Reg.No	٠.	221	20	152
VG5'IM	ι.	LL	LΖU	132

Exno: 4a)	
Date:	Dynamic form creation using javascript
28.01.2025	

Aim:

To develop a cgpa calculator using html, css and javascript to calculate the cgpa with credit and grade with subject name.

Procedure:

- 1. User enters subject name, selects grade, and inputs credit.
- 2. Click "Add / Edit Subject" \rightarrow Adds data to the table.
- 3. User can edit or delete subjects using respective buttons.
- 4. Click "Calculate CGPA" → Displays CGPA based on entered subjects.
- 5. Click "Reset" \rightarrow Clears all data and resets CGPA.

Code:

Index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content= "width=device-width,</pre>
                                     initial-scale=1.0">
    <title>CGPA Calculator</title>
    <link rel="stylesheet" href="/style.css">
</head>
<style>
     body {
    font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
    background-color: #f0f0f0;
    margin: 20px;
    display: flex;
    align-items: center;
    justify-content: center;
    height: 100vh;
}
#calculator {
    background-color: #ffffff;
    padding: 25px;
    width: 500px;
    border-radius: 10px;
    box-shadow: 0 0 20px rgba(0, 0, 0, 0.1);
    text-align: center;
}
button {
    margin-top: 15px;
    padding: 12px 20px;
```

```
background-color: #4caf50;
    color: #fff;
    border: none;
    border-radius: 5px;
    cursor: pointer;
    font-size: 14px;
}
li {
   margin-bottom: 10px;
    list-style-type: none;
}
input,
select {
    width: calc(100% - 15px);
    padding: 10px;
    margin-bottom: 15px;
    border: 1px solid #ccc;
    border-radius: 5px;
    box-sizing: border-box;
    font-size: 14px;
}
h2,
h3 {
    margin-bottom: 20px;
    color: #333;
}
#cgpa {
    font-size: 24px;
    font-weight: bold;
    color: #4caf50;
}
#subjectList {
   text-align: left;
button.edit,
button.delete {
    padding: 8px 15px;
    margin-left: 5px;
    font-size: 12px;
    cursor: pointer;
}
button.delete {
    background-color: #d9534f;
}
.error {
    color: #d9534f;
    font-size: 16px;
```

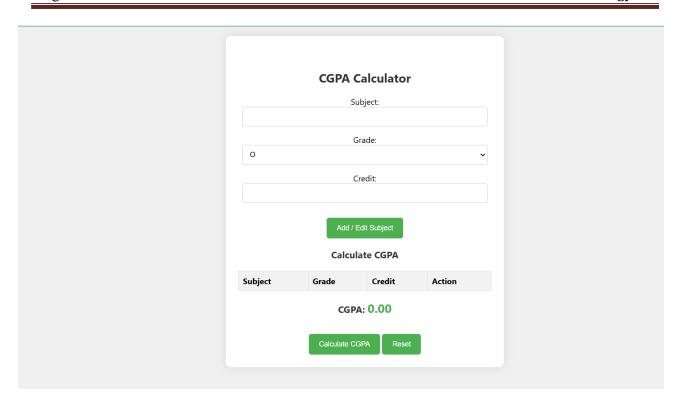
```
}
table {
   width: 100%;
   border-collapse: collapse;
   margin-top: 20px;
}
th,
td {
   border: 1px solid #ddd;
   padding: 8px;
   text-align: left;
}
th {
   background-color: #f2f2f2;
</style>
<body>
    <div id="calculator">
       <span id="creditError" class="error"></span>
        <span id="inputError" class="error"></span>
       <h2>CGPA Calculator</h2>
       <form id="cgpaForm">
           <label for="subject">Subject:</label>
           <input type="text" id="subject" required>
           <label for="grade">Grade:</label>
           <select id="grade">
               <option value="0">0</option>
               <option value="A+">A+</option>
               <option value="A">A</option>
               <option value="B">B+</option>
               <option value="B+">B</option>
               <option value="C">C</option>
           </select>
           <label for="credit">Credit:</label>
           <input type="number" id="credit" min="1" required>
           <button type="button" onclick="addSubject()">
                 Add / Edit Subject
             </button>
       </form>
       <h3>Calculate CGPA</h3>
        <thead>
               Subject
```

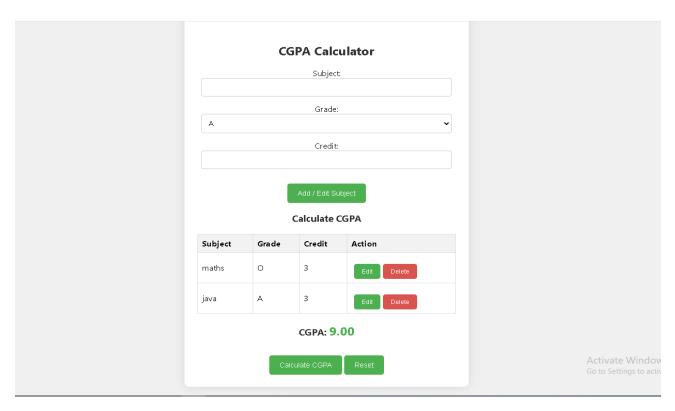
```
Grade
                   Credit
                   Action
               </t.r>
           </thead>
           <h3>CGPA: <span id="cgpa">0.00</span></h3>
       <button type="button" onclick="calculateCGPA()">
             Calculate CGPA
         </button>
       <button type="button" onclick="resetForm()">
         </button>
   </div>
   <script>
        let subjects = [];
let editIndex = -1;
function addSubject() {
   const subjectInput =
       document.getElementById('subject');
   const grade =
       document.getElementById('grade').value;
   const creditInput =
       document.getElementById('credit');
   const credit =
       parseInt(creditInput.value);
   // Validate credit input
   const inputError =
       document.getElementById('inputError')
   const creditError =
       document.getElementById('creditError');
   if (!subjectInput.value || isNaN(credit)) {
       inputError.textContent =
           'Please fill out all fields.';
       return;
   else if (credit < 1 || credit > 20) {
       creditError.textContent =
           'Credit must be between 1 and 20';
       return;
   } else {
       creditError.textContent = '';
   if (editIndex !== -1) {
       subjects[editIndex] =
       { subject: subjectInput.value, grade, credit };
       editIndex = -1;
    } else {
       subjects.push (
           { subject: subjectInput.value, grade, credit });
```

```
}
    displaySubjects();
    clearForm();
function displaySubjects() {
    const subjectList =
        document.getElementById('subjectList');
    subjectList.innerHTML = '';
    subjects.forEach((s, index) => {
        const row = document.createElement('tr');
        const subjectCell = document.createElement('td');
        subjectCell.textContent = s.subject;
        const gradeCell = document.createElement('td');
        gradeCell.textContent = s.grade;
        const creditCell = document.createElement('td');
        creditCell.textContent = s.credit;
        const actionCell = document.createElement('td');
        const editButton = document.createElement('button');
        editButton.className = 'edit';
        editButton.textContent = 'Edit';
        editButton.onclick = () => editSubject(index);
        const deleteButton = document.createElement('button');
        deleteButton.className = 'delete';
        deleteButton.textContent = 'Delete';
        deleteButton.onclick = () => deleteSubject(index);
        actionCell.appendChild(editButton);
        actionCell.appendChild(deleteButton);
        row.appendChild(subjectCell);
        row.appendChild(gradeCell);
        row.appendChild(creditCell);
        row.appendChild(actionCell);
        subjectList.appendChild(row);
    });
}
function editSubject(index) {
    const subjectInput =
        document.getElementById('subject');
    const selectedSubject = subjects[index];
    subjectInput.value = selectedSubject.subject;
    document.getElementById('grade').value =
        selectedSubject.grade;
    document.getElementById('credit').value =
```

```
selectedSubject.credit;
    editIndex = index;
}
function deleteSubject(index) {
    subjects.splice(index, 1);
    displaySubjects();
}
function calculateCGPA() {
    const totalCredits = subjects.reduce(
    (sum, s) => sum + s.credit, 0);
    const weightedSum = subjects.reduce(
    (sum, s) => sum + getGradePoint(s.grade) * s.credit, 0);
    const cgpa = totalCredits === 0 ? 0 :
    (weightedSum / totalCredits).toFixed(2);
    document.getElementById('cgpa').textContent = cgpa;
}
function getGradePoint(grade) {
    // Assign grade points as per your grading system
    switch (grade) {
        case '0': return 10.0;
        case 'A+': return 9.0;
        case 'A': return 8.0;
        case 'B': return 7.0;
        case 'B+': return 6.0;
        case 'C': return 5.0;
        default: return 0.0;
    }
}
function clearForm() {
    document.getElementById('subject').value = '';
    document.getElementById('grade').value = 'A';
    document.getElementById('credit').value = '';
}
function resetForm() {
   subjects = [];
    editIndex = -1;
    document.getElementById('subjectList').innerHTML = '';
    document.getElementById('cgpa').textContent = '0.00';
    clearForm();
}
    </script>
</body>
</html>
```

Output:





Page 7

Problem	Implementation	Time	Viva	Total
Understanding	& Output	Management	(5)	(20)
(5)	(5)	(5)		

Result:

Thus the program for CGPA calculator using javascript was exexuted successfully and output was verified..