**HTML AND CSS NOTES**

Studying with Mosh: <https://youtu.be/qz0aGYrrlhU>

Youtube SuperSimpleDev Link: <https://youtu.be/G3e-cpL7ofc>

HTML is the standard markup language for Web pages. With HTML you can create your own Website. HTML stands for Hyper Text Markup Language. HTML is the standard markup language for creating Web pages. HTML describes the structure of a Web page

HTML consists of a series of elements. HTML elements tell the browser how to display the content. HTML are bulding blocks. HTML elements label pieces of content such as "this is a heading", "this is a paragraph", "this is a link", etc. Javascript is used forn adding functionality or behaviour to web pages.

**ELEMENTS EXPLAINED AND ELABORATED**

The <!DOCTYPE html> declaration defines that this document is an HTML5 document

The <html> element is the root element of an HTML page

The <head> element contains meta information about the HTML page

The <title> element specifies a title for the HTML page (which is shown in the browser's title bar or in the page's tab)

The <body> element defines the document's body, and is a container for all the visible contents, such as headings, paragraphs, images, hyperlinks, tables, lists, etc.

The <h1> element defines a large heading

The <p> element defines a paragraph

**HTML Attributes**

All HTML elements can have attributes. Attributes provide additional information about elements. Attributes are always specified in the start tag. Attributes usually come in name/value pairs like: name="value"

**HTML FORMATTING ELEMENTS**

<b> - Bold text

<strong> - Important text

<i> - Italic text

<em> - Emphasized text

<mark> - Marked text

<small> - Smaller text

<del> - Deleted text

<ins> - Inserted text

<sub> - Subscript text

<sup> - Superscript text

**HTML QUOTATION AND CITATION ELEMENTS**

<abbr> - Defines an abbreviation or acronym

<address> - Defines contact information for the author/owner of a document

<bdo> - Defines the text direction

<blockquote> - Defines a section that is quoted from another source

<cite> - Defines the title of a work

<q> - Defines a short inline quotation

**HTML LINKS**

**By default, the linked page will be displayed in the current browser window. To change this, you must specify another target for the link.**

The target attribute specifies where to open the linked document.

The target attribute can have one of the following values:

\_self - Default. Opens the document in the same window/tab as it was clicked

\_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

Use the <a> element to define a link

Use the href attribute to define the link address

Use the target attribute to define where to open the linked document

Use the <img> element (inside <a>) to use an image as a link

Use the mailto: scheme inside the href attribute to create a link that opens the user's email program

**HTML TABLES**

<table> - Defines a table

<th> - Defines a header cell in a table

<tr> - Defines a row in a table

<td> - Defines a cell in a table

<caption> - Defines a table caption

<colgroup> - Specifies a group of one or more columns in a table for formatting

<col> - Specifies column properties for each column within a <colgroup> element

<thead> - Groups the header content in a table

<tbody> - Groups the body content in a table

<tfoot> - Groups the footer content in a table

**HTML LIST**

<ul> - Defines an unordered list

<ol> - Defines an ordered list

<li> - Defines a list item

<dl> - Defines a description list

<dt> - Defines a term in a description list

<dd> - Describes the term in a description list

**HTML LAYOUT ELEMENTS**

<header> - Defines a header for a document or a section

<nav> - Defines a set of navigation links

<section> - Defines a section in a document

<article> - Defines an independent, self-contained content

<aside> - Defines content aside from the content (like a sidebar)

<footer> - Defines a footer for a document or a section

<details> - Defines additional details that the user can open and close on demand

<summary> - Defines a heading for the <details> element

**HTML CHARACTER ENTITY**

Result Description Entity Name Entity Number Try it

non-breaking space &nbsp; &#160;

< less than &lt; &#60;

> greater than &gt; &#62;

& ampersand &amp; &#38;

" double quotation mark &quot; &#34;

' single quotation mark (apostrophe) &apos; &#39;

¢ cent &cent; &#162;

£ pound &pound; &#163;

¥ yen &yen; &#165;

€ euro &euro; &#8364;

© copyright &copy &#169;

® registered trademark &reg; &#174;

**HTML INPUT ELEMENTS**

<input type="text"> - Displays a single-line text input field

<input type="radio"> - Displays a radio button (for selecting one of many choices)

<input type="checkbox"> - Displays a checkbox (for selecting zero or more of many choices)

<input type="submit"> - Displays a submit button (for submitting the form)

<input type="button"> - Displays a clickable button

**HTML FORM ELEMENTS**

<form> - Defines an HTML form for user input

<input> - Defines an input control

<textarea> - Defines a multiline input control (text area)

<label> - Defines a label for an <input> element

<fieldset> - Groups related elements in a form

<legend> - Defines a caption for a <fieldset> element

<select> - Defines a drop-down list

<optgroup> - Defines a group of related options in a drop-down list

<option> - Defines an option in a drop-down list

<button> - Defines a clickable button

<datalist> - Specifies a list of pre-defined options for input controls

<output> - Defines the result of a calculation

**CSS**

CSS stands for Cascading Stylesheet hence it is used for styling webpages and making it beautiful. The css has three methods for addition namely, Inline CSS: Directly in the html element(NO!), Internal CSS: It is the using of <style> tags within a single document and External CSS: It is use for linking an external.css file. There are class and ids which are kinda the same. IDs are element that are unique. Classes can be changed.

Classes in css uses (dot)"." whiles Ids in css uses (hash)"#".

**CSS POSITIONING**

CSS positioning value are Static(Default position hence renders the element in order of the document flow), Relative(It can be pushed in where you want), Absolute(Allow whatever position in a realtive element), Fixed(Same position), Initial and Inherit.

**FONTS**

The font-style property is mostly used to specify italic, normal or leaning text. eg: font-style: normal, font-style: italic, font-style: oblique.

The font-weight property specifies the weight of a font. eg: font-weight: normal, font-weight: bold.

The font-variant property specifies whether or not a text should be displayed in a small-caps font. In a small-caps font, all lowercase letters are converted to uppercase letters. However, the converted uppercase letters appear in a smaller font size than the original uppercase letters in the text. eg: font-variant: normal, font-variant: small-caps.

**THE Z-INDEX PROPERTY**

The z-index property specifies the stack order of an element. The z-index are positioned, they can overlap other elements. The z-index property specifies the stack order of an element (which element should be placed in front of, or behind, the others). An element can have a positive or negative stack order.

**What are Pseudo-classes?**

A pseudo-class is used to define a special state of an element.

For example, it can be used to:

* Style an element when a user mouses over it
* Style visited and unvisited links differently
* Style an element when it gets focus

|  |  |  |
| --- | --- | --- |
| * **Selector** | **Example** | **Example description** |
| [:active](https://www.w3schools.com/cssref/sel_active.asp) | a:active | Selects the active link |
| [:checked](https://www.w3schools.com/cssref/sel_checked.asp) | input:checked | Selects every checked <input> element |
| [:disabled](https://www.w3schools.com/cssref/sel_disabled.asp) | input:disabled | Selects every disabled <input> element |
| [:empty](https://www.w3schools.com/cssref/sel_empty.asp) | p:empty | Selects every <p> element that has no children |
| [:enabled](https://www.w3schools.com/cssref/sel_enabled.asp) | input:enabled | Selects every enabled <input> element |
| [:first-child](https://www.w3schools.com/cssref/sel_firstchild.asp) | p:first-child | Selects every <p> elements that is the first child of its parent |
| [:first-of-type](https://www.w3schools.com/cssref/sel_first-of-type.asp) | p:first-of-type | Selects every <p> element that is the first <p> element of its parent |
| [:focus](https://www.w3schools.com/cssref/sel_focus.asp) | input:focus | Selects the <input> element that has focus |
| [:hover](https://www.w3schools.com/cssref/sel_hover.asp) | a:hover | Selects links on mouse over |
| [:in-range](https://www.w3schools.com/cssref/sel_in-range.asp) | input:in-range | Selects <input> elements with a value within a specified range |
| [:invalid](https://www.w3schools.com/cssref/sel_invalid.asp) | input:invalid | Selects all <input> elements with an invalid value |
| [:lang(*language*)](https://www.w3schools.com/cssref/sel_lang.asp) | p:lang(it) | Selects every <p> element with a lang attribute value starting with "it" |
| [:last-child](https://www.w3schools.com/cssref/sel_last-child.asp) | p:last-child | Selects every <p> elements that is the last child of its parent |
| [:last-of-type](https://www.w3schools.com/cssref/sel_last-of-type.asp) | p:last-of-type | Selects every <p> element that is the last <p> element of its parent |
| [:link](https://www.w3schools.com/cssref/sel_link.asp) | a:link | Selects all unvisited links |
| [:not(selector)](https://www.w3schools.com/cssref/sel_not.asp) | :not(p) | Selects every element that is not a <p> element |
| [:nth-child(n)](https://www.w3schools.com/cssref/sel_nth-child.asp) | p:nth-child(2) | Selects every <p> element that is the second child of its parent |
| [:nth-last-child(n)](https://www.w3schools.com/cssref/sel_nth-last-child.asp) | p:nth-last-child(2) | Selects every <p> element that is the second child of its parent, counting from the last child |
| [:nth-last-of-type(n)](https://www.w3schools.com/cssref/sel_nth-last-of-type.asp) | p:nth-last-of-type(2) | Selects every <p> element that is the second <p> element of its parent, counting from the last child |
| [:nth-of-type(n)](https://www.w3schools.com/cssref/sel_nth-of-type.asp) | p:nth-of-type(2) | Selects every <p> element that is the second <p> element of its parent |
| [:only-of-type](https://www.w3schools.com/cssref/sel_only-of-type.asp) | p:only-of-type | Selects every <p> element that is the only <p> element of its parent |
| [:only-child](https://www.w3schools.com/cssref/sel_only-child.asp) | p:only-child | Selects every <p> element that is the only child of its parent |
| [:optional](https://www.w3schools.com/cssref/sel_optional.asp) | input:optional | Selects <input> elements with no "required" attribute |
| [:out-of-range](https://www.w3schools.com/cssref/sel_out-of-range.asp) | input:out-of-range | Selects <input> elements with a value outside a specified range |
| [:read-only](https://www.w3schools.com/cssref/sel_read-only.asp) | input:read-only | Selects <input> elements with a "readonly" attribute specified |
| [:read-write](https://www.w3schools.com/cssref/sel_read-write.asp) | input:read-write | Selects <input> elements with no "readonly" attribute |
| [:required](https://www.w3schools.com/cssref/sel_required.asp) | input:required | Selects <input> elements with a "required" attribute specified |
| [:root](https://www.w3schools.com/cssref/sel_root.asp) | root | Selects the document's root element |
| [:target](https://www.w3schools.com/cssref/sel_target.asp) | #news:target | Selects the current active #news element (clicked on a URL containing that anchor name) |
| [:valid](https://www.w3schools.com/cssref/sel_valid.asp) | input:valid | Selects all <input> elements with a valid value |
| [:visited](https://www.w3schools.com/cssref/sel_visited.asp) | a:visited | Selects all visited links |

All CSS Pseudo Elements

|  |  |  |
| --- | --- | --- |
| **Selector** | **Example** | **Example description** |
| [::after](https://www.w3schools.com/cssref/sel_after.asp) | p::after | Insert content after every <p> element |
| [::before](https://www.w3schools.com/cssref/sel_before.asp) | p::before | Insert content before every <p> element |
| [::first-letter](https://www.w3schools.com/cssref/sel_firstletter.asp) | p::first-letter | Selects the first letter of every <p> element |
| [::first-line](https://www.w3schools.com/cssref/sel_firstline.asp) | p::first-line | Selects the first line of every <p> element |
| [::selection](https://www.w3schools.com/cssref/sel_selection.asp) | p::selection | Selects the portion of an element that is selected by a user |

**JAVASCRIPT**

|  |  |
| --- | --- |
| **Keyword** | **Description** |
| var | Declares a variable |
| let | Declares a block variable |
| const | Declares a block constant |
| if | Marks a block of statements to be executed on a condition |
| switch | Marks a block of statements to be executed in different cases |
| for | Marks a block of statements to be executed in a loop |
| function | Declares a function |
| return | Exits a function |
| try | Implements error handling to a block of statements |

The basic datatype for JavaScript are, Boolean, Number, String , BigInt, Object, Undefined, Null and Symbol.