

```
<!DOCTYPE html>
<html lang="ja">
<head>
  <meta charset="UTF-8">
  <title>成績グラフビジュアライザー</title>
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <script src="https://cdn.jsdelivr.net/npm/chart.js"></script>
  <style>
    body {
      font-family: sans-serif;
      max-width: 600px;
      margin: auto;
      padding: 20px;
    }
    h1 {
      text-align: center;
    }
    input, select, button {
      margin: 5px 0;
      width: 100%;
      padding: 8px;
    }
    table {
      width: 100%;
      margin-top: 20px;
      border-collapse: collapse;
    }
    th, td {
      padding: 8px;
      border: 1px solid #ccc;
      text-align: center;
    }
    canvas {
      margin-top: 30px;
    }
  </style>
</head>
<body>
  <h1>成績グラフビジュアライザー</h1>

  <input type="text" id="subject" placeholder="授業名を入力">
```

```
<select id="grade">
  <option value="A">A</option>
  <option value="B">B</option>
  <option value="C">C</option>
  <option value="D">D</option>
  <option value="F">F (不可)</option>
</select>
<button onclick="addEntry()">追加</button>
```

```
<table id="gradeTable">
  <thead>
    <tr><th>授業名</th><th>評価</th><th>削除</th></tr>
  </thead>
  <tbody></tbody>
</table>
```

```
<canvas id="gradeChart" width="400" height="400"></canvas>
```

```
<script>
const subjectInput = document.getElementById("subject");
const gradeSelect = document.getElementById("grade");
const tableBody = document.querySelector("#gradeTable tbody");
const chartCanvas = document.getElementById("gradeChart");
```

```
let grades = JSON.parse(localStorage.getItem("grades") || "[]");
```

```
function addEntry() {
  const subject = subjectInput.value.trim();
  const grade = gradeSelect.value;
```

```
  if (!subject) {
    alert("授業名を入力してください");
    return;
  }
```

```
  grades.push({ subject, grade });
  subjectInput.value = "";
  saveAndRender();
}
```

```
function deleteEntry(index) {
```

```
grades.splice(index, 1);
saveAndRender();
}
```

```
function saveAndRender() {
  localStorage.setItem("grades", JSON.stringify(grades));
  renderTable();
  renderChart();
}
```

```
function renderTable() {
  tableBody.innerHTML = "";
  grades.forEach((entry, index) => {
    const row = document.createElement("tr");
    row.innerHTML = `
      <td>${entry.subject}</td>
      <td>${entry.grade}</td>
      <td><button onclick="deleteEntry(${index})">削除</button></td>
    `;
    tableBody.appendChild(row);
  });
}
```

```
function renderChart() {
  const counts = { A: 0, B: 0, C: 0, D: 0, F: 0 };
  grades.forEach(entry => counts[entry.grade]++);
```

```
const chartData = {
  labels: Object.keys(counts),
  datasets: [{
    label: '評価分布',
    data: Object.values(counts),
    backgroundColor: ['#4CAF50', '#2196F3', '#FFC107', '#FF9800', '#F44336']
  }]
};
```

```
if (window.gradeChart) window.gradeChart.destroy();
window.gradeChart = new Chart(chartCanvas, {
  type: 'pie',
  data: chartData
});
```

```
}
```

```
    saveAndRender(); // 初期描画
```

```
</script>
```

```
</body>
```

```
</html>
```