1. Write a program to Show an alert

<script>

    // JavaScript code to show an alert

alert("This is an alert!");

</script>

1. What will be the result for these expressions?

🡪5 > 4 evaluates to true because 5 is greater than 4.

🡪"apple" > "pineapple" evaluates to false because in lexicographical order, "apple" comes before "pineapple".

🡪"2" > "12" evaluates to true because when comparing strings, each character is compared based on their Unicode code point value, and "2" has a higher code point value than "1".

🡪undefined == null evaluates to true because both undefined and null are considered equal when using loose equality (==) comparison.

🡪undefined === null evaluates to false because === performs strict equality comparison, which also checks for the equality of types. undefined and null have different types.

🡪null == "\n0\n" evaluates to false because when using loose equality (==), null is only equal to undefined, null, or itself.

🡪7. null === +"\n0\n" is a syntax error because 7. null is not a valid syntax in JavaScript. Please correct it, and I can evaluate the expression for you.

1. Will alert be shown?

if ("0") { alert( 'Hello'); }

🡪Yes, the alert will be shown with that code. The condition ("0") evaluates to true because any non-empty string in JavaScript is considered truthy.

Therefore, the code inside the curly braces {} will execute, resulting in the alert displaying "Hello".

1. What is the code below going to output? alert( null || 2 || undefined );

The code will output 2 when executed.2 is truthy, so the expression evaluates to 2 and the alert will display 2

1. What is JavaScript. How to use it?

-> JavaScript is a versatile programming language primarily used for creating interactive and dynamic content on websites.

-> Originally developed for web browsers, it has now expanded its reach to various platforms like server-side programming, mobile app development, game development, and more.

**JavaScript into HTML**: You can include JavaScript code directly within HTML files using <script> tags.

<script>

    class **a** {

         ra() {

            document**.**write("hii")

        }

    }

    const bo**=new** a

    bo**.**ra()

</script>

**External JavaScript files**: can also write JavaScript code in external files with a .js extension and include them in your HTML files using the <script> tag's src attribute.

<script *src***=**"math.js"></script>

1. How many type of Variable in JavaScript?

In JavaScript, there are primarily three types of variables.

**var**: var was traditionally used for variable declaration in JavaScript. Variables declared with var are function-scoped or globally scoped, but they are not block-scoped

**let**: Introduced in ECMAScript 6 (ES6), let allows you to declare block-scoped variables

**const**: const is used to declare variables that cannot be reassigned. It creates a constant whose value cannot be changed once it has been initialized.

*// var example*

var x **=** 10**;**

console**.**log(x)**;** *// Output: 10*

*// let example*

let y **=** 20**;**

{

    let y **=** 30**;** *// This is a different variable y with block scope*

    console**.**log(y)**;** *// Output: 30*

}

console**.**log(y)**;** *// Output: 20*

*// const example*

const z **=** 50**;**

*// z = 60; // This will throw an error because you cannot reassign a constant variable*

const person **=** {

    name**:** 'John'**,**

    age**:** 30

}**;**

person**.***age* **=** 35**;** *// This is allowed, as it's modifying the content of the object, not reassigning the variable*

console**.**log(person**.***age*)**;** *// Output: 35*

1. Define a Data Types in js?

Number: Represents numeric values, including integers and floating-point numbers. Example: let num = 10;

String: Represents textual data enclosed within single (' ') or double (" ") quotes. Example: let str = "Hello, world!";

Boolean: Represents a logical value indicating true or false. Example: let isTrue = true;

Undefined: Represents a variable that has been declared but not assigned a value. Example: let x;

Null: Represents an intentional absence of any value or object. Example: let y = null;

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

 function mul() {

*// Check if no arguments are provided*

**if** (*arguments***.**length **===** 0) {

**return** 0**;**

    }

*// Check if only one argument is provided*

**if** (*arguments***.**length **===** 1) {

**return** *arguments*[0]**;**

    }

*// Initialize the result with the first argument*

    let result **=** *arguments*[0]**;**

*// Loop through the rest of the arguments and multiply with the result*

**for** (let i **=** 1**;** i **<** *arguments***.**length**;** i**++**) {

        result **\*=** *arguments*[i]**;**

    }

**return** result**;**

}

*// Test cases*

console**.**log(mul())**;** *// Output: 0*

console**.**log(mul(5))**;** *// Output: 5*

console**.**log(mul(2**,** 3))**;** *// Output: 6*

console**.**log(mul(2**,** 3**,** 4))**;** *// Output: 24*

console**.**log(mul(2**,** 3**,** 4**,** 5))**;** *// Output: 120*

1. What the deference between undefined and undeclare in JavaScript?

**Undefined**:

undefined is a primitive value that represents a variable that has been declared but not assigned any value, or a property that does not exist within an object

let x**;**

console**.**log(x)**;** *// Output: undefined*

let obj **=** {}**;**

console**.**log(obj**.***property*)**;** *// Output: undefined*

**Undeclared**:

An undeclared variable is a variable that has been referenced in code without being declared using var, let, or const.

1. 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:

  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("Spread love everywhere you go. Let no one ever come to you without leaving happier. - Mother Teresa")**;**

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

the typeof operator returns a string indicating the type of the operand. So, typeof '10' will return 'string', not the number 10. To convert the string '10' to the number 10, you can use the parseInt() function or the Number() constructor. Here's how you can do it:

var strNumber **=** '10'**;**

var num **=** parseInt(strNumber)**;**

console**.**log(**typeof** num **===** 'number')**;** *// output: true*

console**.**log(num **===** 10)**;** *// true*

This code converts the string '10' to the number 10 and then checks if the type is exactly 'number' and if the value is exactly 10

1. Write a JavaScript Program to find the area of a triangle?

function calculateTriangleArea(**a,** **b,** **c**) {

    var s **=** (a **+** b **+** c) **/** 2**;**

*// the area using Heron's formula*

    var area **=** Math**.**sqrt(s **\*** (s **-** a) **\*** (s **-** b) **\*** (s **-** c))**;**

**return** area**;**

}

*// Example usage:*

var side1 **=** 5**;** *// length of the first side*

var side2 **=** 6**;** *// length of the second side*

var side3 **=** 7**;** *// length of the third side*

var area **=** calculateTriangleArea(side1**,** side2**,** side3)**;**

console**.**log("The area of the triangle with sides"**,** side1 **+** ","**,** side2 **+** ","**,** "and"**,** side3**,** "is:"**,** area)**;**

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

function daysUntilChristmas() {

*// Get today's date*

    var today **=** **new** Date()**;**

*// Get the current year*

    var currentYear **=** today**.**getFullYear()**;**

*// Christmas will be on December 25 of the current year*

    var christmas **=** **new** Date(currentYear**,** 11**,** 25)**;** *// Note: month is 0-based, so 11 is December*

*// If Christmas has already passed for this year, set it for next year*

**if** (today **>** christmas) {

        christmas**.**setFullYear(currentYear **+** 1)**;**

    }

*// Calculate the difference in milliseconds between today and Christmas*

    var differenceInMillis **=** christmas **-** today**;**

*// Convert milliseconds to days*

    var daysLeft **=** Math**.**ceil(differenceInMillis **/** (1000 **\*** 60 **\*** 60 **\*** 24))**;**

**return** daysLeft**;**

}

*// Example usage:*

var daysLeft **=** daysUntilChristmas()**;**

console**.**log("There are"**,** daysLeft**,** "days left until Christmas!")**;**

1. What is Condition Statement?

If the condition evaluates to true, the code inside the block is executed; otherwise, it is skipped.

decr**.**addEventListener('click'**,** () **=>** {

**if** (count **>** 0) {

        count**--;**

        counter**.***innerHTML* **=** count**;**

    }

})

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

1. WAP to convert years into days and days into years?

*// Function to convert years into days*

function yearsToDays(**years**) {

    var days **=** years **\*** 365.25**;** *// Assuming 1 year = 365.25 days (accounting for leap years)*

**return** days**;**

}

*// Function to convert days into years*

function daysToYears(**days**) {

    var years **=** days **/** 365.25**;** *// Assuming 1 year = 365.25 days (accounting for leap years)*

**return** years**;**

}

*// Example usage:*

var years **=** 5**;**

var days **=** 1826**;** *// Approximately 5 years in days*

console**.**log(years **+** " years is equal to " **+** yearsToDays(years) **+** " days.")**;**

console**.**log(days **+** " days is equal to " **+** daysToYears(days) **+** " years.")**;**

yearsToDays() takes a number of years as input and returns the equivalent number of days, assuming 1 year is equal to 365.25 days to account for leap years.

daysToYears() takes a number of days as input and returns the equivalent number of years, assuming 1 year is equal to 365.25 days.

1. Convert temperature Fahrenheit to Celsius? (Conditional logic Question)

To convert temperature from Fahrenheit to Celsius, you can use the following formula



*C* is the temperature in Celsius,

*F* is the temperature in Fahrenheit.

function fahrenheitToCelsius(**fahrenheit**) {

    var celsius **=** (5 **/** 9) **\*** (fahrenheit **-** 32)**;**

**return** celsius**;**

}

*// Example usage:*

var fahrenheitTemp **=** 68**;** *// Example Fahrenheit temperature*

var celsiusTemp **=** fahrenheitToCelsius(fahrenheitTemp)**;**

console**.**log(fahrenheitTemp **+** " degrees Fahrenheit is equal to " **+** celsiusTemp**.**toFixed(2) **+** " degrees Celsius.")**;**

In this program, the fahrenheitToCelsius() function takes a temperature value in Fahrenheit as input and returns the equivalent temperature in Celsius using the provided formula

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

The expression (5 > 3 && 2 < 4) evaluates to true. This is because both conditions are true: 5 is greater than 3, and 2 is less than 4. The && operator (logical AND) requires both conditions to be true for the overall expression to be true. Therefore, the result is true.

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

In JavaScript, the `&&` operator returns the first falsy value it encounters, or the last value if all values are truthy. In this case, all values are truthy (`true`, `1`, and `"hello"`), so the result of the expression `(true && 1 && "hello")` would be the last truthy value encountered, which is `"hello"`. Therefore, the result is `"hello"`.

1. What is the result of the expression true && false || false && true?

This expression involves logical AND (`&&`) and logical OR (`||`) operators.

Here's the evaluation of the expression step by step:

1. `true && false` evaluates to `false`.

2. `false && true` also evaluates to `false`.

3. `false || false` evaluates to `false`.

So, the result of the expression `true && false || false && true` is `false`.

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

a loop is a control flow statement that allows code to be executed repeatedly based on a certain condition. There are several types of loops in JavaScript, including:

for loop: Executes a block of code a specified number of times.

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

FoLoop:- loop is a programming construct that allows you to repeatedly execute a block of code until a certain condition is met.

* There are 4 types of loop.

For loop: This loop iterates over a block of code a specified number of times. It consists of three optional expressions: initialization, condition, and increment/decrement.

Ex:--

const cars **=** ["BMW"**,** "Volvo"**,** "Saab"**,** "Ford"]**;**

let i **=** 2**;**

let len **=** cars**.**length**;**

let text **=** ""**;**

**for** (**;** i **<** len**;** i**++**) {

  text **+=** cars[i] **+** "<br>"**;**

}

document**.**getElementById("demo")**.***innerHTML* **=** text**;**

**while loop**: This loop executes a block of code as long as the specified condition is true.

Ex:-

let text **=** ""**;**

let i **=** 0**;**

**while** (i **<** 10) {

  text **+=** "<br>The number is " **+** i**;**

  i**++;**

}

document**.**getElementById("demo")**.***innerHTML* **=** text**;**

**do...while loop**: This loop is similar to the while loop, but it ensures that the block of code is executed at least once before checking the condition.

**Ex**:-

let text **=** ""

let i **=** 0**;**

**do** {

  text **+=** "<br>The number is " **+** i**;**

  i**++;**

}

**while** (i **<** 10)**;**

document**.**getElementById("demo")**.***innerHTML* **=** text**;**

**Switch case**:Switch case, on the other hand, is a conditional statement used to perform different actions based on different conditions. It's an alternative to using multiple if...else statements when you have a single variable with multiple possible values.

*// month=10*

*// switch (month) {*

*//     case 1:*

*//         document.write("janyary")*

*//         break;*

*//         case 2:*

*//         document.write("fab")*

*//         break;*

*//         case 3:*

*//         document.write("mar")*

*//         break;*

*//         case 4:*

*//         document.write("april")*

*//         break;*

*//         case 5:*

*//         document.write("may")*

*//         break;*

*//         case 6:*

*//         document.write("jun")*

*//         break;*

*//         case 7:*

*//         document.write("july")*

*//         break;*

*//         case 8:*

*//         document.write("aug")*

*//         break;*

*//         case 9:*

*//         document.write("sep")*

*//         break;*

*//         case 10:*

*//         document.write("oct")*

*//         break;*

*//         case 11:*

*//         document.write("nov")*

*//         break;*

*//         case 12:*

*//         document.write("dec")*

*//         break;*

*//     default:*

*//         document.write("plase enter correct value")*

*//         break;*

*// }*

1. What is the use of is Nan function?

isNaN() to check whether a particular value is a valid number before performing arithmetic operations or other calculations. This helps in handling unexpected inputs gracefully and avoiding errors due to invalid calculations.

 isNaN(123)**;**      *// false, 123 is a valid number*

isNaN('hello')**;**  *// true, 'hello' cannot be converted to a number*

isNaN(NaN)**;**      *// true, NaN is Not-a-Number*

1. What is the difference between && and || in JavaScript?

* && (Logical AND):

It returns true if both operands are true.

If the first operand is false, it returns false without evaluating the second operand, because the logical AND operation requires both operands to be true.

If the first operand is true, it evaluates the second operand. If both operands are true, it returns true. If the second operand is false, it returns false.

* || (Logical OR):

It returns true if at least one of the operands is true.

If the first operand is true, it returns true without evaluating the second operand, because the logical OR operation requires at least one operand to be true.

If the first operand is false, it evaluates the second operand. If both operands are false, it returns false

1. What is the use of Void (0)?

void is a JavaScript operator that takes an expression as its operand and evaluates that expression, then returns undefined. It's typically used in front of an expression to ensure that the result of the expression is always undefined.

In void(0), (0) is simply an expression that evaluates to 0. Since void returns undefined, void(0) effectively returns undefined.

The benefit of using void(0) over just undefined is that it ensures the result is undefined without relying on the global scope. This can be helpful in cases where the global scope's undefined might have been altered (though this is rare and generally not recommended).

<script>

   function doSomething() {

*// Do something...*

**return** **void**(0)**;** *// Explicitly return undefined*

}

*// Another usage example:*

<a *href***=**"javascript:void(0);" *onclick***=**"doSomething();">Click Me</a>

</script>

1. Check Number Is Positive or Negative in JavaScript?

We define a function called checkNumber that takes a single parameter number.

Inside the function, we use an if-else if-else conditional statement to check the value of the number.

If the number is greater than 0, we log "The number is positive."

If the number is less than 0, we log "The number is negative."

If the number is equal to 0, we log "The number is zero."

Example usage demonstrates how to use this function with different numbers.

 function checkNumber(**number**) {

**if** (number **>** 0) {

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

    } **else** **if** (number **<** 0) {

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

    } **else** {

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

    }

}

*// Example usage:*

checkNumber(5)**;**     *// Output: The number is positive.*

checkNumber(**-**10)**;**   *// Output: The number is negative.*

checkNumber(0)**;**     *// Output: The number is zero.*

1. Find the Character Is Vowel or Not ?

function isVowel(**character**) {

*// Convert the character to lowercase to handle both cases (upper and lower)*

    var lowerCaseCharacter **=** character**.**toLowerCase()**;**

*// Define a set of vowels*

    var vowels **=** ['a'**,** 'e'**,** 'i'**,** 'o'**,** 'u']**;**

*// Check if the character is included in the set of vowels*

**if** (vowels**.**includes(lowerCaseCharacter)) {

**return** true**;**

    } **else** {

**return** false**;**

    }

}

*// Example usage:*

console**.**log(isVowel('a'))**;**  *// Output: true*

console**.**log(isVowel('b'))**;**  *// Output: false*

console**.**log(isVowel('E'))**;**  *// Output: true (case-insensitive)*

console**.**log(isVowel('z'))**;**  *// Output: false*

We define a function called isVowel that takes a single parameter character.

We convert the character to lowercase using toLowerCase() method to handle both uppercase and lowercase characters uniformly.

We define an array vowels containing the vowels.

We use the includes() method to check if the character is included in the vowels array.

If the character is a vowel, the function returns true, otherwise, it returns false.

Example usage demonstrates how to use this function with different characters.

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

 function checkNumber(**number**) {

**if** (number **>** 0) {

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

    } **else** **if** (number **<** 0) {

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

    } **else** {

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

    }

}

*// Example usage:*

checkNumber(5)**;**     *// Output: The number is positive.*

checkNumber(**-**10)**;**   *// Output: The number is negative.*

checkNumber(0)**;**     *// Output: The number is zero.*

We define a function called checkNumber that takes a single parameter number.

Inside the function, we use conditional statements to check the value of the number.

If the number is greater than 0, the function returns "Positive".

If the number is less than 0, the function returns "Negative".

If the number is equal to 0, the function returns "Zero".

Example usage demonstrates how to use this function with different numbers.

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

function checkEvenOrOdd(**number**) {

**return** number **%** 2 **===** 0 **?** "Even" **:** "Odd"**;**

}

*// Example usage:*

console**.**log(checkEvenOrOdd(4))**;**   *// Output: Even*

console**.**log(checkEvenOrOdd(7))**;**   *// Output: Odd*

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

function findMaxNumber(**a,** **b,** **c**) {

**return** a **>** b **?** (a **>** c **?** a **:** c) **:** (b **>** c **?** b **:** c)**;**

}

*// Example usage:*

console**.**log(findMaxNumber(5**,** 8**,** 3))**;**   *// Output: 8*

console**.**log(findMaxNumber(10**,** 2**,** 7))**;**  *// Output: 10*

console**.**log(findMaxNumber(1**,** 1**,** 1))**;**   *// Output: 1 (All numbers are equal)*

1. Write to find minimum number among 3 numbers using ternary operator in JS?

function findMinNumber(**a,** **b,** **c**) {

**return** a **<** b **?** (a **<** c **?** a **:** c) **:** (b **<** c **?** b **:** c)**;**

}

*// Example usage:*

console**.**log(findMinNumber(5**,** 8**,** 3))**;**   *// Output: 3*

console**.**log(findMinNumber(10**,** 2**,** 7))**;**  *// Output: 2*

console**.**log(findMinNumber(1**,** 1**,** 1))**;**   *// Output: 1 (All numbers are equal)*

1. Write to find the largest of three numbers in JS?

 function findLargestNumber(**a,** **b,** **c**) {

    let largest **=** a**;** *// Assume 'a' is the largest initially*

**if** (b **>** largest) {

        largest **=** b**;** *// Update 'largest' if 'b' is greater*

    }

**if** (c **>** largest) {

        largest **=** c**;** *// Update 'largest' if 'c' is greater*

    }

**return** largest**;**

}

*// Example usage:*

console**.**log(findLargestNumber(5**,** 8**,** 3))**;**   *// Output: 8*

console**.**log(findLargestNumber(10**,** 2**,** 7))**;**  *// Output: 10*

console**.**log(findLargestNumber(1**,** 1**,** 1))**;**   *// Output: 1 (All numbers are equal)*

1. Q.29 Write to show

i. Monday to Sunday using switch case in JS?

ii. Vowel or Consonant using switch case in JS?

i. Monday to Sunday using switch case in JS?

function displayDayOfWeek(**dayNumber**) {

**switch** (dayNumber) {

**case** 1:

            console**.**log("Monday")**;**

**break;**

**case** 2:

            console**.**log("Tuesday")**;**

**break;**

**case** 3:

            console**.**log("Wednesday")**;**

**break;**

**case** 4:

            console**.**log("Thursday")**;**

**break;**

**case** 5:

            console**.**log("Friday")**;**

**break;**

**case** 6:

            console**.**log("Saturday")**;**

**break;**

**case** 7:

            console**.**log("Sunday")**;**

**break;**

**default**:

            console**.**log("Invalid day number. Please enter a number between 1 and 7.")**;**

    }

}

*// Example usage:*

displayDayOfWeek(1)**;**  *// Output: Monday*

displayDayOfWeek(4)**;**  *// Output: Thursday*

displayDayOfWeek(7)**;**  *// Output: Sunday*

displayDayOfWeek(9)**;**  *// Output: Invalid day number. Please enter a number between 1 and 7.*

ii. Vowel or Consonant using switch case in JS?

function checkVowelOrConsonant(**character**) {

**switch** (character**.**toLowerCase()) {

**case** 'a':

**case** 'e':

**case** 'i':

**case** 'o':

**case** 'u':

            console**.**log("Vowel")**;**

**break;**

**default**:

            console**.**log("Consonant")**;**

    }

}

*// Example usage:*

checkVowelOrConsonant('a')**;**   *// Output: Vowel*

checkVowelOrConsonant('b')**;**   *// Output: Consonant*

checkVowelOrConsonant('E')**;**   *// Output: Vowel*

checkVowelOrConsonant('z')**;**   *// Output: Consonant*

1. What are the looping structures in JavaScript? Any one Example?

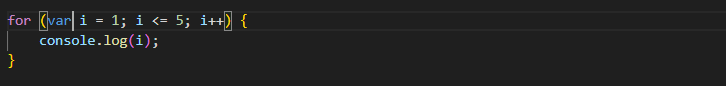
* Ans. There are **Three** types of Loop.

1)for loop

2)while loop

3)do while loop

* **Example:-**



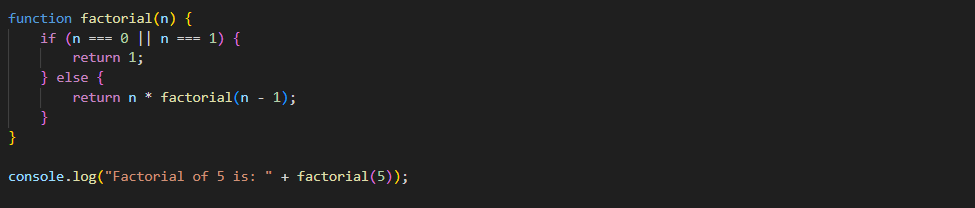
1. **Write a print 972 to 897 using for loop in JS?**

**Ans:-**



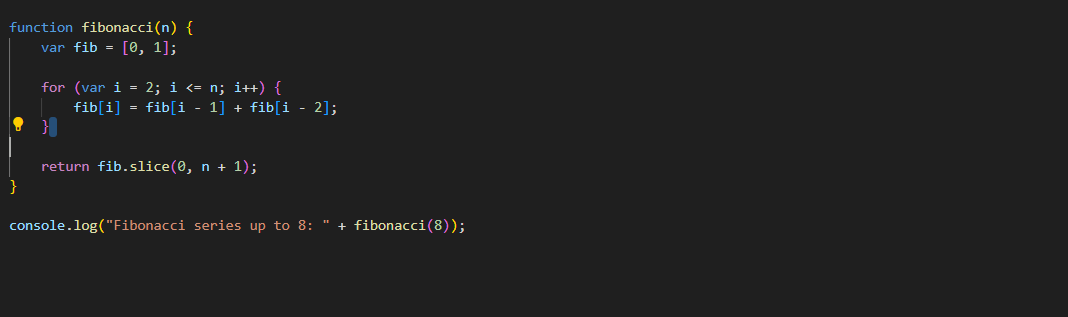
1. **Write to print factorial of given number?**

**Ans:-**



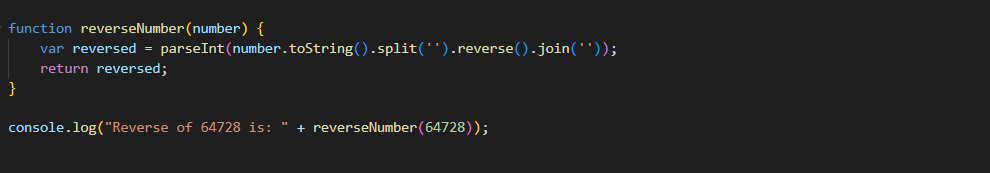
1. **Write to print Fibonacci series up to given numbers?**

**Ans:-**



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

**Ans:-**



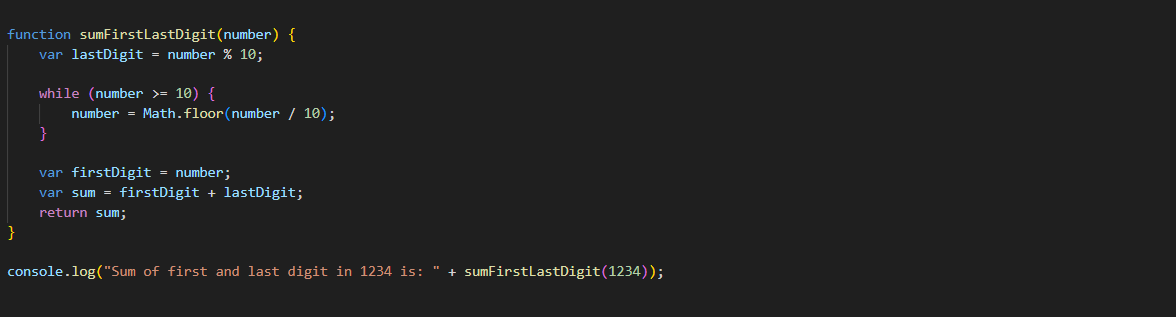
1. **Write a program make a summation of given number (E.g., 1523 Ans: - 11) in JS?**

**Ans:-**



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

**Ans:-**



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

**1 1 1 1 1**

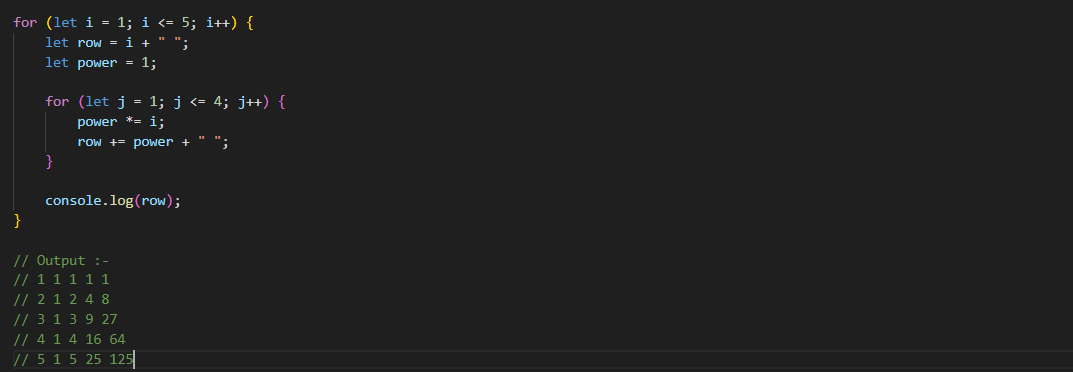
**2 1 2 4 8**

**3 1 3 9 27**

**4 1 4 16 64**

**5 1 5 25 125**

**Ans:-**



**42. Use pattern in console.log in JS?**

**1) 1**

**1 0**

**1 0 1**

**1 0 1 0**

**1 0 1 0 1**

**Ans:-**



**2) A**

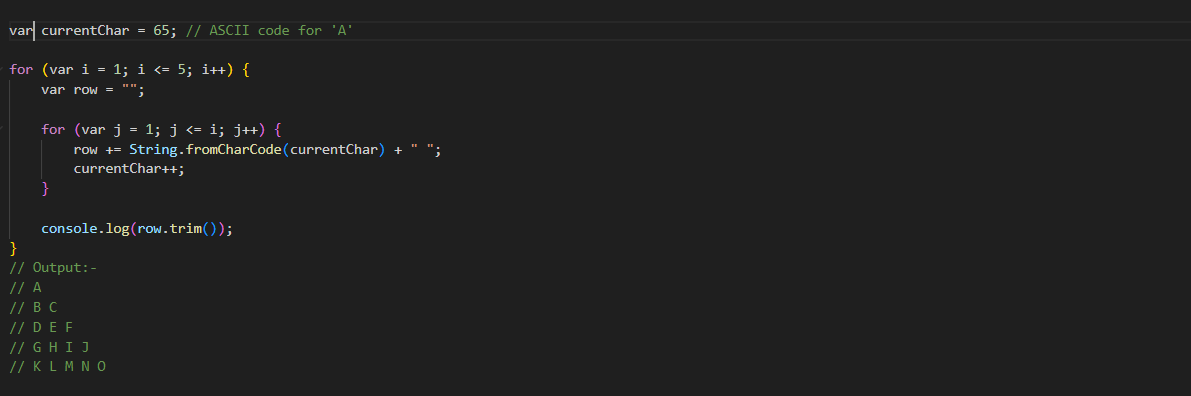
**B C**

**D E F**

**G H I J**

**K L M N O**

**Ans:-**



**3) 1**

**2 3**

**4 5 6**

**7 8 9 10**

**11 12 13 14 15**

**Ans:-**



**4) \***

**\* \***

**\* \* \***

**\* \* \* \***

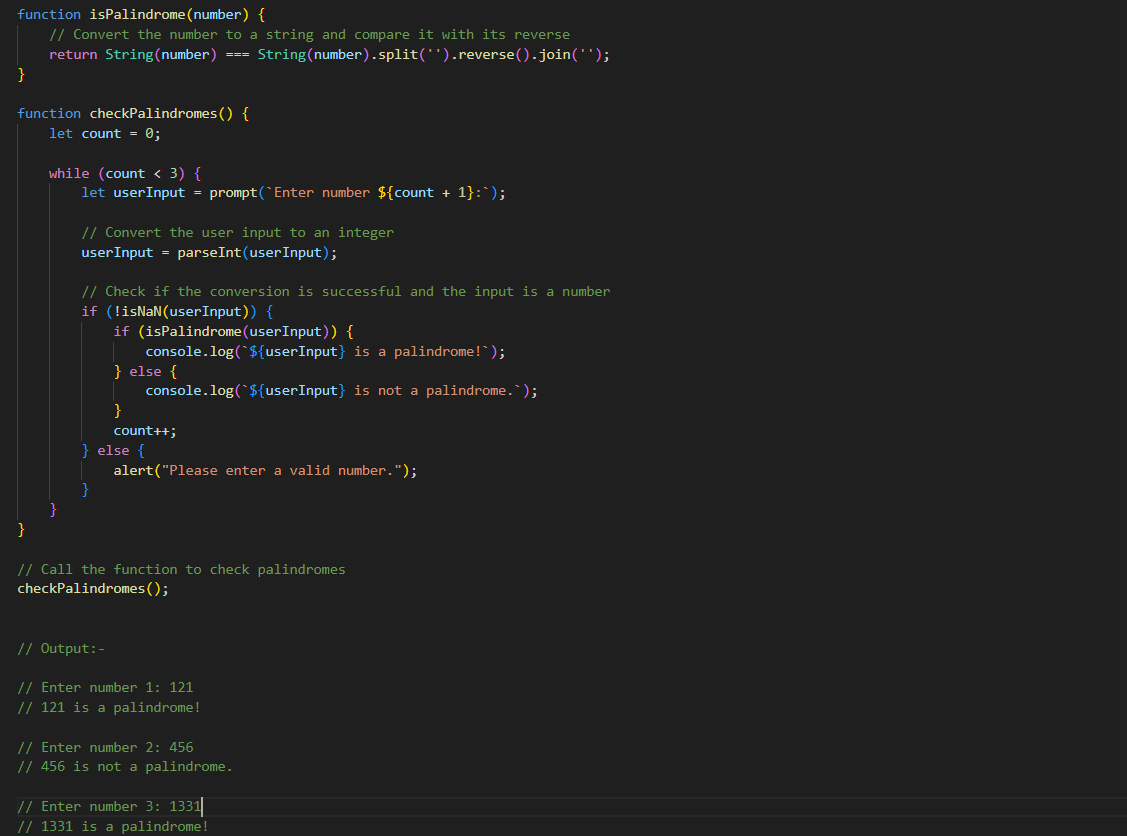
**\* \* \* \* \***

**Ans:-**



1. **Accept 3 numbers from user using while loop and check each numbers palindrome?**

**Ans:-**



**(Array and object Question)**

1. **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 ?**

**Ans:-**



1. **Write a JavaScript program to get the current date?**

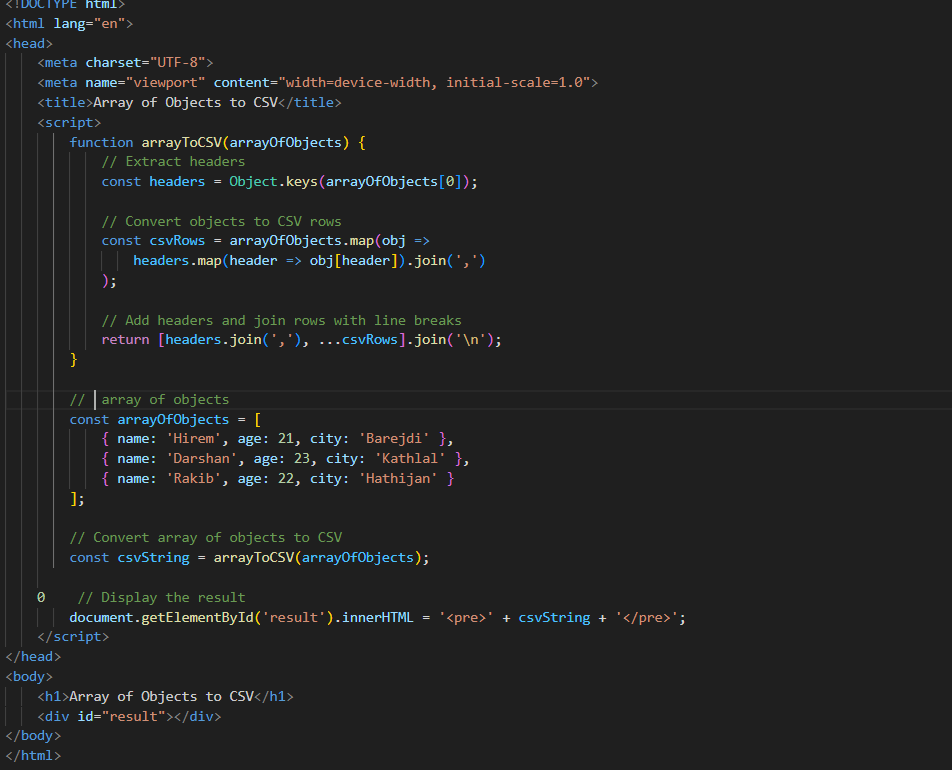
**Ans:-**



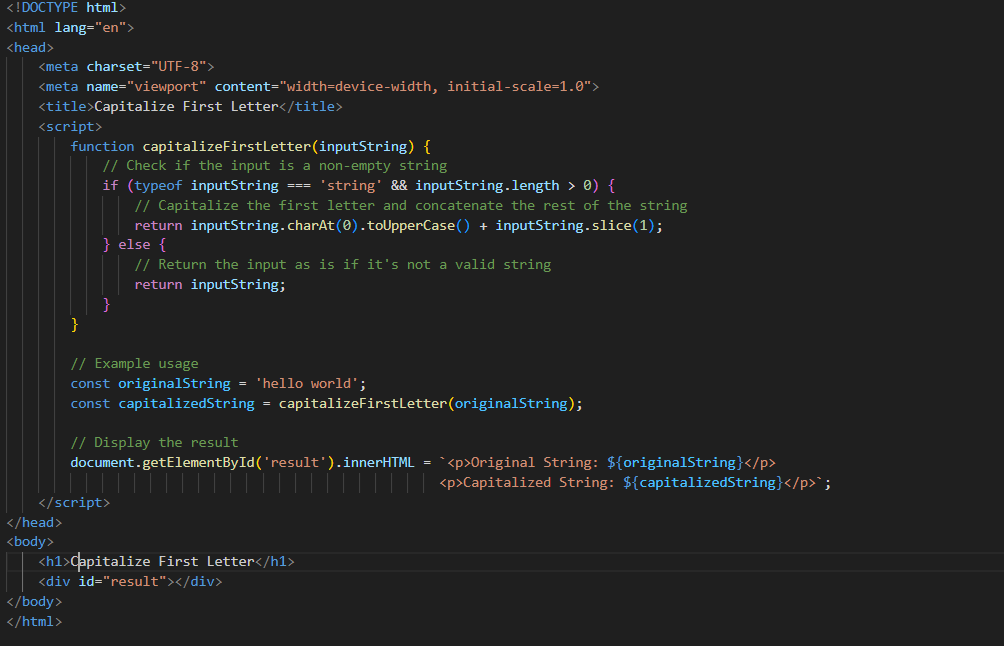
1. **Write a JavaScript program to compare two objects?**

**Ans:-**

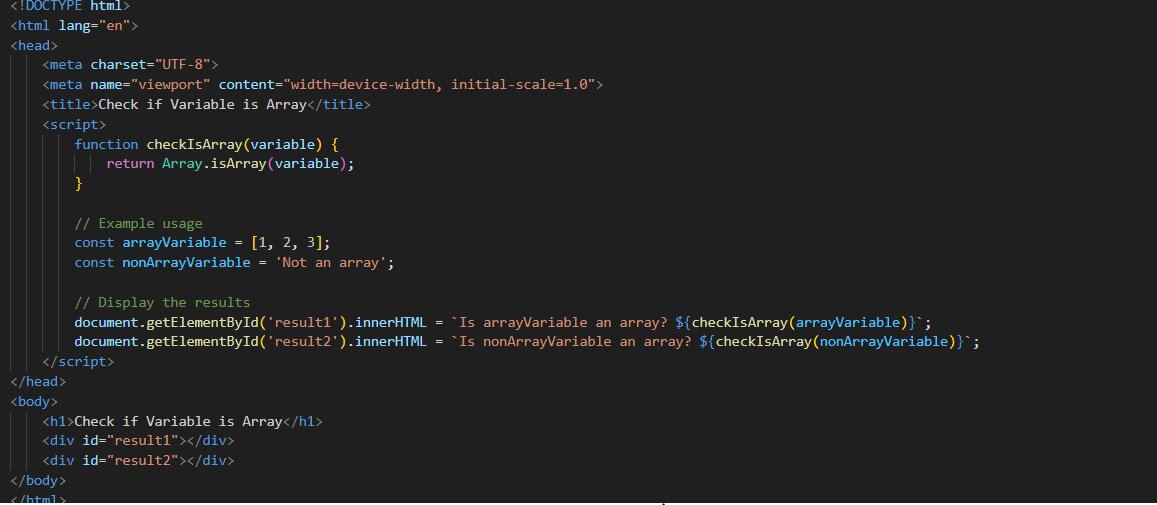


1. **Write a JavaScript program to convert** **an array of objects into CSV string?**

**Ans:-**

1. **Write a JavaScript program to capit**
2. **Write a JavaScript program to determine if a variable is array?**

**Ans:-**



1. **Write a JavaScript program to clone an array?**

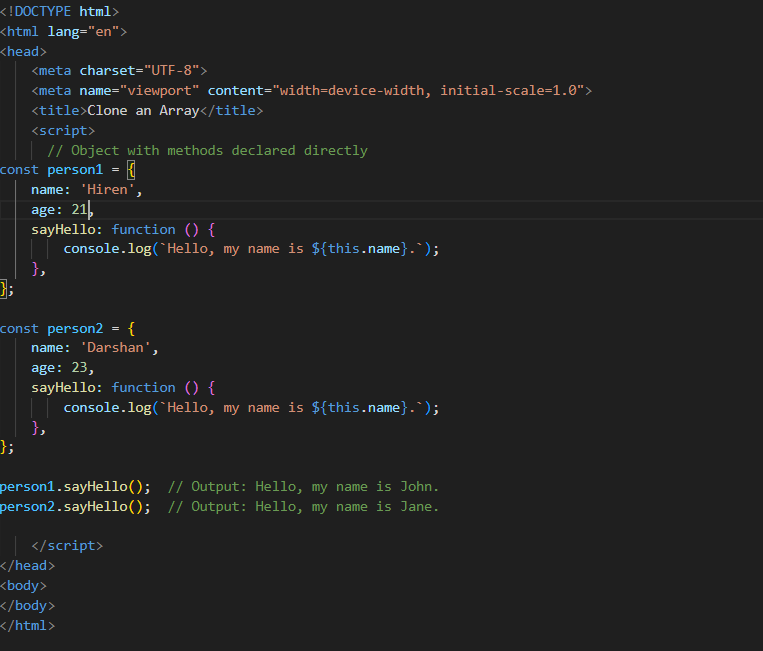
**Ans:-**



1. **What is the drawback of declaring methods directly in JavaScript objects?**

**Ans:-**

* **Declaring Methods Directly in Methods**
* **Example:-**



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

*// Define a string*

var myString **=** "Hello, world!"**;**

*// Print the length of the string to the console*

console**.**log("Length of the string:"**,** myString**.**length)**;**

1. Change all the string characters to capital letters using toUpperCase() method.

*// Define a string*

var myString **=** "Hello, world!"**;**

*// Convert all characters to uppercase*

var uppercaseString **=** myString**.**toUpperCase()**;**

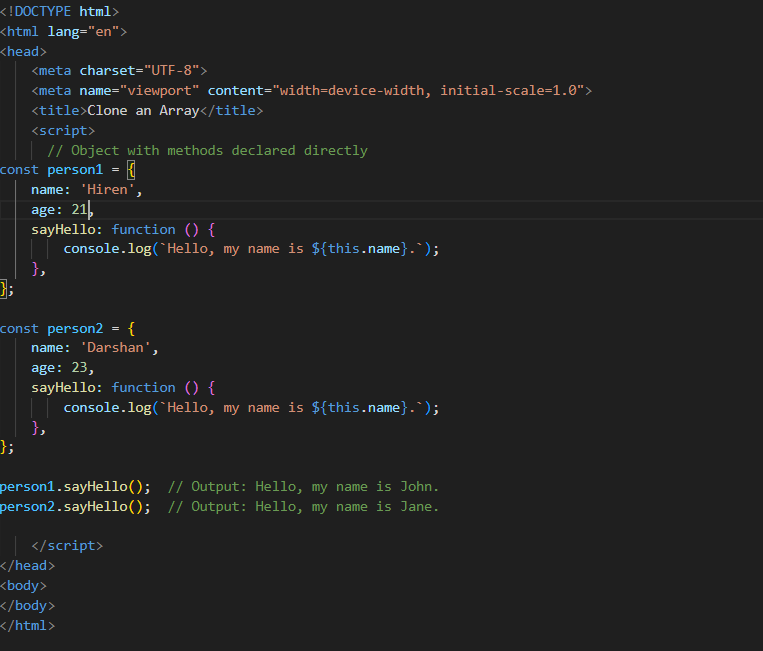
*// Print the uppercase string to the console*

console**.**log(uppercaseString)**;**

1. **What is the drawback of declaring methods directly in JavaScript objects?**

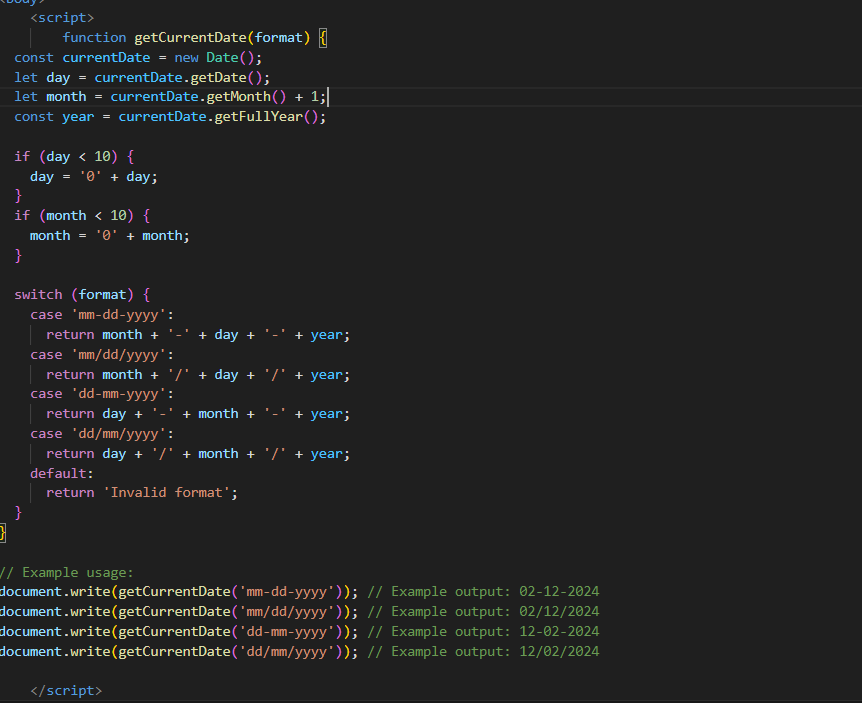
**Ans:-**

* **Declaring Methods Directly in Methods:-**



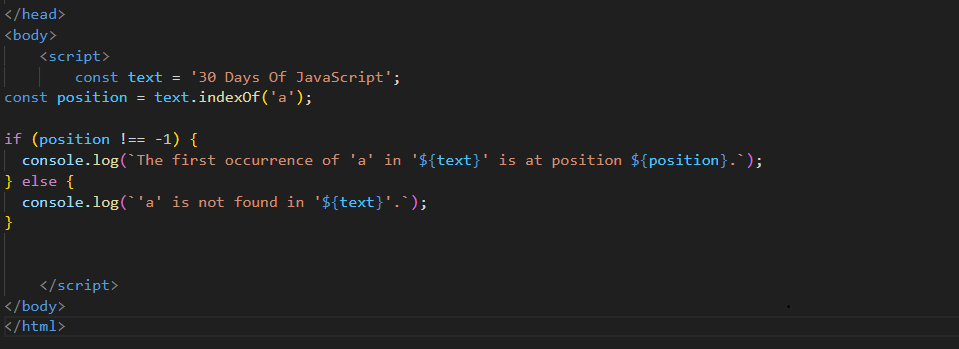
1. **Write a JavaScript program to get the current date. Expected Output : mm-dd-yyyy, mm/dd/yyyy or dd-mm-yyyy, dd/mm/yyyy?**

**Ans:-**



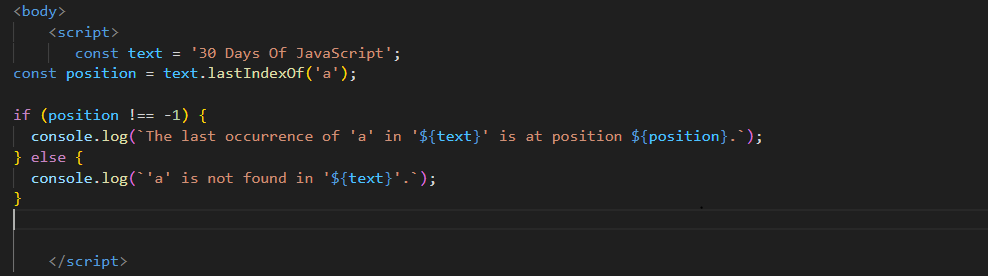
1. **Use indexOf to determine the position of the first occurrence of a in 30 Days Of JavaScript?**

**Ans:-**

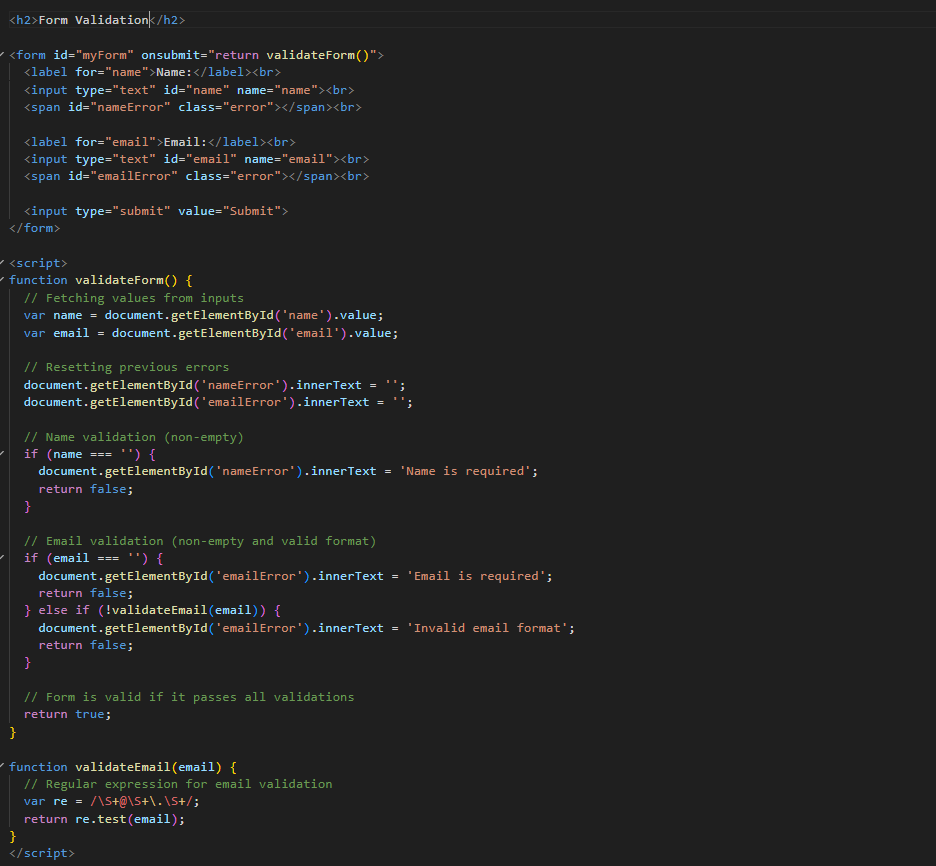


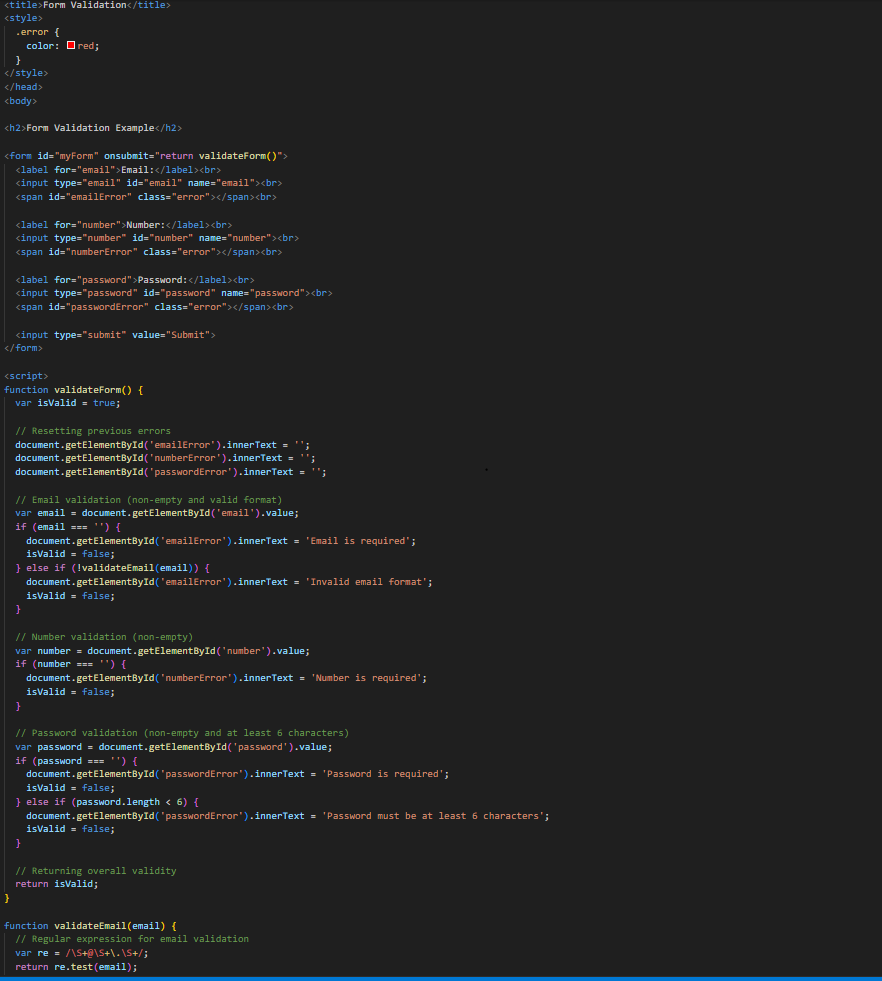
1. **Use lastIndexOf to determine the position of the last occurrence of a in 30 Days Of JavaScript?**

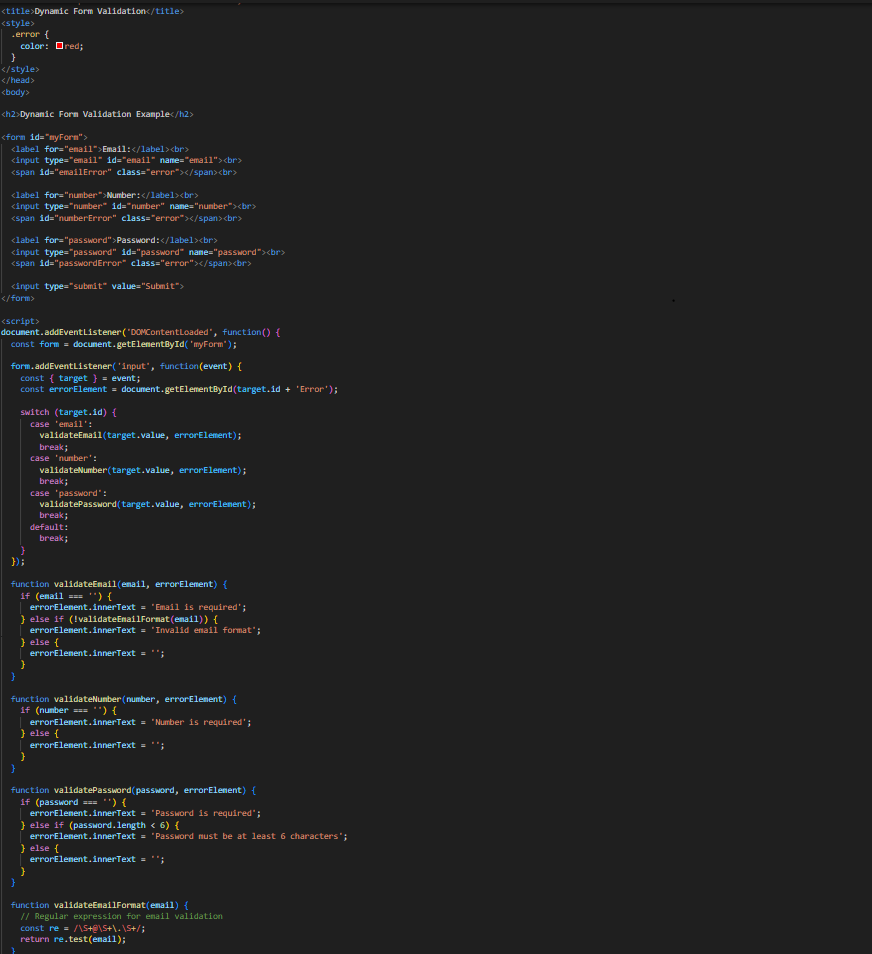
**Ans:-**



1. **Form Validtion in JS?**

**Ans:-** 

1. **Form in Email, number, Password, Validatio**
2. **Dynamic Form Validation in JS?**

**Ans:-** 

1. **how many type of JS Event? How to use it ?**

**Ans:-**

* There are few types of Event in JS.

1. Mouse Event :- These events are triggered by user actions involving the mouse, such as clicking, hovering, or scrolling.
2. Keyboard Event:- Events that are triggered by keyboard interactions, such as pressing keys or releasing keys.
3. Form Event:- Events related to form elements, such as submitting a form, changing input values, or focusing on input fields.
4. Custom Event:- Events that can be defined and triggered by the user or the developer.

* To use these events, you can attach event listeners to DOM elements using methods like addEventListener
* **Example:-**



1. **What is Bom vs Dom in JS?**

**Ans:-**

* **DOM (Document Object Model):** It represents the structure of HTML or XML documents as a tree-like model where each node represents an object in the document.

It allows scripts to dynamically access and modify the content, structure, and style of a document.

* **Example:-**

**console.log(document);**

* **BOM (Browser Object Model):** It provides objects and methods to interact with the browser window.

It includes objects such as window, document, location, history, navigator, etc.

The BOM is not standardized, and its features may vary between different browsers.

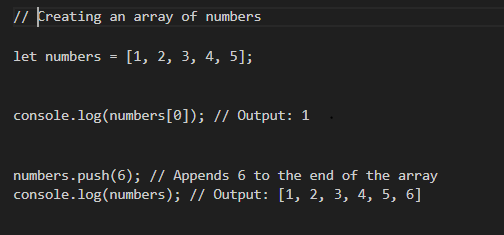
* **Example:-**

**console.log(window);**

1. **Array vs object defences in JS?**

**Ans:-**

* **Array:-**
* Arrays are ordered collections of data.
* Elements in arrays are accessed by their index.
* Arrays are ideal for storing lists of similar items or ordered data.
* **Example:-**



* **Object:-**
* Objects are collections of key-value pairs, where keys are strings (or symbols) and values can be of any data type.
* Objects are unordered, meaning there is no guarantee of the order in which properties are stored.
* They areideal for representing structured data or mapping properties to values**.**
* **Example:-**

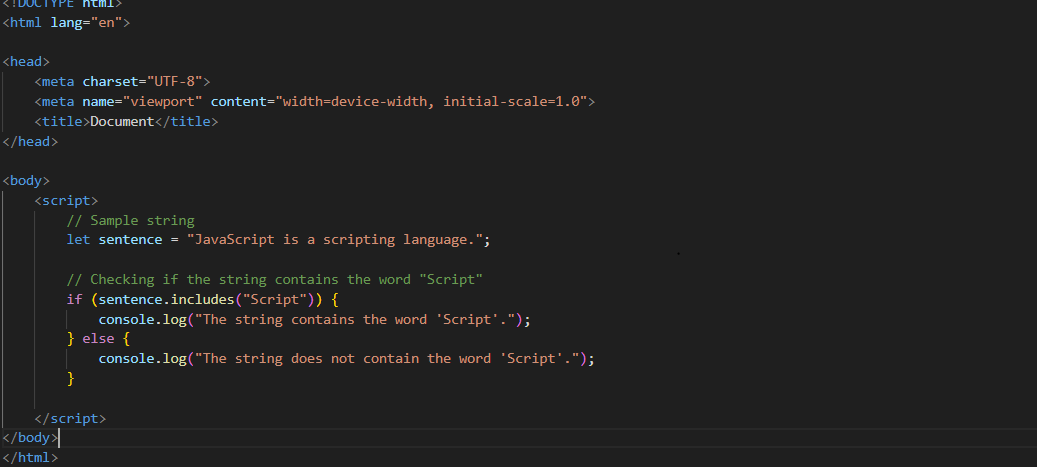


1. **Split the string into an array using split() Method?**

**Ans:-** 

1. **Check if the string contains a word Script using includes() method?**

**Ans:-**



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

**Ans:-**



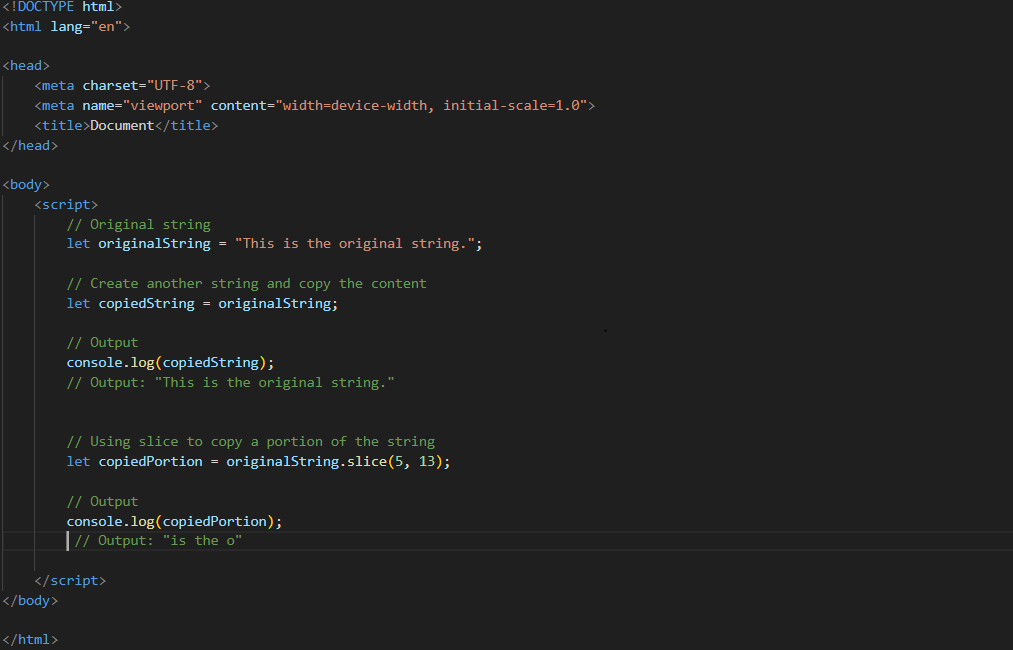
1. **What is Character at index 15 in ’30 Days of JavaScript’ string? Use charAt() method.**

**Ans:-**



**Q.68. copy to one string to another string in JS?**

**Ans:-**



**Q.69. Find the length of a string without using libraryFunction?**

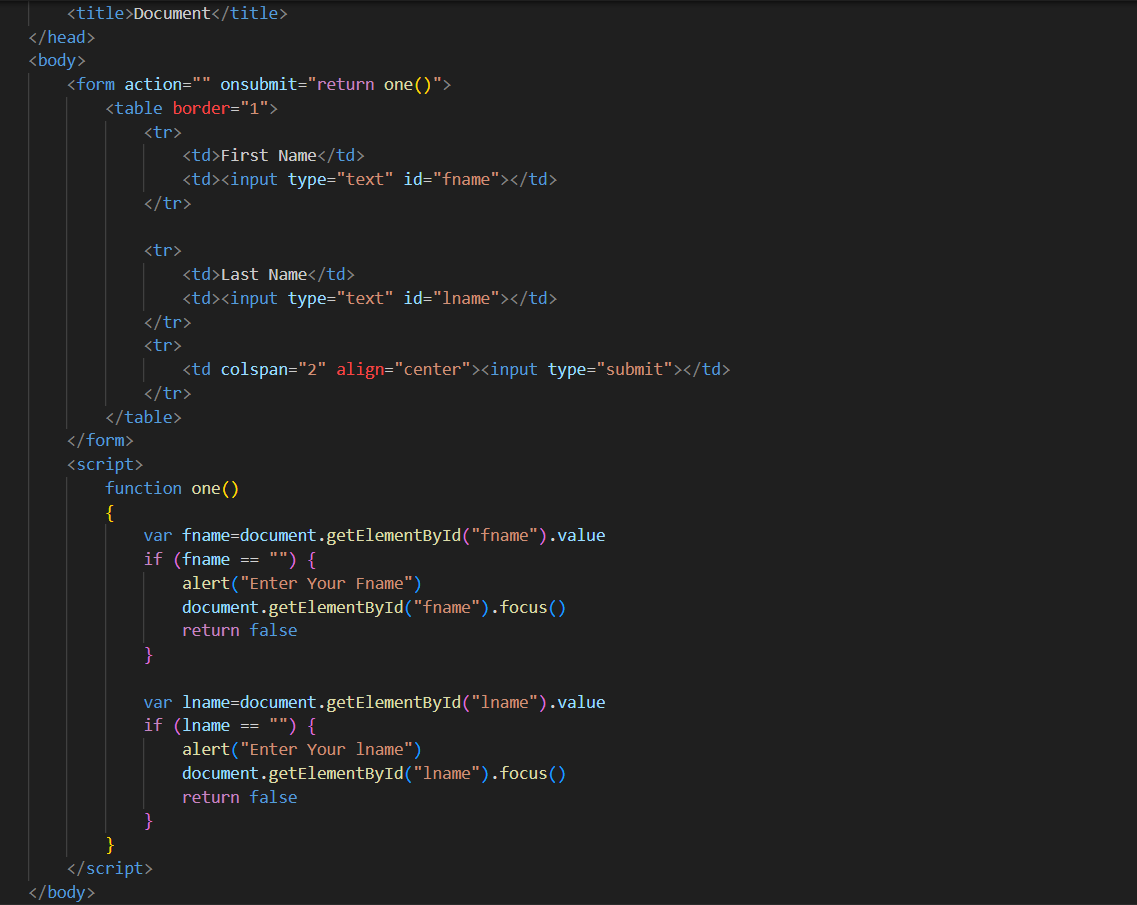
**Ans:-**



**Q.1) What is JavaScript?**

**Ans:-**

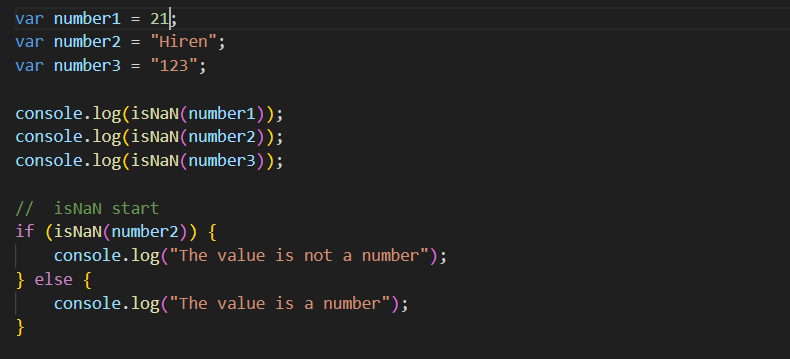
* JavaScript is a high-level, interpreted programming language primarily used for creating interactive and dynamic content on websites.
* It allows you to manipulate the content, structure, and behavior of a web page.
* JavaScript is used in conjunction with HTML and CSS to enhance the user experience.
* **Example:-**



**Q.2) What is the use of is NaN function?**

**Ans:-**

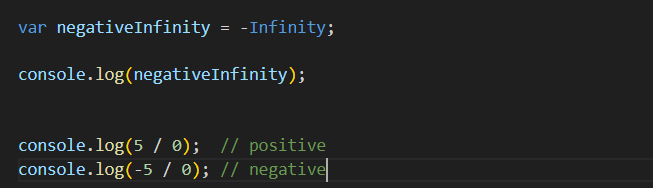
* The NaN full form is " Not a Number."
* It is used to determine whether a value is NaN or can be converted to a number.
* The function returns a Boolean value, true if the value is NaN, and false if it is a valid number or can be converted to one
* **Example:-**



**Q.3) What is negative Infinity?**

**Ans**:-

* Infinity is a special numeric value representing positive infinity.
* there is called -Infinity.
* which represents negative infinity.
* **Example:-**



**Q.4) Which company developed JavaScript?**

**Ans:-**

* JavaScript developed by Netscape Communications Corporation.
* The development was led by Brendan Eich, and the language was introduced in Netscape Navigator browser in 1995.
* It was original name "Mocha" and later "LiveScript" before settling on the name JavaScript.

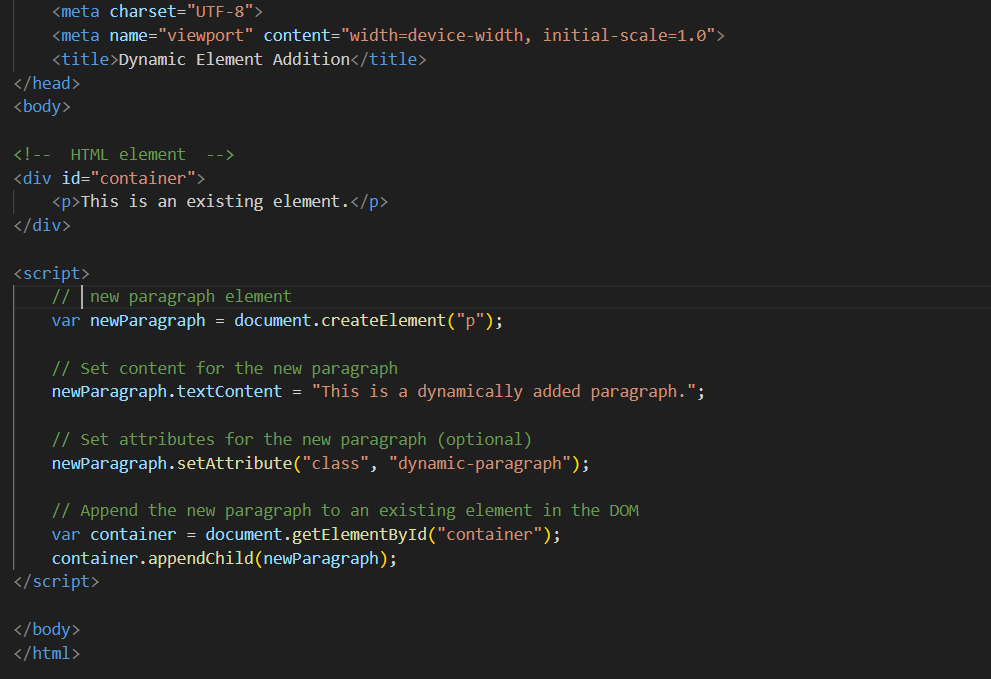
**Q.5) What are undeclared and undefined variables?**

**Ans:-**

* **Undeclared Variable:-**
  + Undeclared variable is used in code without declared using the var, let, or const keyword.
  + Undeclared variable can lead to unexpected behavior and is generally considered a coding mistake.
* **Undefined Variable :-**
  + Undefined variable is a variable that has been declared but not assigned a value.
  + A variable is declared but not initialized, it automatically gets the value undefined.

**Q.6) Write the code for adding new elements dynamically?**

**Ans:-**



**Q.7) What is the difference between View State and Session State?**

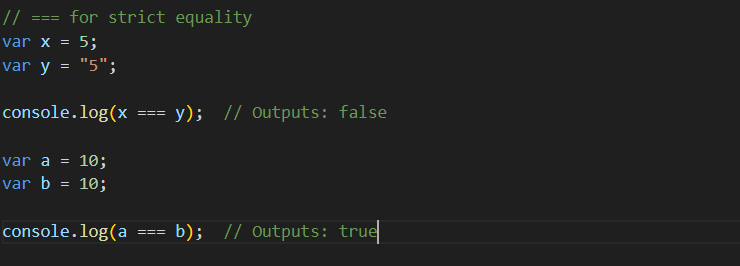
**Ans:-**

* **View State:**
  + - View State is used to persist the state of a single web page across postbacks.
    - It's a client-side state management technique, and the data is stored in a hidden field on the web page.
* **Session State:**
  + Session State is used to persist data across multiple requests and pages for a particular user session.
  + It's a server-side state management technique, and the data is stored on the server.

**Q.8) What is === operator?**

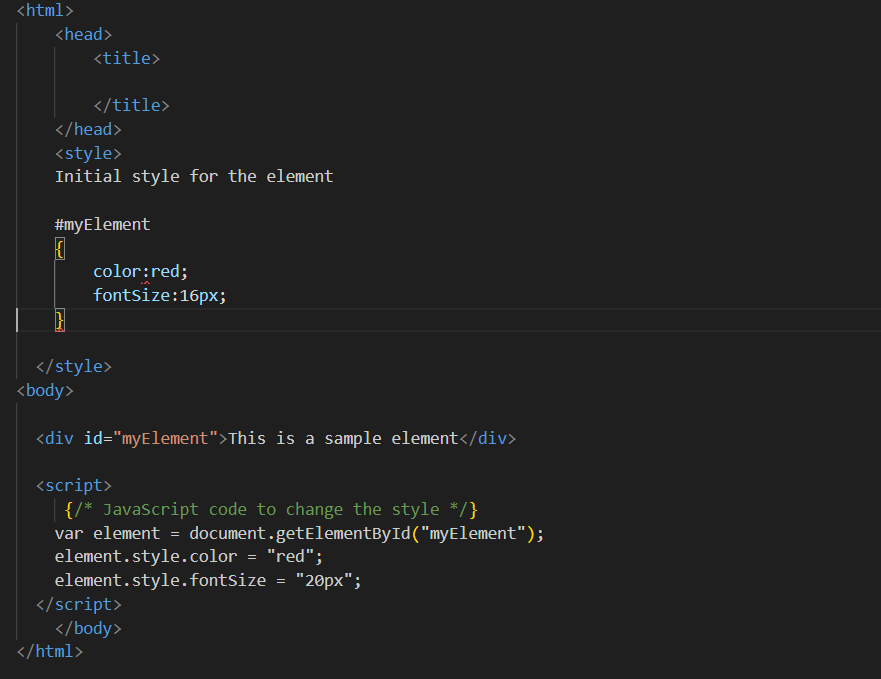
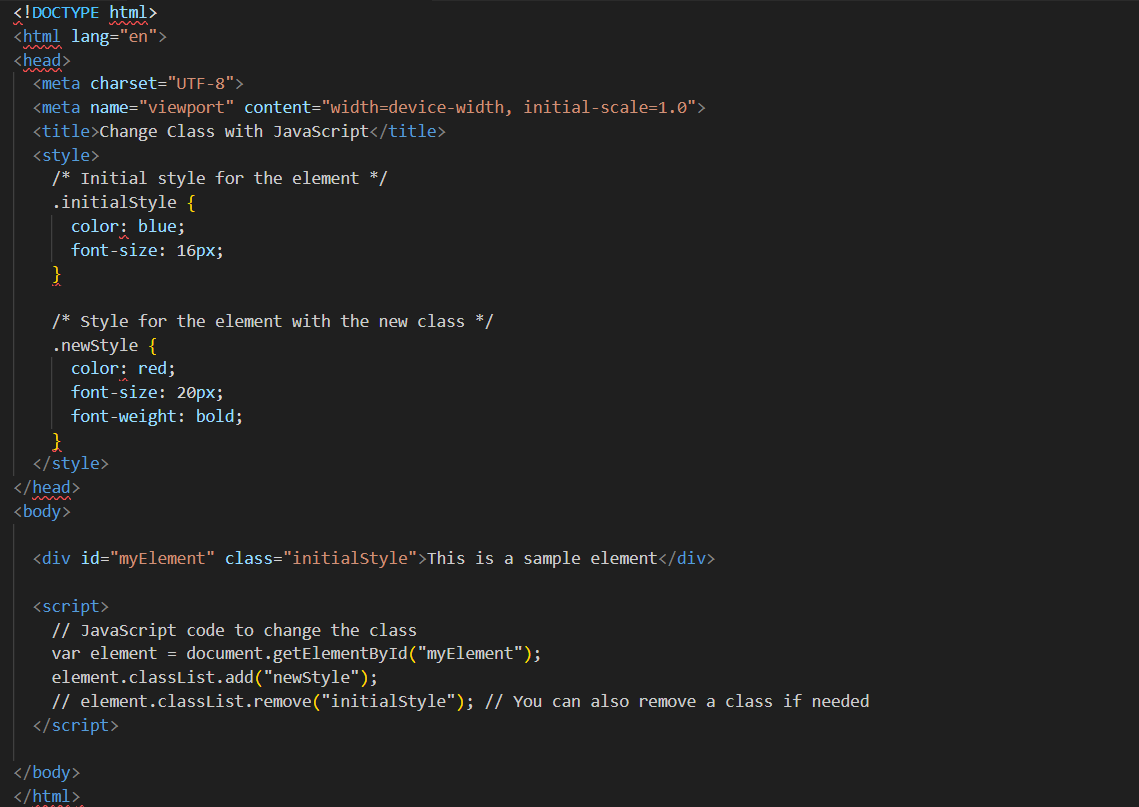
**Ans:-**

* the === operator is a strict equality operator.
* It checks whether its two operands are equal, both in value and in data type.
* If the operands are of the same type and have the same value, === returns true; otherwise, it returns false.
* **Example:-**



**Q.9) How can the style/class of an element be changed?**

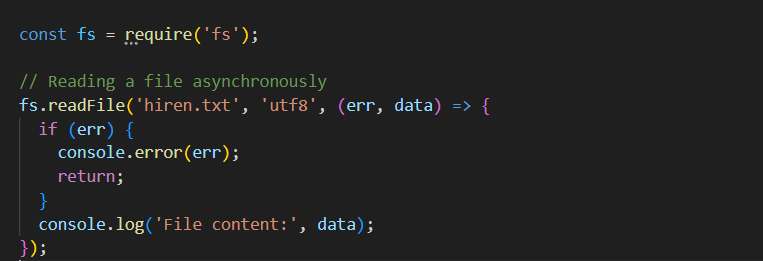
**Ans:-**

1. Changing Style with Java Script:-
2. Changing Class with JavaScript:-

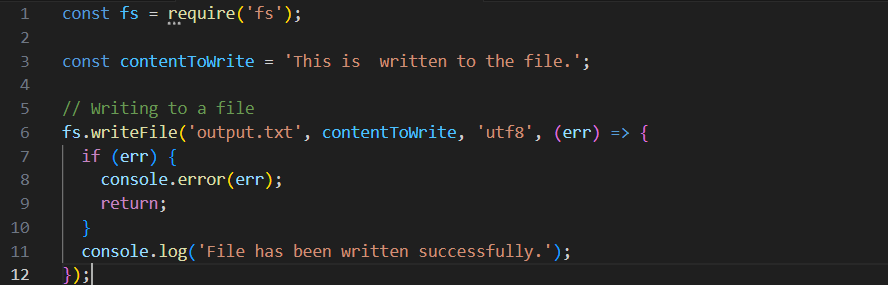
**Q.10) How to read and write a file using JavaScript?**

**Ans:-**

* **Read a File:-**



* **Writing a File:-**



**Q.11)What are all the looping structures in JavaScript?**

**Ans:-**

1. **For loop:-**

**Ex.** for (let i = 0; i < 5; i++)

{

console.log(i);

}

**// Output: 0, 1, 2, 3, 4**

1. **While loop:-**

**Ex.** let i = 0;

while (i < 5)

{

console.log(i);

i++;

}

**// Output: 0, 1, 2, 3, 4**

1. **Do while loop:-**

**Ex.** let i = 0;

Do

{

console.log(i);

i++;

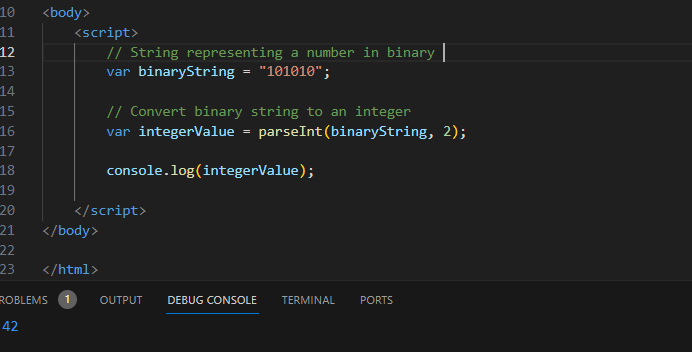
}

while (i < 5);

**// Output: 0, 1, 2, 3, 4**

**Q.12) How can you convert the string of any base to an integer in JavaScript?**

**Ans:-**



**Q.13)What is the function of the delete operator?**

**Ans:-**

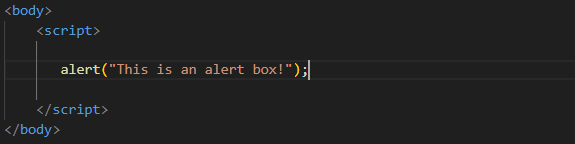


**Q.14)What are all the types of Pop up boxes available in JavaScript?**

**Ans:-**

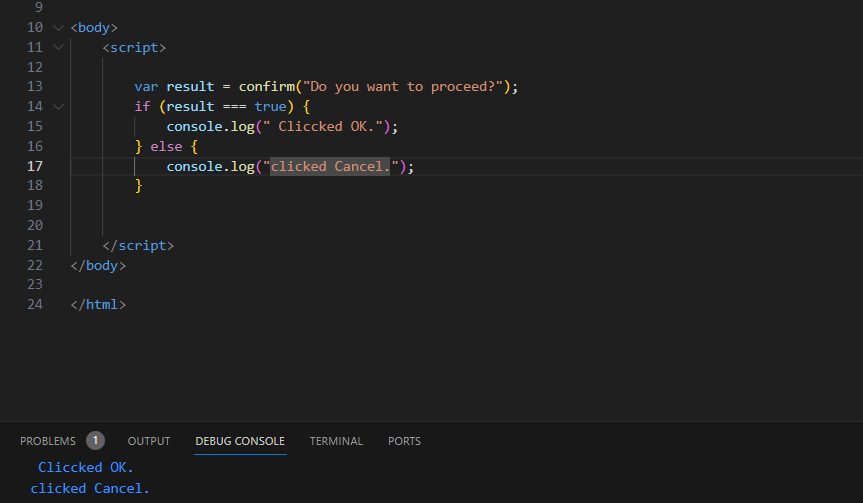
1. **Alert box:-** Displays a message to the user with an "OK" button to acknowledge the message.

**Example:-**



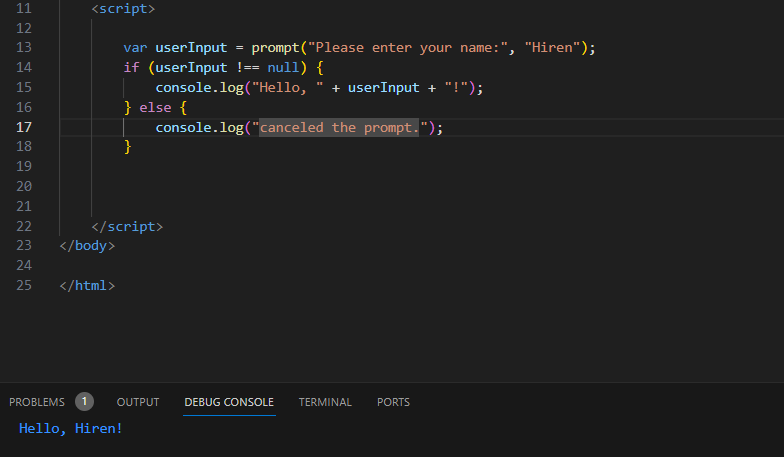
1. **Confirm box:-** Displays a message to the user with "OK" and "Cancel" buttons, allowing the user to confirm or cancel an action.

**Example:-**



1. **Prompt box:-** Displays a message to the user with an input field for the user to enter data. It returns the text entered by the user when they click "OK", or null if they click "Cancel".

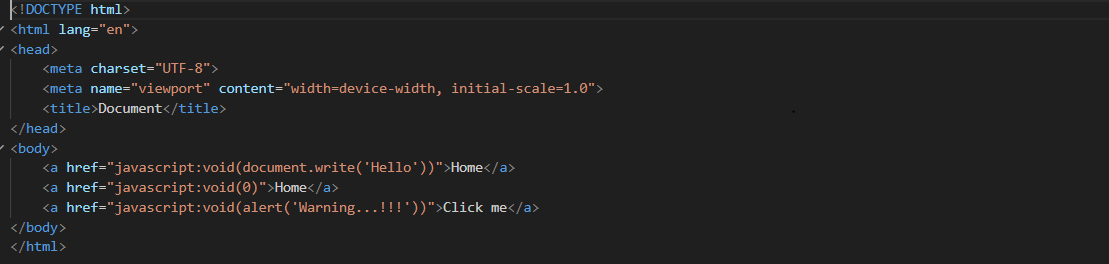
**Example:-**



**Q.15) What is the use of Void (0)?**

**Ans:-**

* Void means return.
* **Example:-**



**Q.16) How can a page be forced to load another page in JavaScript?**

**Ans:-**

* Js can force a page to load another page using the window.location object. There are a few different ways to this:
* window.location.href = "www rakib.com";
* window.location.replace("www.rakib.com");
* window.location.assign("www. rakib .com");