**MODULE-1 HTML**

Q-1: Are the HTML tags and elements the same thing?

Ans: HTML Tags and Elements are sometimes perceived as the same. However, they are not.

1. There is a subtle difference between HTML elements and tags that many people aren't aware of.
2. In HTML, tags represent the structural components of a document, such as <h1> for headings.

-> Example: <b> </b>

3. Elements are formed by tags and encompass both the opening and closing tags along with the content.

-> Example: <b> This is the content. </b>

4. HTML tags are almost like keywords where every single tag has unique meaning.

5. HTML element spicifies the general content.

Q-2: What are tags and attributes in HTML?

Ans: HTML tags are what defines where an HTML element starts and where it ends.

1. There is usually an opening bracket followed by the element’s name and, finally, a closing bracket.
2. In most cases, there is always a start tag and an end tag enclosing an element. Nevertheless, that is not the case for all elements.
3. As far as HTML is concerned, there are several types of a tag.

=> Example: <title>, <p>, <b>

1. In other cases, the tags may contain other aspects.
2. That usually happens when it comes to the opening tags.
3. They may contain an extra attribute, including the likes of height, width, or CSS class name, among others.

=> Example: <p class='info'>

Q-3: What are void elements in HTML?

Ans: A void element is an element whose content model never allows it to have contents under any circumstances.

1. Void elements can have attributes.

2. The following is a complete list of the void elements in HTML: area, base, br, col, command, embed, hr, img, input, keygen, link, meta, param, source, track, wbr.

3. Void elements only have a start tag; end tags must not be specified for void elements.

Q4: What are HTML Entities?

Ans: An HTML entity is a piece of text ("string") that begins with an ampersand (&) and ends with a semicolon (;).

1. HTML entities are frequently used to display reserved characters and invisible characters (like non-breaking spaces).

2. Some special characters are reserved for use in HTML, meaning that your browser will parse them as HTML code

3. To display these characters as text, replace them with their corresponding character entities.

=> Example: &nbsp;

&copy;

&reg;

Q-5: What are different types of lists in HTML?

Ans: There are several types of lists that you can use to organize and display items.

1. Ordered List (<ol>):

-> This type of list is used for items that need to be displayed in a specific order.

-> Items are typically numbered.

-> Example:

<ol>

<li>First Product</li>

<li>Second Product</li>

<li>Third Product</li>

</ol>

2. Unordered List (<ul>):

-> This type of list is used for items that do not need to be displayed in a specific order.

-> Items are typically marked with bullets.

-> Example:

<ul>

<li>First Product</li>

<li>Second Product</li>

<li>Third Product</li>

</ul>

3. Description List (<dl>):

-> This type of list is used for pairs of terms and descriptions.

-> It consists of <dt> (definition term) and <dd> (definition description) elements.

-> Example:

<dl>

<dt>Term 1</dt>

<dd>Description for term 1</dd>

<dt>Term 2</dt>

<dd>Description for term 2</dd>

</dl>

4. Nested Lists:

-> Lists can be nested inside each other to create sub-lists.

-> Example with an unordered list nested inside an ordered list:-

<ol>

<li>First Product</li>

<li>Second Product<li>

<ul>

<li>Subitem 1</li>

<li>Subitem 2</li>

</ul>

</li>

<li>Third Product</li>

</ol>

Q-6: What is the ‘class’ attribute in HTML?

Ans: The HTML class attribute specifies one or more class names for an element.

1. Classes are used by CSS and JavaScript to select and access specific elements.

2. The class attribute can be used on any HTML element.

3. The class name is case sensitive.

4. Different HTML elements can point to the same class name.

=> Example:

<style>

. highlight {

background-color: yellow;

}

</style>

<p class="highlight">This paragraph will have a yellow background. </p>

Q-7: What is the difference between the ‘id’ attribute and the ‘class’ attribute of HTML elements?

Ans: => ID Attribute: Uniquely identifies one element.

=> Class Attribute: Can be applied to multiple elements.

=> ID Attribute: Primarily used for styling or JavaScript.

=> Class Attribute: Also used for styling or JavaScript.

=> ID Attribute: Only one element can have a specific ID.

=> Class Attribute: Multiple elements can share the same class

=> ID Attribute: Written as id="example".

=> Class Attribute: Written as class="example".

=> ID Attribute: Accessed in CSS with #example selector.

=> Class Attribute: Accessed in CSS with .example selector.

=> ID Attribute: Often used for unique page elements.

=> Class Attribute: Commonly used for styling groups of elements.

Q-8: What are the various formatting tags in HTML?

Ans: => HTML provides many predefined elements that are used to change the formatting of text.

=> The formatting can be used to set the text styles, highlight the text, make text superscript and subscript, etc.

=> <b> and <strong> Tags: Both tags are used to make the text bold. The text content of the tag is shown as important information on the webpage.

=> <i> and <em> Tags: Both tags are used to make the text italic and emphasized. Both the element has opening and closing tags.

=> <small> and <big> Tags: The <small> tag is used to set small font-size whereas <big> tag is used to set big font-size.

=> <sup> and <sub> Tags: The <sup> tag is used to superscript a text whereas <sub> tag is used to subscript a text.

=> <ins> and <del> Tag: The <ins> tag is used to underline a text marking the part as inserted or added. It also has an opening and a closing tag. This tag is mainly used in text in place of deleted text whereas the <del> tag is used to delete the text it adds a strike line to the text.

=> HTML <mark> Tag: The <mark> tag is used to highlight a text. It has an opening and closing tag.

Q-9: How is Cell Padding different from Cell Spacing?

Ans: => Cellpadding: Cell padding is the term used to describe the area between a table cell’s border and its content.

=> Cell spacing: The gap between each neighbouring cell is often called “cell spacing.”

=> Cellpadding: It may be made using the HTML table> tag but changes the type property to cell padding.

=> Cell spacing: It may be produced using the HTML table> tag. However, that changes the type property to cell spacing.

=> Cellpadding: It concerns just one cell.

=> Cell spacing: It is exposed to several cells (more than one) at once.

=> Cellpadding: The default value for cell padding is 1.

=> Cell spacing: Cell spacing has a default value of 2.

=> Cellpadding: When compared to cell spacing, it is quite effective. It is, therefore, very commonly used.

=> Cell spacing: Comparatively speaking, it is less efficient than cell padding.

Q-10: How can we club two or more rows or columns into a single row or column in an HTML table?

Ans: The purpose of this article is to explore the method of merging table cells in HTML using the rowspan and colspan attributes.

=> By utilizing rowspan, multiple cells in a row can be merged or combined, while colspan enables the merging of cells in a column within an HTML table.

=> This technique proves essential for creating visually organized and structured tables, and optimizing the presentation of data.

=> Example:

<html>

<body style="text-align:center" border=2px>

<table align="center">

<tr>

<th>Name</th>

<th>Age</th>

</tr>

<tr>

<td>ABC</td> <! -- This cell will take up space on two rows -->

<td rowspan="2">44</td>

</tr>

<tr>

<td>XYZ</td>

</tr>

</table>

</body>

</html>

Q-11: What is the difference between a block-level element and an inline element?

Ans: => INLINE ELEMENTS: Take up only as much width as necessary to display their content.

=> BLOCK ELEMENTS: Take up the full width of their parent container by default.

=> INLINE ELEMENTS: Take up only as much height as necessary to display their content.

=> BLOCK ELEMENTS: Take up the height of their content by default, but can be set to a fixed or percentage height.

=> INLINE ELEMENTS: Do not cause line breaks.

=> BLOCK ELEMENTS: Cause line breaks before and after the element.

=> INLINE ELEMENTS: Only horizontal margins and padding can be applied to them.

=> BLOCK ELEMENTS: All types of margins and padding can be applied to them.

=> INLINE ELEMENTS: <strong>, <em>, <a>, <span>

=> BLOCK ELEMENTS: <div>, <p>, <h1> - <h6>, <ul>, <ol>, <table>, <form>

Q-12: How to create a Hyperlink in HTML?

Ans: => Hyperlinks are one of the most exciting innovations the Web has to offer.

=> They've been a feature of the Web since the beginning, and are what makes the Web a web.

=> Hyperlinks allow us to link documents to other documents or resources, link to specific parts of documents, or make apps available at a web address.

=> A basic link is created by wrapping the text or other content inside an <a> element and using the href attribute, also known as a Hypertext Reference, or target, that contains the web address.

=> Example:

<p>I'm creating a link to

<a href="https://www.mozilla.org/en-US/">the Mozilla homepage</a>.

</p>

Q-13: What is the use of an iframe tag?

Ans: => An inline frame (iframe) is a HTML element that loads another HTML page within the document.

=> It essentially puts another webpage within the parent page.

=> They are commonly used for advertisements, embedded videos, web analytics and interactive content.

=> Example:

<!DOCTYPE html>

<html>

<body>

<h1>The iframe element</h1>

<iframe src="https://www.google.com" title="Online Web Tutorials">

</iframe>

</body>

</html>

Q-14: What is the use of a span tag? Explain with example?

Ans: => Common uses of a span tag include changing the style or color of a part of the text, highlighting a section of text, and inserting icons or small graphics in line with text.

=> It's also useful for applying JavaScript actions to specific parts of inline content.

=> The <span> tag in HTML is an inline-level element used for styling and grouping content within a web page.

=> Unlike block-level elements like <div>, the <span> tag does not inherently introduce any line breaks or block formatting.

=> Instead, it allows you to target and style specific portions of text or inline elements without disrupting the overall flow of the document.

=> Example:

<!DOCTYPE html>

<html>

<head>

<title>Styling Text Example</title>

</head>

<body>

<p>

Welcome to <span style="color: blue;">Shiksha Online</span>, where learning <span style="color: green;">never stops</span>.

</p>

</body>

</html>

Q-15: How to insert a picture into a background image of a web page?

Ans: => Background images are used to make a website more interactive and attractive.

=> It can be applied in many stylings.

=> In the body tag, specify a background image in the background attribute by passing the URL of the image or location path.

=> Example:

<!DOCTYPE html>

<html lang="en">

<head>

<title>Website</title>

</head>

<body background="https://media.geeksforgeeks.org/wp-content/cdn- uploads/20190710102234/download3.png">

<h2>Background Image</h2>

</body>

</html>

Q-16: How are active links different from normal links?

Ans: => The default color for normal and active links is blue.

=> Some browsers recognize an active link when the mouse cursor is placed over that link.

=> Whereas, others recognize active links when the link has the focus.

=> Those that don’t have a mouse cursor over that link is considered a normal link.

Q-17: What are the different tags to separate sections of text?

Ans: => To separate sections of text in HTML, several tags can be used, each serving a different purpose and affecting the structure and semantics of the document.

=> Here are some commonly used tags:

-> <div>: A generic container used to group together related elements for styling or scripting purposes.

-> <section>: Represents a standalone section of content which is thematically related.

-> <header>: Defines the header for a section or page, typically containing introductory content or navigational links.

-> <footer>: Defines the footer for a section or page, typically containing metadata or navigational links.

-> <nav>: Represents a section of the page intended for navigation links.

-> <aside>: Represents a section of the page with content that is tangentially related to the content around it.

-> <main>: Represents the dominant content of the <body> of a document.

-> <h1> to <h6>: Heading tags used to define headings of different levels.

-> <p>: Represents a paragraph.

-> <hr>: Represents a thematic break or a horizontal rule in the content.

Q-18: What is SVG?

Ans: => SVG stands for Scalable Vector Graphics. It is an XML-based markup language for describing two-dimensional vector graphics.

=> SVG allows for the creation of images that can be scaled to different sizes without losing quality, making it ideal for responsive design on the web.

=> SVG supports interactivity and animation. You can add event listeners to SVG elements to make them interactive.

=> SVG graphics can be styled with CSS, similar to how HTML elements are styled.

=> SVG files are often smaller in size compared to other image formats, especially for simple graphics, because they are text-based and can be compressed effectively.

=> Example:

<svg width="100" height="100"xmlns="http://www.w3.org/2000/svg">

<circle cx="50" cy="50" r="40" stroke="black" stroke-width="3" fill="red" />

</svg>

Q-19: What is difference between HTML and XHTML?

Ans: => HTML: Hypertext Markup Language is a markup language used to create web pages and other information that can be displayed in a web browser.

=> XHTML: extensible Hypertext Markup Language is a markup language that is a stricter version of HTML and conforms to XML syntax.

=> HTML: HTML allows for loose syntax, with end tags and attributes often being optional.

=> XHTML: XHTML requires end tags for all elements and attributes to be quoted.

=> HTML: HTML allows for multiple DTDs, including HTML 4.01 and HTML5.

=> XHTML: XHTML requires the use of a specific DTD, such as XHTML 1.0 Strict or XHTML 1.1.

=> HTML: HTML does not support namespaces.

=> XHTML: XHTML supports namespaces, allowing for the integration of other XML languages.

=> HTML: HTML allows for the use of deprecated attributes.

=> XHTML: XHTML does not allow the use of deprecated attributes and requires all attributes to be lowercase.

=> HTML: HTML will continue to be supported by web browsers.

=> XHTML: XHTML support by web browsers is limited and it is now largely replaced by HTML5.

=> HTML: HTML continues to evolve, with the latest version being HTML5.

=> XHTML: XHTML development has largely been discontinued, with future developments focusing on HTML5.

Q-20: What are logical and physical tags in HTML?

Ans: => In HTML, tags can be categorized into logical (semantic) and physical (presentational) tags.

1. Logical (Semantic) Tags:

-> Logical or semantic tags describe the meaning and structure of the content.

-> They provide information about the role of the content within the document, which helps search engines and assistive technologies understand and navigate the content better.

-> They are important for accessibility, SEO, and proper document structure.

-> Example: <p>, <ul>, <ol>, <li>, <a> etc.

2. Physical (Presentational) Tags:

-> Physical or presentational tags are used to directly define the appearance of the content, without providing any semantic meaning.

-> They are mainly concerned with how the content should look.

-> sExample: <tt>, <u>, <i>, <b>