IP-PRACTICAL NO 6

Date of Performance: 27-09-2022

Software Requirement: Visual Studio Code, Notepad, Notepad++, NodeJS

Aim: To use JavaScript to implement array, string, math and date methods in ES6.

Objectives: The aim of this experiment is that the students will be able to:

- Implement JavaScript codes based on arrays, strings, math and date methods.
- Understand different features of JavaScript methods & its usage in code.

Outcomes: After study of this experiment, the students will be able to:

- Execute array, string, math and date methods in JavaScript.
- Implement the usage of different methods in JavaScript.

Prerequisite: Basic knowledge of JavaScript syntax required.

Theory:

The **Array** object, as with arrays in other programming languages, enables storing a collection of multiple items under a single variable name, and has members for performing common array operations.
In JavaScript, arrays aren't primitives but are instead Array objects with the following core characteristics:

- **JavaScript arrays are resizable** and **can contain a mix of different data types**.
- **JavaScript arrays are not associative arrays** array elements must be accessed using nonnegative integers as indexes.
- **JavaScript arrays are zero-indexed**.

The **String** object is used to represent and manipulate a sequence of characters. Strings are useful for holding data that can be represented in text form. Some of the most-used operations on strings are to check their length, to build and concatenate

them using the + and += string operators, checking for the existence or location of substrings with the indexOf() method, or extracting substrings with the substring() method.

Math is a built-in object that has properties and methods for mathematical constants and functions. It's not a function object. Math works with the Number type. It doesn't work with BigInt. Unlike many other global objects, Math is not a constructor. All properties and methods of Math are static.

JavaScript **Date** objects represent a single moment in time in a platform-independent format. Date objects contain a Number that represents milliseconds since 1 January 1970 UTC. A JavaScript date is fundamentally specified as the number of milliseconds that have elapsed since the ECMAScript epoch.

Problem Statement:

Implement array, string, math and date methods in ES6.

Source Code and Output:

→ ARRAY IMPLEMENTATION:

```
console.log('~ARRAY METHODS IMPLEMENTATION~\n')
//1st: displaying all the items of the array
const cities = ['Mumbai', 'Paris', 'Mecca','London', 'New York', 'Seoul'];
for(let i=0; i<cities.length; i++) {
 console.log(`Element at index ${i} is ${cities[i]}.`);
}
//2nd: adding an element to array using push() method:
cities.push('Amsterdam');
console.log('Array after addition is: ')
console.log(cities);
//3rd: removing an element from array using shift() method:
cities.pop();
console.log('Array after removal of an element is: ')
console.log(cities);
//Array of numbers
var arr = [2, 5, 8, 1, 4]
arr.sort();
console.log('\nArray after using sort:');
console.log(arr);
```

```
PS C:\Users\husna\OneDrive\Desktop\Bootstrap> node array.js
~ARRAY METHODS IMPLEMENTATION~

Element at index 0 is Mumbai.
Element at index 1 is Paris.
Element at index 2 is Mecca.
Element at index 3 is London.
Element at index 4 is New York.
Element at index 5 is Seoul.
Array after addition is:
[
  'Mumbai',
  'Paris',
  'Mecca',
  'London',
  'New York',
  'Seoul',
  'Amsterdam'
]
Array after removal of an element is:
[ 'Mumbai', 'Paris', 'Mecca', 'London', 'New York', 'Seoul' ]

Array after using sort:
[ 1, 2, 4, 5, 8 ]
```

→ STRING IMPLEMENTATION:

```
//printing position
var str="HUSNA";
console.log('Character at 2nd position is: ')
console.log(str.charAt(2));
//printing the index
console.log('Index of H is: ')
console.log(str.indexOf('H'));
//printing the last index of
console.log('Last index of A is: ')
console.log(str.lastIndexOf('A'));
//string slicing
var mystr = 'my name is husna';
console.log('The result of String Slicing is: ')
console.log(mystr.slice(1,6));
console.log("\nFound 'is' at index "+mystr.search('is'));
```

```
PS C:\Users\husna\OneDrive\Desktop\Bootstrap> node string.js
Character at 2nd position is:
S
Index of H is:
0
Last index of A is:
4
The result of String Slicing is:
y nam

Found 'is' at index 8
```

→ MATH METHOD IMPLEMENTATION:

```
//1. printing absolute values
console.log('Absolute values are as follows: ');
console.log(Math.abs(-22));
console.log(Math.abs(-9.5));
console.log(Math.abs('-12'));

//2. printing ceil values
console.log('\nCeil values are as follows: ');
console.log(Math.ceil(7.2));
console.log(Math.ceil(0.2));
console.log(Math.ceil(89.4));

//3.printing log
console.log('\nLog values are as follows: ');
console.log(Math.log(4));
console.log(Math.log(9));
console.log(Math.log(33));

//4. printing round values
console.log('\nRound values are as follows: ');
console.log(Math.round(2.7));
console.log(Math.round(4.9));
console.log(Math.round(5.3));

//5. printing cuberoot
console.log('\nCube-root values are as follows: ')
console.log(Math.cbrt(64));
console.log(Math.cbrt(-16));
console.log(Math.cbrt(7));
```

```
PS C:\Users\husna\OneDrive\Desktop\Bootstrap> node math.js
Absolute values are as follows:
22
9.5
12

Ceil values are as follows:
8
1
90

Log values are as follows:
1.3862943611198906
2.1972245773362196
3.4965075614664802

Round values are as follows:
3
5
5

Cube-root values are as follows:
4
-2.5198420997897464
1.9129311827723892
```

→ DATE METHOD IMPLEMENTATION:

```
//1. printing date
var date=new Date();
console.log("Today's date: "+date.getDate());
//2. printing year
var year=new Date();
console.log("Present Year : "+year.getUTCFullYear());
//3. printing minutes
console.log('Minutes value: ')
var min=new Date();
console.log(min.getMinutes());
//4. printing hours
console.log('Hours value: ')
var hour=new Date();
console.log(hour.getHours());
//5. printing milliseconds
console.log('Milliseconds value: ')
var milli =new Date();
console.log(milli.getMilliseconds());
```

```
PS C:\Users\husna\OneDrive\Desktop\Bootstrap> node date.js
Today's date: 24
Present Year : 2022
Minutes value:
51
Hours value:
17
Milliseconds value:
685
```

Conclusion:

From this experiment we understand that JavaScript is a very powerful web language and is used at a large scale. We have learned how to implement various methods in JavaScript such as array, string, math and date methods into our codes.