

Aufgabe 2: Praktische Aufgabe

1. Schreibe die folgende `sum` Funktion so um, dass anstelle von `console.log` eine Callback-Funktion aufgerufen wird.

```
function sum(a, b) {
  const summe = a + b;
  console.log(summe);
}

// Die neue Funktion sollte so aussehen!
function sumNeu(a, b, callback) {
  const summe = a + b;
  // console.log(summe); // Diese Zeile muss angepasst werden
}

function sumCallbackFunction(ergebnis) {
  console.log("Das Ergebnis ist: ", ergebnis);
}

sumNeu(1, 2, sumCallbackFunction);
// Das Ergebnis ist: 3
```

2. Erstelle eine Funktion `waitForFive`, die 5000ms wartet und danach die Zahl 5 returnt. Benutze hierfür ein javascript Promise und die `setTimeout` Funktion. Falls du diese Funktion nicht kennst, recherchiere im Internet.
3. Erstelle eine neue Funktion `asyncCaller`, die die `waitForFive` Funktion aufruft und das Ergebnis in einer Variable Speichert. (Benutze das `await` Keyword)
4. Erstelle eine neue Funktion `sleep`, die auch javascript Promises verwendet und die `setTimeout` verwendet. Die Funktion soll folgendermaßen verwendbar sein:

```
// Die folgende Async-Funktion wird direkt aufgerufen
(async function() {
  await sleep(500); // 500ms warten
})();
```

5. Schreibe die `waitForFive` Funktion so um, dass sie eine callback Funktion akzeptiert, die nach dem Ablauf der 5000ms aufgerufen wird.