1 What is JavaScript. How to use it?

javaScript is client side and server side language. javaScript introduce by browser company named Netscape browser and developed by Brendan Eich in 1995.

For use javascript there are three way-

Embedding code.

Inline code.

External code.

2 How many type of Variable in JavaScript?

There are three types of variable in JavaScript

Var, let , const .

3 Define a Data Types in js?

There are two types of data types in JavaScript.

Primitive data types (string , number ,Boolean , undefined, null ) Primitive data types handle only single value.

Nonprimitive data types (Array , function , object ) Non primitive data types handle multiple values.

Q.4 Write a mul Function Which will Work Properly When invoked With Following Syntax.

The mul function takes an argument . It returns an anonymous function that takes another argument . When you invoke mul(a), it returns a function that multiplies any value passed to it by a .

    <script>

        function mul(a) {

  return function(b) {

    return a \* b;

  };

}

const multiplyBy5 = mul(5);

console.log(multiplyBy5(3));

output : 15

5 What the deference between undefined and undeclare in JavaScript?

Undefined - In javaScript when we create a variable but not declare its value then return Undefined.

Undeclare- An undeclared variable  refers to a variable that doesn’t exist in the memory heap area. It hasn’t been written or declared in the code.

6 Using console.log() print out the following statement: The quote 'There is no exercise better for the heart than reaching down and lifting people up.' by John Holmes teaches us to help one another. Using console.log() print out the following quote by Mother Teresa:

    <script>

      console.log(

        "The quote 'There is no exercise better for the heart than reaching down and lifting people up.' by John Holmes teaches us to help one another."

      );

      console.log(

        "Let's also remember the wisdom of Mother Teresa: 'Not all of us can do great things. But we can do small things with great love.'"

      );

    </script>

7 Check if typeof '10' is exactly equal to 10. If not make it exactly equal?

Type of ’10’ is string

10 is an integar

When we compare both its return false .

    <script>

      const strNumber = "10";

      const parsedNumber = parseInt(strNumber, 10); // Convert to base 10 (decimal)

      console.log(parsedNumber); // Output: 10

      console.log(typeof parsedNumber); // Output: 'number'

    </script>

8 Write a JavaScript Program to find the area of a triangle?

When we know Base and hight

<script>

      const baseValue = prompt("Enter the base of a triangle: ");

      const heightValue = prompt("Enter the height of a triangle: ");

      // Calculate the area

      const areaValue = (baseValue \* heightValue) / 2;

      console.log(`The area of the triangle is ${areaValue}`);

    </script>

9 Write a JavaScript program to calculate days left until next Christmas?

 <script>

      let today = new Date();

      let christmasYear = today.getFullYear();

      // Check if today is after December 25th

      if (today.getMonth() === 11 && today.getDate() > 25) {

        christmasYear += 1; // Set Christmas for the next year

      }

      let christmasDate = new Date(christmasYear, 11, 25);

      let dayMilliseconds = 1000 \* 60 \* 60 \* 24;

      let remainingDays = Math.ceil(

        (christmasDate.getTime() - today.getTime()) / dayMilliseconds

      );

      console.log(`There are ${remainingDays} days remaining until Christmas.`);

    </script>

10 What is Condition Statement?

There are three types of conditional statement in JavaScript (if , if-else , else-if) .

If example –

<script>

      let num = 20;

      if (num % 2 === 0) {

        console.log("Given number is even.");

      }

      if (num % 2 !== 0) {

        console.log("Given number is odd.");

      }

    </script>

// Output: Given number is even.

If- else example -

<script>

      let age = 25;

      if (age >= 18) {

        console.log("You are eligible for a driving license");

      } else {

        console.log("You are not eligible for a driving license");

      }

    </script>

// Output: You are eligible for a driving license

else-if example-

<script>

      let timeOfDay = "morning";

      if (timeOfDay === "morning") {

        console.log("Good morning!");

      } else if (timeOfDay === "afternoon") {

        console.log("Good afternoon!");

      } else {

        console.log("Good evening!");

      }

    </script>

// Output: Good morning!

11 Find circumference of Rectangle formula : C = 4 \* a ?

* The perimeter of a rectangle is given by: [ P = 2(a + b) ] where:

(a) represents the length of the rectangle.

(b) represents the width (or breadth) of the rectangle.

<script>

      function calculatePerimeter(a, b) {

        return 2 \* (a + b);

      }

      // Example usage

      const length = 12; // Replace with your desired length

      const width = 8; // Replace with your desired width

      const perimeter = calculatePerimeter(length, width);

      console.log(`Perimeter of the rectangle: ${perimeter}`);

    </script>

12- WAP to convert years into days and days into years?

**Convert Years to Days-**

To convert years to days, we’ll multiply the given number of years by 365

    <script>

      function yearsToDays(years) {

        return years \* 365;

      }

      // Example usage

      const givenYears = 3; // Replace with your desired number of years

      const days = yearsToDays(givenYears);

      console.log(`${givenYears} years = ${days} days`);

    </script>

**Convert Days to Years**:

To convert days to years, we’ll divide the given number of days by 365

    <script>

      function daysToYears(days) {

        return days / 365;

      }

      // Example usage

      const givenDays = 1095; // Replace with your desired number of days

      const years = daysToYears(givenDays);

      console.log(`${givenDays} days = ${years} years`);

    </script>

13-Convert temperature Fahrenheit to Celsius? (Conditional logic Question)

To convert a temperature from Fahrenheit to Celsius,

[ \text{Celsius} = \frac{{\text{Fahrenheit} - 32}}{9} \times 5 ]

    <script>

      // Convert Fahrenheit to Celsius using a basic function

      function fahrenheitToCelsius(fahrenheit) {

        return ((fahrenheit - 32) \* 5) / 9;

      }

      const givenFahrenheit = 100; // Replace with your desired Fahrenheit temperature

      const celsius = fahrenheitToCelsius(givenFahrenheit);

      console.log(`${givenFahrenheit}°F is ${celsius.toFixed(2)}°C`);

    </script>

14- Write a JavaScript exercise to get the extension of a filename.?

You can split the filename by the dot (.) character and then retrieve the last element of the resulting array, which represents the file extension.

<script>

      function getFileExtension(filename) {

        const parts = filename.split(".");

        return parts.pop(); // Returns the file extension

      }

      // Example usage

      const filename1 = "example.txt";

      const extension1 = getFileExtension(filename1);

      console.log(`Extension for ${filename1}: ${extension1}`);

      const filename2 = "document.pdf";

      const extension2 = getFileExtension(filename2);

      console.log(`Extension for ${filename2}: ${extension2}`);

    </script>

15 What is the result of the expression (5 > 3 && 2 < 4)?

Both condition are true so result is true (this is logical operator(&&)).

16 What is the result of the expression (true && 1 && "hello")?

When using the logical AND(&&) operator, if all operands are truthy, the last truthy operand is returned. In this case, both true and 1 are truthy, so the result is the string "hello".

17 What is the result of the expression true && false || false && true?

the final result is **true** because the logical OR operator returns **true** if at least one of the operands is **true**. In this case, both expressions within the OR evaluate to **false**, but the overall result is still **true.**

18 What is a Loop and Switch Case in JavaScript define that ?

Loops are control structures that allow you to repeat a block of code multiple times.

They help automate repetitive tasks and iterate over collections of data.

For loop- Executes a block of code a specified number of times.

    <script>

      for (let i = 0; i < 5; i++) {

        console.log("Iteration " + i);

      }

    </script>

While loop- Repeats a block of code while a condition is true.

Do while- Similar to While but it always executes the block at least once.

Switch case-

The switch statement is used to perform different actions based on different conditions.

It allows you to test a variable or expression against multiple cases and execute different code for each matching case.

    <script>

      switch (expression) {

        case value1:

          // code to be executed if expression === value1

          break;

        case value2:

          // code to be executed if expression === value2

          break;

        default:

        // code to be executed if none of the cases match

      }

    </script>

19 What is the use of is Nan function?

The isNan() function in JavaScript serves to check whether a given value is a **“Not-a-Number”(Nan).**

20 What is the difference between && and || in JavaScript?

. “&& “ If both operands are true, the "&&" operator returns "true," else it returns "false."

The operator is short-circuiting. The second operand is not examined if the first operand is false since the overall result will still be false regardless of the second operand.

true && true; // true

true && false; // false

false && true; // false

false && false; // false

|| = The operators "||" yield "true" when at least one operand is "true," and "false" if both operands are false.

It is a short-circuiting operator, just like &&. The second operand is not assessed if the first operand is true since the overall result will always be true, regardless of the second operand.

true || true; // true

true || false; // true

false || true; // true

false || false; // false

21 What is the use of Void (0)?

When you use void(0) it evaluates the expression (which is simply 0) and then returns undefined.

Essentially, it means “do nothing” or “return undefined.”

This is commonly used in event handlers or anchor tags to prevent the page from refreshing or navigating to a new URL.

  <body>

    <a href="javascript:void(0);">Click here to do nothing</a>

  </body>

22 Check Number Is Positive or Negative in JavaScript?

**Using if…else if…else:**

<script>

      const number = parseInt(prompt("Enter a number: "));

      if (number > 0) {

        console.log("The number is positive");

      } else if (number === 0) {

        console.log("The number is zero");

      } else {

        console.log("The number is negative");

      }

    </script>

23- Find the Character Is Vowel or Not ?

**Using if…else**:

    <script>

               def check\_vowel\_or\_consonant(char):

          char = char.lower()

          //  # Convert to lowercase for case-insensitivity

          if char in 'aeiou':

              return "Vowel"

          else:

              return "Consonant"

      # Example usage:

      print(check\_vowel\_or\_consonant('c'))  # Output: Consonant

      print(check\_vowel\_or\_consonant('e'))  # Output: Vowel

    </script>

24 Write to check whether a number is negative, positive or zero?

**Using if…else**:

<script>

  def check\_positive\_negative\_zero(num):

    if num > 0:

        return "Positive"

    elif num < 0:

        return "Negative"

    else:

        return "Zero"

# Example usage:

number = float(input("Enter a number: "))

result = check\_positive\_negative\_zero(number)

print(f"The number is {result}.")

</script>

25 Write to find number is even or odd using ternary operator in JS?

    <script>

      const number = 17;

      const isOdd = number % 2 === 1 ? "odd" : "even";

      console.log(isOdd);

      // Output: "odd"

    </script>

26 Write find maximum number among 3 numbers using ternary operator in JS?

    <script>

      const num1 = 10;

      const num2 = 20;

      const num3 = 15;

      const max =

        num1 > num2 ? (num1 > num3 ? num1 : num3) : num2 > num3 ? num2 : num3;

      console.log(

        `Largest number among ${num1}, ${num2}, and ${num3} is ${max}.`

      );

    </script>

27- Write to find minimum number among 3 numbers using ternary operator in JS?

    <script>

      const a = 10;

      const b = 20;

      const c = 30;

      const smallest = a < b ? (a < c ? a : c) : b < c ? b : c;

      console.log(`The smallest number is ${smallest}.`);

    </script>

28-Write to find the largest of three numbers in JS ?

    <script>

      const num1 = parseFloat(prompt("Enter first number: "));

      const num2 = parseFloat(prompt("Enter second number: "));

      const num3 = parseFloat(prompt("Enter third number: "));

      let largest;

      // Check the condition

      if (num1 >= num2 && num1 >= num3) {

        largest = num1;

      } else if (num2 >= num1 && num2 >= num3) {

        largest = num2;

      } else {

        largest = num3;

      }

      // Display the result

      console.log(`The largest number is ${largest}.`);

    </script>

Q.29 Write to show

i. Monday to Sunday using switch case in JS?

    <script>

      // Get the current day of the week (0 = Sunday, 1 = Monday, ..., 6 = Saturday)

      const dayOfWeek = new Date().getDay();

      let day;

      switch (dayOfWeek) {

        case 0:

          day = "Sunday";

          break;

        case 1:

          day = "Monday";

          break;

        case 2:

          day = "Tuesday";

          break;

        case 3:

          day = "Wednesday";

          break;

        case 4:

          day = "Thursday";

          break;

        case 5:

          day = "Friday";

          break;

        case 6:

          day = "Saturday";

          break;

        default:

          day = "Invalid day"; // Just in case

      }

      console.log(`Today is ${day}.`);

    </script

ii. Vowel or Consonant using switch case in JS?

    <script>

      function checkVowelOrConsonant(char) {

        switch (char.toLowerCase()) {

          case "a":

          case "e":

          case "i":

          case "o":

          case "u":

            console.log(`${char} is a vowel.`);

            break;

          default:

            console.log(`${char} is a consonant.`);

        }

      }

      // Example usage:

      checkVowelOrConsonant("a"); // Output: "a is a vowel."

      checkVowelOrConsonant("b"); // Output: "b is a consonant."

    </script>

(Conditional looping logic Question)

30 What are the looping structures in JavaScript? Any one Example?

For loop

 <script>

      let val = 1;

      while (val < 6) {

        console.log(val);

        val += 1;

      }

    </script>

While loop

Do while

31 Write a print 972 to 897 using for loop in JS?

<script>

      for (let i = 972; i >= 897; i--) {

        console.log(i);

      }

    </script>

32 Write to print factorial of given number?

Using for loop-

 <script>

      // Program to find the factorial of a number

      const number = parseInt(prompt("Enter a positive integer: "));

      if (number < 0) {

        console.log("Error! Factorial for negative number does not exist.");

      } else if (number === 0) {

        console.log(`The factorial of ${number} is 1.`);

      } else {

        let fact = 1;

        for (let i = 1; i <= number; i++) {

          fact \*= i;

        }

        console.log(`The factorial of ${number} is ${fact}.`);

      }

    </script>

33 Write to print Fibonacci series up to given numbers?

Using a for loop-

<script>

      // Program to display the Fibonacci series up to n terms

      const n = parseInt(prompt("Enter the number of terms: "));

      let t1 = 0,

        t2 = 1;

      console.log(`Fibonacci Series: ${t1}, ${t2}, `);

      for (let i = 3; i <= n; ++i) {

        const nextTerm = t1 + t2;

        console.log(nextTerm);

        t1 = t2;

        t2 = nextTerm;

      }

    </script>

34 Write to print number in reverse order e.g.: number = 64728 ---> reverse =82746 in JS?

<script>

        const originalNumber = 64728;

        const reversedNumber = Number(originalNumber.toString().split('').reverse().join(''));

        document.write(reversedNumber);

    </script>

35 Write a program make a summation of given number (E.g., 1523 Ans: - 11) in JS?

<script>

        function summation(number) {

            let sum = 0;

            const numberString = Math.abs(number).toString();

            for (let i = 0; i < numberString.length; i++) {

                sum += parseInt(numberString[i]);

            }

            return number < 0 ? -sum : sum;

        }

        const Number = 1523;

        const result = summation(inputNumber);

        document.write(`The summation of the digits of ${Number} is: ${result}`);

    </script>

36 Write a program you have to make a summation of first and last Digit. (E.g., 1234 Ans: - 5) in JS?

<script>

      function sumOfFirstAndLastDigit(number) {

        // Convert the number to a string

        const numString = number.toString();

        // Extract the first and last digits

        const firstDigit = parseInt(numString[0]);

        const lastDigit = parseInt(numString[numString.length - 1]);

        // Calculate the sum

        const sum = firstDigit + lastDigit;

        return sum;

      }

      // Example usage:

      const inputNumber = 1234;

      const result = sumOfFirstAndLastDigit(inputNumber);

      console.log(

        `The sum of the first and last digits of ${inputNumber} is ${result}.`

      );

    </script>

37 Use console.log() and escape characters to print the following pattern in JS?

1 1 1 1 1

2 1 2 4 8

3 1 3 9 27

4 1 4 16 64

5 1 5 25 125

<script>

      // Program to print the given pattern

      console.log("1 1 1 1 1");

      console.log("2 1 2 4 8");

      console.log("3 1 3 9 27");

      console.log("4 1 4 16 64");

      console.log("5 1 5 25 125");

    </script>

38 Use pattern in console.log in JS?

1. 1

1 0

1 0 1

1 0 1 0

1 0 1 0 1

<script>

      for (let i = 1; i <= 5; i++) {

        let row = "";

        for (let j = 1; j <= i; j++) {

          row += j % 2 === 0 ? "0 " : "1 ";

        }

        console.log(row);

      }

    </script>

1. A

B C

D E F

G H I J

K L M N O

<script>

      let char = "A";

      for (let i = 1; i <= 5; i++) {

        let row = "";

        for (let j = 1; j <= i; j++) {

          row += char + " ";

          char = String.fromCharCode(char.charCodeAt(0) + 1);

        }

        console.log(row);

      }

    </script>

1. 1

2 3

4 5 6

7 8 9 10

11 12 13 14 15

 <script>

      let num = 1;

      for (let i = 1; i <= 5; i++) {

        let row = "";

        for (let j = 1; j <= i; j++) {

          row += num + " ";

          num++;

        }

        console.log(row);

      }

    </script>

4) \*

\* \*

\* \* \*

\* \* \* \*

\* \* \* \* \*

<script>

      for (let i = 1; i <= 5; i++) {

        let row = "";

        for (let j = 1; j <= i; j++) {

          row += "\* ";

        }

        console.log(row);

      }

    </script>

(Array and object Question)

40 Write a JavaScript Program to display the current day and time in the following format. Sample Output: Today is Friday. Current Time is 12 PM: 12 : 22 2 ?

<script>

      // Get the current date and time

      const today = new Date();

      // Get the day of the week (0-6, where 0 is Sunday and 6 is Saturday)

      const day = today.getDay();

      // Array of day names

      const daylist = [

        "Sunday",

        "Monday",

        "Tuesday",

        "Wednesday",

        "Thursday",

        "Friday",

        "Saturday",

      ];

      // Display the current day

      console.log(`Today is: ${daylist[day]}.`);

      // Get the current time

      const hours = today.getHours();

      const minutes = today.getMinutes();

      const seconds = today.getSeconds();

      // Determine AM or PM

      const ampm = hours >= 12 ? "PM" : "AM";

      // Convert 24-hour format to 12-hour format

      const formattedHours = hours % 12 || 12;

      // Display the current time

      console.log(

        `Current Time is: ${formattedHours} ${ampm}: ${minutes} : ${seconds}`

      );

    </script>

41 Write a JavaScript program to get the current date?

<script>

      const currentDate = new Date().toDateString();

      console.log(`Today's date is ${currentDate}.`);

    </script>

42 Write a JavaScript program to compare two objects?

 <script>

      const obj1 = { name: "John", age: 30 };

      const obj2 = { age: 30, name: "John" };

      const isEqual = JSON.stringify(obj1) === JSON.stringify(obj2);

      console.log(`Are the objects equal? ${isEqual}`);

    </script>

43 Write a JavaScript program to convert an array of objects into CSV string?

<script>

      function convertArrayOfObjectsToCSV(dataArray) {

        const headers = Object.keys(dataArray[0]);

        const csvRows = [headers.join(",")];

        for (const item of dataArray) {

          const values = headers.map((header) => item[header]);

          csvRows.push(values.join(","));

        }

        return csvRows.join("\n");

      }

      // Example usage:

      const itemsArray = [

        { itemId: 1, itemRef: "Item 001" },

        { itemId: 2, itemRef: "Item 002" },

        { itemId: 3, itemRef: "Item 003" },

      ];

      const csvString = convertArrayOfObjectsToCSV(itemsArray);

      console.log(csvString);

    </script>

44 Write a JavaScript program to capitalize first letter of a string?

<script>

      function capitalizeFirstLetter(string) {

        return string.charAt(0).toUpperCase() + string.slice(1);

      }

      // Example usage:

      const inputString = "this is a test";

      const capitalizedString = capitalizeFirstLetter(inputString);

      console.log(capitalizedString); // Output: "This is a test"

    </script>

45 Write a JavaScript program to determine if a variable is array?

<script>

      const myVariable = [1, 2, 3]; // Replace with your variable

      const isAnArray = Array.isArray(myVariable);

      console.log(`Is the variable an array? ${isAnArray}`);

    </script>

47 What is the drawback of declaring methods directly in JavaScript objects?

If you declare methods directly within each object instance, it leads to code duplication. Each object will have its own copy of the method, consuming more memory.

 <script>

      const person1 = {

        name: "Alice",

        greet: function () {

          console.log(`Hello, I'm ${this.name}.`);

        },

      };

      const person2 = {

        name: "Bob",

        greet: function () {

          console.log(`Hello, I'm ${this.name}.`);

        },

      };

    </script>

48 Print the length of the string on the browser console using console.log()?

To print the length of a string in JavaScript, you can use the .length property.

<script>

      const myString = "Hello, World!";

      console.log(`The length of the string is ${myString.length}.`);

    </script>

49 Change all the string characters to capital letters using toUpperCase() method?

To convert all characters in a string to uppercase, you can use the toUpperCase() method.

<script>

      const myString = "hello, world!";

      const uppercaseString = myString.toUpperCase();

      console.log(uppercaseString); // Output: "HELLO, WORLD!"

    </script>

50 What is the drawback of declaring methods directly in JavaScript objects?

If you declare methods directly within each object instance, it leads to code duplication. Each object will have its own copy of the method, consuming more memory.

<script>

      const person1 = {

        name: "Alice",

        greet: function () {

          console.log(`Hello, I'm ${this.name}.`);

        },

      };

      const person2 = {

        name: "Bob",

        greet: function () {

          console.log(`Hello, I'm ${this.name}.`);

        },

      };

    </script>

51 Write a JavaScript program to get the current date. Expected Output : mm-dd-yyyy, mm/dd/yyyy or dd-mm-yyyy, dd/mm/yyyy?

<script>

      const today = new Date();

      const dd = String(today.getDate()).padStart(2, "0");

      const mm = String(today.getMonth() + 1).padStart(2, "0"); // Months start at 0!

      const yyyy = today.getFullYear();

      // Format options:

      const formattedDate1 = `${mm}-${dd}-${yyyy}`;

      const formattedDate2 = `${mm}/${dd}/${yyyy}`;

      const formattedDate3 = `${dd}-${mm}-${yyyy}`;

      const formattedDate4 = `${dd}/${mm}/${yyyy}`;

      console.log(`mm-dd-yyyy: ${formattedDate1}`);

      console.log(`mm/dd/yyyy: ${formattedDate2}`);

      console.log(`dd-mm-yyyy: ${formattedDate3}`);

      console.log(`dd/mm/yyyy: ${formattedDate4}`);

    </script>

52 Use indexOf to determine the position of the first occurrence of a in 30 Days Of JavaScript?

Certainly! In JavaScript, the indexOf() method allows you to find the position of the **first occurrence** of a specified value within a string.

    <script>

      let text = "30 Days Of JavaScript";

      let position = text.indexOf("a");

      console.log(`The first occurrence of 'a' is at position ${position}.`);

    </script>

53 Use lastIndexOf to determine the position of the last occurrence of a in 30 Days Of JavaScript?

In JavaScript, the lastIndexof() method allows you to find the position of the **last occurrence** of a specified value within a string.

    <script>

      let text = "30 Days Of JavaScript";

      let position = text.lastIndexOf("a");

      console.log(`The last occurrence of 'a' is at position ${position}.`);

    </script>

54 how many type of JS Event? How to use it ?

Mouse Events:

* click: Triggered when the user clicks an element.
* dblclick: Triggered when the user double-clicks an element.
* mousedown: Triggered when the user presses a mouse button over an element.
* mouseup: Triggered when the user releases a mouse button over an element.
* mousemove: Triggered when the user moves the mouse pointer over an element.
* mouseover and mouseout: Triggered when the mouse pointer enters or leaves an element.

Keyboard Events:

* keydown: Triggered when a key is pressed down.
* keyup: Triggered when a key is released.
* keypress: Triggered when a key is pressed and released.

Structured Events:

* submit: Started as soon as a form is sent in.
* change: Initiated when an input element's value shifts.
* input: Always activated when an input element's value fluctuates.
* blur and focus: Initiated when an element experiences a change of focus.

55 What is Bom vs Dom in JS?

DOM (Document Object Model):

* The DOM represents the structure of a document as a tree of objects. Each object corresponds to a part of the document, such as elements, attributes, and text.
* Document: The top-level object that represents the entire HTML or XML document.
* Element: Represents HTML or XML elements in the document, like <div>, <p>, etc.
* Attribute: Represents attributes of elements, like id, class, etc.
* Text: Represents the actual text content within elements.

Usage:

* DOM is used to dynamically manipulate and interact with the content of a web page. It allows you to create, modify, and delete elements, attributes, and text dynamically.

BOM (Browser Object Model):

* The BOM represents the browser itself and provides objects and methods for interacting with the browser window and its components.
* Window: The top-level object that represents the browser window. It provides properties and methods for controlling the browser window
* Navigator: Provides information about the browser and its capabilities.
* Screen: Represents the user's screen and provides information like screen width and height.
* Location: Represents the URL of the current page.
* History: Represents the browser's session history.

Usage:

* BOM is used for tasks related to the browser itself, such as controlling the size and position of the browser window, navigating to different pages, and managing the browser's history.

56 Array vs object defences in JS?

Arrays:

* An array is an ordered, indexed collection of values. Each element in an array is identified by an index, starting from 0.
* Arrays can hold values of any data type, including numbers, strings, objects, or even other arrays.

Syn = const myArray = [1, 'two', { key: 'value' }, true];

Use Cases:

* Best suited for when you have a list of similar or related items and need to perform operations on them collectively.
* Useful for operations like iteration, filtering, mapping, and reducing elements.

Objects:

* An object is an unordered, key-value pair collection of data. Each key (also called a property) is associated with a value.
* Objects can hold values of any data type, and the keys are typically strings or symbols.

Syn = const myObject = {

key1: 'value1',

key2: 42,

key3: { nestedKey: 'nestedValue' },

key4: true

};

Use Cases:

* Best suited for representing entities with properties or attributes.
* Useful when you need to access values using descriptive keys.

57 Split the string into an array using split() Method?

<script>

            public class GFG {

          public static void main(String args[]) {

              String str = "geekss@for@geekss";

              String[] arrOfStr = str.split("@", 2);

              for (String a : arrOfStr)

                  System.out.println(a);

          }

      }

      // Output: geekss for@geekss

    </script>

58 Check if the string contains a word Script using includes() method?

In JavaScript, you can use the includes() method to check if a string contains a specific word or substring.

    <script>

      const str = "The quick brown fox jumps over the lazy dog.";

      console.log(str.includes("fox")); // Expected output: true

    </script>

59 Change all the string characters to lowercase letters using toLowerCase() Method.

<script>

      const originalString = "Hello, World!";

      const lowerCaseString = originalString.toLowerCase();

      console.log(`Original string: ${originalString}`);

      console.log(`Lowercase string: ${lowerCaseString}`);

    </script>

60 What is Character at index 15 in ’30 Days of JavaScript’ string? Use charAt() method.

<script>

      const myString = "30 Days of JavaScript";

      const characterAtIndex15 = myString.charAt(15);

      console.log(`Character at index 15: ${characterAtIndex15}`);

    </script>

61 copy to one string to another string in JS?

<script>

      const originalString = "Hello, World!";

      const copiedString = originalString.slice();

      console.log(copiedString);

      // "Hello, World!"

    </script>