

# Aufgabe: Datenstrukturen Array

## 1. Grundlegende Array-Operationen

Verstehen und Anwenden von grundlegenden Array-Methoden.

- Erstelle ein Array mit den Zahlen 1 bis 10.
- Füge die Zahl 11 am Ende des Arrays hinzu.
- Entferne die erste Zahl aus dem Array.
- Finde und gib die Position der Zahl 5 im Array aus.
- Überprüfe, ob die Zahl 7 im Array enthalten ist.

## 2. Arbeiten mit Schleifen und Arrays

Arrays iterieren und deren Elemente verarbeiten.

- Erstelle ein Array mit den Namen von fünf verschiedenen Städten.
- Iteriere über das Array und gib jede Stadt in Großbuchstaben aus.
- Erstelle ein neues Array, das die Anzahl der Buchstaben in jedem Stadtnamen speichert.

## 3. Filterung und Extra

Verwenden von `filter()` und `reduce()`.

- Erstelle ein Array mit zufälligen Zahlen zwischen 1 und 100 (mindestens 20 Zahlen).
- Filtern Sie alle geraden Zahlen in einem neuen Array heraus.
- Berechne die Summe aller gefilterten geraden Zahlen.

## 4. Zweidimensionale Arrays

Verständnis und Manipulation von zweidimensionalen Arrays.

- Erstelle ein 2D-Array, das eine 3x3-Matrix repräsentiert, die mit den Zahlen 1 bis 9 gefüllt ist.

- Schreibe eine Funktion, die die Diagonalsumme der Matrix (von oben links nach unten rechts) berechnet.
- Verwandle die Matrix in eine 3x3 Matrix mit Nullen in den Ecken.

## **OPTIONAL! FREIWILLIG! NICHT PFLICHT!**

### **5. Deep Dive mit Complex Array Methods**

Fortgeschrittene Array-Transformationen.

- Erstelle ein Array von Objekten, das Informationen über mehrere Bücher enthält (Titel, Autor, Anzahl der Seiten).
- Sortiere das Array basierend auf der Anzahl der Seiten von höchstens nach mindestens.
- Erzeuge ein neues Array, das nur die Titel der Bücher enthält, die mehr als 300 Seiten haben.

### **6. Herausforderung - Spielkarte Implementierung**

Anwendung all dieser Kenntnisse zur Implementierung von kleinen Projekten.

- Erstelle ein Array, das ein Kartenspiel (52 Karten, 4 Farben, 13 Werte) simuliert.
- Schreibe Funktionen zur:
  - `shuffleDeck()`: Mische die Karten zufällig.
  - `drawCard()`: Ziehe eine Karte aus dem Deck und gib ihren Wert zurück.
  - `remainingCards()`: Gib die Anzahl der verbleibende Karten aus.