

# Designing Modernistic Websites

## Trainer Guide

# Designing Modernistic Websites

## Trainer Guide

© 2021 Aptech Limited

All rights reserved.

No part of this book may be reproduced or copied in any form or by any means – graphic, electronic or mechanical, including photocopying, recording, taping, or storing in information retrieval system or sent or transferred without the prior written permission of copyright owner Aptech Limited.

All trademarks acknowledged.

**APTECH LIMITED**

Contact E-mail: [ov-support@onlinevarsity.com](mailto:ov-support@onlinevarsity.com)

Edition I - 2021



The book **Designing Modernistic Websites Trainer Guide** covers basic to advanced concepts of HTML5. The book begins with an explanation of basic HTML tags and attributes. It also explains the structure of Web pages using HTML. Then, it proceeds to explain the concept of Cascading Style Sheets (CSS). CSS is a technology that helps Website designers to provide a consistent formatting across large Websites by separating the content from its styles. Thus, CSS is used to control the look of the Web page by specifying the styles such as color, font, and font size for the HTML content. It can also be used to control the placement of items on a page. The book also explains JavaScript, which is a scripting language used for adding interactivity to Web pages. JavaScript allows programs in an HTML page to respond to user's actions. These responses could be validating the user's input, fetching and displaying the requested page, and so on. The book also covers explanations of jQuery and HTML5 mobile application support. jQuery is a short and fast JavaScript library that simplifies the client side scripting of HTML, animation, event handling, traversing, and developing AJAX based Web applications. The book concludes with a real-world case study based mini project.

The faculty/trainer should teach the concepts in the theory class using the slides. This Trainer's Guide will provide guidance on the flow of the module and also provide tips and additional examples wherever necessary. The trainer can ask questions to make the session interactive and also to test the understanding of the students.

The knowledge and information in this book is the result of the concentrated effort of the Design Team, which is continuously striving to bring to you the latest, the best and the most relevant subject matter in Information Technology. As a part of Aptech's quality drive, this team does intensive research and curriculum enrichment to keep it in line with industry trends and learner requirements.

# Contents

1. Introduction to the Web
2. Introduction to HTML5
3. Formatting Text Using Tags
4. Creating Hyperlinks and Anchors
5. Introduction to CSS3
6. Formatting Using Style Sheets
7. Displaying Graphics and CSS3 Animation
8. Understanding Layouts in CSS3
9. HTML Layout, Forms, and Elements
10. HTML5 Tables, Audio, and Video
11. JavaScript - I
12. JavaScript - II
13. Canvas and Web Storage in HTML5
14. HTML5 Geolocation and APIs
15. Building Mobile Web Applications

# Session 1: Introduction to the Web

## 1.1 Pre-Class Activities

Before you commence the session, you should familiarize yourself with the topics of this session in-depth. Prepare a question or two that will be a key point to relate the current session objectives.

### 1.1.1 Teaching Skills

To teach this session, you should be well versed with evolution of HTML, drawbacks of HTML4, new features of HTML5, CSS, JavaScript, and so on.

You should teach the concepts in the theory class using the images provided. For teaching in the class, you are expected to use slides and LCD projectors.

### In-Class Activities

Follow the order given here during In-Class activities.

Slide 2

Objectives

- Explain the evolution of HTML
- Explain the page structure used by HTML
- List the drawbacks in HTML 4 XHTML
- List the new features of HTML5
- Explain CSS
- Explain JavaScript
- Explain jQuery
- Explain browser support for HTML5

Designing Modernistic Websites © Apeach Limited

2

### Instructions to the Trainer(s):

Give students a brief overview of the current session through the session objectives listed in Slide 2.

## 1.2 In-Class Explanations

Slide 3

### Introduction

- Hypertext Markup Language was introduced in 1990.
- HTML5 was recommended as a standard by W3C in 1997.
- HTML5 is the next version and will be the new standard.
- Majority of the browsers support HTML5 element and Application Programming Interface (API).

Designing Modernistic Websites © Aptech Limited

3

#### Instructions to the Trainer(s):

- Using Slide 3, explain the introduction of Web and HTML. Explain the students about the evolution of HTML language.
- Tim Berners-Lee is the inventor of the World Wide Web (W3C). In 1989, Tim was working in a computing services section of CERN when he came up with the concept of sharing the data globally through a hypermedia. Since then, the use of Internet came into existence. There has been constant evolution of the technologies to the W3C.
- HyperText Markup Language (HTML) was introduced as Web language in the year 1990. It is the standard language used for creating Web pages that are accessible over the Internet.
- Mention HTML 4 was recommended as a standard by W3C in 1997. HTML5 is the next version of HTML and will be the new standard. Today, majority of the browsers support HTML5 elements and Application Programming Interfaces (APIs) supported by HTML5.

## Slides 4 and 5

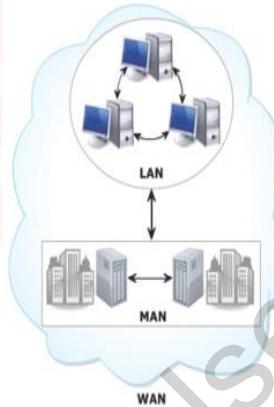
### Evolution of Computing 1-2

Computing started by using stand-alone computers to perform different computing operations.

Later organizations began to connect their computers to share data.

Different types of networks are as follows:

- Local Area Network (LAN)
- Metropolitan Area Network (MAN)
- Wide Area Network (WAN)



Designing Modernistic Websites © Aptech Limited

4

### Evolution of Computing 2-2

Network in a small geographical area

Network that covers city

Network that connects LANs and MANs across the globe

Designing Modernistic Websites © Aptech Limited

5

### Instructions to the Trainer(s):

- Using Slides 4 and 5, explain the evolution of computing to the students.

- Explain about the isolated systems used in different fields such as research and military. These systems were connected after years for processing large amount of data and to get faster results. Thus, sharing of resources was required. However, as devices were very far from each other resulted in networking of computers.
- Networks provide interconnecting between the systems that permits distributed processing of information.
- Organizations began to connect their computers and share data amongst their people.
- Mention the types of networks:
  - Local Area Network (LAN)
  - Metropolitan Area Network (MAN)
  - Wide Area Network (WAN)

## Slides 6 and 7

### Web and Internet 1-2

WAN raised the need to share data across the globe rather than within an organization.

This resulted in the evolution of Web also known as World Wide Web (WWW).

Internet is known as the largest WAN.

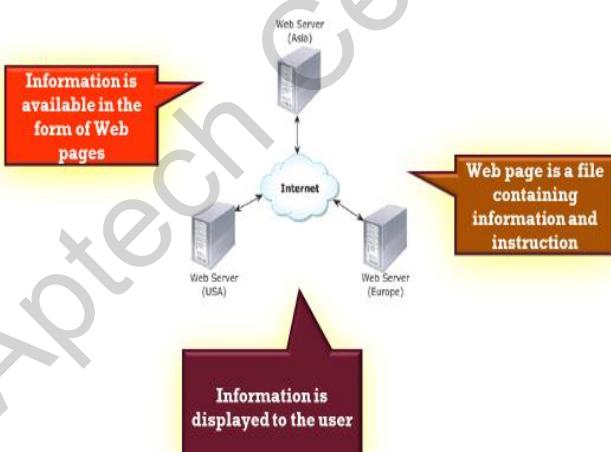
Web is a way to access information using Internet.

Multiple computers are connected to each other irrespective of geographical locations.

Designing Modernistic Websites © Aptech Limited

6

### Web and Internet 2-2



Designing Modernistic Websites © Aptech Limited

7

#### Instructions to the Trainer(s):

- Using Slides 6 and 7, explain the concepts of Web and Internet to the students.
- Mention advent of WANs raised a strong requirement to share data across the globe rather than just sharing the data within the organization. This is because organizations can share their problems, solutions, experiences, and updates along with other organizations and

customers. This would facilitate faster analysis and decision-making process. This resulted in the evolution of the Web, also referred to as World Wide Web or WWW. Internet is known as the largest WAN.

- The Web is a way to access information using the Internet that is referred to as a network of networks. Here, multiple computers are connected to each other irrespective of their geographical locations. Information is made available across the globe in the form of Web pages. The Web pages are created as a part of Web applications or Websites that are hosted on the Web servers.
- The Web page request is sent to a server which can be located anywhere by the user using browser and the server response back to the user by showing a Web page. The URL entered in the address bar of a browser is the request sent to the server and the Web page received is the response from the server.

**In-Class Question:**

**Question:** Which entities are also referred to as Web clients?

**Answer:** The browsers are also referred to as Web clients.

## Web Communication 1-2

Web pages are stored on a Web server to make them available on the Internet for the users.

Web server is a computer with high processing speed and connected to the Internet.

Web server is used to host and display the Web pages on a Web browser.

Web browser displays the Web pages using the HTTP protocol.

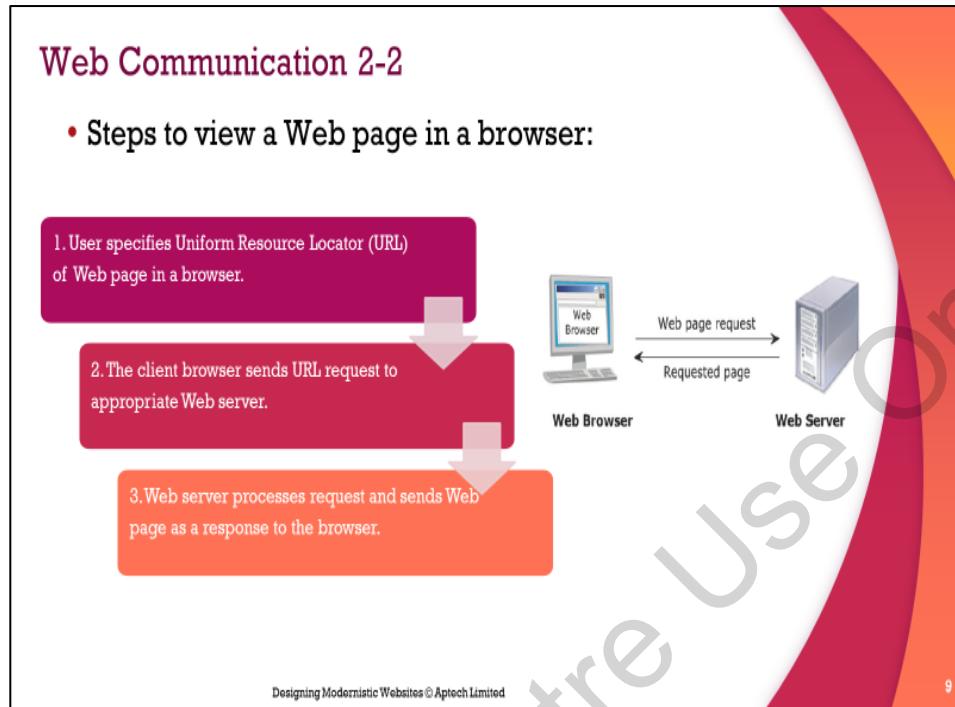
HTTP is a protocol that specifies how a Web page will be retrieved from the Web server.

Designing Modernistic Websites © Aptech Limited

8

**Instructions to the Trainer(s):**

- Using Slide 8, explain students about Web communication.
- Mention, Web pages are stored on a Web server to make them available on the Internet so that users can view them.
- A Web server is a computer with high processing speed connected to the Internet and is used to host Web pages.
- Web browsers such as Microsoft Internet Explorer or Netscape Navigator are used to interpret and display the Web pages using a protocol (set of rules).
- The most popular protocol used to view Web pages is Hypertext Transfer Protocol (HTTP).
- It is a protocol that specifies how a Web page will be retrieved from the Web Server.



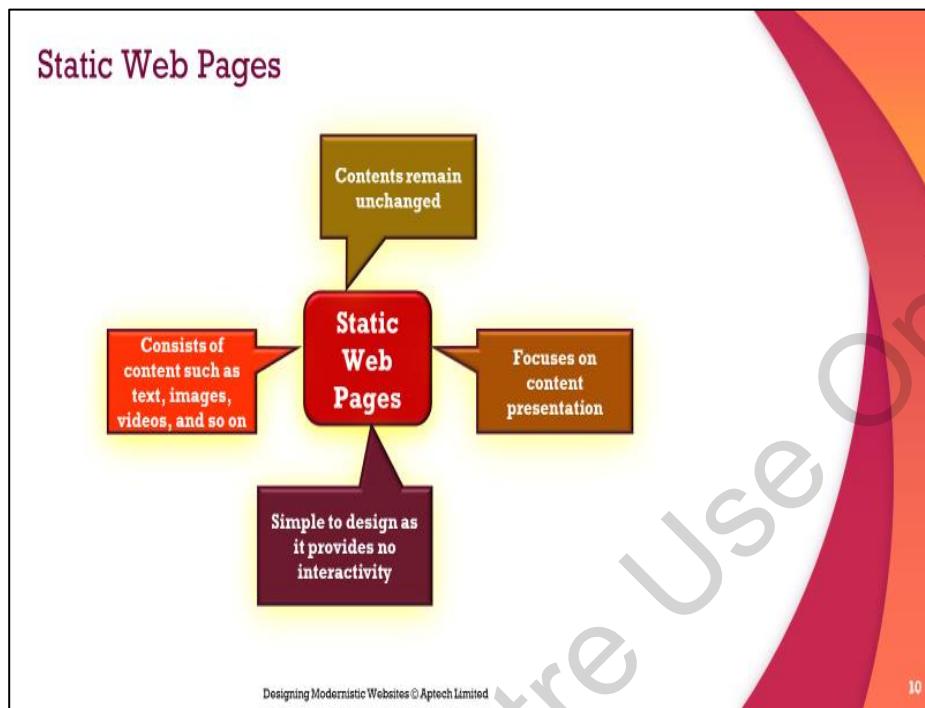
### Instructions to the Trainer(s):

- Using Slide 9, explain the process of Web communication with the help of the figure displayed on the slide.
- Tell the students that HTML is a language for describing web pages. The features of HTML language are as follows:
  - HTML stands for Hyper Text Markup Language.
  - HTML documents are also called Web pages.
  - HTML is a markup language which contains a set of markup tags. Tags are basically the angular bracket <> with keywords.
  - The tags describe document content. For example, <p> Content </p> adds the text as paragraph on the Web page.
  - HTML documents contain HTML tags and plain text.
- The purpose of a Web browser such as Google Chrome, Internet Explorer, Firefox, Safari, and so on is to read HTML documents and display them as Web pages.

### In-Class Question:

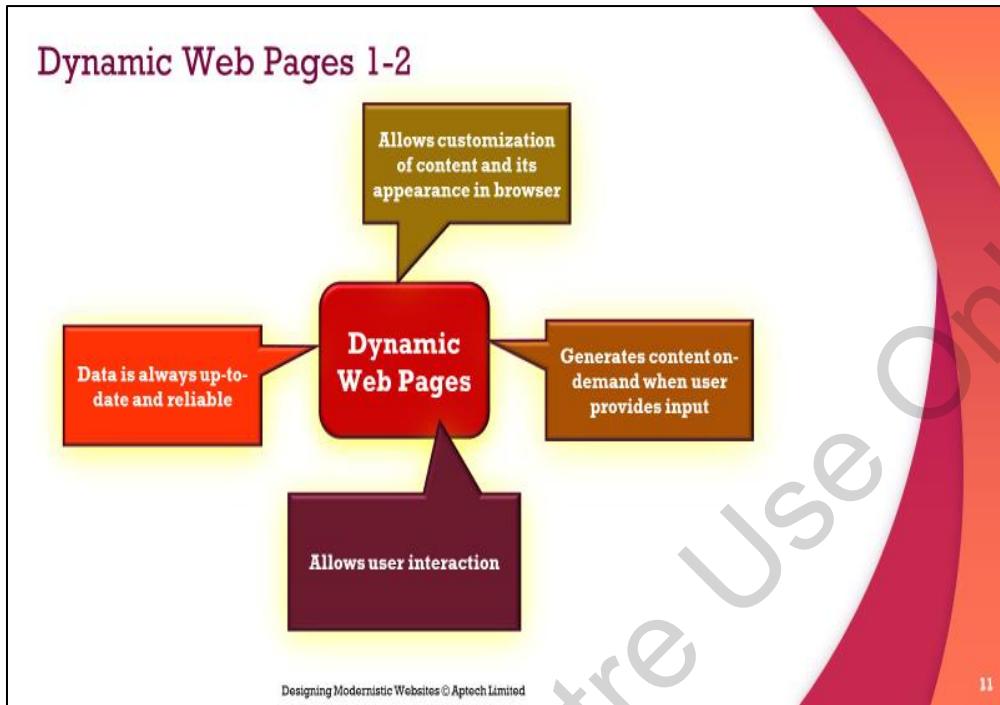
**Question:** What types of contents are observed on a Web page?

**Answer:** Text, Images, Audio, Video, Forms, Hyperlinks, and so on.



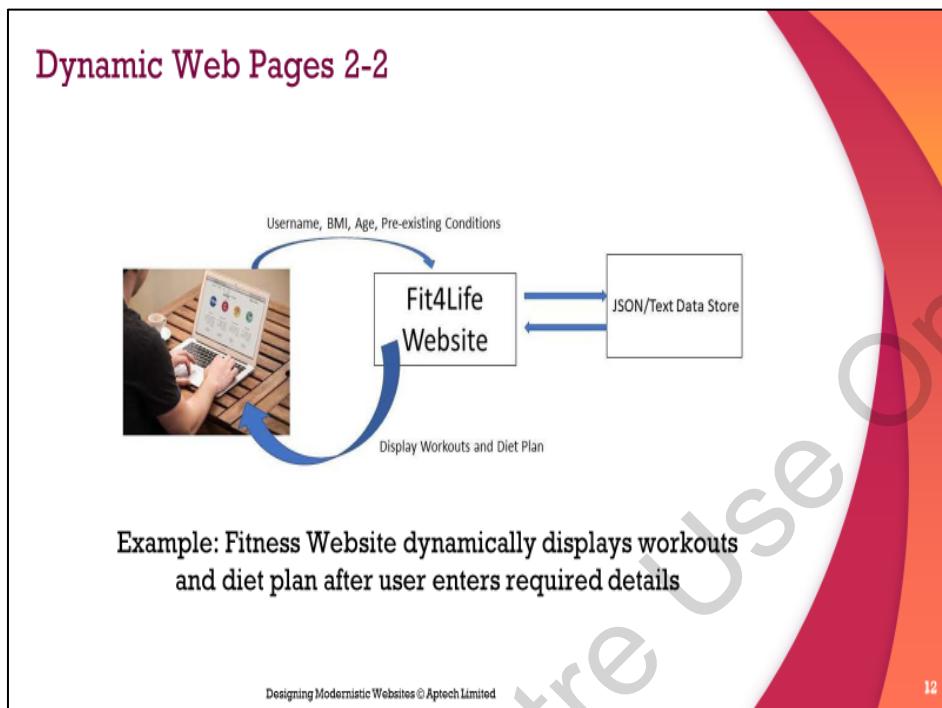
**Instructions to the Trainer(s):**

- Using Slide 10, explain static Web pages to the students.
- Static Web pages contain simple content, not interactive and does not involve any dynamic behavior on the Web pages. Therefore, such a Web page is called a static Web page, as the contents of the Web page remain unchanged. The only way to update a static Web page is to change the content manually.
- Static Web pages include contents developed using HTML tags.
- The static content includes: text, images, audio, video, hyperlink, and so on.



**Instructions to the Trainer(s):**

- Using Slide 11, explain the dynamic Web pages to the students.
- Explain the limitation of static Web page which resulted in development or evolution of dynamic Web page.
- A dynamic Web page generates content 'on-demand' when user provides certain inputs. It accepts the inputs from the user based on which it displays the content in the browser.



**Instructions to the Trainer(s):**

- Using Slide 12, explain students the working of Dynamic Web pages.
- Consider an example of an online store where the users can buy different products by selecting them online. Based on the selected products (input), a page with the total cost is displayed to the user.
- The page interacts with the user based on the action performed by the user.

## Technologies

- Technologies used for creating dynamic Websites:
  - JavaScript, a scripting language, is used for creating dynamic Web pages.
  - CSS specifies the formatting of a Web page for both static and dynamic Web pages.
  - Extensible HTML when used with JavaScript, displays the required user-defined data each time the Web page is loaded in the browser.
  - Dynamic HTML uses JavaScript and CSS to make dynamic Web pages and transform the look and feel of the Web pages.

Designing Modernistic Websites © Aptech Limited

13

### Instructions to the Trainer(s):

- Using Slide 13, explain the technologies used for developing a Website to the students.
- Explain requirements for a Website such as UI interface, database, and also the logical tier for processing data.
- Mention use of Cascading Style Sheet (CSS). CSS are style sheets that specify the formatting of a Web page for both static and dynamic Web pages. The formatting options include font, color, background, spacing, positioning, and borders. It is used in combination with JavaScript to format Web pages dynamically.
- Tell the students the use of JavaScript on Web pages. JavaScript is used to develop interactive Web pages by adding programming on the page. Similarly, explain Extensible HTML (XHTML) and Dynamic HTML (DHTML).
- XHTML is a language that combines HTML with Extensible Markup Language (XML). XML allows defining your own data in a structured format, which can be displayed in any browser. When you use XHTML with JavaScript, the required user-defined data is displayed each time the Web page is loaded in the browser.
- DHTML uses JavaScript and CSS to make dynamic Web pages. It allows you to transform the look and feel of Web pages. It allows Web pages to respond to the user's actions and enables focus on the content changes in the browser.

### History 1-2

- HTML has evolved over the years with the introduction of improved set of standards and specifications.

Designing Modernistic Websites © Aptech Limited

14

### History 2-2

- Basic rules for HTML5 are as follows:

Introduction to new features should be based on HTML, CSS, DOM, and JavaScript.

More markup to be used to replace scripting.

Must be device independent.

Need for external plug-in to be reduced.

Better error handling capabilities.

Better visibility to public the development process.

Designing Modernistic Websites © Aptech Limited

15

#### Instructions to the Trainer(s):

- Using Slides 14 and 15, explain the history of HTML.
- Mention, HTML is derived from Standard Generalized Markup language (SGML). SGML is a markup language that defines the structure of other markup languages. HTML has evolved over the years with the introduction of improved set of standards and specifications.

- HTML 1.0 was the first version of HTML introduced in 1993. At that time, there were very less people involved in designing Websites. HTML 2.0 was introduced in 1995 and included the complete HTML 1.0 specifications with additional features.
- The other versions are as follows:
  - **HTML 3.0:** HTML 3.0 specifications included new features for the Netscape Navigator browser as it became very popular. The new improvements did not work on any other browsers such as Internet Explorer. Therefore, this specification was abandoned.
  - **HTML 3.2:** Additional browser-specific features revolutionized the requirement for standardization of HTML. Therefore, the World Wide Consortium (W3C) organization was formed to specify and maintain the HTML standards. HTML 3.2 was the first specification introduced by W3C in January 1997 and was fully supported by all the Web browsers.
  - **HTML 4.0:** W3C introduced HTML 4.0 in December 1997 with the motive for facilitating support for CSS, DHTML, and JavaScript. However, HTML 4.0 prevailed for a short period and was revised, which led to HTML 4.01 specification in 1999.

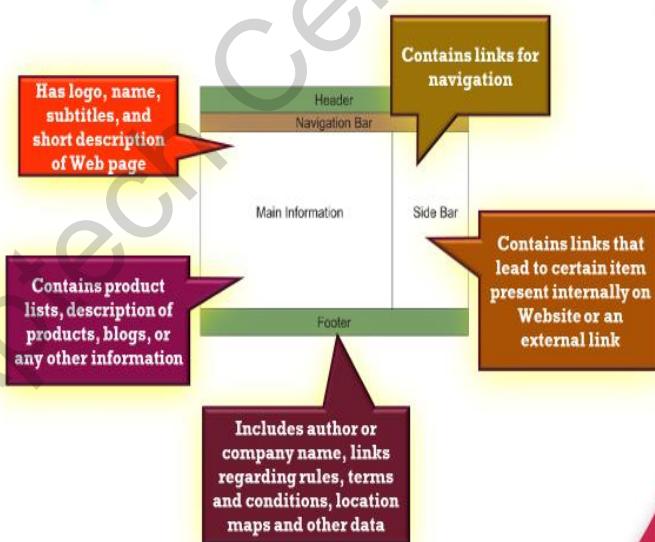
## Layout of a Page in HTML5 1-2

- HTML5 contains a head section containing the unseen elements and the body section containing the visible elements of the document.
- Earlier HTML provided different tags to build and organize the content in the body of the document.
- The <table> tag was an element often used to present the data in an organized manner.
- The <div> tag was another element used to display contents such as images, links, text, menus, forms, and so on.
- HTML5 includes new elements that identify and organize each part of the document body.

Designing Modernistic Websites © Aptech Limited

16

## Layout of a Page in HTML5 2-2



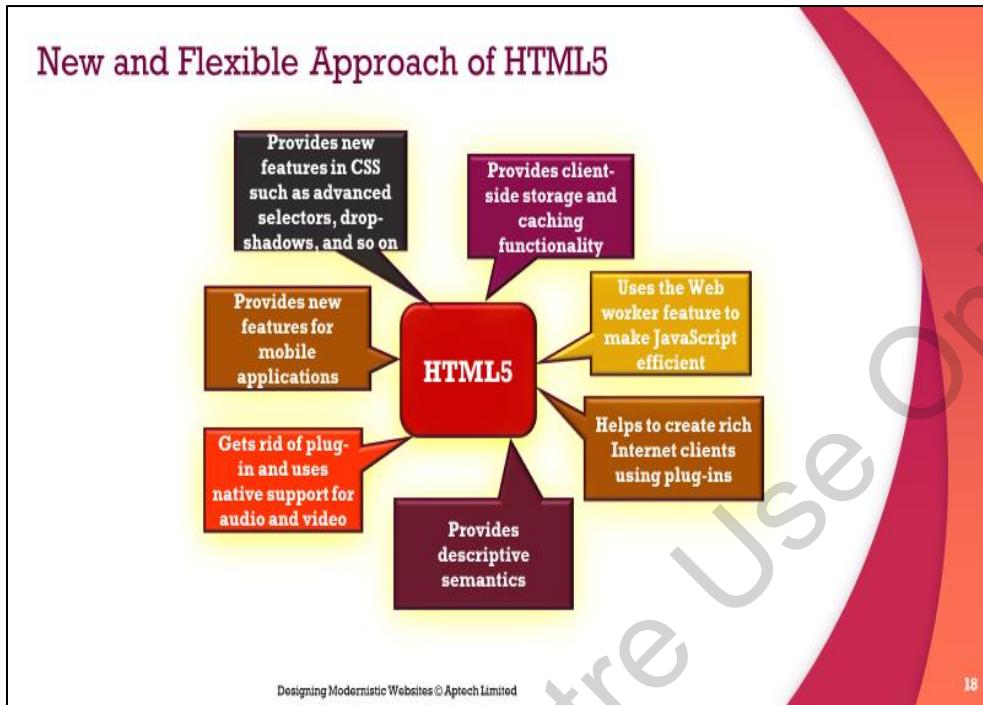
Designing Modernistic Websites © Aptech Limited

17

### Instructions to the Trainer(s):

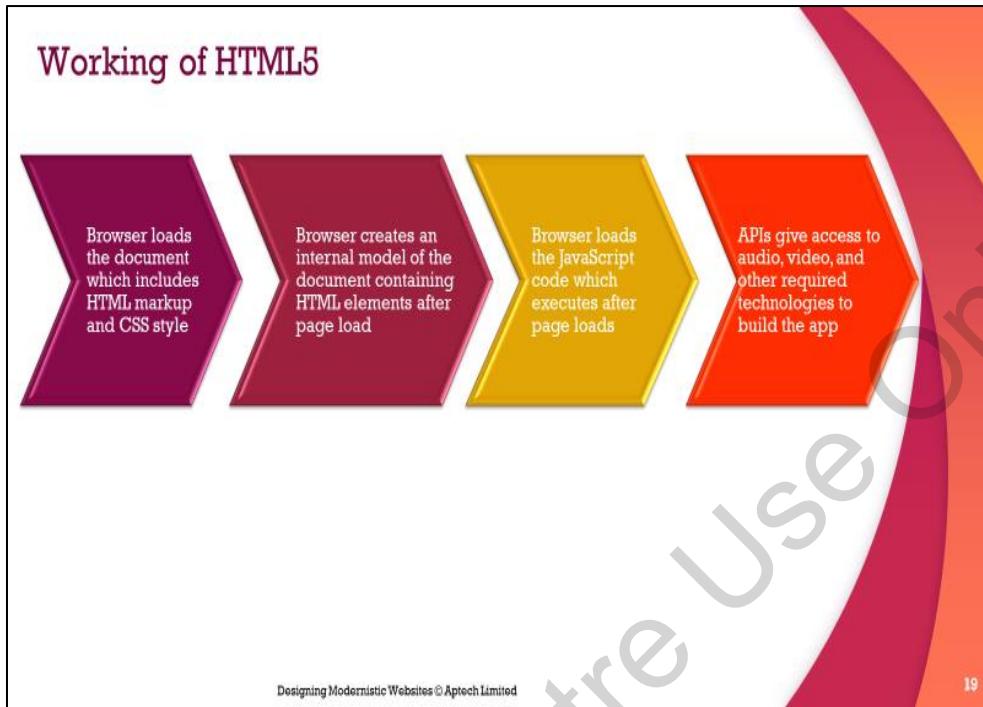
- Using Slides 16 and 17, explain the layout of a page in HTML5.
- Explain that the layout of page in HTML5 is same like that of HTML appended with the support for new tags and Application Programming Interfaces (APIs).

- Explain them the <table> and <div> tags used to layout the page in HTML. However, in HTML5 the main structure does not depend on <div> or <table> tags.
- Explain different sections on the HTML5 page used to layout the Web page. The header on the top usually has the logo, name, subtitles, and short descriptions of the Website or Web page.
- Similarly, the navigation bar includes a menu that contains links for navigation. Web users can navigate to different pages or documents using the navigation bar.
- The most relevant content is generally shown in the middle of the page. The content presented in the main information part of the layout usually has a top priority. It can have a list of products, description of products, blogs, or any other important information.
- The side bar shows a list of links that lead to certain items that may be present internally on the Website or on an external link. For example, in a blog, the last column offers a list of links that can lead to the blog entries, information about the author, and so on. These two sections are extremely flexible. Web designers can perform variety of actions, such as inserting more rows or splitting the columns, to edit the Web page as required.
- The footer at the bottom is used to represent general information about the Website. This can include the author or the company name, links regarding rules, terms and conditions, location maps, and any other additional data.



#### Instructions to the Trainer(s):

- Using Slide 18, explain the new and flexible approach in HTML5 in detail. Explain different aspects of HTML5 by comparing them with the HTML:
- For a multimedia person, HTML5 gets rid of plug-ins and uses new native support for audio and video.
- For a Web designer, HTML5 provides descriptive semantics.
- For a programmer, HTML5 helps to create rich Internet clients. These clients can be built without using plug-ins such as Flash. For this, you can use canvas and JavaScript to create better interfaces and animations. Canvas is a rectangular area on the Web page that uses JavaScript. A developer can control every single pixel in the area. The canvas element has several ways to draw paths, rectangles, filled rectangles, circles, images, and so on.
- For a client-side programmer, the Web workers is one of the features provided that can make JavaScript more efficient. Web workers is a JavaScript based API that is used to run background scripts in a Web application. This helps to mitigate the effect of the background script affecting the main process that is being executed.
- For database administrator, HTML5 has client-side storage and caching functionality.
- For a design expert, CSS in HTML5 has been improved by added features such as advanced selectors, animations, drop-shadows, and so on.
- For a mobile programmer, a lot of features are included for mobile applications. HTML5 is a family of technologies that gives whole new options for building Web pages and applications.



**Instructions to the Trainer(s):**

- Using Slide 19, explain the working of HTML5 to the students.
- HTML5 is made up of a family of technologies.
- HTML consists of markups, improved CSS with CSS3 that provides added options to style your pages. There is also JavaScript and a new set of JavaScript APIs that are available in HTML5.
- Then, explain the process followed by browser to interpret the HTML5 page as mentioned on the slide.

**New Features of HTML5**

## Features of HTML5

Web workers API is added to support background processes without affecting the main process	Web sockets API provides continuous connection between a server and a client	New form controls	<audio> and <video> element available for media playback	Content specific elements helps to structure the document	Provides local storage support	<canvas> element used for drawing
---	--	-------------------	--	---	--------------------------------	-----------------------------------

Designing Modernistic Websites © Aptech Limited

20

**Instructions to the Trainer(s):**

- Using Slide 20, explain the new features of HTML5.
- Some of the new features introduced in HTML5 are as follows:
  - The <canvas> element is used for 2D drawing.
  - New content-specific elements, such as <article>, <nav>, <header>, <footer>, <section>, and so on helps to structure the document.
  - HTML5 has local storage support.
  - The <audio> and <video> elements are available for media playback. New form controls, such as calendar, date, time, e-mail, URL, search, and so on have been provided by HTML5.
  - The Web workers API is added to support background processes without disturbing the main process. The common problems faced by Web applications are slow performance when a large set of data is processed. This is due to the fact that all the processes are executed in a single thread. Web workers help to solve this problem.
  - The Web Sockets API provides a continuous connection between a server and a client by using a specific port. Thus, the Web applications become efficient as the data can be easily exchanged between client and server without reloading the page constantly.
  - Easier access to location specific data which is made available by devices having Global Positioning System (GPS) capabilities. This improved functionality is achieved with the help of API.
  - HTML5 allows Web applications to be executed offline by storing the files and other resources required in the application cache.
  - Web application data is saved locally using Web SQL databases.

## Cascading Style Sheets (CSS)

- Works along with HTML to provide visual styles to document elements.
- Is a rule based language that specifies the formatting instructions for content in an HTML document.
- Purpose is to separate content from its formatting.
- Can define the layout and formatting of content in a separate file with a .css extension.
- Allows rules from different .css files to be merged or edited.

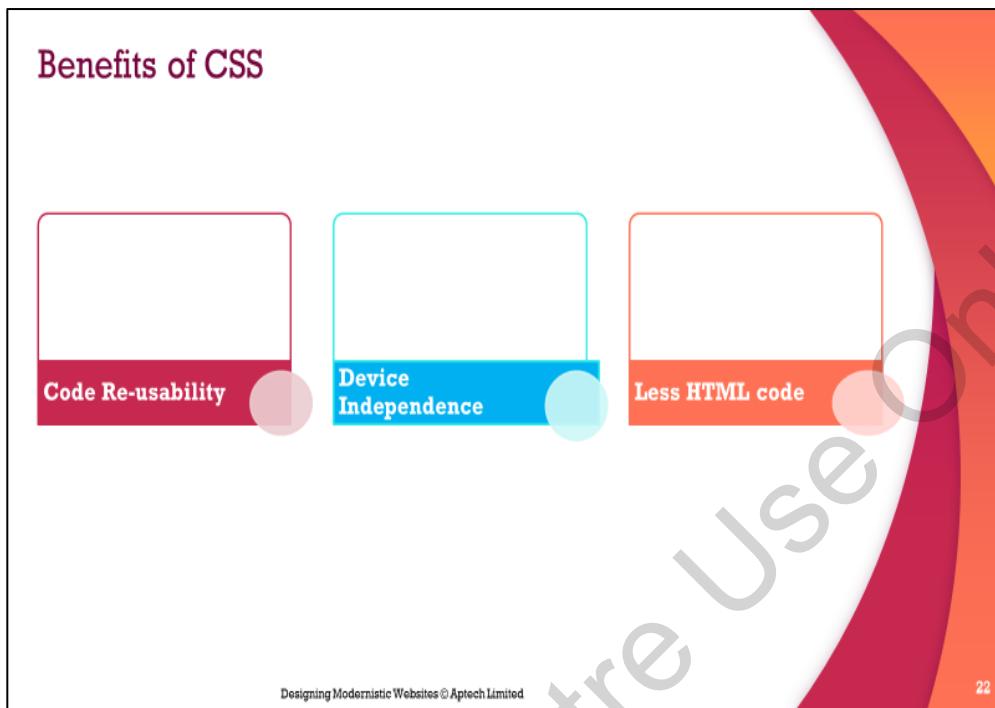
• This task of combining and matching rules from different files is referred to as **cascading**.

Designing Modernistic Websites © Aptech Limited

21

### Instructions to the Trainer(s):

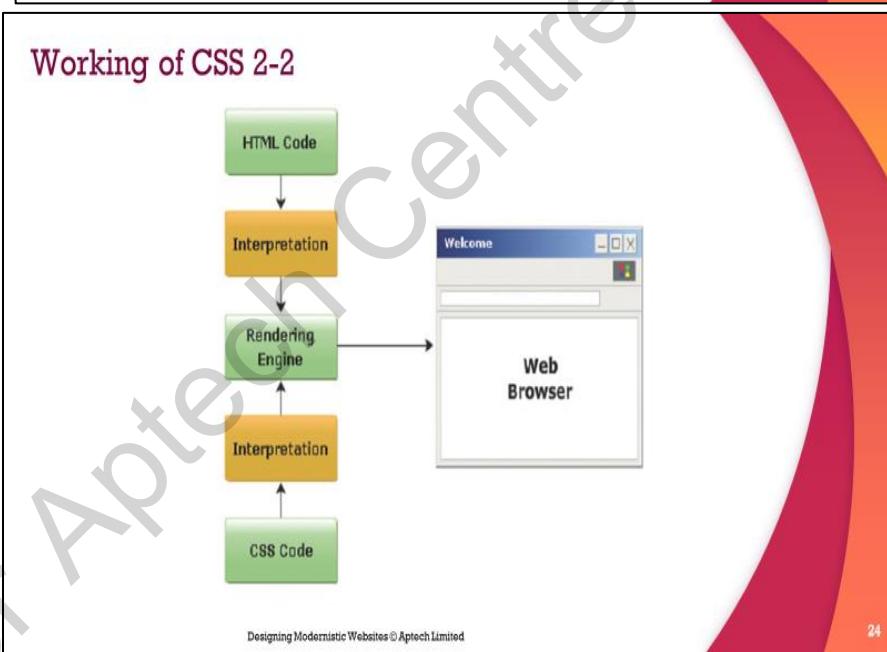
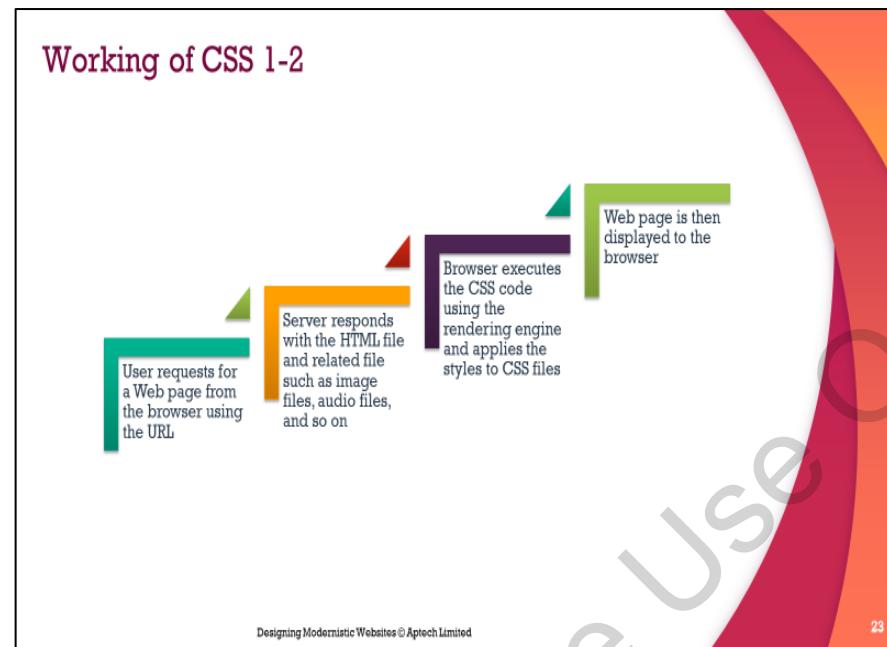
- Using Slide 21, explain the CSS as integral part of HTML5.
- A style sheet is a collection of rules that specifies the appearance of data in an HTML document. HTML is a markup language that focuses only on the layout of the content on a Web page. However, applying layouts to more than one occurrence of an HTML element in an HTML page is a tedious job.
- For example, if you want to change the text in the H2 element to bold, this has to be done manually for all the H2 elements. Such a manual task might result into human errors such as missing an occurrence of the H2 element for applying the bold format. This results in format inconsistency among the H2 elements within an HTML page. Further, the specified formatting might not have same appearance across various devices such as computers and mobiles.
- Explain CSS is a rule-based language, which specifies the formatting instructions for the content specified in an HTML page. Its purpose is to separate HTML content from its formatting so that Web page designers would not worry about the formatting and layout. This is because they can define the layout and formatting of the content in a separate file saved with an extension of .css. In the .css file, the formatting instructions for an element are referred to as a rule set.
- Each rule defines how the content specified within an element should be displayed in a Web browser. While displaying the HTML page, the browser identifies the .css file for the page and applies the rules for the specified elements. You can merge the rules from different .css files or can edit them. This task of combining and matching rules from different files is referred to as cascading.



**Instructions to the Trainer(s):**

- Using Slide 22, explain the benefits of the CSS in detail.
- Multiple HTML pages can use a CSS document. CSS provides some useful benefits that make it an ideal choice to specify the appearance of the content in an HTML page.
- These benefits are as follows:
  - **Code Reusability:** CSS saves time by specifying the formatting options of an element only once and applying them to multiple HTML pages.
  - **Less HTML Code:** CSS helps in reducing the file size of HTML documents by specifying the formatting instructions in another file.
  - **Device Independence:** CSS is designed for different devices to provide the same look and feel of the HTML page across them.

Slides 23 and 24



**Instructions to the Trainer(s):**

- Using Slides 23 and 24, explain about working of CSS.
- The CSS code can be embedded within the HTML code or link the HTML file externally to the CSS file.
- The browser will locate the style sheet irrespective of its location and will apply the style to the HTML page.
- There are certain steps involved in applying a style sheet to an HTML page. Explain these steps as mentioned on the slide.

## Slides 25 and 26

### JavaScript 1-2

#### Functionality of JavaScript

- Allows a user to create 2D drawable surface in your page without using plug-ins.
- Use Web Workers to turbo charge the JavaScript code to perform advanced computation.
- Accesses any Web service and brings back the data to the application in real time.
- Does not require any special plug-ins to play video.
- Allows to create own playback controls using JavaScript and HTML.
- Uses browser local storage and does not require browser cookies.
- Can perform full video processing in the browser.

Designing Modernistic Websites © Aptech Limited

25

### JavaScript 2-2

#### Functionality of JavaScript

- Helps Web designer to insert code snippets into the HTML page without the need for in-depth programming knowledge.
- Can be used to execute events on certain user actions.
- Can manipulate HTML elements using JavaScript.
- Can collect browser information of a Website visitor.

Designing Modernistic Websites © Aptech Limited

26

#### Instructions to the Trainer(s):

- Using Slides 25 and 26, explain the functionality of JavaScript.

- Tell the students that JavaScript helps to build dynamic Web pages by ensuring maximum user interactivity. JavaScript is a scripting language that supports object- oriented programming style. This means that it provides objects for specifying functionalities. An object has a unique identity, state, and behavior. JavaScript being a light-weight programming language is embedded directly into HTML pages.
- JavaScript is also free for use by all. It is the most popular scripting language and is supported by the major browsers.
- Some of the tasks that can be performed using JavaScript and HTML5 are as follows:
  - With HTML5 and JavaScript, you can create a 2D drawable surface in your page without using any plug-ins.
  - Use Web Workers to turbo charge the JavaScript code to perform advanced computation or make an application more responsive.
  - Access any Web service and bring that data back to your application in real time.
  - No necessity for special plug-ins to play video.
  - Create your own video playback controls using HTML and JavaScript.
  - There is no necessity to use browser cookies as the browser local storage can be used.
  - Use JavaScript to perform full video processing in the browser. You can also create special effects and even directly manipulate video pixels.
- Besides the points mentioned JavaScript can also perform the following functionalities:
  - JavaScript helps Web designer to insert code snippets into the HTML pages without the necessity to have in-depth programming knowledge.
  - JavaScript can be used to execute events on certain user actions such as on click of a HTML element, page load, and so on.
  - HTML elements can be manipulated by using JavaScript.
  - The browser information of a Website visitor can be collected by using JavaScript.

**jQuery**

- Is a JavaScript library supported on multiple browsers.
- Simplifies the designing of client-side scripting on HTML pages.
- Library is based on modular approach that allows creation of powerful and dynamic Web applications.

**Features of jQuery**

- Easy to understand syntax that helps to navigate the document.
- Event handling.
- Advanced effects and animation.
- Develop AJAX-based Web applications.

Designing Modernistic Websites © Aptech Limited

27

**Instructions to the Trainer(s):**

- Using Slide 27, explain jQuery and its features.
- Mention that jQuery is a JavaScript library which is supported on multiple browsers. It simplifies the designing of client-side scripting on HTML pages.
- The jQuery library is based on modular approach that allows the creation of powerful and dynamic Web applications. The use of jQuery on HTML pages enable developers to abstract the low-level interaction code with pre-defined library developed on top of the JavaScript. This also helps to keep the client-side script short and concise.
- Explain the features of jQuery library mentioned on the slide. While explaining AJAX- based development feature, mention AJAX stands for Asynchronous JavaScript and XML.
- jQuery is a preferred library used by developers, as it is easy to understand than JavaScript. Also, the features of jQuery enable the development of rich Web applications in a shorter period.

## Summary

- ❖ HTML5 is cooperative project between World Wide Web Consortium (W3C) and the Web Hypertext Application Technology Working Group (WHATWG).
- ❖ New features of HTML5 would include tags such as <canvas>, <article>, <nav>, <header>, <footer>, <section>, <audio>, <video> and so on.
- ❖ Some of the technologies used for creating dynamic Websites JavaScript, CSS, XHTML, and DHTML.
- ❖ A Cascading Style Sheet (CSS) is a rule based language, which specifies the formatting instructions for the content specified in an HTML page.
- ❖ JavaScript is a scripting language that allows you to build dynamic Web pages by ensuring maximum user interactivity.
- ❖ jQuery is a JavaScript library that simplifies the design of client-side scripting on HTML pages.

28

### Instructions to the Trainer(s):

- Show students Slide 28.
- Summarize the session by reading out each point on the slide.

## Session 2: Introduction to HTML5

### 2.1 Pre-Class Activities

Before you commence the session, you should familiarize yourself with the topics of this session in-depth. Prepare a question or two that will be a key point to relate the current session objectives.

#### 2.1.1 Teaching Skills

To teach this session, you should be well versed with basic tags in HTML, data types, attributes, and entities of HTML5.

You should teach the concepts in the theory class using the images provided. For teaching in the class, you are expected to use slides and LCD projectors.

### In-Class Activities

Follow the order given here during In-Class activities.

Slide 2

The slide has a red and orange abstract background with overlapping circles. The title 'Objectives' is at the top left. Below it is a bulleted list of seven items:

- Explain the elements constituting an HTML tag
- Describe DOCTYPE declarations
- Explain the basic tags in HTML
- List the different data types, attributes, and entities of HTML5
- Describe container and standalone tags
- Explain the role of HTML5 in Mobile devices

At the bottom left is the text 'Designing Modernistic Websites © Apiech Limited'. At the bottom right is the number '2'.

#### Instructions to the Trainer(s):

Give students a brief overview of the current session through the session objectives listed in Slide 2.

## 2.2 In-Class Explanations

Slides 3 and 4

### Elements 1-2

- An element organizes the content in a Web page hierarchically, which forms the basic HTML structure.

It consists of tags, attributes, and content. Tags denote the start and end of an HTML element.

A start tag includes an opening angular bracket (<) followed by the element name, zero or more space separated attributes, and a closing angular bracket (>).

Attributes are name/value pairs that describe the element and content format.

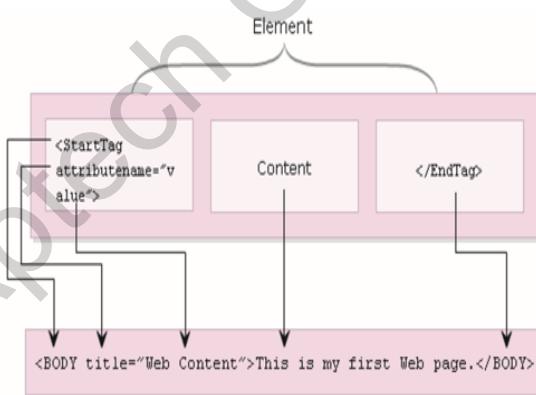
An end tag is written exactly as the start tag, but the forward slash (/) precedes the element name.

Designing Modernistic Websites © Aptech Limited

3

### Elements 2-2

- Following figure shows an element in HTML tag.



Designing Modernistic Websites © Aptech Limited

4

#### Instructions to the Trainer(s):

- Using Slides 3 and 4, explain the elements of a Web page.
- Mention that an HTML page organizes the content into the hierarchical tree structure. The tree structure contains HTML elements. The element organizes the content in a Web page

hierarchically, which forms the basic HTML structure. Each element consists of tags, attributes, and content.

- Tags denote the start and end of an HTML element. A start tag includes an opening angular bracket (<) followed by the element name, zero or more space separated attributes, and a closing angular bracket (>).
- Attributes are name/value pairs that describe the element and content format. An end tag is written exactly as the start tag, but the forward slash (/) precedes the element name.
- Further, explain syntax of the tags to the students in detail using the figure provided on the slide.

## DOCTYPE

- Informs the browser the HTML version number of your document.
- It is the first declaration in the HTML5 document before any other HTML code is written.
- Allows a browser to be more precise in the way it interprets and renders your pages.

The new HTML5 DOCTYPE is as follows:

```
<!DOCTYPE html>
```

- It is the new syntax of HTML5 as well as for all future versions of HTML.
- This DOCTYPE is compatible with older browsers too.

### Instructions to the Trainer(s):

- Using Slide 5, explain the concept of DOCTYPE declaration.
- Mention DOCTYPE element informs the browser the HTML version number of your document. It is the first declaration in the HTML5 document before any other HTML code is written. By using a DOCTYPE, the browser is able to be more precise, in the way it interprets and renders your pages. It is highly recommended to use a DOCTYPE at the beginning of all HTML documents.
- The new HTML5 DOCTYPE declaration is as follows:  

```
<!DOCTYPE html>
```
- Not only this syntax is valid for the DOCTYPE for HTML5, but it is also the DOCTYPE for all future versions of HTML. This DOCTYPE is compatible even with the older browsers.
- Mention that it is not an html tag and also not case sensitive. HTML5 is not based on SGML and therefore does not require a reference to a DTD which is required in HTML 4.01, XHTML 1.0, and XHTML 1.1.
- A Document Type Definition (DTD) defines the legal building blocks of an XML document. It defines the document structure with a list of legal elements and attributes.

### In-Class Question:

**Question:** What is the use of doctype declaration in a Web page?

**Answer:** The browser is able to be more precise, in the way it interprets and renders Web page if doctype declaration is present.

## Basic Tags 1-6

- An HTML document is made up of different elements, tags, and attributes, which specify content and its format.
- HTML is both a structural and presentational markup language.
- Structural markup specifies structure of content, while presentational markup specifies format.
- An HTML page is saved with .html extension.
- Basic structure of an HTML document mainly consists of seven basic elements:



### ➤ HTML

- The `html` element is the root element that marks the beginning of an HTML document.
- It contains start and end tag in the form of `<html>` and `</html>` respectively.
- It is the largest container element as it contains various other elements.

Designing Modernistic Websites © Aptech Limited

6

## Basic Tags 2-6

### ➤ HEAD

- The `head` element provides information about the Web page such as keywords and language used.
- Keywords are important terms existing in a Web page used by the search engines to identify the Web page with respect to search criterion.

### ➤ TITLE

- The `title` element allows you to specify title of the Web page under `<title>` and `</title>` tags.
- The `title` is displayed on the title bar of the Web browser. The `title` element is included within the `head` element.

Designing Modernistic Websites © Aptech Limited

7

## Basic Tags 3-6

### ➤ META

- The meta tag is used for displaying information about the data.
- In HTML5, the content meta tag which was used for specifying the charset or character encoding has been simplified.
- The new <meta> tag is as follows:  
`<meta charset="utf-8" />`
- UTF-8 is the most commonly used character coding that supports many alphabets.
- There are several other attributes associated with the meta tag that can be used to declare general information about the page.
- This information is not displayed in the browser.
- Meta tags provide search engines, browsers, and Web services with the information that is required to preview or acquire a summary of the relevant data of your document.

Designing Modernistic Websites © Aptech Limited

8

## Basic Tags 4-6

### ➤ LINK

- The <link> tag is used to define the association between a document and an external resource.
- It is used to link style sheets. Its type attribute is used to specify the type of link such as 'text/css' which points out to a style sheet.

`<link type="text/css" rel="stylesheet" href="first.css">`

- The type attribute is not included in HTML5.
- The reason is that CSS has been declared as default and standard style for HTML5. So, the new link is as follows:

`<link rel="stylesheet" href="first.css">`

Designing Modernistic Websites © Aptech Limited

9

## Basic Tags 5-6

### ➤ SCRIPT

- With HTML5, JavaScript is now the standard and default scripting language.
- The type attribute tag can be removed from the script tags.
- The new script tag is as follows:

Following example shows use of the script tag:

```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>HTML Webinar</title>
<link rel="stylesheet" href="first.css">
<script src="first.js"></script>
</head>
</html>
```

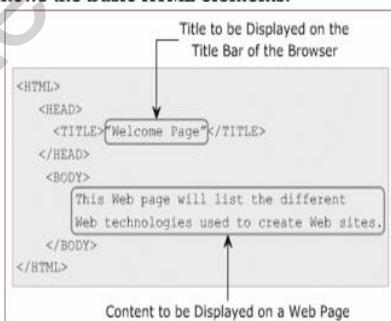
Designing Modernistic Websites © Aptech Limited

10

## Basic Tags 6-6

### ➤ BODY

- The body element enables you to add content on the Web page specified under the <body> and </body> tags.
- Content can include text, hyperlinks, and images. You can display the content using various formatting options such as alignment, color, and background.
- Following figure shows the basic HTML elements:



Designing Modernistic Websites © Aptech Limited

11

### **Instructions to the Trainer(s):**

- Using Slides 6 to 11, explain the basic requirement of the html file. Explain the basic tag in HTML.
- An HTML document is made up of different elements or tags and attributes which specify the content and its format. Therefore, HTML is both a structural and presentational markup language. Structural markup specifies the structure of the content, while the presentational markup specifies the format.
- An HTML page is saved with the .html extension. The basic structure of an HTML document mainly consists of seven basic elements.
- Explain HTML element is the root element that marks the beginning of an HTML document. It contains the start and end tag in the form of <HTML> and </HTML> respectively. It is the largest container element as it contains various other elements.
- The DOCTYPE declaration is done before the <HTML> tag in the HTML page.
- Then, explain <HEAD> element and <Title> element. Mention that the <head> element includes other elements such as title, scripts, styles, and meta information.
- Now, explain the <TITLE> element. It allows you to specify the title of the Web page under the <TITLE> and </TITLE> tags and is displayed on the title bar of the Web browser.
- Explain meta tag is used for displaying information about the data. In HTML5, the content meta tag can be used for specifying the charset or character encoding used on the Web page.
- Tell them that UTF-8 is also being promoted as the new standard for characters. UTF-8 encodes each Unicode character as a variable number of 1 to 4 octets, where the number of octets depends on the integer value assigned to the Unicode character. It is an efficient encoding of Unicode documents that use mostly US-ASCII characters because it represents each character in the range U+0000 through U+007F as a single octet.
- There are several other attributes associated with the meta tag that can be used to declare general information about the page. This information is not displayed in the browser. Meta tags provide search engines, browsers, and Web services, the information that is required to preview or acquire a summary of the relevant data of your document.
- Explain <link> tag that is used to link style sheets in the HTML page. The type attribute is used to specify the type of link such as 'text/css' which points out to a style sheet. The type attribute is not included in HTML5, because CSS has been declared as the default and standard style for HTML5.
- So, the new link can be written as:  

```
<link rel="stylesheet" href="first.css">
```
- Along with this, mention the size attribute is new in HTML5 for <link> tag. The size attribute specifies the sizes of icons for visual media and is used if rel="icon".
- Further, explain the <script> element to the students. Mention that JavaScript is now the standard and default scripting language. Hence, you can remove the type attribute from the script tag.
- Thus, the new script tag is as follows:

```
<script src="first.js"></script>
```

- The `async` attribute can be used with the `<link>` tag. The `async` attribute is a boolean attribute and it specifies that the script will be executed asynchronously as soon as it is available.
- If the `async` attribute is present, then the script will be executed asynchronously, as soon as it is available. If the `async` attribute is not present, then the script is fetched and executed immediately, before the user agent continues parsing the page.
- Finally, explain the `<body>` element to the students. Tell them that this is the most important tag as only contents contained within the `<body>` tag are displayed on the Web page. In other words, it enables you to add content on the Web page.
- The content can include text, hyperlinks, and images. You can display the content using various formatting options such as alignment, color, and background.

**In-Class Question:**

**Question:** Which tag contains the `<meta>` tag in an HTML page?

**Answer:** `<head>`

## Slides 12 and 13

### Data Types 1-2

- A data type specifies type of value assigned to attributes and type of content that is to be displayed on the Web page.
- Data types help in identifying type of formatting such as color and length of data.
- Following table describes different types of content:

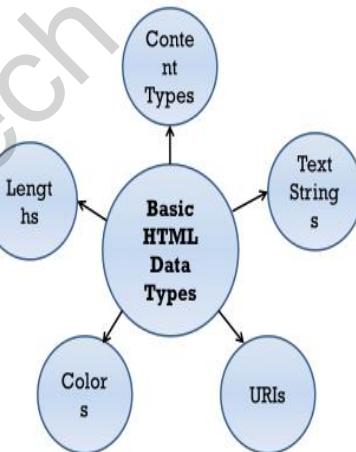
Data Type	Description
Text Strings	Specifies textual content, which is readable by the user.
Uniform Resource Identifiers (URIs)	Specifies the location of Web pages or network files.
Colors	Specifies the color to be applied to the content on the Web page.
Lengths	Specifies the spacing among HTML elements. Length values can be in Pixels, Length, or MultiLength. Pixels refer to the smallest dot on the screen.
Content Types	Specifies the type of content to be displayed on a Web page. Content types include 'text/html' for displaying text, 'image/gif' for displaying image of a .gif format, 'video/mpg' for displaying a video file of .mpg format.

Designing Modernistic Websites © Aptech Limited

12

### Data Types 2-2

- Following figure shows different data types:



Designing Modernistic Websites © Aptech Limited

13

### Instructions to the Trainer(s):

- Using Slides 12 and 13, explain different data types of HTML.

- Data types specify the type of value that can be assigned to the attributes. Different types of value or content include texts, images, hyperlinks, video, and audio. Data types also identify the length of the data that can be stored in it.
- Explain different data types using figure displayed on Slide 13.
- The important basic HTML data types are as follows:
  - **Text Strings:** Specifies textual content which is readable by the user.
  - **Uniform Resource Identifiers (URIs):** Specifies the location of Web pages or network files.
  - **Colors:** Specifies the color to be applied to the content on the Web page.
  - **Lengths:** Specifies the spacing among HTML elements. Length values can be in pixels, length, or multilength. Pixels refer to the smallest dot on the screen. Length is specified as a percentage value of pixels or available space on the screen. Multilength can be specified as pixel or percentage.
  - **Content:** Specifies the type of content to be displayed on a Web page. Examples of content types include ‘text/html’ for displaying text using HTML format, ‘image/gif’ for displaying image of a .gif format and ‘video/mpg’ for displaying a video file of .mpg format.

## Attributes

- HTML attributes help to provide some meaning and context to the elements.
- Following table describes some of the global attributes used in HTML5 elements.

Attribute	Description
class	Specifies class names for an element.
contextmenu	Specifies the context menu for an element.
dir	Specifies the direction of the text present for the content.
draggable	Specifies the draggable function of an element.
dropzone	Specifies whether the data when dragged is copied, moved, or linked, when dropped.
style	Specifies the inline CSS style for an element.
title	Specifies additional information about the element.

Designing Modernistic Websites © Aptech Limited

14

**Instructions to the Trainer(s):**

- Using Slide 14, explain the some of the global attributes of HTML5. The global attributes can be associated with any element or tag.
- Mention HTML attributes that helps to provide some meaning and context to the elements.
- Mention the style attribute is used for providing the inline style to the element. For example, to change the text color of a paragraph, the style attribute can be used, `<p style="color:red;">This is a paragraph.</p>`
- While the class attribute is used to specify the id for the elements for which styles must be applied.

## HTML Entities

- Entities are special characters that are reserved in HTML.
- These entities can be displayed on a HTML5 Website using the following syntax:

**Syntax:**  
`&entity_name; or &#entity_number;`

- Following table shows some of the commonly used HTML entities:

Output	Description	Entity Name	Entity Number
nbsp;	non-breaking space	&nbsp;	&#160;
<	less than	&lt;	&#60;
>	greater than	&gt;	&#62;
&	ampersand	&amp;	&#38;
€	euro	&euro;	&#8364;
©	copyright	&copy;	&#169;

Designing Modernistic Websites © Aptech Limited

15

**Instructions to the Trainer(s):**

- Using Slide 15, explain the HTML entities to the students.
- Mention entities are special characters that are reserved in HTML.
- These entities can be displayed on a HTML5 Website using the following syntax:  
`&entity_name; or &#entity_number;`
- The advantage of using an entity name, instead of a number, is that the name is easier to remember.
- The disadvantage is that browsers may not support all entity names, instead they may support numbers.

## Container and Standalone Tags

There are two types of HTML elements namely, container and standalone elements.

A container element includes the start tag, contents, sub-elements, and end tag.

All the basic HTML elements are container elements.

A standalone element consists of the start tag and attributes followed by the end tag as /> without any content.

### Instructions to the Trainer(s):

- Using Slide 16, explain container and standalone tags.
- Mention there are two types of HTML elements namely, container and standalone elements.
- A container element includes the start tag, contents, sub-elements, and end tag. All the basic HTML elements are container elements.
- A standalone element consists of the start tag and attributes followed by the end tag as /> without any content.
- Provide examples such as <link>, <meta>, and <br> tag are standalone tags.
- Similarly, tags such as <body>, <head>, and so on are container tags.

## HTML5 and Mobile Devices

HTML5 helps to create better and richer mobile applications by using APIs that support advanced Web application features for mobile browsers.

New age smartphones with Apple iOS and Google Android as operating systems support HTML5 compliant browsers.

HTML5 tries to integrate all the features to deploy mobile applications that would be compatible in all the platforms.

HTML5 provides features such as drag-and-drop functionality, video embedding in an application, and even offline capabilities.

As HTML5 is compatible with most mobile operating systems, up to 30% of the cost for development for different operating systems is saved.

Also, there is a reduced dependency in third-party components, thus reducing the licensing costs.

All the required components will be readily available through the browser in HTML5.

Designing Modernistic Websites © Aptech Limited

17

### Instructions to the Trainer(s):

- Using Slide 17, explain the use of HTML5 in mobile devices in detail.
- Mention HTML5 has helped to create better and richer mobile applications. For this, APIs are used in HTML5. These APIs support advanced Web application features for mobile browsers.
- HTML5 is not supported by older mobile devices. New age smartphones with Apple iOS and Google Android as operating systems support HTML5 compliant browsers. Even Microsoft Windows 7 for Mobile will have a newly developed browser to support HTML5 developed Websites and applications.
- However, due to various mobile platforms available on mobile devices, development of mobile applications is difficult. HTML5 has tried to integrate all the features to deploy mobile applications that would be compatible in all the platforms. HTML5 provides features such as drag-and-drop functionality, video embedding in an application, and even offline capabilities.
- As HTML5 is compatible with most mobile operating systems, up to 30% of the cost for development of different operating systems is saved. Also, there is a reduced dependency in third-party components, thus reducing the licensing costs. All the required components will be readily available through the browser in HTML5.

## Benefits of HTML5 for Mobile Development

- HTML5 has included APIs, hence additional plug-ins are not required for mobile browsers.
- Mobile development is easier as knowledge of only HTML5, CSS, and JavaScript is majorly required.
- There is a rising growth of HTML5 for mobile applications due to its enhanced compatibility.
- HTML5 is compatible with most operating system platforms.
- The HTML5 based mobile applications can run on browsers of Android, iOS, Blackberry, Windows Phone, and other mobile operating systems.
- The development cost for creating applications in HTML5 is low.
- Applications based on location and maps will have greater support in HTML5.
- Third-party programs are not required in HTML5.

Designing Modernistic Websites © Aptech Limited

18

### Instructions to the Trainer(s):

- Using Slide 18, explain the benefits of using HTML5 for development of mobile device applications.
- Mention HTML5 as a standard has a long way to go. The fragmented support for HTML5 in different browsers drives the user experience to an even lower common denominator.
- HTML5 apps also have the ability for offline access and usage via the application cache, which means working without a network connection is now possible.
- One of the biggest benefits to IT organizations developing mobile applications in HTML5 is the ability to deploy those apps and updates directly to the user community via the browser. No third party or extra step is required for distribution.

## Summary

- ❖ An element organizes the content in a Web page hierarchically, which forms the basic HTML structure.
- ❖ The DOCTYPE tells the browser the type of your document.
- ❖ A data type specifies the type of value assigned to the attributes and the type of content that is to be displayed on the Web page.
- ❖ Entities are special characters that are reserved in HTML.
- ❖ A container element includes the start tag, contents, sub-elements, and the end tag.
- ❖ A standalone element consists of the start tag and attributes followed by the end tag as /> without any content.
- ❖ HTML5 provides features such as drag-and-drop functionality, video embedding in an application, and even offline capabilities for mobile devices.

19

**Instructions to the Trainer(s):**

- Show students Slide 19.
- Summarize the session by reading out each point on the Slide.

## Session 3: Formatting Text Using Tags

### 3.1 Pre-Class Activities

Before you commence the session, you should familiarize yourself with the topics of this session in-depth. Prepare a question or two that will be a key point to relate the current session objectives.

#### 3.1.1 Teaching Skills

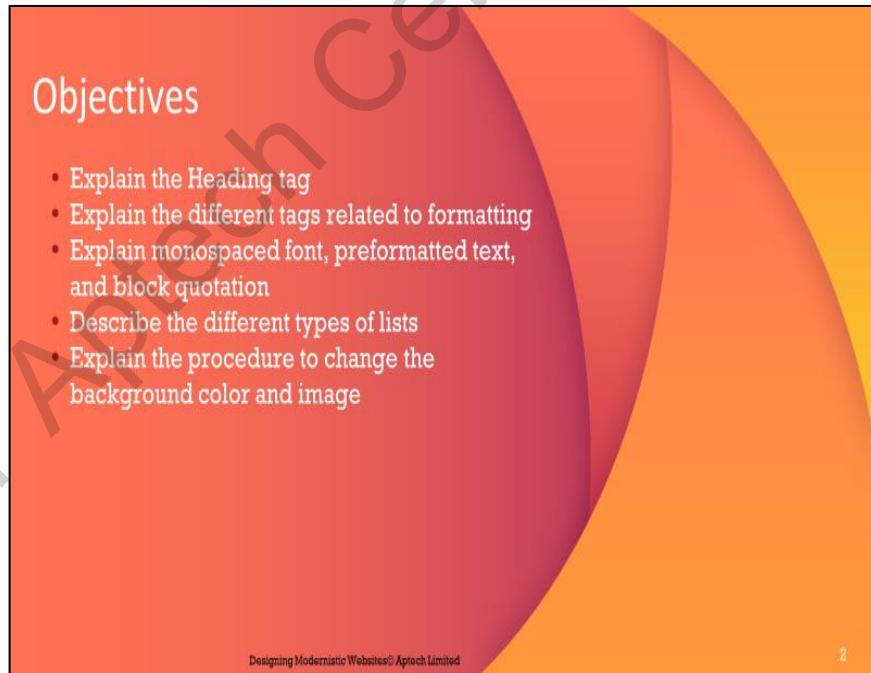
To teach this session, you should be well versed with formation of tags and different types of lists that can be displayed on a Web page. Also, monospaced font, preformatted text, and block quotation should be known.

You should teach the concepts in the theory class using the images provided. For teaching in the class, you are expected to use slides and LCD projectors.

### In-Class Activities

Follow the order given here during In-Class activities.

Slide 2



#### Instructions to the Trainer(s):

Give students a brief overview of the current session through the session objectives listed in Slide 2.

### 3.2 In-Class Explanations

Slide 3

The slide has a decorative background with a red-to-orange gradient and a curved white shape on the right side. The title 'Introduction' is at the top. Below it are four bullet points in colored boxes:

- Text content of Web page forms an important part of a Website.
- Text must be attractive, easy to read, and should be short and crisp.
- Text formatting options such as bold, italics, superscript, subscript, and so on must be applied to attract the user attention.
- Background color and image of the Web page can be specified using HTML.

At the bottom left is the text 'Designing Modernistic Websites © Aptech Limited'. At the bottom right is the number '3'.

#### Instructions to the Trainer(s):

- Using Slide 3, introduce the session to students.
- Mention text content of the Web page form an important part of a Website.
- This text content must not only be informative, but also attractive. It must be easy to read and must have short and crisp sentences for easy understanding of the Web user.
- To attract the attention of the user, headings must be appropriately provided.
- Also, text formatting options such as bold, italics, subscript, superscript, and so on must be applied on the text. Bullets can also be used to list the text in a systematic manner.
- The background color and background image of a Web page can be specified using HTML.

## Slides 4 and 5

### Headings 1-2

Heading elements define headings for contents such as text and images.

Specifies the hierarchical structure of a Web page by grouping the contents.

HTML defines six levels of headings ranging from H1 to H6.

- H1 is the top level heading and is displayed with largest font size
- H6 is the lowest-level heading and is displayed with smallest font size

Designing Modernistic Websites © Aptech Limited

4

### Headings 2-2

- Following Code Snippet demonstrates how to specify the six levels of heading in an HTML page:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Headings</title>
  </head>
  <body>
    <h1>H1 Heading</h1>
    <h2>H2 Heading</h2>
    <h3>H3 Heading</h3>
    <h4>H4 Heading</h4>
    <h5>H5 Heading</h5>
    <h6>H6 Heading</h6>
  </body>
</html>
```

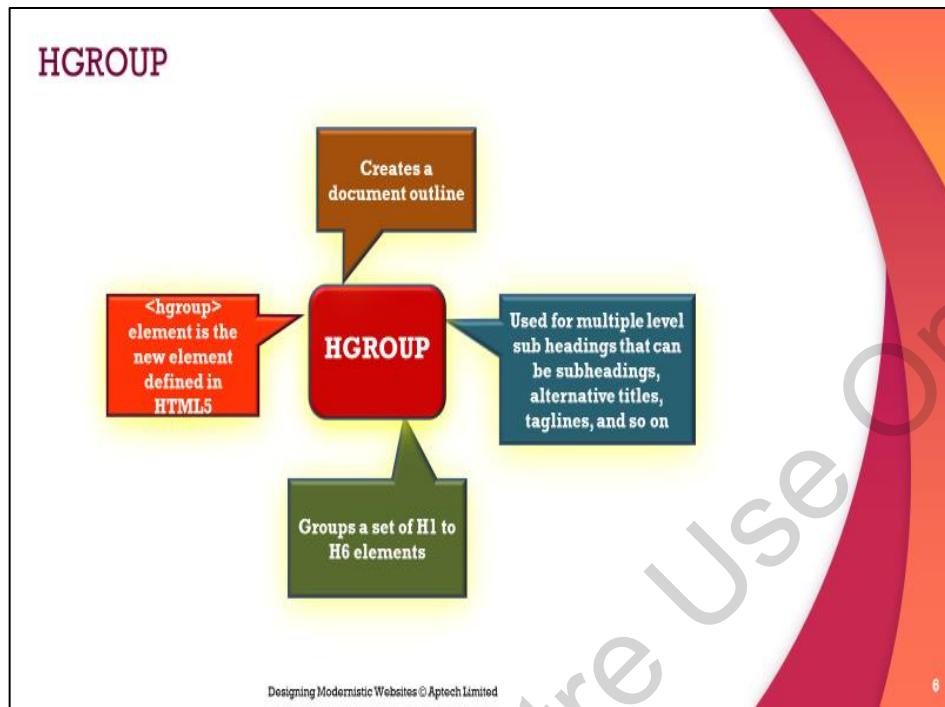


Designing Modernistic Websites © Aptech Limited

5

**Instructions to the Trainer(s):**

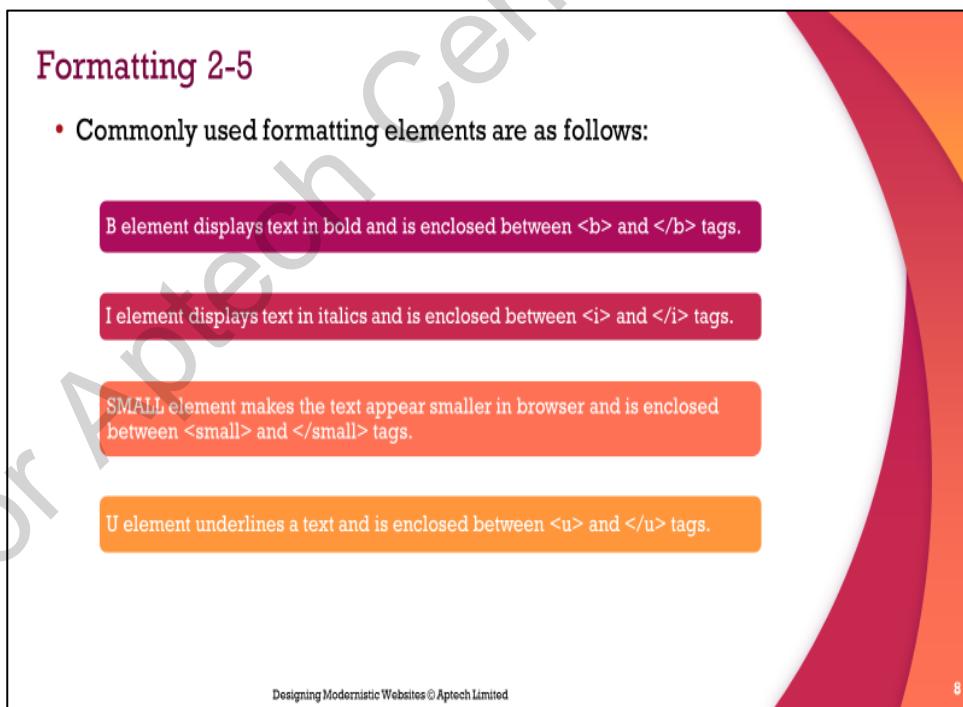
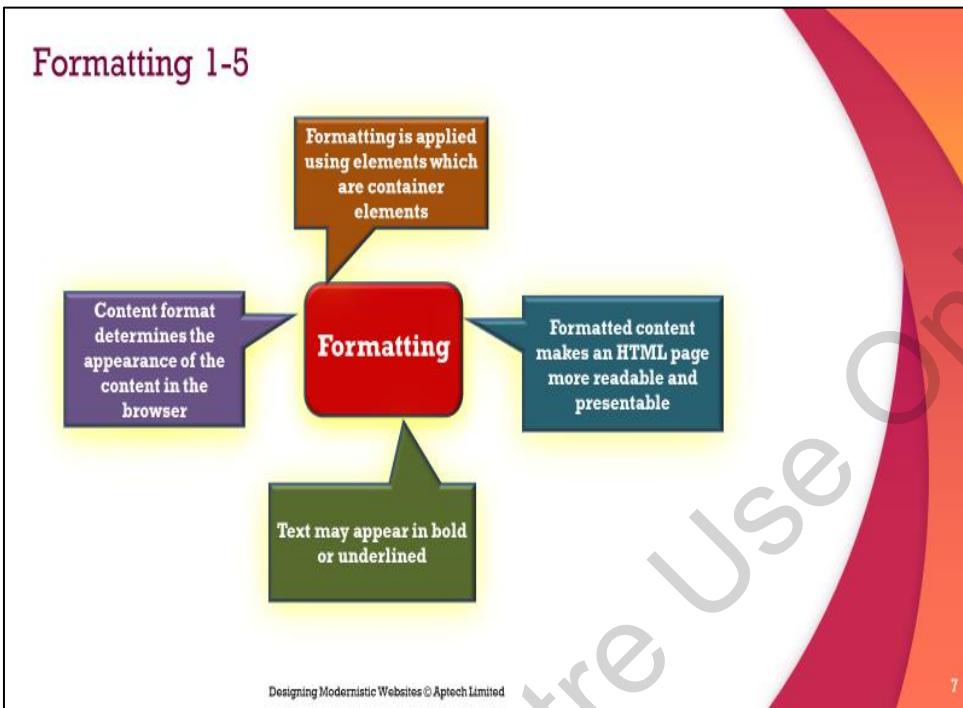
- Using Slides 4 and 5, explain heading tags.
- The heading elements define heading for content such as text and images. They specify a hierarchical structure of a Web page by grouping the contents into different headings.
- Mention that heading under the H1 tags will be displayed with the largest size.
- Each subsequent heading will be displayed in a size lower than its previous heading.
- The heading under the H6 tags will be displayed with the lowest size.



**Instructions to the Trainer(s):**

- Using Slide 6, explain the `hgroup` tag.
- Mention `<hgroup>` element is a new element defined in HTML5. It is used to group titles and their subtitles.
- The element is used to group a set of H1–H6 elements. These are used for headings that have multiple levels that can include subheadings, alternative titles, taglines, and so on.
- The main advantage of using the `<hgroup>` tag is to create a document outline.
- For example, you might have a level 1 heading, followed by a subheading in a level 2 heading. In this instance, the level 2 heading is different to other level 2 headings in the document, because it is an extension of the level 1 heading (i.e. it is a subheading of the heading).
- Therefore, to group the two together, you can use the `<hgroup>` tag.
- The following example demonstrates this scenario:

```
<article>
<hgroup>
<h1>Heading level 1</h1>
<h2>Heading level 2</h2>
</hgroup>
<p>This is example for hgroup element</p>
</article>
```



## Formatting 3-5

- Following Code Snippet demonstrates the use of basic formatting elements:

```
<!DOCTYPE html>
<html>
<head>
<title>Formats</title>
</head>
<body>
<h2>Using HTML Formatting Elements</h2><br>
<b>This text is displayed in bold.</b><br>
<i>This text is displayed in italic.</i><br>
<u>This text is underlined.</u><br>
<small>This text is displayed smaller.</small>
</body>
</html>
```



Designing Modernistic Websites © Aptech Limited

9

## Formatting 4-5

- Some more formatting elements are as follows:

DEL element encloses deleted text and is placed between <del> and </del> tags.

INS element encloses inserted text and is placed between <ins> and </ins> tags.

STRONG element emphasizes the text and is placed between <strong> and </strong> tags.

SUB element displays a text as subscript and is enclosed between <sub> and </sub> tags.

SUP element displays a text as superscript and is enclosed between <sup> and </sup> tags.

Designing Modernistic Websites © Aptech Limited

10

## Formatting 5-5

- Following Code Snippet demonstrates the use of other formatting elements:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Updating and Shifting Text</title>
  </head>
  <body>
    <h3>Updating, Emphasizing, and Shifting Text</h3>
    This is an example of <del>deleted</del>
    <ins>inserted</ins> text.<br/>
    The is an example of <strong>Strong</strong>
    text.<br/>
    The is an example of <sub>subscript</sub>text.<br/>
    The is an example of <sup>superscript</sup>
    text.<br/>
  </body>
</html>
```



Designing Modernistic Websites © Aptech Limited

11

### Instructions to the Trainer(s):

- Using Slides 7 to 11, explain different formatting tags.
- The content format determines how the content will appear in the browser.
- For example, when you visit a Website, some text appears in a specific format such as bold or underlined. This means that the formatted content makes an HTML page look readable and presentable.
- In HTML, formatting is applied to the text by using formatting elements, which are container elements.
- The commonly used HTML formatting elements are as follows:
  - B: The B element displays the text in bold. The text that must be displayed in bold is enclosed between <b> and </b> tags.
  - I: The I element displays the text in italic. The text that must be displayed in italic is enclosed between <i> and </i> tags.
  - SMALL: The SMALL element makes the text appear smaller in a browser. The text that must be displayed smaller is enclosed between <small> and </small> tags.
  - U: The U element applies an underline to the text. The text that must be underlined is enclosed between <u> and </u> tags.
- Mention that HTML provides some more formatting elements that can be applied to the text.
- These formatting elements are as follows:
  - DEL: The DEL element encloses text, which has been deleted from the document. The text to be deleted is placed in the <del> and </del> tags.
  - INS: The INS element encloses text, which has been inserted in the document. The

text to be inserted is placed in the `<ins>` and `</ins>` tags. The `INS` element can be used with `DEL` element to inform the user about the deleted text, which is replaced by the new text.

- **STRONG:** The `STRONG` element emphasizes the text as compared to its surrounding text. The text to be emphasized is placed in the `<strong>` and `</strong>` tags.
- **SUB:** The `SUB` element displays the text as subscript. The text to be displayed as subscript is enclosed in `<sub>` and `</sub>` tags.
- **SUP:** The `SUP` element displays the text as superscript. The text to be displayed as superscript is enclosed in `<sup>` and `</sup>` tags.

**In-Class Question:**

**Question:** Which tag is used to format a text as subscript?

**Answer:** `<SUB>` tag is used to format a text as subscript.

## Monospaced and Preformatted Text 1-2

Monospaced font allows the same amount of horizontal space between fonts irrespective of font size, shape, and type.

Monospaced fonts are used for programming code snippets, instruction texts, and ASCII characters.

<pre> tag is used for preformatted text content.

<pre> tag applies a fixed-font width to the text content.

<pre> tag allows you to copy-paste the content along with the formatting from the source.

## Monospaced and Preformatted Text 2-2

- Following table lists some of the predefined tags and their description:

| Tag    | Description                                    |
|--------|--|
| <em>   | Used for emphasized text                       |
| <dfn>  | Used for definition term                       |
| <code> | Used for computer code                         |
| <samp> | Used for sample output from a computer program |
| <cite> | Used for citation                              |

**Instructions to the Trainer(s):**

- Using Slides 12 and 13, explain the monospaced and preformatted text.
- Explain by using monospaced font in HTML5, a Web developer can provide the same amount of horizontal space between the fonts, even if the font size, shape, and type are not the same.
- Monospaced fonts are used for programming code scripts, instruction texts, and ASCII characters.
- The `<pre>` tag is used to apply preformatted text content to a Web page and has a fixed-font with. It also maintains a standard formatting for spaces and line breaks.
- The `<pre>` tag is usually used when you want to copy paste content from a source, but do not want to change the formatting of the same.
- The content would be copied while maintaining all the line breaks and spaces.

## Formatting a Block Quotation

- To define a long quotation or block quotation, `<blockquote>` tags are used.
- `<blockquote>` tag indents the quotation in browsers.
- Following Code Snippet demonstrates the use of `<blockquote>` tags:

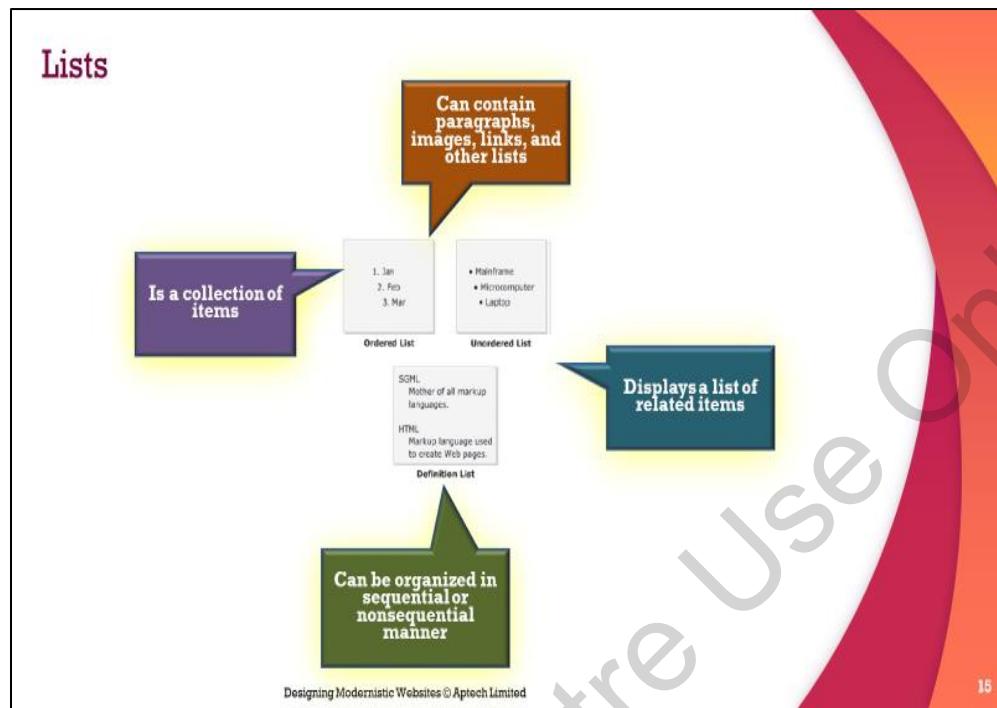
```
<blockquote>  
"When one door closes, another opens; but we often look so long and so  
regretfully upon the closed door that we do not see the one which has opened  
for us." -Alexander Graham Bell  
</blockquote>
```

Designing Modernistic Websites © Aptech Limited

14

### Instructions to the Trainer(s):

- Using Slide 14, explain formatting a block quotation concept to students.
- Mention to define a long quotation or block quotation, the `<blockquote>` tag can be used.
- When the `<blockquote>` tag is used, the quotation is indented in browsers.
- Also, explain the example mentioned on the Slide.



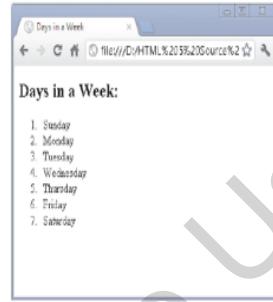
**Instructions to the Trainer(s):**

- Using Slide 15, explain different lists.
- A list is a collection of items which might be organized in a sequential or non-sequential manner. You can use a list to display related items that belong to a particular category.
- For example, to display the types of computers, such as mainframe, microcomputer, and laptops you would organize these items one below the other under the Types of Computers category.
- Similarly, HTML allows you to display related items in a list on a Web page.
- A list in HTML can contain paragraphs, line breaks, images, links, and other lists.
- The items within a list are displayed on a Web page one below the other using bullets.
- HTML supports three types of lists. They are as follows:
  - Ordered
  - Unordered
  - Definition

## Ordered Lists 1-2

- List is displayed using a numbered or alphabetic bullet
- Two elements used for creating an ordered list are as follows:
  - OL – Creates an ordered list
  - LI – Specifies an item and it is a sub-element of the OL element
- Following Code Snippet demonstrates the use of ol and li tags:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Days in a Week</title>
  </head>
  <body>
    <h2>Days in a Week:</h2>
    <ol>
      <li>Sunday</li>
      <li>Monday</li>
      <li>Tuesday</li>
      <li>Wednesday</li>
      <li>Thursday</li>
      <li>Friday</li>
      <li>Saturday</li>
    </ol>
  </body>
</html>
```



Designing Modernistic Websites © Aptech Limited

16

## Ordered Lists 2-2

- Following table lists some of the different numbering styles and their description:

| Property's Value | Example       |
|------------------|---------------|
| decimal          | 1, 2, 3...    |
| lower-alpha      | a, b, c...    |
| upper-alpha      | A, B, C...    |
| lower-roman      | i, ii, iii... |
| upper-roman      | I, II, III... |

- list-style-type property is used to specify a numbering style for the ordered list.
- It is the property of the style attribute, which is specified with the <ol> tags.

Designing Modernistic Websites © Aptech Limited

17

### **Instructions to the Trainer(s):**

- Using Slides 16 and 17, explain the ordered lists.
- Mention an ordered list is a list of items arranged in a particular order. Here, the order of the items is important as it indicates a sequential flow.
- For example, to display the days in a week or months in a year, you would use numbered bullets.
- Similarly, HTML allows you to implement ordered lists where each item in the list is displayed using a numbered or alphabetic bullet.
- HTML provides two elements for creating an ordered list. These are as follows:
  - OL: Creates an ordered list.
  - LI: Specifies an item and it is a sub-element of the OL element.
- Also, explain the example and the output in figure for ordered list provided on the slide.
- Mention different numbering styles, such as roman numerals or alphabetic bullets can be applied to an ordered list.
- Explain different numbering styles that can be specified in an ordered list.
- The list-style-type property is used to specify a numbering style for the ordered list. It is the property of the style attribute, which is specified within the <ol> tag.
- The list-style-type property of the style attribute in the code is set to lower-roman. The property and its value are separated by a colon. This means that the days of the week will be displayed sequentially by applying the lower-case roman numbers as bullets.
- The following example shows how to use the list-style-type property to display the list:

```
<ol style='list-style-type: upper-roman'>
<li>Coffee</li>
<li>Tea</li>
<li>Coca Cola</li>
</ol>
```

- Some of the property values for list-style-type property are as follows:
  - disc – displays filled circle
  - armenian – displays armenian numbering
  - circle – displays circle
  - cjk-ideographic – displays plain ideographic number
  - decimal – displays number
  - decimal-leading-zero – displays leading zeros such as 01, 02, and so on
  - lower-alpha – lower-alpha such as a, b, c, and so on
  - lower-roman – lower-roman such as i, ii, iii, iv, and so on
  - none – no marker
  - square – displays square
  - upper-alpha – displays a, b, c, and so on
  - upper-latin – displays a, b, a, and so on
  - upper-roman – displays i, ii, iii, iv, and so on

## Unordered Lists 1-3

- Items are arranged in random order
- Two elements used for creating an unordered list are as follows:
  - UL – Creates an unordered list
  - LI – Specifies an item and it is a sub-element of the OL element
- Following Code Snippet demonstrates the use of UL and LI tag.

```
<!DOCTYPE html>
<html>
  <head>
    <title>Features of EasyPad</title>
  </head>
  <body>
    <h2>Features of EasyPad</h2>
    <ul>
      <li>Opens many files at a time</li>
      <li>Unlimited undo and redo</li>
      <li>Reads and writes both
          Windows and Unix files</li>
    </ul>
  </body>
</html>
```



Designing Modernistic Websites © Aptech Limited

18

## Unordered Lists 2-3

- The list-style-type property specifies the type of bullet to be applied to an unordered list.
- There are three types of bullets defined for the unordered lists:
  - Disc
  - Square
  - Circle
- The default value is disc, which is applied to the unordered list, even if the list-style-type property is not specified.
- Following Code Snippet demonstrates how to apply the square bullet to an unordered list.

```
<!DOCTYPE html>
<html>
<head>
<title>Wild Animals</title>
</head>
<body>
<h2>Wild Animals</h2>
<ul style="list-style-type:square">
<li>Lion</li>
<li>Tiger</li>
<li>Leopard</li>
<li>Wolf</li>
</ul>
</body>
</html>
```

Designing Modernistic Websites © Aptech Limited

19

### Unordered Lists 3-3

- The `list-style-type` property of the style attribute is set to square.
- Hence, the unordered list of wild animals will be displayed using the square bullet as shown in the figure.



Designing Modernistic Websites © Apnech Limited

20

#### Instructions to the Trainer(s):

- Using Slides 18 to 20, explain the unordered list.
- An unordered list is a list where the items are arranged in a random order. This means that you will create an unordered list when the order of related items is not important.
- For example, to list the features of a product, you would create an unordered list. Each item in an unordered list is displayed using symbolic bullets such as circles and squares. These bullets are similar to the bullets provided by Microsoft Office Word. HTML provides the `UL` element to create an unordered list.
- The `UL` element contains the `<ul>` tag and multiple `<li>` sub-elements. The `<ul>` tag marks the beginning of the unordered list. Each of the sub-elements starts with the `<li>` tag displayed with the default symbolic bullet, which is a small black disc.
- The `list-style-type` property specifies the type of bullet to be applied to an unordered list. There are three types of bullets defined for the unordered lists in HTML. These bullet types are namely, disc, square, and circle. The default value is disc which is applied to the unordered list, even if the `list-style-type` property is not specified.
- Using Slide 19, explain the bullets used in unordered list. Also, explain that the `list-style-type` property of the style attribute is set to square in the figure provided on Slide 20.

#### In-Class Question:

**Question:** Which property of the `style` tag is used for specifying the bullet type in the unordered list?

**Answer:** The `list-style-type` property.

## Definition List 1-4

- Refers to a collection of terms with their corresponding descriptions
- Contains the terms along with their descriptions
- Appears with the term indented on the left followed by description on the right or on next line
- Elements required to create a definition list are as follows:

DL – Is a container element that consists of DT and DD sub elements. Specifies that the definition list will be created using these elements.

DT – Specifies the term to be defined or described.

DD – Specifies the definition or description of the term.

## Definition List 2-4

- Steps to create a definition list are as follows:
  1. Specify the DL element to indicate that you want to create a definition list.
  2. Use the DT element to specify the term such as Common Noun.
  3. Use the DD element to specify the description of the term.

### Definition List 3-4

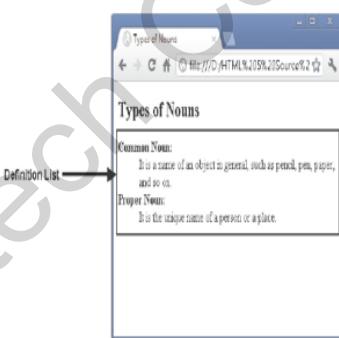
- Following Code Snippet demonstrates the way to create a definition list.

```
<!DOCTYPE html>
<html>
  <head>
    <title>Types of Nouns</title>
  </head>
  <body>
    <h2>Types of Nouns</h2>
    <dl>
      <dt><b>Common Noun:</b></dt>
      <dd>It is a name of an object in general, such as pencil, pen, paper, and so on.</dd>
      <dt><b>Proper Noun:</b></dt>
      <dd>It is the unique name of a person or a place.</dd>
    </dl>
  </body>
</html>
```

Designing Modernistic Websites © Aptech Limited

23

### Definition List 4-4



Designing Modernistic Websites © Aptech Limited

24

#### Instructions to the Trainer(s):

- Using Slides 21 to 24, explain the concept of definition list to students.
- A definition list refers to a collection of terms with their corresponding descriptions. For example, you can display a glossary on a Web page by creating a definition list, which will contain the terms and their descriptions.

- A definition list appears with the term indented on the left followed by the description on the right or on the next line. By default, the description text appears on the next line and is aligned with respect to the term.
- You can specify a single line or multiple lines of description for each term. HTML provides three elements to create a definition list.
- These elements are as follows:
  - DL: Is a container element that consists of the DT and DD sub-elements. It specifies that a definition list will be created using these elements.
  - DT: Specifies the term to be defined or described.
  - DD: Specifies the definition or description of the term.
- Consider a scenario, where you want to create a Web page that will display the types of nouns with their descriptions. To do this, you must create a definition list. The steps to create a definition list are as follows:
  - Specify the DL element to indicate that you want to create a definition list.
  - Use the DT element to specify the term such as Common Noun.
  - Use the DD element to specify the description of the term.

## Background and Foreground Colors 1-2

Background properties specify the background color and image for the Web pages.

Background property is a shorthand property that specifies all the background properties in just one declaration.

bgcolor attribute specifies the background color of a document.

- Syntax for bgcolor is:

```
<body bgcolor="color_name|hex_number|rgb_number">
```

where,

color\_name - Specifies the background color with a color name (such as "red")

hex\_number - Specifies the background color with a hex code (such as "#ff0000")

rgb\_number - Specifies the background color with an rgb code (such as "rgb(255,0,0)")

Designing Modernistic Websites © Aptech Limited

25

## Background and Foreground Colors 2-2

Another way to specify a background color for a Web page is by using the style="background-color:color" attribute.

This attribute must be added to the style attribute of the <body> tag.

The foreground color can be specified by using the style="color:color" attribute.

- Example demonstrating the specification of background and foreground color is:

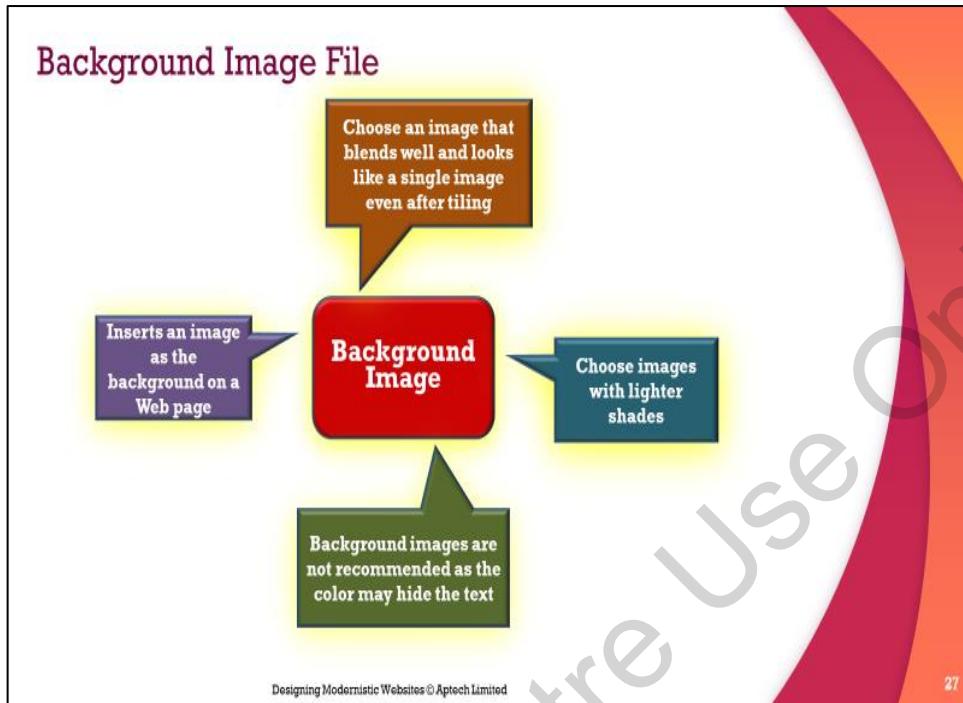
```
<body style="background-color: navy; color: yellow">
```

Designing Modernistic Websites © Aptech Limited

26

**Instructions to the Trainer(s):**

- Using Slides 25 and 26, explain the process to specify background color and image for Web page.
- HTML provides background properties that specify the background color and image for the Web pages. To specify a background for a Web page, use the background property. The background property is a shorthand property that specifies all the background properties in just one declaration. The bgcolor attribute specifies the background color of a document.
- Explain the other way to specify the background color. Another way to specify a background color for a Web page is by using the style="background-color: color" attribute. This attribute must be added to the <body> tag.
- An example for applying a background color using the style attribute is as follows:  
`<body style="background-color: yellow">`
- The color name 'yellow' can also be replaced by the hex code or the rgb code.
- The default text color of the page is specified as the foreground color. The foreground color can be specified by using the style="color: color" attribute. An example for applying a background and foreground color using the style attribute is as follows:  
`<body style="background-color: navy; color: yellow">`



**Instructions to the Trainer(s):**

- Using Slide 27, explain the image file used as background.
- A Website developer can also insert an image as the background on a Web page. These background images are not recommended as sometimes, the colors in the image may hide the text content. Hence, it is best to choose images with lighter shades.
- Also, as the image is tiled, it is best to choose an image that blends well and looks like a single image even after it is tiled.
- The code snippet demonstrates use of image in the background.

```
<html>
<body background="bgimage1.jpg">
<h1>Hello world!</h1>
</body>
</html>
```

## Summary

- ❖ The heading elements define headings for contents such as text and images.
- ❖ The `<hgroup>` element is used to group titles and their subtitles.
- ❖ Monospaced fonts are used for programming code scripts, instruction texts, and ASCII characters.
- ❖ The `<pre>` tag is used to apply preformatted text content to a Web page.
- ❖ To define a long quotation or block quotation, the `<blockquote>` tag can be used.
- ❖ A list is a collection of items, which might be organized in a sequential or nonsequential manner. HTML supports three types of lists namely, ordered, unordered, and definition.
- ❖ HTML provides background properties that specify the background color and image for the Web pages.

28

**Instructions to the Trainer(s):**

- Show students Slide 28.
- Summarize the session by reading out each point on the slide.

## Session 4: Creating Hyperlinks and Anchors

### 4.1 Pre-Class Activities

Before you commence the session, you should familiarize yourself with the topics of this session in-depth. Prepare a question or two that will be a key point to relate the current session objectives.

#### 4.1.1 Teaching Skills

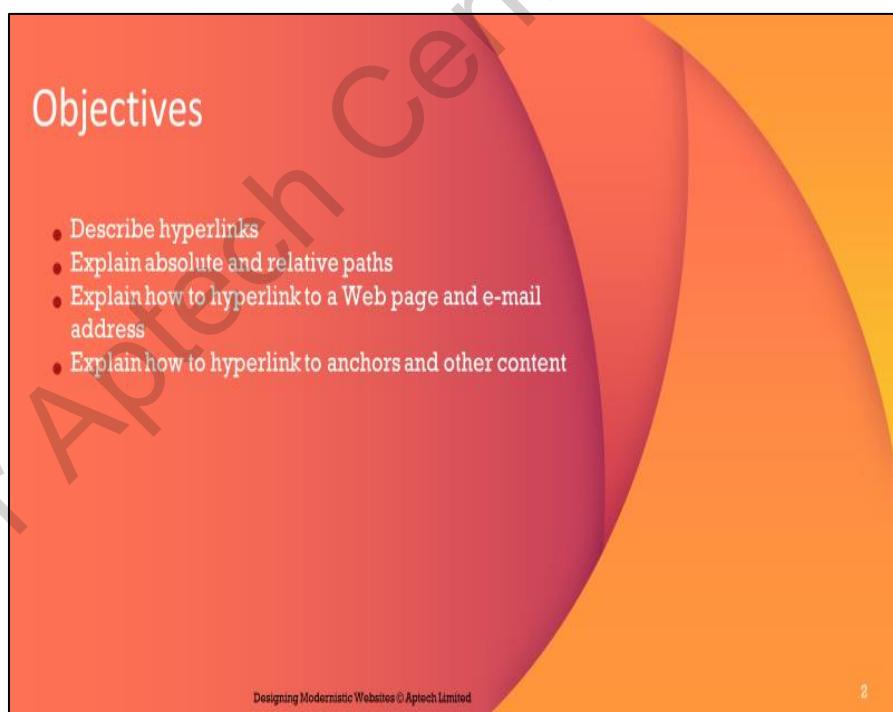
To teach this session, you should be well versed with creating hyperlinks and linking them to Web pages, e-mail address, and other content.

You should teach the concepts in the theory class using the images provided. For teaching in the class, you are expected to use slides and LCD projectors.

### In-Class Activities

Follow the order given here during In-Class activities.

Slide 2



#### Instructions to the Trainer(s):

Give students a brief overview of the current session through the session objectives listed in Slide 2.

## 4.2 In-Class Explanations

Slides 3 and 4

### Hyperlinks 1-2

- A hyperlink is referred to as a link, linking to another Web page, or to a section in the same Web page.
- The `A` (anchor) element is used to create a hyperlink.
- One can specify a text or an image as a hyperlink.
- When mouse is moved over such content, the cursor changes into a hand with its index finger pointing towards the content.
- This means that clicking the link will take the user to the respective link.
- To specify the linked page section or linked Web page, attributes of the `A` element has to be used.
- Following table lists the attributes of the `A` element:

| Attribute             | Description   |
|-----------------------|---|
| <code>href</code>     | Specifies the URL of the Web page to be linked or the value of the <code>name</code> attribute. |
| <code>hreflang</code> | Indicates the language of the destination URL.  |
| <code>name</code>     | Specifies the section of the Web page, which is to be linked.                                   |

Example:

This is a link



Designing Modernistic Websites © Aptech Limited

3

### Hyperlinks 2-2

- The `<a>` tag is used to provide a hyperlink.
- This contains `href` attribute that would contain the link to a URL or path of a Web page.
- An example of a `href` attribute code is as follows:  
`<a href="http://www.aptech-worldwide.com/">`
- The description and reference text that will serve as a hyperlink must be provided before closing the `<a>` tag by using `</a>`.
- An example of a hyperlink along with its output is as follows:

```
<html>
<head>
</head>
<body>
    <a href="http://www.aptech-worldwide.com/">
        Click to view the Aptech Web site</a>
    </body>
</html>
```

Output:



Designing Modernistic Websites © Aptech Limited

4

**Instructions to the Trainer(s):**

- Using Slides 3 and 4, explain hyperlinks.
- Mention hyperlink is referred to as a link. It refers for linking to another Web page or to a section in the same Web page.
- Explain the students on how to use the hyperlink to navigate the Web pages. Anything, a text or an image can be provided as a hyperlink.
- When you move the mouse over a hyperlink content, the cursor changes into a hand with its index finger pointing towards the content. This means that clicking it, will take you to the respective link.
- Explain the basic syntax to provide a hyperlink `<a>` tag that is used to provide a hyperlink.
- The `href` attribute is used to provide a URL or path of a Web page.
- An example of a `href` attribute code, `<a href=" http://www.aptech-worldwide.com/">`

## Target Attribute

- The `target` attribute of the `A` element specifies location where the linked Web page will open when a link is clicked.
- One can assign values to the `target` attribute.
- Following table lists some of the values of the `target` attribute:

| Value               | Description   |
|---------------------|---|
| <code>_blank</code> | Loads the target URL in a new blank window.                             |
| <code>_self</code>  | Loads the target URL in the same window as that of the current Webpage. |
| <code>_top</code>   | Loads the target URL in the complete area of window.                    |

Designing Modernistic Websites © Aptech Limited

5

### Instructions to the Trainer(s):

- Using Slide 5, explain the concept of target attribute in detail.
- The `target` attribute of the `<a>` tag or element specifies the location where the linked Web page will open when a link is clicked.
- Then, explain the values that can be assigned to the `target` attribute as shown in the table listed on the slide.

## Absolute and Relative Paths 1-2

- Absolute paths are links that contain the complete address to get to a Web page.
- Absolute paths are the best way to link to a Website.
- The syntax of an absolute path is as follows:

**Syntax:**

```
<a href="http://www.aptech-worldwide.com/pages/about-us/aboutus_aboutaptechworldwide.html">Aptech Web site</a>
```

- Relative paths are links that are provided when the files of a Web page are in the same folder as the page displaying the link.
- The syntax of a relative path is as follows:

**Syntax:**

```
<a href="aboutus_aboutaptechworldwide.html"> Aptech Web site</a>
```

Designing Modernistic Websites © Aptech Limited

6

## Absolute and Relative Paths 2-2

- To link to the files present in the subfolder, you need to provide the path to the subfolder.
- For example, if the file `aboutus_aboutaptechworldwide.html` is in a subfolder named `about-us` then, the syntax is as follows:

**Syntax:**

```
<a href="about-us/aboutus_aboutaptechworldwide.html"> Aptech Web site</a>
```

- Files that are present in folders that are one level up can also be linked using a relative path. The syntax to link to a file one level up is as follows:

**Syntax:**

```
<a href="../../aboutus_aboutaptechworldwide.html">Aptech Web site</a>
```

Designing Modernistic Websites © Aptech Limited

7

### **Instructions to the Trainer(s):**

- Using Slides 6 and 7, explain the absolute and relative path to the students.
- Absolute paths are links that contain the complete address to get to a Web page. Absolute paths are the best way to link to a Website. Explain the syntax provided on the slide to explain absolute path which includes the domain name of the Website.
- For example, `<a href=" http://www.aptech-worldwide.com/pages/about-us/ aboutus_aboutaptechworldwide.html ">Aptech Web site</a>`.
- Relative paths are links that are provided when the files of a Web page are in the same folder as the page displaying the link. Explain the syntax of a relative path.
- For example, `<a href="aboutus_aboutaptechworldwide.html">Aptech Website</a>`.
- To link to the files present in the subfolder, you must provide the path to the subfolder.
- For example, if the file `aboutus_aboutaptechworldwide.html` is in a subfolder named, `about us` then the syntax is as follows: `<a href="about-us/aboutus_aboutaptechworldwide.html">Aptech Web site</a>`
- Similarly, files that are present in folders that are one level up can also be linked using a relative path.
- The syntax to link to a file one level up is as follows: `<a href="../aboutus_aboutaptechworldwide.html">Aptech Web site</a>`

### **In-Class Question:**

**Question:** What is the correct HTML for creating a hyperlink?

- `<a url="http://www.aptechworldwide.com">Aptech World Wide</a>`
- `<a>http://www.aptechworldwide.com</a>`
- `<a name="http://www.aptechworldwide.com">Aptech WorldWide </a>`
- `<a href="http://www.aptechworldwide.com">Aptech WorldWide </a>`

**Answer:** `<a href="http://www.aptechworldwide.com">Aptech World Wide</a>`

## Hyperlink to an E-mail Address

- Hyperlinks can be applied to e-mail addresses in the same way as they can be given for Web pages.
- Various tasks can be performed when a hyperlink is given to an e-mail, such as starting default e-mail client, creating a new message, adding the subject line, and so on.
- To add an e-mail to a hyperlink, the `href` attribute must be used and followed by `mailto:email address`.
- Following code snippet shows the way to hyperlink an e-mail address:

```
<a href="mailto:customercare@aptech.ac.in">Customer Care</a>
```

- To automatically add a subject line in the new e-mail message, the `?subject=` attribute must be inserted after the e-mail address.
- Following code snippet shows the way to add a subject line to a hyperlinked e-mail address:

```
<a href="mailto:customercare@aptech.ac.in?subject=E-mail to Customer Care">Customer Care</a>
```

### Instructions to the Trainer(s):

- Using Slide 8, explain how to hyperlink to an e-mail address.
- Hyperlinks can be even applied to e-mail addresses in the same way, as they can be given for Web pages. There are various tasks that can be performed when a hyperlink are given to an e-mail. Some of these tasks include starting the default e-mail client, creating a new message, inserting the recipients address, adding the subject line, and so on.
- To add an e-mail to a hyperlink, the `href=` attribute must be used and followed by `mailto: address` attribute. Clicking `mailto` link, opens users default email program or software. A new email page is created with 'To' field containing the address of the name specified on the link by default.
- To automatically add a subject line in the new e-mail message, the `?subject=` attribute must be inserted after the e-mail address.

## Hyperlink to Other Content Types

- Hyperlinks can also be used to link other files and documents.
- Some commonly linked file types on Web pages using hyperlinks are zipped files (.zip), executable files (.exe), documents (.doc), PDF reader files (.pdf), and so on.
- Hyperlinks can also be used to link to graphical .jpg and .gif files.
- To specify a file instead of the Web page, the name of the file must be provided in the <a> tag as shown in the following code snippet:

```
<a href="Compressed.zip">Click to download the compressed zip  
file </a>
```

### Instructions to the Trainer(s):

- Using Slide 9, explain how to hyperlink other contents.
- Hyperlinks can be used to not only refer to another Web page or e-mail address, but also can be used to link to other files and documents.
- Some of the files that are commonly linked on Web pages using hyperlinks are zipped files (.zip), executable files (.exe), documents (.doc), PDF reader files (.pdf), and so on. Hyperlinks can also be used to link to graphical .jpg and .gif files.
- To specify a file instead of the Web page, the name of the file must be provided in the <a> tag.

## Summary

- ❖ A hyperlink is referred to as a link. It refers to linking to another Web page or to a section in the same Web page.
- ❖ The A (anchor) element is used to create a hyperlink.
- ❖ The target attribute of the A element specifies the location where the linked Web page will open when a link is clicked.
- ❖ Absolute paths are links that contain the complete address to get to a Web page.
- ❖ Relative paths are links that are provided when the files of a Web page are in the same folder as the page displaying the link.
- ❖ To add an e-mail to a hyperlink, the href= attribute must be followed by mailto:email address.
- ❖ Hyperlinks can also be used to link to files and documents such as zipped files (.zip), executable files (.exe), documents (.doc), PDF reader files (.pdf), and so on.

10

**Instructions to the Trainer(s):**

- Show students Slide 10.
- Summarize the session by reading out each point on the slide.

## Session 5: Introduction to CSS3

### 5.1 Pre-Class Activities

Before you commence the session, you should familiarize yourself with the topics of this session in-depth. Prepare a question or two that will be a key point to relate the current session objectives.

#### 5.1.1 Teaching Skills

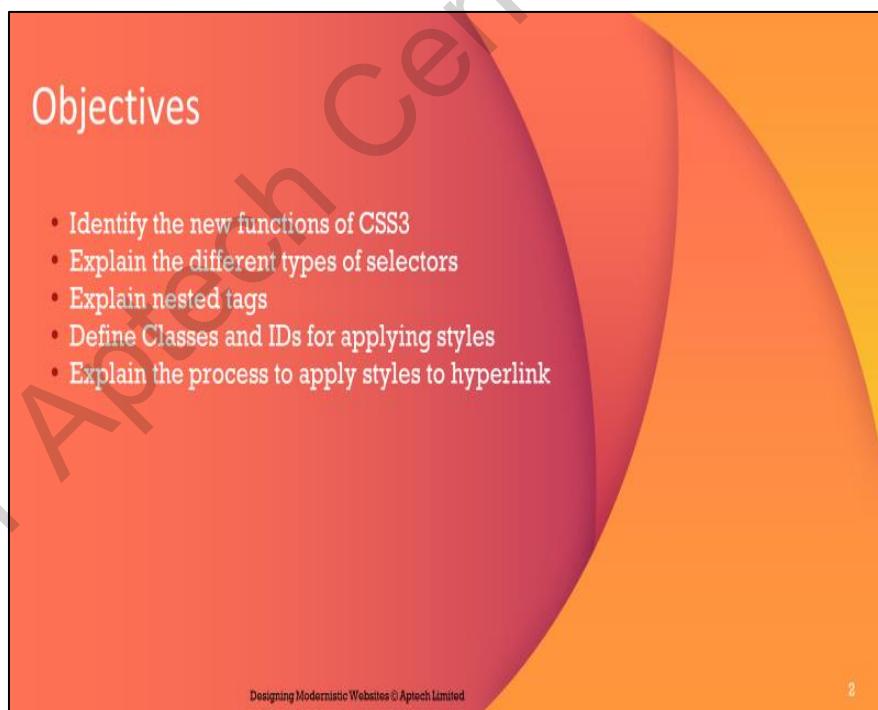
To teach this session, you should be well versed with functions of CSS3, applying styles to hyperlinks, and different types of selectors.

You should teach the concepts in the theory class using the images provided. For teaching in the class, you are expected to use slides and LCD projectors.

### In-Class Activities

Follow the order given here during In-Class activities.

Slide 2



### Instructions to the Trainer(s):

Give students a brief overview of the current session through the session objectives listed in Slide 2.

## 5.2 In-Class Explanations

Slide 3

### Introduction

- Cascading Style Sheet (CSS) is a style sheet language.
- It informs the browser how to present a document.
- It uses a markup language for describing the presentation semantics of a document.
- It defines how HTML elements are to be displayed.

Designing Modernistic Websites © Aptech Limited

3

#### Instructions to the Trainer(s):

- Using Slide 3, explain CSS in detail.
- CSS is a style sheet language used for informing the browser how to present a document. It uses markup language for describing the presentation semantics of a document.
- In other words, an HTML document defines the content of the file, whereas the CSS file defines how HTML elements are to be displayed.
- The main purpose of CSS is to primarily enable the separation of document content from document presentation which includes elements such as the margins, colors, and fonts.

## Cascading Style Sheet 3 (CSS3)

- Used for adding style such as fonts, colors, and spacing to Web documents.
- Has multiple levels and profiles.
- Updates each level of CSS from the earlier version, by adding new features.
- Denotes version as CSS1, CSS2, CSS3, and CSS4, where the numbers are different for each version or level.
- Is divided into multiple documents called 'modules' and each of these modules have new capabilities or extends the features present in CSS2.
- Started drafting of CSS3 when publication of the original CSS2 recommendation was released.

Designing Modernistic Websites © Aptech Limited

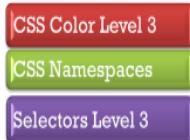
4

### Instructions to the Trainer(s):

- Using Slide 4, explain students the concept of CSS3.
- CSS is a mechanism used for adding styles such as fonts, colors, and spacing to Web documents.
- CSS has multiple levels and profiles. Each level of CSS is updated from the earlier version, by adding new features.
- CSS version are denoted as CSS1, CSS2, CSS3, and CSS4, where the numbers are different for each version or level.
- CSS3 is divided into multiple documents called 'modules'. Each of these modules have new capabilities or extends the features present in CSS2.
- Drafting of CSS3 started when publication of the original CSS2 recommendation was released. The first CSS3 drafts were released on June 1999.
- CSS3 extends variety of new ways to create an impact with any designs, with quite a few important changes.

## Modules 1-4

- As CSS3 is available as modules and is still evolving, there are many modules having different stability and status.
- Only three modules are released as recommendations and they are as follows:



- Modules that are stable and in recommendation stage are as follows:



Designing Modernistic Websites © Aptech Limited

5

## Modules 2-4

- Modules that are in testing phase and in recommendation stage are as follows:



- Modules that are in refining phase and in working draft stage are as follows:

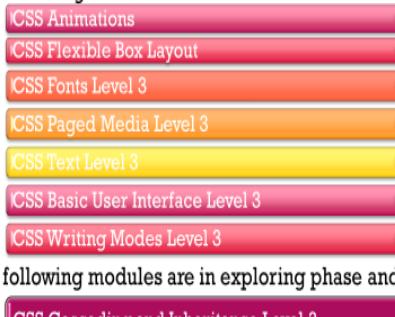


Designing Modernistic Websites © Aptech Limited

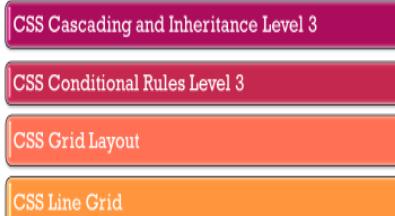
6

## Modules 3-4

- Modules that are in revising phase and in working draft and recommendation stage are as follows:



- Some of the following modules are in exploring phase and in working draft stage:



Designing Modernistic Websites © Aptech Limited

7

## Modules 4-4

- Modules that are in rewriting phase and in working draft stage are as follows:



- Modules that are in abandoned phase and in working draft stage are as follows:



Designing Modernistic Websites © Aptech Limited

8

### Instructions to the Trainer(s):

- Using Slides 5 to 8, explain modules available in CSS3.
- Explain that since CSS3 is available as modules and is still evolving, there are many modules having different stability and status.

- Out of the 50 modules published by the CSS working group, only three modules are released as recommendations and they are as follows:
  - CSS Color Level 3
  - CSS Namespaces
  - Selectors Level 3
- Rest of the modules are in refining, revising, and testing phase.
- Some of the most popular CSS3 modules that are supported in almost all new versions of the browsers are:
  - Selectors
  - Box Model
  - Backgrounds and Borders
  - Image Values and Replaced Content
  - Text Effects
  - 2D/3D Transformations
  - Animations
  - Multiple Column Layout
  - User Interface

## Slides 9 and 10

### CSS Syntax 1-2

Syntax of CSS consists of three parts namely, **selector**, **property**, and **value**.

#### Selector

- Is an HTML element for which you want to specify the style or the formatting instruction.

#### Property of a selected element

- Is a CSS property that specifies the type of the style to be applied to the selector.

#### Value

- Refers to the value of the CSS property and a CSS property can have multiple values. Property and the value for a selector are separated with a colon (:). They are enclosed within the curly brackets ({}), that is known as the declaration block.

Designing Modernistic Websites © Aptech Limited

9

### CSS Syntax 2-2

- Various combinations available to specify rules for HTML elements are as follows:

You can specify multiple selectors for a single property by grouping the selectors. To group the selectors, the selectors are separated by commas followed by a declaration block of properties and values.

You can specify multiple property-value pairs for a selector, which are separated by a semicolon (;) within the declaration block.

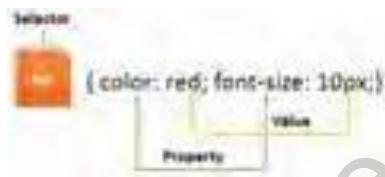
You can specify properties for multiple selectors. Here, the comma-separated selectors are followed with multiple property-value pairs.

Designing Modernistic Websites © Aptech Limited

10

### **Instructions to the Trainer(s):**

- Using Slides 9 and 10, explain CSS syntax.
- The general syntax of CSS consists of three parts namely, selector, property, and value.
  - **Selector** - A selector is an HTML element for which you want to specify the style or the formatting instruction.
  - **Property** - A property of a selected element is a CSS property that specifies the type of the style to be applied to the selector. CSS allows controlling the appearance of the content by providing various properties. These properties include text properties, positioning properties, font properties, color properties, and so on.
  - **Value** - The property and the value for a selector are separated with a colon (:). They are enclosed within the curly brackets ({} ) that is known as the declaration block.
- Following figure explains how to declare a CSS for an HTML element.



- Note that you can have various combinations to specify rules for HTML elements.
  1. You can specify multiple property-value pairs for a selector, which are separated by a semicolon (;) within the declaration block.
  2. You can specify multiple selectors for a single property by grouping the selectors. To group the selectors, the selectors are separated by commas followed by a declaration block of properties and values.
  3. You can specify properties for multiple selectors. Here, the comma-separated selectors are followed with multiple property-value pairs.

### **In-Class Question:**

**Question:** What should be the CSS code to define the red color for the page?

**Answer:** body {color: red;}

## Length Measurement Units 1-4

CSS uses various units of measurements for specifying size of the font, width, and height of margins, and so on.

These units measure the horizontal and vertical length of the content.

CSS supports two types of length measurement units namely, relative and absolute.

Designing Modernistic Websites © Aptech Limited

11

## Length Measurement Units 2-4

Relative length specifies the length units related to other length property that are calculated in comparison to a current value.

- Following table lists some of the relative length units:

| Relative Length | Description  |
|-----------------|--|
| em              | Specifies the font size (height) of a particular font. The em unit is relative to the value of the font-size property of the selector.             |
| ex              | Specifies the 'x-height' of a particular font. The 'x-height' value is approximately half the font size or the height of the lowercase letter 'x'. |
| px              | Specifies the size in pixels, which is relative to the screen of the device.   |

Designing Modernistic Websites © Aptech Limited

12

## Length Measurement Units 3-4

Absolute lengths are specified when the Web page designer is aware of the physical properties of the output device and are specific and fixed values.

- Following table lists some of the absolute length units:

| Relative Length | Description   |
|-----------------|---|
| in              | Specifies the size in inches, where 1 inch = 2.54 centimeters   |
| cm              | Specifies the size in centimeters                               |
| mm              | Specifies the size in millimeters                               |
| pt              | Specifies the size in points, where 1 point = 1/72th of an inch |
| pc              | Specifies the size in picas, where 1 pica = 12 points           |

Designing Modernistic Websites © Aptech Limited

13

## Length Measurement Units 4-4

Percentage allows specifying the length of the content, which is relative to another value.

- Shows use of percentage in defining the style:

```
H1
{
    font-size: 120%;
    line-height: 200%;
}
```

Designing Modernistic Websites © Aptech Limited

14

### Instructions to the Trainer(s):

- Using Slides 11 to 14, explain length measurement units. Mention CSS uses various units of measurements for specifying size of the font, width, and height of margins, and so on.
- These units measure the horizontal and vertical length of the content. CSS supports two types of length measurement units namely, relative and absolute.

- Relative length specifies the length units related to other length property that are calculated in comparison to a current value. Explain different relative length provided in the table on Slide 12.
- Then, explain the absolute length. Absolute lengths are specified when the Web page designer is aware of the physical properties of the output device. These are specific and fixed values. Then, explain the absolute length provided in the table on Slide 13.
- Explain the CSS code on Slide 14. Tell the students that percentage allows specifying the length of the content which is relative to another value. Then, explain the CSS code specifies the styles for the H1 element. The font-size property is set to a value of 120%. This means that the size of the header will appear 20% greater than its current size. The line-height property is set to the value 200%. This means that the height of the line will be double the value of the font-size property.

For Aptech Centre Use Only

## Types of Style Sheets

Three types of style sheets namely, inline, internal or embedded, and external style sheets.

An inline style sheet uses the style attribute within an HTML element to specify the style for HTML elements.

An internal style sheet is also included within the HTML document and is defined using the style element.

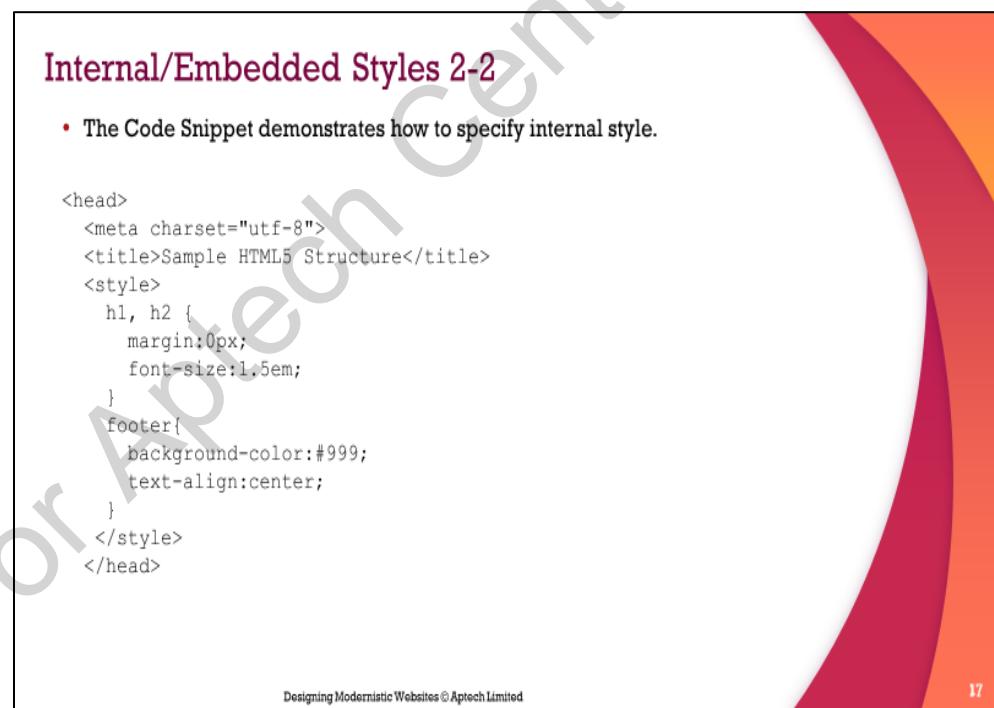
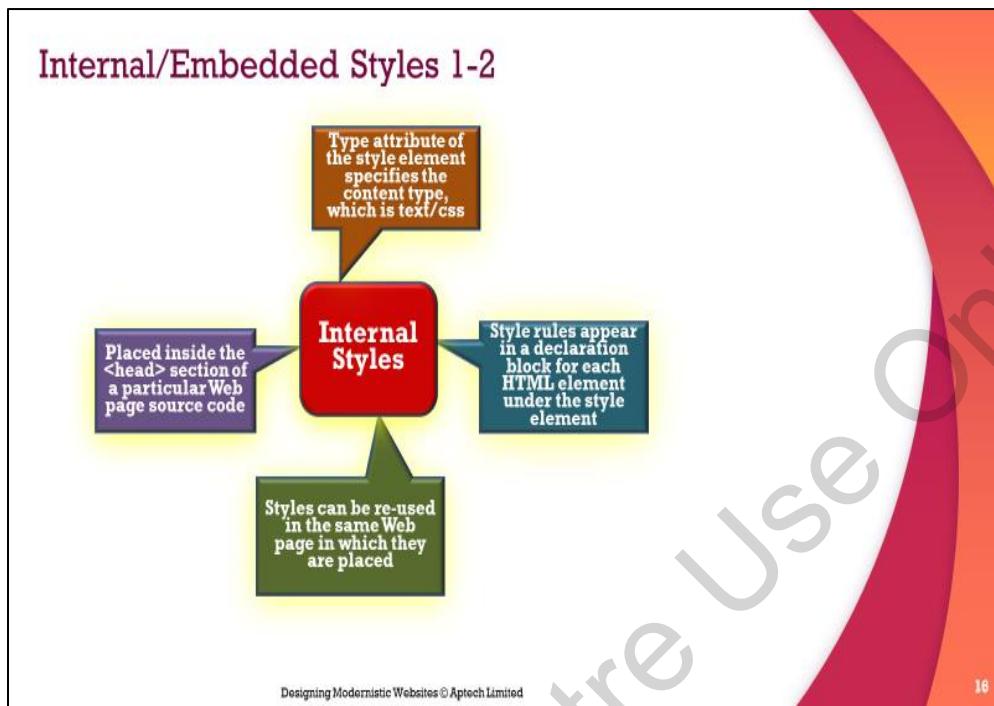
Designing Modernistic Websites © Aptech Limited

15

**Instructions to the Trainer(s):**

- Using Slide 15, explain the types of style sheets.
- Tell the students that the browser reads the style sheet and formats the content on the Web page accordingly.
- There are three types of style sheets namely, inline, internal, or embedded, and external style sheets. Here, explain the two types of style sheets.
- **Inline style sheet** - Uses the style attribute within an HTML element to specify the style for HTML elements.
- **Internal style sheet** - Is included within the HTML document. However, it is defined using the style element within the style element. The style rules appear in a declaration block for each HTML element under the style element. The type attribute of the style element specifies the content type, which is text/css. This means that the content under the style element is CSS code. You can specify any combinations of specifying style rules. The style rules specified for an element will be applied to all the sub-elements. Internal style sheets are useful when styles are to be applied to a specific Web page.

Slides 16 and 17



**Instructions to the Trainer(s):**

- Using Slides 16 and 17, explain the internal style sheet.

- Explain the students that the internal style sheets are placed inside the <head> section on the HTML Web page.
- Explain the CSS properties applied to <h1>, <h2>, and <footer> tags. This can be re-used in the same Web page multiple times.

## Inline Styles

The diagram consists of a central red rounded rectangle labeled "Inline Styles". Four arrows point from surrounding boxes to this central box:

- A red arrow points from a red box containing the text "Are placed directly inside an HTML element".
- A brown arrow points from a brown box containing the text "Cannot be reused at any point of time in a Web page".
- A yellow arrow points from a yellow box containing the text "Web designer cannot use the style builder to create an inline style".
- A blue arrow points from a blue box containing the text "The Code Snippet demonstrates the use of inline style".

**• The Code Snippet demonstrates the use of inline style.**

```
<p style="font-size: 14px; color: purple;"></p>
```

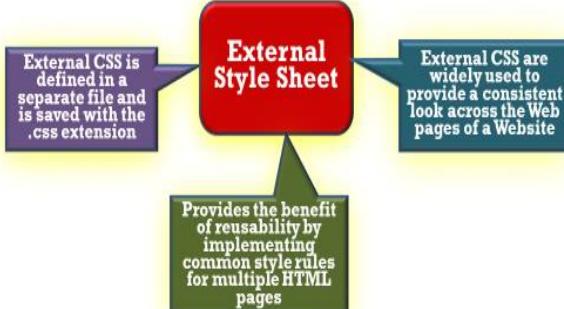
Designing Modernistic Websites © Aptech Limited

18

**Instructions to the Trainer(s):**

- Using Slide 18, explain the inline style sheets.
- Inline styles are placed directly inside an HTML element.
- A Web designer cannot use the style builder to create an inline style.
- Inline style cannot be reused at any point of time in a Web page.

## External Style Sheet 1-2



- The Code Snippet demonstrates the use of external CSS.

```
BODY {  
background-color: gray;  
font-family: arial;  
font-style: italic;  
}
```

Designing Modernistic Websites © Aptech Limited

19

## External Style Sheet 2-2

- Code Snippet shows an example of HTML code using an external CSS style sheet demonstrated earlier.

```
<!DOCTYPE html>  
<html>  
  <head>  
    <LINK rel="stylesheet" type="text/css" href="body.css"/>  
    <title>Webex e-Server</title>  
  </head>  
  <body>  
    This is the fastest Web server...!!  
  </body>  
</html>
```

Designing Modernistic Websites © Aptech Limited

20

**Instructions to the Trainer(s):**

- Using Slides 19 and 20, explain the external style sheets.
- An external CSS is defined in a separate file and is saved with the .css extension. It provides the benefit of reusability by implementing common style rules for multiple HTML pages.
- Hence, external CSS are widely used to provide a consistent look across the Web pages of a Website.
- Explain the code snippet given. <Link> tag is used to link the external style sheet with the Web page.

## Selectors

Selectors refer to the HTML elements with the styles that the users want to apply to them.

The four different types of CSS selectors are as follows:

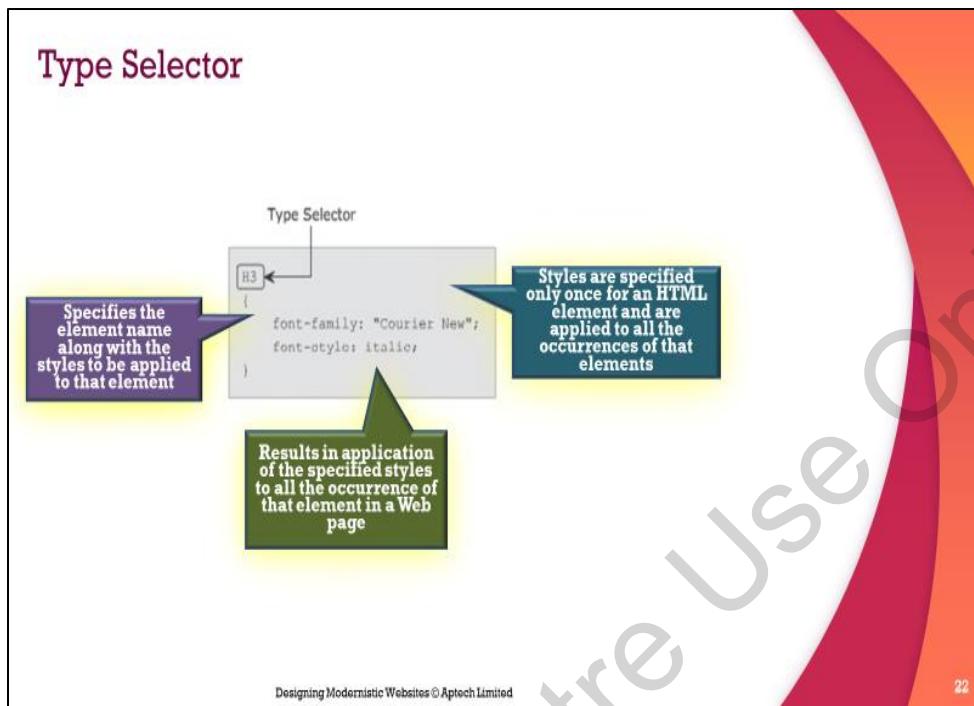
```
graph TD; Selectors[Selectors] --> TypeSelector[Type selector]; Selectors --> ClassSelector[Class selector]; Selectors --> IDSelector[ID selector]; Selectors --> UniversalSelector[Universal selector]
```

Designing Modernistic Websites © Aptech Limited

21

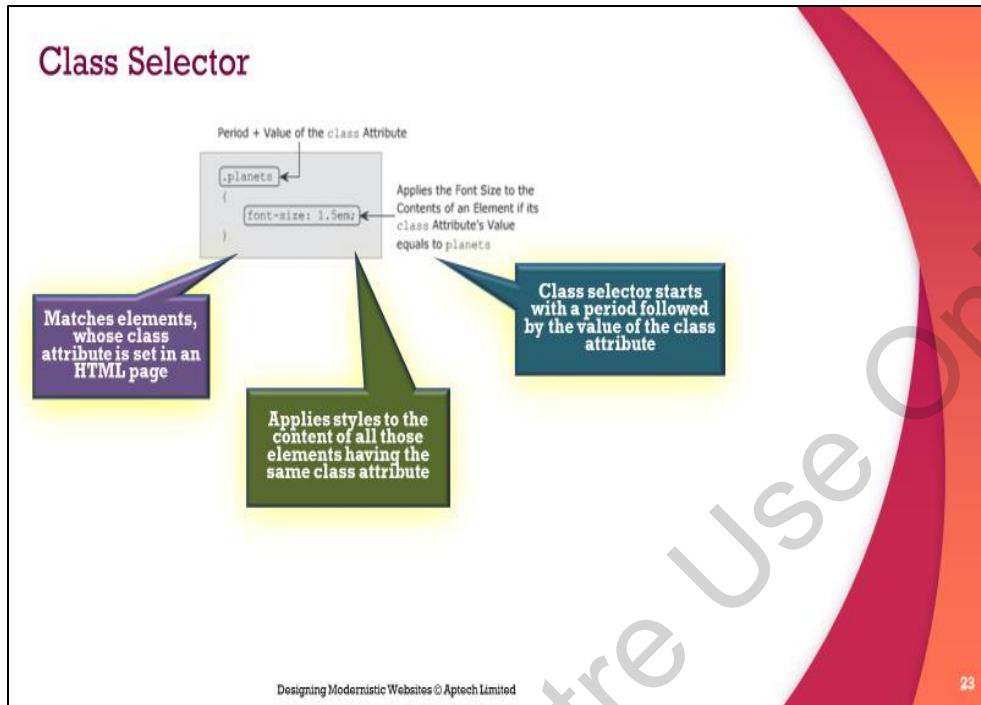
### Instructions to the Trainer(s):

- Using Slide 21, explain selectors.
- Selectors refer to the HTML elements with the styles, the users want to apply to them.
- Four different types of CSS selectors are as follows:
  - Type selector
  - Class selector
  - ID selector
  - Universal selector



**Instructions to the Trainer(s):**

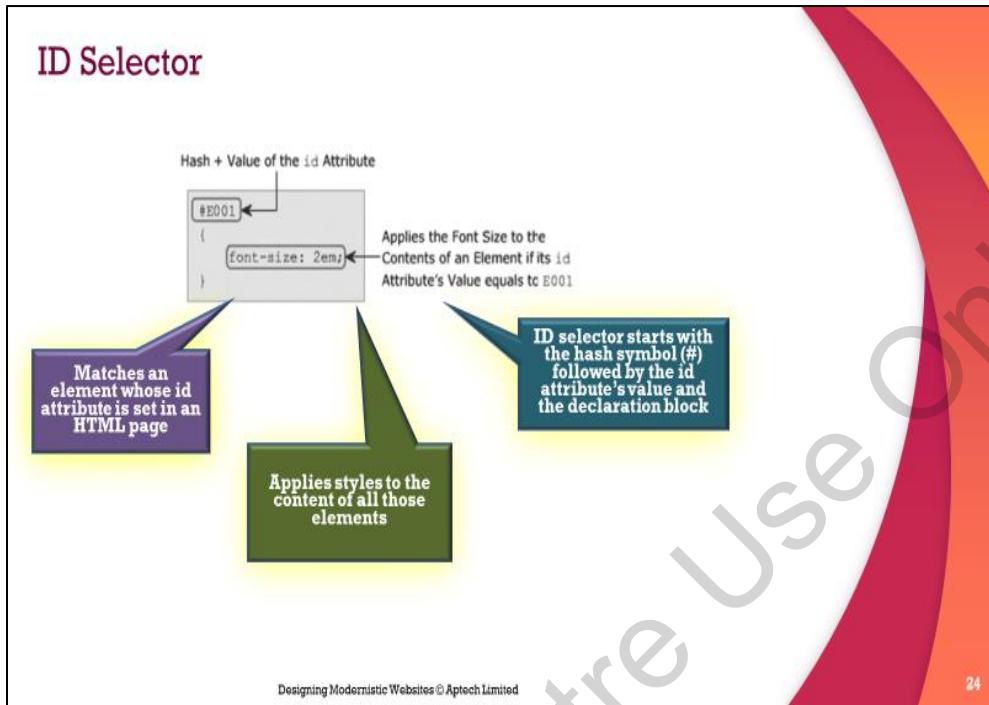
- Using Slide 22, explain the type selector.
- A type selector simply specifies the element name along with the styles to be applied to that element. This results in applying the specified styles to all the occurrence of that element in a Web page.
- Here, the styles are specified only once for an HTML element and are applied to all the occurrences of that element.
- Following example matches all H1 elements on the page:  
`h1 { font-family: sans-serif }`



**Instructions to the Trainer(s):**

- Using Slide 23, explain the class selector.
- A class selector matches elements whose `class` attribute is set in an HTML page and applies styles to the content of all those elements. For example, if there are `<span>` and `<div>` elements in a Web page with their class attributes set, the style specified for the class selector will be applied to both the elements.
- For example, the following code shows the matching class selectors for `<div>` and `<span>`.

```
<html>
<head>
    <style>
        .foo {
            font-size: 1.8 em;
        }
    </style>
</head>
<body>
    <span class="foo">Matches Span </span>
    <div class="foo" title="Help">Matches Div </div>
</body>
```



**Instructions to the Trainer(s):**

- Using Slide 24, explain the ID selector.
- An ID selector matches an element whose id attribute is set in an HTML page and applies styles to the content of that element.
- The ID selector specifies styles for an element whose id attribute is set to a unique value.
- An ID selector starts with the hash symbol (#) followed by the id attribute's value and the declaration block.
- The additional example for ID selector is as follows:

```
<div id="top">
    <h1>About Us</h1>
    <p class="intro"> We are first in IT Industry to introduce
the grid computing.
    </p>
    <p class="intro"> Go through our services offered in grid
computing. </p>
</div>
```

Thus, the css for the mentioned Web page is as follows:

```
#top {
    background-color: #ccc; padding: 20px
}
.intro {
    color: red;
    font-weight: bold;
}
```

## Universal Selector

```
* {  
    font-family: Verdana, Calibri, sans-serif;  
}
```

Can be applied to all elements in the document

Applies the specified styles to the content of all the elements

Represented by an asterisk (\*) sign

Designing Modernistic Websites © Aptech Limited

25

The slide has a decorative background with red and orange curved stripes on the right side. The title 'Universal Selector' is at the top left. Below it is a code snippet: '\* { font-family: Verdana, Calibri, sans-serif; }'. Three callout boxes point to different parts of the slide: one to the code with the text 'Can be applied to all elements in the document'; another to the code with 'Applies the specified styles to the content of all the elements'; and a third to the asterisk (\*) symbol with 'Represented by an asterisk (\*) sign'. At the bottom left is the text 'Designing Modernistic Websites © Aptech Limited' and at the bottom right is the number '25'.

**Instructions to the Trainer(s):**

- Using Slide 25, explain universal selector concept to students.
- The universal selector can be applied to all elements in the document.
- This means that it applies the specified styles to the content of all the elements.
- It is represented by an asterisk (\*) sign.
- For example, universal selector is used to define the font family for all the elements, as shown in code snippet on the slide.

## Generic Cascading Order 1-2

- W3C has defined some rules for applying styles to an HTML element. These rules are:

Gather all the styles that are to be applied to an element.

Sort the declarations by the source and type of style sheet. The source specifies the origin from where the styles are rendered.

Highest priority is given to the external style sheet defined by an author. The next priority is of the reader, which can be a software that reads the content, and the last priority is of the browser.

Sort the declarations by the priority of a selector, where the ID selector has the highest priority.

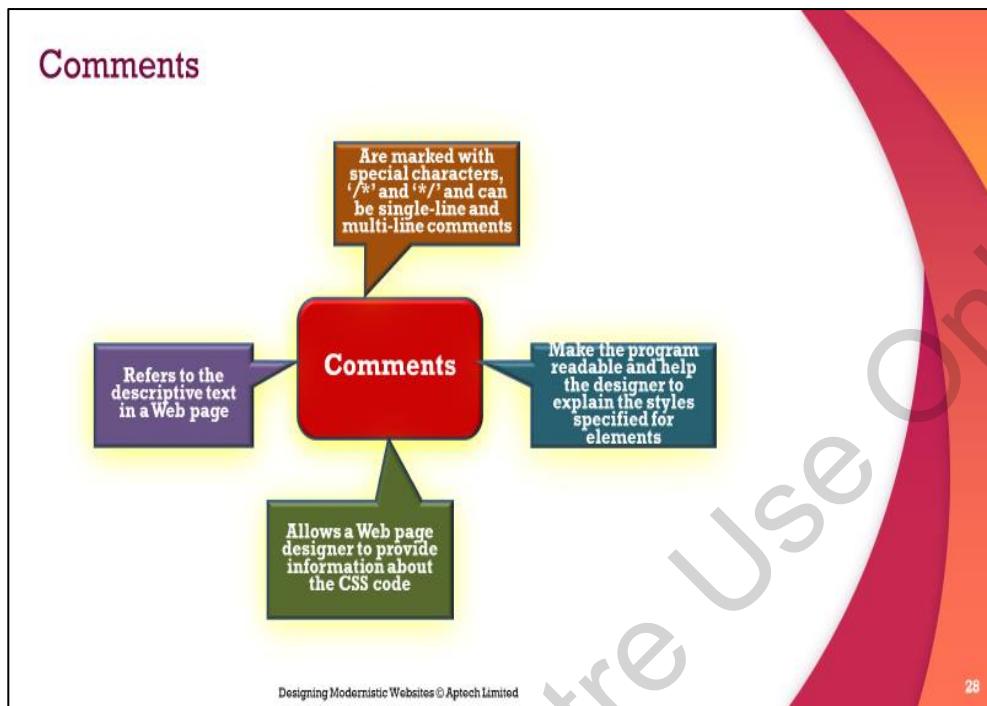
Sort the declaration according to the specified order.

## Generic Cascading Order 2-2

	Lowest Priority	Highest Priority		
	Source	Browser	Reader	Author
CSS Type	External	Internal	Inline	
Selector	Type	Class	ID	

**Instructions to the Trainer(s):**

- Using Slides 26 and 27, explain the generic cascading order.
- Consider a scenario where you have multiple style sheets defined for an HTML page.
- These style sheets might have various selectors and multiple styles defined for an HTML element. Therefore, W3C has defined some rules for applying styles to an HTML element.
- These rules are as follows:
  - Gather all the styles that are to be applied to an element.
  - Sort the declarations by the source and type of style sheet. The source specifies the origin from where the styles are rendered.
- The highest priority is given to the external style sheet defined by an author. The next priority is of the reader, which can be a software that reads the content (screen reader software), and the last priority is of the browser.
  - Sort the declarations by the priority of a selector, where the ID selector has the highest priority.
  - Sort the declaration according to the specified order.



**Instructions to the Trainer(s):**

- Using Slide 28, explain the comments in CSS.
- A comment refers to the descriptive text that allows a Web page designer to provide information about the CSS code.
- Comments make the program more readable and help the designer to explain the styles specified for elements. This is helpful when other Web designers analyze the CSS code.
- The browser can identify comments as they are marked with special characters, which are '/\*' and '\*/'. When the browser encounters these symbols, the text within them are ignored and are not displayed in the browser. You can have single-line and multi-line comments in the CSS file.
- Example on how to apply comments in a CSS file:

```
/*This is a multiple lines comment*/ h1
{
color:red;
/* This is to align text*/ text-align:center;
}
```

## Pseudo Classes 1-4

Sometimes, unknowingly, the same Web page is opened that you have already visited.

You might feel the need for a mechanism that could differentiate the already visited links from the remaining ones.

This is possible by using pseudo classes.

Pseudo classes allow the users to apply different styles to the elements such as buttons, hyperlinks, and so on.

- Syntax for declaring Pseudo classes are as follows:

```
selector_name:state_name {property: value}
```

Designing Modernistic Websites © Aptech Limited

29

## Pseudo Classes 2-4

- Following table lists the different states of an element:

State	Description
active	Defines a different style to an element that is activated by the user.
hover	Defines a different style to an element when the mouse pointer is moved over it.
link	Defines a different style to an unvisited hyperlink.
visited	Defines a different style to the visited hyperlink.

Designing Modernistic Websites © Aptech Limited

30

## Pseudo Classes 3-4

- Following table lists the selector name and its descriptions:

Selector Name	Description
:link	Is used for selecting all unvisited links
:active	Is used for selecting the active link
:hover	Is used for selecting links on mouse over
:visited	Is used for selecting all visited links
:focus	Is used for selecting the input element which has focus
:first-letter	Is used for selecting the first letter of every <p> element
:first-line	Is used for selecting the first line of every <p> element
:first-child	Is used for selecting every <p> elements that is the first child of its parent
:before	Is used for inserting content before every <p> element
:after	Is used for inserting content after every <p> element

Designing Modernistic Websites © Aptech Limited

31

## Pseudo Classes 4-4

- Pseudo classes specify the styles to be applied on an element depending on its state.
- In CSS3, a selector can contain multiple pseudo-classes.
- These pseudo-classes should not be mutually exclusive.
- Code snippets demonstrates the use of CSS code specifying the different styles for the visited links, unvisited links, and for the links when the mouse hovers over it.

```
a:link {  
color: white;  
background-color: black;  
border: 2px solid white;  
}  
  
a:visited {  
color: white;  
background-color: brown;  
border: 2px solid white;  
}  
  
a:hover {  
color: black;  
background-color: white;  
border: 2px solid black;  
}
```

Specifies the styles  
for an unvisited link

Specifies the styles  
for a visited link

Specifies the styles  
when a mouse hovers  
over it

Designing Modernistic Websites © Aptech Limited

32

### Instructions to the Trainer(s):

- Using Slides 29 to 32, explain the pseudo classes.
- Pseudo classes allow the users to apply different styles to the elements such as buttons, hyperlinks, and so on.

- Explain using a scenario where a Website consists of multiple Web pages linked through hyperlinks. Browse through various Web pages by randomly clicking the links within the main page.
- At times, it might happen that unknowingly the same Web page gets open that you have already visited.
- In such a case, you might feel the necessity for a mechanism that could differentiate the already visited links from the remaining ones. In CSS, this is possible by using pseudo classes.
- Pseudo classes specify the styles to be applied on an element depending on its state. In CSS3, a selector can contain multiple pseudo-classes.
- These pseudo-classes should not be mutually exclusive. For example, the selectors `a:visited:hover` and `a:link:hover` are applicable, but `a:link:visited` is not applicable, because `:link` and `:visited` are mutually exclusive selectors.
- CSS code specifies different styles for the visited links, unvisited links, and for the links when the mouse hovers over it.

## Purpose of Pseudo Elements

- Consider a scenario where you are designing a Website that explains the important technical terms.
- While defining such terms, you might feel the need to emphasize more on the first letter by applying different styles.
- Pseudo elements provide you with a flexibility to apply styles to a specific part of the content such as a first letter or first line.
- Pseudo element adds some special effects to HTML elements such as `<p>`, `<body>`, and so on.
- Syntax for declaring pseudo elements is:

The diagram illustrates the syntax for declaring a pseudo element. It shows a sequence of components: a 'selector\_name' (boxed), a colon, a 'pseudo\_element' (boxed), and a brace followed by '(property: value)' (boxed). Above the sequence, a note states 'Can be either :first-line or :first-letter.' An arrow points from this note to the colon. Below the sequence, two annotations provide further details: 'Is an element name.' points to 'selector\_name', and 'Is any CSS property such as color, border, and font.' points to '(property: value)'.

Designing Modernistic Websites © Aptech Limited

33

### Instructions to the Trainer(s):

- Using Slide 33, explain the purpose of pseudo elements.
- Mention pseudo elements provide you with a flexibility to apply styles to a specific part of the content such as a first letter or first line. This allows you to control the appearance of that specific content without affecting the rest of the content.
- Explain with help of scenario, where you are designing a Website that explains the important technical terms. While defining such terms, you might feel to emphasize more on the first letter by applying different styles. It becomes difficult if you try to apply styles only on the first letter of a line or paragraph. This can be achieved by using the pseudo elements.
- Pseudo element adds some special effects to HTML elements such as `<p>`, `<body>`, and so on.

## Pseudo Elements

The `:first-line` pseudo element allows you to apply styles to the first line.

- The Code Snippet declares the style that will be applied to the first line in the paragraph.

```
p:first-line  
{  
font-family: "Tahoma";  
font-weight: bolder;  
background-color: #FFFFCC;  
}
```

Specifies the styles to be applied to the first line of the paragraph content

The `:first-letter` pseudo element allows you to apply styles to the first letter.

- The Code Snippet declares the style that will be applied to the first letter in the paragraph.

```
p:first-letter  
{  
font-family: "fantasy";  
font-size: xx-large;  
font-weight: bold;  
}
```

Specifies the styles to be applied to the first letter of the paragraph content

Designing Modernistic Websites © Aptech Limited

34

### Instructions to the Trainer(s):

- Using Slide 34, explain the pseudo elements.
- :first-line:** The `:first-line` pseudo element allows you to apply styles to the first line. HTML code where the `:first-line` pseudo element will be used.
  - :first-letter:** The `:first-letter` pseudo element allows you to apply styles to the first letter. HTML code for the `:first-letter` pseudo element is shown in Slide 34.

## Styles to Hyperlink

- CSS can be used to change the appearance and behavior of hyperlinks.
- There are two other ways to assign hyperlink styles namely, div specific, and Link specific.
- A div specific hyperlink styles can be created and assigned to a specific div and will have all the hyperlinks present within the div to follow the specified rules.
- Class specific hyperlink styles generally uses a class than an id. A point to note that an id can only be used once on a page whereas a class can be used multiple times as required.

Designing Modernistic Websites © Aptech Limited

35

### Instructions to the Trainer(s):

- Using Slide 35, explain students how to assign hyperlink styles.
- There are two other ways to assign hyperlink styles. They are as follows:
  1. Div specific
  2. Link specific
- A hyperlink styles can be created and assigned to a specific div. This will have all the hyperlinks present within the div to follow the specified rules. It is irrelevant if the div is an (#) id or (.) class.
- Specific styling can be assigned to a specific type of hyperlink. This is achieved by creating the style rules in the CSS. For this type of hyperlink styling, a class is used generally than an id.
- A point to note that an id can only be used once on a page, whereas a class can be used multiple times as required.

## Summary

- ❖ CSS is a mechanism for adding style such as fonts, colors, and spacing to Web documents. CSS has multiple levels and profiles.
- ❖ The general syntax of CSS consists of three parts namely, selector, property, and value.
- ❖ Selectors refer to the HTML elements with the styles that are applied to them and they can be Type, Class, ID, or Universal selectors.
- ❖ A comment refers to the descriptive text that allows a Web page designer to provide information about the CSS code.
- ❖ Pseudo classes allow the users to apply different styles to the elements such as buttons, hyperlinks, and so on.
- ❖ Pseudo elements allow the developer to apply styles to a specific part of a content such as first letter or first line.
- ❖ A hyperlink style can be assigned either through DIV or through link class.

36

### Instructions to the Trainer(s):

- Show students Slide 36.
- Summarize the session by reading out each point on the slide.

## Session 6: Formatting Using Style Sheets

### 6.1 Pre-Class Activities

Before you commence the session, you should familiarize yourself with the topics of this session in-depth. Prepare a question or two that will be a key point to relate the current session objectives.

#### 6.1.1 Teaching Skills

To teach this session, you should be well versed with CSS properties to format text using various font properties. You should also aware yourself with the paragraph indentation and applying borders to various elements on the Web page.

You should teach the concepts in the theory class using the images provided. For teaching in the class, you are expected to use slides and LCD projectors.

### In-Class Activities

Follow the order given here during In-Class activities.

Slide 2

The slide has a red-to-orange gradient background. In the top-left corner, the word "Objectives" is written in white. Below it is a bulleted list of nine items, each preceded by a small red circle. At the bottom left, there is very small, faint text that appears to be a copyright notice. At the bottom right, there is a small number "2".

Objectives

- List and explain text and font styles
- Describe inline spans
- Explain paragraph indentation and application of border
- Explain horizontal paragraph alignment
- Explain vertical spacing within a paragraph
- Describe selector specificity and pseudo selectors
- Explain box model
- Illustrate the use of positioning and float property

Designing Modernistic Websites © Aptech Limited

2

#### Instructions to the Trainer(s):

Give students a brief overview of the current session through the session objectives listed in Slide 2.

## 6.2 In-Class Explanations

Slide 3

### Introduction

In modern Web designing, style sheets allow various styles for formatting menus, texts, borders, or paragraphs.

Designing Modernistic Websites © Aptech Limited

3

### Instructions to the Trainer(s):

- Using Slide 3, introduce the session to students.
- Explain the students about Web designing.
- Tell the students various CSS properties to format the text using font styles, inline spans, paragraph indentation, borders, and so on.

## Text Properties

Property	Description
color	Specifies the color of the text.
text-align	Specifies the horizontal alignment of text in an element.
text-decoration	Specifies the decoration of the text in an element.
text-indent	Specifies the indentation of first line of text in an element in length or %.
text-transform	Specifies the casing of text in an element.
word-spacing	Increases or decreases the space between words.

Designing Modernistic Websites © Aptech Limited

4

**Instructions to the Trainer(s):**

- Using Slide 4, explain the text properties.
- Mention text properties specify and control the appearance of the text in a Web page.
- Tell the students that they can change the color of a text, increase, or decrease the space between characters, align a text, and so on using the CSS text properties.

## Font Properties

Property	Description
font-family	Specifies the font and generic family or a specific family name such as 'Serif' or 'Times New Roman'.
font-size	Specifies the size of the font and can have an absolute or relative value.
font-style	Specifies the style of the font.
font-variant	Specifies whether the text should be displayed in small-caps.

Designing Modernistic Websites © Aptech Limited

5

### Instructions to the Trainer(s):

- Using Slide 5, explain the font styles.
- Mention font properties allow you to specify the font for the text. They allow you to change different font attributes of the text such as font, size, and style of the text.
- The browser must support the font specified by the font properties. Otherwise, it will display the default font, which is dependent on the browser.
- Explain the properties of the font styles with its description mentioned on Slide 5.

## Slides 6 to 8

### Text Styles 1-3

**text-align Property**

Property	Description
left	Aligns the text to the left of the Web page.
right	Aligns the text to the right of the Web page.
center	Aligns the text in the middle of the Web page.
justify	Justifies the text on both sides of the Web page.

**text-indent Property**

Value	Description
length	Specifies fixed indentation. The default value is 0.
%	Specifies an indentation as a percentage of the width of the parent element. The parent element is the element within which the selector element is defined.

**text-transform Property**

Value	Description
none	Specifies that the text will be displayed with the same casing as written within the element.
capitalize	Specifies that the first letter of each word will be capitalized.
uppercase	Specifies only uppercase letters.
lowercase	Specifies only lowercase letters.

Designing Modernistic Websites © Aptech Limited

6

### Text Styles 2-3

```
<!DOCTYPE HTML>
<html>
<head>
<link rel="stylesheet" type="text/css" href="TextProperties.css"/>
</head>
<body>
<h2>Client Contact Information</h2>
<div>
<h4>Dynamic Solutions</h4>
<p>Tel Number - 445 558 7744</p>
<p>Fax Number - 703 740 6539</p>
</div>
</body>
</html>
```

**Output**

**Client Contact Information**

DYNAMIC SOLUTIONS

TEL NUMBER - 445 558 7744

FAX NUMBER - 703 740 6539

Designing Modernistic Websites © Aptech Limited

7

## Text Styles 3-3

text-decoration Property	
Value	Description
none	Displays normal text without any formatting.
underline	Displays a line under the text.
overline	Displays a line over the text.
line-through	Displays a line through the text.
blink	Flashes the text.

word-spacing Property	
Value	Description
normal	Specifies normal spacing between words and it is the default value.
length	Specifies fixed space between words.

```
<!DOCTYPE HTML>
<html>
<head>
<link rel="stylesheet" type="text/css"
 href="ParaProperties.css"/>
<title>Solar System</title>
</head>
<body>
<h3>Nine Planets</h3>
<div>
<p>Mercury, Venus, Earth, Mars, Jupiter, Saturn,
Uranus, Neptune, Pluto</p>
</div>
</body>
</html>
```

**Output**

**Nine Planets:**

Mercury, Venus, Earth, Mars, Jupiter, Saturn,  
Uranus, Neptune, Pluto

Designing Modernistic Websites © Aptech Limited

8

### Instructions to the Trainer(s):

- Using Slides 6 to 8, explain text styles.
- Different text styles such as `text-align`, `text-indent`, and `text-transform` provide different values that allow specifying the alignment, indentation, and casing of text in an element. The `text-align` property allows the text to be centered, or left or right aligned, or justified.
- Explain with the help of table, different values for `text-align`.
- Using Slide 6, explain `text-indent` property and `text-transform` property.
- Explain the `text-indent` property is used for specifying the indentation of the text using the table.
- Also, explain the `text-transform` property is used for changing the case of letters in a text using the table.

### In-Class Question:

**Question:** Which property can be applied to make the text flash on the Web page?

**Answer:** `text-decoration` property with `blink` as value is used for making the text blink or flash on the Web page.

## Slides 9 and 10

### Inline Span 1-2

Attribute	Value	Description
class	classname	Specifies the text direction for the content in an element.
dir	rtl ltr	Specifies the text direction for the content in an element.
id	id	Specifies a unique id for an element.
lang	language_code	Specifies a language code for the content in an element.
style	style_definition	Specifies an inline style for an element.
title	text	Specifies extra information about an element.
xml:lang	language_code	Specifies a language code for the content in an element, in XHTML documents.

Different Attributes and Values Used in <Span> Tag

Designing Modernistic Websites © Aptech Limited

9

### Inline Span 2-2

```
<p>My mother has <span style="color: lightblue">light blue</span>  
eyes.</p>  
  
Or  
  
<span class="eyesonly">light blue</span>
```

My mother has **blue** eyes.

Output

Designing Modernistic Websites © Aptech Limited

10

### Instructions to the Trainer(s):

- Using Slides 9 and 10, explain the inline span content to the students.

- The `<span>` tag groups inline-elements in a document. For example, if one word in a sentence must be bold or colored without using the `<b>` tag, then a `<span>` tag is used which can be present within an existing tag.
- Mention `<span>` tag provides no visual change by itself. The `<span>` tag provides a way to add a hook to a part of a text or a part of a document.
- Explain the code snippet given in the slide. Also, explain the attributes and values of `span` tag and its description.

## Indenting Paragraph 1-3

- Indenting sets off the text from its normal position, either to the left or to the right.
- Three types of indentation: First line indent, Padding, and Margin

```
<!DOCTYPE HTML>
<html>
  <head>
    <title>Font Gallery</title>
  <style>
    p {text-indent: 150px}
  </style>
  </head>
  <body>
    <p>The font styles properties allow you to specify the font for the text. They allow you to change the different font attributes of the text such as font, size, and style of the text. The browser must support the font specified by the font properties. Otherwise, it will display the default font, which is dependent on the browser.
    </p>
  </body>
</html>
```

Designing Modernistic Websites © Aaptch Limited

11

## Indenting Paragraph 2-3

```
<!DOCTYPE HTML>
<html>
  <head>
    <title>Font Gallery</title>
  <style>
    p {padding: 20px }
  </style>
  </head>
  <body>
    <p>
```

The font styles properties allow you to specify the font for the text. They allow you to change the different font attributes of the text such as font, size, and style of the text. The browser must support the font specified by the font properties. Otherwise, it will display the default font, which is dependent on the browser.

Padding  
The font styles properties allow you to specify the font for the text. They allow you to change the different font attributes of the text such as font, size, and style of the text. The browser must support the font specified by the font properties. Otherwise, it will display the default font, which is dependent on the browser.

### Output of Padding Property

Designing Modernistic Websites © Aaptch Limited

12

## Indenting Paragraph 3-3

```
Inline style  
<p style="margin: 20px">  
Internal CSS  
p {margin: 20px}
```

margin

The font styles properties allow you to specify the font for the text. They allow you to change the different font attributes of the text such as font, size, and style of the text. The browser must support the font specified by the font properties. Otherwise, it will display the default font, which is dependent on the browser.

### Margin Property

Designing Modernistic Websites © Aptech Limited

13

#### Instructions to the Trainer(s):

- Using Slides 11 to 13, explain indenting paragraph. Indenting is the process of setting off the text from its normal position, either to the left or to the right. In paragraph style, there are three types of indentation:
- **First line indent** - The `text-indent` property is used in the CSS for indenting the first line of a paragraph.
- Code Snippet demonstrates inline style for `<p>` tag and an internal CSS code for first line indent.
- Tell the students that using Slide 12 they will understand the padding property which is used to add a specified amount of space between the border of an element and its contents. The padding clears an area around the content (inside the border) of an element. The padding is affected by the background color of the element. The top, right, bottom, and left padding can be changed independently using separate properties.
- Using Slide 12 and 13, explain the code for padding property applied to `<p>` tag.
- The margin property is used to add a specified amount of white space around an element. The margin clears an area around an element (outside the border). The margin does not have a background color and is completely transparent.
- The top, right, bottom, and left margin can be changed independently using separate properties.

#### In-Class Question:

**Question:** Which property is used to specify amount between its border and content of an element?

**Answer:** `padding` property is used to specify amount between its border and content of an element.

## Border Style 1-3

<b>border-style Properties</b>	<b>Description</b>
border-left-style	Sets an element's left border.
border-right-style	Sets an element's right border.
border-top-style	Sets an element's top border.
border-bottom-style	Sets an element's bottom border.

**border-style Properties**

<b>Value</b>	<b>Description</b>
dashed	Specifies a dashed border.
dotted	Specifies a dotted border.
double	Specifies two borders.
groove	Specifies a 3D grooved border.
inset	Specifies a 3D inset border.
outset	Specifies a 3D outset border.
ridge	Specifies a ridged border.
solid	Specifies a solid border.

**Values of the border-style Properties**

Designing Modernistic Websites © Aptech Limited

14

## Border Style 2-3

```
<!DOCTYPE HTML>
<html>
<head>
<link rel="stylesheet" type="text/css" href="Styles.css"/>
<title>MagnaSoftwares</title>
</head>
<body>
<div id="heading">
<h2>Welcome to MagnaSoftwares</h2>
</div>
</body>
</html>
```



**Output of border-style Properties**

Designing Modernistic Websites © Aptech Limited

15

## Border Style 3-3

CSS code for shorthand border-style properties

```
<!DOCTYPE HTML>
<html>
<head>
<link rel="stylesheet" type="text/css" href="Styles1.css"/>
<title>Corpse - World's Largest Flower</title>
</head>
<body>
<figure></figure>
<h2>World's Largest Flower </h2>
<p>Corpse flower is the world's
largest flower.<br/> Its diameter is
about a meter. .<br/>
It grows in openings in rainforests on limestone hills of
Sumatra, Indonesia.</ p>
</body>
</html>
```



World's Largest Flower

Corpse flower is the world's largest flower.  
Its diameter is about a meter.  
It grows in openings in rainforests on limestone hills of Sumatra, Indonesia.

### Output of Shorthand border-style Properties

Designing Modernistic Websites © Apech Limited

16

#### Instructions to the Trainer(s):

- Using Slides 14 to 16, explain the border styles.
- Mention borders are rectangular outlines that surround an element. Borders present around text and image, emphasize the content inside the text box. CSS border properties specify the style, color, and width of the border.
- Explain the properties of border given in the table.
- Using Slide 14, explain the border-style properties. Explain the value that can be specified to border-style properties along with its description.

## Border Color 1-2

**border-color Properties**

Property	Description
border-bottom-color	Specifies color for the bottom border.
border-left-color	Specifies color for the left border.
border-right-color	Specifies color for the right border.
border-top-color	Specifies color for the top border.

**Values of border-color Properties**

Value	Description
color	Specifies color to be applied to the border by using either the RGB or hexadecimal value, or the color name itself.
transparent	Specifies that the border is transparent.

Designing Modernistic Websites © Aptech Limited

17

## Border Color 2-2

```
<!DOCTYPE HTML>
<html>
<head>
<link rel="stylesheet" type="text/css"
 href="StylesNew.css"/>
<title>HealthCare</title>
</head>
<body>
<div class="tips">
<h2>Five Essential Health Tips</h2>
<ol>
<li>Quit Smoking</li>
<li>Reduce stress</li>
<li>Protect Yourself from Pollution</li>
<li>Avoid Excessive Drinking</li>
<li>Exercise Regularly</li>
</ol>
</div>
</body>
</html>
```

**Five Essential Health Tips**

- 1. Quit Smoking
- 2. Reduce stress
- 3. Protect Yourself from Pollution
- 4. Avoid Excessive Drinking
- 5. Exercise Regularly

**Output**

Designing Modernistic Websites © Aptech Limited

18

**Instructions to the Trainer(s):**

- Using Slides 17 and 18, explain `border-color` property.
- The `border-color` property in CSS applies colors to all the four borders. You can also apply four different colors to four borders.
- There are other border color properties that allow you to individually specify colors of the left, right, top, or bottom border.
- Using Slide 17, explain the `border-color` property value and its description.
- The `border-color` property accepts different color values that determine different shades of color to be applied to the borders. The table lists the values of different `border-color` properties.

## Border Width 1-3

Values of border-width Properties

Property	Description
border-bottom-width	Specifies width of the bottom border.
border-left-width	Specifies width of the left border.
border-right-width	Specifies width of the right border.
border-top-width	Specifies width of the top border.

border-width Properties

Value	Description
medium	Specifies a medium border.
length	Accepts an explicit value that specifies the thickness of border.
thick	Displays a thick border.
thin	Specifies a thin border.

Designing Modernistic Websites © Aptech Limited

19

## Border Width 2-3

```
.banner
{
text-align:center
;
background:#C0C0C0;
border-style:solid;
border-left-style: none;
border-right-style: none;
border-top-width: thick;
border-bottom-width: thick;
font-family: fantasy;
}
```



Output of border-width Properties

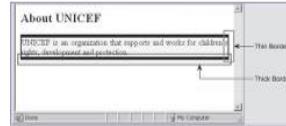
Designing Modernistic Websites © Aptech Limited

20

## Border Width 3-3

Shorthand property: border-width

```
.aboutus  
{  
background-  
color:  
#FFFFCC;  
text-align:  
justify;  
border-style:  
solid;  
border-width: thick thin thick thin;  
}
```



Output of Shorthand Code of border-width Properties

Designing Modernistic Websites © Aptech Limited

21

### Instructions to the Trainer(s):

- Using Slides 19 to 21, explain the border-width property.
- The border-width property is a shorthand property that specifies the width for all the four borders. There are other border-width properties that allow you to individually specify the left, right, top, or bottom borders. Explain the table that lists different border-width properties.
- Using Slide 19, explain the border-width properties to specify the way the border will appear.
- The width of the border can be specified or altered by using the predefined values of the border width properties.
- Mention border-width property is used to set the width of the border. The width is set in pixels or by using one of the three pre-defined values: thin, medium, or thick.
- Using Slide 20, explain the HTML and CSS code for border-width property.

## Shorthand Border 1-2

Value	Description
border-bottom	Specifies width, style, and color for the bottom border.
border-left	Specifies width, style, and color for the left border.
border-right	Specifies width, style, and color for the right border.
border-top	Specifies width, style, and color for the top border.

### Shorthand Border Properties

```
.imppnote
{
background-color: #FFFFCC;
border-top: dashed thin #FF0000; border-bottom:
ridge thick #0000FF; border-right: dotted thin
#FF8040; border-left: inset medium #FF00FF;
}
ul{
list-style: square;
}
```

Note

- Mobility are not allowed during class hours
- Each student should carry his/her identity card/receipt

### Output of Border Properties

Designing Modernistic Websites © Aptech Limited

22

## Shorthand Border 2-2

```
<!DOCTYPE HTML>
<html>
<head>
<title>Flower Gallery</title>
<style>
.flower{
border:solid thin #FF0000;
}
</style>
</head>
<body>
<h2>Flower Gallery</h2>
<table>
<tr>
<td></td>
<td valign="top"><h3>Lilac is a species of flowering plants in the olives family. They are shrubs that range from 2 to 10m in height.</h3>
</td>
</tr>
<tr>
<td></td>
<td valign="top"><h3>Sunflower is a flowering plant whose stem can grow as high as 3m.</h3>
</td>
</tr>
</table>
</body>
</html>
```

Designing Modernistic Websites © Aptech Limited

23

**Instructions to the Trainer(s):**

- Using Slides 22 and 23, explain the shorthand border.
- The border shorthand property in CSS specifies all the properties such as style, width, and color for all the four borders. It allows the user to specify different properties in just one declaration. These properties can be set individually by using different shorthand border properties.
- Explain the table that lists different shorthand border properties. The values of different border properties determine the type of effect to be applied to the borders.
- Mention to shorten the code, it is possible to specify all the individual border properties in one property. This is called a shorthand property.

## Horizontal Alignment

Value	Description
left	Aligns the text to the left.
right	Aligns the text to the right.
center	Centers the text.
justify	Aligns text to both left and right margins by adding space between words (such as in newspapers and magazines).
inherit	Specifies that the value of the text-align property should be inherited from the parent element.

Values of text-align Properties

Designing Modernistic Websites © Aptech Limited

24

### Instructions to the Trainer(s):

- Using Slide 24, explain the horizontal alignment.
- In CSS, text-align property is used for horizontal alignment of text in an element. This property aligns the inline content of a block. Explain the table that lists all values of text-align property.
- The text-align property applies only to block-level elements, such as paragraphs. Hence, text-align cannot change the alignment of a single word without changing the alignment of the entire line.

## Vertical Alignment

Value	Description
normal	A normal line height. This is default.
number	A number that will be multiplied with the current font size to set the line height.
length	A fixed line height in px, pt, cm, and so on.
%	A line height in percent of the current font size.
inherit	Specifies that the value of the line-height property should be inherited from the parent element.

**Values of line-height Properties**

Designing Modernistic Websites © Aptech Limited

25

**Instructions to the Trainer(s):**

- Using Slide 25, explain the vertical alignment.
- In CSS, line-height property is used for vertical alignment of text in an element. This property is also a component of the 'font' shorthand property. It can be applied on block-level elements, table cells, table captions, and so on.

**In-Class Question:**

**Question:** Which property is used to align the content vertically?

**Answer:** line-height property.

## Selector Specificity 1-2

- Selector specificity is the priority given to a selector on which style declarations will be applied.
  - There is a specificity hierarchy for selectors.
  - Four categories define the specificity level of a selector: Inline styles, IDs, Classes, attributes, and Elements and pseudo-classes, and Elements and pseudo-elements

`p {color: yellow;}`

Selector

Designing Modernistic Websites © Aptech Limited

26

## Selector Specificity 2-2

Rules	Example Code Snippet
If the rule is written twice, the lower rule will be applied.	<pre>&lt;!DOCTYPE html&gt; &lt;html&gt; &lt;head&gt; &lt;style&gt; h1 {background-color: red;} h1 {background-color: yellow;} &lt;/style&gt; &lt;/head&gt; &lt;body&gt;&lt;h1&gt;This is my line.&lt;/h1&gt; &lt;/body&gt; &lt;/html&gt;</pre>
ID selectors have a higher specificity than attribute selectors.	<pre>&lt;!DOCTYPE html&gt; &lt;html&gt; &lt;head&gt; &lt;style&gt; div#a {background-color: yellow;} #a {background-color: blue;} div[id=a] {background-color: red;} &lt;/style&gt; &lt;/head&gt; &lt;body&gt;&lt;div id="a"&gt;This is my line.&lt;/div&gt; &lt;/body&gt; &lt;/html&gt;</pre>

Rules	Example Code Snippet
Contextual selectors are more specific than a single element selector.	<pre>CSS file: #content h1 {background-color: blue;}  HTML file: &lt;style&gt; #content h1 { background-color: red; } &lt;/style&gt;</pre>
A class selector is given preference to element selectors.	<pre>&lt;!DOCTYPE html&gt; &lt;html&gt; &lt;head&gt; &lt;style&gt; .intro {background-color: blue;} h1 {background-color: red;} &lt;/style&gt; &lt;/head&gt; &lt;body&gt; &lt;h1 class="intro"&gt;This is my line.&lt;/h1&gt; &lt;/body&gt; &lt;/html&gt;</pre>

Designing Modernistic Websites © Aptech Limited

27

### Instructions to the Trainer(s):

- Using Slides 26 and 27, explain selector specificity.

- Mention that selector specificity is the set of the rules applied to CSS selectors to determine which style is applied to an element.
- Using Slide 27, mention that there are different specificity rules.
- Use the table and explain the specificity rules and discuss the output for each.
- For more information about selector specificity, refer to the following:  
<https://developer.mozilla.org/en-US/docs/Web/CSS/Specificity>

**In-Class Question:**

**Question:** What are the four categories that define the specificity level of a selector?

**Answer:** Inline styles, IDs, Classes, attributes, and pseudo-classes, and Elements and pseudo-elements.

Pseudo-Class Selectors	Example Code Snippet	Output
:hover	<pre>&lt;!DOCTYPE html&gt; &lt;html&gt; &lt;head&gt; &lt;style&gt; div { background-color: green; color: white; padding: 25px; text-align: center; } div:hover { background-color: blue; } &lt;/style&gt; &lt;/head&gt; &lt;body&gt; &lt;div&gt;Place the mouse over here to change the color.&lt;/div&gt; &lt;/body&gt; &lt;/html&gt;</pre>	<p>(Before Mouse Over)</p> <p>Place the mouse over here to change the color</p> <p>(After Mouse Over)</p> <p>Place the mouse over here to change the color</p>
:before	<pre>*para{ font-size: 18px; } *para:before{ content: "- BEFORE -"; background-color: green; } *para:after{ content: "- AFTER -"; background-color: green;</pre>	<p>BEFORE first paragraph AFTER</p>

Designing Modernistic Websites © Aptech Limited

28

### Instructions to the Trainer(s):

- Using Slide 28, explain pseudo selectors.
- Mention that these classes can be used to define the state of an element. For example, fading out an input and so on. Ask the students if they can come up with more examples.
- Use the table on the slide and explain the two pseudo selectors.
- For more information about pseudo selectors, refer to the following: <https://css-tricks.com/pseudo-class-selectors/>

The slide has a red and orange decorative border on the right side. The title 'CSS Combinators' is at the top. A text box contains: 'A combinator indicates the relationship between selectors.' Below it is a code snippet:

```
<html>
<head>
<style> div + p {
background-color: gray;
}
</style>
</head>
<body>
<h2>Example of Adjacent Sibling Selector</h2>
<div>
<p>This is the first line.</p>
<p>This is the second line.</p>
</div>
<p>This is the third line.</p>
<p>This is the fourth line.</p>
<div>
<p>This is the fifth line.</p>
<p>This is the sixth line.</p>
</div>
<p>This is the seventh line.</p>
<p>This is the eighth line.</p>
</body>
</html>
```

A separate box shows the output: 'This is the first line.', 'This is the second line.', **'This is the third line.'**, 'This is the fourth line.', 'This is the fifth line.', 'This is the sixth line.', **'This is the seventh line.'**, 'This is the eighth line.'

At the bottom left is the text 'Designing Modernistic Websites © Aptech Limited'. At the bottom right is the number '29'.

### Instructions to the Trainer(s):

- Using Slide 29, explain CSS combinator.
- Say that a combinator describes the relationship between the selectors.
- There are four combinators:
  - Descendant selector. The symbol is (space).
  - Child selector. The symbol is >
  - Adjacent sibling selector. The symbol is +
  - General sibling selector. The symbol is ~
- Use the code snippet on Slide 29 and the output to show the use of adjacent sibling selector.

## Box Model

- Box model refers to the design and layout of HTML element.
- Includes margins, borders, padding, and content of the element.

```
<!DOCTYPE html>
<html>
<head>
<style> div {
width: 100px;
border: 5px solid red;
padding: 10px; margin: 0;
}
</style>
</head>
<body>
<div>Hi! Check the padding and border</div>
</body>
</html>
```

Output of Code Snippet



Output After Changing Padding Value



Designing Modernistic Websites © Aptech Limited

30

### Instructions to the Trainer(s):

- Using Slide 30, explain box model.
- Say that a box model is a container that includes several properties. It includes borders, margin, padding, and the content.
- Using the code snippet on Slide 30 and outputs, explain box model.
- For more information about the box model, refer to the following:  
[https://developer.mozilla.org/en-US/docs/Web/CSS/CSS\\_Box\\_Model/Introduction\\_to\\_the\\_CSS\\_box\\_model](https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_Box_Model/Introduction_to_the_CSS_box_model)  
[https://developer.mozilla.org/en-US/docs/Web/CSS/CSS\\_Box\\_Model/Mastering\\_margin\\_collapsing](https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_Box_Model/Mastering_margin_collapsing)

### In-Class Question:

**Question:** What surrounds the padding area?

**Answer:** Borders

## Positioning

➤ Use the position property to define the positioning method for an element.  
➤ Syntax: position: static|absolute|fixed|relative|sticky

This line uses the static position.  
This is the second line. When no position is specified, static becomes the default position.

This is the closest parent element.  
This is the relative position.

Static

This is the relative position.

absolute

The menu bar above uses the fixed position  
Scrolling will not affect the bar.  
Scroll down to check.

This is line 1.  
This is line 2.  
This is line 3.

Fixed

Designing Modernistic Websites © Aptech Limited

31

### Instructions to the Trainer(s):

- Using Slide 31, explain students the concept of positioning.
- Mention that the position property can be used to define the positioning method for an element.
- Explain some of the examples given on the slide:
  - Static displays elements in the order given in the document.
  - Absolute positions element relative to its closest positioned ancestor element.
  - Fixed positions element relative to the browser window.
- For more information about positioning, refer to the following:  
<https://css-tricks.com/almanac/properties/p/position/>

## Float

Use the float property to position and format content.

```
<!DOCTYPE html>
<html>
<head>
<style>
img {
    float: left;
}
</style>
</head>
<body>
<p>
The two types of pollination are: self-pollination and cross-pollination. Self-pollination happens when the pollen from the anther is deposited on the stigma of the same flower, or another flower on the same plant. Cross-pollination is the transfer of pollen from the anther of one flower to the stigma of another flower on a different individual of the same species.</p>
</body>
</html>
```

The two types of pollination are: self-pollination and cross-pollination. Self-pollination happens when the pollen from the anther is deposited on the stigma of the same flower, or another flower on the same plant. Cross-pollination is the transfer of pollen from the anther of one flower to the stigma of another flower on a different individual of the same species.

**Output**

Designing Modernistic Websites © Aptech Limited

32

### Instructions to the Trainer(s):

- Using Slide 32, explain the use of float property.
- Mention that the float property can be used to float an element to the left and right.
- For more information about the float property, refer to the following:  
<https://www.geeksforgeeks.org/what-is-float-property-in-css/>

## Summary

- ❖ The text styles specify and control the appearance of the text in a Web page.
- ❖ Indenting is the process of offsetting text from its normal position, either to the left or to the right.
- ❖ CSS border properties specify the style, color, and width of the border.
- ❖ The border-color property accepts different color values that determine different shades of color to be applied to borders.
- ❖ Values of different border properties determine the type of effect to be applied to the borders.
- ❖ In CSS, the text-align property is used for horizontal alignment of text in an element.
- ❖ In CSS, the line-height property is used for vertical alignment of text in an element.
- ❖ Selector specificity can be used to prioritize a selector on which style declarations will be applied.
- ❖ A pseudo-class selector defines a particular state of an element.
- ❖ A combinator indicates the relationship between selectors.
- ❖ It is important to know about box model so that elements are displayed well in all browsers with respect to height and width.
- ❖ The position property can be used to position and format content.

33

### Instructions to the Trainer(s):

- Show students Slide 33.
- Summarize the session by reading out each point on the Slide.

# Session 7: Displaying Graphics and CSS3 Animation

## 7.1 Pre-Class Activities

Before you commence the session, you should familiarize yourself with the topics of this session in-depth. Prepare a question or two that will be a key point to relate the current session objectives.

### 7.1.1 Teaching Skills

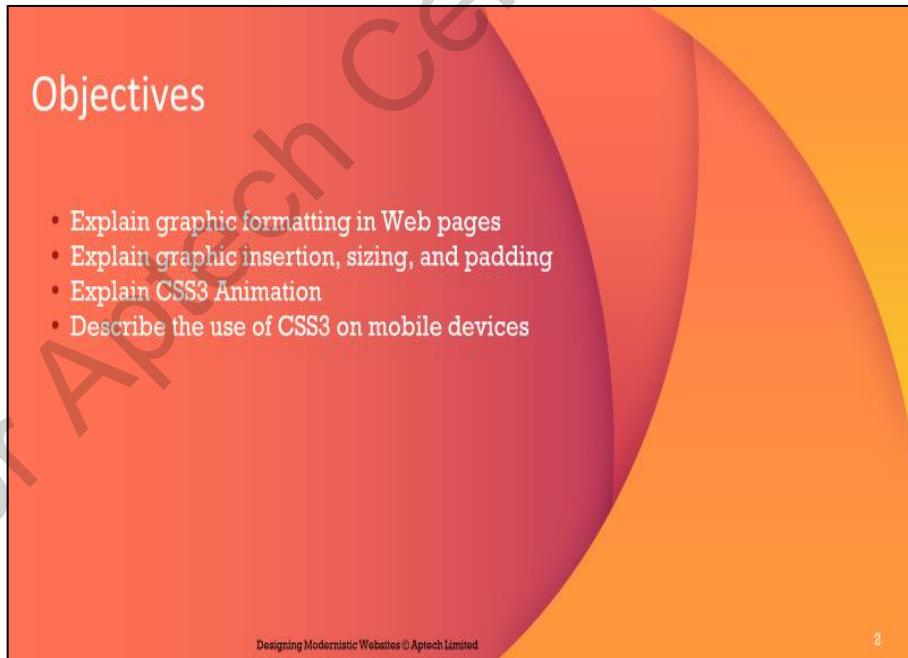
To teach this session, you should be well versed with graphic formatting in Web pages. Also, the graphic insertion, sizing, and padding should be known. Along with this, you should prepare yourself with how to apply CSS3 animation.

You should teach the concepts in the theory class using the images provided. For teaching in the class, you are expected to use slides and LCD projectors.

## In-Class Activities

Follow the order given here during In-Class activities.

Slide 2



### Instructions to the Trainer(s):

Give students a brief overview of the current session through the session objectives listed in Slide 2.

## 7.2 In-Class Explanations

Slide 3

### Introduction

- After release of HTML5 and CSS3 in the market, Web designers have been developing graphics based Web pages.
- CSS3 has allowed designers to style their Web pages graphically with ease.
- Currently, HTML5 applications provide amazing experiences with the use of new CSS3 animations.
- Introduction of mobile applications has allowed users to expand their Web usage to mobile devices.
- CSS3 has introduced new features specifically for mobile devices.



Designing Modernistic Websites © Aptech Limited

3

### Instructions to the Trainer(s):

- Using Slide 3, introduce the session to students.
- Tell the students that this session will introduce them how to apply graphic formatting to the HTML Web pages.
- The session explores CSS3 features to learn how to perform graphic insertion, sizing, and padding.
- They will also learn about the CSS3 animation and the use of CSS3 on mobile devices.

## Graphic Format 1-4

- There are many graphic formats available; the most commonly used are Joint Photographic Experts Group (JPEG), Graphics Interchange Format (GIF), and Portable Network Graphics (PNG).
- The difference between each graphic format depends on the following characteristics:

### ➤ Color Depth

- It is defined by the number of distinct colors that are represented by a hardware or software.
- Color depth is defined by the number of Bits Per Pixel (bpp) and it is also called as bit depth.
- Higher color depth indicates higher range of colors used.

### ➤ Compression/file size

- Compression stores the original images in a reduced number of bytes using an algorithm.
- This image can be expanded back to the original size using a decompression algorithm.

Designing Modernistic Websites © Aptech Limited

4

## Graphic Format 2-4

- Two types of image file compression algorithms used are as follows:

### Lossless compression

- File size is reduced, but preserves a copy of the original uncompressed image.
- Avoids accumulating stages of re-compression when editing images.

### Lossy compression

- A representation of the original uncompressed image is preserved.
- The image appears to be a copy of the original image, but in real it is not a copy.
- Lossy compression achieves smaller file sizes when compared with lossless compression.
- Lossy compression algorithms allow variable compression that comprises on image quality for file size.

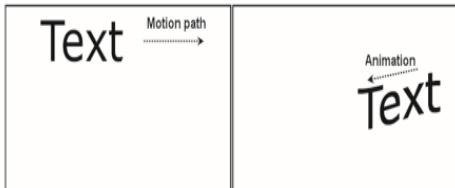
Designing Modernistic Websites © Aptech Limited

5

## Graphic Format 3-4

### ➤ Animation

- Some graphic format consists of a series of frames that are played one after the other giving an impression of animation.
- Following figure shows an animated graphic:



### ➤ Transparency

- It is very common on the Web to display an image on a Web page that appears directly against the background color of the page.
- The background color of the Web page shows through the transparent portion of the image.

Designing Modernistic Websites © Aptech Limited

6

## Graphic Format 4-4

- In a transparent image, one and only one color can be hidden.
- If the color chosen to make transparent is same as the background of the inserted image, then an irregularly shaped image appears to float on the page.
- Following figure shows a transparent image:



Designing Modernistic Websites © Aptech Limited

7

### Instructions to the Trainer(s):

- Using Slides 4 to 7, explain the graphic format.
- Tell the students there are many graphic formats available; the most commonly used are Joint Photographic Experts Group (JPEG), Graphics Interchange Format (GIF), and Portable Network Graphics (PNG).

- The difference between each graphic format depends on the following characteristics:
  - **Color Depth** – It is defined by the number of distinct colors that are represented by a hardware or software. Color depth is defined by the number of Bits Per Pixel (BPP) and it is also called as bit depth. Higher the color depth indicates higher range of colors used. For example, a color depth of 8-bit for GIF image would offer maximum of 256 variations.  
Similarly, a color depth of 24-bit will give 16,000,000 variations.
  - **Compression/file size** – Graphic files are large, so images are compressed using various techniques. Compression stores the original images in a reduced number of bytes using an algorithm. This image can be expanded back to the original size using a decompression algorithm. In some compression formats, images with less complexity results in smaller compressed file sizes.
- Using Slide 5, explain the two types of image file compression algorithms namely, lossless and lossy.
- Using Slides 6 and 7, explain students the animation and transparency graphic format.

**In-Class Question:**

**Question:** What is the effect of a compression done on an image?

**Answer:** Compression stores the original images in a reduced number of bytes using an algorithm.

## Graphic Format for the Web 1-2

- JPEG and PNG graphics provide maximum compatibility with all the devices accessing Web pages.
- For photos, use of JPEG graphic format and for screen-shots and drawings use of PNG graphic format is recommended.

### ➤ JPEG

- Uses a lossy compression which means that image quality is lost in the process of compressing the image.
- For continuous tone pictures such as photos, JPEG should be used.
- Most JPEG editors allow user to specify amount of detail that the user is prepared to lose.
- If the quality is reduced, then loss is visible; JPEG is about half the size of PNG.

### ➤ PNG

- Uses lossless compression, which means there is no loss of any image detail.
- Designed for transferring images on the Internet and not for professional-quality print graphics.
- Therefore, it does not support non-RGB color spaces such as CMYK.
- Supports high color and partial transparency using alpha channels.

Designing Modernistic Websites © Aptech Limited

8

## Graphic Format for the Web 2-2

### ➤ GIF

- Uses a lossless compression which means that there is no loss in quality when the image is compressed.
- The uncompressed image stores its information in a linear fashion.
- Each line of pixels is read from left to right.
- An interlaced GIF file stores the lines of the image in a different order.
- Animated graphics are stored in GIF format.

Compatibility and appearance are the keywords on the Web.

The inserted images must be visible and undistorted when appearing on any recipient's device.

The Web designer can make assumptions that the Website will open in a computer which will have minimum resolution of 800x600 pixel display capability.

If a mobile based Web page needs to be created then the specifications will change.

Designing Modernistic Websites © Aptech Limited

9

**Instructions to the Trainer(s):**

- Using Slides 8 and 9, explain various graphic format available for the Web.
- For Web pages, use of JPEG and PNG graphics are recommended, as it provides maximum compatibility with all the devices that might be accessing the Web page.
- For photos, use of JPEG graphic format and for screen-shots and drawings use of PNG graphic format is recommended.
- Both these formats compress the picture information to reduce the download time and increase the downloading speed.
- The GIF uses a lossless compression where there is no loss of quality when the image is compressed.
- Then, explain the features of each graphic format to the students.

## Graphic Insertion 1-6

The `IMG` element is an empty element, which allows the user to insert an image in a Web page.

It allows insertion of images and diagrams.

The commonly used graphic formats that are supported are namely, GIF, JPEG, BITMAP (BMP), and PNG.

The `<img>` tag reserves a space for the image and does not insert the image in the HTML page.

It creates a link between the image and the HTML page.

## Graphic Insertion 2-6

- Following table lists the commonly used attributes of the `IMG` element:

Data Type	Description
<code>src</code>	Specifies the path of an image that is to be displayed.
<code>height</code>	Specifies the height of an image.
<code>width</code>	Specifies the width of an image.

- Following Code Snippet demonstrates displaying an image using the `IMG` element:

```
<body>

</body>
```

- The code uses the `src` attribute of the `IMG` element to insert a JPEG image.
- The attribute specifies the name of the image and also indicates that the image is present in the same folder where the HTML file is saved.
- The width and height of the image is set to 225 and 151 pixels respectively by using the `width` and `height` attribute.
- A pixel refers to the smallest dot on the monitor screen.

## Graphic Insertion 3-6

- An image can also be stored in a subfolder of the folder containing the HTML file.
- In such cases, a reference to the image is made by using the sub folder name as shown in the following Code Snippet:

```
<body>

</body>
```

- To align the image, the `float` style attribute can be used to specify the inline style for the element.
- This will force the image to be aligned to the left or right side of the screen and wrap the surrounding text around the image.
- Following Code Snippet demonstrates the use of the float style:

```
<body>

</body>
```

Designing Modernistic Websites © Aptech Limited

12

## Graphic Insertion 4-6

- Following table lists the values of `float` property in the `<img>` tag:

Value	Description
left	The element floats to the left.
right	The element floats to the right.
none	The element does not float and is the default value.
inherit	The element specifies that the value of the float property should be inherited from the parent element.

HTML5 introduced a new `<figure>` tag that acts as a container containing the `<img>` tag.

It is not a replacement for `<img>` tag, but acts as a container into which the `<img>` tag is placed.

The `<figure>` tag specifies self-contained content, such as illustrations, diagrams, photos, code listings, and so on.

The content of the `<figure>` element is related to the main flow, its position is independent of the main flow.

Designing Modernistic Websites © Aptech Limited

13

## Graphic Insertion 5-6

- Following Code Snippet demonstrates the use of `<figure>` tag:

```
<figure>

</figure>
```

- The main advantage of using `<figure>` tag is that it allows the user to use `<figcaption>` tag along with it.
- The `<figcaption>` tag allows the user to add a caption to the image.
- The caption always appears along with the image even if the image floats in Web site layout.
- Following Code Snippet demonstrates the use of `<figcaption>` tag:

```
<figure>

<figcaption>This diagram shows the logo of a
      product.</figcaption>
</figure>
```

Designing Modernistic Websites © Aptech Limited

14

## Graphic Insertion 6-6

- The `<figure>` tag can also assign styles and other attributes to the `<figure>` element using an external or internal style sheet.
- A single caption to a group of images can be added using the `<figure>` tag.
- Following Code Snippet demonstrates how to assign a single caption to a group of images:

```
<figure>



<figcaption>The different types of flowers</figcaption>
</figure>
```

- Following figure shows output of a single caption to a group of images:



Designing Modernistic Websites © Aptech Limited

15

### Instructions to the Trainer(s):

- Using Slides 10 to 15, explain the graphic insertion.

- Tell the students that the graphics can be added on the Web pages using `IMG` and `Figure` element.
- The `IMG` element is an empty element, which allows the user to insert an image on a Web page. It allows insertion of images and diagrams. The commonly used graphic formats that are supported are namely, GIF, JPEG, BMP, and PNG.
- The `<img>` tag reserves a space for the image and does not insert the image in the HTML page. It creates a link between the image and the HTML page.
- Use Code Snippet to explain how to display an image in a Web page using the `IMG` element. The code uses the `src` attribute of the `IMG` element to insert a JPEG image. The attribute specifies the name of the image and also indicates that the image is present in the same folder where the HTML file is saved. The width and height of the image is set to 225 and 151 pixels respectively by using the `width` and `height` attribute. A pixel refers to the smallest dot on the monitor screen.
- Using Slide 11, explain the `src` attribute if stored in subfolder of the folder.
- Mention the `alt` attribute which is the other important attribute of the `<img>` tag. The required `alt` attribute specifies an alternate text for an image, if the image cannot be displayed.
- Using Slide 13, explain the `float` property and `<figure>` tag.
- HTML5 introduced a new `<figure>` tag. The `<figure>` tag acts as a container containing the `<img>` tag. In other words, it is not a replacement for `<img>` tag, but acts as a container into which the `<img>` tag is placed. The `<figure>` tag specifies self-contained content, such as illustrations, diagrams, photos, code listings, and so on.
- Using Slide 14, explain the `<figure>` and `<figcaption>` tag.
- To align the image the `float` style attribute can be used to specify the inline style for the element. This will force the image to be aligned to the left or right side of the screen and wrap the surrounding text around the image. Code Snippet demonstrates the use of the `float` style.
- The main advantage of using `<figure>` tag is that it allows the user to use the `<figcaption>` tag along with it. The `<figcaption>` tag allows the user to add a caption to the image. The caption always appears along with the image even if the image floats in Website layout.
- Using Slide 15, explain the `<figure>` tag attributes. The `<figure>` tag can also assign styles and other attributes to the `<figure>` element using an external or internal style sheet. A single caption to a group of images can be added using the `<figure>` tag.

## CSS Image Sizing and Padding 1-3

- Size of an image is specified in pixels.
- The `height` and `width` property sets the height and width of the image.
- One can specify the width and the height will be resized or vice versa.
- Following Code Snippet demonstrates CSS code for setting the image height and width property:

```
p.ex  
{  
height:100px;  
width:100px;  
}
```

Designing Modernistic Websites © Aptech Limited

16

## CSS Image Sizing and Padding 2-3

- Following table lists different CSS properties and values of images:

Property	Description	Values
<code>height</code>	Sets the height of an element	<ul style="list-style-type: none"><li>• Auto</li><li>• Length</li><li>• %</li><li>• inherit</li></ul>
<code>max-height</code>	Sets the maximum height of an element	<ul style="list-style-type: none"><li>• none</li><li>• length</li><li>• %</li><li>• inherit</li></ul>
<code>max-width</code>	Sets the maximum width of an element	<ul style="list-style-type: none"><li>• none</li><li>• length</li><li>• %</li><li>• Inherit</li></ul>
<code>min-height</code>	Sets the minimum height of an element	<ul style="list-style-type: none"><li>• length</li><li>• %</li><li>• Inherit</li></ul>
<code>min-width</code>	Sets the minimum width of an element	<ul style="list-style-type: none"><li>• length</li><li>• %</li><li>• Inherit</li></ul>
<code>width</code>	Sets the width of an element	<ul style="list-style-type: none"><li>• auto</li><li>• length</li><li>• %</li><li>• inherit</li></ul>

Designing Modernistic Websites © Aptech Limited

17

## CSS Image Sizing and Padding 3-3

- Following table lists various values used with height and width properties:

Value	Description
auto	The browser calculates the height and is the default value
length	Defines the length in pixels (px) % Defines the height of the containing block in percent format
length	Defines the length in pixels (px) % Defines the height of the containing block in percent format
inherit	Specifies that the value of the property should be inherited from the parent element

Designing Modernistic Websites © Aptech Limited

18

### Instructions to the Trainer(s):

- Using Slides 16 to 18, explain the CSS image sizing and padding.
- Size of an image is specified in pixels. The height and width property sets the height and width of the image.
- Explain the Code Snippet that demonstrates how to use CSS for setting the image height and width property.
- Using Slides 17 and 18, explain different properties of image tag.

## Padding 1-3

- The CSS padding property is used to specify the space between the element border and the element content.
- The background color of the element affects the padding property.
- Using separate properties such as top, right, bottom, and left, different padding values can be specified and the padding can be changed separately.
- Following table lists various values used in padding property:

Value	Description
length	This property specifies a fixed value for padding in pixels, pt, em, and so on.
%	This property specifies a value for padding in % of the containing element.

## Padding 2-3

- Following Code Snippet demonstrates the CSS code used for specifying different padding values for different sides:

```
padding-top:10px;  
padding-bottom:10px;  
padding-right:15px;  
padding-left:15px;
```

- Instead of using different padding for different sides, users can use a shorthand property.
- A shorthand property is one where all the padding properties for different sides are specified in one property.
- The shorthand property for all the padding properties is padding.
- Following Code Snippet demonstrates the use of the shorthand property for padding:

```
padding:25px 50px 75px 100px;
```

where,

top padding is 25px, right padding is 50px, bottom padding is 75px, and left padding is 100px.

## Padding 3-3

- Following table lists all CSS padding properties:

Property	Description
padding	The browser calculates the height and is the default value
padding-bottom	Defines the length in pixels (px)
padding-left	Defines the height of the containing block in percent format
padding-right	Specifies that the value of the property should be inherited from the parent element
padding-top	Sets the top padding of an element

Designing Modernistic Websites © Aptech Limited

21

### Instructions to the Trainer(s):

- Using Slides 19 to 21, explain padding and shorthand padding.
- The CSS padding property is used to specify the space between the element border and the element content. It is used to separate them from the surrounding element. The background color of the element affects the padding property. Using separate properties such as top, right, bottom, and left, different padding values can be specified and the padding can be changed separately.
- Explain the table provided on the slide that lists various values used in padding property. Also, explain the Code Snippet that demonstrates the CSS code used for specifying different padding values for different sides.
- In the code, the value for padding was set for all the sides. Instead of using different padding for different sides, users can use a shorthand property. A shorthand property is one where all the padding properties for different sides are specified in one property. This will result in a shortened code. The shorthand property for all the padding properties is padding. The property can be used to specify one to four values for each of the side.
- Using Slide 21, explain the list of CSS properties with description given.

### In-Class Question:

**Question:** What is shorthand property for applying padding to an element?

**Answer:** A shorthand property is one where all the padding properties for different sides are specified in one property.

## Thumbnail Graphics 1-4

- A thumbnail is a small image or a part of a larger image.
- Clicking the thumbnail image will link to the larger original image, which can be viewed and downloaded. Even a hover effect can be given through CSS and JavaScript.

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>Thumbnail</title>
    <style>
      body {
        margin:0;
        padding:0px 80px;
        background:#ffff;
        font:70% Arial, Helvetica, sans-serif;
        color:#555;
        line-height:100%;
      }
      h1, h2 {
        font-size:180%;
        font-weight: normal;
        color:#555;
      }
      p {
        margin:1em 0;
      }
      p.text {
        width:500px;
      }
      a{
        color:#f20;
        text-decoration:none;
      }
    </style>
  </head>
  <body>
    <h1>Thumbnail</h1>
    <h2>Following Code Snippet demonstrates an HTML code for inclusion of a thumbnail image:</h2>
    <p>
      <a href="#">
        
      </a>
    </p>
    <p>The image is a thumbnail of the original image, which is 500px wide. The link is styled with a color of #f20 and no underline.</p>
  </body>
</html>
```

22

## Thumbnail Graphics 2-4

```
a:hover{
  color:#999;
}
img{
  border:none;
}
/* thumbnail list */
ul#thumbs, ul#thumbs li{
  margin:0;
  padding:0;
  list-style:none;
}
ul#thumbs li{
  float:left;
  margin-right:0px;
  border:1px solid #999;
  padding:2px;
}
ul#thumbs a{
  display:block;
  float:left;
  width:125px;
  height:135px;
  line-height:50px;
  overflow:hidden;
  position:relative;
  z-index:1;
}
ul#thumbs a img{
  float:left;
  position:absolute;
  top:0px;
  left:0px;
}
```

23

## Thumbnail Graphics 3-4

```
/* mouse over */
ul#thumbs a:hover{
  overflow:visible;
  z-index:1000;
  border:none;
}
ul#thumbs a:hoover img{
  border:1px solid #999;
  background:#fff;
  padding:2px;
}
/* mouse over */
/* clearing floats */
ul#thumbs:after, li#thumbs:after{
  content:"";
  display:block;
  height:0;
  clear:both;
  visibility:hidden;
}
ul#thumbs, li#thumbs{
  display:block;
}
ul#thumbs, li#thumbs{
  min-height:1%;
}
* html ul#thumbs, * html li#thumbs{
  height:1%;
}
```

```
/* clearing floats */
/* thumbnail list */
</style>
</head>
<body>
  <h2>Thumbnail</h2>
  <ul id="thumbs">
    <li><a href="HTML5.png" target="_blank">
      
    </a></li>
  </ul>
</body>
</html>
```

24

## Thumbnail Graphics 4-4

- Following figure shows output of thumbnail with hover effect:



Designing Modernistic Websites © Aptech Limited

25

### Instructions to the Trainer(s):

- Using Slides 22 to 25 explain the students on the use of thumbnail graphics.
- The speed of loading a page of a Website is reduced if high-resolution graphics are used.
- High-resolution graphics are required to improve the effectiveness of the site and cannot be avoided. Hence, to avoid this issue, thumbnails are used.
- A thumbnail is a small image, or a part of a larger image. Clicking thumbnail image will link to the larger original image which can be viewed and downloaded. Even a hover effect can be given through CSS and JavaScript.

## Working with CSS3 Transitions 1-6

- In 2007, Apple introduced the CSS transition, which later became a proprietary feature of Safari called CSS Animation.
- Representatives from Apple and Mozilla began adding the CSS transitions module to the CSS Level 3 specification, closely modeled on what Apple had already added to Webkit and moz.

Browsers that support CSS3 Transitions are as follows:

- Apple Safari 3.1 and later which requires the prefix -webkit-
- Google Chrome which requires the prefix -webkit-
- Mozilla Firefox 3.7 alpha and later which requires the prefix -moz-
- Opera 10.5x and later which requires the prefix -o-

Designing Modernistic Websites © Aptech Limited

26

## Working with CSS3 Transitions 2-6

For performing CSS transitions the two required specifications are as follows:

- The CSS property that needs the effect
- The duration of the effect

- Following Code Snippet demonstrates the use of transition effect on the width property for three seconds:

```
div  
{  
    transition: width 3s;  
    -moz-transition: width 3s; /* Firefox 4 */  
    -webkit-transition: width 3s; /* Safari and Chrome */  
    -o-transition: width 3s; /* Opera */  
}
```

- The effect will start when the specified CSS property changes value.
- The CSS property changes its value typically when a user moves a mouse over an element.
- Thus, the user can set the hover for <div> elements.

Designing Modernistic Websites © Aptech Limited

27

## Working with CSS3 Transitions 3-6

- Following Code Snippet demonstrates setting the hover for <div> elements:

```
div:hover
{
    width:200px;
}
```

- Following table lists all the transition properties:

Property	Description
transition	Is a shorthand property and is used for setting the four transition properties into a single property.
transition-property	Is used for specifying the name of the CSS property for which the transition value is set.
transition-duration	Is used for defining the duration of the transition. Default value is 0.
transition-timing-function	Is used for describing how the speed during a transition will be calculated. Default value is 'ease'.
transition-delay	Is used for defining the start of the transition. Default value is 0.

Designing Modernistic Websites © Aptech Limited

28

## Working with CSS3 Transitions 4-6

- Following Code Snippet demonstrates an HTML and CSS code using all transition properties:

```
<!DOCTYPE html>
<html>
<head>
<style type="text/css">
div
{
    width:100px;
    height:100px;
    background:#000000;
    transition-property:width;
    transition-duration:2s;
    transition-timing-function:linear;
    transition-delay:1s;
    /* Firefox 4 */
    -moz-transition-property:width;
    -moz-transition-duration:2s;
    -moz-transition-timing-function:linear;
    -moz-transition-delay:1s;
```

Designing Modernistic Websites © Aptech Limited

29

## Working with CSS3 Transitions 5-6

```
/* Safari and Chrome */
-webkit-transition-property:width;
-webkit-transition-duration:2s;
-webkit-transition-timing-function:linear;
-webkit-transition-delay:1s;
}
div:hover
{
width:500px;
}
</style> </head>
<body>
<p><b>Note:</b> The example</p>
<div></div>

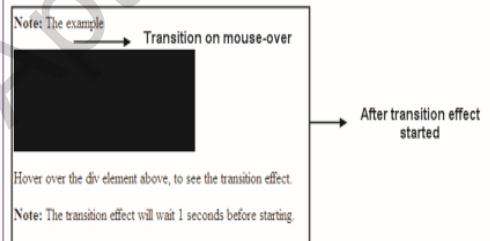
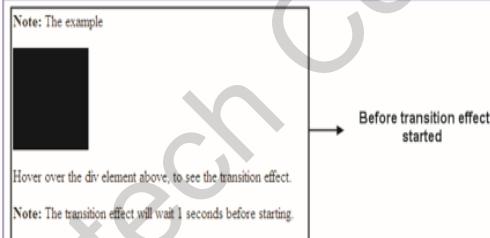
<p>Hover over the div element above, to see the transition effect.</p>
<p><b>Note:</b> The transition effect will wait 1 seconds before starting.</p>
</body></html>
```

Designing Modernistic Websites © Aptech Limited

30

## Working with CSS3 Transitions 6-6

- Following figure shows output of all transition properties:



Designing Modernistic Websites © Aptech Limited

31

### Instructions to the Trainer(s):

- Using Slides 26 to 31, explain the working of CSS3 transition properties.

- Different browsers that support the CSS3 transition properties are Apple Safari, Mozilla, Google Chrome, Opera, and so on.
- Using Slide 28 list and explain different transition properties.
- The transition-duration property specifies how many seconds (s) or milliseconds (ms) a transition effect takes to complete. Always specify the transition-duration property, otherwise the duration is 0, and the transition will have no effect.
- The transition-timing-function property specifies the speed curve of the transition effect. This property allows a transition effect to change speed over its duration.
- Explain the style tag used for the CSS code and the <p>, <div> tag used.
- Using Slide 31 explain the output of the transition properties.

## CSS3 Animation

- CSS3 animations can animate transitions of one CSS style configuration to another.

Two components of animation are as follows:

- An animation style describing the animation.
- A keyframes set that specifies the start and end states of the animation's CSS style and possible intermediate waypoints along the way.

Advantages of CSS3 animations over script-based animation techniques are as follows:

- Easy to use and anybody can create them without the knowledge of JavaScript.
- Executes well even under reasonable system load.
- Allows the browser to control the animation sequence, optimize performance, and efficiency.

Designing Modernistic Websites © Aptech Limited

32

### Instructions to the Trainer(s):

- Using Slide 32, explain the CSS3 animation.
- Tell the students that CSS3 animations can animate transitions of one CSS style configuration to another. The two components of animation are as follows:
  - An animation style describing the animation.
  - A keyframe set that specifies the start and end states of the animation's CSS style and possible intermediate waypoints along the way.
- Some of the advantages of CSS3 animations over script-based animation techniques are as follows:
  - Easy to use and anybody can create them without the knowledge of JavaScript.
  - Executes well even under reasonable system load. As simple animations perform poorly in JavaScript, the rendering engine uses the frame-skipping techniques to allow smooth flow of animation.
  - Allows the browser to control the animation sequence, optimize performance and efficiency by reducing the update frequency of animations executing in tabs that aren't currently visible.

### In-Class Question:

**Question:** What is the keyframes set used for?

**Answer:** Keyframes set specifies the start and end states of the animation's CSS style and possible intermediate waypoints along the way.

## Configuring the Animation 1-6

- A CSS animation sequence can be created by styling the element with the `animation` property.
- This property can be used to configure the timing, duration, and sequence of the animation.
- `@keyframes` rule defines the appearance of the animation.
- The keyframe is used to describe the rendering of the element in the animation sequence.
- Following table lists the `@keyframes` rule and all the animation properties:

Property	Description
<code>@keyframes</code>	Is used for specifying the animation.
<code>animation</code>	Is a shorthand property representing all the animation properties, except the <code>animation-play-state</code> property.
<code>animation-name</code>	Is used for specifying the name of the <code>@keyframes</code> animation.
<code>animation-duration</code>	Is used for specifying the duration of an animation cycle in seconds or milliseconds. Default value is 0.
<code>animation-timing-function</code>	Is used for describing the progress of animation over one cycle of its duration. Default value is 'ease'.

Designing Modernistic Websites © Aptech Limited

33

## Configuring the Animation 2-6

Property	Description
<code>animation-delay</code>	Is used for specifying the start value of animation. Default value is 0.
<code>animation-iteration-count</code>	Is used for specifying the number of times an animation is played. Default value is 1.
<code>animation-direction</code>	Is used for specifying whether or not the animation should play in reverse on alternate cycles. Default value is 'normal'.
<code>animation-play-state</code>	Is used for specifying the state of the animation, that is whether it is running or paused. Default value is 'running'.

- The syntax for `@keyframes` is as follows:

### Syntax:

```

@keyframes myfirst
{
  from {background: red;}
  to {background: yellow;}
}

@moz-keyframes myfirst /* Firefox */
{
  from {background: red;}
  to {background: yellow;}
}

-webkit-keyframes myfirst /* Safari and Chrome */
{
  from {background: red;}
  to {background: yellow;}
}

```

Designing Modernistic Websites © Aptech Limited

34

## Configuring the Animation 3-6

Following Code Snippet demonstrates HTML and CSS code of @keyframes rule and all the animation properties:

```
<!DOCTYPE html>
<html>
<head>
<style type="text/css">
div {
width:200px;
height:200px;
background:red;
position:relative;
border-radius:100px;
animation-name:myfirst;
animation-duration:4s;
animation-timing-function:linear;
animation-delay:1s;
animation-iteration-count:infinite;
animation-direction:alternate;
animation-play-state:running;
/* Firefox */
-moz-border-radius:100px;
-moz-animation-name:myfirst;
-moz-animation-duration:4s;
-moz-animation-timing-function:linear;
-moz-animation-delay:1s;
-moz-animation-iteration-count:infinite;
-moz-animation-direction:alternate;
-moz-animation-play-state:running;
```

Designing Modernistic Websites © Aptech Limited

35

## Configuring the Animation 4-6

```
/* Safari and Chrome: */
-webkit-border-radius:100px;
-webkit-animation-name:myfirst;
-webkit-animation-duration:4s;
-webkit-animation-timing-function:linear;
-webkit-animation-delay:1s;
-webkit-animation-iteration-count:infinite;
-webkit-animation-direction:alternate;
-webkit-animation-play-state:running;
}
@keyframes myfirst
{
0% {background:red; left:0px; top:0px;}
25% {background:yellow; left:300px; top:0px;}
50% {background:blue; left:300px; top:300px;}
75% {background:green; left:0px; top:300px;}
100% {background:red; left:0px; top:0px;}
@-moz-keyframes myfirst /* Firefox */
{
0% {background:red; left:0px; top:0px;}
25% {background:yellow; left:300px; top:0px;}
}
```

Designing Modernistic Websites © Aptech Limited

36

## Configuring the Animation 5-6

```
50% {background:blue; left:300px; top:300px;}  
75% {background:green; left:0px; top:300px;}  
100% {background:red; left:0px; top:0px;}  
}  
@-webkit-keyframes myfirst /* Safari and Chrome */  
{  
0% {background:red; left:0px; top:0px;}  
25% {background:yellow; left:200px; top:0px;}  
50% {background:blue; left:200px; top:200px;}  
75% {background:green; left:0px; top:200px;}  
100% {background:red; left:0px; top:0px;}  
}  
</style>  
</head>  
<body>  
<p><b>Note:</b> Animation</p>  
<div></div>  
</body>  
</html>
```

Designing Modernistic Websites © Aptech Limited

37

## Configuring the Animation 6-6

- Following figure shows the output of @keyframes rule and all the animation properties:



Designing Modernistic Websites © Aptech Limited

38

### Instructions to the Trainer(s):

- Using Slides 33 to 38 explain configuring the animation.
- A CSS animation sequence can be created by styling the element with the animation property. This property can be used to configure the timing, duration, and sequence of the animation.

@keyframes rule define the appearance of the animation. The keyframe is used to describe the rendering of the element in the animation sequence.

- Explain the table that lists the @keyframes rule and all the animation properties.
- Mention the @keyframes rule is where the animation is created. Specify a CSS style inside the @keyframes rule and the animation will gradually change from the current style to the new style.
- When an animation is created in the @keyframe rule, you must bind it to a selector, otherwise the animation will have no effect.
- Bind the animation to a selector (element) by specifying at least these two properties:
  - The name of the animation
  - The duration of the animation
- The animation created using @keyframes must be bound with the selector for effective execution. For this, specify the name of the animation and the duration of the animation to the selector.
- Using Slide 38 explain the output of @keyframes rules and animation properties used.

## Using CSS3 on Mobile Devices 1-2

There are different ways to provide Web pages for mobile devices.

The user can make use of style sheet for the handheld devices (all mobile browsers do not recognize it).

iPhone's Safari and Opera's Mini browsers support a new feature of CSS3 called media queries.

These queries allow the user to specify a conditional expression for media type.

- Following Code Snippet shows the use of a conditional expression for displaying a link element where the maximum screen width for mobile devices is 480 pixels:

```
<link rel="stylesheet" href="styles/mobile.css" media="only  
screen and (max-device-width: 480px)"/>
```

- The user can also specify another link element for screen media with a minimum screen width of 481 pixels.
- In other words, the style sheet for this element can be used for standard computer screens.

## Using CSS3 on Mobile Devices 2-2

- Most mobile Websites are created to precede the domain name of the main site with m for example [m.aptech-education.com](http://m.aptech-education.com).
- To detect a mobile device, a Web site can use JavaScript on the client, a scripting language on the server, or Wireless Universal Resource File (WURFL) on the server.

Five ways to provide Web pages for mobile devices are as follows:

- Define a style sheet for mobile devices
- Include a link to a mobile version of the Website
- Use JavaScript to detect mobile devices and redirect
- Use a server-side scripting language to detect and redirect
- Use the WURFL to detect mobile devices

### Instructions to the Trainer(s):

- Using Slides 39 and 40, explain using CSS3 on mobile devices.
- Explain the students that there are different ways to provide Web pages for mobile devices.

- The user can make use of style sheet for the handheld devices (all mobile browsers do not recognize it). iPhone's Safari and Opera's Mini browsers support a new feature of CSS3 called media queries. These queries allow the user to specify a conditional expression for media type. Code Snippet shows the use of a conditional expression for displaying a link element where the maximum screen width for mobile devices is 480 pixels.
- Mention that the user can also specify another link element for screen media with a minimum screen width of 481 pixels. In other words, the style sheet for this element can be used for standard computer screens.
- Mention interfaces similar to native apps, without the restrictions of each platform and without the need to multiply the versions of the app for each one of them in smartphones. This way, apps are easier to develop and they can be more easily cross-platform spread.
- Separate Websites must be developed for mobile devices. The home page of the main site should provide a link that connects to the mobile Website. This technique identifies the mobile device of the user and renders the mobile Website automatically in the best view possible. Most mobile Websites are created to precede the domain name of the main site with m for example m.aptech-education.com To detect a mobile device, a Website can use JavaScript on the client, a scripting language on the server, or Wireless Universal Resource File (WURFL) on the server.
- Then, explain the ways to provide Web pages for mobile devices.

## Coding for Optimum Browser Compatibility 1-3

Web browser compatibility measures are undertaken to provide predictability and consistency across the preferable Web browsers of the targeted end users.

Cross browser compatibility means a Website that is attuned and reliable in looks, layout, color, functionality, interactivity, and proportion.

Cross browser compatibility is across all existing Web browsers, regardless of the browsers' insignificance or popularity differences from version to version.

Multi-browser compatibility is constant and it is functionally rendered across the most commonly used browsers in a client's target market.

HTML5 uses different standards and is supported by various browsers. These browsers provide different versions of support.

Rendering engines are a set of tools that are used in most browsers that supports different HTML features.

## Coding for Optimum Browser Compatibility 2-3

- Some of the rendering engines of different browsers are as follows:

### Gecko

- The Gecko engine is the main engine of Mozilla Firefox and a number of related browsers.
- It has support for various HTML5 features.

### Trident

- The Trident engine is used by different versions of Internet Explorer (IE).
- Currently, HTML5 is not majorly supported by the Trident engine.

### WebKit

- The WebKit engine is supported mainly for the Safari browser used in Apple Macs, iPhones, iPads, and other Apple products.
- This engine is based on the open source KHTML project.

### Presto

- Presto is the engine used in the Opera browsers.
- Opera browsers are considered to be a technically superior browser, but market share of Opera browsers is still low.

## Coding for Optimum Browser Compatibility 3-3

- Best practices for optimum browser compatibility are as follows:

### Test the Website in different browsers

- Review the Web site's appearance and functionality on multiple browsers to ensure that all the users are getting the same experience according to the design.
- Preferably test on different versions of the same browser also as they can show the Website differently.

### Write a good clean HTML code

- To ensure that the page looks same in all browsers is to write Web pages using valid HTML and CSS codes and then, test it in many browsers.
- Using External CSS can help pages render and load faster.

Designing Modernistic Websites © Aptech Limited

43

### Instructions to the Trainer(s):

- Using Slides 41 to 43, explain the optimum browser compatibility.
- Mention Web browser compatibility measures are undertaken by Web developers who are committed to producing Web products that provide predictability and consistency across the preferable Web browsers of the targeted end users.
- Cross browser compatibility means a Website that is attuned and reliable in looks, layout, color, functionality, interactivity, and proportion across all existing Web browsers, regardless of the browsers' insignificance or popularity differences from version to version.
- Multi-browser compatibility is constant and it is functionally rendered across the most commonly used browsers in a client's target market.
- HTML5 uses different standards and is supported by various browsers. These browsers provide different version of support.
- Then, explain different rendering engines used for different browsers.
- Finally, conclude the explanation by describing the best practices for optimum browser compatibility.

## Summary

- ❖ The text styles specify and control the appearance of the text in a Web page.
- ❖ Indenting is the process of offsetting text from its normal position, either to the left or to the right.
- ❖ CSS border property specifies the style, color, and width of the border.
- ❖ The border-color property accepts different color values that determine the different shades of color to be applied to the borders.
- ❖ The values of the different border properties determine the type of effect to be applied to the borders.
- ❖ In CSS, the text-align property is used for horizontal alignment of text in an element.
- ❖ In CSS, the line-height property is used for vertical alignment of text in an element.

44

### Instructions to the Trainer(s):

- Show students Slide 44.
- Summarize the session by reading out each point on the Slide.

## Session 8: Understanding Layouts in CSS3

### 8.1 Pre-Class Activities

Before you commence the session, you should familiarize yourself with the topics of this session in-depth. Prepare a question or two that will be a key point to relate the current session objectives.

#### 8.1.1 Teaching Skills

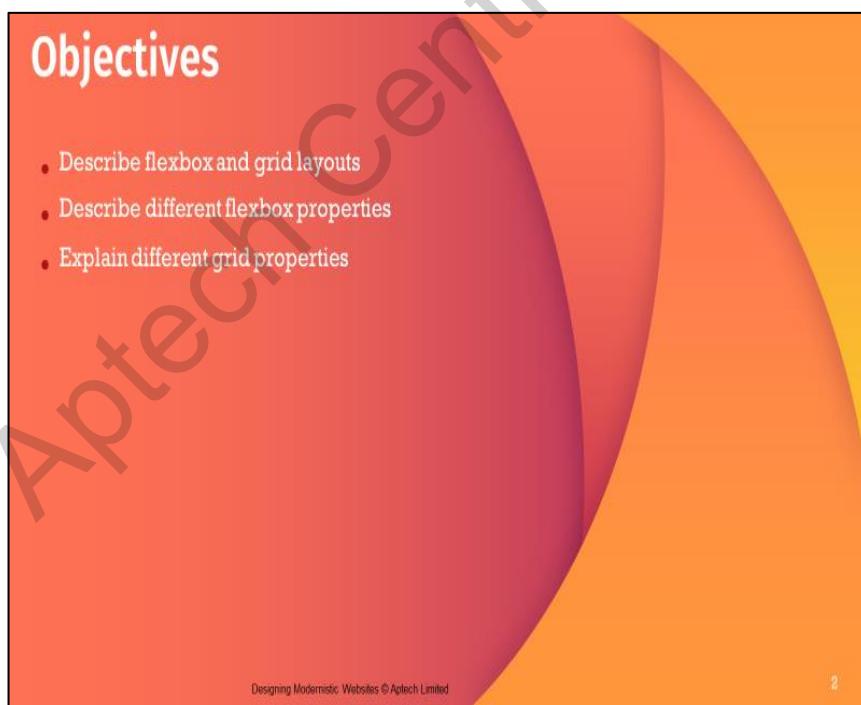
To teach this session, you should be well versed with CSS3 layouts.

You should teach the concepts in the theory class using the images provided. For teaching in the class, you are expected to use slides and LCD projectors.

#### In-Class Activities

Follow the order given here during In-Class activities.

Slide 2

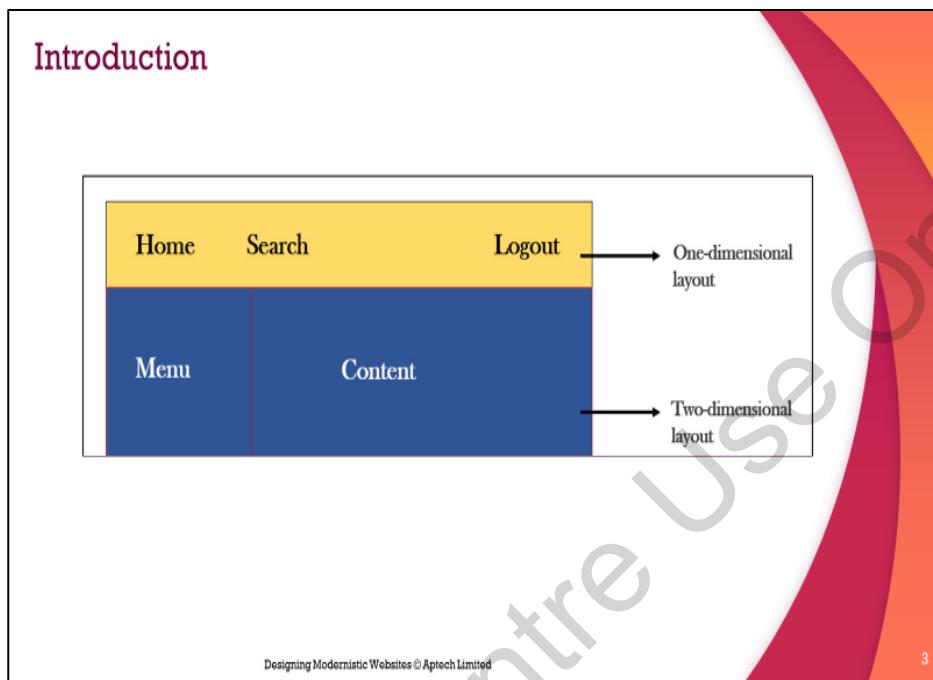


#### Instructions to the Trainer(s):

Give students a brief overview of the current session through the session objectives listed in Slide 2.

## 8.2 In-Class Explanations

Slide 3



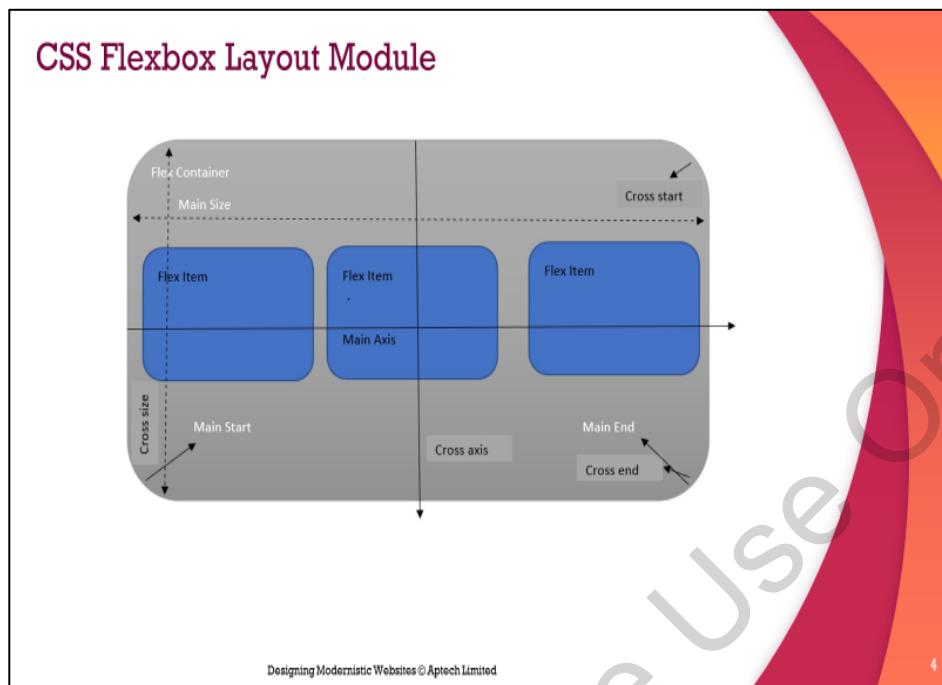
**Instructions to the Trainer(s):**

- Using Slide 3, introduce the session to students.
- Tell the students that Flexbox and Grid are the two popular layouts in the modern CSS.
- Flexbox provides one-dimensional layout and defines layout either for a row or a column.
- Grid provides two-dimensional layout and defines layout of both row and column, simultaneously.

**In-Class Question:**

**Question:** Which one-dimensional layout defines layout either for a row or a column?

**Answer:** Flexbox.



### Instructions to the Trainer(s):

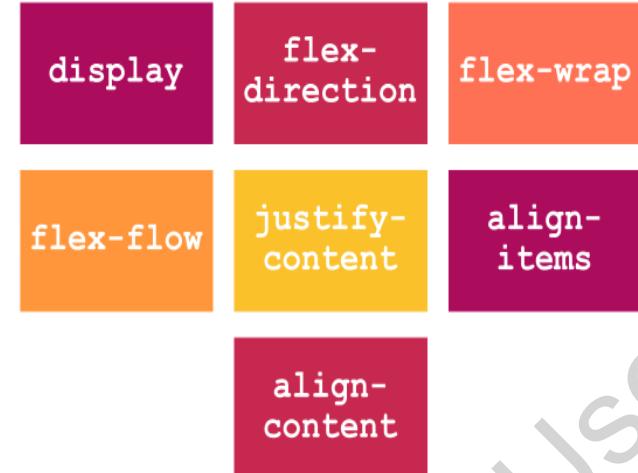
- Using Slide 4, explain the flexbox layout.
- Describe the flexbox components to the students.
  - Flex container is the basic element of the module.
  - All items in a flex container are called flex items. A flex container is a parent element and flex items are its children.
  - The direction of flex items is the main axis. The flex items can be stacked vertically or horizontally. This is the primary axis.
  - Main start and main end are the start and end points for flex items in the flex container.
  - Main size is the width or height of a flex item.
  - Cross axis is the secondary axis and is always perpendicular to the main axis. If the main axis is horizontal, the cross axis is vertical, and vice-versa.
  - The start point of cross axis is cross-start. The end point of cross axis is cross-end.
  - Cross size is the width or height of a flex item in the cross dimension.
- For more information about flexbox layout module, refer to the following:  
[https://developer.mozilla.org/en-US/docs/Web/CSS/CSS\\_Flexible\\_Box\\_Layout/Basic\\_Concepts\\_of\\_Flexbox](https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_Flexible_Box_Layout/Basic_Concepts_of_Flexbox)

### In-Class Question:

**Question:** Which axis is always perpendicular to the main axis?

**Answer:** Cross axis.

### Properties of Flexbox Container 1-3



Designing Modernistic Websites © Aptech Limited

5

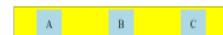
### Properties of Flexbox Container 2-3



Example of Flex  
Direction - column



flex-wrap Property:  
wrap-reverse



space-around Attribute  
Of justify-content  
Property



flex-flow Property: row  
wrap



flex-end Attribute Of  
justify-content  
Property

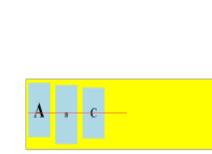


stretch Attribute of  
align-items Property

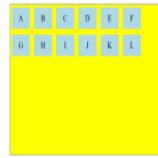
Designing Modernistic Websites © Aptech Limited

6

### Properties of Flexbox Container 3-3



baseline Attribute of align-items Property



flex-start Attribute of align-content Property

Designing Modernistic Websites © Aptech Limited

1

#### Instructions to the Trainer(s):

- Using Slides 5 to 7, explain the properties of flexbox container.
- Tell the students that display defines the type of flexbox of the parent container. In a flexible layout, the display property is set to flex. All the child elements in the container are called the ‘flex items’.
- Flex-direction defines how the flex items should be stacked, whether horizontally or vertically. In vertically stack, which is column, the flex items can be stacked from top to bottom or bottom to top. In a horizontal stack, which is row, the flex items can be stacked from left to right or right to left.
- Flex-wrap specifies the wrapping of flex items. You can keep all flex items on one line or wrap flex items on multiple lines, from top to bottom. You can also wrap flex items on multiple lines from bottom to top.
- Using flex-flow, both flex-direction and flex-wrap properties can be set together.
- justify-content aligns flex items at the center, beginning, or end of the container. Spacing can also be adjusted after, before, or between the flex lines. A flex line refers to the direction of the text.
- align-items property helps align the flex items in the middle, top, or bottom of the container. The flex items can also be stretched to fill the container.
- align-content aligns the flex lines. Spacing can be adjusted before, between, or after the flex lines.
- For more information about flexbox container properties, refer to the following:  
[https://developer.mozilla.org/en-US/docs/Web/CSS/CSS\\_Flexible\\_Box\\_Layout/Basic\\_Concepts\\_of\\_Flexbox](https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_Flexible_Box_Layout/Basic_Concepts_of_Flexbox)

#### In-Class Question:

**Question:** Which property defines the type of flexbox of the parent container?

**Answer:** Display.

### Properties of Flexbox Item 1-4

order

flex-grow

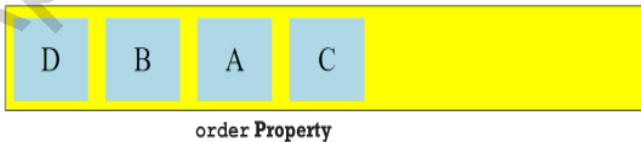
flex-shrink

Designing Modernistic Websites © Aptech Limited

8

### Properties of Flexbox Item 2-4

```
<div class="flex-container">
<div style="order: 3">A</div>
<div style="order: 2">B</div>
<div style="order: 4">C</div>
<div style="order: 1">D</div>
</div>
```

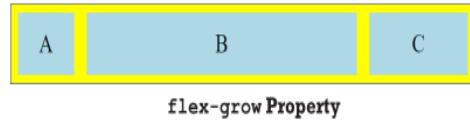


Designing Modernistic Websites © Aptech Limited

9

## Properties of Flexbox Item 3-4

```
<div class="flex-container">
  <div style="flex-grow: 1">A</div>
  <div style="flex-grow: 6">B</div>
  <div style="flex-grow: 2">C</div>
</div>
```



Designing Modernistic Websites © Aptech Limited

10

## Properties of Flexbox Item 4-4

```
<div class="flex-container">
  <div>A</div>
  <div>B</div>
  <div style="flex-shrink: 0">C</div>
  <div style="flex-shrink: 0">D</div>
  <div>E</div>
  <div>F</div>
  <div>G</div>
  <div>H</div>
  <div>I</div>
</div>
```



Designing Modernistic Websites © Aptech Limited

11

**Instructions to the Trainer(s):**

- Using Slides 8 to 11, explain the properties of flexbox item.
- Tell the students that using the properties of flex item, arrangement of flex items can be controlled.
  - order defines the order of the flex items.
  - flex-grow grows a flex item relative to the rest of the flex items.
  - flex-shrink shrinks a flex item relative to the rest of the flex items.
- For more information about flexbox item properties, refer to the following:  
<https://www.learnhowtoprogram.com/user-interfaces/advanced-css-features/customizing-flexbox-flex-item-properties>

**In-Class Question:**

**Question:** Which property allows to specify flex-grow, flex-shrink, and flex-basis properties altogether?

**Answer:** Flex.

Slides 12 and 13

## CSS Grid Layout Module 1-2

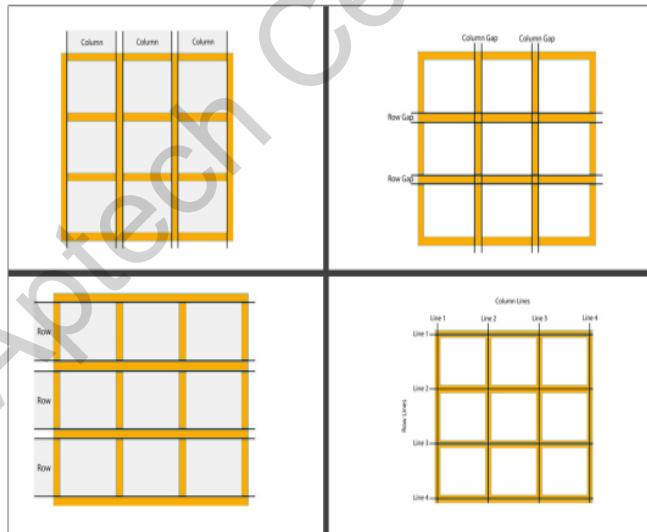
A	B	C
D	E	F

Grid Layout

Designing Modernistic Websites © Aptech Limited

12

## CSS Grid Layout Module 2-2



Designing Modernistic Websites © Aptech Limited

13

**Instructions to the Trainer(s):**

- Using Slides 12 and 13, explain grid layout.
- Tell the students that a grid layout has one parent element with one or more child elements.
- Items in a grid container are called grid items, which are placed inside columns and rows.
- Using Slide 13, explain students the certain components of grid.
  - Grid container holds elements of a grid such as row.
  - Child elements of a grid container are called grid items. A grid container is the parent of all grid items.
  - Grid lines are the dividing lines and make the structure of grid. Vertical grid line refers to column grid line and horizontal grid line refers to row grid line.
  - Grid cell is the single unit of a grid, bound by two row grid lines and two column grid lines.
  - Grid track is a generic term for a grid column or row.
  - Grid area includes grid cells.
  - Grid gap exists between rows and columns.
  - Grid column is the space between two adjacent vertical grid lines.
  - Grid row is the space between two adjacent horizontal grid lines.
- For more information about grid layout module, refer the following:  
<https://webkit.org/blog/7434/css-grid-layout-a-new-layout-module-for-the-web/>

**In-Class Question:**

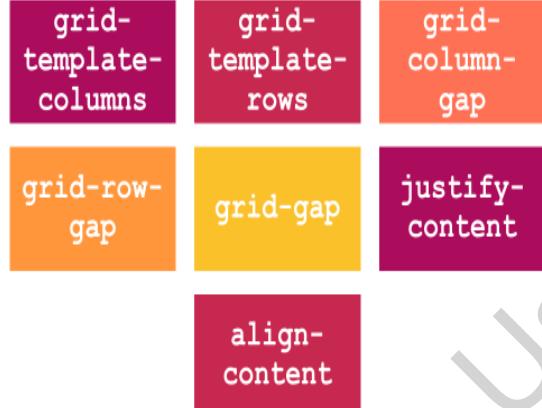
**Question:** What is the term that includes grid cells?

**Answer:** Grid area.

**Question:** What is the parent of all grid items called?

**Answer:** Grid container.

## Grid Container Properties 1-2



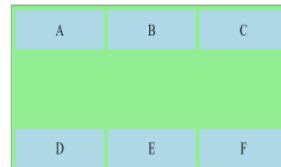
Designing Modernistic Websites © Aptech Limited

14

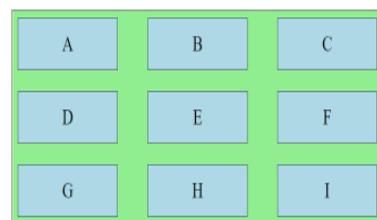
## Grid Container Properties 2-2



justify-content, grid-template-columns,  
and grid-gap Properties



align-content Property



grid-gap Property

Designing Modernistic Websites © Aptech Limited

15

**Instructions to the Trainer(s):**

- Using Slides 14 and 15, explain the grid container properties.
- Tell the students that layout of the grid can be controlled using the grid container properties.
  - grid-template-columns defines number and size of columns.
  - grid-template-rows defines height of rows.
  - grid-column-gap defines size of the gap between columns.
  - grid-row-gap defines size of the gap between rows.
  - grid-gap allows to use both grid-row-gap and grid-column-gap properties together.
  - justify-content helps align grid items along the inline row axis.
  - align-content helps vertically align whole grid along the block column axis.

**In-Class Question:**

**Question:** Which property can be used to give rows equal amount of space between and around them?

**Answer:** space-evenly.

## Grid Items 1-4

grid-column  
grid-row  
grid-area

Designing Modernistic Websites © Aptech Limited

16

## Grid Items 2-4

```
<!DOCTYPE html>
<html>
<head>
<style>
.grid-container {
display: grid;
grid-template-columns: auto auto auto auto auto auto;
grid-gap: 10px;
background-color: lightgreen;
padding: 10px;
}
.grid-container > div {
background-color: lightblue;
text-align: center;
padding: 20px 0;
font-size: 30px;
}
.item2 {
grid-column: 2 / span 5;
}
</style>
</head>
```

```
<body>
<div class="grid-container">
<div class="item1">A</div>
<div class="item2">B</div>
<div class="item3">C</div>
<div class="item4">D</div>
<div class="item5">E</div>
<div class="item6">F</div>
<div class="item7">G</div>
<div class="item8">H</div>
<div class="item9">I</div>
<div class="item10">J</div>
<div class="item11">K</div>
<div class="item12">L</div>
</div>
</body>
</html>
```

A	B				
C	D	E	F	G	H
I	J	K	L		

grid-column Property

17

## Grid Items 3-4

```
<!DOCTYPE html>
<html>
<head>
<style>
.grid-container {
display: grid;
grid-template-columns: auto auto auto auto auto auto;
grid-gap: 10px;
background-color: lightgreen;
padding: 10px;
}
.grid-container > div {
background-color: lightblue;
text-align: center;
padding: 20px 0;
font-size: 30px;
}
.item1 {
grid-row: 1 / span 2;
}
</style>
</head>
```

```
<body>


A



B



C



D



E



F



G



H



I



J

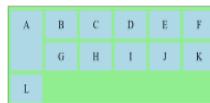


K



L


</body>
</html>
```



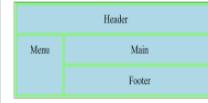
grid-row Property

Designing Modernistic Websites © Aptech Limited

18

## Grid Items 4-4

```
<!DOCTYPE html>
<html>
<head>
<style>
.item1 { grid-area: header; }
.item2 { grid-area: menu; }
.item3 { grid-area: main; }
.item4 { grid-area: footer; }
.grid-container { display:
grid;
grid-template-areas:
'header header header header header' 'menu menu main main main'
'menu footer footer footer footer',
grid-gap: 10px;
background-color: lightgreen;
padding: 10px;
}
</style>
</head>
```



grid-area Property

Designing Modernistic Websites © Aptech Limited

19

### Instructions to the Trainer(s):

- Using Slides 16 to 19, explain the grid item properties.
- Tell the students that using the grid item properties, grid items can be styled.
- grid-column specifies column(s) to place an item and where the item will start and end.
- grid-row specifies the row to place an item and where the item will start and end.
- grid-area helps use grid-row-start, grid-column-start, grid-row-end, and grid-column-end properties together. The properties help specify where to start and end the grid item with respect to column and row.

## Summary

- ❖ The text styles specify and control the appearance of the text in a Web page.
- ❖ Indenting is the process of offsetting text from its normal position, either to the left or to the right.
- ❖ CSS border property specifies the style, color, and width of the border.
- ❖ The border-color property accepts different color values that determine the different shades of color to be applied to the borders.
- ❖ The values of the different border properties determine the type of effect to be applied to the borders.
- ❖ In CSS, the text-align property is used for horizontal alignment of text in an element.
- ❖ In CSS, the line-height property is used for vertical alignment of text in an element.

44

### Instructions to the Trainer(s):

- Show students Slide 20.
- Summarize the session by reading out each point on the Slide.

# Session 9: HTML Layout, Forms, and Elements

## 9.1 Pre-Class Activities

Before you commence the session, you should familiarize yourself with the topics of this session in-depth. Prepare a question or two that will be a key point to relate the current session objectives.

### 9.1.1 Teaching Skills

To teach this session, you should be well versed with HTML5 semantic tags. Also, the HTML5 semantic tag layouts and the usage of navigation bar should be known. Along with this, you should prepare yourself with text-based and graphical navigation bar. The session also covers the image mapping, divisions, and forms in HTML5.

You should teach the concepts in the theory class using the images provided. For teaching in the class, you are expected to use slides and LCD projectors.

## In-Class Activities

Follow the order given here during In-Class activities.

Slide 2

The slide features a red-to-orange gradient background. On the right side, there is a large, semi-transparent orange circle. At the top left, the word "Objectives" is written in white. Below it is a bulleted list of 15 items, each preceded by a small black dot. At the bottom left, the text "Designing Modernistic Websites © Apitech Limited" is visible, and at the bottom right, there is a small number "2".

- Explain HTML5 semantic tags
- Explain HTML5 semantic tag layouts
- Explain the usage of navigation bar
- Describe a text-based and graphical navigation bar
- Explain image mapping
- Explain divisions in HTML5
- Describe HTML5 forms
- Explain the working of new input types in HTML5
- Explain the new Form attributes
- Explain the new Form elements
- Define hidden fields

Instructions to the Trainer(s):

Give students a brief overview of the current session through the session objectives listed in Slide 2.

## 9.2 In-Class Explanations

Slide 3



The screenshot shows a web browser window with a red header bar. The main content area displays a book store interface. At the top, there's a banner for 'WICKED APPETITE' and 'Big Fall Releases' featuring books by ECKMANN TOL, NICHOLAS SPARKS, and JANET EVANOVICH. Below the banner, there's a section titled 'New & Notable' with several book thumbnails. A large watermark reading 'For Aptech Centre Use Only' is diagonally across the page. In the bottom right corner of the slide frame, there's a small number '3'.

Introduction

HTML5 provides:

- Semantic markup for easy understanding.
- New features to make Web forms a lot easier to write.
- Hidden element feature for better security.

Designing Modernistic Websites © Aptech Limited

### Instructions to the Trainer(s):

- Using Slide 3, introduce the session to students.
- Tell the students that they will learn about semantic tags, new features of Web forms, and hidden element features.

## HTML5 Semantic Tags

### Structural Semantic Tags

- Are block level elements
- Structure pages

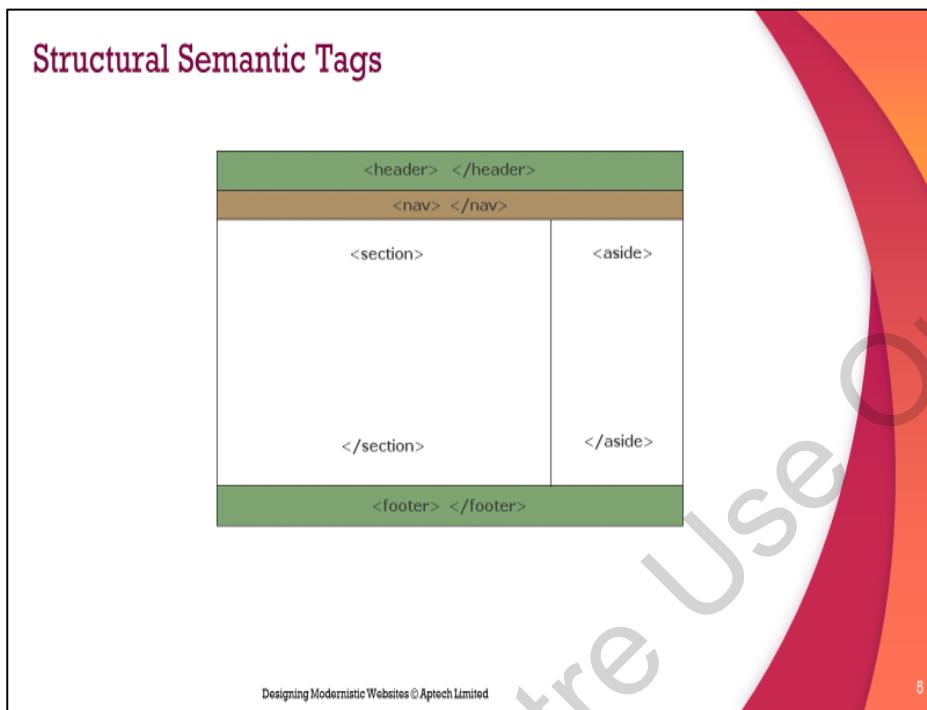
### Text-level Semantic Tags

- Inline elements
- Makes text more expressive

Designing Modernistic Websites © Aptech Limited

**Instructions to the Trainer(s):**

- Using Slide 4, explain the HTML5 semantic tag.
- Mention HTML5 has evolved by introducing new elements that brought semantics to higher level.
- New tags were developed to create stable semantic structure. The earlier version of HTML had the universal tag div which was used to accomplish various tasks in the HTML structure.
- HTML5 has introduced two types of semantic tags. They are namely, text-level and structural.



**Instructions to the Trainer(s):**

- Using Slide 5 explain the structural semantic tags.
- They are the block level elements and are used to structure pages. The new structural semantic elements are as follows:
  - Section
  - Header
  - Footer
  - Aside
  - Nav
  - Article
- All these tags are newly introduced in HTML5.

Text-level Semantic Tags	
Text-level Semantic Tag	Description
<a>	Helps creates hyperlinks, such as to Web pages and files.
<b>	Bolds text.
 	Gives a line break. This is usually used to write an address.
<em>	Emphasizes text.
<mark>	Highlights text for reference or notation purposes
<time>	Indicates a specific period in time. The datetime attribute might be included so that custom features such as reminders can be added.
<u>	Indicates text to be underlined

Designing Modernistic Websites © Aptech Limited

6

**Instructions to the Trainer(s):**

- Using Slide 6, explain the text-level semantic tag.
- The text level semantic tags are currently inline elements and they are as follows:
  - a
  - b
  - br
  - em
  - mark
  - time
  - u

## Navigation Bar 1-8

Text-based

Graphical

Home | News | Contact | About

This is a Text-based Navigation Bar

This is a Graphical Navigation Bar

Designing Modernistic Websites © Aptech Limited

7

## Navigation Bar 2-8

```
<!DOCTYPE html>
<html>
<head>
</head>
<body>
<nav>
<a href="/home/"><font size="6">Home</font></a> | 
<a href="/news/"><font size="6">News</font></a> | 
<a href="/contact/"><font size="6">Contact</font></a> | 
<a href="/about/"><font size="6">About</font></a>
</nav>
<h1>This is a Text-based Navigation Bar</h1>
</body>
</html>
```

Home News Contact About

This is a Text-based Navigation Bar

**Text-based Navigation Bar**

Designing Modernistic Websites © Aptech Limited

8

## Navigation Bar 3-8

```
<!DOCTYPE html>
<html>
<head>
<style>

/* Main Navigation */ #nav {
    position: relative; width: 620px; margin: 0 auto; margin-top: 50px; padding: 10px;
}

ul#navigation { margin: 0px auto; position: relative; float: left; border-left: 1px solid #c4dbe7; border-right: 1px solid #c4dbe7; }

ul#navigation li { display: inline; font-size: 12px; font-weight: bold; margin: 0; padding: 0; float: left; position: relative; border-top: 1px solid #c4dbe7; border-bottom: 2px solid #c4dbe7; }

ul#navigation li a {
    padding: 10px 25px;
}
```

Designing Modernistic Websites © Aptech Limited

9

## Navigation Bar 4-8

```
color: #616161;
text-shadow: lpx lpx 0px #fff; text-decoration: none; display: inline-block;
border-right: 1px solid #fff;
border-left: 1px solid #C2C2C2;
border-top: 1px solid #fff;
background: #f5f5f5;
-webkit-transition: color 0.2s linear, background 0.2s linear;
-moz-transition: color 0.2s linear, background 0.2s linear;
-o-transition: color 0.2s linear, background 0.2s linear;
transition: color 0.2s linear, background 0.2s linear;
}

ul#navigation li a:hover {
background: #f8f8f8;
color: #222222;
}

ul#navigation li a.first {
border-left: 0 none;
}

ul#navigation li a.last {
border-right: 0 none;
}
```

Designing Modernistic Websites © Aptech Limited

10

## Navigation Bar 5-8

```
ul#navigation li:hover>a {  
background:#00FF00;  
}  
/* Drop-Down Navigation */ ul#navigation li:hover>ul  
{  
/*these two styles are very important,  
being the ones which make the drop-down to appear on hover */ visibility:visible;  
opacity:1;  
}  
ul#navigation ul, ul#navigation ul li ul { list-style:none;  
margin:0;  
padding:0;  
  
/*the next two styles are very important,  
being the ones which make the drop-down to stay hidden */ visibility:hidden;  
opacity:0;  
position:absolute;  
z-index: 99999; width:180px; background:#f8f8f8;  
box-shadow:1px 1px 3px #ccc;
```

Designing Modernistic Websites © Aptech Limited

11

## Navigation Bar 6-8

```
/*css3 transitions for smooth hover effect */  
-webkit-transition:opacity 0.2s linear, visibility 0.2s linear;  
-moz-transition:opacity 0.2s linear, visibility 0.2s linear;  
-o-transition:opacity 0.2s linear, visibility 0.2s linear;  
transition:opacity 0.2s linear, visibility 0.2s linear;  
}  
  
ul#navigation ul {  
top:  
43px;  
left: 1px;  
}  
ul#navigation ul li {  
top: 0;  
left: 18px; /* strongly related to width:180px; from above */  
}
```

Designing Modernistic Websites © Aptech Limited

12

## Navigation Bar 7-8

```
ul#navigation ul li
{
    clear:both;
    width:100%;
    border:none;
    border-bottom:1px solid #c9c9c9;
}

ul#navigation ul li a
{
    background:none;
    padding:7px 15px;
    color:#616161;
    text-shadow:1px 1px 0px #fff;
    text-decoration:none;
    display:inline-block;
}
```

Designing Modernistic Websites © Aptech Limited

13

## Navigation Bar 8-8

```
border:none; float:left;
clear:both; width:150px;
}
</style>
</head>
<body>
<nav id="nav">
<ul id="navigation">
- <a href="#"><font size="4">Home</img></font></a></li>
<li><a href="#"><font size="4">News</img></font></a></li>
<li><a href="#"><font size="4">Contact</img></font></a></li>
<li><a href="#"><font size="4">About</img></font></a></li>
</ul>
</nav>
<br/>
<br/>
<br/>
<br/>
<h1>This is a Graphical Navigation Bar</h1>
</body>
</html>

```

Designing Modernistic Websites © Aptech Limited

14

### Instructions to the Trainer(s):

- Using Slides 7 to 14, explain the navigation bar.
- Navigation is one of the most important elements in Web design.
- Web-layouts do not have any specific physical representation that a user can depend on except for a consistent navigation menu. It is one of the most important design elements

which provide the users with a sense of orientation and guide them through the Website. Thus, it can be said that navigation is one segment of a Website's information architecture, but it plays an important role as it is the most visible segment to the end user.

- Mention in Web designing, a navigation menu is always on navigation bars, which can be horizontal or vertical.
- A navigation bar is a section of a Website or online page intended to support visitors in browsing through the online document. Typically, Web pages will have a primary and a secondary navigation bar on all pages which will include links to the most important sections of the Website.
- There are two types of Navigation bar:
  - **Text based:** Text-based navigation bars are not associated with icons, but are easy to create and can be displayed in any Web browsers. The advantage of using a text-based navigation bar is that it reduces the loading time of a page. Although a text-based navigation bar is easy to create, it is not interesting because there is very less interaction or visual appeal to the visitor. Text links are hard to distinguish from the regular text that appears on a Web page. It can be displayed either horizontally or vertically. The font (best to use Web safe fonts), color, and link colors can be determined by the user through the font pane.
  - **Graphical:** Graphical navigation bar is more captivating than text-based navigation bar as it uses icons. The usability of the page increases with a good choice of icon for the navigation bar. It can also make the Website more noticeable for the user visiting the Website. In other words, graphical navigation bar is better than text-based navigation as it gives a visual appeal to the visiting users. The only disadvantage is that, since it uses images, it takes longer time for a page to load. Also, the Web page will be useless for users using a non-graphic browser.

**In-Class Question:**

**Question:** Which element is used for creating the navigation bar in HTML5?

**Answer:** Nav element.

## Image Map

- An image map includes clickable areas.
- The `<map>` tag defines an image map.

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

<map name="cakemap">
<area shape="circle" coords="0,0,200,600" href="4.html"
alt="cake" />
</map>
</body>
</html>
```



Output of Image Map

Designing Modernistic Websites © Aptech Limited

16

### Instructions to the Trainer(s):

- Using Slide 15, explain students the image map concept.
- Image maps are images with clickable areas. These areas in image-maps when clicked will link to another page.
- The image maps have to be used intelligently to make it effective.
- If they are not used appropriately they can confuse the users.
- The `<map>` tag is used to define an image-map.
- The `<map>` element contains a number of `<area>` elements for defining the clickable areas in the image map.
- In HTML5, if the id attribute of the `<map>` tag is specified, then it must have the same value as the name attribute. Using slide 33 explain the attribute of the `<map>` tag.

## Slides 16 and 17

### Divisions 1-2

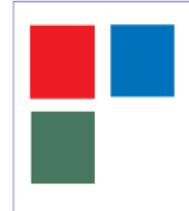
- Division is represented by the `<div>` tag.
- It defines a division or a section in an HTML document.
- The `<div>` tag is:
  - Used as a container for HTML elements. This can be styled with CSS or manipulated with JavaScript.
  - Styled by using the class or id attribute.
- It can hold any content.

Designing Modernistic Websites © Aptech Limited

16

### Divisions 2-2

```
.lCard{ width: 100px;  
height:100px;  
background-color:blue; padding:  
6px; position:fixed; left:450px;  
top:100px;  
}  
.rCard{  
width: 100px; background-  
color:red; padding: 7px;  
position:relative; top:93px;  
left:300px;  
}  
.bCard{ width: 100px;  
height:100px;  
background-color:green; padding:  
6px; position:absolute;  
left:310px; bottom:320px;  
}
```



Output of Division Positioning

Designing Modernistic Websites © Aptech Limited

17

### Instructions to the Trainer(s):

- Using Slides 16 and 17, explain the division in HTML page or `<div>` tag.

- The `<div>` tag can be used when there is no other semantically appropriate element left that suits the purpose in a Web page development.
- It can be commonly used for stylistic purposes such as wrapping some semantically marked-up content in a CSS-styled container.
- Using Slide 17, explain the HTML code for `<div>` tag and tips for using it. Some of the features of `<div>` tag in Website development are as follows:
  - The `<div>` tag is a block-level element.
  - The `<div>` tag can contain any other tag and can also be nested.
  - In HTML5, the `<div>` tag can be found inside any element that can contain flow elements, such as other `<div>`, `<address>`, `<section>`, and `<table>`.

## Introduction to HTML5 Forms 1-3

- An HTML form is used to collect user input.
- The user input is usually sent to a server for processing.
- The `<form>` element creates an HTML form for user input.

Enter Your Name:

Enter Your Age:

Are You an Employed Person:

Yes  
 No

Designing Modernistic Websites © Aptech Limited

18

## Introduction to HTML5 Forms 2-3

Type	Description
<code>&lt;input type="text"&gt;</code>	Used for a single-line text input field
<code>&lt;input type="radio"&gt;</code>	Used for a radio button
<code>&lt;input type="checkbox"&gt;</code>	Used for a check box
<code>&lt;input type="submit"&gt;</code>	Used for a submit button
<code>&lt;input type="button"&gt;</code>	Used for a clickable button

Common Input Types in HTML5

Designing Modernistic Websites © Aptech Limited

19

## Introduction to HTML5 Forms 3-3

```
<!DOCTYPE html>
<html>
<head>
<title>FormDemo</title>
</head>
<body>
<form method="get" action="test.html">
<label>Enter Your Name:</label><br/>
<input type="text" value="" id="name" /><br/>
<label>Enter Your Age:</label><br/>
<input type="text" value="" id="age" /><br/>
<label>Are You an Employed Person:</label><br/>
<input type="radio" value="Yes" id="Yes" />
<label for="Yes">Yes</label><br>
<input type="radio" value="No" id="No" />
<label for="No">No</label><br>
<br/>
<input type="submit" value="Submit" />
</form>
</body>
</html>
```

Enter Your Name:

Enter Your Age:

Are You an Employed Person:

Yes

No

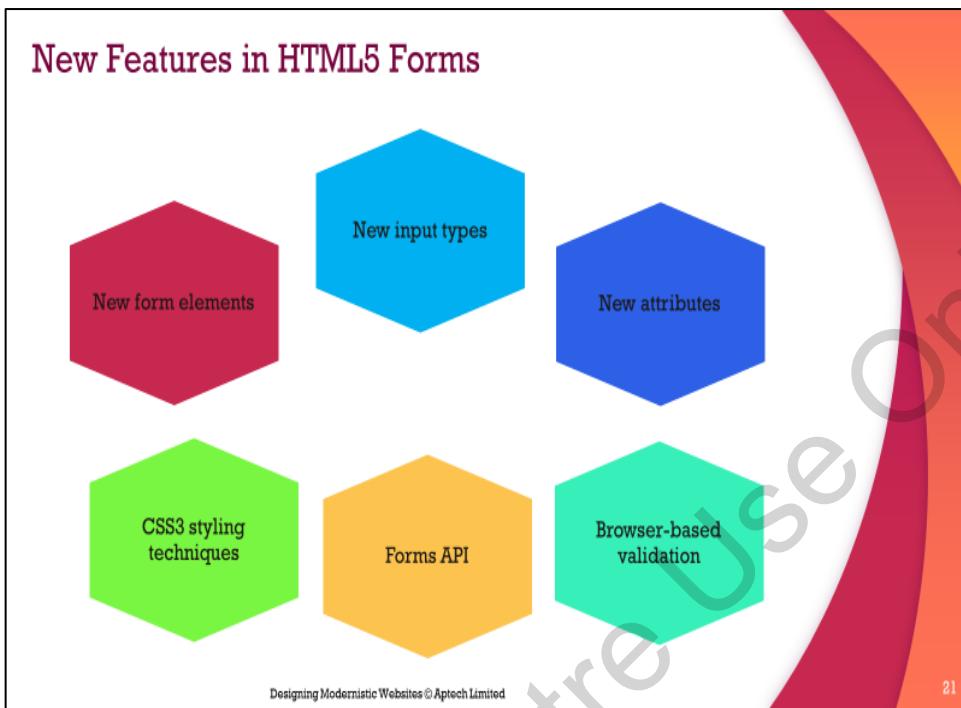
Output Showing HTML Form

Designing Modernistic Websites © Aptech Limited

20

### Instructions to the Trainer(s):

- Ask students where all do they get to fill forms online to understand the purpose of forms.
- Using Slide 18, introduce students to HTML5 forms.
- Using Slide 19, explain different input types of forms. The <form> element can hold different types of input elements, such as, check boxes and radio buttons.
- Using Slide 20, explain a simple markup for a form that accepts user data and transfers control to another HTML page.
- Tell the students, that form data is often sent to a file on the server for processing when user clicks the Submit button.



**Instructions to the Trainer(s):**

- Using Slide 21, explain the new features of HTML 5 form.
- Mention HTML5 Web forms bring great improvements related to form creation for the Web developers and also for users interacting with them.
- Tell the students that all the new features help in creating a delightful user experience.

Slides 22 and 23

## New Form Elements 1-2

Element	Description
progress	Represents the completion progress of a task on the page
meter	Represents a scale of known range
datalist	Represents a set of options used with list attribute to make a drop-down control
output	Represents the result of a calculation

New Elements in HTML5

Designing Modernistic Websites © Aptech Limited

22

## New Form Elements 2-2

```
<label> Downloading status: </label>
<progress value="35" max="100" >
</progress>
<input type="submit" value="submit"/>
```



Progress Element

Designing Modernistic Websites © Aptech Limited

23

**Instructions to the Trainer(s):**

- Using Slides 22 and 23, explain the new elements of HTML5 form.

- HTML5 has introduced a range of new elements that are expanding the options for more number of elements related to input on the forms.
- Describe the list of the new elements in HTML5.
- Tell the students that these elements are to be used with JavaScript to make them more functional.
- For more information about new elements, refer to the following:  
<https://www.wideskills.com/html5-tutorial/03-new-form-elements-in-html5>
- Using Slide 23, explain the code. The progress element contains two attributes, max, and value.
  - The max attribute declares the maximum value for the task to be processed.
  - The value attribute indicates how much task has been processed so far.

**In-Class Question:**

**Question:** What element displays the result of a calculation?

**Answer:** Output.

## New Input Types 1-2

Type	Description
email	Represents the completion progress of a task on the page
search	Represents a scale of known range
url	Represents a set of options used with list attribute to make a drop-down control
tel	Represents the result of a calculation
number	Represents a numeric value in the input field
range	Represents a numeric value to be selected from a range of numbers
date	Represents a calendar which is shown at each click upon the field
week	Represents date in year-week format
month	Represents a value with year-month format
time	Represents a value in hours and minutes format
datetime	Represents a full date and time input field with a time zone
color	Represents a predefined interface for selecting color

Designing Modernistic Websites © Aptech Limited

24

## New Input Types 2-2

```
<label for="url">Enter your Web page address:</label>
<input type="url" value="" id="urlname" name="urltext"
       maxlength="255" />
<input type="submit" value="submit"/>
```

Error Message for Incorrect URL

Number Input Type

Designing Modernistic Websites © Aptech Limited

25

### Instructions to the Trainer(s):

- Using Slides 24 and 25, explain the new input types.

- Explain the list of the new input types supported by HTML5 that specify the kind of input expected from the users on the Web page.
- Tell the students that HTML 5 introduces several input types such as DateTime-local, time, week, and month. This is to improve the user experience and to make the forms more interactive.
- Explain the new input types with some code snippet and outputs given on Slide 25.
- For more information about new elements, refer to the following: <https://www.c-sharpcorner.com/article/list-of-new-input-types-in-html-5/>

**In-Class Question:**

**Question:** Which attribute of the input element determines what kind of input will be displayed on the user's browser?

**Answer:** type attribute.

## New Form Attributes 1-2

Type	Description
placeholder	Represents a hint that help users to enter the correct data in the field
required	A Boolean attribute that validates the entry in the field
multiple	A Boolean attribute that allows multiple values to be entered in the field
autofocus	Focuses the input element on page load
pattern	Represents a regular expression for validating the field's value
form	Allows the elements to reference the form by including the form name

Designing Modernistic Websites © Aptech Limited

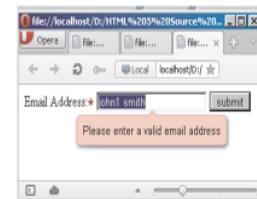
26

## New Form Attributes 2-2

```
<label>Phone number:</label>
<input type="tel" value="" size="4" maxlength="5" tabindex="1" required="true" placeholder="Code" pattern="[\+0-9]{1,4}" title="Format: (+) 99(99)"/>
<label>-</label>
<input type="tel" value="" size="10" maxlength="12" tabindex="13" required="true" placeholder="Number" pattern="[\d]{8,}" title="Minimum 8 numbers"/>
```



Message of Pattern Attribute



Validation of  
Multiple E-mail  
Address

Designing Modernistic Websites © Aptech Limited

27

### Instructions to the Trainer(s):

- Using Slides 26 and 27, explain the new attributes introduced in HTML 5.
- Explain the list of new attributes in HTML5.
- HTML5 has introduced several new attributes that can be used with form and input elements.

- Attributes help the elements to perform their tasks.
- With new Input Types and Attributes, the <input> element becomes more dynamic.
- Mention, there are many other attributes such as autocomplete, step, formnovalidate, formaction, and so on.
- Using Slide 27, explain the code snippet for pattern attribute.
- In the code, [+0-9] pattern specifies that only special character '+' as well as numbers are allowed.
- {1, 4} refers to the length of the numbers, that is between 1 and 4.
- {8,} means minimum eight numbers are allowed in the tel input type field.
- For more information about new form attributes, refer to the following:  
<https://wwwencodedna.com/html5/input-types-and-attributes-in-html5.htm>

## Browser-based Validation and CSS Styling Techniques

- The new attributes such as required and pattern can be used with the input elements to perform validation.
- Web developers need not have to write separate JavaScript code for client-side validation.
- Input field with required attribute can be styled using CSS.
  - This makes it easier for user to navigate and complete the form.

### Instructions to the Trainer(s):

- Using Slide 28, explain the concept of browser-based validation and CSS Styling techniques.
- HTML4 supported the use of custom JavaScript or libraries to perform validation on the client-side browsers. These validations ensure that the input fields are checked before the form is submitted to the server for further processing.
- The new attributes in HTML5, such as required and pattern can be used with the input elements to perform validation. This relieves the Web developers from writing the custom JavaScript code for performing client-side validation on the Web pages.
- HTML5 also provides advanced validation techniques that can be used with JavaScript to set custom validation rules and messages for the input elements.
- Mention, that browser-based validation or client-side validation may not only reduce the round trip, but may also prevent from vulnerabilities such as injection flaws and malicious file execution.
- A Web developer can enhance the form elements with the pseudo-class selectors, such as :required, :valid, and :invalid.

For example, the input fields which cannot be left blank while submitting the form can be displayed with an outline. To achieve this, input field with required attribute can be styled using CSS. Applying CSS styles make it easier for user to navigate and complete the form.

## Forms API

Events and Methods	Description
setCustomValidity (message)	Sets the custom error message that is displayed when the form is submitted by the user
checkValidity()	Checks the validity of the e-mail address entered by the user
oninvalid	Allows script to run only when the element is invalid
onforminput	Allows script to run when the form gets an input from the user
onformchange	Represents a regular expression for validating the field's value
form	Allows script to run when the form changes

Designing Modernistic Websites © Aptech Limited

29

**Instructions to the Trainer(s):**

- Using Slide 29, explain the form API.
- HTML5 has also introduced JavaScript API for forms. This API is used to customize validations and processing performed on the forms.
- The new form's API provides new methods, events, and properties to perform complex validations combining fields or calculations. Explain the list of events and methods.

## Hidden Elements 1-3

- Hidden Elements remain out of the sight of user when filling out a form.
- The fields can have a default value.
- The information in hidden fields is processed by the server after the form is submitted.
- Common uses are as follows:
  - To track edited content
  - To improve Website security

Designing Modernistic Websites © Aptech Limited

30

## Hidden Elements 2-3

```
<><div class="container">
<form>
  <div class="input-group">
    <label for="title">Post title:</label>
    <input type="text" id="title" name="title" value="My latest trip to
      Bulgaria">
  </div>
  <div class="input-group">
    <label for="content">Post content:</label>
    <textarea id="content" name="content" cols="60" rows="5">
      Hope You enjoy it!
    </textarea>
  </div >
  <div class="input-group">
    <button type="submit">Update post</button>
  </div>
  <input type="hidden" id="postId" name="postId" value="23678">
</form>
</div>
```

Designing Modernistic Websites © Aptech Limited

31

## Hidden Elements 3-3

```
.container{  
display:  
flex;  
justify-content: center;  
align-items: center;  
height: 100vh;  
}  
.container form{  
padding: 50px;  
border-radius:  
2px;  
border: 1px solid black;  
}  
.input-group{  
margin-bottom: 10px;  
display: flex;  
}  
input ,  
textarea{  
flex: 6;  
}  
label{  
line-height: 2;  
flex: 2;  
}  
textarea{  
height: 60px;  
}
```

title=My+latest+trip+to+Bulgaria+post&content=+Hope+You+enjoy+it  
0D%0A++++&postId=23678

Data Sent to Server

Designing Modernistic Websites © Apnech Limited

32

### Instructions to the Trainer(s):

- Using Slides 30 to 32, explain about hidden elements.
- Using Slide 30, mention how they are used as security tokens.
- Tell the students that when a user edits a blog using a form, the ID of the record could be used as a hidden field. On form submission, the server-side component knows from the ID the specific record that must be updated with submitted data.
- Forms on banking Websites especially include security measures. A security token ensures that the right user is filling out the form and keeps malicious users away.
  - Explain how hidden fields can prevent CSRF attack.
  - Double submit cookies are an effective way to prevent CSRF attacks.
- Explain what is double submit cookies and how it works.
- Using Slide 31, explain the code snippet for the form.
- Ask the students to note the hidden input 23678 for the hidden field postId in the code.
  - The hidden input value is set by the developer.
  - The hidden field is prefilled.
  - When the user submits the information, the server would know the record it has to update.
- Using Slide 32, explain the CSS for the HTML form and the form data that will be sent to the server.

## Data Attributes

- A data attribute helps store custom data on an HTML element.
- It can be called using JavaScript.
- Syntax: <element data-attribute\_name= "value">

### Instructions to the Trainer(s):

- Using Slide 33, explain data attributes.
- For more information about data attributes, refer to the following:  
<https://www.geeksforgeeks.org/html-data-attributes/>

## Summary

- ❖ HTML5 has introduced two types of semantic tags. They are namely, text-level and structural.
- ❖ Some of the structural semantic tags include section, header, footer, and so on.
- ❖ Text-level semantic tags include mark, time, meter, and progress.
- ❖ Text-based navigation bars are created as stand-alone navigation bars that are not associated with icons.  
Text-based navigation bar is easy to create and can be displayed in any Web browsers.
- ❖ Graphical navigation bar is better than text-based navigation as it gives a visual appeal to the visiting users.
- ❖ Div can be used when there is no other semantically appropriate element left that suits the purpose in a Web page development.
- ❖ HTML5 introduces new form elements such as new input types, new attributes, browser-based validation, CSS3 styling techniques, and forms API.
- ❖ HTML5 provides new input types that are data-specific user interface elements such as email, URL, number, range, date, tel, and color.
- ❖ The new form elements introduced in HTML5 are namely, datalist, progress, meter, and output.
- ❖ In HTML5, one can use the submit input type for form submission.
- ❖ Hidden elements remain out of the sight of user and can be used to transmit sensitive information to the server.

34

### Instructions to the Trainer(s):

- Show students Slide 34.
- Summarize the session by reading out each point on the Slide.

# Session 10: HTML5 Tables, Audio, and Video

## 10.1 Pre-Class Activities

Before you commence the session, you should familiarize yourself with the topics of this session in-depth. Prepare a question or two that will be a key point to relate the current session objectives.

### 10.1.1 Teaching Skills

To teach this session, you should be well versed with creating and formatting tables. You will learn how to resize the table and adjust column width. You should also be well-versed with the concepts of multimedia files. Then, you should aware yourself with the supported media types in HTML5 such as audio and video elements.

You should teach the concepts in the theory class using the images provided. For teaching in the class, you are expected to use slides and LCD projectors.

## In-Class Activities

Follow the order given here during In-Class activities.

Slide 2

**Objectives**

- Describe how to create and format tables
- Explain the table size and the width of a column
- Explain the process of merging table cells
- Explain the page layout for tables
- Describe the need for multimedia in HTML5
- List the supported media types in HTML5
- Explain the audio elements in HTML5
- Explain the video elements in HTML5
- Explain the accessibility of audio and video elements
- Describe how to deal with non-supporting browsers

Designing Modernistic Websites © Apreach Limited

2

### Instructions to the Trainer(s):

Give students a brief overview of the current session through the session objectives listed in Slide 2.

## 10.2 In-Class Explanations

Slide 3

### Introduction

Tables allow the user to view the data in a structured and classified format.

Tables can contain any type of data such as text, images, links, and other tables.

The user can create tables for displaying timetables, financial reports, and so on.

Example:

Test Done	Amount in \$
X Ray	60
X Ray	30

Designing Modernistic Websites © Apnech Limited

3

### Instructions to the Trainer(s):

- Using Slide 3, introduce students the process of creating and formatting tables.
- Tell the students that they will learn to view the data in a structured and classified format.
- Tables are created to display timetables, financial reports, and so on.
- Students can view the table displayed on Slide 3.

## Slides 4 and 5

### Creating and Formatting Tables 1-2

A table is made up of rows and columns. The intersection of each row and column is called as a cell.

A row is made up of a set of cells that are placed horizontally.

A column is made up of set of cells that are placed vertically.

The user can represent the data in a tabular format by using the `<table>` element in HTML.

The `<tr>` element divides the table into rows and the `<td>` element specifies columns for each row.

By default, a table does not have a border.

The `border` attribute of the `<table>` element specifies a border for making the table visible in a Web page.

Designing Modernistic Websites © Aptech Limited

### Creating and Formatting Tables 2-2

- The Code Snippet demonstrates how to create a table.

```
<!DOCTYPE HTML>
<html>
  <head>
    <title>Languages</title>
  </head>
  <body>
    <h2>Main Languages</h2>
    <table border="1">
      <tr>
        <td>English</td>
        <td>German</td>
      </tr>
      <tr>
        <td>French</td>
        <td>Italian</td>
      </tr>
    </table>
  </body>
</html>
```



Designing Modernistic Websites © Aptech Limited

#### Instructions to the Trainer(s):

- Using Slides 4 and 5, explain the process of creating and formatting tables.
- Mention that table is made up of rows and columns.

- The intersection of each row and column is called as a cell.
- A row is made up of a set of cells that are placed horizontally.
- A column is made up of set of cells that are placed vertically.
- The user can represent the data in a tabular format by using the `<table>` element in HTML.
- The `<tr>` element divides the table into rows and the `<td>` element specifies columns for each row.
- By default, a table does not have a border. The border attribute of the `<table>` element specifies a border for making the table visible in a Web page.

**In-Class Question:**

**Question:** Which tag is used for defining a row in table?

**Answer:** `<td>` tag is used for defining a row in the table.

## Table Headings

- The user can specify the heading for each column in HTML.
- To specify the heading for columns in a table, use the `<th>` element.
- The text included within the `<th>` element appears in bold.
- The Code Snippet demonstrates how to create a table with a heading.

```
<!DOCTYPE HTML>
<html>
  <head>
    <title>List of Students </title>
  </head>
  <body>
    <h2>List of Students</h2>
    <table border="1">
      <tr>
        <th>Name</th>
        <th>Age</th>
        <th>Place</th>
      </tr>
      <tr>
        <td>Mark</td>
        <td>17</td>
        <td>Madrid</td>
      </tr>
      <tr>
        <td>John</td>
        <td>19</td>
        <td>London</td>
      </tr>
    </table>
  </body>
</html>
```

Designing Modernistic Websites © Aptech Limited

Name	Age	Place
Mark	17	Madrid
John	19	London

### Instructions to the Trainer(s):

- Using Slide 6, explain the table headings and HTML code.
- Explain that one can specify the heading for each column in HTML. To specify the heading for columns in a table, use the `<th>` element.
- The text included within the `<th>` element appears in bold. Explain the code snippet that demonstrates how to create a table with a heading.
- In this code, the `<table>` element creates a table with a border of one pixel. The `<th>` element provides three column headings namely, Name, Age, and Place.

## Colspan Attribute 1-2

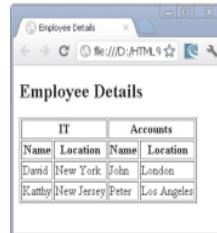
- Spanning refers to a process of extending a cell across multiple rows or columns.
- To span two or more columns, use the `colspan` attribute of the `<td>` and `<th>` elements.
- The `colspan` attribute allows the user to span a cell along a horizontal row.
- The value of the `colspan` attribute specifies the number of cells across which a specific cell shall be expanded.

Designing Modernistic Websites © Aptech Limited

7

## Colspan Attribute 2-2

```
<!DOCTYPE HTML>
<html>
  <head>
    <title>Employee Details</title>
  </head>
  <body>
    <h2>Employee Details</h2>
    <table border="1">
      <tr>
        <th colspan="2">IT</th>
        <th colspan="2">Accounts</th>
      </tr>
      <tr>
        <th>Name</th>
        <th>Location</th>
        <th>Name</th>
        <th>Location</th>
      </tr>
      <tr>
        <td>David</td>
        <td>New York</td>
        <td>John</td>
        <td>London</td>
      </tr>
      <tr>
        <td>Kathy</td>
        <td>New Jersey</td>
        <td>Peter</td>
        <td>Los Angeles</td>
      </tr>
    </table>
  </body>
</html>
```



IT		Accounts	
Name	Location	Name	Location
David	New York	John	London
Kathy	New Jersey	Peter	Los Angeles

Designing Modernistic Websites © Aptech Limited

8

### Instructions to the Trainer(s):

- Using Slides 7 and 8, explain the `colspan` attribute.

- The user might feel the must span two or more cells while working with tables.
- Spanning refers to a process of extending a cell across multiple rows or columns. To span two or more columns, use the `colspan` attribute of the `<td>` and `<th>` elements.
- The `colspan` attribute allows the user to span a cell along a horizontal row.
- The value of the `colspan` attribute specifies the number of cells across which a specific cell shall be expanded.
- Then, explain the code snippet which demonstrates how to create a table and span header cells across two cells vertically.
- Explain the code that creates a table with a border of one pixel. The `<th>` element specifies two column headings namely, IT and Accounts.
- Each of these header cells horizontally span across the two cells by setting the `colspan` attribute of the `<th>` element to 2. Each of these headings has two sub-headings namely, Name and Location, which specify the name and location of employees.
- The first and second rows display the details of the employees.

## Rowspan Attribute 1-3

- The `rowspan` attribute spans a data cell across two or more rows.
- It allows the user to span a data cell along a vertical column.
- Like the `colspan` attribute, the `rowspan` attribute can be used within the `<td>` and `<th>` elements.
- The Code Snippet demonstrates how to span a cell across multiple rows.

```
<!DOCTYPE HTML>
<html>
  <head>
    <title>Automobile Gallery</title>
  </head>
  <body>
    <table border="1">
      <tr>
        <th>Manufacturer</th>
        <th>Model</th>
        <th>Price</th>
      </tr>
      <tr>
        <th rowspan="3">Audi</th>
        <td>A4</td>
        <td>34.5</td>
      </tr>
```

Designing Modernistic Websites © Aptech Limited

9

## Rowspan Attribute 2-3

```
<tr>
  <td>A5</td>
  <td>42.6</td>
</tr>
<tr>
  <td>A6</td>
  <td>30.75</td>
</tr>
<tr>
  <th rowspan="2">BMW</th>
  <td>328i</td>
  <td>28.25</td>
</tr>
<tr>
  <td>530d</td>
  <td>47.5</td>
</tr>
</table>
</body>
</html>
```

- The code creates a table with a border width of one pixel.

Designing Modernistic Websites © Aptech Limited

10

## Rowspan Attribute 3-3

- Three <th> elements within the <tr> element specify column headings namely, **Manufacturer**, **Model**, and **Price**.
- The rowspan attribute of the <th> element combines the three rows of the **Manufacturer** column into a common brand namely, **Audi**.
- Three different models and the respective prices of the **Audi** brand are displayed in three different rows.
- Similarly, the rowspan attribute of the <th> element combines the next two rows of the **Manufacturer** column into a common brand called **BMW**.
- Following figure displays the rowspan attribute effect:

The screenshot shows a table titled "Automobile Gallery" with three columns: Manufacturer, Model, and Price. The data is as follows:

Manufacturer	Model	Price
Audi	A4	34.5
	A5	42.6
	A6	30.75
BMW	328i	28.25
	530d	47.5

Designing Modernistic Websites © Aptech Limited

11

### Instructions to the Trainer(s):

- Using Slides 9 to 11, explain the attribute `rowspan` and HTML code for it.
- The `rowspan` attribute spans a data cell across two or more rows.
- It allows spanning a data cell along a vertical column. Like the `colspan` attribute, the `rowspan` attribute can be used within the `<td>` and `<th>` elements.
- Explain the code snippet for spanning a cell across multiple rows. Using Slide 11, explain the output for `rowspan` attribute.
- Explain the code that creates a table with a border width of 1 pixel. The three `<th>` elements within the `<tr>` element specify column headings namely, Manufacturer, Model, and Price.
- The `rowspan` attribute of the `<th>` element combines the three rows of the Manufacturer column into a common brand namely, Audi. The three different models and the respective prices of the Audi brand are displayed in three different rows.
- Similarly, the `rowspan` attribute of the `<th>` element combines the next two rows of the Manufacturer column into a common brand called BMW.
- Explain the figure that displays the `rowspan` effect.

## Horizontal Alignment 1-2

- Alignment determines the representation of text along the left, right, or center positions.
- In HTML, by default, the data within the table is aligned on the left side of the cell.
- HTML5 has deprecated the align attribute.
- The four possible values for setting the horizontal alignment are as follows:

### left:

- Aligns the data within a cell on the left side. This is the default value for table content.

### center:

- Aligns the data within the cell on the center. This is the default value for table headings.

### right:

- Aligns the data within the cell on the right side.

### justify:

- Aligns the data within the cell by adjusting the text at the edges.

Designing Modernistic Websites © Aptech Limited

12

## Horizontal Alignment 2-2

- To set the alignment with style you can use the text-align attribute to specify the horizontal alignment.
- The Code Snippet demonstrates how to center align the table data.

```
<!DOCTYPE HTML>
<html>
<head>
  <title>Automobile Gallery</title>
</head>
<body>
  <table border="1">
    <tr>
      <th>Sr.No.</th>
      <th>Medicine Name</th>
      <th>Price</th>
    </tr>
    <tr style="text-align: center;">
      <td>1</td>
      <td>Captopril</td>
      <td>12.45</td>
    </tr>
    <tr style="text-align: center;">
      <td>2</td>
      <td>Ceftriaxone</td>
      <td>6.94</td>
    </tr>
    <tr style="text-align: center;">
      <td>3</td>
      <td>Cliprofloxacin</td>
      <td>56.21</td>
    </tr>
  </table>
</body>
</html>
```

Sr.No.	Medicine Name	Price
1	Captopril	12.45
2	Ceftriaxone	6.94
3	Cliprofloxacin	56.21

Designing Modernistic Websites © Aptech Limited

13

**Instructions to the Trainer(s):**

- Using Slides 12 and 13, explain the horizontal alignment or the align attribute.
- Alignment determines the representation of text along the left, right, or centre positions. In HTML, by default, the data within the table is aligned on the left side of the cell.
- Sometimes, the user might must align the data to some other position for improving the readability or focusing on some data. HTML5 has deprecated the align attribute.
- The four possible values for setting the horizontal alignment are as follows:
  - **left:** Aligns the data within the cell on the left side. This is the default value for table content.
  - **center:** Aligns the data within the cell on the center. This is the default value for table headings.
  - **right:** Aligns the data within the cell on the right side.
  - **justify:** Aligns the data within the cell by adjusting the text at the edges.

## Vertical Alignment 1-2

- Users can vertically align the position of data earlier by using the valign attribute.
- HTML5 has deprecated the valign attribute.
- The possible values of vertical alignment are as follows:

### top:

- Vertically aligns the data within the cell at the top.

### middle:

- Vertically aligns the data within the cell at the center.

### bottom:

- Vertically aligns the data within the cell at the bottom.

- To set the alignment with the style, you can use the text-align attribute to specify the vertical alignment use the following syntax:

```
<td style= "text-align: center; vertical-align: middle"> Aptech  
Web site </a>
```

Designing Modernistic Websites © Aptech Limited

14

## Vertical Alignment 2-2

- The Code Snippet demonstrates how to align the data vertically within the table using the style attribute.

```
<!DOCTYPE HTML>  
<html>  
<head>  
<title>CelinaBatteries</title>  
</head>  
<body>  
<table border="1">  
<tr>  
<th>Sr.No.</th>  
<th>Product ID</th>  
<th>Product Description</th>  
</tr>  
<tr>  
<td style="text-align: center; vertical-align: middle">1  
</td>  
<td style="text-align: center; vertical-align: middle">P101  
</td>  
<td>1.5 Volts AA Ultra Alkaline</td>  
</tr>  
<tr>  
<td style="text-align: center; vertical-align: middle">2  
</td>  
<td style="text-align: center; vertical-align: middle">M105  
</td>  
<td>9 Volts pp3 Super Alkaline</td>  
</tr>  
</table>  
</body>  
</html>
```

Designing Modernistic Websites © Aptech Limited

Sr.No	Product ID	Product Description
1	P101	1.5 Volts AA Ultra Alkaline
2	M105	9 Volts pp3 Super Alkaline

15

### Instructions to the Trainer(s):

- Using Slides 14 and 15, explain the vertical alignment.
- Mention users can vertically align the position of data earlier by using the valign attribute.

- HTML5 has deprecated the `valign` attribute. The possible values of vertical alignment are as follows:
  - **top**: Vertically aligns the data within the cell at the top.
  - **middle**: Vertically aligns the data within the cell at the center.
  - **bottom**: Vertically aligns the data within the cell at the bottom.
- To set the alignment with the style, you can use the `text-align` attribute to specify the vertical alignment use the following syntax:  
`<td style= "text-align: center; vertical-align: middle">`
- The style can also be applied to individual rows, cells, or to the entire table.

## Margin Attributes

- The data in a table might appear cluttered, which may affect the readability.
- This might make it difficult to comprehend data as the data.
- To overcome this issue, use the cell margin attributes.
- Cell padding allows the user to control the look of the content on a page.

### ➤ Padding

- Padding is the amount of space between the content and its outer edge.
- For tables, padding is referred to as a space between the text and the cell border.
- Suppose, if the user wants to set the padding attribute for the individual cells then padding attribute can be used in a style as follows:  
`<td style="padding: 4px">`

Designing Modernistic Websites © Aptech Limited

16

### Instructions to the Trainer(s):

- Using Slide 16, explain the margin attribute.
- The data in a table might appear cluttered, which may affect the readability. This might make it difficult to comprehend data as the data. To overcome this issue, use the cell margin attributes. Cell padding allows the user to control the look of the content on a page.
- **Padding:** Padding is the amount of space between the content and its outer edge. For tables, padding is specified as a space between the text and the cell border.
- Suppose, if the user wants to set the padding attribute for the individual cells, then he/she can use the padding attribute in a style as follows:  
`<td style="padding: 4px">`

## Slides 17 and 18

### Caption Element 1-2

- To specify the main heading for the table, use the `<caption>` element.
- The `<caption>` element defines a caption for the table. It is a sub-element of the `<table>` element.
- It must be present immediately after the `<table>` tag.
- The `<caption>` element allows the user to specify a title for your entire table.
- There can be only one caption for a table.
- The Code Snippet demonstrates how to specify a heading for a table.

```
<!DOCTYPE HTML>
<html>
<head>
    <title>Travel Expense Report</title>
</head>
<body>
    <table border="1">
        <caption>Travel Expense Report</caption>
        <tr>
            <th>&nbsp;</th>
            <th>Meals</th>
            <th>Hotels</th>
            <th>Transport</th>
        </tr>
    </table>

```

Designing Modernistic Websites © Aptech Limited

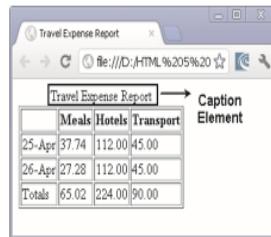
17

### Caption Element 2-2

```

        <tr>
            <td>25-Apr</td>
            <td>37.74</td>
            <td>112.00</td>
            <td>45.00</td>
        </tr>
        <tr>
            <td>26-Apr</td>
            <td>27.28</td>
            <td>112.00</td>
            <td>45.00</td>
        </tr>
        <tr>
            <td>Totals</td>
            <td>65.02</td>
            <td>224.00</td>
            <td>90.00</td>
        </tr>
    </table>
</body>
</html>

```



Designing Modernistic Websites © Aptech Limited

18

### Instructions to the Trainer(s):

- Using Slides 17 and 18, explain the `caption` element and HTML code for it.

- Mention a user can add a heading to a table in HTML. To specify the main heading for the table, use the `<caption>` element. The `<caption>` element defines a caption for the table. It is a sub-element of the `<table>` element. It must be present immediately after the `<table>` tag.
- Unlike the `<th>` element that is used to specify a heading to an individual row or column, the `<caption>` element allows the user to specify a title for the entire table. There can be only one caption for a table.
- Explain the code snippet for specifying a heading for a table using `caption` element.
- Using Slide 18, explain the output for the `<caption>` tag code.
- The code creates a table of border width of 1 pixel. The `<caption>` element that is used inside the `<table>` element specifies a caption to the entire table as **Travel Expense Report**.

## Table Size and Width of a Column

- The table size can be expanded when the user wants to add rows and columns in the table.
- The user can use the `<style>` section to set the default width for the table to 100% of the browser window.
- To set the width of a column in pixels, one can use style attribute in the `<td>` tag.
- The Code Snippet demonstrates how to create a table with specific width for a column.

```
<!DOCTYPE HTML>
<html>
<head>
<title>Tables</title>
</head>
<body>
<h2>Table</h2>
<table border="1">
<tr>
<td style ="width: 200px">Flowers</td>
<td style ="width: 80px">Fruits</td>
</tr>
<tr>
<td style ="width: 200px">Vegetables</td>
<td style ="width: 80px">Trees</td>
</tr>
</table>
</body>
</html>
```

Flowers	Fruits
Vegetables	Trees

Designing Modernistic Websites © Aptech Limited

19

### Instructions to the Trainer(s):

- Using Slide 19, explain the table size and column width.
- Mention that user can decide the size of the table based on his/her requirements while creating a Website.
- The table size can be expanded when the user wants to add rows and columns in the table.
- The user can use the `<style>` section to set the default width for the table to 100% of the browser window. For setting the width of a column in pixels, you can use style attribute in the `<td>` tag.
- Explain the code snippet for creating a table with specific width for a column. Explain the figure that displays the table size and column width.

## Merging Table Cells 1-2

- To change the cells of a table to different height and width, colspan and rowspan attributes can be used.
- Consider a scenario, where the user wants to merge a cell into adjacent cells to the right side.
  - The colspan attribute can be used to specify the number of columns to span.
  - The rowspan attribute can be used to specify the number of rows.
- The Code Snippet demonstrates creating a table having five columns and five rows, but many of the cells span multiple columns or rows.

```
<!DOCTYPE HTML>
<html>
<head>
    <title>Favorite Destination</title>
</head>
<body>
    <h2>Report</h2>
    <table border="1" width="100%" height="100%">
        <tr>
            <td colspan="2" rowspan="2">Results</td>
            <td colspan="3">Range</td>
        </tr>
```

Designing Modernistic Websites © Aptech Limited

20

## Merging Table Cells 2-2

```
<tr>
    <td>18 to 20</td>
    <td>25 to 50</td>
    <td>Over 50</td>
</tr>
<tr>
    <td rowspan="3">Your favorite vacation destination</td>
    <td>Dubai</td>
    <td>25%</td>
    <td>50%</td>
    <td>25%</td>
</tr>
<tr>
    <td>Bangkok</td>
    <td>40%</td>
    <td>30%</td>
    <td>30%</td>
</tr>
<tr>
    <td>Switzerland</td>
    <td>30%</td>
    <td>20%</td>
    <td>50%</td>
</tr>
</table>
</body>
</html>
```

Results		Range		
Your favorite vacation destination	Dubai	25%	50%	25%
	Bangkok	40%	30%	30%
	Switzerland	30%	20%	50%

Designing Modernistic Websites © Aptech Limited

21

**Instructions to the Trainer(s):**

- Using Slides 20 and 21, explain the concept of merging table cells.
- Explain the necessity for merging cell. Suppose if the user wants to change the cells of a table to different height and width, then `colspan` and `rowspan` attributes can be used.
- Consider a scenario, where the user wants to merge a cell into adjacent cells to the right side.
- The `colspan` attribute can be used to specify the number of columns to span. Similarly, the user can use the `rowspan` attribute to specify the number of rows.
- Explain the code snippet that demonstrates creating a table having five columns and five rows.

**In-Class Question:**

**Question:** Which attributes are used for merging table columns and rows?

**Answer:** `Colspan` and `rowspan` attributes are used for merging columns and rows, respectively.

## Apply Borders by Using Styles 1-2

- CSS can be used for applying borders as it is the best reliable and flexible method.
- One can format the table by using style based border for `<table>` and `<td>` tags.
- To evaluate the attributes used are as follows:

### `border-width:`

- Used to control the thickness of the border and the values are specified in pixels.

### `border-color:`

- Used to control the color of the border and specifies the color by either name, or RGB value, or hexadecimal number.

### `border-style:`

- Used to control the line style. Users can choose between solid, dashed, groove, dotted, outset, ridge, inset, or none.

Designing Modernistic Websites © Aptech Limited

22

## Apply Borders by Using Styles 2-2

To set all these attributes at one time, the user can use the `border` attribute and place the settings in the order of width, color, and style respectively.

To format the sides of the border individually, replace the `border` attribute with `border-bottom`, `border-top`, `border-right`, or `border-left` attribute.

The user can apply these attributes to the entire table or individual cells and also create rules in the `<style>` area.

Designing Modernistic Websites © Aptech Limited

23

### **Instructions to the Trainer(s):**

- Using Slides 22 and 23, explain the process of applying border using styles.
- Mention that users can use CSS for applying borders as it is the best reliable and flexible method.
- The user must select the CSS method for Websites that will be active for many years as the old formatting methods will not be used in future. You can format the table by using style based border for `<table>` and `<td>` tags.
- To evaluate this, the attributes used are as follows:
  - The `border-width` attribute is used to control the thickness of the border and the values are specified in pixels.
  - The `border-color` attribute is used to control the color of the border and specifies the color by name, or RGB value, or hexadecimal number.
  - The `border-style` attribute is used to control the line style. Users can choose value from solid, dashed, groove, dotted, outset, ridge, inset, or none.
- Suppose, if the user wants to set all these attributes at one time, then the user can use the `border` attribute and place the settings in the following order namely, width, color, and style respectively.
- The user can also format the sides of the border individually by replacing the `border` attribute with `border-bottom`, `border-top`, `border-right`, or `border-left` attribute.
- The user can apply these attributes to the entire table or individual cells and also create rules in the `<style>` area.

## Tables for Page Layout 1-3

- Tables are used for structuring the content and to organize the data in an appropriate manner.
- Tables allow the user to arrange the data horizontally or vertically according to the requirement.
- Each and every Website has a unique way of presenting data to their customers or users.
- Many Websites use pop-ups for providing information to their customers.
- The Code Snippet demonstrates a simple example of using table for structuring the content of a Web page.

```
<!DOCTYPE HTML>
<html>
<head>
    <title>Page Layout </title>
    </head>
    <style>
```

Designing Modernistic Websites © Aptech Limited

24

## Tables for Page Layout 2-3

```
#navlayout {
    width: 100%;
    float: left;
    margin: 0 0 3em 0;
    padding: 0;
    list-style: none;
    background-color: #f2f2f2;
    border-bottom: 1px solid #ccc;
    border-top: 1px solid #ccc; }
#navlayout li {
    float: left; }
#navlayout li a {
    display: block;
    padding: 8px 15px;
    text-decoration: none;
    font-weight: bold;
    color: #069;
    border-right: 1px solid #ccc; }
#navlayout li a:hover {
    color: #c00;
    background-color: #fff; }
</style>
```

Designing Modernistic Websites © Aptech Limited

25

## Tables for Page Layout 3-3

```
<body>
    
    <h1>Blossoms Gallery</h1>
    <h5><i>The Best sellers for flowers since 1979</i></h5>
    <navlayout>
        <hr>
        <ul id="navlayout">
            <li><a href="#">Home</a></li>
            <li><a href="#">Contact Us</a></li>
            <li><a href="#">About Us</a></li>
            <li><a href="#">FAQs</a></li>
        </ul>
    </navlayout>
    <table>
        <tr>
            <td>
                <b>Flowers are now in stock! </b>
                <i>We have just received a large shipment
                    of flowers with prices as low as $19.
                </i>
            </td>
        </tr>
    </table>
</body>
</html>
```



26

### Instructions to the Trainer(s):

- Using Slides 24 to 26, explain the use of table for page layout.
- Nowadays, there are many new techniques used for developing attractive Web pages. Tables are used for structuring the content. In other words, tables are used by the user to organize the data in an appropriate manner.
- With the help of tables, the user can arrange the data horizontally or vertically according to his/her requirements.
- Community Websites such as Facebook has different page layouts, the user uses the navigation tabs to move from one page to another. Similarly, the look and feel of each page is different.
- While accessing Websites such as Yahoo, Rediff, and so on, users can view that the home page is very informative with a number of links, images, and so on.
- Each and every Website has its unique way of presenting data to their customers or users.
- Many Websites use popups for providing information to their customers.

## Introduction to Multimedia

- Traditionally, Web browsers were capable of handling only graphics and text.
- User had to install a distinct program, plug-in, or an ActiveX control to play some video.
- Earlier, Web designers and Web developers used to set up Web pages to play audio and video on the Web using Adobe Flash player.

Designing Modernistic Websites © Aptech Limited

27

**Instructions to the Trainer(s):**

- Using Slide 27, introduce the concept of multimedia to students.
- Traditionally, Web browsers were capable of handling only graphics and text.
- Suppose, if a user had to play some video, then, a distinct program, plug-in, or an ActiveX control had to be installed.
- Earlier, Web designers and Web developers used to set up Web pages using Adobe Flash player to play audio and video on the Web.

## Multimedia in HTML5

Multimedia is a combination of various elements such as video, graphics, sound, and text.

Common way of inserting a multimedia content on Web pages is by embedding a video or audio file in the Web page.

HTML5 has made lives easier by introducing `<audio>` and `<video>` elements.

HTML5 has provided the developers with the features to embed media on the Web pages in a standard manner.

### Instructions to the Trainer(s):

- Using Slide 28, explain multimedia in HTML5.
- Multimedia is a combination of various elements such as video, graphics, sound, and text. A common way of inserting multimedia content on Web pages is by embedding a video or audio file in a Web page.
- Consider the earlier situations where a Website developer did not have the facility of including videos or audios directly on their Website until and unless the browser had the required plug-in installed. These days, Website developers want their visitors to not only download, but also view movies online on their Website. Today, HTML5 provides this facility.
- Explain the students that HTML5 has made lives easier by introducing `<audio>` and `<video>` elements. Thus, the user does not have to depend on Flash to access the audio and video files.

## Supported Media Types in Audio and Video

There are various video and audio codecs which are used for handling of video and audio files.

Codec is a device or a program used for encoding and decoding digital data stream.

Different codecs have different level of compression quality.

For storing and transmitting coded video and audio together, a container format is used.

There are a number of container formats which includes Ogg (.ogg), the Audio Video Interleave (.avi), Flash Video (.flv), and many others.

Different browsers support different container format. WebM is a new open source video container format supported by Google.

- Following table lists the common audio and video formats:

Container	Video Codec	Audio Codec
Mp4	H.264	AAC
Ogg	Theora	Vorbis
WebM	VP8	Vorbis

Designing Modernistic Websites © Aptech Limited

29

### Instructions to the Trainer(s):

- Using Slide 29, explain the supported media types in audio and video.
- Mention that there are various video and audio codecs which are used for handling of video and audio files.
- The codec is a device or a program used for encoding and decoding digital data stream. These different codecs have different level of compression quality.
- For storing and transmitting coded video and audio together, a container format is used. There are a number of container formats which includes Ogg (.ogg), the Audio Video Interleave (.avi), Flash Video (.flv), and many others. WebM is a new open source video container format supported by Google. Different browsers support different container format.
- Explain the list of the common audio and video formats listed in the slide.

## Audio Formats

- There are three supported file formats for the `<audio>` element in HTML5.
- Following table lists audio file formats supported by Web browsers:

Browser Support	MP3	WAV	Ogg
Opera 10.6	No	Yes	Yes
Apple Safari 5	Yes	Yes	No
Google Chrome 6	Yes	Yes	Yes
FireFox 4.0	No	Yes	Yes
Internet Explorer 9	Yes	No	No
Edge 17-91 and 92 onwards	Yes	Yes	Yes

Designing Modernistic Websites © Aptech Limited

30

### Instructions to the Trainer(s):

- Using Slide 30, explain the audio formats.
- There are three supported file formats for the `<audio>` element in HTML5.
- Table lists the audio file formats supported by the Web browsers.

## Video Formats

- There are three supported file formats for the <video> element in HTML5.
- Following table lists video file formats supported by Web browsers:

Browsers Support	MP4	WebM	Ogg
Opera 10.6 onwards	No	Yes	Yes
Apple Safari 5 onwards	Yes	No	No
Google Chrome 6 onwards	Yes	Yes	Yes
FireFox 4.0 onwards	No	Yes	Yes
Internet Explorer 9 onwards	Yes	No	No
Edge 79 onwards	Yes	Yes	Yes

Designing Modernistic Websites © Aptech Limited

31

### Instructions to the Trainer(s):

- Using Slide 31, explain the video formats.
- There are three supported file formats for the <video> element in HTML5.
- Table lists which of the browsers support these three formats namely MP4, WebM, and Ogg.

## Audio Elements in HTML5

- `<audio>` element will help the developer to embed music on the Website.
- `<audio>` tag specifies the audio file to be used in the HTML document.
- `src` attribute is used to link the audio file.
- The Code Snippet displays the embedding of an audio file in the Web page using the `<audio>` tag.

```
<!doctype html>
<html>
  <head>
    <title>audio element</title>
  </head>
  <body>
    <audio src="d:\sourcecodes\audio.mp3"
    controls autoplay loop>
    html5 audio not supported
    </audio>
  </body>
</html>
```

Audio formats frequently used are wav, ogg, and mp3.

Designing Modernistic Websites © Aptech Limited



32

### Instructions to the Trainer(s):

- Using Slide 32, explain the `audio` element in HTML5.
- The `audio` element will help the developer to embed music on the Website and allow the user to listen to music. The `<audio>` tag specifies the audio file to be used in the HTML document. It contains the `src` attribute that is used to link the audio file.
- Explain the code snippet which displays the embedding of an audio file in a Web page using the `<audio>` tag. The music is played in the background when the page is loaded on the browser.
- The `src` attribute is mandatory, the `<audio>` tag includes several other options. Explain the code snippet to add an audio file on the Web page.
- Then, explain the existing HTML5 specification does not specify the formats supported by the browser in the `<audio>` tag. The audio formats frequently used are wav, ogg, and mp3.
- Then, mention that if you want to specify an alternative audio file to be selected based on its supported format in the browser. Then, the `<source>` tag is used as a child element of `<audio>` tag.
- The `<source>` tag has three attributes namely, `src`, `media`, and `type`. Example:  
`<audio>`  
`<source src="maddi.ogg" type="audio/ogg">`  
`<source src="maddi.mp3" type="audio/mpeg">`  
`</audio>`

### In-Class Question:

**Question:** What are the attributes of the `<source>` tag used within the `audio` element?

**Answer:** `src`, `media`, and `type`.

## Audio Tag Attributes

Attributes provide additional information to the browser about the tag.

HTML5 has a number of attributes for controlling the look and feel of various functionalities.

HTML5 has the following attributes for the <audio> element.

Audio Attributes	Description
autoplay	This attribute identifies whether to start or not the audio once the object is loaded. The attribute accepts a boolean value which when specified will automatically start playing the audio as soon as possible without stopping
autobuffer	This attribute starts the buffering automatically
controls	This attribute identifies the audio playback controls that should be displayed such as resume, pause, play, and volume buttons
loop	This attribute identifies whether to replay the audio once it has stopped
preload	This attribute identifies whether the audio has to be loaded when the page loads and is ready to execute
src	This attribute specifies the location or the URL of the audio file that has to be embedded

Designing Modernistic Websites © Aptech Limited

33

### Instructions to the Trainer(s):

- Using Slide 33, explain the `audio` tag attributes in detail.
- HTML tags normally consist of more than one attribute. Attributes provide additional information to the browser about the tag.
- HTML5 has a number of attributes for controlling the look and feel of various functionalities.
- According to HTML5 specifications, the `<audio>` element has the following attributes: muted, `autoplay`, and `controls`.
- The `muted` attribute is a Boolean attribute. When present, it specifies that the audio output should be muted.
- The `controls` attribute is a Boolean attribute. When present, it specifies that audio controls that should be displayed. Audio controls include:
  - Play
  - Pause
  - Seeking
  - Volume
- The `autoplay` attribute is a Boolean attribute. When present, the audio will automatically start playing as soon as it can do so without stopping.

## Creating Audio Files

- To play the audio in older browsers then the `<embed>` tag will be used.
- `<embed>` tag has two attributes, `src` and `autoplay`.
- `src` attribute is used to specify the source of the audio.
- `autoplay` attribute controls the audio and determines whether the audio should play as soon as the page loads.

- The Code Snippet demonstrates the use of `<embed>` tag in the `<audio>` element.

```
<!DOCTYPE HTML>
<html>
<body>
<audio autoplay loop>
<source src="sampaudio.mp3">
<source src="sampaudio.ogg">
<embed src="sampaudio.mp3">
</audio>
</body>
</html>
```

Designing Modernistic Websites © Aptech Limited

34

### Instructions to the Trainer(s):

- Using Slide 34, explain the process of creating audio files.
- Suppose, if the user plays the audio in older browsers then the `<embed>` tag will be used. The `<embed>` tag has two attributes namely, `src` and `autoplay`.
- The `src` attribute is used to specify the source of the audio and the `autoplay` attribute controls the audio and determines whether the audio should play as soon as the page loads.
- Explain code snippet which demonstrates the use of `<embed>` tag in the `<audio>` element.
- Mention that `<audio>` element in HTML5 supports multiple formats. The content included within the `<embed>` tag is automatically played by default. Suppose, if the user does not want to play the audio file automatically then he/she can set the value of the `autoplay` attribute to "false".
- Mention, `<embed>` tag also supports another attribute named `loop`. The `loop` attribute determines whether the audio clip will be replayed continuously or not. If the value of the `loop` attribute is set to true or infinite then, the music will be played continuously. If the `loop` attribute is not specified, then it is same as setting the value to false.

## Video Elements in HTML5

- `<video>` element is a new feature added in HTML5.
- `<video>` element is for embedding the video content on the Web page.
- `<video>` element if not supported by the browser then the content between the start tag and end tag is displayed.
- `src` attribute is used to link to the video file.
- The Code Snippet demonstrates the use of `<video>` element.

```
<!DOCTYPE HTML>
<html>
  <head>
  </head>
  <body>
    <video src="D:\Source codes\movie.mp4">
      Your browser does not support the video.
    </video>
  </body>
</html>
```



Designing Modernistic Websites © Aptech Limited

35

### Instructions to the Trainer(s):

- Using Slide 35, explain the `video` element in HTML5.
- The `video` element is a new feature added in HTML5. The user can use the `<video>` element for embedding the video content on the Web page. The easiest way to specify the video is by using the `src` attribute which gives the URL of the video file to be used.
- Suppose, if the browser does not support the `<video>` element then, the content between the start tag and end tag is displayed on the browser.
- Mention, `<video>` element allows multiple `<source>` elements. `<source>` elements can link to different video files. The browser will use the first recognized format.
- Explain code snippet which demonstrates the use of the `<video>` element. In the code, the `src` attribute is used for specifying the location of the mp4 video file format used by the `<video>` tag.
- While adding the `<video>` element in the code, the user can specify messages between the `<video>` and `</video>` tag to check if the browser is supporting the `<video>` tag or not.

## Video Tag Attributes

HTML5 specification provides a list of attributes that can be used with the `<video>` element.

HTML5 has the following attributes for the `<video>` element.

- Following table lists some of the `<video>` tag attributes:

Video Attributes	Description
autoplay	Specifies that the browser will start playing the video as soon as it is ready
muted	Allows to mute the video initially, if this attribute is existing
controls	Allows displaying the controls of the video, if the attribute exists
loop	Specifies that the browser should repeat playing the existing video once more if the loop attribute exists and accepts a boolean value
preload	Specifies whether the video should be loaded or not when the page is loaded
src	Specifies the location of the video file to be embedded

Designing Modernistic Websites © Aptech Limited

36

### Instructions to the Trainer(s):

- Using Slide 36, explain the attributes of `video` tag.
- The HTML5 specification provides a list of attributes that can be used with the `video` element. Explain the table listed on the slide for `<video>` tag attributes.
- Some of the additional attributes of `video` elements are `height`, `width`, `poster`, and `src`.

## Preloading the Video 1-2

- <video> element comprises a preload attribute that allows the browser to download or buffering the video while the Web page containing the video is being downloaded.
- preload attribute has the following values:

**None** - allows the browser to load only the page. The video will not be downloaded while the page is being loaded.

**Metadata** - allows the browser to load the metadata when the page is being loaded.

**Auto** - is the default behavior as it allows the browser to download the video when the page is loaded. The browser can avoid the request.

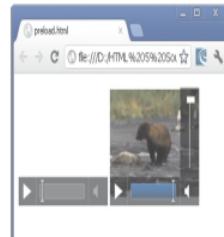
Designing Modernistic Websites © Aptech Limited

37

## Preloading the Video 2-2

- The Code Snippet demonstrates the use of none and metadata values for the preload attribute.

```
<!DOCTYPE HTML>
<html>
  <head>
  </head>
  <body>
    <video width="160" height="140" src="D:\Source
Codes\movie.mp4" controls preload="none" muted>
      Your browser does not support the video.
    </video>
    <video width="160" height="140" src="D:\ Source
Codes\movie.mp4" controls preload="metadata" muted>
      Your browser does not support the video.
    </video>
  </body>
</html>
```



Designing Modernistic Websites © Aptech Limited

38

### Instructions to the Trainer(s):

- Using Slides 37 and 38, explain preloading the video using preload attribute.

- The `video` element comprises a `preload` attribute that allows the browser to download or buffer the video while the Web page containing the video is being downloaded. If the video is preloaded, then it decreases the initial delay once the user has started the playback.
- The `preload` attribute has the following values:
  - **None:** This attribute allows the browser to load only the page. The video will not be downloaded while the page is being loaded.
  - **Metadata:** This attribute allows the browser to load the metadata when the page is being loaded.
  - **Auto:** This is the default behavior as it allows the browser to download the video when the page is loaded. The browser can avoid the request.
- Using Slide 38, explain the HTML code with `preload` attribute. In the code, there are two `video` elements.
- Each `video` element contains the `preload` attribute with value `none` and `metadata` respectively.
- Figure on Slide 38 displays the effect of both the values `none` and `metadata` on the Web page.

## Setting the Video Size

- User can specify the size of the video with the `height` and `width` attribute of the `<video>` element.
- If these attributes are not provided then the browser sets the video with the key dimensions of the video.
- The Code Snippet demonstrates how to apply the `height` and `width` attributes to the `<video>` element.

```
<!DOCTYPE HTML>
<html>
<head>
</head>
<title> Video Size</title>
<style>
video{
    background-color: black;
    border: medium double black;
}
</style>
<body>
<video src="D:\Source Codes\movie.mp4" controls preload="auto" width="360" height="340">
    Your browser does not support the video.
</video>
</body>
</html>
```



Designing Modernistic Websites © Apneet Limited

39

### Instructions to the Trainer(s):

- Using Slide 39, explain the process of setting the video size.
- The user can specify the size of the video with the `height` and `width` attribute of the `video` element.
- Suppose, if these attributes are not provided then the browser sets the video with the key dimensions of the video. This will result in changing the page layout as the Web page is adjusted to accommodate the video.
- Explain code snippet which demonstrates how to apply the `height` and `width` attributes to the `<video>` element.
- In the code, the `style` attribute is used to specify the `background-color` and `border` style of the video. The code also specifies the `preload`, `height`, and `width` attributes for the `video` element.

## Converting the Video Files

- There are many problems with browser vendors for supporting various video formats on the Web sites.
- Following are some of the video formats supported by the significant browsers:

**Ogg/Theora** - is an open source, royalty-free, and patent-free format available. This format is supported by browsers such as Opera, Chrome, and Firefox.

**WebM** - is a royalty-free and patent-free format supported by Google. This format is supported by browsers such as Opera, Chrome, and Firefox.

**H.264/MP4** - are supported on iPhone and Google Android devices.

**Micro Video Controller** - converter creates all files that the user requires for HTML5 <video> element that works on the cross browser.

### Instructions to the Trainer(s):

- Using Slide 40, explain the process of converting the video files.
- Mention video element used in HTML5 is a great feature, but how will the user get the video files in a correct format.
- There are many problems with browser vendors for supporting various video formats on the Websites. Thus, there are converters available which can convert the format of video files appropriately to be supported in HTML5 browsers.
- The **VLC converter** can be used on Windows to covert video files into Ogg format.
- For MP4 videos, you can use **HandBrake** which is an open-source application available for Windows, Mac OS X, and Linux. Similarly, **WebM Encoder 1.2** is a simple utility to covert video files to the WebM format.

## Slides 41 and 42

### Accessibility of Audio and Video Elements 1-2

- Enterprises across the world are employing people with varied skills and abilities.
- It may include people with limited abilities or disabilities such as people with visual, cognitive, or mobility impairments.
- Accessibility is the level of ease with which computers can be used and be available to a wide range of users.
- While developing an application a lot of assumptions are to be considered and some of them are as follows:

Users can check the content on laptop, mobile, tablet, or desktop.

Users can listen to the audio by using headphones or speakers.

Users can understand the language in which the media was delivered.

Users can successfully play and download the media.

Designing Modernistic Websites © Aptech Limited

41

### Accessibility of Audio and Video Elements 2-2

- Earlier, assumptions made will meet the requirements of a vast majority of users accessing the application.
- Not all users will fall in this category.
- Another set of assumptions are to be considered for users and they are as follows:

Users who have hearing and visual impairment and thus, cannot listen to the audio or view the video.

Users who are not familiar with the language that the content is delivered.

Users who uses keyboards and screen readers to access the content on Web.

Users who cannot view or hear the media content because of their working environment or due to device restrictions.

Designing Modernistic Websites © Aptech Limited

42

#### Instructions to the Trainer(s):

- Using Slides 41 and 42, explain the accessibility of audio and video element.

- Mention, enterprises across the world are employing people with varied skills and abilities. They may even include people with limited abilities or disabilities such as people with visual, cognitive, or mobility impairments.
- Accessibility is the level of ease with which computers can be used and be available to a wide range of users, including people with disabilities.
- Applications can be accessed through various sources. If the application considers the requirements of the target audience, then it will be appreciated and used by number of users.
- Explain the students about the assumptions for multimedia content.
- Then, explain the HTML5 powerful feature named Web Video Text Tracks (WebVTT) to make applications accessible to such users.
- WebVTT is a file format used to mark up the external text tracks. This format allows the user to give a textual description of the content in the video. This description is then used by different accessibility devices to define the content to those users who cannot see it.

## Track Element 1-3

Track element provides an easy, standard way to add captions, subtitles, chapters, and screen reader descriptions to the <audio> and <video> elements.

Track elements are also used for other types of timed metadata.

Source data for this track element is in a form of a text file that is made up of a list of timed cues.

Cue is a pointer at an accurate time point in the length of a video.

Cues contain data in formats such as Comma-Separated Values (CSV) or JavaScript Object Notation.

Track element is not supported in many major browsers and is now available in IE 10 and Chrome 18+.

## Track Element 2-3

- Following table lists the track element attributes:

Container	Description
src	Contains the URL of the text track data
srlang	Contains the language of the text track data
kind	Contains the type of content for which the track definition is used
default	Indicates that this will be the default track if the user does not specify the value
label	Specifies the title to be displayed for the user

## Track Element 3-3

- The Code Snippet demonstrates how a track element is used in combination with `<video>` element for providing subtitles.

```
<video controls>
  <source src="myvideo.mp4" type="video/mp4" />
  <source src="myvideo.webm" type="video/webm" />
  <track src="eng.vtt" label="English subtitles" kind="subtitles"
    srclang="en" >
</video>
```

- The Code Snippet demonstrates how a track element is used in combination with `<video>` element providing subtitles in another language.

```
<video controls>
  <source src="myvideo.mp4" type="video/mp4" />
  <source src="myvideo.webm" type="video/webm" />
  <track src="de.vtt" label="German subtitles" kind="subtitles"
    srclang="de" >
</video>
```

Designing Modernistic Websites © Apnech Limited

45

### Instructions to the Trainer(s):

- Using Slides 43 to 45, explain the `track` element in HTML5.
- The `track` element provides an easy, standard way to add captions, subtitles, chapters, and screen reader descriptions to the `<audio>` and `<video>` elements.
- Then, explain the table provided on slide 20 that lists the attributes of the `track` element.
- Using Slide 45, explain the HTML codes for `track` element. First code specifies the `src`, `label`, and `srclang` attributes in the `track` element.
- Here, the `srclang` is set to `en` that is English language. Second code demonstrates a `track` element used in combination with `<video>` element providing subtitles in the French language.

### In-Class Question:

**Question:** Which attribute is used to specify the type of element to be tracked in `track` tag?

**Answer:** `kind` attribute.

## Accessibility for Audio and Video Element

- Accessibility supports for <audio> and <video> elements are as follows:

- Audio Support

Firefox - Expose controls with accessibility APIs, however individual controls do not interact with keyboard. Access to keyboard is provided by the Firefox specific shortcuts.

Opera - Has only keyboard support.

IE 9 - Expose controls with accessibility APIs, however individual controls do not interact with keyboard.

- Video Support

Firefox - Cannot interact with individual controls.

Opera - Has only keyboard support.

IE 9 - Does not allow individual controls to interact with keyboard.

### Instructions to the Trainer(s):

- Using Slide 46, explain the accessibility support for audio and video element in various browsers.
- Mention that the lower versions of these browsers do not provide the support for multimedia elements.

## Non-Supporting Browsers

- Different browsers such as Firefox, Chrome, Opera, and Safari support <audio> and <video> elements.
- Google Chrome 17 and lower versions have no support for <audio> elements.
- Safari browser does not support <audio> element in HTML5

Designing Modernistic Websites © Aptech Limited

47

**Instructions to the Trainer(s):**

- Using Slide 47, explain the concept of non-supporting browsers.
- <audio> and <video> elements are supported by browsers such as Chrome, Opera, Firefox, and so on.
- Lower versions do not support <audio> elements.

## Summary

- ❖ Tables allow the user to view your data in a structured and classified format.
- ❖ Padding is the amount of space between the content and its outer edge.
- ❖ The caption element defines a caption for a table. It is a sub-element of the <table> element.
- ❖ Spanning refers to a process of extending a cell across multiple rows or columns.
- ❖ The rowspan attribute spans a data cell across two or more rows.
- ❖ The colspan attribute allows the user to specify the number of columns a cell should span.
- ❖ The border attribute of the table element allows the user to specify a border for making the table visible in a Web page.
- ❖ Tables allow the user to organize the data. It enables the developer to design a Web page having an attractive page layout.
- ❖ Multimedia is a combination of various elements such as video, graphics, sound, and text.
- ❖ There are various media types used for audio and video files on different Websites.
- ❖ The <audio> element will help the developer to embed music on the Website and allow the user to listen to music.
- ❖ Users can play the audio in older browsers using the <embed> tag.
- ❖ The <video> element is used for embedding the video content on the Web page.
- ❖ Preload attribute identifies whether the audio has to be loaded when the page loads and is ready to execute.
- ❖ WebM is a new open source video container format supported by Google.

48

### Instructions to the Trainer(s):

- Show students Slide 48.
- Summarize the session by reading out each point on the Slide.

## Session 11: JavaScript - I

### 11.1 Pre-Class Activities

Before you commence the session, you should familiarize yourself with the topics of this session in-depth. Prepare a question or two that will be a key point to relate the current session objectives.

#### 11.1.1 Teaching Skills

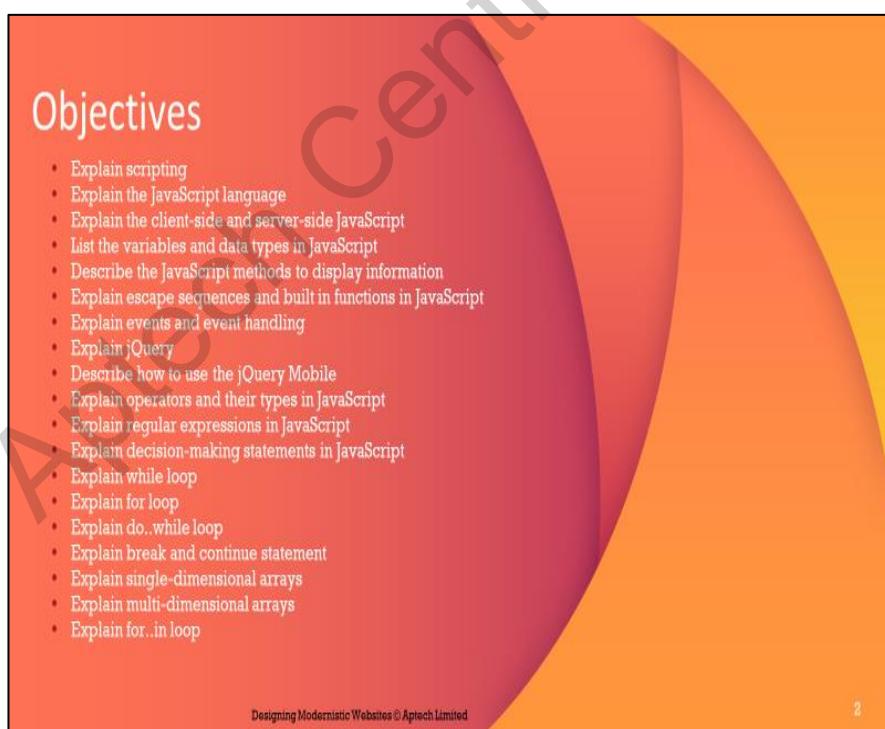
To teach this session, you should be well versed with JavaScript language, jQuery, and loops

You should teach the concepts in the theory class using the images provided. For teaching in the class, you are expected to use slides and LCD projectors.

#### In-Class Activities

Follow the order given here during In-Class activities.

Slide 2



#### Instructions to the Trainer(s):

Give students a brief overview of the current session through the session objectives listed in Slide 2.

## 11.2 In-Class Explanations

Slide 3

### Introduction

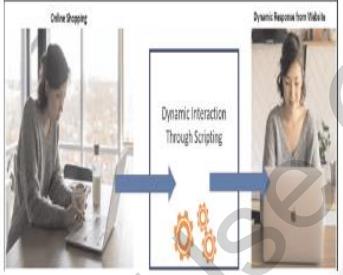
Consider an organization that provides a Website that allows its customers to view their products.

The company has received frequent customer feedbacks to provide the shopping facility online.

For example, details such as credit card number, email, and phone number entered by the customer must be in a proper format.

The developer can handle all these critical tasks by using a scripting language.

A scripting language refers to a set of instructions that provides some functionality when the user interacts with a Web page.



Designing Modernistic Websites © Aptech Limited

3

### Instructions to the Trainer(s):

- Using Slide 3, tell the students that this session introduces them to basics of scripting.
- It introduces them to JavaScript language which is used to provide dynamic functionality and validation on the HTML pages. They will learn about JavaScript variable, data types, built-in functions, and methods.
- Using Slide 3, you can explain the example on JavaScripting.

## Scripting

Scripting refers to a series of commands that are interpreted and executed sequentially and immediately on occurrence of an event.

This event is an action generated by a user while interacting with a Web page.

Examples of events include button clicks, selecting a product from a menu, and so on.

A scripting language refers to a set of instructions that provides some functionality when the user interacts with a Web page.

Scripting languages are often embedded in the HTML pages to change the behavior of the Web pages according to the user's requirements.

**Client-side Scripting:**

- Refers to a script being executed on the client's machine by the browser.

**Server-side Scripting:**

- Refers to a script being executed on a Web server to generate dynamic HTML pages.

```
graph TD; Scripting[Scripting] --> ClientSide[Client-side Scripting]; Scripting --> ServerSide[Server-side Scripting];
```

Designing Modernistic Websites © Aptech Limited

4

### Instructions to the Trainer(s):

- Using Slide 4, explain scripting.
- A scripting language refers to a set of instructions that provides some functionality when the user interacts with a Web page.
- Scripting languages are often embedded in the HTML pages to change the behavior of the Web pages according to the user's requirements.
- Tell the students that some of the scripts can be written to be executed on the occurrence of an event on the Web page. Examples of events include button clicks, selecting a product from a menu, and so on.
- Explain the necessity for scripting and types of scripting languages as explained on the Slides.

## JavaScript

JavaScript is a scripting language that allows building dynamic Web pages by ensuring maximum user interactivity.

JavaScript language is an object-based language, which means that it provides objects for specifying functionalities.

In real life, an object is a visible entity such as a car or a table having some characteristics and capable of performing certain actions.

Similarly, in a scripting language, an object has a unique identity, state, and behavior.

The identity of the object distinguishes it from other objects of the same type.

The state of the object refers to its characteristics, whereas the behavior of the object consists of its possible actions.

The object stores its identity and state in fields (also called variables) and exposes its behavior through functions (actions).

Designing Modernistic Websites © Aptech Limited

5

### Instructions to the Trainer(s):

- Using Slide 5, explain JavaScript.
- Explain the object-oriented features of JavaScript language to the students.
- Tell them that an object-based language allows you to work with real-world objects for specifying functionalities. As In real life, an object is a visible entity such as a car or a table.
- Similarly, software every object has some characteristics and is capable of performing certain actions. Similarly, in a scripting language, an object has a unique identity, state, and behavior.
- The object stores its identity and state in fields (also called variables) and exposes its behavior through functions (actions).

**Versions of JavaScript**

- The first version of JavaScript was developed by Brendan Eich at Netscape in 1995 and was named JavaScript 1.0.
- Following table lists various versions of JavaScript language:

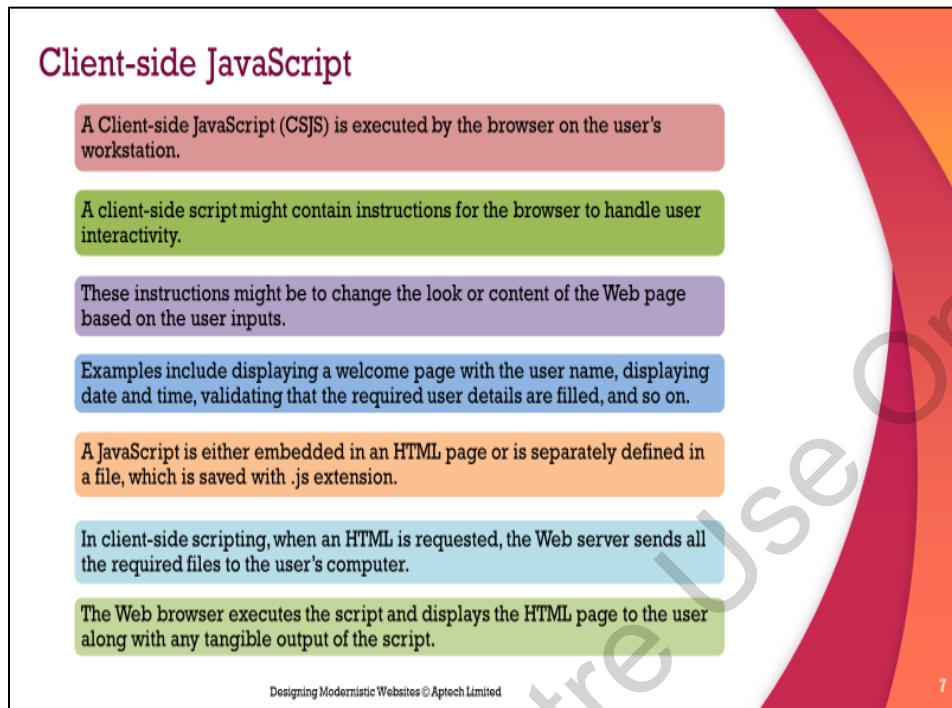
Edition	Name	Description
1	ECMAScript 1 (1997)	First edition
2	ECMAScript 2 (1998)	Supported by Internet Explorer from version 4.0
3	ECMAScript 3 (1999)	Added regular expressions and try/catch and was supported by Internet Explorer 5.0, Netscape Navigator 4.0, and Opera 5.0 onwards
5	ECMAScript 5 (2009)	Added features such as 'strict mode' JSON support, String.trim(), Array.isArray(), and Array iteration methods. Is supported by Internet Explorer 6.0 and Mozilla Firefox 1.0 onwards
6	ECMAScript 2015	Was published in 2015 and added features such as let and const, default parameter values, Array.find(), and Array.findIndex()
7	ECMAScript 2016	Was published in 2016 and added exponential operator and Array.prototype.includes
8	ECMAScript 2017	Was published in 2017 and added features such as string padding, Object.entries, Object.values, async functions, and shared memory
9	ECMAScript 2018	Was published in 2018 and added features such as rest/spread properties, asynchronous iteration, and Promise.finally()
10	ECMAScript 2019	Was published in 2019 and added features such as Array.prototype.flat, Array.prototype.flatMap, and Object.fromEntries
11	ECMAScript 2020	Was published in June 2020 and introduces a BigInt primitive type for arbitrary-sized integers, nullish coalescing operator, and globalThis object
12	ECMAScript 2021	Was published in June 2021 and introduces replaceAll method for Strings, Promise.any, AggregateError, WeakRef, and other features

Designing Modernistic Websites © Aptech Limited

6

### Instructions to the Trainer(s):

- Using Slide 6, explain the versions of JavaScript.
- The first version of JavaScript was developed by Brendan Eich at Netscape in 1995 and was named JavaScript 1.0.
- Netscape Navigator 2.0 and Internet Explorer 3.0 supported JavaScript 1.0.
- Over the period, it gradually evolved with newer versions where each version provided better features and functionalities as compared to their previous versions.
- Explain the table that lists various versions of JavaScript language.



The slide features a decorative background with a red-to-orange gradient on the right side and a large, semi-transparent watermark reading "EduCentral Centre Only" diagonally across the center.

## Client-side JavaScript

A Client-side JavaScript (CSJS) is executed by the browser on the user's workstation.

A client-side script might contain instructions for the browser to handle user interactivity.

These instructions might be to change the look or content of the Web page based on the user inputs.

Examples include displaying a welcome page with the user name, displaying date and time, validating that the required user details are filled, and so on.

A JavaScript is either embedded in an HTML page or is separately defined in a file, which is saved with .js extension.

In client-side scripting, when an HTML is requested, the Web server sends all the required files to the user's computer.

The Web browser executes the script and displays the HTML page to the user along with any tangible output of the script.

Designing Modernistic Websites © Aptech Limited

7

### Instructions to the Trainer(s):

- Using Slide 7, explain Client-side JavaScript.
- JavaScript is a scripting language, which can be executed on the client-side and on the server-side.
- A client-side JavaScript (CSJS) is executed by the browser on the user's workstation. A client-side script might contain instructions for the browser to handle user interactivity.
- These instructions might be to change the look or content of the Web page based on the user inputs.
- Examples include displaying a welcome page with the username, displaying date and time, validating that the required user details are filled, and so on.
- Mention, JavaScript is either embedded in an HTML page or is separately defined in a file, which is saved with .js extension.
- In client-side scripting, when an HTML is requested, the Web server sends all the required files to the user's computer.
- The Web browser executes the script and displays the HTML page to the user along with any tangible output of the script.

## Server-side JavaScript

- A Server-side JavaScript (SSJS) is executed by the Web server when an HTML page is requested by a user and the output is displayed by the browser.
- A server-side JavaScript can interact with the database, fetch the required information specific to the user, and display it to the user.
- Server-side scripting fulfills the goal of providing dynamic content in Web pages.
- Unlike client-side JavaScript, HTML pages using server-side JavaScript are compiled into bytecode files on the server.
- A JavaScript is either embedded in an HTML page or is separately defined in a file, which is saved with .js extension.
- Compilation is a process of converting the code into machine-independent code.
- This machine-independent code is known as the bytecode, which is an executable file that the Web server runs to generate the desired output.

Designing Modernistic Websites © Aptech Limited

8

### Instructions to the Trainer(s):

- Using Slide 8, explain Server-side JavaScript.
- A server-side JavaScript (SSJS) is executed by the Web server when an HTML page is requested by a user.
- The output of a server-side JavaScript is sent to the user and is displayed by the browser.
- In this case, a user might not be aware that a script was executed on the server to produce the desirable output.
- A server-side JavaScript can interact with the database, fetch the required information specific to the user, and display it to the user.
- This means that server-side scripting fulfills the goal of providing dynamic content in Web pages.

## <script> Tag

- The <script> tag defines a script for an HTML page to make them interactive.
- The browser that supports scripts interprets and executes the script specified under the <script> tag when the page loads in the browser.
- You can directly insert a JavaScript code under the <script> tag.
- You can define multiple <script> tags either in the <head> or in the <body> elements of an HTML page.
- In HTML5, the type attribute specifying the scripting language is no longer required as it is optional.

• The Code Snippet demonstrates the use of the tag.

```
<!doctype html>
<html>
<head>
<script>
    document.write("Welcome to the Digital World");
</script>
</head>
<body>
    .....
</body>
</html>
```

There are two main purposes of the <script> tag, which are as follows:

- Identifies a given segment of script in the HTML page.
- Loads an external script file.

Designing Modernistic Websites © Aptech Limited

9

### Instructions to the Trainer(s):

- Using Slide 9, explain the <script> tag.
- Mention, <script> tag defines a script for an HTML page to make them interactive. The browser that supports scripts interprets and executes the script specified under the <script> tag when the page loads in the browser.
- Mention, JavaScript code under the <script> tag. You can define multiple <script> tags either in the <head> or in the <body> elements of an HTML page.
- In HTML5, the type attribute specifying the scripting language is no longer required as it is optional.
- Then, explain the HTML code using <script> tag and uses of <script> tag. There are two main purposes of the <script> tag, which are as follows:
  - Identifies a given segment of script in the HTML page
  - Loads an external script file

### In-Class Question:

**Question:** Which tag is used to insert JavaScript code in a HTML file?

**Answer:** <script> tag is used to insert JavaScript code in a HTML file.

## Variables in JavaScript

A variable refers to a symbolic name that holds a value, which keeps changing.

For example, age of a student and salary of an employee can be treated as variables.

In JavaScript, a variable is a unique location in computer's memory that stores a value and has a unique name.

The name of the variable is used to access and read the value stored in it.

A variable can store different types of data such as a character, a number, or a string.

Designing Modernistic Websites © Aptech Limited

10

### Instructions to the Trainer(s):

- Using Slide 10, explain Variables in JavaScript.
- A variable refers to a symbolic name that holds a value, which keeps changing.
- For example, age of a student and salary of an employee can be treated as variables. A real-life example for variables includes the variables used in algebraic expressions that store values.
- In JavaScript, a variable is a unique location in the computer's memory that stores a value and has a unique name. The name of the variable is used to access and read the value stored in it.
- A variable can store different types of data such as a character, a number, or a string.
- Therefore, a variable acts as a container for saving and changing values during the execution of the script.

## Declaring Variables

- Declaring a variable refers to creating a variable by specifying the variable name.
- For example, one can create a variable named to store the name of a student.

In JavaScript,

- the var keyword is used to create a variable by allocating memory to it.
- a keyword is a reserved word that holds a special meaning.
- the variable can be initialized at the time of creating the variable or later.
- initialization refers to the task of assigning a value to a variable.
- once the variable is initialized, you can change the value of a variable as required.
- variables allow keeping track of data during the execution of the script.
- one can declare and initialize multiple variables in a single statement.
- while referring to a variable, you are referring to the value of that variable.

Syntax:

```
var <variableName>;
```

where,

- var: Is the keyword in JavaScript.
- variableName: Is a valid variable name.

Designing Modernistic Websites © Aptech Limited

11

### Instructions to the Trainer(s):

- Using Slide 11, explain how to declare variables.
- Declaring a variable refers to creating a variable by specifying the variable name. For example, you can create a variable named studName to store the name of a student. Here, the variable name studName is referred to as an identifier.
- In JavaScript, the var keyword is used to create a variable by allocating memory to it. A keyword is a reserved word that holds a special meaning in JavaScript.
- Mention, variable can be initialized at the time of creating the variable or later.
- Initialization refers to the task of assigning a value to a variable. Once the variable is initialized, you can change the value of a variable as required.
- Variables allow keeping track of data during the execution of the script. While referring to a variable, you are referring to the value of that variable.
- In JavaScript, you can declare and initialize multiple variables in a single statement.
- Then, explain the syntax to declare variables.

## Variable Naming Rules

- JavaScript is a case-sensitive language.
- The variables X and x are treated as two different variables.
- JavaScript consists of certain rules for naming a variable as follows:

In JavaScript, a variable name

- can consist of digits, underscore, and alphabets.
- must begin with a letter or underscore character.
- cannot begin with a number and cannot contain any punctuation marks.
- cannot contain any kind of special characters such as +, \*, %, and so on.
- cannot contain spaces.
- cannot be a JavaScript keyword.

Designing Modernistic Websites © Aptech Limited

12

### Instructions to the Trainer(s):

- Using Slide 12, explain variable naming rules.
- Mention, JavaScript is a case-sensitive language.
- This means that if you specify X and x as variables, both of them are treated as two different variables.
- Similarly, in JavaScript, there are certain rules, which must be followed while specifying variables names.
- Then, explain the rules for a variable name listed on the slide.

## Data Types in JavaScript 1-2

- To identify the type of data that can be stored in a variable, JavaScript provides different data types.
- Data types in JavaScript are classified into two broad categories namely, primitive and composite data types.
- Primitive data types contain only a single value, whereas the composite data types contain a group of values.

### ➤ PRIMITIVE DATA TYPES

- A primitive data type contains a single literal value such as a number or a string.
- A literal is a static value that you can assign to variables.

- Following table lists the primitive data types:

Primitive Data Type	Description
boolean	Contains only two values namely, true or false
null	Contains only one value namely, null. A variable of this value specifies that the variable has no value. This null value is a keyword and it is not the same as the value, zero
number	Contains positive and negative numbers and numbers with decimal point. Some of the valid examples include 6, 7.5, -8, 7.5e-3, and so on
string	Contains alphanumeric characters in single or double quotation marks. Single quotes are used to represent a string, which itself consists of quotation marks. A set of quotes without any characters within it is known as the null string

Designing Modernistic Websites © Aptech Limited

13

## Data Types in JavaScript 2-2

### ➤ COMPOSITE DATA TYPES

- A composite data type stores a collection of multiple related values, unlike primitive data types.
- In JavaScript, all composite data types are treated as objects.
- A composite data type can be either predefined or user-defined in JavaScript.

- Following table lists the composite data types:

Composite Data Type	Description
Objects	Refers to a collection of properties and functions. Properties specify the characteristics and functions determine the behavior of a JavaScript object
Functions	Refers to a collection of statements, which are instructions to achieve a specific task
Arrays	Refers to a collection of values stored in adjacent memory locations

Designing Modernistic Websites © Aptech Limited

14

**Instructions to the Trainer(s):**

- Using Slides 13 and 14, explain the data types in JavaScript.
- A Web page designer can store different types of values such as numbers, characters, or strings in variables.
- However, the Web page designer must know what kind of data a particular variable is expected to store.
- To identify the type of data that can be stored in a variable, JavaScript provides different data types.
- A Web page designer does not have to specify the data type while declaring variables. Thus, JavaScript is referred to as the loosely typed language.
- This means that a variable holding a number can also hold a string value later. The values of variables are automatically mapped to their data types when the script is executed in the browser.
- Mention, data types in JavaScript are classified into two broad categories namely, primitive and composite data types.
  - Primitive data types contain only a single value, whereas the composite data types contain a group of values.
- A primitive data type contains a single literal value such as a number or a string. A literal is a static value that you can assign to variables.
- Using Slide 13, explain different types of primitive data types. Explain, table which lists the primitive data types.
- Using Slide 14, explain the composite data type and its types. Mention, composite data type stores a collection of multiple related values, unlike primitive data types. In JavaScript, all composite data types are treated as objects.
- A composite data type can be either predefined or user-defined in JavaScript.
- Explain, table which lists the composite data types.

## Methods

- JavaScript allows you to display information using the methods of the document object.
- The document object is a predefined object in JavaScript, which represents the HTML page and allows managing the page dynamically.
- Each object in JavaScript consists of methods, that fulfill a specific task.
- There are two methods of the document object, that display any type of data in the browser:
  - `write()`: Displays any type of data.
  - `writeln()`: Displays any type of data and appends a new line character.
- The syntax demonstrates the use of `document.write()` method, which allows you to display information in the displayed HTML page.

### Syntax:

```
document.write("<data>" + variables);
```

where,

- **data**: Specifies strings enclosed in double quotes.
- **variables**: Specify variable names whose value should be displayed on the HTML page.

Designing Modernistic Websites © Aptech Limited

15

### Instructions to the Trainer(s):

- Using Slide 15, explain methods in JavaScript.
- JavaScript allows you to display information using the methods of the document object. The document object is a predefined object in JavaScript, which represents the HTML page and allows managing the page dynamically.
- Tell the students that the HTML page elements are referred to as object in JavaScript. To work with these objects JavaScript accesses them as objects. The object model on HTML page is also referred to as HTML Document Object Model (DOM).
- The browser creates a DOM of the Web page when it is loaded.
- There are two methods of the document object, which display any type of data in the browser. These methods are as follows:
  - `write()` : Displays any type of data.
  - `writeln()` : Displays any type of data and appends a new line character.

## Using Comments

- Comments provide information about a piece of code in the script.
- When the script is executed, the browser identifies comments as they are marked with special characters and does not display them.
- JavaScript supports two types of comments. These are as follows:

➤ **SINGLE LINE COMMENTS**

- Single-line comments begin with two forward slashes (//). You can insert single-line comments as follows:

```
// This statement declares a variable named num.  
var num;
```

➤ **MULTI-LINE COMMENTS**

- Multi-line comments begin with a forward slash followed by an asterisk /\*) and end with an asterisk followed by a forward slash (\*/).
- You can insert multiple lines of comments as follows:

```
/* This line of code  
declares a variable */  
var num;
```

Designing Modernistic Websites © Apneet Limited

16

### Instructions to the Trainer(s):

- Using Slide 16, explain comments in JavaScript.
- A Web page designer might code complex script to fulfil a specific task.
- In JavaScript, a Web page designer specifies comments to provide information about a piece of code in the script.
- Comments describe the code in simple words so that somebody who reads the code can understand the code.
- Comments are small piece of text that makes the program more readable. While the script is executed, the browser can identify comments as they are marked with special characters and do not display them.
- JavaScript supports two types of comments. These are as follows:
  - Single-line Comments
  - Multi-line Comments

## Escape Sequence Characters

- An escape sequence character is a special character that is preceded by a backslash (\).
- Escape sequence characters are used to display special non-printing characters such as a tab space, a single space, or a backspace.
- In JavaScript, the escape sequence characters must be enclosed in double quotes.
- Following table lists the escape sequence characters:

Escape Sequence	Non-Printing Character
\b	Back space
\f	Form feed
\n	New line
\r	Carriage return
\t	Horizontal tab
\'	Single quote
\"	Double quote
\\\	Backslash
\aaa	Matches a Latin-1 encoding character using octal representation, where aaa are three octal numbers. For example, \251 represents the copyright symbol
\xaa	Matches a Latin-1 encoding character using hexadecimal representation, where aa are two hexadecimal numbers. For example, \x61 represents the character 'a'
\uaaaa	Represent the Unicode encoding character, where aaaa are four hexadecimal numbers. For example, the character \u0020 represents a space

Designing Modernistic Websites © Aptech Limited

17

### Instructions to the Trainer(s):

- Using Slide 17, explain the escape sequence characters used in JavaScript.
- An escape sequence character is a special character that is preceded by a backslash (\) .
- Escape sequence characters are used to display special non-printing characters such as a tab space, a single space, or a backspace. These non-printing characters help in displaying formatted output to the user to maximize readability.
- The backslash character specifies that the following character denotes a non-printing character.
- For example, \t is an escape sequence character that inserts a tab space similar to the Tab key of the keyboard. In JavaScript, the escape sequence characters must always be enclosed in double quotes.
- Explain the multiple escape sequence characters in JavaScript that provide various kind of formatting with the help of table listed on the slides.

## Built-in Functions

- A function is a piece of code that performs some operations on variables to fulfill a specific task.
- It takes one or more input values, processes them, and returns an output value.
- Following table lists the built-in JavaScript functions:

Function	Description	Example
alert()	Displays a dialog box with some information and OK button	alert("Please fill all the fields of the form"); Displays a message box with the instruction
confirm()	Displays a dialog box with OK and Cancel buttons. It verifies an action, which a user wants to perform	confirm("Are you sure you want to close the page?"); Displays a message box with the question
parseInt()	Converts a string value into a numeric value	parseInt("25 years");
parseFloat()	Converts a string into a number with decimal point	parseFloat("10.33"); Returns 10.33
eval()	Evaluates an expression and returns the evaluated result	eval("2+2"); Returns 4
isNaN()	Checks whether a value is not a number	isNaN("Hello"); Returns true
prompt()	Displays a dialog box that accepts an input value through a text box. It also accepts the default value for the text box	prompt("Enter your name", "Name"); Displays the message in the dialog box and Name in the text box.

Designing Modernistic Websites © Aptech Limited

18

### Instructions to the Trainer(s):

- Using Slide 18, explain the built-in functions.
- A function is a piece of code that performs some operations on variables to fulfil a specific task. It takes one or more input values, processes them, and returns an output value.
- JavaScript provides built-in functions that are already defined to fulfil a certain task.
- Explain table which lists the built-in functions.

## Events

- An event occurs when a user interacts with the Web page.
- Some of the commonly generated events are mouse clicks, key strokes, and so on.
- The process of handling these events is known as event handling.
- Following figure displays the event:



Designing Modernistic Websites © Aptech Limited

19

### Instructions to the Trainer(s):

- Using Slide 19, explain the concept 'events'.
- Consider a scenario where you want to design an Employee registration Web form. This form allows the users to fill in the appropriate details and click the submit button.
- When user clicks the Submit button, the form data is submitted to the server for validation purposes. In this case, when the user clicks the button, an event is generated. The submission of form refers to the action performed on click of the button.
- An event occurs when a user interacts with the Web page. Some of the commonly generated events are mouse clicks, key strokes, and so on.
- The process of handling these events is known as event handling.
- Explain the figure displays the event on mouse click.
- Discuss more events occurrence noticed by the students while working on Operating System (OS) or accessing the Web pages with forms.
- Some more examples of events are:
  - An event occurring on page load in the browser.
  - While moving from one field to another.

## Event Handling

Event handling is a process of specifying actions to be performed when an event occurs. This is done by using an event handler.

An event handler is a scripting code or a function that defines the actions to be performed when the event is triggered.

When an event occurs, an event handler function that is associated with the specific event is invoked.

The information about this generated event is updated on the `event object`.

The `event object` is a built-in object, which can be accessed through the `window object`.

It specifies the event state, including information such as the location of mouse cursor, element on which an event occurred, and state of the keys in a keyboard.

Designing Modernistic Websites © Aptech Limited

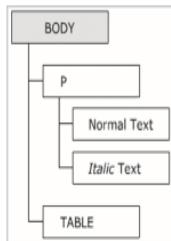
20

### Instructions to the Trainer(s):

- Using Slide 20, explain event handling.
- Mention, event handling is a process of specifying actions to be performed when an event occurs. This is done by using an event handler. An event handler is a scripting code or a function that defines the actions to be performed when the event is triggered.
- When an event occurs, an event handler function that is associated with the specific event is invoked. The information about this generated event is updated on the `event object`. The `event object` is a built-in object, which can be accessed through the `window object`.
- It specifies the event state, which includes information such as the location of mouse cursor, element on which an event occurred, and state of the keys in a keyboard.

## Event Bubbling

- Event bubbling is a mechanism that allows a user to specify a common event handler for all child elements.
- This means that the parent element handles all the events generated by the child elements.
- For example, consider a Web page that consists of a paragraph and a table. The paragraph consists of multiple occurrences of italic text.
- To change the color of each italic text of a paragraph when the user clicks a particular button, instead of declaring an event handler for each italic text, one can declare it within the `<P>` element.
- Following figure displays the event bubbling:



Designing Modernistic Websites © Aptech Limited

21

### Instructions to the Trainer(s):

- Using Slide 21, explain concept of event bubbling.
- Explain the event bubbling mechanism in JavaScript along with the figure displaying the event bubbling.

## Life Cycle of an Event

- An event's life starts when the user performs an action to interact with the Web page.
- It finally ends when the event handler provides a response to the user's action.
- Steps involved in the life cycle of an event are as follows:

1. The user performs an action to raise an event.
2. The event object is updated to determine the event state.
3. The event is fired.
4. The event bubbling occurs as the event bubbles through the elements of the hierarchy.
5. The event handler is invoked that performs the specified actions.

### Instructions to the Trainer(s):

- Using Slide 22, explain life cycle of an event.
- An event's life starts when the user performs an action to interact with the Web page. It finally ends when the event handler provides a response to the user's action.
- Explain the steps involved in the life cycle of an event as specified in the slide.

## Keyboard Events

- Keyboard events are the events that occur when a key or a combination of keys are pressed or released from a keyboard.
- These events occur for all keys of a keyboard.
- Different keyboard events are as follows:

### Onkeydown

- Occurs when a key is pressed down.

### Onkeyup

- Occurs when the key is released.

### Onkeypress

- Occurs when a key is pressed and released.

Designing Modernistic Websites © Aptech Limited

23

### Instructions to the Trainer(s):

- Using Slide 23, explain different keyboard events.
- Keyboard events are the events that occur when a key or a combination of keys are pressed or released from a keyboard. These events occur for all keys of a keyboard.
- Different keyboard events are as follows:
  - Onkeydown : Occurs when a key is pressed down.
  - Onkeyup : Occurs when the key is released.
  - Onkeypress : Occurs when a key is pressed and released.

## Mouse Events

- Mouse events occur when the user clicks the mouse button.
- Following table lists the mouse events:

Event	Description
onmousedown	Occurs when the mouse button is pressed
onmouseup	Occurs when the mouse button is released
onclick	Occurs when the mouse button is pressed and released
ondblclick	Occurs when the mouse button is double-clicked
onmousemove	Occurs when the mouse pointer is moved from one location to other
onmouseover	Occurs when the mouse pointer is moved over the element
onmouseout	Occurs when the mouse pointer is moved out of the element

Designing Modernistic Websites © Aptech Limited

24

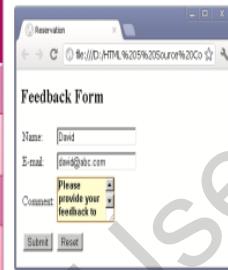
### Instructions to the Trainer(s):

- Using Slide 24, explain the mouse events.
- Mouse events occur when the user clicks the mouse button.
- Explain table which lists the mouse events.

## Focus and Selection Events

- The focus events determine the activation of various elements that uses the element.
- It allows a user to set or reset focus for different elements.
- The selection events occur when an element or a part of an element within a Web page is selected.
- Following table lists the focus and selection events:

Data Type	Description
onfocus	Occurs when an element receives focus
onblur	Occurs when an element loses focus
onselectstart	Occurs when the selection of an element starts
onselect	Occurs when the present selection changes
ondragstart	Occurs when the selected element is moved



Designing Modernistic Websites © Apneet Limited

25

### Instructions to the Trainer(s):

- Using Slide 25, explain different focus and selection events.
- The focus events determine the activation of various elements that uses the input element. It allows you to set or reset focus for different input elements.
- The selection events occur when an element or a part of an element within a Web page is selected. Explain table which lists the focus and selection events.

### In-Class Question:

**Question:** Which event occurs when an element receives focus?

**Answer:** onfocus event occurs when an element receives focus.

## jQuery

- Following are key features supported by jQuery:

Event Handling	jQuery has a smart way to capture a wide range of events, such as user clicks a link, without making the HTML code complex with event handlers.
Animations	jQuery has many built-in animation effects that the user can use while developing their Websites.
DOM Manipulation	jQuery easily selects, traverses, and modifies DOM by using the cross-browser open source selector engine named Sizzle.
Cross Browser Support	jQuery has a support for cross-browser and works well with following browsers: <ul style="list-style-type: none"><li>Internet Explorer 6 and above</li><li>Firefox 2.0 and above</li><li>Safari 3.0 and above</li><li>Chrome</li><li>Opera 9.0 and above</li></ul>
Lightweight	jQuery has a lightweight library of 19 KB size.
AJAX Support	jQuery helps you to develop feature-rich and responsive Web sites by using AJAX technologies.
Latest Technology	jQuery supports basic XPath syntax and CSS3 selectors.

Designing Modernistic Websites © Aptech Limited

26

### Instructions to the Trainer(s):

- Using Slide 26, explain the jQuery.
- jQuery is a short and fast JavaScript library developed by John Resig in 2006 with a wonderful slogan: Write less and do more. It simplified the client-side scripting of HTML.
- jQuery also simplifies HTML files animation, event handling, traversing, and developing AJAX based Web applications. It helps in rapid Web application development.
- jQuery is designed for simplifying several tasks by writing lesser code.
- Explain the features of jQuery specified on the slide.

## Using jQuery Library

- To work with jQuery perform following steps:

1. Download the jQuery library from the <http://jquery.com/> Website.

2. Place the jquery-1.7.2.min.js file in the current directory of the Website.

- The user can include jQuery library in their file.
- The Code Snippet shows how to use a jQuery library.

```
<!DOCTYPE HTML>
<html>
<head>
<title>The jQuery Example</title>
    // Using jQuery library
    <script src="jquery-1.7.2.min.js">
        // The user can add our JavaScript code here
    </script>
</head>
<body>
</body>
</html>
```

Designing Modernistic Websites © Aptech Limited

27

### Instructions to the Trainer(s):

- Using Slide 27, explain the how to use JQuery library.
- There is an easy way to use jQuery library. To work with jQuery perform the following steps:
  1. Download the jQuery library from the <http://jquery.com/> Web site
  2. Place the jquery-1.7.2.min.js file in the current directory of the Website. The user can include jQuery library in their file.
- Code Snippet shows how to use a jQuery library.

## Calling jQuery Library Functions

- While jQuery is reading or manipulating the Document Object Model (DOM) object, the users can add the events when the DOM object is ready.
- If the user wants the event on their page then, the user has to call the event in the `$(document).ready()` function.
- Users also register the ready event for the document.
- Place the `jquery-1.7.2.min.js` file in the current directory and specify the location of this file in the `src` attribute.
- The Code Snippet shows how to call jQuery library function and ready event in DOM.

### Instructions to the Trainer(s):

- Using Slide 28, explain the process of calling a jQuery library function.
- Users can do many tasks while jQuery is reading or manipulating the DOM object.
- The users can add the events only when the DOM object is ready. If the user wants the event on their page then the user has to call the event in the `$(document).ready()` function.
- All the content inside the event will be loaded as soon as the DOM is loaded, but before the contents of the page are loaded. The users also register the ready event for the document.
- Place the `jquery-1.7.2.min.js` file in the current directory and specify the location of this file in the `src` attribute.

## jQuery Mobile

- jQuery mobile is a Web User Interface (UI) development framework that allows the user to build mobile Web applications that work on tablets and smartphones.
- The jQuery mobile framework provides many facilities that include XML DOM and HTML manipulation and traversing, performing server communication, handling events, image effects, and animation for Web pages.
- Basic features of jQuery mobile are as follows:

<b>Simplicity</b>	<b>Accessibility</b>	<b>Enhancements and Degradation</b>	<b>Themes</b>	<b>Smaller Size</b>
This framework is easy to use and allows developing Web pages by using markup driven with minimum or no JavaScript.	The framework supports Accessible Rich Internet Applications (ARIA) that helps to develop Web pages accessible to visitors with disabilities.	The jQuery mobile is influenced by the latest HTML5, JavaScript, and CSS3.	This framework provides themes that allow the user to provide their own styling.	The size for jQuery mobile framework is smaller for CSS it is 6 KB and for JavaScript library it is 12 KB.

Designing Modernistic Websites © Apnech Limited

29

### Instructions to the Trainer(s):

- Using Slide 29, explain the jQuery mobile and its basic functions.
- jQuery mobile is a Web User Interface (UI) development framework that allows the user to build mobile Web applications that work on tablets and smartphones.
- The jQuery mobile framework provides many facilities that include XML, DOM, and HTML manipulation and traversing, performing server communication, handling events, image effects, and animation for Web pages.
- Basic features of jQuery mobile are as follows:
  - **Simplicity:** This framework is easy to use and allows developing Web pages by using markup driven with minimum or no JavaScript.
  - **Accessibility:** The framework supports Accessible Rich Internet Applications (ARIA) that helps to develop Web pages accessible to visitors with disabilities.
  - **Enhancements and Degradation:** The jQuery mobile is influenced by the latest HTML5, JavaScript, and CSS3.
  - **Themes:** This framework provides themes that allow the user to provide their own styling.
  - **Smaller Size:** The size for jQuery mobile framework is smaller for CSS it is 6 KB and for JavaScript library it is 12 KB.

## Introduction to JavaScript Operators

An operator specifies the type of operation to be performed on the values of variables and expressions.

JavaScript provides different types of operators to perform simple to complex calculations and evaluations.

Certain operators are also used to construct relational and logical statements. These statements allow implementing decision and looping constructs.

### Instructions to the Trainer(s):

- Using Slide 30, introduce the JavaScript Operators.
- An operator specifies the type of operation to be performed on the values of variables and expressions.
- JavaScript provides different types of operators to perform simple to complex calculations and evaluations.
- Certain operators are also used to construct relational and logical statements.
- These statements allow implementing decision and looping constructs.

## Basics of Operators

An operation is an action performed on one or more values stored in variables.

The specified action either changes the value of the variable or generates a new value.

An operation requires minimum one symbol and some value.

Symbol is called an operator and it specifies the type of action to be performed on the value.

Value or variable on which the operation is performed is called an operand.

- Three main types of operators are as follows:

Unary operators - Operates on a single operand.

Binary operators - Operates on two operands.

Ternary operators - Operates on three operands.

### Instructions to the Trainer(s):

- Using Slide 31, explain the basics of operators in JavaScript.
- An operation is an action performed on one or more values stored in variables.
- The specified action either changes the value of the variable or generates a new value. Here, the symbol is called an operator and it specifies the type of action to be performed on the value.
- The value or variable on which the operation is performed is called an operand.
- For example,  $X*2$  is an expression, where  $X$  and  $2$  are operands and  $*$  is an operator. There are three main types of operators, which are as follows:
  - Unary operators** - Operates on a single operand. For example, the expression  $y = -x$ .
  - Binary operators** - Operates on two operands. For example, the expression  $sum = y + x$ .
  - Ternary operators** - Operates on three operands. For example, the expression  $age >= 18 ? "Eligible" : "Not Eligible"$ .

## Operators and their Types

Operators help in simplifying expressions.

JavaScript provides a predefined set of operators that allow performing different operations.

JavaScript operators are classified into six categories based on the type of action they perform on operands.

- Six categories of operators are as follows:
  - Arithmetic operators
  - Relational operators
  - Logical operators
  - Assignment operators
  - Bitwise operators
  - Special operators

Designing Modernistic Websites © Aptech Limited

32

### Instructions to the Trainer(s):

- Using Slide 32, explain the operators and its types.
- Operators help in simplifying expressions. JavaScript provides a predefined set of operators that allow performing different operations.
- JavaScript operators are classified into six categories based on the type of action they perform on operands.
- These six categories of operators are as follows:
  - Arithmetic operators
  - Relational operators
  - Logical operators
  - Assignment operators
  - Bitwise operators
  - Special operators

## Arithmetic Operators

- Are binary operators.
- Perform basic arithmetic operations on two operands.
- Operator appears in between the two operands, which allow you to perform computations on numeric and string values.

• Following table lists arithmetic operators:

Arithmetic Operator	Description	Example
+ (Addition)	Performs addition. In case of string values, it behaves as a string concatenation operator and appends a string at the end of the other	45 + 56
- (Subtraction)	Performs subtraction. If a larger value is subtracted from a smaller value, it returns a negative numeric value	76 - 78
/ (Division)	Divides the first operand by the second operand and returns the quotient	24 / 8
% (Modulo)	Divides the first operand by the second operand and returns the remainder	90 % 20
* (Multiplication)	Multiples the two operands	98 * 10

Designing Modernistic Websites © Aptech Limited

33

### Instructions to the Trainer(s):

- Using Slide 33, explain the arithmetic operators.
- Arithmetic operators are binary operators, as they perform basic arithmetic operations on two operands.
- The operator appears in between the two operands, which allows you to perform computations on numeric and string values.
- These computations include addition, subtraction, multiplication, and division.
- Explain table which lists the arithmetic operators with their descriptions and an example of each type on Slide 33.

### In-Class Question:

**Question:** What is the modulo (%) operator used for?

**Answer:** Modulo (%) operator is for finding remainder in a division operation.

## Increment and Decrement Operators

Increment and decrement operators are unary operators.

Increment operator (++) increases the value by 1, while the decrement operator (--) decreases the value by 1.

These operators can be placed either before or after the operand.

Operator if placed before the operand, expression is called pre-increment or pre-decrement. Operator if placed after the operand, expression is called post-increment or post-decrement.

- Following table lists increment and decrement operators:

Expressions	Type	Result
numTwo = ++numOne;	Pre-increment	numTwo = 3
numTwo = numOne++;	Post-increment	numTwo = 2
numTwo = --numOne;	Pre-decrement	numTwo = 1
numTwo = numOne--;	Post-decrement	90 % 20

Designing Modernistic Websites © Aptech Limited

34

### Instructions to the Trainer(s):

- Using Slide 34, explain the increment and decrement operators of JavaScript.
- The increment and decrement operators are unary operators, as they operate only on a single operand.
- The increment operator (++) increases the value by 1, while the decrement operator (--) decreases the value by 1. These operators can be placed either before or after the operand.
- If the operator is placed before the operand, the expression is called pre-increment or pre-decrement.
- If the operator is placed after the operand, the expression is called post-increment or post-decrement.

## Relational Operators

Are binary operators that make a comparison between two operands.

After making a comparison, they return a boolean value namely, true or false.

Expression consisting of a relational operator is called as the relational expression or conditional expression.

- Following table lists relational operators:

Relational Operators	Description	Example
<code>== (Equal)</code>	Verifies whether the two operands are equal	<code>90 == 91</code>
<code>!= (Not Equal)</code>	Verifies whether the two operands are unequal	<code>99 != 98</code>
<code>==== (Strict Equal)</code>	Verifies whether the two operands are equal and are of the same type	<code>numTwo === 1</code>
<code>!== (Strict Not Equal)</code>	Verifies whether the two operands are unequal and whether are not of the same type	<code>90 !== 20</code>
<code>&gt; (Greater Than)</code>	Verifies whether the left operand is greater than the right operand	<code>97 &gt; 95</code>
<code>&lt; (Less Than)</code>	Verifies whether the left operand is less than the right operand	<code>94 &lt; 96</code>
<code>&gt;= (Greater Than or Equal)</code>	Verifies whether the left operand is greater than or equal to the right operand	<code>92 &gt;= 93</code>
<code>&lt;= (Less Than or Equal)</code>	Verifies whether the left operand is less than or equal to the right operand	<code>99 &lt;= 100</code>

Designing Modernistic Websites © Aptech Limited

35

### Instructions to the Trainer(s):

- Using Slide 35, explain relational operators in JavaScript.
- Relational operators are binary operators that make a comparison between two operands. After making a comparison, they return a boolean value namely, true or false.
- The expression consisting of a relational operator is called as the relational expression or conditional expression.
- Explain table which lists the relational operators along with their descriptions and an example of each type.

## Logical Operators

Are binary operators that perform logical operations on two operands.

They belong to the category of relational operators, as they return a boolean value.

- Following table lists the logical operators:

Logical Operators	Description	Example
&& (AND)	Returns true, if either of the operands are evaluated to true. If first operand evaluates to true, it will ignore the second operand	$(x == 2) \&\& (y == 5)$ Returns false
! (NOT)	Returns false, if the expression is true and vice-versa	$!(x == 3)$ Returns true
(OR)	Returns true, if either of the operands are evaluated to true. If first operand evaluates to true, it will ignore the second operand	$(x == 2)    (y == 5)$ Returns true

Designing Modernistic Websites © Aptech Limited

36

### Instructions to the Trainer(s):

- Using Slide 36, explain the logical operators.
- Logical operators are binary operators that perform logical operations on two operands. They belong to the category of relational operators, as they return a boolean value.
- Explain table which lists various logical operators and an example of each type, assuming that x is 2 and y is 2.

## Assignment Operators

Assignment operators assign the value of the right side operand to the operand on the left side by using the equal to operator (=).

Simple assignment operator - Is the '=' operator which is used to assign a value or result of an expression to a variable.

Compound assignment operator - Is formed by combining the simple assignment operator with the arithmetic operators.

- Following table lists the assignment operators:

Expressions	Description	Example
numOne += 6;	numOne = numOne + 6	numOne = 12
numOne -= 6;	numOne = numOne - 6	numOne = 0
numOne *= 6;	numOne = numOne * 6	numOne = 36
numOne %= 6;	numOne = numOne % 6	numOne = 0
numOne /= 6;	numOne = numOne / 6	numOne = 1

Designing Modernistic Websites © Aptech Limited

37

### Instructions to the Trainer(s):

- Using Slide 37, explain the assignment operators.
- Assignment operators assign the value of the right side operand to the operand on the left side by using the equal to operator (=) .
- The assignment operator is divided into two categories in JavaScript that is as follows:
  - Simple assignment operator - Is the '=' operator which is used to assign a value or result of an expression to a variable. For example, `result = numOne + numTwo;`
  - Compound assignment operator - Is formed by combining the simple assignment operator with the arithmetic operators. For example, `salary -= eval(salary * tax / 100);`
- Explain table which demonstrates the use of assignment operator by assuming the value of the variable numOne as 6 .

## Bitwise Operators

- Represent their operands in bits (zeros and ones) and perform operations on them.
- They return standard decimal values.
- Compound assignment operator - Is formed by combining the simple assignment operator with the arithmetic operators.

Following table lists the bitwise operators in JavaScript:

Bitwise Operators	Description	Example
& (Bitwise AND)	Compares two bits and returns 1 if both of them are 1 or else returns 0	00111000 Returns 00011000
- (Bitwise NOT)	Inverts every bits of the operand and is a unary operator	-00010101 Returns 11010101
(Bitwise OR)	Compares two bits and returns 1 if the corresponding bits of either or both the operands is 1	00111000 Returns 00111100
<sup>a</sup> (Bitwise XOR)	Compares two bits and returns 1 if the corresponding bit of either, but not both the operands is 1	<sup>a</sup> ^ b Returns 7 (00000111)

Designing Modernistic Websites © Aptech Limited

38

### Instructions to the Trainer(s):

- Using Slide 38, explain the bitwise operator.
- Bitwise operators represent their operands in bits (zeros and ones) and perform operations on them. However, they return standard decimal values.

## Special Operators

There are some operators in JavaScript which do not belong to any of the categories of JavaScript operators.

Such operators are referred to as the special operators.

- Following table lists the special operators in JavaScript:

Special Operators	Description
, (comma)	Combines multiple expressions into a single expression, operates on them in the left to right order and returns the value of the expression on the right.
? : (conditional)	Operates on three operands where the result depends on a condition. It is also called as ternary operator and has the form condition ? value1:value2. If the condition is true, the operator obtains value1 or else obtains value2.
typeof	Returns a string that indicates the type of the operand. The operand can be a string, variable, keyword, or an object.

Designing Modernistic Websites © Aptech Limited

39

### Instructions to the Trainer(s):

- Using Slide 39, explain special operators.
- There are some operators in JavaScript which do not belong to any of the categories of JavaScript operators.
- Such operators are referred to as the special operators.
- Explain table which lists the most commonly used special operators in JavaScript.

## Operator Precedence

- Following table lists the precedence of the operators from the highest to the lowest and their associativity:

Precedence Order	Operator	Description	Associativity
1	()	Parentheses	Left to Right
2	++, --	Post-increment and Post-decrement operators	Not Applicable
3	typeof, ++, --, !, ~, ~!	Pre-increment and Pre-decrement operators, Logical NOT, Bitwise NOT, and Unary negation	Right to Left
4	*, /, %	Multiplication, Division, and Modulo	Left to Right
5	+, -	Addition and Subtraction	Left to Right
6	<, <=, >, >=	Less than, Less than or equal, Greater than, and Greater than or equal	Left to Right
7	==, ===, !=, !==	Equal to, Strict equal to, Not equal to, and Strict not equal to	Left to Right
8	&,  , ^, &&,	Bitwise AND, Bitwise OR, Bitwise XOR, Logical AND, and Logical OR	Left to Right
9	? :	Conditional operator	Right to Left
10	=, +=, -=, *=, /=, %=	Assignment operators	Right to Left
11	,	Comma	Left to Right

Designing Modernistic Websites © Aptech Limited

40

### Instructions to the Trainer(s):

- Using Slide 40, explain the operator precedence.
- Operators in JavaScript have certain priority levels based on which their execution sequence is determined.
- For example, the division operator (/) has a higher priority than the subtraction (-) operator.
- Therefore, the division operator will be carried out first, if an expression involves both these operators.
- Further, an execution order is also defined for the operators within expression. This order is referred to as the associativity, which is either from left to right or vice-versa depending upon the operators.
- Explain table which lists the precedence of the operators from the highest to the lowest and their associativity.

## Regular Expressions

Is a pattern that is composed of set of strings, which is to be matched to a particular textual content.

Allow handling of textual data effectively as it allows searching and replacing strings.

Allows handling of complex manipulation and validation, which could otherwise be implemented through lengthy scripts.

- There are two ways to create regular expressions which are as follows:

### Literal Syntax:

- Refers to a static value
- Allows specifying a fixed pattern, which is stored in a variable
- Syntax is as follows:  
var variable\_name = /regular\_expression\_pattern/;

### RegExp() Constructor:

- Is useful when the Web page designer does not know the pattern at the time of scripting.
- Method dynamically constructs a regular expression when the script is executed.
- Syntax is as follows:  
var variable\_name = new RegExp("regular\_expression\_pattern", "flag");

Designing Modernistic Websites © Aptech Limited

41

### Instructions to the Trainer(s):

- Using Slide 41, explain the regular expression.
- A regular expression is a pattern that is composed of set of strings, which is to be matched to a particular textual content. For example, you can specify a pattern for US postal code that the code will not contain more than five digits. When the user enters the postal code, the digits entered by the user will be verified against the pattern to ensure that the postal code is valid.
- Regular expressions allow handling textual data effectively, as it allows searching and replacing strings. They allow handling complex manipulation and validation that could otherwise be implemented through lengthy scripts.
- In JavaScript, there are two ways to create regular expressions which are as follows:
  - Literal Syntax:** A literal refers to a static value. Therefore, a literal syntax allows specifying a fixed pattern, which is stored in a variable. This method of specifying patterns is useful when the Web page designer knows the pattern at the time of scripting.
  - RegExp () Constructor:** The RegExp () constructor is useful when the Web page designer does not know the pattern at the time of scripting. This means that the method dynamically constructs a regular expression when the script is executed. The RegExp () constructor is a function that returns a reference to the built-in RegExp object.

## RegExp Methods and Properties

- RegExp object supports methods that are used for searching the pattern in a string. They are as follows:

`test(string)` - Tests a string for matching a pattern and returns a Boolean value of true or false. The Boolean value indicates whether the pattern exists or not in the string. The method is commonly used for validation.

`exec(string)` - Executes a string to search the matching pattern within it. The method returns a null value, if pattern is not found. In case of multiple matches, it returns the matched result set.

Attribute	Description
<code>\$n</code>	Represents the number from 1 to 9. It stores the recently handled parts of a parenthesized pattern of a regular expression.
<code>aif</code>	Indicates whether the given regular expression contains a g flag. The g flag specifies that all the occurrences of a pattern will be searched globally, instead of just searching for the first occurrence.
<code>aifc</code>	Indicates whether the given regular expression contains an i flag.
<code>aiff</code>	Stores the location of the starting character of the last match found in the string. In case of no match, the value of the property is -1.
<code>asc</code>	Stores the copy of the pattern.

Designing Modernistic Websites © Aptech Limited

42

### Instructions to the Trainer(s):

- Using Slide 42, explain the RegExp methods and properties.
- Mention, RegExp object supports methods that are used for searching the pattern in a string.
- These methods are as follows:
  - `test (string)` – Tests a string for matching a pattern and returns a boolean value of true or false. The boolean value indicates whether the pattern exists in the string. This method is commonly used for validation.
  - `exec (string)` – Executes a string to search the matching pattern within it. The method returns a null value, if pattern is not found. In case of multiple matches, it returns the matched result set.

## Categories of Pattern Matching

- Different categories of pattern matching character that are required to create a regular expression pattern are as follows:



Designing Modernistic Websites © Aptech Limited

43

### Instructions to the Trainer(s):

- Using Slide 43, explain the categories of pattern matching.
- There are different categories of pattern matching character that are required to create a regular expression pattern. Mention the categories displayed on the slide.

## Position Matching

Characters or symbols in this category allow matching a substring that exists at a specific position within a string.

- Following table lists various position matching symbols:

Symbol	Description	Example
<code>^</code>	Denotes the start of a string	<code>/^Good/</code> matches "Good" in "Good night", but not in "A Good Eyesight"
<code>\$</code>	Denotes the end of a string	<code>/art\$/</code> matches "art" in "Cart" but not in "artist"
<code>\b</code>	Matches a word boundary. A word boundary includes the position between a word and the space	<code>/ry\b/</code> matches "ry" in "She is very good"
<code>\B</code>	Matches a non-word boundary	<code>/\Ban/</code> matches "an" in "operand" but not in "anomaly"

Designing Modernistic Websites © Aptech Limited

44

### Instructions to the Trainer(s):

- Using Slide 44, explain the position matching concept.
- Mention, characters or symbols in this category allow matching a substring that exists at a specific position within a string.
- Explain list of various position matching symbols.

## Character Classes

Characters or symbols in this category are combined to form character classes for specifying patterns.

These classes are formed by placing a set of characters within the square brackets.

- Following table lists various character classes symbols:

Symbol	Description	Example
[xyz]	Matches one of the characters specified within the character set	/Good/ matches "Good" in "Good night", but not in "A Good Eyesight"
[!xyz]	Matches one of the characters not specified within the character set	!/BC RT/ Matches "RT" but, not "BRT" or "CRT"
.	Denotes a character except for the new line and line terminator	/s./ Matches "sat", "sit", "set", and so on
\w	Matches alphabets and digits along with the underscore	/\w/ Matches "800" in "800%"
\W	Matches a non-word character	/\W/ Matches "%" in "800%"
\d	Matches a digit between 0 to 9	/\d/ Matches "4" in "A4"
\D	Searches for a non-digit	/\D/ Matches "ID" in "ID 2246"
\s	Searches any single space character including space, tab, form feed, and line feed	/\s/w/ Matches "bar" in "scroll bar"
\S	Searches a non-space character	/\\$w*/ Matches "scroll" in "scroll bar"

Designing Modernistic Websites © Aptech Limited

45

### Instructions to the Trainer(s):

- Using Slide 45, explain the character classes in JavaScript.
- Mention, characters or symbols in this category are combined to form character classes for specifying patterns.
- These classes are formed by placing a set of characters within the square brackets.
- For example, the / [ a-zA-Z0-9 ] / pattern matches all alphabets and digits.
- Explain, table which lists various character classes symbols.

## Repetition

Characters or symbols in this category allow matching characters that reappear frequently in a string.

- Following table lists various repetition matching symbols:

Symbol	Description	Example
{x}	Matches x number of occurrences of a regular expression	/\d{6}/ Matches exactly six digits"
{x,}	Matches either x or additional number of occurrences of a regular expression	/\s{4,}/ Matches minimum four whitespace characters
{x,y}	Matches minimum x to maximum y occurrences of a regular expression	/\d{6,8}/ Matches minimum six to maximum eight digits
?	Matches minimum zero to maximum one occurrences of a regular expression	/\s?m/ Matches "lm" or "l m"
*	Matches minimum zero to multiple occurrences of a regular expression	/im*/ Matches "i" in "Ice" and "imm" in "immaculate", but nothing in "good"

Designing Modernistic Websites © Aptech Limited

46

### Instructions to the Trainer(s):

- Using Slide 46, explain students the concept of repetition.
- Characters or symbols in this category allow matching characters that reappear frequently in a string.
- Explain table which lists various repetition matching symbols.

## Alternation and Grouping

Characters or symbols in this category allow grouping characters as an individual entity or adding the 'OR' logic for pattern matching.

- Following table lists various alternation and grouping character symbols:

Symbol	Description	Example
( )	Organizes characters together in a group to specify a set of characters in a string	/xyz)+(uvw)/ Matches one or more number of occurrences of "xyz" followed by one occurrence of "uvw"
	Combines sets of characters into a single regular expression and then matches any of the character set	/(xy) (uv) (st)/ Matches "xy" or "uv" or "st"

Designing Modernistic Websites © Aptech Limited

47

### Instructions to the Trainer(s):

- Using Slide 47, explain the concept of alteration and grouping.
- Characters or symbols in this category allow grouping characters as an individual entity or adding the 'OR' logic for pattern matching.
- Explain table which lists various alternation and grouping character symbols.

## Back References

Characters or symbols in this category allow grouping characters as an individual entity or adding the 'OR' logic for pattern matching.

- Following table lists various alternation and grouping character symbols:

Symbol	Description	Example
<code>\n</code>	Matches a parenthesized set within the pattern, where n is the number of the parenthesized set to the left	<code>/(\w+)\s+\1/</code> Matches any word occurring twice in a line, such as "hello hello". The \1 specifies that the word following the space should match the string, which already matched the pattern in the parentheses to the left of the pattern. To refer to more than one set of parentheses in the pattern, you would use \2 or \3 to match the appropriate parenthesized clauses to the left. You can have maximum nine back references in the pattern

Designing Modernistic Websites © Aptech Limited

48

### Instructions to the Trainer(s):

- Using Slide 48, explain the back references.
- Characters or symbols in this category allow referring back to a sub-expression in the same regular expression.
- This is useful when matching the remaining sub-expression of a regular expression is based upon the result of matching the previous sub-expression.
- Explain table which lists the back reference matching symbol.

## Decision-making Statements

Statements are referred to as a logical collection of variables, operators, and keywords that perform a specific action to fulfill a required task.

Statements help you build a logical flow of the script.

In JavaScript, a statement ends with a semicolon.

JavaScript is written with multiple statements, wherein the related statements are grouped together and referred to as a block of code and are enclosed in curly braces.

Decision-making statements allow implementing logical decisions for executing different blocks to obtain the desired output.

They execute a block of statements depending upon a Boolean condition that returns either true or false.

### Instructions to the Trainer(s):

- Using Slide 49, explain the decision-making statements in JavaScript.
- Statements are referred to as a logical collection of variables, operators, and keywords that perform a specific action to fulfil a required task.
- For example, the line of code that declares a variable is a statement. Statements help you build a logical flow of the script.
- In JavaScript, a statement ends with a semicolon.
- JavaScript is written with multiple statements, wherein the related statements are grouped together. Such a group of statements is referred to as a block of code and the statements within it are enclosed in curly braces.
- Mention, decision-making statements allow implementing logical decisions for executing different blocks to obtain the desired output. They execute a block of statements depending upon a Boolean condition.
- This condition is an expression that returns either true or false.

The slide compares two types of conditional statements: **if-else-if Statement** and **Nested if Statement**.

**if-else-if Statement:**

- Allows you to check multiple conditions and specify a different block to be executed for each condition.
- Flow of these statements begins with the `if` statement followed by multiple `else if` statements and finally by an optional `else` block.
- Entry point of execution in these statements begins with the `if` statement.
- If the condition in the `if` statement is false, the condition in the immediate `else if` statement is evaluated.
- Also referred to as the `if-else-if` ladder.

**Nested if Statement:**

- Comprises multiple `if` statements within an `if` statement.
- Flow of the nested-`if` statements starts with the `if` statement, which is referred to as the outer `if` statement.
- Outer `if` statement consists of multiple `if` statements, which are referred to as the inner `if` statements.
- Inner `if` statements are executed only if the condition in the outer `if` statement is true.
- Each of the inner `if` statements is executed but, only if the condition in its previous inner `if` statement is true.

Designing Modernistic Websites © Aptech Limited

50

### Instructions to the Trainer(s):

- Using Slide 50, explain the `if-else-if` and `nested if` statement.
- The `if-else-if` statements allow you to check multiple conditions and specify a different block to be executed for each condition. The flow of these statements begins with the `if` statement followed by multiple `else if` statements and finally by an optional `else` block. The entry point of execution in these statements begins with the `if` statement. If the condition in the `if` statement is false, the condition in the immediate `else if` statement is evaluated. The `if-else-if` statements are also referred to as the `if-else-if` ladder.
- The nested-`if` statements comprises multiple `if` statements within an `if` statement. The flow of the nested-`if` statements starts with the `if` statement, which is referred to as the outer `if` statement. This outer `if` statement consists of multiple `if` statements, which are referred to as the inner `if` statements.

## switch-case Statement

A program becomes quite difficult to understand when there are multiple if statements.

To simplify coding and to avoid using multiple if statements, switch-case statement can be used.

switch-case statement allows comparing a variable or expression with multiple values.

### Instructions to the Trainer(s):

- Using Slide 51, explain switch-case statement.
- A program becomes quite difficult to understand when there are multiple if statements.
- To simplify coding and to avoid using multiple if statements, switch-case statement can be used as a different approach to code the same logic.
- The switch-case statement allows comparing a variable or expression with multiple values.

## Introduction to Loops

Loops allow you to execute a single statement or a block of statements multiple times.

They are widely used when you want to display a series of numbers and accept repetitive input.

A loop construct consists of a condition that instructs the compiler the number of times a specific block of code will be executed.

If the condition is not specified within the construct, the loop continues indefinitely. Such loop constructs are referred to as infinite loops.

- JavaScript supports three types of loops that are as follows:
  - `while Loop`
  - `for Loop`
  - `do-while Loop`

### Instructions to the Trainer(s):

- Using Slide 52, introduce students the concept of loops.
- Consider a scenario where you want to accept and display ten numbers to the user.
- Instead of writing the same lines of code again and again for 10 times, you can use loops.
- Loops allow you to execute a single statement or a block of statements multiple times. They can be used when you want to display a series of numbers and accept repetitive input.
- Loop helps you to execute a particular block for a specified number of times.
- Explain the types of loops supported in JavaScript.

## while Loop

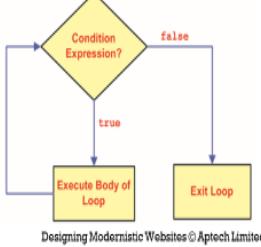
The `while` loop executes a block of code as long as the given condition remains `true`.

The `while` loop begins with the `while` keyword, which is followed by parentheses containing a boolean condition.

If this condition returns `true`, the block of statements within the `while` loop are executed.

Once the condition becomes `false`, the `while` statement stops the execution of loop and transfers the control to next statement appearing after the block.

- Following figure shows the flow of execution of the `while` loop:



Designing Modernistic Websites © Aptech Limited

53

### Instructions to the Trainer(s):

- Using Slide 53, explain the `while` loop in details.
- Mention, `while` loop executes a block of code as long as the given condition remains `true`.
- The `while` loop begins with the `while` keyword, which is followed by parentheses containing a boolean condition. If this condition returns `true`, the block of statements within the `while` loop are executed.
- After every iteration, the program control is transferred back to the `while` statement, where the condition is again checked for another round of execution.
- This process is continued till the specified condition becomes `false`.
- Once the condition becomes `false`, the `while` statement stops the execution of loop and transfers the control to next statement appearing after the block.
- Figure shows the flow of execution - `while` loop.

**for Loop**

- The **for** loop is similar to the **while** loop as it executes the statements within the loop as long as the given condition is true.
- Unlike the **while** loop, the **for** loop specifies the loop control statements at the top instead in the body of the loop.
- The **for** loop begins with the **for** keyword, which is followed by parentheses containing three expressions, each of which are separated by a semicolon.
- The three expressions are referred to as **initialization expression**, **condition expression**, and **increment/decrement expression** respectively.

**do-while Loop**

- The **do-while** loop is similar to the **while** loop. This is because both the **do-while** and **while** loops execute until the condition becomes false.
- However, the **do-while** loop differs by executing the body of the loop at least once before evaluating the condition even if the condition is false.
- The **do-while** loop starts with the **do** keyword and is followed by a block of statements.
- At the end of the block, the **while** keyword is specified that is followed by parentheses containing the condition.
- When the condition returns false, the block of statements after the **do** keyword are ignored and the next statement following the **while** statement is executed.

Designing Modernistic Websites © Aptech Limited

54

### Instructions to the Trainer(s):

- Using Slide 54, explain the **for** loop and **do-while** loop.

#### **for** loop:

- Mention that the **for** loop is similar to the **while** loop in functionality. It executes the statements within the loop as long as the given condition is **true**.
- Unlike the **while** loop, the **for** loop specifies the loop control statements at the top instead in the body of the loop.
- The **for** loop begins with the **for** keyword, which is followed by parentheses containing three expressions, each of which are separated by a semicolon.
- The three expressions are referred to as **initialization expression**, **condition expression**, and **increment/decrement expression** respectively.

#### **do-while** loop:

- Mention **do-while** loop is similar to the **while** loop. This is because both the **do-while** and **while** loops execute until the condition becomes false.
- However, the **do-while** loop differs by executing the body of the loop at least once before evaluating the condition. Thus, even if the condition is **false**, the **do-while** loop executes at least once.
- The **do-while** loop starts with the **do** keyword and is followed by a block of statements.
- At the end of the block, the **while** keyword is specified that is followed parentheses containing the condition.

- When the specified condition returns `false`, the block of statements after the `do` keyword are ignored and the next statement following the `while` statement is executed.

**In-Class Question:**

**Question:** What is difference in do-while loop as compared to while loop?

**Answer:** The `do-while` loop differs by executing the body of the loop at least once before evaluating the condition.

## Jump Statements

### break Statement

The break statement can be used with decision-making such as switch-case and loop constructs such as for and while loops.

The break statement is denoted by using the break keyword. It is used to exit the loop without evaluating the specified condition.

The control is then passed to the next statement immediately after the loop.

### continue Statement

The continue statement is mostly used in the loop constructs and is denoted by the continue keyword.

It is used to terminate the current execution of the loop and continue with the next repetition by returning the control to the beginning of the loop.

This means, the continue statement will not terminate the loop entirely, but terminates the current execution.

Designing Modernistic Websites © Aptech Limited

55

### Instructions to the Trainer(s):

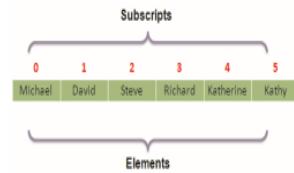
- Using Slide 55, explain the jump statements.
- Jump statements comprises the two types:
  - break statement
  - continue statement
- **break statement:**
  - break statement can be used with decision-making statements, such as switch-case and loop constructs, such as for and while loops.
  - It is denoted by using the break keyword.
  - It is used to exit the loop without evaluating the specified condition.
  - The control is then passed to the next statement immediately after the loop.
- **continue statement:**
  - continue statement is mostly used in the loop constructs.
  - The continue statement is denoted by the continue keyword.
  - It is used to terminate the current execution of the loop and continue with the next repetition by returning the control to the beginning of the loop.
  - This means, the continue statement will not terminate the loop entirely, but terminates the current execution.

## Arrays

- An array is a collection of values stored in adjacent memory locations.
- These array values are referenced using a common array name and must be of the same data type. These values can be accessed by using subscript or index numbers.
- JavaScript supports two types of arrays:
  - Single-dimensional array
  - Multi-dimensional array

### Single-dimensional Array

In a single-dimensional array, the elements are stored in a single row in the allocated memory.



### Multi-dimensional Array

- A multi-dimensional array stores a combination of values of a single type in two or more dimensions.
- These dimensions are represented as rows and columns similar to those of a Microsoft Excel sheet.

Employee Salaries	0	1	2	3
	BASIC	HRA	ALLOWANCE	TOTAL
0	14350	10500	1500	26350
1	34350	4050	1000	39400
2	6150	4500	1250	13900
3	4920	4500	2250	11670
4	12300	9000	2000	23300

Designing Modernistic Websites © Aptech Limited

56

### Instructions to the Trainer(s):

- Using Slide 56, explain arrays.
- An array is a collection of values stored in adjacent memory locations. These array values are referenced using a common array name. The values of an array variable must be of the same data type.
- These values that are also referred to as elements and can be accessed by using subscript or index numbers. The subscript number determines the position of an element in an array list.
- Arrays are of two types: single-dimensional and multi-dimensional arrays.
- In single-dimensional array, the elements are stored in a single row in the allocated memory. Explain the figure that shows the allocation of single-dimensional array.
- A multi-dimensional array stores a combination of values of a single type in two or more dimensions. These dimensions are represented as rows and columns similar to those of a Microsoft Excel sheet. A two-dimensional array is an example of the multi-dimensional array.

## Accessing Single-dimensional Arrays

➤ Accessing Array Elements Without Loops

➤ Accessing Array Elements With Loops

### Array Methods

An array is a set of values grouped together and identified by a single name. In JavaScript, the `Array` object allows you to create arrays.

It provides the `length` property that allows you to determine the number of elements in an array.

various methods of the `Array` object allow to access and manipulate the array elements.

Designing Modernistic Websites © Aptech Limited

57

### Instructions to the Trainer(s):

- Using Slide 57, explain how to access single-dimensional arrays.
- Tell the students that arrays can be accessed as follows:
  - Accessing Array Elements Without Loops
  - Accessing Array Elements With Loops
- **Array methods:**
  - In JavaScript, the `Array` object allows you to create arrays.
  - It provides the `length` property that allows you to determine the number of elements in an array.
  - Various methods of the `Array` object allow to access and manipulate the array elements.

## for..in Loop

The `for..in` loop is an extension of the `for` loop. It enables to perform specific actions on the arrays of objects.

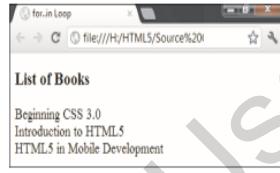
The loop reads every element in the specified array and executes a block of code only once for each element in the array.

### Syntax:

```
for (variable_name in array_name)
{
    //statements;
}
```

where,

- `variable_name`: Is the name of the variable.
- `array_name`: Is the array name.



Designing Modernistic Websites © Aptech Limited

58

### Instructions to the Trainer(s):

- Using Slide 58, explain the `for..in` loop.
- Mention, `for..in` loop is an extension of the `for` loop.
- It enables to perform specific actions on the arrays of objects.
- The loop reads every element in the specified array and executes a block of code only once for each element in the array.

## Slides 59 and 60

### Summary 1-2

- ❖ Scripting refers to a series of commands that are interpreted and executed sequentially and immediately on an occurrence of an event.
- ❖ JavaScript is a scripting language, which can be executed on the client-side and on the server-side.
- ❖ A variable refers to a symbolic name that holds a value, which keeps changing.
- ❖ A primitive data type contains a single literal value such as a number or a string.
- ❖ A function is a piece of code that performs some operations on variables to fulfill a specific task.
- ❖ Event handling is a process of specifying actions to be performed when an event occurs.
- ❖ Event bubbling is a mechanism that allows you to specify a common event handler for all child elements.
- ❖ jQuery mobile is a Web User Interface development framework that allows the user to build mobile Web applications that works on tablets and smartphones.
- ❖ An operator specifies the type of operation to be performed on the values of variables and expressions.
- ❖ JavaScript operators are classified into six categories based on the type of action they perform on operands.

59

### Summary 2-2

- ❖ There are six category of operators namely, Arithmetic, Relational, Logical, Assignment, Bitwise, and Special operators.
- ❖ Operators in JavaScript have certain priority levels based on which their execution sequence is determined.
- ❖ A regular expression is a pattern that is composed of set of strings, which is to be matched to a particular textual content.
- ❖ In JavaScript, there are two ways to create regular expressions namely, literal syntax and RegExp() constructor.
- ❖ Decision-making statements allow implementing logical decisions for executing different blocks to obtain the desired output.
- ❖ A loop construct consists of a condition that instructs the compiler the number of times a specific block of code will be executed.
- ❖ JavaScript supports three types of loops that include: while loop, for loop, and do-while loop.
- ❖ The break statement is used to exit the loop without evaluating the specified condition.
- ❖ The continue statement terminates the current execution of the loop and continue with the next repetition by returning the control to the beginning of the loop.
- ❖ JavaScript supports two types of arrays namely, Single-dimensional array and Multi-dimensional array.
- ❖ The for..in loop is an extension of the for loop that enables to perform specific actions on the arrays of objects.

60

#### Instructions to the Trainer(s):

- Show students Slides 59 and 60.
- Summarize the session by reading out each point on the Slide.

## Session 12: JavaScript - II

### 12.1 Pre-Class Activities

Before you commence the session, you should familiarize yourself with the topics of this session in-depth. Prepare a question or two that will be a key point to relate the current session objectives.

#### 12.1.1 Teaching Skills

To teach this session, you should be well versed with functions, objects, and methods in JavaScript.

You should teach the concepts in the theory class using the images provided. For teaching in the class, you are expected to use slides and LCD projectors.

#### In-Class Activities

Follow the order given here during In-Class activities.

Slide 2

The slide features a red-to-orange gradient background. On the right side, there is a large, semi-transparent orange circle. At the top left, the word "Objectives" is written in white. Below it is a bulleted list of ten items:

- Explain functions
- Explain parameterized functions
- Explain return statement
- Describe objects
- Explain different browser objects
- Describe DOM and its objects
- Identify the use of Promise.any
- Explain Private class methods
- Explain JSON

At the bottom left, the text "Designing Modernistic Websites © Aptech Limited" is visible. At the bottom right, there is a small number "2".

#### Instructions to the Trainer(s):

Give students a brief overview of the current session through the session objectives listed in Slide 2.

## 12.2 In-Class Explanations

Slides 3 to 5

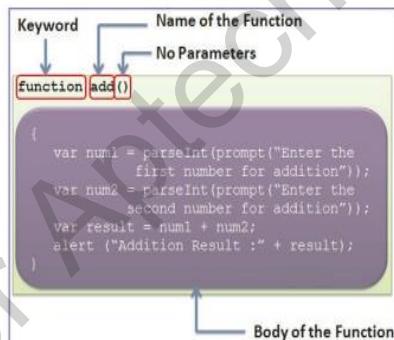
### Functions 1-3

- A function is an independent reusable block of code that performs certain operations.
- It is always created under `script` element.
- A function is declared using `function` keyword.
- The keyword is followed by the name of the function and parameters enclosed within the parenthesis.
- A function needs to be invoked.
  - To invoke a function, specify the function name followed by parenthesis outside the function block.

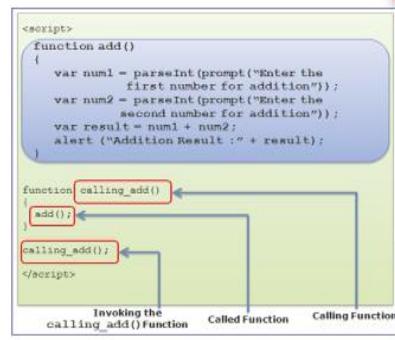
Designing Modernistic Websites © Aptech Limited

3

### Functions 2-3



Declaration and Definition of a Function

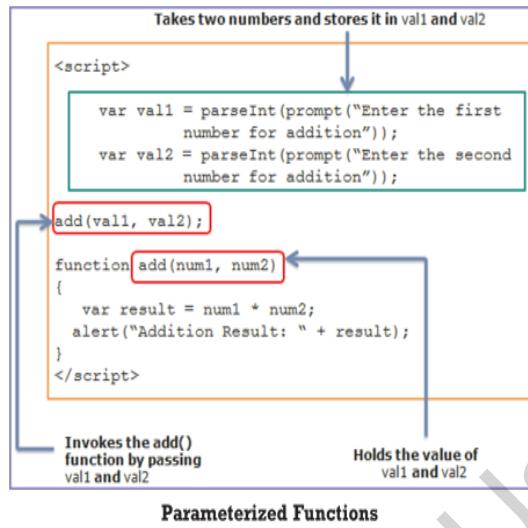


Invoking of Function

Designing Modernistic Websites © Aptech Limited

4

## Functions 3-3



Designing Modernistic Websites © Aptech Limited

5

### Instructions to the Trainer(s):

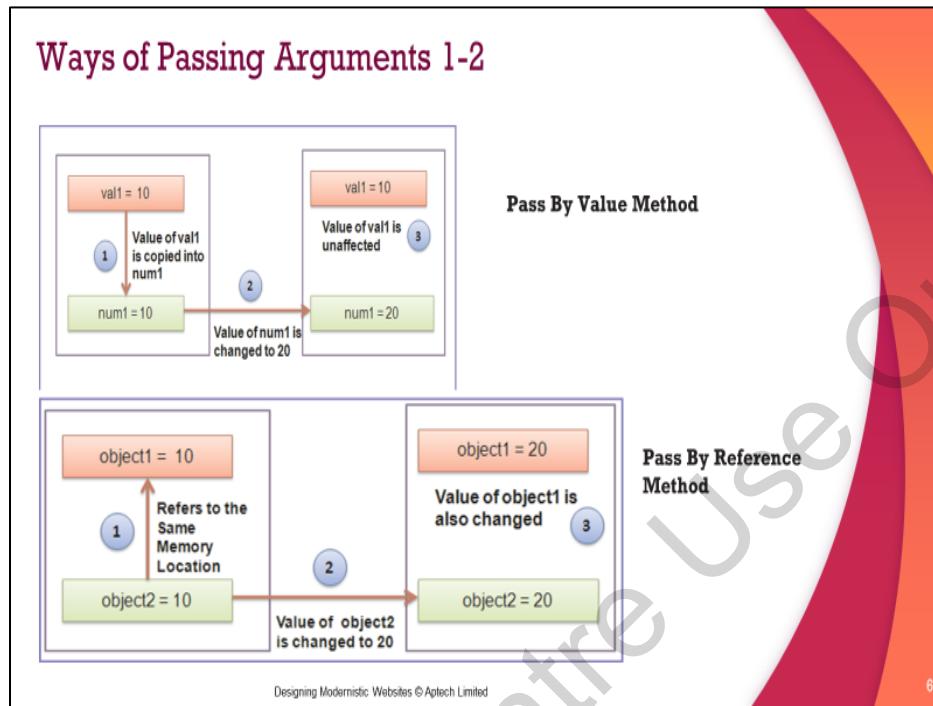
- Using Slides 3 to 5, explain the functions in JavaScript.
- A function is an independent reusable block of code that performs certain operations on variables and expressions to fulfill a task.
- A JavaScript function is always created under the script element. JavaScript supports both user-defined and built-in functions.
- JavaScript allows declaring a function using the function keyword. The keyword is followed by the name of the function and parameters enclosed within the parenthesis () .
- A function must be invoked or called to execute it in the browser. To invoke a function, specify the function name followed by parenthesis outside the function block.
- Parameterized functions refer to JavaScript functions that take parameters. These parameters hold values on which the function must perform operations. Parameterized functions can be created to accept values for performing operations.

### In-Class Question:

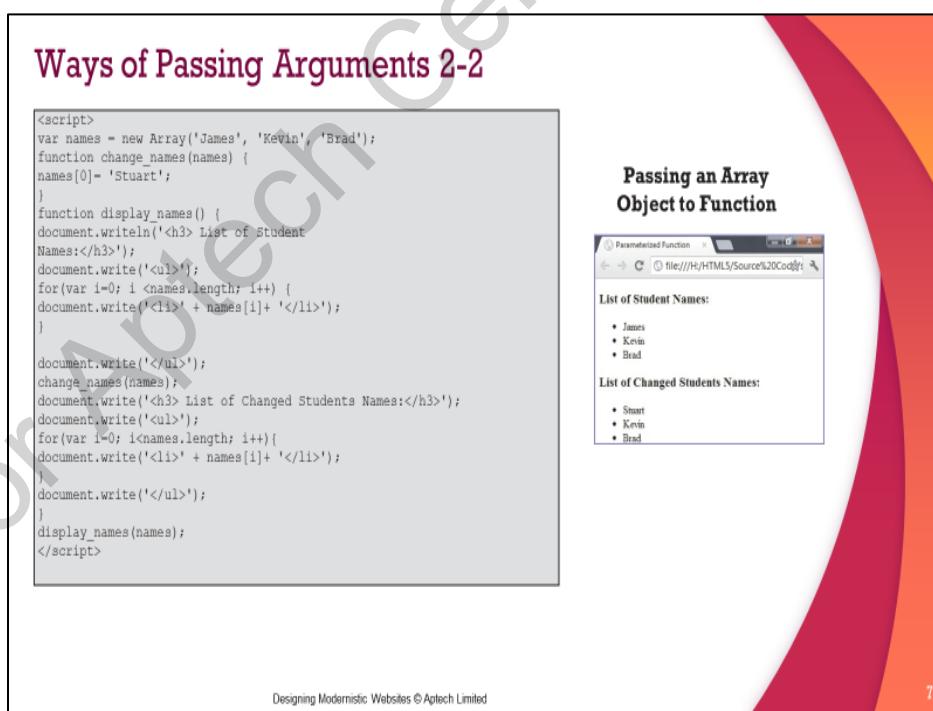
**Question:** How to define a function?

**Answer:** The function by specifying the operations or instructions within the curly braces { and } .

## Slides 6 and 7



6



7

**Instructions to the Trainer(s):**

- Using Slides 6 and 7, explain the ways of passing arguments.
- There are two ways of passing arguments to a function namely, pass by value and pass by reference.
- The description about these is as follows:
  - **Passing by value** - Refers to passing variables as arguments to a function. In the pass by value method, the called function do not change the values of the parameters passed to it from the calling function. This is because each parameter occupies different memory locations. This is because each parameter occupies different memory locations.
  - **Passing by reference** - Refers to passing objects as arguments to a function. In the pass by reference method, the called function modifies the value of parameters passed to it from the calling function. This change is reflected when the control passes back to the calling function.

## return Statement

JavaScript sends result to the calling function by using `return` statement.

```
<script>
function factorial(num){
    if(num==0)
        return 0;
    else if(num==1)
        return 1;
    else {
        var result=num;
        while(num>1)
        {
            result= result * (num-1);
            num--;
        }
        return result;
    }
}
var num=prompt('Enter number:','');
var result=factorial(num);
alert('Factorial of '+num+' is '+result+'.');
</script>
```

Designing Modernistic Websites © Aptech Limited



Factorial of Number

Factorial of 6 is 720.

OK

8

**Instructions to the Trainer(s):**

- Using Slide 8, explain the `return` statement.
- JavaScript allows sending the result back to the calling function by using the `return` statement.
- The `return` statement begins with `return` keyword followed by the variable or value, which must be returned to the calling function. The `return` statement can also be used to halt the execution of the function and to return the control to the calling function. This is required when a particular condition is false or when there are chances of unexpected results during the code execution.
- Explain code snippet which demonstrates the script that calculates the factorial of a number using a function and display the output to the user.
- The code defines a function named `factorial()` which takes the `num` variable as the parameter. The execution of the script starts from the `prompt()` function, which takes the number from the user and stores it in the `num` variable. Next, the `factorial()` function is invoked by passing the `num` argument. If the user enters the value as 0 or 1, the function returns the value as 0 or 1 respectively. For any other number, the function calculates the factorial and returns the output value by using the `return` statement. The output is stored in the `result` variable, which is displayed to the user.

## Objects

- Objects are entities with properties and methods.
  - Properties specify the characteristics or attributes of an object.
  - Methods identify the behavior of an object.
- Objects can be built-in or custom.

	<b>Object: Car</b>
	<b>Properties</b> Make - ford Color - green Wheels - four
	<b>Methods</b> run() stop()
	<b>Object: Bird</b>
	<b>Properties</b> Type - pigeon Color - gray Wings - two
	<b>Methods</b> eat() fly()

Designing Modernistic Websites © Aptech Limited

9

### Instructions to the Trainer(s):

- Using Slide 9, explain objects.
- Objects are entities with properties and methods and resemble to real life objects. Properties specify the characteristics or attributes of an object, while methods identify the behavior of an object. For example, consider a real life object namely, Car.
- The attributes of the Car object can include color, car number, and model. The methods of the car could be `run()` that specifies the running behavior of the car.
- Similarly, in JavaScript, objects have their own properties and methods. Figure shows objects with their properties and methods.

## Creating Custom Objects

- The `object` object is the parent object.
  - All JavaScript objects are derived from this object.
- An object can be created using the built-in `Object` object or by defining a template.

Syntax using the built-in `Object` object:

```
var object_name = new Object();
```

Syntax using the template:

```
function object_type(list of parameters)
{
    // Body specifying properties and methods
}
```

Example:

```
<script>
//create an object using direct method
var doctor_details=new Object();
//create an object using new keyword
studOne = new student_info ('James', '23', 'New Jersey');
</script>
```

Designing Modernistic Websites © Aptech Limited

10

### Instructions to the Trainer(s):

- Using Slide 10, explain creating custom objects.
- The `Object` object is the parent object from which all JavaScript objects are derived. The custom objects can be derived from this object by using the `new` keyword.
- There are two main methods to create a custom object namely, direct method and template method.
- In the code, `doctor_details` object is created using the `Object` object. After creating the object, properties and methods can be specified. Similarly, `student_info` object is created using `new` keyword. The values 'James', '23', and 'New Jersey' are the properties of the `student_info` which are initialized by constructor function during creation.

Slides 11 and 12

## Creating Properties for Custom Objects 1-2

```
<script>
var student_details=new Object();
student_details.first_name= 'John';
student_details.last_name= 'Fernando';
student_details.age= '15';
alert ('Student\'s name: '+student_details.first_name+ '' + student_details.last_name);
</script>
```



student\_details Object

Designing Modernistic Websites © Aptech Limited

11

## Creating Properties for Custom Objects 2-2

```
<script>
// To define the object type
function employee_info(name, age, experience)
{
    this.name=name;
    this.age=age;
    this.experience=experience;
}
// Creates an object using new keyword
empMary=new employee_info('Mary', '34', '5years');
alert ("Name:"+empMary.name+ "\n" +"Age: "+empMary.age+ "\n" +"Experience: "+empMary.experience);
</script>
```



employee\_info Object

Designing Modernistic Websites © Aptech Limited

12

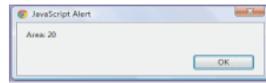
### Instructions to the Trainer(s):

- Using Slides 11 and 12, explain the process of creating properties of custom objects.

- Mention that, properties specify the characteristics of an object. They can be specified for objects created through Object or template method.
- To create and access a property of an object created using Object object, specify the object name followed by a period and the property name.
- Explain code snippet which demonstrates the script that creates the student\_details object and adds properties namely, first\_name, last\_name, and age along with their values.
- The code specifies three properties of the student\_details object namely, first\_name, last\_name, and age along with their values. The values of these properties are displayed in the browser using the write() method.

## Creating Methods for Custom Objects

```
<script>
    var square =new Object();
    square.length=parseInt("5");
    square.cal_area=function()
    {
        var area=(parseInt(square.length)*parseInt("4"));
        return area;
    }
    alert ("Area: "+square.cal_area());
</script>
```



Output of the Area of Square

Designing Modernistic Websites © Aptech Limited

13

### Instructions to the Trainer(s):

- Using Slide 13, explain how to create methods for custom objects.
- Methods are similar to JavaScript functions, but there is a slight difference between them. A method is always associated with an object and is executed by referring to that object. On the other hand, a function is not associated with any object and is executed independently. Similar to functions, the custom methods can also take parameters.
- One or more methods can be specified after creating an object using the `Object` object or while creating the template. To invoke a method, they must be specified with the object name followed by a period, method name, and parenthesis with parameters, if any.
- Explain code snippet which demonstrates the code that defines a custom method to calculate the area of a square.

## Built-in Objects

- The built-in objects are static objects.
- They help extend the functionality in the script.
- Some of these objects are: String, Math, and Date.

Designing Modernistic Websites © Aptech Limited

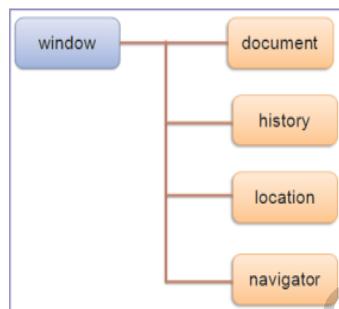
14

### Instructions to the Trainer(s):

- Using Slide 14, explain the built-in objects in JavaScript.
- The built-in objects are static objects which can be used to extend the functionality in the script.
- Some of these objects are: String, Math, and Date.
- The browser objects, such as window, history, and navigator are used to work with the browser window, whereas the HTML objects, such as form, anchor, and so on are used to access elements on a Web page.

## Browser Objects

- Browser objects help manipulate various aspects of a Web browser.
- They exist on all pages displayed in the browser.



Designing Modernistic Websites © Aptech Limited

18

### Instructions to the Trainer(s):

- Using Slide 15, explain browser objects.
- Give examples such as accessing browser history and changing current URL.
- Explain the hierarchy of browser objects. Mention that:
  - The window object is the top-level object in JavaScript hierarchy and represents a browser window.
  - All the objects in the hierarchy are descendants of the window object.
  - The history object contains a set of URLs visited by a user in a browser window.
  - The location object allows complete access of information of the URL loaded in the browser window.
- For more information about browser objects, refer to the following:  
<https://codescracker.com/js/js-browser-objects.htm>

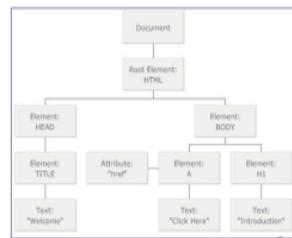
### In-Class Question:

**Question:** Which object includes browser information, such as the look and feel of the browser?

**Answer:** Window object.

## Document Object Model (DOM)

- DOM is a cross-platform and language-independent interface.
- It considers an XML or HTML document as a tree structure.
  - Each node is an object representing a part of the document.
- DOM represents a document with a logical tree.
  - Each branch of the tree ends in a node.
  - Each node contains objects.

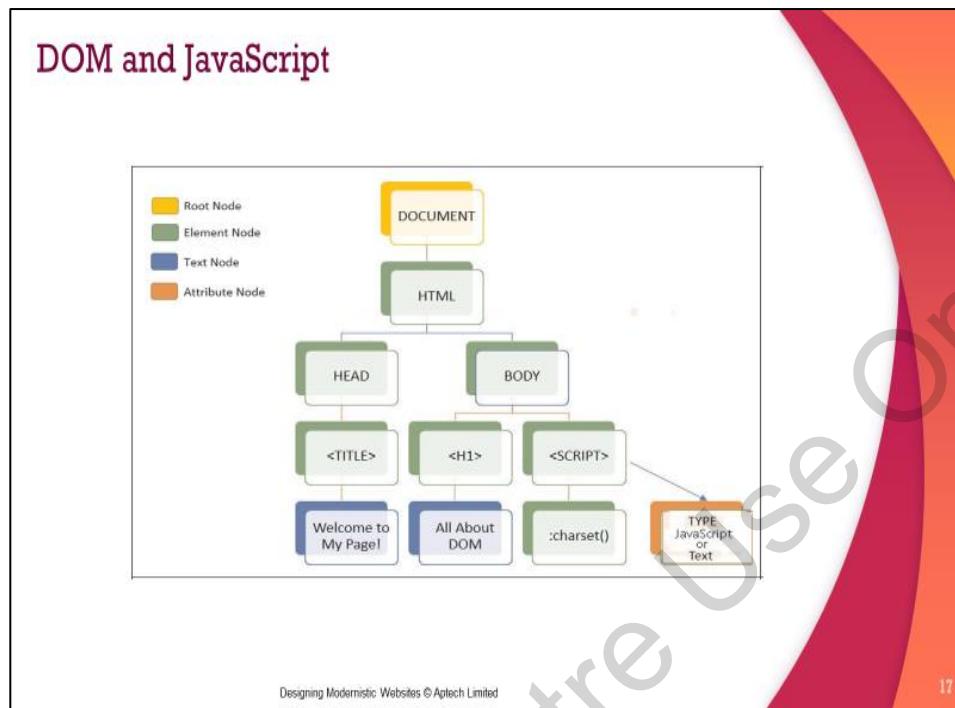


Designing Modernistic Websites © Aptech Limited

16

### Instructions to the Trainer(s):

- Using Slide 16, explain DOM.
- Mention that with DOM methods, one can change the structure, style or content of a document.
- With the HTML DOM, JavaScript can access and change all the elements of an HTML document.
- When a Web page is loaded, the browser creates a DOM of the page.
- The HTML DOM is a programming interface for HTML and defines:
  - HTML elements as objects
  - properties of all HTML elements
  - methods to access all HTML elements
  - events for all HTML elements
- For more information about DOM, refer to the following: [https://developer.mozilla.org/en-US/docs/Web/API/Document\\_Object\\_Model/Introduction](https://developer.mozilla.org/en-US/docs/Web/API/Document_Object_Model/Introduction)



**Instructions to the Trainer(s):**

- Using Slide 17, mention that DOM is an essential component for JavaScript.
- It helps in identifying the HTML documents, XML documents, Web pages, and associated components, such as header of the document and tables.

**In-Class Question:**

**Question:** Are JavaScript and DOM separate entities?

**Answer:** Yes

## New Features in JavaScript DOM

- Arrow functions help create functions in a simple manner.
- Arrow functions are useful to work with functions that require another function as an argument.

```
document.addEventListener("DOMContentLoaded" ,  
    ()=>{ console.log("loaded");  
})
```

Designing Modernistic Websites © Aptech Limited

18

### Instructions to the Trainer(s):

- Using Slide 18, explain arrow functions as the new feature in JavaScript DOM.
- Tell the students that they must be defined before they are used.
- For more information about arrow functions, refer to the following:  
<https://www.javascripttutorial.net/es6/javascript-arrow-function/>

## New Features in JavaScript DOM

- The `for...of` loop statement creates a loop that repeats over iterable objects, such as arrays, maps, strings, and more.

```
const webFrameworks = ["React", "Angular", "Rails",  
"Node.js"];  
let text = "";  
for (let x of webFrameworks) {  
    text += x;  
}  
console.log(text);
```

Designing Modernistic Websites © Aptech Limited

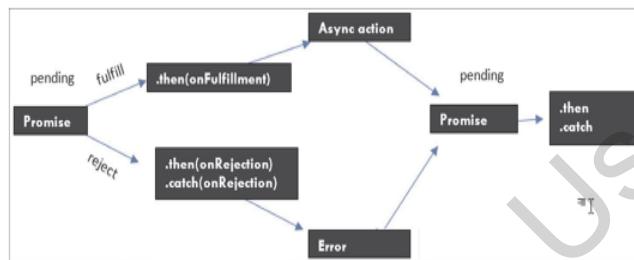
19

### Instructions to the Trainer(s):

- Using Slide 19, explain the new features in JavaScript DOM.
- Tell the students that the iterable objects can be items such as arrays, maps, and strings.
- Explain an example of looping over an array using the code snippet given on the slide.
  - In the code, a constant array is declared to hold the names of different Web Frameworks.
  - A `for...of` loop is created to iterate through each element in the array.
  - The elements form a concatenated string, `text`.
- Discuss the output of the code snippet in the class.
- For more information about `for...of`, refer to the following: [https://exploringjs.com/es6/ch\\_for-of.html](https://exploringjs.com/es6/ch_for-of.html)

## JavaScript Promises

- Promises are a new feature in JavaScript.
- Promise represents eventual success or failure of an asynchronous operation.
- Promises can handle multiple asynchronous operations and provide better error handling.



Designing Modernistic Websites © Aptech Limited

20

### Instructions to the Trainer(s):

- Using Slide 20, explain JavaScript promises.
- Tell the students that a promise has four states: fulfilled, rejected, pending, and settled.
- Explain promise chain. Mention that:
  - When a callback function returns a Promise, it searches for a method.
  - Based on the method, the Promise chains on another call.
  - All successive calling methods are called the promise chain.
- For more information about JavaScript promises, refer to the following:  
<https://www.digitalocean.com/community/tutorials/understanding-javascript-promises>

### In-Class Question:

**Question:** What are the four states of a promise?

**Answer:** pending, fulfilled, rejected, and settled.

## Private Class Features

- A private method means only those objects belonging to the same class can access it.
- To declare a private class field, prefix the name of the class field with # (hash) tag.
- Private fields can be accessed on the class constructor from within the class declaration.

```
// Create new class class  
MyClass {  
    // Declare private class field  
    #myPrivateField = 'This is a personal account.'  
}
```

Designing Modernistic Websites © Aptech Limited

21

### Instructions to the Trainer(s):

- Using Slide 21, elaborate on the private method.
- Tell the students that by default, class fields are public.
- One can access a class property either by creating a new instance of the class and accessing the property on that instance or by declaring the property as a static property so that the class does not have to be instantiated.
- To access private class fields from outside the class, create a new method and return the private class field:
  - Define the method as public or static.
  - For public method, instantiate the class. Next, call the method on the new instance and get the value of the private field.
  - Static methods can be called without instantiating the class.
- For more information about private method, refer to the following:  
[https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Working\\_With\\_Private\\_Class\\_Features](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Working_With_Private_Class_Features)

Slides 22 and 23

## JavaScript Object Notation (JSON) 1-2

```
[  
  "page":1,  
  "results": [  
    {  
      "first_air_date": "2005-03-26",  
      "genre_ids": [ 28,  
                    12,  
                    18,  
                    878  
                  ],  
      "id": 57243,  
      "original_name": "Doctor Who",  
      "origin_country": [  
        "GB"  
      ],  
      "name": "Doctor Who"  
    },  
    {  
      "first_air_date": "2007-09-24",  
      "genre_ids": [ 18,  
                    35  
                  ],  
      "id": 1418,  
      "original_name": "The Big Bang Theory",  
      "origin_country": [  
        "US"  
      ],  
      "name": "The Big Bang Theory"  
    }  
  ]
```

Designing Modernistic Websites © Aptech Limited

22

## JavaScript Object Notation (JSON) 2-2

```
[  
  {  
    "first_air_date": "2015-08-23",  
    "genre_ids": [ 18,  
                  27  
                ],  
    "id": 62286,  
    "original_name": "Fear the Walking Dead",  
    "origin_country": [  
      "US"  
    ],  
    "name": "Fear the Walking Dead"  
  },  
  {  
    "total_pages": 3116,  
    "total_results": 62309  
  }  
]
```

Designing Modernistic Websites © Aptech Limited

23

**Instructions to the Trainer(s):**

- Using Slides 22 and 23, explain JSON.
- Tell the students that JSON structures data in a readable form and transmits data between a Web application and a server.
- JSON includes keys and values:
  - Key is a string enclosed in quotation marks.
  - Value can be an array, object, string, number, or Boolean values.
- For more information about JSON, refer to the following: <https://www.json.org/json-en.html>

**In-Class Question:**

**Question:** How is a key enclosed in JSON data?

**Answer:** It is enclosed in quotation marks.

**JSON Serialization and Deserialization**

Serialization - an object is converted into a string so that it can be recreated

Deserialization - a string is converted into an object

Designing Modernistic Websites © Aptech Limited

24

**Instructions to the Trainer(s):**

- Using Slide 24, explain JSON Serialization and Deserialization.
- Tell the students that serialization and deserialization are essential to save the state of an object so that it can be created again. This way data can be stored and exchanged.
- Small data is stored in byte form. Large and complex data cannot be stored in bytes.
- Serialization converts large data into bytes. When the receiver gets the original data through deserialization.
- Discuss examples of serialization and deserialization.
- For more information about serialization and deserialization, refer to the following:  
<https://betterprogramming.pub/serialization-and-deserialization-ba12fc3fbe23>

**In-Class Question:**

**Question:** What is the process of creating object from sequence of bytes called?

**Answer:** Deserialization.

## Summary

- ❖ A function is reusable piece of code which performs calculations on parameters and other variables.
- ❖ The return statement passes the resultant output to the calling function after the execution of the called function.
- ❖ Objects are entities with properties and methods and resemble to real life objects.
- ❖ There are two ways to create a custom object namely, by directly instantiating the object or by creating a constructor function.
- ❖ JavaScript provides various built-in objects, such as String, Math, and Date.
- ❖ JavaScript also provides browser objects, such as window, history, location, and navigator.
- ❖ DOM is a standard technique for dynamically accessing and manipulating HTML elements.
- ❖ The DOM provides a document object which is used within JavaScript to access all HTML elements presented on the page.

25

### Instructions to the Trainer(s):

- Show students Slide 25.
- Summarize the session by reading out each point on the Slide.

## Session 13: Canvas and Web Storage in HTML5

### 13.1 Pre-Class Activities

Before you commence the session, you should familiarize yourself with the topics of this session in-depth. Prepare a question or two that will be a key point to relate the current session objectives.

#### 13.1.1 Teaching Skills

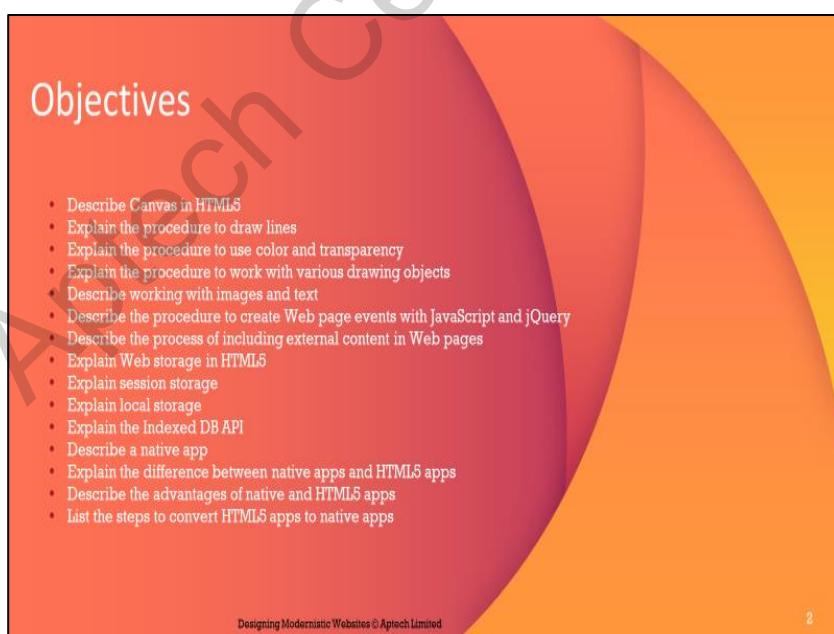
To teach this session, you should be well versed with <canvas> element in HTML5. You should also aware yourself with drawing objects, images, text, and create Web page events with JavaScript and jQuery. Also, you should understand the Web storage in HTML5, native apps, and HTML5 apps

You should teach the concepts in the theory class using the images provided. For teaching in the class, you are expected to use slides and LCD projectors.

### In-Class Activities

Follow the order given here during In-Class activities.

Slide 2



### Instructions to the Trainer(s):

Give students a brief overview of the current session through the session objectives listed in Slide 2.

## 13.2 In-Class Explanations

Slides 3 to 5

### Canvas Element 1-3

The `<canvas>` element in HTML5 can be used to draw shapes on Websites as well as to dynamically draw graphics using JavaScript.

The `<canvas>` element is represented like a rectangle on a page and allows the user to draw arcs, text, shapes, gradients, and patterns.

The `<canvas>` in HTML5 is like `<div>`, `<table>`, or `<a>` tag except that the content used in it is rendered through JavaScript.

The `<canvas>` element does not contain any drawing abilities, instead, drawing is done using a JavaScript code.

To make use of `<canvas>` element, a user has to add `<canvas>` tag on the HTML page.

Using `<canvas>` with JavaScript improves overall performance of Websites and avoids requirement to download images from sites.

Designing Modernistic Websites © Apitoch Limited

3

### Canvas Element 2-3

- The Code Snippet demonstrates the use of `<canvas>` element.

```
<!DOCTYPE HTML>
<html>
  <head>
    <title> Canvas </title>
    <style>
      canvas{border: medium double red; margin: 4px}
    </style>
  </head>
  <body>
    <canvas width="278" height="200"></canvas>
  </body>
</html>
```

- In the code, `<style>` element is used to display border of `<canvas>` element.
- The height and width attributes specify size of `<canvas>` element on the page.

To draw a `<canvas>` element, the user can use a context object.

The context object contains the drawing functions for a specific style of graphics.

Two-Dimensional (2d) context is used to work with 2d operations.

Designing Modernistic Websites © Apitoch Limited

4

## Canvas Element 3-3

The <canvas> element in DOM exposes the `HTMLCanvasElement` interface.

This interface provides the methods and properties for changing the presentation and layout of canvas elements.

The `HTMLCanvasElement` has a `getContext(context)` method that returns the drawing context for the canvas.

- The Code Snippet demonstrates the 2d context object for the canvas.

```
<!DOCTYPE HTML>
<html>
  <head>
    <title> Canvas </title>
    <script>
      window.onload = function()
      {
        var canvas = document.getElementById('mCanvas');
        var ctext = canvas.getContext('2d');
        ctext.beginPath();
        ctext.rect(18, 50, 200, 100);
        ctext.fillStyle = "DarkBlue";
        ctext.fill();
      };
    </script>
  </head>
  <body>
    <canvas id="mCanvas" width="578" height="200"></canvas>
  </body>
</html>
```

Designing Modernistic Websites © Aptech Limited

5

### Instructions to the Trainer(s):

- Using Slides 3 to 5, explain the <canvas> element.
- Canvas is one of the most interesting features added in HTML5.
- The <canvas> element is used to draw graphics on the Web pages. It can also be used to dynamically draw graphics using JavaScript.
- This improves the overall performance of Websites and avoids the requirement to download images from the sites.
- The <canvas> element is represented like a rectangle on a page and allows the user to draw arcs, texts, shapes, gradients, and patterns.
- The <canvas> element is simple and easy to use with JavaScript.
- The <canvas> element does not contain any drawing abilities; instead, the drawing is done using a JavaScript code.
- Mention that the DOM exposes the `HTMLCanvasElement` interface to work with the canvas element.
- This interface provides the methods and properties for changing the presentation and layout of canvas elements.
- The `HTMLCanvasElement` has a `getContext(context)` method that returns the drawing context for the canvas.

## Drawing a Line in Canvas 1-3

- You can create lines in a canvas using the `stroke()`, `beginPath()`, `lineTo()`, and `moveTo()` methods.
- Following is the syntax to create a line in canvas:

### Syntax:

```
ctext.beginPath();
ctext.moveTo(x,y);
ctext.lineTo(x,y);
ctext.stroke();
```

where,

- `ctext` - specifies a context object
- `beginPath()` - Specifies a new drawing path
- `moveTo()` - Specifies the creation of new sub path to the given position
- `lineTo()` - Specifies the drawing of a line from the context position to the given position
- `stroke()` - Specifies how to assign a color to the line and display it

Designing Modernistic Websites © Aptech Limited

6

## Drawing a Line in Canvas 2-3

- The Code Snippet demonstrates creating a line in HTML5 canvas.

```
<!DOCTYPE HTML>
<html>
<head>
<title>Line</title>
<style>
body
{
    margin: 0px;
    padding: 0px;
}
#Canvas
{
    border: 1px solid red;
}
</style>
<script>
window.onload = function() {
    var canvas = document.getElementById("mCanvas");
    var ctext = canvas.getContext("2d");
    ctext.beginPath();
    ctext.moveTo(100, 150);
    ctext.lineTo(250, 50);
    ctext.lineWidth = 5;
    ctext.strokeStyle = "blue";
    ctext.stroke();
}
</script>
</head>
<body>
<canvas id="mCanvas" width="360" height="200"></canvas>
</body>
</html>
```

Designing Modernistic Websites © Aptech Limited

7

## Drawing a Line in Canvas 3-3

- In the code, the `height` and `width` attributes are defined.
- The initializer function has the DOM object which is accessed through the `id` attribute and gets a 2d context by using the `getContext()` method.
- The `beginPath()` method is called through the context object to draw the path of the line.
- The `moveTo(100, 150)` method is called that creates a new path for the given point to place the drawing cursor and moves the position of the window to the upper-left corner by giving the `x` and `y` coordinates.
- The `lineTo(250, 50)` method is called to draw the line from the context point to given point.
- The `lineWidth` property is specified as 5 to define the width of the line on the canvas.
- The `strokeStyle` property sets the color of the line to blue.
- The `stroke()` method assigns the color to the line.

Designing Modernistic Websites © Aptech Limited

8

### Instructions to the Trainer(s):

- Using Slides 6 to 8, explain how to draw line in canvas.
- The functions used to draw a line are `stroke()`, `beginPath()`, `lineTo()`, and `moveTo()` methods.
- The `stroke()` method actually draws the path you have defined with all those `moveTo()` and `lineTo()` methods. The default color is black.
- The `lineTo()` method adds a new point and creates a line from that point to the last specified point in the canvas (this method does not draw the line).
- The `moveTo()` method moves the path to the specified point in the canvas, without creating a line.

### In-Class Question:

**Question:** Which methods are used to define the starting point and ending point of the line?

**Answer:** `moveTo(x, y)` and `lineTo(x, y)`

## Working with Drawing Objects 1-11

- HTML5 canvas allows the user to work with different types of drawing objects.
- Following objects can be drawn on a canvas element:

### ➤ Rectangle

- With HTML5 canvas, the user can create a rectangle using the `rect()` method.
- The HTML5 canvas is placed by using the `x` and `y` parameters and appropriately sized through `height` and `width` properties.
- Following table lists the common properties and methods of various shapes:

Properties and Methods	Description
<code>fillStyle</code>	The values for this property can be gradient, pattern, or a CSS color. The default property style is solid black, but user can set color according to the requirements.
<code>fillRect(x, y, width, height)</code>	Enables the user to draw a rectangle with the existing fill style.
<code>strokeStyle</code>	The values for this property can be gradient, pattern, or a CSS color.
<code>strokeRect(x, y, width, height)</code>	Enables the user to draw a rectangle with the existing stroke style. This property is used to draw the edges of the rectangle.
<code>clearRect(x, y, width, height)</code>	Used to clear the pixels in a rectangle.

Designing Modernistic Websites © Aptech Limited

9

## Working with Drawing Objects 2-11

- The Code Snippet demonstrates how to create a rectangle in HTML5 canvas.

```
<!DOCTYPE HTML>
<html>
  <head>
    <style>
      #mCanvas {
        border: 1px solid green;
      }
      body {
        margin: 0px;
        padding: 0px;
      }
    </style>
    <script>
      window.onload = function() {
        var canvas = document.getElementById('mCanvas');
        var ctext = canvas.getContext('2d');
        ctext.beginPath();
        ctext.rect(30, 50, 150, 100);
        ctext.fillStyle = "Magenta";
        ctext.fill();
        ctext.lineWidth = 5;
        ctext.strokeStyle = 'black';
        ctext.stroke();
      };
    </script>
  </head>
  <body>
    <canvas id="mCanvas" width="278" height="200"></canvas>
  </body>
</html>
```

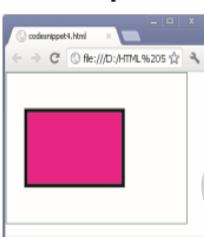
Designing Modernistic Websites © Aptech Limited

10

## Working with Drawing Objects 3-11

- In the code, the `height` and `width` attributes are defined.
- The initializer function has the DOM object which is accessed through the `id` attribute and gets a 2d context by using the `getContext()` method.
- The `beginPath()` method is called through the context object to draw the rectangle.
- The `rect(30,50,150,100)` method takes `x, y, height, and width` as the parameters.
- The `fillStyle` property fills the rectangle with magenta color.
- The `fill()` method is used to paint the rectangle.
- The `lineWidth` property is specified as 5 to define the width of line on the canvas.
- The `strokeStyle` property sets the stroke style of the rectangle to black.
- The `stroke()` method assigns the color to the rectangle.

Designing Modernistic Websites © Aptech Limited



11

## Working with Drawing Objects 4-11

### ➤ Arcs

- With HTML5 canvas, the user can create an arc by using the `arc()` method.
- Arcs are represented using a start angle, an end angle, a radius, a center point, and the drawing direction (anticlockwise or clockwise).
- The syntax to draw an arc in HTML5 is as follows:

#### Syntax:

```
arc(x, y, radius, startAngle, endAngle, anticlockwise)
```

where,

- `x, y` - Specifies the coordinates of the center of an arc
- `radius` - Specifies the distance from the center to any point on the circle
- `startAngle, endAngle` - Specifies the start and end points in the arc
- `anticlockwise` - Draws the arc clockwise or anticlockwise and accepts a boolean value

Designing Modernistic Websites © Aptech Limited

12

## Working with Drawing Objects 5-11

- The Code Snippet demonstrates how to create an arc in HTML5 canvas.

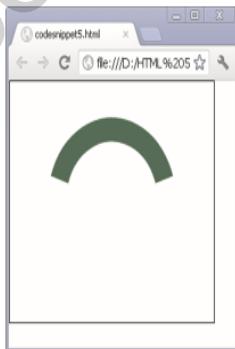
```
<!DOCTYPE HTML>
<html>
<head>
<style>
body
{
    margin: 0px;
    padding: 0px;
}
#mCanvas {
    border: 1px solid black;
}
</style>
<script>
window.onload = function() {
    var canvas = document.getElementById("mCanvas");
    var ctext = canvas.getContext("2d");
    var x = canvas.width / 2;
    var radius = 75;
    var startAngle = 1.1 * Math.PI;
    var endAngle = 1.9 * Math.PI;
    var ctrClockwise = false;
    ctext.beginPath();
    ctext.arc(x, y, radius, startAngle, endAngle, ctrClockwise);
    ctext.lineWidth = 25;
    // line color
    ctext.strokeStyle = "DarkGreen";
    ctext.stroke();
};
</script>
</head>
<body>
<canvas id="mCanvas" width="278" height="250"></canvas>
</body></html>
```

Designing Modernistic Websites © Aptech Limited

13

## Working with Drawing Objects 6-11

- In the code, the `beginPath()` method is called through the context object to draw an arc by using the `arc()` method which has `x`, `y`, and `radius` as the parameters.
- The `startAngle` and the `endAngle` are the start and end points of the arc.
- The `anticlockwise` specifies the direction of the arc between the two start and end points.
- Following figure displays an arc in HTML5 canvas.



Designing Modernistic Websites © Aptech Limited

14

## Working with Drawing Objects 7-11

### ► Circle

- In HTML5, you can draw a circle using the `arc()` method.
- You have to set the start angle with 0 and the end angle is specified as  $2 * \text{PI}$ .
- Following is the syntax to draw a circle in HTML5 as follows:

#### Syntax:

```
arc(x,y, radius,startAngle,endAngle, anticlockwise)
```

where,

- `x,y` - Specifies the coordinates of the center of an arc
- `radius` - Specifies the distance from the center to any point on the circle
- `startAngle,endAngle` - Specifies the start and end points in the arc
- `anticlockwise` - Draws the arc clockwise or anticlockwise and accepts a boolean value

Designing Modernistic Websites © Aptech Limited

15

## Working with Drawing Objects 8-11

- The Code Snippet demonstrates how to create a circle using HTML5.

```
<!DOCTYPE HTML>
<html>
  <head>
    <style>
      body {
        margin: 0px;
        padding: 0px;
      }
      #mCanvas
      {
        border: 1px solid blue;
      }
    </style>
    <script>
      window.onload = function() {
        var canvas = document.getElementById("mCanvas");
        var ctext = canvas.getContext("2d");
        var ctrX = canvas.width / 2;
        var ctrY = canvas.height / 2;
        var radius = 70;
        ctext.beginPath();
        ctext.arc(ctrX, ctrY, radius, 0, 2 * Math.PI, false);
        ctext.fillStyle = "DarkOrchid";
        ctext.fill();
        ctext.lineWidth = 4;
        ctext.strokeStyle = "black";
        ctext.stroke();
      };
    </script>
  </head>
  <body>
    <canvas id="mCanvas" width="356" height="150"></canvas>
  </body>
</html>
```

Designing Modernistic Websites © Aptech Limited

16

## Working with Drawing Objects 9-11

- In this code, a circle is defined by using the `arc()` method which has `ctrX`, `ctrY`, and `radius` as the parameters.
- To define the arc with the points the `startAngle` is set to 0 and the `endAngle` is specified as  $2\pi$  PI.
- The `anticlockwise` defines the direction of the path of an arc between the two start and end points.

### ➤ Bezier Curves

- Using HTML5 canvas, you can create a Bezier curve using the `bezierCurveTo()` method.
- Bezier curves are represented with the two control points, context points, and an end point.

Designing Modernistic Websites © Aptech Limited

17

## Working with Drawing Objects 10-11

- The Code Snippet demonstrates how to create a Bezier curve using HTML5.

```
<!DOCTYPE HTML>
<html>
<head>
<style>
body
{
    margin: 0px;
    padding: 0px;
}
#mCanvas
{
    border: 1px solid maroon;
}
</style>
<script>
window.onload = function()
{
    var canvas = document.getElementById("mCanvas");
    var ctext = canvas.getContext("2d");
    ctext.beginPath();
    ctext.moveTo(198, 130);
    ctext.bezierCurveTo(140, 10, 388, 10, 288, 100);
    ctext.lineWidth = 15;
    // line color
    ctext.strokeStyle = "purple";
    ctext.stroke();
}
</script>
</head>
<body>
<canvas id="mCanvas" width="378" height="200"></canvas>
</body>
</html>
```

Designing Modernistic Websites © Aptech Limited

18

## Working with Drawing Objects 11-11

### ➤ Quadratic Curves

- HTML5 canvas allows the user to create quadratic curves using the `quadraticCurveTo()` method.
- Quadratic curves are represented through the context point, an end point, and a control point.

The Code Snippet demonstrates how to create a quadratic curve using HTML5.

```
<!DOCTYPE HTML>
<html>
<head>
<style>
body
{
    margin: 0px;
    padding: 0px;
}
#mCanvas
{
    border: 1px solid #9c9999;
}
window.onload = function()
{
    var canvas = document.getElementById("mCanvas");
    var ctext = canvas.getContext("2d");
    ctext.beginPath();
    ctext.moveTo(178, 150);
    ctext.quadraticCurveTo(220, 0, 320, 150);
    ctext.lineWidth = 15;
    // line color
    ctext.strokeStyle = "Fuchsia";
    ctext.stroke();
}
</script>
</head>
<body>
<canvas id="mCanvas" width="378" height="200"></canvas>
</body>
</html>
```

Designing Modernistic Websites © Aptech Limited

19

### Instructions to the Trainer(s):

- Using Slides 9 to 19, explain the working with drawing object.
- HTML5 canvas allows the user to work with different types of drawing objects.
- Using Slides 9 to 11 explain the rectangle object.
- With HTML5 canvas, the user can create a rectangle using the `rect()` method.
- The HTML5 canvas is placed by using the `x` and `y` parameters and appropriately sized through height and width properties. There is a collection of methods and properties that are used to draw different types of shapes. Explain the list of common properties and methods of various shapes.
- Using Slides 12 to 14, explain working with drawing object, arc.
- With HTML5 canvas, the user can create an arc by using the `arc()` method. Arcs are represented using a start angle, an end angle, a radius, a center point, and the drawing direction (anticlockwise or clockwise).
- Using Slides 15 to 17, explain the working with drawing object, circle.
- In HTML5, circle can be drawn using the `arc()` method. You have to set the start angle with `0` and the end angle with `2 * PI`.
- Using Slides 17 to 19, explain the Bezier curves and quadratic curves.
- Using HTML5 canvas, you can create a Bezier curve using the `bezierCurveTo()` method. Bezier curves are represented with the two control points, context points and an end point.
- HTML5 canvas allows the user to create quadratic curves using the `quadraticCurveTo()` method. Quadratic curves are represented through the context point, an end point, and a control point.

**In-Class Question:**

**Question:** Which method is used to moves the path to the specified point in the canvas, without creating a line?

**Answer:** `moveTo()` method is used to moves the path to the specified point in the canvas, without creating a line.

## Working with Images

- In HTML5, the user can draw image objects on canvas using the `drawImage()` method.
- The `drawImage()` method can also draw parts of an image and increase or reduce the size of the image.
- The image object can be a video, an image, or another canvas element.

- The Code Snippet demonstrates how to create an image using HTML5.

```
<!DOCTYPE HTML>
<html>
<head>
<style>
body {
    margin: 0px;
    padding: 0px;
}
#mCanvas {
    border: 1px solid #9C9999;
}
</style>
<script>
window.onload = function() {
    var canvas = document.getElementById("mCanvas");
    var ctext = canvas.getContext("2d");
    var imgObj = new Image();
    imgObj.onload = function()
    {
        ctext.drawImage(imgObj, 69, 50);
    };
    imgObj.src = "bird.jpg";
}
</script>
</head>
<body>
<canvas id="mCanvas" width="368" height="300"></canvas>
</body>
</html>
```

Designing Modernistic Websites © Aptech Limited

20

### Instructions to the Trainer(s):

- Using Slide 20, explain the drawing of images on the canvas.
- In HTML5, the user can draw image objects on canvas using the `drawImage()` method.
- The `drawImage()` method can also draw parts of an image and increase or reduce the size of the image.
- This method accepts nine parameters, depending on editing that is required on the image.
- The image object can be a video, an image, or another canvas element.

## Slides 21 and 22

### Working with Text 1-2

- HTML5 canvas enables you to set the font, style, and size of text by using the font properties.
- The font style can be italic, normal, or bold.
- To set the text color, the `fillStyle` property of the canvas can be used.
- The Code Snippet demonstrates how to set the font, size, style, and color of the text on a HTML5 canvas.

```
<!DOCTYPE HTML>
<html>
  <head>
    <style>
      body {
        margin: 0px;
        padding: 0px;
      }
      #mCanvas {
        border: 1px solid blue;
      }
    </style>
  <script>
    window.onload = function() {
      var canvas = document.getElementById("mCanvas");
      var ctext = canvas.getContext("2d");
      ctext.font = "italic 30pt Calibri";
      ctext.fillStyle = "MediumVioletRed";
      ctext.fillText("Welcome to HTML5!", 40, 100);
    }
  </script>
</head>
<body>
  <canvas id="mCanvas" width="300" height="170"></canvas>
</body>
</html>
```

Designing Modernistic Websites © Aptech Limited

21

### Working with Text 2-2

- The Code Snippet demonstrates the use of stroke text in HTML5 canvas.

```
<!DOCTYPE HTML>
<html>
  <head>
    <style>
      body {
        margin: 0px;
        padding: 0px;
      }
      #mCanvas {
        border: 1px solid black;
      }
    </style>
  <script>
    window.onload = function() {
      var canvas = document.getElementById("mCanvas");
      var ctext = canvas.getContext("2d");
      var x = 80;
      var y = 110;
      ctext.font = "40pt Calibri";
      ctext.lineWidth = 2;
      // stroke color
      ctext.strokeStyle = "Brown";
      ctext.strokeText("HTML5", x, y);
    }
  </script>
</head>
<body>
  <canvas id="mCanvas" width="360" height="200"></canvas>
</body>
</html>
```

Designing Modernistic Websites © Aptech Limited

22

**Instructions to the Trainer(s):**

- Using Slides 21 and 22, explain how to work with text.
- HTML5 canvas enables you to set the font, style, and size of text by using the font properties.
- The font style can be italic, normal, or bold. For setting the text color, you can use the `fillStyle` property of the canvas.

## Using Transparency for Text in Canvas

The Code Snippet demonstrates the use of globalAlpha property.

```
<!DOCTYPE HTML>
<html>
<head>
<style>
body {
    margin: 0px;
    padding: 0px;
}
#mCanvas {
    border: 1px solid black;
}
</style>
<script>
window.onload = function() {
    var canvas = document.getElementById("mCanvas");
    var ctext = canvas.getContext("2d");
    ctext.fillStyle = "indigo";
    ctext.strokeStyle = "black";
    ctext.lineWidth=2;
    ctext.font = "italic 30pt Calibri";
    ctext.fillText("HTML5", 40, 100);
    ctext.strokeText("HTML5", 40, 100);
    ctext.fillStyle="blue";
    ctext.globalAlpha=0.5;
    ctext.fillRect(100, 10, 150, 100);
};
</script>
</head>
<body>
<canvas id="mCanvas" width="350" height="170"></canvas>
</body>
</html>
```

Designing Modernistic Websites © Aptech Limited

23

### Instructions to the Trainer(s):

- Using Slide 23, explain transparency of text in canvas.
- The `globalAlpha` drawing state property, is the method which can be applied universally.
- The `globalAlpha` property is a value that ranges between 0 (fully transparent) and 1 (fully opaque).

## Using Events with jQuery 1-3

- jQuery also offers different events to deal with common interactions when the user moves the mouse or switches between two actions while clicking.
  - Following are the events:
    - **hover()** event
- The mouseenter and mouseleave are the two events often used together.  
jQuery provides a `hover()` function that accepts two parameters.  
The first parameter executes when the mouse moves over the element and the second function executes when the mouse moves away from the element.

The Code Snippet demonstrates the hover event.

```
<!DOCTYPE html>
<html>
<head>
<script src="jquery-1.7.2.min.js"></script>
<script>
$(document).ready(function(){
  $("p").hover(function(){
    $("p").css("background-color","red");
  },function(){
    $("p").css("background-color","maroon");
  });
});
</script>
</head>
<body>
<p>Hover the mouse on this line.</p>
</body>
</html>
```

Designing Modernistic Websites © Aptech Limited

24

## Using Events with jQuery 2-3

### ➤ **toggle()** event

- The `toggle()` event works in a similar manner as that of the `hover()` event, except that it responds to mouse clicks.
- The `toggle()` function accepts more than two functions as arguments.
- All the functions passed to the `toggle()` event will react to its corresponding click action.
- The Code Snippet demonstrates the `toggle` event.

```
<!DOCTYPE html>
<html>
<head>
<script src="jquery-1.7.2.min.js"></script>
<script>
$(document).ready(function(){
  $("p").toggle(function(){
    $("body").css("background-color","blue");
    function(){
      $("body").css("background-color","pink");
    }
  });
});
</script>
</head>
<body>
<p>Click to change the colors.</p>
</body>
</html>
```

Designing Modernistic Websites © Aptech Limited

25

## Using Events with jQuery 3-3

- Following figure displays the toggle effect to blue:



- Following figure displays the toggle effect to pink:



- Following figure displays the toggle effect to grey:



Designing Modernistic Websites © Aptech Limited

26

### Instructions to the Trainer(s):

- Using Slides 24 to 26, explain the events with jQuery.
- A Web page can respond to user actions through events. Some of the examples of the events are as follows:
  - Moving a mouse over an element
  - Selecting a radio button
  - Clicking an element on the page
- jQuery also offers different events to deal with common interactions when the user moves the mouse or switch between two actions while clicking.
- Using Slide 24 explain the `hover()` event.
- The `mouseenter` and `mouseleave` are the two events often used together. For example, when a user moves a mouse over a menu, a tooltip appears and when the user moves the mouse off the menu, the tooltip disappears. Combining these two events is very common, therefore, jQuery provides a `hover()` function that accepts two parameters. The first parameter executes when the mouse moves over the element and the second function executes when the mouse moves away from the element.
- Using Slide 25, explain the `toggle()` event.
- The `toggle()` event works in a similar manner as that of the `hover()` event, except that it responds to mouse clicks. The `toggle()` function accepts more than two functions as arguments. For example, you want to perform some action on the first click, another action on the second click, and one more action on the third click. All the functions passed to the `toggle()` event will react to its corresponding click action.

## Inclusion of External Content in Web Pages

HTML5 introduces the `<eventsource>` tag that allows the user to push external content in the Web page. This model is referred to as push model.

Since the `<eventsource>` tag is not supported in many browsers, users make use of the `<embed>` tag for this purpose.

The `<embed>` tag is a new element in HTML5 and it is represented as a container for an interactive content or an external application.

The `<embed>` tag is often used to add elements such as image, audio, or video on a Web page.

- The Code Snippet demonstrates the use of `<embed>` tag.

```
<embed src="mymovie.mp3" />
```

- In this code, the `src` attribute specifies the path of an external file to embed.

### Instructions to the Trainer(s):

- Using Slide 27, explain how to include the external content in Web pages.
- HTML5 introduces the `<eventsource>` tag that allows the user to push external content in the Web page. This model is referred to as push model.
- Since the `<eventsource>` tag is not supported in many browsers, users make use of the `<embed>` tag for this purpose.
- The `<embed>` tag is a new element in HTML5 and it is represented as a container for an interactive content or an external application.
- The `<embed>` tag is often used to add elements such as image, audio, or video on a Web page.
- In the code, the `src` attribute specifies the path of an external file to embed.

## Cookies and Web Storage

Traditionally, over the last few decades, Web applications have been using cookies to store small amounts of information on a user's computer.

A cookie is a file that stores user-related information and may either be temporary or permanent.

A cookie can be created for login details which can be saved for a specified period on a user's computer.

- Drawbacks of cookies are as follows:
  - Cookies slow down the performance of Web application, as they are included with every HTTP request
  - Cookies cannot be considered as safe means for transmission of sensitive data
  - Cookies cannot store large amount of information, as they have a limitation of size of four kb
- W3C has designed a specification named Web Storage API which offer a solution to store data on the client-side

Is a W3C specification and certain browsers refer to it as 'DOM Storage'.

Provides functionality for storage of data on the client-side that is on user's machine.

Stores data that can cater for both temporary as well as permanent needs.

Offers more control than traditional cookies, and is easy to work with.

Was originally a part of the HTML5 specification, but now it is present in a separate specification and stores a maximum of five mb of information per domain.

Designing Modernistic Websites © Aptech Limited

28

### Instructions to the Trainer(s):

- Using Slide 28, explain the concept of cookies and Web Storage.
- Traditionally, over the last few decades, Web applications have been using cookies to store small amounts of information on a user's computer.
- A cookie is a file that stores user-related information and may either be temporary or permanent. Thus, in this case, a cookie can be created for login details which can be saved for a specified period on a user's computer.
- To overcome these drawbacks and offer a solution to store data on the client-side, W3C has designed a specification named, Web Storage API.
- The Web storage provides the functionality using which data can be stored on the client-side for a session or beyond the session.

## Web Storage versus Cookies

- Some key differences between cookies and Web storage are as follows:

Cookies are meant to be read on the server-side, whereas Web storage is available only on the client-side.

Cookies are sent along with each HTTP request to the server, whereas Web storage data is not carried over to the server.

Cookies result in bandwidth overhead and thus lead to high costs, as they are sent with each HTTP request. The Web storage is stored on the user's hard drive, so it costs nothing to use.

With cookies, the information data that could be stored is four kb, whereas with Web storage, a large amount of data can be stored upto five mb.

### Instructions to the Trainer(s):

- Using Slide 29, explain the Web storage versus cookies.
- Explain the difference between Web storage and cookies as mentioned on the Slide.

## Browser-specific Web Storage

Web storage is browser-specific and the location where the Web storage data is stored depends on the browser.

Each browser's storage is separate and independent, even if it is present on the same machine.

HTML5 Web storage is implemented natively in most Web browsers, so one can use it even when third-party browser plug-in is not available.

- Following table lists the support of various browsers for HTML5 Web storage:

Browser	Version
IE	8.0+
Firefox	3.6+
Safari	4.0+
Chrome	5.0+
Opera	10.5+

Designing Modernistic Websites © Aptech Limited

30

**Instructions to the Trainer(s):**

- Using Slide 30, explain the browser-specific Web storage.
- Web storage is browser-specific. If a user visits a site in Google Chrome, any data will be stored to Google Chrome's Web storage store.
- Similarly, if the user revisits that same site in Firefox, the data saved earlier through Google Chrome will be unavailable.
- The location where the Web storage data is stored depends on the browser.
- Each browser's storage is separate and independent, even if it is present on the same machine.
- HTML5 Web storage is implemented natively in most Web browsers, so one can use it even when a third-party browser plug-in is not available.

## Exploring Web Storage

Two types of HTML5 Web storage are namely, session storage and local storage.

Both session and local storage enable to store around five mb of data per domain.

To check for browser support of HTML5 Web storage, a property named `localStorage` or `sessionStorage` is available as a global variable for the window object.

If there is no support, the `localStorage` or `sessionStorage` property will be undefined.

```
<!DOCTYPE html>
<html>
<head>
<title>Support for Web Storage</title>
<script>
function checkSupport() {
  if ('sessionStorage' in window) && window['sessionStorage'] != null)
  {
    alert("Your browser supports Web Storage");
    return;
  }
  alert("Your browser does not support Web Storage");
}
</script>
</head>
<body onload="checkSupport();">
</body>
</html>
```

Code Snippet demonstrates the script to check the support for HTML5 Web storage in the browser.

Designing Modernistic Websites © Aptech Limited

31

### Instructions to the Trainer(s):

- Using Slide 31, explain the Web storage in detail.
- The two types of HTML5 Web storage are namely, session storage and local storage.
  - **localStorage** - The data will be stored in the browser permanently.
  - **sessionStorage** – The data will be stored within the browser only for a single session.
- Both session and local storage enable to store around 5 MB of data per domain.
- To check for browser support of HTML5 Web storage, a property named `localStorage` or `sessionStorage` is available as a global variable for the window object.
- If there is no support, the `localStorage` or `sessionStorage` property will be undefined.

### In-Class Question:

**Question:** Which property is used for checking the support Web storage?

**Answer:** The `localStorage` or `sessionStorage` properties are used for checking the support Web storage.

## Session Storage 1-3

Keeps track of data specific to one window or tab and discards it as soon the user closes the tab (or window) that he/she was working with.

Lasts for the entire duration of the session and hence, is not persistent.

Makes use of named key/value pairs which are enclosed within double quotes.

Stores the data using the named key, whereas the data is retrieved by referring to that key.

Key is a string, whereas the value stored in the key can be of any data type such as string, boolean, integer, or float. Regardless of the type of data that is stored, it is actually stored internally as a string.

Storing and retrieving data of other types requires the use of functions to convert them into the appropriate data types.

- Following table lists some examples of named key/value pairs belonging to various data types:

Key	Value
Name	Sarah
book	C Programming
Email	info@me.com
car	Toyota Innova
age	28
uservalid	true

Designing Modernistic Websites © Aptech Limited

32

## Session Storage 2-3

Storing and retrieving data - `setItem()` and `getItem()` methods are used to store and retrieve data from session storage respectively.

- Syntax to use `setItem()` and `getItem()` methods is as follows:

- To assign data

```
sessionStorage.setItem(key, value);
```

where,

key: Is the named key to refer to the data.

value: Is the data to be stored.

- To retrieve data

```
var item = sessionStorage.getItem(key);
```

where,

item: Is the variable into which the data will be saved.

key: Is the named key to refer to the data.

- To remove data

```
sessionStorage.removeItem(key);
```

where,

key: Is the named key to refer to the data.

- To clear data

```
sessionStorage.clear();
```

Designing Modernistic Websites © Aptech Limited

33

## Session Storage 3-3

- Code Snippet demonstrates how to set and retrieve a name using sessionStorage object.

```
<!DOCTYPE html>
<html>
  <head>
    <title>Working with Session Storage</title>
    <script>
      function testStorage() {
        if ('sessionStorage' in window) {
          window['sessionStorage'] !== null)
        {
          sessionStorage.setItem('name', 'Sarah');
          alert('The name is: ' +
            sessionStorage.getItem('name'));
        }
      }
    </script>
  </head>
  <body onload="testStorage();">
  </body>
</html>
```



Designing Modernistic Websites © Aptech Limited

34

### Instructions to the Trainer(s):

- Using Slides 32 to 34, explain the session storage.
- Session storage keeps track of data specific to one window or tab and discards it as soon as the user closes the tab (or window) that he/she was working with.
- Thus, even if you are visiting the same site in two different windows, each window will have its own individual session storage object.
- This means that each window contains separate session storage object with distinct data.
- Session storage lasts for the entire duration of the session and hence, is not persistent.
- Session storage makes use of named key/value pairs. The data is stored using the named key, whereas the data is retrieved by referring to that key. Both the key-value pairs are enclosed within double quotes.
- The key is a string, whereas the value stored in the key can be of any type of data, such as string, boolean, integer, or float. Regardless of the type of data that is stored, it is actually stored internally as a string.
- Therefore, storing and retrieving data of other types requires the use of functions to convert them into the appropriate data types.

## Local Storage 1-3

Enables to save data for longer periods on the user's computer, through the browser.

Data is persistent and can be retrieved when a user visits the site again.

Is used, if data needs to be stored for more than a single session.

Works in a similar fashion as session storage and uses similar functions, such as `setItem()`, `getItem()`, `removeItem()`, and `clear()`.

Designing Modernistic Websites © Aptech Limited

35

## Local Storage 2-3

- **Code Snippet** demonstrates use of local storage to store the value of **username** field and later, retrieve the value in another Web page.

```
<!DOCTYPE html>
<html>
<title> Local Storage </title>
<script>
function store() {
    if ('localStorage' in window) && window['localStorage'] !== null) {
        var username = document.getElementById('username').value;
        localStorage.setItem('username', username);
    } else {
        alert ('your browser does not support storage');
    }
}
function cancel_store() {
    if ('localStorage' in window) && window['localStorage'] !== null) {
        localStorage.removeItem('username');
    } else {
        alert ('your browser does not support storage');
    }
}
</script>
</head>
<body>
<form method="get" action="success.html">
    Username: <input type="text" id="username" value="" size="20" onblur="store()"/>
    <input type="submit" value="Submit"/>
    <input type="reset" Value="Cancel" onclick="cancel_store()"/>
</body>
</html>
```

Designing Modernistic Websites © Aptech Limited

36

## Local Storage 3-3

- Code Snippet shows the success.html page that retrieves value from the local storage and displays it in the browser.

```
<!DOCTYPE html>
<head>
    <title> Local Storage </title>
    <script>
        function print() {
            var username = localStorage.getItem('username');
            document.getElementById('lblMsg').innerHTML = 'Username:
                is <b>' + username + '</b>';
        }
    </script>
</head>
<body onload="print()">
    <label id="lblMsg"></label><br>
</body>
</html>
```



Designing Modernistic Websites © Aptech Limited

37

### Instructions to the Trainer(s):

- Using Slides 35 to 37, explain the local storage.
- Unlike session storage, local storage enables to save data for longer periods on the user's computer, through the browser.
- The data is persistent and can be retrieved when a user visits the site again.
- In other words, local storage is used, if data must be stored for more than a single session.
- A simple scenario would be to count the number of times a person has visited a Web page.
- In terms of methods, local storage works in a similar fashion as session storage. It uses the same functions, such as `setItem()`, `getItem()`, `removeItem()`, and `clear()`.

## Indexed Database API 1-3

A database is an organized collection of data.

Databases, such as relational database stores the data in the form of tables.

A table comprises rows and columns that are used to store data.

The representation of data from a table is in the form of records.

HTML5 has introduced a new Web Storage API which can host Web databases locally within the user browser.

Web databases are not like relational databases in terms of functionality.

## Indexed Database API 2-3

Indexed Database API is a specification also known as IndexedDB.

It is basically an object store that can be used to store and manipulate data on the client-side.

The object store is the primary storage mechanism that stores the object in the database managed locally within the browser.

It enables to create an object store of a particular type in which objects can be persisted using JavaScript.

IndexedDB enables to create Web applications with rich query abilities and which can work both online and offline.

IndexedDB supports two types of API namely, synchronous and asynchronous.

The synchronous API can be used with WebWorkers, whereas asynchronous API can be used for Web applications.

## Indexed Database API 3-3

IndexedDB API is implemented using `window.indexedDB` object.

Browsers implement the IndexedDB object with their own prefixes. For example, Chrome browser uses the `webkit` prefix, whereas Mozilla supports `-moz` prefix.

- Following table lists the browser support for the IndexedDB API:

IE	Firefox	Chrome	Safari	Opera	iOS Safari
6.0	-	-	-	-	3.2
7.0	8.0moz	-	-	-	4.0-4.1
8.0	9.0moz	16.0webkit	5.0	-	4.2-4.3
9.0	10.0moz	17.0webkit	5.1	11.6	5.0
10.0	11.0moz	18.0webkit	6.0	12.0	-
-	12.0moz	19.0webkit	-	-	-

Designing Modernistic Websites © Aptech Limited

40

### Instructions to the Trainer(s):

- Using Slides 38 to 40, explain the Indexed database API.
- HTML5 has introduced a new Web Storage API which can host Web databases locally within the user browser. However, Web databases are not like relational databases in terms of functionality.
- Indexed Database API is a specification also known as IndexedDB. It is basically an object store that can be used to store and manipulate data on the client-side.
- The object store is the primary storage mechanism that stores the object in the database managed locally within the browser.
- It enables to create an object store of a particular type in which objects can be persisted using JavaScript.
- Thus, IndexedDB enables to create Web applications with rich query abilities and which can work both online and offline.
- The IndexedDB API is implemented using `window.indexedDB` object.
- As the current specification is still in the evolving stage, browsers implement the `IndexedDB` object with their own prefixes. For example, Chrome browser uses the `webkit` prefix, whereas Mozilla supports `-moz` prefix.

## Slides 41 and 42

### Indexed DB API 1-2

- Some of the basic constructs of IndexedDB API are as follows:

#### Database

The IndexedDB database comprises more than one object store. Each database contains a name that identifies the origin of the database and a version number which identifies the lifetime of the database.

#### Object Store

Is the main mechanism to store data in a database. They hold the data stored in the database in the form of records.

#### Keys

Each record stored in the database is identified by a unique key.

#### Values

Are the data stored in the records.

#### Key Path

Is a string that defines how the browser should extract key from a value. The key from a value can be extracted either in the object store or index.

#### Index

Is used when the data from the object store is retrieved based on some other values other than a key.

#### Transaction

Any addition or retrieval of the data in a database is performed by using transaction. Each transaction has a mode, scope, and a request list.

Designing Modernistic Websites © Aptech Limited

41

### Indexed DB API 2-2

- Some of the basic constructs of IndexedDB API are as follows:

**Requests** - Operations, such as reading or writing on the database is performed using a request. Each request contain attributes, such as flag, source object, result, and error.

**Cursor** - Is a mechanism used to retrieve multiple records from a database.

**Key Range** - Records from the object stores and indexes are retrieved using keys or key ranges. A key range refers to retrieval of data between specified bounds based on the keys.

Designing Modernistic Websites © Aptech Limited

42

### Instructions to the Trainer(s):

- Using Slides 41 and 42, explain the basic constructs of IndexedDB API.

- Tell the students that the basic constructs are Database, Object Store, Keys, Values, Key Path, Index, and Transaction.
- Also, some of the basic constructs of IndexedDB PI are:
  - Requests
  - Cursor
  - Key Range

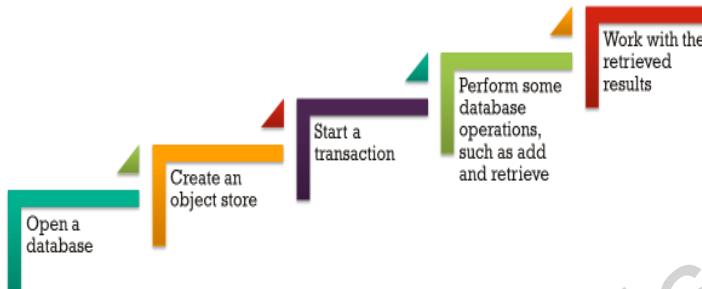
**In-Class Question:**

**Question:** Which construct is used to retrieve multiple records?

**Answer:** Cursor is used to retrieve multiple records.

## Implementing IndexedDB API 1-5

- Steps to implement the IndexedDB API in a Web application are as follows:



Designing Modernistic Websites © Aptech Limited

43

## Implementing IndexedDB API 2-5

- **Opening a Database**

- Code Snippet shows the code to open a database

```
var indexedDB = window.indexedDB || window.webkitIndexedDB || window.mozIndexedDB;
var request = indexedDB.open("CompanyDB", 1);
request.onsuccess = function (event) {
    ...
};
request.onerror = function (event) {
    console.log("IndexedDB error: " + event.target.errorCode);
};
```

- **Updating Version of a Database**

After the database is opened, it can be structured by providing a version number which helps to set up the database.

- Code Snippet shows the code that specifies the version number to the database

```
var setVRequest = db.setVersion("1.99");
setVRequest.onsuccess = function(event) {
    ...
}
```

Designing Modernistic Websites © Aptech Limited

44

## Implementing IndexedDB API 3-5

- **Creating the Object Store**

Structure of IndexedDB database facilitates the storage of multiple object stores.

Object store is created using `createObjectStore()` method which accepts two arguments namely, the store name and a parameter object.

- Code snippet demonstrates the code to create an object store named `employee` in the `CompanyDB` database.

```
var employeeData = [  
    { name: "John Smith", email: "john@company.com" },  
    { name: "Jill Patrick", email: "jill@company.com" },  
    { name: "Rock Ethan", email: "rock@company.com" },  
    { name: "Daniel Andrew", email: "daniel@company.com" }  
];  
var objectStore = db.createObjectStore("employee", {  
    keyPath: "id", autoIncrement: true });  
for (i in employeeData) {  
    objectStore.put(employeeData[i]);  
    alert("Record added");  
}
```

Designing Modernistic Websites © Aptech Limited

45

## Implementing IndexedDB API 4-5

- **Creating a Transaction**

To perform database operation, such as retrieving data from the object store, IndexedDB provides a `IDBTransaction` object.

This object can be created in three mode namely, `read-only`, `read-write`, and `snapshot`.

- Code Snippet demonstrates the code to retrieve data from the `employee` object store using `get()` function of the transaction object.

```
var trans = db.transaction(["employee"],  
    IDBTransaction.READ_WRITE).objectStore("employee");  
var request = trans.get(2);  
request.onerror = function(event) {  
    // Handle errors!  
};  
request.onsuccess = function(event) {  
    // Do something with the request.result!  
    alert("Name: " + request.result.name);  
};
```

Designing Modernistic Websites © Aptech Limited

46

## Implementing IndexedDB API 5-5

- **Opening a Cursor**

Cursor is used to retrieve multiple records from an object store.

They can be used when the value of key path is not known. They are part of a transaction and are opened for a particular object store.

- **Code Snippet demonstrates the code to retrieve multiple records from the employee object store.**

```
store = db.transaction("employee").objectStore("employee");
store.openCursor().onsuccess = function(event) {
    var cursor = event.target.result;
    if (cursor) {
        alert("Name for id " + cursor.key + " is " + cursor.value.name);
        cursor.continue();
    }
};
```

Designing Modernistic Websites © Aptech Limited

47

### Instructions to the Trainer(s):

- Using Slides 43 to 47, explain how to implement IndexedDb API.
- The steps to implement the IndexedDB API in a Web application are as follows:
  - Open a database
  - Create an object store
  - Start a transaction
  - Perform some database operations, such as add and retrieve
  - Work with the retrieved results

## Limitations of IndexedDB API

- Design limitations for IndexedDB API used for client-side storage of data are as follows:

Internationalized sorting deals with sorting of string data. As the database does not follow any international order for storing data, internationalized sorting is not supported by the API.

IndexedDB API does not synchronize client-side database with the server-side databases.

IndexedDB API supports querying the client-side database, but does not support the use of operators, such as LIKE that is used by Structured Query Language (SQL).

### Instructions to the Trainer(s):

- Using Slide 48, explain the limitations of IndexedDB API.
- The IndexedDB API is used for client-side storage of data, but it has some design limitations.
- Explain these limitations to the students mentioned on the slide.

## Converting HTML5 Apps to Native Apps

A native application also known as native app is an application program that is built for using it on a particular device or platform.

A native app, when compared with Web app is installed on a device and has a faster response, because it has a direct user interface.

Designing Modernistic Websites © Aptech Limited

49

**Instructions to the Trainer(s):**

- Using Slide 49, explain the conversion of HTML5 apps to native apps.
- A native application is also known as native app. It is an application program that is built to be used on a particular device or platform.
- A native app, when compared with Web app, is installed on a device and has a faster response, because it has a direct user interface.
- For example, BlackBerry Messenger (BBM) is a native app available on BlackBerry mobile devices.

## Difference between Native Apps and HTML5 Apps

HTML5 Web apps are accessible and used on any devices through Web browser similar to the mobile Website.

Web apps have the ability of offline access which means that the user need not have a network connection.

- Following table lists differences between native apps and HTML5 apps:

Native Apps	HTML5 Apps
Native Apps runs on iOS and Android devices that can be downloaded or purchased from the online app stores.	HTML5 Apps runs on a Web server, usually in a Web browser.
Native Apps use programming language, such as Java for Android devices and Objective C for iOS devices.	Web developers use HTML, JavaScript, and CSS. They need to acquire the skills of Java and objective C for writing native applications.

Designing Modernistic Websites © Aptech Limited

50

### Instructions to the Trainer(s):

- Using Slide 50, explain the difference between native apps and HTML5 apps.
- HTML5 is the latest version of HTML language providing a simple building block for Web pages. This is the first version of markup language that supports the use of multimedia without using any additional plug-in and is supported by many devices and computer systems.
- HTML5 Web apps are accessible and used on any devices through Web browser similar to the mobile Website.
- The Web apps have the ability of offline access which means that the user does not require to have a network connection.
- Explain the list of differences between the native apps and HTML5 apps.

## Advantages of HTML5 Apps

- Some of the reasons to develop HTML5 applications are as follows:

Users cannot identify the differences	Cannot identify whether they are working on a hybrid HTML5-native application or a fully native application or an HTML5 application.
Users adjust styles for devices	HTML5 apps can be viewed on any devices that contains Web browser.
Upcoming functionalities	HTML5 does not support all features on a device, but it is coming up with new functionalities.
Improving Performance	Many developers learn new methods to improve the performance of Web.
Independent device	If the developers want that their application to be used by a large number of users, then they should design and develop applications for both mobile users as well as desktop users.
Developers are not locked in app stores	HTML5 developers are not restricted to an app store. Instead, they can create applications and sell them like any other Web page.

Designing Modernistic Websites © Aptech Limited

51

### Instructions to the Trainer(s):

- Using Slide 51, explain the advantages of HTML5 apps.
- The main advantage of using HTML5 is to create applications that execute on a wide range of devices easily.
- App development on HTML5 is cheaper as compared to native app development. Developers do not have to learn any new programming language and the development becomes much easier.
- There are many reasons to develop HTML5 applications rather than native applications. Explain these reasons mentioned on the slide.

## Advantages of Native Apps

- Major advantage of native apps over HTML5 apps is that they are faster than HTML5 apps. Native apps provide more benefits over HTML5 apps. These are as follows:

Providing access to device hardware	Uploading Files	Push notifications	Accessing device files	Superior graphics than HTML5	Offline access
There are no APIs available for accelerometers, cameras, or any other device hardware for HTML5 apps.	Native apps can access the file system in Android and some files in iOS. However, the HTML5 file API does not work on Android or iOS.	The push notifications are sent always with an open IP connection to applications on the iOS device.	Native apps communicate with files on devices, such as contacts and photos. However, these files cannot be seen from HTML5 app.	HTML5 has a canvas element, but it will not create a full 3D experience.	HTML5 provides access to offline Web applications. However, a native app is stored on local machine, so that users do not require access to the Web to work with the application.

Designing Modernistic Websites © Aptech Limited

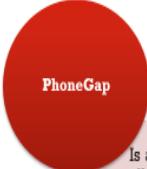
52

**Instructions to the Trainer(s):**

- Using Slide 52, explain the advantages of native apps.
- The major advantage of native apps over HTML5 apps is that they are faster than HTML5 apps.
- Similar to normal Web pages, HTML5 apps are slow, because these apps work on HTTP that uses a request/response cycle mechanism.
- When an HTTP request is made, it takes more time for the applications to execute, as it has to wait for the request to go and return back with a response.
- Native apps provide many more benefits over HTML5 apps. Explain these benefits mentioned on the Slide.

## Converting HTML5 Apps to Native Apps

- Users have a choice of developing their application in HTML5 and convert them into a native app
- Users can use some tools to convert an HTML5 app to Native app and they are as follows:



**PhoneGap**

Is an HTML5 app that allows the user to create native apps with Web technologies and is accessible to app stores and APIs.



**Appcelerator**

Is a cross-platform mobile application development support and allows the users to create Android, iOS, and mobile Web apps.

Designing Modernistic Websites © Aptech Limited

53

### Instructions to the Trainer(s):

- Using Slide 53, explain the concept of converting HTML5 apps to native apps.
- Users can use tools to convert HTML5 app to a native app. The following are the best tools used for converting an HTML5 app to native app:
  - **PhoneGap:** PhoneGap is an HTML5 app that allows the user to create native apps with Web technologies and is accessible to app stores and APIs. PhoneGap controls the Web technologies.
  - **Appcelerator:** Appcelerator is a cross-platform mobile application development support. It allows the users to create Android, iOS, and mobile Web apps. Native applications are developed using a JavaScript code base with Eclipse as the IDE.

## Summary

- ❖ The <canvas> element is a drawing area where the user can draw graphics, use images, add animations, and also add text for enhancing the user experience on Web pages.
- ❖ To create a line, on a canvas one can use the stroke(), beginPath(), lineTo(), and moveTo() methods.
- ❖ Arcs are represented using a start angle, an end angle, a radius, a center point, and the drawing direction (anticlockwise or clockwise).
- ❖ With HTML5 canvas, the user can create a rectangle using the rect() method.
- ❖ Bezier curves are represented with the two control points, context points, and an end point.
- ❖ HTML5 canvas allows the user to create quadratic curves using the quadraticCurveTo() method.
- ❖ HTML5 canvas enables the user to draw image object on canvas using the drawImage() method.
- ❖ Web Storage is a W3C specification that provides functionality for storing data on the client-side for both temporary as well as permanent needs.
- ❖ HTML5 Web applications make use of Web storage to implement client-side persistent storage and they are: session storage and local storage.
- ❖ Session storage keeps track of data specific to one window or tab.
- ❖ The setItem() and getItem() methods are used to store and retrieve the data from session storage.
- ❖ Local storage enables to save data for longer periods on the user's computer, through the browser.
- ❖ IndexedDB API is basically an object store that can be used to store and manipulate data on the client-side.
- ❖ A native application also called as native app is an application program that is built for a particular device or platform.

54

**Instructions to the Trainer(s):**

- Show students Slide 54.
- Summarize the session by reading out each point on the Slide.

## Session 14: HTML5 Geolocation and APIs

### 14.1 Pre-Class Activities

Before you commence the session, you should familiarize yourself with the topics of this session in-depth. Prepare a question or two that will be a key point to relate the current session objectives.

#### 14.1.1 Teaching Skills

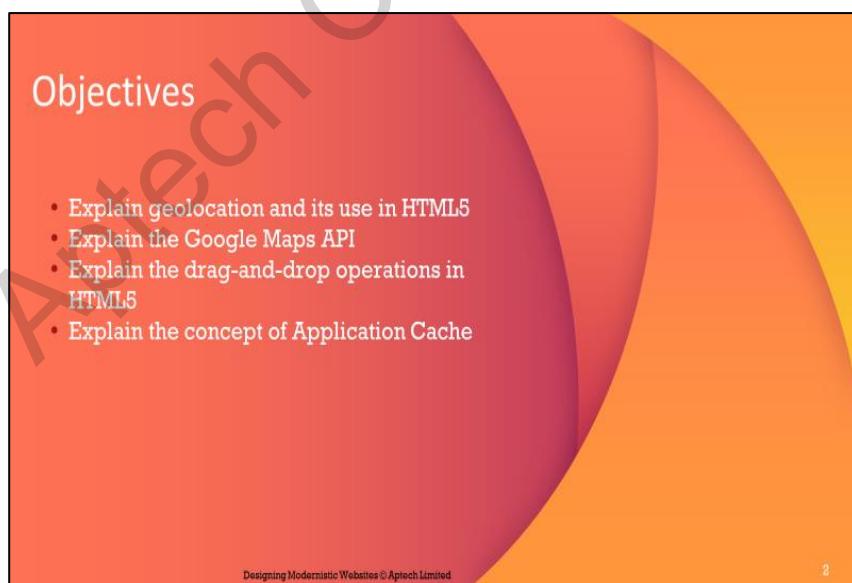
To teach this session, you should be well-versed with the new APIs supported by HTML5 such as Geolocation API and Google Maps API that are used to determine and display the location on a map. Along with this, you should prepare yourself to explain the drag-and-drop mechanism which is used to perform the drag-and-drop operations. Also, learn about the description of Application Cache.

You should teach the concepts in the theory class using the images provided. For teaching in the class, you are expected to use slides and LCD projectors.

### In-Class Activities

Follow the order given here during In-Class activities.

Slide 2



### Instructions to the Trainer(s):

Give students a brief overview of the current session through the session objectives listed in Slide 2.

## 14.2 In-Class Explanations

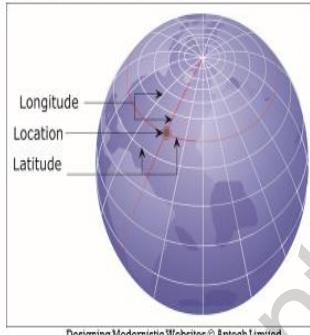
Slides 3 and 4

### Geolocation 1-2

Geolocation in computing terminology determines the current location of a user on the devices.

The location of the user is represented as a single point that comprises two components: latitude and longitude.

- Following figure shows the representation of latitude and longitude with respect to a location on the globe:



Designing Modernistic Websites © Aptech Limited

3

### Geolocation 2-2

- Different sources through which devices can determine the information about the location are as follows:

#### Global Positioning System (GPS)

- GPS is a satellite navigation system that provides information about the location on any part of the globe.
- The GPS system is maintained by the government of the United States.

#### IP Address

- Location information can be derived from IP Address which is assigned to devices, such as desktops, printers, and so on connected on a network.

#### GSM/CDMA Cell IDs

- These are used by the cell phones.

#### WiFi and Bluetooth MAC address

- These are used by devices that have wireless network connection.

#### User Input

- It is a software tool which can be used on any device requesting for location information.
- The information retrieved by the tool is based on the data provided by the user. For example, a zip code.

Designing Modernistic Websites © Aptech Limited

4

**Instructions to the Trainer(s):**

- Using Slides 3 and 4, explain the term, Geolocation.
- Consider a scenario where you are visiting a new city and are unaware of specific locations and routes. You want to get information regarding hotels in your locality, such as their exact address, tariffs, and so on. In such a situation, an application which can provide relevant information about the hotels based on your current location would be useful. A feature that can detect location and list relevant information based on that location is called Geolocation.
- Geolocation is a term used to identify the geographic location of a person, place, or an object. Today, modern devices such as computers, smartphones, tablets, and so on provide Internet-enabled browsers through which the geographic locations of a user or an object can be detected.
- Geolocation in computing terminology indicates a feature that determines the current location of a user on devices. The location of the user is represented as a single point that comprises two components namely, latitude and longitude. The components can be used further to retrieve more information for the user, such as businesses in the neighborhood or other users within the same coverage area.
- Figure shows the representation of latitude and longitude with respect to a location on the globe.
- There are different sources through which devices can determine the information about the location.

**In-Class Question:**

**Question:** Which source is based on satellite navigation to provide information about any location on the globe?

**Answer:** GPS

## Geolocation API 1-2

In HTML5, the Geolocation API is a specification by W3C for providing a consistent way to develop location-aware Web applications.

The Geolocation API provides a high-level interface to retrieve location information related to the hosting devices.

The interface hides the details, such as how the information is gathered or which methods were used to retrieve the information.

The object that holds implementation of the Geolocation API is the Geolocation object.

This object is used in JavaScript to retrieve the geographic information about the devices programmatically.

The browser processes the script and returns the location to the Geolocation API.

The Geolocation API is supported on most of the modern browsers available on desktop and mobile phones.

## Geolocation API 2-2

- Following table lists the browsers providing support for Geolocation API:

Browser	Version Support
Safari	5.0+
Chrome	5.0+
Firefox	3.5+
Internet Explorer	9.0+
Opera	10.6+
iOS (Mobile Safari)	3.2+
Android	2.0+

**Instructions to the Trainer(s):**

- Using Slides 5 and 6, explain the Geolocation API.
- The Geolocation API provides a high-level interface that can be used by developers to retrieve location information related to the hosting devices. The interface hides the details, such as how the information is gathered or which methods were used to retrieve the information. This helps the developer to concentrate on geographic information rather than its processing methods.
- The object that holds implementation of the Geolocation API is the `Geolocation` object.
- `Geolocation` object is used in JavaScript to retrieve the geographic information about the devices programmatically. The browser processes the script and returns the location to the Geolocation API.
- Explain the list of browsers providing support for Geolocation API.

## Implementing Geolocation Object 1-3

The Geolocation object is available as a new property of the navigator object.

The navigator object is a browser object that allows a user to retrieve information about the specific location.

- Following syntax shows how to create a Geolocation object in JavaScript:

### Syntax:

```
var geolocation = window.navigator.geolocation;
```

where,

- window: Is the top level object in JavaScript object hierarchy.

Designing Modernistic Websites © Aptech Limited

7

## Implementing Geolocation Object 2-3

- The Code Snippet demonstrates the script that tests the existence of geolocation object within a browser.

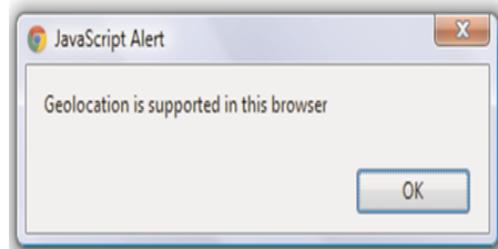
```
<!DOCTYPE html>
<html>
  <head>
    <title> Testing Support for Geolocation in Browsers</title>
    <script>
      function display_location_enabled()
      {
        // Default message
        var str = "Geolocation is not supported in this browser";
        if (window.navigator.geolocation)
        {
          str = "Geolocation is supported in this browser";
        }
        alert (str);
      }
    </script>
  </head>
  <body>
    <input type="button" value="Geolocation Support"
      onClick="display_location_enabled()"></input>
  </body></html>
```

Designing Modernistic Websites © Aptech Limited

8

### Implementing Geolocation Object 3-3

- In the code, the 'if' statement checks existence of the geolocation property in the browser.
- If browser provides implementation for the property, then an alert window displays the message 'Geolocation is supported in this browser'.
- Otherwise, the default message is displayed.
- Following figure shows the existence of geolocation object in the Chrome browser:



Designing Modernistic Websites © Aptech Limited

9

#### Instructions to the Trainer(s):

- Using Slides 7 to 9, explain how to implement Geolocation object.
- The Geolocation object is available as a new property of the navigator object.
- The navigator object is a browser object that allows a user to retrieve information about the specific location.
- Explain the syntax of creating Geolocation object.
- Then, explain the code snippet which demonstrates the script that tests the existence of Geolocation object within a browser.

## Geolocation Methods

- The geolocation object provides three methods that can be used to determine the current position of the user.
- Following table lists the methods of the geolocation object:

Method	Description
getCurrentPosition()	Retrieves the current geographic location information of the user
watchPosition()	Retrieves the geographic information of the device at regular intervals
clearWatch()	Terminates the current watch process

Designing Modernistic Websites © Aptech Limited

10

### Instructions to the Trainer(s):

- Using Slide 10, explain Geolocation methods.
- The Geolocation object provides three methods that can be used to determine the current position of the user.
- Explain these methods as listed in the table on the Slide.

## Retrieve User Information 1-5

The current position of a user is retrieved using the method  
getCurrentPosition(successCallback,errorCallback,options)

This function accepts three parameters, out of which two are optional,  
errorCallback and options.

The first parameter, successCallback is the name of the function which is  
invoked after the position of a device is found successfully.

The second parameter, errorCallback is the name of the function which will  
be called, if an error occurs in retrieving the position.

The last parameter, options represents a PositionOptions object.

## Retrieve User Information 2-5

- The Code Snippet demonstrates the script that will retrieve the current location of the user.

```
<!DOCTYPE html>
<html >
<head>
<title>Geolocation API</title>
<script>
function getLocation()
{
if (navigator.geolocation) {
navigator.geolocation.getCurrentPosition(showPosition);
}
else{
alert ("Geolocation is not supported in this browser.");
}
}
function showPosition(position)
{
alert('Latitude: ' + position.coords.latitude + '\n' +
'Longitude: ' + position.coords.longitude);
}
</script>
</head>
```

## Retrieve User Information 3-5

```
<body>
  <input type="button" value=" Display Location"
        onClick="getLocation()"/>
</body>
</html>
```

- In the code, the `getCurrentPosition()` function obtains the position which is passed as a parameter to the `showPosition()` function.
- The `showPosition()` function obtains the coordinates of a location through `position` object.
- The `position` object is defined in the Geolocation API and holds the current location of the device.
- It contains attribute named `coords` that retrieves the latitude and longitude of the location.
- The values retrieved for latitude and longitude are in decimal degrees.

Designing Modernistic Websites © Aptech Limited

13

## Retrieve User Information 4-5

- Following table lists the attributes of `position` object:

Attribute	Description
<code>coords</code>	An object of type <code>Coordinates</code> that provides different properties, such as latitude, longitude, altitude, accuracy, speed, and so on.
<code>timestamp</code>	An object of type <code>DOMTimeStamp</code> .

- Following figure shows the notifications for the Web page containing geolocation code:



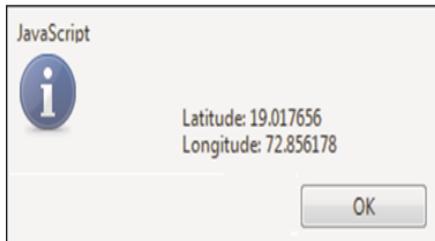
- The browser seeks permission from the user to share their location information with the application.

Designing Modernistic Websites © Aptech Limited

14

## Retrieve User Information 5-5

- Following figure shows a message displaying current location of the user, when the **Share My Location** button is clicked:



Designing Modernistic Websites © Aptech Limited

15

### Instructions to the Trainer(s):

- Using Slides 11 to 15, explain the steps and code snippet to retrieve user information.
- The current position of a user is retrieved using the `getCurrentPosition(successCallback, errorCallback, options)` method. This function accepts three parameters, out of which two are optional, `errorCallback` and `options`.
- The first parameter, `successCallback` is the name of the function which is invoked after the position of a device is found successfully.
- The second parameter, `errorCallback` is the name of the function which will be called, if an error occurs in retrieving the position.
- The last parameter, `options` represents a `PositionOptions` object.
- Explain the code snippet using Slide 12 which demonstrates the markup that will retrieve the current location of the user.
- In the code, the `getCurrentPosition()` function obtains the position which is passed as a parameter to the `showPosition()` function. The `showPosition()` function obtains the coordinates of a location through `position` object.
- The `position` object is defined in the Geolocation API and holds the current location of the device. It contains attribute named `coords` that retrieves the latitude and longitude of the location. The values retrieved for latitude and longitude are in decimal degrees.
- Explain the list of attributes of the `position` object on Slide 14.
- Figure shows the notifications for the Web page containing geolocation code. The browser seeks permission from the user to share their location information with the application.

Figure shows a message displaying current location of the user, when the **Share My Location** button is clicked.

**In-Class Question:**

**Question:** What are the parameters of the `getCurrentPosition()` method?

**Answer:** It has three parameters, `successCallback`, `errorCallback`, and `options`.

## Handling Errors 1-4

- An application could fail in gathering geographic location information. In that case, the `geolocation` object calls an `errorCallback()` function.
- The `errorCallback()` function handles errors by obtaining a `PositionError` object from the API.

### ➤ HTML

- The `PositionError` object holds information related to errors occurred while finding the geographic location of the user.
- Following table lists the properties of `PositionError` object:

Property	Description
<code>code</code>	Returns a numeric value for the type of error occurred.
<code>message</code>	Returns a detailed message describing the error encountered. The message can be used for debugging.

## Handling Errors 2-4

- Following table lists different error codes returned by the `code` property of the `PositionError` object:

Code	Constant	Description
1	<code>PERMISSION_DENIED</code>	Application does not have permission to access Geolocation API.
2	<code>POSITION_UNAVAILABLE</code>	Position of the device could not be obtained.
3	<code>TIMEOUT</code>	Unable to retrieve location information within the specified interval.

## Handling Errors 3-4

The Code Snippet demonstrates the error handling routine for the geolocation code.

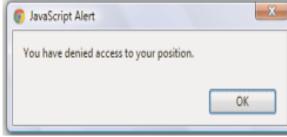
```
<!DOCTYPE html>
<html>
<head>
    <title>Handling Error</title>
<script>
    function getLocation() {
        function showPosition(position) {
            alert('Latitude: ' + position.coords.latitude + '\n' +
                'Longitude: ' + position.coords.longitude);
        }
        function errorHandler(error) {
            switch (error.code) {
                case error.PERMISSION_DENIED:
                    alert ('You have denied access to your position.');
                    break;
                case error.POSITION_UNAVAILABLE:
                    alert ('There was a problem getting your position.');
                    break;
                case error.TIMEOUT:
                    alert ('The application has timed out attempting to
                        get your position.');
                    break;
            }
        }
    </script> </head>
<body>
    <input type="button" value="Display Location"
        onClick="getLocation()" />
</body>
</html>
```

Designing Modernistic Websites © Aptech Limited

18

## Handling Errors 4-4

- In the code, if application fails to find the current location of the user, then the `errorHandler()` function is invoked.
- The function is passed as the second parameter in the `getCurrentPosition()` method and is used to handle the errors occurred in the application.
- It obtains the `error` object which is of type `PositionError` from the API and compares it with the error codes specified in the `switch-case` statement.
- Depending on the error occurred, the appropriate `case` statement is executed and an `alert` message is displayed to the user.
- Following figure shows the output displaying error message for geolocation application:



- The reason for the error is that the Chrome browser blocks the URL whose file path starts with file:///.

Designing Modernistic Websites © Aptech Limited

19

### Instructions to the Trainer(s):

- Using Slides 16 to 19, explain error handling.

- An application could fail in gathering geographic location information. In that case, the Geolocation object calls an `errorCallback()` function.
- The `errorCallback()` function handles errors by obtaining a `PositionError` object from the API. The `PositionError` object holds information related to errors occurred while finding the geographic location of the user.
- Mention the properties and methods of the `PositionError` object.
- Explain the code snippet using slides 17 and 18 which demonstrates the error handling routine for the geolocation code.
- Figure shows the output displaying error message for geolocation application. The reason for displaying error is that the Chrome browser blocks the URL whose file path starts with `file:///`.

## PositionOptions Object 1-3

- PositionOptions object is an optional third parameter passed to the `getCurrentPosition()` method.
- This object defines properties that are optional and are used by an application while retrieving the geolocation information.
- Following table lists the attributes of PositionOptions object:

Attribute	Description
<code>enableHighAccuracy</code>	Indicates that the application wants to receive the most accurate results for geolocation. The default value of the attribute is false.
<code>maximumAge</code>	Obtains the cached position object whose age is less than the specified maximumAge limit (in milliseconds). If age limit is set to 0, then the application must obtain a new position object.
<code>timeout</code>	Indicates the maximum time length (in milliseconds) for which the application can wait to obtain the position object.

Designing Modernistic Websites © Aptech Limited

20

## PositionOptions Object 2-3

- The Code Snippet demonstrates the script to set attributes of PositionOptions object.

```
<script>
    var options = {
        enableHighAccuracy: true,
        maximumAge: 50000,
        timeout: 60000
    };
    function getLocation() {
        if (navigator.geolocation) {
            navigator.geolocation.getCurrentPosition(showPosition,
                errorHandler, options);
        }
        else{
            alert ("Geolocation is not supported in this browser.");
        }
    ...
</script>
```

Designing Modernistic Websites © Aptech Limited

21

## PositionOptions Object 3-3

- In the code, an object named options is set with attributes.
- The attribute, maximumAge enables the application to use a cached position object which is not older than 50 seconds.
- Also, the timeout limit is set to 60 seconds for an application, before notifying an error.
- The options set of values is passed as third parameter to the getCurrentPosition() method.

### Instructions to the Trainer(s):

- Using Slides 20 to 22, explain PositionOptions object.
- PositionOptions object is an optional third parameter passed to the getCurrentPosition() method. This object defines properties that are optional and are used by an application while retrieving the location information.
- Explain the attributes of the PositionOptions object and code snippet that demonstrates the script to set the attributes of PositionOptions object.
- In the code, an object named options is set with attributes. The attribute maximumAge enables the application to use a cached position object which is not older than 50 seconds. Also, the timeout limit is set to 60 seconds for an application, before notifying an error.
- Options are passed as third parameter to the getCurrentPosition() method.

## Google Maps API 1-5

Google Maps API is used to display locations on a map based on the values of their coordinates, latitude and longitude.

The Google Maps API must be configured in JavaScript, before it can be referenced further on the page.

It contains a Map object which is instantiated and displayed on a Web page.

- Following syntax shows the configuration of Google Maps API in JavaScript:

### Syntax:

```
<script src="http://maps.google.com/maps/api/js?sensor=false">  
</script>
```

where,

- **src:** Is the URL of Google Maps API.
- **sensor:** Parameter sent with the URL. It indicates whether application uses any sensor such as GPS system.

Designing Modernistic Websites © Aptech Limited

23

## Google Maps API 2-5

- The Code Snippet demonstrates how to load and initialize the Google Maps API in the `<script>` tag.
- The code will execute after the page is loaded completely and will invoke a function in response to the `onload` event.

```
<!DOCTYPE html>  
<html>  
  <head>  
    <title> Load and Initialize Google Maps </title>  
    <style>  
      html { height: 100% }  
      body { height: 100%; width: 100%; margin: 10% }  
      #map_canvas { height: 50%; width: 50% }  
    </style>  
  <script  
    src="http://maps.google.com/maps/api/js?sensor=false">  
  </script>
```

Designing Modernistic Websites © Aptech Limited

24

## Google Maps API 3-5

```
function initialize()
{
    // Loading Google Maps
    var num = new google.maps.LatLng(51.528663,-0.173171);
    var myOptions = {
        zoom: 16,
        center: num,
        mapTypeId: google.maps.MapTypeId.HYBRID
    };
    var mymap = new google.maps.Map(document.getElementById("map_canvas"), myOptions);
    var marker = new google.maps.Marker({
        position: num,
        map: mymap,
        title:"Lord's Cricket Ground, London!"
    });
}
</script>
</head>
<body onload="initialize() ">
    <div id="map_canvas"></div>
</body>
</html>
```

Designing Modernistic Websites © Aptech Limited

25

## Google Maps API 4-5

- Following table lists some of the `myOptions` properties:

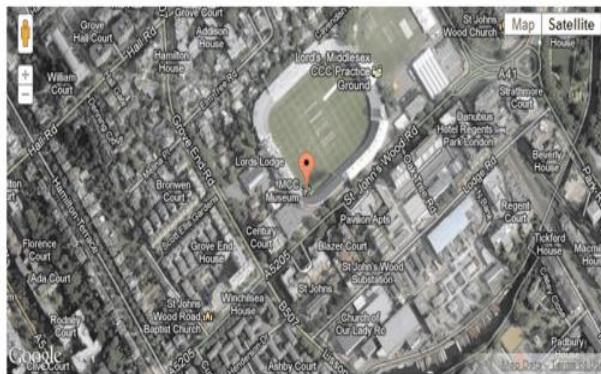
Property	Description
zoom	Sets the initial resolution at which map is displayed. A lower zoom value 0 represents a full map of the Earth. Similarly, a higher zoom value displays a map with high resolution.
center	Centers the map on a specific point by creating an object of type <code>LatLng</code> which holds the location coordinates.
mapTypeId	Sets an initial map type. The map types supported are: ROADMAP for normal, SATELLITE for photographic tiles, HYBRID for roads and city names, and TERRAIN for water features.

Designing Modernistic Websites © Aptech Limited

26

## Google Maps API 5-5

- Following figure displays the object on the Web page that is centered on Lord's Cricket Ground in London:



Designing Modernistic Websites © Aptech Limited

27

### Instructions to the Trainer(s):

- Using Slides 23 to 27, explain the Google Maps API.
- The Google Maps API is used to display locations on a map, based on the values of their coordinates - latitude and longitude. The Google Maps API must be configured in JavaScript, before it can be referenced further on the page. It contains a Map object which is instantiated and displayed on a Web page.
- Explain the syntax that shows the configuration of Google Maps API in JavaScript.
- Explain the code snippet which demonstrates how to load and initialize the Google Maps API in the <script> tag. The code will execute after the page is loaded completely and will invoke a function in response to the onload event.
- In the code, the URL `http://maps.google.com/maps/api/js?sensor=false` defines symbols and definitions to be loaded for the Google Maps API. Then, the function `initialize()` is invoked after the page is loaded completely. This function creates the object of type Map and initializes it with the map initialization variables.
- In the function, `var myOptions = {};`, is an object of type options that contains properties, such as `zoom`, `center`, and `mapTypeId`. These properties are used to initialize the map.
- Then, the statement, `new google.maps.Map (document.getElementById ("map_canvas"), myOptions);` creates an instance of Map object. The object is displayed in a container on the Web page specified with the <div> element.
- Figure on Slide 27 displays the Map object on the Web page that is centered on **Lord's Cricket Ground in London**.

### In-Class Question:

**Question:** Which of the following properties are used to initialize the map?

**Answer:** `zoom`, `center`, and `mapTypeId`

## Tracking User's Location 1-3

- The Geolocation object is used by the Google Maps API to display the geolocation information in the applications.
- The Code Snippet demonstrates the code that displays current location of a user on the map using Geolocation object.

```
<!DOCTYPE html>
<html lang="en">
<head>
<style>
    html, body {
        width: 100%;
        height: 100%;
        padding: 10%
    }
    #map_canvas {
        height: 50%;
        width: 50%;
    }
</style>
<script src="http://maps.google.com/maps/api/js?sensor=false">
</script>
<script>
    // Check support for Geolocation in the browser
    if (navigator.geolocation) {
        // Locate position and invoke function
        navigator.geolocation.getCurrentPosition(displayPosition, errorFunction);
    }
    else {
        alert('Geolocation is not enabled in your browser');
    }
</script>
```

Designing Modernistic Websites © Aptech Limited

28

## Tracking User's Location 2-3

```
// Creates the Map object
var map = new
google.maps.Map(document.getElementById("map_canvas"), myOptions);
// Displays icon on the located position
var marker = new google.maps.Marker({
    position: latlng,
    map: map,
    title:"User location"
});
// Error callback function
function errorFunction(pos) {
    alert('Error!');
}
</script> </head>
<body>
    <div id="map_canvas"></div>
    <div id="user_location"></div>
</body> </html>
```

```
// Success function
function displayPosition(position) {
    var my_lat = position.coords.latitude;
    var my_lng = position.coords.longitude;
    var div_info =
document.getElementById('user_location');
    div_info.innerHTML = '<h1>Latitude is : ' +
my_lat + ' and Longitude is : ' + my_lng + '</h1>';
    // Load Google Maps
    var latlng = new google.maps.LatLng(my_lat,
my_lng);
    var myOptions = {
        zoom: 2, //the initial resolution is set at
        // which map is displayed
        center: latlng, //centers the map
        mapTypeId: google.maps.MapTypeId.ROADMAP //sets
        // the map type
    };
}
```

Designing Modernistic Websites © Aptech Limited

29

## Tracking User's Location 3-3

- The code uses the `getCurrentPosition()` method and retrieves the current position of the user.
- Then, it passes the information to `displayPosition()` function, which retrieves the coordinates, latitude and longitude.
- The retrieved coordinates are set into the properties of the `Options` object named `myOptions` and initialize the `Map` object.
- Finally, the `Map` object is displayed along with the current position information in the `<div>` element.
- Following figure shows the output displaying the current location of the user on the Google Maps:



Latitude is :19.017656 and Longitude is 72.856178

Designing Modernistic Websites © Aptech Limited

30

### Instructions to the Trainer(s):

- Using Slides 28 to 30, explain the code to track user's location.
- The `Geolocation` object is used by the Google Maps API to display the geolocation information in the applications.
- Explain the code snippet which demonstrates the code that displays current location of a user on the map using `Geolocation` object.
- The code uses the `getCurrentPosition()` method and retrieves the current position of the user. Then, it passes the information to `displayPosition()` function, which retrieves the coordinates namely, latitude and longitude. The retrieved coordinates are set into the properties of the `options` object named `myOptions` and initialize the `Map` object.
- Finally, the `Map` object is displayed along with the current position information in the `<div>` element. Figure shows the output displaying the current location of the user on the Google Maps.

## Drag and Drop

HTML5 defines drag-and-drop operations that are based on events. Currently, drag-and-drop operations are supported by all major browsers.

The event-based mechanism allows the elements to be copied, reordered, or deleted on a Web page.

The drag-and-drop operation involves the use of a pointing device, such as mouse on a visual medium.

To perform the drag operation, a mousedown event is triggered followed by multiple mousemove events.

Similarly, the drop operation is performed when a user releases the mouse.

The benefit of drag-and-drop mechanism is that it has brought the drag-and-drop operations on the browser level.

This makes programming easier, thus eliminating the need of complex JavaScript code written in earlier HTML versions.

Designing Modernistic Websites © Aptech Limited

31

### Instructions to the Trainer(s):

- Using Slide 31, explain drag-and-drop operations.
- HTML5 defines drag-and-drop operations that are based on events. The event-based mechanism allows the elements to be copied, reordered, or deleted on a Web page.
- The drag-and-drop operation involves the use of a pointing device, such as mouse on a visual medium.
- To perform the drag operation, a mousedown event is triggered followed by multiple mousemove events.
- Similarly, the drop operation is performed when a user releases the mouse.
- Then, explain the benefits of drag-and-drop operations listed on the slide. Currently, drag-and-drop operations are supported by all major browsers.

## Drag Operation

- The steps required to make any element draggable on a Web page are as follows:

1. Set the `draggable` attribute of an element to be dragged.

2. Set an `ondragstart` event on the element which stores the data being dragged.

3. Store the data into the `DataTransfer` object.

- The Code Snippet shows how to set the `draggable` attribute of an image element.

```
<!DOCTYPE html>
<html>
  <head>
    <title>Drag and Drop API</title>
  </head>
  <body>
    <div id="div" style="border: red 2px solid; height:125px;
      width:75px; padding: 10px">
      
    </div>
  </body> </html>
```

Designing Modernistic Websites © Aptech Limited

32

### Instructions to the Trainer(s):

- Using Slide 32, explain the drag operation.
- The steps required to make any element draggable on a Web page are as follows:
  - Set the `draggable` attribute of an element to be dragged
  - Set an `ondragstart` event on the element which stores the data being dragged
  - Store the data into the `DataTransfer` object
- Explain the code snippet which shows how to set the `draggable` attribute of an image element.
- In the code, the `<img>` element contains `draggable` attribute that is set to `true`. The value `true` indicates that the element is eligible for dragging.

## Drag Events

- During various stages of the drag-and-drop operation, a number of events are fired.
- These events are mouse-based events.
- Following table lists various events triggered during the drag operation:

Event	Description
dragstart	Triggers when an element is started to be dragged by the user.
drag	Triggers when an element is being dragged using a mouse.
dragleave	Triggers when the drag and drop operation is completed.

Designing Modernistic Websites © Aptech Limited

33

### Instructions to the Trainer(s):

- Using Slide 33, explain drag events.
- During various stages of the drag-and-drop operation, a number of events are fired. These events are mouse-based events.
- Explain various events triggered during the drag operation.

## DataTransfer Object 1-3

- The `dataTransfer` object reveals the **drag data store** that contains the dragged data in the drag-and-drop operation.
- It allows getting and setting of the data being dragged.
- In other words, the `dataTransfer` object holds the data during drag-and-drop operation.
- The `dataTransfer` Object enables to define two types of information.
- These are as follows:
  - The data type of the draggable element
  - The value of the data being stored in the data store

Designing Modernistic Websites © Aptech Limited

34

## DataTransfer Object 2-3

- The Code Snippet demonstrates how to associate an element with `dragstart` event to store the data being dragged.

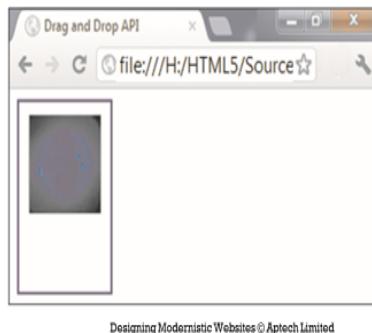
```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Drag and Drop API</title>
    <script>
      function drag_image(event)
      {
        event.dataTransfer.setData("image", event.target.id);
      }
    </script>  </head>
  <body>
    <div id="div1" style="border: blue 2px solid; height:125px;
width:75px; padding: 10px">
      
    </div>
  </body>
</html>
```

Designing Modernistic Websites © Aptech Limited

35

## DataTransfer Object 3-3

- In the code, the `<img>` element has been set with an event listener for the `dragstart` event.
- When the image is dragged, then the `dragstart` event is fired and calls `drag_image()` function.
- The function uses the `dataTransfer` object to store the data during drag-and-drop operation.
- The string 'image' represents the data type and `event.target.id` represents the value of `id` attribute of the draggable element.
- Following figure shows the output of the image element to be dragged:



36

### Instructions to the Trainer(s):

- Using Slides 34 to 36, explain the `dataTransfer` object.
- The `dataTransfer` object reveals the drag data store that contains the dragged data in the drag-and-drop operation.
- It allows getting and setting of the data being dragged.
- In other words, the `dataTransfer` object holds the data during drag-and-drop operation.
- The `dataTransfer` object enables to define two types of information. These are as follows:
  - The data type of the draggable element
  - The value of the data being stored in the data store
- Explain the code snippet which demonstrates how to associate an element with `dragstart` event to store the data being dragged.

### In-Class Question:

**Question:** What is the `dataTransfer` object used for?

**Answer:** The `dataTransfer` object is used to hold the data that is being dragged during a drag- and-drop operation.

## Drop Operation

After the element has been set up for dragging, it can be dropped on some element on the Web page.

By default, elements on the page are not set up to receive dragged elements.

Thus, the behavior of element acting as a drop element must be changed.

This can be done by creating event listeners for the drop element.

The drop element is also referred to as target element.

### Instructions to the Trainer(s):

- Using Slide 37, explain the drop operation.
- After the element has been set up for dragging, it can be dropped on some element on the Web page.
- By default, elements on the page are not set up to receive dragged elements.
- Thus, the behavior of element acting as a drop element must be changed.
- This can be done by creating event listeners for the drop element.
- The drop element is also referred to as target element.

## Drop Events 1-4

- For any element to receive the drop operation, it must be associated with the drop events.
- Following table lists the events of the drop operation:

Event	Description
dragenter	Triggers when a draggable element is being dragged on the target element for the first time.
dragleave	Triggers when an element is dragged outside the target element.
dragover	Triggers when an element is dragged inside the target element.
drop	Triggers when an element is dropped in the target element.

Designing Modernistic Websites © Aptech Limited

38

## Drop Events 2-4

The Code Snippet demonstrates how to set up event listeners to drop the image element on the target element.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Drag and Drop API</title>
    <script>
        function drag_image(event)  {
            event.dataTransfer.setData("image", event.target.id);
        }
        function allow_drop(event)
        {
            event.preventDefault();
        }
        function drop_image(event)  {
            var data=event.dataTransfer.getData("image");
            event.target.appendChild(document.getElementById(data));
        }
    </script> </head>
<body>
    <div id="div1" style="border: blue 2px solid; height:125px;
        width:75px; padding: 10px">
        
    </div>
    <br/>
    <div id="div2" style="border: red 2px solid; height:125px;
        width:75px; padding: 10px" ondrop="drop_image(event)"
        ondragover="allow_drop(event)">
    </div>
</body>
</html>
```

Designing Modernistic Websites © Aptech Limited

39

## Drop Events 3-4

- In the code, the `<div>` element with id attribute, set as 'div2', is associated with two event listeners namely, `ondragover` and `ondrop`.
- The `ondropover` calls the `allow_drop()` function which prevents the default behavior of the target element.
- By default, browsers do not support dropping of one elements on the other element.
- To prevent the default behavior, the statement, `event.preventDefault()` is invoked.
- Then, the `drop` event is fired on the target element.
- It calls the function `drop_image()` which uses `getData()` method to retrieve image that is set as 'image'.
- Finally, it appends the dragged image as a element into the target element, `div2`.

Designing Modernistic Websites © Aptech Limited

40

## Drop Events 4-4

- Following figure shows the output of the drop operation, after the image is dragged on the target element:



Designing Modernistic Websites © Aptech Limited

41

### Instructions to the Trainer(s):

- Using Slides 38 to 41, explain the concept of drop-events.
- Consider a situation where a user is travelling outside the coverage area of Internet Service Provider (ISP). In this case, the user will not be able to access Web applications due to the network connection failure.

- HTML5 supports offline Web applications that allow a user to work with them without being online. Offline Web applications work by saving all the Web pages locally on the user's system. This feature is known as the Application Cache.
- The Application Cache enables all resources, such as HTML, JavaScript, images, and CSS pages of a Web application to be stored locally on the system.
- Following are the steps that can be taken to cache resources locally on the system:
  - Create a manifest file to define the resources that must be saved.
  - Reference the manifest file in each Web page designed to use cached resources.

## Offline Web Applications API

- HTML5 supports offline Web applications that allow a user to work with them without being online.
- The offline Web applications works by saving all the Web pages locally on the user's system.
- This concept is also known as **Application Cache**.
- The **Application Cache** enables all resources, such as HTML, JavaScript, images, and CSS pages of an Web application to be stored locally on the system.
- Following are the steps that can be taken to cache resources locally on the system:
  1. Create a manifest file to define the resources that need to be saved.
  2. Reference the manifest file in each Web page designed to use cached resources.

Designing Modernistic Websites © Aptech Limited

42

### Instructions to the Trainer(s):

- Using Slide 42, explain Offline Web Applications API.
- HTML5 supports offline Web applications that allow a user to work with them without being online.
- The offline Web applications works by saving all the Web pages locally on the user's system.
- This concept is also known as Application Cache.
- The Application Cache enables all resources, such as HTML, JavaScript, images, and CSS pages of a Web application to be stored locally on the system.
- Steps that are taken to cache resources locally on the system:
  - Create a manifest file to define the resources that must be saved.
  - Reference the manifest file in each Web page designed to use cached resources.

## Creating a Manifest File 1-2

- The manifest file is a text file that defines the caching behavior for resources used by the Web page.
- The file should be saved with the .manifest extension.
- The Code Snippet demonstrates creation of a manifest file.

### CACHE:

```
# Defines resources to be cached.  
check.js  
styles.css  
images/figure1.jpg
```

### FALLBACK:

```
# Defines resources to be used if non-cached resources cannot be  
# downloaded  
Other_images/ figure2.png
```

### NETWORK:

```
# Defines resources that will not be cached.  
figure3.png
```

## Creating a Manifest File 2-2

- Following are the sections defined in the .manifest file:

### CACHE

- This section defines resources, such as check.js, styles.css, and figure1.png to be stored locally.

### FALLBACK

- This section defines alternative resource to be used, when the actual resource is not available.

### NETWORK

- This section specifies resources to be accessed when there is a network connection. Resources in this section are not cached.

**Instructions to the Trainer(s):**

- Using Slides 43 and 44, explain how to create the manifest file.
- The manifest file is a text file that defines the caching behavior for resources used by the Web page. The file should be saved with the .manifest extension.
- Explain the code snippet which demonstrates how to create a manifest file. Following are the sections defined in the .manifest file:
  - **CACHE:** This section defines resources, such as check.js, styles.css, and figure1.png to be stored locally.
  - **FALLBACK:** This section defines alternative resource to be used, when the actual resource is not available. For example, figure2.png is defined as a fallback image. If a browser cannot access figure1.jpg in the images folder, then figure2.png will replace the unavailable image at the time of rendering the markup on the Web page. The unavailability of the image can be due to network connection or server problem.
  - **NETWORK:** This section specifies resources to be accessed when there is a network connection. Resources in this section are not cached.

## Declaring a Manifest 1-3

- To associate a manifest with a Web page, assign `.manifest` file to the attribute named `manifest` specified with the `html` element.
- The Code Snippet demonstrates how to add the `.manifest` file in an HTML document.

```
<!doctype html>
<html manifest="appcache.manifest">
  <head>
    <title> Web Page </title>
    <link rel="stylesheet" href="styles.css"/>
    <script type="text/javascript" src="check.js"></script>
  </head>
  <body>
    <input type="button" value="click Here..." onClick="display()"/>
    
  </body>
</html>
```

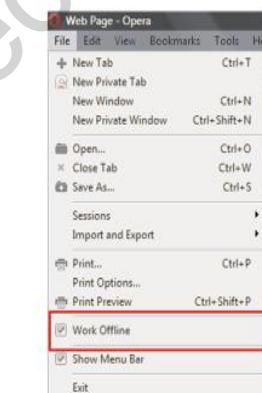
- In the code, the "appcache.manifest" is specified with the `<html>` tag.
- The interpretation of the manifest file is similar to any other file reference.
- The document uses a relative file path, as both the manifest file and HTML document are located in the same directory.
- By default, a Web page declaring manifest is cached automatically.

Designing Modernistic Websites © Aptech Limited

45

## Declaring a Manifest 2-3

- The benefit of Application Cache is that it improves performance of a Web page by reducing the number of requests made to the Web server.
- The Web server hosts the Web application to be accessed on the network.
- Following figure shows how to enable the **Work Offline** mode in the Opera browser:



Designing Modernistic Websites © Aptech Limited

46

## Declaring a Manifest 3-3

- Following figure shows the cached Web page in the Opera browser:



Designing Modernistic Websites © Aptech Limited

47

### Instructions to the Trainer(s):

- Using Slides 45 to 47, explain declaring a manifest.
- To associate a manifest with a Web page, assign `.manifest` file to the attribute named `manifest` specified with the `html` element.
- Explain the code snippet which demonstrates how to add the `.manifest` file in an HTML document.
- In the code, the “`appcache.manifest`” is specified with the `<html>` tag.
- The interpretation of the manifest file is similar to any other file reference.
- The document uses a relative file path, as both the manifest file and HTML document are located in the same directory.
- By default, a Web page declaring manifest is cached automatically.
- Explain the benefits of the Application Cache listed on Slide 46.
- Figure on Slide 46 shows how to enable the Work Offline mode in the Opera browser. This enables to cache the resources of the Web application pages locally.

## Summary

- ❖ Geolocation determines the current location of a user on devices.
- ❖ The location is represented as a single point on a map that comprises two components: latitude and longitude.
- ❖ The Geolocation API is a specification provided by the W3C which provides a consistent way to develop location-aware Web applications.
- ❖ Google Maps API is used to display the user's location on the map.
- ❖ The object of type Map is created in JavaScript, before it can be referenced in an HTML document.
- ❖ The drag-and-drop operations defines an event-based mechanism using which elements on a Web page can be copied, reordered, or deleted.
- ❖ HTML5 supports offline Web applications that allow a user to work with them without being online.

48

**Instructions to the Trainer(s):**

- Show students Slide 48.
- Summarize the session by reading out each point on the Slide.

# Session 15: Building a Mobile Web Application

## 15.1 Pre-Class Activities

Before you commence the session, you should familiarize yourself with the topics of this session in-depth. Prepare a question or two that will be a key point to relate the current session objectives.

### 15.1.1 Teaching Skills

To teach this session, you should be well-versed with different categories and platforms for mobile devices. Along with this, you should prepare yourself about the architectural and design requirements of mobile devices and use of HTML5 to design Web pages for mobiles. Also, explain the support of HTML5 for mobile website and best practices for optimization of mobile Website.

You should teach the concepts in the theory class using the images provided. For teaching in the class, you are expected to use slides and LCD projectors.

## In-Class Activities

Follow the order given here during In-Class activities.

Slide 2



### Instructions to the Trainer(s):

Give students a brief overview of the current session through the session objectives listed in Slide 2.

## 15.2 In-Class Explanations

Slide 3

**Mobile Application Environment**

- Today, access to the Web is not limited to only desktop systems, but is also available on portable and wireless devices, such as mobile devices.

A mobile device, also known as a handheld device, is a small portable computing device with a small display screen and keyboard.

A mobile device has an operating system on which various types of application software are executed.

These application software are also known as apps.

The most commonly used apps are mobile browsers that display the Web pages.

Designing Modernistic Websites © Aptech Limited

### Instructions to the Trainer(s):

- Using Slide 3, explain mobile application environment.
- Today, mobile devices such as smartphones and tablets are equipped with browsers and network access that provide a better Web experience to their users.
- As the number of mobile users has increased, so the necessity for a mobile Web experience has increased, that is identical to a desktop user experience.
- In other words, mobile users now look for applications targeting their mobiles that are similar to the ones on their desktops. This has led to the emergence of mobile Web application development.
- Different companies follow distinctive strategies for developing mobile applications depending on the requirement of their users.
- Then, explain the students about mobile devices and their apps as mentioned on Slide 3.

## Types of Mobile Devices 1-5

- Different categories of mobile phones available in the market are as follows:

### ➤ Basic Mobile Devices

- Very basic models with only call and Short Message Service (SMS) facility.
- Do not provide support for Web browsers or network access.

### ➤ Low-end Mobile Devices

- Provide more features than a basic mobile device, typically Web support
- Preferred by users who do not need heavy Internet usage.
- Include a basic camera and a basic music player.
- Manufacturers, such as Nokia, Motorola, Sony Ericsson, Samsung, and so forth have gained popularity for offering low cost handsets in the global market.



Designing Modernistic Websites © Aptech Limited

4

## Types of Mobile Devices 2-5

### ➤ Mid-end Mobile Devices

- These types of mobile devices have gained popularity due to their increased user experience and moderate cost.
- Some key features of these devices include: medium sized-screen, HTML supported browser, a decent camera, games, and support for applications.
- They have a proprietary Operating System (OS) that is not well-known and is also not portable across various platforms.



Designing Modernistic Websites © Aptech Limited

5

## Types of Mobile Devices 3-5

### ➤ High-end Mobile Devices

- These types of mobile devices have advanced features, such as an accelerometer, advanced camera features, and Bluetooth.
- They have a better look and feel as compared to mid-end mobile devices.



Designing Modernistic Websites © Aptech Limited

6

## Types of Mobile Devices 4-5

### ➤ Smartphones

- These are mobile devices with multitasking capabilities.
- These devices have a full browser support similar to desktop browsers with wireless LAN and 3G connection.
- They have several advanced features that are as follows:
  - Digital Compass
  - Global Positioning System (GPS)
  - Touch screen
  - Camera with video recording
  - TV out
  - Bluetooth
  - Accelerometer



Designing Modernistic Websites © Aptech Limited

7

## Types of Mobile Devices 5-5

### ➤ Tablets and Notebooks

- These devices are larger than mobile phones.
- They are mobile computers with a touch screen virtual keyboard and stylus or digital pen.
- Features of tablets include: multi-touch display, better user experience, high quality screen resolution, better Web support, and multitasking OS with high speed.
- Some of the tablets available in the market are Lenovo Tablet PC, Samsung Galaxy Tab, and HCL Me Tab.
- Following figure shows a tablet device:



Designing Modernistic Websites © Aptech Limited

8

### Instructions to the Trainer(s):

- Using Slides 4 to 8, explain different types of mobile devices.
- Explain different types of mobile device models available in the market based on their features, manufacturers, and cost.
- With the advent of mobile devices, the format and designing of Websites is changing keeping users in the mind.
- Using Slide 7 and 8, tell the students that the main concern of the Web designer should be the seamless access to the Web pages on the smartphones and tablet devices.

## Mobile Platforms 1-2

A mobile device platform is similar to a software platform.

It is basically responsible to interact with the device hardware and run software/services on the mobile device.

The mobile platforms are categorized as proprietary and open source.

Proprietary platforms are those which are designed and developed by the mobile device manufacturers.

These platforms are developed for specific devices and are not supported on all platforms.

Open source platforms are those which are freely available to the users.

The users can download the source code and alter them as per their requirements.

## Mobile Platforms 2-2

- Brief description of the platforms available on mobile devices is as follows:

### iOS

It is a mobile OS developed by Apple Inc. and was initially referred to as iPhone OS. It is derived from Mac OS X, which is based on the UNIX platform. It was originally developed for iPhone and iPadTouch, but later extended to support other devices, such as iPad and Apple TV. iOS cannot be installed on non-Apple platforms.

### Windows Mobile

It is a mobile OS that runs on top of the Windows Mobile platform.

### Android

Android is an open source OS developed by Google. It is currently used by smartphones and tablet computers. It can be customized by device manufacturers and community developers to extend the functionality of the devices.

**Instructions to the Trainer(s):**

- Using Slides 9 and 10, explain different types of mobile platforms and their features.
- Using Slide 9, tell the students that a mobile device platform is similar to software platform, and is responsible to interact and run with services on mobile devices.
- Using Slide 10, list and explain different platforms available for mobile devices. These are as follows:
  - iOS
  - Windows Mobile
  - Android
- For more information on mobile platform, refer to  
<http://mobileplatforms.wikidot.com/>

**In-Class Question:**

**Question:** What are the features of Android OS?

**Answer:** Android is a free and open-source OS, developed by Google. It can be customized by device manufacturers and is currently used by many smartphones and tablet devices.

## Design Aspects of Mobile Website 1-2

- Some of the basic considerations needed for designing a Websites for intended mobile device are as follows:
  - Resolution and Physical Dimension
  - Page Orientation
  - Input methods

### ➤ Resolution and Physical Dimension

The resolution means the number of pixels (width and height) on the screen of the mobile device.

The resolution of mobile devices is measured in terms of the physical dimensions of the screen.

The screen dimensions are either measured diagonally in terms of inches/centimeters or in terms of width and height.

The relation between the physical dimension and resolution is termed as Pixels per Inch (PPI) or Dots per Inch (DPI).

The higher DPI results in good print-quality graphics on the mobile device.

## Design Aspects of Mobile Website 2-2

### ➤ Page Orientation

- The mobile devices are also categorized based on their orientation, vertical and horizontal.
- The vertical orientation devices are also referred to as portrait devices with taller display.
- Similarly, the horizontal orientation devices are referred as landscape devices with wider display.

### ➤ Input Methods

- Numeric keypad
- Alphanumeric keypad (Simple or QWERTY)
- Virtual keypad on screen
- Multi-touch
- External keypad
- Voice and handwriting recognition

**Instructions to the Trainer(s):**

- Using Slides 11 and 12, explain the design aspects of mobile Website.
- The design process of a mobile Website is similar to that of a traditional Website developed for desktop Web browsers. Still, there are some differences between them that must be taken into consideration.
- An ideal mobile Website is supported as well as rendered properly by maximum possible browsers and operating system.
- Some of the basic considerations required for designing Websites for intended mobile device are as follows:
  - **Resolution and Physical Dimension** - The resolution means the number of pixels (width and height) on the screen of the mobile device. As there are no standards defined for screen resolution, it varies depending on its model and manufacturer.
  - **Page Orientation** - The mobile devices are also categorized based on their orientation namely, vertical and horizontal. The vertical orientation devices are also referred to as portrait devices with taller display. Similarly, the horizontal orientation devices are referred as landscape devices with wider display.
  - **Input Methods** - There are different methods to input data on the mobile devices. A mobile device can support more than one input method.

**In-Class Question:**

**Question:** Which device takes care of the screen rotation while changing the view from landscape to portrait and vice-versa?

**Answer:** Hardware accelerators availability on the phones.

## Architectural Aspects 1-5

- The Website developed for a mobile device is a collection of Web pages.
- Thus, it is essential to understand a few architectural concepts that can help to create meaningful mobile services.
- Some of the concepts that relate to its architecture are as follows:

### ➤ Navigation

- Navigation is the path followed by a user to travel in a Website.
- As compared to the navigation tree of a desktop site, almost 80% of the information of a desktop site will not be useful to a mobile Website.
- Thus, the main focus should be on 20%.

Design Web pages based on the use cases.

Arrange Web pages depending on the frequent requirements of the mobile users.

Restrict the depth of a mobile page to three clicks for a specific use case.

## Architectural Aspects 2-5

Design minimum input controls for the form pages.

Desktop Website normally has a welcome screen. In case of mobile Websites, avoid developing welcome screens.

While designing a service, decide its usability.

Approximate the number of mobile pages required to separate services, after the home page.

### ➤ Perspective

- The perspective of a mobile user is different from a desktop user in terms of needs and accessibility.
- Hence, a user-centric design approach should be followed for designing mobile Websites.
- This ensures that a user completes the task easily and successfully.

## Architectural Aspects 3-5

Some of the possible users' contexts are as follows:

- What is the location of the user?
- Why is a mobile Website accessed by the user?
- What are the needs of the user?
- What solution is offered by a mobile application to solve the user's problem?
- Where is the user present while accessing a Website?

### ➤ Enhancement

- Enhancement is a simple and powerful technique that can be adopted while designing a mobile Website.
- This technique defines compatibility of Website and allows access to basic content, services, and functionality on all type of mobile devices.
- Also, it provides a better Web experience on devices with higher standards.

Designing Modernistic Websites © Aptech Limited

15

## Architectural Aspects 4-5

Some of the core principles for enhancing mobile Websites are as follows:

- Basic content and functionality are accessible in all browsers.
- Enhanced layout and behavior must be provided through external style sheets and JavaScript that are linked with the Web pages.
- Markup elements used on the pages must have proper semantic.
- Web browser settings on a user's device should be considered.

### ➤ Use of Web Standards

- The Web standards, such as HTML, CSS, and JavaScript followed in the mobile Website design must be correctly used.
- This increases the possibility of displaying pages on large number of devices.
- The well-formedness of the markup tags used on a page can be achieved by validating them.
- Also, the use of certain HTML elements can be avoided while designing the Web pages for mobile devices.

Designing Modernistic Websites © Aptech Limited

16

## Architectural Aspects 5-5

- The brief description of these elements is as follows:

### Use of HTML tables

- As the screen size of mobile devices is small, so the use of tables in layouts should be avoided.
- It makes the scrolling difficult and also slows down the page loading in the browser.

### Pop-up windows

- The Websites with pop-up windows makes the site impractical to work with.
- Also, all mobile browsers do not provide support for them.

### Use of graphics

- The use of graphics increases download time of the pages.
- Also, they can obstruct the layout of the old mobile browsers, resulting in incorrect display of the page.

### Use of frames

- Many mobile devices do not provide the support for frames due to usability problems.
- Also, the HTML5 new specification does not provide the support for frames.

Designing Modernistic Websites © Aptech Limited

17

### Instructions to the Trainer(s):

- Using Slides 13 to 17, explain the architectural aspects of developing mobile apps.
- The architecture for designing a mobile Website begins with the understanding of the information and service that is offered to the mobile user.
- Explain the concepts that relate to the mobile architecture, such as:
  - Navigation
  - Perspective
  - Enhancement
  - Use of Web Standards

## Setting Up the Environment 1-2

- Mobile Web applications are developed to be run on different mobile devices.
- Hence, they need to be tested in several different environments.
- The tools required to develop a mobile Web application are namely, Integrated Development Environment (IDE) and emulators.
- These are described as follows:

### ➤ IDE

- An IDE is a tool used for coding the markup, JavaScript, and CSS.
- Some of these tools are as follows:
  - Adobe Dreamweaver
  - Aptana Studio
  - Notepad++
  - Eclipse
  - Editplus (text editor)
- Latest versions of these tools provide better support for mobile markups.
- They also provide support for validating pages against mobile Web standards.

## Setting Up the Environment 2-2

### ➤ Emulators

- An emulator is a software that translates compiled code to native platform on which the application is executed.
- The emulator runs as a desktop application that allows testing and debugging of a mobile application.
- It offers the environment similar to a real mobile device on which an application will be executed.
- Emulators are developed by manufacturers and are often offered free to users.
- They are either standalone applications or bundled with a Software Development Kit (SDK) for native development.
- Some of the popular emulators are as follows:
  - Android
  - iOS
  - Windows Mobile
  - Opera Mobile

**Instructions to the Trainer(s):**

- Using Slides 18 and 19, explain the process of setting up environment for developing mobile apps.
- Desktop Web applications are created and tested in the environment for which they are developed.
- Mobile Web applications are developed to be run on different mobile devices. Hence, they must be tested in several different environments.
- The tools required to develop a mobile Web application are namely, Integrated Development Environment (IDE) and emulators.
- Explain that an IDE is a tool used for coding the markup, JavaScript, and CSS.
- Today, in the market, there are different tools which provide the facility to easily build a mobile Web application. Some of these tools are as follows:
  - Adobe Dreamweaver
  - Microsoft Expression Web
  - Aptana Studio
  - Eclipse
  - EditPlus (text editor)
- Latest versions of these tools provide better support for mobile markups. They also provide support for validating pages against mobile Web standards.
- Explain emulators using Slide 19. The testing of a mobile Web application can be done using an emulator. An emulator is software that translates the compiled code to the native platform on which the application is executed.
- Emulators are developed by manufacturers and are often offered free to users. They are either standalone applications or bundled with a Software Development Kit (SDK) for native development.
- Some of the popular emulators that either run as standalone applications or in an SDK are as follows:
  - Android
  - iOS
  - webOS
  - Blackberry
  - Windows Phone
  - Opera Mobile

**In-Class Question:**

**Question:** What is an emulator?

**Answer:** An emulator is software that translates the compiled code to the native platform on which the application is executed.

## HTML Support on Mobiles

- Today, majority of smartphones and tablets are providing good support for HTML5.
- Most Android and iOS mobile devices as well as tablets use browsers that are based on Webkit.
- The Webkit is a layout engine supported by browsers, such as Google Chrome and Apple Safari to render Web pages.

The features suited for mobile devices are as follows:

- Video
- Audio
- Drag and drop
- Accessing browser history
- Geolocation API for accessing location
- Web storage API to save data on mobile devices
- Offline Web applications (Applications with no Internet connection)

Designing Modernistic Websites © Aptech Limited

20

### Instructions to the Trainer(s):

- Using Slide 20, explain HTML5 support for mobile devices.
- Most Android and iOS mobile devices as well as tablets use browsers based on Webkit.
- Features for mobile devices are as follows:
  - Video
  - Audio
  - Drag and drop
  - Accessing browser history
  - Geolocation API
  - Web storage API
  - Offline Web Applications

### In-Class Question:

**Question:** Which layout engine is supported by Google chrome and Apple Safari to render Web pages?

**Answer:** Webkit layout engine

## HTML5 Markup

- The Web pages developed for a mobile Web application have the same structure as traditional Web pages.

A Web page contains following sections:

- Heading Structure
- Document Structure

Designing Modernistic Websites © Aptech Limited

21

**Instructions to the Trainer(s):**

- Using Slide 21, explain the HTML5 markup.
- The Web pages developed for a mobile Web application have the same structure as traditional Web pages.
- A Web page contains the following sections:
  - Heading Structure
  - Document Structure

## Heading Structure 1-7

- The heading structure is represented by a <head> element defined in an HTML Web page.
- It defines a <meta> tag that is used specifically for mobile browsers.
- The brief description for some of the tags defined under element is as follows:

### ➤ Viewport Meta Tag

- This is a new technique used to inform the browser that the Web page is optimized for a mobile device.
- A viewport is the rectangular display area on the screen, where the content of a Web page are displayed by the browser.
- It contains attributes, such as width and height that can be set to larger or smaller values depending on the total visible area on the screen.
- Following table lists the attributes of viewport meta tag:

Attribute	Description	Value
width	Defines the horizontal size of the viewport in pixels	Integer value (in pixels) or constant device-width
height	Defines the vertical size of the viewport in pixels	Integer value (in pixels) or constant device-height
initial-scale	Sets the scale of the page for its initial display	Floating value between 0.1 to n

Designing Modernistic Websites © Aptech Limited

22

## Heading Structure 2-7

Attribute	Description	Value
minimum-scale	Defines the minimum zoom scale of the viewport	Floating value between 0.1 to n
maximum-scale	Defines the maximum zoom scale of the viewport	Floating value between 0.1 to n
user-scalable	Allows scaling of application on the mobile devices. That is, users can zoom in and out in the application	no or yes

- The Code Snippet demonstrates the viewport meta tag to set device width for a mobile Web page.

```
<!DOCTYPE html>
<head>
<title>Mobile</title>
<!-- <meta name="viewport" content="width=device-width,
      user-scalable=no"> -->
</head>
<body>
<header> Mobile Design </header>
<NAV>
<a href="home.html">Home</a> | <a href="aboutUs.html">
About Us</a> |
<a href="contactUs.html">Contact Us</a>
</NAV>
<section id="intro">
```

Designing Modernistic Websites © Aptech Limited

23

## Heading Structure 3-7

```
<p>This is the introductory text to my mobile Web  
application.  
</p>  
<p>Mobile development is more than cross-browser, it  
should be cross-platform. The vast number of mobile  
devices makes thorough testing a practical  
impossibility, leaving developers nostalgic for the  
days when they only had to support legacy browsers.  
</p>  
</section>  
</body>  
</html>
```

- In the code, viewport width has been set to "device-width" which sets device width to 320px.
- As all phones does not support the same width, so setting "device-width" allows the mobile browsers to set the width according to the device width.
- Also, setting the attribute user-scalable=no prevents the user from increasing the display scale of the application.
- The default width taken for the viewport is 980px which is approximately the desktop size.
- Users can view the mobile version of a Website by using Chrome Devtools feature of Google Chrome browser.

Designing Modernistic Websites © Aptech Limited

24

## Heading Structure 4-7

- Steps to view the mobile version of a Website on Chrome are as follows:

1. Launch the Website in Google Chrome browser.
2. Open DevTools by pressing F12. You will see different tools available as shown in figure:

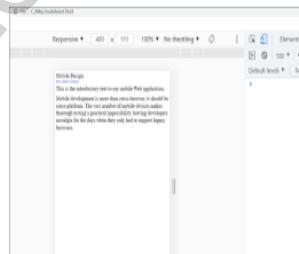


3. Click the **Toggle Device toolbar** icon present in the Tools list. Refer to figure:



Toggle Device Toolbar

The icon will turn blue when the device mode is turned ON. At the same time, the display of the Website on the left will change similar to how it would be on a mobile device.

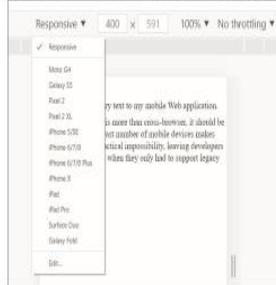


Designing Modernistic Websites © Aptech Limited

25

## Heading Structure 5-7

4. Click the Responsive drop-down on the left of the screen to choose a mobile device to simulate as shown in figure:



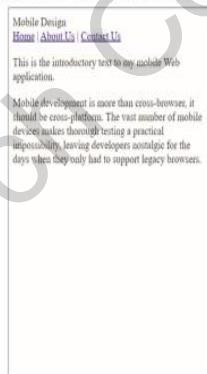
5. Choose a device, say Pixel 2. The content will be accordingly rendered on the screen. Figure shows the output of Code Snippet 1 without a viewport tag, as rendered on a Google Pixel 2 mobile.



26

## Heading Structure 6-7

Following figure displays the Web page after removing the comments from Code Snippet 1 and setting the viewport meta tag:



27

## Heading Structure 7-7

### ➤ Title Tag

- Apart from `<meta>` tag, `<head>` element also contains a `<title>` tag.
- The text selected for `<title>` tag should be meaningful, short, and precise.
- It should be between four and eight words, as some old mobile devices truncate the long titles after 10 or 12 words.

### ➤ Icons

- To add icons to a mobile Web page, images in or format could be used.
- These formats are compatible with mobile devices, as they are easy to export and are optimized in size.
- For example,  
`<link rel="icon" type="image/png" href="mobile.png" />`
- From HTML5 onwards, Android supports the `apple-touch-icon-precomposed` meta tag in order to display high-resolution icons.

Designing Modernistic Websites © Apech Limited

28

### Instructions to the Trainer(s):

- Using Slides 22 to 28, explain the heading structure.
- The heading structure is represented by a `<head>` element defined in an HTML Web page. It provides information about the Web page such as keywords and language used, which is not displayed on the Web page.
- Keywords are important terms existing in a Web page used by the search engines to identify the Web page with respect to the search criterion. It defines a `<meta>` tag that is used specifically for mobile browsers.
- A `<meta>` tag has two factors that must be considered while designing a mobile Web application are its initial display (zoom) scale and orientation. Thus, it is necessary to inform mobile browsers to consider these factors while displaying a Web page.
- A `<meta>` tag indicates that the document is optimized for mobile devices and is used to control the display scale, while displaying HTML content on the device. It is specific to mobile browsers.
- Similarly, a non-standard variation of `<meta>` tag is specified by giving an alternate `<link>` tag. This tag is mostly used with desktop Web pages and defines an alternative URL for displaying the same content on different media such as handheld devices.  
`<link rel="alternate" media="handheld" href="http://mysite.com" />`
- Then, explain viewport meta tag using Slide 22. It is supported on many smartphones, such as iPhone, Android based phones, and browsers such as Internet Explorer Mobile, Opera Mini, and Opera Mobile.

- A viewport is the rectangular display area on the screen, where the content of a Web page is displayed by the browser. It contains attributes, such as width and height that can be set to larger or smaller values depending on the total visible area on the screen.
- Explain the syntax of viewport tag: `<meta name="viewport" content="width=device-width, user-scalable=no">`. Then, explain the list of attributes of the viewport meta tag.
- Figure on Slide 26 displays the Web page on Opera Mobile Emulator, before setting the viewport meta tag.
- Figure on Slide 27 displays the Web page on Opera Mobile Emulator, after removing the comments from code snippet. The code sets the viewport meta tag.

## Document Structure 1-11

- The document structure is represented by a `<body>` element in the HTML Web page.
- The `<body>` element of a mobile Web application defines content displayed to the user.
- Some of the elements used in the `<body>` element of a mobile Web page are as follows:

### ➤ Layouts

- The HTML5 new tags that provide semantics for the layout of an HTML document are as follows:
  - `<article>`- An independent portion of the document or site
  - `<aside>`- Content that is tangential to the main part of the page or site
  - `<figcaption>`- Caption for a figure
  - `<figure>`- A figure or quotation pulled out of the flow of text
  - `<footer>`- The footer of a document or section
  - `<header>`- The header of a document or section
  - `<hgroup>`- A group of headings
  - `<nav>`- A navigation section
  - `<section>`- Identifies a block of content

Designing Modernistic Websites © Aptech Limited

29

## Document Structure 2-11

### ➤ Images

- Images can be used in mobile Web applications for pictorial representation.
- Almost all mobile browsers understand formats, such as GIF, JPEG, and PNG.
- The `<img>` tag is used to display image on a Web page.
- The attributes of `<img>` tag, such as `width`, `height`, and `alt` should be specified, as it reduces the rendering time of the image.
- The Code Snippet demonstrates a mobile Web page with an `<img>` tag.

```
<!DOCTYPE html>
<head>
<title>Images</title>
<meta name="viewport" content="width=device-width,
    user-scalable=no"/>
</head>
<body>
<article>
<h2> Gift Basket </h2>

</article>
</body>
</html>
```

Designing Modernistic Websites © Aptech Limited

30

## Document Structure 3-11

- Following figure displays image on a mobile Web page in Chrome Dev Tools:



Designing Modernistic Websites © Aptech Limited

31

## Document Structure 4-11

### ➤ Lists

- A mobile Web application supports different types of lists.
- Various list types are as follows:

#### Ordered lists

- Used for navigational menus and are defined using <ol> tag on a Web page.

#### Unordered lists

- Used for presenting objects of same type and are defined using <ul> tag on a Web page.

#### Definition lists

- Used for presenting information as key/value pairs and are defined using <dl> tag on a Web page.

Designing Modernistic Websites © Aptech Limited

32

## Document Structure 5-11

- The Code Snippet demonstrates the use of a definition list to present the capacity list of different hardware components on a mobile Web page.

```
<!DOCTYPE html>
<head>
    <title>List on Mobile</title>
    <meta name="viewport" content="width=device-width, user-scalable=no" />
    <link rel="apple-touch-icon-precomposed" href="mobile.png" />
</head>
<body>
    <header>
        <h5> Hardware Components Capacity List </h5>
    </header>
    <section>
        <dl>
            <dt>RAM Memory</dt>
            <dd>4.00GB</dd>
            <dt>Hard Disk</dt>
            <dd>500GB</dd>
            <dt>LAN</dt>
            <dd>Wifi, Bluetooth</dd>
            <dt>CPU</dt>
            <dd>2.93GHz</dd>
        </dl>
    </section>
</body>
</html>
```

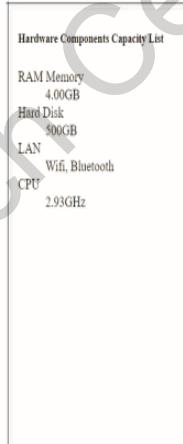
- In the code, the `<dl>` tag is used to represent the information as key and value pair format.

Designing Modernistic Websites © Aptech Limited

33

## Document Structure 6-11

- Following figure displays the definition list on a mobile Web page:



Designing Modernistic Websites © Aptech Limited

34

## Document Structure 7-11

### ➤ Links

- Hyperlinks are used to link pages in a Web application.
- A hyperlink is defined using `<a>` tag with `href` attribute.
- The `href` attribute is set to the URL of a resource.
- The `<a>` tag should also have `accesskey` attribute specified with it.
- The `accesskey` attribute is a keyboard shortcut and is useful for mobile devices that have support for access keys.

Designing Modernistic Websites © Aptech Limited

35

## Document Structure 8-11

### ➤ Links

- The Code Snippet demonstrates the use of `<a>` and `<ul>` tag to create a navigation list on a mobile Web page.

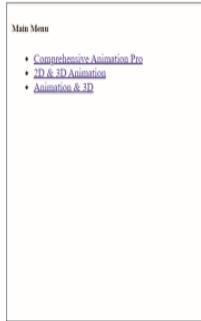
```
<!DOCTYPE html>
<head>
<title>Navigation list</title>
<meta name="viewport" content="width=device-width, user-scalable=no" />
<link rel="apple-touch-icon-precomposed" href="mobile.png" />
</head>
<body>
<header>
<h5> Main Menu </h5>
</header>
<NAV>
<ul>
<li><a title="Comprehensive Animation"
      href="comprehensive.html" accesskey="1">Comprehensive Animation Pro</a></li>
<li><a title="2D and 3D" href="animation_3d.html"
      accesskey="2">2D & 3D Animation </a></li>
<li><a title="3D Animation" href="animation_2d.html"
      accesskey="3">Animation & 3D</a></li>
</ul>
</NAV>
</body>
</html>
```

Designing Modernistic Websites © Aptech Limited

36

## Document Structure 9-11

- Following figure displays the navigation list on a mobile Web page:



### ➤ Tel Scheme

- As mobile devices are basically phones, hence, links can be created to perform phone call actions.
- This is achieved using the `tel:<phone number>` scheme embedded with a hyperlink.
- The `tel` scheme is useful in situations, such as accessing helpdesk systems or voicemail systems.

Designing Modernistic Websites © Aptech Limited

37

## Document Structure 10-11

- The Code Snippet demonstrates a `tel` scheme defined on a mobile Web page.

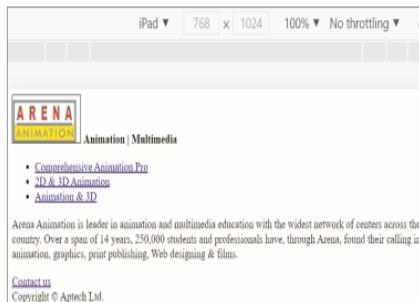
```
<!DOCTYPE html>
<head>
    <title> Mobile Application </title>
    <meta name="viewport" content="width=device-width, minimum-scale=1.0,maximum-scale=1.0"/>
</head>
<body>
    <HEADER>
        <SECTION> <b> Animation | Multimedia </b>
    </SECTION>
    <NAV><ul>
        <li><a title="Comprehensive Animation" href="#">Comprehensive Animation Pro</a></li>
        <li><a title="2D and 3D" href="#">2D & 3D Animation </a></li>
        <li><a title="3D Animation" href="#">Animation & 3D</a>
    </li>
    </ul>
    </NAV>
    </HEADER>
    <SECTION id="intro">
        <p> Arena Animation is leader in animation and multimedia education with the widest network of centers across the country. Over a span of 14 years, 250,000 students and professionals have, through Arena, found their calling in animation, graphics, print publishing, web designing & films.
    </p>
    </SECTION>
    <FOOTER>
        <p>
            <a href="tel:+91 22 2827 2300">Contact us </a> <br/>
            Copyright ©; 2012 Aptech Ltd.</p>
    </FOOTER>
</body>
</html>
```

Designing Modernistic Websites © Aptech Limited

38

## Document Structure 11-11

- Following figure displays the output on iPad tablet simulated in Google Chrome:



Designing Modernistic Websites © Aptech Limited

39

### Instructions to the Trainer(s):

- Using Slides 29 to 39, explain document structure.
- Using Slides 29 and 30, explain how to design layouts and images for mobiles.
- The document structure is represented by a `<body>` element in an HTML Web page. The `<body>` element of a mobile Web application defines the content that is displayed to the user. Some of the elements used in the `<body>` element of a mobile Web page are as follows:
  - Layouts:** Tell the students that in the earlier sessions, they have learnt about the HTML5 new tags that provide semantics for the layout of an HTML document. Describe the tags listed on Slide 29 which are used to design the layout of the mobile application.
  - Images:** After explaining the image formats, explain them that Images should not be used for setting background, buttons, links, or presenting titles on a mobile Web page. This is because images increase the number of request to the Web server and also load time of the Web page. The `<img>` tag is used to display image on a Web page. Also, the attributes of `<img>` tag such as width, height, and alt should be specified, as it reduces the rendering time of the image.
- Using Slide 32, explain the use of lists in the mobile Web page design.
- A mobile Web application supports different types of lists. Various list types are as follows:
  - Ordered lists** - Used for navigational menus and is defined using `<ol>` tag on a Web page.
  - Unordered lists** - Used for presenting objects of same type and are defined using `<ul>` tag on a Web page.

- **Definition lists** - Used for presenting information as key/value pairs and are defined using `<dl>` tag on a Web page.

- Using Slides 35 to 37, explain the use of links in the mobile Web page design.
- Hyperlinks are used to link pages in a Web application. They are one of the important elements on Web pages and this holds true in case of mobile Web applications too.
- A hyperlink is defined using `<a>` tag with `href` attribute. The `href` attribute is set to the URL of a resource.
- Apart from `href`, the `<a>` tag should also have `accesskey` attribute specified with it.
- The `accesskey` attribute is a keyboard shortcut and is useful for mobile devices that have support for access keys.
- `Tel` scheme as mobile devices are basically phones, hence, links can be created to perform phonecall actions. This is achieved using the `tel:<phone number>` scheme embedded with a hyperlink. The `tel` scheme is useful in situations, such as accessing helpdesk systems or voicemail systems.

**In-Class Question:**

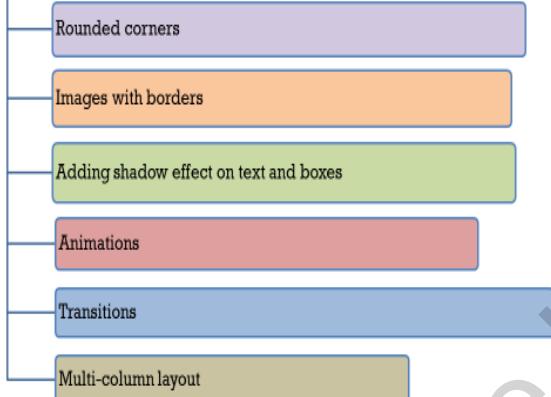
**Question:** Which attribute is useful for mobile devices that have support for access keys?

**Answer:** `<li>` tag is used to define the item in a list.

## CSS for Mobile 1-8

- CSS3 provides properties for adding colors, selectors, borders, backgrounds, and so on for effective appearance of a Web page.

Most modern mobile browsers support following features of CSS3:



Designing Modernistic Websites © Aptech Limited

40

## CSS for Mobile 2-8

- Modern browsers have provided new styles for CSS3 that are specific to each browser.
- To add these styles on a Web page, the relevant properties need to be prefixed with the browser specific keyword.
- The property prefixed with the keyword represents the browser on which it is supported.
- Following table lists the keywords with their supported browsers:

Keyword	Browser
-moz	Firefox
-ms	Internet Explorer
-o	Opera
-webkit	Google Chrome and Safari

Designing Modernistic Websites © Aptech Limited

41

## CSS for Mobile 3-8

- The Code Snippet demonstrates the CSS3 properties for a Web page that was developed in the code.

```
<!DOCTYPE html>
<head>
<title> Mobile Application </title>
<meta name="viewport" content="width=device-width"/>
<style>
html, body {
margin: 0;
padding: 0;
border: 0;
font-size: 100%;
font-weight: normal;
vertical-align: baseline;
background: transparent;
}
body {
line-height: inherit;
}
nav {
width:500px;
height:60px;
background-color:#A4A4FF;
margin:0px;
margin-top:5px;
padding:0px;
}
```

Designing Modernistic Websites © Aptech Limited

42

## CSS for Mobile 4-8

```
ul#navigation {
margin:0px;
border-left:1px solid #c4bbe7;
border-right:1px solid #c4dbe7;
padding-top:5px;
}
ul#navigation li {
display:inline;
font-size:12px;
font-weight:bold;
margin:0;
padding:0;
float:left;
position:relative;
border-top:1px solid #c4dbe7;
border-bottom:2px solid #c4dbe7;
}
ul#navigation li a {
padding:0px;
color:#1E1E1E;
text-shadow:1px 1px 0px #fff;
text-decoration:none;
display:inline-block;
border-top:1px solid #fff;
background: #f6f6f5;
-webkit-transition:color 0.2s linear, background 0.2s linear;
-moz-transition:color 0.2s linear, background 0.2s linear;
-o-transition:color 0.2s linear, background 0.2s linear;
transition:color 0.2s linear, background 0.2s linear;
}
```

```
ul#navigation li a:hover {
background:#f8f8f8;
color:red;
}
p {
border: 0;
font-size: 100%;
font-weight: normal;
vertical-align: baseline;
background: transparent;
}
a {
margin: 0;
padding: 0;
font-weight: normal;
}
#intro {
background-color: yellow;
border: solid black 2px;
width: 600px;
height: 150px;
}
</style>
</head>
<body>
```

Designing Modernistic Websites © Aptech Limited

43

## CSS for Mobile 5-8

```
<HEADER>
<SECTION>
<b> Animation | Multimedia </b>
</SECTION>
<NAV id="nav">
<ul id="navigation">
<li><a title="Comprehensive Animation" href="#">Comprehensive Animation Pro</a></li>
<li><a title="2D and 3D" href="#">2D & 3D Animation </a></li>
<li><a title="3D Animation" href="#">Animation & 3D</a>
</li>
</ul></NAV>
</HEADER>
<SECTION id="intro">
<p style="font-size: 12px; font-style: italic; color:#0000FF">
Arena Animation is leader in animation and multimedia education with the widest network of centers across
the country. Over a span of 14 years, 250,000 students and professionals have, through Arena, found their
calling in animation, graphics, print publishing, web designing & films.
</p>
</SECTION>
<FOOTER>
<p>
<a href="tel:+91 22 2827 2300">Contact us </a> <br/>
Copyright © 2012 Aptech Ltd.
</p>
</FOOTER>
</body>
</html>
```

Designing Modernistic Websites © Aptech Limited

44

## CSS for Mobile 6-8

- Following figure displays the output of a mobile Web page with the CSS3 styles applied:



Designing Modernistic Websites © Aptech Limited

45

## CSS for Mobile 7-8

### ➤ Media Queries for Browser Detection

- Media queries are used to target specific features, such as screen width, orientation, and resolution of the devices.
- The use of a media query is to display HTML pages on various devices, such as computers and mobile devices with different styles based on their media types.
- In media queries, expressions are added for specific media type, then checking for condition is done, and finally, respective style sheet is applied to a Web page.

Media queries are used in two ways that are as follows:

Inline within a CSS style sheet

In the <link> tag as "media" attribute

Designing Modernistic Websites © Aptech Limited

46

## CSS for Mobile 8-8

- The Code Snippet shows the markup to apply a style sheet named screen.css to a device with screen and set the viewing-width of the area to 480.

```
<link type="text/css" rel="stylesheet" media="only screen and  
(max-device-width: 480px)" href="screen.css" />
```

- The Code Snippet shows the code to change the background color of a Web page depending on the device width.

```
@media only screen and (max-device-width: 480px) {  
body {  
background-color: #666;  
}}
```

- The Code Snippet shows the markups to serve style sheets based on the orientation of the device.

```
<link rel="stylesheet" media="all and (orientation: portrait)"  
href="portrait_orientation.css" />  
<link rel="stylesheet" media="all and (orientation: landscape)"  
href="landscape_orientation.css" />
```

Designing Modernistic Websites © Aptech Limited

47

### Instructions to the Trainer(s):

- Using Slides 40 to 47, explain the CSS for mobile.
- CSS3 provides properties for adding colors, selectors, borders, backgrounds, and so on for effective appearance of a Web page.

- Most modern mobile browsers support following features of CSS3:
  - Rounded corners
  - Images with borders
  - Adding shadow effect on text and boxes
  - Animations
  - Transitions
  - Multi-column layout
- Apart from W3C specifications that are laid for CSS3, modern browsers have provided new styles for CSS3. These styles are specific to each browser.
- Therefore, to add these styles on a Web page, the relevant properties must be prefixed with the browser specific keyword.
- The property prefixed with the keyword represents the browser on which it is supported. Table lists the keywords with their supported browsers.
- Using Slides 52 and 53, explain media queries for browser detection.
- Explain the students with a scenario which describes the use of properties that are used to display a Web page for a particular media which can be a screen or print. To provide the CSS for different media, media queries are used.
- Media queries are used to target specific features such as screen width, orientation, and resolution of the devices. The use of a media query is to display HTML pages on various devices, such as computers and mobile devices with different styles based on their media types.
- Inform the students that earlier CSS2 allowed creating style sheets for specific media types, such as screen and print. However, CSS3 has been enhanced further with the features of media queries.
- Media queries are used in two ways that are as follows:
  - Inline within a CSS style sheet
  - In the <link> tag as “media” attribute

## Slides 48 and 49

### Optimizing a Site for a Mobile 1-2

- Mobile Websites should be optimized for better performance.
- Some of the best practices that can be followed for mobile applications are as follows:

Design of a mobile Website should be simple to fit on small screens.

Avoid horizontal scrolling as some phones do not support horizontal scrolling and hide the content on the screen.

Use buttons, instead of providing many tiny links, as this can annoy the mobile users.

Create cookies to store the user's choice for viewing the full version of the site.

Avoid creating complex forms with many input fields, as data entry can be difficult on mobile devices compared to the desktops.

Limit the use of images due to bandwidth restrictions on mobile devices.

Designing Modernistic Websites © Apitoch Limited

48

### Optimizing a Site for a Mobile 2-2

Add mobile specific functionalities, such as built-in GPS facility or call-in action links.

Use of good foreground and background colors is important as they makes the sites readable on small screens.

Select the technologies that are compatible with old mobile devices.

Also, provide alternatives for functionalities, such as cookies, tables, style sheets, fonts, colors, and so on.

Avoid use of pop-up windows, tables for layout, frames, and image maps in the mobile Website design.

Designing Modernistic Websites © Apitoch Limited

49

### Instructions to the Trainer(s):

- Using Slides 48 and 49, explain how to optimize a site for mobile.

- Mention the requirements of mobile users are different from desktop users, in terms of screen size and connectivity issues. Thus, mobile Websites should be optimized for better performance.
- Explain some of the best practices that can be followed for mobile applications as listed on the slides.

For Aptech Centre Use Only

## Summary

- ❖ A mobile device is a small portable computing device with a small display screen and keyboard.
- ❖ Different categories of mobile devices available in the market are: basic model, low-end mobile devices, mid-end mobile devices, high-end mobile devices, smartphones, and tablets.
- ❖ A mobile platform is basically responsible to interact with the device hardware and run software/services on the mobile device.
- ❖ Different platforms for mobile devices include: iOS, Windows Mobile, and Android.
- ❖ An ideal mobile Website is supported and rendered properly by maximum possible browsers and OS.
- ❖ Two factors that need to be considered, while designing mobile Web application are its initial display (zoom) scale and orientation.
- ❖ The use of media query is to display HTML pages on different devices with different styles based on their media types.

50

**Instructions to the Trainer(s):**

- Show students Slide 50.
- Summarize the session by reading out each point on the Slide.