### **HTML(Hyper Text Markup Language)**

It is the most basic building block of web, it defines the meaning and structure of web. It uses special elements .

1. <head>g area contains meta data which can not be seen outside.
2. <body> area contains data that can be seen on the browser.
3. <h1> is the biggest heading, we should have only one h1 per page.
4. h2,h3,h4… will be the consecutive subtopics.
5. <p> tag/paragraph tag holds information about sub topics
6. <style> tag hold the CSS content
7. W3C markup validation service: [The W3C Markup Validation Service](https://validator.w3.org/) go here to check and validate your file.
8. meta data: <meta >
9. href: In HTML, href is an attribute that is used to specify the URL (Uniform Resource Locator) of a link. The href attribute is used in conjunction with the <a> tag to create hyperlinks that allow users to navigate between different web pages or resources on the internet.
10. The href attribute can also be used with other HTML elements, such as the <link> tag, to specify the location of external resources like stylesheets or scripts.
11. <hr> – horizontal rules, add horizontal lines in webpage content.
12. White-space-collapsing: We can add spaces in between contents but html file will not allow those spaces and will not be seen in the web page.
13. <br>- line break- use inside p tag to make points
14. In-line elements and block level elements.
15. <em> – to provide emphasis on particular sentence or word.
16. <strong> - for making bold word
17. Special entity- &nbps; -it create space
18. <abbr title=”give description”> - abbreviation
19. <address> - put location

List – Ordered, Unordered and description list.

* <li> - list of items -it will create list of items and not inside a paragraph tag so it will not add extra spaces.
* <ol> - ordered list \_ it gives numbering of each item on the list
* <ul> - it gives bullet points of each list.
* <dl> - description list

<dd> -description details

<dt> - description term

* <a> - anchor tag - href and give url to re-direct
* <section> -
* mailto: <email address> -for email
* [tel: <telephone](tel:%3ctelephone) number>
* <img src= »image » location” alt = «description «  title = « « >

**Cumulative layout shift:** Cumulative Layout Shift (CLS) is a performance metric that measures the amount of unexpected layout shift of visual elements on a web page. CLS can be caused by a variety of factors, such as images or videos loading slowly, or dynamic content that is added to the page after it has already loaded.

* To avoid this we always should use width and height attributes.
* Loading attribute is always set to eager by default, now if any image which is below the fold should set to “laz” that meaning the image will only load when cursor will go down. Lazy loading is very goof performance technique when there are lots of images to load in the browser.
* <fugure> and <figcaption> - for space and indentation and to put the image into figure and add a caption
* <code>&lt;h1&gt;Hello World&lt;/h1&gt;</code>
* To write html code directly
* <header> - inside header navigator should be there.
* <footer>
* <main> - only one main per page
* If there is more than one navigator then you can label by below:

<nav *aria-label*="primary navigator">

* Inside main there will be hierarchy. First article then section
* Aside->details->summary-> p tag

Text

Description automatically generated

Graphical user interface, text

Description automatically generated

* <mark> - to highlight the text
* <div> - divider. It has no meaning by itself
* <span> - It has no meaning by itself
* <br> -<table>-<tr(table row)>-<td(table data)>-
* For table header, we have to use <hr>

<tr>

          <th>&nbsp;</th>

                 <th *scope*="col">Time</th>

                  <th *scope*="col">Activity</th>

</tr>

* rowspan=”2” and colspan=”2” to combined two row or column together.

To add form and inputs:

1. hr – article -h2 -p-form tag-label-input(type – name – id – placeholder – autocomplete(it will suggest the same thing whatever you put in the first time) – required(for the form to have input from user before it submits))- autofocus
2. id should be exactly same as name attribute
3. only one input can have autofocus

<form *action*="https://httpbin.org.get" *method*="get"></form>

        <label *for*="firstName">First Name</label>

        <input *type*="text" *name*="firstName" *id*="firstName" *placeholder*="Jane" *autocomplete*="on" required autofocus>

1. Autocomplete is not required for password.

For choosing option:

p>

        <label *for*="coffee">Favourite Coffee</label>

        <select *name*="coffee" *id*="coffee">

            <option *value*="regular">Regular</option>

            <option *value*="medium">Medium</option>

            <option *value*="grande">Grande</option>

        </select>

        <select *name*="coffee" *id*="coffee">

        <option *value*="coffee latte" *selected*>Latte</option>

        <option *value*=" coffee espresso">Espresso</option>

        <option *value*="cappuccino">Cappuccino</option>

        </select>

    </p>

**Note:** If we want to select something by default we have to write selected

1. Optgroup to organize the choices.
2. fieldset and lebel – to put item inside a box and for a heading
3. <id> - should be exactly in match with label for attribute
4. <name> - should be same for the same category