

Objects - Objekte

Objekte:

Objekte sind Platzhalter / Container für mehrere Werte.

Die einzelnen Werte (values) werden Schlüsselworten (keys) zugeordnet, so dass es key-value-pairs (Paare) ergibt.

Die Werte können mixed sein, also number, string und boolean. Ebenso können Arrays, Objekte und Funktionen als Werte gespeichert werden.

Ein Objekt wird durch das Schlüsselwort **var** und geschweiften Klammern `{}` definiert, die einzelnen key-value-pairs sind durch Kommas getrennt.

Syntax:

```
var objektName = {  
    key_1 : value_1,  
    key_2 : value_2,  
    key_3 : value_3  
}
```

Die einzelnen Werte eines Objekts werden durch den Objektnamen und den Keynamen angesprochen, entweder durch Punkt getrennt oder der Keyname als string in eckigen Klammern:

`objektName.key_1` oder `objektName['key_1']`

```
var person = {  
    name: "Graf Koks von der Gasanstalt",  
    kreuz_gemacht: true,  
    demokratisch_proEuropa: true  
}
```

// Das Objekt person wird hier initialisiert, es hat 3 key-value-pairs

```
console.log(person.name);
```

// gibt auf der Console aus: Graf Koks von der Gasanstalt

```
console.log(person.kreuz_gemacht);
```

// gibt auf der Console aus: true

```
console.log(person.demokratisch_proEuropa);
```

// gibt auf der Console aus: true

Objekte können als Parameter an Funktionen übergeben werden.

```
var person = {  
  name: "Graf Koks von der Gasanstalt",  
  kreuz_gemacht: true,  
  demokratisch_proEuropa: true  
}
```

```
function wahl(waehler) {  
  
  if (waehler.kreuz_gemacht === true) {  
    console.log(`${waehler.name} hat seine Bürgerpflicht erfüllt.`);  
  } else {  
    console.log(`${waehler.name} ist Nichtwähler`);  
    return;  
  }  
  
  if (waehler.demokratisch_proEuropa === true) {  
    console.log(`${waehler.name} ist pro EU eingestellt`);  
  }  
}  
wahl(person);
```

// gibt auf der Console aus:

Graf Koks von der Gasanstalt hat seine Bürgerpflicht erfüllt.

Graf Koks von der Gasanstalt ist pro EU eingestellt

Auch ein Array kann ein Wert für einen Key in einem Objekt sein.

```
var hand = {  
  breite: `10 cm`,  
  laenge: `20 cm`,  
  finger: [`Daumen`, `Zeigefinger`, `Mittelfinger`, `Ringfinger`, `kleiner Finger`]  
}
```

```
console.log(hand.laenge);
```

// gibt auf der Console aus: 20 cm

```
console.log(hand.finger[2]);
```

// gibt auf der Console aus: Mittelfinger

Objekte können ebenso als Items eines Arrays gespeichert werden.

```
var student_1 = {  
  name: `Matthias`,  
  geschlecht: `m`,  
  alter : 19  
}  
var student_2 = {  
  name: `Sarah`,  
  geschlecht: `m`,  
  alter : 22  
}  
var student_3 = {  
  name: `Onur`,  
  geschlecht: `m`,  
  alter : 20  
}
```

```
var alleStudenten = [student_1, student_2, student_3];
```

// hier wird ein Array definiert, die einzelnen Items sind die Objekte student_1, student_2 und student_3

```
console.log(alleStudenten[2].name);
```

// gibt auf der Console aus: Onur

Methoden: Funktionen als Wert in einem Objekt

Ebenso kann ein Wert in einem Objekt eine Funktion sein, dann ist diese Funktion Mitglied des Objektes.

Dann ist die Funktion eine Methode des Objekts.

```
var testPerson = {  
  name: `Testperson`,  
  gewichtZumBerechnen: 10000,  
  groesseZumBerechnen: 300,  
  bodyMaxIndex: function (vorname, gewicht, groesse) {  
    this.name = vorname;  
    this.gewichtZumBerechnen = gewicht;  
    this.groesseZumBerechnen = groesse;  
    var BMI = this.gewichtZumBerechnen / (Math.pow(this.groesseZumBerechnen,  
    2));  
    console.log(`${this.name} hat einen BMI von ${BMI}`);  
  }  
}
```

```
testPerson.bodyMaxIndex(`Bud Spencer +`, 125, 1.93);
```

// Hier wird vom Objekt testperson der key bodyMaxIndex als Funktion mit drei Parametern aufgerufen und ausgeführt .

Wichtig ist hier das Schlüsselwort **this**.

Dadurch werden in dem Objekt beim Ausführen der Methode die Werte für die keys name, gewichtZumBerechnen und groesseZumBerechnen mit den Parametern vorname, gewicht und groesse überschrieben.

Auf der Console wird ausgegeben:

Bud Spencer + hat einen BMI von 33.557947864372196

Questions? You're welcome!

holger.zerbe@web.de