

# Assignment

*Web Technologies*

**By: Devdeepsinh Zala**  
**20BCM015**  
**Trimester – 7**  
**7BCEI**

1. Design a web page which have simple guessing number game. Write a JavaScript function which generate should generate an integer randomly and ask the user to guess the integer. Based on the number guessed, it should display more appropriate message and change the background colour of web (if Greater then background color Green , Smaller then Red or Correct then Yellow).

```
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport">
  <title>Guessing Game</title>
</head>

<body>
  <h1>Mystery Number Guessing</h1>
  <div id="form">
    <label for="guess">Enter your guess (1 - 10)</label>
    <input type="number" name="guess" id="guess" required>
    <input type="submit" value="Check" id="guessbtn">
  </div>
  <script>
    let num = Math.floor(Math.random() * 10);
    console.log("Myster number is: " + num);
    let tries = 0
    let limit = 5
    document.getElementById("guessbtn").addEventListener("click",
      function () {
        let guess = document.getElementById("guess").value;
        if (guess == num) {
          document.body.style.backgroundColor = 'yellow';
          alert('You guessed it correct!');
        }
        else if (guess < num) {
          document.body.style.backgroundColor = 'red';
          alert('Too low');
        }
        else {
          document.body.style.backgroundColor = 'green';
          alert('Too high');
        }
        tries++;
        if (tries == limit) {
          alert('You lose :(\nThe number was ' + num);
        }
      })
  </script>
</body>

</html>
```

Output:

# Mystery Number Guessing

Enter your guess (1 - 10)

# Mystery Number Guessing

Enter your guess (1 - 10)

# Mystery Number Guessing

Enter your guess (1 - 10)

## 1. What is DOM? Explain in detail.

The Document Object Model (DOM) is a programming interface for web documents. It represents the page so that programs can change the document structure, style, and content. The DOM represents the document as nodes and objects; that way, programming languages can interact with the page. A web page is a document that can be either displayed in the browser window or as the HTML source. In both cases, it is the same document but the Document Object Model (DOM) representation allows it to be manipulated. As an object-oriented representation of the web page, it can be modified with a scripting language such as JavaScript. The Document Object Model (DOM) is a programming interface for web documents. It represents the page so that programs can change the document structure, style, and content. The DOM represents the document as nodes and objects; that way, programming languages can interact with the page. A web page is a document that can be either displayed in the browser window or as the HTML source. In both cases, it is the same document but the Document Object Model (DOM) representation allows it to be manipulated. As an object-oriented representation of the web page, it can be modified with a scripting language such as JavaScript. For example, the DOM specifies that the `querySelectorAll` method in this code snippet must return a list of all the `<p>` elements in the document:

```
const paragraphs = document.querySelectorAll("p");
alert(paragraphs[0].nodeName);
```

\

2. Design a web page which have simple student grading system. Write a JavaScript to find the grade of a student based on the following policy. The script should change the background colour of web as per the grading.

**Class test: 12% weightage, Tutorial-12%, SE:16%, LPW:20%, SEE:40%.**

**Grade is decided based on the below range of total marks.**

Grade	Range of total marks	Background colour of web
A+	91-100	Green
A	81-90	Parrot
B+	71-80	Yellow
B	61-70	Pink
C+	51-60	Orange
C	>40	Red
Fail	<40	Black

```
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport">
  <title>Grading System</title>
  <style>
    table,
    th,
    td,
    tr {
      border: 1px solid black;
      border-collapse: collapse;
    }

    th {
      padding: 10px;
    }

    td {
      padding: 10px;
    }

    h1 {
      text-align: center;
      margin-bottom: 3%;
    }

    .mid {
      text-align: center;
    }

    #btn {
      padding-left: 45%;
      padding-right: 40%;
      margin-top: 15px;
    }
  </style>
</head>

<body>
  <div class="mid">
    <h1>Grading System</h1>
    <div class="table">
      <table>
        <tr>
          <th>Grade</th>
          <th>Range of total marks</th>
          <th>Background colour of web</th>
        </tr>
        <tr>
          <td>A+</td>
          <td>91-100</td>
          <td>Green</td>
        </tr>
        <tr>
          <td>A</td>
          <td>81-90</td>
          <td>Parrot</td>
        </tr>
        <tr>
          <td>B+</td>
          <td>71-80</td>
          <td>Yellow</td>
        </tr>
        <tr>
          <td>B</td>
          <td>61-70</td>
          <td>Pink</td>
        </tr>
        <tr>
          <td>C+</td>
          <td>51-60</td>
          <td>Orange</td>
        </tr>
        <tr>
          <td>C</td>
          <td>>40</td>
          <td>Red</td>
        </tr>
        <tr>
          <td>Fail</td>
          <td><40</td>
          <td>Black</td>
        </tr>
      </table>
    </div>
    <div class="btn">
      <button>Submit</button>
    </div>
  </div>
</body>
</html>
```

```
}

div {
    float: left;
}

.second {
    margin-left: 10%;
    justify-content: center;
}
```

```
label {
    margin-left: 10px;
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<h1>Student Grading System</h1>
```

```
<div id="first">
```

```
<table>
```

```
<th>
```

```
Criterion
```

```
</th>
```

```
<th>
```

```
Marks (/100)
```

```
</th>
```

```
<tr>
```

```
<td>Class test (12%)</td>
```

```
<td><input type="number" step="any" id="class_test" class="mid" min="0" max="100"></td>
```

```
</tr>
```

```
<tr>
```

```
<td>Tutorial (12%)</td>
```

```
<td><input type="number" step="any" id="tutorial" class="mid" min="0" max="100"></td>
```

```
</tr>
```

```
<tr>
```

```
<td>SE (16%)</td>
```

```
<td><input type="number" step="any" id="se" class="mid" min="0" max="100"></td>
```

```
</tr>
```

```
<tr>
```

```
<td>LPW (20%)</td>
```

```
<td><input type="number" step="any" id="lpw" class="mid" min="0" max="100"></td>
```

```
</tr>
```

```
<tr>
```

```
<td>SEE (40%)</td>
```

```
<td><input type="number" step="any" id="see" class="mid" min="0" max="100"></td>
```

```
</tr>
```

```
</table>
```

```
<input type="submit" value="Submit" id="btn">
```

```
</div>
```

```
<div class="second">
```

```
<label>Marks</label>
```

```
<input type="text" id="total_marks">
```

```
<label>Grade</label>
```

```
<input type="text" id="grade">
```

```
<label>BG-Color</label>
```

```
<input type="text" id="color">
```

```
</div>
<script>
    document.getElementById('btn').addEventListener('click', function () {
        console.log('Button clicked')
        let class_test = (12 / 100) *
            (document.getElementById('class_test').value);
        let tutorial = (12 / 100) *
            (document.getElementById('tutorial').value);
        let se = (16 / 100) * (document.getElementById('se').value);
        let lpw = (20 / 100) * (document.getElementById('lpw').value);
        let see = (40 / 100) * (document.getElementById('see').value);
        let total_marks = class_test + tutorial + se + lpw + see;
        document.getElementById('total_marks').value = total_marks;
        if (total_marks > 90) {
            document.body.style.backgroundColor = 'green';
            document.getElementById('color').value = 'Green';
            document.getElementById('grade').value = 'A+';
        }
        else if (total_marks > 80) {
            document.body.style.backgroundColor = `rgb(97, 179, 59)`;
            document.getElementById('color').value = 'Parrot green';
            document.getElementById('grade').value = 'A';
        }
        else if (total_marks > 70) {
            document.body.style.backgroundColor = 'yellow';
            document.getElementById('color').value = 'Yellow';
            document.getElementById('grade').value = 'B+';
        }
        else if (total_marks > 60) {
            document.body.style.backgroundColor = 'pink';
            document.getElementById('color').value = 'Pink';
            document.getElementById('grade').value = 'B';
        }
        else if (total_marks > 50) {
            document.body.style.backgroundColor = 'orange';
            document.getElementById('color').value = 'Orange';
            document.getElementById('grade').value = 'C+';
        }
        else if (total_marks > 40) {
            document.body.style.backgroundColor = 'red';
            document.getElementById('color').value = 'Red';
            document.getElementById('grade').value = 'C';
        }
        else {
            document.body.style.backgroundColor = 'grey';
            document.getElementById('color').value = 'Black';
            document.getElementById('grade').value = 'Fail';
        }
    })
</script>
</body>

</html>
```

Output:

## Student Grading System

Criterion	Marks (/100)
Class test (12%)	89
Tutorial (12%)	95
SE (16%)	97
LPW (20%)	94
SEE (40%)	97

Submit

Marks 95.19999999999999 Grade A+ BG-Color Green

## Student Grading System

Criterion	Marks (/100)
Class test (12%)	90
Tutorial (12%)	90
SE (16%)	90
LPW (20%)	90
SEE (40%)	90

Submit

Marks 90 Grade A BG-Color Parrot green

## Student Grading System

Criterion	Marks (/100)
Class test (12%)	75
Tutorial (12%)	75
SE (16%)	75
LPW (20%)	75
SEE (40%)	75

Submit

Marks 75 Grade B+ BG-Color Yellow

## Student Grading System

Criterion	Marks (/100)
Class test (12%)	<input type="text" value="65"/>
Tutorial (12%)	<input type="text" value="65"/>
SE (16%)	<input type="text" value="65"/>
LPW (20%)	<input type="text" value="65"/>
SEE (40%)	<input type="text" value="65"/>

Marks  Grade  BG-Color

## Student Grading System

Criterion	Marks (/100)
Class test (12%)	<input type="text" value="55"/>
Tutorial (12%)	<input type="text" value="55"/>
SE (16%)	<input type="text" value="55"/>
LPW (20%)	<input type="text" value="55"/>
SEE (40%)	<input type="text" value="55"/>

Marks  Grade  BG-Color

## Student Grading System

Criterion	Marks (/100)
Class test (12%)	<input type="text" value="45"/>
Tutorial (12%)	<input type="text" value="45"/>
SE (16%)	<input type="text" value="45"/>
LPW (20%)	<input type="text" value="45"/>
SEE (40%)	<input type="text" value="45"/>

Marks  Grade  BG-Color

## Student Grading System

Criterion	Marks (/100)
Class test (12%)	<input type="text" value="35"/>
Tutorial (12%)	<input type="text" value="35"/>
SE (16%)	<input type="text" value="35"/>
LPW (20%)	<input type="text" value="35"/>
SEE (40%)	<input type="text" value="35"/>

Marks  Grade  BG-Color



### 3. Explain the following terms, localStorage JSON

#### LocalStorage

It is a data storage type of web storage. This allows the JavaScript sites and apps to store and access the data without any expiration date. This means that the data will always be persisted and will not expire. So, data stored in the browser will be available even after closing the browser window.

Some essential points of localStorage need to be noted:

- o localStorage is not secure to store sensitive data and can be accessed using any code. So, it is quite insecure.
- o It is an advantage of localStorage over cookies that it can store more data than cookies. You can store 5MB data on the browser using localStorage.
- o localStorage stores the information only on browser instead in database. Thereby the localStorage is not a substitute for a server-based database.
- o localStorage is synchronous, which means that each operation executes one after another.

#### JSON

The JavaScript JSON is an acronym of JavaScript Object Notation. It provides a format for storing and transporting data. It is a lightweight human readable collection of data that can be accessed in a logical manner.

- o It generates and stores the data from user input.
- o It can transport the data from the server to client, client to server, and server to server.
- o It can also build and verifying the data.