

1) What is HTML & what is its purpose?

HTML is the code that is used to structure a web page and its content. Basically, it tells the user's computer what things are. It also provides access to an incredible amount of functionality that's already built directly into the browser.

2) What is the difference b/w HTML & XHTML?

HTML

1) HTML stands for Hypertext markup language

2) It was developed by Tim Berners-Lee

3) It was developed in 1991

4) It is extended from SGML

5) The format is a document file format

6) What are the new features introduced in HTML5?

The new features are

semantic elements

audio & video support

canvas elements

Geo locations API

Local storage

Responsive images

Web workers

Drag & Drop API

XHTML

1) XHTML stands for Extensible Hypertext Markup language

2) It was developed by W3C

3) It was released in 2000

4) It is extended from XML & HTML

5) The format is a markup language

4) How do you include comments in HTML?

→ In HTML a comment is a section of text that is not processed by the web browser. Comments are enclosed in `<!-- -->` tags. These tags tell the browser that the text inside should be a comment & should not be rendered on the front end.

5) Explain the difference b/w `<div>` & `` tags

`<div>`

``

1) The `<div>` tag is a block level element

1) The `` tag is an inline element

2) It is best to attach it to a section of a web page

2) It is best to attach a CSS to a small section of a div in a web page

3) It accepts align attribute

3) It does not accept align attribute

4) This tag should be used to wrap a section for highlighting that section

4) This tag should be used to wrap any specific word that you want to highlight in your website webpage.

6) What are semantic elements in HTML5 & why are they important?

→ A semantic element describes to both the browser & the developer. Ex of non-semantic elements

`<div>` & `` tells nothing about its content. Ex of semantic elements `<Form>` `<table>` & `<article>` clearly defines its content.

The semantic HTML tags help the search engine & other user device to determine the importance & context of web pages.

7) What is the purpose of `<header>`, `<nav>`, `<section>` & `<footer>` tags in HTML 5?

→ The `<header>` tag specifies the introduction or a group of navigation elements for the document.

The `<header>` tag is supported in all major browsers.

The `<footer>` tag is supported in all major browsers.

The `<footer>` tag is present only in HTML5, it gives additional information copyright, author of document, privacy, contact.

The `<nav>` HTML element represents a section of a page whose purpose is to provide navigation links, either within the current document or to other documents.

The `<section>` tag divides the content into sections & subsections. The section tag is used when requirements of two headers or footers @ any other section of document are needed.

8) How do you create a hyperlink in HTML?

→ To make a hyperlink in an HTML page we use `<a>` & `` tags, which are the tags used to define the stroke. The `<a>` tag indicates where the hyperlink starts & `` tag indicates where it ends. Whatever text gets added inside the tags will work as a hyperlink.

9) What is the diff b/w `` & `` elements?

→ `ol` tag is used to create an ordered list of items but `ul` tag is used to create an unordered list of items in the HTML.

10) How do you embed an image in HTML?

→ The HTML `` tag is used to embed an image in a web page. Images are not technically inserted into a webpage; images are linked to web pages. The `` tag creates a holding space for the referenced image.

11) Explain the difference between the `` & `` tags

→ Adding to the confusion is the fact that while HTML 4 defined `` as indicating a stronger emphasis, HTML 5 defines `` as representing "strong importance" for its contents.

While `` is used to change the meaning of a sentence as spoken emphasis does ("I love carrots" vs "I love carrots"). `` is used to give portions of a sentence added importance (e.g., "warning! This is very dangerous.") Both `` and `` can be nested to increase the relative degree of importance.

12) How do you create a table in HTML?

→ An HTML table is created with an opening `<table>` tag & a closing `</table>` tag. Inside these tags data is organized into rows & columns by using opening & closing `<tr>` tags & opening & closing `<td>` data `</td>` tags. Table row `<tr>` tags are used to create a row of data.

- 13) what is the purpose of the <form> tag in HTML & how do you create a form?
- the HTML form tag is required when you want to collect information that visitors provide.
- A complete guide to HTML Form tags
- 1) The <form> tag is essentially a container for inputs
 - 2) The <input> element. The HTML <input> is a fundamental form elements
 - 3) The <label> Element. A Label can be assigned with a <label> element -
 - a) multiple-line text input control
 - b) HTML form example.
- 14) what are some new input type introduced in HTML5?
- HTML5 introduced several input types like date, datetime, datetime-local, time, week, month, email, tel, url, search, range, number and numbers to improve user experience & to make our forms more interactive.
- 15) How do you include audio & video content in HTML?
- HTML, Add <audio> & <video> elements to the page; make them display the default browser controls. give both of them <source> elements so that browsers will find the audio format they support best & load it. These should include type attribute.

16) What is the purpose of the <iframe> tag & how it is used?

→ The <iframe> tag specifies an inline frame. An inline frame is used to embed another document within the current HTML document. You can define an inline frame with HTML tag <iframe>, the <iframe> tag is not somehow related to the <frame> tag, instead it can appear anywhere in your document.

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

→ Internal CSS is placed inside the <style> element which goes inside the <head> of the HTML document. External CSS exists in a separate file called an external stylesheet. It requires a <link> element placed inside the head section of an HTML file.

18) What is the role of the alt attribute in tag?

→ The required alt attribute specifies an alternate text for an image. If the image cannot be displayed, the alt attribute provides alternative info for an image if a user for some reason cannot view it.

19) How do you create a numbered list with custom numbering style in HTML?

→ A list is a connected items which consecutively. Using HTML, you can create two kinds of lists: unordered list & ordered list.

An ordered list is marked with the numbers by default.

You can create an ordered list using `` tag & define list items using `<i></i>`.

Q1) What is responsive web design, & why it is important?

→ RWD involves creating web pages that render well across a variety of devices & screen sizes. In this way, web designers are able to create a website experience that accommodates to many different ways users access & interact with websites.

Q2) How do you make a website responsive using CSS?

→ set your media query ranges
size layout elements with percentages
implement responsive images.

Responsive typography for your website text.
Test responsiveness.

Q3) What is a media query in CSS, & how is it used for responsive design?

→ media queries allow you to apply CSS styles depending on a device's general type (such as print vs screen)
① other characteristics such as screen resolution
media queries are used for conditionally apply styles with the CSS @media & @import at-rules.

23) Explain the diff b/w a fluid layout & a fixed layout in terms of responsiveness?

→ In a fixed layout ~~the~~ publisher sees what the readers get.

However, in the case of fluid layout, the publisher has less control over what the user sees & so they may overlook certain aspects simply because the page looks fine on their specific screen resolutions.

24) How do you make images responsive in CSS?

→ To make an image responsive, you need to give a new value to its width property. Then the height of the image will adjust itself automatically.

25) What are breakpoints in responsive design, & how are they determined?

→ A breakpoint is a point, usually a specific width, at which a webpage's style is adapted in a particular way in order to provide the best possible user experience.

They are determined by two concepts

a) Breakpoints based on device

b) Breakpoint based on content.

96) How can you hide elements on specific screen
size using CSS?

→ To hide element in a responsive layout, we need to set the CSS display property set to its "none" value along with the @media rule. The content of the second `<p>` elements having a "hidden-mobile" class will be hidden on device smaller than 767px.

27) What is the purpose of the max-width property in responsive CSS?

→ The max-width CSS property sets the maximum width of an element. It prevents the used value of the width property from becoming larger than the value specified by max-width.

28) How do you create a responsive navigation menu using CSS?

→ A div container having a class `nav` acts as a container for our navbar functionality using anchor tags.

A heading & a paragraph element in active class highlights the active link.

29) Explain the concept of mobile-first design & how it relates to responsive CSS.

→ mobile first means designing for mobile before designing for desktop or any other device (this will make the page display faster on smaller devices). This means that we must make some changes in CSS.

30) what is CSS flexbox, & what problem does it solve?

→ CSS flexbox was designed as a one-dimensional layout model, & as a method that could offer space distribution b/w items in an interface & powerful alignment capabilities.

CSS flexbox layout allows you to easily format HTML. Flexbox makes it simple to align items vertically & horizontally using rows & columns.

31) Explain the difference b/w flex containers & flex items.

→ Flex items are the direct children of flex containers, A flex container (the large yellow area in image) is an HTML element whose display property's value is flex or inline-flex.

32) How do you create a flex-container in CSS?

→ You create a flex container by setting the display property of an element to one of the flexbox layout values.

33) What is the purpose of the flex-grow, flex-shrink & flex-basis properties?

→ The flex-grow property specifies how much the items will grow relative to the rest of the flexible items inside the same container.

The flex-shrink property specifies how the items will shrink relative to the rest of the flexible items inside the same container.

The flex basis property sets the initial main size of a flex item. It sets the size of the content box unless otherwise set with box-sizing.

24) How do you align flex items horizontally & vertically with in a flex container?

→ set the display of an "big box" to "flex" & add align-items property to specific the vertical alignment of contents inside the flex container. Also use the -webkit-extension. Set the justify-content property of an "big box" to "center" to align the content horizontally.

25) Explain the difference b/w justify-content & align-items properties in flexbox.

→ An align-items effect the alignment of items on the current line

align-content effects the alignment across lines of a flex container. This means that this property has no effect on single line containers.

26) How can you control the order of flex items using CSS flexbox?

→ New layout method such as flexbox & grid bring with them the possibility of controlling the order of content. We will take a look at ways in which you can change the visual order of your content when using flexbox.

37) what are flexbox break points, & how can they be used for responsive design?

→ Responsive designs can largely be achieved using breakpoints & flexbox at implementation time. But it's not immediately clear how to prototype them by hand in Figma. I found a couple of plugins, the Breakpoints plugin & the auto layout plugin.

38) What are HTML attributes?

→ An HTML attribute is a piece of markup language used to adjust the behaviour or display of an HTML element.

39) How do you add attribute to an HTML element?

→ HTML DOM element setAttribute()

1. Ex. Add a class attribute to an element.

setAttribute("class", "demoClass"); --

2. Change an input field to an input button: myInput.setAttribute("type", "button"); Before: --

3. Add href attribute to an <a> element: myAnchor.setAttribute("href", "https://www.w3.com"); Before

40) What is the purpose of the id attribute in HTML & how it is unique?

→ The id attribute specifies a unique id for an HTML element. The id attribute is most used to point to a style in a style sheet, & by JavaScript to manipulate the elements with specified id.

41) What is the difference b/w the class attribute & the id attribute?

Id

Class

Syntax	In HTML, for an element the ID name starts with # symbol followed by unique name assigned to it	"class" assigned to an element has its name start with ". " followed by class name
selector	Only one ID selector can be attached to an element	multiple class selector can be attached to an element
uniqueness	ID is unique in a page & can only apply to at most one element	The class can be applied to multiple elements so it could be multiple times on a single page

42) Explain the role of href attribute in HTML, particularly in the context of links & anchors.

→ The href attribute specifies the URL of the page link goes to. If the href attribute is not present, the <a> tag will not be a hyperlink.

43) How do you add alternative text to an image using the alt attribute?

→ To add an text to picture, shape, chart, or image as graphic, right click on the object choose format picture. In the format picture panel, choose the layout of properties icon. Then choose ALT text. Add a title for your object then a description.

44) What is the purpose of the target attribute in HTML links & what are its possible values?

→ The target attribute specifies a name or a keyword that indicate where to display the response that is received after submitting the form. The target attribute defines a name or keyword for a browser context.

45) How do you use the src attribute to embed an external resource, such as an image or video, in HTML?

→ The HTML <input> src attribute is used to specify the URL of the image to be used as a submit button. The attribute can only be used with <input type="image">, Attribute value: it contains a single value URL that specifies the link of the source images.

46) What is the purpose of the disabled attribute & how it is used in HTML form elements?

→ The disabled attribute is a boolean attribute, when present, it specifies that the element should be disabled. A disabled element is unusable. The disabled attribute can be set to keep a user from using the element until some other condition has been met.

47) Is there any relation b/w Java & Javascript

→ Javascript has no direct relation to Java besides being used for web technologies.

48) Is Javascript a compiled or interpreted language
→ Javascript is an interpreted language & does not require to compile before execution.

49) Is Javascript a case-sensitive language

→ All Javascript identifiers are case sensitive.

50) What is node.js?

→ Node.js is easily employed as a server-side proxy, where it can handle a large amounts of simultaneous connections in a non-blocking manner.

51) What is the difference b/w let & var

→ The diff b/w let & var is in the scope of the variables they create; variables declared by let are only available inside the block where they defined. Variables declared by var are available throughout the function in which they are declared.

52) What are the diff b/w undeclared & undefined variable

undeclared	undefined
1) There are the variables that do not exist in our memory map	1) These variables are ones that do exist in memory but
2) The variables are considered to be undeclared because of programs do not withdraw with var, let or const	The variables are considered to be undefined because it is assigned by Javascript to them
3) If we try to access them in the code execution phase then Javascript will throw a reference error.	If we try to access these variables we'll get the undefined as value.

53) What is hoisting?

→ Hoisting is javascript's default behavior of moving all declarations to the top of the current scope.

54) what is scope in Javascript?

→ In Javascript, scope refers to the context context of your code, this context determines where you can access certain variables & functions.

55) what are reserved words? can we use reserved words as identifiers

→ A reserved word is one that looks like a normal word but is not allowed to be used as a normal word.

reserved words cannot be used as identifiers.

56) why do you need strict mode? how do you declare strict mode.

→ Strict mode makes it easier to write "secure" Javascript, strict mode changes previously accepted "bad syntax" into real errors.

strict mode is declared by adding "use strict;" to the beginning of a script as a function.

57) what are global variables?

→ In computer programming, a global variable is a variable with global scope meaning that it is visible throughout the program, unless shadowed.

58) What are the problems with global variables?

→ Global variables are subject to a number of problems: violates information hiding, negates the open-closed principle, introduces implicit coupling & side effects.

59) What is NaN property?

→ The NaN property represents a "not-a-Number" value. This property indicates that a value is not a legal number.

60) What is the purpose of delete operator?

→ The delete operator removes a given property from an object. On successful deletion, it will return true, else false will be returned.

61) What is the difference b/w null & undefined?

→ Some of significant diff b/w null & undefined are

a) Javascript always assigns undefined to indicate the absence of a value by default & it never automatically assigns null as a value. A variable will only return null if a programmer or an API specifically assigns the value ~~null~~ null to variable.

b) Although both null & undefined are primitive values in javascript because of a historical error & transpired, typeof(null) returns "object", on the other hand, typeof(undefined) is undefined.

c) Even though null & undefined are loosely equal they are not strictly equal.

62) What are the bitwise operators available in Javascript?

→ Bitwise operators treat its operands as a set of 32-bit binary digits & perform actions. However, the result is shown as a decimal value.

63) Can I redeclare let & const variables

variables defined with let cannot be redeclared. You cannot accidentally redeclare a variable declared with let.

64) Does const variable make the value immutable?
→ The const declaration creates an immutable reference to a value. It does not mean the value it holds holds is immutable -- Just that the variable identifier cannot be reassigned.

65) What is ES6? List down some of the features of

- ES6?

→ ES6 stands for ECMAScript 6; ECMAScript is another official name for Javascript

Features of ES6

- 1) let & const keywords
- 2) Arrow functions
- 3) multi line strings
- 4) Default parameters
- 5) Template literals
- 6) Destructuring assignments
- 7) Promises

66) What are the possible ways to create objects in Javascript?

→ There are different ways to create new objects:
create single object, using an object literal, create
a single object with the keyword new, define an
object constructor, & then create objects of the
constructed type.

67) What is the difference b/w slice & splice?

→ slice returns a piece of the array but it doesn't
affect the original array. Splice changes the original
array by removing, replacing & returns the
affected values.

68) What is a higher order function?

→ A higher order function is a function that takes
one or more functions as arguments, or
returns a function as its result.

69) What is the currying function?

→ Currying is the transformation of a function
with multiple arguments into a sequence of
single argument functions.

70) What are arrow functions?

→ The javascript arrow functions is a shorter
way of writing a function expression that was
introduced in ECMAScript 6.

71) what is a spread operator?

→ The spread operator is a new addition to the set of operators in JavaScript ES6. It takes in an iterable (eg an array) & expands into individual elements.

72) What is rest parameter?

→ The rest parameter syntax allows a function to accept an indefinite number of arguments as an array, providing a way to represent variadic functions in JavaScript.

73) What happens if you do not use rest parameters as a last argument?

→ A syntax error is thrown stating that it has to be the last formal argument & the code is not executed.

74) What are regular expression patterns?

→ A regular expression is a pattern that the regular expression engine attempts to match in input text. A pattern consists of one or more character literals, operators or constructs.

75) What are regular expression?

→ A regular expression is a pattern that the regular expression engine attempts to match in input text.

76) How do you search a string for a pattern?
→ compile a string regular expression to a pattern, using compile API method of pattern class
→ matchers (char sequence input) API method of pattern to create a matchers that will match the given string input against this pattern.

77) What is the purpose of switch-case?
→ The switch case in Java executes one statement from multiple ones. Thus it like an if-else-if ladder statement. The switch statement is used to test the equality of a variable against several values specified in the case.

78) What are primitive data types?

A primitive data type is pre-defined by the programming language. The size & type of variable values are specified & it has no additional methods.
Ex:- byte, short, int, long, float, double

79) What are the diff ways to access object properties?
→ You can access the properties of an object in javascript in 3 ways

- 1) dot property accessor: object.property
- 2) square brackets property accessor: object['property']
- 3) object destructuring: const {property} = object

80) What are the function parameter rules.

→ Javascript function definitions do not specify data types for parameters.

Javascript functions do not perform type checking on the passed arguments

They do not check the number of arguments received received

81) Different ways which create infinite loops

→ we can create an infinite loop using while Statement. If the condition of while loop is always true, we get an infinite loop.

82) what are template literals

→ Template literals are literals delimited with back tick (`) characters allowing for multiline strings, string interpolation with embedded expressions & special constructs called tagged templates.

83) What are default values in destructuring assignment

→ we can follow the syntax below to set the default value when destructuring objects in javascript

```
const {prop1, prop2 = default_value, prop3 = default_value}
```

= {prop1: value1, prop2: value2}; In the above syntax we destructuring the object in to prop1, prop2 & prop3 variables.

84) How do you swap variables in destructive assignment

→ $[a, b] = [b, a]$ is a destructive assignment that swaps the variables a & b . At the first step on the right side of the destructuring, a temporary array $[b, a]$ (which evaluates to $[2, 1]$) is created. Then the destructuring of the temporary array occurs; $[a, b] = [2, 1]$.

85) Is that possible to use expressions in switch cases?

→ The switch case in Java works like an if-else ladder i.e. multiple conditions can be checked at once. switch is provided with an expression that can be a constant or literal expression that can be evaluated.

86) What are the diff b/w for... of & for... in statements

→ The main difference b/w them is for what they iterate over. one for... in statement iterates over the enumerable string properties of an object. while the for... of statement iterates over values that the iterable object defines to be iterated over.

87) what are the differences b/w arguments object & rest parameters.

→ There are three main differences b/w rest parameters & the arguments

object : the argument object is not a real array, while rest parameters are Array instances, meaning methods like `sort()`, `map()`, `forEach()` & `pop()` can be applied on it directly.

88) what are the diff b/w spread operator & rest parameters.

→ The main diff b/w rest & spread is that the rest operator puts the rest of some specific over-supplied values into a javascript array. But the spread syntax expands iterables into individual elements.

89) Explain all the array methods, what are the outputs & whether the method modifies the original array

→ Javascript array methods

1. map() :- This method creates a new array with the results of calling a provided function on every element in a given array.

2 filter() :- This method creates a new array with only elements that passes the condition inside the provided function.

3 sort() :- This method is used to arrange/sort array's elements either in ascending Ⓛ descending Ⓜ orders.

4 forEach() :- This method helps to loop over array by executing a provided callback function for each element in an array.

5 concat() :- This method is used to merge two Ⓛ more arrays & returns a new array, without changing the existing arrays.

6 every() :- This method checks every element in the array that passes the condition returning true or false as appropriate.

7 some() :- This method checks if at least one element in the array that passes the condition returning true Ⓛ false

8 includes() :- This method checks if an array includes the element that passes the condition returning true Ⓛ false

9 join() :- This method returns a new string by concatenating all of the array's elements separated by the specified separator

10 reduce() :- This method applies a function against an accumulator & each element in the array to reduce it to a single value.

11 find() :- This method returns the value of the first element in an array that pass the test in a testing function.

12 findIndex() :- This method returns the index of the first element in an array that pass the test in a testing function.

13 indexOf() :- This method returns the index of the first occurrence of the specified element in the array.

14 fill() :- This method fills the element in an array with a static value & returns the modified array.

15 slice() :- This method returns a new array with specified start to end elements.

16 reverse() :- This method reverse an array in place. Element at last index will be first & elements at 0 index will be last.

17 push() :- This method adds one or more elements to the end of array & returns the new length of the array.

18 pop() :- This method removes the last element from the end of array & returns that element.

19 shift():- This method removes the first element from an array & returns that element

20 unshift():- This method adds one or more elements to the beginning of an array & returns the new length of the array.

~~21~~ mutating methods are array methods that modify the original array, while non-mutating methods return a new array without modifying the original
~~push()~~ push() = Adds one or more elements to the end of an array & returns the new length of the array, pop() = removes the last element from an array & returns it.