HTML & CSS

1. What is HTML and what is its purpose?

HTML, which stands for hypertext Markup Language, is the standard markup language used to create and design web pages. It is the backbone of most web content & provides a structure for organizing and presenting information on the internet. HTML is not a programming language; rather, it is a markup language that defines the structure & layout of content on a web page.

The primary purpose of html is to structure content on the web. It uses a system of tags to define different elements within a document, such as headings, paragraphs, lists, links, images and more. These tags help web browsers interpret and display the content in a structured and visually appealing manner.

2. What is the difference between HTML and XHTML?

HTML	XHTML
1. Html has more lenient syntax, allowing for certain deviations and errors. Browsers are designed to handle such errors & still render the page.	1. Xhtml has a stricter syntax based on xml rules. It requires well-formed documents, meaning that elements must be properly nested, closed, and in lowercase.
2. HTML documents have a more relaxed structure. eg:- empty elements like line breaks or image can be written as or <img> without a closing slash & its valid.	2. Xhtml requires self-closing tags for empty elements, such as or <img/> following the XML syntax.
3. ~~Browsers~~ The MIME type for html documents is typically 'text/html'.	3. The MIME type for xhtml documents is usually 'application/xhtml+xml'.

3. What are the new features introduced in html5?

1. Semalic Elements:-

Html5 introduced several new semantic elements that provide a more meaning structure to web documents. eg:- include <header>, <footer>, <nav>, <article>, <section> & <aside>. These elements help discribe the purpose and meaning of different parts of a webpage, making it more accessible & search engine-friendly.

2. New form Elements:-

HTML5 introduced new input types such a <input type="data"> <input type="email">, <input type="url"> & <input type="number">. These input types make it easier to collect specific types of data & provides a bitter user experience.

4. How do you include comments in html?

In html, you can include comments using the following syntax.

<!-- this is a comment -->

Anything blew <!-- and --> will be treated as a comment and not displayed in a browser. Comment are useful for adding explanations or notes within your html code without affecting the rundered output.

. Comments are used for documentation, provided explanations about code.

. Comments are do not appias in the rundered output of the webpage.

5. Explain the difference between and tags.

. (division) is a block-level element used to group other html elements together.

. is an inline element used to apply styles to a specific parts of the text within a larger block of content.

6. what are Semantic elements in HTML5 and why are they important?

1. Clarity and Readability :- Semantic elements improve the document's structure, making it easier for developers to understand the purpose of different sections.

2. Accessibility: They enhance accessibility for users with disabilities by providing more meaningful information to assistive technologies.

3. Search Engine optimization :- Search Engines use semantic markup to better understand the content and context of a page, potentially improving its search ranking.

7. What is the purpose of the <header>, <nav>, <section> and <footer> tags in HTML5?

. <header> is used to define a header for a document or a section. It often contains headings, logos and navigation elements.

· <nav> is used to define a set of navigation links, facilitating navigation within the document & to other documents.

· <section> is a generic container for grouping related content. It helps organize content into thematic groups.

· <footer> is used to define a footer for a document it typically contains metadata, copyright information.

8. How do you Create a hyperlink in HTML?

To Create a hyperlink in HTML, use the <a> <anchor> tag with the href attribute to specify the URL. Here's an example:

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

9. What is the difference between <ol> and <ul> Elements

The <ol> (ordered list) element is used to create a numbered list, where each list item is preceded by a number. (unordered list) elements is used to Create a bulleted list, where each list item is preceded by a bullet point.

10. How do you embed an image in HTML?

To embed an image in HTML, you use the <img> (image element). Here's a basic example.

```
<img src="image.jpg"
alt="Description of the image">
```

- src attribute specifies the path @ URL of the image file.
- alt attribute provides alternative texts for accessibility and is displayed if the image cannot be loaded.

11. Explain the difference between the & tag

- is used to define text with strong importance. It typically renders the text in a blod font, but the primary purpose is to convey that the text is of strong significance.

Example:

```
<p> This is <strong>important</strong> information. </p>
```

- is used to emphasize text. it typically renders the text in italics, but like , the emphasis is the key, not the specific styling.

Example:

```
<p> This text is <em> Emphasized</em>.</p>
```

12. How do you Create a table in HTML?

```
<table>
  <tr>
    <th>Header 1</th>
    <th>Header 2</th>
  </tr>
  <tr>
    <td>Row 1, Cell 1</td>
    <td>Row 1, Cell 2</td>
  </tr>
  <tr>
    <td>Row 2, Cell 1</td>
    <td>Row 2, Cell 2</td>
  </tr>
</tabe>
```

In this Example:-

- <table> defines the table.
- <tr> defines a table row.
- <th> defines a table header cell.
- <td> defines a table data cell

13. What is the purpose of the <form> tag in HTML and how do you Create a form?

The <form> tag in HTML is used to Create an interactive form that allows users to input data, which can be submitted to a server for processing.

```
<form action="/Submit-form"
method="post">
  <label for="username">Username:</
label>
  <input type="text" id="username"
name="username" required>
  <label for="password">password:</
label>
  <input type="password"
id="password" name="password"
required>
  <input type="Submit"
Value="Submit">
</form>
```

14. What are some new input types introduced in HTML5?

1. Email: Speifically for Email addresses. It helps Validate that the entered text is in a valid email format.

```
<input type="email"
name="user_email" required>
```

2 Url: Used for website URLs. It helps Validate that the entered text is a valid URL.

```
<input type="url"
name="website-url" required>
```

3. tel : designed for telephone numbers. It allows for input and validation of phone number.

```
<input type="tel"
name="phone_number" required>
```

15. How do you include audio and video content in HTML

Audio:

```
<audio controls>
  <source src="audio.mp3"
type="audio/mp3">
    Your browser does not support the audio tag.
</audio>
```

Video:

```
<video controls width="600">
    <source src="video.mp4"
type="video/mp4">
      your browser doesnot support the video tag.
</video>.
```

Replace "audio.mp3" and vedio.mp4" with the actual URLs of your audio & vedio files. The text within the Elements will be displayed if the browser doesn't supports the audio & video tag.

16. What is the purpose of the <iframe> tag and how is it used?

The <iframe> (Inline frame) tag in HTML is used to Embed another HTML document or external content within the current document. It allows you to display content from another source, link a web page, in a specific area of your webpage.

Example:

```
<iframe
src="https://www.Example.Com"
width="600" height="400"
title="Embedded Content"></iframe>.
```

17. How do you add CSS styles to HTML elements?

1. Inline styles: you can apply styles directly to an HTML element using the style attribute.

```
<p style="Color: blue;
font-size: 16px;">This is a styled pragraph.</p>
```

2. Internal style: you can include CSS styles within the HTML document using the <style> element in the document head.

```
<head>
  <style>
  P{
    color: green;
```

```
        font-size : 18px;
    }
    </style>
</head>
<body>
    <p>This is a styled paragraph.</p>
</body>
```

18. What is the role of the alt attribute in <img> tags?

The alt attribute in <img> tags is used to provide alternative text for an image. This text is displayed if the image. This text is displayed if the image cannot be loaded @ if the user is using a screen reader.

Example:

```
<img src='Example.jpg' alt="A
Picturesque mountain landscape">
```

19. How do you Create a number list with Custom numbering styles in HTML?

In HTML, you can Create a numbered list Using the <ol> (ordered list) element. To apply Custom numbering styles, you can use the CSS

Example:

```
<style>
  ol.upper-roman {
    list-style-type: upper-roman;
  }
  ol.lower-alpha {
    list-style-type: lower-alpha;
  }
</style>
<ol class="upper-roman">
    <li>Item 1</li>
    <li>Item 2</li>
    <li>Item 3</li>
</ol>
<ol class="lower-alpha">
    <li>Item A</li>
    <li>Item B</li>
    <li>Item C</li>
</ol>
```

20. what is the difference between <Script async> Es <Script defer>

1. Execution Timing:

- Async: The script is executed asynchronously as soon as it is available, without blocking the HTML parsing.

```
<Script async
 src="script.js"></script>
```

- defer: The script is executed in order, but deferred until the HTML parsing is complete.

```
<script defer
 src="script.js"></script>
```

2. Parsing Behavior:

- async: Does not block HTML parsing, allowing the HTML parsing to continue while the script is being fetched.
- defer: Also does not block HTML parsing, but the script is deferred until the parsing is complete.

21. What is responsive web design, and why is it important?

Responsive web design is an approach to designing and building websites that ensure a seamless and optimal user experience across various devices & screen sizes. The goal is to create website that adapt and respond to the user device.

Importance:

1. Importance User Experience: Responsive design provides a consistent and user-friendly experience across devices.

2. Broader audience reach: with the increasing use of smartphones and tables, ensuring that your website is accessible.

3. Search Engine optimization: google & other search Engines prioritize mobile-friendly websites in search results.

22. How do you make a website responsive using CSS?

1. Viewport Meta Tag: Ensure you have the viewpoint meta tag in the <head> of the HTML document.

```
<meta name = "viewport"
Content = " width=device-width,
Initial - scales = 1.0">
```

2. Flexible Images: Ensure images can scale based on the Container size by using the max-width: 100%. CSS rule

```
img {
    max-width: 100%.
    height: auto;
}
```

23. What is a Media query in CSS, and how is it used for responsive design?

A media query is CSS a way to apply different styles based on various characteristics of the user's device, such as its screen size, resolution, or other features. Media queries enable you to create responsive designs.

Example of using a responsive design:—

```
/* Default styles for all screens */
body {
    font-size: 16px;
}
/* Media query for screens with a maximum width of 600 pixels */ @ media screen and (max-width: 600px)
{
  body {
      font-size: 14px;
  }
}
```

24. Explain the difference between a fluid layout and and a fixed layout in terms of responsiveness?

Fluid layout for Responsiveness :-

Fluid layouts are favored for responsive design because they allow content to adjust gracefully to different screens, providing a better user experience on a variety of devices.

Fixed Layout challenges :-

Fixed layouts may require additional adjustments @ a separate mobile version to cater to different screen sizes, potentially leading to less optimal user experience.

25. How do you make images responsive in CSS?

To make images responsive in CSS, you can use the max-width: 100% rule. This ensures that the image will not exceed the width of its container, allowing it to scale proportionally and maintain its aspect ratio. Here's an example :-

```
img {
  max-width: 100%;
  height: auto; /* Ensure the
image's aspect ratio is maintained
*/
}
```

26. What are breakpoints in responsive design and how are they determind?

Breakpoints are determind based on the design and content of a website & they are typically associated with changes in layout, styling & the presentation of content.

1. Device-specific Breakpoints:-
   - These breakpoints are based on the typical screen sizes of specific devices, such as smartphones, tablets & desktops.

2. Content-Based Breakpoints:-
   - Breakpoints can also be determined by the content and layout of a specific website.

3. Fluid Grid Breakpoint:-
   Responsive design often involves using a fluid grid layout.

27. How can you hide elements on specific screen sizes using CSS?

- The default styles apply to all screen sizes.
- Inside the media query (@media screen & (max-width: 600px)). the element-class .selector is used to target the element you want to hide.
- The 'display: none' property value pair is used to hide the element the screen width is 600 pixels or smaller.

28. What is the purpose of the max-width property in responsive CSS?

The max-width property in CSS commonly used in responsive web design to control the maximum width of an element. It is often employed in conjuction with media queries to create layouts that adapt to different screen sizes or devices.

- The default style outside the media query apply to all screen sizes.
- The margin: 0 auto; rule centers the container horizontally within its parent.

29. How do you create a responsive navigator menu using CSS?

- The navigation menu is structured using an unordered list (<ul>) for the menu items.
- The menu icon is initially hidden on larger screens and displayed on smaller screens.
- Flexbox is used for layout, providing a flexible styles & responsive structure for the navigator menu.
- The show class is used to toggle the display of the navigation list when the menu icon is clicked.

30. Explain the concept of mobile-first design and how it relates to responsive CSS.

• Starting with the Basics:-

In a mobile-first approch, the design begins with the essential elements & features that are crucial for mobile user.

• Progressive Enhancement:-

As the screen size increases, features and design elements can be progressively enhanced to take advantage of the additional space & capabilities.

• Media Queries:-

Responsive CSS plays a key role in mobile-first design through the use of media queries.

31. Explain the concept of mobile-first design and how it relates to responsive CSS.

• Starting with the Basics:

In a mobile-first approach the design begins with the essential elements and features that are crucial for mobile users.

• Progressive Enhancement:-

As the screen size increases, features and design elements can be progressively enhanced to take advantage of the additional space and capability.

32. What is css flexbox, and what problem does it solves?

. Vertical Centering:-
Archiving vertical centering in css has historically been challenging.

• Equal Hight columns:-
Creating equal height colums in traditional layouts often required complex techniques such as flex colums or javascript.

. Distribution of space:-
Flexbox makes it easy to distribute space along a single axis & control how extra space.

33. Explain the difference blw flex Containes and flex items.

• Flex Containes:-
A flex container is an element that as the property display: flex @ 'display: inline-flex' applied to it.

. Flex items :-
Flex items are the children of a flex container. They are the elements that are laid out within the flex container becomes a flex items.

34. How do you create a flex container in CSS?

· The justify-content property defines how the items are aligned along the main axis.

· The align-items property defines how the items are aligned along the cross axis.

· The flex-wrap property is set to wrap if you want to the items to wrap to the next line.

35. What are the main properties used to control the layout in flexbox?

· display:-

value: flex | inline-flex

Description: Sets an elements as flex containes. The value flex creates a block-level flex container.

· flex-direction:-

values: row | row-reverse | column | column-reverse.

Description: Defines the main axis of the flex containes

· justify-content:-

value: flex-start | flex end | center | space-between space around | space evenly.

Description: Aligns flex items along the main axis of the flex container.

36. How do you specify the direction of flex items within a flex-container?

• row:-
Flex items are placed in a row along the main axis. starting from the left on the right.

• row-reverse:-
Flex items are placed in a row along the main axis, but the order is reversed starting from the end.

• column:-
Flex items are placed in a column along the main axis starting from the top.

• column-reverse:-
Flex items are placed in a column along the axis, but the order is reversed, starting from the bottom.

37. How do you align flex items horizontally and vertically within a flex container?

• Align Horizontal:
• The justify-content property align flex items along the main axis.
• Possible values for justify-content include.
flex start, flex end, center, space-between, space-around, space-evenly.

38. How can you control the order of flex items using CSS flexbox?

·<integer> A neumeric value rep the order of the flex item.

item1 : has an order of 2 it appear second.

item2 : has an order of 2 so it appear first.

item3 : has an order of 3 so it appear third.

The order property allows you to rearrange flex items visually without changing thier source order in the html.

39. What are flexbox breakpoints and how can they be used for responsive design?

1. Analyze your design and design specific viewpoint widths where the layouts or styling needs to change to improve usability & visual appeal.

2. Use media queries:-

Apply media queries in you css code to target specific viewpoint ranges. Media queries allow you to condition apply styles based on charicteristics such as srun width.

3. Adjust flexbox properties.

4. Optimize for mobile-first.

40. Explain the difference blw justify-content and align-items properties in Flexbox.

1. justify content :-
Axis : Main Axis.
Description : - The justify content property align flex items along the main axis of the flex container.
values : flex start, flex end, center, space-between, space-around, space-evenly.

2. Align-items :-
Axis :- Cross Axis
Description : The align items property align flex items along the cross axis of the flex container.
values : flex-start, flex-end, center, baseline, stretch.

41. What are html attributes?
. href is an attribute, and "https://example.com" is its value. it specifies the hyperlink reference.
. target is another attribute and '_blank' its value.

1. class and id :-
Used for styling and javaScript / css target.

2. src and alt :-
used in <img> elements for specification and image source and alternative text.

3. href and target:-
used in <a> elements for creating hyperlinks and specifying the target.

4. disabled:-
used in form elements to disable user interaction.

42. Explain the difference b/w global attributes and element-specific attributes in html.

1. Global Attributes:-
Global Attributes are attributes that can be used with any html element, regardless of its type.

2. Element-specific Attributes:-
Element-specific attributes are attributes that are designed to be used with specific html elements.

Key Difference:-

- Applicability:-
  - Global Attributes :- Can be used with any html element.
  - Element-Specific Attributes :-

Designed for use the specific types of elements.

3. Flexibility :-
Offer flexibility in terms of applying similar properties to various elements.

43. How do you add attributes to an html elements?

1. Single Attributes:-
This example add a class attributes to a elements.

2. Multiple attributes-
Here both src and alt attributes are added to an <img> element. Boolean

3. Multiple attributes:-
Here both are boolean meaning they don't require a value.

4. Custom data Attribute:-
Customen data attributes are used for storing private data or metadata.

44. What is the purpose of the id attribute in html, and how is unique?

1. uniqueness:-
Every id value within a document must be unique. No one elements should have the same id.

2. Usage in js:-
js can use the getElementById method to access and manipulated an element with a specific id.

3. usuage on css:-
The id can be used a selector in css to apply styles to a specific element.

45. What is the difference blw the class attributes and the id attribute?

1. class Attribute:-

- Purpose:- The class attribute is used to apply one (or) more class names to an html element. Classes are commonly used for styling multiple elements in a consistent way or for selecting elements with js.

a. Uniqueness:-

Multiple elements can share the same class name, and an element can have multiple class names separated by spaces.

2id Attributes:-

- purpose:- The id attributes is used to provide a unique identifier for specific html element.
- uniqueness:- Each id value must be unique with the html document.

46. Explain the role of the href attributes in html particularly in the context of links and anchors.

- Linking to External Resources:-

Linking to other website, pages or resources on the internet

 Visit Example

- Linking to Internal Resources:-

Linking to other pages within the same website.

 Learn more About us

3. Creating Email Links :-

Generating email links by specific an email address in the href attribute.

 send us an Email

4. Anchors for In-page Navigation :-

Creating anchors within a page for in-page navigation.

47. How do you add alternative text to an image using the alt attribute ?

1. Accessibility :-

Screen readers use the alt text to describe images to user with visual impairments.

2. SEO :-

Search engines use alt text to understand the context of image contribute to better search rankings.

3. Placeholder text :-

If the image fails to load the alt text is displayed as a placeholder.

* src specific the path to the image file.

* alt provides a descriptive text for the image.

Its recommended to make alt text concise and descriptive conveying the essential information about the image.

48. What is the purpose of the target attributes in html links and what are its possible values?

1. self (Default)
Open the linked document or resource in the same window or tab that the link was clicked.

2. _blank:-
Opens the linked document or resources in a new browser window or tab.

3. _parent:-
Opens the linked document or resource in the parent frame or window of the current frame or window.

4. _top:-
Opens the linked document or resource in the top-level browsing context.

49. How do you use the src attribute to embed an external resource, such as an image or video in html?

1. Embedding an image:-
To embed an external image, use the <img> element and set the src attribute to the url of the image.

2. Embedding a Video:-
To embed an external video, use the <video> element and set the src attribute to the URL of the video file.

3. Embedding an Audio?

To embed an external audio file, use the <audio> element and set the src attributes to the url of the audio file.

So. What is the purpose of the disabled attributes and how is it used in html form element?

1. Text input (<input type = "text">) textarea (<textarea>) and password Input (<input type = "password">);

2. Checkbox (<input type = "checkbox">) and Radio Button (<input type = "radio">):

3. Select Dropdown (<select>) and its options (<option>)

4. Button (<button>):

5. Form (<form>):

· Disabling an entire form will disable all the form controls within it.

JavaScript :-

51. Is there any relation between java and javascript.

Despite their similar names, java and javascript are distinct programming language with different purpose and characteristics. Java is a general-purpose, object-oriental programming language, while javascript is a scripting language primarily used for web development to add interactivity to web pages.

52. Is javascript a Compiled or interpreted language.

Javascript is primarily an interpreted language. Web browsers have built-in javascript engine that interpret and execute the javascript code directly. However, modern javascript code directly. However, This means that the javascript code is translated into machine code just before it is executed, optimizing its Performance.

53. Is javascript a case-sensitive language.

Yes. Javascript is a case-sensitive language. This means that it distinguishes b/w upperCase and lowerCase letters For example Variables named "MyVar" and "myvar" would be considered different variables

It's essential to be consistent with the casing when working with javascript to avoid errors related to variable names, function names.

54. What is nodejs.

Node.js is a javascript runtime environment built on the chrome v8 javascript engine. It allows developers to run javascript on the server side, enabling server-side scripting and the development of scalable network applications. It has a large ecosystem of libraries and packages available through npm, making it a popular choice for web development.

55. What is the difference between let and var.

1. Scope :-

• Variable declared with var are function-scoped. This means they are only visible within the function where they are declared.

• Variables declared with let are block-scoped. during in which they are defined. This includes loops, conditionals, and any other block.

2. Reassignment :-

• Variables declared with var can be re-declared within the same scope without raising an error.

• Variables declared with let cannot be re-declared within the same scope.

56. What are the differences between undeclared and Undefined Variables.

• Undeclared Variables:-

- An undeclared variable is one that has been used without being declared using var, let.

- If you try to access the value of an undeclared variable, it will typically result in a ReferenceError.

• Undefined Variables:-

- An undefined variable is one that has been declared but has not been assigned a value.

- Accessing the value of an undefined variable is allowed in javascript, and it will result in the value Undefined.

57. What is Hoisting.

Hoisting is a javascript behavior where Variable and function declarations are moved to the top of their Containing Scope during the Compilation phase.

1. Variable Hoisting:-

variable declared with var are hoisted and.

hoisted and initialized with the value undefined.

```
Console.log(x); // undefined var x = 5;
```

2. Function Hoisting :-

Function declarations are hoisted along with their definitions.

```
SayHello(); // "Hello!"
function sayHello(){
  Console.log("Hello!");
}
```

58. What is Scope in Javascript.

1. Global Scope :-

• Variables declared outside of any function or block have a global scope.

• Global Variable are accessible throughout the entire program, including inside functions.

2. Local Scope :-

• Variable declared inside a function or a block have a local scope.

• Local variable are only accessible within the function or block when they are declared.

59. What are reserved words? Can I use reserved words as identifiers.

Reserved words in a programming language are words that are predefined and reserved for a specific purpose. These words usually have a special meaning in the language's syntax.

Example :-

```
Var if = 5; // Error : Unexpected token 'if'
```

if you need to choose a name for a variable or function, make sure it's not a reserved word. Most code editors Provide syntax highlighting that can help identify reserved words.

60. Why do you need strict mode? How do you declare strict mode.

Strict mode in javascript is a feature that helps catch common coding errors and prevents the use of certain "bad practices" that might otherwise go unnoticed.

1. Error prevention:

Strict mode catches common coding mistakes and throws errors, which can helps you identify and fix problem early in the development process.

2. Security:-

Some features consider insecure or deprecated are disabled in strict mode, making the code more secure.

61. What are global variables.

Global variable in javascript are variables that are declared outside of any function or block, making them accessible throughout the entire program.

1. Scope pollution:

Since global variables are accessible from anywhere there's a risk of unintentional variable name clashes.

2. Maintainability:

Code that relies heavily on global variables might be harder to maintain and debug.

62. What are the problems with global variables.

1. Scope pollution:

global variables are accessible from any part of the code. If you have many global variables, there's a higher chance of unintentional variable name clashes, making the code.

2. Unintended modification :-

Since global variable are accesible and modifiable from anywhere, its challenging to track where a variable is modified.

3. Code maintainability :-

Code that heavily relies on global variables may be harder to understand, maintain, and debug.

63. What is NaN property.

NaN stands for "Not-a-Number" in JavaScript. It is a special value that represents the result of an operation that should return a valid number but doesn't

1. Mathematical operations:

operations that involve undefined values data types may result in NaN.

2. Undefined & Unrepresentable:

Some operations, like dividing zero by zero, Produce a result that is undefined as a real number.

3. Parsing operations:

Parsing functions like parseInt may return NaN if they encounter characters.

64. What is the purpose of delete operator.

The delete operator in javascript is used to remove a property from an object or to delete an element from an array.

1. Deleting object properties:

you can use delete to remove a property from an object.

2. Deleting Array Elements:

In arrays, delete can be used to remove an element, but it leaves an undefined gap in the array.

65. What is the difference between null and undefined.

1. Undefined :-

- Undefined typically means that a variable has been declared but has not been declared but has not been assigned a value.
- It's important to note that undefined is a primitive value in javascript.

2. Null :-

- Null is a value that represents the intentional absence of any object value.
- Unlike undefined, null is an object.

66. What are the bitwise operators available in JavaScript.

1. Bitwise AND (&):
   - sets each bit to 1 if both bits are

2. Bitwise OR (|):
   - sets each bit to 1 if at least one of the corresp-onding bits is 1.

3. Bitwise Not (~):
   - inverts the bits, changing 1s to 0s and 0s to 1s.

4. Bitwise XOR (^):
   - sets each bit to 1 if only one of the corresponding bits is 1.

5. Left Shift (<<):
   - Shifts the bits to the right by a specified number of positions, filling in with the sign bit.

67. Can I redeclare let and Const variables.

Variables declared with let and Const have blockScope, meaning they exist only within the block, statement, or Expression on which they are defined. Redeclaring a Variable within the same scope violates this rule & leads

```
let myVar = 10;
myVar = 20; // valid.
```

Cannot reassign a value to a Const variable:

```
Const myConst = 10;
myConst = 20; // TypeError:
Assignment to Constant Variable.
```

68. Does Const Variable makes the value immutable.

No, declaring a variable with const in javascript does not make the value itself immutable.

Example:

```
Const myconst = 10;
myConst = 20; // TypeError:
Assignment to Constant variable.
```

However, it's important to note that const applies to the variable binding, not the value it holds. If the value is an object or an array, the properties or elements of that object can still be modified.

69. What is ES6? List down some of the features of ES6.

ES6, also known as ECMAScript 2015, is the sixth edition of the ECMAScript standard, which is the specification that javascript is based on.

1. Arrow Function:-

Shorter syntax for defining anonymous functions.

2. Let and const :

Introduces block-scoped variables with let and Constants with Const.

3. Template literals:

Provides a more Convenient way to concatenate strings using backticks.

10. In what are the possible ways to create objects in javascript.

1. Object literal:

The simplest way to create an object is by using an object literal.

2. Constructor Function:

you can create objects using Constructor functio

3. object.create method:

The object.create() method allows you to creat a new object with a specified prototype object.

4. Factory Function:

A factory function is a function that returns an object.

71. What is the difference b/w slice and splice.

- Slice :-

The slice method is used to extract a portion of an array and create a new array containing the extracted elements.

- Syntax :-

```
array.slice(start, end);
```

2. parameters :-

- Start :- The index at which to start changing the array.
- deleteCount : The number of element to remove from the array.

3. Modification of original Array

- Slice : Does not modify the original array, it creates a new array with selected element.

4. Splice : Modifies the original array by removing replacing or adding element in place.

Return Value :-

- Slice : Returns a new array containing the selected elements
- splice : Returns an arrays containing the removed elements or an empty array if no element were removed

72. What is the difference b/w a. == and === operators

b. = and ==

c. %= and =.

a. == and === Operators:-

• == (Equality Operator)

• Compares value for equality after performing type coercion if necessary.

• === (Strict Equality Operator):-

• Compares values for equality without type coercion. Both the value and the type must be the same for the result to be true.

b. = and ==

• = (Assignment Operator)

• Used to assign a value to a variable.

• Ex:- var x = 5; assign the value 5 to the variable x.

• == (Equality Operator)

• Compares value for equality after performing type coercion if necessary.

• Ex:- x == "5" would evaluate to true if x is 5 because of type coersion.

C. %= and = (Rimainder Assignment Operator and Assignment Operator).

- performs a rumainder operation and assign the result to the variable.
- Ex:- x %=3; is equivalent to x = x %. 3; and updates the value of x to its rumainder when devided by 3.
- = (Assignment Operator):-
- Used to assign a value to a variable.
- Ex y =10; assign the value 10 to the variable.

#3. What is a higher order function.

1. Takes a function as an Argument:-
   - The higher-order function can accept another function as an argument.

2. Returns a function:-
   - The higher-order function can rulurn a function as its rusult.

2. Callback Functions:-
   - Passing function as arguments to be executed later, often asynchronously.

3. Function Composition:-
   - Combining multiple function to create a new function.

- Higher - order - Components :-
- In the context of react, a higher - order component is a function that takes a components and returns a new.

74. What is the currying function?

1. partial App^n :-
You can create specialized versions of a function by applying some of its arguments and leaving the other for later.

2. Reusability :-
Curried functions are more modular and can be reused in different contexts.

75. What are arrow function?

1. Traditional Function Expression :-

```
var add = function (x, y) {
    return x+y;
}
```

2. Arrow Function :-

```
const add = (x, y) => x+y;
```

3. No Arguments Object :-
Arrow function do not have thier own argument object. instead, they inherit it from the enclosing scope.

4. No Binding of super:-
Arrow function do not bind thier own super value.

76. What is spread Operator?
1. The of coping Arrays:-
let originalArray = [1, 2, 3]
let copyArray = [... originalArray]

2. Concatenation Arrays:-
let array1 = [1, 2, 3]
let array2 = [4, 5, 6]
let combinedArray = [...array1, ...array2].

3. Creating Object Copies:-
let originalObject = {key1: "value1", key2 = "value2"}
let copyObject = {... originalObject}.

77. What is rest parameter?
• The sum function use the rest parameter ...numbers to collect all the arguments passed to the function into an array called number.
• The reduced method is then used to sum up all the no. in the array.

Collecting Remaining Arguments:-
The rest parameter collects any remaining arguments into an array. In the eg: no is an array containing all the arguments after the first one.

78. What happens if you do not use rest parameter as a last argument.
Attempting to run this code would result in a syntax error and your script or function won't execute successfully
In this corrected eg:- The rest parameter ...rest is the last parameter and it will correctly collect any remaining arguments into an array. Allow the rule that the rest parameter must be the last formal parameter in the function declaration.

79. What are regular expression patters?
* Literals:-
Regular characters such as letter or no match themselves
eg:- the pattern abc matches the string "abc" exactly.
* MetaCharacters:-
. Special characters that have a specific meaning in regex
* character Classes:-
Square brackets ([ ]) define a character class which matches

any one of the characters within the brackets.

4. Quantifier:-

Specifiq the no of occurrnce of the preuding charictis or group.

80. What is a Rigular Expression?

1. Literals charicters:-

- Rigular charecters, like letter or no, match themselves.

2 Metacharacters:-

- Special charecters with a specific meaning in rugular expressions.

3. charactis Classes:-

Enclosis in square bracket [ ], charicters classes match any single charictis within the brackets.

81. How do you search a string for a patterns.

1. The string text contains the text you want to search.

2. The rugular expression pattern is /fox/ which will match the substring "fox" in the text.

3. The test() method of the rugular expression is used to chick if the pattern is present in the text

82. What is the purpose of switch-case?

1. Fall-through:-
By default each case block ends with a break statement to exit the switch statement.

2. Expression:-
The value inside the parentheses of the switch statement is the expression being evaluated.

3. Default Case:-
The default case is optional and provides a block of code to execute when none of the cases match.

4. Strict Equality:-
Switch uses strict equality (===) for comparision so type and value must match for a case to be considered a match.

83. What are the conversations to be followed for the usage of switch case.

1. Indentation:-
Indent case statements inside the switch block. This enhance code readability and makes it clear which code belongs to each case.

2. Break Statements:-
Use break statements to exit each case block. This prevents fall-through to subsequent cases. If fall-through behaviour is intentional, make sure to comment and document it clearly.

3. Default Case:-
Include a default case, even if its empty. This helps handle unexpected values and provides a clear default behaviour.

4. Consitent Formatting:-
Maintaining a consistent and clear formal for switch statements throughout your codebase.

84. What are the different ways to access object properties.

1. Dot Notation:-
• Use the Dot (.) notation to access properties if you known the property name at development time.

2. Bracket Notation:-
Use square bracket ([]) and a string containing the property name to access property.

3. Computed property names:-
If using ES6 or later you can use computed property names to dynamically access properties.

85. What are the function parameter rules.
1. parameter Declaration:-
Declare parameters inside the parentheses of a function definition.

2. parameter Naming:-
Choose meaningful names for parameters that reflects thier purpose.

3. Default Values:-
ES6 introduced default parameter values, allowing you to specify a default value if an argument is not provided

86. Different ways which create infinite loops?
1. Missing Break statement:-
Forgetting to include a break statement in a loop can result in the loop running indefinitely.

2. Incorrect Loop Conditions:-
. Using loop conditions that never evaluated to false will result in an infinite loop.

3. Improperly Implemented Recursion:-
Recursion functions can accidentlly become infite if the case is not properly defined or if the recursion call does not approch the base case.

87. What are default values in distructuring assignment?
1. If the property or element is present in the source object or array, the variable receives its value.
2. If the property or element is not present or is undefined the variable receives the specified default values.

88. What are template literals?
1. String Interpolation:-
Expression inside ${} within the backticks are evaluated and thier result are inserted into the string.
2. Multiline Strings:-
Template literals make it easy to create multiline string without the need for concatenation or escaping newlines.
. Expression Evaluation:-
you can include any valid js expression inside ${} within template literals.

89. How do you swap variables in disturing assignment?

- Swapping variables with array Destructing.

```
let a = 5
let b = 10
[a, b] = [b, a]
Console.log(a) => 10
Console.log(b) => 5
```

① [b, a] creati an array with the values of b & a in reversed order.
② [a, b] = [b, a] distructuring the array assignment the value of b ~~to~~ a and the value of a to b, effectively swapping thier values.

- Swapping variablis with object Distructring.

```
let a = 5
let b = 10
({a, b} = {b, a});
Console.log(a) => 10
Console.log(b) => 5
```

① {b, a} cratis an object literal with properlis b & a.
② ({a, b} = {b, a}) distructuring the object, assigning the value of b to a & the value of a to b.

90. What are primitive data-typis?

1. String:-
Riprisents sequnis of charicturs and is enclosid in single (' '), doble (" "). or back-tick (` `) quotes.

2. Numbers-
Rip numeric valus, including integers and floating point numbirs.

3. Boolean :-

Rep a logical value, either true or false.

4. Undefined :-

Rep an unitialized or undefind value.

5. Null :-

Rep the intentional absence of any object value.

91. Is that possible to use expression in Switch cases ?

No its not possible to use expressions directly in switch cases in js. The switch statiment is designed to evaluate a single expression against different constant values. Each case in a switch statiment must be a literal values or a constant expression.

92. What are the difference between for ... of and for in statiment ?

for of :-

for of is specifically designed for iterating over iterable objects such as array, string, set, maps and other objects the implument the iterable protocol.

2. values:-
It directly iterates over the values of the iterable, providing the actual values rather than the indices.

for in:-

1. Enumerable properties:-
for in iterates over the enumeric properties of an object, including properties in its prototype chain.

2. Keys / Indices:-
It provides accss to the keys or indices rather than the values. In the ex above, it logs property names and not thier corresponding values.

93. What are the differnces between arguments object and rest paramters?

1. Availability:-
The arguments object is available in all function scope, regardless of whether the function scopes, regardless of whether the function is declared with parameters.

2. Not an Actual Array:-
The arguments object is not a real array, it is

an array like object. It does not have array methods. like forEach, map etc.

3. No named Parameter :—

arguments includes all parameters passed to the function whether named or not. It doesnot respect the named parameter defined in the function signature.

94. What are the diffrence blw spread operator and rest operator?

1. Usage :—

The spread operator is used to split an array or object into individual elments or properties.

1. The rest operator is used in function parameter to to collect multiple arguments into a single array.

2. It can be used to spread elments of an array or properties of an object in function arguments

2. It gathers all remaining arguments into an array.

95. Explain all the array method, what are the outputs and whether the method modifies the original Array.

1. push() Method :— Add one or more element to the end of an array.

2. pop() Method:-
Removes the last elements from an array.

3. UnShift():-
Adds one or more elements to the beginning of an array.

4. Shift():-
Removes the first element from an array.

5. concat():-
Combines two or more arrays.

6. Slice():-
Returns a shallow copy of a portion of an array into a new array.

7. Splice():-
changes the contents of an array by removing or replacing existing elements.

8. forEach():-
Executes a provided function once for each Element array Element.