



Programmieren
Starten

Zusammenfassung

Schleifen

JavaScript - Dynamische Webseiten entwickeln



FOR SCHLEIFE

Die for-Schleife wird verwendet, wenn eine bestimmte Anzahl an Durchläufen benötigt wird. Im Schleifenkopf befindet sich die **Initialisierung der Zählervariable**, gefolgt von der **Schleifenbedingung** und einer **Anweisung**.

Beispiel:

```
for (let i = 0; i < 10; i++) { //Die Schleife läuft so lange i kleiner als 10 ist
    console.log(i);
}
//Ausgabe (untereinander): 0 1 2 3 4 5 6 7 8 9
```

WICHTIG

Die for-Schleife läuft so lange, bis die Bedingung **unwahr** (false) wird.

FOR OF SCHLEIFE

Die for-of-Schleife kann verwendet werden, wenn man ein iterierbares Objekt, also beispielsweise ein Array durchläuft.

Beispiel:

```
var cars = [
    {
        "brand": "Mercedes",
        "age": 10
    },
    {
        "brand": "VW",
        "age": 5
    },
    {
        "brand": "Toyota",
        "age": 3
    }
]
```

```
    ]  
  
    for (let car of cars) {  
        console.log(car);  
    }  
}
```

WHILE SCHLEIFE

Eine while-Schleife wird ausgeführt, **solange** deren Bedingung **true** ist. Sobald die Bedingung also false ist, wird die while-Schleife verlassen.

Beispiel:

```
let number = 0;  
  
while (number < 3) {  
    n++;  
}  
console.log(number); //Ausgabe: 3
```

DO WHILE SCHLEIFE

Die do-while-Schleife ist im Grunde wie die normale while-Schleife, jedoch mit dem Unterschied, dass bei der do-while-Schleife **zuerst** der **Code ausgeführt** und **anschließend** die **Schleifenbedingung überprüft** wird. Das bedeutet, dass Anweisungen innerhalb der Schleife in jedem Fall **mindestens 1-mal** ausgeführt werden.

Beispiel:

```
var counter = 0;  
  
do {  
    counter++;  
    console.log(counter);  
} while (counter < 100); //Ausgabe: 1 2 3 4 5 ... 100
```

VERSCHACHTELTE SCHLEIFEN

Eine verschachtelte Schleife setzt sich aus mehreren Schleifen zusammen. Dabei wird zuerst die äußere Schleife ausgeführt und anschließend die Innere.

Beispiel:

```
var days = 7;
var weeks = 5;

for (let i = 1; i <= weeks; i++) {

    console.log("Woche: " + i);

    for (let j = 1; j <= days; j++) {
        console.log("Tag: " + j);
    }
} //Ausgabe: Woche: 1 Tag: 1 Tag: 2 Tag: 3 Tag: 4 ...
```

BREAK UND CONTINUE

break

Die break-Anweisung **beendet** eine **Schleife**. Mit break kann beispielsweise überprüft werden, ob sich ein bestimmter Wert in einem Array befindet. Falls das nicht der Fall ist, soll die Schleife direkt verlassen werden.

continue

Die continue-Anweisung **beendet** den **aktuellen Durchlauf** der **Schleife**, wenn eine bestimmte Bedingung erfüllt wird und macht beim nächsten Schleifendurchlauf weiter. Mit continue kann beispielsweise nach einer bestimmten Kondition gefiltert werden.