Practical – 3

AIM 1: Print prime Numbers up to given value in the input box

Source Code:

HTML CODE:

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
    <center>
    <h1>ENTER PRIME NUMBER</h1>
    <input type="text" id="num">
    <button onclick="printPrimeNumbers()">submit</button>
    <ul id="result"></ul>
    <script src="./JS/prime.js"></script>
    </center>
</body>
</html>
```

JAVASCRIPT CODE:

```javascript
console.log("connected..");
function printPrimeNumbers() {
    var num = document.getElementById("num").value;
    var primes = "";
    for (var i = 2; i <= num; i++) {
        var isPrime = true;
        for (var j = 2; j <= i/2; j++) {
            if (i % j == 0) {
                isPrime = false;
                break;
            }
        }
        if (isPrime) {
            primes += i + " ";
        }
    }
    document.getElementById("result").innerHTML = primes;
}
```

Output:

ENTER PRIME NUMBER

20   submit

2 3 5 7 11 13 17 19

AIM 2: Write Script to reverse the given input string

Source Code:

HTML CODE:

```html
<!DOCTYPE html>
<html>
  <head>
    <title>Reverse Input</title>
  </head>
  <body>
    <center>
      <h1>ENTER THE NUMBER</h1>
      <input type="text" id="inputBox"></input>
      <button onclick="reverseInput()">Reverse</button>
      <p id="output"></p>
      <script src="./JS/reverse.js">
      </script>
    </center>
  </body>
</html>
```
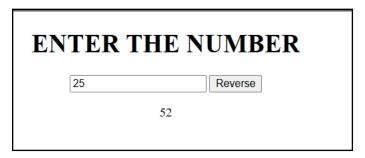
JAVASCRIPT CODE:

```javascript
function reverseInput() {
    var inputBox = document.getElementById("inputBox");
    var input = inputBox.value;
    var reversedInput = input.split("").reverse().join("");
    document.getElementById("output").innerHTML = reversedInput;
}
```

Output:

ENTER THE NUMBER

```
25                    Reverse
          52
```

AIM 3: Create Dynamic Multiplication Table using inputs

Source Code:

HTML CODE:

```html
<!DOCTYPE html>
<html>
<head>
  <title>Dynamic Multiplication Table</title>
</head>
<body>
  <center>
    <h1>DYNAMIC MULTIPLICATION TABLE</h1>
    <form>
     ENTER THE NUMBER: <input type="number" id="num" min="1" max="10">
    <button type="button" onclick="createTable()">Create Table</button>
    </form>
     <br>
    <table id="table"></table>
    <script src="./JS/mul.js"></script>
  </center>
</body>
```

</html>

JAVASCRIPT CODE:

```
function createTable() {

    var num = document.getElementById("num").value;

    var table = document.getElementById("table");

    table.innerHTML = "";

     for (var i = 1; i <= 10; i++) {

      var row = table.insertRow(-1);

      var cell1 = row.insertCell(0);

      var cell2 = row.insertCell(1);

      var cell3 = row.insertCell(2);

     cell1.innerHTML = num + "x" + i;

     cell2.innerHTML = "=";

     cell3.innerHTML = num * i;

     }

  }
```

Output:

AIM 4: Find the Age from input date.(Ex. 17 Yrs, 3 Monts,13 Days)

Source Code:

HTML CODE:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
    <center>
    <label><h1>ENTER THE BIRTHDATE:</h1></label>
    <br>
    <input type="text" id="birthdate" placeholder="yyyy-mm-dd">
    <button onclick="calculateAge()">Calculate age</button>
    <p id="output"></p>
    <script src="./JS/age.js">
    </script>
    </center>
</body>
</html>
```

JAVASCRIPT CODE:

```javascript
function calculateAge() {

    var birthdate = new Date(document.getElementById("birthdate").value);

    var today = new Date();


    var years = today.getFullYear() - birthdate.getFullYear();

    var months = today.getMonth() - birthdate.getMonth();

    var days = today.getDate() - birthdate.getDate();

    if (days < 0) {

        days += 30;

        months--;

    }

    if (months < 0) {

        months += 12;

        years--;

    }

    if (months === 0 && days === 0 && today.getDate() < birthdate.getDate()) {

        years--;

    }

    document.getElementById("output").innerHTML = "Your age is: " + years + " Yrs, " +
months + " Months, " + days + " Days.";

}
```

Output:

ENTER THE BIRTHDATE:

2003-04-01   Calculate age

Your age is: 19 Yrs, 9 Months, 13 Days.

AIM 5: Find the No. of Days between two given dates

Source Code:

HTML CODE:

```html
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
</head>
<body>
  <center>
  <label><h1>ENTER START DATE:</h1></label>
  <br>
  <input type="text" id="startDate" placeholder="yyyy-mm-dd">
  <br>
  <label><h1>ENTER END DATE:</h1></label>
  <br>
```
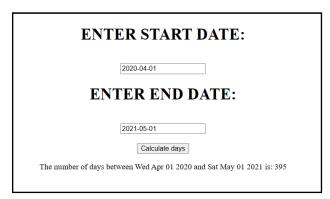
```html
<input type="text" id="endDate" placeholder="yyyy-mm-dd">

<br><br>

<button onclick="calculateDays()">Calculate days</button>

<p id="output"></p>

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

</script>

</center>
</body>
</html>
```

JAVASCRIPT CODE:

```javascript
function calculateDays() {

    var startDate = new Date(document.getElementById("startDate").value);

    var endDate = new Date(document.getElementById("endDate").value);

    var timeDiff = Math.abs(endDate.getTime() - startDate.getTime());

    var diffDays = Math.ceil(timeDiff / (1000 * 3600 * 24));

    document.getElementById("output").innerHTML = "The number of days between " +
startDate.toDateString() + " and " + endDate.toDateString() + " is: " + diffDays;

}
```

Output:

Conclusion:

In this practical we learnt about Print prime Numbers up to given value in the input box, To write Script to reverse the given input string, Create Dynamic Multiplication Table using inputs, Find the Age from input date, Find the No. of Days between two given dates.

Course Outcome:

Hoisting: It is a concept that enables us to extract values of variables and functions even before initializing/assigning value without getting errors and this is happening due to the 1st phase (memory creation phase) of the Execution Context.

Scoping: The scope is a region of the program where a variable can be accessed. In other words, scope determines the accessibility/visibility of a variable.

A JavaScript function is a block of code designed to perform a particular task. A JavaScript function is executed when "something" invokes it (calls it).

There are different types of JavaScript operators:

- Arithmetic Operators
- Assignment Operators
- Comparison Operators
- Logical Operators
- Conditional Operators
- Type Operators

JavaScript supports different kinds of loops:

- for - loops through a block of code a number of times
- for/in - loops through the properties of an object
- for/of - loops through the values of an iterable object
- while - loops through a block of code while a specified condition is true
- do/while - also loops through a block of code while a specified condition is true

The JavaScript date object can be used to get year, month and day. You can display a timer on the webpage by the help of JavaScript date object. You can use different Date constructors to create a date object. It provides methods to get and set day, month, year, hour, minute and seconds.