Placement Prep Resource- HTML Concepts Assignment

1. What is HTML?

Ans: The **HyperText Markup Language**, or **HTML** is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript.

Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

HTML elements are the building blocks of HTML pages. With HTML constructs, images and other objects such as interactive forms may be embedded into the rendered page. HTML provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. HTML elements are delineated by *tags*, written using angle brackets. Tags such as `<img />` and `<input />` directly introduce content into the page. Other tags such as `<p>` surround and provide information about document text and may include other tags as sub-elements. Browsers do not display the HTML tags, but use them to interpret the content of the page.

2. What do you mean by a markup language?

   Ans: A markup language is a computer language that uses tags to define elements within a document. It is human-readable, meaning markup files contain standard words, rather than typical programming syntax. While several markup languages exist, the two most popular are HTML and XML.

3. Can you share examples of other markup languages and how they differ from HTML?

Ans: The **main difference** between markup language and programming language is that **a markup language defines a set of rules for encoding documents in a format that is both human-readable and machine-readable while a programming language provides a set of commands and syntax that can be used to write computer programs which are understood by the computer.**
A markup language is a type of language used to annotate text and embed tags in accurately styled electronic documents, irrespective of computer platforms, operating systems, applications or programs. However, a programming language is a language that provides a set of rules, syntax, and commands to write computer programs that implements algorithms.

4. What version of HTML do you use in your projects? How is HTML 5 different from HTML 4?

Ans **HTML** stands for *Hyper Text Markup Language*. It is used to design web pages using a markup language. HTML is the combination of Hypertext and Markup language. Hypertext defines the link between the web pages. A markup language is used to define the text document within tag which defines the structure of web pages. This language is used to annotate (at the note for computer) text so that a machine can understand it and manipulate text accordingly. Most of the markup (e.g. HTML) languages are human readable. The language uses tags to define what manipulation has to be done on the text. It is used for structuring and presenting the content on the web pages. HTML5 is the fifth version of HTML. Many elements are removed or modified from HTML5.

5. What are attributes in HTML?

Ans: **HTML attributes** are special words used inside the opening tag to control the element's behaviour. HTML attributes are a modifier of an *HTML element type*. An attribute either modifies the default functionality of an element type or provides functionality to certain element types unable to function correctly without them. In HTML syntax, an attribute is added to an *HTML start tag*.

Several basic attributes types have been recognized, including: (1) *required attributes*, needed by a particular element type for that element type to function correctly; (2) *optional attributes*, used to modify the default functionality of an element type; (3) *standard attributes*, supported by many element types; and (4) *event attributes*, used to cause element types to specify scripts to be run under specific circumstances.

6. What are data- attributes good for?

Ans: HTML5 is designed with extensibility in mind for data that should be associated with a particular element but need not have any defined meaning. data-* attributes allow us to store extra information on standard, semantic HTML elements without other hacks such as non-standard attributes, or extra properties on DOM.

7. Describe the difference between &<script>, <script async> and <script defer>.

Ans: <script> = used to define a client-side script

<script async≥= If async is present: The script is executed asynchronously with the rest of the page (the script will be executed while the page continues the parsing) If async is not present and defer is present: The script is executed when the page has finished parsing.

<script defer≥= The defer attribute tells the browser to only execute the script file once the HTML document has been fully parsed

8. Why is it generally a good idea to position CSS <link>s between <head></head&> and JS <script>s just before </body>? Do you know any exceptions?

Ans: The **main reason** as to why JS files are linked at the bottom of the body is **because** whenever a browser encounters any JS, it parses it and executes that on the spot. Hence if it was to be added at the top, it would make the page rendering slow and thus it would take more time for page load. Moreover since the **DOM won't** be rendered fully, JS won't be able to manipulate the elements.

However if you use **Jquery,** that won't be an issue since it would execute only after the document is ready. But since in any case, the **browser would parse** it, it would slow the page load.

On the contrary, CSS files are linked in the head because they get applied regardless of DOM already rendered or not. Hence the webpage looks elegant as soon as the page loads. However just like JS you can link the CSS at the end which would mean that the webpage first loads with just plain HTML and then the CSS is applied to it. This shift is clearly visible to the user and moreover an important thing to remember is that the page would load with bare minimum HTML and if the user has **slow Internet connection,** the CSS load would take considerable amount of time, which means that the webpage shows just the HTML meanwhile. This might make the user close the website without waiting for it to load fully.

To avoid such things, a CSS file is linked at the head while a JS file is linked at the bottom.