

JS-Funktionen

Lernziele

- Schreiben von Funktionen in JavaScript
 - Aufrufen von Funktionen
 - Verwendung von Funktionsparametern
 - Erlernen, was 'Scope' ist
-

Funktionen

Funktionen sind ein grundlegendes Konzept in JavaScript. Sie enthalten eine Reihe von Anweisungen – mit anderen Worten: Sie enthalten JavaScript-Code. Funktionen müssen definiert werden. Sobald eine Funktion definiert ist, kann sie beliebig oft aufgerufen werden.

Funktionsdeklarationen

Eine Funktion kann mit einer **Funktionsdeklaration** definiert werden, die aus folgenden Bestandteilen besteht:

- dem Schlüsselwort `function`
- dem Funktionsnamen
- dem Funktionskörper (JavaScript-Anweisungen / JavaScript-Code)

```
function greet() {  
  console.log("Hi Freunde!");  
  console.log("Schön, hier zu sein.");  
}
```

! Das Definieren einer Funktion führt nicht dazu, dass der JavaScript-Code im Funktionskörper ausgeführt wird. Sie müssen die Funktion aufrufen, damit der Code ausgeführt wird.

Parameter

Funktionen können Parameter akzeptieren. Parameter können im Funktionskörper wie vordefinierte Variablen verwendet werden. Bei der Deklaration einer Funktion sind wir frei in der Wahl der Namen für die Parameter, aber es sollten aussagekräftige, kurze Namen gewählt werden.

```
function printLetter(name) {  
  console.log(  
    "Hi " + name + ", ich hoffe, es geht dir gut. Liebe Grüße, Johnny"  
  );  
}
```

```
function printSum(first, second, third) {  
  const summe = first + second + third;  
  console.log("Die Summe deiner Zahlen ist: " + summe);  
}
```

Function Calls

Wenn Funktionen definiert sind, können Sie sie aufrufen, indem Sie ihren Namen schreiben, gefolgt von Klammern ("runden Klammern"). Wenn die Funktionen Parameter verwenden, können Sie diese als Argumente in den Klammern übergeben.

```
greet();  
/*  
Dies wird folgendes in die Konsole ausgegeben:  
Hi Freunde!  
Schön, hier zu sein.  
*/  
  
printLetter("Max");  
printLetter("Jordan");  
/*  
Dies wird folgendes in die Konsole ausgegeben:  
Hi Max, ich hoffe, es geht dir gut. Liebe Grüße, Johnny  
Hi Jordan, ich hoffe, es geht dir gut. Liebe Grüße, Johnny  
*/  
  
printSum(1, 2, 3);  
printSum(3, 4, 5);  
/*  
Dies wird folgendes in die Konsole ausgegeben:  
Die Summe deiner Zahlen ist: 6  
Die Summe deiner Zahlen ist: 12  
*/
```

Scope

Der Scope definiert, wo Variablen sichtbar sind und wo sie referenziert werden können. In JavaScript gibt es verschiedene Arten von Scope, zum Beispiel:

- global Scope
- funktion Scope

Funktion Scope

Variablen, die **innerhalb einer Funktion** definiert sind, sind von außen nicht zugänglich. Aber alle Variablen **außerhalb der Funktion** können vom Funktionskörper aus aufgerufen werden:

```
const globalVariable = "ein Text";

function myFunction() {
  const localVariable = true;

  console.log(globalVariable);
  console.log(localVariable);
}

myFunction();
// gibt aus:
// ein Text
// true

console.log(localVariable); // Fehler! Variable außerhalb der Funktion
nicht verfügbar
```

Globaler Scope

Eine Variable befindet sich im **global scope**, wenn sie außerhalb einer Funktion in einer JavaScript-Datei deklariert wird. Globale Variablen sind sichtbar und können von überall in dieser JavaScript-Datei nach der Deklaration aufgerufen werden.

Ressourcen

[MDN docs: Functions](#) [MDN docs: Scope](#)