Q1.Can you explain the difference between HTML and HTML5?

Ans: Differences Between HTML and HTML5 are

SL No.	HTML	HTML5
1.	(HTML stands for Hypertext Markup Language. It the older version of HTML.	HTML stands for Hypertext Markup Language. It the newest version of HTML.
2.	It does not support video and audio.	It supports both support video and audio .
3.	In html we have to give an id or class to every section for example, , , , etc	In html5 we do not need like html .Here it gives the facility of tags for example <header></header> <footer></footer> <nav></Nav> <form></form> etc
4.	HTML is compatible with almost all browsers because it has been present for a long time, and the browser made modifications to support all the features.	In HTML5, we have many new tags, elements, and some tags that have been removed/modified, so only some browsers are fully compatible with HTML5.
5.	In HTML, vector graphics are possible with tools Like Silver light, Adobe Flash, VML, etc.	It supports vector graphics by default.

6.	Doctype declaration in html is too long <! DOCTYPE HTML PUBLIC "- // W3C // DTD HTML 4.01 // EN" "http://www.w3.org/TR/html4/strict.dtd">	The DOCTYPE declaration in html5 is very simple "<! DOCTYPE html>
7.	It does not allow to draw shapes like circles, rectangles, triangles.	Using html5, you can draw shapes like circles, rectangles, triangles**.**
8.	Works with all older browsers	A new browser supports this.

Q2. What is the difference between ID and Class selector?

Ans: The difference between Id and class selectors are that the “class” selector can be common for multiple division or throughout the webpage but “id” selector will be unique for division of webpage.

For example:

For id selector

,

,

For class selector

,

,

We use the id in style sheet by appending “#” (hashtag) symbol before the id sector name while in class selector we use “.”(Dot) symbol before the class selector name.

For Example:-

```
# select1{
},
.select{
}
```

Q3. What is the difference between null value and undefined value?

Ans: If we assigned the value “Undefined” In JavaScript. This means the variable is untouched and can be assigned any value in future. It also means that nobody knows the value that this variable is going to hold at the time of declaration. “Undefined” also means a variable has been declared but has not yet been assigned a value for example:

```
var name;
alert(name);
```

it shows undefined in dialogue box.

Null Value: null is a special value meaning "no value". null is a special object because typeof null returns 'object'.

```
let supervisor = null;
```

Q4. What is Scope in JavaScript? Name the different types of Scopes.

Ans: In JavaScript “scope” is a function. which determines the accessibility (visibility) of variables.

Variables defined inside a function are not accessible (visible) from outside the function.

In java script there are two types of scope.

a. Local Scope: Variables declared within a JavaScript function, become **LOCAL** to the function.

```
function myFunction() {
  var color = "blue";
}
```

b. Global Scope: A variable declared outside a function, becomes **GLOBAL**. A global variable has **global scope**: All scripts and functions on a web page can access it.

```
var color = "blue";
function myFunction() {
}
```

Q5. What is CSS ? Explain how many types css?

Ans: CSS is the language we use to style a Web page. CSS stands for Cascading Style Sheets. It describes how HTML elements are to be displayed on screen, paper, or in other media. CSS saves a lot of work. It can control the layout of multiple web pages all at once. External stylesheets are stored in CSS files.

Three are three types of CSS

a. **Inline CSS** : Inline css is used directly in tag by using “style” attributes in current page.

   For example:

b. **Embeded CSS**: It is use between Style tag in same page.
   For example:

   <style>
   . myclass{

   }
   <style>

   </style>

c. **External CSS:** In External css we make separate page/file or file od css and save it .css extention and link it in html page
   For example:
   <link rel="stylesheet" type="text/css" href="filename.css">

Q6. What is HTML5 Web Storage?

Ans: It is a simple client side database that allows the users to persist data in the form of key/value pairs. With web storage, web applications can store data locally within the user's browser. Before HTML5, application data had to be stored in cookies, included in every server request. Web storage is more secure, and large amounts of data can be stored locally, without affecting website performance. Unlike cookies, the storage limit is far larger (at least 5MB) and information is never transferred to the server.

Web storage is per origin (per domain and protocol). All pages, from one origin, can store and access the same data.

HTML web storage provides two objects for storing data on the client:

- window.localStorage - stores data with no expiration date
- window.sessionStorage - stores data for one session (data is lost when the browser tab is closed)

Before using web storage, check browser support for localStorage and sessionStorage:

```
if (typeof(Storage) !== "undefined") {
  // Code for localStorage/sessionStorage.
} else {
  // Sorry! No Web Storage support..
}
```

Q7. What's the difference between Canvas and SVG?

Ans:

Sl no	Canva	SVG
1	Raster based (composed of pixel)	Vector based (composed of shapes)

	Canvas has poor scalability. Hence it is not suitable for printing on higher resolution.	SVG has better scalability. So it can be printed with high quality at any resolution.
	Canvas gives better performance with smaller surface or larger number of objects.	SVG gives better performance with smaller number of objects or larger surface.
	Canvas can be modified through script only.	SVG can be modified through script and CSS.
	Single element similar to <img> in behavior. Canvas diagram can be saved to PNG or JPG format.	Multiple graphical elements, which become the part of the page's DOM tree.

Q8. What is doctype?

Ans: All HTML documents must start with a <!DOCTYPE> declaration. The declaration is not an HTML tag. It is an "information" to the browser about what document type to expect. The <!DOCTYPE> declaration is NOT case sensitive.
<!DOCTYPE html>, <!DocType html>, <!Doctype html> , <!doctype html>
In older documents (HTML 4 or XHTML), the declaration is more complicated because the declaration must refer to a DTD (Document Type Definition).

HTML 4.01: <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
XHTML 1.1: <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN" "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

Q9. Explain what is Bootstrap?

Ans: Bootstrap is the most popular CSS Framework for developing responsive and mobile view websites. Bootstrap 5 is the newest version of bootstrap. It provides the in-built classes of css which we can use it to our html page.

Q10. Explain what is Bootstrap 3 & 4 Grid System?

Ans: **Bootstrap 3 Grid system:** Bootstrap 3 grid system is responsive, and the columns will re-arrange depending on the screen size: On a big screen it might look better with the content organized in three columns, but on a small screen it would be better if the content items were stacked on top of each other.
Bootstrap's grid system allows up to 12 columns across the page.
Either we can use as 12 columns or use at once as col-lg-12 , col-md-12, col-sm-12 , col-sx-12 or use 12 columns by dividing 2 or 4 part.

Col-md-1	Col-md-1	Col-md-1	Col-md-1	Col-md-1	Col-md-1v	Col-md-1	Col-md-1	Col-md-1	Col-md-1	Col-md-1	Col-md-1

Col-md-2	Col-md-2	Col-md-2	Col-md-2	Col-md-2	Col-md-2

Col-md-3	Col-md-3	Col-md-3	Col-md-3

Grid Classes of bootstrap 3:

- xs (for phones - screens less than 768px wide)
- sm (for tablets - screens equal to or greater than 768px wide)
- md (for small laptops - screens equal to or greater than 992px wide)
- lg (for laptops and desktops - screens equal to or greater than 1200px wide)

Some Bootstrap grid system rules:

- Rows must be placed within a .container (fixed-width) or .container-fluid (full-width) for proper alignment and padding
- Use rows to create horizontal groups of columns
- Content should be placed within columns, and only columns may be immediate children of rows
- Predefined classes like .row and .col-sm-4 are available for quickly making grid layouts
- Columns create gutters (gaps between column content) via padding. That padding is offset in rows for the first and last column via negative margin on .rows
- Grid columns are created by specifying the number of 12 available columns you wish to span. For example, three equal columns would use three .col-sm-4
- Column widths are in percentage, so they are always fluid and sized relative to their parent element

Bootstrap 4 Grid system: Bootstrap4 grid system is responsive, and the columns will re-arrange depending on the screen size: On a big screen it might look better with the content organized in three columns, but on a small screen it would be better if the content items were stacked on top of each other. Bootstrap4 grid system allows up to 12 columns across the page.
Either we can use as 12 columns or use at once as col-lg-12 , col-md-12, col-sm-12 , col-sx-12 or use 12 columns by dividing 2 or 4 part.

Col-md-1	Col-md-1	Col-md-1	Col-md-1	Col-md-1	Col-md-1v	Col-md-1	Col-md-1	Col-md-1	Col-md-1	Col-md-1	Col-md-1

Col-md-2	Col-md-2	Col-md-2	Col-md-2	Col-md-2	Col-md-2

Col-md-3	Col-md-3	Col-md-3	Col-md-3

Grid Classes of bootstrap 4:

- .col- (extra small devices - screen width less than 576px)
- .col-sm- (small devices - screen width equal to or greater than 576px)
- .col-md- (medium devices - screen width equal to or greater than 768px)
- .col-lg- (large devices - screen width equal to or greater than 992px)
- .col-xl- (xlarge devices - screen width equal to or greater than 1200px)

The classes above can be combined to create more dynamic and flexible layouts.

Some Bootstrap 4 grid system rules:

- The biggest **difference between Bootstrap 3 and Bootstrap 4** is that Bootstrap 4 now uses flexbox, instead of floats. One big advantage with flexbox is that grid columns without a specified width will automatically layout as "equal width columns" (and equal height). Example: Three elements with .col-sm will each automatically be 33.33% wide from the small breakpoint and up.
- Rows must be placed within a .container (fixed-width) or .container-fluid (full-width) for proper alignment and padding
- Use rows to create horizontal groups of columns

- Content should be placed within columns, and only columns may be immediate children of rows
- Predefined classes like .row and .col-sm-4 are available for quickly making grid layouts
- Columns create gutters (gaps between column content) via padding. That padding is offset in rows for the first and last column via negative margin on .rows
- Grid columns are created by specifying the number of 12 available columns you wish to span. For example, three equal columns would use three .col-sm-4
- Column widths are in percentage, so they are always fluid and sized relative to their parent element

Q11. What flex in Bootsrap 4?

Quickly manage the layout, alignment, and sizing of grid columns, navigation, components, and more with a full suite of responsive flexbox utilities. For more complex implementations, custom CSS may be necessary.

We apply display utility to create flex box. By doing so we transform direct children elements into flex items. Flex containers and items are able to be modified further with additional flex properties.

For example:

This is flex bos caontainer

Responsive variations also exist for .d-flex and .d-inline-flex.

- .d-flex

- .d-inline-flex
- .d-sm-flex
- .d-sm-inline-flex
- .d-md-flex
- .d-md-inline-flex
- .d-lg-flex
- .d-lg-inline-flex
- .d-xl-flex
- .d-xl-inline-flex

Q12. What is jQuery?

Ans: Jquery is javascript library . it is lightweight. By using jquerywe hav to write less code do more work The purpose of jQuery is to make it much easier to use JavaScript on your website.

jQuery takes a lot of common tasks that require many lines of JavaScript code to accomplish, and wraps them into methods that you can call with a single line of code.

jQuery also simplifies a lot of the complicated things from JavaScript, like AJAX calls and DOM manipulation.

The jQuery library contains the following features:

- HTML/DOM manipulation
- CSS manipulation
- HTML event methods
- Effects and animations

- AJAX
- Utilitie

Q13.What is Contextual classes of table in Bootstrap?

Ans: The Bootstrap Contextual Class allows us to change the background color of our table rows or individual cells.

Some Contextual Classes are:

.table-active

.table-primary

.table-secondary

.table-success

.table-danger

.table-warning

.table-info

.table-light

.table-dark

We can apply it whole table or in individual table row (tr), table data (td)

<tr class="table-active">...</tr>

<tr class="table-primary">...</tr>

<tr class="table-secondary">...</tr>

<tr class="table-success">...</tr>

<tr class="table-danger">...</tr>

<tr class="table-warning">...</tr>

<tr class="table-info">...</tr>

<tr class="table-light">...</tr>

<tr class="table-dark">...</tr>

Q14. Explain types of lists supported by Bootstrap.

There are three different types of lists in bootstrap:

- Unordered lists — A list of items in which the order does not explicitly matter.

  The list items in unordered lists are marked with bullets.
- Ordered lists — A list of items in which the order does explicitly matter. The list. Items in ordered lists are marked with numbers.
- Definition list — A list of terms with their associated descriptions.

Q15.Differences between display none and visibility hidden.

Ans: Display None: When we use the display none the element gets hidden and no space taken by that item in the browser.

Visibility Hidden: When we use the Visibility Hidden the element gets hidden but take space in the browser.