Week 1 Practice Questions

1. Which of the following statements is not true regarding JavaScript? [MCQ]

A. JavaScript always requires a browser for its execution.
B. JavaScript cannot be invoked using command line.
C. JavaScript always requires a document context in order to be executed.
D. All of the above

Answer: D

Solution: Being a programming language, the programs written in JavaScript language can be executed independently, without always needing a document context using some JavaScript runtimes like NodeJS, Deno etc.

2. What will be the output on console if the following JavaScript program is executed using command line? [MCQ]

Code:

```
{
      let x = 5;
}
console.log(x)
```

A. 5
B. undefined
C. null
D. Reference Error

Answer: D

Solution: The variables declared using keyword “let” have block scope. And, they cannot be used or referred to, outside the scope or block, which they have been declared in. If done, it will generate an error.

3. What is AJAX? [MSQ]

A. AJAX is an acronym for Asynchronous JavaScript and XML.
B. It is a technology used to send request to the server without reloading the page via JavaScript.
C. Only XML data can be transported using AJAX.
D. All of the above

Answer: A and B

Solution: AJAX (Asynchronous JavaScript and XML) is a technology used to communicate with the servers behind the scenes, and use the server responses to make the user experience better. It is not mandatory to use XML data while working with AJAX, it can even be done using JSON format. In fact, JSON is preferred over XML these days.

4. Consider the following variable declaration,

let x

What will be printed on console for console.log(x)? [MCQ]

A. undefined
B. null
C. 0
D. It will throw a syntax error.

Answer: A

Solution: The variables declared using keyword “let” will result undefined, if used or referred before assigning.

5. Consider the following code snippet,

```
const x = 14
x = 15
```

What will be the output of console.log(x)? [MCQ]

A. It will throw a TypeError.
B. 15
C. 14
D. undefined

Answer: A

Solution: An assignment to a variable declared using “const” keyword afterwards will always result in a Type Error. It can only be assigned once, that too during declaration.

6. Consider the following JavaScript program,

```
const obj = {color: 'red',
  changeColor: function (color) {this.color = color}}

obj.changeColor("Yellow")
```

What will be the output of the statement: console.log(obj.color)? [MCQ]

A. SyntaxError
B. Red
C. Yellow
D. Undefined

Answer: C

Solution: The value of “this” keyword is obj (calling object) itself. So, the object’s method “changeColor” will change the color property from “red” to “Yellow”.

7. What will be the output of the following program? [MCQ]

```
const obj = {
  name: 'Rohit',
  changeName: function (name) {
    this.name = name
  },
}

obj.changeName('Mohit')
console.log(obj.name)
```

A. undefined
B. Rohit
C. Mohit

D. null

Answer: C

Solution: The value of “this" keyword is obj (calling object) itself. So, the object’s method “changeName” will change the name property from “Rohit” to “Mohit”.

8. Which of the following is a correct way to define a function in JavaScript? [MSQ]

A. const sum = (x, y) => {return x+y}
B. const sum = (x, y) => x+y
C. const sqr = x => {return x*x}
D. const sqr = x => x*x

Answer: A, B, C and D

Solution: If the function has only one argument, the parenthesis can be skipped. And if it has only return statement, { } can be skipped as well. So, all the given options refer to valid function definitions.

9. Consider the following HTML document,

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <script>
      document.getElementById('div1').style.backgroundColor = 'yellow'
    </script>
    <title>Document</title>
  </head>
  <body>
    <div
      id="div1"
      style="height: 100px; width: 100px; background-color: red"
    ></div>
  </body>
</html>
```

What will be the background color of an element having ID ‘div1’ in the rendered webpage corresponding to this document? [MCQ]

A. red
B. black
C. yellow
D. None of the above

Answer: A

Solution: Till the time script will run, DOM is not yet created so element with ID ‘div1’ will not be found. So, script will not have any effect on the style.

10. Consider the following HTML document,

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <title>Document</title>
  </head>
  <body>
    <div
      id="div"
      style="background-color: black; height: 50px; width: 50px"
    ></div>
    <script>
      const colors = {
        color1: 'red',
        color2: 'green',
        color3: 'yellow',
      }
      let i = 1
      let color = null
      setInterval(() => {
        if (i % 2 != 0) {
          color = colors.color1
        }
        else if (i % 4 == 0) {
          color = colors.color2
```

```
        }
        else {
          color = colors.color3
        }
        document.getElementById('div').style.backgroundColor = color
        i++
      }, 1000)
    </script>
  </body>
</html>
```

What will be the background color of an element having ID ‘div’ after 6 seconds? [MCQ]

A. black
B. yellow
C. green
D. red

Answer: B

Solution: The setInterval() function will execute the callback function after every 1 second. So ‘i’ will be incremented after every 1 second, and after 6 seconds the value of ‘i’ will be 6, and it is divisible by 2 but not by 4. So, code inside else block will be executed, which will make the color ‘yellow’.