Module – 1: Node - JavaScript Fundamental

Q1 : What is the difference between Java & JavaScript?

JAVA	JAVASCRIPT
This is OOP or Object-Oriented programming language	This is an object based scripting language
A stand-alone language	Not stand-alone, incorporated into HTML program for operation
Strongly typed language is used, and data type of variable is decided before declaring or using it	Language utilized is loosely typed, so that the user does not have to worry about the data type before the declaration
Code has to be compiled	The code is all text
Slightly more complex	Easier in comparison
Large amount of memory is required	Memory consumption is lesser
Programs are saved with “.java” extension	Programs are saved with JavaScript, “.js” extension
Compiled on the server before it is executed on the client side	JavaScript is interpreted on the client side
Is static and the code once written can be run on any computing platform	Dynamic and is a cross-platform language
Stored in the / client host machine under the “Byte” code	Stored in host or client machine as “source” code

Q2. What is JavaScript?

JavaScript is a lightweight, interpreted programming language with object-oriented capabilities that allows you to build interactivity into otherwise static HTML pages.

JavaScript is a text-based programming language used both on the client-side and server-side that allows you to make web pages interactive

Q3. What are the data types supported by JavaScript?

1.String
2.Number
3.Bigint
4.Boolean
5.Undefine
6.Null
7.Symbol
8.Object

Q4. What are the scopes of a variable in JavaScript?

1. Global Scope
2.Local Scope
3.Block Scope
4.Function Scope

1. Global Scope : Any variable declared outside of a function is said to have Global Scope.

2. Local Scope : Any variable that you declare inside a function is said to have Local Scope. You can access a local variable can within a function

3. Block Scope : Declaring the variable within a block and accessing the variable within the block.

4. Function Scope: You cannot access variables defined inside a function from outside the function or from another function

Q5. What is Callback?

A callback function is a function passed into another function as an argument, which is then invoked inside the outer function to complete some kind of routine or action. The above example is a synchronous callback, as it is executed immediately.

Q6. What is Closure? Give an example.

A closure is the combination of a function bundled together (enclosed) with references to its surrounding state (the lexical environment). In other words, a closure gives you access to an outer function's scope from an inner function.

```
//ex of closure
function a() {
       var value = 100;

       function b() {
           console.log(value);
       }

       return b;
   }
   var b = a();

   b();
</script>
```

Q7. What is the difference between the operators '==' & '===‘?

Equality Operator == Equality Operator does not check the type of the operand. It tries to convert them to string, number, or Boolean.
EX.
Var a=10
Var b=10

console.log(a== b) //true

Strict Equality Operator === The strict Equality operator returns false if the types are different.
EX.
Var a=10
Var b="10"

console.log(a=== b) //false

Q8. What is the difference between null & undefined?

NULL: The null value represents the intentional absence of any object value

UNDEFINED:A variable that has not been assigned a value is of type undefined . A method or statement also returns undefined if the variable that is being evaluated does not have an assigned value. A function returns undefined if a value was not returned .

Q8.What would be the result of 2+5+”3″?

```
<script>
    const sum = (a, b, c) => {
        const result = a + b + c;
        console.log(result);
    };
    sum(2, 5, "3"); //73
</script>
```

Q10. What is the difference between Call & Apply?

The call() method takes arguments separately.
The apply() method takes arguments as an array.