

aus: H. Schröder, R. Steyer: JavaScript 1.8, Herdt 2014

3.2 Allgemeine Notationsregeln

Anweisungen

Über Anweisungen beschreiben Sie die Funktionalität eines Programms. Die Anweisungen werden vom Browser interpretiert und ausgeführt. Eine Anweisung wird in JavaScript durch die Angabe eines Semikolons `;` beendet.

Die letzte Anweisung in einem JavaScript-Block (einer Gruppe von Anweisungen) kann auf das Semikolon verzichten, was zu unklaren Situationen führen kann. Sie sollten deshalb eine Anweisung immer mit einem Semikolon beenden. Im Buch wird davon ausgegangen, dass **jede** Anweisung von einem Semikolon beschlossen wird.

Beispiel

```
document.write("Test");
```

Kommentare

In JavaScript können Sie einzeilige und mehrzeilige Kommentare verwenden. Sie werden vom Browser nicht interpretiert und dienen der Dokumentation des Quellcodes. Quellcode sollten Sie immer so kommentieren, dass Sie nach längerer Zeit den Quellcode nachvollziehen können und dass der Quellcode jederzeit auch von einer anderen Person verstanden und gepflegt werden kann.

//	Diese zwei Zeichen teilen dem Browser mit, dass alle folgenden Zeichen bis zum Ende der Zeile zu ignorieren sind.
/* */	Mehrzeilige Kommentare werden zwischen die Zeichen /* und */ geschrieben. Dabei kennzeichnet die Zeichenfolge /* den Anfang und die Zeichenfolge */ das Ende eines mehrzeiligen Kommentars.

Beispiel für einen einzeiligen Kommentar

```
// Ausgabe der Quadratzahlen von 1 bis 10
for(i = 1; i <= 10; i++) document.write(i * i, "<br>");
```

3.3 Reservierte Wörter

Reservierte Wörter, auch **Schlüsselwörter** genannt, sind feste, vorgegebene Wörter der Sprache JavaScript. Da jedes reservierte Wort eine feste, vorgegebene Bedeutung besitzt, dürfen Sie diese nicht als Namen von Variablen oder Funktionen verwenden. Die folgende Übersicht enthält alle reservierten Wörter von JavaScript.

abstract	debugger	final	instanceof	public	transient
boolean	default	finally	int	return	true
break	delete	float	interface	short	try
byte	do	for	long	static	typeof
case	double	function	native	super	undefined
catch	else	goto	new	switch	var
char	enum	if	null	synchronized	void
class	export	implements	package	this	volatile
const	extends	import	private	throw	while
continue	false	in	protected	throws	with