

HTML - Grundlehrgang

1. Einführung

HTML ist die Abkürzung für *Hypertext Markup Language*, eine Dokumentbeschreibungssprache für das Internet. Für die Übertragung von HTML-Seiten ist das HTTP-Protokoll (*Hyper Text Transport Protocol*) zuständig. Es gibt viele Erweiterungen von HTML (z.B. Java-Script, CSS, DHTML, XML, ...) sie alle basieren jedoch auf HTML.

HTML-Dateien bestehen immer aus reinem Text ohne irgendwelchen Formatierungen. HTML ist eine Interpretersprache. HTML-Dokumente sind plattformunabhängig. HTML (bzw. der auswertende Browser) arbeiten nach dem Prinzip der Fehlertoleranz. Damit können auch Dokumente mit Syntaxfehlern weitgehend ausgewertet werden. Programmiersprachen wie z.B. Turbo-Pascal sind nicht Fehlertolerant.

2. Grundlagen

2.1 Tags

HTML-Anweisungen bestehen aus Steueranweisungen die aus sogenannten *Tags* aufgebaut sind. Ein Tag beginnt immer mit `<` und endet mit `>`. html unterscheidet nicht zwischen Groß- und Kleinbuchstaben. Ich schreibe alles was mit Befehlen zu tun hat klein. Bei Dateinamen ist Vorsicht geboten. Bekommt ein html Dokument z.B. einen kleingeschriebenen Dateinamen, ein zugehöriger Link aber eine andere Groß-Klein-Schreibweise so funktioniert es auf einem Windows-Rechner problemlos. Auf einem UNIX oder LINUX Rechner wird die Datei nicht gefunden da LINUX zwischen Groß- und Kleinschreibung unterscheidet.. Gleiches gilt für Dateinamen von Bildern. Ich empfehle daher: alle Dateinamen klein, keine Leerstellen und keine deutschen Sonderzeichen.

Es gibt zwei Formen von Tags: einen einleitenden `` (fett ein) und einen beendenden `` (fett aus). Es gibt auch einzelstehende Tags wie z.B. `
` für einen Zeilenvorschub. Auch wenn Farben umgeschaltet werden ist ein Ende-Tag nicht zwingend erforderlich. Wenn du dir unsicher bist setze immer ein Ende-Tag.

2.2 Grundstruktur

Die Endung htm oder html macht aus einem einfachen Klartext ein HTML-Dokument.

Eine HTML Seite beginnt immer mit der Anweisung

```
<html>  
    und endet mit  
</html>
```

Dies ist der äußere Container.

Im Inneren gibt es zwei Bereiche: Einen den man im Browser nicht sieht,; der header und einen sichtbaren der body. Im header werden Hintergrundinformationen über die Seite zusammengefasst (wichtig für Suchmaschinen). Der body enthält alle Informationen die der Browser anzeigen soll. Das Ganze sieht dann so aus:

```
<html>  
  <head>  
  </head>  
  <body>  
  </body>  
</html>
```

2.3 Kommentare

Ein HTML-Dokument mit Kommentaren zu versehen ist ganz leicht. Du verwendest einfach eine Anweisung die der Browser nicht versteht und deshalb ignoriert. So ist z.B. `<!-- dies ist ein kommentar -->` ein Kommentar.

Die folgende Struktur für Kommentare hat sich jedoch durchgesetzt: `<!-- hier steht der kommentar -->` .

2.4 Der Header

Eine der wichtigsten Angaben im header ist der Titel weil Suchmaschinen den Titel zum Bewerten und einordnen von Webseiten benutzen.

```
<title> Meine erste Homepage </title>
```

Achtung: title ist die englische Schreibweise. Die Ähnlichkeit zum deutschen Wort Titel führt leicht zu Fehlern (Mit diesem Fehler habe ich sehr unangenehme Erfahrungen gemacht).

3. Ein erstes einfaches HTML-Dokument

```
<html>
<head> <title> beispiel1 </title> </head>
<body text="#0000E0" bgcolor="#00FF40">           <!--text blau, hintergrund grün-->
<h1>Eine große Überschrift</h1>
<h3>Noch eine Überschrift</h3>
<br>                                               <!--zeilenumbruch-->
Jetzt beginnt normaler Text! <br> Und eine neue Zeile.
<p>                                               <!--ein absatz-->
<table width="80%" border="1" >                 <!--eine tabelle mit 3 spalten-->
<tr>                                              <!--und zwei zeilen-->
  <td width="20%">Eins </td>
  <td width="30%">Zwei </td>
  <td width="50%">  Drei </td>                 <!--leerzeichen werden nicht dargestellt-->
</tr>
<tr>
  <td>Vier </td>
  <td>Fünf </td>
  <td>&nbsp; &nbsp; &nbsp; &nbsp; Sechs </td>                 <!--&nbsp; erzeugt erzwungene leerzeichen-->
</tr>
</table>
</p>                                               <!--absatz ende-->
<a href="seite2.html">zur zweiten Seite</a>         <!--ein link zur seite 2-->
<p><a href="seite2.html" target="_blank">zur Seite 2 </a></p>   <!--noch ein link zur seite 2-->
 <!--ein bild einfügen-->
<br>
<a href="bfestung.jpg">  </a>
                                                    <!--aus einem kleinen bild wird ein grosses-->
<br> <a href="18.03.00_sunny1.xls">Solardaten Tagesverlauf </a>   <!-- excel tabelle aufrufen-->
<p> <a href="pdfstest.pdf">PDF - Datei </a> </p>           <!--pdf-datei-->

</body> </html>
```

4. Anker innerhalb einer Seite; Textmarken

```
<html><head><title>anker</title></head>
<body text="#000000" bgcolor="#FFFFFF">
<a name="oben"> <h2>Hier ist oben</h2> </a>
<a href="#unten"> nach unten</a>
<p>text anfang</p><p></p><p></p><p></p><p></p><p></p><p></p><p></p>
<p></p><p></p><p></p><p></p><p></p><p></p><p></p><p></p><p></p>
<p></p><p></p><p></p><p></p><p></p><p></p><p></p><p></p><p></p>
<p></p><p></p><p></p><p></p><p></p><p></p><p></p><p></p><p></p>
<a name="unten"> <h2>Hier ist unten</h2> </a>
<a href="#oben"> nach oben</a>
</body></html>
```

5. Pfadangaben

Bei Verweisen spielt die Angabe des Pfades in dem die Datei steht eine entscheidende Rolle. Es gibt zwei Möglichkeiten:

1. absolute Pfadangabe z.B. <http://suse.de> für das Internet oder `file://c:/zubl/html_lehrgang` auf dem eigenen Rechner. Die Methode mit `file ...` funktioniert jedoch nicht mehr wenn die Dateien auf einen anderen Rechner oder ins Internet gebracht werden.

2. Relative Pfadangabe z.B. für das Einfügen eines Bildes `` hier muss das Bild im gleichen Verzeichnis stehen. `` hier steht das Bild in einem Unterverzeichnis der HTML-Dateien. `` hier steht es drei Verzeichnisse höher.

Tip für Anfänger: alles (HTML-Dateien, Bilder, ...) in das gleiche Verzeichnis !!! Dabei den Bildernamen ein `bb ..` voranstellen, um sie von den HTML-Dateien zu trennen z.B. `bb_schoenau.jpeg`.

6. Text formatieren

Der Text wird mit vordefinierter Textgröße, Textfarbe und Hintergrundfarbe innerhalb des Browsers dargestellt. Diese Darstellung kann der Designer einer Webseite nicht beeinflussen. Sie sind von der Plattform des Betrachters abhängig.

Auch wenn man im Editor Leerzeichen (mehr als Eins) und Zeilenschaltungen eingefügt werden, werden sie im Browser nicht dargestellt. Die Größenveränderungen des Browsers werden sich jedoch auswirken. Der Text wird immer in der Breite angepasst.

Bei Netscape wird die Schrift unter: Bearbeiten; Einstellungen; Schriftart geändert.

6.1 Absätze

 (break) erzeugt einen Zeilenumbruch; <p></p> (paragraph) erzeugt einen neuen Absatz, das Ende-Tag kann entfallen.

6.2 Ausrichtung

<div align="center"> Dieser Text steht in der Mitte </div>. div erzeugt auch einen Zeilenvorschub.

<div align="justify"> erzeugt Blocksatz.

<div align="right"> hier steht der Text rechts.

6.3 Trennlinien

Beispiel:<hr width="80%" align="center" size="3" noshade> . Der voreingestellte Wert für Size ist 2. noshade erzwingt eine Trennlinie ohne Schatten.

6.4 Überschriften

<h1></h1> bis <h6> erzeugt Überschriften in verschiedenen Größen und einen neuen Absatz.

<h3 align="center"> In der Mitte </h3>

6.5 Sonderzeichen

lt	less than	<	<	gt	greater than	>	>
amp	ampersand	&	&	quot	quotation	"	"
auml	a Umlaut	ä	ä	Auml	A Umlaut	Ä	Ä
ö, ü, Analog							
szlig	sz gefolgt von lig	ß	ß				

6.6 Textformate

Man unterscheidet zwischen logischen und physischen Befehlen zur Hervorhebung von Text.

Logische Anweisungen überlassen weitgehend dem Browser des Betrachters wie der Text aussehen soll.

Physische Anweisungen legen genauer fest wie der Text dargestellt werden soll. Jedoch kann es auch dabei Unterschiede geben.

Logische Anweisungen:

<cite>	Zitat	<code>	Quellcode, meist nichtproportional
<dfn>	Definitionen		Betonung; meist kursiv
	meist fett	<var>	Variablenamen; meist kursiv

Physische Anweisungen

	fett	<big>	größer als Standartschrift
<i>	kursiv; italic	<small>	kleiner als Standartschrift
<sub>	tiefgestellt	<sup>	hochgestellt
<u>	unterstrichen		

Die font Anweisung;

die Größen 1 ... 7 sind relative Schriftgrößen, die sich an der Standartschriftgröße des Browsers orientieren

Beispiel: Text

Auch und sind möglich.

7. Grafiken

Im Internet existieren zwei Grafikformate: gif und jpeg (aus DOS-Zeiten jpg)

Das gif-Format: Pixelformat; (kein Vektorformat; Berechnung von grafischen Formen); 256 Farben;

hochkomprimiert; gif-Bilder können als Animation laufen.

Das jpeg-Format Pixelformat; 16,7 Millionen Farben; hochkomprimiert

Beispiele: `` nicht verzerrt weil nur die Breite angegeben wurde
`` möglicherweise stark verzerrt
`<center> </center>`
`<div align="right"> </div>`
``

8. Frames

Mit Frames wird der Anzeigebereich eines Browsers in einzelne Segmente unterteilt, die unabhängig voneinander mit Inhalt gefüllt werden können. Ursprünglich von der Firma Netscape entwickelt, sind Frames ein fester Bestandteil von HTML 4 geworden. Frames sollten von allen neueren Browsern verstanden werden, wenngleich sie bei Designern sehr umstritten sind.

Beispiel1: kleines frame auf der linken Seite (für die Inhaltsangabe) und dem restlichen Bereich für den eigentlichen Inhalt.

```
<html> <head> <title>frames1</title></head>
```

```
<frameset cols="100, *" frameborder="yes" border="0">  
  <frame src="links.html" name="navigation"> </frame>  
  <frame src="rechts.html" name="hauptfenster"> </frame>  
</frameset>
```

```
</html>
```

```
<html> <head> <title>links</title> </head>  
<body text="#000000" bgcolor="#FF0020"> </body>  
</html>
```

```
<html> <head> <title>rechts</title> </head>  
<body text="#000000" bgcolor="#00C000"> </body>  
</html>
```

Beispiel2: frames oben und unten. Statt cols muss rows verwendet werden.

Beispiel3: kleines frame links; rechtes frame in Titel (oben) und Hauptfenster (unten) unterteilt.

```
<html> <head> <title>frames3</title> </head>
```

```
<frameset cols="100, *" frameborder="yes" border="0">  
  <frame src="links.html" name="navigation"> </frame>  
  <frameset rows="70, *">  
    <frame src="oben.html" name="titel"></frame>  
    <frame src="unten.html" name="main"></frame>  
  </frameset>  
</frameset>
```

```
</html>
```

```
<html> <head> <title>oben</title> </head>  
<body text="#000000" bgcolor="#707070"> </body>  
</html>
```

```
<html> <head> <title>oben</title> </head>  
<body text="#000000" bgcolor="#B000FF"> </body>  
</html>
```

Warum bekommen frames einen Namen (z.B. main)?

Wenn ein Verweis von einem frame in ein anderes führt muss es eine Möglichkeit geben das frame anzusprechen. Dies geschieht mit target.

Beispiel: Erstes Kapitel
 zurück
 zur Testseite
 target=" _blank " öffnet ein neues Fenster
 zur Testseite
 target=" _top " öffnet Zieldatei im gesamten Framesetfenster

Weitere Einstellungen:

<frame src="test.html" noresize="noresize" /> *Benutzer kann Größe des frames nicht ändern*
<frame src="test.html" scrolling="yes" /> *Rollbalken; auto, yes, no*

9. Weitere Tabellen; Zellen verbinden

```
<html>
<head>
<title>tabelle</title>
</head>
<body>
<table width="100%" border="1">
  <tr height="40%"> <td>1. </td> <td>1.1 </td></tr>
  <tr height="40%"> <td colspan="2">2. zwei verbundene Spalten</td>
  <tr> <td>3. </td> <td>3.1 </td></tr>
  <tr> <td>4. </td> <td rowspan="3">4.1 drei verbundene Zeilen</td></tr>
  <tr> <td>5. </td>
  <tr> <td>6. </td>
</table>
</body>
</html>
```

10. Imagemaps

```
<html> <head> <title>imagemaps</title> </head>
<body text="#000000" bgcolor="#FFFFFF" link="#FF9F00" alink="#FF0020" vlink="#FF0000">



<map name="karte">
  <area shape="rect" coords="10,10, 100, 100" href="seite2.html" alt="Hallo" />
  <area shape="circle" coords="300,180, 40" href="seite2.html" alt="Insel" />
  <area shape="rect" coords="350,50, 400, 100" href="seite2.html" alt="Stern" />
  <area shape="polygon" coords="100,100, 200, 100, 150, 200, 100, 100" href="seite2.html" alt="Polygon" />
  <area shape="default" coords="0,0, 400, 300" href="seite2.html" alt="Rest" />
</map>
</body> </html>
```

11. Style-Sheets

Mit Style-Sheets kann das Layout von Webseiten genau festgelegt werden. Es kann eine klare Trennung von Struktur und Layout vorgenommen werden (wie bei Formatvorlagen von Textverarbeitungsprogrammen). Style-Sheets haben jedoch auch Nachteile. Auf der Zielplattform muß eine große Kompatibilität zu einem vorgegebenem Standard gegeben sein (diverse Schriften, leistungsstarke Grafikkarte, nicht zu geringe Auflösung ...). Ein weiterer Nachteil, nicht alle Style-Sheet Anweisungen werden von allen Browsern unterstützt.

Beispiel 1

```
<html>
<head>
<title>cssscript1</title>
<style type="text/css">
  <!--p.basic {font-size:11pt; color:black;}
  p.green {font-size:18pt; color:green;}
  h1 { font-size:12pt; color:#FF0000; font-family:arial; }
  b { font-size:24pt;color:#0000FF;font-family:arial; }
  -->
</style>
</head>
<body>
  <p class="green">Dies ist optimal für Überschriften</p>
  <p class="basic">Mit der Formatierung basic schreiben wir normal</p>
  Normaler Text<br>
  <h1>Text mit dem neuen h1-Format</h2>
  <b>Text mit dem neuen b-Format</b>
</body></html>
```

Die Style-Anweisungen werden als Kommentar „getarnt“ damit Browser die keine Style-Sheets verarbeiten sie übergehen.

Beispiel 2

```
<html>
<head>
<style type="text/css">
  <!--
  @import url(cssvorlage1.css);
  -->
</style>
</head>
<body>
  Dies ist ein normaler Text.<br>
  <b>Dies ist ein fetter Text.</b><br>
  <i id="neu">Dies ist ein kursiv-formatierter Text.</i><br>
  <p class="neu">Neues Textformat.</p>
  <h3>Überschrift H2</h3>
  <h4>Überschrift H4</h4>
</body></html>
```

Hier stehen die Formatierungen in einer eigenen Datei: cssvorlage1.css. Sie kann somit von mehreren html-Seiten benutzt werden. Diese Datei wird mit dem Html-Editor erstellt und als css-Datei gespeichert

```
<!--
BODY { margin-top:100px;margin-bottom:10px;margin-left:100px;}
B { font-family:Courier New,Courier;font-size:30pt; color:#FF0000; }
I { font-family:Courier New,Courier;font-size:20pt; color:#0000FF; }
P.neu { font-size:16pt;color:#987654;font-family:Arial,Courier New,Times; }
H1,H2,H3 { color:pink; }
H4,H5 { color:green; }
-->
```

HTML-Tags	Bedeutung	
<html></html>	HyperTextMarkupLanguage	Anfang und Ende des Dokumentes
<head></head>	Kopf	Anfang und Ende des Heads
<title></title>	Titel	Inhalt erscheint in der Kopfzeile
<body></body>	Körper	Anfang und Ende des Textkörpers
<h1></h1>	Heading1	sehr große Überschrift, fett
bis		
<h6></h6>	Heading6	sehr kleine Überschrift, fett
	bold	fett
<i></i>	italics	kursiv
<u></u>	underlined	unterstrichen
<p></p>	paragraph	Absatz
 	break	Zeilenwechsel
<hr>	horizontal rule	fügt eine Linie ein
	list	Liste, eingerückt
	ordered list	Liste mit Zahlen
	unordered list	Liste mit Punkten
<body background="bild_3.JPG">		Hintergrundbild
< a href="http://www.suse.de"> Link ins www 		Link; anchor hyper reference
<a href="bild1.jpg" 		Link zum großen Bild bild1.jpg
nach unten		Querverweis auf internen Anker
hier ist unten		unten, Name des internen Ankers
 text 		Link auf eigene e-mail-adresse
		image source
<nobr></nobr>	verhindert Zeilenumbruch	
 	erzwungenes Leerzeichen	
<sub>	Tiefstellen	
<sup>	Hochstellen	
<small>	Text kleiner	
<big>	Text größer	
<div align="center"></div>		Ausrichtung von Text; left, right, justify
		Schriftgröße, Farbe, Art
 		

<p>Home</p> <p>1. Tabelle mit verbundenen Zellen</p> <p>2. Java Script Schleifen</p> <p>3. Java Script Verzweigungen</p> <p>4. Imagemap</p> <p>5. PDF – Datei</p> <p>6. Bericht über Python</p> <p>7. Privates</p>	<div data-bbox="608 286 1390 338" style="border: 1px solid black; padding: 2px;">Überschrift</div> <p>Startseite</p>
--	--

1.	1.1
2. zwei verbundene Spalten	
3.	3.1
4.	4.1 drei verbundene Zeilen
5.	
6.	