JavaScript Basics Exam Answer

Question 1: Answer the following questions :

1. Difference between var and let keyword in JavaScript?

Ans: Var can be declared and accessed globally. Let can be declared globally, but it’s access is limited to the block in which it’s declared.

1. Difference between “== “ and “=== “operators?

Ans: The main difference between the two operators is how they compare values. The (==) operator compares the values of two variables after performing type conversion if necessary. On the other hand, the (===) operator compares the values of two variables without performing type conversion.

1. Difference between while and for loop?

Ans: For loop is used to iterate over a sequence of items. While loop is used to repeatedly execute a block of statements while a condition is true.

1. Difference between if and switch?

Ans: The difference between if-else and switch is if statement evaluates integers, characters, pointers, floating-point numbers, and booleans. A switch statement evaluates just character or numeric data types.

Question 2 : What is the output:

1. Output: object (it returns the type of function which is object)
2. Output: 12 (it concatenated 1 with the character 2)
3. Output: 10 (y stores the value of x which is 10)
4. Output:

foo

undefined

bar

undefined

1. Output: 16 (the final value stored in i is printed)

Question 3 : Write Js Code to solve these problems:

1- Check if string is palindrome or not

Hint : A palindrome is a word or sentence that’s spelled the

same way both forward and backward

For example : eye is palindrome

Welcome isn’t palindrome

Ans:

function isPalindrome(str) {

// Remove spaces and convert to lowercase for uniformity

str = str.replace(/\s+/g, '').toLowerCase();

// Check if the string is equal to its reverse

return str === str.split('').reverse().join('');

}

// Examples

console.log(isPalindrome("eye")); // true

console.log(isPalindrome("Welcome")); // false

console.log(isPalindrome("A man a plan a canal Panama")); // true

2- Create a function to check if the given number is even

Hint: return true if number is even and false if not.

Solve this problem in two ways

1- Use only if statement without else

2- Use ternary operator

Ans: -Using an if statement without else:

function isEven (number) {

if (number % 2 === 0) {

return true;

}

return false;

}

// Examples

console.log(isEven(4)); // true

console.log(isEven(7)); // false

- Using a ternary operator:

function isEven(number) {

return number % 2 === 0 ? true : false;

}

// Examples

console.log(isEvenTernary(4)); // true

console.log(isEvenTernary(7)); // false

3- Find Largest Number In Array

Ex: [10 , 100 , 50 , 4] => output : 100

Ans:

function LargestNumber(arr) {

return Math.max(...arr);

}

// Example

const numbers = [10, 100, 50, 4];

console.log(LargestNumber(numbers)); // 100

4- Write a function that takes an integer hours and converts it

to seconds.

Ans:

function hoursToSeconds(hours) {

const secondsInAnHour = 3600;

return hours \* secondsInAnHour;

}

// Examples

console.log(hoursToSeconds(1)); // 3600

console.log(hoursToSeconds(2)); // 7200

console.log(hoursToSeconds(8)); // 28800

console.log(hoursToSeconds(0.5)); // 1800

5- Write a JS code to find the power of a number using for loop

Ans:

function power(base, exponent) {

let result = 1;

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

result \*= base;

}

return result;

}

// Examples

console.log(power(2, 3)); // 8

console.log(power(5, 2)); // 25

console.log(power(3, 4)); // 81

console.log(power(10, 0)); // 1

Question 4 : Using html , css and js make this possible:

HTML:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Background Generator</title>

<link rel="stylesheet" href="styles.css">

</head>

<body>

<div class="container" id="container">

<h1>BACKGROUND GENERATOR</h1>

<input type="color" id="color1" value="#ff0000"> <!-- Color 1 -->

<input type="color" id="color2" value="#00ff00"> <!-- Color 2 -->

</div>

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

</body>

</html>

CSS:

body {

font-family: Arial, sans-serif;

display: flex;

justify-content: center;

align-items: center;

height: 100vh;

background-color: #f0f0f0;

margin: 0;

}

.container {

text-align: center;

background-color: #fff;

padding: 20px;

border-radius: 8px;

box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);

width: 100%;

max-width: 600px;

height: 350px;

margin-top: 20px;

}

h1 {

margin-top: 0;

font-size: 50px;

letter-spacing: 18px;

color: #201c1c;

}

input[type="color"] {

margin: 10px;

}

JS:

// Function to update gradient and CSS code

function updateGradient() {

const color1 = document.getElementById('color1').value;

const color2 = document.getElementById('color2').value;

const gradientDiv = document.getElementById('container');

gradientDiv.style.background = `linear-gradient(to right, ${color1}, ${color2})`;

}

// Event listeners for color input changes

document.getElementById('color1').addEventListener('input', updateGradient);

document.getElementById('color2').addEventListener('input', updateGradient);

// Initial update when the page loads

updateGradient();

Question 5:

1. Refactor this code by doing it in another two ways:
2. const keyNum = Number(prompt('enter key num : '))
3. if (keyNum === 1 ) {
4. console.log("The key is on") ;
5. } else {
6. console.log("The key is off");
7. }

* Tenary Operator:
* const keyNum = Number(prompt('enter key num : '))
* const message = keyNum === 1 ? "The key is on" : "The key is off";
* console.log(message);
* Switch Case:
* const keyNum = Number(prompt('Enter key num: '));
* switch (keyNum) {
* case 1:
* console.log("The key is on");
* break;
* default:
* console.log("The key is off");
* }

2- Refactor this code using for loop:

let i = 5;

while (i >= 1 ) {

  console.log(i);

  i--;

}

Ans:

for (i = 5 ; i >= 1 ; i-- ) {

  console.log(i);

 }

3- True or false

1. JavaScript is async , blocking and single threaded language

Ans: True

1. Const arr = [1 , 2 , 3 , 4]

The output of console.log(typeof(arr)) will be array

Ans: False (Output will be "object" because arrays are a type of object in JavaScript)