

STUDENT'S NAME	
CLASS	SUBJECT
ROLL NO.	DATE

Q) What is HTML and what is its purpose?

- HTML stands for HyperText Markup Language.
- It is the standard markup language for creating web pages. HTML describes the structure of a web page by using a series of elements. These elements tell the browser how to display the content of the page.

#### Purpose -

- To identify different types of contents on a web page.
- To specify the formatting of the content.
- To create link to other web page.

Q) What is the difference between HTML & XHTML?

→ HTML (HyperText Markup Language) and XHTML (Extensible HyperText Markup Language) are both markup languages used for structuring and presenting content on the web.

The difference are :-

① Syntax : HTML has more forgiving syntax, allowing for certain sloppy or incorrect coding practices. On the other hand, XHTML follows stricter rules and is based on XML syntax, which requires well-formed and properly nested tags.

② Case sensitivity : HTML is generally case-insensitive when it comes to element and attribute names. XHTML, however, is case-sensitive, requiring all tags and attributes to be written in lower case.

③ Document structure : In HTML, the document structure is often more relaxed, allowing for missing closing tags and other inconsistency. XHTML, being based on XML, enforces a strict structure where all tags must be properly closed & nested.

**ARUN'S**

**MAX**

Compatibility :- HTML is widely supported by web browsers including older versions, which may be more tolerant of errors. HTML, being strict, may encounter issues with older browsers that do not conform XML standards.

Q) What are the new features introduced in HTML5?

→ ① Semantic elements :- HTML5 introduced a set of semantic elements that provide more meaningful structure of web developments. These elements include <header> <footer> <nav> <article> <section> <aside> <main> and many more. They improve accessibility, SEO and code readability.

② Canvas :- HTML5 introduced the <canvas> element, which provides a 2D drawing API. It allows developers to dynamically draw graphics, animations, and visualizations using JS.

③ Form Enhancement : HTML5 introduced several new input types and attributes to enhance form handling. Examples :- <input type="email">, <input type="date">, <input type="range">. Additionally <datalist> allows for the creation of pre-defined options for input fields..

④ Geolocation : HTML5 introduced support for geolocation, enabling websites to access a user's geographical location through JS.

4) How do you include comments in HTML?

→ In HTML, we can include comments to add explanation to the code.

• <!-- Comment -->

STUDENT'S NAME	
CLASS	SUBJECT
ROLL NO.	DATE

Q) Explain the difference between `<div>` and `<span>`.

Ans. The key difference between `<div>` and `<span>` lies in their default display behavior and the scale of the content they are typically used to group.

`<div>` is a block-level element, meaning it takes up the full width available and creates a line break before and after its content by default.

`<span>`, on the other hand, is an inline-level element that does not create line breaks and is typically used to target smaller sections of text.

Q) What are semantic elements in HTML5 and why they are important?

Ans. These elements provide additional context and information about the content, making it more understandable to both humans and machines.

Importance:

① Accessibility →

② SEO → SEO rely on the structure and semantics of the HTML to understand the content of web page.

By using semantic elements we can give search engines with more accurate info about the content.

③ Code readability

④ Maintainability.

⑤ CSS styling and selectivity → `<div> class / style` It allows more specific and targeted CSS selection, as we can style or target specific selection based on their semantic meaning.

- 7) What is the purpose of the `<header>`, `<nav>`, `<section>` & `<footer>`
- `<header>` → Represents the introductory or navigational section, consist of branding, site title, logo. It often appears at the top of web page.
  - `<nav>` → Represents a section of a document that contains navigation links. It is used to create menus, navigation bars, or other navigation elements that help users navigate through different sections of page.
  - `<section>` → Used to divide the content into groups. It includes heading, para, imgs, lists.
  - `<footer>` → Represents footer section of doc. contain info such as copyright notice, contact details.

- 8) How do you create a hyperlink in HTML?
- We can use `a` element which allows to create clickable links that navigate to another page, a specific `<div>` section within page.  
`href`: The attribute specifies the URL or destination that the link should point to. It can be absolute or (`http://ex.com`) or relative URL (`page.html`).  
OR we can use `id` to navigate within page.  
 $\text{href} = \text{id}$  of section.

- 9) What is the difference b/w `<ol>` and `<ul>`?
- Ordered list: The `<ol>` element is used to create a numbered or ordered list. Each element in list `<li>` within the `<ol>` is automatically assigned a number or symbol to indicate its position.
  - Unordered list: The `<ul>` element is used to create a bulleted or unordered list. The `<li>` within `<ul>` are typically represented by bullet points.

18) How do you embed images in HTML?

- To embed an image in HTML, we can use `<img>` tag.  
The `<img>` tag has two required attributes:
  - src: This attribute specifies the URL.
  - alt: provides an alternative text.

19) Explain the difference between the `<strong>` and `<em>` tags.

- The `<strong>` is used to indicate text that is of strong importance. This could be something like warning, a head line or a technical term. The `<strong>` is typically displayed in bold.

The `<em>` is used to indicate text that is emphasized.

The `<em>` will be displayed in italic.

20) How do you create a table in HTML?

- We can use the `<table>` tag. The `<table>` tag is a container for the table row and cells.

Inside the `<table>` tag we need `<tr>` (table <sup>row</sup> ~~head~~) to write headers and create cells and inside `<tr>` we use `<th>` (table head) to give the header of the table columns and we use `<td>` (table data) to give data to the cells below the headers.

We can also need `<thead>` `<tbody>` and `<tfoot>`.

`<thead>`

`<tr>`

`<th></th>`

`</tr>`

`</thead>`

`<tbody>`

`<tr>`

`<th></th>`

`</tr>`

**ARUN'S**

`</tbody>`

STUDENT'S NAME

CLASS

SUBJECT

ROLL NO.

DATE

~~Q. let a = "String"  
 let b =  
 console.log(a + b); → String 1.  
 console.log(typeof a);~~

~~Q. const item = {  
 a: 1,  
 b: 2,  
 c: 3}~~

Q. What is the purpose of <form> tag and how to create it?

→ The <form> tag in HTML is used to create a form. A form is a way for users to input data into web page. The data entered into the form is sent to a server.

The form tag has following attributes:

- \* action: This attribute specifies the URL of some side script that will process form.data.

- \* method: This attribute specifies the method that will be used to send the form data to server. The possible values are get and post.

- \* enctype: This attribute specifies the encoding type that will be used to send the data.

The <input> tag is used to create an input field in a form. We can use the attributes like

type → specifies the type of input. text, password, checkbox, radio, select, textarea, button.

name → specifies name of input area. Used to identify data.

placeholder → specifies a placeholder text that is displayed in the input field when the field is empty.

value → specifies the initial value of the input field.

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

① Email :

② Search → Create search boxes.

- \* The main difference between text and search is how the browser styles it. The search is displayed with rounded corners, they also sometimes display an 'x' mark which will clear the field when clicked.

③ tel → Used to type phone numbers.

Creates input field for phone numbers.

- \* When clicked on the field the dynamic keyboards will display numeric keyboards.

④ url → Create a field to enter url.

⑤ number → Create a field to enter number and also provides buttons to increase or decrease number.

⑥ range → Create a slider. We can see this quite often on sites like home-buying where we want to set a max property price to filter by. We can give min & max values.

⑦ date time localtime : To pick date and time.

⑧ month

⑨ week.

15) How do you include audio and video in HTML?

To include audio and video we can use `<audio>` and `<video>` tag.

The `<audio>` tag has ~~more~~ attributes like.

- \* `src` : specifies the url of audio file.
- \* `controls` : specifies whether the audio player should be displayed.
- \* `autoplay` : specifies whether the audio should start playing automatically.
- \* `loop` : specifies whether the audio should loop.

STUDENT'S NAME	
CLASS	SUBJECT
ROLL NO.	DATE

the `<video>` tag has following attributes:

- \* `src`: specifies url.
- \* `controls`: specifies whether video player should be displayed.
- \* `autoplay`:
- \* `loop`
- \* `width`:
- \* `height`:

- Q) What is the purpose of `<iframe>`, and how is it used?
- The `<iframe>` in HTML is used to embed another HTML document in the current document. This can be useful for displaying content from another website.
- To embed another documents we use some attribute.
- \* `src`:
  - \* `width`:
  - \* `height`:
  - \* `frameborder`: specify whether the border should be displayed around the frame.
  - \* `scrolling`: specifying whether scrollbars should be displayed.

- Q) How to add CSS styles to HTML elements?

→ There ways:-

Inline CSS:- This is the simplest way to add CSS styles.

We can add inline CSS styles of HTML elements using the `style` attribute.

Internal CSS:- This is the way to add CSS styles to HTML elements that is more organized than inline CSS. We can add internal CSS styles by using the `<style>` element in the `<head>` section.

External CSS:- This is the most flexible way to add CSS styles to HTML elements. We can add external CSS styles by using the `<link>` element in the `<head>` section.

STUDENT'S NAME	
CLASS	SUBJECT
ROLL NO.	DATE

18)

What is the role of alt attribute in `<img>` tag.

→ The alt attribute in `<img>` is used to provide alternative text for `img`. The text is displayed if the `img` cannot be displayed for some reason.

The alt attribute is also used by search engines to index `img`. This means that if a user searches for an image that is relevant to the alt text,

19). How do you create a numbered list with custom numbering list styles in HTML.

Two types:

→ Using the `<ol>` tag : The `<ol>` tag is used to create an ordered list. The `<ol>` has an attribute called 'type' that can be used to specify the numbering styles. like `i, a, A, 1, and I`

→ Using CSS : We can use CSS to customize the numbering of an `ol`. we can give some properties like

`list-style-type: upper-roman, lower-roman, decimal, etc.`

20) What is difference between `<script async>` and `<script defer>`?

→ The main difference between `<script async>` and `<script defer>` is when the JS files are executed. The `<script async>` attribute tells the browser to execute the JS file as soon as it loaded. The `<script defer>` attribute tells the browser to execute the JS file after the rest of the page has loaded.

→ If we want to load the JS file as soon as possible for ex:- to initialize chat widget. we use `<script async>`.

→ If we want lazy loading we use `<script defer>`.

ARUN'S

MAX

21) What is responsive web design and why it is important.

→ Responsive web design is a web design approach that makes web pages render well on a variety of devices and screen sizes. RWD uses flexible layouts; media queries to adapt the layouts of web pages to specific device.

Responsive web design is important

① It ensures that the website is accessible to users of all devices.

② It improves website SEO. Search engines like Google now take into account the mobile-friendliness of websites when ranking them in search results. RWD → Search rank.

③ Improve website performance. RWD are smaller in size than non RWD which means it will load faster.

22) How do you make website responsive using CSS?

→ ① Use CSS media queries: Media Queries allows us to apply diff styles based on the device or screen size. By defining specific breakpoints, we can target diff devices and adjust the layout accordingly.

② Create fluid layout: Instead of using fixed pixel values for width and height, we can use relative units such as % or viewport-based units (vw, vh).

③ Using flexible grid system: CSS grid or flexbox are powerful layout tools that help create responsive design.

④ Hide or reorder content :- In some cases, we may want to hide or rearrange certain content on smaller screen to improve UX. using CSS props like, display, visibility or order.

STUDENT'S NAME	
CLASS	SUBJECT
ROLL NO.	DATE

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

- A media query in CSS is a technique that allows us to apply different styles and CSS rules based on specific conditions, such as the characteristics of the device or the view port size. It helps to create responsive designs that adapt and respond to diff screen sizes.

Using this

- ① Adjust layout for diff screen.
- ② Modify screen typography.
- ③ Adopt images and media.
- ④ Customize navigation.
- ⑤ Apply responsive styling.

24). Exp the diff b/w fluid layout & fixed layout in terms of responsiveness

- A fluid layout is inherently responsive as it automatically adjust and fills the available space based on relative units. A fixed layout, on the other hand, maintains a constant size and requires manual adjustments or break points to accommodate diff screen sizes.

25) How do you make images responsive in CSS?

- We use CSS props & techniques
  - ①  $\text{max-width}$  → To make sure the img width should not exceed the width of its container. If it is 100%.
  - ② View port units → Relative to viewport dimension.
  - ③ CSS flexbox or grid →
  - ④ Media queries.

26)

What are break points in responsive design, and how they are determined.

→ In responsive design, break points are specific points or ranges of viewport width where the layout and design of a website need to be adjusted to accommodate different screen sizes.

Determining bp involves considering various factors.

① Device capabilities :- varying screen sizes, resolution and capabilities.

② Content layout & readability :- Analyzing how the website's content is structured and how it behaves on diff screen sizes.

We have to ensure that the content is readable and usable across the device.

③ Design aesthetics :- Considering visual design

④ User behavior.

→

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

→

We can use CSS media query & CSS props.

① Display: none

② Visibility: hidden → Reserve space

③ Opacity: 0 → Transparent

④ Positioning & offsetting → position: absolute or fixed and adjust the 'top' & 'left' values to position an element outside the view port

⑤

27)

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

→ The purpose of max-width in responsive CSS is to set a maximum width for an element. It is commonly used in conjunction with other CSS properties and mediaquery to create responsive design that adapts to diff resizes.

ARUN'S

MAX

STUDENT'S NAME	
CLASS	SUBJECT
ROLL NO.	DATE

\* If the container of element that has given property max-width becomes narrower than the max-width value, the element will shrink to fit into container maintaining its aspect ratio.

2a) How do you create a responsive navigation menu using css.

- ① Create a unordered list inside nav element tag.
- ② CSS Styling
- ③ Media queries

3b) Explain the concepts of mobile first design & how it relate to responsive css.

→ Mobile-first design is an approach to web design and development that prioritizes designing for mobile device first, and then progressively enhancing the layout and functionality for larger screens. It recognises the increasing dominance of mobile devices in web traffic and aims to provide the best user experience on mobile devices, while still ensuring a seamless experience on larger screens.

→ The concept of mobile first design is based on the principle that it is easier to enhance an existing design for larger screen. By starting with a mobile-first approach, designers focus on the essential elements and content that are crucial for a mobile experience.

→ Responsive CSS is closely related to mobile-first design because it enables the implementation of the mobile first approach.

3c) What is CSS flex box, and what problem does it solve?

→ CSS flexbox is a layout module in CSS that provides a flexible way to arrange and distribute elements within a container.

It introduces a set of properties and behaviors that allows for efficient and responsive design of web page layouts.

Using flex box :-

- ① Horizontal & vertical alignment
- ② Equal height columns → equal height columns within a row.
- ③ Dynamic spacing & resizing → enables automatic adjustment of item sizes based on available space.
- ④ Reordering elements
- ⑤ Responsive design.

Q

Diff. b/w flex container & flex items.

→ Flex containers.

It acts as a parent or wrapper for group of flex items. This flex container has property that can control the layout and behavior of its child flex items.

Some commonly used properties are.

- ① flex-direction :- row, row-reverse, column, column-reverse
- ② justify-content :- center
- ③ align-items :- horizontal.
- ④ flex-wrap :- Determined whether flex items should wrap to the next line if they exceed the width of container.  
(default) nowrap, wrap, wrap-reverse

Flex items :-

These are the individual elements contained within a flex container. They are the direct children of the flex container and are affected by its parent props.

- ① flex-grow :- grow when extra space is available.
- ② flex-shrink :-
- ③ flex-basis :- defines the initial size of flex items before any space is distributed.
- ④ Order :- specify the order in which flex items should appear.

ARUN'S

MAX

STUDENT'S NAME	
CLASS	SUBJECT
ROLL NO.	DATE

33) How do you create a flex container in CSS?

- ① Create a container element using `<div>`, `<section>` or `<header>`.
- ② Apply the flex container property: using `display: flex`. This property turns the elements into flex containers.
- ③ We can customize using CSS properties like:
  - flex direction,
  - justify content,
  - align items & `flex-start`, `flex-end`, `center`, `baseline`, `stretch`
  - flex-wrap .
- ④ We can add flex items inside the container element.

34) What are the main properties used to control the layout in flexbox?

- ① display - flex
- ② flex direction → To set the direction of the main axis.  
`row`, `row-reverse`, `columns`, `column-reverse`.
- ③ justify content : controls how flex items are aligned along the main axis.
- ④ flex-start ④ flex-end ④ center ④ space-between ④ space-around.
- ⑤ space evenly
- ⑥ align items : For multiline containers, this property controls how lines of flex items are aligned along the cross axis.

35) How do you specify the direction of flex items within a flex container?

- We can use `flex-direction`, it determines the main axis along which the flex items are laid out.
- ⑦ Set `flex direction` prop on the flex container element.
- That has `display:flex` property.
- ⑧ `row` ⑧ `row-reverse` ⑧ `column`, ⑧ `column-reverse`

36) What is the purpose of the flex-grow, flex-shrink & flex-basis property.

- ① Flex-grow: This property determines how much a flex item should grow relative to other flex items. In same container when there is extra space or available.
- ② Flex-shrink : determines how much a flex item should shrink relative to other flex items when there is insufficient space.
- ③ flex-basis : gets the initial size of a flex item along the main axis before any remaining space is distributed.

37) How do you align flex items horizontally & vertically within a flex container?

- Horizontally aligning flex items:
  - ④ justify-content: align along the main axis (horizontally). 'flex-start', 'flex-end', 'center', 'space-between', 'space-around'.
- Vertically aligning
  - ⑤ align items horizontally vertically.  
'flex-start', 'flex-end', 'center', 'baseline', 'stretch'.

38) Exp the diff b/w justify-content and align-items.

- justify-content is used to align flex items horizontally. It determines how flex items are distributed horizontally and how space is distributed b/w and around them.

align-items : controls the alignment of flex vertically. It determines how flex items are positioned vertically.

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

- We can control the order of flex items using the 'order' property. This prop allows us to change the order of flex items appear within the container regardless of source order.

**ARUN'S**

**MAX**

STUDENT'S NAME	
CLASS	SUBJECT
ROLL NO.	DATE

- We can apply the 'order' prop to the flex items. we want to reorder. By default it is 0.  
item & order : <integer>; }.

Flex items with lower values will appear first and items with -ve will appear first

- What are flex box breakpoints, and how can they be used for responsive design?

→ Flex box breakpoints are specific points or ranges of viewport width where we can adjust the layout of flex containers and flex items to create responsive design using CSS media queries. They are used to accommodate different screen sizes and optimize the display of content on various devices.

How we can use-:

- ① Determining the desired break points.

- ② Defining media queries.

③ media screen and (max-width : 768px) {

  container {

    }

  }

- ④ Modify flex container and flex item properties

- What are HTML attributes?

→ HTML attributes are additional piece of information or settings that can be added to elements. They provide extra details or instructions to the browser about the element should appear or behave. All are specified within opening tag:

42) Explain the diff b/w global attributes and element-specific attributes in HTML.

→ Global attribute:-

- \* There are attributes that can be applied to any HTML element, regardless of type or purpose.
- \* These all provide general functionality and can be used in various elements throughout an HTML doc.
- \* Ex: as class, id, style, title, data, and event related attr like onclick and onload.
- \* These affect the behavior, presentation, or interaction of elements and can be accessed by CSS or JS.
- \* These are not specific to any particular element.

Element-specific attributes,

- \* There are attr that are specific to certain HTML elements and are not applicable to all elements.
- \* Ex: - 'src' and 'alt' for img & 'href' for a.
- \* These are intended to provide specific functionality or define unique characteristics of corresponding elements.

43) How do you add attributes to an HTML element?

→ To add attributes to an HTML element, you we include them within the opening tag of the element.

① Within the opening tag of element we can add attributes we want like name = value.

② We can add multiple attr by separating them with space

44) What is the purpose of the id attribute, and how it is unique?

→ The id attribute serves as a unique identifier for an element. Its purpose is to provide a means to select and manipulate specific elements using CSS or JS. The 'id' allows developers to uniquely identify and reference individual element in doc.

STUDENT'S NAME	
CLASS	SUBJECT
ROLL NO.	DATE

- Q1) Uniqueness :- no two element can have same id.  
 Q2. selection & manipulations. We can target directly.

Q3) diff b/w class & id attributes?

→ Q1) Uniqueness,

id → No two element should have same Id.

class → Elam more elements can share the same class name.

Q2) Purpose :-

id → primarily used to uniquely identify and target specific element.

class → Used to categorize the group elements that has similar characteristics.

Q3) id → we have to give single value id = value.

class → we can give multiple values. class = value1, value2, value3

Q4) Selection:

#id, .classname

Q5) Explain the role of the href attribute in HTML - particularly in the context of links and anchors.

→ href is used to specify the destination URL or the location to which the link should navigate when clicked.

href plays as crucial role in creating hyperlinks and enabling navigation b/w web pages.

\* href can take both absolute and relative URL's

→ relative URL's → "about.html"

→ Absolute URL's → "http://www.example.com".

Q6) How do you add alternative text to an img using the alt attr.

→ Q1. add the 'alt' attribute to the <img> and provide a descriptive text within the attribute value.

a) What is the purpose of the target attribute in HTML links, and what are possible values?

→ The target attribute in HTML links is used to define how the linked URL should be opened when the link is clicked. It specifies the target window in which the linked content should open. It adds behavior to the links, determining whether the URL should open in a new tab, same tab, in a specific named window or in a frame.

Values:

- ① "-self": Open in same browser
- ② "-blank": new tab
- ③ "-parent": Opens in the parent frame. If the current document is nested within a frame set.
- ④ "top": Opens in full body of the window, replacing any frames. If the doc is not nested.
- ⑤ "<window-name>": Opens in specific named window.

b) How do you use the src attribute to embed an external resource, such as an img or video.

→ ① Start with the element we want to use to embed the resource. If we want to embed img we can use `img`.

② Add the src to specify the URL & URL may be relative and absolute.

③ For embedding videos we use `video` and set the src to URL of video file.

c) What is the purpose of disabled attribute, and how it is used in forms?

→ The 'disabled' in HTML is used to disable or make form elements unresponsive, preventing user interaction or input.

STUDENT'S NAME	
CLASS	SUBJECT
ROLL NO.	DATE

① Input elements :-

disabled need provide input tag.

② Selects:-

→ → → → → select tag    <select disabled>  
 option >    <option>  
 </select>

③ Text area:-    <text area disabled>    </text area>

④ Submit & <input type="submit" value="Submit" disabled>

81) Is there any relation b/w Java & JS?

→ No, they ~~base~~ are distinct and have different purpose, syntax, and use case.

82) Is JS a compiled or interpreted language?

→ Interpreted language. When a web page containing JS is loaded in browser, the JS code is executed by browser's JS engine without a separate compilation step.

83) Is JS a case sensitive language?

→ Yes, to declare variable, ~~function~~ we use small camel case.

84) What is node JS?

→ Node.js is an open source JS runtime environment that allows developer to run JS code outside of web browser.

85) What is difference b/w let and var.

Scope:-

let → Block scope

var → Global scope.

### Hoisting:

let: Variables declared with let are not hoisted. This means they are not accessible before they are declared.  
var: Can be hoisted.

### Redeclaration:

let/var can variable with let cannot be redeclared within the same scope.

var → can be redeclared.

Ques) Diff b/w undeclared and undefined variable

→ Undeclared:

- \* An undeclared variable refers to a variable that has not been declared using var, let and const.
- \* If we try to use these, it will result in Reference error.

Undefined:

- \* Refers to variable that has been declared, but no value has been assigned to it.
- \* When a variable has been declared but not explicitly assigned a value, it is automatically initialized to undefined.

Ques) What is hoisting?

→ Hoisting is the mechanism that moves all variables and function declarations to the top of their scope before code execution. This means we can use a variable or function before it is declared, as long as the declaration is within same scope.

Ques) What is scope in JS:

→ Scope refers to visibility of variables and func. A variable or func is only visible within its scope. There are 3 main types of scope.

STUDENT'S NAME	
CLASS	SUBJECT
ROLL NO.	DATE

### ④ Global scope

Variables declared outside of any fn are said to have global scope. They are accessible from anywhere.

### ⑤ Local scope

Variables declared inside a fn are said to have local scope. They are accessible inside that fn.

### ⑥ Block scope

Variables declared inside a block of code are block scoped.

Q1) What are reserved words? Can I use reserved words as identifiers?

Ans. Reserved words are words that have a special meaning. They cannot be used as identifiers.

var var = "foo" // Error: var is a reserved word.

Why? → ① confusing.

② reserved for future use.

Q2) Why do you need strict mode? How do you declare strict mode.

→ Strict mode is a way to write JS code that is more secure & reliable. It does this by enforcing a strict set of rules for how JS code can be written.

Reasons to use strict mode are

① To prevent errors.

② To improve security. → Strict mode prevents us from modifying the argument object.

③ To make code readable.

Q3) What are global variables?

→ There are the variables declared outside of fn. They are accessible from anywhere.

64) What are the problems with global variable.

- • Global variables can be accessed from anywhere in the program, which can make it difficult to track down where they are being used and changed.
- Global variables can be overwritten by other files or scripts, which can cause unexpected behavior.
- They can make our code less modular & reusable.

65) What is NaN property?

- NaN stands for "Not a Number". It is a special value that is returned by JS interpreter when an operation cannot be performed because the operands are not numbers.

66) What is the purpose of delete operator

- Used to delete a property from an object or an element from an array. It can also be used to delete a variable.

67) What is the diff b/w null and undefined?

- Null is a value that can be assigned to a variable. It is used to indicate that the variable does not have a value.

Undefined is a value that is not assigned to a variable. It is used to indicate that the variable does not exist.

68) What are bitwise operators available in JS?

→ & : Bitwise AND

| : Bitwise OR

^ : Bitwise XOR

~ : Bitwise NOT

<< : Bitwise left shift

>> : Bitwise Right shift

**ARUN'S**

Bitwise operators **MAX** are used to perform bitwise operations on numbers. Bitwise operators are performed on the binary representation of numbers.

STUDENT'S NAME	
CLASS	SUBJECT
ROLL NO.	DATE

Q9) Can I redeclare let and const.

→ No.

Q10) Does the const variable makes the value immutable.

→ No declaring a variable with the const keyword does not make the value immutable. The const keyword prevents variable from being redeclared.

Q11) What is ESG? List down some of the features of ESG.

→ ESG also known as ECMAScript 6, is a version of JS. It was released in 2015.

① Classes → for OOPS

② Modules → to organise & reuse codes.

③ Arrow fn.

④ Spread syntax.

⑤ Template literals.

⑥ Default parameters → To provide default values for fn parameters.

⑦ Rest parameter → To capture an indefinite no. of args into an array.

⑧ Enhanced object literals → To add methods & props to objects.

to objects

Q12) What are the possible ways to create objects.

→ Object literals & it is the most common way to create objects. They are created using a comma-separated list of key-value pairs.

const obj = {

  foo: 'bar',

};

New keyword :-

const now = new Date();

The object constructor:-

const obj = new Object();

ARUN'S

MAX

76) Diff b/w slice & splice.

→ Slice : Used to create a new array from ~~comple~~  
copying some part of original array.  
Slice (1, 2)

Splice : Used to add and remove elements from  
existing array.

splice (1, 2, {elements to be added}).

77) What is the difference b/w:

(i) == and === operator.

== → compares values.

==== → compares values & types of

(ii) = & ==

= → used to assign value to a variable.

== → used to compare value.

(iii) %= & =\*

%= → assignment operator that performs a remainder operation.

=\* → simple assignment operator.

78) What is Higher Order fn.?

→ This is the fn. that takes another fn. as an argument or returns a function as a result.

79) What is currying fn.?

→ Currying is a technique in functional programming where a fn. is partially applied to a series of args, producing a new fn. that takes the remaining args as input. This can be useful for breaking down a complex fn. into smaller, more manageable pieces.

STUDENT'S NAME	
CLASS	SUBJECT
ROLL NO.	DATE

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

const curriedAdd

const curriedAdd = Curry(add);

const result = curriedAdd(10)(20)

c.log(result) => 30.

Q2) What are arrow fn.

→ These are new type of fn introduced in JS ES6.

They are a concise way to write anonymous fn.

\* Shorter syntax.

\* Implicit return → need not to write return.

\* Lexical scope. → means they can only see the variables that are declared in the surroundings. This makes arrows fn more predictable & easy to debug.

\* Readability

\* Efficiency

Q3) What are spread operators?

→ The spread operator in JS is a special syntax that allows us to spread the elements of an iterable object into another iterable object.

Q4) What is a rest parameter?

→ A rest parameter in JS is a special type of parameter that can be used to capture an indefinite number of args.

Q5) What happens if we do not use rest parameter as a last argument.

→ The fn will not be able to capture an indefinite number of args; instead, the rest parameter will be treated as regular arguments.

86) What are regular expression patterns.

→ Regular expression patterns in JS are a way to match a specific pattern of text. They are used in a variety of contexts, such as searching for text in a string, validating user inputs, and parsing data.

87) What are regular expression ranges

→ Same as previous

88) How do you search a string for pattern.

→ We can search a string for pattern using string.prototype.match() method. The match() method takes a regular expression pattern as an argument and returns an array of all the matches.

89) What is the purpose of switch case.

→ A switch case statement is a control flow that allows us to execute diff blocks of code depending on values of variables.

90) What are the conventions to be followed for the usage of switch case blocks.

→ Use descriptive case labels :- Case Monday instead of case 1  
Use break statement :- To exit switch  
Use default case.  
Order the case in ascending order.

91) What are primitive data types.

→ These are the most basic data types, that can be stored in variables.

- ① Number ② String ③ Boolean ④ Null ⑤ Undefined
- ⑥ Symbol ⑦ BigInt

STUDENT'S NAME	
CLASS	SUBJECT
ROLL NO.	DATE

Q3) What are the different ways to access object properties.

→ ① Dot notation:

Object name. Key.

② Bracket notation:

Object name ['key'].

Q3) What are fn parameter rules.

→ \* parameter names should be unique.

\* → must be valid variable name

\* parameter types are optional.

\* Default parameter values are optional.

\* Last → → → → →

Q3) Different ways which create infinite loops

→ \* While loop with a condition that always evaluates true  
while (true) {

    // Do something

}

\* For loop with infinite counter:

for (let i=0; i<infinity; i++) {

    // Do something

}

\* Recursive fn with no base case

fn factorial (n) {

    if (n==0) {

        return 1;

    } else

        return n \* factorial (n-1);

}

3

The fn will call itself indefinitely because there is no base case.

\* Function with condition that always evaluate false  
fn infiniteLoop() {  
 while (false) {  
 do {  
 }  
 }  
}

Q8) What are template literals.

→ These are literals deformed with backtic(`), allowing multiline strings, string interpolation with embedded expressions.

100) What are the difference between for of and for in.

→ For of is used to iterate through elements of the array.  
For in is used to iterate through index or position of the array.

Q7) What are default values in destructuring assignment.

→ The default values in destructuring assignment can be used to set a default value for a destructured variable if the property or element is not present in the object or array.

const person = {

age: 30,

occ: developer,

}

const { name = "Hello" } = person.

c: log(name) : → hello.

STUDENT'S NAME	
CLASS	SUBJECT
ROLL NO	DATE

Methods	Functionality	Return o/p	Modify original arr.
concat()	combines two or more arrays into new array.	Yes	No.
every()	return true if the provided function returns true for all the elements.	No	Also.
filter()	return a new array containing only elements for which provided function returns true.	No	No.
find()	return the first element in the array for which the provided function returns true.	No	No.
findIndex()	return the index of first element which provided function returns true for each element in array.	No	No.
includes()	return true if the array contains the specific element.	Yes	No.
indexof()	return the index of specified element by providing element with specified separator.	Yes	No.
join()	convert the array into string by joining elements with specified separator.	Yes	No.
lastIndexOf()	return the index of specified character in the array starting from the end.	Yes	No.
map()	return new array containing the results.	Yes	No.
pop()	remove the last element.	Yes	No.
push()	add element at the end.	Yes	No.
reduce()	reducing the array into single value.	Yes	No.
reverse()	reverse the order of elements.	Yes	No.
shift()	remove the first element.	Yes	No.
slice()	creating new array from copied elements from original array.	Yes	No.
sort()	sort the elements in ascending order.	Yes	No.
splice()	removes or add elements.	Yes	No.

Q1)

How do you swap variables in destructuring assignment.

$[a, b] = [b, a]$

Q2)

Is it possible to use expressions in switch cases?

→ Yes,

```
let day = "monday"
switch (day) {
    case monday*:
        c.log ("Monday");
    break
```

Q3). What are the differences between arguments object and rest parameters

→ The arguments object is a property of all fn, while rest parameters must be explicitly declared.

② The argument object is not an array, while rest parameters are

③ The argument object is mutable, while rest parameters are immutable.

Q4). What are the diff b/w spread operator & rest operator.

→ The spread operator is used to spread the contents of an iterable object, such as an array object, into individual arguments.

Let num = [1, 2, 3]

const sum = (...num) => {

return numbers.reduce((a, b) => a+b);

c.log (sum(...num)); // 6.

The rest operator is used to collect all the remaining args after the first few have been assigned to named parameters.