

Q.1) what are tags & attributes in HTML.

Ans.

1) HTML Tags :- Tags are the starting & ending part of an HTML element. They begin with < symbol & ends with > symbol. Whatever written in between <> this symbol are called tags.

Eg:- <head> </head>
 <body> </body>

 <h2>, <i> </i>

2) HTML Attributes :- It used to define the character of an HTML element. It always placed in the opening tag of an element. It generally provides additional styling to the element.

Eg:- <p align="center"> This is page </p>.

HTML Tags

HTML attributes.

1) HTML tags are used to hold the HTML elements.
 HTML attributes are used to describe the characteristics of an HTML element in detail.

2) HTML tags starts with < and ends with >.

HTML attributes are found only in the starting page.

3) HTML tags are almost like keywords where every tag has unique meaning.

HTML attributes specify various additional properties to the existing HTML element.

Q.2) What are the advantages of collapsing white space.

Ans:-

white spaces refers to empty or blank values in the code which the browser reads & renders. HTML has a special feature of collapsing these white spaces. If you put extra consecutive white spaces or newlines in the code it will regard it as one white space this is known as collapsing of white space.

* Advantages of collapsing white spaces :-

- 1) While you are writing HTML your webpage you want the code to be more understandable / readable to the user.
- 2) Collapsing white spaces decreases the transmission time between the server & the client because collapsing features removes unnecessary bytes that are occupied by the white spaces.
- 3) By mistake, if you leave extra white space, the browser will ignore it and display the UI perfectly.

E.g:- The <hi> tag containing lot of spaces b/w the short form & the full form if you run this code the browser will be see the following output with perfect UI.

```
<!DOCTYPE html>
<html>
<head>
<title> My page </title>
</head>
<body>
<p> This is my page. My name  
is Naresh </p>.
</body>
</html>.
```

p.3) what are HTML Entities.

Ans:

Some characters are reserved in HTML if you use the less than (<) or greater than (>) signs in your text, the browser might mix them with tags. character entities are used to display reserved characters in HTML.

To display less than sign (<) we must write : < ; or < ;

* Advantage of using an HTML entity name :- An entity name is easy to remember.

* Disadvantage of using an HTML entity name :- Browser may not support all entity names, but the support for entity numbers is good.

* Non-breaking Space.

A commonly used in HTML is the non-breaking space : ; A non-breaking space is a space that will not break into a new line.

Eg:- Naresh vilasao
Kapade .

Q. 4) what are different types of lists in HTML
Ans'

HTML lists allow web developers to group a set of related items in lists.

Eg :-

```
<!DOCTYPE html>
<html>
<head>
<title> List </title>
</head>
<body>
<h1> Unordered List </h1>
<ul>
<li> coffee </li>
<li> Tea </li>
<li> milk </li>
</ul>
</body>
<h1> ordered List </h1>
<ol>
<li> coffee </li>
<li> Tea </li>
<li> milk </li>
</ol>
</html>
```

output :-

Unordered list

- coffee
- Tea
- milk

ordered list

1. coffee
2. Tea
3. milk

* Unordered HTML list

An unordered list starts with `` tag. each list item starts with `` tag. the list items will be marked with bullets (small black circles) by default:

E.g. ``

```
<li> coffee </li>
<li> Tea </li>
<li> milk </li>
</ul>
```

* ordered list :-

An ordered list starts with `` tag. each list item starts with `` tag. the list items will be marked with bullets & numbers by default.

E.g. ``

```
<li> coffee </li>
<li> Tea </li>
<li> milk. </li>
</ol>
```

* HTML Description lists :-

HTML also supports description lists.
A description list is a list of terms, with description of each term.

The `<dl>` tag defines description list, the `<dt>` tag defines the term (name), & the `<dd>` tag describes each term.

E.g.:-

```
<dl>
  <dt> coffee <dd>
    <dd>- black hot drink </dd>
  <dt> MILK <dd>
    <dd>- white cold drink </dd>
</dl>
```

Output :-

Coffee
- black hot drink.
MILK
- white cold drink.

Q.5) Describe HTML layout structure.

Ans:-

* HTML layouts:-

HTML layouts provides a way to arrange web pages in well-mannered, well-structured & in responsive form or we can say that HTML layout specifies a way in which the web pages can be arranged. web-page layout works with visual elements of an HTML element of an HTML document.

* Web page :-

Webpage layout is the most important part to keep in mind while creating a website so that our website can appear professional with the great look. You can also use CSS & JavaScript based frameworks for creating layouts for responsiveness & dynamic website designing.

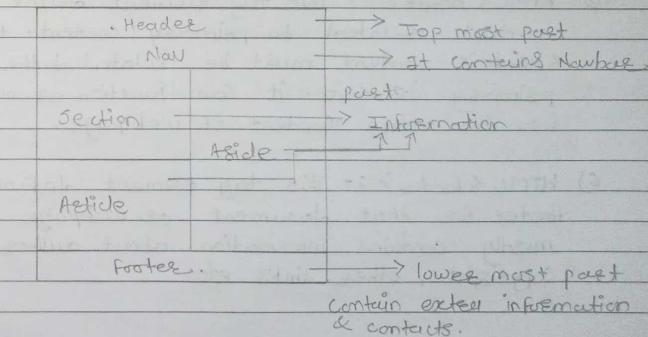


Fig: HTML layout Structure.

- 1) HTML <header> :- This tag element is used to create header section of web pages. the header contains introductory content , heading element logo for webpage.
- 2) HTML <nav> :- This tag element is a container for the main block of navigation links . it can contain links for the same page & for other pages .
- 3) HTML <section> :- This tag element represent a separate section of a web page which contains related elements grouped together . it can contain text,images, tables, videos etc .
- 4) HTML <article> :- This tag element is used to self-contained article such as big story, huge article etc .
- 5) HTML <aside> :- This tag element define aside content related to primary content . the <aside> content must be related to the primary content . it can function as aside bar for the main content of web page .
- 6) HTML <footer> :- This tag element define the footer for that document or webpage . it mostly contains information about author, copyright , other links etc .

7) HTML <details>:- This tag element is used to add extra details about the webpage & user can hide or show the details as per requirement.

8) HTML <summary>:- This tag element is used with <details> tag in webpage . It is used as summary, captions about the content .

E.g. <!DOCTYPE Html>
 <html>
 <head>
 <title> Webpage </title>
 </head>
 <body>
 <header style="background:#fff; height:100px; width:100%;>
 <h1> welcome to my webpage </h1>
 </header>
 <nav>
 <h2 style="text-align:center;"> Navigation
 links </h2>

 link1
 link2
 link3

 </nav>

`<section style = "background-color: red;
width: 100%; border: 1px solid white;">
<h2> intro to my HTML </h2>
<p> HTML is a frontend scripting
and coding language. the abbreviation of
HTML is Hypertext markup language (HTML)</p>
</section>.`

`<article>
<h2> This my article </h2>
<p> write your content for this page </p>
</article>`

`<aside style = "background-color: blue;">
<h2> sidebar section </h2>
</aside>`

`<#footer> style = "background-color: green;">
width: 100%; text-align: center;</#footer>
<h3> footer section </h3>
<p> © copyrights 2022-2023 </p>
</#footer>
<details>
<summary> This is my webpage </summary>
<p> are you like it </p>
</details>.`

P.6) How to optimize website assets loading.
Ans.

optimizing your website means improving its performance across various areas, including traffic, conversion rates & usability. despite meeting too many kpi's & goals, the goal of website optimization is to ensure a seamless digital journey on your website.

* Benefits of web optimization :-

1) An increase in traffic :- increasing traffic to your website through web optimization is one of the most effective ways to do so. the majority of your website's impressions & clicks will come from the search engines. if its results are at the top. hence, your website will receive a lot of traffic you are the top.

2) Enhance user experience :- whenever you are optimizing your website, you should remember that it is crucial to optimize it for the best results. making your website easy to navigate and friendly should be one of your top priority. if your website E-commerce then today's deals, new products should display on landing page.

- 5) Effortless marketing :- Since web optimization targets people searching online for products & services, it costs less compared to other marketing strategies. A website gets more qualified traffic through search engine optimization.
- 6) provide what search engine wants :- The "Spiders" are such search engine programs that automatically scan your site. The "Spiders" typically comb through the code of your websites.

* Tips to improve web optimization :-

- 1) Prefetching should be enabled :- Prefetching allows a browser to store the information a user likely to access soon. A user can see the necessary information instantly on their screen after clicking on link that has already stored in browser's cache.
- 2) Provide fast loading times! - Have you visited a website that took forever to load in browser? There are some websites which take more time because its size of website is big & it contains heavy graphics & images, videos etc. Showing spinner instead make it animation & it good to show skeleton of the card or frames.

- 3) Scripts should be placed at bottom & style should be placed at top :- It is beneficial to place style sheets in the head of website as it helps it load faster as render web pages incrementally. A majority of internet browsers allows for simultaneous downloading of two components at once. Parallel download are usually blocked by scripts, which means you can not download anything until the script has finished downloading.
- 4) Asynchronous Scripts :- Asynchronous loading of scripts will also optimize the loading speed of webpage since scripts are independent of webpage loading. The page can now be rendered faster since scripts no longer need to load before rendering.

Q. 7) what are the various formatting tags in HTML

Ans.

The HTML formatting is process of formatting text for better look & feel. HTML provides us ability to format text without using CSS. There are many formatting tags are available in HTML.

These tags are used to text bold, italic, underline etc. There are 14 options available in HTML for text appears.

1) Bold text :- HTML **and **formatting elements. The HTML **element is a physical tag which display text in bold.******

E.g. `<html>
<head>
<title> bold </title>
</head>
<body>
<p> My name is Nazeesh </p>
</body>
</html>`

2) Italic text :- HTML *and *formatting elements. The HTML *element is physical tag which display text in italic.***

E.g. `<html>
<head>
<title> Italic </title>
</head>
<body>
<p><i> my name is Nazeesh </i> I am from Nanded, Maharashtra
</p>
</body>
</html>`

3) Underline text :- HTML formatting element. The HTML element is physical tag which display text in underline.

E.g. `<html>
<head>
<title> underline </title>
</head>
<body>
<p><u> Hello my name is Nazeesh </u></p>
</body>
</html>`

4) HTML masked formatting :- If you want to mask or highlight a text, you should write the content within `<mark> --- </mark>`

E.g:- <html>
<head>
<title> MarksFormat </title>
</head>
<body>
<p> Hello my name is Nagash,
& I am from <mark> Nanded </mark>
</p>
</body>
</html>

Q.8) Explain how to indicate the character set being used by a document in HTML.

Ans-

For a browser to understand the set of characters used in HTML document, various HTML element encoding character set representations has been used like ASCII, ISO-8859-1, UTF-8 etc.

The character set being used by an HTML document is indicated using the charset attribute of a <meta> tag inside the <head> element of HTML.

In HTML-5, developers were encouraged to use the UTF-8 character set, which covers all characters & symbols. therefore, UTF-8 is the default character set for HTML-5.

E.g:- <meta charset="UTF-8">
UTF stands for Unicode Transformation Format, where '8' in UTF-8 means it uses 8-bit blocks to represent characters.

* ways to indicate character set :- character set is also abbreviated as charset, which is an attribute in an HTML document to tell the character coding.

Character set in HTML document can be represented by following two ways or.

Q) Using Meta tag :- The meta tag is used to specify the character encoding in an HTML document. the meta tag defines meta-data about HTML document which is not displayed on the webpage.

```
<head>
  <meta charset="UTF-8">
</head>
```

2) using Script tag :- The script tag is used to specify the character encoding in an external file. the script tag defines client-side Script. the script tag refers to an external Script file through the "src" attribute.

```
<script src="Script.js" charset="UTF-8">
</script>
```

E.g:-

```
<!DOCTYPE html>
<html>
  <head>
    <title> charset </title>
    <meta charset="UTF-8">
  </head>
  <body>
    <h2> Welcome To my Site </h2>
  </body>
</html>
```

P.9) In how many ways can be position an HTML element.

Ans:-

The position property specifies the type of position method used for an element (static, fixed, absolute or sticky).

Elements are then position using the top, bottom, left & right properties. however these properties will not work unless the position property is set first. they also work differently depending on the position value.

i) position: static;

HTML elements are positioned static by default. Static positioned elements are not affected by the top, bottom, left & right properties. An element with position: static; is not positioned in any special way; it is always positioned according to the normal flow of the page.

E.g:- <div> element has position: static;

```
div{ static {
```

```
position: static;
border: 3px solid orange;
```

}

2) position: relative;

An element with position: relative; is positioned relative to its normal position. Setting the top, right, bottom, and left properties of relatively-positioned element will cause it to be adjusted away from its normal position. Other content will not be adjusted to fit into any gap left by the element.

E.g.: - <div> element has position: relative</div>
div.relative {

position: relative;

left: 30px;

border: 3px solid #73AD21;

}

3) position: fixed;

An element with position: fixed; is positioned relative to the viewport, which means it always stays in the same place even if the page is scrolled. The top, right, bottom, and left properties are used to position the element.

E.g. div.fixed {

position: fixed;

bottom: 0;

right: 0;

width: 300px;

border: 3px solid #73AD21;

}

4) position: absolute;

An element with position: absolute; is positioned relative to the nearest positioned ancestor. Absolute positioned elements are removed from the normal flow, and can overlap elements.

E.g.: - <div> element has position: relative;</div>
<div> element has position: absolute;</div>
div.relative {

position: relative;

width: 400px;

height: 200px;

border: 3px solid #73AD21;

}

div.absolute {

position: absolute;

top: 80px;

right: 0;

width: 200px;

height: 100px;

border: 2px solid #fff;

}

5) position: sticky;

An element with position: sticky; is positioned based on the user's scroll position.

A sticky element toggles between relative & fixed, depending on the scroll position.

In this example, the sticky element stick to the top of the page, when you reach its scroll position.

Eg: div.stick {

position: sticky;

top: 0;

border: 2px solid red;

}

Q.10) In how many ways you can display HTML elements.

Ans - In HTML there are three ways to display elements. These elements are inline, inline-block & block.

1) inline :- Displays an element as an inline element. Any height and width properties will have no effect. The display property specifies the display behaviour of an element. It's a must know for controlling layout of an element.

This one displays the element inline or on the same line. In other words, inline elements do not start on new line & only takes up as much width as its content.

Eg: .inline-element {

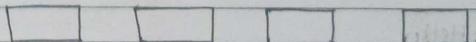
display: inline;

}

Here are few elements that have a default inline property.

1) span 2) a 3) img.

Output: display: inline;



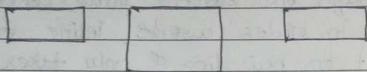
2) inline-block :- Displays an element as an inline-level block container. You can set height & width values.

Ans:- Height, it's essentially the same thing as inline, except that you can set height & width value.

E.g:- `inline-block-element {`

```
display: inline-block;  
width: 200px;  
height: 200px;
```

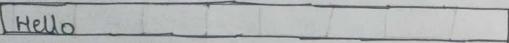
Output:- inline-block



3) block :- block starts on new line & takes up the full width available. So that means block elements will occupy the entire width of its parent element.

E.g:- `block {
display: block;
}`

Output:- display block.



Q.11) How to specify the link in HTML and explain the target attribute.

Ans.

Links are found in nearly all web-pages. Links allows users to click their way from page to page.

HTML links are hyperlinks. You can click on a link & jump to another document. When you move the mouse over a link, the mouse cursor will turn into a little hand.

* HTML Links - Syntax :- The HTML `a` tag defines a hyperlink. It has the following syntax.

` Link text `

The most important attribute of the `a` element is the `href` attribute, which indicates the link's destination. The link text is the part that will be visible to the reader. Clicking on the link text, will send the reader to the specific URL address.

E.g:- ` Visit our search engine `

* By default, links will appear as follows in all browsers.

- 1) An unvisited link is underlined & blue.
- 2) A visited link is underlined & purple.
- 3) An active link is underlined & red.

* The target attributes of HTML links:-

By default, the linked page will be displayed in current browser window. To change this, you must specify another target for the link.

The target attribute specifies where to open the linked document.

The target attribute can have one of the following values:

- 1) -self :- Default. open the document in the same window/tab as it was clicked.
- 2) -blank :- opens the document in new window or tab.
- 3) -parent :- opens the document in the parent frame.
- 4) -top :- opens the document in the full body of the window.

E.g:- ` Visit `

Q.12) How to include Javascript code in HTML.

Ans:-

In HTML, Javascript code is inserted between `<script>` & `</script>` tags.

E.g:- `<script>`

```
document.getElementById("demo").innerHTML  
= "My first Javascript";  
</script>
```

You can place any number of scripts in an HTML document. Scripts can be placed in the `<body>`, or in the `<head>` section of an HTML page, or in both.

i) E.g:- A Javascript function is placed in the `<head>` section of an HTML page. The function is invoked (called) when a button is clicked.

```
<!DOCTYPE html> <html>  
<head>  
<script>  
function myFunction(){  
document.getElementById("demo").innerHTML  
= "Hello world";}  
</script>  
<head>  
<body>  
<p id="demo"> welcome to world </p>  
<button type="button" onclick="myFunction()">  
Try it </button>  
</body>  
</html>.
```

- 2) A javascript function is placed in the `<body>` section of an HTML page. The function is invoked (called) when a button is clicked.

```
E.g. <!DOCTYPE html>
<html>
<head>
<body>
  <p id="demo"> Paragraph </p>
  <button type="button" onclick="myFunction()">
    Try it </button>
<script>
  function myFunction() {
    document.getElementById("demo").innerHTML = "paragraph changed";
  }
</script>
</body>
</html>
```

- 3) External Javascript in scripts can also be placed in external files.

External file : main.js

```
function myFunction() {
  document.getElementById("demo").innerHTML =
  " para change";}
```

External scripts are practical when the same code is used in many different webpages. Javascripts files have the file extensions `.js`. To use an external script, put the name of the script file in the `src (source)` attribute of a `<script>` tag.

E.g. :- `<script src="main.js"></script>`
You can place an external script reference in `<head>` or `<body>` as you like.

Q. 13) when do we use scripts in the head & when to use script in the body.

Ans:-

Scripts can be placed inside the body or the head section of an HTML page or inside both the head & body. We can also place JavaScript outside the HTML file which can be linked by specifying its source in the Script tag.

* Javascript in head :- A Javascript function is placed inside the head section of an HTML page & the function is invoked when a button is clicked.

E.g. `<html>`
`<head>`
`<script>`
`function gfg() {`
`document.getElementById("demo").`
`innerHTML = "Geeks for Geeks";}`
`</script>`
`</head>`
`<body>`
`<p id="demo"> geeksforgeeks </p>`
`<button type="button" onclick="gfg()">`
Tey. `</button>`
`</body>`
`</html>`

* Javascript in body :- A Javascript function is placed inside the body section of an HTML page & the function is invoked when a button is clicked.

E.g:- `<html>`
`<head> <head>`
`<body>`
`<p id="demo"> Nazeesh. </p>`
`<button type="button" onclick="gfg()">`
Teyit `</button>`
`<script>`
`function gfg() {`
`document.getElementById("demo").`
`innerHTML = "Nazeesh vilas Kanade"`
g.
`</script>`
`</body>`
`</html>`

* External Javascript :- Javascript can also be used as external files. Javascript files have file extension .JS. To use an external script put the file name of the script file in the SRC attribute of Script tag. External Scripts cannot contain Script tags.

E.g:- main.js.
`function gfg() {`
`document.getElementById("demo").`
`innerHTML = "paragraph changed"`
}.

```

ideexhtml :- <html>
<head>
</head>
<body>
<p id="demo"> gfg </p>
<button type="button" onclick =
    "gfg()>
    Try it </button>
<script>sec = "main.js"</script>
</body>
</html>

```

* Advantages of external Javascript:-

- 1) Cached Javascript files can speed up page loading.
- 2) It makes Javascript & HTML easier to read & maintain.
- 3) It separates the HTML & Javascript Code.

Q.14) what are forms & how to create forms in HTML.

Ans:-

An HTML form is a section of document which contains contents such as textfields, password fields, checkboxes, radio buttons & submit button, menus etc.

An HTML form facilitates the user to enter data that is to be sent to the server for processing such as name, email address, password, phone number etc.

* HTML form Syntax:-

```

<form action = "server.php" method = "getpost">
    " input contents eg. textfield, textarea,
    radio button.
</form>

```

* HTML <input> element :- The HTML <input> element is fundamental form element. It is used to create form fields, to take input from user.

E.g:- <body>
 <form>

```

        Enter your name <input>
        <input type = "textfield" name = "user">
        </form>
    </body>

```

- * HTML TextArea in form :- The textarea tag in HTML is used to insert multiple-line text in a form. The size of textarea can be specify either using "rows" or "cols" attributes or by CSS.

E.g:- <body>

<form>

Enter your address

<textarea rows="2" cols="20">

<textarea>

</form>

</body>

- * Radio button in an HTML form :- To create a radio button, we use <input> tag following by type="radio" to provide users to choose a limited numbers of classes.

Syntax:- <input type="radio" name="radio" value="radio"/>

E.g:- <head>

<title> Eradio </title>

</head>

<body> <form>

<label> Male <input type="radio"

name="gender" value="male"/> <label>

<label> Female <input type="radio"

name="gender" value="female"/>

</label>

</form>

</body>