HTML cheat sheet

**STRUCTURE**

<!DOCTYPE html> Tells the system that this is an html document. Put on top of page.

<html></html> This tags tells the user all the elements within this tag is written in html code.

<body></body> Tag is for structuring the main details within the html.

<head></head> Tag is before the <body> that tells the metadata for webpage located at the top of the page.

<title></title> serves as the title of the page within head that is located on the tab or title bar.

<head>

<title>bruh</title>

</head>

<section> is used to define a section within the body. Can be broken down as divs.

<div></div> Helps divide the page, can be used in groups. Divs can also have ids

<div id= ”bruhmoment”></div>

<hr> basically helps break between paragraph using a horizontal line

Comments are written as the following

<!-- Insert Comment -->

**Structure-Body**

Tags within Body

<main> The main content

<header> Basically the top part of the main content

<nav> The Navigation portion at the top

<section> Section of the page

<article> Can hold content such as articles, etc.

<aside> Additional info for article.

<figure> Put photo here

<figcaption> Encapsulated within figure to describe photo.

<audio> Put audio. Can use autoplay and controls

<source> Encapsulates audio. Define src and type.

<video> Put video. Define src and can use autoplay, loop, and controls.

<embed> can put media content either locally or online

<footer> The footer

**BASIC TAGS**

<h1></h1> heading 1 up to heading 6

h1: 2em, h2: 1.5em, h3: 1.17em, h4: 1.12em, h5: 0.83em, h6: 0.75em

<p></p> paragraph

<span></span> Can be used withing text tag to add special effects.

<em></em> emphasize text

<strong></strong> strong text

<a></a> “a” tag defines hyper link <a href= “https://www.alzwang.com” target= “\_blank”>  
 target refers to what happens in page. \_blank = new page, \_self = self.  
 download refers to downloading a document listed

a tags can also be used to go to different parts of a page

For example <p id= “top”>This is the top</p>

<a href= “#top”>Click to go to top</a>

This command allows us to click on the link to go to the p with the id = “top”.

They can also be used to go to different directories of your website

<a href= “bruh.html”>Goes to bruh.html</a>

<br> Breaks the line

<ul></ul> Unordered list, used to help organize <li> without any order.

<ul>

<li>bruh</li>

<li>moment</li>

</ul>

<ol></ol> Ordered list, basically ul but with order.

<img /> Displays image from code. <img src= “../../media/images/bruhmoment.jpg”/>

Alt can be used in case image doesn’t load

<img src= “../../media/images/bruhmoment.jpg” alt= “says bruh moment”/>

Can also adjust size

<img src= “../../media/images/bruhmoment.jpg” alt= “says bruh moment” height= “500” width= “500”/>

<video /> Similar to how images work. Can add controls to simulate pause, play, rewind, etc.

<video src- “../../media/videos/mp4.mp4” controls>

**TABLE**

<table></table> Makes a table. Can use props such as border= “1” to get a border.

<tr> is a row

<td> is data. Can use props such as colspan = “int” to define width.

<th> is a heading. Can use scope=”” to define type of data

<tbody> basically the body of table if it gets too big.

<thead> basically the heading of the table.

<tfoot> Footer of table

<table>

<thead>

<th scope=“col”>Yeet stats</th>

</thead>

<tbody>

<tr>

<td>bruh</td>

<td>moment</td>

</tr>

</tbody>

<tfoot>

<td>Results: Bruh</td>

</tfoot>

</table>

**FORM**

<form> is used to collect data and send it to a targeted location.

action= “” is where the info is being sent. method = “” is what type of information is sent.

Can contain elements such as <input> which is what we can use so users can input information.

<input> has type= “text”, password, number, range, checkbox, radio, submit, etc and can be named by its group name= “topTextBox” can also add value by default

value= “insert value”

<label> basically give a title for the form. Must be connected with the id attribute of input using for and id.

<form action= “../../data/data.html” method= “POST”>

<label for= “username”>Username: </label>

<input type= “text” name= “username” id= “username”>

<label for= “password”>Password: </label>

<input type= “password” name= “password” id= “password”>

<label for= “num”>Choose a number </label>

<input type= “number” name= “num” id= “num” value= “0” step= “1”(in/decrements by 1)>

<label for= “value”>From 1 to 100, give our system a rating</label>

<input type= “range” min= “0” max= “100” step= “1”>

<label for= “survey”>Check this box if you like this survey</label>

<input type= “checkbox” name= “survey” id= “survey”>

<h3>Yes or No?</h3>

<input type= “radio” name= “question” id= “yes” label= “yes”>

<label for= “yes”>Yes</label>

<input type= “radio” name= “question” id= “no” label= “no”>

<label for= “no”>No</label>

<input type= “submit” value= “send”>

</form>

<select> is another type of form used. Instead of inputting words or buttons, it is a dropdown menu

<option> are the stuff within select that provides options

<form action= “../../data/data.html” method= “POST”>

<label for= “form”>Select 1 - 3</label>

<select name= “form” id= “form”>

<option value= “1”>1</option>

<option value= “2”>2</option>

<option value= “3”>3</option>

</select>

</form>

<datalist> Similar to option but needs to be referred in an <input> with list= “name of datalist”

<form action= “../../data/data.html” method= “POST”>

<label for= “input”>Select 1 - 3</label>

<input type= “text” id= “input” name= “input” label= “input” list= “data”>

<datalist id= “data”>

<option value= “1”/>

<option value= “2”/>

<option value= “3”/>

</datalist>

</form>

<textarea> basically text but has an area

<textarea id= “thing” name= “thing” rows= “5” cols= “30”>Default text</textarea>

**Form Validation**

Adding required would be important when wanting info boxes not to be blank.

<input type= “text” id= “firstName” name= “information” required>

You can also set min and max of values

<input type= “num” id= “firstNum” name= “information” min = 1 max = 4 required>

You can also set min and max lengths for passwords and text

<input type= “password” id= “userPassword” name= “information” minlength= 4>

You can also create a pattern users will follow. [a-zA-Z0-9] = accepted values {4, 8} range of values

<input type= “text” id= “pinNumber” name= “information” pattern= “[a-zA-Z0-9] {4,8}” required>

**STYLING**

<style> Basically the styling tag used within an html document. AKA inline styling

Can be used to adjust the attributes of text tags, to defining the structure and background of structural tags.

<style>

p {

color: red;

font-size: 20;

}

</style>

Linking css file

<link href= “LocationofFile” type= “text/css” rel= “stylesheet”>

<link href= “../style/main.css” type= “text/css” rel= “stylesheet”>

**<script>**

This is basically where you use JavaScript code in html

defer: you want to refer it at the end of the script to indicate to run this script last.

async: similar placement with defer. Basically the script will run async from the actual webpage and will execute after it is downloaded.