### What is HTML?

### https://s3.ap-south-1.amazonaws.com/myinterviewtrainer-domestic/public_assets/assets/000/000/585/original/html-intro.png?1625063534

HTML stands for Hyper Text Markup Language. It is a language of World Wide Web. It is a standard text formatting language which is used to create and display pages on the Web. It makes the text more interactive and dynamic. It can turn text into images, tables, links.

**2. What are Tags?**

HTML tags are composed of three things: an opening tag, content and ending tag. Some tags are unclosed tags.

HTML documents contain two things:

* content, and
* tags

When a web browser reads an HTML document, the 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.

#### Syntax

1. **<tag>** content **</tag>**

Content is placed between tags to display data on the web page.

Tags are the primary component of the HTML that defines how the content will be structured/ formatted,

**3. What are attributes in HTML?**

Attributes are used along with the HTML tags to define the characteristics of the element. For example, <p align=” center”>Interview questions</p>, in this the ‘align’ is the attribute using which we will align the paragraph to show in the center of the view.

**4. What are void elements in HTML?**

HTML elements which do not have closing tags or do not need to be closed are Void elements. For Example <br />, <img />, <hr />, etc.

**5. What is the advantage of collapsing white space?**

In HTML, a blank sequence of whitespace characters is treated as a single space character, Because the browser collapses multiple spaces into a single space character and this helps a developer to indent lines of text without worrying about multiple spaces and maintain readability and understandability of HTML codes.

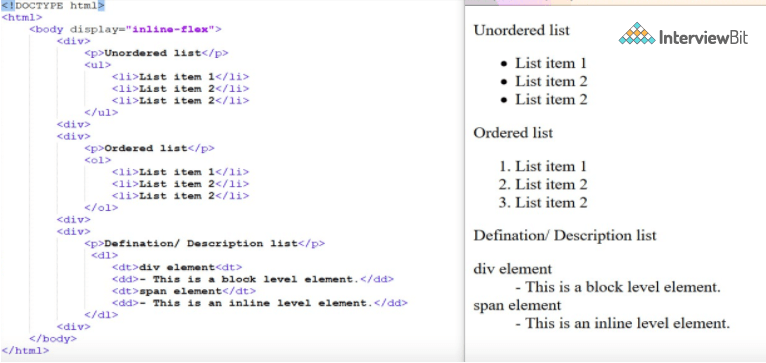
**6. What are HTML Entities?**

In HTML some characters are reserved like ‘<’, ‘>’, ‘/’, etc. To use these characters in our webpage we need to use the character entities called HTML Entities. Below are a few mapping between the reserved character and its respective entity character to be used.

| **Character** | **Entity Name** | **Entity Number** |
| --- | --- | --- |
| < | &lt; | &#60; |
| > | &gt; | &#62; |
| & | &amp; | &#38; |
| (non-breaking space) Eg. 10  PM | &nbsp; Eg. <p>10&nbsp&nbspPM</p> | &#160; |

The HTML character entities are used as a replacement for 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.

**7. What are different types of lists in HTML?**

****

There are many common lists which are used to design a page. You can choose any or a combination of the following list types:

* Ordered list - The ordered list displays elements in numbered format. It is represented by <ol> tag.
* Unordered list - The unordered list displays elements in bulleted format. It is represented by <ul> tag.
* Definition list - The definition list displays elements in definition form like in dictionary. The <dl>, <dt> and <dd> tags are used to define description list.

### 8. How do you keep list elements straight in an HTML file?

You can keep the list elements straight by using indents.

**9. What is the ‘class’ attribute in HTML?**

The class attribute is used to specify the class name for an HTML element. Multiple elements in HTML can have the same class value. Also, it is mainly used to associate the styles written in the stylesheet with the HTML elements.

**10. What is the difference between the ‘id’ attribute and the ‘class’ attribute of HTML elements?**

Multiple elements in HTML can have the same class value, whereas a value of id attribute of one element cannot be associated with another HTML element.

### 11. What is the difference between HTML elements and tags?

HTML elements communicate to the browser to render text. When the elements are enclosed by brackets <>, they form HTML tags. Most of the time, tags come in a pair and surround content.

HTML elements are defined by a starting tag, may contain some content and a closing tag.For example, <h1>Heading 1</h1> is a HTML element but just <h1> is a starting tag and </h1> is a closing tag.

All HTML tags don’t have an end tag because, There are some HTML tags that don't need a closing tag. For example: <image> tag, <br> tag.

**12. Define multipart form data?**

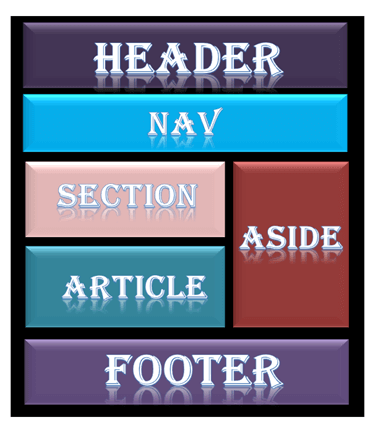
Multipart form data is one of the values of the enctype attribute. It is used to send the file data to the server-side for processing. The other valid values of the enctype attribute are text/plain and application/x-www-form-urlencoded.

**13. Describe HTML layout structure.**

Every web page has different components to display the intended content and a specific UI. But still, there are few things which are templated and are globally accepted way to structure the web page, such as:

* **<header>**: Stores the starting information about the web page.
* **<footer>**: Represents the last section of the page.
* **<nav>**: The navigation menu of the HTML page.
* **<article>**: It is a set of information.
* **<section>**: It is used inside the article block to define the basic structure of a page.
* **<aside>**: Sidebar content of the page.

HTML layout specifies a way in which the web page is arranged.

  
Every website has a specific layout to display content in a specific manner.

Following are different HTML5 elements which are used to define the different parts of a webpage.

* <header>: It is used to define a header for a document or a section.
* <nav>: It is used to define a container for navigation links
* <section>: It is used to define a section in a document
* <article>: It is used to define an independent, self-contained article
* <aside>: It is used to define content aside from the content (like a sidebar)
* <footer>: It is used to define a footer for a document or a section

**14. How to optimize website assets loading?**

To optimize website load time we need to optimize its asset loading and for that:

* **CDN hosting** - A CDN or content delivery network is geographically distributed servers to help reduce latency.
* **File compression** - This is a method that helps to reduce the size of an asset to reduce the data transfer
* **File concatenation** - This reduces the number of HTTP calls
* **Minify scripts** - This reduces the overall file size of js and CSS files
* **Parallel downloads** - Hosting assets in multiple subdomains can help to bypass the download limit of 6 assets per domain of all modern browsers. This can be configured but most general users never modify these settings.
* **Lazy Loading** - Instead of loading all the assets at once, the non-critical assets can be loaded on a need basis.

**15. What are the various formatting tags in HTML?**

 HTML has various formatting tags:

* **<b>** - makes text bold
* **<i>** - makes text italic
* **<em>** - makes text italic but with added semantics importance
* **<big>** - increases the font size of the text by one unit
* **<small>** - decreases the font size of the text by one unit
* **<sub>** - makes the text a subscript
* **<sup>** - makes the text a superscript
* **<del>** - displays as strike out text
* **<strong>** - marks the text as important
* **<mark>** - highlights the text
* **<ins>** - displays as added text

The HTML formatting is a process of format the text for a better look and feel. It uses different tags to make text bold, italicized, underlined.

**16. What are the different kinds of Doctypes available?**

The three kinds of Doctypes which are available:

* Strict Doctype
* Transitional Doctype
* Frameset Doctype

No, the <!DOCTYPE html> declaration is not an HTML tag. There are many type of HTML e.g. HTML 4.01 Strict, HTML 4.01 Transitional, HTML 4.01 Frameset, XHTML 1.0 Strict, XHTML 1.0 Transitional, XHTML 1.0 Frameset, XHTML 1.1 etc. So, <!DOCTYPE html> is used to instruct the web browser about the HTML page.

### If I do not put <!DOCTYPE html> will HTML 5 work - No, because the browser will not be able to identify that it is an HTML document and HTML 5 tags do not function properly.

**17. Explain how to indicate the character set being used by a document in HTML?**

The character set is defined in <meta> tag inside <head> element.

<!DOCTYPE **html**>

<**html**>

<**head**>

<**meta** charset="UTF-8">

...

...

</**head**>

...

</**html**>

**18. What is the difference between <strong>, <b> tags and <em>, <i> tags?**

The effect on a normal webpage of the tags <strong>, <b>  and <em>, <i> is the same. <b> and <i> tags stands for bold and italic. These two tags only apply font styling and bold tag <b>, just adds more ink to the text, these tags don't say anything about the text.

Whereas, <strong> and <em> tags represent that the span of text is of strong importance or more importance and emphatic stress respectively than the rest of the text. These tags have semantic meaning.

**19. What is the significance of <head> and <body> tag in HTML?**

<head> tag provides the information about the document. It should always be enclosed in the <html> tag. This tag contains the metadata about the webpage and the tags which are enclosed by head tag like <link>, <meta>, <style>, <script>, etc. are not displayed on the web page. Also, there can be only 1 <head> tag in the entire Html document and will always be before the <body> tag.

<body> tag defines the body of the HTML document. It should always be enclosed in the <html> tag. All the contents which needs to be displayed on the web page like images, text, audio, video, contents, using elements like <p>, <img>, <audio>, <heading>, <video>, <div>, etc. will always be enclosed by the <body> tag. Also, there can be only 1 body element in an HTML document and will always be after the <head> tag.

### 20. What is the use of an iframe tag?

An iframe is used to display a web page within a web page.

**Syntax:**

1. **<iframe** src="URL"**></iframe>**

**Example:**

1. **<iframe** src="demo\_iframe.html" width="200px" height="200px"**></iframe>**

**Target to a link:**

1. **<iframe** src="http://www.javatpoint.com" name="iframe\_a"**></iframe>**

### 21. How to create a nested webpage in HTML?

The HTML iframe tag is used to display a nested webpage. In other words, it represents a webpage within a webpage. The HTML <iframe> tag defines an inline frame. For example:

1. <!DOCTYPE html**>**
2. **<html>**
3. **<body>**
4. **<h2>**HTML Iframes example**</h2>**
5. **<p>**Use the height and width attributes to specify the size of the iframe:**</p>**
6. **<iframe** src="https://www.javatpoint.com/" height="300" width="400"**></iframe>**
7. **</body>**
8. **</html>**

Yes, we can display a web page inside another HTML web page. HTML provides a tag <iframe> using which we can achieve this functionality.

<**iframe** src=”url of the web page to embed” />

**22. How is Cell Padding different from Cell Spacing?**

Cell Spacing is the space or gap between two consecutive cells. Whereas, Cell Padding is the space or gap between the text/ content of the cell and the edge/ border of the cell. Please refer to the above figure example to find the difference.

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

HTML provides two table attributes “rowspan” and “colspan” to make a cell span to multiple rows and columns respectively.

**24. In how many ways can we position an HTML element? Or what are the permissible values of the position attribute?**

There are mainly 7 values of position attribute that can be used to position an HTML element:

1. **static**: Default value. Here the element is positioned according to the normal flow of the document.
2. **absolute**: Here the element is positioned relative to its parent element. The final position is determined by the values of left, right, top, bottom.
3. **fixed**: This is similar to absolute except here the elements are positioned relative to the <html> element.
4. **relative**: Here the element is positioned according to the normal flow of the document and positioned relative to its original/ normal position.
5. **initial**: This resets the property to its default value.
6. **inherit**: Here the element inherits or takes the property of its parent.

**25. In how many ways you can display HTML elements?**

1. **inline**: Using this we can display any block-level element as an inline element. The height and width attribute values of the element will not affect.
2. **block**: using this, we can display any inline element as a block-level element.
3. **inline-block**: This property is similar to inline, except by using the display as inline-block, we can actually format the element using height and width values.
4. **flex**: It displays the container and element as a flexible structure. It follows flexbox property.
5. **inline-flex**: It displays the flex container as an inline element while its content follows the flexbox properties.
6. **grid**: It displays the HTML elements as a grid container.
7. **none**: Using this property we can hide the HTML element.

Below are some of the display types which are rarely used:

1. table
2. inline-table
3. table-cell
4. table-column
5. table-row
6. inline-grid
7. list-item
8. inherit
9. initial
10. table-caption

Yes, it is possible using the “display” property with its value as “block”, to change the inline element into a block-level element.

**26. What is the difference between “display: none” and “visibility: hidden”, when used as attributes to the HTML element.**

When we use the attribute “visibility: hidden” for an HTML element then that element will be hidden from the webpage but still takes up space. Whereas, if we use the “display: none” attribute for an HTML element then the element will be hidden, and also it won’t take up any space on the webpage.

**27. How to specify the link in HTML and explain the target attribute?**

HTML provides a hyperlink - <a> tag to specify the links in a webpage. The ‘href’ attribute is used to specify the link and the ‘target’ attribute is used to specify, where do we want to open the linked document. The ‘target’ attribute can have the following values:

1. **\_self**: This is a default value. It opens the document in the same window or tab as it was clicked.
2. **\_blank**: It opens the document in a new window or tab.
3. **\_parent**: It opens the document in a parent frame.
4. **\_top**: It opens the document in a full-body window.

### 28. What is the use of the required attribute in HTML5?

It forces a user to fill text on the text field or text area before submitting the form. It is used for form validation.

**Example:**

1. Name: **<input** type="text" name="name" required**>**

### 29. How to create a hyperlink in HTML?

The HTML provides an anchor tag to create a hyperlink that links one page to another page. These tags can appear in any of the following ways:

* Unvisited link - It is displayed, underlined and blue.
* Visited link - It is displayed, underlined and purple.
* Active link - It is displayed, underlined and red.

**30. Difference between link tag <link> and anchor tag <a>?**

The anchor tag <a> is used to create a hyperlink to another webpage or to a certain part of the webpage and these links are clickable, whereas, link tag <link> defines a link between a document and an external resource and these are not clickable.

**31. How to include javascript code in HTML?**

HTML provides a <script> tag using which we can run the javascript code and make our HTML page more dynamic.

<!DOCTYPE **html**>

<**html**>

<**body**>

<**h1**>

<**span**>This is a demo for </**span**>

<**u**><**span** id="demo"></**span**></**u**>

</**h1**>

<**script**>

document.getElementById("demo").innerHTML = "script Tag"

</**script**>

</**body**>

</**html**>

**32. When to use scripts in the head and when to use scripts in the body?**

If the scripts contain some event-triggered functions or jquery library then we should use them in the head section. If the script writes the content on the page or is not inside a function then it should be placed inside the body section at the bottom. In short, follow below three points:

1. Place library scripts or event scripts in the head section.
2. Place normal scripts that do not write anything on the page, in the head section until there is any performance issue.
3. Place scripts that render something on the web page at the bottom of the body section.

**33. What are forms and how to create forms in HTML?**

The HTML form is used to collect the user inputs. HTML provides a <form> tag to create forms. To take input from the user we use the <input> tag inside the form so that all collected user data can be sent to the server for processing. There are different input types like ‘button’, ‘checkbox’, ‘number’, ‘text’, ‘password’, ‘submit’ etc.

<**form** action="/submit\_data.php">

<**label**>Enter your name: </**label**>

<**input** type="text" name="name" />

<**label**>Enter Mobile number </**label**>

<**input** type="number" name="mobile\_no"/>

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

</**form**>

**34. How to handle events in HTML?**

HTML allows event trigger actions in browsers using javascript or JQuery. There are a lot of events like ‘onclick’, ‘ondrag’, ‘onchange’, etc.

<!DOCTYPE **html**>

<**html**>

<**body** style="padding-top:50px">

<**h3** id="event\_demo">0</**h3**>

<**input** type="button" onclick="myFunction()" value="Click Me" />

<**input** type="reset" onclick="reset()" value="Reset" />

</**body**>

<**script**>

**function** **myFunction**() {

**var** value = document.getElementById("event\_demo").innerHTML

value = parseInt(value) + 1;

document.getElementById("event\_demo").innerHTML = value;

}

**function** **reset**() {

document.getElementById("event\_demo").innerHTML = 0;

}

</**script**>

</**html**>

**35. What are some of the advantages of HTML5 over its previous versions?**

Some advantages of HTML5 are:-

* It has Multimedia Support.
* It has the capabilities to store offline data using SQL databases and application cache.
* Javascript can be run in the background.
* HTML5 also allows users to draw various shapes like rectangles, circles, triangles, etc.
* Included new Semantic tags and form control tags.

These were the target area of the HTML5 specs:

* Introduction of new element tags to better structure the web page such as <header> tag.
* Forming a standard in cross-browser behavior and support for different devices and platforms
* Backward compatible with the older version HTML web pages
* Introduction of basic interactive elements without the dependency of plugins such as <video> tag instead of the flash plugin.

**36. How can we include audio or video in a webpage?**

HTML5 provides two tags: <audio> and <video> tags using which we can add the audio or video directly in the webpage.

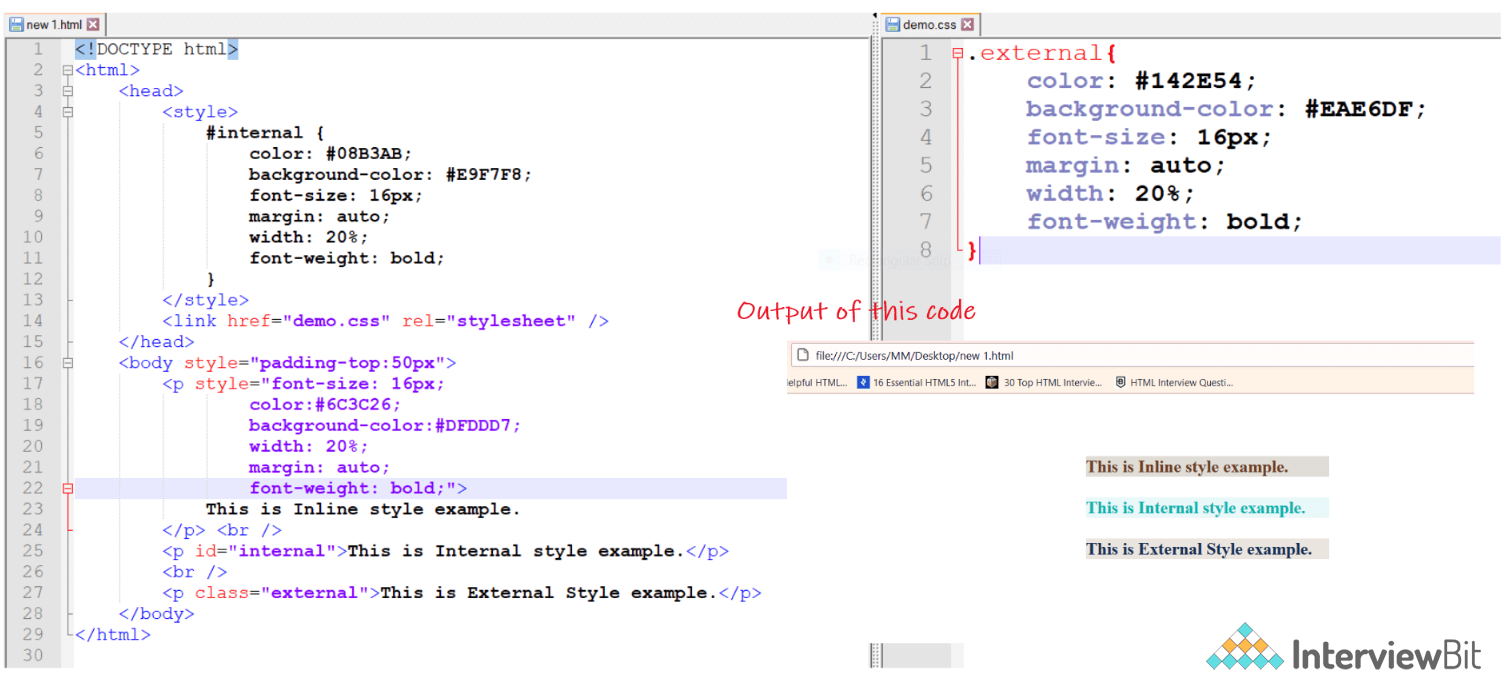
HTML 5 supports three types of video format:

* mp4
* WebM
* Ogg

**37. In how many ways can we specify the CSS styles for the HTML element?**

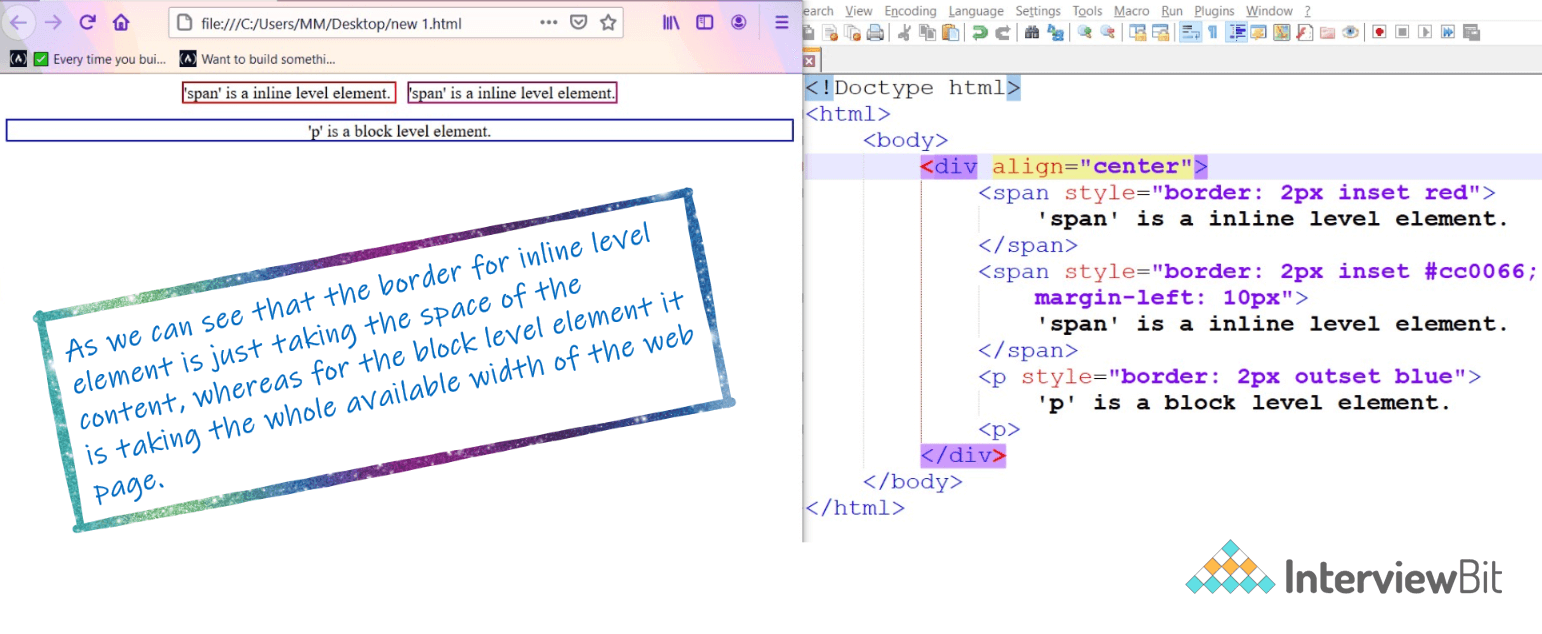
There are three ways in which we can specify the styles for HTML elements:

* **Inline**: Here we use the ‘style’ attribute inside the HTML element.
* **Internal:** Here we use the <style> tag inside the <head> tag. To apply the style we bind the elements using ‘id’ or ‘class’ attributes.
* **External**: Here we use the <link> tag inside <head> tag to reference the CSS file into our HTML code. Again the binding between elements and styles is done using ‘id’ or ‘class’ attributes.



**38. Inline and block elements in HTML5?**

| **Inline** | **Block** |
| --- | --- |
| Inline elements just take up the space that is absolutely necessary for the content and does not start from a new line. Example:- <span>, <a>, <strong>, <img>, <button>, <em>, <select>, <abbr>, <label>, <sub>, <cite>, <abbr>, <script>, <label>, <i>, <input>, <output>, <q>, etc. | Block elements start on a new line and consume the full width of the page available. Example:- <div>, <p>, <header>, <footer>, <h1>...<h6>, <form>, <table>, <canvas>, <video>, <blockquote>, <pre>, <ul>, <ol>, <figcaption>, <figure>, <hr>, <article>, <section>, etc. |



**39. What is the use of figure tag in HTML 5?**

The figure tag is used to add a photo in the document on the web page. It is used to handle the group of diagrams, photos, code listing with some embedded content.

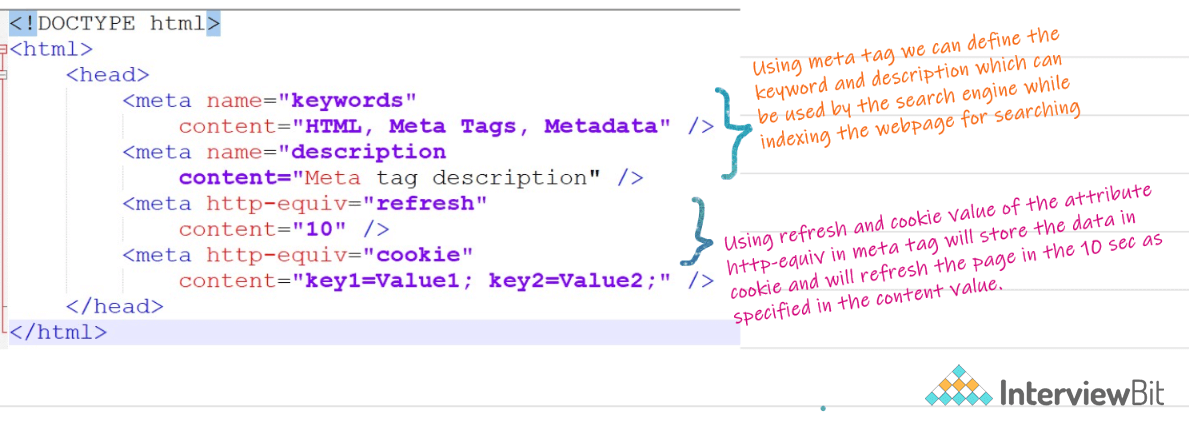
1. **<p>**The Taj Mahal is widely recognized as "the jewel of Muslim art in India and one of the universally admired masterpieces of the world's heritage."**</p>**
2. **<figure>**
3. **<img** src="htmlpages/images/tajmahal.jpg" alt="Taj Mahal"**/>**
4. **</figure>**

**34. What is the difference between <figure> tag and <img> tag?**

The <figure> tag specifies the self-contained content, like diagrams, images, code snippets, etc. <figure> tag is used to semantically organize the contents of an image like image, image caption, etc., whereas the <img> tag is used to embed the picture in the HTML5 document.

**40. How to specify the metadata in HTML5?**

To specify we can use <meta> tag which is a void tag,i.e., it does not have a closing tag. Some of the attributes used with meta tags are name, content, http-equiv, etc. The below image tells how to specify the metadata.



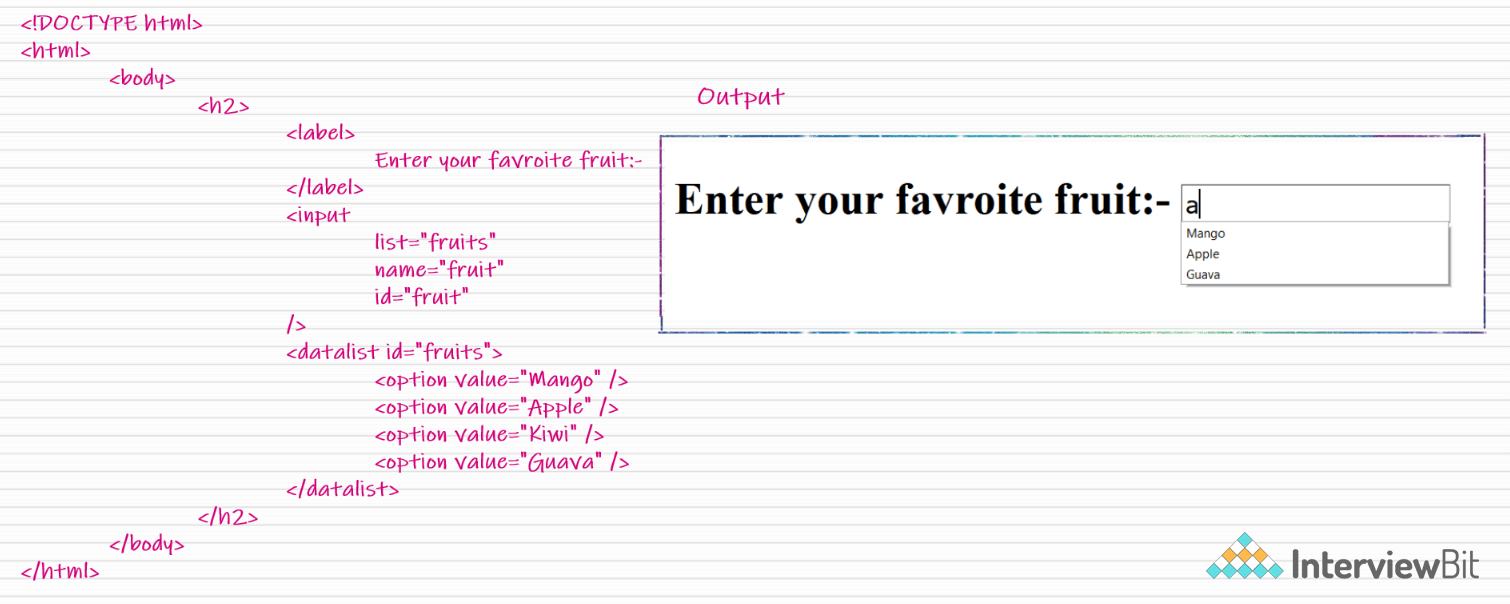
### 41. What is datalist tag?

The HTML 5 datalist tag provides an autocomplete feature on the form element. It facilitates users to choose the predefined options to the users to select data.

1. **<label>**
2. Enter your favorite cricket player: Press any character**<br** **/>**
3. **<input** type="text" id="favCktPlayer" list="CktPlayers"**>**
4. **<datalist** id="CktPlayers"**>**
5. **<option** value="Sachin Tendulkar"**>**
6. **<option** value="Brian Lara"**>**
7. **<option** value="Jacques Kallis"**>**
8. **<option** value="Ricky Ponting"**>**
9. **<option** value="Rahul Dravid"**>**
10. **<option** value="Shane Warne"**>**
11. **<option** value="Rohit Sharma"**>**
12. **<option** value="Donald Bradman"**>**
13. **<option** value="Saurav Ganguly "**>**
14. **<option** value="AB diVilliers"**>**
15. **<option** value="Mahendra Singh Dhoni"**>**
16. **<option** value="Adam Gilchrist"**>**
17. **</datalist>**
18. **</label>**

**42. Is the <datalist> tag and <select> tag same?**

No. The <datalist> tag and <select> tag are different. In the case of <select> tag a user will have to choose from a list of options, whereas <datalist> when used along with the <input> tag provides a suggestion that the user selects one of the options given or can enter some entirely different value.



**43. Define Image Map?**

Image Map lets a developer map/link different parts of images with the different web pages. It can be achieved by the <map> tag in HTML5, using which we can link images with clickable areas.

<**img** src=”image\_url” , usemap=”#workspace” />

<**map** name=”workspace”>

<**area** shape=”rect” coords=”34, 44, 270, 350” , href=”xyz.html” />

<**area** shape=”rect” coords=”10, 120, 250, 360” , href=”xyz.html” />

</**map**>

Image map facilitates you to link many different web pages using a single image. It is represented by <map> tag. You can define shapes in images that you want to make part of an image mapping.

### 44. What is semantic HTML?

Semantic HTML is a coding style. It is the use of HTML markup to reinforce the semantics or meaning of the content. For example: In semantic HTML <b> </b> tag is not used for bold statement as well as <i> </i> tag is used for italic. Instead of these we use <strong></strong> and <em></em> tags.

Semantic elements are those which describe the particular meaning to the browser and the developer. Elements like <form>, <table>, <article>, <figure>, etc., are semantic elements.

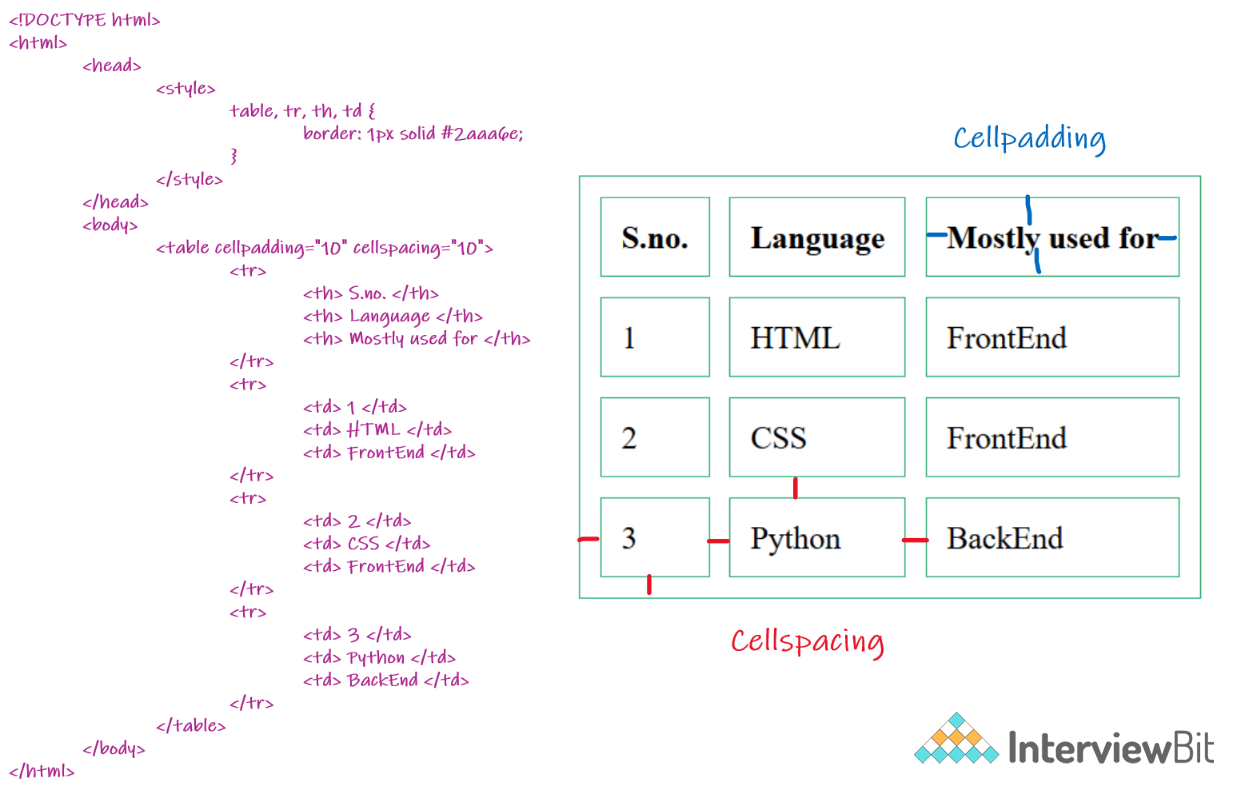
**45. Convert the below data into Tabular format in HTML5?**

S.no., Language, Mostly used for

1, HTML, FrontEnd

2, CSS, FrontEnd

3, Python, BackEnd



The **HTML table tag** is used to display data in tabular form (row \* column). It also manages the layout of the page, e.g., header section, navigation bar, body content, footer section. Here is the list of tags used while displaying the data in the tabular form:

|  |  |
| --- | --- |
| **Tag** | **Description** |
| <table> | It defines a table. |
| <tr> | It defines a row in a table. |
| <th> | It defines a header cell in a table. |
| <td> | It defines a cell in a table. |
| <caption> | It defines the table caption. |
| <colgroup> | It specifies a group of one or more columns in a table for formatting. |
| <col> | It is used with <colgroup> element to specify column properties for each column. |
| <tbody> | It is used to group the body content in a table. |
| <thead> | It is used to group the header content in a table. |
| <tfooter> | It is used to group the footer content in a table. |

**46. What is the difference between <meter> tag and <progress> tag?**

<progress> tag should be used when we want to show the completion progress of a task, whereas if we just want a scalar measurement within a known range or fraction value. Also, we can specify multiple extra attributes for <meter> tags like ‘form’, ‘low’, ‘high’, ‘min’, etc.

The progress tag is used to represent the progress of the task only while the meter tag is used to measure data within a given range.

**47. Is drag and drop possible using HTML5 and how?**

Yes, in HTML5 we can drag and drop an element. This can be achieved using the drag and drop-related events to be used with the element which we want to drag and drop.

### 48. What is SVG?

HTML SVG is used to describe the two-dimensional vector and vector/raster graphics. SVG images and their behaviors are defined in XML text files. So as XML files, you can create and edit an SVG image with the text editor. It is mostly used for vector type diagrams like pie charts, 2-Dimensional graphs in an X, Y coordinate system.

1. **<svg** width="100" height="100"**>**
2. **<circle** cx="50" cy="50" r="40" stroke="yellow" stroke-width="4" fill="red" **/>**
3. **</svg>**

[.](https://www.javatpoint.com/html-svg)

**49. How to support SVG in old browsers?**

To support old browsers instead of defining the resource of svg in src attribute of <img> tag, it should be defined in srcset attribute and in src the fallback png file should be defined.

<**img** src="circle.png" alt="circle" srcset="circle.svg">

### 50. What is the canvas element in HTML5?

The <canvas> element is a container that is used to draw graphics on the web page using scripting language like JavaScript. It allows for dynamic and scriptable rendering of 2D shapes and bitmap images. There are several methods in canvas to draw paths, boxes, circles, text and add images. For Example:

1. **<canvas** id="myCanvas1" width="300" height="100" style="border:2px solid;"**>**
2. Your browser does not support the HTML5 canvas tag.
3. **</canvas>**

### 51. Difference between SVG and Canvas HTML5 element?

| **SVG** | **Canvas** |
| --- | --- |
| SVG is a vector based i.e., composed of shapes. | It is Raster based i.e., composed of pixels. |
| SVG works better with a larger surface. | Canvas works better with a smaller surface. |
| SVG can be modified using CSS and scripts. | Canvas can only be modified using scripts. |
| SVG is highly scalable. So we can print at high quality with high resolution. | It is less scalable. |

### 52. Is audio tag supported in HTML 5?

Yes. It is used to add sound or music files on the web page. There are three supported file formats for HTML 5 audio tag.

1. mp3
2. WAV
3. Ogg

Let's see the code to play mp3 file using HTML audio tag.

1. **<audio** controls**>**
2. **<source** src="koyal.mp3" type="audio/mpeg"**>**
3. Your browser does not support the html audio tag.
4. **</audio>**

Instead of koyal.mp3, you can pass any mp3 file name.

HTML5 supports the following three types of audio file formats:

1. Mp3
2. WAV
3. Ogg

**53. Explain the concept of web storage in HTML5.**

This web storage helps in storing some of the static data in the local storage of the browser so that we do not need to fetch it from the server every time we need it. There is a size limit based on different browsers. This helps in decreasing the load time and a smooth user experience. There are two types of web storage that are used to store data locally in HTML5:

* **Local Storage** - This helps in storing data that will be retained even though the user reopens the browser. It is stored for each webapp on different browsers.
* **Session Storage** - This is used for one session only. After the user closes the browser this gets deleted.

**54. What is Microdata in HTML5?**

It is used to help extract data for site crawlers and search engines. It is basically a group of name-value pairs. The groups are called items, and each name-value pair is a property. Most of the search engines like Google, Microsoft, Yandex, etc follow schema.org vocabulary to extract this microdata.

<**div** itemscope itemtype="http://schema.org/SoftwareApplication">

<**span** itemprop="name">Interviewbit Games</**span**> -

REQUIRES <**span** itemprop="operatingSystem">ANDROID</**span**><**br**>

<**link** itemprop="applicationCategory" href="http://schema.org/GameApplication"/>

<**div** itemprop="aggregateRating" itemscope itemtype="http://schema.org/AggregateRating">

RATING:

<**span** itemprop="ratingValue">4.6</**span**> (

<**span** itemprop="ratingCount">8864</**span**> ratings )

</**div**>

<**div** itemprop="offers" itemscope itemtype="http://schema.org/Offer">

Price: Rs.<**span** itemprop="price">1.00</**span**>

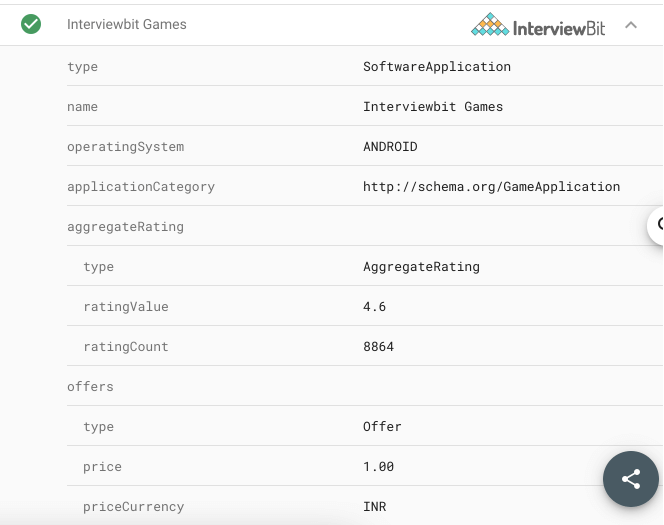
<**meta** itemprop="priceCurrency" content="INR" />

</**div**>

</**div**>

* **itemid** – The unique, global identifier of an item.
* **itemprop** – Used to add properties to an item.
* **itemref** – Provides a list of element ids with additional properties.
* **itemscope** – It defines the scope of the itemtype associated with it.
* **itemtype** – Specifies the URL of the vocabulary that will be used to define itemprop.

The above example will be parsed by Google as



**55. Which tag is used for representing the result of a calculation? Explain its attributes.**

The <output> tag is used for representing the result of a calculation. It has the following attributes:

* **for** - It defines the relationship between the elements used in calculation and result.
* **form**- This is used to define the form the output element belongs to.
* **name** - The name of the output element.

<**form** oninput = "result.value=parseInt(n1.value)+parseInt(n2.value)">

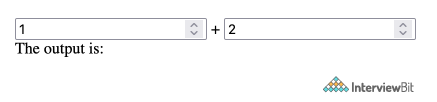
<**input** type = "number" name = "n1" value = "1" /> +

<**input** type = "number" name = "n2" value = "2" /><**br** />

The output is: <**output** name = "result"></**output**>

</**form**>

The above example looks like



**56. What is new about the relationship between the <header> and <h1> tags in HTML5?**

As HTML5 was all about better semantics and arrangements of the tags and elements, the <header> tag specifies the header section of the webpage. Unlike in previous version there was one <h1> element for the entire webpage, now this is the header for one section such as <article> or <section>. According to the HTML5 specification, each <header> element must at least have one <h1> tag.

**57. Explain HTML5 Graphics.**

HTML5 supports two kinds of graphics:

* **Canvas** - It is like drawing on a whitepaper or a blank webpage. We can add different graphic designs on web pages with available methods for drawing various geometrical shapes.

<!DOCTYPE **HTML**>

<**html**>

<**head**>

</**head**>

<**body**>

<**canvas** width="300" height="100" style="border:2px solid;"></**canvas**>

</**body**>

</**html**>

* **SVG** - Scalable Vector Graphics are used mostly for diagrams or icons. It follows the XML format.

<!DOCTYPE **html**>

<**html**>

<**body**>

<**svg** width="400" height="110">

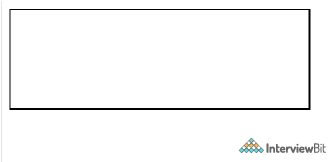
<**rect** width="300" height="100" style="fill:#FFF;stroke-width:2;stroke:#000" />

</**svg**>

</**body**>

</**html**>

Both of the above examples produce this output and represent two different approaches provided by HTML5 to implement graphical aspects in the webpage.



**58. Explain new input types provided by HTML5 for forms?**

Following are the significant new data types offered by HTML5:

* Date - Only select date by using type = "date"
* Week - Pick a week by using type = "week"
* Month - Only select month by using type = "month"
* Time - Only select time by using type = "time".
* Datetime - Combination of date and time by using type = "datetime"
* Datetime-local - Combination of  date and time by using type = "datetime-local." but ignoring the timezone
* Color - Accepts multiple colors using type = "color"
* Email - Accepts one or more email addresses using type = "email"
* Number - Accepts a numerical value with additional checks like min and max using type = "number"
* Search - Allows searching queries by inputting text using type = "search"
* Tel - Allows different phone numbers by using type = "tel"
* Placeholder - To display a short hint in the input fields before entering a value using type = "placeholder"
* Range - Accepts a numerical value within a specific range using type = "range"
* Url - Accepts a web address using type = "url”

<**form**>

<**div**>

<**label**>Date:</**label**>

<**input** type="date" id="date" />

<**br**>

<**label**>Week:</**label**>

<**input** type="week" id="week" />

<**br**>

<**label**>Month:</**label**>

<**input** type="month" id="month" />

<**br**>

<**label**>Time:</**label**>

<**input** type="time" id="time" />

<**br**>

<**label**>Datetime:</**label**>

<**input** type="datetime" id="datetime" />

<**br**>

<**label**>Datetime Local:</**label**>

<**input** type="datetime-local" id="datetime-local" />

<**br**>

<**label**>Color:</**label**>

<**input** type="color" id="color"/>

<**br**>

<**label**>Email:</**label**>

<**input** type="email" id="email" placeholder="email address" />

<**br**>

<**label**>Number:</**label**>

<**input** type="number" id="number" />

<**br**>

<**label**>Search:</**label**>

<**input** type="search" id="search" />

<**br**>

<**label**>Phone:</**label**>

<**input** type="tel" id="phone" placeholder="Phone Number" pattern="\d{10}$" />

<**br**>

<**label**>Range:</**label**>

<**input** type="range" id="range" />

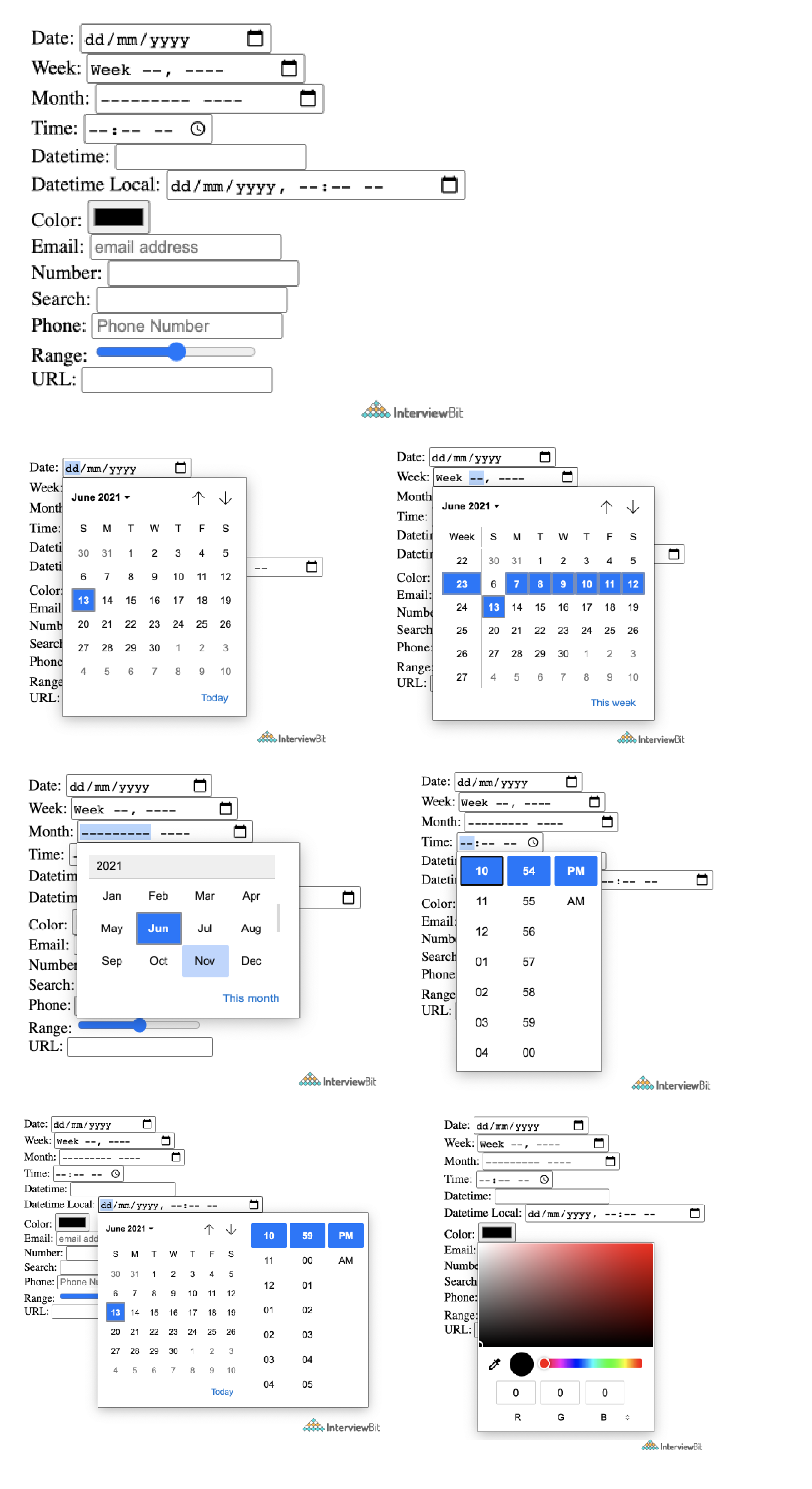
<**br**>

<**label**>URL:</**label**>

<**input** type="url" id="url"/>

</**div**>

</**form**>



**59. What are the New tags in Media Elements in HTML5?**

* **<audio>** - Used for sounds, audio streams, or music, embed audio content without any additional plug-in.
* **<video>** - Used for video streams, embed video content etc.
* **<source>** - Used for multiple media resources in media elements, such as audio, video, etc.
* **<embed>** - Used for an external application or embedded content.
* **<track>** - Used for subtitles in the media elements such as video or audio.

<**label**>

Video:

</**label**>

<**video** width="320" height="240" controls>

<**source** src="video.mp4" type="video/mp4">

<**track** src="subtitles.vtt" kind="subtitles" srclang="en" label="English">

</**video**>

<**br**>

<**label**>

Embed:

</**label**>

<**embed** type="video/webm" src="https://www.youtube.com/embed/MpoE6s2psCw" width="400" height="300">

<**br**>

<**label**>

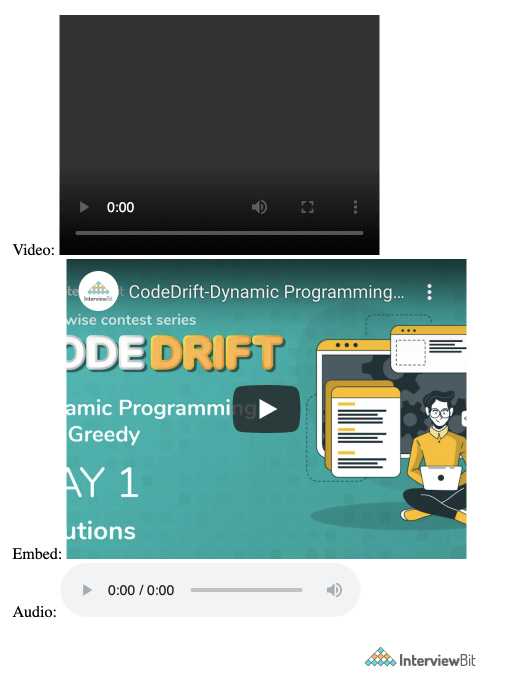
Audio:

</**label**>

<**audio** controls>

<**source** src="audio.mp3" type="audio/mpeg">

</**audio**>



**60. Why do you think the addition of drag-and-drop functionality in HTML5 is important? How will you make an image draggable in HTML5?**

The drag and drop functionality is a very intuitive way to select local files. This is similar to what most of the OS have copy functionality thus making it very easy for the user to comprehend. Before the native drag and drop API, this was achievable by writing complex Javascript programming or external frameworks like jQuery.

To enable this functionality there is a draggable attribute in the <img> tag and need to set ondrop and ondragover attribute to an eventhandler available in scripts.

<!DOCTYPE **HTML**>

<**html**>

<**head**>

<**script**>

**function** **allowDrop**(ev) {

ev.preventDefault();

}

**function** **drop**(ev) {

...

}

</**script**>

</**head**>

<**body**>

...

<**div** id="div1" ondrop="drop(event)" ondragover="allowDrop(event)" style="border: 1px solid #aaaaaa; width:350px; height: 70px;"></**div**>

<**br**>

<**img** id="drag1" src="img\_logo.gif" draggable="true" width="336" height="69">

...

</**body**>

</**html**>

**61. Why do we need the MathML element in HTML5?**

MathML stands for Mathematical Markup Language. It is used for displaying mathematical expressions on web pages. For this <math> tag is used.

<!DOCTYPE **HTML**>

<**html**>

<**head**>

</**head**>

<**body**>

<**math**>

<**mrow**>

<**mrow**>

<**msup**>

<**mi**> a </**mi**>

<**mn**> 2 </**mn**>

</**msup**>

<**mo**> + </**mo**>

<**msup**>

<**mi**> b </**mi**>

<**mn**> 2 </**mn**>

</**msup**>

<**mo**> + </**mo**>

<**mn**> 2 </**mn**>

<**mn**> a </**mn**>

<**mn**> b </**mn**>

</**mrow**>

<**mo**> = </**mo**>

<**mn**> 0 </**mn**>

</**mrow**>

</**math**>

</**body**>

</**html**>

This displays the equation a2 + b2 + 2ab = 0.

**62. What are the server-sent events in HTML5?**

The events pushed from the webserver to the browsers are called server-sent events. DOM elements can be continuously updated using these events. This has a major advantage over straight-up polling. In polling, there is a lot of overhead since every time it is establishing an HTTP connection and tearing it down whereas, in server-sent events, there is one long-lived HTTP connection. To use a server-sent event, <eventsource> element is used. The src attribute of this element specifies the URL from which sends a data stream having the events.

<**eventsource** src = "/cgi-bin/myfile.cgi" />

**63. What are Web Workers?**

These are added to bring parallelism and async capability. It runs in the background to do the computationally expensive tasks without yielding to make the page responsive. It is achieved by starting a separate thread for such tasks. These are not meant to perform UI operations. There are three types of web workers:

* **Dedicated Workers**- These are workers that are utilized by a single script.
* **Shared Workers** -These are workers that are utilized by multiple scripts running in different windows, IFrames, etc.
* **Service Workers** - These act as proxy servers between web applications, the browser, and the network. Mostly used for push notifications and sync APIs.

<**p**>Count numbers: <**output** id="result"></**output**></**p**>

<**button** onclick="startWorker()">Start Worker</**button**>

<**button** onclick="stopWorker()">Stop Worker</**button**>

<**script**>

**var** w;

**function** **startWorker**() {

**if**(**typeof**(Worker) !== "undefined") {

**if**(**typeof**(w) == "undefined") {

w = **new** Worker("demo\_workers.js");

}

w.onmessage = **function**(event) {

document.getElementById("result").innerHTML = event.data;

};

}

}

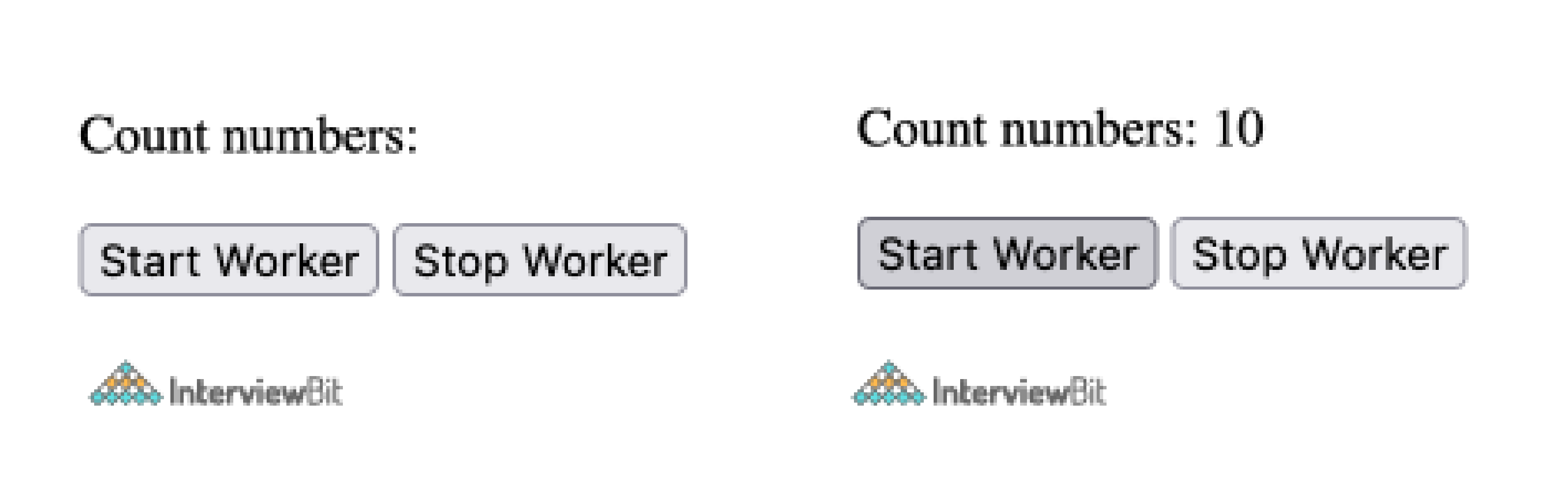
**function** **stopWorker**() {

w.terminate();

w = undefined;

}

</**script**>



### 64. What are the new <input> types for form validation in HTML5?

The new input types for form validation are email, URL, number, tel, and date.

**Example:**

1. **<input** type="email"**>**

**65. What is the usage of a novalidate attribute for the form tag that is introduced in HTML5?**

Its value is a boolean type that indicates whether or not the data being submitted by the form will be validated beforehand. By making this false, forms can be submitted without validation which helps users to resume later also.

<**form** action = "" method = "get" novalidate>

Name:<**br**><**input** type="name" name="sname"><**br**>

Doubt:<**br**><**input** type="number" name="doubt"><**br**>

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

</**form**>

**66. What are raster images and vector images?**

**Raster Images** - The raster image is defined by the arrangement of pixels in a grid with exactly what color the pixel should be. Few raster file formats include PNG(.png), JPEG(.jpg), etc.  
**Vector Images** - The vector image is defined using algorithms with shape and path definitions that can be used to render the image on-screen written in a similar markup fashion. The file extension is .svg

**67. What is a manifest file in HTML5?**

The manifest file is used to list down resources that can be cached. Browsers use this information to make the web page load faster than the first time. There are 3 sections in the manifest file

* **CACHE Manifest** - Files needs to be cached
* **Network** - File never to be cached, always need a network connection.
* **Fallback** - Fallback files in case a page is inaccessible

CACHE MANIFEST

# 2012-06-16 v1.0.0

/style.css

/logo.gif

/main.js

NETWORK:

login.php

FALLBACK:

/html/ /offline.html

<!DOCTYPE **HTML**>

<**html** manifest="tutorial.appcache">

...

...

</**html**>

**68. What is the Geolocation API in HTML5?**

Geolocation API is used to share the physical location of the client with websites. This helps in serving locale-based content and a unique experience to the user, based on their location. This works with a new property of the global navigator object and most of the modern browsers support this.

**var** geolocation = navigator.geolocation;

**.**

<!DOCTYPE **html**>

<**html**>

<**body**>

<**p**>Click "try it" button to get your coordinates.</**p**>

<**button** onclick="getLocation()">Try It</**button**>

<**p** id="demo"></**p**>

<**script**>

**var** x = document.getElementById("demo");

**function** **getLocation**() {

**if** (navigator.geolocation) {

navigator.geolocation.getCurrentPosition(showPosition);

} **else** {

x.innerHTML = "Geolocation functionality is not supported by this browser.";

}

}

**function** **showPosition**(position) {

x.innerHTML = "Latitude: " + position.coords.latitude +

"<br>Longitude: " + position.coords.longitude;

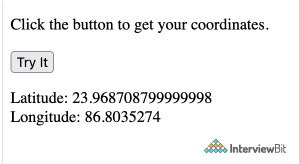
}

</**script**>

</**body**>

</**html**>

The above example asks for user permission for accessing the location data via geolocation API and after clicking the button the coordinates of the physical location of the client get displayed.



**69. Explain Web Components and it’s usage.**

These are used to create reusable custom elements which are very difficult in traditional HTML. It consists of three technologies:

* **Custom elements** - These are JavaScript APIs that help in defining custom elements and their behavior.
* **Shadow DOM** - These are JavaScript APIs that attach an encapsulated shadow DOM tree to an element to keep the element’s features private and unaffected by other parts.

<!DOCTYPE **html**>

<**html**>

<**head**>

<**meta** charset="utf-8">

<**title**>composed and composedPath demo</**title**>

<**script** src="main.js" defer></**script**>

</**head**>

<**body**>

<**h1**><**code**>composed</**code**> and <**code**>composedPath</**code**> demo</**h1**>

<**open-shadow** text="I have an open shadow root"></**open-shadow**>

<**closed-shadow** text="I have a closed shadow root"></**closed-shadow**>

</**body**>

</**html**>

customElements.define('open-shadow',

**class** **extends** **HTMLElement** {

**constructor**() {

super();

**const** pElem = document.createElement('p');

pElem.textContent = this.getAttribute('text');

**const** shadowRoot = this.attachShadow({mode: 'open'});

shadowRoot.appendChild(pElem);

}

}

);

customElements.define('closed-shadow',

**class** **extends** **HTMLElement** {

**constructor**() {

super();

**const** pElem = document.createElement('p');

pElem.textContent = this.getAttribute('text');

**const** shadowRoot = this.attachShadow({mode: 'closed'});

shadowRoot.appendChild(pElem);

}

}

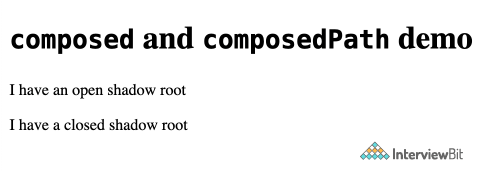
);

document.querySelector('html').addEventListener('click', e => {

console.log(e.composed);

console.log(e.composedPath());

});



Here 2 custom elements are defined <open-shadow> and <closed-shadow> which takes their text content and inserts them into a shadow DOM as content of a <p> element.

* **HTML templates** - The markup templates are written using <template> and <slot> elements which can be reused multiple times as the basis of a custom element's structure.

<!DOCTYPE **html**>

<**html**>

<**head**>

<**meta** charset="utf-8">

<**title**>Simple template</**title**>

<**script** src="main.js"></**script**>

</**head**>

<**body**>

<**h1**>Simple template</**h1**>

<**template** id="my-paragraph">

<**style**>

**p** {

**color**: white;

**background-color**: #666;

**padding**: 5px;

}

</**style**>

<**p**><**slot** name="my-text">My default text</**slot**></**p**>

</**template**>

<**my-paragraph**>

<**span** slot="my-text">Let's have some different text!</**span**>

</**my-paragraph**>

<**my-paragraph**>

<**ul** slot="my-text">

<**li**>Let's have some different text!</**li**>

<**li**>In a list!</**li**>

</**ul**>

</**my-paragraph**>

</**body**>

</**html**>

customElements.define('my-paragraph',

**class** **extends** **HTMLElement** {

**constructor**() {

super();

**const** template = document.getElementById('my-paragraph');

**const** templateContent = template.content;

this.attachShadow({mode: 'open'}).appendChild(

templateContent.cloneNode(true)

);

}

}

);

**const** slottedSpan = document.querySelector('my-paragraph span');

console.log(slottedSpan.assignedSlot);

console.log(slottedSpan.slot);



Here we are reusing the <my-paragraph> template.

**70. How many types of heading does an HTML contain?**

The HTML contains six types of headings which are defined with the <h1> to <h6> tags. Each type of heading tag displays different text size from another. So, <h1> is the largest heading tag and <h6> is the smallest one. For example:

1. **<h1>**Heading no. 1**</h1>**
2. **<h2>**Heading no. 2**</h2>**
3. **<h3>**Heading no. 3**</h3>**
4. **<h4>**Heading no. 4**</h4>**
5. **<h5>**Heading no. 5**</h5>**
6. **<h6>**Heading no. 6**</h6>**

**71. How to insert a copyright symbol on a browser page?**

You can insert a copyright symbol by using &copy; or &#169; in an HTML file.

### 72. Does a hyperlink only apply to text?

No, you can use hyperlinks on text and images both. The HTML anchor tag defines a hyperlink that links one page to another page. The "href" attribute is the most important attribute of the HTML anchor tag.

#### Syntax

1. **<a** href = "..........."**>** Link Text **</a>**

**73. What is a style sheet?**

A style sheet is used to build a consistent, transportable, and well-designed style template. You can add these templates on several different web pages. It describes the look and formatting of a document written in markup language.

### 74. Can you create a multi-colored text on a web page?

Yes. To create a multicolor text on a web page you can use <font color ="color"> </font> for the specific texts you want to color.

### 75. Is it possible to change the color of the bullet?

### The color of the bullet is always the color of the first text of the list. So, if you want to change the color of the bullet, you must change the color of the text.20) What is a marquee?

Marquee is used to put the scrolling text on a web page. It scrolls the image or text up, down, left or right automatically. You should put the text which you want to scroll within the <marquee>......</marquee> tag.

**76. How many tags can be used to separate a section of texts?**

Three tags are used to separate the texts.

* <br> tag - Usually <br> tag is used to separate the line of text. It breaks the current line and conveys the flow to the next line
* <p> tag - The <p> tag contains the text in the form of a new paragraph.
* <blockquote> tag - It is used to define a large quoted section. If you have a large quotation, then put the entire text within <blockquote>.............</blockquote> tag.

### 77. How to make a picture of a background image of a web page?

To make a picture a background image on a web page, you should put the following tag code after the </head> tag.

1. **<body** background = "image.gif"**>**

Here, replace the "image.gif" with the name of your image file which you want to display on your web page.

### 78. What are empty elements?

HTML elements with no content are called empty elements. For example: <br>, <hr> etc.

### 79. What is the use of a span tag? Give one example.

The span tag is used for following things:

* For adding color on text
* For adding background on text
* Highlight any color text

**Example:**

1. **<p>**
2. **<span** style="color:#ffffff;"**>**
3. In this page we use span.
4. **</span>**
5. **</p>**

### 80. Why is a URL encoded in HTML?

An URL is encoded to convert non-ASCII characters into a format that can be used over the Internet because a URL is sent over the Internet by using the ASCII character-set only. If a URL contains characters outside the ASCII set, the URL has to be converted. The non-ASCII characters are replaced with a "%" followed by hexadecimal digits.

### 81. What are the different new form element types in HTML 5?

Following is a list of 10 frequently used new elements in HTML 5:

* Color
* Date
* Datetime-local
* Email
* Time
* Url
* Range
* Telephone
* Number
* Search

### 82. Is there any need to change the web browsers to support HTML5?

No. Almost all browsers (updated versions) support HTML 5. For example Chrome, Firefox, Opera, Safari, IE.

### 83. What is the use of figcaption tag in HTML 5?

The <figcaption> element is used to provide a caption to an image. It is an optional tag and can appear before or after the content within the <figure> tag. The <figcaption> element is used with <figure> element and it can be placed as the first or last child of the <figure> element.

1. **<figure>**
2. **<img** src="htmlpages/images/tajmahal.jpg" alt="Taj Mahal"**/>**
3. **<figcaption>**Fig.1.1 - A front view of the great Taj Mahal in Agra.**</figcaption>**
4. **</figure>**

### 84. What is button tag?

The button tag is used in HTML 5. It is used to create a clickable button within the HTML form on the web page. It is generally used to create a "submit" or "reset" button. Let's see the code to display the button.

1. **<button** name="button" type="button"**>**Click Here**</button>**

### 85. What is the use of details and summary tag?

The details tag is used to specify some additional details on the web page. It can be viewed or hidden on demand. The summary tag is used with details tag.  [.](https://www.javatpoint.com/html-details-tag)

### 86. How are tags migrated from HTML4 to HTML5?

|  |  |  |
| --- | --- | --- |
| **No.** | **Typical HTML4** | **Typical HTML5** |
| 1) | <div id="header"> | <header> |
| 2) | <div id="menu"> | <nav> |
| 3) | <div id="content"> | <section> |
| 4) | <div id="post"> | <article> |
| 5) | <div id="footer"> | <footer> |

### Header and Footer Example

**HTML 4 Header and Footer:**

1. **<div** id="header"**>**
2. **<h1>**Monday Times**</h1>**
3. **</div>**
4. .
5. .
6. .
7. **<div** id="footer"**>**
8. **<p>**&copy; JavaTpoint. All rights reserved.**</p>**
9. **</div>**

**HTML 5 Header and Footer:**

1. **<header>**
2. **<h1>**Monday Times**</h1>**
3. **</header>**
4. .
5. .
6. .
7. **<footer>**
8. **<p>**© JavaTpoint. All rights reserved.**</p>**
9. **</footer>**

### Menu Example

**HTML 4 Menu:**

1. **<div** id="menu"**>**
2. **<ul>**
3. **<li>**News**</li>**
4. **<li>**Sports**</li>**
5. **<li>**Weather**</li>**
6. **</ul>**
7. **</div>**

**HTML 5 Menu:**

1. **<nav>**
2. **<ul>**
3. **<li>**News**</li>**
4. **<li>**Sports**</li>**
5. **<li>**Weather**</li>**
6. **</ul>**
7. **</nav>**

**87. What are different approaches to make an image responsive?**

* **Art direction** - Using <picture> element the landscape image fully shown in desktop layout can be zoomed in with the main subject in focus for a portrait layout.

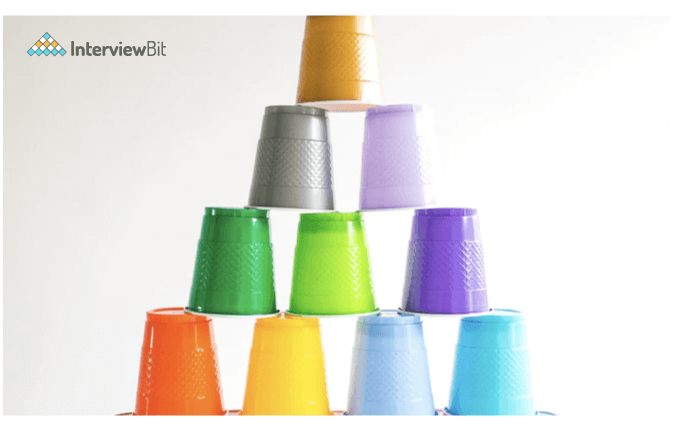
<**picture**>

<**source** media="(min-width: 650px)" srcset="img\_cup.jpg">

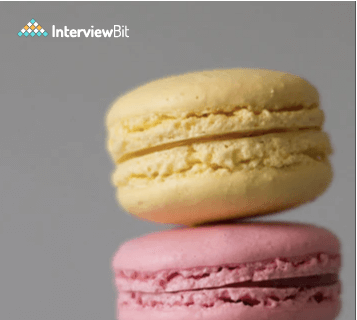
<**img** src="img\_marsh.jpg" style="width:auto;">

</**picture**>

***Bigger Screen (>650px)***



***For any other screen***



* **Resolution switching**- Instead of zoom and crop the images can be scaled accordingly using vector graphics. Also, this can be further optimized to serve different pixel density screens as well.

***For example SVG***

<**svg** width="100" height="100">

<**circle** cx="50" cy="50" r="40"

stroke="green" stroke-width="4" fill="yellow" />

</**svg**>

