**HTML Basics**

* **HTML List**
  + An Unordered list
  + An Ordered list
* **Unordered HTML List**
  + - An unordered list starts with <ul> tag. Each list items starts with the <li> tag.
    - The list items will be marked with bullets (by default).

|  |  |
| --- | --- |
| <ul>   <li>Coffee</li>   <li>Tea</li>   <li>Milk</li> </ul> | * Coffee * Tea * Milk |

* **Unordered HTML List – Choose list item marker**
  + - The CSS list-style-type property is used to define the style of the list item marker:

|  |  |
| --- | --- |
| **Value** | **Description** |
| disc | Sets the list item marker to be bullet(default) |
| circle | Sets the list item marker to be circle |
| Square | Sets the list item marker to be square |
| None | The list items will not marked |

**Example:**

|  |  |
| --- | --- |
| <ul style="list-style-type:disc;">   <li>Coffee</li>   <li>Tea</li>   <li>Milk</li> </ul> | * Coffee * Tea * Milk |

* **Ordered HTML List**
  + - An order list starts with the <ol> tag. Each list item starts with the <li> tag.
    - The list item will be marked with number by default:

|  |  |
| --- | --- |
| <ol>   <li>Coffee</li>   <li>Tea</li>   <li>Milk</li> </ol> | 1. Coffee 2. Tea 3. Milk |

* **Ordered HTML List – The type attribute**
  + - The **type** attribute of the **<ol>** tag, defines the type of the list item marker:

|  |  |
| --- | --- |
| **Type** | **Description** |
| type=”1” | The list items will be numbered with numbers (default) |
| type=”A” | The list items will be numbered with uppercase letters |
| type=”a” | The list items will be numbered with lowercase letters |
| type=”I” | The list items will be numbered with uppercase roman numbers |
| type=”i” | The list item will be numbered with lowercase roman numbers |

**Example:**

|  |  |
| --- | --- |
| <ul type=”1”>   <li>Coffee</li>   <li>Tea</li>   <li>Milk</li> </ul> | 1. Coffee 2. Tea 3. Milk |
| <ul type=”A”>   <li>Coffee</li>   <li>Tea</li>   <li>Milk</li> </ul> | 1. Coffee 2. Tea 3. Milk |

* **HTML Description List** 
  + - HTML also supports description lists.
    - A description list is a list of terms, with a description of each item.
    - The <dl> tag defines the description list, the <dt> tag defines the term (name), and the <dd> tag describes each term.

|  |  |
| --- | --- |
| <dl>   <dt>Coffee</dt>   <dd>- black hot drink</dd>   <dt>Milk</dt>   <dd>- white cold drink</dd> </dl> | Coffee  - black hot drink  Milk  - white cold drink |

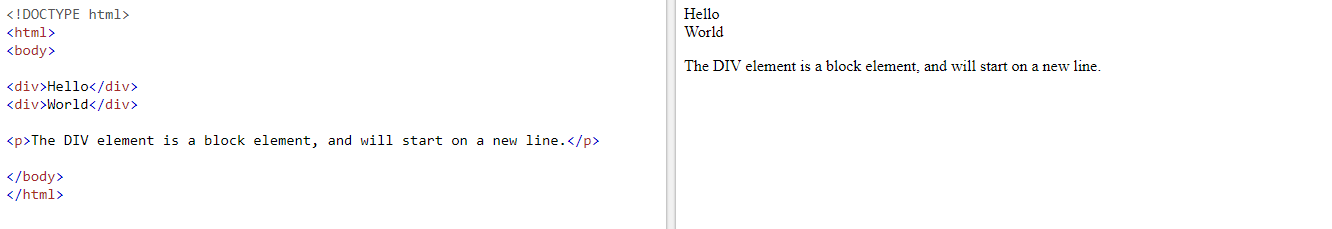
* **Control List Counting**
  + - By default, an ordered list will start counting from 1. If we want to start counting from a specified number, we can use the start attribute.

|  |  |
| --- | --- |
| <ol start=”50”>   <li>Coffee</li>   <li>Tea</li>   <li>Milk</li> </ol> | 1. Coffee 2. Tea 3. Milk |

* **Horizontal List with CSS**
  + - 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.

|  |
| --- |
| <!DOCTYPE html> <html> <head> <style> ul {   list-style-type: none;   margin: 0;   padding: 0;   overflow: hidden;   background-color: #333333; }  li {   float: left; }  li a {   display: block;   color: white;   text-align: center;   padding: 16px;   text-decoration: none; }  li a:hover {   background-color: #111111; } </style> </head> <body>  <ul>   <li><a href="#home">Home</a></li>   <li><a href="#news">News</a></li>   <li><a href="#contact">Contact</a></li>   <li><a href="#about">About</a></li> </ul>  </body> </html> |

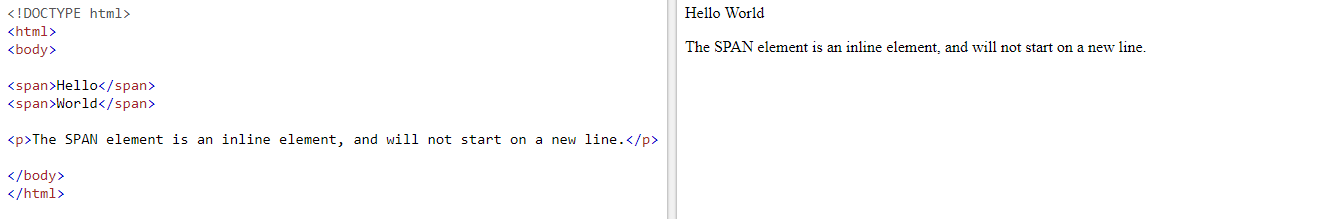
* **HTML Block and Inline Elements**
  + - Every HTML element has a default display value depending on what type of element it is.
    - The default display value for most elements is block or inline.
  + **Block Level Element**
    - A block-level element always starts on a new line and takes up the full width available (stretches out to the left and right as far as it can).
    - The <div> element is a block-level element

****

* + **Block level element in HTML:**

|  |  |
| --- | --- |
| **TAG** | **Description** |
| <address> |  |
| <article> |  |
| <aside> |  |
| <blockquote> |  |
| <canvas> |  |
| <dd> |  |
| <div> |  |
| <dl> |  |
| <dt> |  |
| <fieldset> |  |
| <figcaption> |  |
| <figure> |  |
| <footer> |  |
| <form> |  |
| <h1> to <h6> |  |
| <header> |  |
| <hr> |  |
| <li> |  |
| <main> |  |
| <nav> |  |
| <noscript> |  |
| <ol> |  |
| <p> |  |
| <pre> |  |
| <section> |  |
| <table> |  |
| <tfoot> |  |
| <ul> |  |
| <video> |  |

* + **Inline Elements:**
    - An inline element does not start on a new line and only takes up as much width as necessary.
    - This is an inline element <span>

****

* + **Inline elements in HTML:**

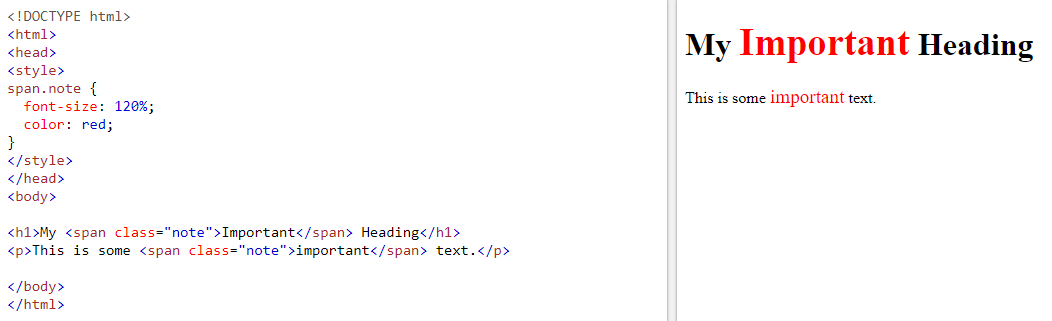
|  |  |
| --- | --- |
| **TAG** | **Description** |
| <a> |  |
| <abbr> |  |
| <acronym> |  |
| <b> |  |
| <bdo> |  |
| <big> |  |
| <br> |  |
| <button> |  |
| <cite> |  |
| <code> |  |
| <dfn> |  |
| <em> |  |
| <i> |  |
| <img> |  |
| <input> |  |
| <kbd> |  |
| <label> |  |
| <map> |  |
| <object> |  |
| <output> |  |
| <q> |  |
| <samp> |  |
| <script> |  |
| <select> |  |
| <small> |  |
| <span> |  |
| <strong> |  |
| <sub> |  |
| <sup> |  |
| <textarea> |  |
| <time> |  |
| <tt> |  |
| <var> |  |

* **HTML Class Attribute**
  + - The HTML **class** attribute is used to define equal styles for elements with same class name.
    - So, all HTML elements with the same **class** attribute will have the same format and style.
    - Here we have three <div> elements that point to the same class name:

|  |
| --- |
| <!DOCTYPE html> <html> <head> <style> .cities {   background-color: black;   color: white;   margin: 20px;   padding: 20px; }  </style> </head> <body>  <div class="cities">   <h2>London</h2>   <p>London is the capital of England.</p> </div>  <div class="cities">   <h2>Paris</h2>   <p>Paris is the capital of France.</p> </div>  <div class="cities">   <h2>Tokyo</h2>   <p>Tokyo is the capital of Japan.</p> </div>  </body> </html> |

****

* **Using The Class Attribute On Inline Elements**
  + - The HTML **class** attribute can also be used on inline elements:

****

* + - The **class** attribute can be used on **any** HTML element.
    - The **class** name is case sensitive!
* **Multiple Classes**
  + - HTML elements can have more than one class name “city”, also style elements with the class name “main”.

|  |
| --- |
| <h2 class="city main">London</h2> <h2 class="city">Paris</h2> <h2 class="city">Tokyo</h2> |

* + - In the example above, the first **<h2>** element belongs to both “city” and “main” class.
* **HTML The id Attribute**
  + - The id attribute specifies a unique id for an HTML element (The id values must be unique with in the HTML document).
    - The id value can be used by CSS and JavaScript to perform certain tasks for a unique element with the specified id value.
    - In CSS, to select an element with specific id, write a (#) character, followed by the id of the element.

****

* + - The id attribute can be used on any HTML element.
    - The id value is case-sensitive
    - The id value must contain at least one character, and must not contain whitespace (spaces, tabs, etc.).
* **Difference between class and id**
  + - An HTML element can only have one unique id that belongs to that single element, while the class name can be used by multiple elements.
* **HTML Iframes**
  + - An iframe is used to display a webpage with in a page.
    - An HTML iframe is defined with the <iframe> tag

|  |
| --- |
| <iframe src="URL"></iframe> |

* + - The src attribute specifies the URL (web address) of the inline frame page.
* **Iframe – set Height and Width**
  + - Use the **height** and **width** attributes to specify the size of the iframe.
    - The attribute values are specified in pixels by default, but they can also be in percent (like “80%”).

|  |
| --- |
| <iframe src="demo\_iframe.htm" height="200" width="300"></iframe> |

* + - We can use CSS to set **height** and **width** of the iframe.

|  |
| --- |
| <iframe src="demo\_iframe.htm" style="height:200px;width:300px;"></iframe> |

* **Iframe – Remove the Border**
  + - By default, an iframe has a border around it.
    - To remove the border, and the **style** attribute and use the CSS **border** property:

|  |
| --- |
| <iframe src="demo\_iframe.htm" style="border:none;"></iframe> |

* + - With CSS, we also change the size, style and color of the iframe’s border:

|  |
| --- |
| <iframe src="demo\_iframe.htm" style="border:2px solid red;"></iframe> |

* **Iframe – Target for a Link**
  + - An iframe can be used as the target frame for a link.
    - The **target** attribute of the link must refer to the **name** attribute of the iframe:

|  |
| --- |
| <iframe src="demo\_iframe.htm" name="iframe\_a"></iframe>  <p><a href="https://www.w3schools.com" target="iframe\_a">W3Schools.com</a></p> |

* **HTML File Paths**

|  |  |
| --- | --- |
| **Path** | **Description** |
| <img src=”picture.jpg”/> | picture.jpg is located in the same folder as the current page |
| <img src=”images/picture.jpg”/> | picture.jpg is located in the images folder in the current folder |
| <img src=”/images/picture.jpg”/> | picture.jpg is located in the images folder at the root of the current web |
| <img src=”../picture.jpg”/> | picture.jpg is located in the folder one level up from the current folder |

* + - A file path describes the location of a file in a web site’s folder structure.
    - File paths are used when linking to external files like:
      * Web pages
      * Images
      * Style Sheets
      * Java Scripts
* **HTML Absolute File Paths**
  + - An absolute file path is the full URL to an internet file:

|  |
| --- |
| <img src="https://www.w3schools.com/images/picture.jpg" alt="Mountain"> |

* **HTML Relative File Paths**
  + - A relative file path points to a file relative to the current page.

|  |
| --- |
| <img src="/images/picture.jpg" alt="Mountain"> |

* + - In this example, the file path points to a file in the images folder located in the current folder.
* **HTML <head> Element**
  + - The **<head>** element is a container for metadata (data about data) and is placed between the **<html>** tag and the **<body>** tag.
    - HTML metadata is data about the HTML document. Metadata is not displayed.
    - Metadata typically define the document title, character set, styles, links, scripts, and other meta information.
    - The following tag describe metadata: <title>, <style>, <meta>, <link>. <script> and <base>
* **HTML <title> Element**
  + - The **<title>** element defines the title of the document, and is required in all HTML/XHTML documents.
    - The **<title>** element:
      * **Defines a title in the browser tab**
      * **Provides a title for the page when it is added to favorites.**
      * **Displays a title for the page in search engine results**

|  |
| --- |
| <head>   <title>Page Title</title> </head> |

* **HTML <style> Element**
  + - The <style> element is used to define style information for a single HTML page.

|  |
| --- |
| <style>   body {background-color: powderblue;}   h1 {color: red;}   p {color: blue;} </style> |

* **HTML <link> Element**
  + - The <link> element is used to link to external style sheets:

|  |
| --- |
| <link rel="stylesheet" href="mystyle.css"> |

* **HTML <meta> Element**
  + - The <meta> element is used to specify which character set is used, page description, keywords, author, and other metadata.
    - Metadata is used by browsers (how to display content), by search engines (keywords), and other web services.
    - Define the character set used:

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

* + - Define a description of our webpage:

|  |
| --- |
| <meta name="description" content="Free Web tutorials"> |

* + - Define keywords for search engines:

|  |
| --- |
| <meta name="keywords" content="HTML, CSS, XML, JavaScript"> |

* + - Define the author of a page:

|  |
| --- |
| <meta name="author" content="John Doe"> |

* + - Refresh document every 30 seconds:

|  |
| --- |
| <meta http-equiv="refresh" content="30"> |

**Example:**

|  |
| --- |
| <head>  <meta charset="UTF-8"> <meta name="description" content="Free Web tutorials"> <meta name="keywords" content="HTML,CSS,XML,JavaScript"> <meta name="author" content="John Doe">  </head> |

* **Setting the Viewport**
  + - HTML5 introduced a method to let web designers take control over the view port, through the <meta> tag.
    - The viewport is the user’s visible area of a web page. It varies with the device, and will be smaller on a mobile phone than on a computer screen.
    - We should include following <meta> viewport element in all our web pages:

|  |
| --- |
| <meta name="viewport" content="width=device-width, initial-scale=1.0"> |

* + - A <meta> viewport element gives the browser instructions 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 browser.
* **HTML <script> Element**
  + - The **<script>** element is used to define client-side Java Scripts.
    - This Java Script writes “Hello JavaScript!” into an HTML element with id=”demo”:

|  |
| --- |
| <script> function myFunction {   document.getElementById("demo").innerHTML = "Hello JavaScript!"; } </script> |

* **HTML <base> Element**
  + - The <base> element specifies the base URL and base target for all relative URLs in a page:

|  |
| --- |
| <base href="https://www.w3schools.com/images/" target="\_blank"> |