

Frontend Technologies for Beginners

Trainer Guide

Frontend Technologies for Beginners

Trainer Guide

© 2023 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

Edition 1 - 2023



PREFACE

HyperText Markup Language (HTML) is the markup language popularly used by developers to design Websites. This book 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

Table of Contents

1. Introduction to the Web and HTML5
2. Formatting Text and Using Hyperlinks and Anchors
3. Introduction to CSS3
4. Formatting Using Style Sheets
5. Displaying Graphics and CSS3 Animation
6. Understanding Layouts in CSS3
7. New Features of CSS3
8. HTML Layout, Forms, and Elements
9. HTML5 Tables, Audio, and Video
10. JavaScript - I
11. JavaScript - II
12. ECMAScript 2022 and Its New Features
13. Canvas and Web Storage in HTML5
14. HTML5 Geolocation and APIs
15. Building Mobile Web Applications
16. Mini Project

The Thumbnails in the Trainer Guide display a different visual design for the slides as compared to the Classroom Presentations for the sake of visual clarity.

Session 1: Introduction to the Web and HTML5

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

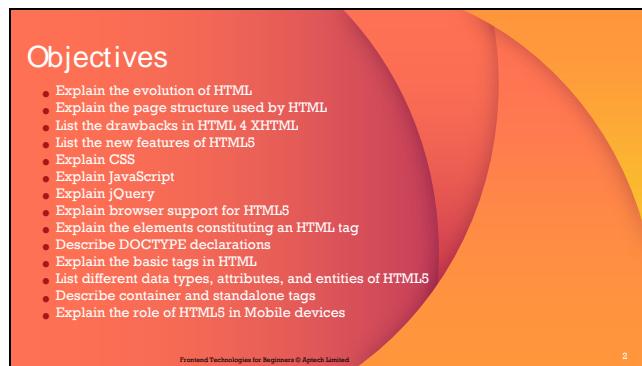
To teach this session, you should be well versed with evolution of HTML, drawbacks of HTML4, new features of HTML5, CSS, JavaScript, and so on. You should also be familiar with basic tags in HTML, data types, attributes, and entities of HTML5.

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



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

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

- 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).

At the bottom left is the text 'Frontend Technologies for Beginners © Aptech Limited'. At the bottom right is the number '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)

Frontend Technologies for Beginners © 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

Frontend Technologies for Beginners © 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 necessity 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).

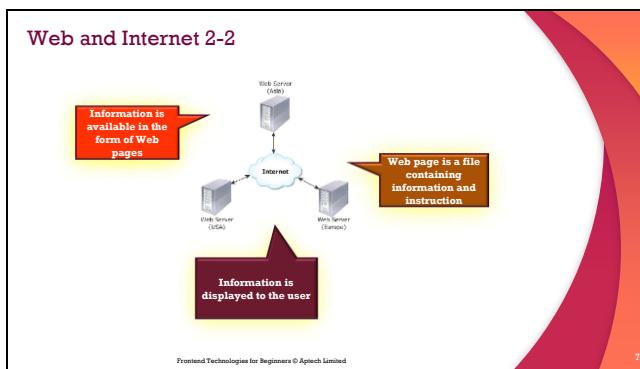
Internet is known as the largest WAN.

Web is a way to access information using Internet.

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

Fundamental Technologies for Beginners © Aptech Limited

6



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. The 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: Browsers are also referred to as Web clients.

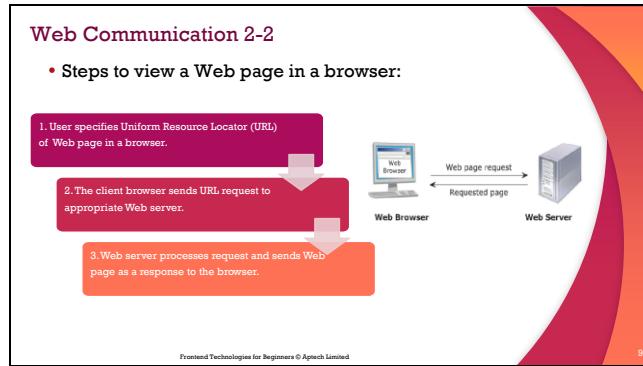
Web Communication 1-2

- 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.

Frontend Technologies for Beginners © Aptech Limited

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.



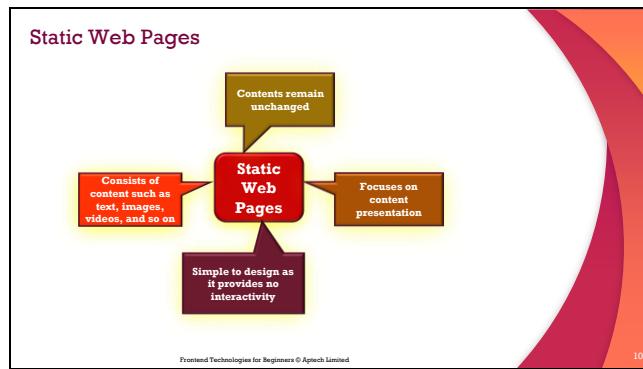
Instructions to the Trainer(s):

- 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:

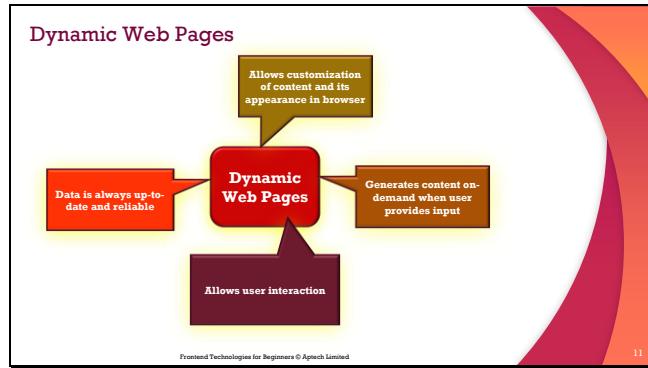
Question: What types of contents are observed on a Web page?

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.
- 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.

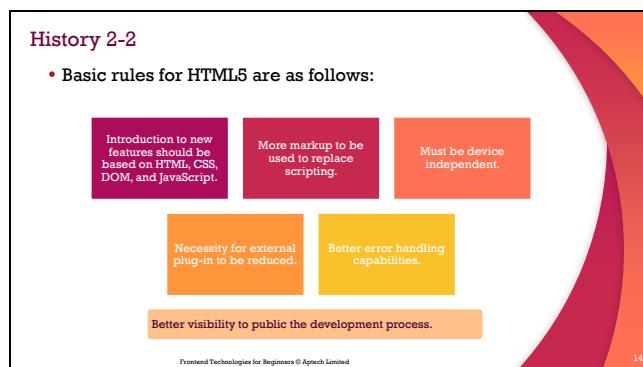
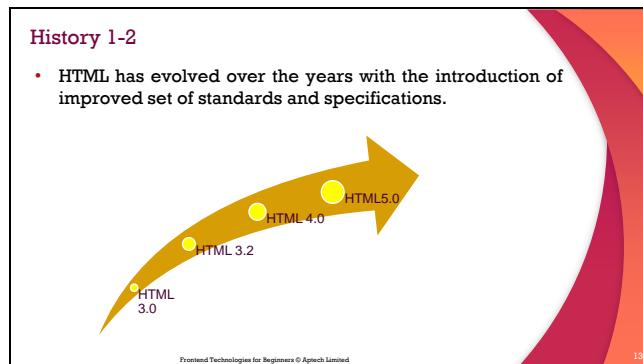
Frontend Technologies for Beginners © Aptech Limited

12

Instructions to the Trainer(s):

- Using Slide 12, 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.

Slides 13 and 14

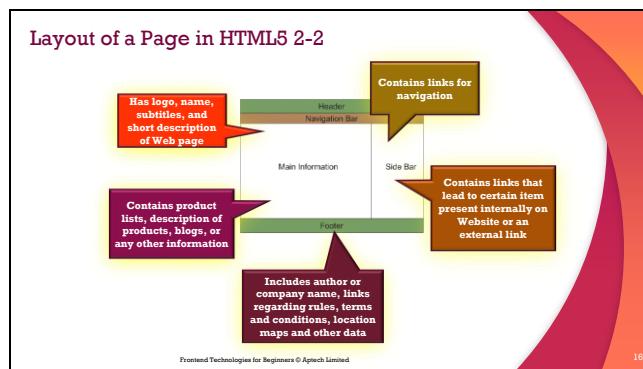
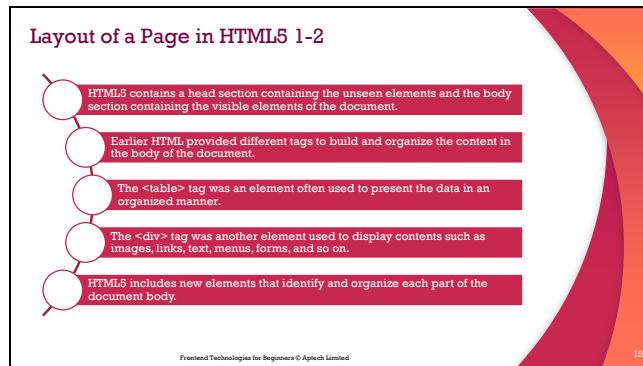


Instructions to the Trainer(s):

- Using Slides 13 and 14, 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. 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.

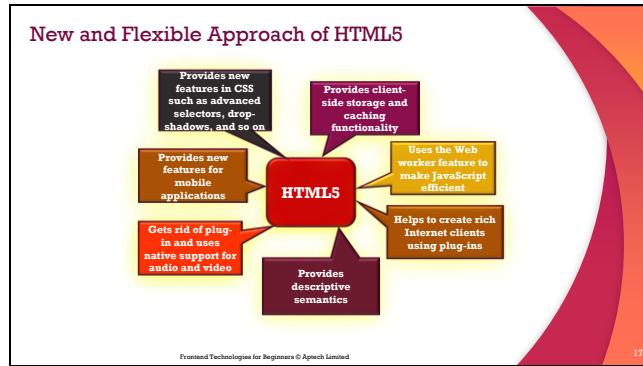
Slides 15 and 16



Instructions to the Trainer(s):

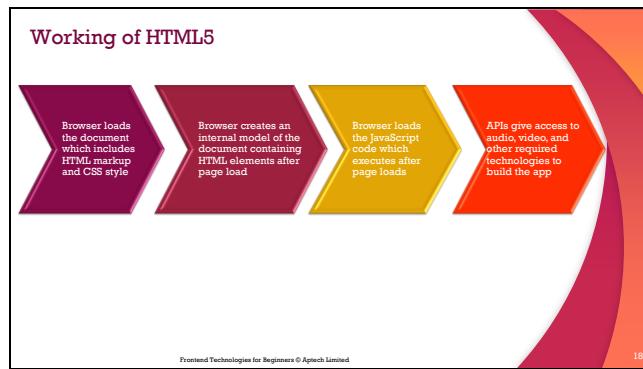
- Using Slides 15 and 16, explain the layout of a page in HTML5.
- Explain that the layout of page in HTML5 is same like that of HTML appended with the support for new tags and Application Programming Interfaces (APIs).
- Explain them the <table> and <div> tags used to layout the page in HTML. However, 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.



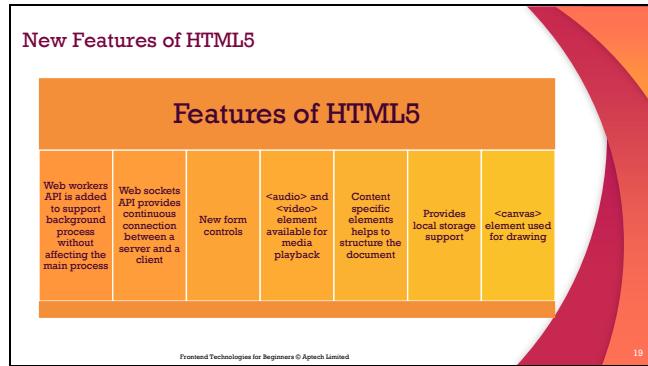
Instructions to the Trainer(s):

- Using Slide 17, 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.



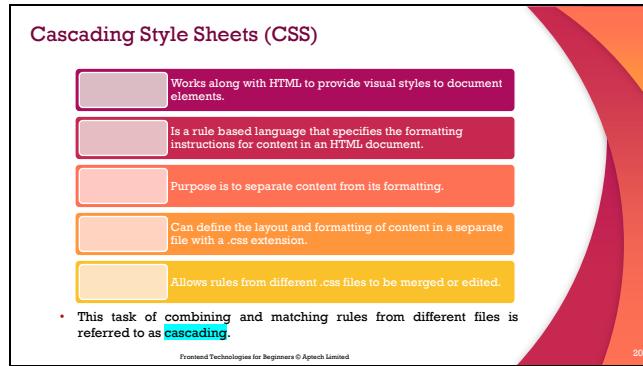
Instructions to the Trainer(s):

- Using Slide 18, 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.



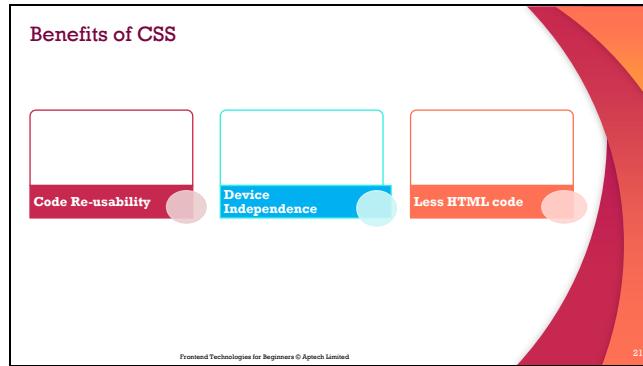
Instructions to the Trainer(s):

- Using Slide 19, 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.



Instructions to the Trainer(s):

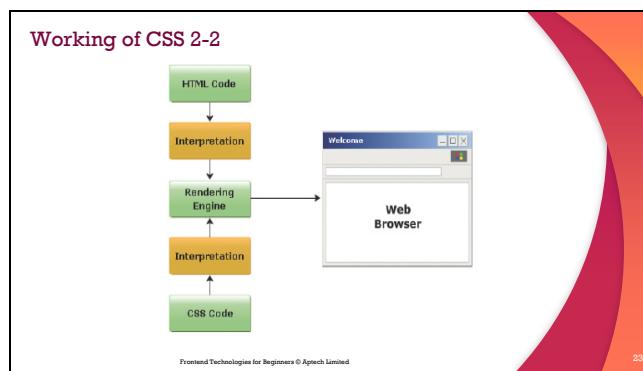
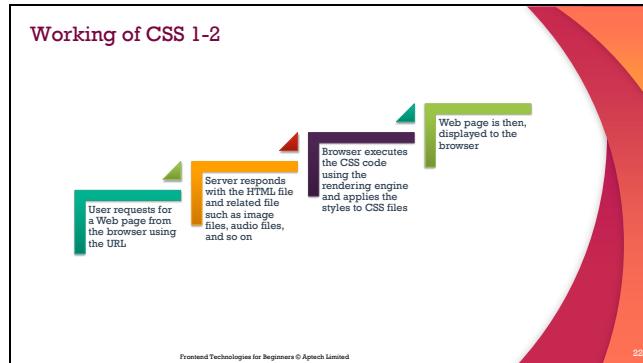
- Using Slide 20, explain the CSS as integral part of HTML5.
- A style sheet is a collection of rules that specifies the appearance of data in an HTML document. HTML is a markup language that focuses only on the layout of the content on a Web page. However, applying layouts to more than one occurrence 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 21, 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 22 and 23



Instructions to the Trainer(s):

- Using Slides 22 and 23, 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 24 and 25

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.

Frontend Technologies for Beginners © Aptech Limited

24

JavaScript 2-2

Functionality of JavaScript

- Helps Web designer to insert code snippets into the HTML page without the necessity 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.

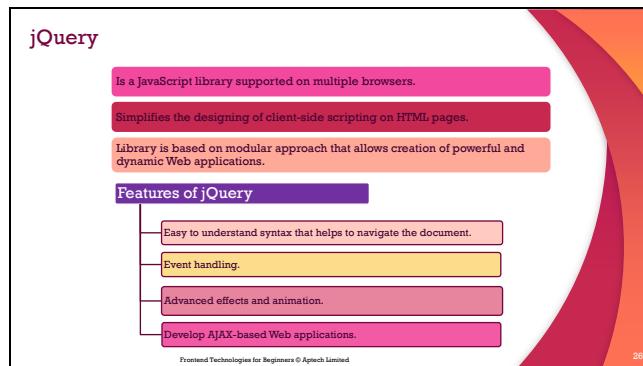
Frontend Technologies for Beginners © Aptech Limited

25

Instructions to the Trainer(s):

- Using Slides 24 and 25, 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 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.



Instructions to the Trainer(s):

- Using Slide 26, 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.

Slides 27 and 28

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.

Frontend Technologies for Beginners © Aptech Limited

27

Elements 2-2

- Following figure shows an element in HTML tag.

Frontend Technologies for Beginners © Aptech Limited

28

Instructions to the Trainer(s):

- Using Slides 27 and 28, 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.

Frontend Technologies for Beginners © Aptech Limited

29

Instructions to the Trainer(s):

- Using Slide 29, 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.

Slides 30 to 35

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.

Frontend Technologies for Beginners © Aptech Limited

30

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.

Frontend Technologies for Beginners © Aptech Limited

31

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.

Frontend Technologies for Beginners © Aptech Limited

32

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">`

Frontend Technologies for Beginners © Aprach Limited

33

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>
```

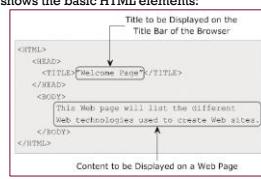
Frontend Technologies for Beginners © Aprach Limited

34

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:



Frontend Technologies for Beginners © Aprach Limited

35

Instructions to the Trainer(s):

- Using Slides 30 to 35, 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>

Slide 36 and 37

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.

Frontend Technologies for Beginners © Aptech Limited

36

Data Types 2-2

- Following figure shows different data types:

```
graph TD; A((Basic HTML Data Types)) --> B((Text Strings)); A --> C((Lengths)); A --> D((Colors)); A --> E((URIs)); F((Content Types)) --> A
```

Frontend Technologies for Beginners © Aptech Limited

37

Instructions to the Trainer(s):

- Using Slides 36 and 37, 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.
- 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.

Slide 38

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.

Frontend Technologies for Beginners © Aprach Limited

38

Instructions to the Trainer(s):

- Using Slide 38, explain some of the global attributes of HTML5. 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
non-breaking space		 	
<	less than	<	<
>	greater than	>	>
&	ampersand	&	&
€	euro	€	€
©	copyright	©	©

Frontend Technologies for Beginners © Aptech Limited

39

Instructions to the Trainer(s):

- Using Slide 39, 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 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.

Frontend Technologies for Beginners © Aptech Limited

40

Instructions to the Trainer(s):

- Using Slide 40, 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
 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 on third-party components, thus reducing the licensing costs.
- All the required components will be readily available through the browser in HTML5.

Frontend Technologies for Beginners © Aptech Limited

41

Instructions to the Trainer(s):

- Using Slide 41, 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.
- 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 on 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.

Frontend Technologies for Beginners © Aptech Limited

42

Instructions to the Trainer(s):

- Using Slide 42, explain the benefits of using HTML5 for development of mobile device applications.
- Mention that 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.

Slides 43 and 44

- **Ein Element** entspricht der Zeichenkette "Zeil", wenn es zusammen mit **Wörtern** eines der **Zeichenketten** ist.
- **Die DOKTÖR** ist die **Zeichenkette** der **Zeichen** von **your document**.
- **ZU** kann **zwei** spezielle **Zeichen** sein, die **zu** **zwei** **verschiedene** **Typen** von **Elementen** gehören. Ein **Zeichen**, das **zu** **zwei** **verschiedenen** **Typen** von **Elementen** gehört, ist ein **Zeichen**, das **zu** **zwei** **verschiedenen** **Typen** von **Elementen** gehört.
- **BRÜDER** sind **spezielle** **Zeichenketten**, die **in** **HTML** **verwendet** werden.
- **A** **zeichenkette** **besteht** **aus** **einem** **Zeichen**, **oder** **aus** **mehreren** **Zeichen**, **die** **an** **einander** **gefügten** **sind**.
- **A** **zeichenkette** **besteht** **aus** **einem** **Zeichen**, **oder** **aus** **mehreren** **Zeichen**, **die** **an** **einander** **gefügten** **sind**.
- **HTML** **providiert** **verschiedene** **Arten** **von** **Zeichenketten**, **die** **verschiedene** **Werte** **enthalten** **können**. **Einige** **Zeichenketten** **enthalten** **Werte**, **die** **nicht** **als** **Zeichenketten** **ausgedrückt** **werden**.

Instructions to the Trainer(s):

- Show students Slides 43 and 44.
 - Summarize the session by reading out each point on the Slides.

Session 2: Formatting Text and Using Hyperlinks and Anchors

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

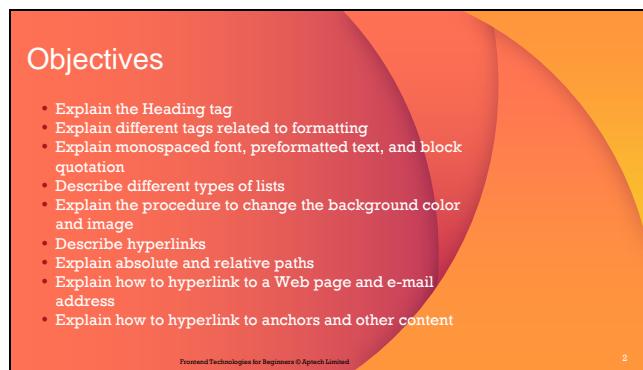
To teach this session, you should be well versed with formation of tags and different types of lists that can be displayed on a Web page. Also, monospaced font, preformatted text, and block quotation should be known. You should be well versed with creating hyperlinks and linking them to Web pages, e-mail addresses, 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.

.2 In-Class Explanations

Slide 3

The slide has a decorative background with a red-to-orange gradient curve on the right side. The title 'Introduction' is at the top left. Below it are four bullet points in orange 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.

Frontend Technologies for Beginners © Aprach Limited

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 to 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.

Frontend Technologies for Beginners © 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>
```

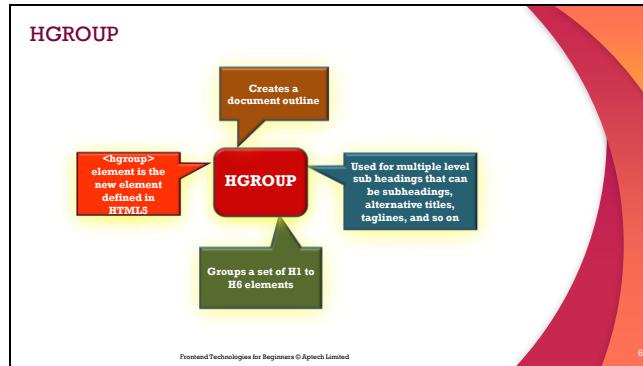


Frontend Technologies for Beginners © Aptech Limited

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 the 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.

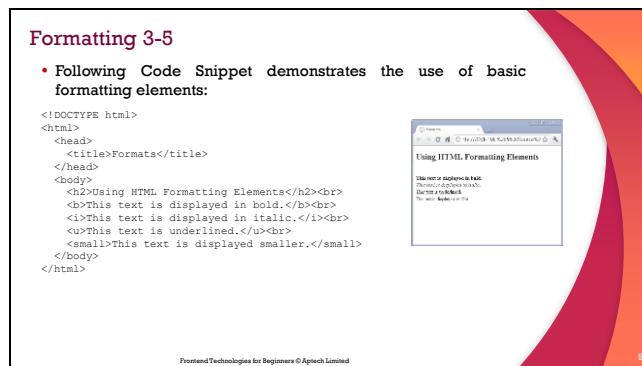
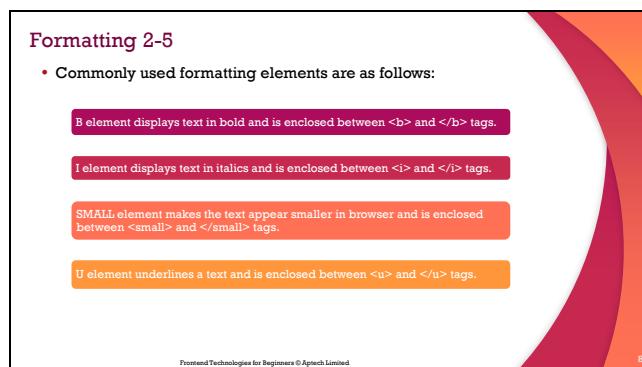
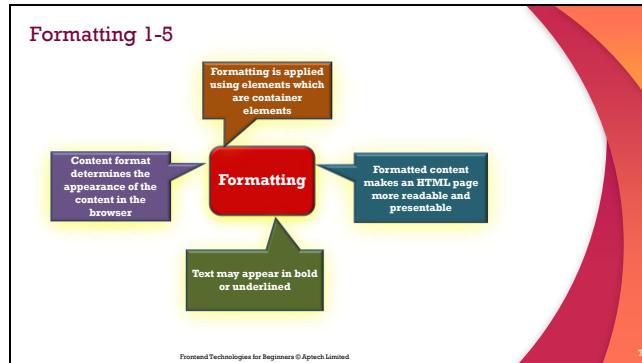


Instructions to the Trainer(s):

- 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.
- 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>
```

Slides 7 to 11



Formatting 4-5

- Some more formatting elements are as follows:

DEL element encloses deleted text and is placed between and tags.

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

STRONG element emphasizes the text and is placed between and tags.

SUB element displays a text as subscript and is enclosed between _{and} tags.

SUP element displays a text as superscript and is enclosed between ^{and} tags.

Frontend Technologies for Beginners © Aprach 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>
    <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>
```



Frontend Technologies for Beginners © Aprach 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.
- 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 and 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 `` and `` 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 `` and `` tags.
- **SUB:** The `SUB` element displays the text as subscript. The text to be displayed as subscript is enclosed in `_{` and `}` tags.
- **SUP:** The `SUP` element displays the text as superscript. The text to be displayed as superscript is enclosed in `^{` and `}` 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.

Slides 12 and 13

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.

Frontend Technologies for Beginners © Aptech Limited

12

Monospaced and Preformatted Text 2-2

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

| Tag | Description |
|--------|--|
| | 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 |

Frontend Technologies for Beginners © Aptech Limited

13

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 width. 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:

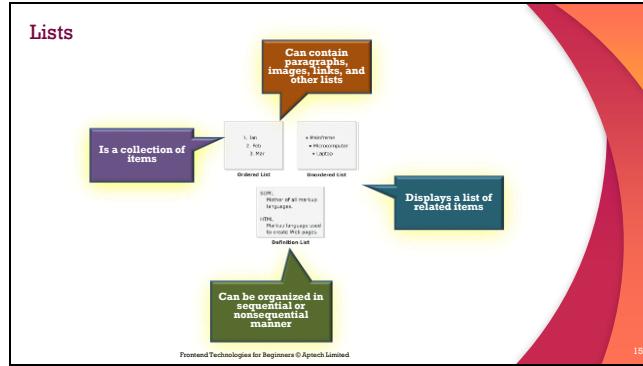
```
<blockquote>
  "When one door closes, another opens; but we often look so long and so
  regretfully upon the closed door that we do not see the one which has opened
  for us." -Alexander Graham Bell
</blockquote>
```

Frontend Technologies for Beginners © 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

Slides 16 and 17

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>
```



Frontend Technologies for Beginners © Aptech Limited

16

Ordered Lists 2-2

- Following table lists some of 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 tags.

Frontend Technologies for Beginners © 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 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.

- 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

Slides 18 to 20

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>
```



Frontend Technologies for Beginners © Apnach 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>
```

Frontend Technologies for Beginners © Apnach 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.



Frontend Technologies for Beginners © Apnach 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 `` tag and multiple `` sub-elements. The `` tag marks the beginning of the unordered list. Each of the sub-elements starts with the `` 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.

Slides 21 to 24

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.

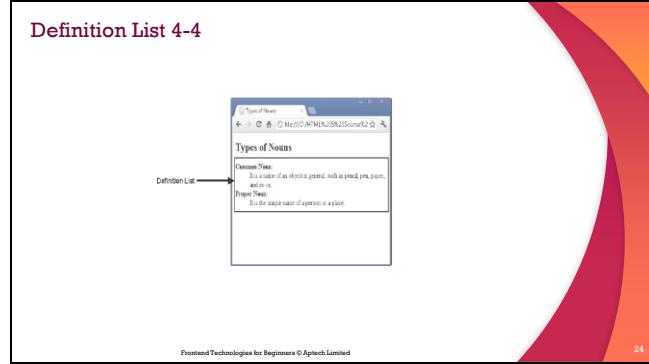
Definition List 2-4

- 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, 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>
```



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.

Slides 25 and 26

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)")

Frontend Technologies for Beginners © Apricot 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.

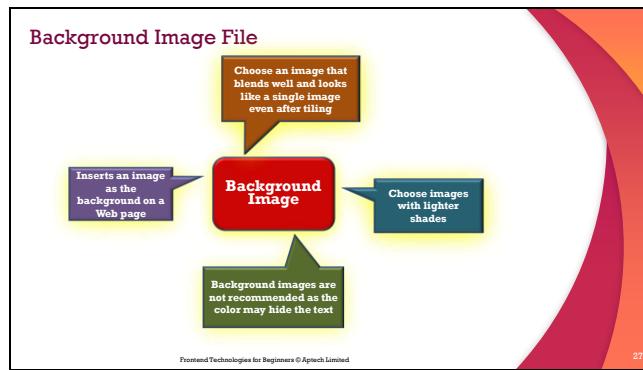
- Example demonstrating the specification of background and foreground color is:
`<body style="background-color: navy; color: yellow">`

Frontend Technologies for Beginners © Apricot 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>
```

Slides 28 and 29

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

Example: [This is a link](#)

Frontend Technologies for Beginners © Aptech Limited

28

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:

- The description or reference text that will serve as a hyperlink must be provided before closing the <a> tag by using .
- An example of a hyperlink along with its output is as follows:

```
<html>
  <head>
  </head>
  <body>
    <a href="http://www.contoso.com/">
      Click to view the Contoso Website</a>
    </body>
</html>
```

Output:

Frontend Technologies for Beginners © Aptech Limited

29

Instructions to the Trainer(s):

- Using Slides 28 and 29, explain hyperlinks.
- Mention hyperlink is referred to as a link. It refers to linking to another Web page or to a section in the same Web page.
- Explain to the students how to use the hyperlink to navigate the Web pages. Anything, such as 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.contoso.com/>">
- Mention that contoso here represents a fictional company and the site does not actually exist.

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 Web page. |
| <code>_top</code> | Loads the target URL in the complete area of window. |

Frontend Technologies for Beginners © Aptech Limited

30

Instructions to the Trainer(s):

- Using Slide 30, 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.

Slides 31 and 32

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.contoso.com/pages/about-us/aboutus_aboutcontoso.html">Contoso Website</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_aboutcontoso.html"> Contoso Website</a>
```

Frontend Technologies for Beginners © Aptech Limited

31

Absolute and Relative Paths 2-2

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

Syntax:

```
<a href="about-us/aboutus_aboutcontoso.html">Contoso Website</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_aboutcontoso.html">Contoso Website </a>
```

Frontend Technologies for Beginners © Aptech Limited

32

Instructions to the Trainer(s):

- Using Slides 31 and 32, 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, `Contoso Website`.
- 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, `Contoso Website`.
- To link to the files present in the subfolder, you must provide the path to the subfolder.
- For example, if the file `aboutus_aboutcontoso.html` is in a subfolder named, `about us` then, the syntax is as follows: `Contoso Website`
- 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: `Contoso Website`

In-Class Question:

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

Answer: Contoso

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:
`Customer Care`
- 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:
`Customer Care`

Frontend Technologies for Beginners © Aptech Limited

33

Instructions to the Trainer(s):

- Using Slide 33, explain how to hyperlink to an e-mail address.
- Hyperlinks can even be 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>
```

Frontend Technologies for Beginners © Aptech Limited

Instructions to the Trainer(s):

- Using Slide 34, 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.

Slide 35

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.
- ❖ 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.

35

Instructions to the Trainer(s):

- Show students Slide 35.
- Summarize the session by reading out each point on the slide.

Session 3: Introduction to CSS

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

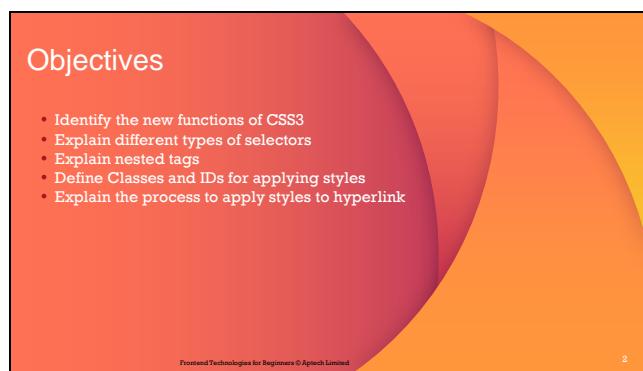
To teach this session, you should be well versed with functions of CSS3, applying styles to hyperlinks, and different types of selectors.

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.

3.2 In-Class Explanations

Slide 3

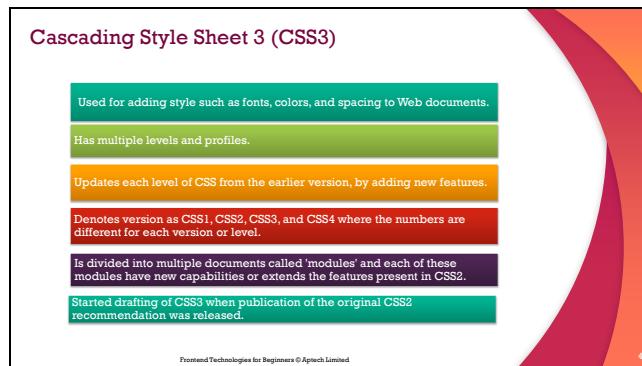
The slide has a white background with a decorative red and orange curved graphic on the right side. The title 'Introduction' is at the top left. Below it are four horizontal bars, each containing a statement about CSS:

- 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.

At the bottom left is the text 'Frontend Technologies for Beginners © Apache Limited'. At the bottom right is the number '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.



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 versions 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 in June 1999.
- CSS3 extends a variety of new ways to create an impact with any designs, with quite a few important changes.

Slides 5 to 8

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:
 - CSS Color Level 3
 - CSS Namespaces
 - Selectors Level 3

- Modules that are stable and in recommendation stage are as follows:
 - Media Queries
 - CSS Style Attributes

Frontend Technologies for Beginners © Aptech Limited

5

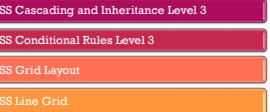
Modules 2-4

- Modules that are in testing phase and in recommendation stage are as follows:
 - CSS Backgrounds and Borders Level 3
 - CSS Image Values and Replaced Content Level 3
 - CSS Marquee
 - CSS Multi-column Layout
 - CSS Speech
 - CSS Mobile Profile 2.0
 - CSS TV Profile 1.0
- Modules that are in refining phase and in working draft stage are as follows:

Frontend Technologies for Beginners © Aptech Limited

6

Modules 3-4

- Modules that are in revising phase and in working draft and recommendation stage are as follows:
 - CSS Animations
 - CSS Flexible Box Layout
 - CSS Fonts Level 3
 - CSS Paged Media Level 3
 - CSS Text Level 3
 - CSS Basic User Interface Level 3
 - CSS Writing Modes Level 3
- Some of the following modules are in exploring phase and in working draft stage:
 - CSS Cascading and Inheritance Level 3
 - CSS Conditional Rules Level 3
 - CSS Grid Layout
 - CSS Line Grid

Frontend Technologies for Beginners © Aptech Limited

7

Modules 4-4

- Modules that are in rewriting phase and in working draft stage are as follows:
 - CSS Line Layout Level 3
 - CSS Ruby
 - CSS Syntax Level 3
- Modules that are in abandoned phase and in working draft stage are as follows:
 - Behavioral Extensions to CSS
 - CSS Hyperlink Presentation

Frontend Technologies for Beginners © Apricot 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
- The 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 | Property of a selected element | Value |
|---|---|--|
| • Is an HTML element for which you want to specify the style or the formatting instruction. | • Is a CSS property that specifies the type of the style to be applied to the selector. | • 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. |

Frontend Technologies for Beginners © 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.

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

- 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 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;}`

Slides 11 to 14

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.

Frontend Technologies for Beginners © 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. |

Frontend Technologies for Beginners © 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 |

Frontend Technologies for Beginners © 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:

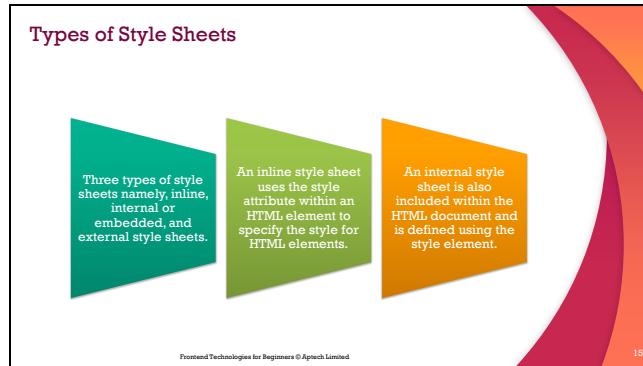
```
H1
{
    font-size: 120%;
    line-height: 200%;
}
```

Frontend Technologies for Beginners © Apricot Limited

14

Instructions to the Trainer(s):

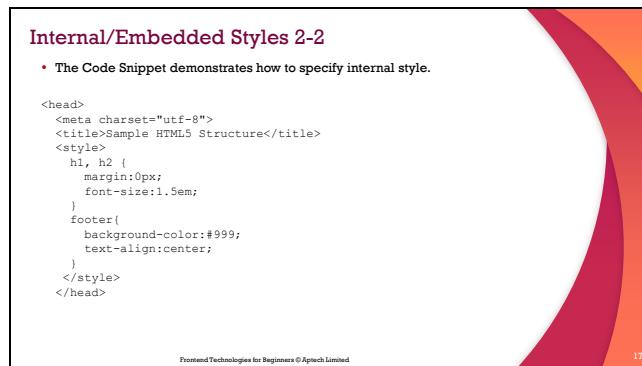
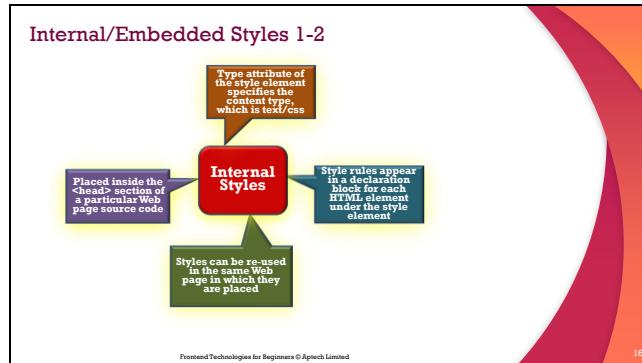
- Using Slides 11 to 14, explain length measurement units. Mention CSS uses various units of measurements for specifying size of the font, width, and height of margins, and so on.
- These units measure the horizontal and vertical length of the content. CSS supports two types of length measurement units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



Instructions to the Trainer(s):

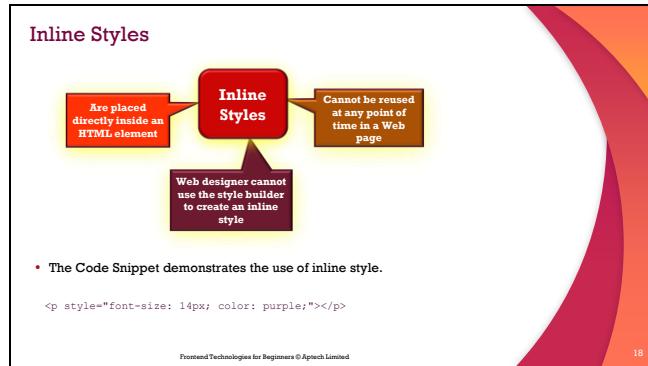
- Using Slide 15, explain the types of style sheets.
- Tell the students that the browser reads the style sheet and formats the content on the Web page accordingly.
- There are three types of style sheets namely, inline, internal, or embedded, and external style sheets. Here, explain the two types of 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.



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.

Slides 19 and 20

External Style Sheet 1-2

External CSS is defined in a separate file and is saved with .css extension

External Style Sheet

External CSS are widely used to provide a consistent look across the Web pages of a Website

Provides the benefit of reusability by implementing common style rules for multiple HTML pages

- The Code Snippet demonstrates the use of external CSS.

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

Frontend Technologies for Beginners © Aptech Limited

19

External Style Sheet 2-2

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

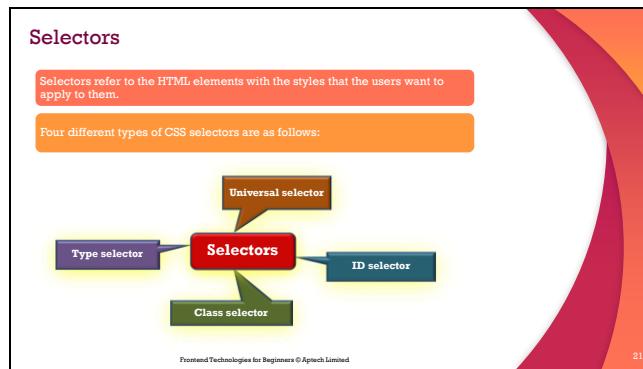
```
<!DOCTYPE html>
<html>
  <head>
    <link rel="stylesheet" type="text/css" href="body.css"/>
    <title>Webex e-Server</title>
  </head>
  <body>
    This is the fastest Web server..!!
  </body>
</html>
```

Frontend Technologies for Beginners © Aptech Limited

20

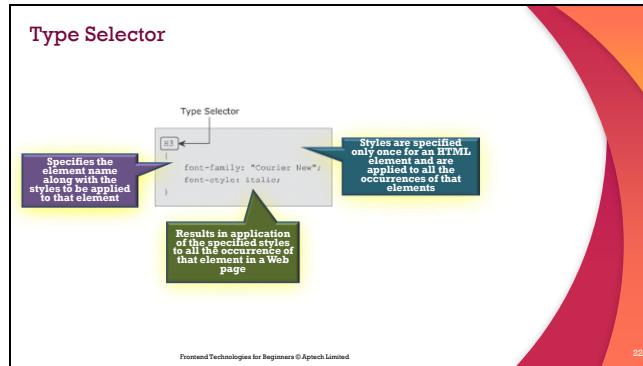
Instructions to the Trainer(s):

- 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.



Instructions to the Trainer(s):

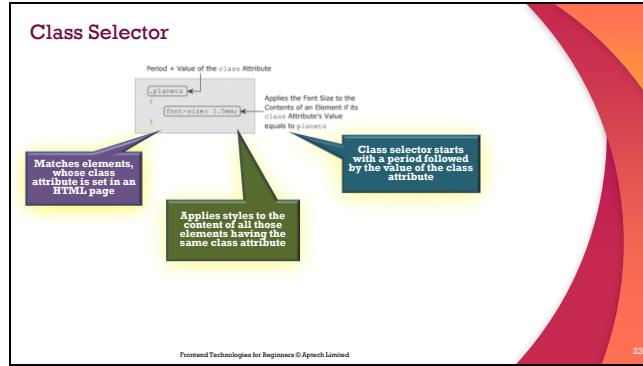
- Using Slide 21, explain selectors.
- Selectors refer to the HTML elements with the styles, the users want to apply to them.
- Four different types of CSS selectors are as follows:
 - Type selector
 - Class selector
 - ID selector
 - Universal selector



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:

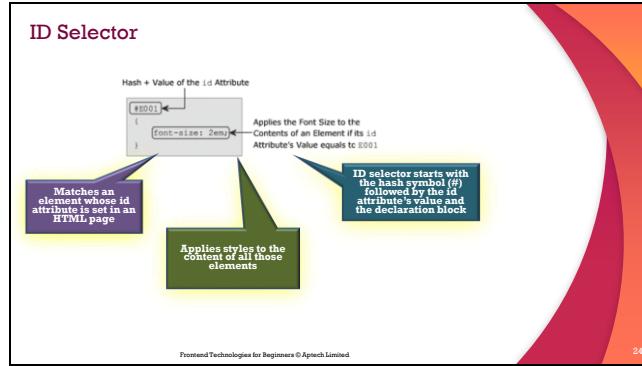
```
h1 { font-family: sans-serif }
```



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 `` 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, following code shows the matching class selectors for `<div>` and ``.

```
<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>
```



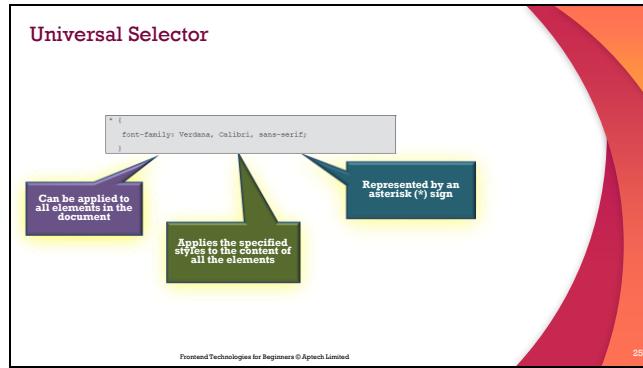
Instructions to the Trainer(s):

- 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;
}
```



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.

Slides 26 and 27

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.

Frontend Technologies for Beginners © Aptech Limited

26

Generic Cascading Order 2-2

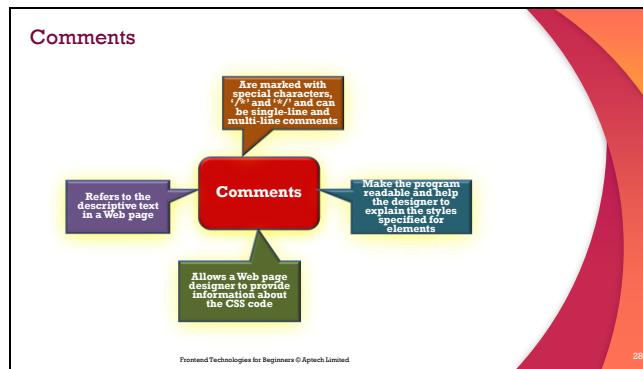
	Source	Browser	Reader	Author
CSS Type	External	Internal	Inline	
Selector	Type	Class	ID	

Frontend Technologies for Beginners © Aptech Limited

27

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;
}
```

Slides 29 to 32

Pseudo Classes 1-4

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

You might feel the necessity 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}
```

Frontend Technologies for Beginners © Aptech Limited

29

Pseudo Classes 2-4

- Following table lists 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.

Frontend Technologies for Beginners © 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

Frontend Technologies for Beginners © 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 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

Frontend Technologies for Beginners © 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 necessity 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:

```
Can be either first-line or first-letter  
selector name:pseudo_element {property: value}  
Is an element name.  
Is any CSS property such as color, border, and font.
```

Frontend Technologies for Beginners © 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.

The slide has a decorative red and orange curved graphic on the right side.

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: bold;  
background-color: #FFFFCC;  
}
```

A callout box points to the background color line with the text: "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;  
}
```

A callout box points to the font size line with the text: "Specifies the styles to be applied to the first letter of the paragraph content".

Frontend Technologies for Beginners © 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.

Frontend Technologies for Beginners © 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:
 - Div specific
 - 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.

Slide 36

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 4: Formatting Using Style Sheets

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

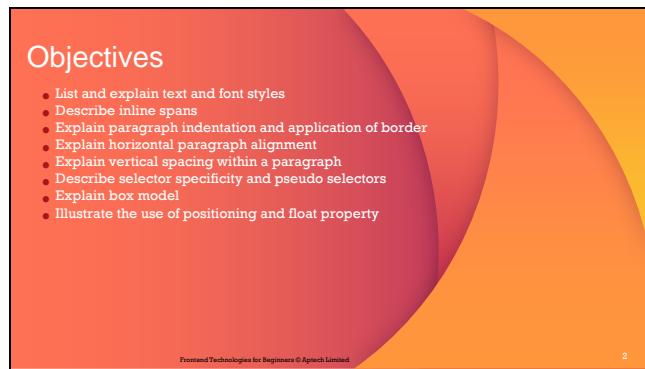
To teach this session, you should be well versed with CSS properties to format text using various font properties. You should also familiarize yourself with the paragraph indentation and applying borders to various elements on the Web page.

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



Instructions to the Trainer(s):

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

4.2 In-Class Explanations

Slide 3



Instructions to the Trainer(s):

- Ask the class to describe how modern Web designing uses style sheets, based on what they learnt until now. Let them answer. Guide their responses with better answers, if necessary.
- Tell students that:
 - HTML and CSS are two of the core technologies for building Web pages.
 - HTML provides structure of the page, whereas CSS provides visual layout.
 - You can use CSS to describe the presentation of Web pages. This includes colors, layout, and fonts. CSS is independent of HTML. Style sheets can be shared across pages, and customized to different environments.
 - One can consider graphics, scripting, HTML, and CSS as key tools to build Web pages and applications.

Slide 4

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.

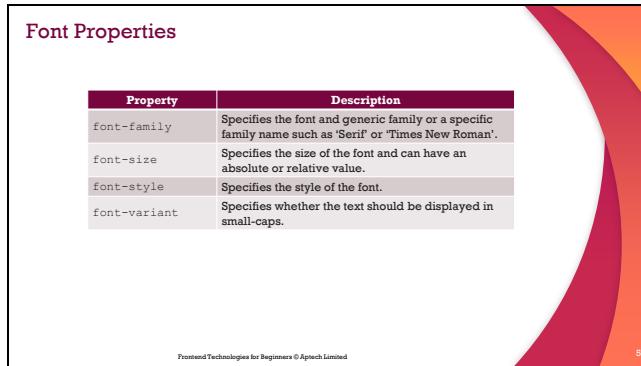
Frontend Technologies for Beginners © Aptech Limited

4

Instructions to the Trainer(s):

- Mention that the text properties of CSS can be used to control the appearance of the text in a Web page.
- Discuss different changes that can be applied to the text.
- Use Slide 4 to explain different text properties.

Slide 5



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.

Frontend Technologies for Beginners © Aptech Limited

Instructions to the Trainer(s):

- Tell the students that different font attributes such as font family, font size, and font style can be applied to text using font properties.
- Use Slide 5 to explain font properties.
- Give some examples to illustrate the use of these properties.
- For more information about font properties, refer to following links:

<https://www.tutorialrepublic.com/css-reference/css-font-property.php>
<https://developer.mozilla.org/en-US/docs/Web/CSS/font>

In-class Question:

Question: Which property specifies the casing of text in an element?

Answer: `text-transform`

Slide 6

Text Styles 1-3												
text-align Property												
<table border="1"><thead><tr><th>Property</th><th>Description</th></tr></thead><tbody><tr><td>left</td><td>Aligns the text to the left of the Web page.</td></tr><tr><td>right</td><td>Aligns the text to the right of the Web page.</td></tr><tr><td>center</td><td>Aligns the text in the middle of the Web page.</td></tr><tr><td>justify</td><td>Justifies the text on both sides of the Web page.</td></tr></tbody></table>		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
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.											
<table border="1"><thead><tr><th>Value</th><th>Description</th></tr></thead><tbody><tr><td>length</td><td>Specifies fixed indentation. The default value is 0.</td></tr><tr><td>%</td><td>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.</td></tr></tbody></table>		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.					
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												
<table border="1"><thead><tr><th>Value</th><th>Description</th></tr></thead><tbody><tr><td>none</td><td>Specifies that the text will be displayed with the same casing as written within the element.</td></tr><tr><td>capitalize</td><td>Specifies that the first letter of each word will be capitalized.</td></tr><tr><td>uppercase</td><td>Specifies only uppercase letters.</td></tr><tr><td>lowercase</td><td>Specifies only lowercase letters.</td></tr></tbody></table>		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.	
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.											
Frontend Technologies for Beginners © Aptech Limited												
6												

Instructions to the Trainer(s):

- Tell the students that there are different properties to specify the alignment, indentation, and casing of text in an element.
- Use Slide 6 to explain each property.
- For more information about text styles, refer to following links:

<https://developer.mozilla.org/en-US/docs/Web/CSS/text-align>

<https://developer.mozilla.org/en-US/docs/Web/CSS/text-overflow>

Slide 7

The slide is titled "Text Styles 2-3". It contains a code block and an output preview.

```
<!DOCTYPE HTML>
<html>
<head>
<link rel="stylesheet" type="text/css" href="TextProperties.css"/>
<title>Client</title>
</head>
<body>
<h2>Client Contact Information</h2>
<div>
<p>Ergonomic Solutions</p>
<p>Tel Number - +44 558 7744</p>
<p>Fax Number - 703 740 6539</p>
</div>
</body>
</html>
```

Output

Client Contact Information
ERGONOMIC SOLUTIONS
TEL: +44 558 7744
FAX: 703 740 6539

Frontend Technologies for Beginners © Aptech Limited

Instructions to the Trainer(s):

- Use the code snippet on Slide 7 and explain the HTML code for DIV element.

- Discuss the CSS code with the students. It should be:

```
div
{
text
align:
left;
text-
indent:
2em;
text-transform: uppercase;
}
```

- Explain the CSS code and describe how it specifies the text styles for the DIV element.
- Show the output of the code.

Slide 8

The slide is titled "Text Styles 3-3". It contains two tables: one for the "text-decoration" property and one for the "word-spacing" property. The "text-decoration" table has four rows: "none" (normal text), "underline" (a line under the text), "overline" (a line over the text), and "line-through" (a line through the text). The "word-spacing" table has two rows: "normal" (normal spacing) and "length" (fixed space between words). To the right of the tables is an HTML code snippet for a "Solar System" page. The code includes an H3 header ("Nine Planets"), a P paragraph with the text "Mercury, Venus, Earth, Mars, Jupiter, Saturn, Uranus, Neptune, Pluto", and a CSS link to "ParaProperties.css". Below the code is a "Output" section showing the rendered HTML with the text "Nine Planets" and the list of planets.

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

Instructions to the Trainer(s):

- Using Slide 8, explain the `text-decoration` and `word-spacing` properties.
- Say that the properties help specify decoration and word spacing of text in an element.
- Use the code snippet to show the HTML code for header and paragraph.
- Explain the use of `word-spacing` property and `text-decoration` property in the CSS code for `body` and `h3` elements. It should be:

```
body{
  word-spacing: 2 px;
}
h3{
  text-decoration: underline;
}
```

- Show the output.
- For more information about text styles, refer to following links:
<https://edu.gcfglobal.org/en/basic-css/text-styling-in-css/1/>
<https://developer.mozilla.org/en-US/docs/Web/CSS/text-decoration>
<https://developer.mozilla.org/en-US/docs/Web/CSS/word-spacing>

In-class Question:

Question: Which text-style property aligns the text to the right of the Web page?

Answer: `text-align`

Slide 9

Inline Span 1-2		
Attribute	Value	Description
class	classname	Specifies the text direction for the content in an element.
dir	rtl	Specifies the text direction for the content in an element.
ltr		
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 Tag

Frontend Technologies for Beginners © Aptech Limited

Instructions to the Trainer(s):

- Using Slide 9, explain inline span. Say that a element can be used to format a part of text.
 - Use the tag to mark up a part of a text, or a part of a document.
- Use the table and explain different inline span attributes.

Slide 10

The slide has a decorative red and orange curved graphic on the right side.

Inline Span 2-2

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

Output

My mother has **blue** eyes.

Frontend Technologies for Beginners © Aptech Limited

10

Instructions to the Trainer(s):

- Use the code snippet on Slide 10 and explain the CSS inline style for `` tag.
- Explain the CSS external style for `` tag. It should be:
`.eyesonly {font-color: lightblue}`
- Show the output.
- For more information about inline span, refer to following link:
<https://www.geeksforgeeks.org/span-tag-html/>

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 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>
```

Frontend Technologies for Beginners © Aptech Limited

11

Instructions to the Trainer(s):

- Ask students what is indentation and why is it used.
- Mention that indentation is a good way to indicate readers that they are about to get into another topic or another section.
- Explain the three types of indentation.
- Explain the inline style for `<p>` tag and the internal CSS code for first line indent. It should be this:

```
Inline style
<p style="text-indent: 50px">
Internal CSS
p {text-indent: 50px}
```
- Use the code snippet on Slide 11 to explain the use of the `text-indent` property in the HTML file.
- Discuss output of the code that uses `text-indent` property.

Indenting Paragraph 2-3

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

Output of Padding Property

The following screenshot shows the output of the padding property applied to the paragraph element. The text is centered and has a 20px padding on all four sides.

Frontend Technologies for Beginners © Aptech Limited

12

Instructions to the Trainer(s):

- Explain the use of inline style for `<p>` tag and the internal CSS code for the padding property. It should be:
 Inline style
 `<p style="padding: 20px">`
 Internal CSS
 `p {padding: 20px}`
- Use the code snippet on Slide 12 and explain the use of the padding property.
- Show the output that demonstrates use of padding property.

The slide has a decorative red and orange curved graphic on the right side.

Indenting Paragraph 3-3

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

Margin Property

The font styles 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, weight, color, etc. You can also apply the font styles properties to the entire page by applying them to the body element. If you don't apply the font styles properties to the body element, it will display the default font, which is dependent on the browser.

Frontend Technologies for Beginners © Aptech Limited

13

Instructions to the Trainer(s):

- Use the code snippet on Slide 13 and explain the use of inline style for `<p>` tag and the internal CSS code for margin property.
- Show the output.

In-class Question:

Question: What are the three types of indentation that can be applied in a paragraph?

Answer: First line indent, Padding, and Margin

Border Style 1-3		
border-style Properties	Description	Value
border-left-style	Sets an element's left border.	dashed
border-right-style	Sets an element's right border.	dotted
border-top-style	Sets an element's top border.	double
border-bottom-style	Sets an element's bottom border.	groove
		inset
		outset
		ridge
		solid
border-style Properties		Values of the border-style Properties

Instructions to the Trainer(s):

- Use Slide 14 to explain the border style properties and values.
- Tell the students that borders are present around text and images to emphasize the content.
- Using the CSS border properties, the style, color, and width of the border can be specified.

Slide 15

The slide has a decorative background with a red-to-orange gradient and a white circle on the right side.

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>
```

Welcome to Magnasoftwares

Output of border-style Properties

Frontend Technologies for Beginners © Aptech Limited

15

Instructions to the Trainer(s):

➤ Use the code snippet on Slide 15 and explain the HTML code to apply border styles.

➤ Discuss the CSS code for border styles. It should be:

```
#heading
{
background:
#FFEFDD;
text-align:
center;
border-left-style:
ridge; border-right-
style: groove;
border-top-style:
dashed; border-
bottom-style:
double;
}
```

➤ Show the output.

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

Frontend Technologies for Beginners © Aptech Limited

16

Instructions to the Trainer(s):

- Use the code snippet on Slide 16 and explain the HTML code.
- Discuss and explain the CSS code for shorthand border-style properties. It should be:

```
.largest_flower
{
    border-style: ridge groove dashed double;
}
```
- Show the output.

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

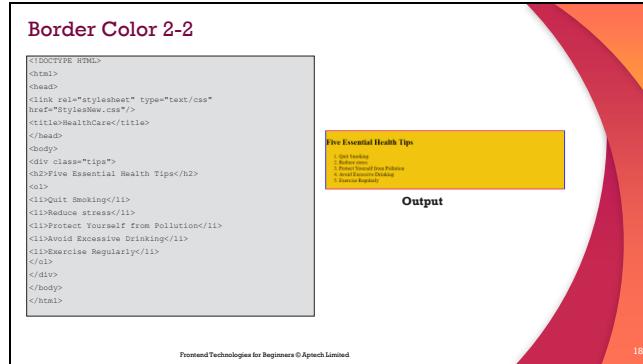
Value	Description
color	Specifies color to be applied to the border by using either the RGB or hexadecimal value, or the color name itself.
transparent	Specifies that the border is transparent.

Frontend Technologies for Beginners © Aptech Limited

17

Instructions to the Trainer(s):

- Using Slide 17, mention that the `border-color` property applies colors to the four borders.
- Four different colors can be applied to the four borders.
- Use the table in the Slide and explain different border color properties and the values.



Instructions to the Trainer(s):

- Explain the code snippet on Slide 18 and discuss the CSS for it. It should be:

```
.tips
{
border-style: solid;
background: #F1C40F;
border-
bottom-
color:#E91E63
; border-top-
color:
#E91E63;
border-right-
color:
#0000FF;
border-left-
color:
#0000FF;
}
```

- Show the output.
➤ For more information about border color, refer to following link:
<https://developer.mozilla.org/en-US/docs/Web/CSS/border>

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.

Frontend Technologies for Beginners © Aptech Limited

19

Instructions to the Trainer(s):

- Mention that the `border-width` property is a shorthand property to specify the width for all the four borders.
- Using the table shown on Slide 19, explain the border-width properties and different values.

In-class Question:

Question: Is color name allowed as a value for border color?

Answer: Yes

Border Width 2-3

```
.banner {  
    text-align:center;  
    background:#CCCCCC;  
    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

The screenshot shows a browser window with a title bar "EASYBANK - WHOLE WORLD ONE BANK". Inside the window, there is a banner with the text "EASYBANK - WHOLE WORLD ONE BANK". The banner has a thick black border at the top and a thick black border at the bottom, while the sides are solid but not thick. A callout bubble points to the right border with the text "Top Border".

Frontend Technologies for Beginners © Aptech Limited

20

Instructions to the Trainer(s):

- Using Slide 20, explain and show the use of border-width property using the code snippet and the output.

Border Width 3-3

Shorthand property: border-width

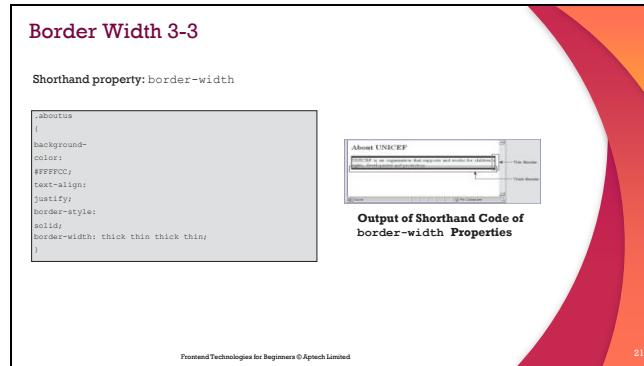
```
.aboutus {background-color: #FFFFCC; text-align: justify; border-style: solid; border-width: thick thin thick thin;}
```

About UNICEF

Output of Shorthand Code of border-width Properties

Frontend Technologies for Beginners © Aptech Limited

21



Instructions to the Trainer(s):

- Mention that the shorthand property for setting the border is border-width.
- Explain the use with the code snippet on Slide 21.
- Show the output.

The slide is titled "Shorthand Border 1-2". It contains a table with two columns: "Value" and "Description". The table includes entries for border-bottom, border-left, border-right, and border-top. Below the table is a section titled "Shorthand Border Properties" with a code snippet. A yellow box labeled "Output" shows the rendered HTML with borders applied. The slide has a decorative background with a red-to-orange gradient.

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

```
.Impnote {  
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;  
}
```

Output of Border Properties

Frontend Technologies for Beginners © Aptech Limited

Instructions to the Trainer(s):

- Say that the border properties can also be set individually by using different shorthand border properties.
- Use the table on Slide 22 and explain different shorthand border properties.
- Explain the usage through the code snippet and output.

Slide 23

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>Ivy is a species of flowering plants in the olive family. They are shrubs that range from 2 to 10m in height.</h3></td>
</tr>
<tr>
<td></td>
<td><h3>Sunflower is a flowering plant whose stem can grow as high as 3m.</h3></td>
</tr>
</table>
</body>
</html>
```

Frontend Technologies for Beginners © Aptech Limited

23

Instructions to the Trainer(s):

- Using Slide 23, explain the HTML and CSS code to apply the image border property.
- Discuss the output.

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

Frontend Technologies for Beginners © Aptech Limited

Instructions to the Trainer(s):

- Using Slide 24, mention that `text-align` property aligns the text horizontally in an element.
- Using the table, explain the values of different `text-align` properties.

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

Frontend Technologies for Beginners © Aptech Limited

Instructions to the Trainer(s):

- Using Slide 25, explain vertical alignment.
- Mention that the `line-height` property is used for vertical alignment of text in an element.
- The property is also a component of the ‘font’ shorthand property.
- Use the table and explain different values of the `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 pseudo-classes, and Elements and pseudo-elements

Frontend Technologies for Beginners © Aptech Limited

26

The diagram shows a vertical hierarchy:
Element
↳ Selector

Instructions to the Trainer(s):

- Using Slide 26, 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.
- For more information about selector specificity, refer to following link:
<https://developer.mozilla.org/en-US/docs/Web/CSS/Specificity>

Selector Specificity 2-2	
Rules	Example Code Snippet
If the rule is written twice, the lower rule will be applied.	<!DOCTYPE html> <html> <head> <style> h1 {background-color: red;} h1 {background-color: yellow;} </style> </head> <body><h1>This is my line.</h1> </body> </html>
ID selectors have a higher specificity than attribute selectors.	<!DOCTYPE html> <html> <head> <style> #one {background-color: yellow;} #one {background-color: blue;} </style> </head> <body><div id="one">This is my line.</div> </body> </html>

Rules	Example Code Snippet
Contextual selectors are more specific than single element selector.	CSS file: <style> #content h1 {background-color: blue;}</style> HTML file: <html><head><style> #content h1 {background-color: red;}</style></head><body><div id="content"><h1>This is my line.</h1></div></body></html>

Frontend Technologies for Beginners © Aptech Limited

27

Instructions to the Trainer(s):

- Using Slide 27, mention that there are different specificity rules.
- Use the table and explain the specificity rules.
- Discuss the output for each.

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 Selectors		
Pseudo-Class Selectors	Example Code Snippet	Output
:hover	<pre><!DOCTYPE html> <html> <head> <style> div { background-color: green; color: white; padding: 25px; text-align: center; } div:hover { background-color: blue; } </style> </head> <body> <div>Place the mouse over here to change the color or </div> </body> </html></pre>	<div style="background-color: green; color: white; padding: 25px; text-align: center;">Place the mouse over here to change the color</div> <div style="background-color: blue; color: white; padding: 25px; text-align: center;">Place the mouse over here to change the color</div>
:before	<pre>#para{ font-size: 18px; } #para:before{ content: "- BEFORE -"; background-color: green; } #para:after{ content: "- AFTER -"; background-color: green; }</pre>	<div style="font-size: 18px;">- BEFORE -</div> <div style="font-size: 18px;">- AFTER -</div>

Instructions to the Trainer(s):

- 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 following link: <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 left. Below it, a section titled 'A combinator indicates the relationship between selectors.' contains the following 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>
```

To the right, a section titled 'Example of Adjacent Sibling Selector' 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.
```

Below this is a caption: 'Output of Using Adjacent Sibling Selector'.

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.

The slide has a decorative red and orange curved graphic on the right side. The main content area has a white background with a black border.

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;
background: #e6f2ff;
padding: 10px; margin: 0;
}
</style>
<body>
<div>Hi! Check the padding and border</div>
</body>
</html>
```

Output of Code Snippet

Hi! Check the padding and border

Output After Changing Padding Value

Hi! check the padding and border

Frontend Technologies for Beginners © Aptech Limited

30

Instructions to the Trainer(s):

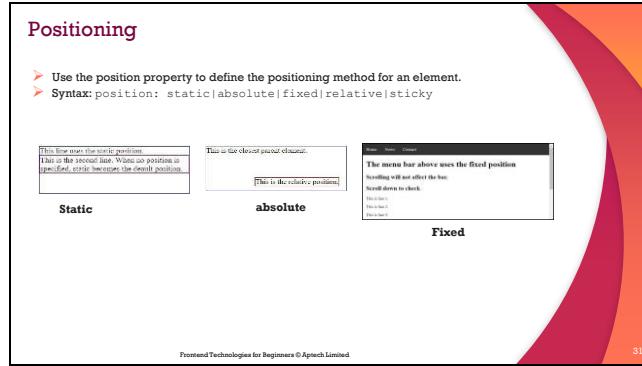
- Using Slide 30, explain box model.
- Say that a box model is a container that includes several properties. It includes borders, margin, padding and the content.
- Using the code snippet on Slide 30 and outputs, explain box model.
- For more information about the box model, refer to following links:

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

In-class Question:

Question: What surrounds the padding area?

Answer: Borders



Instructions to the Trainer(s):

- Mention that the position property can be used to define the positioning method for an element.
- Using the Slide 31, explain the syntax of the property.
- 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 following link: <https://css-tricks.com/almanac/properties/p/position/>

Float

Use the float property to position and format content.

```
<!DOCTYPE html>
<html>
<head>
<title>
</title>
</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>
```

Output

The screenshot shows a slide with a white background and a red-to-orange gradient border on the right. The title 'Float' is at the top left. Below it is a heading 'Use the float property to position and format content.' followed by a block of CSS code. To the right of the code is a small image of a flower and a text box explaining the two types of pollination: self-pollination (pollen from anther to stigma of the same flower) and cross-pollination (pollen from anther of one flower to stigma of another flower on a different plant). At the bottom left is the footer 'Frontend Technologies for Beginners © Aptech Limited'. The number '32' is in the bottom right corner.

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 following link:
<https://www.geeksforgeeks.org/what-is-float-property-in-css/>

Slide 33

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):

Using Slide 33, summarize the session by reading out each point on the Slide.

Session 5: Displaying Graphics and CSS3 Animation

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

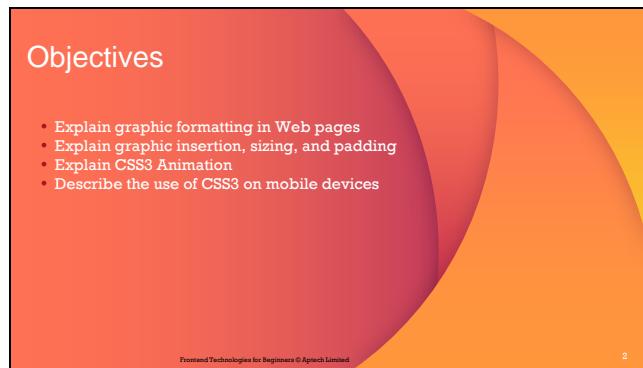
To teach this session, you should be well versed with graphic formatting in Web pages. Also, the graphic insertion, sizing, and padding should be known. Along with this, you should prepare yourself with how to apply CSS3 animation.

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.

5.2 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.



Frontend Technologies for Beginners © Apktech 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.

Slides 4 to 7

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.

Frontend Technologies for Beginners © 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.
- It may appear 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.

Frontend Technologies for Beginners © 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.

Frontend Technologies for Beginners © 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:

Frontend Technologies for Beginners © Apsach Limited

Instructions to the Trainer(s):

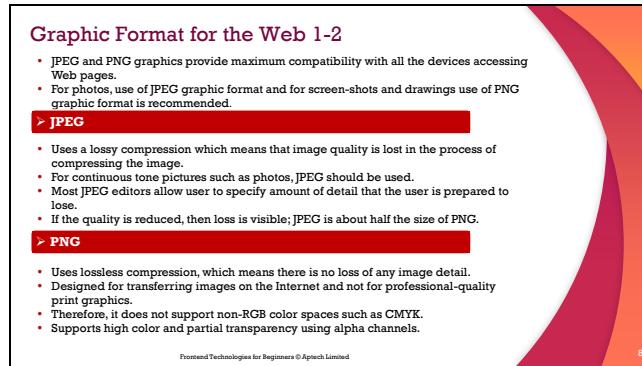
- Using Slides 4 to 7, explain the graphic format.
- Tell the students there are many graphic formats available; the most commonly used are Joint Photographic Experts Group (JPEG), Graphics Interchange Format (GIF), and Portable Network Graphics (PNG).
- The difference between each graphic format depends on following characteristics:
 - **Color Depth** – It is defined by the number of distinct colors that are represented by hardware or software. Color depth is defined by the number of Bits Per Pixel (BPP) and it is also called as bit depth. The 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.

Slides 8 and 9



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**

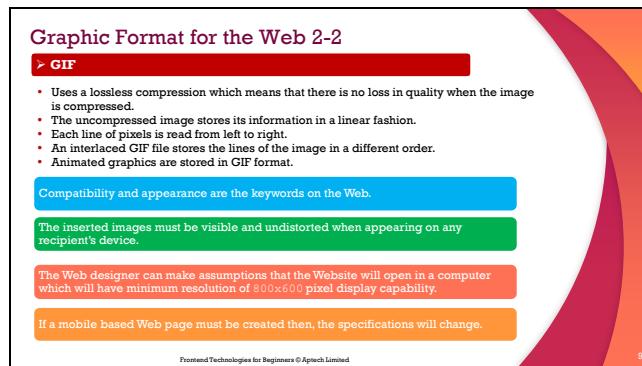
- Uses a lossy compression which means that image quality is lost in the process of compressing the image.
- For continuous tone pictures such as photos, JPEG should be used.
- Most JPEG editors allow user to specify amount of detail that the user is prepared to lose.
- If the quality is reduced, then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

Frontend Technologies for Beginners © 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 must be created then, the specifications will change.

Frontend Technologies for Beginners © 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.

Slides 10 to 15

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 `` 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.

10

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:

```

```

- 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.

11

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>
```

12

Graphic Insertion 4-6

- Following table lists the values of float property in the `` 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 `` tag.
It is not a replacement for `` tag, but acts as a container into which the `` 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.

Frontend Technologies for Beginners © Aptech Limited

13

Graphic Insertion 5-6

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

```
<figure>
![Logo](logo.gif)
</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>
![Logo](logo.gif)
<figcaption>This diagram shows the logo of a product.</figcaption>
</figure>
```

Frontend Technologies for Beginners © 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>
![Flowers](flower1.jpg)
![Flowers](flower2.jpg)
![Flowers](flower3.jpg)
<figcaption>The different types of flowers</figcaption>
</figure>
```

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



Frontend Technologies for Beginners © 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 `` 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 `` 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 `` tag. In other words, it is not a replacement for `` tag, but acts as a container into which the `` 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.
- 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.

Slides 16 to 18

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;
}
```

Frontend Technologies for Beginners © ApnaGhar Limited

18

CSS Image Sizing and Padding 2-3

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

Property	Description	Values
height	Sets the height of an element	• Auto • Length • % • inherit • none
max-height	Sets the maximum height of an element	• inherit • none • length • percentage
max-width	Sets the maximum width of an element	• inherit • none • length • percentage
min-height	Sets the minimum height of an element	• length • % • inherit
min-width	Sets the minimum width of an element	• length • % • inherit • auto
width	Sets the width of an element	• length • % • inherit

Frontend Technologies for Beginners © ApnaGhar 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

Frontend Technologies for Beginners © ApnaGhar 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.

Slides 19 to 21

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.

Frontend Technologies for Beginners © Agnieszka Limited

19

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.
```

Frontend Technologies for Beginners © Agnieszka Limited

20

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

Frontend Technologies for Beginners © Agnieszka 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.

Slides 22 and 23

Thumbnail Graphics 1-2

- 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.

Frontend Technologies for Beginners © Aprach Limited

22

Thumbnail Graphics 2-2

- Following figure shows output of thumbnail with hover effect:

Frontend Technologies for Beginners © Aprach Limited

23

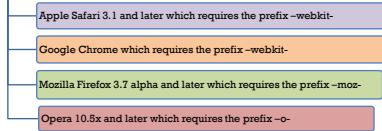
Instructions to the Trainer(s):

- Using Slides 22 and 23, 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.

Slides 24 to 27

- ### Working with CSS3 Transitions 1-4
- 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:

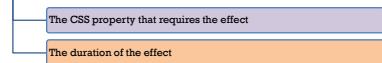


Frontend Technologies for Beginners © Aptech Limited

24

Working with CSS3 Transitions 2-4

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



- 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.

Frontend Technologies for Beginners © Aptech Limited

25

Working with CSS3 Transitions 3-4

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

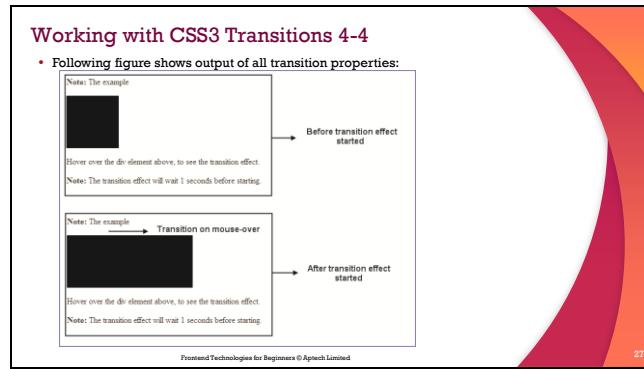
```
div:hover  
{  
    width:200px;  
}
```

- 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.

Frontend Technologies for Beginners © Aptech Limited

26



Instructions to the Trainer(s):

- Using Slides 24 to 27, 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 27, explain the output of the transition properties.

Slide 28

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.

Frontend Technologies for Beginners © ApnaGuru Limited

Instructions to the Trainer(s):

- Using Slide 28, explain the CSS3 animation.
- Tell the students that 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 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 are not currently visible.

In-Class Question:

Q: 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.

Slides 29 to 31

Configuring the Animation 1-3

- 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'.

Frontend Technologies for Beginners © Aptech Limited

29

Configuring the Animation 2-3

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:

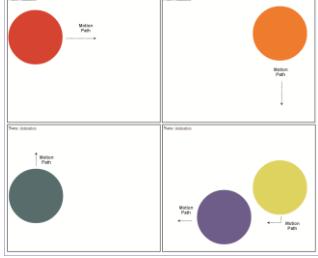
```
@keyframes myfirst
{
  from {background: red;}
  to {background: yellow;}
}
@-moz-keyframes myfirst /* Firefox */
{
  from {background: red;}
  to {background: yellow;}
}
@-webkit-keyframes myfirst /* Safari and Chrome */
{
  from {background: red;}
  to {background: yellow;}
}
```

Frontend Technologies for Beginners © Aptech Limited

30

Configuring the Animation 3-3

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



Frontend Technologies for Beginners © Aptech Limited

31

Instructions to the Trainer(s):

- Using Slides 29 to 31, 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 31, explain the output of @keyframes rules and animation properties used.

Slides 32 and 33

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.

Frontend Technologies for Beginners © Aprach Limited

32

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.contoso.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

Frontend Technologies for Beginners © Aprach Limited

33

Instructions to the Trainer(s):

- Using Slides 32 and 33, 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.contoso.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.

Slides 34 to 36

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.

Frontend Technologies for Beginners © Agnieszka Limited

34

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.

Frontend Technologies for Beginners © Agnieszka Limited

35

Coding for Optimum Browser Compatibility 3-3

- Best practices for optimum browser compatibility are as follows:
 - Test the Website in different browsers**
 - Review the Website's appearance and functionality on multiple browsers to ensure that 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.

Frontend Technologies for Beginners © Agnieszka Limited

36

Instructions to the Trainer(s):

- Using Slides 34 to 36, 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.

Slide 37

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 different shades of color to be applied to the borders.
- ❖ The 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.

37

Instructions to the Trainer(s):

- Show students Slide 37.
- Summarize the session by reading out each point on the Slide.

Session 6: Understanding Layouts in CSS3

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

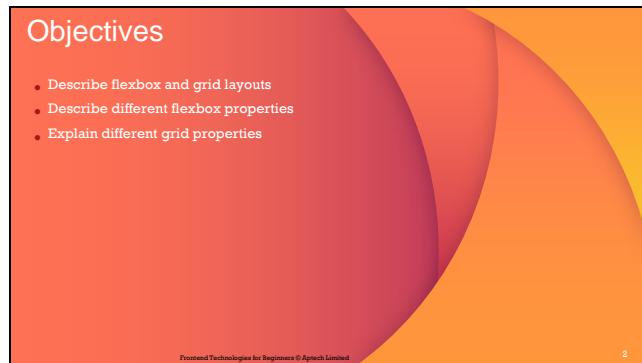
To teach this session, you should be well versed with CSS3 layouts.

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

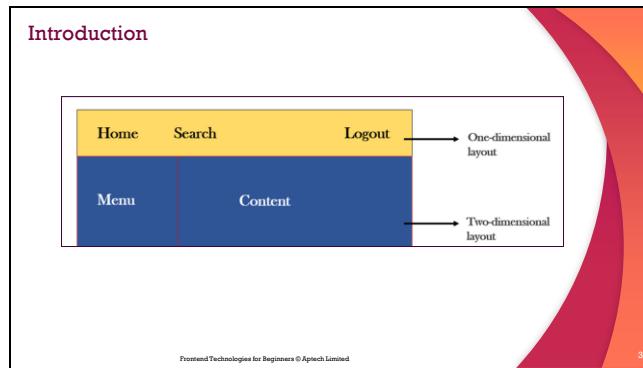


Instructions to the Trainer(s):

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

6.2 In-Class Explanations

Slide 3



Instructions to the Trainer(s):

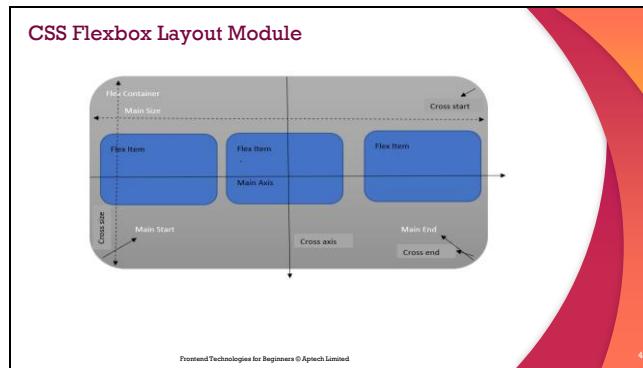
- Using Slide 3, explain Flexbox and Grid are the two popular layouts in the modern CSS.
- Flexbox provides one-dimensional layout and defines layout either for a row or a column.
- Grid provides two-dimensional layout and defines layout of both row and column, simultaneously.

In-class Question:

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

Answer: Flexbox.

Slide 4



Instructions to the Trainer(s):

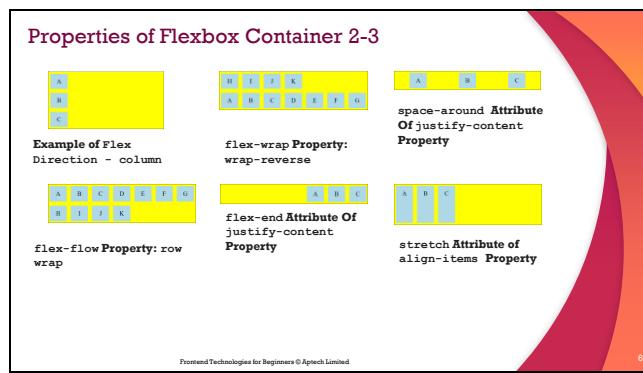
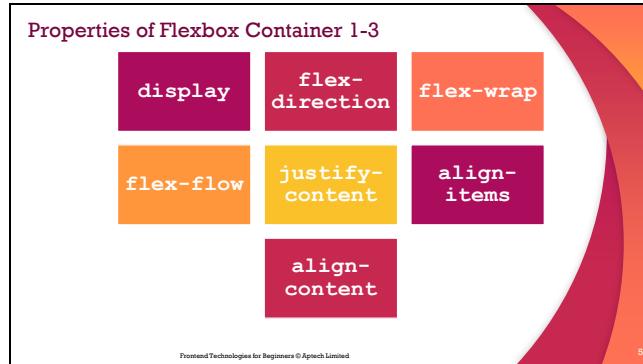
- Using Slide 4, explain the flexbox layout.
- Describe the flexbox components to the students.
 - Flex container is the basic element of the module.
 - All items in a flex container are called flex items. A flex container is a parent element and flex items are its children.
 - The direction of flex 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 following link:
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.

Slides 5 and 6



Instructions to the Trainer(s):

- Using Slides 5 and 6, 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 '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.

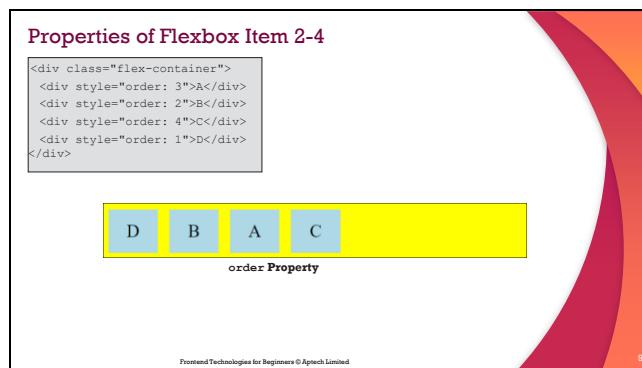
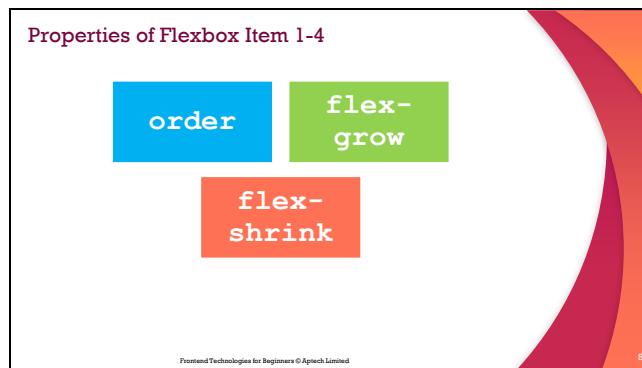
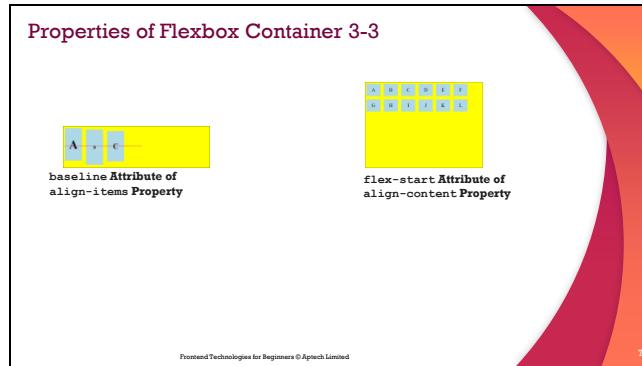
- For more information about flexbox container properties, refer to following link:
https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_Flexible_Box_Layout/Basic_Concepts_of_Flexbox

In-class Question:

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

Answer: display.

Slides 7 to 10



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>
```

flex-grow Property

Frontend Technologies for Beginners © ApnaGangotri Limited

10

Instructions to the Trainer(s):

- Using Slides 7 to 10, 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 following link:
<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`.

Slide 11

Properties of Flexbox Item 4-4

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

A B C D E F G H I
flex-shrink Property

Frontend Technologies for Beginners © Aptech Limited

11

Instructions to the Trainer(s):

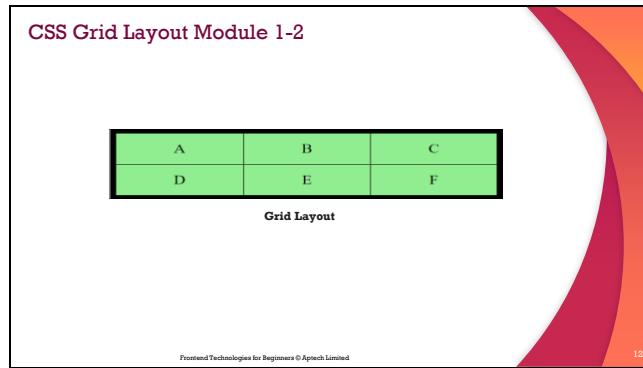
- Using Slide 11, explain grid layout.
- Tell the students that a grid layout has one parent element with one or more child elements.
- Items in a grid container are called grid items, which are placed inside columns and rows.

In-class Question:

Question: How many parent element does a grid layout have?

Answer: One parent element with one or more child elements.

Slide 12



Instructions to the Trainer(s):

- Using Slide 12, explain 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 to following link:
<https://webkit.org/blog/7434/css-grid-layout-a-new-layout-module-for-the-web/>

In-class Question:

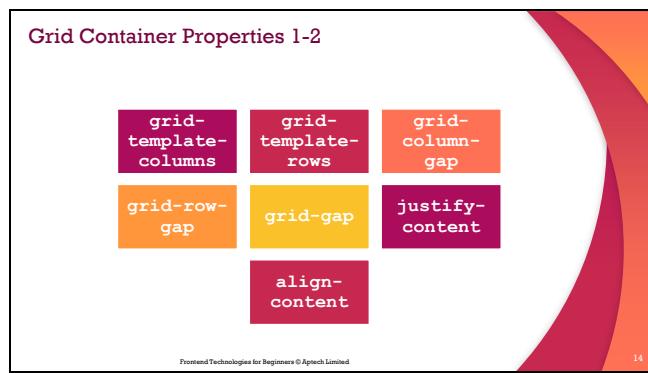
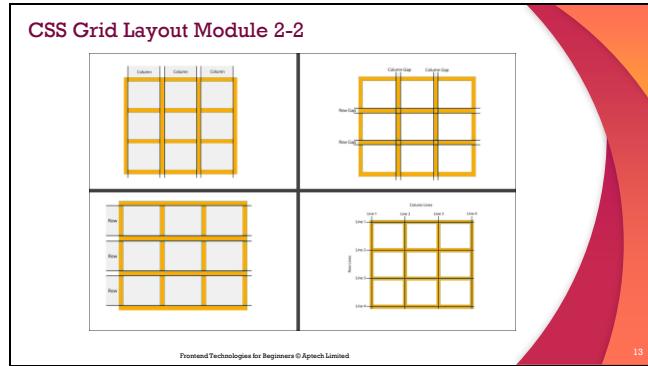
Question: What is the term that includes grid cells?

Answer: Grid area.

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

Answer: Grid container.

Slides 13 and 14



Instructions to the Trainer(s):

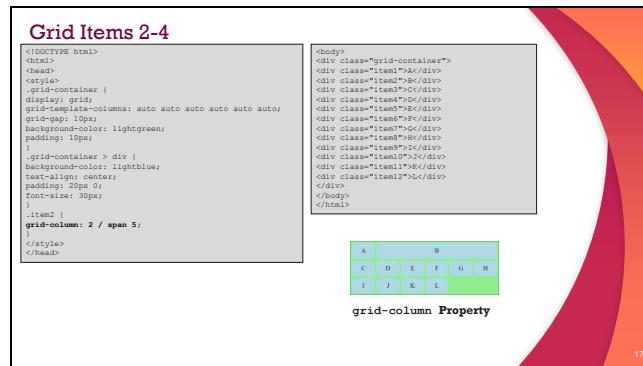
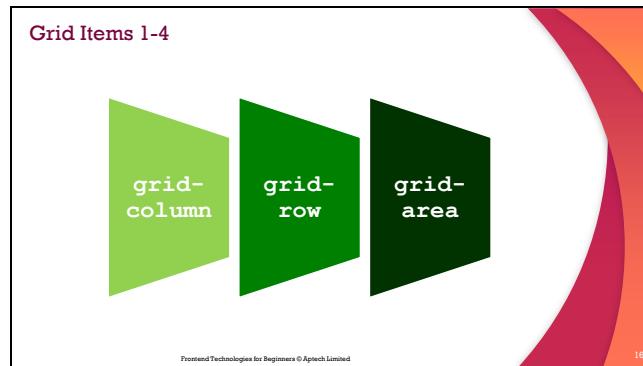
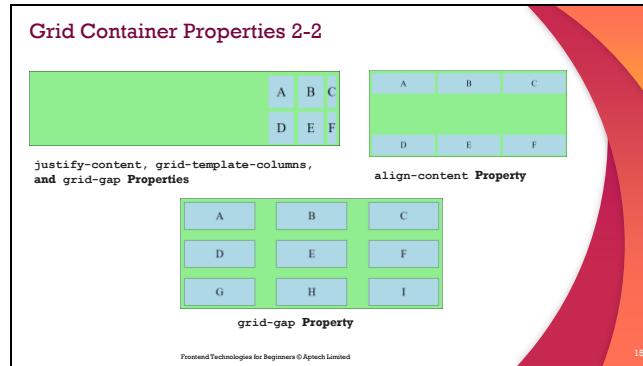
- Using Slides 13 and 14, explain the grid container properties.
- Tell the students that layout of the grid can be controlled using the grid container properties.
 - grid-template-columns defines number and size of columns
 - grid-template-rows defines height of rows.
 - grid-column-gap defines size 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-around .

Slides 15 to 19



Grid Items 3-4

```
<!DOCTYPE html>
<html>
<head>
<style>
.grid-container {
  display: grid;
  grid-template-columns: auto auto auto auto auto auto;
  grid-template-rows: auto auto;
  background-color: lightgreen;
  padding: 10px;
}
.grid-container > div {
  background-color: lightblue;
  text-align: center;
  padding: 20px 0;
  font-size: 30px;
}
.item {
  grid-row: 1 / span 2;
}
</style>
</head>
<body>
<div class="grid-container">
<div class="item">A</div>
<div class="item">B</div>
<div class="item">C</div>
<div class="item">D</div>
<div class="item">E</div>
<div class="item">F</div>
<div class="item">G</div>
<div class="item">H</div>
<div class="item">I</div>
<div class="item">J</div>
<div class="item">K</div>
<div class="item">L</div>
</div>
</body>
</html>
```

grid-row Property

Frontend Technologies for Beginners © Apache 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 main main main"
  "main main main main"
  "footer footer footer footer";
grid-gap: 10px;
background-color: lightgreen;
padding: 10px;
}
</style>
</head>
<body>
<div class="grid-container">
<div class="item1">Header</div>
<div class="item2">Menu</div>
<div class="item3">Main</div>
<div class="item4">Footer</div>
</div>
</body>
</html>
```

grid-area Property

Frontend Technologies for Beginners © Apache Limited

19

Instructions to the Trainer(s):

- Using Slides 15 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. Properties helps specify where to start and end the grid item with respect to column and row.

In-class Question:

Question: Which property helps use `grid-row-start`, `grid-column-start`, `grid-row-end`, and `grid-column-end` properties together?

Answer: `grid-area` .

Slide 20

Summary

- ❖ Flexbox provides one-dimensional layout.
- ❖ Grid provides two-dimensional layout.
- ❖ A container is the parent element and child elements of the container are flex items.
- ❖ A grid layout has one parent element with one or more child elements.
- ❖ Items in a grid container are called grid items.
- ❖ All grid items are placed inside columns and rows.
- ❖ Layout of grid, flexbox, flex items, and grid items can be controlled using different properties.
- ❖ A data attribute helps store custom data or extra information on an HTML element.

20

Instructions to the Trainer(s):

Using Slide 20, summarize the session by reading out each point on the Slide.

Session 7: New Features of CSS3

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 which 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 new subgrid feature of CSS3, viewport units, new features of CSS3 such as :has(), when/else, nesting of selectors, 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

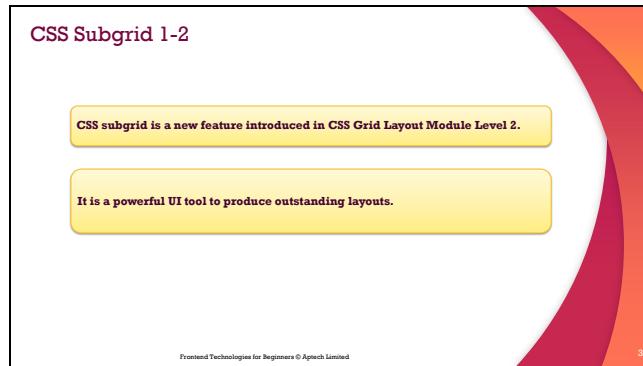


Instructions to the Trainer(s):

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

7.2 In-Class Explanations

Slide 3



Instruction(s) to the trainer:

Initiate the session with a general discussion that CSS has introduced some new features for ease of development.

Using Slide 3, explain CSS subgrid and tell students that it is a recently developed feature and is a part of the CSS Grid Layout Module Level 2. It is an incredibly useful UI tool that allows designers to create impressive layouts.

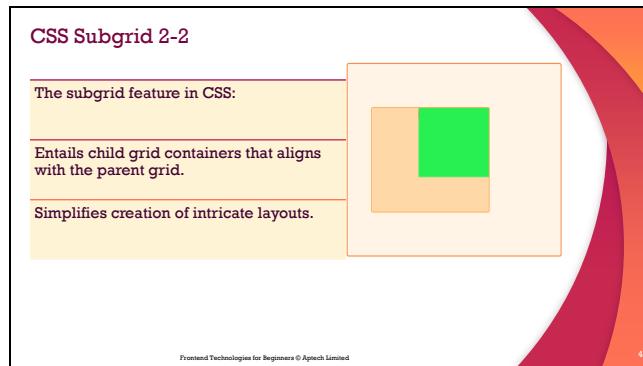
Further, explain that without subgrid nested grids can become disorganized when the grid is resized or expanded. This happens because each grid item can behave independently due to their varied content sizes.

Reiterate that one of the main issues with nested grids is that each grid item behaves independently, which can make it difficult to maintain consistent spacing and alignment between grid items.

Refer to following links for more information on CSS subgrid:

- <https://www.w3.org/TR/css-grid-2/#subgrid-items>
- <https://css-tricks.com/css-grid-layout-module-level-2/>

Slide 4



Instruction(s) to the trainer:

Using Slide 4, explain that subgrid solves the problem of disorganized nesting by allowing nested grids to inherit the column grid from the parent container.

Tell students that this means that the nested grids align with the parent grid making it easier to maintain a consistent layout across all grid items. Next, consider following example of a grid container comprising four grid items; header, sidebar, main content area, and footer:

```
.container { display: grid;
grid-template-rows: auto 1fr auto; /* Header, main content
area and footer */
grid-template-columns: 100px 1fr; /* Sidebar, main content
area */
grid-gap: 20px; } .

main { display: grid; grid-template-columns: repeat(3, 1fr);
/* Subgrid with 3 columns */
grid-gap: 20px; }
```

Explain to students that within the main content area, there is a subgrid that consists of three grid items. To ensure that the subgrid aligns with the parent grid, developer can assign the subgrid value to the grid-template-columns property of the .main container. They can also use the grid-template-columns property of the .subgrid container to determine the size and number of columns for the subgrid.

Subgrids also make it easier to create responsive layouts because the grid items within the nested grids can be defined independently of the parent grid.

This means that designers can create different layouts for different screen sizes and devices making it easier to adapt a Website to different platforms.

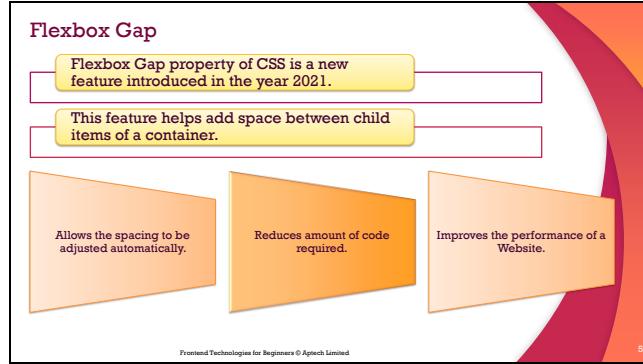
Refer to Code Snippet 1 of Learner Guide and explain the code to students. Explain the output as well.

Mention about the site <https://caniuse.com/> and its purpose. Tell students that this Website helps developers which browsers support a specific feature. For subgrid, one can use this link <https://caniuse.com/?search=subgrid> to know browser support for this feature.

Refer to following link for more information on use of subgrid:

➤ https://www.youtube.com/watch?v=Y5qGz_6TnOI

Slide 5



Instruction(s) to the trainer:

Using Slide 5, explain the next new feature, which is flexbox gap. Tell students that flexbox gap is a new feature in CSS that:

- Allows developers to add spacing between items in a flex container.
- Provides a convenient and easy way to add gaps between flex items without using additional markup or CSS.

Explain that earlier developers had to use different techniques to add spacing between flexbox items. For example, using margins, padding, or pseudo-elements.

Further, explain that flexbox gap is supported in modern browsers and is part of the CSS Box Alignment Module Level 3.

It is similar to the grid-gap property that allows for spacing between grid items in a CSS grid container.

Explain that with flexbox gap, developers can set the gap size between flex items using the gap property.

Consider following code snippet:

```
#flexbox { display: flex;  
flex-wrap: wrap;  
width: 200px;  
gap: 10px 20px; }
```

Tell students that in the code snippet, gap: 10px 20px; sets the gap between the flex items. 10 px represents the vertical gap and 20px represents the horizontal gap.

Continue explaining that the gap property can also be set to a single value to specify the same gap size for both rows and columns.

Refer to Code Snippet 2 of Learner Guide and explain the code to students. Explain the output as well.

Refer to following links for more information on use of flexbox gap:

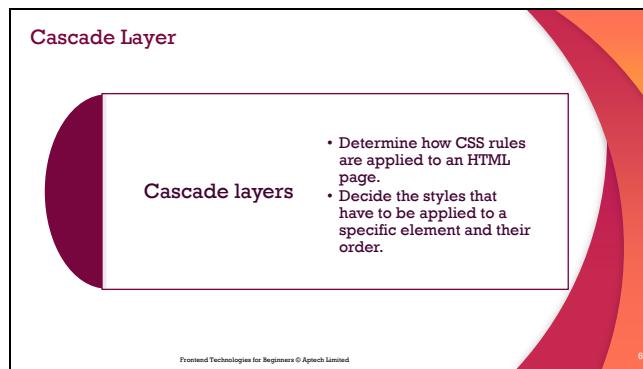
- <https://coryrylan.com/blog/css-gap-space-with-flexbox>
- <https://www.youtube.com/watch?v=Jj68iVdSw0g>

In-class Question:

Question: How is a subgrid implemented?

Answer. Subgrid is implemented by using a new property called grid-template-columns. This property is used to define the size and position of grid columns within a CSS grid container. When using subgrid, the grid-template-columns property is set to subgrid for the nested grid container. This allows the nested grid to inherit the column grid from the parent container. The parent grid must have a defined column grid for the subgrid to inherit.

Slide 6



Instruction(s) to the trainer:

Using Slide 6, explain the next new feature, which is cascade layer.

Explain that in CSS, the way styles are applied to an element is determined by the cascade. Styles are applied from each layer in order with later layers overriding earlier ones.

Next, consider following example:

```
@layer components, variations;  
@layer components { .button { color: #fff; background-color: #d73a7c; } }
```

```
@layer variations { .button--ghost { background-color: transparent; color: #474747; border: 2px solid #e0e0e0; } .button--facebook { background-color: var(--brand-fb); } }
```

Example Source: <https://ishadeed.com/article/cascade-layers/>

Tell students that the given code snippet allows the browser to group styles from identical @layer definitions and interpret them together according to the sequence. This ensures that the variation of a component will have a higher priority compared to the base styles.

Further, discuss that the order in which styles appear in the CSS file or document also affects their application. Tell students that by understanding the cascade, they can create well-organized and efficient CSS code that ensures that styles are applied correctly to the HTML elements in a Web page.

Refer to following link for more information on use of Cascade layers:

➤ <https://css-tricks.com/css-cascade-layers/>

Slide 7

New Viewport Units

- ✓ Viewport is the viewer area or the area that is as seen by the user in the browser.
- ✓ Viewports vary depending on the size of the device.
- ✓ There are four viewports in CSS – vw, vh, vmin, and vmax.
- ✓ The units, vmin and vmax, are recent additions to viewport units.

Fundamental Technologies for Beginners © Apricot Limited

7

Instruction(s) to the trainer:

Using Slide 7, explain the next feature, which is viewport units.

Start the class by explaining the importance of viewer area in responsive Websites.

Explain that viewport units in CSS are a type of length measurement that are relative to the size of the viewport, which is the visible area of a Web page on a user screen.

Further, discuss the types of viewport units. There are four viewport units in CSS, which are vw (width), vh (height), vmin (minimum), and vmax (maximum). These units allow developers to set the size and position of elements on a Web page relative to the size of the viewport rather than using fixed pixel values. This is very useful for creating responsive Web designs that can adapt to different screen sizes.

Continue explaining the recent additions which are vmin and vmax. Explain that vmin represents the smaller of vw and vh. For example, vmin can be used to set the font size of a heading. It can be smaller of the viewport width or height.

vmax is the bigger unit of vw and vh. For example, it can be used to set the width of an image to the larger of either the viewport width or height. This allows the image to scale up or down to fit the available space while maintaining its aspect ratio.

Refer to Code Snippet 3 of Learner Guide and explain the code to students. Explain the output as well.

Conclude by saying that viewport units are useful for creating designs that are responsive to different screen sizes because they adjust automatically based on the size of the viewport. However, also discuss that it is important to use them in conjunction with other CSS layout techniques to ensure that the design looks good on all devices.

Refer to following link for more information on use of viewport units:

<https://www.youtube.com/watch?v=lWFqGsXxJ1E>

In-class Question:

Question: What is the use of cascade layer?

Answer: The cascade layer in CSS is used to determine the order in which styles are applied to an HTML element.

Slide 8

The slide has a yellow header bar with the text "has()". Below it is a yellow callout box containing two bullet points: "✓ :has() is a CSS pseudo-class introduced in CSS Selectors Level 4 specification." and "✓ It allows selection of a specific parent element based on their descendant." At the bottom of the slide is a red box containing the syntax: "Syntax" followed by "parent:has(child) { /* Styles to apply to the parent element */ }". The slide is titled "Frontend Technologies for Beginners © ApnaGuru Limited" and has a small number "8" in the bottom right corner.

Instruction(s) to the trainer:

Using Slide 8, explain the next feature, which is `has()`.

Explain that the `:has()` pseudo-class can allow developers to select elements based on whether they contain a specific selector.

For example, developers can use `:has(p)` to select all elements that contain a `<p>` element.

Explain to the students that with the `:has()` pseudo-class, they can select elements based on whether they contain a specific selector. For example, developers can use `:has(p)` to select all elements that contain a `<p>` element. Developers can also style parent elements based on child elements, which means that they can use the `:has()` pseudo-class to select parent elements based on whether they contain a specific child element. This allows applying styles to the parent element based on the content of the child element.

Refer to Code Snippet 4 of Learner Guide and explain the code to students. Explain the output as well.

Next, explain another benefit. For example, use `article: has(h1 + p)` to select all `<article>` elements that contain an `<h1>` element followed by a `<p>` element.

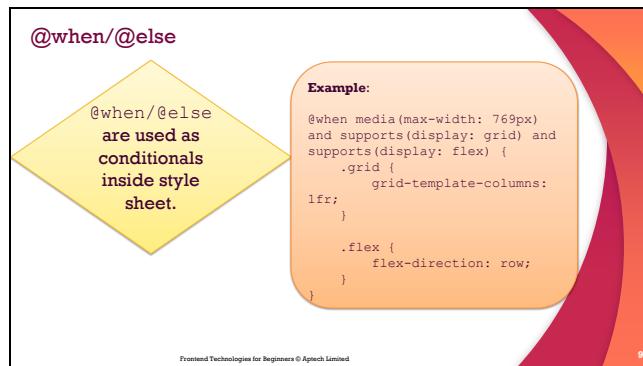
After explaining, ask the question, does `has()` improves performance?

Answer: Yes, it improves performance by allowing selection of elements based on their contents, the `:has()` pseudo-class reduces the requirement for JavaScript-based solutions.

Refer to following link for more information on use of has () :

<https://www.youtube.com/watch?v=lWFqGsXxJ1E>

Slide 9



Instruction(s) to the trainer:

Using Slide 9, explain that using conditionals are imperative in certain situations. Two new conditional rules are @when and @else.

For example, explain that @when allows developers to use multiple queries in one rule block making it easier than using a specific conditional rule, such as @support.

This means that single @when rule can be used to wrap other CSS conditional rules, such as @media and @support.

Consider following Code Snippet:

```
@when screen and (max-width: 780px) and supports(display: flex) { .my-element { color: red; display: flex; } } @else { .my-element { display: block; } }
```

Source: <https://fjolt.com/article/css-when-else-statements>

The code checks whether the screen width is less than or equal to 780 pixels and whether the browser supports the display: flex property.

If both conditions are true, the styles are applied to the .my-element class which sets its color to red and its display property to flex.

Refer to Code Snippet 5 of Learner Guide and explain the code to students. Explain the output as well.

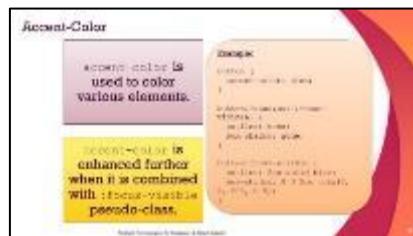
Conclude by saying that conditional statements were not a part of the standard CSS. Therefore, it is beneficial that they are now included. This will make the creation of media queries much simpler. While it is possible to use third-party packages such as SASS to include logic in CSS, the addition of conditional statements to CSS will eliminate the requirement for a pre-processor or any additional build steps.

Refer to following links for more information on use of @when/@else:

<https://css-tricks.com/proposal-for-css-when/>

<https://blog.logrocket.com/extending-css-when-else-chains-first-look/>

Slide 10



Instruction(s) to the trainer:

Using Slide 10, explain the accent-color feature. Explain that the accent-color is a property that sets the accent-color for user interface controls generated by some elements. For example, the color for buttons, inputs, and checkboxes. This gives a visual appeal to the Website.

To set the accent-color in CSS, developers can use the accent-color property. For example, you can set accent-color as blue for all buttons in a Website.

Refer to Code Snippet 6 of Learner Guide and explain the code to students. Explain the output as well.

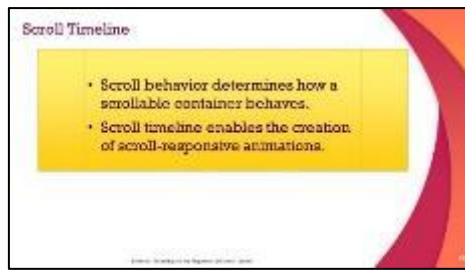
Explain that the accent-color property can also be used with the :focus pseudo-classes to change the accent-color when the user interacts with the element. For example, accent-color feature can be used to change the accent-color of an input field when it receives focus. You can combine the accent-color property with other CSS properties to style the input field further, such as changing the border color or background color on focus.

Refer to following links for more information on use of accent-color:

https://www.w3schools.com/cssref/css4_pr_accent-color.php

<https://css-tricks.com/almanac/properties/a/accent-color/>

Slide 11



Instruction(s) to the trainer:

Using Slide 11, explain that scroll behavior determines how a scrollable container behaves when a user scrolls through a Web page using a mouse, trackpad, or touch gesture.

Further, explain that CSS provides various properties that control the behavior of scrollable elements when users interact with them. For example, developers can use the CSS scroll timeline to create animations that respond to the scrolling behavior allowing for engaging visual effects or animating elements as users scroll into view.

Explain that when using CSS scroll timeline, a timeline of key frames is defined. It specifies the properties to animate and the scroll position at which they should be applied. As the user scrolls, the browser interpolates between these key frames to create a smooth animation.

Refer to Code Snippets 7, 8, and 9 of Learner Guide and explain the code to students. Explain the output as well.

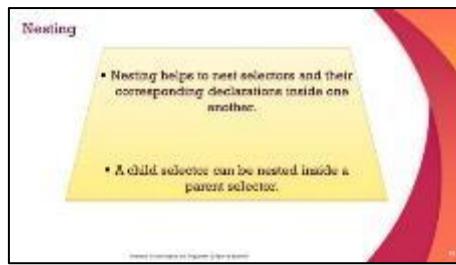
Conclude by explaining that overall CSS scroll behavior gives developers significant control over how scrollable elements behave on their Web pages. By utilizing a combination of CSS properties and scroll timeline animations, developers can create interactive and captivating scrolling experiences for their users.

Refer to following links for more information on use of scroll timeline:

<https://developer.mozilla.org/en-US/docs/Web/CSS/scroll-timeline>

<https://css-tricks.com/practical-use-cases-for-scroll-linked-animations-in-css-with-scroll-timelines/>

Slide 12



Instruction(s) to the trainer:

Using Slide 12, explain that nesting in CSS refers to the practice of placing a CSS rule or selector inside another CSS rule or selector creating a parent-child relationship. This allows developers to write more specific and targeted CSS code. This also makes the code easier to read and organize. For example, nesting also allows targeting specific elements within the container and applying styles to them.

Refer to Code Snippet 10 of Learner Guide and explain the code to students. Explain the output as well.

Conclude by explaining that nesting in CSS can be a powerful tool but it is important to use it carefully and not to overcomplicate the code.

Finally, discuss if nesting can pose any performance issues?

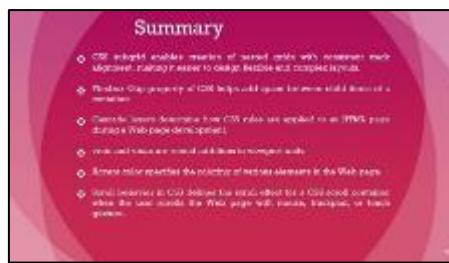
Yes, nesting in CSS can pose performance issues when it leads to excessive specificity in the selectors.

Refer to following links for more information on nesting:

<https://www.youtube.com/watch?v=V7Xinz27XQA>

<https://www.w3.org/TR/css-nesting-1/>

Slide 13



Instructions to the Trainer(s):

Show students Slide 13.

Summarize the session by reading out each point on the slide.

Session 8: HTML Layout, Forms, and Elements

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 concept of HTML layout, forms, and 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

- 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

Frontend Technologies for Beginners © ApnaGuru Limited

Instructions to the Trainer(s):

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

Slide 3

Introduction

Frontend Technologies for Beginners © Agnieszka Limited

HTML5 provides:

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

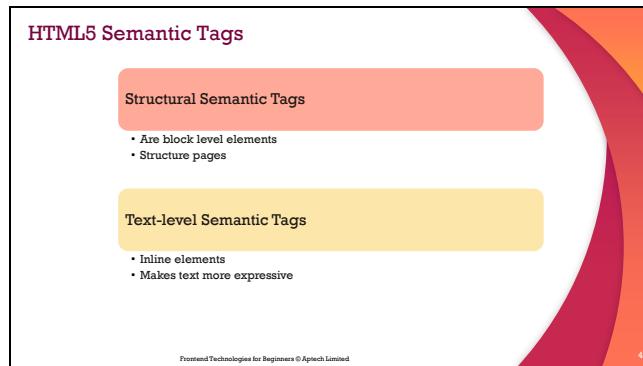
3

8.2 In-Class Explanations

Instructions to the Trainer(s):

- Using Slide 3, ask the students what attracts them in a Website. What kind of Website do they like? Let the students answer.
- Then, mention that a Web page is displayed in a browser. It might include information, such as logo, text, pictures, hyperlinks, and so on. An attractive Web page looks appealing to a user.
- HTML5 includes a set of markup elements to create a structured layout for Web pages.
 - These elements are called Semantic Markup as it conveys their meaning and purpose to the developer and browser.
- HTML5 includes new form controls and functionalities that helps create appealing and accessible forms.
- HTML5 also allows to use hidden elements in situations where certain data does not have to be seen or modified by users.
 - Ask for use cases from students.
 - Tell them hidden elements can be used for:
 - Ensuring form security.
 - Taking information of users from tracking parameters, such as URLs. Information from the URL can be taken to know if the visitors have come from a particular social media campaign.
 - Defining audience. For example, you can collect information about language and location from form submitters.

Slides 4 and 5



HTML5 Semantic Tags

Structural Semantic Tags

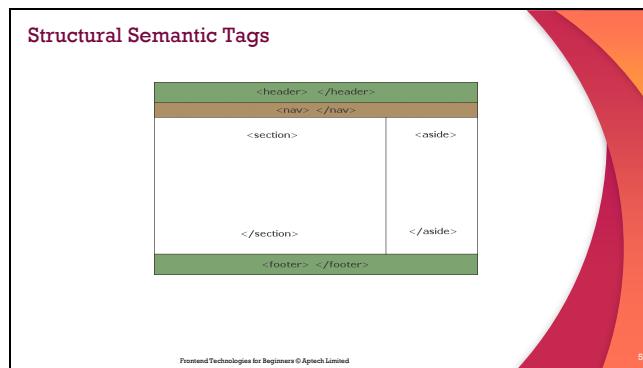
- Are block level elements
- Structure pages

Text-level Semantic Tags

- Inline elements
- Makes text more expressive

Frontend Technologies for Beginners © Aptech Limited

4



Structural Semantic Tags

<header>	</header>
<nav>	</nav>
<section>	<aside>
</section>	</aside>
<footer> </footer>	

Frontend Technologies for Beginners © Aptech Limited

5

Instructions to the Trainer(s):

- Using Slides 4 and 5, tell the students that there are two types of semantic tags: Structural and Text-level.
- Mention that a semantic tag clearly gives its meaning to both the browser and developer. For example, `<form>` and `<table>` clearly indicate the purpose of the element.
- Non-semantic tags, such as ``, indicate nothing about the element.
- Structural semantic tags identify the parts of a Web page, such as the header element indicates the introductory elements of a section. Main navigation menu links should be placed in the nav element.
- Text-level semantic tags are inline elements and are used to make the text more expressive. For example, you can bold and emphasize text.

In-class Question:

Question: Which semantic tags are used to make text expressive?

Answer: Text-level semantic tags.

Explain different structural tags.

In-class Question:

Question: Is it mandatory for the aside element to be aligned to the right or left of a Web page?

Answer: No. It can be at the top, the bottom, or even in the middle of a Web page.

Slide 6

HTML5 Semantic Tags	
Tag	Description
Section	The section element represents a section of a Web page. It is used for grouping related content and is different from other content groups present on the Web page. It is similar to a div tag though section element has more semantic meaning. In other words, section element is more meaningful as the content inside the section tags should be related.
Header	The header element represents the header of a Web page. It can be used either at the top of the document or at the top of a section. Though most of the Websites currently uses a single header at the top of the page called masthead, but a Web developer can have multiple headers in a single HTML5 document. This element is used as a container containing a group of introductory content or a set of navigational links.
Footer	The footer is similar to the header and can be present as the footer either for the document or for the section. There can be multiple footer elements in an HTML5 document. A footer element has information about the Web document. The typical contents which are placed in footer include Authors information, Copyright information, and Text-based navigation bar.
Aside	The aside element is used for representing the content that is related to the main text of the document. It aligns itself as a sidebar. As compared with other structural tags its importance is not related with its position within a document or its relation with other content. It is not intended to be aligned with the main content aligned to the right or left of a Web page. It can be at the top, the bottom, or even in the middle of a Web page.
Nav	The nav element represents a section of a Web page that contains navigation links/ menus to other Web pages or to other parts within the Web page. In other words, it allows the user to navigate through the Web page and site. This section is created for major navigational information such as a navigation bar for the entire site or for a subsection menu.
Article	The article element represents a section of content that is independent of a Web page or site content. It is self-contained and stands on its own. The possible sources for the article tag could be Blog post, News story, Comment, Review, and Forum post.

Instructions to the Trainer(s):

- Using Slide 6, explain various HTML5 semantic tags.

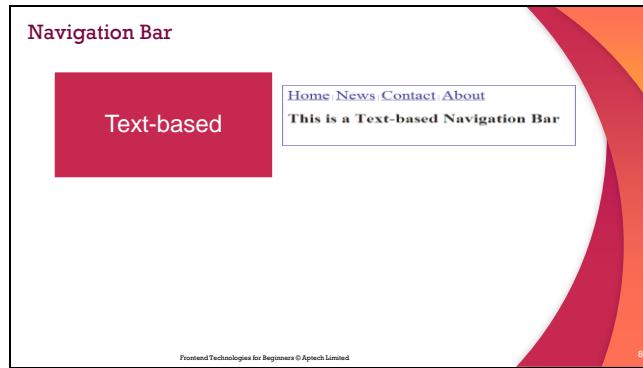
Slide 7

Text-level Semantic Tags	
Tag	Description
Mark	The <mark> tag is used for defining marked or highlighted text because of its relevance to the context. For example, a mark tag can be used for highlighting words in a Web page that a visitor searched for.
Time	The <time> tag is used for defining either the time, or a date in the Gregorian calendar. It is used optionally with a time and a time-zone offset. This element can be used to encode dates and times in a machine-readable format. For example, a Web user can add birthday reminders or scheduled events to the user's calendar and enable the search engines to produce better search results. Attributes and value of <time> tag are as follows: datetime: Provides the date/time given by the element's content pubdate: It is used for specifying publication date and time of the document
Meter	The <meter> tag displays markup or scalar measurement within a defined range. Absolute scalar values, such as height or weight, are not represented automatically by the meter tag. For this, the user must specify the height and weight within the known range of values. It is also used for displaying fractional value.
Progress	The <progress> tag can be used with JavaScript to display the progress of a task.

Instructions to the Trainer(s):

- Using Slide 7, explain certain text-level semantic tags.
- Tell the students that the tags provide meaning, structure, or style of a word or line.

Slide 8



Instructions to the Trainer(s):

- Using Slide 8, explain the importance of a navigation bar.
- Without navigation, users might not be able to move from page to page within a Website.
- You can commonly find navigational bars at the top of the browser and footer of the site. Sidebar navigation bar is also very common to see. Tell the students that they can find all these navigation bars in an ecommerce site.
- Tell the students that there are two types of navigation bars: text-based and graphical.
 - Text-based navigation bar reduces the loading time of a page.
 - However, it is less interactive and visually appealing to the visitor.
 - Graphical navigation bar gives a visual appeal to the site. Rollover effect can be added to this bar to make it interactive. The state of the image changes on mouse over. When the mouse is moved off the image, the image swaps back to the previous view.
- Explain the code from the Learner Guide that shows how to create a text-based navigation bar.

Additional Content:

Graphical Navigation Bar

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.

Graphical Navigation Bar with Rollover Effects

They are similar to the graphical navigation bar except for the additional feature. Moving the mouse over the linked image leads to a change in the state of image. In other words, the state change of image leads to an image swapping process. When the mouse is moved off the image, the image swaps back to the previous view. This rollover effect creates an interactive activity between the Website and the visitor.

This rollover effect has two different activities that include the image in the original view and the changed image after mouse rollover.

Note - The pre-load time of a Web page will increase if the rollover effect is used as all the images is required to be fully loaded on the Web page.

Following Code Snippet demonstrates CSS and HTML code for a graphical navigation bar.

```
<!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;
color:#616161;
text-shadow:1px 1px 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:#282828;
}
ul#navigation li a.first {
border-left: 0 none;
}
ul#navigation li a.last {
border-right: 0 none;
}
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;
/* 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 ul {
top: 0;
left: 181px; /* strong related to width:180px; from above */
}
ul#navigation ul li {
clear:both;
width:100%;
border:0 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;
border:0 none;
float:left;
clear:both;
width:150px;
}
</style>
</head>
<body>
<nav id="nav">
<ul id="navigation">
<li><a href=""><font size="4">Home</img></font></a></li>
<li><a href=""><font size="4">News</font></a></li>
<li><a href=""><font size="4">Contact</font></a></li>
<li><a href=""><font size="4">About</font></a></li>
</nav>
<br/>
<br/>
<br/>
<br/>
<h1>This is a Graphical Navigation Bar</h1>
</body>
</html>
```

Image Map

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.

Follow these guidelines to create an image map:

Use the `` tag to insert and link an image. In the `` tag, use the `usemap` attribute to define the image map name.

Use the `<map>` tag to create a map with the same name. Inside this `<map>` tag, define the clickable areas with the `<area>` tag.

Following Code Snippet demonstrates the use of 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>
```

In-class Question:

Question: Which navigation bar is suitable for visual appeal?

Answer: Graphical.

Slide 9

Divisions

- 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.

Frontend Technologies for Beginners © ApnaGhar Limited

9

Instructions to the Trainer(s):

- Using Slide 9, explain the `<div>` tag.

In-class Question:

Question: What type of element is the `<div>` tag?

Answer: Block-level element.

Slide 10

The slide has a title 'Divisions' at the top left. Below it is a code block:

```
.lCard{ width: 100px; height:100px; background-color:blue; padding: 6px; position:fixed; left:140px; top:100px; } .rCard{ width: 100px; background-color:red; padding: 7px; position:relative; top:93px; left:30px; } .bCard{ width: 100px; height:100px; background-color:green; padding: 6px; position:absolute; left:310px; bottom:320px; }
```

To the right of the code is a visual representation showing three colored boxes (red, blue, green) arranged in a 2x2 grid pattern. Below this is the text 'Output of Division Positioning'. At the bottom left of the slide is the text 'Frontend Technologies for Beginners © Agnieszka Limited'.

10

Instructions to the Trainer(s):

- Using Slide 10, explain the CSS code for <div> tags with different position properties.

Slide 11

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

Frontend Technologies for Beginners © Agnieszka Limited

11

Instructions to the Trainer(s):

- Ask students for some examples where they get to fill forms online. Discuss this and help them understand the purpose of forms.
- Using Slide 11, explain HTML forms.

In-class Question:

Question: What is the purpose of controls in forms?

Answer: The controls, such as radio buttons and text boxes provide a visual interface to the user to interact with them. A user provides data through these controls that is sent to the server for further processing.

Slide 12

Introduction to HTML5 Forms 2-3

Type	Description
<input type="text">	Used for a single-line text input field
<input type="radio">	Used for a radio button
<input type="checkbox">	Used for a check box
<input type="submit">	Used for a submit button
<input type="button">	Used for a clickable button

Common Input Types in HTML5

Frontend Technologies for Beginners © ApnaGhar Limited

Instructions to the Trainer(s):

- Using Slide 12, explain different input types of forms.
- The <form> element can hold different types of input elements, such as, check boxes and radio buttons.

Slide 13

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>
```

Frontend Technologies for Beginners © ApnaGhar Limited

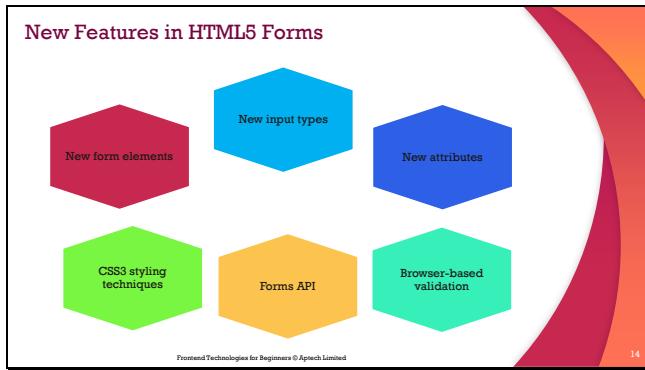
Output Showing HTML Form

13

Instructions to the Trainer(s):

- Using Slide 13, 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.

Slide 14



Instructions to the Trainer(s):

- Using Slide 14, list the new features in HTML5.
- Tell the students that all the new features help in creating a delightful user experience.

Slide 15

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

Frontend Technologies for Beginners © ApnaGhar Limited

15

Instructions to the Trainer(s):

- Using Slide 15, explain the new form elements.
- Explain the new form elements with example code snippet and outputs given on Slide 16.
- 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 following link:
<https://www.wideskills.com/html5-tutorial/03-new-form-elements-in-html5>

Slide 16

New Form Elements 2-2

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



Progress Element

Frontend Technologies for Beginners © Apache Limited

16

Instructions to the Trainer(s):

- Using Slide 16, 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.

Slide 17

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

Frontend Technologies for Beginners © Apache Limited

17

Instructions to the Trainer(s):

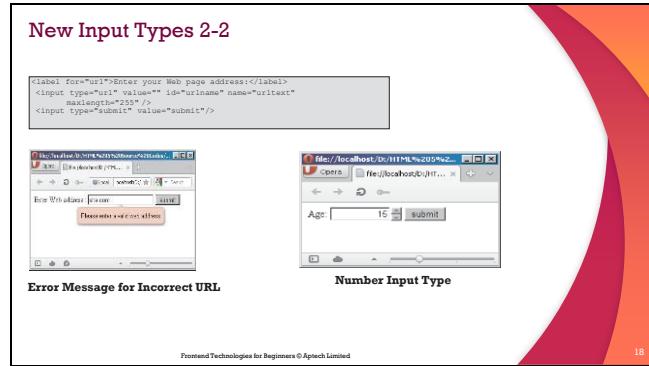
- Using Slide 17, mention that HTML 5 introduces several input types such as DateTime-local, time, week, and month.
- This is to improve the user experience and to make the forms more interactive.
- Use the Slide and explain the input types.
- Explain the new input types with some code snippet and outputs given on Slide 18.
- For more information about new elements, refer to following link: <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.

Slide 18



Instructions to the Trainer(s):

- Using Slide 18, mention that while validating the URL, browsers only check for entry with forward slash (/). For example, a URL such as **x://mynewsite.com** will be considered as valid, even though x:// is not a real protocol.

Slide 19

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

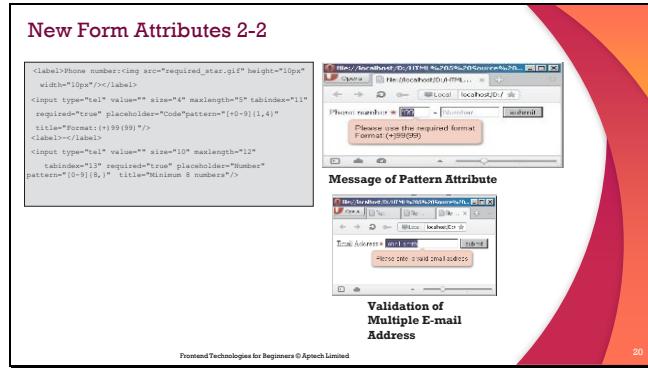
Frontend Technologies for Beginners © ApnaGhar Limited

19

Instructions to the Trainer(s):

- Using Slide 19, mention that the HTML `<input>` is one of the most important form elements.
- With new Input types and attributes, the `<input>` element becomes more dynamic.
- Use Slide 19 to explain certain new form attributes.
- Explain the new form attributes with example code snippet and outputs given on Slide 20.
- For more information about new form attributes, refer to following link:
<https://www.encodedna.com/html5/input-types-and-attributes-in-html5.htm>

Slide 20



Instructions to the Trainer(s):

- Using Slide 20, 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.

Slide 21

Browser-based Validation and CSS Styling Techniques

- New attributes such as `required` and `pattern` can be used with the input elements to perform validation.
- Web developers do 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.

Frontend Technologies for Beginners © ApnaGhar Limited

21

Instructions to the Trainer(s):

- Using Slide 21, explain browser-based validation and CSS styling techniques.
- Give example of CSS styling. Example: -input fields which cannot be left blank while submitting the form can be displayed with an outline.

Slide 22

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

Frontend Technologies for Beginners © ApnaGhar Limited

22

Instructions to the Trainer(s):

- Using Slide 22, mention that the new Forms API provides new methods, events, and properties to perform complex validations combining fields or calculations.
- Use the table on the Slide and explain the events and methods.

Slide 23

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

Frontend Technologies for Beginners © ApnaGuru Limited

23

Instructions to the Trainer(s):

- Using Slide 23, explain hidden elements.
- 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.

Slide 24

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>
```

Frontend Technologies for Beginners © Apache Limited

24

Instructions to the Trainer(s):

- Using Slide 24, 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.

Slide 25

The slide has a red and orange decorative header. The title 'Hidden Elements 3-3' is at the top left. Below it is a code block with CSS rules for a container, input-group, input, textarea, and label. To the right is a box labeled 'Data Sent to Server' containing the raw URL-encoded data: 'title=My+latest+trip+to+Bulgaria&post&content=+Hope+You+enjoy+it%0A++++&postId=23678'. At the bottom left is a small note: 'Frontend Technologies for Beginners © Apache Limited'.

```
.container{display:flex;justify-content:center;align-items:center;height:100vh;} .container form{position:fixed;bottom:2px;right:2px;border-radius:2px;background-color:white;outline:none;box-shadow:0 0 10px 0px black;}.input-group{margin-bottom:10px;display:flex;}.input,.textarea{flex:6;}.label{line-height:2;clear:2;}.textarea{height:60px;}
```

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

Data Sent to Server

Frontend Technologies for Beginners © Apache Limited

Instructions to the Trainer(s):

- Using Slide 25, explain the CSS for the HTML form and the form data that will be sent to the server.

Slide 26

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">

Frontend Technologies for Beginners © Apache Limited

26

Instructions to the Trainer(s):

- Using Slide 26, explain data attributes.
- For more information about data attributes, refer to following link:
<https://www.geeksforgeeks.org/html-data-attributes/>

In-class Question:

Question: Can the use of data attribute help users perform a narrower search on the Website?

Answer: Yes.

Slide 27

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.
- ❖ A data attribute helps store custom data or extra information on an HTML element.

27

Instructions to the Trainer(s):

Using Slide 27, summarize the session by reading out each point on the Slide.

Session 9: HTML5 Tables, Audio, and Video

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 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 familiarize 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

The slide features a title 'Objectives' at the top left. Below it is a bulleted list of 13 items, each starting with a bullet point. The background consists of overlapping red and orange curved shapes on a white base. At the bottom left is the text 'Frontend Technologies for Beginners © ApnaGuru Limited' and at the bottom right is the number '2'.

- 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 necessity 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

Frontend Technologies for Beginners © ApnaGuru Limited 2

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

Slides 3 and 4

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.

Fundamental Technologies for Beginners © Aptech Limited

3

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>
```



Fundamental Technologies for Beginners © Aptech Limited

4

Instructions to the Trainer(s):

- Using Slides 3 and 4, introduce students to 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.
- Then, 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.

Slide 5

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">
      <thead>
        <th>Name</th>
        <th>Age</th>
        <th>Place</th>
      </thead>
      <tbody>
        <tr>
          <td>Mark</td>
          <td>17</td>
          <td>Madrid</td>
        </tr>
        <tr>
          <td>John</td>
          <td>19</td>
          <td>London</td>
        </tr>
      </tbody>
    </table>
  </body>
</html>
```

Frontend Technologies for Beginners © ApnaGuru Limited



5

Instructions to the Trainer(s):

- Using Slide 5, 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.

Slides 6 and 7

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.

Frontend Technologies for Beginners © Aptech Limited

6

Colspan Attribute 2-2

```
<!DOCTYPE HTML>
<html>
<head>
<title>Employee Details</title>
</head>
<body>
<h2>Employee Details</h2>
<table border="1">
<thead>
<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>
</thead>
<tbody>
<tr>
<td>John</td>
<td>New York</td>
<td>John</td>
<td>London</td>
</tr>
<tr>
<td>Mike</td>
<td>New Jersey</td>
<td>Peter</td>
<td>Los Angeles</td>
</tr>
</tbody>
</table>
</body>
</html>
```

The Code Snippet demonstrates how to create a table and span header cells across two cells vertically.

Frontend Technologies for Beginners © Aptech Limited

7

Instructions to the Trainer(s):

- Using Slides 6 and 7, explain the `colspan` attribute.
- The user might want that the column 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.

Slides 8 to 10

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>
    </table>
  </body>
</html>
```

Frontend Technologies for Beginners © Aptech Limited

8

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>38.25</td>
</tr>
<tr>
  <td>30d</td>
  <td>47.5</td>
</tr>
</tbody>
</table>
</body>
</html>
```

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

Frontend Technologies for Beginners © Aptech Limited

9

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:

Manufacturer	Model	Price
Audi	A1	34.5
Audi	A4	34.5
Audi	A6	30.75
BMW	328i	38.25
	30d	47.5

Frontend Technologies for Beginners © Aptech Limited

10

Instructions to the Trainer(s):

- Using Slides 8 to 10, 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 10, explain the output for `rowspan` attribute.
- Explain the code that creates a table with a border width of one 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.

Slide 11

Horizontal Alignment

- 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.
- 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.

Frontend Technologies for Beginners © ApnaGhar Limited

11

Instructions to the Trainer(s):

- Using Slide 11, 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

- 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 following syntax:

```
<td style="text-align: center; vertical-align: middle"> Aptech  
Web site </a>
```

Frontend Technologies for Beginners © Aptech Limited

12

Instructions to the Trainer(s):

- Using Slide 12, 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 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.

Slide 13

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">`

Frontend Technologies for Beginners © Agnieszka Limited

13

Instructions to the Trainer(s):

- Using Slide 13, 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">`

Slide 14

Caption Element

- 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.

Frontend Technologies for Beginners © Agnieszka Limited

14

Instructions to the Trainer(s):

- Using Slide 14, 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 given in the Learner Guide for specifying a heading for a table using `caption` element.
- The code creates a table of border width of one 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>
```



Frontend Technologies for Beginners © Apache Limited

15

Instructions to the Trainer(s):

- Using Slide 15, 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.

Slide 16

Merging Table Cells

- 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.

Frontend Technologies for Beginners © ApnaGhar Limited

16

Instructions to the Trainer(s):

- Using Slide 16, 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.

Slides 17 and 18

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.

Frontend Technologies for Beginners © Agnieszka Limited
17

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.

Frontend Technologies for Beginners © Agnieszka Limited
18

Instructions to the Trainer(s):

- Using Slides 17 and 18, 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 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.

Slide 19

Tables for Page Layout

- 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 popups for providing information to their customers.

Frontend Technologies for Beginners © Agnieszka Limited

19

Instructions to the Trainer(s):

- Using Slide 19, 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.

Slide 20

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.

Frontend Technologies for Beginners © ApnaGhar Limited

20

Instructions to the Trainer(s):

- Using Slide 20, 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.
- Explain multimedia in HTML5 by beginning with definition of multimedia. Say that it 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.

Slide 21

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 common audio and video formats:

Container	Video Codec	Audio Codec
Mp4	H.264	AAC
Ogg	Theora	Vorbis
WebM	VP8	Vorbis

Frontend Technologies for Beginners © ApnaGuru Limited

21

Instructions to the Trainer(s):

- Using Slide 21, 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.

Slide 22

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

Frontend Technologies for Beginners © Apache Limited

22

Instructions to the Trainer(s):

- Using Slide 22, 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.

Slide 23

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

Frontend Technologies for Beginners © ApnaGuru Limited

23

Instructions to the Trainer(s):

- Using Slide 23, 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.

Frontend Technologies for Beginners © ApnaGuru Limited

24

Instructions to the Trainer(s):

- Using Slide 24, 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`.

Slide 25

The slide has a decorative background with a red-to-orange gradient and a white circular shape on the right.

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 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

Frontend Technologies for Beginners © ApnaGuru Limited

25

Instructions to the Trainer(s):

- Using Slide 25, 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 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.

Slide 26

Creating Audio Files

- To play the audio in older browsers then the `<embed>` tag will be used.
- `<embed>` tag has two attributes, `src` and `autostart`.
- `src` attribute is used to specify the source of the audio.
- `autostart` 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 autostart loop>
      <source src="sampleaudio.mp3">
      <source src="sampleaudio.ogg">
      <embed src="sampleaudio.mp3">
    </audio>
  </body>
</html>
```

Frontend Technologies for Beginners © Agnieszka Limited

Instructions to the Trainer(s):

- Using Slide 26, 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 `autostart`.
- The `src` attribute is used to specify the source of the audio and the `autostart` 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 `autostart` 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`.

Slide 27

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>
```



Frontend Technologies for Beginners © Apache Limited

27

Instructions to the Trainer(s):

- Using Slide 27, 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.

Slide 28

Video Tag Attributes

HTML5 specification provides a list of attributes that can be used with the `<video>` element.

HTML5 has following attributes for the `<video>` element.

- Following table lists some of the `<video>` tag attributes:

Video Attributes	Description
<code>autoplay</code>	Specifies that the browser will start playing the video as soon as it is ready
<code>muted</code>	Allows to mute the video initially, if this attribute is existing
<code>controls</code>	Allows displaying the controls of the video, if the attribute exists
<code>loop</code>	Specifies that the browser should repeat playing the existing video once more if the <code>loop</code> attribute exists and accepts a boolean value
<code>preload</code>	Specifies whether the video should be loaded or not when the page is loaded
<code>src</code>	Specifies the location of the video file to be embedded

Frontend Technologies for Beginners © Aptech Limited

28

Instructions to the Trainer(s):

- Using Slide 28, 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`.

Slides 29 and 30

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 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.

Frontend Technologies for Beginners © Aptech Limited

29

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>
```

Frontend Technologies for Beginners © Aptech Limited

30

Instructions to the Trainer(s):

- Using Slides 29 and 30, 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 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 30, 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 30 displays the effect of both the values `none` and `metadata` on the Web page.

Slide 31

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 Code\\movie.mp4" controls preload="auto" width="360" height="340">
Your browser does not support the video.
</video>
</body>
</html>
```

Frontend Technologies for Beginners © Apache Limited



31

Instructions to the Trainer(s):

- Using Slide 31, 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.

Frontend Technologies for Beginners © ApnaGhar Limited

32

Instructions to the Trainer(s):

- Using Slide 32, 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 convert 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 convert video files to the WebM format.

Slides 33 and 34

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.

Frontend Technologies for Beginners © Agnieszka Limited

33

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.

Frontend Technologies for Beginners © Agnieszka Limited

34

Instructions to the Trainer(s):

- Using Slides 33 and 34, 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.

Slides 35 and 36

Track Element 1-2

- 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+.

Frontend Technologies for Beginners © Apneach Limited

35

Track Element 2-2

- Following table lists the track element attributes:

Container	Description
src	Contains the URL of the text track data
srclang	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

Frontend Technologies for Beginners © Apneach Limited

36

Instructions to the Trainer(s):

- Using Slides 35 and 36, 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 36 that lists the attributes of the `track` element.
- 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.

Slide 37

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.

Frontend Technologies for Beginners © Agnieszka Limited

37

Instructions to the Trainer(s):

- Using Slide 37, explain the accessibility support for audio and video element in various browsers.
- Mention that the lower versions of these browsers do not provide support for multimedia elements.

Slide 38

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.

Frontend Technologies for Beginners © ApnaGhar Limited

38

Instructions to the Trainer(s):

- Using Slide 38, 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.

Slide 39



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.

39

Instructions to the Trainer(s):

- Show students Slide 39.
- Summarize the session by reading out each point on the Slide.

Session 10: JavaScript - I

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 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

The slide features a red header with the word "Objectives". Below the header is a bulleted list of 25 items, each starting with a bullet point. The list includes topics such as scripting, the JavaScript language, client-side and server-side JavaScript, data types, escape sequences, events, and various loop constructs. At the bottom of the slide, there is a small footer with the text "Frontend Technologies for Beginners © ApnaGuru Limited" and the number "2".

- Explain scripting
- Explain the JavaScript language
- Explain the client-side and server-side JavaScript
- List the primitive data types in JavaScript
- Describe the JavaScript methods to display information
- Explain escape sequences and built-in functions in JavaScript
- Explain events and event handling
- Explain jQuery
- Explain operators and their types in JavaScript
- Explain regular expressions in JavaScript
- Explain decision-making statements in JavaScript
- Explain while loop
- Explain for loop
- Explain do...while loop
- Explain break and continue statement
- Explain single-dimensional arrays
- Explain multi-dimensional arrays
- Explain for..in loop

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

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).

Frontend Technologies for Beginners © ApnaGuru 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 variables, data types, built-in functions, and methods.
- Explain a simple example on JavaScript coding as given in the Learner Guide.
- Then, define scripting language. 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.
- Then, 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).

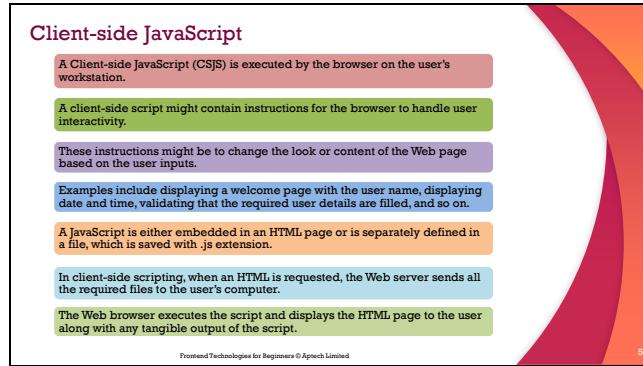
Slide 4

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
1	ECMAScript 1 (1997)
2	ECMAScript 2 (1998)
3	ECMAScript 3 (1999)
5	ECMAScript 5 (2009)
6	ECMAScript 2015
7	ECMAScript 2016
8	ECMAScript 2017
9	ECMAScript 2018
10	ECMAScript 2019
11	ECMAScript 2020
12	ECMAScript 2021

Instructions to the Trainer(s):

- Using Slide 4, 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.

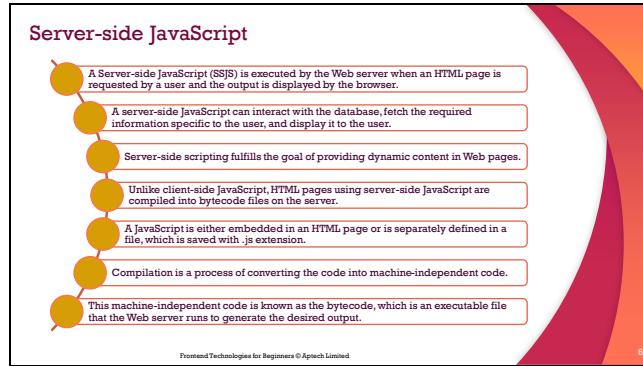
Slide 5



Instructions to the Trainer(s):

- Using Slide 5, 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.

Slide 6



Instructions to the Trainer(s):

- Using Slide 6, 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 fulfils the goal of providing dynamic content in Web pages.

Slide 7

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>
```

These 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.

Frontend Technologies for Beginners © ApnaGhar Limited

Instructions to the Trainer(s):

- Using Slide 7, 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 that 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:

Q: 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.

Slide 8

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.
```

Frontend Technologies for Beginners © ApnaGhar Limited

Instructions to the Trainer(s):

- Using Slide 8, 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.
- 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.
- 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.

Slides 9 and 10

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, -9, 7.5e3, 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

Frontend Technologies for Beginners © Aptech Limited

9

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

Frontend Technologies for Beginners © Aptech Limited

10

Instructions to the Trainer(s):

- Using Slides 9 and 10, 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.
- Explain different types of primitive data types. Explain, table which lists the primitive data types.

- 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 the table on Slide 10 which lists the composite data types.

Slide 11

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.

Frontend Technologies for Beginners © Agnieszka Limited

11

Instructions to the Trainer(s):

- Using Slide 11, 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.

Slide 12

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;`

Frontend Technologies for Beginners © ApnaGhar Limited

12

Instructions to the Trainer(s):

- Using Slide 12, 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

Slide 13

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
\xa	Matches a Latin-1 encoding character using hexadecimal representation, where aa are two hexadecimal numbers. For example, \d31 represents the character 'a'
\uaaaa	Represents the Unicode encoding character, where aaaaa are four hexadecimal numbers. For example, the character \u0020 represents a space

Instructions to the Trainer(s):

- Using Slide 13, 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 subsequent 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.

Slide 14

Built-in Functions		
Function	Description	Example
<code>alert()</code>	Displays a dialog box with some information and OK button.	<code>alert("Please fill all the fields of the form");</code> Displays a message box with the instruction
<code>confirm()</code>	Displays a dialog box with OK and Cancel buttons. It verifies an action, which a user wants to perform.	<code>confirm("Are you sure you want to close the page?")</code> Displays a message box with the question
<code>parseInt()</code>	Converts a string value into a numeric value.	<code>parseInt("25 years");</code>
<code>parseFloat()</code>	Converts a string into a number with decimal point.	<code>parseFloat("10.33");</code> Returns 10.33
<code>eval()</code>	Evaluates an expression and returns the evaluated result.	<code>eval("2*2");</code> Returns 4
<code>isNaN()</code>	Checks whether a value is not a number.	<code>isNaN("Hello");</code> Returns true
<code>prompt()</code>	Displays a dialog box that accepts an input value through a text box. It also accepts the default value for the text box.	<code>prompt("Enter your name", "Name");</code> Displays the message in the dialog box and Name in the text box

Frontend Technologies for Beginners © ApnaGhar Limited

14

Instructions to the Trainer(s):

- Using Slide 14, 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.

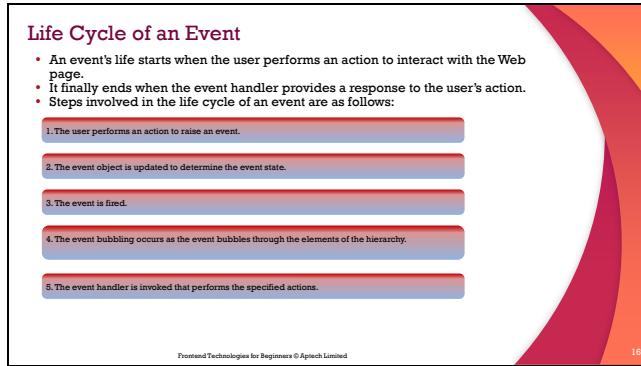
Slide 15

The diagram is titled "Event Handling and Event Bubbling". It features a central white circle with a red border containing the text: "Event handling is a process of specifying actions to be performed when an event occurs. This is done by using an event handler." Below this are four colored boxes: green (event handler), purple (event object), blue (event bubbling), and yellow (event bubbling details). The green box contains: "An event handler is a scripting code or a function that defines the actions to be performed when the event is triggered." The purple box contains: "When an event occurs, an event handler function that is associated with the specific event is invoked." The blue box contains: "Event bubbling is a mechanism that allows a user to specify a common event handler for all child elements." The yellow box contains: "This means that the parent element handles all the events generated by the child elements." At the bottom, a pink box states: "For example, consider a Web page that consists of a paragraph and a table. The paragraph consists of multiple occurrences of italic text." The footer of the slide reads: "Frontend Technologies for Beginners © Apache Limited" and "15".

Instructions to the Trainer(s):

- Using Slide 15, 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 that 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.
- 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.
- Explain concept of event bubbling.
- Explain the event bubbling mechanism in JavaScript along with the figure displaying the event bubbling.

Slide 16



Instructions to the Trainer(s):

- Using Slide 16, 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.

Slide 17

Keyboard and Mouse 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.

- 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

Frontend Technologies for Beginners © ApnaGhar Limited

17

Instructions to the Trainer(s):

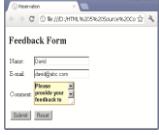
- Using Slide 17, explain different keyboard events.
- Keyboard events are the events that occur when a key or a combination of keys are pressed or released from a keyboard. These events occur for all keys of a keyboard.
- Different keyboard events are as follows:
 - Onkeydown : Occurs when a key is pressed down.
 - Onkeyup : Occurs when the key is released.
 - Onkeypress : Occurs when a key is pressed and released.
- Explain the mouse events.
- Mouse events occur when the user clicks the mouse button.
- Explain table which lists the mouse events.

Slide 18

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



Frontend Technologies for Beginners © ApnaGhar Limited

18

Instructions to the Trainer(s):

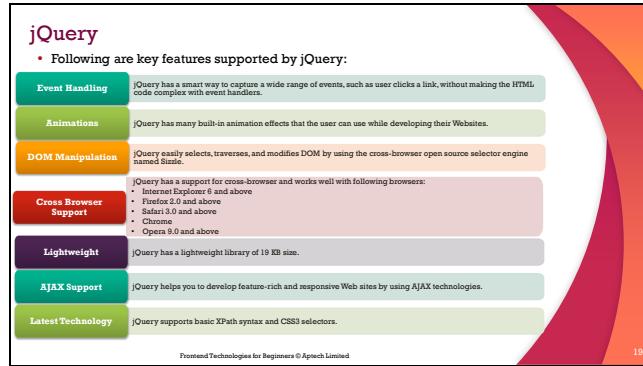
- Using Slide 18, explain different focus and selection events.
- The focus events determine the activation of various elements that uses the input element. It allows you to set or reset focus for different input elements.
- The selection events occur when an element or a part of an element within a 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.

Slide 19



Instructions to the Trainer(s):

- Using Slide 19, 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.
- Explain the how to use JQuery library. Say that there is an easy way to use jQuery library. To work with jQuery, one can perform following steps:
 - Download the jQuery library from the <http://jquery.com/> Website
 - 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 in the Learner Guide shows how to use a jQuery library.
- 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.
- Mention about jQuery mobile saying that it used to be a Web User Interface (UI) development framework that allowed the user to build mobile Web applications that works on tablets and smartphones. The jQuery mobile framework provided many facilities that included XML, DOM, and HTML manipulation and traversing, performing

server communication, handling events, image effects, and animation for Web pages. However, the library is deprecated now and not recommended for use.

Slide 20

Basics of JavaScript 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.

Presented Technologies for Beginners © Aptech Limited

20

Instructions to the Trainer(s):

- Using Slide 20, 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.
- 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 $age >= 18 ? \text{"Eligible"} : \text{"Not Eligible"}$.
- 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

Slide 21

The slide has a decorative background with a red-to-orange gradient and a white circular graphic on the right. At the top, the title 'Arithmetic Operators' is displayed in a pink font. Below the title are three green callout boxes with black borders: the first says 'Are binary operators.', the second says 'Perform basic arithmetic operations on two operands.', and the third says 'Operator appears in between the two operands, which allow you to perform computations on numeric and string values.' A bulleted list follows: '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

Frontend Technologies for Beginners © ApnaGuru Limited

Instructions to the Trainer(s):

- Using Slide 21, 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 the table which lists the arithmetic operators with their descriptions and an example of each type.

In-Class Question:

Q: What is the modulo (%) operator used for?

Answer: Modulo (%) operator is for finding remainder in a division operation.

Slide 22

The slide has a decorative background with a red-to-orange gradient on the right side. The title 'Increment and Decrement Operators' is at the top. Below it are several colored boxes containing text:

- Red box: 'Increment and decrement operators are unary operators.'
- Green box: 'Increment operator (++) increases the value by 1, while the decrement operator (--) decreases the value by 1.'
- Purple box: 'These operators can be placed either before or after the operand.'
- Blue box: '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.'

A bulleted list follows:

- Following table lists increment and decrement operators:

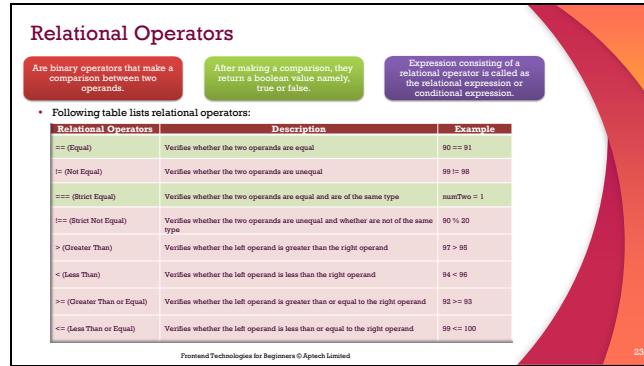
Expressions	Type	Result
numTwo = ++numOne;	Pre-increment	numTwo = 3
numTwo = numOne++;	Post-increment	numTwo = 2
numTwo = --numOne;	Pre-decrement	numTwo = 1
numTwo = numOne--;	Post-decrement	90 % 20

Frontend Technologies for Beginners © ApnaGhar Limited

Instructions to the Trainer(s):

- Using Slide 22, 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.

Slide 23



The slide contains three callout boxes at the top:

- A red box: Are binary operators that make a comparison between two operands.
- A green box: After making a comparison, they return a boolean value namely, true or false.
- A purple box: 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>> (Greater Than)</code>	Verifies whether the left operand is greater than the right operand	<code>97 > 95</code>
<code>< (Less Than)</code>	Verifies whether the left operand is less than the right operand	<code>94 < 96</code>
<code>>= (Greater Than or Equal)</code>	Verifies whether the left operand is greater than or equal to the right operand	<code>92 >= 93</code>
<code><= (Less Than or Equal)</code>	Verifies whether the left operand is less than or equal to the right operand	<code>99 <= 100</code>

Frontend Technologies for Beginners © ApnaGuru Limited

23

Instructions to the Trainer(s):

- Using Slide 23, 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.

Slide 24

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

Frontend Technologies for Beginners © ApnaGuru Limited

24

Instructions to the Trainer(s):

- Using Slide 24, 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.

Slide 25

The slide has a decorative background with a red-to-yellow gradient curve on the right side. At the bottom left, there's a small watermark-like text: 'Frontend Technologies for Beginners © ApnaGhar Limited'.

Assignment Operators

Assignment operators assign the value of the right operand to the operand on the left 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

Frontend Technologies for Beginners © ApnaGhar Limited

Instructions to the Trainer(s):

- Using Slide 25, explain the assignment operators.
- Assignment operators assign the value of the right operand to the operand on the left 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 .

Slide 26

The slide has a decorative background with a red-to-orange gradient on the right side. At the top, the title 'Bitwise Operators' is centered. Below the title are three green callout boxes with white text: 'Represent their operands in bits (zeros and ones) and perform operations on them.', 'They return standard decimal values.', and 'Compound assignment operator - Is formed by combining the simple assignment operator with the arithmetic operators.' A bullet point states: '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	01110000 Returns 00111100
^a (Bitwise XOR)	Compares two bits and returns 1 if the corresponding bit of either, but not both the operands is 1	a ^ b a = 5 (00000111) b = 3 (00000111)

Frontend Technologies for Beginners © ApnaGhar Limited

26

Instructions to the Trainer(s):

- Using Slide 26, 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.

Slide 27

The slide has a decorative background with a red and orange gradient on the right side.

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.

Frontend Technologies for Beginners © ApnaGhar Limited

27

Instructions to the Trainer(s):

- Using Slide 27, 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.

Slide 28

Operator Precedence			
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 Modulus	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

Instructions to the Trainer(s):

- Using Slide 28, 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.
`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.
`var variable_name = new RegExp("regular_expression_pattern","flag");`

Frontend Technologies for Beginners © ApnaGuru Limited

29

Instructions to the Trainer(s):

- Using Slide 29, 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.

Slide 30

The slide has a decorative background with a red-to-orange gradient and a white header bar. The title 'RegExp Methods and Properties' is at the top. Below it is a bulleted list: '• RegExp object supports methods that are used for searching the pattern in a string. They are as follows:' followed by two items: 'test(string)' and 'exec(string)'. A table below lists five attributes with their descriptions:

Attribute	Description
\$n	Represents the number from 1 to 9. It stores the recently handled parts of a parenthesized pattern of a regular expression.
aif	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.
aifc	Indicates whether the given regular expression contains an i flag.
aiff	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.
asc	Stores the copy of the pattern.

At the bottom right of the slide is the number '30'.

Instructions to the Trainer(s):

- Using Slide 30, 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.
- Explain that there are five categories of pattern matching. These categories are as follows:
 - Position Matching
 - Character Classes
 - Repetition
 - Alternation and Grouping
 - Back Reference

Slide 31

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
^	Denotes the start of a string	/^Good/ matches "Good" in "Good night", but not in "A Good Eyesight"
\$	Denotes the end of a string	/art\$/ matches "art" in "Cart" but not in "artist"
\b	Matches a word boundary. A word boundary includes the position between a word and the space	/ry\b/ matches "ry" in "She is very good"
\B	Matches a non-word boundary	/^Ba\b/ matches "an" in "operand" but not in "anomaly"

Frontend Technologies for Beginners © ApnaGhar Limited

31

Instructions to the Trainer(s):

- Using Slide 31, 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.

Slide 32

The slide has a decorative background with a red and orange gradient on the right side. At the top, there's a purple box containing the text: "Characters or symbols in this category are combined to form character classes for specifying patterns." Below it is a blue box with the text: "These classes are formed by placing a set of characters within the square brackets." A bullet point states: "Following table lists various character classes symbols:". The table is as follows:

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 night!"
[^xyz]	Matches one of the characters not specified within the character set	/*[^G]*/ Matches "RIT", but, not "KIT" or "CIT"
.	Denotes a character except for the new line and line terminator	/.*/ Matches "sat", "at", "set", and so on
\w	Matches alphabets and digits along with the underscore	/\w/ Matches "600" in "600%"
\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*/ Matches " " in "scroll bar"
\S	Searches a non-space character	/\S*/ Matches "scroll" in "scroll bar"

Frontend Technologies for Beginners © ApnaGang Limited

32

Instructions to the Trainer(s):

- Using Slide 32, 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.

Slide 33

The slide has a purple header bar with the title 'Repetition'. Below it is a purple callout box containing the text: 'Characters or symbols in this category allow matching characters that reappear frequently in a string.' A bulleted list follows: '• 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	/\w{4,5}/ Matches minimum four whitespace characters
(x,y)	Matches minimum x to maximum y occurrences of a regular expression	/\w{6,8}/ Matches minimum six to maximum eight digits
?	Matches minimum zero to maximum one occurrences of a regular expression	/\w?m/ Matches "im" or "i m"
*	Matches minimum zero to multiple occurrences of a regular expression	/im*/ Matches "i" in "ice" and "imm" in "immaculate", but nothing in "good"

Frontend Technologies for Beginners © ApnaGhar Limited

33

Instructions to the Trainer(s):

- Using Slide 33, 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.

Slide 34

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
0	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"

Frontend Technologies for Beginners © ApnaGhar Limited

34

Instructions to the Trainer(s):

- Using Slide 34, 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.

Slide 35

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
\(\)\n	Matches a parenthesized set within the pattern, where n is the number of the parenthesized set to the left	/(\w+)\n+\1/

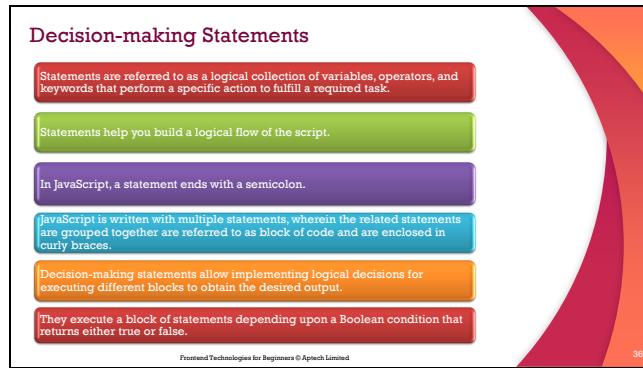
Frontend Technologies for Beginners © ApnaGhar Limited

35

Instructions to the Trainer(s):

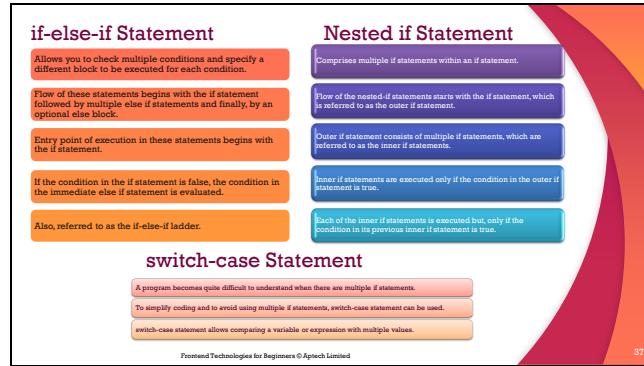
- Using Slide 35, explain the back references.
- Characters or symbols in this category allow referring back to a sub-expression in the same regular expression.
- This is useful when matching the remaining sub-expression of a regular expression is based upon the result of matching the previous sub-expression.
- Explain table which lists the back reference matching symbol.

Slide 36



Instructions to the Trainer(s):

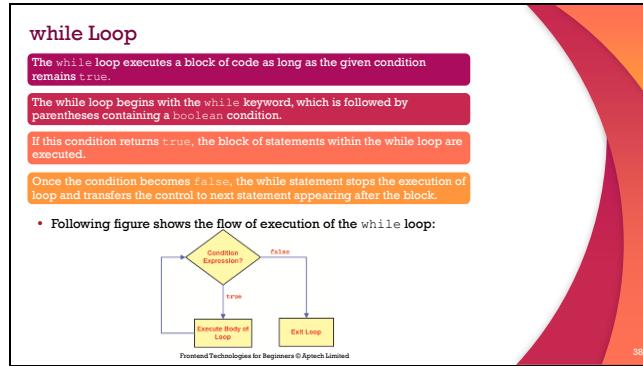
- Using Slide 36, explain the decision-making statements in JavaScript.
- Statements are referred to as a logical collection of variables, operators, and keywords that perform a specific action to fulfil a required task.
- For example, the line of code that declares a variable is a statement. Statements 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.



Instructions to the Trainer(s):

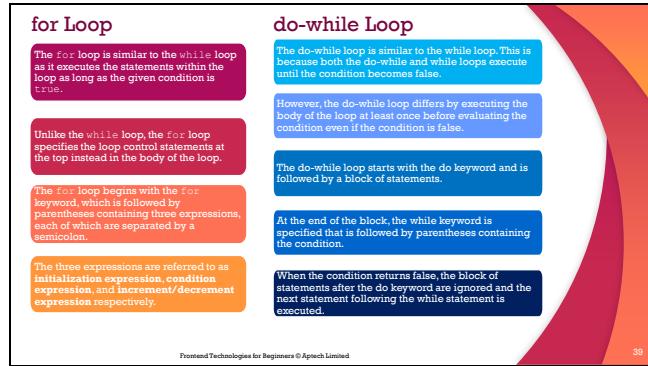
- Using Slide 37, explain the if-else-if and nested if statement.
- The if-else-if statements allow you to check multiple conditions and specify a different block to be executed for each condition. The flow of these statements begins with the if statement followed by multiple else if statements and finally 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.
- 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.

Slide 38



Instructions to the Trainer(s):

- Using Slide 38, 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.
- 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.



Instructions to the Trainer(s):

- Using Slide 39, 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.

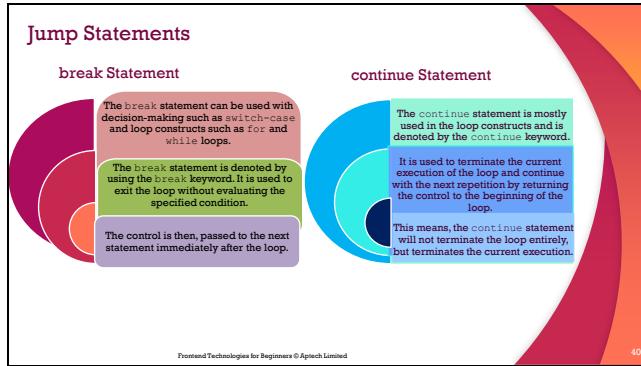
do-while loop:

- 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.



Instructions to the Trainer(s):

- Using Slide 40, 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.

Slide 41

The slide is titled 'Arrays' and contains two main sections: 'Single-dimensional Array' and 'Multi-dimensional Array'.

Single-dimensional Array: An array is a collection of values stored in adjacent memory locations. These values can be accessed by using subscript or index numbers. JavaScript supports two types of arrays:

- Single-dimensional array
- Multi-dimensional array

In a single-dimensional array, the elements are stored in a single row in the allocated memory. A diagram shows an array of six elements: Michael, David, Steve, Richard, Katherine, and Kelly. Brackets above the array are labeled 'Subscripts' (0, 1, 2, 3, 4, 5) and brackets below are labeled 'Elements'.

Employee Salary	0	1	2	3	4	5
Michael	14150	10500	1500	26550		
David	34350	40500	1000	36400		
Steve	15500	10500	1500	17000		
Richard	4900	4000	2250	11150		
Katherine	32300	9000	2000	23300		
Kelly						

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.

Frontend Technologies for Beginners © Agnieszka Limited

Instructions to the Trainer(s):

- Using Slide 41, 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.

Slide 42

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.

Frontend Technologies for Beginners © ApnaGhar Limited

Instructions to the Trainer(s):

- Using Slide 42, 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.

Slide 43

The slide is titled "for..in Loop". It contains the following text:
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.

A screenshot of a browser window titled "for..in Loop" shows a list of books: "List of Books", "Beginning CSS 3.0", "Introducing HTML5", and "HTML5 in Mobile Development".

Frontend Technologies for Beginners © Apache Limited

Instructions to the Trainer(s):

- Using Slide 43, explain the `for..in` loop.
- Mention that `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.

Slide 44

Summary

- ❖ JavaScript is a scripting language, which can be executed on the client-side and on the server-side.
- ❖ 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 and event bubbling is a mechanism that allows you to specify a common event handler for all child elements.
- ❖ JavaScript operators are classified into six categories based on the type of action they perform on operands.
- ❖ There are six categories of operators namely, Arithmetic, Relational, Logical, Assignment, Bitwise, and Special operators.
- ❖ 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.

44

Instructions to the Trainer(s):

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

Session 11: JavaScript - II

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

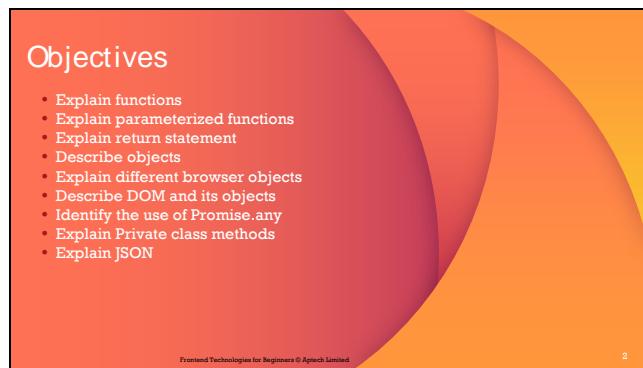
To teach this session, you should be well versed with functions, DOM, the new features of JavaScript, and how JavaScript handles large data.

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 on Slide 2.

11.2 In-Class Explanations

Slides 3 and 4

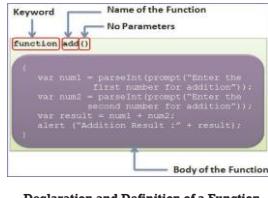
Functions 1-3

- 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 must be invoked.
 - To invoke a function, specify the function name followed by parenthesis outside the function block.

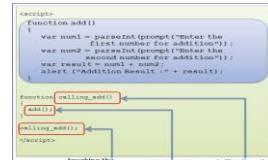
Frontend Technologies for Beginners © Aptech Limited

3

Functions 2-3



Declaration and Definition of a Function



Invoking of Function

Frontend Technologies for Beginners © Aptech Limited

4

Instructions to the Trainer(s):

- Ask students if they know of any popular applications built using JavaScript.
- Mention that:
 - While HTML and CSS give structure and style to Web pages, JavaScript gives interactive elements that engage a user.
 - Common examples of JavaScript include the search box on Amazon and interactive maps.
- Using Slides 3 and 4, explain functions.

Consider a scenario where a Web page has been designed to greet the user with his/her name on the click of a button. A code can be used here to accomplish this task, but may result in the same output on repetitive execution. However, writing these statements each time for the same action is tedious, time consuming, and error prone. To make the code more task-oriented and manageable, JavaScript allows to group statements before they are actually invoked. This can be achieved by using the concept of functions.

A function is a reusable block of code that is executed on the occurrence of an event. The event can be a user action on the page or a call within the script.

- Tell the students that:
 - Functions are one of the important building blocks in JavaScript.
 - A function in JavaScript is similar to a set of statements that performs a task.
 - To use a function, it must be defined somewhere in the scope from which it must be called.
 - A function might take parameters, which are variables or values on which it performs operations. After performing operations, a function might return the resultant value to display it in the browser. For example, a function named add() might take two numbers on which the addition operation will be performed and will return the result of addition.
 - A JavaScript function is always created under the script element. JavaScript supports both user-defined and built-in functions.
- Some of the places in the code where a function can be invoked:
 - On an event. For example, on the click of a button, a function can be invoked.
 - A function can invoke another function in JavaScript code.
 - A function can be invoked itself.
- Mention, 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 (). If the function does not take any parameters, then it must be specified with the empty parenthesis.
- Once the function is declared, you must define the function by specifying the operations or instructions within the curly braces {and}. These curly braces indicate the start and end of the function block, which is collectively referred to as the body of the function.
- There are certain conventions that must be followed for naming functions. They are as follows:
 - Can consist of letter, digits, and underscore
 - Can begin only with a letter or an underscore
 - Cannot be a JavaScript keyword
 - Cannot begin with a digit
 - Cannot contain spaces
- Explain the syntax of creating a function.
- A function must be defined, before it can be invoked in the script. Also, there can be multiple functions defined within the script element.
- Explain the figure shows the declaration and definition of a function.
- 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.
- A function can be defined and invoked even in an external JavaScript file. Also, a function can be called from another function in JavaScript. The function that invokes another function is called the **calling** function; whereas the function that is called is referred to as the **called** function.
- Functions provide the benefit of code reusability by allowing the user to call a function multiple times.
- For more information about functions, refer to following link:
<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Functions>

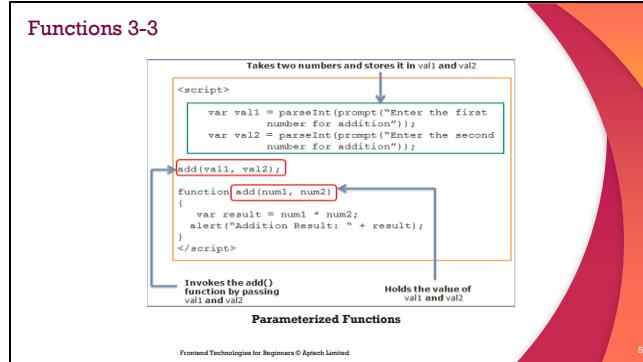
In-class Questions:

Question: What should be done after a function is declared?

Answer: Once the function is declared, define the function by specifying the operations or instructions within the curly braces.

Question: How to define a function?

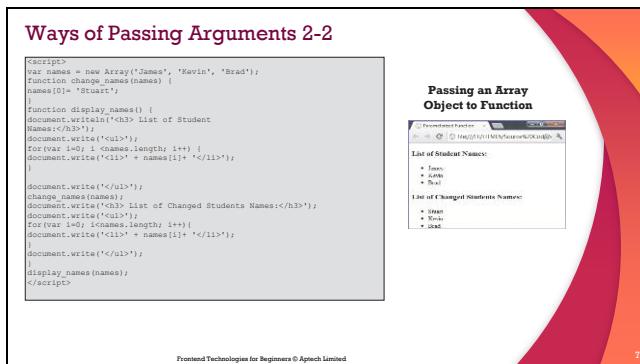
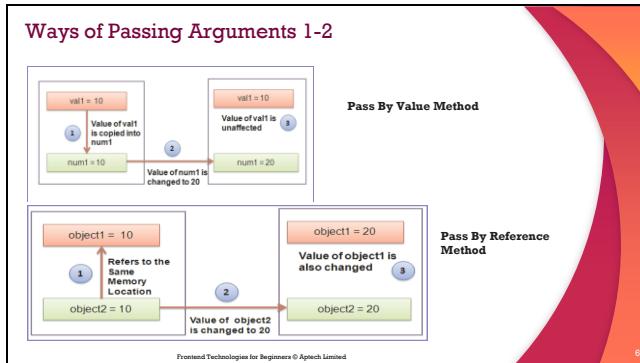
Answer: A function can be defined by specifying the operations or instructions within the curly braces {and}.



Instructions to the Trainer(s):

- Using Slide 5, explain parameterized functions.
- Parameterized functions take parameters to hold values.
- Function perform operations using these values.
- 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.
- Figure on Slide 5 shows the parameterized functions. The num1 and num2 parameters will hold the values of val1 and val2 arguments to perform the operations.
- The num1 and num2 parameters are only accessible within the function. The parameters of a function are variables that are declared in the function declaration. Here, num1 and num2 are the parameters of the function.
- Similarly, val1 and val2 are the arguments whose values are passed to the parameters, num1 and num2, while invoking the function. Both the set of variables, that is, arguments and parameters of the function will occupy different memory space.
- Alternatively, one can use same variable names for arguments and parameters while creating and invoking functions. In either of the case, the variables will occupy different memory space.

Slides 6 and 7



Instructions to the Trainer(s):

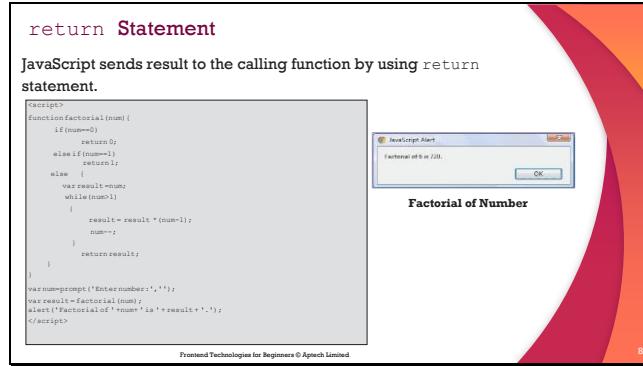
- Using Slides 6 and 7, explain the two ways of passing arguments.
- There are two ways of passing arguments to a function namely, pass by value and pass by reference. The description about these is as follows:
 - **Passing by value** - Refers to passing variables as arguments to a function. In the pass 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.
- Explain code snippet on Slide 7 which demonstrates the code to pass Array object as a parameter to a function.
- In the code, the function `change_names (names)` takes the `names` array as parameter. It changes the value at the 0th position in the array. The function is further invoked in the `display_names()` function. The `display_names ()` function displays the values from the array before and after the value is changed at the 0th position in the array.

- For more information about passing arguments, refer to following link:
<https://www.dummies.com/web-design-development/javascript/how-to-pass-and-use-arguments-to-code-with-javascript/>

In-class Question:

Question: What does the called function do in the pass by reference method?

Answer: 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.



Instructions to the Trainer(s):

- Using Slide 8, explain the use of 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.
- Similarly, the return statement can be used to return a collection of values stored in arrays.
- For more information about return statement, refer to following link:
<https://www.freecodecamp.org/news/javascript-return-statements/>

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.

Object: Car



Properties	Make : ford Color : green Model : focus Run()
Methods	

Object: Bird



Properties	Type : pigeon Color : gray Wings : two
Methods	Eat() Fly()

Frontend Technologies for Beginners © 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.
- JavaScript provides built-in objects and allows creating user-defined objects. The description of the object is as follows:
 - **Built-in Objects** - Are pre-defined objects which are already defined. Their properties and methods must be called to fulfil a task. An example of a pre-defined object is the Array object.
 - **Custom Objects** - Are user-defined objects, which the developer explicitly creates in the script and defines their properties and methods. For example, to store the doctor details, such as name, age, hospital name, and so on, an object named doctor can be created.

Tips:

Object are mutable: They are addressed by reference, not by value.

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:

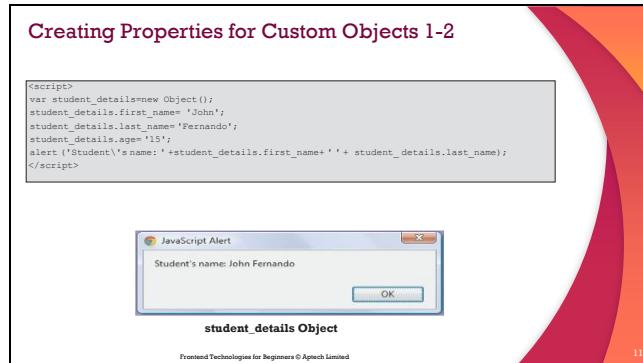
```
<script>
//create an object using direct method
var doc = detailsnewObject();
//create an object using new keyword
studOne = newstudent_info ('James', '23', 'New Jersey');
</script>
```

Frontend Technologies for Beginners © Aptech Limited

10

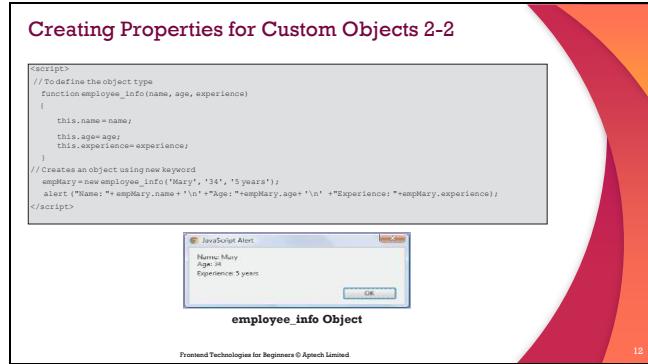
Instructions to the Trainer(s):

- Using Slide 10, explain how to create custom objects.
- The object is the parent object from which all JavaScript objects are derived.
- The new keyword is used to get custom objects.
- There are two methods to create a custom object:
 - An object can be created using the built-in Object object. This is called the **direct** method.
 - An object can be created by defining a template and initializing it with the new keyword.
- Explain the syntax for both.
- Explain the code snippet that creates objects using direct method and template method.
- For more information about custom objects, refer to following link:
<http://www.universalclass.com/articles/computers/javascript/working-with-javascript-objects-and-arrays.htm>



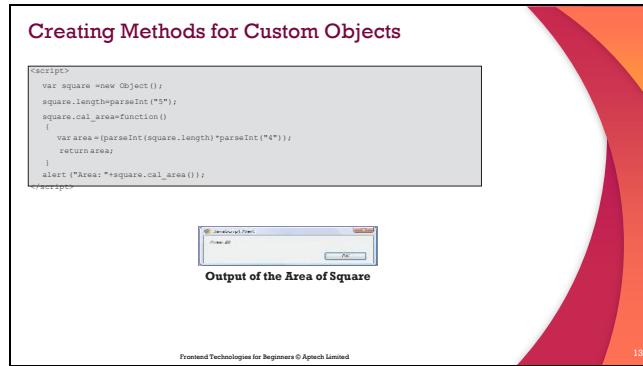
Instructions to the Trainer(s):

- Using Slide 11, explain how to create properties for custom objects.
- Mention that properties for the object are specified using the `Object` or template method.
- To create and access a property of an object using the `Object` object, specify the object name followed by a period and the property name.
- The code specifies three properties of the `student_details` object: `first_name`, `last_name`, and `age` along with their values.
- The values of the properties are displayed in the browser using the `write()` method.



Instructions to the Trainer(s):

- Using Slide 12, mention that when using the template method to create a custom object, a constructor function is used to declare properties for an object.
- Explain the code snippet that creates the **employee_info** object.
 - The code specifies three properties: **name**, **age**, and **experience** along with their values in the constructor function.
 - The object named **empMary** passes the values as the arguments and invokes the constructor function and initializes the properties to their values.
 - The **this** keyword references the current object whose properties are being initialized.
 - It resolves conflict between the property and the parameter, both of which have the same name, when assigning the value to the properties of an object.
- Show the output.



Instructions to the Trainer(s):

- Using Slide 13, mention that even a custom method can take parameters.
- Once an object is created, one or more methods can be specified.
- A method is invoked when it is specified with an object name followed by a period, method name, and parenthesis with parameters, if any.
- Explain how the code snippet defines a custom method to calculate the area of a square and show the output.
- Mention that in the template method, a function is assigned to the custom method. Such functions are known as method functions.
 - To invoke the function, specify the object name followed by a period and the method name.

In-class Question:

Question: How do you invoke a method function?

Answer: By specifying the object name followed by a period and the method name.

Slide 14

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.

Frontend Technologies for Beginners © Aprach Limited

14

Instructions to the Trainer(s):

- Using Slide 14, JavaScript objects are categorized as built-in objects, browser objects, and HTML objects.
- Mention that built-in objects that JavaScript supports, operate independently of whatever page the browser has loaded.
- Explain the string object.
- Strings are a set of characters surrounded by single or double quotes.
- These can include alphabets, numbers, spaces, and symbols, such as %, @, &, and so on.
- The String object help perform different text operations, such as searching for a specific character occurrence and retrieving a substring.
- Explain the Math object.
- The Math object helps perform mathematical operations on numeric values.
- It is a pre-defined object that provides static properties and methods to perform mathematical operations and can be invoked directly with the object name.
- No object instantiation is required.
- Explain the Date object.
- It helps define and manipulate the date and time values programmatically.
- It supports both the Universal Time Coordinated (UTC) and Greenwich Mean Time (GMT) conventions.
- The Date object calculates dates in milliseconds from 01 January, 1970.
- For more information on built-in objects, refer to following link:
https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects

In-class Question:

Question: How is a variable treated if it contains true or false values?

Answer: It is treated as a Boolean object.

Browser Objects

- Browser objects help manipulate various aspects of a Web browser.
- They exist on all pages displayed in the browser.

```
graph TD; window --> document; window --> history; window --> location; window --> navigator;
```

Frontend Technologies for Beginners © Aptech Limited

15

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 following link:
<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)

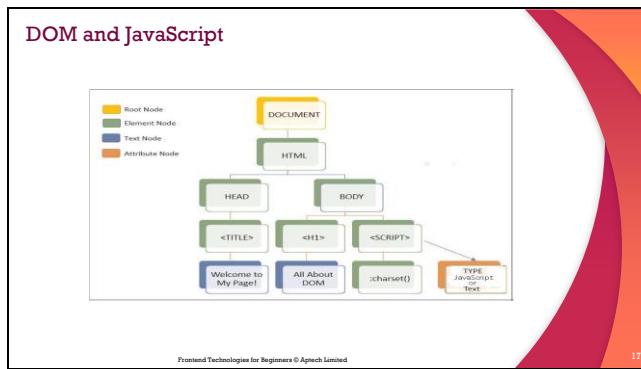
- DOM is a cross-platform and language-independent interface.
- It considers an XML or HTML document as a tree structure.
 - Each node is an object representing a part of the document.
- DOM represents a document with a logical tree.
 - Each branch of the tree ends in a node.
 - Each node contains objects.

Frontend Technologies for Beginners © 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 following link:
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");  
})
```

Frontend Technologies for Beginners © 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 following link:
<https://www.javascripttutorial.net/es6/javascript-arrow-function/>

In-class Question:

Question: What happens to syntax when arrow functions are used?

Answer: The syntax becomes short.

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.

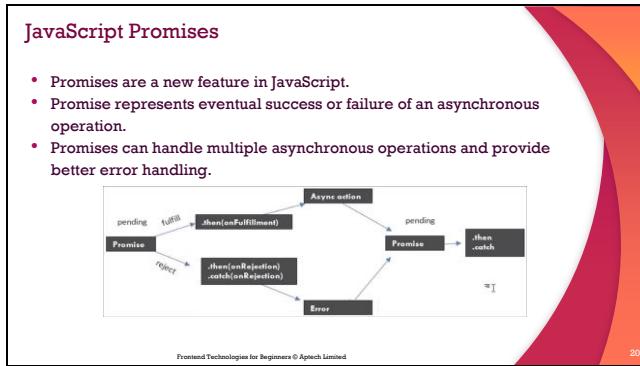
```
const webFrameworks = ["React", "Angular", "Rails", "Node.js"];
let text = "";
for (let x of webFrameworks) {
    text += x;
}
console.log(text);
```

Frontend Technologies for Beginners © Aptech Limited

19

Instructions to the Trainer(s):

- Using Slide 19, explain the `for of` loop statement.
- 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 following link:
https://exploringjs.com/es6/ch_for-of.html



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 following link:
<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.' }
```

Frontend Technologies for Beginners © 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 following link:
- https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Working_With_Private_Class_Features

In-class Question:

Question: What happens to a class when a method is public?

Answer: It should be instantiated.

Slides 22 and 23

JavaScript Object Notation (JSON) 1-2

```
[{"page":1, "results": [{"first_air_date":"2005-03-26", "genre_ids":[28, 12, 18, 19], "id":157243, "original_name":"Doctor Who", "origin_country":[{"name": "UK"}], "name":"Doctor Who"}, {"first_air_date":"2007-09-24", "genre_ids":[18, 35], "id":1418, "original_name":"The Big Bang Theory", "origin_country":[{"name": "US"}], "name":"The Big Bang Theory"}]}
```

Frontend Technologies for Beginners © Aptech Limited

22

JavaScript Object Notation (JSON) 2-2

```
[{"first_air_date":"2015-08-23", "genre_ids":[18, 27, 1, "id":62286, "original_name":"Fear the Walking Dead", "origin_country":[{"name": "US"}], "name":"Fear the Walking Dead"}, {"total_pages":3116, "total_results":62309}]
```

Frontend Technologies for Beginners © 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 following link: <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.



Instructions to the Trainer(s):

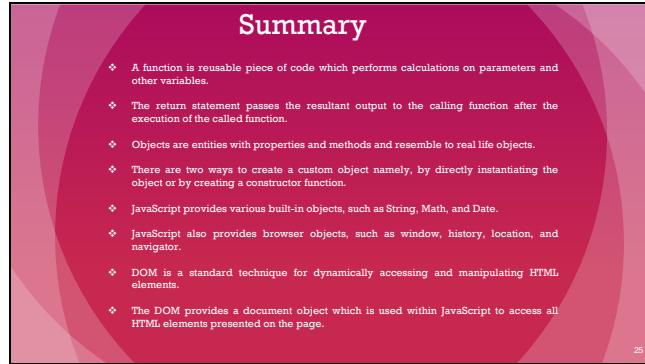
- Using Slide 24, explain serialization and deserialization.
- Tell the students that serialization and deserialization are essential to save the state of an object so that it can be created again. This way data can be stored and exchanged.
- Small data is stored in byte form. Large and complex data 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 following link:
<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

Slide 25



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):

Using Slide 25, summarize the session by reading out each point on the Slide.

Session 12: ECMAScript 2022 and Its New Features

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 which will be a key point to relate the current session objectives.

12.1.1 Teaching Skills

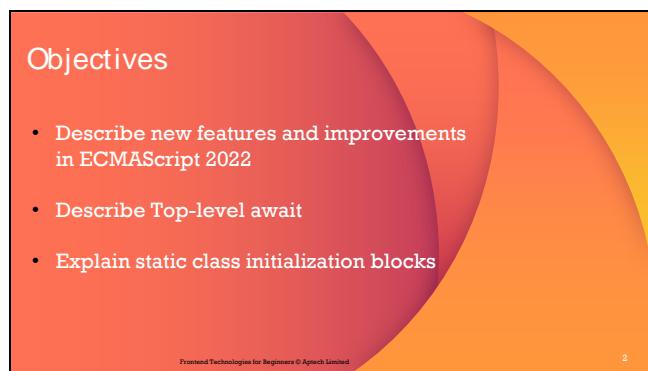
To teach this session, you should be well versed with new features of ECMAScript 2022.

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

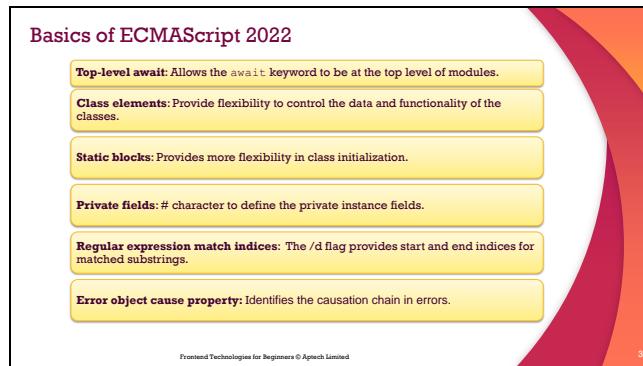


Instructions to the Trainer(s):

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

12.2 In-Class Explanations

Slide 3



Instruction(s) to the trainer:

Using Slide 3, explain that in June 2022, the 13th edition of the ECMAScript language was released, which is commonly referred to as ES2022. This new version includes enhancements compared to its previous iteration.

Introduce each of the features listed on the Slide.

Tell students that the latest version of ECMAScript (ES2022) introduces a new feature called `top-level await`, which permits the use of the `await` keyword outside of a function. Explain that it is different from earlier versions where `await` was only allowed inside functions that were declared as `async`. Tell students that this feature will be discussed in detail.

Next, tell students about new class elements. These include public and private instance fields, public and private static fields, private instance methods and accessors, as well as private static methods and accessors. Tell them that these enhancements offer developers flexibility and control on the data and functionality of classes.

Further, explain that another new feature introduced in ES2022 is the ability to add static blocks within classes. This offers increased flexibility during class initialization because it allows for per-class evaluation initialization. It allows creation of modular and maintainable code.

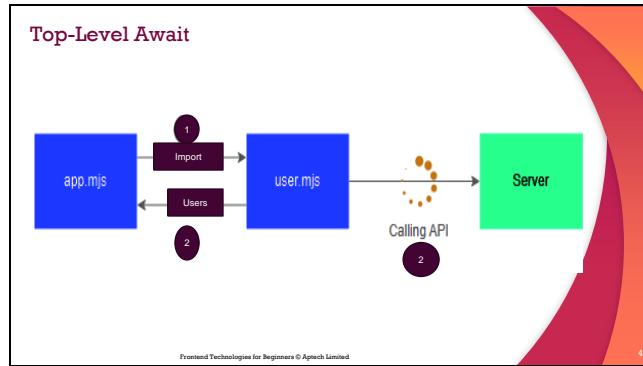
Next, explain about private fields. Private fields can only be accessed within the class itself and not from outside the class or its instances. Tell students that the `#` character allows defining the private instance fields.

Continue and explain the next feature, regular expression match indices. Tell students that the `/d` flag allows for identification of start and end indices for matched substrings.

Finally, tell them that another feature cause has been introduced in the Error object. It helps in identifying the causal chain in errors, making it simpler to trace the source of an error.

Refer to following links for more information on new features of ECMAScript 2022:

- https://www.youtube.com/watch?v=w1_qt9nyAFA
- <https://softuni.org/dev-talks/all-the-new-javascript-features-coming-up-with-ecmascript-2022/>
- <https://enlear.academy/7-new-javascript-features-in-ecmascript-2022-64a330f6eae2?gi=4ee7fd1f4088>
- <https://blog.devgenuis.io/ecmascript-2022-is-officially-released-what-should-we-pay-attention-to-5e207ed61a46>
- <https://www.infoworld.com/article/3665748/8-new-javascript-features-to-start-using-today.html>



Instruction(s) to the trainer:

Using Slide 4, explain the top-level await. Explain that the figure shows an architecture where, a JavaScript module, `app.mjs`, imports data from another module called `user.mjs`. Tell students that as shown in the figure, both `app.mjs` and `user.mjs` use the top-level await feature. This allows them to wait for data to be fetched before continuing with their respective operations. Continue explaining that `user.mjs` makes a request to an external server to fetch the user data and because of the top-level await feature, the module waits for the response to be received before continuing with any further operations. After the data is fetched, it is exported by `user.mjs` so that it can be utilized by other modules. Conclude by saying that the use of top-level await in both modules allows for an organized way of handling asynchronous operations.

Next, explain Code Snippet 1 of LG. Also, explain the output.

Additional Code Example:

Explain following Code Snippet from an external source: <https://v8.dev/features/top-level-await>:

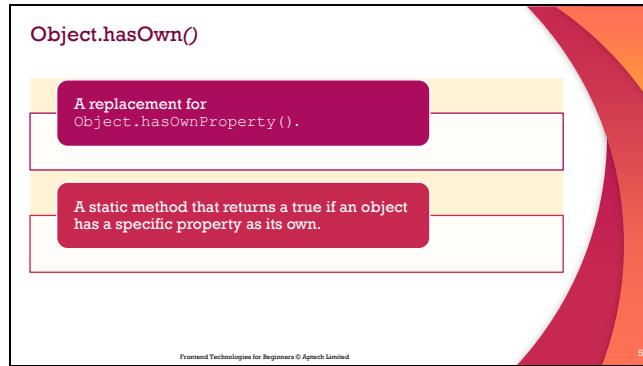
```
const strings = await import(`./i18n/${navigator.language}`);
```

Explain that in the code, the `import()` function with top-level await imports module at runtime. `await` waits for the resolution of the Promise returned by `import()`.

Refer to following link for more information on top-level await:

<https://typescript.tv/new-features/top-level-await-in-typescript-3-8/>

Slide 5



Instruction(s) to the trainer:

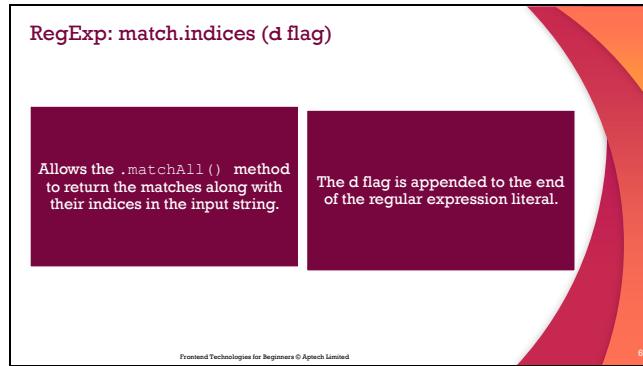
Using Slide 5, explain the next property, which is the Object.hasOwnProperty(). Tell students that Object.hasOwnProperty() is a static method that can be used instead of Object.hasOwnProperty(). It returns a Boolean value. Further, explain that it indicates whether an object has a specific property same as its own. However, if the property is inherited, it will return false.

Next, explain Code Snippet 3 of LG. Also, explain the output.

Refer to following link for more information on use of Object.hasOwnProperty() property and explain the example.

<https://www.youtube.com/watch?v=9w0gA13EXYU>

Slide 6



Instruction(s) to the trainer:

Using Slide 6, explain the next new feature, which is the d flag. Tell students it is a regular expression syntax that allows the .matchAll() method to return not only the matches found in the input string, but also their indices. This is achieved by appending the d flag to the end of the regular expression literal.

Next, explain Code Snippet 4 of LG. Also, explain the output.

Continue explaining that the d flag makes it easy for developers to work with multiple matches in a single regular expression.

Additional Code Example:

You can take following example from <https://github.com/tc39/proposal-regexp-match-indices#examples>:

```
/ indices are relative to start of the input string: const s1
= "xaaaz";
const m1 = re1.exec(s1); m1.indices[0][0] === 1;
m1.indices[0][1] === 5; s1.slice(...m1.indices[0]) === "aaaz";
m1.indices[1][0] === 4;
m1.indices[1][1] === 5; s1.slice(...m1.indices[1]) === "z";
m1.indices.groups["Z"][0] === 4;
m1.indices.groups["Z"][1] === 5;
s1.slice(...m1.indices.groups["Z"]) === "z";
// capture groups that are not matched return `undefined`:
const m2 = re1.exec("xaaay");
m2.indices[1] === undefined; m2.indices.groups["Z"] ===
undefined;
```

Explain the usage of d flag in the code. It is added so that the .exec() method can return both the match and its index in the input string. The indices property retrieves the start and

end indices of the entire match.

Conclude by saying that the `d` flag enables accurate management of matches in intricate patterns. With the `d` flag, it is possible to retrieve the matched substrings and get their exact locations in the input string.

Slide 7

Summary

- ❖ The ECMAScript 2022 is the 13th version of the ECMAScript language released in June 2022.
- ❖ Private field is a feature in ECMAScript that allows developers to define private properties in classes that are encapsulated and not directly accessible from outside the class.
- ❖ The static initialization blocks provide a way to initialize static properties in a more flexible manner.
- ❖ `Object.hasOwn()` property is a static method that is a replacement for `Object.hasOwnProperty()` and returns a Boolean value indicating if an object has a specific property as its own.
- ❖ The `d` flag in regular expressions is a new addition that allows the `.matchAll()` method to return the matches along with their indices in the input string.

7

Instructions to the Trainer(s):

Show students Slide 7.

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

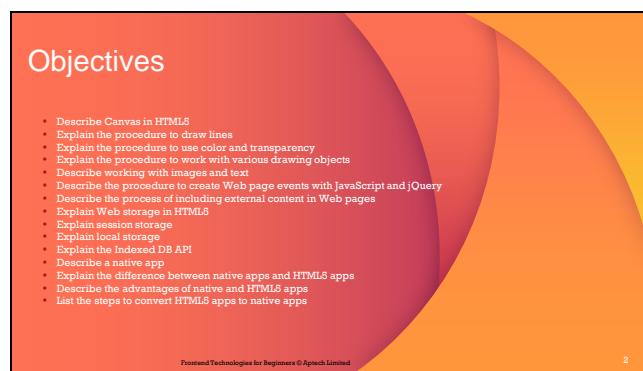
To teach this session, you should be well versed with `<canvas>` element in HTML5. You should also familiarize yourself with drawing objects, images, text, and create Web page events with JavaScript and jQuery. Also, you should understand the Web storage in HTML5, native apps, and HTML5 apps.

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

Slide 3

Canvas Element

- 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.

Frontend Technologies for Beginners © Aptech Limited

3

Instructions to the Trainer(s):

- Using Slide 3, 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.

Slide 4

Drawing a Line in Canvas

- 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

Frontend Technologies for Beginners © ApnaGhar Limited

4

Instructions to the Trainer(s):

- Using Slide 4, 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:

Q: Which methods are used to define the starting point and ending point of the line?

Answer: `moveTo(x, y)` and `lineTo(x, y)`

Slides 5 to 8

Working with Drawing Objects 1-4

- 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 fill style is solid black, but user can set color according to the requirement.
<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.

Frontend Technologies for Beginners © Apneach Limited

5

Working with Drawing Objects 2-4

➤ 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

Frontend Technologies for Beginners © Apneach Limited

6

Working with Drawing Objects 3-4

➤ 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 $2 * \pi$.
- 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

Frontend Technologies for Beginners © Apneach Limited

7

Working with Drawing Objects 4-4

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.

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.

Frontend Technologies for Beginners © Apsach Limited

8

Instructions to the Trainer(s):

- Using Slides 5 to 8, explain the working with drawing object.
- HTML5 canvas allows the user to work with different types of drawing objects.
- 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.
- 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).
- 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`.
- 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:

Q: 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.

Slide 9

The slide has a white background with a decorative red and orange curved bar on the right side. The title 'Working with Images' is at the top left. Below it is a bulleted list of three items. At the bottom left is a small text 'Frontend Technologies for Beginners © ApnaGang Limited'. At the bottom right is the number '9'.

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.

Frontend Technologies for Beginners © ApnaGang Limited

9

Instructions to the Trainer(s):

- Using Slide 9, 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.

Slide 10

Working 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.
- To set the text color, the `fillStyle` property of the canvas can be used.

Frontend Technologies for Beginners © ApnaGhar Limited

10

Instructions to the Trainer(s):

- Using Slide 10, 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.
- 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).

Slides 11 to 13

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.

Frontend Technologies for Beginners © Agnieszka Limited

11

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.

Frontend Technologies for Beginners © Agnieszka Limited

12

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:

Frontend Technologies for Beginners © Agnieszka Limited

13

Instructions to the Trainer(s):

- Using Slides 11 to 13, 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.
- 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.
- 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.

Slide 14

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.

Frontend Technologies for Beginners © ApnaGuru Limited

14

Instructions to the Trainer(s):

- Using Slide 14, 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

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.

- 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 requirements.
 - 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.

Source: Technologies for Developers (Project) - United

Instructions to the Trainer(s):

- Using Slide 15, 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.

Slide 16

Web Storage vs. 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.

Frontend Technologies for Beginners © ApnaGuru Limited

16

Instructions to the Trainer(s):

- Using Slide 16, 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+

Frontend Technologies for Beginners © ApnaGuru Limited

17

Instructions to the Trainer(s):

- Using Slide 17, 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.

Slide 18

The slide has a decorative red and orange curved graphic on the right side. At the bottom right is the number '18'. The text on the slide is as follows:

- 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.

Frontend Technologies for Beginners © ApnaGuru Limited

Instructions to the Trainer(s):

- Using Slide 18, 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:

Q: Which property is used for checking the support Web storage?

Answer: The `localStorage` or `sessionStorage` properties are used for checking the support Web storage.

Slides 19 and 20

Session Storage 1-2

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

Frontend Technologies for Beginners © Apache Limited

19

Session Storage 2-2

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();`

Frontend Technologies for Beginners © Apache Limited

20

Instructions to the Trainer(s):

- Using Slides 19 and 20, 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

- 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 has 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()`.

© Microsoft Corporation. All rights reserved. © Agustín, Instituto de la Música

Instructions to the Trainer(s):

- Using Slide 21, 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()`.

Slides 22 to 24

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.

Frontend Technologies for Beginners © Aptech Limited

22

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.

Frontend Technologies for Beginners © Aptech Limited

23

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	6.0	-	4.2-4.3
9.0	10.0moz	17.0webkit	8.1	11.6	5.0
10.0	11.0moz	18.0webkit	6.0	12.0	-
-	12.0moz	19.0webkit	-	-	-

Frontend Technologies for Beginners © Aptech Limited

24

Instructions to the Trainer(s):

- Using Slides 22 to 24, 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 25 and 26

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.

Frontend Technologies for Beginners © Apache Limited

25

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.

Frontend Technologies for Beginners © Apache Limited

26

Instructions to the Trainer(s):

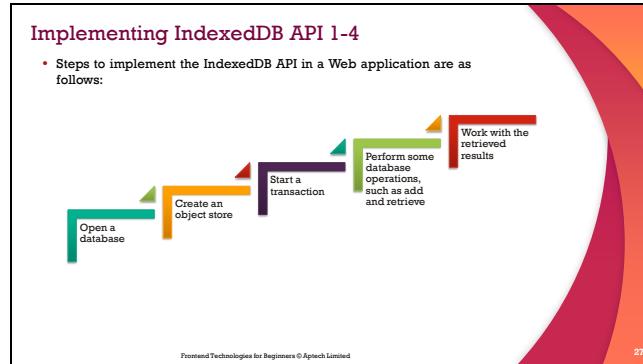
- Using Slides 25 and 26, explain the basic constructs of IndexedDB API.
- Tell the students that the basic constructs are Database, Object Store, Keys, Values, Key Path, Index, and Transaction.
- Also, some of the basic constructs of IndexedDB API are:
 - Requests
 - Cursor
 - Key Range

In-Class Question:

Q: Which construct is used to retrieve multiple records?

Answer: Cursor is used to retrieve multiple records.

Slides 27 to 30



Implementing IndexedDB API 2-4

- 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) {
    ...
}
```

Frontend Technologies for Beginners © Aptech Limited

28

Implementing IndexedDB API 3-4

- 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.

Frontend Technologies for Beginners © Aptech Limited

29

Implementing IndexedDB API 4-4

- **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.
- **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.

Frontend Technologies for Beginners © Aptech Limited

30

Instructions to the Trainer(s):

- Using Slides 27 to 30, 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

Slide 31

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).

Frontend Technologies for Beginners © ApnaGhar Limited

31

Instructions to the Trainer(s):

- Using Slide 31, 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.

Slide 32

The slide has a title 'Converting HTML5 Apps to Native Apps' at the top. Below the title are two green-bordered boxes containing text. The first box contains: 'A native application also known as native app is an application program that is built for using it on a particular device or platform.' The second box contains: '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.' At the bottom left is the text 'Frontend Technologies for Beginners © ApnaGhar Limited'. At the bottom right is the number '32'.

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.

Frontend Technologies for Beginners © ApnaGhar Limited

32

Instructions to the Trainer(s):

- Using Slide 32, 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, iTunes is a native app available on iPhone mobile devices.

Slide 33

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 does not require to have a network connection.

- Following table lists differences between native apps and HTML5 apps:

Native Apps	HTML5 Apps
Native Apps run 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 must sacrifice the effects of "less" and "objective C" for writing native applications.

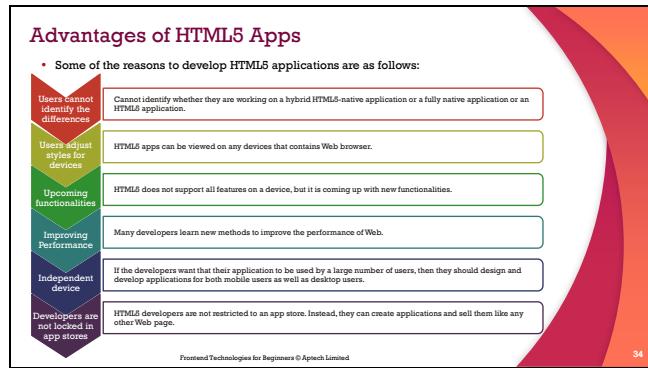
Created by Instructor: Dileepa De Silva

33

Instructions to the Trainer(s):

- Using Slide 33, 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.

Slide 34



Instructions to the Trainer(s):

- Using Slide 34, 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.

Slide 35

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 accessing cameras, cameras, or any other device hardware for HTML5 apps.	Native apps can access the file system on 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's not a full 3D experience.	HTML5 provides access to offline Web application. However, a native app is stored on local machine, so that user do not require access to the Web to work with the application.

Frontend Technologies for Beginners © ApnaGhar Limited

35

Instructions to the Trainer(s):

- Using Slide 35, 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.

The infographic is titled "Converting HTML5 Apps to Native Apps". It features two circular icons: a red one for "PhoneGap" and a dark purple one for "Appcelerator". Both icons have descriptive text boxes below them. The PhoneGap box states: "Is an HTML5 app that allows the user to create native apps with Web technologies and is accessible to app stores and APIs." The Appcelerator box states: "Is a cross-platform mobile application development support and allows the users to create Android, iOS, and mobile Web apps." At the bottom left is the text "Frontend Technologies for Beginners © ApnaGang Limited" and at the bottom right is the number "36".

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.

Frontend Technologies for Beginners © ApnaGang Limited

36

Instructions to the Trainer(s):

- Using Slide 36, explain the concept of converting HTML5 apps to native apps.
- Users can use tools to convert HTML5 app to a native app. 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.

Slide 37



Instructions to the Trainer(s):

- Show students Slide 37.
- 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

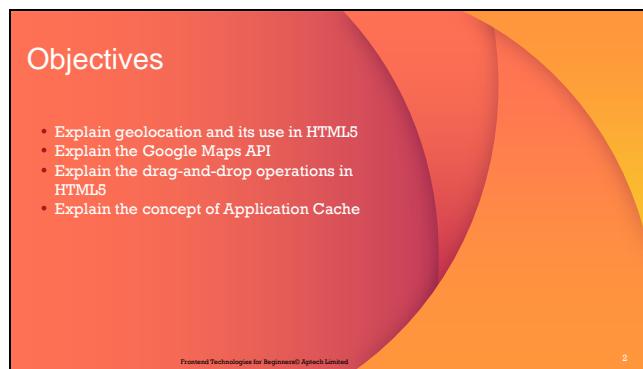
To teach this session, you should be well-versed with the new APIs supported by HTML5 such as Geolocation API and Google Maps API that are used to determine and display the location on a map. Along with this, you should prepare yourself to explain the drag-and-drop mechanism which is used to perform 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

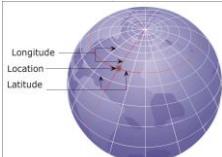
Slides 3 and 4

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:



Frontend Technologies for Beginners © Apricorn 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.

Frontend Technologies for Beginners © Apricorn 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

Slides 5 and 6

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.

Frontend Technologies for Beginners © Aprach Limited

5

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+

Frontend Technologies for Beginners © Aprach Limited

6

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.

Slide 7

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.

Frontend Technologies for Beginners © ApnaGhar Limited

7

Instructions to the Trainer(s):

- Using Slide 7, 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.

Slide 8

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
<code>getCurrentPosition()</code>	Retrieves the current geographic location information of the user
<code>watchPosition()</code>	Retrieves the geographic information of the device at regular intervals
<code>clearWatch()</code>	Terminates the current watch process

Frontend Technologies for Beginners © ApnaGuru Limited

8

Instructions to the Trainer(s):

- Using Slide 8, 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.

Slides 9 to 11

Retrieve User Information 1-3

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.

Frontend Technologies for Beginners © Apache Limited

9

Retrieve User Information 2-3

- 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.

Frontend Technologies for Beginners © Apache Limited

10

Retrieve User Information 3-3

- Following figure shows a message displaying current location of the user, when the `Share My Location` button is clicked:

Frontend Technologies for Beginners © Apache Limited

11

Instructions to the Trainer(s):

- Using Slides 9 to 11, 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 in the Learner Guide 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.
- 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`.

Slides 12 and 13

Handling Errors 1-2

- 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.

Frontend Technologies for Beginners © Aptech Limited

12

Handling Errors 2-2

- 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.

Frontend Technologies for Beginners © Aptech Limited

13

Instructions to the Trainer(s):

- Using Slides 12 and 13, 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 in the Learner Guide 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:///`.

Slide 14

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 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.

Frontend Technologies for Beginners © ApnaGhar Limited

14

Instructions to the Trainer(s):

- Using Slide 14, 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.

Slides 15 to 17

Google Maps API 1-3

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.

Frontend Technologies for Beginners © Aprach Limited

15

Google Maps API 2-3

- Following table lists some of the myOptions properties:

Property	Description
zoom	Set 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 LatLang 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.

Frontend Technologies for Beginners © Aprach Limited

16

Google Maps API 3-3

- Following figure displays the object on the Web page that is centered on Lord's Cricket Ground in London:



Frontend Technologies for Beginners © Aprach Limited

17

Instructions to the Trainer(s):

- Using Slides 15 to 17, 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 17 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`

Slides 18 to 20

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: 10px;
  }
  #map_canvas {
    height: 50px;
    width: 50px;
  }
</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, errorCallback);
  } else {
    alert('Geolocation is not enabled in your browser');
  }
</script>
Frontend Technologies for Beginners © Apeach Limited
```

18

Tracking User's Location 2-3

```
// Creates the Map object
var map = new
google.maps.Map(document.getElementById("map_canvas", myOptions);
// Displays marker on the located position
var marker = new google.maps.Marker({
  position: latlng,
  map: map,
  title:"User location"
});
// Error catch function
function errorCallback(pos) {
  alert('Error!!');
}
</script> </head>
<body>
  <div id="map_canvas"></div>
  <div id="user_location"></div>
</body>
Frontend Technologies for Beginners © Apeach Limited
```

19

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

20

Instructions to the Trainer(s):

- Using Slides 18 to 20, 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.

Slide 21

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 necessity of complex JavaScript code written in earlier HTML versions.

Frontend Technologies for Beginners © ApnaGhar Limited

21

Instructions to the Trainer(s):

- Using Slide 21, 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.

Slide 22

The slide has a decorative background with a red-to-yellow gradient curve on the right side. The title 'Drag Operation' is at the top left. Below it is a bulleted list: 'The steps required to make any element draggable on a Web page are as follows:' followed by three numbered steps: 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. At the bottom left is the text 'Frontend Technologies for Beginners © ApnaGuru Limited'. At the bottom right is the number '22'.

- 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.

Frontend Technologies for Beginners © ApnaGuru Limited
22

Instructions to the Trainer(s):

- Using Slide 22, explain the drag operation.
- 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 `` element contains `draggable` attribute that is set to `true`. The value `true` indicates that the element is eligible for dragging.

Slide 23

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.

Frontend Technologies for Beginners © ApnaGhar Limited

23

Instructions to the Trainer(s):

- Using Slide 23, 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.

Slide 24

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

Frontend Technologies for Beginners © ApnaGhar Limited

24

Instructions to the Trainer(s):

- Using Slide 24, 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 in the Learner Guide 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.

Slide 25

The slide has a decorative background with a red-to-yellow gradient curved shape on the right side. The title 'Drop Operation' is at the top. Below it are five bullet points in colored boxes:

- 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.

Frontend Technologies for Beginners © Agnieszka Limited

25

Instructions to the Trainer(s):

- Using Slide 25, 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.

Slides 26 and 27

Drop Events 1-2

- 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:

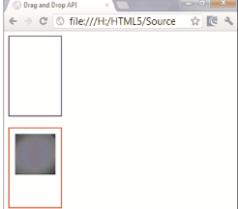
Event	Description
dragenter	Triggers when a draggable element is being dragged on the target element for the first time.
dragleave	Triggers when an element is dragged outside the target element.
dragover	Triggers when an element is dragged inside the target element.
drop	Triggers when an element is dropped in the target element.

Frontend Technologies for Beginners © Agnieszka Limited

26

Drop Events 2-2

- Following figure shows the output of the drop operation, after the image is dragged on the target element:



Frontend Technologies for Beginners © Agnieszka Limited

27

Instructions to the Trainer(s):

- Using Slides 26 and 27, 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.

Slide 28

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.
- 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 must be saved.
 2. Reference the manifest file in each Web page designed to use cached resources.

Frontend Technologies for Beginners © ApnaGhar Limited

28

Instructions to the Trainer(s):

- Using Slide 28, 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.

Slides 29 and 30

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
```

Frontend Technologies for Beginners © Agnieszka Limited

29

Creating a Manifest File 2-2

- Following are the sections defined in the .manifest file:

CACHE	<ul style="list-style-type: none">• This section defines resources, such as check.js, styles.css, and figure1.png to be stored locally.
FALLBACK	<ul style="list-style-type: none">• This section defines alternative resource to be used, when the actual resource is not available.
NETWORK	<ul style="list-style-type: none">• This section specifies resources to be accessed when there is a network connection. Resources in this section are not cached.

Frontend Technologies for Beginners © Agnieszka Limited

30

Instructions to the Trainer(s):

- Using Slides 29 and 30, 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.

Slides 31 to 33

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>
```

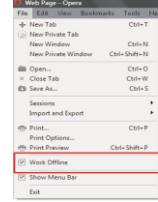
- In the code, the “`appcache.manifest`” is specified with the `<html>` tag.
- The interpretation of the manifest file is similar to any other file reference.
- The document uses a relative file path, as both the manifest file and HTML document are located in the same directory.
- By default, a Web page declaring manifest is cached automatically.

Frontend Technologies for Beginners © Apachek Limited

31

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:

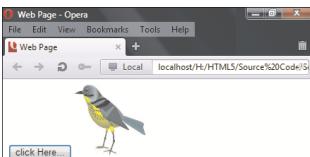


Frontend Technologies for Beginners © Apachek Limited

32

Declaring a Manifest 3-3

- Following figure shows the cached Web page in the Opera browser:



Frontend Technologies for Beginners © Apachek Limited

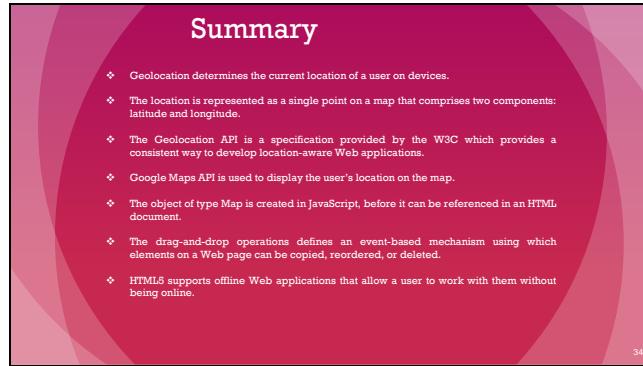
33

Instructions to the Trainer(s):

- Using Slides 31 to 33, 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.

- Figure on Slide 32 shows how to enable the Work Offline mode in the Opera browser. This enables to cache the resources of the Web application pages locally.

Slide 34



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.

34

Instructions to the Trainer(s):

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

Session 15: Building Mobile Websites

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 which will be a key point to relate the current session objectives.

15.1.2 Teaching Skills

To teach this session successfully, you should be familiar with HTML5 frameworks for mobile Websites. You should also have an in depth knowledge of the importance of creating a responsive Web page.

For teaching in the class, you are expected to use slides and LCD projectors.

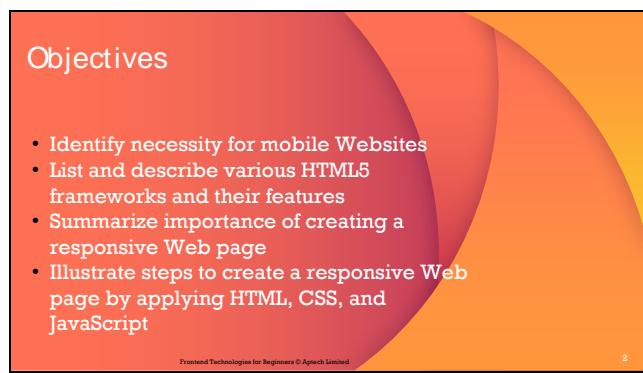
Tips:

It is recommended that you test the understanding of the students by asking questions in between the class.

In-Class Activities:

Follow the order given here during In-Class activities.

Slide 2

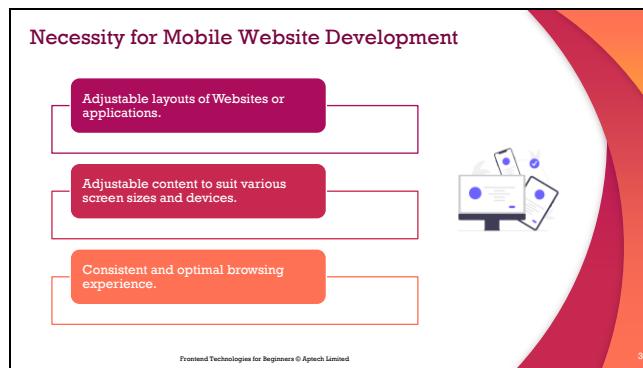


Overview of the Session:

Give the students a brief overview of the current session in the form of session objectives. Show the students slide 2 of the presentation.

15.2 In-Class Explanations

Slide 3



Instruction(s) to the trainer:

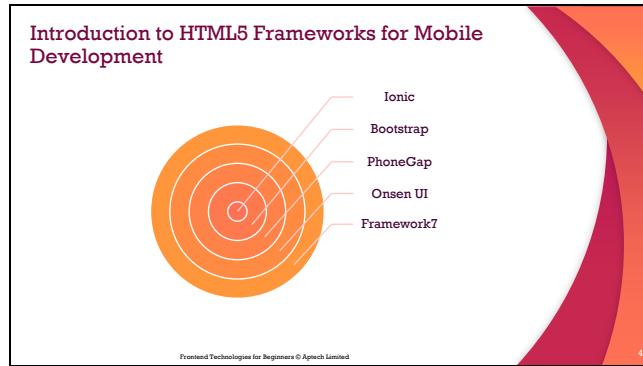
Using Slide 3, give an overview to students on the necessity for mobile Website development.

Tell them that the usage of mobile devices for accessing Websites has increased significantly and it is not restricted to bigger screens, such as desktops and laptops anymore. Therefore, it is crucial for Web developers to create a Web page that can adapt to different screen sizes and devices, which is known as responsive design. This approach allows Websites or applications to modify their layout and content, providing users with a seamless and ideal browsing experience.

Continue to tell that in today's digital age, having a mobile-friendly Website is no longer an option, but a necessity. With extensive number of unique mobile phone users worldwide, it is clear that mobile devices are the primary means of accessing the Internet. In fact, mobile devices account for majority of all Web traffic globally. Furthermore, research shows that majority of users will not recommend a business with a poorly designed mobile site, and they leave the site altogether if it takes more than three seconds to load. With these statistics in mind, it is clear that a mobile-friendly Website is essential for businesses to remain competitive and attract and retain customers.

Refer to following link for more information on the necessity for mobile Website development:

- <https://cynoteck.com/blog-post/reasons-why-your-website-needs-a-mobile-application/>
- <https://magora-systems.com/mobile-website-development/>
- <https://developer.mozilla.org/en-US/docs/Web/Guide/Mobile>



Instruction(s) to the trainer:

Using Slide 4, give students an introduction to the HTML5 Frameworks for Mobile Development.

Tell them that Web developers who are familiar only with HTML, CSS, and JavaScript may find it challenging to develop native mobile applications. This is because the programming languages used to create mobile apps for platforms such as Android and iOS are completely different from one another. Additionally, the process of learning these programming languages can be difficult due to their steep learning curve.

HTML5 frameworks are becoming increasingly popular for mobile development due to their flexibility and compatibility across multiple devices and platforms. According to a research, using an HTML5 framework can reduce the development time for mobile apps by 20-30%, allowing businesses to get their product to market faster. With the increasing demand for mobile apps and the requirement for cross-platform compatibility, HTML5 frameworks are a smart choice for businesses looking to develop mobile apps efficiently and cost-effectively.

Explain in detail to students various HTML5 frameworks that are easily accessible and can assist in creating a mobile version of Websites. These frameworks come with pre-designed components and templates that simplify the development process and are outlined as follows:

- Ionic
- Bootstrap
- PhoneGap
- Onsen UI
- Framework7

Each framework has its distinctive features, advantages, and disadvantages. However, depending on the development objectives and necessities, the most suitable framework can

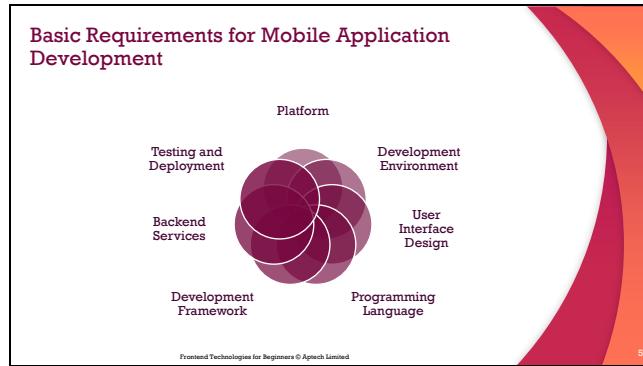
be selected. Additionally, some frameworks may demand additional skill sets to work with them.

Refer to following links for more information on introduction to the HTML5 frameworks for mobile development:

<https://technostacks.com/blog/mobile-app-development-frameworks/>

<https://colorlib.com/wp/html5-frameworks/>

Slide 5



Instruction(s) to the trainer:

Using Slide 5, explain to students about the basic requirements for mobile application development.

Tell them that developers can create not only mobile Websites but also mobile applications.

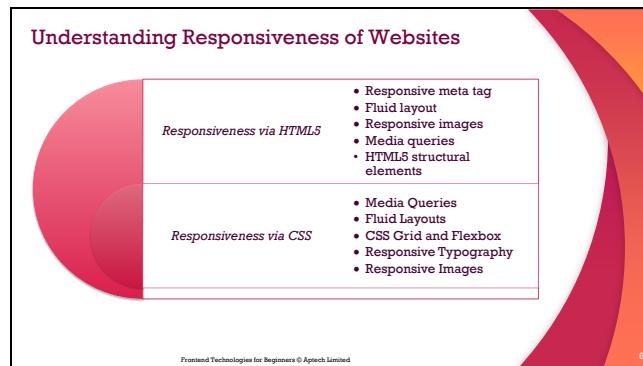
Creating a successful and high-quality mobile application demands careful consideration of various requirements and best practices. This includes determining the target platform and audience, as well as prioritizing the User Interface (UI) during the development process. A well-designed UI is crucial in ensuring optimal performance and security, as it influences how smoothly users can navigate through the app. Additionally, thorough testing is necessary to eliminate bugs and errors, and ongoing maintenance is essential to fix bugs, add features, and ensure compatibility with the latest operating systems and devices. By adhering to these guidelines, developers can create HTML5-based mobile applications that meet high quality standards and provide an excellent user experience.

Now, explain in detail some of the basic requirements to be considered while developing a mobile application with HTML5 that are as follows:

- Platform
- Development Environment
- User Interface Design
- Programming Language
- Development Framework
- Testing and Deployment

Refer to following links for more information on the basic requirements for mobile application development:

- <https://www.ibm.com/docs/en/spm/7.0.9?topic=requirements-defining-mobile-application>
- <https://www.upwork.com/resources/how-to-develop-an-app>
- <https://www.geeksforgeeks.org/basic-prerequisites-of-learning-android-app-development/>



Instruction(s) to the trainer:

Using Slide 6, explain to students about understanding responsiveness of Websites.

Tell them that the increasing use of various digital devices with varying screen sizes and resolutions necessitates the creation of responsive Web pages that can adjust to different devices. This ensures a uniform user experience across a wide range of devices, including desktop computers, laptops, tablets, and smartphones.

Responsive Web design aims to develop a Website that is adaptable and convenient for users on various devices, while maintaining the quality of user experience and content legibility. To achieve this, flexible grids, resizable images, and CSS media queries are used to modify the layout and content of a Website to fit the screen size and orientation of the device being used.

A Website that is responsive can enhance the user experience, encourage higher engagement, and improve conversion rates, which can ultimately contribute to the success of businesses and organizations.

Further, explain them about responsiveness via HTML5. HTML5 is a type of coding language that has advanced capabilities and enhancements compared to its prior versions. It includes an integrated function for creating responsive designs. The process of producing a responsive design in HTML5 involves using different methods, including incorporating a responsive meta tag, fluid layout, responsive images, media queries, and HTML5 structural elements. They are as follows:

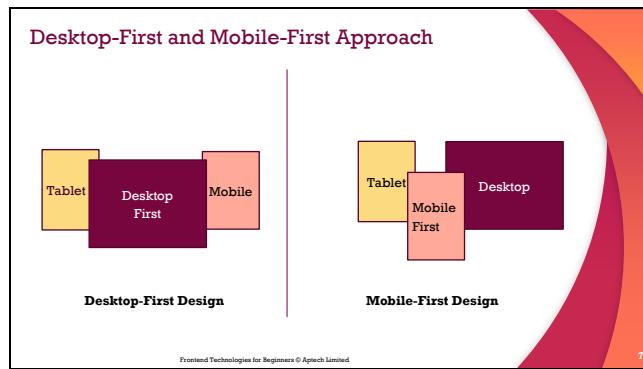
- Responsive meta tag
- Fluid layout
- Responsive images
- Media queries
- HTML5 structural elements

Also, explain them about responsiveness via CSS. HTML5, in conjunction with CSS and specifically CSS3, can be used to create a responsive Web page. There are several CSS techniques commonly used to achieve responsive Web design, which are as follows:

- Media Queries
- Fluid Layouts
- CSS Grid and Flexbox
- Responsive Typography
- Responsive Images

Refer to following links for more information on understanding responsiveness of Websites:

- https://www.w3schools.com/html/html_responsive.asp
- <https://www.webfx.com/blog/web-design/what-is-responsive-web-design/>
- <https://web.dev/responsive-web-design-basics/>
- https://developer.mozilla.org/en-US/docs/Learn/CSS/CSS_layout/Responsive_Design



Instruction(s) to the trainer:

Using Slide 7, explain to students about desktop-first and mobile-first approach.

Tell them that the development of Websites has been revolutionized by the digital age, and there are two distinct approaches to designing and developing responsive Websites for different devices and screen sizes that are desktop-first and mobile-first approach.

Continue to explain the key difference between mobile-first approach and desktop-first approach. Tell them that in the case of designing a Website using a desktop-first approach, user may have a lot of content and features to showcase, but if they do not adapt well to smaller screens or slower internet speeds, they end up with a Website that is hard to navigate and frustrating for users. On the other hand, designing a Website using a mobile-first approach is similar to building a puzzle with limited pieces. The user must carefully choose which elements are essential and optimize them for speed and usability, creating a seamless experience for users on any device. By prioritizing the most important elements and designing them to be adaptable, a mobile-first approach can result in a Website that is both technically sound and user-friendly. Therefore, just as solving a puzzle, it is important to start with the basics and build up from there to create a cohesive, effective Website.

Refer to following links for more information on desktop-first and mobile-first approach:

- <https://www.websitebuilderexpert.com/building-websites/mobile-first-design/>
- <https://www.lambdatest.com/blog/mobile-first-design/>
- <https://redblink.com/mobile-first-design-benefits-challenges/>
- <https://enou.co/blog/mobile-first-design/>



Instruction(s) to the trainer:

Using Slide 8, explain to students about building Mobile Site with HTML5, CSS, and JavaScript.

Begin by explaining in detail to students the essential steps to consider when building a mobile Website which are as follows:

- Plan the Website
- Set up the development environment
- Write the HTML code
- Style with CSS
- Add interactivity with JavaScript
- Testing and debugging
- Deployment

Further, explain in detail and with the help that there are some different approaches to developing a mobile Website which are as follows:

HTML Meta Viewport Tag: This tag is a useful tool for optimizing Website layout on mobile devices and allows the width and scale of the viewport to be adjusted.

Refer to Code Snippet 1 of Learner Guide that provides an example of the code without using the meta viewport tag.

In the output of this code, the menus appear unclear on the browser emulator, making them difficult to read. To fix this issue, the viewport meta tag can be added. This will allow the Website to adjust its layout and scale based on the size of the user screen. Now, refer to the modified code displayed in Code Snippet 2 of Learner Guide that display the optimized Website.

Existing CSS Functionality: CSS provides various properties such as max-width, min-width, and media queries that can be used to position images appropriately on mobile devices,

based on their screen size. To illustrate this, explain Code Snippet 2 of Learner Guide and the modified code shown in Code Snippet 3 of Learner Guide.

Implementation of CSS3 @Media: The @media rule is a CSS rule that allows developers to create responsive designs that adapt to different devices and screen sizes. This helps ensure that Websites look visually appealing and user-friendly across all devices.

Further, explain Code Snippet 5 of Learner Guide as an example of @media rule, where a simple CSS specification is used to set the background color.

Now, to apply this rule only for screens with a width less than or equal to 600 pixels, the @media rule is added. For this, explain Code Snippet 6 of Learner Guide.

Further, explain Code Snippet 7 of Learner Guide that demonstrates how the @media rule can be used to apply different styles to the body element based on the screen size of the device.

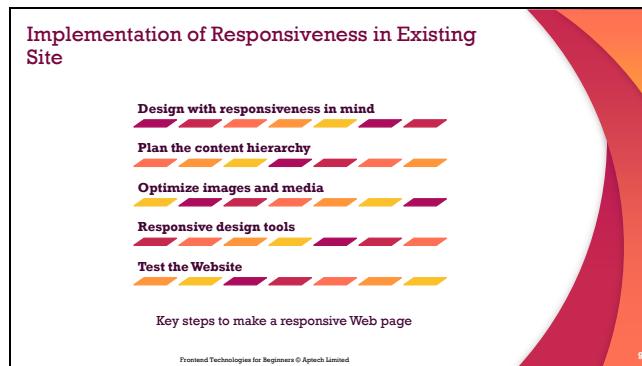
One Column To Summarize All Text Columns: Refer to Code Snippet 8 of Learner Guide to demonstrate how content should be stacked vertically in columns.

To implement this, explain Code Snippet 9 of Learner Guide.

Further, refer to the Code Snippet 10 of Learner Guide that is the complete code for the mobile Website, which incorporates these principles.

Refer to following links for more information on building mobile site with HTML5, CSS, and JavaScript:

- <https://blog.openreplay.com/building-a-mobile-app-using-html-css-and-javascript>
- <https://www.geeksforgeeks.org/build-an-android-app-with-html-css-and-javascript-in-android-studio/>



Instruction(s) to the trainer:

Using Slide 9, explain to students about the implementation of responsiveness in existing Website.

Tell them that the implementation of responsiveness in an existing Website is crucial in today's digital age where users access Websites from a variety of devices, such as smartphones, tablets, laptops, and desktops. Without responsiveness, a Website may not load properly or display correctly on different devices, which can result in a poor user experience and lost business opportunities.

Consider an example, a customer trying to access a clothing store Website on a smartphone. If the Website is not responsive, the text and images may be too small to read or the layout may be distorted, making it difficult for the customer to navigate. Frustrated, the customer may leave the Website and go to a competitor site that is easier to use on the mobile device. On the other hand, if the clothing store Website is responsive, it will adjust to fit the screen size of the customer device, providing an optimized user experience. This can lead to increased engagement, longer site visits, and ultimately, more sales.

Further, explain some important steps to make a Web page responsive are as follows:

- Design with responsiveness in mind
- Plan the content hierarchy
- Optimize images and media
- Responsive design tools
- Test the Website

Further, tell them that to make an existing Website responsive, a viewport meta tag can be included to specify the size of the viewport based on the screen sizes and orientations. The HTML <meta> tag can be used to achieve this.

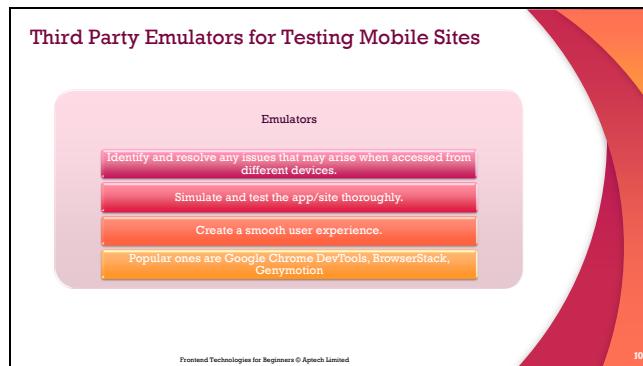
Refer to Code Snippet 11 of LG to provide an example of how to add a viewport meta tag to the HTML code.

To improve the responsiveness of a Website, the options that can be implemented are as follows:

- Using percentage (%) rather than pixels (px) to handle images.
- Using @media queries to apply different styles to div tags and images for different screen sizes.
- Using source types for images to enhance responsiveness.
- Applying a fluid grid to create Websites that can adjust to various screen sizes and devices.
- Defining typography within the site to ensure that the text is readable on screens of all sizes. For example, adjusting the font size based on the width of the parent container.

Refer to following links for more information on the implementation of responsiveness in existing site:

- <https://www.browserstack.com/guide/how-to-create-responsive-website>
- <https://kinsta.com/blog/responsive-web-design/>



Instruction(s) to the trainer:

Using Slide 10, explain to students about the third party emulators for testing mobile sites.

Tell them that before deploying a mobile site, it is crucial to test it thoroughly to ensure seamless and efficient user experiences. If a user installs an app and encounters issues such as bugs, slowness, or difficulty using it, they may uninstall the app, resulting in a negative experience for the user and a poor reputation for the developer. Therefore, it is important to thoroughly simulate and test the app/site using third-party emulators.

Next, explain that emulators are programs that help to identify and resolve any issues that may arise when accessed from different devices. Popular emulators include Google Chrome DevTools, BrowserStack, Genymotion, and others.

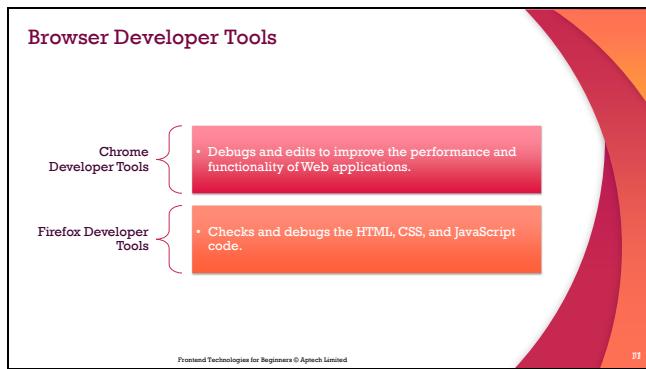
Explain that emulators can mimic slow Internet speeds, low battery life, and even network interruptions to test Website performance under these conditions. This provides developers with a more comprehensive understanding of how their Website will function in the real world helping them to identify and address potential issues before the site is launched.

A real-life example of a third-party emulator is Firebase Test Lab, which is Google cloud-based testing infrastructure for mobile apps. Firebase Test Lab allows developers to test their mobile applications on a range of virtual and physical devices and provides detailed reports on app performance and issues. It can also simulate user interactions, such as tapping and swiping, to test app responsiveness and usability.

Third-party emulators are important for testing mobile sites and applications because they provide a cost-effective and efficient way to test across multiple devices and platforms. They can also identify issues that may not be immediately obvious on physical devices or in a lab environment.

Refer to following links for more information on the third party emulators for testing mobile sites:

- <https://www.browserstack.com/emulators-simulators>
- <https://www.emizentech.com/blog/emulator-vs-simulator-for-mobile-testing.html>



Instruction(s) to the trainer:

Using Slide 11, explain to students about the browser developer tools.

Tell them that modern Web browsers come equipped with comprehensive developer tools that provide a variety of functions, such as the ability to evaluate the HTML, CSS, and JavaScript code on a Web page. If there are any errors in the code, the browser will display them and indicate their location in the code. The console outputs can also be viewed in the browser, providing insight into potential areas for optimization. Two examples of such browsers are as follows:

- **Chrome Developer Tools:** Google Chrome Developer Tools are a set of Web developer tools integrated into the Google Chrome Web browser. These tools offer a variety of features to Web developers, including debugging and editing, to enhance the performance and functionality of Web applications. If a user has an existing mobile site, they can follow a few steps to view the output of a mobile Website in a browser emulator which are as follows:
 - Open the Website in the Chrome browser.
 - Open the developer tools in Chrome by pressing F12, which will bring up a screen.
 - Click on "Toggle Device Toolbar" located in the top right corner of the screen.
 - This will open a new window.
 - From the drop-down menu at the top of the window, select a specific type of device to see how the site will look on that device. This allows the user to experience how the developed site will appear on different devices.
- **Firefox Developer Tools:** Firefox Developer Tools are a useful resource for checking and debugging HTML, CSS, and JavaScript code. Steps to view the output of a mobile Website are as follows:
 - Open the Website on the Firefox browser.

- Press F12 to open the developer tools screen.
- From the bottom right corner of the screen, select the "Responsive Design Mode" option.
- From the top of the screen, select a specific type of mobile device to view the Website.

Refer to following links for more information on the browser developer tools:

- https://developer.mozilla.org/en-US/docs/Learn/Common_questions/Tools_and_setup/What_are_browser_developer_tools
- <https://www.globalmediainsight.com/blog/web-development-tools/>

Slide 12

Summary

- ❖ The trend of viewing Websites on mobile screens has been on the rise making it essential to create responsive Webpages.
- ❖ Framework7, Ionic, and Bootstrap are some of the HTML5 frameworks to develop mobile apps.
- ❖ The main goal of responsive Web design is to create a flexible and user-friendly Website that can be accessed from any device of any size and orientation.
- ❖ CSS applies style to the HTML elements.
- ❖ The mobile-first design is preferred approach to designing Web pages.
- ❖ Emulators are programs that allows to identify and resolve any issues that arises when accessed from different devices.
- ❖ Firefox Developer tools and Chrome Developer Tools provide a range of features for Web developers such as debugging and editing.

12

Instruction(s) to the trainer:

Show students Slide 12. Summarize the session by reading out each point on the slide.