



- ▶ Neu in HTML5
- ▶ Tabellen
- ▶ Video und Audio
- ▶ Canvas
- ▶ CSS
- ▶ Formulare
- ▶ Mikrodaten
- ▶ Scripting

# HTML5

Das umfassende Praxis- und Referenzwerk

GÜNTER BORN



Alle Beispiele zum Download  
auf der Website zum Buch

ALWAYS LEARNING

PEARSON

# HTML5



# HTML5

Das umfassende Praxis- und Referenzwerk

GÜNTER BORN



Markt+Technik

## Bibliografische Information der Deutschen Nationalbibliothek

Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über <<http://dnb.d-nb.de>> abrufbar.

Die Informationen in diesem Buch werden ohne Rücksicht auf einen eventuellen Patentschutz veröffentlicht. Warennamen werden ohne Gewährleistung der freien Verwendbarkeit benutzt. Bei der Zusammenstellung von Texten und Abbildungen wurde mit größter Sorgfalt vorgegangen. Trotzdem können Fehler nicht vollständig ausgeschlossen werden. Verlag, Herausgeber und Autoren können für fehlerhafte Angaben und deren Folgen weder eine juristische Verantwortung noch irgendeine Haftung übernehmen. Für Verbesserungsvorschläge und Hinweise auf Fehler sind Verlag und Herausgeber dankbar.

Alle Rechte vorbehalten, auch die der fotomechanischen Wiedergabe und der Speicherung in elektronischen Medien. Die gewerbliche Nutzung der in diesem Produkt gezeigten Modelle und Arbeiten ist nicht zulässig.

Fast alle Hardware- und Softwarebezeichnungen und weitere Stichworte und sonstige Angaben, die in diesem Buch verwendet werden, sind als eingetragene Marken geschützt. Da es nicht möglich ist, in allen Fällen zeitnah zu ermitteln, ob ein Markenschutz besteht, wird das Symbol ® in diesem Buch nicht verwendet.

10 9 8 7 6 5 4 3 2 1

13 12 11

ISBN 978-3-8272-4730-8

© 2011 by Markt+Technik Verlag,  
ein Imprint der Pearson Deutschland GmbH,  
Martin-Kollar-Straße 10–12, D-81829 München/Germany  
Alle Rechte vorbehalten  
Covergestaltung: Thomas Arlt, [tarlt@adesso21.net](mailto:tarlt@adesso21.net)  
Lektorat: Dorothea Krist, [dkrist@pearson.de](mailto:dkrist@pearson.de)  
Korrektorat: Marita Böhm, München  
Fachlektorat: Georg Weiterer, Münzenberg  
Herstellung: Elisabeth Prümm, [epruemm@pearson.de](mailto:epruemm@pearson.de)  
Satz: Reemers Publishing Services GmbH, Krefeld  
Druck und Verarbeitung: Drukarnia Dimograf, Bielsko-Biala  
Printed in Poland



## Kapitel 3

# Aufzählungen, Nummerierungen und Listen

In diesem Kapitel erfahren Sie, welche Möglichkeiten HTML5 zum Gestalten von Aufzählungen, Nummerierungen und Listen bietet.

### 3.1 Arbeiten mit Aufzählungen

Zum optischen Hervorheben von Textstellen werden die betreffenden Absätze häufig mit vorangestellten Symbolen versehen. Als Symbole werden Punkte, Pfeile etc. verwendet. Man bezeichnet diese Symbole auch als Schmuckpunkte oder Bullets, und die betreffenden Dokumentstellen werden als Aufzählung interpretiert.

#### 3.1.1 Wie werden Aufzählungen realisiert?

HTML5 stellt eigene Elemente für Aufzählungen bereit. In der englischen Sprache werden Aufzählungen und Nummerierungen als Listen bezeichnet. Zur Unterscheidung wird eine einfache Aufzählung mit Schmuckpunkten vor jedem Absatz als unsortierte Liste (unordered list) interpretiert. Für die Auszeichnung solcher Listen steht das *ul*-Element zur Verfügung. Die Syntax, um eine solche Liste mit Aufzählungen zu realisieren, ist relativ einfach:

- Die betreffenden Absätze mit der Aufzählung sind durch einen einleitenden und einen abschließenden Tag der Form `<ul>...</ul>` einzufassen. Diese beiden Tags sorgen zusätzlich dafür, dass die betreffenden Absätze etwas nach rechts eingezogen werden.

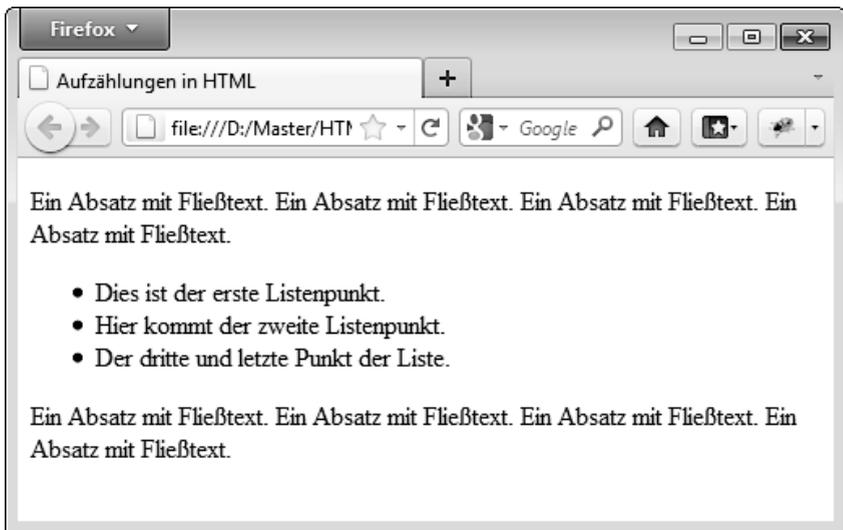
- Die Absätze der Auflistung sind zusätzlich durch den vorangestellten `<li>`-Tag und einen angehängten `</li>`-Tag zu kennzeichnen.

Nachfolgend sehen Sie eine einfache Sequenz, die eine Aufzählungsliste in HTML realisiert.

```
<ul>  
<li>Dies ist der erste Listenpunkt.</li>  
<li>Hier kommt der zweite Listenpunkt.</li>  
<li>Der dritte und letzte Punkt der Liste.</li>  
</ul>
```

Sie erkennen, dass die komplette Liste durch die Tags `<ul>...</ul>` eingefasst ist. Jeder Listenpunkt wird anschließend mit dem Tag `<li>` eingeleitet.

**Abbildung 3.1:**  
Darstellung einer  
Aufzählung im  
Firefox



In Abbildung 3.1 sehen Sie die Darstellung einer Aufzählung im Firefox. Der Browser versteht die drei Absätze mit einem (Schmuck-)Punkt und zieht den Text auch etwas nach rechts ein.

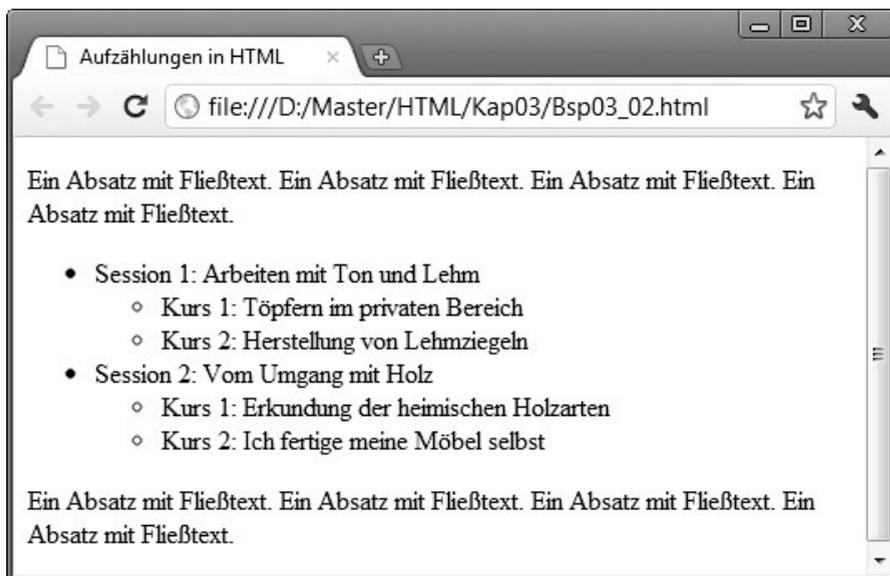
**INFO**

Der HTML5-Entwurf erfordert keinen abschließenden `</li>`-Tag in den Absätzen. In diesem Buch wird dieser abschließende `</li>`-Tag jedoch gemäß den XHTML-Empfehlungen angegeben. Frühere HTML-Versionen unterstützten zudem ein `type`-Attribut mit Werten wie »disc«, »square« etc., um die Form des Symbols vorzugeben. Aktuelle Browser kennen dieses Attribut zwar noch, aber im HTML5-Entwurf ist es nicht mehr vorgesehen.

Sie finden den HTML-Code des Beispiels in der Datei `lBeispielKap03lBsp03_01.html` des Beispielarchivs.

### 3.1.2 Gestufte Aufzählungen realisieren?

In verschiedenen Fällen reicht eine einfache Aufzählung nicht aus, um einen bestimmten Sachverhalt darzustellen. So kann es beispielsweise sein, dass ein Aufzählungspunkt seinerseits wieder aus mehreren Teilthemen besteht. Dann ist es häufig erwünscht, diese Teilthemen mit getrennten Symbolen und vergrößerten Einzügen hervorzuheben. Man spricht in diesem Zusammenhang von gestuften Aufzählungen (Abbildung 3.2).



**Abbildung 3.2:**  
Gestufte  
Aufzählung

Dies lässt sich in HTML mit geschachtelten `<ul>`-Tags realisieren. Hierzu ist lediglich innerhalb einer `<ul>...</ul>`-Sequenz ein weiteres `ul`-Element einzufügen.

```
<ul>
<li>Session 1: Arbeiten mit Ton und Lehm</li>
  <ul>
    <li>Kurs 1: Töpfern im privaten Bereich</li>
    <li>Kurs 2: Herstellung von Lehmziegeln</li>
  </ul>
<li>Session 2: Vom Umgang mit Holz </li>
  <ul>
    <li>Kurs 1: Erkundung der heimischen Holzarten </li>
```

```
<li>Kurs 2: Ich fertige meine Möbel selbst </li>
</ul>
</ul>
```

Im Browser werden die Listenpunkte in geschachtelte Aufzählungen umgesetzt. Der Browser zeigt die obige Sequenz als zweistufige Aufzählung.



Sie finden den HTML-Code des Beispiels in der Datei *lBeispielKap03lBsp03\_02.html* des Beispielarchivs.

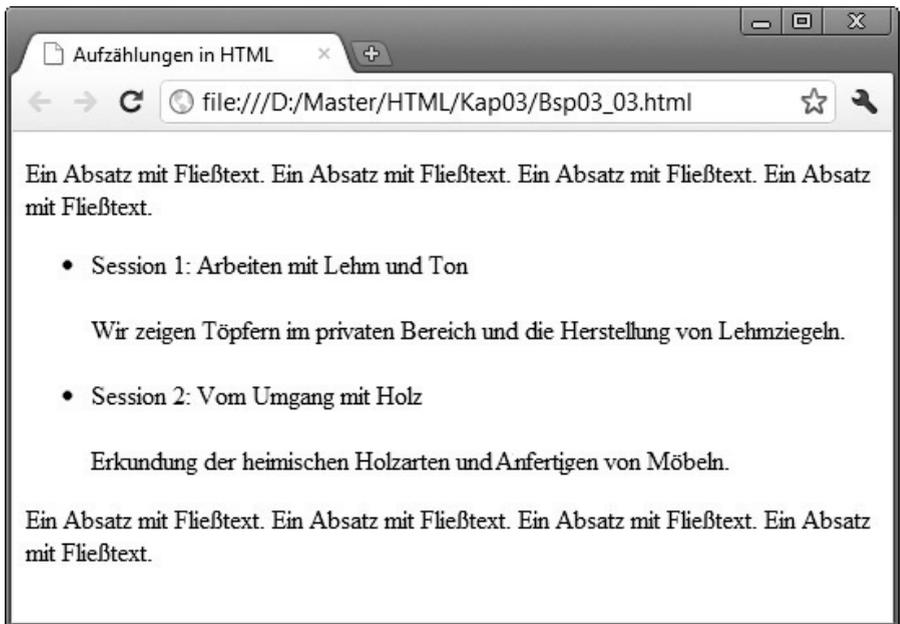


Theoretisch lassen sich Elemente wie *blockquote* und *ul* auch verwenden, um Absatzeinzüge zu realisieren. In HTML5 empfiehlt es sich aber, diese Absatzeinzüge über CSS-Anweisungen, die dem *<p>*-Tag zugeordnet sind, umzusetzen.

### Hängende Absatzeinzüge in Dokumenten

Gelegentlich besteht der Wunsch, in Aufzählungen weitere Absätze ohne eigenen Schmuckpunkt einzufügen (Abbildung 3.3).

**Abbildung 3.3:**  
Hängende  
Absatzeinzüge



Diese eingezogenen Absätze ohne eigenes Aufzählungssymbol lassen sich in HTML innerhalb eines *li*-Elements durch Einfügen von Zeilenumbrüchen mittels des *<br/>*-Tags oder durch Einfassen der Absätze in *p*-Elemente realisieren.

```
<ul>
<li><p>Session 1: Arbeiten mit Lehm und Ton</p>
<p>Wir zeigen Töpfern im privaten Bereich und die
Herstellung von Lehmziegeln.</p></li>
<li>Session 2: Vom Umgang mit Holz <br/><br/>
Erkundung der heimischen Holzarten und Anfertigen
von Möbeln.<br/></li>
</ul>
```

Im Beispielcode wurde das *p*-Element im ersten *li*-Element zum Trennen der Absätze verwendet, während im zweiten *li*-Element der `<br />`-Tag zum Einleiten eines Zeilenumbruchs zum Einsatz gelangt.

Sie finden den HTML-Code des Beispiels in der Datei *lBeispielKap03lBsp03\_03.html* des Beispielarchivs.



## Grafische Aufzählungssymbole

Gelegentlich trifft man auf Aufzählungen auf Webseiten, bei denen die Absätze mit einer vorangestellten Grafik versehen sind. HTML5 selbst sieht für so etwas nichts vor. In den vorhergehenden Abschnitten wurde gezeigt, wie sich Aufzählungen in HTML-Dokumenten realisieren lassen. Sie haben aber auch gesehen, dass die Darstellung browserabhängig ist. Vielen Webdesignern reichen daher die in HTML gebotenen Optionen zur Darstellung von Aufzählungssymbolen nicht aus. Was fehlt, sind Optionen, um weitere Aufzählungssymbole zum Beispiel in Form von Grafiken im Dokument unterzubringen.

Im Vorgriff auf das folgende Kapitel soll jetzt aber demonstriert werden, wie sich grafische Symbole als Schmuckpunkte in HTML-Dokumenten einbinden lassen. Voraussetzung ist lediglich, dass Grafikdateien mit den entsprechenden Aufzählungssymbolen (Punkten, Sternchen, Kästchen etc.) vorliegen. Um eine solche Grafik als »Pseudo«-Schmuckpunkt in ein HTML-Dokument aufzunehmen, wird diese mit dem `<img>`-Tag eingebunden. Dies sieht in der allgemeinen Form folgendermaßen aus:

```
Text des Absatzes ...
```

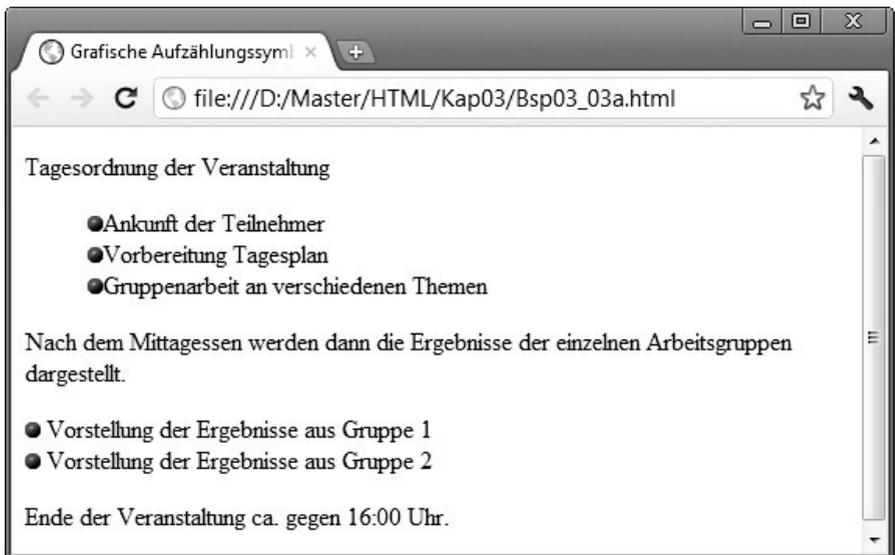
Der `<img>`-Tag bewirkt, dass die im *src*-Attribut angegebene Grafikdatei geladen und vom Browser angezeigt wird. Der Name der Grafikdatei wird anstelle des Platzhalters *datei.ext* eingesetzt. Sie können hierbei auf GIF- oder auf JPEG-Dateien zurückgreifen. Der folgende HTML-Quellcode zeigt, wie sich ein Dokument mit solchen Schmuckpunkten versehen lässt.

```
<p>Tagesordnung der Veranstaltung</p>
<ol>
Ankunft der Teilnehmer <br>
```

```
Vorbereitung Tagesplan <br>
Gruppenarbeit an verschiedenen Themen <br>
</ol>
<p>Nach dem Mittagessen werden dann die Ergebnisse der einzelnen
Arbeitsgruppen dargestellt.</p>
 Vorstellung der Ergebnisse aus Gruppe 1<br>
 Vorstellung der Ergebnisse aus Gruppe 2<br>
<p>Ende der Veranstaltung ca. Gegen 16:00 Uhr.</p>
```

Das vom Browser angezeigte Ergebnis sehen Sie in Abbildung 3.4. In der Aufzählungssequenz wurden die Absätze mit den Tags `<ol>... </ol>` eingefasst.

**Abbildung 3.4:**  
Aufzählung mit  
Grafiken



Im Gegensatz zu einer normalen Aufzählung fehlen die `<li>`-Tags am Zeilenanfang. Daher ist jeder Aufzählungsabsatz am Zeilenende mit jeweils einem `<br>`-Tag abzuschließen. Andernfalls zeigt der Browser die einzelnen Aufzählungspunkte direkt in einer Zeile an.

Bei der Analyse des angezeigten Ergebnisses fällt weiterhin auf, dass der Abstand zwischen Schmuckpunkt (Grafik) und Text in der obersten Aufzählungssequenz zu gering ist. Aus diesem Grund wurde in der zweiten Sequenz ein Leerzeichen zwischen Grafik und Textanfang eingefügt. Außerdem wurde auf die Verwendung des `<ol>`-Tags verzichtet.



Sie sollten diesen Ansatz zur Gestaltung von Aufzählungen nicht einsetzen, da die Darstellung vom verwendeten Browser abhängt. Zudem gibt es Probleme, wenn der hinter dem Grafiksymbol stehende Text länger als eine Zeile wird. Bei farbigen Hintergründen für die Webseiten benötigten Sie zudem transparente GIF-Dateien für die Schmuckpunkte. Wie sich Aufzählungen mit grafischen Symbolen mittels CSS-Anweisungen besser realisieren lassen, wird in *Kapitel 10* gezeigt.

Sie finden den HTML-Code des Beispiels zum Experimentieren in der Datei *lBeispl Kap03lBsp03\_03a.html* des Beispielarchivs.



## 3.2 Nummerierungen in HTML

Neben Aufzählungen können Sie in HTML Textabsätze durch Nummerierungen hervorheben. Ähnlich wie bei Aufzählungen zieht der Browser bei Nummerierungen die Absatztexte etwas nach rechts ein. Allerdings wird anstelle des Aufzählungssymbols eine Nummerierung in Form einer Zahl oder eines Buchstabens verwendet.

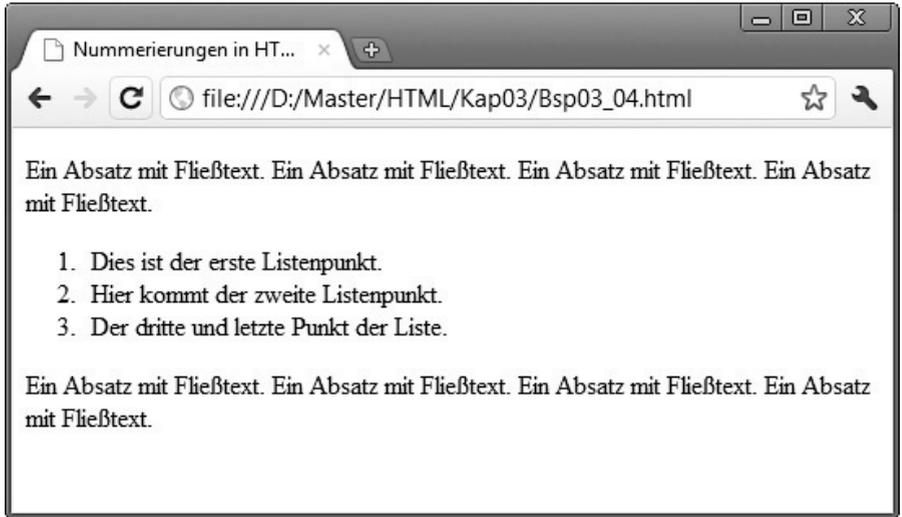
### 3.2.1 Nummerierung mit dem ol-Element

In HTML gibt es das *ol*-Element (steht für »ordered list«, sortierte Liste) zum Erstellen einer Nummerierung. Ein Nummerierungsabschnitt besitzt folgenden Aufbau im HTML-Dokument:

```
<ol>
<li>Dies ist der erste Listenpunkt. </li>
<li>Hier kommt der zweite Listenpunkt. </li>
<li>Der dritte und letzte Punkt der Liste. </li>
</ol>
```

Die komplette Liste wird durch die Tags `<ol>...</ol>` eingefasst und jeder Listenpunkt anschließend mit dem Tag `<li>` eingeleitet. In obiger Sequenz werden die Listenpunkte zusätzlich mit einem `</li>`-Tag abgeschlossen. Gemäß der HTML-Spezifikation ist dieser abschließende `</li>`-Tag aber nicht mehr erforderlich. Sofern Sie sich jedoch an die XHTML-Spezifikationen anlehnen, müssen Sie den abschließenden `</li>`-Tag verwenden. Die obigen Anweisungen (ergänzt um etwas Text) werden im Browser gemäß Abbildung 3.5 dargestellt.

**Abbildung 3.5:**  
Beispiel einer  
Nummerierung



Sie finden den HTML-Code des Beispiels in der Datei *lBeispielKap03lBsp03\_04.htm* des Beispielarchivs.

### 3.2.2 Unterbrochene Nummerierung

Ähnlich wie bei Aufzählungen kommt es häufiger vor, dass bei einer Nummerierung einzelne Absätze nicht einbezogen werden sollen. Ein entsprechendes Beispiel sehen Sie in Abbildung 3.6. Hierzu müssen Sie lediglich den betreffenden Absatz innerhalb der `<ol>...</ol>`-Tags einfügen und durch einen Absatzwechsel (`<p>`) vom Nummerierungsabsatz abtrennen. Sobald der einleitende `<li>`-Tag fehlt, unterdrückt der Browser das Nummerierungssymbol. Beim nächsten Nummerierungsabsatz wird die logisch folgende Nummer benutzt. Nachfolgend sehen Sie den HTML-Quellcode, der zum Erzeugen des Beispiels aus Abbildung 3.6 benutzt wurde.

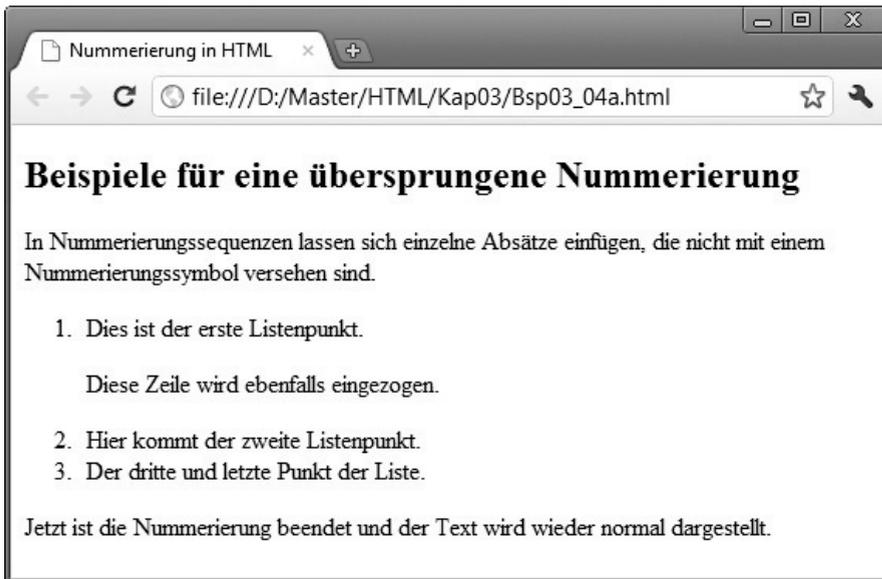
```
<!DOCTYPE html>
<html>
<head>
<title>Nummerierung in HTML</title>
</head>
<body>
<h2>Beispiele für eine übersprungene Nummerierung</h2>
<p>In Nummerierungssequenzen lassen sich einzelne Absätze einfügen, die nicht mit einem Nummerierungssymbol versehen sind.</p>
<ol>
<li>Dies ist der erste Listenpunkt.
```

```
<p>Diese Zeile wird ebenfalls eingezogen. </p></li>
<li>Hier kommt der zweite Listenpunkt. </li>
<li>Der dritte und letzte Punkt der Liste. </li>
</ol>
<p>Jetzt ist die Nummerierung beendet und der Text wird
wieder normal dargestellt. </p>
</body>
</html>
```

Sie finden den HTML-Code des Beispiels zum Experimentieren in der Datei *lBeispl Kap03lBsp03\_04a.html* des Beispielarchivs.



**Abbildung 3.6:**  
Beispiel einer  
übersprungenen  
Nummerierung



### 3.2.3 Gestufte Nummerierungen

In verschiedenen Dokumenten werden Nummerierungen geschachtelt, d. h., in einem Nummerierungsabsatz treten weitere (Unter-)Nummerierungen auf. Eine solche gestufte Nummerierung lässt sich durch Schachteln der `<ol>`-Tags innerhalb einer `<ol>...</ol>`-Sequenz erreichen.

```
<!DOCTYPE html>
<html>
<head>
<title> Gestufte Nummerierung in HTML </title>
</head>
<body>
<h2>Unsere Kurse für Senioren</h2>
<p>In diesem Winter können Sie folgende Kurse belegen: </p>
```

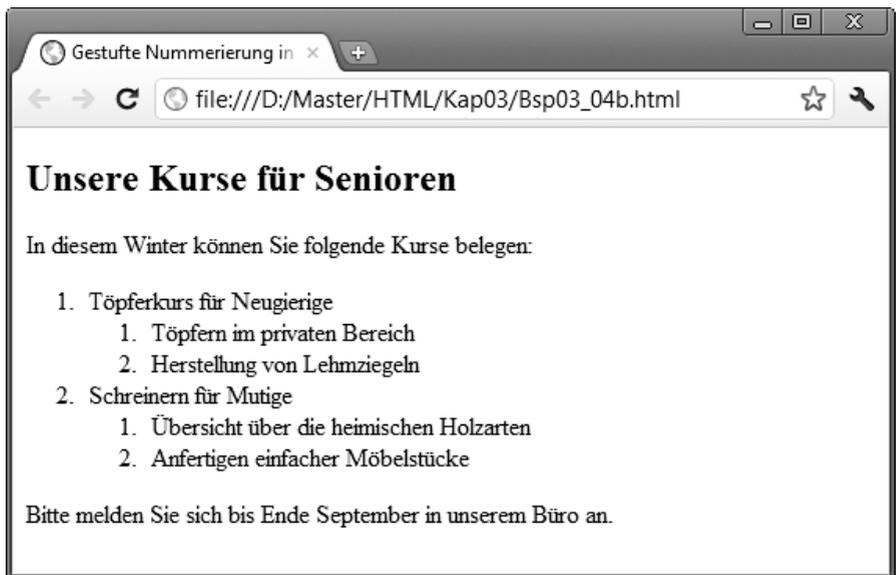
```

<ol>
  <li>Töpferkurs für Neugierige
    <ol>
      <li>Töpfern im privaten Bereich </li>
      <li>Herstellung von Lehmziegeln </li>
    </ol>
  </li>
  <li>Schreinern für Mutige <ol>
    <li>Übersicht über die heimischen Holzarten </li>
    <li>Anfertigen einfacher Möbelstücke </li>
  </ol>
</li>
</ol>
<p>Bitte melden Sie sich bis Ende September in unserem Büro an. </p>
</body>
</html>

```

Die obige Sequenz nutzt diese Erkenntnisse und generiert die in Abbildung 3.7 gezeigte Nummerierung.

**Abbildung 3.7:**  
Beispiel einer gestuften Nummerierung



Der Browser zieht die zweite Ebene der Nummerierung einfach nach rechts ein. Sofern Sie in dieser Ebene eine gestufte Nummerierung der Form 1.1, 1.2 etc. erwartet haben, werden Sie enttäuscht. HTML sieht bisher nur eine einstufige Nummerierung vor, die auch in den Folgeebenen nicht erweitert wird. Der Browser setzt lediglich die Nummerierung beim Beginn jeder neuen Ebene auf 1 zurück.



Sie finden den HTML-Code des Beispiels zum Experimentieren in der Datei *lBeispiel Kap03lBsp03\_04b.html* des Beispielarchivs.

### 3.2.4 Wechsel der Nummerierungswerte

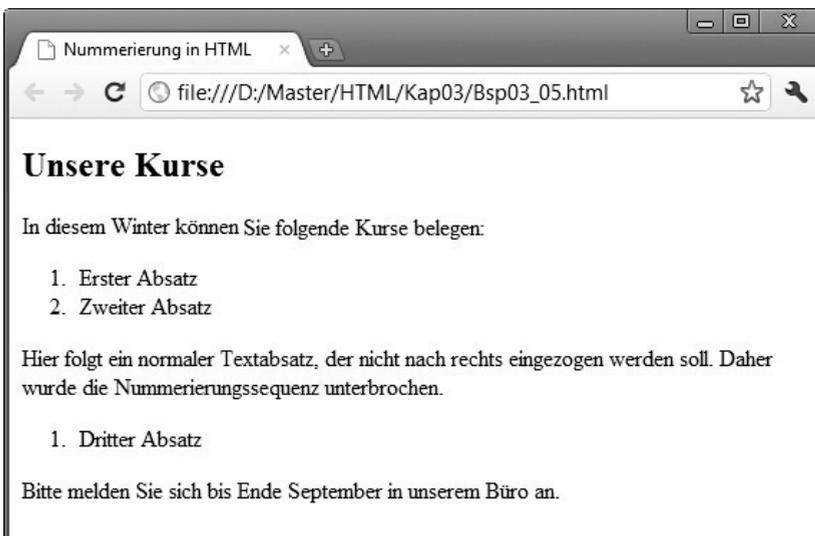
Benötigen Sie HTML-Dokumente, bei denen normale Absätze in Nummerierungssequenzen eingeschoben sind? Falls die Absätze nicht eingezogen dargestellt werden dürfen, müssen Sie diese außerhalb der `<ol>...</ol>`-Sequenz unterbringen. Bei der Wiederaufnahme der Nummerierung stehen Sie aber vor dem Problem, dass der Browser die Zählweise bei der `<ol>`-Sequenz wieder bei 1 aufnimmt. Die folgende Sequenz verdeutlicht das Problem:

```
<ol>
<li>Erster Absatz </li>
<li>Zweiter Absatz </li>
</ol>
```

Hier folgt ein normaler Textabsatz, der nicht nach rechts eingezogen werden soll. Daher wurde die Nummerierungssequenz unterbrochen.

```
<ol>
<li>Dritter Absatz </li>
</ol>
```

In Abbildung 3.8 sehen Sie, wie der Browser eine solche unterbrochene Nummerierung darstellt. Der letzte Punkt der Nummerierung beginnt wieder mit der Ziffer 1, was in diesem Fall aber nicht beabsichtigt ist. Wie lässt sich veranlassen, dass der Browser die Nummerierung (in diesem Fall mit 3) fortsetzt?



**Abbildung 3.8:**  
Darstellung einer unterbrochenen Nummerierung



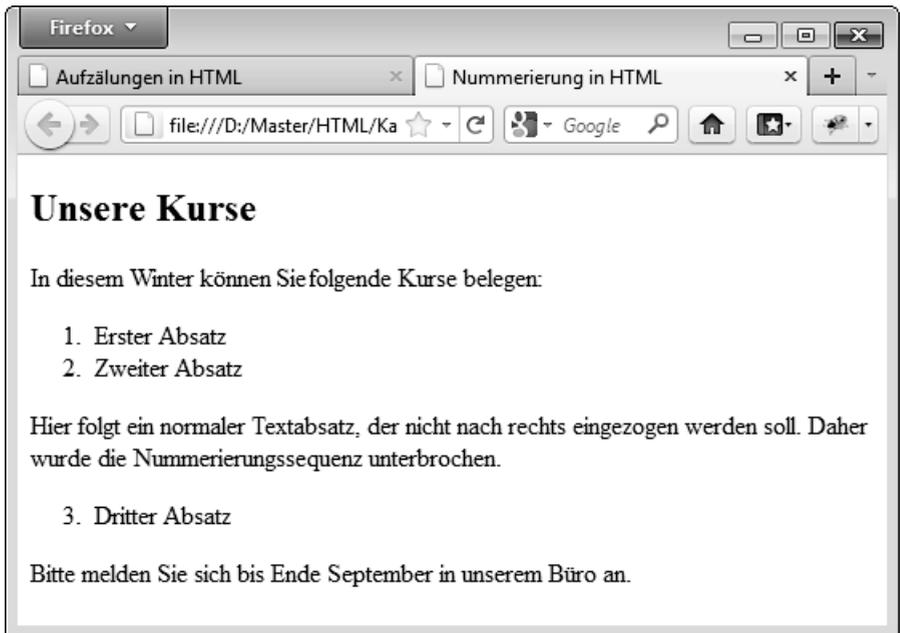
Sie finden den HTML-Code des Beispiels zum Experimentieren in der Datei *lBeispl Kap03lBsp03\_05.html* des Beispielarchivs.

HTML5 sieht hierzu das *value*-Attribut vor, dem Sie einen Integerwert zuweisen können. Die folgende Sequenz korrigiert die Nummerierung so, dass im dritten Absatz die Zahl 3 vorangestellt wird.

```
<ol>
<li>Erster Absatz </li>
<li>Zweiter Absatz </li>
</ol>
Hier folgt ein normaler Textabsatz, der nicht nach
rechts eingezogen werden soll. Daher wurde die
Nummerierungssequenz unterbrochen.
<ol>
<li value="3">Dritter Absatz </li>
</ol>
```

Einziges Problem bei diesem Ansatz ist, dass ältere Browser dieses neu in HTML5 eingeführte Attribut nicht unterstützen. In Google Chrome (ab Version 10) und im Firefox (ab Version 4) wurde die HTML-Sequenz korrekt dargestellt (Abbildung 3.9).

**Abbildung 3.9:**  
Korrigierte  
Nummerierung





Sie finden den HTML-Code des Beispiels zum Experimentieren in der Datei *lBeispielKap03lBsp03\_05a.html* des Beispielarchivs.

Möchten Sie eine komplette Nummerierungssequenz mit einem Startwert beginnen? Seit HTML 3.2 ist für den `<ol>`-Tag das *start*-Attribut vorgesehen. Diesem Attribut lässt sich ein Integerwert mit dem Startwert zuweisen. Dies wird in folgendem HTML-Codeauszug demonstriert.

`<p>In diesem Winter können Sie folgende Kurse belegen: </p>`

```
<ol>
<li>Erster Absatz</li>
<li>Zweiter Absatz</li>
</ol>
<p>Hier folgt ein normaler Textabsatz, der nicht nach
rechts eingezogen werden soll. Daher wurde die
Nummerierungssequenz unterbrochen.</p>
<ol>
<li value="3">Dritter Absatz</li>
</ol>
```

`<p>Hier folgt ein weiterer normaler Textabsatz, der
nicht nach rechts eingezogen werden soll. Daher wurde
die Nummerierungssequenz unterbrochen.</p>`

```
<ol start="4">
<li>Vierter Absatz</li>
</ol>
```

`<p>Bitte melden Sie sich bis Ende September in unserem Büro an. </p>`

In der dritten Aufzählungssequenz wird dem `<ol>`-Tag das *start*-Attribut mit dem Wert 4 zugewiesen. Dann beginnt der Browser die Nummerierung mit diesem Startwert (Abbildung 3.10).

Sie finden den HTML-Code des Beispiels zum Experimentieren in der Datei *lBeispielKap03lBsp03\_05b.html* des Beispielarchivs.



**Abbildung 3.10:**  
Korrigierte  
Nummerierung  
über das  
*start*-Attribut



### 3.2.5 Umkehren der Nummerierung

Der HTML5-Entwurf sieht noch das *reversed*-Attribut im `<ol>`-Tag vor, mit dem sich die Nummerierung umkehren lässt.

```
<ol reversed="reversed">
<li>Erster Absatz</li>
<li>Zweiter Absatz</li>
<li>Dritter Absatz</li>
</ol>
```

Allerdings unterstützte keiner der mir beim Testen zugänglichen Browser dieses Attribut .



Sie finden den HTML-Code des Beispiels zum Experimentieren in der Datei *lBeispielKap03lBsp03\_05c.html* des Beispielsarchivs.

### 3.2.6 Wechsel des Nummerierungssymbols

Insbesondere bei gestuften Nummerierungen stört es, dass der Browser keine mehrstufigen Nummerierungen der Form 1.1, 1.2 etc. anzeigen kann. Um diesen Nachteil auszugleichen, bietet es sich an, die Nummerierung von arabischen Ziffern auf Buchstaben oder römische Zahlen umzustellen. Um einen

solchen Wechsel des Nummerierungssymbols zu ermöglichen, wurde in älteren HTML-Versionen das *type*-Attribut mit den Werten aus dem oberen Teil der Tabelle 3.1 auf Seite 100 im `<ol>`-Tag verwendet. Allerdings ist das Attribut seit HTML 4 als überholt gekennzeichnet. In HTML5 sollte das Nummerierungssymbol über die CSS-Eigenschaft *list-style-type* mit den Werten aus dem unteren Teil der Tabelle 3.1 auf Seite 101 zugewiesen werden. Die folgende HTML-Sequenz zeigt, wie sich die neue CSS-Eigenschaft *list-style-type* über das *style*-Attribut vorgeben lässt.

```
<!DOCTYPE html>
<html>
<head>
<title>
Wechsel des Nummerierungssymbols in HTML
</title>
</head>
<body>
<h3>Tagesordnung der Vertriebskonferenz</h3>
<ol>
<li><b>Programm am ersten Tag</b></li>
<ol style="list-style-type:lower-roman;">
<li>Begrüßung der Teilnehmer</li>
<li>Maßnahmen zur Verbesserung des
Kundenkontakts</li>
</ol>
<li><b>Programm am zweiten Tag</b></li>
<ol>
<li style="list-style-type:lower-latin;">
Vorstellung neuer Produkte</li>
<li style="list-style-type:upper-latin;">
  Diskussion neuer Vertriebsstrukturen</li>
</ol>
</ol>
</body>
</html>
```

Das Ergebnis dieser Sequenz ist in Abbildung 3.11 zu sehen. Die erste Nummerierungsebene wird mit normalen arabischen Ziffern ausgezeichnet. Für die zweite Nummerierungsebene wurden im oberen Abschnitt dagegen kleine römische Ziffern vorgesehen.

**Abbildung 3.11:**  
 Beispiel für  
 verschiedene  
 Nummerierungs-  
 symbole



Dies lässt sich erreichen, indem das *style*-Attribut im `<ol>`-Tag in der folgenden Form angegeben wird:

```
<ol style='list-style-type: lower-roman; '>
```

Der Wert »lower-roman« sorgt dafür, dass der Browser die komplette Nummerierungssequenz bis zum abschließenden `</ol>`-Tag mit kleinen römischen Buchstaben versieht.

Die CSS-Eigenschaft *list-style-type* lässt sich auch innerhalb des `<li>`-Tags mit angeben. Dadurch lässt sich erreichen, dass ggf. nur der betreffende Absatz mit einem anderen Aufzählungssymbol versehen wird. Diese Möglichkeit wurde in der unteren Hälfte des in Abbildung 3.11 gezeigten Dokuments genutzt. Hier sind die Absätze wahlweise mit kleinen und großen Buchstaben nummeriert. Sie finden die zulässigen Werte für das *type*-Attribut sowie für die CSS-Eigenschaft *list-style-type* in Tabelle 3.1.

**Tabelle 3.1:**  
 Werte für das  
*type*-Attribut und  
 für die CSS-  
 Eigenschaft  
*list-style-type*

Wert	Auszeichnung
1	Nummerierung mit arabischen Ziffern (Standard, falls das <i>type</i> -Attribut fehlt).
a	Nummerierung mit kleinen Buchstaben der Form a, b, c etc.
A	Nummerierung mit großen Buchstaben der Form A, B, C etc.
i	Nummerierung mit kleinen römischen Ziffern der Form i, ii, iii, iv, v, vi etc.
I	Nummerierung mit großen römischen Ziffern der Form I, II, III, IV, V, VI etc.

**Tabelle 3.1:**  
Werte für das *type*-Attribut und für die CSS-Eigenschaft *list-style-type* (Forts.)

Wert	Auszeichnung
<b>Werte für list-style-type bei Nummerierungen</b>	
decimal	Dezimale Nummerierung der Art 1., 2., 3. etc.
lower-roman	Nummerierung der Art i., ii., iii., iv etc.
upper-roman	Nummerierung der Art I., II., III., IV. etc.
lower-alpha lower-latin	Nummerierung der Art a., b., c. etc.
upper-alpha upper-latin	Nummerierung der Art A., B., C. etc.
<b>Werte für list-style-type bei Aufzählungen</b>	
disc	Gefüllter Kreis als Aufzählungszeichen
circle	Leerer Kreis als Aufzählungszeichen
square	Viereck als Aufzählungszeichen
none	Kein Aufzählungszeichen

Sie finden den HTML-Code des Beispiels zum Experimentieren in der Datei *lBeispl Kap03lBsp03\_06.html* des Beispiellarchivs.



## 3.3 Weitere Listen in HTML

In HTML gibt es einige zusätzliche Tags, mit denen sich Listen erzeugen lassen. Die nachfolgenden Absätze enthalten eine kurze Übersicht über die betreffenden Tags sowie Hinweise zur Verwendbarkeit.

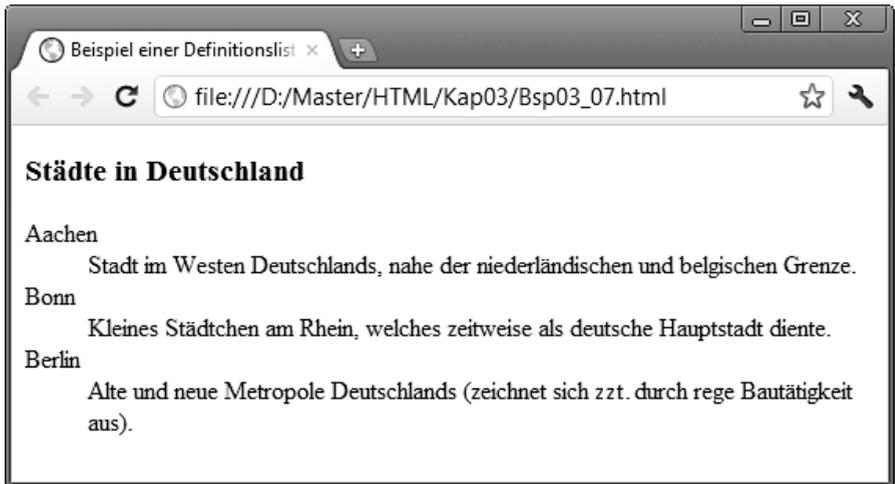
### 3.3.1 Listendarstellung mit dem dl-Element

In HTML gibt es eine besondere Art von Listen, die häufig für Definitionen verwendet wird. Diese als »Definition List« bezeichnete Listenart benötigt drei Tags, `<dl>`, `<dd>` und `<dt>`, um die betreffenden Texte auszuzeichnen. Der Tag wird meist benutzt, um Definitionen (z. B. Glossare) zu realisieren. Zum Aufbau der betreffenden Liste muss der Text ähnlich wie bei Aufzählungen zwischen die Tags `<dl>...</dl>` gestellt werden. (Die Buchstaben *dl* stehen übrigens für den Begriff »definition list«.) Jeder Eintrag der Liste besteht aus einer Definition gefolgt von einem erläuternden Text. Dies sieht dann so aus:

```
<d1>
<dt>HTML</dt><dd>Eine Sprache zur Erstellung von Webdokumenten.</dd>
...
</d1>
```

Die Definition wird durch den `< dt >`-Tag (*dt* steht dabei für »definition term«) eingeleitet und optional durch einen `< /dt >`-Tag abgeschlossen. Der Browser hebt diesen Begriff bei der Anzeige in einer getrennten Zeile hervor. Hinter dieser Definition schließt sich ein `< dd >`-Tag, ein erläuternder Text für die betreffende Definition sowie ein optionaler `< /dd >`-Tag an.

**Abbildung 3.12:**  
Anzeige einer  
Definition List



Die nachfolgende HTML-Sequenz zeigt Ihnen, wie Sie die betreffenden Tags zum Aufbau einer solchen Liste einsetzen können. Das Ergebnis ist in Abbildung 3.12 zu sehen. Jeder Eintrag der Definitionsliste wird in eine eigene Zeile gestellt. Darunter folgt etwas nach rechts eingezogen der zur jeweiligen Definition gehörende Text. Durch Verwendung des `< dt >`-Tags wird der Text der Definition von der Erläuterung getrennt. Sie haben daher die Möglichkeit, die Definitionsterme mit getrennten Formatierungen (z. B. `< dd > < b > Berlin < b >`) auszuzeichnen.

```
<!DOCTYPE html>
<html>
<head>
<title> Beispiel einer Definitionsliste </title>
</head>

<body>
<h3>Städte in Deutschland</h3>
<d1>
```

```

<dt>Aachen</dt>
<dd>Stadt im Westen Deutschlands, nahe der
    niederländischen und belgischen Grenze.
</dd>
<dt>Bonn</dt>
<dd>Kleines Städtchen am Rhein, welches
    zeitweise als deutsche Hauptstadt diente. </dd>
<dt>Berlin</dt>
<dd>Alte und neue Metropole Deutschlands (zeichnet
    sich zzt. durch rege Bautätigkeit aus).
</dd>
</dl>
</body>
</html>

```

Sie finden das Beispiel in der Datei */Beisp/Kap03/Bsp03\_07.html* des Beispielarchivs.

WWW

Der seit HTML 2.0 definierte `<dir>`-Tag zur Erzeugung von Listen im Directory-Stil wurde in der HTML-4.0-Spezifikation als »deprecated element« bezeichnet und ist nicht mehr vorhanden. Ein Attribut mit dem Namen *dir* (»direction«) legt dagegen die Ausgaberrichtung (links nach rechts oder umgekehrt) fest. Das in früheren HTML-Versionen benutzbare Attribut *compact* ist in HTML5 entfallen.

INFO

### 3.3.2 Listen mit dem menu-Element gestalten

Ein weiterer etwas exotischer Tag, der noch aus den HTML-2.0-Zeiten stammt, ist der `<menu>`-Tag. Dieser Tag lässt sich verwenden, um eine Liste (eigentlich ein Menü) in HTML zu erstellen. Ähnlich wie bei den anderen Listen müssen Sie die Menüeinträge in eine `<menu>...</menu>`-Tag-Sequenz einbinden. Jedem der Einträge ist weiterhin ein `<li>`-Tag voranzustellen. Die folgende HTML-Sequenz zeigt, wie der Tag eingesetzt wird.

```

<!DOCTYPE html>
<html>
<head>
<title> Beispiel einer Definitionsliste </title>
</head>
<body>
<h3> Früchte der Saison</h3>
<menu>
<li>Apfel</li>
<li>Birne</li>

```

```

<li>Papaya</li>
<li>Traube</li>
</menu>
<p>Beispiel aus dem HTML5-Entwurf</p>
<menu type="toolbar">
  <li>
    <menu label="File">
      <button type="button"
        onclick="fnew()">Neu...</button>
      <button type="button"
        onclick="fopen()">Öffnen...</button>
      <button type="button"
        onclick="fsave()">Speichern</button>
      <button type="button"
        onclick="fsaveas()">Speichern unter...</button>
    </menu>
  </li>
  <li>
    <menu label="Edit">
      <button type="button"
        onclick="ecopy()">Kopieren</button>
      <button type="button"
        onclick="ecut()">Ausschneiden</button>
      <button type="button"
        onclick="epaste()">Einfügen</button>
    </menu>
  </li>
  <li>
    <menu label="Help">
      <li><a href="Bsp03_06.html">Hilfe</a></li>
      <li><a href="Bsp03_07.html">Kontakt</a></li>
    </menu>
  </li>
</menu>
</body>
</html>

```

Der Aufbau der Liste gleicht der Sequenz, die zur Realisierung von Aufzählungen mit dem `<ul>`-Tag benutzt wird. Lediglich der `<ul>`-Tag wurde an dieser Stelle durch den `<menu>`-Tag ersetzt. Im unteren Teil des Beispiels habe ich noch HTML-Code aus dem HTML5-Entwurf eingefügt, der die drei Menüschaltflächen *File*, *Edit* und *Help* erzeugen soll, die ihrerseits dann Untermenüs anzeigen. Wie dies nach den Vorstellungen des W3C umgesetzt werden soll, ist in Abbildung 3.13 in der rechten unteren Ecke als Einblendung zu

INFO

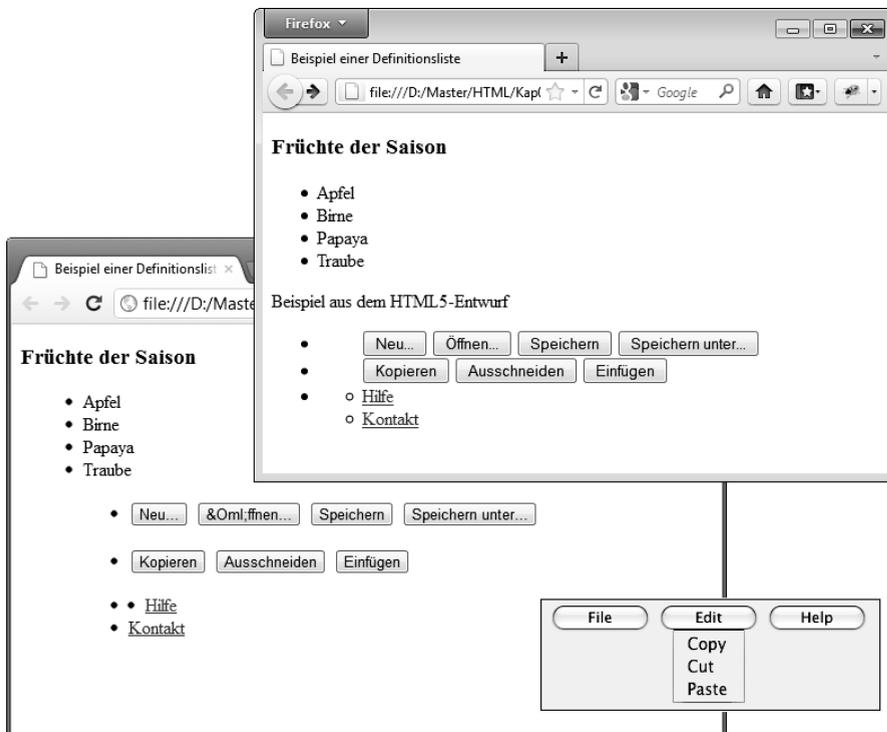
WWW

sehen. Im Hintergrund von Abbildung 3.13 sind dagegen die Darstellungen im Firefox 5/6 und Google Chrome 12/13 gegenübergestellt. Die aktuell verfügbaren Browser können mit dem *menu*-Element noch nichts anfangen und setzen es auch noch unterschiedlich um. Das Ergebnis entspricht in etwa der Anzeige einer mit `<ul>` erstellten Liste.

Der `<menu>`-Tag bringt eigentlich nicht viel Neues und ist zur Listengestaltung recht überflüssig. Den gleichen Effekt erreichen Sie mit dem `<ul>`-Tag. Daher wurde auch dieser Tag in der HTML-4.0-Spezifikation als »deprecated element« bezeichnet. Im HTML5-Entwurf habe ich das *menu*-Element aber wieder zur Listen- und Menügestaltung gefunden.

Sie finden das Beispiel in der Datei `/Beisp/Kap03/Bsp03_08.html` des Beispielarchivs.

**Abbildung 3.13:** Anzeige einer mit `<menu>` erzeugten Liste



### 3.3.3 Command für Menüs

Der HTML5-Entwurf sieht als Ergänzung zum *menu*-Element noch den `<command>`-Tag vor. Dieser soll die Ausführung eines Befehls bei Anwahl eines Menüeintrags ermöglichen. Der HTML5-Entwurf gibt hierzu folgendes Beispiel an.

```

<menu type="toolbar">
  <command type="radio"
    radiogroup="alignment" checked="checked"
    label="Left" icon="icons/all.png"
    onclick="setAlign('left')">
  <command type="radio" radiogroup="alignment"
    label="Center" icon="icons/alC.png"
    onclick="setAlign('center')">
  <command type="radio" radiogroup="alignment"
    label="Right" icon="icons/alR.png"
    onclick="setAlign('right')">
  <hr>
  <command type="command" disabled
    label="Publish" icon="icons/pub.png"
    onclick="publish()">
</menu>

```

Im `<command>`-Tag lassen sich ein Bedienelement (Menübefehl) sowie eine Ereignisbehandlungsroutine festlegen. Wird das Menüelement aufgerufen, startet das angegebene Skript zur Ereignisbehandlung. Beim Testen des Beispiels konnte aber wegen des verwendeten `menu`-Elements keiner der verfügbaren Browser die Bedienelemente anzeigen.



Sie finden das Beispiel in der Datei `/Beisp/Kap03/Bsp03_09.html` des Beispielarchivs.

# Copyright

Daten, Texte, Design und Grafiken dieses eBooks, sowie die eventuell angebotenen eBook-Zusatzdaten sind urheberrechtlich geschützt. Dieses eBook stellen wir lediglich als **persönliche Einzelplatz-Lizenz** zur Verfügung!

Jede andere Verwendung dieses eBooks oder zugehöriger Materialien und Informationen, einschließlich

- der Reproduktion,
- der Weitergabe,
- des Weitervertriebs,
- der Platzierung im Internet, in Intranets, in Extranets,
- der Veränderung,
- des Weiterverkaufs und
- der Veröffentlichung

bedarf der **schriftlichen Genehmigung** des Verlags. Insbesondere ist die Entfernung oder Änderung des vom Verlag vergebenen Passwortschutzes ausdrücklich untersagt!

Bei Fragen zu diesem Thema wenden Sie sich bitte an: [info@pearson.de](mailto:info@pearson.de)

## Zusatzdaten

Möglicherweise liegt dem gedruckten Buch eine CD-ROM mit Zusatzdaten bei. Die Zurverfügungstellung dieser Daten auf unseren Websites ist eine freiwillige Leistung des Verlags. **Der Rechtsweg ist ausgeschlossen.**

## Hinweis

Dieses und viele weitere eBooks können Sie rund um die Uhr und legal auf unserer Website herunterladen:

**<http://ebooks.pearson.de>**