

Übung: Notenrechner mit JavaScript

In dieser Übung erstellen wir einen einfachen **Notenrechner** mit **HTML** und **JavaScript**, der folgende Aufgaben erfüllt:

- Eingabe einer Liste von Noten.
 - Berechnung der **Durchschnittsnote** und der **höchsten Note**.
 - Ausgabe der Ergebnisse auf der Webseite.
-

HTML- und JavaScript-Code

```
<!DOCTYPE html>
<html lang="de">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Notenrechner</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      margin: 20px;
    }
    .container {
      max-width: 400px;
      margin: 0 auto;
    }
    input {
      width: 100%;
      padding: 10px;
      margin: 5px 0;
      border: 1px solid #ccc;
      border-radius: 5px;
    }
    button {
      width: 100%;
      padding: 10px;
      background-color: #28a745;
      color: white;
      border: none;
      border-radius: 5px;
      cursor: pointer;
    }
    button:hover {
      background-color: #218838;
    }
    .output {
      margin-top: 20px;
      padding: 10px;
      border: 1px solid #ccc;
      border-radius: 5px;
      background-color: #f9f9f9;
    }
  </style>
</head>
<body>
  <div class="container">
    <h1>Notenrechner</h1>
```

```

    <p>Geben Sie die Noten ein (mit Kommas getrennt):</p>
    <input type="text" id="gradesInput" placeholder="z.B. 85, 90, 78,
92">
    <button onclick="calculateGrades()">Berechnen</button>

    <div class="output" id="output">
        <p>Ergebnisse werden hier angezeigt.</p>
    </div>
</div>

<script>
    // Funktion zur Berechnung des Durchschnitts
    function calculateAverage(arr) {
        let sum = 0;
        for (let i = 0; i < arr.length; i++) {
            sum += arr[i];
        }
        return sum / arr.length;
    }

    // Funktion zur Ermittlung der höchsten Note
    function findHighestGrade(arr) {
        return Math.max(...arr);
    }

    // Hauptfunktion zur Verarbeitung der Eingabe und Anzeige der
    Ergebnisse
    function calculateGrades() {
        // Eingabe abrufen und in ein Array umwandeln
        const gradesInput =
document.getElementById("gradesInput").value;
        const gradesArray = gradesInput.split(",").map(Number);

        // Validierung: Nur gültige Zahlen zulassen
        if (gradesArray.some(isNaN)) {
            document.getElementById("output").innerHTML = "<p
style='color: red;'>Bitte geben Sie nur gültige Zahlen ein!</p>";
            return;
        }

        // Ergebnisse berechnen
        const average = calculateAverage(gradesArray);
        const highest = findHighestGrade(gradesArray);

        // Ergebnisse anzeigen
        document.getElementById("output").innerHTML = `
            <p><strong>Durchschnittsnote:</strong> $
{average.toFixed(2)}</p>
            <p><strong>Höchste Note:</strong> ${highest}</p>
            <p><strong>Eingegebene Noten:</strong> $
{gradesArray.join(", ")}</p>
        `;
    }
</script>
</body>
</html>

```

Aufgaben und Herausforderungen

1. Niedrigste Note berechnen und anzeigen

- o Füge eine Funktion hinzu, die die niedrigste Note in der Liste berechnet.
 - o Zeige die niedrigste Note in den Ergebnissen an.
 - 2. **Warnung bei weniger als zwei Noten**
 - o Ergänze eine Überprüfung, die eine Warnung ausgibt, wenn der Benutzer weniger als zwei Noten eingibt.
 - 3. **Dynamisches Styling basierend auf der Durchschnittsnote**
 - o Falls die Durchschnittsnote **80 oder höher** ist, zeige sie in **grün** an.
 - o Falls die Durchschnittsnote unter **80** liegt, zeige sie in **rot** an.
 - 4. **Validierung für leere Eingaben**
 - o Füge eine Überprüfung hinzu, die sicherstellt, dass der Benutzer etwas eingibt, bevor auf "Berechnen" geklickt wird.
 - 5. **Runde die höchste und niedrigste Note**
 - o Stelle sicher, dass die höchste und niedrigste Note auf eine Dezimalstelle gerundet angezeigt wird.
-

Bonus-Challenge

Erweitere die Funktionalität, um eine **Bewertungsskala** hinzuzufügen:

- z. B. **90-100**: "Sehr gut", **80-89**: "Gut", **70-79**: "Befriedigend", usw.
 - Zeige die Bewertung basierend auf der Durchschnittsnote an.
-

Mit diesen Herausforderungen kannst du tiefer in die Logik von JavaScript eintauchen und mehr über Arrays, Funktionen und DOM-Manipulation lernen. Viel Spaß beim Programmieren! 🤖