

JavaScript / AJAX Cheat Sheet

Variablen und Operationen

- **Variablen**
`var text = „felix“;`
`var zahl = 12;`
- **Textausgabe**
`document.write(„Hallo Welt!“);`
`alert(„Hallo Welt!“);`
- **Kommentare**
`// Einzeilig, /* Mehrzeilig */`
- **Konkatenation**
`var concat = „Hallo“ + zahl;`
- **Arithmetische Operationen**
`+, -, *, /, %`
- **logische Operationen**
`&&` (und), `||` (oder), `!` (nicht)
- **Vergleiche**
`==` (gleich), `===` (identischer typ und wert), `!=` (ungleich), `!==` (nicht identisch)
`<` (kleiner), `>` (größer), `<=` (kleiner-gleich), `>=` (größer-gleich)
- **Inkrement/Dekrement**
`++a` oder `--a` (Prä)
`a++` oder `a--` (Post)

Verzweigungen

- **if-then-else**
`if (a > b) {`
 `alert(„a ist größer b“);`
`}`
`else {`
 `alert(„a ist NICHT größer b“);`
`}`
- **switch-case-Anweisung**
`switch(i){`
 `case 1: alert(„Eins“); break;`
 `case 2: alert(„Zwei“); break;`
 `default: alert(„Fehler“); break;`
`}`

Schleifen

- **while-Schleife**

```
var i = 1;
while (i <= 10) {
    document.write(i++);
}
```

- **do-while-Schleife**

```
var i = 0;
do {
    document.write(i++);
} while (i <= 10);
```

- **for-Schleife**

```
for (var i = 1; i <= 10; i++) {
    document.write(i);
}
```

- **for-each-Schleife**

```
for each (var value in array){
    document.write(value);
}
```

- **break**

Bricht die ganze Schleife ab und führt den darauffolgenden Code aus.

- **continue**

Bricht den aktuellen Schleifendurchlauf ab und starten den darauffolgenden.

Funktionen

- **Funktion mit Rückgabewert**

```
function quadrat (zahl){
    return zahl * zahl;
}
```

Der Rückgabe-Wert sowie Übergabeparameter sind optional.

Arrays

- **Array anlegen**

```
var array = new Array (1, 2, 3, 4, 5, 6);
```

- **Mehrdimensionales Array**

```
var array = new Array (new Array(1,2,3), new Array(4,5,6));
```

- **Länge eines Arrays:** var length = array.length;

Zeitfunktionen (Endlosaufruf einer Funktion)

```
var a = setInterval(„ausgabe()“, 1000); //1sek clearInterval(a);
```

```
Nur einmal x ms warten: var a = setTimeout(„ausgabe()“, 1000); clearTimeout(a);
```

Document Object Model (DOM)

- **Bezug/Manipulation von Tag-Inhalten**

```
document.getElementById(„headline“).innerHTML = „Neue Ueberschrift“;  
document.getElementsByName(„text“)[0].innerHTML = „Neuer Text“;
```

Speziell XML

```
xml.getElementsByTagName(„schlagzeile“)[0].firstChild.nodeValue/data;  
xml.documentElement.getAttribute(„href“); (documentElement = unten „feed“)
```

Beispiel

```
<?xml version="1.0" encoding="UTF-8"?>  
<feed>  
  <schlagzeile>Lorem ipsum</schlagzeile>  
  <schlagzeile>dolor sit</schlagzeile>  
  <schlagzeile>  
    <eins>ahmet</eins>  
    <zwei>fuu-bar</zwei>  
  </schlagzeile>  
  <schlagzeile href="attribut">bar-fuu</schlagzeile>  
</feed>
```

```
xml.getElementsByTagName(„schlagzeile“)[0].firstChild.nodeValue;  
xml.getElementsByTagName(„schlagzeile“)[1].firstChild.nodeValue;
```

```
xml.getElementsByTagName(„schlagzeile“)[2].getElementsByTagName(„eins“)[0].firstChild.nodeValue;  
xml.getElementsByTagName(„schlagzeile“)[2].getElementsByTagName(„zwei“)[0].firstChild.nodeValue;
```

```
xml.getElementsByTagName(„schlagzeile“)[3].getAttribute(„href“);
```

AJAX

- **Grundlegender Verbindungsaufbau**

```
var http = new XMLHttpRequest();
```

```
function start() {  
  http.open(„GET“, „datei.xml“, true);  
  http.onreadystatechange = callback;  
  http.send(null);  
}
```

```
function callback() {  
  if (http.readyState == 4) {  
    var xml = http.responseXML;      // DOM kompatibel, oder...  
    var text = http.responseText;   // ...Inhalt als String  
    // verarbeite...  
  }  
}
```

...

```
<body onload="start()" >
```