Assignment 1:

Question 1. Explore and explain the various methods in console function Explain them.

Ans:

- In javascript, the console is an object which provides access to the browser debugging console. We can open a console in web browser by using: *Ctrl + Shift + K* for windows and *Command + Option + K* for Mac. The console object provides us with several different methods, like :
    - log()
    - error()
    - warn()
    - clear()
    - time() and timeEnd()
    - table()
    - count()
    - group() and groupEnd()
    - custom console logs

1. **console.log():**

- Mainly used to log(print) the output to the console. We can put any type inside the log(), be it a string, array, object, boolean etc.

2. **console.error():**

- Used to log error message to the console. Useful in testing of code. By default the error message will be highlighted with red color.

3. **console.warn()**

- Used to log warning message to the console. By default the warning message will be highlighted with yellow color.

4. **console.clear()**

- Used to clear the console. The console will be cleared, in case of Chrome a simple overlayed text will be printed like : ‘Console was cleared’ while in firefox no message is returned.

5. **console.time() and console.timeEnd()**

- Whenever we want to know the amount of time spend by a block or a function, we can make use of the time() and timeEnd() methods provided by the javascript console

object. They take a label which must be same, and the code inside can be anything(function, object, simple console).

6. **console.table()**

- This method allows us to generate a table inside a console. The input must be an array or an object which will be shown as a table.

7. **console.count()**

- This method is used to count the number that the function hit by this counting method.

8. **console.group() and console.groupEnd()**

- group() and groupEnd() methods of the console object allows us to group contents in a separate block, which will be indented. Just like the time() and the timeEnd() they also accepts label, again of same value.

9. **Custom Console Logs**

- User can add Styling to the console logs in order to make logs Custom . The Syntax for it is to add the css styling as a parameter to the logs which will replace %c in the logs as shown in the example below .

Question 2. Write the difference between var, let and const with code examples.

Ans:

- **Var:**
- The JavaScript variables statement is used to declare a variable and, optionally, we can initialize the value of that variable.

- **Let:**
- The **let** statement declares a local variable in a block scope. It is similar to **var**, in that we can optionally initialize the variable.

- **Const:**
- const statement values can be assigned once and they cannot be reassigned. The scope of const statement works similar to let statements.

Question 3. Write a brief intro on available data types in JavaScript.

Ans:

- JavaScript provides different **data types** to hold different types of values. There are two types of data types in JavaScript.

    1. Primitive data type
    2. Non-primitive (reference) data type

- JavaScript is a **dynamic type language**, means you don't need to specify type of the variable because it is dynamically used by JavaScript engine. You need to use **var** here to specify the data type. It can hold any type of values such as numbers, strings etc.

- JavaScript primitive data types
- There are five types of primitive data types in JavaScript. They are as follows:

1. **String:** represents sequence of characters e.g. "hello"
2. **Number:** represents numeric values e.g. 100
3. **Boolean:** represents boolean value either false or true
4. **Undefined:** represents undefined value
5. **Null:** represents null i.e. no value at all

- JavaScript non-primitive data types
- The non-primitive data types are as follows:

1. **Object:** represents instance through which we can access members
2. **Array:** represents group of similar values
3. **RegExp:** represents regular expression