Lesson: HTML

Learn HTML: Lessons

- ✔ HTML stands for Hyper Text Markup Language.
- ✔ HTML is mostly widely used language to write web pages.
- ✔ Hypertext are called hyperlinks, which helps web pages to link together.
- ✔ The link available on the web pages are called Hypertext.
- ✔ HTML is a markup language which means markup a text document with tags that instructs the web browser to how to structure and display a web page.
- ✔ Today HTML is widely used to design a web page with the help of different tags available in HTML language.
- ✔ HTML tags are enclosed in angle brackets.
- ✔ Most of the tags have their own enclosing tags.
- ✔ Some Basic tags are:

Tags	Description
<!DOCTYPE HTML>	This tag defines the document type and HTML version.
<html></html>	This tag encloses the complete HTML document.
<head></head>	This tag represent document header, which is the parent tag of other tags like title, meta e.t.c.
<title></title>	The title tag represents the Document Title.
<body></body>	The body tags represents the document body which have other HTML tags like h1, p and div.
to	This represents the HTML heading tags. It starts form h1 and go further to h6.
	This tag represent the paragraph of a HTML document.

- ✔ Word Wide Web Consortium (W3C) represents HTML and recommends lowercase tags to be used in an HTML document.
- ✔ The visible part of the document is under the body opening and closing tags.

The **<!DOCTYPE>** Tag:

- ➢ This tag is used to declare the current version of the HTML used in the HTML document.
- ➢ There are many other doc-type used in HTML document.
- ➢ It must appear only once at the top of the page, before any HTML tag.
- ➢ The doctype declaration is not case sensitive.

HTML Headings:

- ➢ HTML headings are defined with the <h1> to <h6> tags.

Lesson: HTML

- <h1></h1> defines the most important heading.
- <h6></h6> defines the least important heading.

HTML Paragraphs:

- HTML paragraph is defined with the pair of tags.

HTML Anchor Tag:

- The <a>tags known as anchor tag.
- HTML links are defined with the pair of <a>Home tags.
- The link destination is specified in the href attribute.
- Attributes are used to provide additional information about HTML elements.

HTML Images:

- HTML images are defined with the <*img*>tag.
- *The source file, alternative text, height* and *width* are provided as attributes.

HTML Elements:

- An HTML elements is defined by a start tag, some content, and an end tag.
- The HTML element is everything from the start tag to the end tag.
- Example: <h1>Heading</h1>, Some Text. .
- Element content is between the start tag and the end tag.
- Some HTML elements have no content like <*br/*> tag.
- These elements are called empty tags or empty elements.
- Empty elements do not have end tag.
- They will end in the start tag.

Nested HTML Elements:

- HTML elements can be nested.
- All HTML documents consist of nested HTML elements.
- The <*html*> element is the root element and it defines the whole HTML document.
- It has a start tag <*html*> and an end tag <*/html*>.
- Inside the <*html*>element there is a body element <*body*>.
- The <*body*> element defines the document body.
- It has a start tag <*body*> and an end tag <*/body*>.
- Inside the <*body*> element there are two elements <*h1*>,<*p*>.
- The <*h1*> heading defines the heading.
- It has a start tag <*h1*> and an end tag <*/h1*>.
- The <*p*> element defines the paragraph.
- It has a start tag <*p*> and an end tag <*/p*>.

Lesson: HTML

Never Skip the end tag:

- Some HTML elements will display correctly even if you forget the end tag.
- Never relay on this.
- Unexpected results and errors may occur, if you forget the end tag.

Empty HTML Elements:

- HTML elements with no content are called empty elements.
- The <*br/*> tag defines the line break, and is an empty element without a closing tag.
- The <*hr/*> tags is called the horizontal line. This is an empty tag, which closes in the start tag.

HTML is not Case-Sensitive:

- HTML tags are not case-sensitive <*P*> means same as <*p*>.
- The HTML standard does not require lowercase tags.
- W3C recommends lowercase in HTML, and demands lowercase for stricter documents types like XHTML.

HTML Attributes:

- HTML attributes provides additional information about HTML elements.
- All HTML elements can have attributes.
- Attribute usually comes in name/value pair.
- Example: title=”Hello World”
- The <*a*> defines the hyperlink. The href attribute specifies the URL of the page the link goes to.
- The <*img*> tag is used to embed an image in a HTML page. The *src* attribute specifies the path to the image to be displayed.
- The style attribute is used to add styles to an element, such as color, background-color or font-size.
- Remember to always include *Lang* attribute in <*html*> opening tag.
- It helps to declared the language used in formation of the web-page.
- This is used for search engines and web browsers.
- Country codes can also be added to the language code in the *lang* attribute.
- The first two characters defines the language of the HTML page, and the last two characters defines the country.
- The *title* attribute defines some extra information about an element.
- The value of the *title* attribute will be displayed as tool-tip when you mouse over the element.

Absolute URLs:

- Links to an external image that is hosted on another website have absolute URLs.
- You cannot control external images, they can be deleted or moved anytime.
- External images might have copyright.

Lesson: HTML

Relative URLs:

- Links to an image that is hosted within the website have relative URLs.
- If the URL begins without a slash it will be relative to the current page.
- If the URL begins with slash, it will be relative to the domain.
- It is almost best way to use relative URLs.
- They will not break if you changed the domain.
- The <*img*> tag should also contain the *width* and *height* attributes which specifies the width and height of the image.
- The <*img*> tag have another attribute called *alt* attribute abbreviated as alternate text. It is used instead of an image, which is unable to load due to slow internet connection or may have a broken link due to mistype.

HTML Headings:

- HTML headings are titles or subtitles that you want to display on a web page.
- HTML Headings are defines with <*h1*> to <*h6*> tags.
- <*h1*> defines the most important headings, while <*h6*> defines the least important headings.
- Browsers automatically adds white space also called a margin, before and after the headings.
- Search engines use the headings to index the structure and content of your web page.
- Users often skim the web page by its headings.
- It is important to use the headings to show the document structure.
- <*h1*> headings should be use for the main headings and so on according to the level of headings up to <*h6*> headings.
- Use HTML headings only for HTML headings only. Don't use them to look text bold or big.
- Each HTML headings has a default size.
- You can specify the size for any heading with the style attribute, by using the CSS property *font-size* and it value.

HTML Paragraphs:

- A paragraph always starts on a new line and is usually a block of text.
- The HTML <*p*> element defines the paragraph.
- A paragraph always start on a new line, and browser automatically adds white space before and after a paragraph.
- The white space also called a margin.

Lesson: HTML

HTML Display:

- You cannot be sure that how HTML will behave.
- Large, small screens will create different results.
- You cannot change the display by adding extra spaces or extra lines in you HTML file.
- The browser will automatically remove any extra spaces and lines when the HTML file will displayed in the web browser.

HTML Horizontal Rule(HR):

- The *<hr>* tag defines a break in an HTML page and is most often displayed as a horizontal rule.
- The *<hr>* element is use to separate content in an HTML page.

***HTML Line Break(
):***

- The *
* tag is used to add line break in an HTML paragraph.
- You can use *
* tag if you want a line break, without starting a new line.
- The *
* tag is an empty tag, which means that is has no end tag.

HTML PRE Element:

- The HTML *<pre></pre>* element defines preformatted text.
- The text inside the *<pre></pre>* element is displayed in a fixed-width font (courier) and it preserves both spaces and line breaks.

HTML <style> Element:

- The HTML <style> attribute is used to add styles to an element such as
  - ✔ *color*
  - ✔ *font-size*
  - ✔ *etc*
- Setting the style of an HTML element can be done with the *style* attribute.
- The HTML *style* attribute example: <p style=”color: black;”>Some Text.</p>
- The *color* is a CSS property.
- The value *black* is the value assigned to the *color* property.
- The CSS *background-color* property defines the background color for an HTML element.
- The CSS *color* property defines the text color for an HTML element.
- The CSS *font-family* property defines the font to be used for an HTML element.
- The CSS *font-size* property defines the text size for an HTML element.
- The CSS *text-align* property defines the horizontal alignment of the element.
- HTML have several elements for defining text with a special meaning.

Lesson: HTML

HTML Formatting Elements:

- HTML formatting elements were designed to to display special types of text such as:
  - ✔ <*b*></*b*> - Bold Text.
  - ✔ <*strong*></*strong*> - Important Text.
  - ✔ <*i*></*i*> - Italic text.
  - ✔ <*em*></*em*> - Emphasized Text.
  - ✔ <*mark*></*mark*> - Marked Text.
  - ✔ <*small*></*small*> - Smaller Text.
  - ✔ <*del*></*del*> - Deleted Text.
  - ✔ <*ins*></*ins*>- Inserted Text.
  - ✔ <*sup*></*sup*> - Subscript Text.
  - ✔ <*sup*></*sup*> - Superscript Text.
- The HTML <*b*> element defines bold text, without any extra importance.
- The HTML <*strong*> element defines text with strong importance. The content inside typically displayed in bold.
- The HTML <*i*> element defines the content in italic.
- The <*i*> tag is often used to indicate a technical term, a phrase from another language, a thought, a ship name. ETC.
- The HTML <*small*> element defines the smaller text.
- The HTML <*mark*> element defines the text that should be marked or highlighted.
- The HTML <*del*> element defines text that has been deleted from a document. Browser will usually strike a line through deleted text.
- The HTML <*ins*> element defines a text that has been inserted into a document.
- Browser will usually underline inserted text.
- The HTML <*sub*> element defines subscript text.
- Subscript text appear half a character below the normal line, and is sometimes rendered in a smaller font.
- Subscript text can be used for chemical formulas.
- Subscript Example: Ruby Gemstone Chemical Formula: (AI_2O_3).
- The HTML <*sup*> element defines superscript text.
- Superscript text appears half a character above the normal line, and is sometimes rendered in a smaller text.
- Superscript text can be used for mathematical formula : ($a^2+2ab+b^2$).

HTML Element: <blockquote>:

- HTML <*blockquote*> element defines a section that is quoted from another source.
- Browser usually indent <*blockquote*> elements.

HTML element: <q>

- The HTML <*q*> tag defines a short quotation.
- Browser normally inserts quotations mark around the quotations.

Lesson: HTML

HTML element: <abbr>

- The HTML <*abbr*> tag defines an abbreviations or an acronym.
- Marking abbreviations can give useful information to browsers, translations systems and search-engines.
- Use the *title* attribute with <*abbr*> tag to show the description for the abbreviation or acronym when ever user mouse over the specified that element.

HTML Element: <address>

- The HTML <*address*> element defines the contact information for the author or owner of a document or an article.
- The contact information can be an email address, URL, physical address, social media link, phone number e.t.c.
- The text in the <*address*> element usually renders in *italic* font style, and browsers will always adds a line break before and after the *address* element.

HTML Element: <cite>

- The HTML <*cite*> element defines the title of a creative work.
- The text in the <*cite*> element usually renders in *italic* font style.

HTML Element: <bdo>

- BDO stands for Bi-directional override.
- The HTML <*bdo*> tag is used to override the current text direction.

HTML Comments:

- HTML comments are not displayed in the web browser.
- They can help your document HTML source code.
- You can add comment to your HTML source code like this:
- Example:<!-- HTML Comment →
- There is an exclamation mark in the start tag, but not in the end tag.
- With the help of comments you can able to place the notifications as well as reminders in your HTML source code.
- Comments are also helpful in debugging HTML.
- You can comment out the line of code, one by one, to find the exact error in the code.

HTML Colors:

- HTML colors are specified with predefined color names, RGB, HEX code, RGBA, HSL, HSLA values.
- In HTML, a color can be specified by using a color name, such as blue or BLUE.
- You can set the colors of text too, by assigning the value of color to property *color*.
- Borders in HTML can also be colored of your choice.
- By default border will be in white color.
- An RGB color value represents RED, GREEN, BLUE light sources.

Lesson: HTML

- An RGBA color value is an extension of RGB with and alpha channel called opacity.
- In HTML a color can be specified as an RGB values using this formula:
- Formula: *rgb(red, green, blue)*.
- Each parameter (red, green, blue) defines the intensity of the color with a value between (0) to (255).
- To display *black* color with RGB, use (0) value in (RED,GREEN,BLUE) parameters.
- To display *white* color through RGB, use the value (255) in all (RED,GREEN,BLUE) parameters.
- Shades of *gray* color are often defined using equal values for all (3 (RED,GREEN,BLUE)) parameters.
- In RGBA color, (A) alpha channel is a number between (0.0) fully transparent to (1.0) to not transparent at all.
- A hexadecimal color is defined with: (#RRGGBB) format.
- (RR) denotes *RED* color, (GG) is for *GREEN* color and (BB) used for *BLUE* color respectively.
- The hexadecimal value are between (0 – ff), where (*00*) is for lowest value which refers to *black* color and *(ff*) is for *white* color and is the highest value.
- To display *black* color set all parameters to (0) like this. Color: #000000;
- To display *white* color set all parameters to (f) like this. Color: #ffffff.
- In hexadecimal color mode, shades of *Grey* is defines when all (3) parameters value is equal.
- HSL stands for Hue, Saturation and Lightness.
- HSLA color values are an extension of HSL with an Alpha channel which is an opacity of the color.
- Hue is the degree of the color on the color wheel. Starting from (0) to (360).
- (0) is RED, (120) is green and (240) is blue.
- Saturation is the percentage value.
- (0%) means a shade of Grey color and (100%) is the full color.
- Lightness is also the percentage value (0%) is black, (100%) is white light.
- Saturation can be defined as the color intensity. 100% is pure color and no shades of Grey color.
- 50% is 50% Grey color but you can still see the color.
- 0% is completely Grey and you no longer will see the color.
- The lightness of the color can be defined as how much the light you want on your chosen color will be.
- Where 0% means no light. Black Color.
- 50% means 50% light. Neither dark nor light.
- 100% means full light. White color.
- Shades of Grey color are often defined by setting the values for Hue to (0) and for Saturation to (0%) and then adjust the lightness from (0% - 100%), to achieve darker or lighter color.

Lesson: HTML

- In HSLA color values, the Alpha channel is the value between (0.00 – 1.0). Where (0.00) is fully transparent and (1.0) is not transparent at all.

HTML Styles (CSS):

- CSS stands for cascading style sheet.
- CSS saves a lot of work. It can control the layout of multiple web pages all at once.
- Cascading Style Sheet is used to format the layout of the web page.
- With CSS, you can control the color, font, the size of the text, the spacing between elements, how elements are positioned and laid out, what background images and background colors to be used, different displays for different devices and screen sizes and much more.
- The word cascading means that a style applied to a parent element will also applied to all children elements within the parent.
- If you set the color of the body text to *blue*, all headings, paragraphs and other text elements within the body will also get the same color unless you specify some other colors.
- An inline CSS is used to apply a unique style to a single HTML element.
- An inline CSS uses the *style* attribute of an HTML element to style the element.
- An internal CSS is used to define a style for a single HTML element.
- An internal CSS is defined in the <*head*> section of a HTML page. Within a <*style*> element.
- An external style sheet is used to define the styles for many HTML pages.
- To use an external style sheet, add a link to the style sheet file in the <*head*> section of every HTML page.
- <*link*> tag is used to link external style sheet to a web page.
- The external style sheet can be written with the help of any text editor.
- The external style sheet file must not contain any HTML code in it.
- The external style sheet file must be saved with the .*CSS* extension.
- External style sheet can be referenced with a full URL or with a relative path to the current web page.
- With the use of external style sheet, you can change the look of entire website by just changing the one file.
- The CSS *color* property define the text color to be use.
- The CSS *font-family* property defines the font style to use for the text.
- The CSS *font-size* property defines the text size to be used for the desired text.
- The CSS *border* property defines a border around an HTML element.
- You can define the border for nearly all HTML elements.
- The CSS *padding* property defines a padding (some space) between the text and the text border.
- The CSS *margin* property defines the space outside the border;

Lesson: HTML

HTML Links:

- Links are found in nearly in all web pages
- Links allows users to click their way from page to page.
- HTML links are called Hyperlinks.
- You click on a link and jump to another document.
- When you mouse the mouse over a link the mouse arrow will turn into a little handle.
- A link does not have to be a text like.
- A link can be an image or another HTML element.
- The HTML *<a>* tag defines the hyperlink. The most important attribute of the *<a>* element is the *href* attribute, which indicates the link destination.
- The link text will be the part that will be visible to the reader. Clicking on the link text will send the user to the specified URL address.
- An unvisited link is underlined and in blue color by default.
- The color of the visited link is purple and also underlined.
- An active link is in red color and is underlined also.
- HTML links can be styled by CSS to get the brand new look.
- By default, the link page will be displayed in the current browser window.
- To change this behavior of the link, you must specify the value for the *target* attribute of the link.
- The *target* attribute specifies where to open the linked document.
- The possible values for the *target* attribute are as follows:
  - ✔ _self – Default.
  - ✔ _blank – Opens the document in a new window or tab.
  - ✔ _parent – Opens the document in the parent frame.
  - ✔ _top – Opens the document in the full body of the window.
- Absolute URL: A full web address in the *href* attribute of the HTML link.
- Relative URL: A link to the page within the same website in the *href* attribute of the HTML link.
- To use an image as a link just write *<img>* tag inside the *<a>* tag.
- To use link as email address, use *mailto* inside the *href* attribute to create a link that opens the user's email program to let them send a new email.
- To use an HTML button as a link you have to add some JavaScript code.
- JavaScript allows you to specify what happens at certain events such as a click of a button.
- The *title* attribute of the *<a>* tag specifies extra information about the element.
- The information is most often shown as a *tool-tip* text when the mouse moves over the element.
- A link can also be styled as a button.
- HTML links can be used to create bookmarks, so that readers can jump to specific parts of a web page.
- Bookmark can be useful if a web page very long.
- To create a bookmark, first create a bookmark, then add a link to it.

- When the linked is clicked, the page will scroll down up to the location with the bookmark.
- You can also add to a bookmark to another page.

HTML Images:

- Images can improve the design and the appearance of the web page.
- The HTML <*img*> tag is used to embed an image in a web page.
- Images are not technically inserted into a web page, but images are linked to a web page.
- The <*img*> tag creates a holding space for the referenced image.
- The <*img*> tag is empty tag and only contains *attributes* for the specific image, and does not have a losing tag.
- The <*img*> tag has two most important required *attributes* (*src* and *alt*).
- The *src* attribute of the *img* tag specifies the path (URL) ot the actual image.
- Always make sure that the required image should exist at the same location as the web page, or else your visitors will get broken link icon instead of the actual image.
- The broken link icon and the *alternate* text are shown if the browser cannot find the actual required image at the specified location.
- The required *alt* attribute provides an alternate text for an image, if the user for some reason cannot view the image.
- The value of the *alt* attribute should describe the image.
- If the browser image cannot find the image, then it will display the value of the alternate text attribute.
- A screen reader is a software program that reads the HTML code and allows the user to listen the code content.
- Screen reader is useful for the people who are visually impaired or learning disabled.
- You can use the *style* attribute or *img* tag *attributes* to set the height and width of the image. Preferred way is to use *style* attribute of the img tag.
- The *width* and *height* attributes always defines the width and height of the image in pixels.
- Always specifies the *width* and *height* of an image.
- If the width and height are not specified then the web page might flicker while the image loads.
- If the images are in the sub-folder, you must include the folder name in the *src* attribute.
- Some websites points to an external image on another server.
- To point an image on another server you must specify an absolute URL in the *src* attribute of the <*img*> tag.
- External images might be copyrighted.
- If you have not ask for the permission to use their images, then you may be an assailant of copyright laws.
- In addition to copyright law, you cannot control the external images.
- It can be removed or changed anytime on the internet.

Lesson: HTML

- HTML allows animated GIFs.
- GIF stands for **G**raphic **I**nterchange **F**ormat.
- To use an image as a link use the <*img*> tag inside the <*a*> tag.
- Use the CSS *float* property to let the image float to the right or to the left of a text.
- Loading large images takes some time and can make you web page slow.
- With HTML image maps,you can create clickable areas on an image.
- The HTML tag <*map*> defines the image map.
- An image map is an image with a clickable areas.
- The areas are defined with one or more <*area*> tag.
- The idea behind the image map is that you should be able to perform different actions depending n the image you click.
- To create an image map, you need an image and some HTML code that describes the clickable area.
- The image is inserted using <*img*> tag the only difference from other images is that you must add a *usemap* attribute.
- The *usemap* value starts with a # tag followed by the image of the map and used to create a relationship between the image and the image map.
- You can use any image as an image map.
- The <*map*> element is used to create an image map and is linked to an image map by using the *name* attribute.
- The *name* attribute must have the same value and the <*img*>'s *usemap* attribute.
- A clickable area is defined using <*area*> element.
- You must define the shape of clickable area an the values are:
  - Rectangle(shape = "rect").
  - Circle(shape = "circle").
  - Polygon(shape="poly").
  - Default(shape="default").
- You must also define some coordinates top be able to place the clickable area onto the image.
- The coordinate for the rectangle shape come in pair, one for the x-axis and second for the y-axis.
- In circle area first locate the coordinate of the center of the circle. Then specify the radius of the circle.
- The polygon shape contains many points which creates a shape formed with straight lines.
- This shape can be used to create any other shape also.
- A clickable are can also trigger a JavaScript function.
- Add a click event in the <*area/*> tag to execute a JavaScript function.
- A background image can be specified for almost any HTML element.
- To add a background image on an HTML element, use the HTML *style* attribute as well as CSS *background-image* property.
- Background-images are part of the CSS, so it can be used with CSS only.

Lesson: HTML

- If you want the entire page have a background image, you must specify the background image on the <*body*> element.
- If the background image is smaller than the element, the image will repeat itself. In both horizontally and vertically until it reaches the end of the element.
- To avoid the background image to repeat itself, use the *background-repeat* property with *value:norepeat*.
- If you want image to cover entire element you can set the *background-size* property to *cover*.
- To make sure the entire element is always covered set the *background-attachment* property to *fixed*.
- This way the entire element will be covered by the background image, with no stretching the image, and the image will remain in its original skin.
- If you want the background image to stretch to fit the entire element you can set the *background-size* property to 100% in both *height* and *width*.
- The HTML <*picture*> element allows us to display different pictures for different devices of screen sizes.
- The HTML <*picture*> element gives web developers more flexibility in specifying image resources.
- The <*picture*> element contains one or more <*source*> elements each refers to a different pictures through the *srcset* attribute.
- This way the browser can choose the image that best fits the current view and or device.
- Each <*source*> element has a *media* attribute that defines when the image is the most suitable.
- Always specify the an <*img/*> element as the child element of the <*picture*> element.
- <*img/*> element is used by browsers that do not supports the <*picture*> element or if none of the <*source*> element matches.
- There are two main purpose for the <*picture*> element to be used.
  - Bandwidth.
  - Format Support.
- The browser will use the first support <*source*>element with matching attribute values and ignore anything else following <*source*> element.

HTML Tables:

- HTML tables allows web developers to arrange data into rows and columns.
- The <*table*> tag defines an HTML table
- Each table row is defines with the <*tr*> tag.
- Each table header is defined with the <*th*>tag.
- Each table data cell is defined with <*td*> tag.
- By default, the data in the table header (<*th*>) tag are bold and center.
- By default the data in table cell (<*td*>) are regular and left-align.
- The <*td*> element are the data containers of the table.
- They contain all sorts of HTML elements text, images, lists, other tables e.t.c.

Lesson: HTML

- To add border to a table use the CSS *border* property. Define borders for both table rows and table columns.
- In order to collapse table border into only one border for the table, use the CSS *border-collapse* property with value *collapse* assigned.
- Cells padding specify the space between the cell content and its border.
- If you don't specify the cell padding, the table cells will be displayed without padding.
- To set the padding use the *padding* property of CSS.
- To left align the table headings use the CSS *text-align* property with value *left*.
- Border spacing specifies the space between the cells.
- To set the border spacing for a table cells, use the CSS *border-spacing* property.
- If the tables has collapse border, *border-spacing* property has no effect.
- To make a cell span more than one column, use the *colspan* attribute of CSS.
- To make a cell span more than one row, use the *rowspan* attribute.
- To add a caption to a table use the <*caption*> tag.
- The <*caption*> tag must be inserted immediately after the <*table*> tag.
- To define special style table use the *id* attribute of the CSS within the <*table*> tag.

HTML Lists:

- HTML lists allows web developers to group a set of related items.
- An unordered list starts with <*ul*> tag.
- Each list items starts with the <*li*> tag.
- The list items marked with bullets by defaults.
- An ordered list starts with <*ol*> tag.
- Each list item starts with <*li*> tag.
- The <*li*> tag in <*ol*> list will be marked with numbers by default.
- HTML also supports description list.
- A description list is a list of terms, with the description of each terms.
- The <*dl*> tag defines the description list.
- The <*dt*> tag defines the description term or title for the list.
- The <*dd*> describes the each term.
- List can be nested.
- Nested means list inside in another list.
- A list item <*li*> can contain a new list and other HTML elements like images and links.
- HTML lists can be styled in many different ways with CSS.
- One popular way is to style a list horizontally to create a navigation menu.
- An ordered list can be numerical or alphabetical.
- The *type* attribute of the <*ol*> tag, defines the numbering type used for the list item markers.
- The *list-style-type* CSS property is used in <*ul*> tag to change the bullets styles used in the list.
- By default an ordered list will starts counting from 1.

Lesson: HTML

- If you want to start the counting from a desired number you can use the *start* attribute followed by you chosen number in the <*ol*> tag.

HTML Block and Inline Element:

- Every HTML element has a default display value, depending what type of element it is.
- There are two display values
  - ✔ Block.
  - ✔ Inline.
- A block level element always starts on a new line and takes up the full width available.
- <*div*> element is a block level element.
- An inline element does not start on a new line and it only takes up as much width as necessary.
- An inline element cannot contain block level element.
- The <*div*> element is most often used as a container for other HTML elements.
- The <*div*> element has no required attributes by *style*, *class*, and *id* are most often used.
- When used together with the CSS, the <*div*> element can be used to style blocks of content.
- The <*span*> element is an inline container used to mark up a part of a text or a part of a document.
- The <*span*> element has no required attributes, but you can use *id*, *class* and *style* attribute with that.
- When used together with the CSS, the <*span*> element can be used to style part of the text.

HTML Attribute:(class)

- The HTML *class* attribute is used to specify the class for an HTML element.
- Multiple HTML elements can share the same class.
- The *class* attribute is often used to point to a class name in a style sheet.
- It can also be used by the JavaScript to access and manipulate elements with the specific class name.
- HTML *class* attribute can be used on any HTML element.
- The *class* name is case-sensitive.
- HTML elements can belongs to more than one class.
- To define multiple classes, separate the *class names* with a space. The element will be styled according to all the classes specified.
- Different HTML elements can point to the same *class* name.
- The *class name* can also be used by JavaScript to perform certain tasks for certain elements.
- JavaScript can access the element by method *getElementsByClassName()*;

Lesson: HTML

HTML Attribute (ID):

- The HTML *id* attribute is used to specify the unique id for an HTML element.
- You cannot have more that one element with the same id in an HTML document.
- The *id* attribute is used to point to a specific style declaration in a style sheet.
- It is also used by JavaScript to access and manipulate the element with the specific id.
- The syntax for writing id attribute is that: type the hash symbol (#) followed by the name for id attribute and then the pairs of opening and closing tags.
- Between these tag you can specify the styles for the specified id.
- The *id* name is case-sensitive.
- The *id* name must contain at least one character and must not contain white spaces or tabs.
- A *class name* can be used by multiple HTML elements while an *id name* must only be used by one HTML element within the page.
- HTML bookmarks are used to allow readers to jumps to specific parts of the web page.
- Bookmarks are very useful if your web page is very long.
- To use a bookmark, you must create it, and then link to it.
- JavaScript can access an element with a specific *id* with the *getElementById()* method.

HTML Tag <iframes>:

- An HTML *iframe* is used to display a web page within a web-page.
- The HTML <*iframe*> tag specifies an in-line frame.
- An inline frame is used to embed another document within the current document.
- It is always a good practice to include the *title* attribute for the <*iframe*> tag. This is used by the screen readers to read out what the content of the *iframe* is.
- Use the *height* and *width* attributes to specify the size of the iframe. The *height* and *width* is specified in pixels by default.
- You can also use the *height* and *width* attribute of the *style attribute* too.
- An iframe can be used as the target an anchor tag.
- The *target* attribute of the anchor tag must be refer to the *name* attribute of the iframe tag.

HTML JavaScript:

- JavaScript makes HTML pages more dynamic and interactive.
- The HTML <*script*> tag is used to define a client-side script I.e: JavaScript.
- The <*script*> element either contain script statements, or it points to an external script file through the *src* attribute.
- Common uses of JavaScript are image manipulation, form validation and dynamic changes of content.
- To select and HTML element JavaScript element is most often used *document.getElementById()* method for the selection.
- JavaScript can change content.
- JavaScript can change style.

Lesson: HTML

- JavaScript can change attribute.
- The HTML <*noscript*> tag defines an alternate content to be displayed to users that have disabled scripts in their web browsers or have a web browser that doesn't support scripts.

HTML File Paths:

- A file path describes the location of a file in a website folder structure.
- File paths are used when linking to external files like web-pages, images, style sheet and JavaScript.
- An absolute file path is the full URL to a file. Absolute file path can also be referred as complete file path.
- A relative file path refers to a file relative to the current page.
- It is best practice to use relative files paths if possible.
- When using relative file paths your web pages will not be bound to your current base URL.
- All links will work on your own computer (localhost), as well as on your current public domain and also on your future public domains.

HTML <head> Element:

- The HTML <*head*> element is a container for: <*title*>, <*style*>, <meta>, <*link*>, <*script*>and <*base*>.
- The <*head*> element is the container for metadata that is data about data and is placed between the <*html*> tag and the <*body*> tag.
- HTML metadata is about the HTML document.
- Meta data is not displayed.
- Metadata typically defines the document title, character set, styles, scripts, and other meta information.

HTML <title> Element:

- The <*title*> element defines the title of the document.
- The title element must be text-only, and it is shown in the browser's title bar or in the page's tab.
- The <*title*> tag is required in HTML document.
- The content of the page title is very important for Search Engine Optimization (SEO).
- The title is used by Search Engine algorithms to decide the order when listing pages in search engine results.
- The <*title*> element defines the title in the browser toolbar.
- The <*title*> element defines the title for the page when it is added to favorites.
- The <*title*> element displays a title for the page in search engine results.
- Try to make the title as accurate and meaningful as possible.
- The <*style*> element is used to define style information for a single HTML page.

Lesson: HTML

HTML <link> element:

- The <*link*> element defines the relationship between the current document and an external resource.
- The <*link*> element is most often used to link to an external style sheet.

HTML <meta> element:

- The <*meta*> element is typically used to specify the character set, page description, keywords, authors of the document and viewport settings.
- The metadata will not be displayed on the page but are used by the web browsers, how to reload the page or to display the content. By search engines keywords and other web services.
- The viewport is the users visible area of the web page. It varies with the device.
- It will be smaller on a mobile phone than on a computer screen.
- This instructs the browser on how to control the page's dimensions and scaling.
- The *width=device-width* part sets the width of the page to follow the screen width of the device which will vary depending on the device.
- The *initial-scale=1.0* part sets the initial zoom level when the page is first loaded by the devices browser.
- The <*script*> tag is used to define client-side JavaScript.

HTML <base> element:

- The <*base*> element specifies the URL and or target for all relative URLs in a page.
- The <*base*> tag must have either an *href* or a *target* attribute present or both.
- There can be only a single <*base*> element in a document.

HTML Layouts Elements and Technique:

- Websites often display content in multiple columns like a magazine.
- HTML have several semantics elements that define the several parts of a web page.
- <*header*> - Defines the header for the document or a section.
- <*nav*> - Defines the set of navigation links.
- <*section*> - Defines the section of a document.
- <*article*> - Defines an independent self-contained content.
- <*aside*> - Defines the content aside from the content like a sidebar.
- <*footer*> - Defines a footer for a document or a section.
- <*details*> - Defines additional details that the use can open and close on demand.
- <*summary*> - Defines the heading for the <*details*> element.
- There are four different techniques to create multi column layouts.
- Each technique has its own pros and cons.
- These techniques are:
  - CSS framework.
  - CSS float property.
  - CSS flexbox.
  - CSS grid.

Lesson: HTML

CSS Framework:

- If you want to create your layout fast you can use CSS framework like Bootstrap.

CSS Float-property:

- It is common to do entire website layout using the CSS *float* property.
- Float is easy to learn, you just need to remember how the *float* and *clear* property works.
- Floating elements are tied to the document flow which may harm the flexibility of the web-page.

CSS Flexbox Layout:

- Use of flexbox ensures that elements behaves predictably when the page layout must accommodate different screen sizes and different display devices.

CSS Grid Layout:

- The CSS grid layout module offers a grid-based layout system with rows and columns, making it easier to design web pages without having to use *floats* and *positioning*

HTML Responsive Web Design:

- Responsive web design is about creating web pages that look good on all devices.
- A responsive web design will automatically adjust itself for different screen sizes and viewports.
- Responsive web design is about HTML, CSS to automatically hide, resize, shrink and enlarge a web-page to look good on all devices (desktops, laptops, tablets, mobiles and other).
- To create a responsive website add the following *<meta>* tag to all your web-pages.
- Tag: *<meta name = "viewport" content="width='device-width', initial-scale: '1.0'"/>*
- this will set the viewport of your web page which will give the browser instruction on how to control the page's dimensions and scaling.
- Responsive images are images that scale nicely to fit any browser size.
- If the CSS *width* property for image is set to *100%* then the image will be responsive and scale up and down.
- A better solution is to use *max-width* property for images.
- If the *max-width* property is set to *100%* the image will scale down if it has to, but never scale up to be larger then its original size.
- The HTML *<picture>* element allows you to define different images for different browser window sizes.
- Viewport is the browser window size, 1vw = 1% of viewport width.
- If the viewport is *50cm* wide. 1Vw = 0.5cm

Lesson: HTML

Media Queries:

- in addition to resize text and images it is also common to use media queries in responsive web design.
- With media queries you can completely redesign the web page for different browser sizes.

Responsive Web Page:

- A responsive web page should look good on large desktop screens and on small mobile phones.
- All popular CSS frameworks offer responsive web design, and they are free as well ass easy to use.
- Bootstrap uses HTML, CSS and jQuery to develop responsive web pages.

HTML Computer Code Elements:

- HTML contains several elements for defining computer code and user input.
- The HTML <*kbd*> elements is used to define keyboard input. The content inside is displayed in the browsers default monospace font.
- The HTML <*samp*> element is used to define sample output from a computer program. The content inside is displayed in the browsers default monospace font.
- The HTML <*code*> element is used to define a piece of computer code. The content inside is displayed in browser default monospace font.
- The <*code*> element does not preserve extra white space and line-breaks. To fix this you can write the <*code*> element inside a <*pre*> element.
- The HTML <*var*> element is used to define a variable in programming or in a mathematical expression. The content inside is typically displayed in italic.

HTML Semantic Elements:

- Elements with meaning are called Semantics Element.
- A semantic element clearly defines its meaning to both the browser and the developer.
- Some non-semantics elements are: <*div*> and <*span*>.
- Some semantic elements are: <*form*>, <*table*> and <*article*>.
- Many website contains HTML code like <*div class=”header”*> and also <*div class=”footer”*> to indicate header and footer.
- A semantic web allows data to be shared and reused across applications, enterprises and communities.

HTML <section> Element:

- The HTML <*section*> element defines a section in a HTML document.
- A section is a thematic grouping of content, typically with a heading.
- A web page could be normally split into sections for introduction, content and contact information.

Lesson: HTML

HTML <article> Element:

- The HTML <*article*> element specifies independent, self-contained content.
- An article makes sense on its own and it should be possible to distribute it independently from the rest of the website.
- You can use CSS to style the <*article*> element.

HTML <header> Element:

- The HTML <*header*> element represents a container for introductory content or a set of navigational links.
- You can have several <*header*> elements in one HTML document.
- However <*header*> cannot be placed within a <*footer*>, <*address*> or another <*header*> elements.

HTML <footer> Element:

- The HTML <*footer*> element defines a footer for a document or a section.
- You can have several <*footer*> elements in one document.

HTML <nav> Element:

- The HTML <*nav*>element defines a set of navigation links.
- Not all links of the document should be inside <*nav*> element.
- The <*nav*> element is intended only for major block of navigation links.
- Screen readers can use this element to determine whether to omit the initial rendering of this element.

HTML <aside> Element:

- The HTML <*aside*> element defines some content aside from the content it is placed in like a sidebar.
- The <*aside*> content should be indirectly related to the surrounding content.
- Use CSS to style the <*aside*> element.

HTML <figure> Element:

- The <*figure*> tag specifies self-contained content like illustrations, diagrams, photos, code listings etc.
- The <*figcaption*> tag defines a caption for the <*figure*> element.
- The <*figcaption*> element can be placed as the first or as the last child element of the <*figure*> element.
- The <*img*> element defines the actual image or illustration.

HTML Style Guide and Coding Conventions:

- A well written HTML code is helpful for others to read and understand your code.
- Always declared the document type as the first line in your document.
- HTML allows mixing uppercase and lowercase letters in element names.

Lesson: HTML

- Mixing uppercase and lowercase elements names looks bad.
- Developer use lowercase names.
- Lowercase looks cleaner.
- Lowercase is easier to write.
- In HTML you do not have to close all elements.
- We strongly recommends closing all HTML elements.
- We recommends lowercase attribute names.
- HTML allows attribute values without quotes.
- We recommend quoting attribute values.
- Always specify the *alt* attribute for images. This attribute is important if the image for some reasons cannot be displayed.
- Always define the *height* and *width* of images. This reduce flickering, because the browser can reserve space for the image before loading.
- HTML allows spaces around equal signs. But space-less is easier to read and group entities better together.
- When using an HTML editor, it is not convenient to scroll right and left to read the HTML code.
- Try to avoid too long code lines.
- Do not add blank lines, spaces or indentations without any reasons.
- For readability add blank lines, to separate large or logical code blocks.
- For readability add two spaces of indentation.
- Do not use the tab key while indentation. Use spacebar instead for two line spaces.
- The <*title*> element is requires in HTML.
- The content of a page title is very important for search engine optimization (SEO).
- The page title is used by the search engines algorithms to decide the order when listing pages in search results.
- The <*title*> element displays the title for the page in search engine results.
- Try to make the title as accurate and meaningful as possible.
- An HTML page will validate without the <*html*> and <*body*> tags.
- It is strongly recommended that to always add the <*html*> and <*body*> tag.
- Omitting <*body*> tag can produce errors in older browsers.
- Omitting <*html*> and <*body*> can also crash DOM and XML software.
- The HTML <*head*> tag can also be omitted.
- Browser will add all elements before <*body*> to a default <*head*> element.
- We recommend using <*head*> element.
- In HTML it is optional to close empty elements.
- If you expect XML/XHTML software to access your page keep the closing slash(/) because it is required in XML and XHTML.
- You should always add the *lang* attribute inside the <*html*> tag to declared the language of the web-page. This is meant to assist search engines and browsers.
- To ensure proper interpretations and correct search engine indexing both the language and character coding should be defined as early as possible in an HTML document.

Lesson: HTML

- The viewport is the visible area of the web page. It varies with the device, it will be smaller on a mobile phone than on a computer screen. This instructs the browser on how to control the page dimensions and scaling.
- Some web servers (Apache, Unix) are case sensitive about file names.
- Other web servers (Microsoft IIS) are not case-sensitive.
- If you use the mix of uppercase or lowercase you have to be aware of this.
- If you move from a case insensitive to case sensitive servers even small errors can break your website.
- To avoid these kinds of silly errors, always use lowercase file names.
- When a URL does not specify the filename at the end. The server just adds the default file name (index.html, index.htm, default.html, default.htm).
- Try to use very first file name as any one of them (index.html, indexhtm, default.html, default.htm).
- Servers can be configured with more than one default file name.
- Usually you can setup as many default file names as you want.

HTML Comments:

- Short comments should be written on one line.
- Example Comment: <!-- This is HTML Single-line Comment →
- Comment which can more then one line should be written like this
- Example Comment:
- <!--
- This is
- Multi line
- Comments-→
- long comments are easier to observe if they are indented with two spaces.

Using Style Sheets:

- Use simple syntax for linking to style sheet the *type* attribute is not necessary.
- Short CSS rules can be written compressed.
- Long CSS rules should be written on multiple lines.

Loading JavaScript in HTML:

- Use simple syntax for loading external scripts the *type* attribute is not necessary.
- Using untidy HTML code can result in JavaScript errors.

File Extensions:

- HTML files should have a (.html) or (.htm) file extension.
- CSS file should have (.css) file extension.
- JavaScript should have (.js) file extensions.
- There is no difference between (.html) or (.htm) file extension. Both will be treated as HTML by any web browser and web server.

Lesson: HTML

HTML Entities:

- Reserved characters in HTML must be replaced with character entities.
- If you use less than (<) or greater than (>) signs in your text the browser might mix them with HTML element tags.
- Character entities are used to display reserved characters in HTML.
- The advantage of using entity names is that entity names are easier to remember.
- Disadvantage of using entity names is that browsers may not support all entities names, but entities numbers are good to go.
- A common used entity in HTML is the non-breaking space ( ).
- A non-breaking space is a space that will not break into a new line.
- Two words separated by a non-breaking space will stick together, will not break into two lines. This is handy when breaking the words might be disruptive.
- Another common use of non-breaking space is to prevent browsers from truncating spaces in HTML pages.
- If you write 10 spaces in your text, the browser will remove 9 of them.
- To add real spaces to your text you can use ( ) character entity.
- Entity name are case-sensitive.
- Copyright Symbol Entity Number(©): (©).
- Registered Trade Mark Entity Number(®): (®).
- Trade Mark Entity Number(™): (ℤ).
- A diacritical mark is a glyph added to a letter.
- Some diacritical marks like grave (ỳ) and acute (ý) are called accents.
- Diacritical marks can appear both above and below a letter, inside a letter, and between two letters.
- Diacritical marks can be used in combination with alphanumeric characters to produce a character that is not present in the character set, encoding is used IN THE PAGE.
- Symbols that are not present on your keyboard can also be added by using HTML entities.
- Many mathematical, technical and currency symbols are not present on a normal keyboard.
- To add a such symbol to an HTML page, you can use an entity name or entity number, a decimal or a hexadecimal reference for the symbol.

HTML Emojis:

- Emojis are characters from the UTF-8 character set.
- Emojis look like images or icons, but they are not.
- They are letters or characters from the UTF-8 Unicode character set.
- UTF-8 almost covers all of the characters and symbols of the world.
- To display an HTML page correctly, a web browser must know the character set used in the page.
- This is specified in the *<meta>* tag.
- If not specified, UTF-8 is the default character set in HTML.

Lesson: HTML

- Many UTF-8 characters cannot be typed from the keyboard.
- They can always be displayed by using numbers called (Entity Numbers).
- To let the browser display the character you are willing to display, you must start the entity number with (&#) and end it with semicolon (;).
- Example: #&169; ™ ® for (Copyright, Trade Mark and Registered Trade Mark) Respectively.
- Emojis are also characters from the UTF-8 alphabet.
- Since emojis are characters they can be copied, displayed, and sized just like any other character in HTML

HTML Encoding:

- To display an HTML page correctly a web browser must know which character set to use.
- ASCII was the first character encoding standard.
- ASCII: American Standard Code for Information Interchange
- ASCII defines 128 different character that could be used on the internet: number (0 – 9), English letters (A – Z, a - z), and special characters some of them are ($, !, <,>,+,-,(,)).
- ISO-8859-1 was the default character set for HTML4.
- This character set supported 256 different character codes.
- HTML also supports UTF-8.
- ANSI (Windows-1252) was the original Windows character set.
- ANSI: America National Standards Institute.
- ANSI is identical to ISO-8859-1 character encoding.
- ANSI have more 32 extra characters added to ISO-8859-1 character encoding.
- The default character set for HTML5 is UTF-8.
- UTF-8 covers almost all characters and symbols in the world.
- ASCII uses the values from (0 – 31) and (127) for control characters.
- ASCII uses values from (32 to 126) for letters, digits and symbols.
- ASCII does not use values from (128 – 255).
- ANSI is identical to ASCII for the values from (0 – 127).
- ANSI has a proprietary set of characters for the values from (128 – 159).
- ANSI is identical to UTF-8 for the values from (160 – 255).
- ISO-8859-1 is identical to ASCII for the values from (0 – 127).
- ISO-8859-1 does not used the values from (128 – 159).
- ISO-8859-1 is identical to UTF-8 for the values from (160 – 255).
- UTF-8 is identical to ASCII for the values from (0 – 127).
- UTF-8 does not use the values from (128 – 159).
- UTF-8 is identical to both ANSI and ISO-8859-1 for the values from (160 – 255).
- UTF-8 continues from the value (256) with more than (10000) different characters.

Lesson: HTML

HTML Uniform Resource Locator:

- A URL is another word for a web address.
- A URL can be composed of words (google.com) or an Internet Protocol (IP) address (172.217.19.4)
- Most people enter the name when surfing, because names are easier to remember than numbers.
- Web browsers request pages from web servers by using a URL.
- A Uniform Resource Locator (URL) is used to address a document or other data on the web.
- **Scheme**: defines the type of internet service, most common is HTTP and https.
- **Prefix**: defines a domain prefix. Default for HTTP is www.
- **Domain**: defines the Internet Domain Name.
- **Port**: defines the port number at the host. Default for HTTP is 80.
- **Path**: defines a path at the server. If omitted the root directory of the site.
- **Filename**: defines the name of a document or resource.
- URLs can only be sent over the internet using the ASCII character set.
- If the character is used outside the ASCII character set the URL has to be converted.
- URL converts non-ASCII characters into a formatted that can be transmitted over the internet.
- URL converts non-ASCII characters with a (%) followed hexadecimal digits.
- URL cannot contain spaces.
- URL normally replaces a space with a plus (+) sign or (%20) literal.
- Your browser will encode input according to the character set used in your page.

HTML vs XHTML:

- XHTML is a stricter, XML based version of HTML.
- XHTML stands for Extensible Hypertext Markup Language.
- XHTML is defined as an XML based application.
- XHTML is supported by all major browsers.
- XML is a markup language where all document must be marked up correctly. Well-Defined.
- XHTML was developed to make HTML more extensible and flexible to work with other data formats such XML.
- Browsers ignore errors in HTML pages, and try to display the website even if it has some errors in the markup.
- XHTML comes with much stricter error handling.

Lesson: HTML

Important To XHTML:

- <!doctype html> is mandatory.
- The xmlns attribute in <html> is mandatory.
- <html>,<head>. <title> and <body> tags are mandatory.
- Elements must always be properly nested.
- Elements must always be closed and in lowercase.
- Attribute names and values must always be in lowercase.
- Attribute values must always be quoted.
- Attribute minimization is forbidden.

General URL Schemes:

Scheme	Short For	User for
HTTP	Hype Text Transfer Protocol	Common Web pages
https	Hyper Text Transfer Protocol Secure	Encrypted Web Pages
ftp	File Transfer Protocol	Downloading or Uploading Files
file		A file on your computer

HTML Forms:

- ✔ The HTML <form> element is used to create an HTML form for user input.
- ✔ The <form> element is a container for different types of elements such as text fields, text boxes, radio buttons, submit buttons etc.
- ✔ The HTML <input> element is the most useful form element.
- ✔ An <input> element can be displayed in many forms depending upon **type** attribute.
- ✔ Some type attribute values are:
  - Text.
  - Radio.
  - Checkbox.
  - Submit.
  - Button.
  - Reset.
  - File
- ✔ The <input type=”text”/> defines a single line input field for text input.
- ✔ The form itself is not visible.
- ✔ The default width of an input field is 20 characters.
- ✔ The <label> element defines the label for many form elements.
- ✔ The <label> element is useful for screen reader users, because the screen reader read out loud the label when the user focus on the input element.
- ✔ The label element is also useful for users who have difficulty clicking on very small regions such as radio buttons and check boxes because when the user clicks on the label element it toggles the radio button/checkbox element.

Lesson: HTML

- ✔ The *for* attribute for the label element should be equal to the *id* attribute of the <*input*> element to bind them together.
- ✔ The <*input type=”radio”*/> defines the radio button.
- ✔ Radio button let the user select one of a limited number of choices.
- ✔ The <*input type=”checkbox”*/> defines the checkbox.
- ✔ Check boxes let users select number of options from a limited number of choices.
- ✔ The <*input type-”submit”*/> defines the button for submitting the form data to a form handler.
- ✔ The form-handler is typically a file on the server with the script for handling the form data.
- ✔ The form handler is specified in the action attribute of the <form> element.
- ✔ Each input element has name attribute to be submitted.
- ✔ If the *name* attributed is omitted, the *value* of the input field will not be sent at all.
- ✔ The *action* attribute defines the action to be performed when the form is submitted.
- ✔ The form data is sent to the file on the server. When the user clicks on the submit button.
- ✔ If the action attribute is omitted the action is set to the current page.
- ✔ The target attribute specifies where to display the response that is received after submitting the form.
- ✔ The target attribute can have on of the following values:
  - _blank.
  - _self.
  - _parent
  - _top
  - framename.
- ✔ The default value of the target attribute is _self which means response will be open in the current window.
- ✔ The *method* attribute specifies the HTTP method to be used with HTML forms.
- ✔ The form-data can be sent as URL variables with (method=”get”) or as HTTP post transaction with (method=”post”).
- ✔ The default method used with HTTP method is get.
- ✔ The auto-complete attribute specifies whether a form should have an auto-complete on or off.
- ✔ When auto-complete is on the browser will automatically complete values based upon the user has entered before.
- ✔ The novalidate is a Boolean attribute. When present it specifies that the form-data should not be validated when submitted.
- ✔ The <*select*> element defines the drop-down list.
- ✔ The <*option*> element defines the options for the select element.
- ✔ By default the very first item is selected by default in the drop-down list.
- ✔ To define a pre-selected option by default, use the selected attribute in the option tag.
- ✔ Use the size attribute is select element to specify the number of visible values.

Lesson: HTML

- ✔ Use the multiple attribute value in select element to allow multiple selection in drop down list.
- ✔ The <*textarea*> element defines a multiple lines text area for input data.
- ✔ The row attribute in text area specifies the number of lines in a text area.
- ✔ The col attribute of text area specifies the visible width of a text area.
- ✔ You can also define the text area size by using CSS.
- ✔ The <*button*> element defines a clickable button. Always specify the type attribute for the button element. Different browsers will have different type for the button element
- ✔ the <*fieldset*> element is used to groups related data in a form.
- ✔ The <*legend*> element defines the caption for the <*fieldset*> element.
- ✔ The list attribute of the input element must refer to the id attribute of the datalist element.
- ✔ The <*output*> element represents the result of a calculations one performed by a script
- ✔ The formaction attribute specifies the URL of the file the will process the input when the form is submitted. This attribute override the action attribute of the form element. This attribute works with the submit and image types.
- ✔ The input enctype attribute specifies how the form data should be encoded when submitted, only for form method=”post”. It works with types submit and image.
- ✔ The input formmethod attribute specifies HTTP method for sending form-data to the action URL. This method will override the method attribute of the form element. This attribute works with submit and image types.
- ✔ The form data can be sent as URL variables (method = “get”), or as an HTTP post transaction (method = “post”).
- ✔ The input formtarget attribute specifies a name or a keyword that indicates where to display the response that is received after submitting the form. This attribute overrides the target attribute of the form element. It works with input types: submit and image.
- ✔ The input element formnovalidate attribute specifies that an input element should not be validated when submitted. This attribute overrides the novalidate attribute of the form element. It works with input type submit only.
- ✔ The novalidate attribute is a form element attribute. This attribute specifies that the form-data should not be validated when submitting the to the server script.
- ✔

HTTP GET Method:

- ✔ Append the form data to the URL, in name and values pair.
- ✔ Never use GET method to send sensitive data.
- ✔ The submitted data is visible in the URL.
- ✔ The length of the URL is limited to (2048 characters).
- ✔ Useful for form submission where the user want to bookmark the result.
- ✔ Get method is good for non-secured data like query strings in google.

Lesson: HTML

HTTP POST Method:

- ✔ Append the form data inside the body of the HTTP request.
- ✔ POST has no size limitations and can be used to send large amounts of data.
- ✔ Form submission with POST cannot be bookmarked.
- ✔ Always use POST method if the form data contains sensitive or personal information.

HTML <form> Attribute:

- ✔ The input form attribute specifies the form the input element belong to.
- ✔ The value of the attribute must be equal to the id attribute value of the form tag.

HTML SVG Graphics:

- ✔ SVG defines vector based graphics in XML format.
- ✔ SVG stands for Scaleable Vector Graphics.
- ✔ SVG is used to define graphics for the web.
- ✔ SVG is aW3C recommendations.
- ✔ The HTML <svg> element is a container for SVG graphics.
- ✔ SVG has several methods for drawing paths, boxes, circles, text and graphics images.

SVG vs Canvas:

- ✔ SVG is a language for describing 2D graphics in HTML.
- ✔ Canvas draws 2d graphics on the fly,with JavaScript.
- ✔ SVG is XML based which means that every element is available within the SVG DOM.
- ✔ You can attach JavaScript event handlers in SVG elements.
- ✔ In SVG, each drawn shape is remembered as an object.
- ✔ If attributes of an SVG object are changed, the browser can automatically re-render the shape.
- ✔ Canvas is rendered Pixel by Pixel.
- ✔ In canvas, once the graphics is drawn it is forgotten by the browser. If its position should change, the entire scene needs to be drawn, including any objects that might have been covered by the graphics.

Comparison Canvas vs SVG:

Canvas	SVG
✔ Resolution Dependent. ✔ No support for event handler. ✔ Poor text rendering capabilities. ✔ You can save the resulting image as .png, .jpg. ✔ Well suited for graphics-intensive games.	✔ Resolution independent. ✔ Support for event handlers. ✔ Best suited for application with large rendering areas. ✔ Slow rendering if possible, any thing that uses DOM will be slow. ✔ Not suited for game applications.

Lesson: HTML

HTML Multimedia:

- ✔ Multimedia on the Web is:
  - ➢ Sound.
  - ➢ Music.
  - ➢ Videos.
  - ➢ Movies.
  - ➢ Animations.
  - ➢ Images.
  - ➢ Films.
  - ➢ Records.
- ✔ Multimedia can have many formats.
- ✔ Web page often contains multimedia elements of different types and formats.
- ✔ The very first browser has support for text only, limited to a single font in a single color.
- ✔ Latest browsers has support for color, fonts, images, multimedia.
- ✔ Multimedia elements like audios and videos are stored in media files.
- ✔ The most common way to discover the type of a file, is to look at the file extension.
- ✔ Multimedia files have different formats and extensions like:
  - ➢ .wav
  - ➢ .mp3
  - ➢ .mp4
  - ➢ .mpg
  - ➢ .wmv
  - ➢ .avi
- ✔ There are many video formats are there.
- ✔ The mp4, WebM and OGG format are supported by HTML.
- ✔ The mp4 format is recommended by YouTube.
- ✔ Mp3 is the best format for compressed recorded music.
- ✔ The mp3 has become synonymous with digital music.
- ✔ If your website is about recorded music, mp3 is the best choice.

HTML <video>:

- ✔ The HTML <video> element is used to show a video on a web-page.
- ✔ The control attribute add the video controls, like play, pause and volume.
- ✔ It is always goo idea to add height and width attributes. If height and width are not set the page might flicker while the video loads.
- ✔ The <source> element allows you to specify alternative video files which the browser may choose from. The browser will use the first recognized video format.
- ✔ The text between <video> and </video> tag will only be displayed in browsers that do not support the <video> element.
- ✔ To start the video automatically, use the *autoplay* attribute. The autoplay feature does not work on mobile devices.
- ✔ There re three supported video formats:

- MP4.
- WebM.
- Ogg.

✔ The media type for Mp4 will be video/mp4.
✔ The media type for WebM will be video/webm.
✔ The media type for Ogg will be video/ogg.
✔ The HTML DOM defines the methods, properties and events for the <*video*> element. This allows you to play, pause video, as well as set the duration and volume for the current video.
✔ There are also DOM events that can notify you when a video begins to play, is paused etc.

HTML <audio> Element:

✔ The HTML <*audio*> element is used to play an audio file on the web page.
✔ The controls attribute add the audio controls like play, pause and volume.
✔ The <source> element allows you to specify alternative audio files which the browser may choose from. The browser will use the first recognized format.
✔ The text between <audio> and </audio> elements will only be displayed in browsers that do not support the <audio> element.
✔ There are three <audio> supported formats:
- MP3.
- WAV.
- Ogg.

✔ The media type of MP3 is audio/mpeg.
✔ The media type for WAV is audio/wav.
✔ The media type for OGG is audio/ogg.
✔ The HTML DOM defines properties, methods and event for the <audio> element. This allows you to play, pause audios as well as set duration and volume for the audio files.
✔ There are also DOM events that ca notify you when an audios begin to play, is paused etc.

HTML Plug-ins:

✔ Plug-ins are computer programs that extends the standard functionality of the browser.
✔ Plug-ins were designed to be used for many different purposes such as to:
- Run Java Applets.
- Run Microsoft ActiveX Controls.
- Display Flash Movies.
- Display Maps.
- Scans for computer viruses.
- Verify a bank id.

✔ Many browser does not support Java Applets and Plugins.
✔ ActiveX Controls are now not supported by any latest web browsers.
✔ The support for Flash Player has been turned off in modern web browsers.
✔ The <*object*> element is supported by every latest web browsers.
✔ The <*object*> element defines the embed object within an HTML document.

Lesson: HTML

- ✔ It was designed to embed plug-ins, like Java Applets, PDF readers, Flash Players in web pages.
- ✔ They can also be used to embed HTML into HTML or images if you like.
- ✔ The HTML <*embed*> element is supported in all major modern web browsers.
- ✔ The <*embed*> element is also defined as an embed object within an HTML document.
- ✔ Web browsers have supported <embed> element for a long time. It has not been a pert of the HTML specifications before HTML5.
- ✔ The <embed> does not have a closing tag.
- ✔ The <*embed*> element can also be used to include HTML into HTML.

HTML YouTube Videos:

- ✔ The easier way to play videos in HTML is to use YouTube.
- ✔ Converting videos to different video formats can be difficult and time-consuming.
- ✔ An easier solution is to YouTube play the videos in your web page.
- ✔ YouTube will display an ID, when you save or play a video.
- ✔ You can use this ID and refer to your video in the HTML code.
- ✔ To play a video on a web page you will need to:
  - ➢ Upload the video on YouTube.
  - ➢ Take a note of the video ID.
  - ➢ Define an <*iframe*> in your document.
  - ➢ Let the src attribute point to the video URL.
  - ➢ Use the width and height attribute to specify the dimensions of the YouTube Player.
  - ➢ Add any other parameters to the URL.
- ✔ You can start auto playing your videos when a user visits the page, by adding autoplay=1 to the YouTube URL.
- ✔ Automatically playing your videos can annoying your user and end up causing more harm than good.
- ✔ Chrome added stricter auto-play policies in 2018.
- ✔ chromium browsers does not allow auto-play videos in all cases.
- ✔ Muted auto-play is allowed.
- ✔ Add mute = 1 and autoplay = 1 to let your video start playing automatically but muted.
- ✔ Add loop=1 to let your videos forever.
- ✔ Value 0 is by default. The video will play only once.
- ✔ Value 1, the video will loop forever.
- ✔ Add controls=0 to display controls in the video player.
- ✔ Value 0. Player controls does not display.
- ✔ Value 1, Default. Player controls will display.

HTML Geo-location:

- ✔ The HTML Geo-location API is used to locate a user's position.
- ✔ Since this can compromise the privacy the position is not available unless the user approves it.
- ✔ Geo-location is more accurate for devices with GPS.
- ✔ As of chrome 50, the Geo-Location API will work only secure context.

Lesson: HTML

- ✔ If your website is hosted on non-secured context, the request to get the user location will no longer function.
- ✔ The getcurrentposition() method is used to return the users position.
- ✔ The second parameter of the getCurrentPosition() method is used to handle errors.
- ✔ It specifies a function to execute if it fails to get the user location.
- ✔ To display the result in a map you need to access a map service like Google Maps.
- ✔ Geo-location is very location for location specific information:
  - ➢ Up-to-date local information.
  - ➢ Show the point of interest near the user.
  - ➢ Turn by turn navigation, called GPS.
- ✔ The getCurrentPosition() method returns an object on success.
- ✔ The longitude, latitude and accuracy is always returned.
- ✔ The Geo-location also has other interesting methods:
  - ➢ watchPosition() - returns updated position of the user.
  - ➢ ClearWatch() - stops getting user updated position.

HTML Drag and Drop API:

- ✔ In HTML, any element can be dragged and dropped.
- ✔ Drag and Drop is very common feature.
- ✔ It is when you “grab” an object and drag it to another location.
- ✔ To make and element drag-gable set the drag-gable value to true.
- ✔ Then specify what should happen when the element is dragged.
- ✔ The ondragover event specifies where the dragged data can be dropped.
- ✔ By default, data/element cannot be dropped in other elements.
- ✔ To allow drop we must prevent the default.
- ✔ This can be achieved by calling the method event.preventDefault() method for the ondragover event.
- ✔ When the dragged data is dropped, a drop event occurs.

HTML Web Storage:

- ✔ HTML web storage, better than cookies.
- ✔ With the use of web storage, a web application can store data locally within the user’s browser.
- ✔ Before HTML5, application data can be stored in cookies, included in every server request.
- ✔ Web storage is much more secured and large amount of data can be stored locally, without effecting website performance.
- ✔ Unlike cookies, the storage is far larger(5MB) and information is not transferred to the server.
- ✔ Web storage is per origin means per domain and per protocol.
- ✔ All pages can store and access the same data from one origin.
- ✔ HTML web storage provides two objects for storing data on the client.
  - ➢ Window.localStorege – stores data with no expiration date.
  - ➢ Window.sessionStorege – stores data for one session, data is lost when the browser tab is closed.

Lesson: HTML

- ✔ Name/value pair is always stored as string, remember to convert them to another format when needed.

HTML Web workers:

- ✔ A web worker is a JavaScript running in the background, without affecting the performance of the web page.
- ✔ When executing the scripts in the web page the page becomes unresponsive until the script is finished.
- ✔ You can continue whatever you want to do, while the web worker runs in the background.
- ✔ Before creating a web worker, check whether user's browser supports it.
- ✔ Web workers are used for more CPU intensive task.
- ✔ When a web worker object is created it will continue to listen for messages even after the external scripts is finished executing, until it is terminated.
- ✔ To terminate a web worker and free computer resources use the terminate() method.
- ✔ If you set the worker variable to undefined after it has been terminated you can reuse the code.
- ✔ Since web workers are in external files, they do not have the access to the following JavaScript objects.
  - ➢ The window object.
  - ➢ The document object.
  - ➢ The parent object.

HTML SSE API:

- ✔ Server Sent Events (SSE) allow a web page to get updates from a server.
- ✔ A server sent event is when a web page automatically gets updates from a server.
- ✔ Before to this a web page has to ask from the server for the latest updates.
- ✔ With server-send events updates are automatically.
- ✔ The EventSource object is used to receive server-sent events notification.

Lesson: HTML

Note: All References Links Listed Below.

Reference: HTML Tags.
Reference: HTML Language Codes.
Reference: HTML Attributes.
Reference: HTML Color Names.
Reference: HTML File Path.
Reference: HTML Global Attributes.
Reference: HTML Events Attributes.
Reference: UTF-8 Mathematical Operations Symbols.
Reference: UTF-8 Greek and Coptic.
Reference: UTF-8 Currency Symbols.
Reference: UTF-8 Arrows.
Reference: Miscellaneous Symbols.
Reference: Emojis Unicode.
Reference: HTML Character Set.
Reference: PHP Tutorials.
Reference: W3C HTML Validator.
Reference: Word Wide Web Consortium.
Reference: Information about all types of File Formats.
Reference: HTML Audio and Video DOM.
Reference: Keyboard Shortcuts