapter 4 teaches you to create a Website with a CSS Framework. Chapter 5 covers JavaScript and AngularJS. Along with HTML and CSS, JavaScript forms the basis of front-end web development, allowing the creating of interactive elements.

Chapter 6 discusses Photo Editor, and Chapter 7 covers Web Publishing and Browsing which discusses topics such as Web hosting, components of Web Publishing and Publishing Tools.

KEY FEATURES

- Design and create effective web pages
- Integrate graphics in web pages
- Integrate various tools and techniques like HTML, CSS, JavaScript, AngualarJS etc
- Design and edit margins using tools
- Embed the images in Web pages

WHAT WILL YOU LEARN

- Web Design, HTML Basics
- Cascading Style Sheet, CSS Framework
- JavaScript and Angular JS
- Photo Editor, Web Publishing and Browsing

WHO THIS BOOK IS FOR

- This book aims at imparting a basic level of IT literacy to computer novices and will help them to learn the practical applications of the concepts.
- Students, Programmers, researchers, and software developers who wish to learn the basics of web programming.

AUTHOR BIO

Prof. Satish Jain has obtained a B.Sc. degree from Agra University in First Division and is a gold medal winner. He obtained a B.E. (Electronics) degree from the Indian Institute of Science (I.I.Sc.), Bangalore, with Distinction. He joined Indian Airforce as a signal officer and held different technical appointments during 21 years of service. He was specially selected by the IAF to undergo Masters of Engineering course in Aerospace science at I.I.Sc., Bangalore, and M.Tech in Computer Science Engineering at IIT Kanpur. After taking retirement from the IAF, he set up the Computer Science Department in different organizations and educational institutes.

₹360
ISBN 978-93-8984-574-7

BPB PUBLICATIONS
www.bpbonline.com

Scanned with OKEN Scanner