

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

JavaScript direkt einbinden

Über das `<script>`-Tag können Sie JavaScript-Code in ein HTML-Dokument einfügen. Der JavaScript-Bereich wird durch Angabe des `</script>`-Tags wieder beendet.

```
<script type="text/javascript">
  <!-- Hier werden die JavaScript-Anweisungen eingefügt -->
</script>
```

Der JavaScript-Code kann sowohl im `head`-Bereich (engl.: head = Kopf) als auch im `body`-Bereich (engl.: body = Körper, Rumpf) einer HTML-Datei eingefügt werden. Beim Laden eines HTML-Dokuments werden die JavaScript-Anweisungen sequenziell abgearbeitet. In der Regel werden Skripts im `<head>`-Tag eingefügt, um oft verwendete Funktionen und globale Variablen anzulegen (deklarieren). Skripts im `<body>`-Tag werden z. B. dazu genutzt, eine Ausgabe in das HTML-Dokument durchzuführen.

Innerhalb des `<script>`-Tags wird über das Attribut `type` der MIME-Typ der Skriptsprache angegeben, wobei diese Angabe mit HTML5 nicht mehr gefordert wird. Die Verwendung des alternativen Attributs `language` ist laut Standard grundsätzlich veraltet und sollte daher nicht mehr verwendet werden.

- ✓ Durch die Angabe des optionalen Attributs `defer` (ohne Wert) im `<script>`-Tag teilen Sie dem Browser mit, dass der JavaScript-Code keine Ausgabe im HTML-Dokument erzeugt. Diese Angabe ist rein informell und dient dazu, dass der Browser dadurch das HTML-Dokument schneller darstellen kann, da der JavaScript-Code nicht sofort analysiert (geparst) werden muss, z. B. `<script type="..." defer>`.
- ✓ Wenn Sie über JavaScript Ausgaben im HTML-Dokument vornehmen, wird dieser Code erneut interpretiert und die Webseite neu aufgebaut. Auf diese Weise können Sie über JavaScript beispielsweise JavaScript-Code erzeugen, der ebenfalls interpretiert und ausgeführt wird. Für das Erzeugen umfangreicher dynamischer Webseiten ist dies eine gängige Praxis.

In einer Webseite kann es mehrere Skriptbereiche geben. Diese bilden einen gemeinsamen Namensraum. Namensraum bedeutet, dass Elemente, die Sie in einem Skriptbereich deklariert haben, auch in den anderen (nachfolgenden) Skriptbereichen verfügbar sind.

Einbinden als externe Dateien

JavaScript-Code können Sie auch als separate Dateien einbinden. Dies ist z. B. nützlich, wenn Sie Funktionen in mehreren HTML-Dokumenten verwenden möchten. Durch die Definition der Funktionen in einer Datei steht Ihnen eine zentrale Stelle zur Verfügung, an der Sie Quellcode anpassen können, ohne alle entsprechenden HTML-Dokumente bearbeiten zu müssen.

```
<script type="text/javascript" src="URL"></script>
```

Die (ausschließliche) Verwendung externer JavaScript-Dateien ist bei modernen Webseiten der Regelfall. Nur damit können Sie die Trennung von Struktur und Funktionalität erreichen, was bei komplexeren Webseiten unabdingbar ist. Beachten Sie, dass in den Beispielen im Buch **rein aus didaktischen Gründen** meist mit internen Skript-Containern gearbeitet wird. Damit lassen sich Zusammenhänge leichter erklären. In der Praxis können Sie diese internen Skript-Container jederzeit in externe Skripte auslagern.

Über das Attribut `src` beim `<script>`-Tag geben Sie die URL zur Datei an, in der sich der JavaScript-Code befindet. Wie bei der Definition von Hyperlinks können Sie relative und absolute Pfadangaben verwenden. JavaScript-Dateien besitzen standardmäßig die Endung `*.js`, wobei das aber nicht zwingend ist.

Eine JavaScript-Datei kann im `head`-Bereich oder `body`-Bereich eines HTML-Dokuments eingebunden werden. Vorzugsweise sollten Sie diese jedoch im `head`-Bereich einbinden, wenn in der JavaScript-Datei Deklarationen durchgeführt werden. Erzeugen Sie jedoch direkt Ausgaben in der JavaScript-Datei, sollte die Datei in den `body`-Bereich eingebunden werden, da Sie im `head`-Bereich einer Webseite keine Ausgaben erzeugen können.

Beispiel: *kap03/javascript.js*

In der Datei befindet sich lediglich eine Ausgabeanweisung, die einen Text in ein HTML-Dokument schreibt.

```
document.write('Ich befinde mich in einer externen Datei.');
```

Beim Erstellen von JavaScript-Code in externen Dateien werden HTML-Anweisungen wie `<script type="text/ javascript">` weggelassen. Sie geben nur die JavaScript-Anweisungen an.

Beispiel: *kap03/javascript-extern.html*

Binden Sie die JavaScript-Datei *javascript.js* im `body`-Bereich des HTML-Dokuments ein. Dadurch wird die Anweisung in der Datei ausgeführt und die betreffende Zeichenkette ausgegeben.

```
...  
<head>  
  <meta charset="utf-8">  
  <title>JavaScript-Test</title>  
</head>  
<body>  
  <script type="text/javascript" src="javascript.js"></script>  
...  
</body>
```

Falls JavaScript nicht verfügbar ist

Ist die Verwendung von JavaScript im Browser nicht verfügbar, wurde in der Vergangenheit auf der Webseite oft eine alternative Information angezeigt. Diese wurde zwischen die Tags `<noscript>...</noscript>` eingefügt und mit üblichen HTML-Anweisungen gestaltet. In den meisten Fällen wurde dieser Bereich genutzt, um auf die fehlende JavaScript-Funktionalität des Browsers hinzuweisen. Bei modernen Webseiten wird der Besucher ohne JavaScript-Unterstützung automatisch auf eine alternative Webseite weitergeleitet, die ohne JavaScript auskommt. Oder der Besucher wird vollkommen ignoriert und bekommt eine sinnlose Webseite angezeigt.

Früher wurden für nicht scriptfähige Browser im `<script>`-Tag die Skriptanweisungen vollständig mit HTML-Kommentaren (`<!--` und `-->`) umschlossen. Da mittlerweile kein Browser mehr verwendet wird, der JavaScript-Code innerhalb eines `<script>`-Tags als HTML-Code interpretieren würde, ist die Notation der HTML-Kommentare überflüssig.

Ausführungsreihenfolge

JavaScript-Anweisungen werden in der Reihenfolge ausgeführt, in der sie in einem HTML-Dokument verwendet werden. Dadurch werden JavaScript-Anweisungen, die im `head`-Bereich eines HTML-Dokuments eingebunden werden, vor JavaScript-Anweisungen im `body`-Bereich ausgeführt. Führen Sie Ausgaben direkt in das Dokument durch, erscheinen diese beim Laden des HTML-Dokuments an der entsprechenden Stelle.

- ✓ Beim Laden eines JavaScripts werden nur die Anweisungen ausgeführt, die sich nicht innerhalb einer Funktion befinden.
- ✓ Führen Sie JavaScript-Code aus, der z. B. beim Klick auf einen Hyperlink Ausgaben im Dokument durchführt, wird der Inhalt des HTML-Dokuments überschrieben.