JAVASCRIPT ASSIGNMENT

1. What is JavaScript. How to use it?

Ans :

=> JavaScript is a high-level, interpreted programming language that is commonly used to make web pages interactive and dynamic.

Client-side scripting language: Primarily executed in a web browser, making web pages interactive and responsive to user actions.

2) How many type of Variable in JavaScript?

Ans :

=> There are primarily three ways to declare variables in JavaScript.

1. let (introduced in ES6)
2. const (introduced in ES6)

1. var (traditional, but less preferred due to potential scoping issues)

3) Define a Data Types in js?

Ans:

=> JavaScript has two main categories of data types:

1. Primitive Data Types:

* Number: Represents numeric values, including integers (whole numbers) and floating-point numbers (decimals). Examples: 10, -5.2, 3.14159.
* String: Represents a sequence of characters enclosed in single or double quotes. Examples: "Hello, world!", 'This is a string'.
* Boolean: Represents logical values: true or false. Examples: true (something is true), false (something is false).
* Null: Represents the intentional absence of a value. It signifies that there is no value assigned to the variable. Example: null.
* Undefined: Represents a variable that has been declared but not yet assigned a value. Example: let x; // x is undefined.
* Symbol (introduced in ES6): A unique and immutable (cannot be changed) primitive value that represents an identifier.

2. Non-Primitive Data Types:

* Object: A collection of key-value pairs, where keys are unique strings and values can be any data type, including other objects.
* Array: An ordered collection of items, where each item can be accessed using an index

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

Ans:

=>function mul(x) { return function(y) { return function(z) {

// Return the product of all arguments return x \* y \* z;

};

};

}

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

Ans:

=>

Undefined: A variable has been declared but not yet assigned a value.

Undeclared:The variable has not been declared at all. It doesn't exist in the current scope.

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:

Ans : =>

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? Ans:

=> This code checks if the type of '10' (which is a string) is exactly equal to the type of 10 (which is a number).

If not, it converts '10' to a number using the Number() function. After conversion, both will be of the same type, and the code will print 10.

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

Ans:

=>You can write a JavaScript program to find the area of a triangle using the formula:

Area = (base \* height) / 2

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

Ans :

=>

// Function to calculate the number of days left until next Christmas

function daysUntilChristmas() { // Get the current date let currentDate = new Date();

// Get the current year

let currentYear = currentDate.getFullYear();

// Create a new Date object for Christmas of the current year

let christmasDate = new Date(currentYear, 11, 25); // Month is 0-based index (11 represents December)

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

if (currentDate.getMonth() === 11 && currentDate.getDate()

> 25) {

christmasDate.setFullYear(currentYear + 1);

}

// Calculate the difference in milliseconds between the current date and Christmas

let timeDiff = christmasDate.getTime() - currentDate.getTime();

// Calculate the number of days left until Christmas let daysLeft = Math.ceil(timeDiff / (1000 \* 3600 \* 24));

return daysLeft;

}

// Example usage:

let daysLeft = daysUntilChristmas();

console.log("Days left until next Christmas:", daysLeft);

10) What is Condition Statement?

Ans:

=> A condition statement, often referred to simply as a "conditional," is a programming construct that allows you to execute different blocks of code based on whether a specified condition evaluates to true or false.

1. if statement
2. else if statement
3. switch statement
4. Find circumference of Rectangle formula : C = 4 \* a ?

Ans:

=> The formula you provided, C=4×a, calculates the perimeter of a rectangle, not the circumference. In a rectangle, the perimeter is the sum of all its sides.

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

Ans:

=> function yearsToDays(years) { var days = years \* 365; return days;

}

function daysToYears(days) { var years = days / 365; return years;

}

function main() {

var choice = parsent(prompt("Enter 1 to convert years to days or 2 to convert days to years: "));

if (choice === 1) {

var years = parseFloat(prompt("Enter the number of years: "));

var days = yearsToDays(years);

console.log(`${years} years is equal to ${days} days.`);

} else if (choice === 2) { var days = parseFloat(prompt("Enter the number of days:

"));

var years = daysToYears(days);

console.log(`${days} days is equal to ${years} years.`);

} else {

console.log("Invalid choice. Please enter 1 or 2.");

}

}

main();

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

Ans:

=> function fahrenheitToCelsius(fahrenheit) { var celsius;

// Formula to convert Fahrenheit to Celsius: (F - 32) \* 5/9 if (typeof fahrenheit === 'number') { celsius = (fahrenheit - 32) \* 5 / 9; return celsius;

} else {

return "Please enter a valid numerical temperature.";

}

}

var fahrenheitTemperature = parseFloat(prompt("Enter temperature in Fahrenheit: ")); var celsiusTemperature =

fahrenheitToCelsius(fahrenheitTemperature);

if (typeof celsiusTemperature === 'number') { console.log(`${fahrenheitTemperature}°F is equal to ${celsiusTemperature.toFixed(2)}°C.`);

} else {

console.log(celsiusTemperature);

}

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

Ans:

=> <!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Filename Extension Exercise</title>

<script>

function getFileExtension(filename) {

// Split the filename into an array using dot as the delimiter

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

// Check if there's more than one part (at least one dot in the filename)

if (parts.length > 1) {

// Return the last part as the extension return parts[parts.length - 1];

} else {

// No extension found return "No extension found";

}

}

// Example usage

const filename1 = "example.txt"; const filename2 = "document.pdf"; const filename3 = "script\_without\_extension";

console.log(getFileExtension(filename1)); // Output: txt console.log(getFileExtension(filename2)); // Output: pdf console.log(getFileExtension(filename3)); // Output: No extension found

</script>

</head>

<body>

<!-- Exercise content goes here -->

</body>

</html>

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

Ans :

=> The expression (5 > 3 && 2 < 4) involves the logical AND operator (&&).

In these result will be true because both the expression are true.

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

Ans :

=>

1. true is a boolean value and is considered truthy.
2. 1 is a truthy value in many programming languages.
3. "hello" is a non-empty string, which is also considered truthy.

The logical AND operator (&&) returns the first falsy value it encounters or the last value if all are truthy. Since all the expressions in this case are truthy, the result of the expression will be the last truthy value, which is "hello"

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

Ans:

=> `true && false || false && true` involves logical AND (`&&`) and logical OR (`||`) operators.

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

Ans:

=> Loop:-

In JavaScript, a loop is a control structure that allows you to repeatedly execute a block of code as long as a specified condition is true.

Switch Case:-

In JavaScript, the switch statement is used to perform different actions based on different conditions. It evaluates an expression, and depending on the value of the expression, it executes the corresponding block of code.

1. What is the use of is Nan function?

Ans:

=> The is NaN function in JavaScript is used to determine whether a value is NaN (Not-a-Number) or not. NaN is a special value in JavaScript that represents the result of an invalid or undefined mathematical operation.

The isNaN function returns a Boolean value indicating whether the provided value is NaN. It can be used with various types of values, not just numbers.

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

=> The main difference between the && (logical AND) and || (logical OR) operators in JavaScript lies in their behavior and the conditions under which they evaluate to true or false.

&& (Logical AND):

Requires both operands to be true.

|| (Logical OR):

Requires at least one operand to be true.

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

Ans :

=> The void keyword in JavaScript is used to evaluate an expression and then return undefined.

The most common use of void is in combination with the value 0 to create a self-executing anonymous function or to prevent a browser from navigating to a new page when clicking on a link.

1. Check Number Is Positive or Negative in JavaScript?

Ans :

=> 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.");

}

}

checkNumber(5); // Output: The number is positive. checkNumber(-3); // Output: The number is negative. checkNumber(0); // Output: The number is zero.

23) Find the Character Is Vowel or Not ?

Ans:

=> function isVowel(char) {

return ['a', 'e', 'i', 'o', 'u'].indexOf(char.toLowerCase()) !== 1;

}

// Example usage:

var character = prompt("Enter a character: ");

if (character.length === 1 && character.match(/[a-zA-Z]/)) { if (isVowel(character)) {

console.log(character + " is a vowel.");

} else {

console.log(character + " is not a vowel.");

}

} else {

console.log("Please enter a single alphabetical character.");

}

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

Ans:

=> function checkNumber(num) {

if (num > 0) { return "Positive"; } else if (num < 0) { return "Negative";

} else { return "Zero";

}

}

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

Ans:

=> let userInput = prompt("Enter a number: "); let number = parseInt(userInput);

let result = (number % 2 === 0) ? "Even" : "Odd";

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

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

Ans :

=> let num1 = parseFloat(prompt("Enter the first number: ")); let num2 = parseFloat(prompt("Enter the second number: ")); var num3 = parseFloat(prompt("Enter the third number: "));

var maxNumber = (num1 >= num2 && num1 >= num3) ? num1 :

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

console.log(`The maximum number is: ${maxNumber}`);

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

Ans: =>

let num1 = parseFloat(prompt("Enter the first number: ")); let num2 = parseFloat(prompt("Enter the second number: ")); let num3 = parseFloat(prompt("Enter the third number: "));

let minNumber = (num1 <= num2 && num1 <= num3) ? num1 :

(num2 <= num1 && num2 <= num3) ? num2 : num3;

console.log(`The minumum number is: ${minNumber}`); 28) Write to find the largest of three numbers in JS?

Ans:

=> let num1 = parseFloat(prompt("Enter the first number: ")); let num2 = parseFloat(prompt("Enter the second number: ")); let num3 = parseFloat(prompt("Enter the third number: "));

if (!isNaN(num1) && !isNaN(num2) && !isNaN(num3)) { var largestNumber;

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

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

} else {

largestNumber = num3;

}

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

} else {

console.log("Invalid input. Please enter valid numbers.");

}

29) Write to show

i. Monday to Sunday using switch case in JS? ii. Vowel or Consonant using switch case in JS?

Ans :

=> let dayNumber = parseInt(prompt("Enter a number (1-7) representing a day of the week:"));

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 input. Please enter a number between 1 and 7.");

}

ii. Vowel or Consonant using switch case in JS?

let character = prompt("Enter a single alphabet character:");

switch (character.toLowerCase()) { case 'a': case 'e': case 'i': case 'o': case 'u':

console.log("Vowel"); break; default:

console.log("Consonant");

}

\* Conditional looping logic Question \*

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

Ans:

=> In JavaScript, there are several looping structures that allow you to repeatedly execute a block of code. The most common ones are: 1. for loop

1. while loop
2. do while loop
3. for in loop
4. for of loop

for loop:- printing 1 to 5 number: for (let i = 0; i < 5; i++) { console.log(i);

}

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

Ans :

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

}

1. Write to print factorial of given number?

Ans :

=> function factorial(n) { if (n === 0 || n === 1) { return 1; } else { let result = 1; for (let i = 2; i <= n; i++) { result \*= i;

}

return result;

}

}

let number = 5; let result = factorial(number);

console.log(`The factorial of ${number} is: ${result}`); 33) Write to print Fibonacci series up to given numbers?

Ans:

=> function fibonacciSeries(limit) { let fibArray = [0, 1];

for (let i = 2; fibArray[i - 1] + fibArray[i - 2] <= limit; i++) { fibArray[i] = fibArray[i - 1] + fibArray[i - 2];

}

return fibArray;

}

let limit = 50;

let series = fibonacciSeries(limit);

console.log(`Fibonacci series up to ${limit}: ${series.join(',

')}`);

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

Ans :

=> function reverseNumber(number) { let numberString = number.toString();

let reversedString = numberString.split('').reverse().join(''); let reversedNumber = parseInt(reversedString);

return reversedNumber;

}

let originalNumber = 64728;

let reversedNumber = reverseNumber(originalNumber); console.log(`Original number: ${originalNumber}`); console.log(`Reversed number: ${reversedNumber}`);

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

Ans :

=> function calculateDigitSum(number) { let numberString = number.toString(); let sum = 0;

for (let i = 0; i < numberString.length; i++) { sum += parseInt(numberString[i]);

}

return sum;

}

let givenNumber = 1523;

let digitSum = calculateDigitSum(givenNumber); console.log(`Summation of ${givenNumber}: ${digitSum}`); 36) Write a program you have to make a summation of first and last Digit. (E.g., 1234 Ans: -5) in JS?

Ans :

=> function calculateFirstAndLastDigitSum(number) { let numberString = number.toString(); let firstDigit = parseInt(numberString[0]);

let lastDigit = parseInt(numberString[numberString.length -

1]);

let sum = firstDigit + lastDigit; return sum;

}

let givenNumber = 1234;

let digitSum = calculateFirstAndLastDigitSum(givenNumber);

console.log(`Sum of the first and last digits of ${givenNumber}: ${digitSum}`);

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 Ans :

=>

// Define the number of rows for the pattern const numRows = 5; for (let i = 1; i <= numRows; i++) { let rowOutput = ''; for (let j = 1; j <= 5; j++) { if (j === 1) { rowOutput += `${i} `;

} else {

rowOutput += `${Math.pow(i, j)} `;

}

}

console.log(rowOutput);

}

38) Use pattern in console.log in JS?

=>

1) 1

1 0

1 0 1

1 0 1 0

1 0 1 0 1

=>

for (let i = 1; i <= 5; i++) { for (let j = 1; j <= i; j++) { if (j % 2 === 0) { console.log('0');

} else { console.log('1');

}

}

console.log('\n');

}

1. A

B C

D E F

G H I J

K L M N O =>

let currentChar = 65; // ASCII code for 'A'

for (let i = 1; i <= 5; i++) { let row = '';

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

row += String.fromCharCode(currentChar) + ' '; currentChar++;

}

console.log(row);

}

1. 1

2 3

4 5 6

7 8 9 10

11 12 13 14 15

=> let counter = 1;

for (let i = 1; i <= 5; i++) { let row = '';

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

}

console.log(row);

}

1. \*
   * \*
   * \* \*
   * \* \* \*
   * \* \* \* \*

=>

for (let i = 1; i <= 5; i++) { let row = '';

for (let j = 1; j <= i; j++) { row += '\* ';

}

console.log(row);

}

39) Accept 3 numbers from user using while loop and check each numbers palindrome?

Ans :

=>

function isPalindrome(number) { const originalNumber = number; let reversedNumber = 0;

while (number > 0) { const digit = number % 10;

reversedNumber = reversedNumber \* 10 + digit; number = Math.floor(number / 10);

}

return originalNumber === reversedNumber;

}

let count = 1; while (count <= 3) {

const userInput = parseInt(prompt(`Enter number ${count}:`));

if (!isNaN(userInput)) { if (isPalindrome(userInput)) {

console.log(`${userInput} is a palindrome.`);

} else {

console.log(`${userInput} is not a palindrome.`);

}

count++; } else {

alert('Invalid input. Please enter a valid number.');

}

}