

Tutorium – Tabellendesign

von Maria Lechner.

1. HTML – Die Sprache des Internets

HTML – Hypertext Markup Language (eine textbasierte Auszeichnungssprache zur Strukturierung von Inhalten). Html ist also keine Programmiersprache!
Mit Html lassen sich die Struktur, der Inhalt und das Verhalten eines Dokuments beschreiben bzw. auszeichnen, und es wird kein Programm im eigentlichen Sinne erstellt. Html sind also reine Textdokumente mit Inhalt (Texte, Überschriften, Bilder, Tabellen etc.), dem man eine Struktur (und somit Aussehen) zuweist.

HTML-Version

HTML 4.01

& verschiedene Sprachvarianten: strict, transitional, frameset

Frühere Versionen von HTML werden kaum noch verwendet und sind zum heutigen Zeitpunkt der Entwicklung auch nicht mehr sinnvoll.

Tags

Mit Tags (dt. Etiketten) kann man dem Dokument ein gewisses Aussehen zuweisen. D. h. man kann mit Tags das Dokument formatieren.

Tags werden in Spitzklammern geschrieben – `<tag>`

Es gibt Start- und Endtags. Letztere kann man von den Starttag durch einen vorangestellten Schrägstrich / unterscheiden.

`<html></html>`

Alleinstehende Tags können mit einem Schrägstrich am Ende geschrieben werden: `
` oder `
`

Whitespace

Sogenannter "whitespace" sind mehrere Leerzeichen, Tabulatoren und Zeilenumbrüche hintereinander. Diese werden vom Browser völlig ignoriert. Wenn nun also in einer Html-Datei ein Leerzeichen oder 7 Leerzeichen eingefügt werden, macht keinen Unterschied.

Nur die Tags beeinflussen die Darstellung der Website.

HTML – Basiselemente

```
<html>
  <head>
    <title>Website</title>
  </head>

  <body>Hallo Welt!</body>
</html>
```

Das *title*-Element definiert den Dokument-Titel, und das *body*-Element enthält den darzustellenden Inhalt eines Dokuments. Das Pendant zum *body*-Element ist das *head*-Element. Es enthält alle wichtigen Dokumentinformationen (wie den Titel), die nicht im Anzeigebereich des Browsers dargestellt werden sollen.

Die **syntaktische Notation** für ein Element, das einen beliebigen Text enthält, sieht also folgendermaßen aus:

```
<Elementname>Text</Elementname>
```

Um ein weiteres Element notieren zu können (sprich *Verschachteln*), wird der Text durch ein weiteres Element ersetzt:

```
<Elementname>
  <Element2>Text</Element2>
</Elementname>
```

Eltern- und Kindelement

Da das *title*-Element dem *head*-Element untergeordnet ist, wird es auch als Kindelement des *head*-Elements genannt. Und anders herum ist *head* das Elternelement des *title*-Elements.

Im Kopfteil (*head*) können noch weitere Elemente stehen, die grundlegende Informationen zum Dokument (wie Titel, Autor, Erstellungsdatum, JavaScript, StyleSheet, uvm.) geben. Diese Infos werden im Anzeigebereich des Browsers nicht dargestellt.

Im Bereich *body* steht der wesentliche Inhalt des HTML-Dokuments (wie Texte, Bilder, Hyperlinks, Tabellen, Listen uä.).

Attribute

Jedes Element besitzt gewissen Eigenschaften oder Auszeichnungsmerkmale. Diese Eigenschaften führen dazu, dass der Text im h1-Element (Überschrift 1. Ordnung) größer dargestellt wird, als wenn er in einem h2-Element (Überschrift 2. Ordnung) stehen würde.

Um nun Texte an die eigenen Wünsche anzupassen, werden *Attribute* und *Parameter* verwendet. Diese werden in Kombination innerhalb des Start-Tags eines Elements gesetzt.

```
<Element Attribut="Parameter">Text</Element>
```

Um mehrere verschiedene Eigenschaften eines Elements verändern zu können, werden die gewünschten Attribute in beliebiger Reihenfolge im Start-Tag notiert.
<Element *Attribut1*="Parameter1" *Attribut2*="Parameter2"
AttributN="ParameterN">Text</Element>

Kommentare und Quelltext-Strukturierung

Ein Kommentar wird durch die Zeichenfolge <!-- eingeleitet und endet mit der Zeichenfolge -->. Zwischen diesen beiden Zeichenfolge steht dann der Kommentartext.

<!-- Das ist ein Kommentar -->

Um bei einer umfangreichen Html-Datei den Überblick zu behalten, ist es sinnvoll, die Datei ordentlich zu strukturieren. Dafür gibt es zwar keine vorgeschriebene Norm, jedoch haben sich im Laufe der Zeit einige Regeln durchgesetzt.

- Start- und End-Tags der Elemente, die weitere Elemente enthalten, werden in unterschiedlichen Zeilen notiert. Die Einrückung soll hier gleich sein.

```
<html>  
  <head><title></title>  
  </head>  
</html>
```

- Start- und End-Tags eines Elements, das kein Kindelement enthält, sollten in einer Zeile notiert werden. Siehe Beispiel oben *title*

- Es ist egal, ob man in Tags in Groß- oder Kleinbuchstaben schreibt, man sollte sich jedoch für eine Schreibweise entscheiden. Also entweder <ELEMENT> oder <element>

HTML-Syntax

Syntax – Grammatik, Schreibweise

Html-Dateien bestehen aus reinem Text, entweder aus ASCII-Text oder Text im Format ISO 8859-1 (Latin-1). Latin-1 ist der ASSCI-Zeichensatz ergänzt um die Zeichen ü ö ä U Ö Ä ß é è ñ.

Unter Windows und Linux wird Latin-1 verwendet, am Mac muss jedoch beim Speichern das richtige Format eingestellt werden.

Entities

"Sonderzeichen" oder Zeichencode nennt man in HTML "character entities". Sie haben die Form *&zeichencode;*

Folgend eine Liste der wichtigsten Entities.

Sonderzeichen

ä ä (Umlaut)
ö ö
ü ü
Ä Ä
Ö Ö
Ü Ü
ß ß (sz-Ligatur)
< < (less than)
> > (greater than)
& &
» »
« «
geschütztes Leerzeichen
© ©
® ®

Es gibt natürlich noch mehr davon. Weitere character entities siehe <http://www.w3.org/TR/html401/sgml/entities.html>

Dokumententyp-Definition

DTD – Dokument Type Definiton

Da wir mit der Html-Version 4.01 arbeiten, verwenden wir folgenden Dokumententyp -

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
```

Als Sprachversion verwenden wir transitional, das es die am wenigsten Strenge Variante von Html ist.

Andere Sprachversionen sind strict und frameset.

Die Variante strict wird verwendet, wenn bestimmte Elemente und Attribute, die einmal zu Html gehört haben, nicht mehr verwendet werden sollen.

Die Variante frameset wird nur in Html angeführt, die die Frame-Technologie verwendet wird.

HTML-Vorlage

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
```

```
<!-- Kommentar zum Inhalt des Dokuments -->
```

```
<html>  
  <head>  
    <title>Titel</title>  
  </head>  
  <body>  
    Inhalt  
  </body>  
</html>
```

2. Textstrukturierung

Absätze

<p> (paragraph)

```
<body>
  <p>Dieser Text steht in einem p-Element.</p>
</body>
```

Textausrichtung

Standardmäßig wird der Text im <p>-Element links ausgerichtet.

Um dies zu beeinflussen, gibt es das Attribut <align> und vier Parameter, um den Text linksbündig, rechtsbündig, zentriert oder als Blocksatz auszurichten.

```
<p align="left">Dieser Absatz ist am linken Seitenrand ausgerichtet.</p>
<p align="right">Dieser Absatz ist am rechten Seitenrand ausgerichtet.</p>
<p align="center">Dieser Absatz ist am linken Seitenrand ausgerichtet.</p>
<p align="justify">Dieser Absatz ist ein Blocksatz.</p>
```

Zeilenumbrüche

 bzw.
 (line break)

```
<p>Dieser Absatz steht in einer Zeile <br />
und dieser Text steht nun in der zweiten Zeile.
</p>
```

Überschriften

Für Überschriften gibt es in Html sechs verschiedene Elemente, von h1 bis h6. Sie unterscheiden sich in der Größe der Darstellung. H1 ist das Element mit der größten Überschrift, das sich abstuft bis h6, welches die kleinste Überschrift ist.

<h1> (header)

Physische Textauszeichnung

Neben den verschiedenen Absatzformaten gibt es spezielle Elemente um Wörter, Sätze, Absätze hervorzuheben. Die Elemente zur Textauszeichnung werden als **Inline-Elemente** bezeichnet. Die Elemente für Absatzformate (p, h1 – 6) gehören zu den **Blockelementen**.

Der Unterschied zwischen dem Inline-Element und dem Blockelement ist, dass nach dem Blockelement ein neuer Absatz beginnt. Inline-Elemente integrieren sich hingegen im Textfluss.

```
<b> (bold, fett)
<i> (italic, kursiv)
<u> (underlined, unterstrichen)
<s> (strike, durchgestrichen)
<big> (groß, zeichnet einen Text größer aus)
<small> (klein, zeichnet einen Text kleiner aus)
<sup> (superscript, hochgestellt)
```

<sub> (subscript, tiefgestellt)

<tt> (teletype, Fernschreiber, zeichnet einen Text mit nicht proportionaler Schrift aus)

Schriftformatierung

Mit Hilfe des font-Elements kann die Schriftgröße, die Schriftfarbe und die Schriftart individuell angepasst werden.

Das font-Element gilt in Html eigentlich als veraltet und sollte normalerweise im CSS (Cascading Style Sheets) stehen. Für Übungszwecke kann es aber verwendet werden.

<body>

 <h1>Schriftformatierung</h1>

 <h1>Umformatierter Absatz</

font></h1>

</body>

 (ändert die Größe)

 (ändert die Farbe, Eingabe des Farbnamens oder Hexadezimal-Wert z. B. #000000 = Schwarz)

 (ändert die Schriftart z. B. serif, sans-serif, cursive, fantasy, monospace)

Dateiweite Farben

Hintergrund und Textfarbe

<body bgcolor="blue"> (blauer Hintergrund)

<body bgcolor="#00FF00"> (grüner Hintergrund)

<body text="red"> (rote Textfarbe)

<body text="#FFFF00"> (gelbe Textfarbe)

Hyperlink Farben

<body link="green" vlink="blue" alink="red">

link = Verknüpfung

vlink = besuchte Verknüpfung

alink = aktive Verknüpfung

Listen

Nummerierte Liste

 (ordered list, sortierte Liste // durchnummeriert)

 Punkt 1

 Punkt 2

Verschachtelte Liste

```
<ol>
  <li>Grundlagen
    <ol>
      <li>Punkt 1</li>
      <li>Punkt 2</li>
    </ol>
  </li>
</ol>
```

Aufzählungslisten

 (mit Aufzählungszeichen - bullet)

```
<ul>
  <li>Punkt 1</li>
  <li>Punkt 2</li>
</ul>
```

Listenattribute

siehe <http://de.selfhtml.org/navigation/css.htm#listen>

3. Tabellen

Tabellen dienen der einfachen und strukturierten Darstellung der verschiedensten Informationen. Auch als Gerüst für das Design einer Homepage sind Tabellen universell einsetzbar.

Aufbau und Strukturierung

Eine Tabelle setzt sich aus einer oder/und mehreren Tabellenzellen zusammen. Beim Definieren einer Tabelle geht man sehr strukturiert vor.

Die wichtigsten Elemente, die man für eine Tabelle benötigt, lauten:

```
<table> (Tabelle)
<tr> (table row, Tabellenreihe)
<td> (table data, Tabellendaten)
<th> (table header, Tabellenkopf)
<caption> (Titel)
```

Diese Elemente dürfen nur in einer bestimmten Reihenfolge verwendet werden.

```
<table>
<!-- Ob die Tabellenbezeichnung über oder unter der Tabelle stehen soll, wird mit
dem align-Attribut und dem Parameter top oder bottom festgelegt. -->
  <caption align="bottom">Tabellenbezeichnung</caption>
  <tr>
    <th>Fahrer</th>
```

```

        <th>Australien</th>
        <th>Brasilien</th>
        <th>Malaysia</th>
    </tr>
    <tr>
        <th>Schumacher</th>
        <td>1. Platz</td>
        <td>2. Platz</td>
        <td>1. Platz</td>
    </tr>
    <tr>
        <th>Schumacher</th>
        <td>1. Platz</td>
        <td>2. Platz</td>
        <td>1. Platz</td>
    </tr>
</table>

```

Größe, Rahmen und Abstände

Für die optische Gestaltung einer Tabelle stellt Html verschiedene Attribute zur Verfügung. Die wichtigsten lauten:

<border> (Rahmen)
 <width> (Breite)
 <height> (Höhe)
 <cellpadding> (Zellenpolster, definiert den Abstand zwischen dem Zellenrand und dem Zelleninhalt.)
 <cellspacing> (Zellenabstand, legt den Abstand zwischen den Zellen fest.)

```

...
<table border="1" width="75%" height="25%" cellspacing="1" cellpadding="2">
...

```

Tabellen und Inhalt ausrichten

Man kann die gesamte Tabelle oder den Inhalt einzelner Zellen verschieden ausrichten – und das sowohl horizontal als auch vertikal.

Tabelle horizontal ausrichten

Mit dem Attribut align und den Parametern left, right oder center.

```

...
<table align="left">
...

```

Zelleninhalt horizontal ausrichten

Auch mit align und folgenden Parametern: left, center, right.

```

...
<th align="center">

```

```
<td align="right">
```

...

Zelleninhalt vertikal ausrichten

Mit dem Attribut valign (vertical align) kann der Inhalt einer Zelle vertikal ausgerichtet werden.

Dieses Attribut besitzt vier Parameter: top, middle, bottom (unten), baseline (Grundlinie).

...

```
<th valign="top">
```

```
<td valign="middle">
```

...

Farben und Schrift

Tabellen lassen sich natürlich auch in Html farblich gestalten. Der Text des Inhaltes kann über das font-Element oder über CSS formatiert werden. Die

Hintergrundfarbe einer Tabelle lässt sich mit dem Attribut bgcolor verändern.

<bgcolor> (Hintergrundfarbe)

...

```
<table bgcolor="blue"> (Tabellenhintergrund)
```

```
  <tr bgcolor="red"> (Tabellenzeile)
```

```
    <td bgcolor="green"> (Tabellenzelle)
```

...

```
<th><font size="5">Schriftgröße 5</font></th>
```

```
<td><font face="monospace">Courier New</font></td>
```

Hintergrundbild

Um Bilder als Hintergrund in eine Tabelle einzufügen, wird das *background*-Attribut benötigt.

```
<table background="bild.gif">
```

```
<tr background="bild.gif">
```

```
<th background="bild.gif">Kopfzelle</th>
```

```
<td background="bild.gif">Datenzelle</td>
```

```
</tr>
```

```
</table>
```

Zellenverbund

Eine weitere Möglichkeit für Tabellen in Html ist das Verbinden von Zellen. So lassen sich neben- oder übereinander liegende Zellen zu einer Zelle zusammenschließen und über mehrere Spalten oder Zeilen ausdehnen.

<colspan> (column span, Spaltenspanne, um nebeneinander liegende Zellen zu verbinden)

```
<table border="1">
```

```
  <tr>
```

```
    <td>1</td><td>2</td><td>3</td><td>4</td> <!--4 Zellen in einer Reihe-->
```

```
  </tr>
```

```

    <tr>
      <td colspan="2">Zelle 1 und 2</td> <!-- Parameter ist hier der Wert,
wieviele Zellen zusammengefügt werden sollen -->
      <td colspan="2">Zelle 3 und 4</td>
    </tr>
</table>

```

Um Zellen, die übereinander liegen, verbinden zu können, verwendet man das *rowspan*-Attribut (row spanning, Zeilenspanne).

```

<table border="1" width="100%">
  <tr>
    <td>1</td> <td>2</td> <td>3</td> <td>4</td>
  </tr>
  <tr>
    <td rowspan="2">5 und 9</td> <td>6</td> <td>7</td> <td>8</td>
  </tr>
  <tr>
    <td>10</td> <td>11</td> <td>12</td>
  </tr>
  <tr>
    <td>13</td> <td>14</td> <td>15</td> <td>16</td>
  </tr>
</table>

```

Über- und nebeneinander liegende Zellen verbinden.
siehe http://de.selfhtml.org/html/tabellen/zellen_verbinden.htm#spalten_zeilenweise

4. Grafiken

Die drei wichtigsten Grafikformate für das Web sind: gif, jpg und png. Alle diese Formate verfügen über spezielle Kompressionsalgorithmen, um die Größe der Datei auch bei hohen Auflösungen und Farbtiefen so gering wie möglich zu halten.

Grafiken einbinden

Um Grafiken in Html einbinden zu können, wird das Element *img* (Image, Abbildung) benötigt.

Dieses Element besitzt keinen Gültigkeitsbereich und verfügt somit auch nicht über ein Ende-Tag.

```

<body>
  <h1>Die Freiheitsstatue</h1>
  
</body>

```

Dem img-Element wird die darzustellende Grafik als Parameter mit dem src-Attribut (source, Quelle) übergeben. Es muss der vollständige Dateiname inklusive der Dateierweiterung und des Pfades angegeben werden. Wenn man keinen Pfad angibt, geht der Browser davon aus, dass die Grafik im selben Ordner liegt wie die Html-Datei.

Alternativ-Text und title.-Attribut

Sollte es einmal vorkommen, dass die Grafik, die man einbinden möchte, nicht dargestellt wird oder angezeigt werden kann, so kann man einen alternativen Text eingeben, der anstelle des Bildes angezeigt wird. Dieses alt-Attribut steht innerhalb des img-Tags.

```

```

Wenn beispielsweise der Mauszeiger länger über einem Bild steht, so wird der Titel des Bildes angezeigt, sofern einer angegeben worden ist. Dies ist möglich mit dem title-Attribut.

```

```

Breite und Höhe

```
 (Originalgröße)
```

```
 (die halbe Höhe wird angezeigt)
```

```
 (Bild wird doppelt so breit angezeigt)
```

```
 (Prozentangaben sind auch möglich)
```

Rahmen

```
 (3 Pixel breiter Rahmen)
```

```
 (0 Pixel breiter Rahmen)
```

Grafik ausrichten

```
 (weitere Attribute sind: center, right)
```

Hintergründe

```
<body background="bild.gif"> (Ein Bild steht über dem gesamten Hintergrund)
```

```
<body bgcolor="#CCCCCC" background="bild.gif">
```

```
  <h1>Ein Hintergrund in Marmor-Stil</h1>
```

```
</body>
```

5. Geknüpftes Netz

Eine Verknüpfung zu einer anderen Seite nennt man Hyperlink, Link, auch Anker oder Verküpfung.

Um in einem Html-Dokument eine Verknüpfung zu einem anderen Dokument anzulegen, wird das a-Element verwendet.

Lokale Links

```
<a href="link.html">Testlink</a>
```

Gleiches Verzeichnis

Liegt das Ziel im gleichen Verzeichnis, so muss man lediglich die Date (mit Erweiterung) angeben.

```
<a href="link.html">Testlink</a>
```

Liegt das Ziel in einem anderen Ordner (hier "Bilder), so muss dieser angegeben werden.

```
<a href="Bilder/link.html">Testlink</a>
```

Übergeordnete Verzeichnisse

Liegt das Ziel beispielsweise in einem Ordner über dem Html-Dokument so muss es folgendermaßen notiert werden:

Ordnerstruktur: html/bilder/test.html

Verknüpfung: von test.html zu einem Bild, dass im bilder-Ordner liegt.

```
<a href="../bilder/bild.gif">Bild im Bilder-Ordner</a>
```

```
<a href="../quelle.html">Zurück zum Quelldokument</a>
```

.. steht für das übergeordnete Verzeichnis.

Globale Links

Lokale Links funktionieren natürlich nur auf dem gleichen Server. Möchte man ein Dokument referenzieren, das auf einem anderen Server liegt, muss man diese explizit angeben.

```
<a href="http://derstandard.at/?url=/?id=3294779%26_range=1">Der Standard</a>
```

Email

Wenn man auf der Website einen Link setzen möchte, der automatisch den E-Mail Client des Benutzers startet, um eine E-Mail zu versenden, muss im href-Attribut des a-Elements die Zeichenfolge mailto: und dann die entsprechende E-Mail-Adresse angegeben werden.

```
<a href="mailto:office@derstandard.at">E-Mail an den Standard</a>
```

Weitere Infos zu Email siehe

```
http://de.selfhtml.org/navigation/html.htm#verweise
```

Neue Fenster

Soll der Link in einem neuen Fenster öffnen, so gilt folgende Notation:

```
<a href="http://www.google.at" target="_blank">Google in einem neuen Fenster</a>
```

Der Parameter _self würde bewirken, dass der Link im gleichen Fenster geöffnet wird.

Grundlegende Programme

Text-Editor

für Mac

TextWrangler

<http://textwrangler.softonic.de/mac> (freeware)

bbedit

<http://bbedit.softonic.de/mac> (Testversion)

für Windows

pspad

<http://pspad.softonic.de/>

FTP-Programm

für Windows

FileZilla (freeware)

http://download.webmonic.com/FileZilla_current_setup.exe

für Mac

Cyberduck (freeware)

<http://cyberduck.softonic.de/mac>

Validieren

HMTL, XHTML, ...

<http://validator.w3.org/>

CSS

<http://jigsaw.w3.org/css-validator/>

Linkliste

Selfhtml <http://de.selfhtml.org/>

W3 Consortium <http://www.w3.org/>