**HTML ASSIGNMENT**

**Module 1) –Foundation**

**Web site:**

* A website is a collection of many web pages, and web pages are digital files that are written using HTML(Hyper Text Markup Language). To make your website available to every person in the world, it must be stored or hosted on a computer connected to the Internet round a clock. Such computers are known as a **Web Server**.
* The website’s web pages are linked with hyperlinks and hypertext and share a common interface and design. The website might also contain some additional documents and files such as images, videos, or other digital assets.
* With the Internet invading every sphere, we see websites for all kinds of causes and purposes. So, we can also say that a website can also be thought of as a digital environment capable of delivering information and solutions and promoting interaction between people, places, and things to support the goals of the organization it was created for.

**Components of a Website:**We know that a website is a collection of a webpages hosted on a web-server. These are the components for making a website.

* **Webhost:**Hosting is the location where the website is physically located. Group of webpages (linked webpages) licensed to be called a website only when the webpage is hosted on the webserver. The webserver is a set of files transmitted to user computers when they specify the website’s address..
* **Address:**Address of a website also known as the URL of a website. When a user wants to open a website then they need to put the address or URL of the website into the web browser, and the asked website is delivered by the webserver.
* **Homepage :**Home page is a very common and important part of a webpage. It is the first webpage that appears when a visitor visits the website. The home page of a website is very important as it sets the look and feel of the website and directs viewers to the rest of the pages on the website.
* **Design :**It is the final and overall look and feel of the website that has a result of proper use and integration elements like navigation menus, graphics, layout, navigation menus etc.
* **Content :**Every web pages contained on the website together make up the content of the website. Good content on the webpages makes the website more effective and attractive.
* **The Navigation Structure:**The navigation structure of a website is the order of the pages, the collection of what links to what. Usually, it is held together by at least one navigation menu.

**How to access Websites?**  
When we type a certain URL in a browser search bar, the browser requests the page from the Web server and the Web server returns the required web page and its content to the browser. Now, it differs from how the server returns the information required in the case of static and dynamic websites

**Types of Website:**

* Static Website
* Dynamic Website
* **Static Website:** In Static Websites, Web pages are returned by the server which are prebuilt source code files built using simple languages such as HTML, CSS, or JavaScript. There is no processing of content on the server (according to the user) in Static Websites. Web pages are returned by the server with no change therefore, static Websites are fast. There is no interaction with databases. Also, they are less costly as the host does not need to support server-side processing with different languages.
* **Dynamic Website:** In Dynamic Websites, Web pages are returned by the server which is processed during runtime means they are not prebuilt web pages, but they are built during runtime according to the user’s demand with the help of server-side scripting languages such as PHP, Node.js, ASP.NET and many more supported by the server. So, they are slower than static websites but updates and interaction with databases are possible. Dynamic Websites are used over Static Websites as updates can be done very easily as compared to static websites (Where altering in every page is required) but in Dynamic Websites, it is possible to do a common change once, and it will reflect in all the web pages.

There are different types of websites on the whole internet, we had chosen some most common categories to give you a brief idea –

**Example:**

* **Blogs:**These types of websites are managed by an individual or a small group of persons, they can cover any topics — they can give you fashion tips, music tips, travel tips, fitness tips. Nowadays professional blogging has become an external popular way of earning money online.
* **E-commerce:**These websites are well known as online shops. These websites allow us to make purchasing products and online payments for products and services. Stores can be handled as standalone websites.
* **Portfolio:**These types of websites acts as an extension of a freelancer resume. It provides a convenient way for potential clients to view your work while also allowing you to expand on your skills or services.
* **Brochure:**These types of websites are mainly used by small businesses, these types of websites act as a digital business card, and used to display contact information, and to advertise services, with just a few pages.
* **News and Magazines:**These websites needs less explanation, the main purpose of these types of websites is to keep their readers up-to-date from current affairs whereas magazines focus on the entertainment.
* **Social Media:**We all know about some famous social media websites like Facebook, Twitter, Reddit, and many more. These websites are usually created to let people share their thoughts, images, videos, and other useful components.
* **Educational:**Educational websites are quite simple to understand as their name itself explains it. These websites are designed to display information via audio or videos or images.
* **Portal:**These types of websites are used for internal purposes within the school, institute, or any business, These websites often contain a login process allowing students to access their credential information or allows employees to access their emails and alerts.

**Html:**

* HTTP stands for HyperText Transfer Protocol.
* It is a protocol used to access the data on the World Wide Web (www).
* The HTTP protocol can be used to transfer the data in the form of plain text, hypertext, audio, video, and so on.
* This protocol is known as HyperText Transfer Protocol because of its efficiency that allows us to use in a hypertext environment where there are rapid jumps from one document to another document.
* HTTP is similar to the FTP as it also transfers the files from one host to another host. But, HTTP is simpler than FTP as HTTP uses only one connection, i.e., no control connection to transfer the files.
* HTTP is used to carry the data in the form of MIME-like format.
* HTTP is similar to SMTP as the data is transferred between client and server. The HTTP differs from the SMTP in the way the messages are sent from the client to the server and from server to the client. SMTP messages are stored and forwarded while HTTP messages are delivered immediately.

**Freatures Of Http :**

* **Connectionless protocol:** HTTP is a connectionless protocol. HTTP client initiates a request and waits for a response from the server. When the server receives the request, the server processes the request and sends back the response to the HTTP client after which the client disconnects the connection. The connection between client and server exist only during the current request and response time only.
* **Media independent:** HTTP protocol is a media independent as data can be sent as long as both the client and server know how to handle the data content. It is required for both the client and server to specify the content type in MIME-type header.
* **Stateless:** HTTP is a stateless protocol as both the client and server know each other only during the current request. Due to this nature of the protocol, both the client and server do not retain the information between various requests of the web pages.

**URL:**

A URL is a type of uniform resource identifier and is address of a resource on the World Wide Web and the protocol used to access it. It is used to indicate the location of a web resource to access the web pages.

The URL sends users to a specific resource online such as video, webpage, or other resources. When you search any query on Google, it will display the multiple URLs of the resource that are all related to your search query. The displayed URLs are the hyperlink to access the webpages.

A URL (Uniform Resource Locator) contains the information, which is as follows:

* The port number on the server, which is optional.
* It contains a protocol that is used to access the resource.
* The location of the server
* A fragment identifier
* In the directory structure of the server, it contains the location of the resource.

**Example:**

<https://www.hostinger.com/>

<https://www.microsoft.com/>

**Where is the URL located?**

A URL is located in the address bar or search bar at the top of the browser window. The URL is always visible in the desktop computers and laptop unless your browser is being displayed in full screen. In most of the smartphones and tablets, when you scroll down the page, the URL will disappear and only show the domain when visible. To visible the address bar, you need to scroll up the page. And, if only the domain is shown and you want to see full address, tapping on the address bar to show the full address.

**Why URL?**

The URL is beneficial, as the written information in the URL provides users the option to switch from one web page to another by clicking only one mouse click.

Every URL is unique and tells users how to access a specific resource.When a user types a URL into the web browser and opens any hyperlink from search results, the browser forwards a request to a webserver to fetch files related to the search query.

A website domain or URL identifies one particular file, and it is the most important part of your website. Usually, by using words that end with .net, .com, or .org, you can get traffic on your website.

**Module 2) Fundamentals Of IT**

**Domain:**

A domain name is the identity of one or more IP addresses; for example, the domain name google.com points to the IP address "74.125.127.147". Domain names are invented as it is easy to remember a name rather than a long string of numbers. It would be easy to enter a domain name in the search bar than a long sequence of numbers.

So, it is the web address of your website that people need to type in the browser URL bar to visit your website. In simple words, suppose your website is a house, then the domain name is its address.

A domain name cannot have more than sixty-three characters excluding .com, .net, .org, .edu, etc. The minimum length of a domain is one character excluding the extensions. It is entered in the URL after the protocol and subdomain as shown in the following example and the image:

E.g. <https://www.google.com>

**How Domain Name Works:**

When the domain name is entered in your web browser, a request is sent to the global network of servers that form the Domain Name System (DNS), which is like a phonebook of the internet.

The server then searches the name servers related to the domain and forwards the request to the name servers. The name servers are big computers, which are managed by hosting companies. The hosting company forwards the request to the webserver where your site is stored. The web server fetches the requested web page or information and forwards it to the browser.

The Domain Names System is managed by Internet Corporation for Assigned Names and Numbers (ICANN). It is a non-profit organization that creates and implements the policies for domain names.

ICANN authorizes the companies called Domain Name Registrars for selling domain names. It also allows them to make changes to domain names registry on your behalf, and to sell domain names, manages their records, renewal, and transfer to other registrars. As a domain name owner, you are required to renew your domain registration before it expires.

**Hosting:**

With web hosting, you will be able to store your online content in a secure place. The text, videos, images, and the code that comprise your website will be stored on a server. With web hosting, anyone that has access to the Internet can access your site.

Self-hosting is running your own site by setting up a network or server yourself. Self-hosting means that you own your site, and you have unlimited capabilities to what you can do with it. Those who choose to self-host their website, besides building their website, are also responsible for uptime and maintenance of their site.

**How Does Web Hosting Work?**

The server that hosts your website is a physical computer that runs continuously to make the site available for visitors all the time. Buying servers for web hosting will allow you to store all the data of your website in the those servers of your provider.

Once a user enters your domain name into their browser’s address bar, the web host’s server will transfer all the files necessary to load your website.

You can host a website yourself, but it requires extensive technical skills. Self-hosting entails setting up and configuring a web server from scratch, including the equipment, infrastructure, hardware, and software. Furthermore, you will also have to handle all the ongoing maintenance.

A web hosting service provider ensures that your website performs optimally and with better security protocols. In addition, it simplifies the many complex aspects of hosting a website – from software installation to technical support.

**Types of Web Hosting Services:**

**Shared Hosting:**

With shared hosting, multiple users share the same server resources, including memory, processing power, and storage space.Because of its simplicity and affordability, shared web hosting is an excellent solution for small businesses and personal websites that do not require advanced configuration or higher bandwidth. Hence, shared hosting is an excellent choice for beginners that need cheap hosting to start.

Pros

* Cost-effective, ideal for small-scale websites
* Technical expertise is not required
* Pre-configured server options
* No need to take care of maintenance and server administration

Cons

* Minimal access to server configuration
* Increased traffic on other websites can affect your website’s speed

**Virtual Private Server (VPS) Hosting:**

With this web hosting type, your website also shares a physical server with other users, but the web host creates a virtual partition for each user. Thus, a site hosted on a virtual private server get an allocated amount of resources.

VPS web hosting is a great option for medium-sized sites, eCommerce shops, and large blogs with a rapidly growing number of visitors.

Pros

* Dedicated server space
* Increased traffic on other websites has no impact on your site’s performance
* Root access to the server
* High customizability

Cons

* Users need technical expertise to manage it
* Even though it’s relatively affordable, some users may have to hire a developer to manage the virtual server, increasing the overall costs

**Cloud Hosting:**

This web hosting solution uses several virtual servers to host sites. Thus, if one server experiences high traffic or a problem, the remaining ones will take over and maintain the website operating.

Since cloud based web hosting relies on a cluster of web servers to function, businesses with multiple websites and large-scale sites like eCommerce shops can benefit the most from it, as it provides little to no downtime.

Pros

* Reduced likelihood of downtime and hardware failure
* Uses load balancing to handle high traffic and prevent DDoS attacks
* Scalability – your website is not limited to the resources of a single server

Cons

* Root access is not always provided
* It is more expensive than VPS and shared hosting

**WordPress Hosting:**

This type of CMS web hosting service provides a WordPress-optimized server environment to help your site load faster and minimize potential issues. However, other types of web hosting still work for websites based on this popular content management system (CMS).

Typically, WordPress-optimized hosting plans come with features such as pre-installed themes, plugins for core functions like caching and security, and other tools.

Pros

* Low cost and beginner-friendly
* Optimized performance for WordPress sites
* Customer support team trained in WordPress issues
* Pre-installed WordPress plugins and themes

Cons

* Not an ideal type of web hosting for non-WordPress websites

### Dedicated Hosting

Dedicated hosting designates a physical server for each website. By going with dedicated hosting, you can configure the server, choose your desired operating system and software, and customize the entire hosting environment to your specifications.

Renting a dedicated server is just as powerful as having your own on-site server, but with the added benefit of getting professional support from your web host. Thus, dedicated hosting is ideal for large online businesses that deal with heavy traffic.

Pros

* Complete control over the server’s configuration
* High reliability
* Root access to the server

Cons

* High cost, more oriented towards large businesses
* Technical and server management knowledge is required

**SEO:**

SEO stands for Search Engine Optimization. It is a process designed to optimize a website for search engines. It helps websites achieve a higher ranking in search engine results when people search for keywords related to their products and services. So, it is a practice of increasing the quantity and quality of traffic to a website through organic search engine results.

How Search Engine Optimization Works:

Search Engines such as Google have their own algorithm or rules to decide the order of pages to show for a search query. These algorithms determine the rankings of the SERPs based on various ranking factors. However, it gives more emphasis on certain metrics to evaluate the quality of a page and accordingly to decide its ranking.

**How do search engines work?**

In order to understand how SEO works, it’s vital to have a basic understanding of how search engines work. Search engines use crawlers (also known as spiders or bots) to gather information across the internet to populate their big databases, called “indexes”. Crawlers begin from a known web page and then follow links from that page to other pages.

For example, if a page Google already indexed on Patagonia.com on the topic of used clothing features internal links to further pages on the site for used jackets, used hiking boots, and used flannel shirts, Google can crawl to those pages via the links provided. Meanwhile, if Patagonia’s main used clothing page links out to an article on TheGuardian.com about the negative impacts of fast fashion, Google can crawl from Patagonia to the news article via the link, thereby discovering that content and potentially indexing it.

The content of the discovered page, and the context of the links the crawler followed from Patagonia to The Guardian, help Google understand what the page is about and how it is relevant to all of the other pages within its index.

If you happen to be the journalist who wrote The Guardian article on fast fashion, the fact that a used outdoor clothing section of a large brand is linking to your piece is an indication to Google that there might be a relationship between the problems of fast fashion and the potential solution of buying used clothing instead of new clothing. These semantic relationships go far towards helping Google determine which results to show for each query they receive from the searching public.

Search engines’ success as businesses depends on the public finding search engine results to be relevant to their needs. The more links a search engine like Google finds pointing from a particular type of content to a particular resource, the more confident it becomes that the linked-to resource is relevant to certain search queries. The search engine then determines that this resource deserves to be ranked highly when people make those queries.

There are three main categories of SEO: on-page SEO, off-page SEO, and technical SEO, all of which combine to help search engines discover, crawl, index, understand, and rank your content, and this article will cover each of these topics.

**Module 3) (HTML)**

**Q.1 Are the HTML tags and elements the same thing?**

**ANS:-**

HTML Tags: HTML tags are like keywords which defines that how web browser will format and display the content. With the help of tags, a web browser can distinguish between an HTML content and a simple content. HTML tags contain three main parts: opening tag, content and closing tag. But some HTML tags are unclosed tags.

HTML Elements: An HTML file is made of elements. These elements are responsible for creating web pages and define content in that webpage. An element in HTML usually consist of a start tag <tag name>, close tag </tag name> and content inserted between them. Technically, an element is a collection of start tag, attributes, end tag, content between them.

**Q.2** **What are tags and attributes in HTML?**

**ANS:-**

**Tags:**

* Definition: Tags are the fundamental building blocks of HTML. They define the structure of the content on a web page.
* Syntax: Tags are typically written as pairs – an opening tag and a closing tag – surrounding the content. The opening tag indicates the beginning of an element, and the closing tag marks its end. For example:
* When a web browser reads an HTML document, browser reads it from top to bottom and left to right. HTML tags are used to create HTML documents and render their properties. Each HTML tags have different properties.
* An HTML file must have some essential tags so that web browser can differentiate between a simple text and HTML text. You can use as many tags you want as per your code requirement.

**Example**:

<p>This is a paragraph.

</p><h1>h</h1>

Types: Tags can be classified into two main types: container tags and empty (self-closing) tags.

* Container Tags: These have both opening and closing tags and enclose content between them.

Example:-

<div>This is a container tag.</div>

EmptyTags: These are self-closing and don't have a separate closing tag. They may include attributes.

Example:-

<img src="image.jpg" alt="An image">

**Attributes:**

* Definition: Attributes provide additional information about HTML elements. They are always included in the opening tag of an element and are written as name-value pairs.
* Syntax: An attribute is typically included within the opening tag of an element and is written as name="value". For example:

**Example**:-

<element attribute\_name="value">content</element>

<a **href**="https://www.example.com">Visit Example</a>

**Common Attributes:**

* id: Provides a unique identifier for an element.
* class: Assigns one or more class names to an element.
* src: Specifies the source URL for elements like images or scripts.
* href: Specifies the hyperlink destination for anchor (<a>) elements.
* alt: Defines alternative text for elements like images.

**Q.3 What are void elements in HTML? With Example.**

**ANS:-**

All the elements in HTML do not require to have start tag and end tag, some elements does not have content and end tag such elements are known as Void elements or empty elements. These elements are also called as unpaired tag.

**<img> (Image):**

* Purpose: HTML img tag is used to display image on the web page. HTML img tag is an empty tag that contains attributes only, closing tags are not used in HTML image element.
* **Example**:

<h2>HTML Image Example</h2>

<img src="good\_morning.jpg" alt="Good Morning Friends"/>

**<br> (Line Break):**

* Purpose: The <br> tag in HTML document is used to create a line break in a text.
* It is generally used in poem or address where the division of line is necessary. It is an empty tag, which means it does not need a company of end tag. If you place the <br> tag in the HTML code, then it works the same as pressing the enter key in a word processor.
* **Example**:

Text <br> Text

**<hr> (Horizontal Rule):**

* Purpose: Represents a thematic break or separation within a page.
* **Example:**

<p>Some content above<hr>Some content below</p>

**<input> (Input):**

* Purpose: The HTML <input> tag is used to represent a form input control in HTML document. This form input control facilitate user to input data and communicate with a website or application. Let's take an example of an HTML form with three input fields, two text fields and one button for submission.
* **Example:**

<form action="#">

First name: <input type="text" name="FirstName" placeholder="enter firstname..."><br>

Last name: <input type="text" name="LastName" placeholder="enter lastname..."><br>

<input type="submit" value="Submit">

</form>

**<meta> (Metadata):**

* Purpose: HTML <meta> tag is used to represent the metadata about the HTML document. It specifies page description, keywords, copyright, language, author of the documents, etc.
* The metadata does not display on the webpage, but it is used by search engines, browsers and other web services which scan the site or webpage to know about the webpage.
* With the help of meta tag, you can experiment and preview that how your webpage will render on the browser.
* The <meta> tag is placed within the <head> tag, and it can be used more than one times in a document.
* **Example:**

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8">

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

<meta name="description" content="s">

<meta name="author" content="r">

<meta http-equiv="refresh" content="5; url=https://www.data.com/html-tags- list">

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

</head>

<body>

<h2>Example of Meta tag</h2>

<p>This example shows the use of meta tag within an HTML document</p>

</body>

</html>

**<link> (Link):**

Purpose: HTML <link> tag is used to specify the relationship between the current document and external source.

The <link> tag is commonly used to link the external Stylesheet for the current document, but it can also use with link site icons. It is placed on the head section of the document.

* **Example:**

<link rel="stylesheet" type="text/css" href="">

**Q.4 What are HTML Entities? With Example.**

**ANS:-**

**Named Entities:**

HTML character entities are used as a replacement of reserved characters in HTML. You can also replace characters that are not present on your keyboard by entities.

These characters are replaced because some characters are reserved in HTML. HTML entities provide a wide range of characters which can allow you to add icons, geometric shapes, mathematical operators, etc.

* Named entities use a specific name or keyword to represent a character. For example:
  + &lt;: Represents the less-than symbol <.
  + &gt;: Represents the greater-than symbol >.
  + &amp;: Represents the ampersand &.
  + &quot;: Represents the double quotation mark "

**Example:-**

<!DOCTYPE html>

<html>

<head>

<title></title>

</head>

<body>

<h3>HTML entity example</h3>

<p> "This is the content written within entity"</p>

<p> <p> Paragraph tag </p>

</body>

</html>

Output:-

Css:

This is an example of <em>HTML</em> entities.

**Numerical Entities:**

* Numerical entities use the Unicode or ASCII code of the character to represent it. They begin with an ampersand (&), followed by the pound sign (#), and then the numerical code, ending with a semicolon (;). For example:
  + &#60;: Represents the less-than symbol <.
  + &#62;: Represents the greater-than symbol >.
  + &#38;: Represents the ampersand &.
  + &#34;: Represents the double quotation mark ".

Example:-

<p>This is an example of &#60;em&#62;HTML&#60;/em&#62; entities.</p>

Css:-

This is an example of <em>HTML</em> entities.

**Q.5 What are different types of lists in HTML? With Example.**

**ANS:-**

**Ordered Lists (<ol>):**

HTML Unordered List or Bulleted List displays elements in bulleted format . We can use unordered list where we do not need to display items in any particular order. The HTML ul tag is used for the unordered list

<ol>

<li>data 1</li>

<li>data 2</li>

<li>data 3</li>

</ol>

**Output:-**

1. data 1
2. data 2
3. data 3

**Unordered Lists (<ul>):**

<ul>

<li>data A</li>

<li>data B</li>

<li>data C</li>

</ul>

**Output:**

* data A
* data B
* data C

**Definition Lists (<dl>):**

<dl>

<dt>Term 1</dt>

<dd>Definition 1</dd>

<dt>Term 2</dt>

<dd>Definition 2</dd>

</dl>

**Output:-**

* Term 1
* Definition 1 Term 2
* Definition 2

<ol>

<li>Ordered List Item 1</li>

<li>Ordered List Item 2

<ul>

<li>Nested Unordered List Item 1</li>

<li>Nested Unordered List Item 2</li>

</ul>

</li>

<li>Ordered List Item 3</li>

</ol>

HTML provides several types of lists that allow you to organize and structure information. The three main types of lists in HTML are:

**Q.6 What is the ‘class’ attribute in HTML? With Example.**

**ANS:-**

The HTML class attribute is used to specify a single or multiple class names for an HTML element. The class name can be used by CSS and JavaScript to do some tasks for HTML elements. You can use this class in CSS with a specific class, write a period (.) character, followed by the name of the class for selecting elements.

A class attribute can be defined within <style> tag or in separate file using the (.) character.

In an HTML document, we can use the same class attribute name with different elements.

**Example:-**

<!DOCTYPE html>

<html>

<head>

<style>

.headings{

color: lightgreen;

font-family: cursive;

background-color: black; }

</style>

</head>

<body>

<h1 class="headings">This is first heading</h1>

<h2 class="headings">This is Second heading</h2>

<h3 class="headings">This is third heading</h3>

<h4 class="headings">This is fourth heading</h4>

</body>

</html>

**Q.7 What is the difference between the ‘id’ attribute and the ‘class’ attribute of HTML elements? With Example.**

**ANS:-**

**id Attribute:**

* **Purpose:** The id attribute is used to uniquely identify a single HTML element on a page. No two elements in the same HTML document should have the same id.
* **Usage:** Typically, id is used for elements that have a unique and specific role or significance on a page.
* **Example:**

<div id="header">This is the header</div>

<p id="main-content">This is the main content</p>

<footer id="page-footer">This is the footer</footer>

**class Attribute:**

* **Purpose:** The class attribute is used to assign one or more class names to an HTML element. Classes are used to group multiple elements together, and the same class can be applied to multiple elements.
* **Usage:** class is commonly used for styling purposes, where you want to apply the same styles to multiple elements, or for JavaScript to select and manipulate multiple elements.
* **Example:**

<p class="highlight">This is a highlighted paragraph.</p>

<p class="highlight">So is this one.</p>

**Q.8 What are the various formatting tags in HTML?**

**ANS:-**

<b> This is a physical tag, which is used to bold the text written between it.

<strong> This is a logical tag, which tells the browser that the text is important.

<i> This is a physical tag which is used to make text italic.

<em> This is a logical tag which is used to display content in italic.

<mark> This tag is used to highlight text.

<u> This tag is used to underline text written between it.

<tt> This tag is used to appear a text in teletype. (not supported in HTML5)

<strike> This tag is used to draw a strikethrough on a section of text. (Not supported in HTML5)

<sup> It displays the content slightly above the normal line.

<sub> It displays the content slightly below the normal line.

<del> This tag is used to display the deleted content.

<ins> This tag displays the content which is added

<big> This tag is used to increase the font size by one conventional unit.

<small> This tag is used to decrease the font size by one unit from base font size.

**Q.9 How is Cell Padding different from Cell Spacing? With Example.**

**ANS:-**

**Cell Padding:**

Cellpadding is the attribute that defines the space in a cell between the cell content and its border. To put it another way, it is the attribute of the table tag (<table>) that specifies the spacing between the cell content and its border. In HTML, the unit of this distance might be represented in pixels or as a percentage. The cellpadding attribute value may alternatively be 0 to meet the needs of the consumers. It is utilized to separate the text from one another, improving the appearance.

Syntax:

You may apply the following syntax to utilize the cellpadding attribute:

<table cellpadding="value" >.....</table>

* **Example:**

<!DOCTYPE html>

<html>

<head>

<style>

td { background:pink;}

</style>

<title>

Web table

</title>

</head>

<body>

<table border="2" cellpadding="15">

<tr>

<td>Java</td>

<td>JavaScript</td>

<td>C++</td>

</tr>

<tr>

<td>Python</td>

<td>Nodejs</td>

<td>Angular</td>

</tr>

</table>

</body>

</html>

**Cell Spacing:**

Cellspacing is something different from cellpadding. In HTML, cellspacing is yet another attribute of the table tag. It regulates the distance between the single cells in a table. By using this feature, developers might simply change the space between the edges of several adjacent cells. It enhances the table's readability. The cell border appears to increase as the number of cellspacing increases.

Cellspacing is most commonly utilized in web design. Different programming languages utilize different syntax for cellspacing, like CSS utilizes the "border-spacing" property, whereas HTML utilizes the "cellspacing" property.

Syntax:

You may apply the following syntax to utilize cellspacing attribute:

<table cellspacing="value" >.....</table>

* **Example:**

<!DOCTYPE html>

<html>

<head>

<style>

td { background:pink;}

</style>

<title>

Web table

</title>

</head>

<body>

<table border="3" cellspacing="25">

<tr>

<td>Java</td>

<td>JavaScript</td>

<td>C++</td>

</tr>

<tr>

<td>Python</td>

<td>Nodejs</td>

<td>Angular</td>

</tr>

</table>

</body>

</html>

**Q.10 How can we club two or more rows or columns into a single row or column in an HTML table? With Example.**

**ANS:-**

**Rowspan:-**

### <table>

### <tr>

### <th>Name</th>

### <td>chirag</td>

### </tr>

### <tr>

### <th rowspan="2">Phone</th>

### <td>555-1234</td>

### </tr>

### <tr>

### <td>555-8745</td>

### </tr>

### </table>

**Colspan:**

<table>

<tr>

<th colspan="2">Name</th>

<th>Age</th>

</tr>

<tr>

<td>brd</td>

<td>chirag</td>

<td>27</td>

</tr>

<tr>

<td>ed</td>

<td>Jack</td>

<td>24</td>

</tr>

</table>

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

**ANS:-**

**Block-level elements:**

They consume the entire width available irrespective of their sufficiency. They always start in a new line and have top and bottom margins. It does not contain any other elements next to it.

* Start on a new line.
* Extend the full width of the container by default (unless styled otherwise).
* Allow setting width, height, margins, and padding.
* Typically used for structural elements and grouping content.

**Examples:**

<h1>-<h6> : This element is used for including headings of different sizes ranging from 1 to 6.

<div>: This is a container tag and is used to make separate divisions of content on the web page.

<hr>: This is an empty tag and is used for separating content by horizontal lines.

<li>: This tag is used for including list items of an ordered or unordered list.

<ul>: This tag is used to make an unordered list.

<ol>: This tag is used to make an ordered list.

<p>: This tag is used to include paragraphs of content in the webpage.

<table>: This tag is used for including the tables in the webpage when there is a need for tabular data.

HTML 5 Semantic block elements:

<header>: This tag is used for including all the main things of the webpage like navbar, logos, and heading of the webpage.

<nav>: This tag helps to navigate through different sections by including different blocks of hyperlinks in the webpage.

<footer>: This contains all information about the authorization, contact, and copyright details of the webpage.

<main>: The main content of the webpage resides in this tag.

<section> : This is used separate different sections in the webpage.

<article>: This tag is used to include different independent articles on the webpage.

<aside>: This tag is used to mention details of the main content aside.

**Inline elements:**

Inline elements occupy only enough width that is sufficient to it and allows other elements next to it which are inline. Inline elements don’t start from a new line and don’t have top and bottom margins as block elements have.

* + Do not start on a new line; they flow within the content.
  + Only take up as much width as necessary, not the full width of the container.
  + Do not allow setting width, height, margins, or padding.
  + Typically used for styling or emphasizing specific parts of text within a block-level element.

**Examples:**

<a>: This tag is used for including hyperlinks in the webpage.

<br>: This tag is used for mentioning line breaks in the webpage wherever needed.

<script> : This tag is used for including external and internal JavaScript codes.

<input>: This tag is used for taking input from the users and is mainly used in forms.

<img>: This tag is used for including different images in the webpage to add beauty to the webpage.

<span>: This is an inline container that takes necessary space only.

<b>: This tag is used in places where bold text is needed.

<label>: The tag in HTML is used to provide a usability improvement for mouse users i.e, if a user clicks on the text within the <label> element, it toggles the control.

**Q.12 How to create a Hyperlink in HTML? With Example.**

**ANS**:- A link or hyperlink is a connection from one web resource to another. Links allow users to move seamlessly from one page to another, on any server anywhere in the world.

A link has two ends, called anchors. The link starts at the source anchor and points to the destination anchor, which may be any web resource, for example, an image, an audio or video clip, a PDF file, an HTML document or an element within the document itself, and so on.

By default, links will appear as follows in most of the browsers:

An unvisited link is underlined and blue.

A visited link is underlined and purple.

An active link is underlined and red.

Links are specified in HTML using the <a> tag.

A link or hyperlink could be a word, group of words, or image.

<a href="url">Link text</a>

Anything between the opening <a> tag and the closing </a> tag becomes the part of the link that the user sees and clicks in a browser. Here are some examples of the links:

**Example:-**

<!DOCTYPE html>

<html>

<head>

<title>Link Tag</title>

<link rel="stylesheet" type="text/css" href="htmlpages/css/link.css">

</head>

<body>

<h2>Example of Link Tag</h2>

<p>This is paragraph which is styled with external style sheet. </p>

</body>

</html>

**Q.13 What is the use of an iframe tag? With Example.**

**Ans:-**

HTML Iframe is used to display a nested webpage (a webpage within a webpage). The HTML <iframe> tag defines an inline frame, hence it is also called as an Inline frame.

An HTML iframe embeds another document within the current HTML document in the rectangular region.

The webpage content and iframe contents can interact with each other using JavaScript.

Iframe Syntax

An HTML iframe is defined with the <iframe> tag:

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

**Example:-**

<!DOCTYPE html>

<html>

<body>

<h2>HTML Iframes example</h2>

<p>Use the height and width attributes to specify the size of the iframe:</p>

<iframe src="https://www.javatpoint.com/" height="300" width="400"></iframe>

</body>

</html>

**Q.14 What is the use of a span tag? Explain with example?**

**Ans:-**

HTML <span> tag is used as a generic container of inline elements. It is used for styling purpose to the grouped inline elements (using class and id attribute or inline style).

The <span> tag does not have any default meaning or rendering.

HTML <span> is much similar as <div> tag, but <div> is used for block-level elements and <span> tag is used for inline elements.

Syntax

<span>Write your content here......</span>

**Example:-**

<!DOCTYPE html>

<html>

<head>

<title>Span Tag</title>

</head>

<body>

<h2>Example of span tag</h2>

<p>I have choosen only

<span style="color: red;">red</span>,

<span style="color: blue;">blue</span>, and

<span style="color: green;">green</span> colors for my painting.

</p>

</body>

</html>

**Q.15 How to insert a picture into a background image of a web page? With Example.**

**Ans:-**

The <background> attribute in the HTML document is used to specify the background image on a HTML page or a table. You can pass the path of an image as a value of background attribute to set the image of your HTML page or table.

Syntax

<tag background="Path\_of\_an\_image">

**Example:-**

<style>

body {

background-image: url('img\_girl.jpg');

background-repeat: no-repeat;

background-attachment: fixed;

background-size: 100% 100%;

}

</style>

**Q.16 How are active links different from normal links?**

**Ans:-**

Active links and normal links are terms commonly used in the context of web design and HTML/CSS. Let's clarify the differences between them:

**Normal Links:**

* Normal links, often referred to as regular or default links, are the standard links used to navigate between different web pages or resources.
* When a user clicks on a normal link, it typically triggers a navigation event to load a new page or resource.
* In HTML, normal links are created using the <a> (anchor) element.

**Example of a normal link:**

<a href="https://www.google.com">Visit google.com</a>

**Active Links:**

An Active link is a hyperlink that is currently being interacted with the user. Whenever the user holds the mouse button on the link and not released yet or if right clicked on it, it will change its color into red, this is when the link will be in active state.

The active state is temporary and ends once the user releases the mouse button. However, we can customize the style of the active links using the CSS properties (a:active).

**example:**

/\* visited link \*/

a:visited {

color: green;

}

/\* mouse over link \*/

a:hover {

color: red;

}

/\* selected link \*/

a:active {

color: yellow;

}

**Q.17 What are the different tags to separate sections of text?**

**Ans:-**

In HTML, there are several tags that you can use to separate and structure different sections of text. Here are some commonly used tags:

**Paragraphs: <p>**

* HTML paragraph or HTML p tag is used to define a paragraph in a webpage. Let's take a simple example to see how it work. It is a notable point that a browser itself add an empty line before and after a paragraph. An HTML <p> tag indicates starting of new paragraph.

**Example**:-

<p>This is first paragraph.</p>

<p>This is second paragraph.</p>

<p>This is third paragraph.</p>

**Headings:**

A HTML heading or HTML h tag can be defined as a title or a subtitle which you want to display on the webpage. When you place the text within the heading tags <h1>.........</h1>, it is displayed on the browser in the bold format and size of the text depends on the number of heading.There are six different HTML headings which are defined with the <h1> to <h6> tags, from highest level h1 (main heading) to the least level h6 (least important heading).

h1 is the largest heading tag and h6 is the smallest one. So h1 is used for most important heading and h6 is used for least important.

**Example:-**

<h1>Heading no. 1</h1>

<h2>Heading no. 2</h2>

<h3>Heading no. 3</h3>

<h4>Heading no. 4</h4>

<h5>Heading no. 5</h5>

<h6>Heading no. 6</h6>

**Divisions:**

We know that every tag has a specific purpose e.g. p tag is used to specify paragraph, <h1> to <h6> tag are used to specify headings but the <div> tag is just like a container unit which is used to encapsulate other page elements and divides the HTML documents into sections.

**Example:-**

<div>

<p>This is a paragraph inside a div.</p>

<p>Another paragraph inside the same div.</p>

</div>

**Sections: <section>**

* The HTML <section> tag is used to define sections in a document. When you put your content on a web page, it may contains many chapters, headers, footers, or other sections on a web page that is why HTML <section> tag is used.
* HTML <section> is a new tag introduced in HTML5.

**Example:-**

<section>

<h2>Section Title</h2>

<p> data</p>

</section>

**Articles: <article>**

The HTML <article> tag defines an independent self-contained content in a document, page, application or a site.

The article tag content makes sense on its own. It is independent and complete from other content shown on the page. This tag is generally used on Forum post, Blog post, News story, comment etc

**Example:-**

<article>

<h2>Article Title</h2>

<p>data</p>

</article>

**Lists: <ul>, <ol>, <li>:**

* Lists are used to represent ordered or unordered sets of items. <ul> is an unordered list, <ol> is an ordered list, and <li> is used for list items.

**Example:-**

<ul>

<li>data 1</li>

<li>data 2</li>

<li>data 3</li>

</ul>

<ol>

<li>First</li>

<li>Second</li>

<li>Third</li>

</ol>

**Q.18 What is SVG?**

**Ans:-**

The HTML SVG is an acronym which stands for Scalable Vector Graphics.

HTML SVG is a modularized language which is used to describe graphics in XML. It describe two-dimensional vector and mixed vector/raster graphics in XML. It is a W3C recommendation. SVG images and their behaviors are defined in XML text files. So as XML files, you can create and edit an SVG image with text editor, but generally drawing programs like inkspace are preferred to create it.SVG is mostly used for vector type diagrams like pie charts, 2-Dimensional graphs in an X,Y coordinate system etc.

The <svg> element specifies the root of a SVG fragment. You can animate every element and every attribute in SVG files.

**Q.19 What is difference between HTML and XHTML?**

**Ans:-**

**XHTML :**

XHTML stands for Extensible Hypertext Markup Language. It can be considered as a part of the XML markup language this is because of XHTML have features of both XML and HTML. XHTML is extended from XML and HTML. XHTML can be considered as a better version of HTML.

**Advantages of XHTML:**

Strict syntax: XHTML has a stricter syntax compared to HTML5, which means that it is more structured and easier to read.

Standardization: XHTML follows the syntax rules of XML, which is a standardized markup language. This makes it easier to create interoperable web pages that work well with different web browsers and devices.

Better for parsing: XHTML is easier to parse and process than HTML5, which makes it a better choice for developers who want to create web pages that can be easily processed by other software tools.

**Disadvantages of XHTML:**

More difficult to code: XHTML has a stricter syntax compared to HTML5, which can make it more difficult to code.

Not backward compatible: XHTML is not backward compatible with older versions of HTML, which means that some older web browsers may not be able to display XHTML documents properly.

Requires more bandwidth: XHTML documents tend to require more bandwidth compared to HTML5 documents due to their stricter syntax and increased number of tags.

**HTML5:**

HTML is the Hypertext Markup Language which is the most widely used language over the internet. HTML is used to create web pages and link them from one to another. Please note HTML is not a programming language, it is a markup language. We can use different other technologies as like CSS and javascript to give a new look to the pages developed by HTML.

**Advantages of HTML5:**

Easier to code: HTML5 has a more relaxed syntax compared to XHTML, which makes it easier to code.

Backward compatibility: HTML5 is designed to be backward compatible with older versions of HTML, which means that it can be used with older web browsers.

Multimedia support: HTML5 includes support for multimedia elements such as video and audio, which makes it easier to create web pages that include multimedia content.

**Disadvantages of HTML5:**

Non-standardization: HTML5 is not a standardized markup language, which means that different web browsers may interpret it differently.

Security issues: HTML5 includes new features such as geolocation and offline storage, which can create security vulnerabilities if not implemented properly.

Incompatibility with older web browsers: Some older web browsers may not be able to display HTML5 documents properly, which can create compatibility issues for developers.

**Q.20 What are logical and physical tags in HTML?**

**Ans:-**

**Logical Tags :**

Logical Tags are used in HTML to display the text according to the logical styles. Following are the Logical tags commonly used in HTML.

Tag Description

<abbr> Defines an abbreviation

<acronym> Defines an acronym

<address> Defines an address element

<cite> Defines citation

<code> Defines computer code text

<blockquote> Defines a long quotation

<del> Defines text

<dfn> Defines a definition term

<ins> Defines inserted text

<kbd> Defines keyboard text

<pre> Defines preformatted text

<q> Defines short quotation

<samp> Defines sample computer code

<strong> Defines strong text

<var> Defines a variable

**Physical Tags**

Physical Tags are used in HTML to provide actual physical formatting to the text. Following are the Physical tags commonly used in HTML.

Tag Description

<b> Defines bold text

<big> Defines big text

<i> Defines italic text

<small> Defines small text

<sup> Defines superscripted text

<sub> Defines subscripted text

<tt> Defines teletype text

<u> Deprecated. Use styles instead