

# HTML-Kurs

## Was ist HTML

HTML ist die englischsprachige Abkürzung von:

**H**yper  
**T**ext  
**M**arkup  
**L**anguage

Compiler (hier Browser): Netscape Navigator  
Internet Explorer  
Opera, iCab (Macintosh)

Dateiendung: wahlweise *.htm* oder *.html*

Da die Interpretation von Internet-Seiten mittels o. g. Browsern erfolgt und die Software-Giganten Netscape und Microsoft in inoffiziellen Markt-Machtkämpfen stehen, wird es dem HTML-Programmierer dahin gehend schwer gemacht, dass eindeutige Codes von den Browsern unterschiedlich interpretiert oder gar übergangen werden. Dies wird insbesondere erkennbar bei dem Einfügen von Javascript, CSS (Cascading Style Sheets), VBScript, DHTML etc. in die HTML-Seiten. Doch dazu später mehr.

## Vorwort

Um eine gut funktionierende Internetseite zu erstellen bedarf es vorab einiger Hinweise. Einige Punkte werden Sie sicher zur Zeit noch nicht verstehen können, werden jedoch später dankbar sein, diese Informationen vorab bekommen zu haben.

Wenn Sie sich mit HTML beschäftigen möchten, können Sie getrost auf reine WYSIWIG-Editoren, z. B. die den Browsern zugehörige Programme Netscape Composer, Front Page Express (Internet Explorer), NetObjects Fusion, Sierra WebArtist oder DATA BECKER's Web Artist usw. verzichten. Sollten Sie ein derartiges Programm haben und bereits schon damit gearbeitet haben, machen Sie sich den Spass und schauen Sie sich diese Seite in einem HTML-Editor (Hier z. B. Hot Metal) an. Sie werden die Hände über dem Kopf zusammenschlagen, wie überflüssig einige der automatisch erstellten HTML-Codes sind.

Und – das werde ich in meinen weiteren Ausführungen immer wieder betonen – je mehr in einer Seite steht, seien es nun Bilder oder Text, desto höher wird die Ladezeit einer Seite, was den Surfer auf Ihrer Seite nicht gerade freut. Erstens sucht er sicher eine bestimmte Information auf Ihrer Seite und nicht die große Grafik auf der Eingangsseite und zweitens tickt ja immer die Online-Uhr.

### Hier nun ein paar Grundgedanken:

- Geben Sie Ihren HTML- und Grafik-Dateien aussagekräftige aber dennoch kurze Namen alles in Kleinbuchstaben mit höchstens 8 Zeichen vor der Dateiendung (HTM,HTML,GIF,JPG). Wenn Sie später Ihre Homepage ins Web stellen möchten, machen Ihnen sonst einige WebSpace-Anbieter Querelen und Sie können Ihre ganze Seite neu machen.
- Packen Sie alle Dateien, welche zu Ihrer Homepage gehören in einen separaten Ordner. Erstellen Sie, wenn möglich, keine Unterordner – auch hier machen manche WebSpace-Provider nicht mit.
- Testen Sie Ihre Seiten mit verschiedenen Browsern und verschiedenen Versionen (Netscape, Internet Explorer, eventuell auch Opera) auf ihre einwandfreie Funktionstüchtigkeit. Dies ist besonders wichtig, wenn Sie Javascript, CSS oder DHTML in Ihre Seiten eingebunden haben.
- Überprüfen Sie das Erscheinungsbild Ihre Seiten unter verschiedenen Bildschirmauflösungen (480 x 640, 600 x 800, 1024 x 768 etc.). Nicht alle Surfer verfügen, wie Sie vielleicht, über einen 19“-Bildschirm mit der höchsten Auflösung. Um alles lesen zu können, müssen Sie an dieser Stelle vielleicht noch kleine Änderungen vornehmen, um tunlichst ein horizontales Scrollen zu vermeiden. Ein guter Mittelweg von Anfang an ist, Ihre Homepage unter einer Auflösung von 600 x 800 zu erstellen.
- Geben Sie Ihrer Startseite (auch bei der Verwendung von *Frames*, dann dem *Frameset*) vorsichtshalber den Namen index.htm oder index.html. Warum? Ja – die WebSpace – Provider!
- Verwenden Sie für Ihre Homepages keine ausgefallenen Schriftarten. Der Browser, der die Schriftarten darstellen soll, greift auf die beim Surfer im Betriebssystem installierten Schriften zu. Ist dann Ihre in der Homepage verwendete Schrift nicht installiert, kann es zu falschen Darstellungen führen und Ihre Homepage wirkt verhunzt. Verwenden Sie z. B. Arial (Diese Seiten sind in Arial geschrieben) oder Times New Roman (Die Befehle sind hier in dieser Schriftart dargestellt). Licht in den Schriftartenschungel bringen erst CSS-Scripts, auf die ich hier nicht näher eingehen will.

## Das Grundgerüst

Das Grundgerüst einer HTML-Datei ist stets gleich.

Bevor es richtig losgeht informiere ich den Browser über meine derzeit verwandte Version von HTML, da diese Programmiersprache ständigem Wandel unterliegt. (derzeit HTML 4)

Das Festhalten dieser Information über die verwendete Sprachversion entspricht den SGML-Vorgaben und ist empfehlenswert.

Syntax:

```
<!doctype html public „-//w3c//dtd html 4.0//en“>
```

Die Befehle in HTML nennt man *Tags*.

Eingeleitet wird eine Seite mit `<html>`.

Es folgt der sogenannte Kopf mit `<head>`

Die Seite sollte auch einen Namen bekommen. Dieser Name ist neben den *Keywords* ausschlaggebend für Suchmaschinen (Lycos, Altavista, Fireball etc.) Diese Suchmaschinen suchen u. a. nach diesen Namen. Außerdem wird dieser Name in der Titelzeile des verwendeten Browsers angezeigt. Zu dumm, wenn dort steht: *Untitled Document* und nicht gerade rühmlich für den Webmaster.

Also: `<title>Meine erste Seite </title>`

Nanu, was ist denn das? – Zweimal dasselbe? – Nein.

Alle Tags in HTML haben einen Anfang und ein Ende – außer natürlich ein paar rühmlichen Ausnahmen.

Anfangs-Tag: `<...>`

End-Tag: `</...>`

Hier kommen dann auch die *meta-Tags* hin. Nähere Erläuterungen dazu etwas weiter unten.

Außerdem gehört hier *JavaScript-Code* oder auch *CSS-Code* hinein, worauf ich in dieser Kurzanleitung nicht näher eingehen möchte.

Danach wird der *Head*-Teil abgeschlossen mit `</head>`.

Bis jetzt ist Ihre Internet-Seite noch leer. Nichts ist im Browser zu sehen. Das ändert sich erst mit der Eingabe

`<body>`

Ab hier ist es jetzt möglich, Schrift und Bilder auf Ihrer Seite zu plazieren.

Abgeschlossen wird dieser Tag wieder mit

`</body>`

und zum Schluß noch

`</html>`

Zusammengefasst sieht das ganze dann so aus:

```
<!doctype html public „-//w3c//dtd html 4.0//en“>
```

```
<html>
```

```
<head>
```

```
<title> Meine erste Seite </title>
```

```
</head>
```

```
<body>
```

*Hier ist dann was zu sehen.*

```
</body>
```

```
</html>
```

## Weitere Angaben

Wenn man eine Site ins Web stellt, möchte man natürlich auch dass sie angesurft wird. Um sicher zu stellen, dass die Site von Suchmaschinen gefunden wird ist der `<meta>`-Tag verantwortlich. Er findet seinen Platz im Head-Bereich der Seite.

`<meta>` kann verschiedene Attribute aufweisen und verschiedene inhaltliche Informationen aufnehmen.

Attribut / Wert:	Beschreibung
<code>name</code>	Nimmt verschiedene inhaltliche Informationen auf
<code>author= "... "</code>	Der Name des Autors der Seite
<code>description= "... "</code>	aussagefähige Inhaltsangabe
<code>date= "... "</code>	Das Erstellungsdatum der Seite
<code>keywords= "... "</code>	Schlüsselwörter für die Suchmaschinen
<code>audience= "... "</code>	Die Zielgruppe für den Seiteninhalt
<code>robots= "... "</code>	Soll die Suchmaschine diese Seite anzeigen (Einsatz bei Framesets)

### Beispiel:

```
<head>
<meta name= "author" content= "Molle Mustermann">
<meta name= "description" content= "Die Gestaltung einer Webseite in HTML">
<meta name= "keywords" content= "HTML,Dateistruktur,Webseite">
<meta name= "date" content= "2000-03-03">
<meta name= "audiende" content= "HTML-Programmierer">
<meta name= "robots" content= "nofollow">
</head>
```

Hinter „`content=`“ sollten dann ihre eigenen Inhalte stehen.

## Kommentare

Um bei der ganzen Programmiererei den Überblick zu behalten, empfiehlt es sich, sogenannte Kommentare einzubauen. Diese werden vom Internet-Browser nicht interpretiert und sind sozusagen nur für Sie im Seiten-Quelltext lesbar. In diese Kommentare können Sie sich ihre persönlichen Informationen und Hinweise schreiben.

```
<!-- Das ist ein Kommentar-->
```

## Text auf meiner Seite

Alle folgenden Tags und Codes finden ihren Platz zwischen den Tags `<body>` und `</body>`.

Attribut/Wert	Beschreibung
<code>&lt;h1&gt; &lt;/h1&gt;</code>	Überschrift für die bessere Gestaltung Ihrer Website. H1 ist die größte, H6 die kleinste Überschrift
<code>&lt;h2&gt; &lt;/h2&gt;</code>	
<code>&lt;h3&gt; &lt;/h3&gt;</code>	
<code>&lt;h4&gt; &lt;/h4&gt;</code>	
<code>&lt;h5&gt; &lt;/h5&gt;</code>	
<code>&lt;h6&gt; &lt;/h6&gt;</code>	
<code>&lt;br&gt;</code>	erzeugt einen Zeilenumbruch und ist ein Tag <u>ohne</u> End-Tag.
<code>&lt;nobr&gt; und &lt;/nobr&gt;</code>	verhindert einen Zeilenumbruch
<code>&lt;p&gt; und &lt;/p&gt;</code>	erzeugt einen neuen Absatz
<code>&amp;nbsp;</code>	verhindert die Trennung von zusammengehörigen Wörtern (erzwungenes Leerzeichen)
	z. B. Eingabe: Im Alter von 25 &nbsp; Jahren ...
<code>align=</code>	Textausrichtung innerhalb eines Absatzes ( <code>&lt;p&gt;</code> , <code>&lt;/p&gt;</code> ). Nach Absatzende sind die Standardeinstellungen wieder gültig.
<code>left</code>	linksbündig
<code>center</code>	zentriert
<code>right</code>	rechtsbündig

## Beispiel:

`<p align=center>Dieser Satz ist zentriert ausgerichtet.</p>`

`<font> </font>`

Tag zur Schriftbestimmung

<code>face= "... "</code>	Schriftart <code>&lt;font face=Arial"&gt;Schriftart Arial&lt;/font&gt;</code>
<code>size= "... "</code>	Schriftgröße
<code>color= "... "</code>	Schriftfarbe
<code>&lt;em&gt;&lt;/em&gt;</code>	kursiv
<code>&lt;strong&gt;&lt;/strong&gt;</code>	fett
<code>&lt;code&gt;&lt;/code&gt;</code>	Formatierung von Listings
<code>&lt;samp&gt;&lt;/samp&gt;</code>	Formatierung von Beispieltexten
<code>&lt;cite&gt;&lt;/cite&gt;</code>	Formatierung von Zitaten
<code>&lt;big&gt;&lt;/big&gt;</code>	größerer Text
<code>&lt;small&gt;&lt;/small&gt;</code>	kleinerer Text
<code>&lt;sub&gt;&lt;/sub&gt;</code>	hochgestellt
<code>&lt;sup&gt;&lt;/sup&gt;</code>	tiefgestellt
<code>&lt;b&gt;&lt;/b&gt;</code>	fett
<code>&lt;tt&gt;&lt;/tt&gt;</code>	Teletype
<code>&lt;i&gt;&lt;/i&gt;</code>	kursiv
<code>&lt;s&gt;&lt;/s&gt;</code>	durchgestrichen
<code>&lt;pre&gt;&lt;/pre&gt;</code>	stellt Text mit eingegebenen Abständen und Einzügen dar

## Listen

Häufig ist es erforderlich, numerierte Aufzählungen oder Aufzählungen mit einem Punkt davor übersichtlich darzustellen.

Die Liste wird mit dem Befehl `<ol>` eingeleitet und mit `</ol>` abgeschlossen. Der Beginn der Numerierung beginnt standardmäßig mit dem Wert 1, fügt man dem `<ol start=5>` zu, beginnt die Aufzählung mit dem Wert 5.

Innerhalb des Listings können verschiedene Aufzählungsarten definiert werden. Der neue Listeneintrag wird in mit `<li></li>` eingeschlossen.

Attribut/Wert	Beschreibung
<code>type=A</code>	Aufzählung: A,B,C...
<code>type=a</code>	Aufzählung: a,b,c...
<code>type=I</code>	Aufzählung: I,II,III,IV...
<code>type=i</code>	Aufzählung: i,ii,iii,iv

### Beispiel:

```
<body>
<ol>
  <li>Erste Aufzählung</li>
  <li>Zweite Aufzählung</li>
  <li>Dritte Aufzählung</li>
</ol>
</body>
```

Aufzählungen mit einem Aufzählungszeichen werden mit `<ul>` eingeleitet und mit `</ul>` abgeschlossen.

<code>&lt;ul type="circle"&gt;</code>	○
<code>&lt;ul type="square"&gt;</code>	■
<code>&lt;ul type="disc"&gt;</code>	●

Haben Ihre Listen zusätzlich noch untergeordnete Listeneinträge brauchen Sie noch folgende Tags:

Attribut/Wert	Beschreibung
<code>&lt;d&gt;&lt;/d&gt;</code>	Erstellt eine Definitionsliste
<code>&lt;dt&gt;&lt;/dt&gt;</code>	Erstellt einen Listeneintrag
<code>&lt;dd&gt;&lt;/dd&gt;</code>	Erläuterung zu einem Listeneintrag

Listen können ineinander verschachtelt werden. Es wird innerhalb einer Liste lediglich eine weitere definiert.

Aber halt – bevor Sie das erste Wort schreiben - schreiben sie es so, dass es alle lesen können. Nicht nur die Browser interpretieren Text verschieden, sondern es gibt auch Surfer die andere Betriebssysteme außer Windows 95/98 verwenden, wie Windows NT, UNIX, OS/2, Linux, Solaris oder MAC, welche nicht die gleichen Zeichensätze verwenden.

Umlaute und Sonderzeichen müssen in HTML daher gesondert codiert werden.

```

ä = &auml;
Ä = &Auml;
ö = &ouml;
Ö = &Ouml;
ü = &uuml;
Ü = &Uuml;
ß = &szlig;

```

Schreiben Sie also anstelle von z. B. „Häusern“ → `H&auml;usern`.

## Jetzt wird's bunt

Die Farbdarstellung von Schrift oder für den Seitenhintergrund ist in HTML recht kompliziert - aber keine Angst - so schwer ist es auch wieder nicht.

Grundsätzlich gibt es zwei Wege, Farben in HTML darzustellen.

Zum ersten kann man Farben mit einem Ausdruck darstellen. Dies beschränkt sich jedoch nur auf eine geringe Anzahl (16) von Farben und das anfangs erwähnte Problem der unterschiedlichen Interpretation durch die verschiedenen Browser ist hier ebenfalls zu bedenken.

Die Zweite und sicherere Lösung ist die Angabe des Farbwertes in hexadezimaler Form. Die etwas schwierigere Art entschädigt durch die mögliche Darstellung von 16,7 Mio Farben auf diesem Weg.

Grundlage bildet das sogenannte RGB-Modell. Alle Farben werden hier aus Rot, Grün und Blau zusammengesetzt. Um dies zu erreichen werden die Farben in unterschiedlichen Farbanteilen von 0 = nicht vorhanden bis 255 = höchster Farbanteil (entspricht der hexadezimalen Zahl FF) dargestellt.

Die folgende Tabelle versucht zu veranschaulichen, wie die einzelnen Farbwerte zusammengesetzt werden.

	Rotanteil	Grünanteil	Blauanteil
Hexadezimaler Wert	FF	00	80
RGB-Farbanteile	255	0	0
Farbdefinition	ff0080 (mehr Rot geht nicht)		

Hexadezimale Umrechnungstabelle:

Dec	0	1	2	3	4	5	6	7
Hex	0	1	2	3	4	5	6	7

Dec	8	9	10	11	12	13	14	15
Hex	8	9	A	B	C	D	E	F

Die 16 Grundfarben mit ihren Ausdrücken und hexadezimalen Farbwerten:

Farbe	Hexwert	Farbe	Hexwert
black	#000000	green	#008000
silver	#c0c0c0	lime	#00ff00
gray	#808080	olive	#808000
white	#ffffff	yellow	#ffff00
maroon	#800000	navy	#000080
red	#ff0000	blue	#0000ff
purple	#800080	teal	#008080
fuchsia	#ff00ff	aqua	#00ffff

Das Zuweisen einer Farbe erfolgt in HTML mit der Übergabe des Farbwertes an ein entsprechendes Attribut.

Dabei können die Attribute sein:

<code>&lt;body bgcolor= "#ff00ff"&gt;</code>	Hier ist der Hintergrund der gesamten Seite fuchsiarben
<code>&lt;font color= "#800000"&gt;</code>	Hier ist die Schrift braun.
<code>&lt;hr color= "blue"&gt;</code>	Der Trennstrich ist blau. (nur Internet-Explorer)

## Der Trennstrich

Der Trennstrich zur optischen Gestaltung Ihrer Website wird mit `<hr>` erzeugt. Dieses Tag braucht keinen End-Tag.

```
<hr width="250", size="3", color="blue" noshade="noshade"/>
```

Dieser Trennstrich hat nicht nur eine Farbe (blau), sondern auch eine bestimmte Länge (width) und Breite (size) und wirft keinen Schatten.

## Seitenhintergrund

Im vorhergehenden Abschnitt haben Sie bereits gelernt, Ihrer Homepage eine bestimmte Hintergrundfarbe zuzuweisen.

Die Farbzuzuweisung erfolgt im Body-Tag, also: `<body color= "#ff00ff">`

Das Einfügen eines Hintergrundbildes hat nichts gemein mit dem Einfügen eines Icons, Clipart oder Fotos auf Ihrer Homepage.

Im Standardmodus wird das Hintergrundbild so oft auf Ihrer Website eingefügt, bis das gesamte Browserfenster damit ausgefüllt wird. Dies ist je nach Bildschirmauflösung der einzelnen Surfer, die Ihre Site im Internet aufsuchen, unterschiedlich. Wählen Sie daher ein kleines Bild im .GIF oder .JPG-Format und achten Sie auf die Dateigröße. Je mehr KB's von Ihrem Webserver zu demjenigen, der Ihre Website aufruft, geschaufelt werden müssen, desto langsamer ist auch der Seitenaufbau – was beim Surfer nicht gerade Freude aufkommen lässt.

Das Einfügen des Hintergrundbildes erfolgt mit: `<body background= "higru.gif">`  
Dabei wird *higru.gif* durch Ihre eigene Bilddatei ersetzt.

Haben Sie ein dunkles Hintergrundbild eingefügt, kann es sein, dass die Standardtextfarbe schwarz nicht mehr so gut lesbar ist.

Möchten Sie auf Ihrer Seite die Textfarbe generell ändern, können Sie sich das Getippe in jedem Absatz sparen und im Body-Tag mit:

```
<body text= "#ffffff">
```

die Textfarbe generell hier z. B. auf *weiss* setzen.

Dies geht natürlich auch mit einem CSS-Befehl, worauf ich nicht näher eingehen möchte, da dies den Rahmen dieser kleinen HTML-Einführung bei weitem sprengen würde.

## Bilder über Bilder

Am Anfang dieses Kapitels möchte ich zu aller erst etwas über die Gestaltung sagen. Dass ein langsames erscheinen von Internet-Seiten im Browser keine Freude beim Surfer aufkommen lässt, habe ich bereits schon bemerkt. Deshalb – wählen Sie bei Fotos stets das .JPG-Format. Jonglieren Sie ein wenig bei der Komprimierung des Bildes. Wägen Sie ab zwischen Qualität des Bildes und Dateigröße!

Entscheiden Sie sich bei Grafiken, wie z. B. Buttons oder Cliparts für das .GIF-Format. Wählen Sie stets das GIF-Format für Grafiken mit klaren Konturen und wenigen klaren Farben. Das GIF-Format beherrscht ausschließlich die 8-Bit-Palette mit einer Darstellung von höchstens 256 Farben. Mittels eines geeigneten Bildbearbeitungsprogrammes speziell fürs Web, wie z. B. Macromedia Fireworks können Sie die GIF-Bilder auch in ihrer Dateigröße etwas beeinflussen, indem Sie die Farbanzahl herabsetzen. Das GIF-Format bietet

auch den Vorteil, dass Sie Farben „transparent“ setzen können. Was so viel heisst, dass z. B. ein runder Button keinen rechteckigen weißen Rahmen bei seiner Darstellung aufweist.

Neuere Browser können mit dem noch relativ jungen und noch weniger verbreiteten Bildformat PNG (Portable Network Graphics) etwas anfangen. Der Vorzug dieses Bildformates gegenüber JPG oder GIF ist eine höhere Bildqualität gepaart mit geringerer Dateigröße.

So, wir haben uns jetzt beispielsweise für ein eingescanntes Foto namens „*auto.jpg*“ entschieden und fügen es mit dem Befehl:

```
<img src= "auto.jpg">
```

ein. Fertig.

Möchten Sie nicht, dass das Foto mit einem Rahmen dargestellt wird, geben Sie zum o. g. Befehl einfach einen weiteren hinzu.

```
<img src= "auto.jpg" border= "0">
```

Die Standardeinstellung ist im Grunde immer ohne Rahmen, ausser – es handelt sich um eine .GIF-Grafik, welche mit einem LINK (Dazu kommen wir gleich.) hinterlegt ist – hier ist es immer ratsam *border= "0"* mit anzugeben, da die Grafik (z. B. ein Pfeil) sonst automatisch mit einem hässlichen Rahmen versehen wird.

So eine feine Sache auch Bilder auf der Homepage sind, sie bremsen die Ladezeit im Browser und deshalb stellen einige Surfer das Anzeigen von Bildern in Ihren Browsern ab und es erscheint nur ein Platzhalter anstelle des Bilder. Um den Surver mitzuteilen, was das Bild darstellt fügen wir einen sogenannten ALT-Tag ein (die *alternative* Textanzeige anstelle des Bildes). Hier bei unserem Beispiel:

```
<img src= "auto.jpg" alt= "Auto" border= "0">
```

## Links und Links

Ein Link ist eine Verbindung zu einer anderen Internet-Seite oder einer anderen Seite Ihrer eigenen Homepage oder einer bestimmten Stelle auf der aufgerufenen Seite. Das Anklicken des LINKs mit der Maus ist ähnlich zu verstehen wie das Umblättern einer Buchseite.

Treffen Sie im Internet auf einen Link, verwandelt sich der Mauszeiger in eine zeigende Hand.

Und so erzeugen Sie einen Link:

```
<a href= "zweite.htm">Zur zweiten Seite</a>
```

<a heißt auf englisch anchor (Anker) und leitet den Link ein, *href* sagt dem Browser die Adresse, die er anzeigen soll – hier zu unseren Zweiten Seite unserer Homepage.

Der Browser stellt den Link im Standardmodus blau unterstrichen dar: [Zur zweiten Seite](#)

Wenn Sie Verweise auf andere Seiten im Internet erzeugen wollen, geben Sie stets die Vollständige URL (Internet-Adresse) ein, also z. B.:

```
<a href= "http://www.gikom-nis.de">Gikom</a>
```

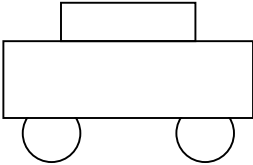
Im Grunde können Sie mit den vorgestellten Befehlen bereits eine Internetseite erstellen. Da Sie jedoch sicher auf eine optisch ansprechende Gestaltung nicht verzichten möchten, kommen Sie nicht darum, gestalterische Hilfsmittel wie Tabellen oder Frames einzusetzen. Zu diesen Optionen kommen wir jetzt.

## Tabellen

Vielleicht haben Sie schon einmal in den Programmen wie Word oder Excel Tabellen erstellt, um Texte und Grafiken auszurichten. Ähnlich ist es in der Programmierung in HTML. Tabellen und ihre vielfältigen Optionen dienen dazu, die Internetseite optisch ansprechend und übersichtlich zu gestalten.

Um Ihnen die Arbeit mit Tabellen näher zu veranschaulichen, möchte ich hier mit einem Beispiel arbeiten.

## Meine erste Seite

<p>Auto</p> <p>(Das ist unser Bild)</p> 	<p>Links sehen Sie eine Abbildung des neuesten Modells.</p>
---	---

Wir gestalten hier eine Internet-Seite mit einer Überschrift *Meine erste Seite* in der Größe der Überschrift `</h2>` und einer Tabelle mit zwei Spalten, einem Bild *-auto.gif-* links und einem erläuterndem Text auf der rechten Seite.

Unsere Seite beginnt also wie folgt:

```
<!doctype html public „-//w3c//dtd html 4.0//en“>
<html>
<head>
<title> Meine erste Seite in HTML </title>
</head>
<body>
<h2>Meine erste Seite</h2>
```

Jetzt fügen wir unsere Tabelle ein.

```
<table border = 0>
```

leitet die Tabelle ein, welche keinen Rahmen erhalten soll. Die einzelnen Zellen der Tabelle erhalten keine dünnen Linien als Umrahmung und die Tabelle bleibt für den Surfer unsichtbar.

```
<tr>
```

Wir erzeugen eine Zeile, die bis zum Abschluß `</tr>` in Spalten aufgeteilt wird.

```
<td><img src= "auto.jpg" alt= "Auto" border = "0" ></td>
```

In die Erste Spalte kommt das Bild

```
<td>Links sehen Sie eine Abbildung des neuesten Modelles</td>
```

In die zweite Spalte kommt der erläuternde Text.

```
</tr>
```

Hier schließen wir die Zeile ab.

```
</table>
```

Wir beenden die Tabelle.

```
</body>
```

Wir beenden den Body-Tag...

```
</html>
```

und auch die HTML-Seite.

Speichern Sie die Seite als *start.htm* in einem neuen Verzeichnis, in dem sie ab jetzt alle html-Dateien und Bilder Ihrer Homepage abspeichern. Vergessen Sie auch nicht, das Bild *auto.gif* in diesem Verzeichnis zu speichern.

Betrachten Sie Ihre Seite im Browser - die Tabelle bleibt für den Betrachter unsichtbar.

In Tabellen können Sie grundsätzlich alles unterbringen, Sie müssen nur beachten, dass Sie alle Formatierungen, ob für Bilder oder für Text in jeder Zelle vornehmen.

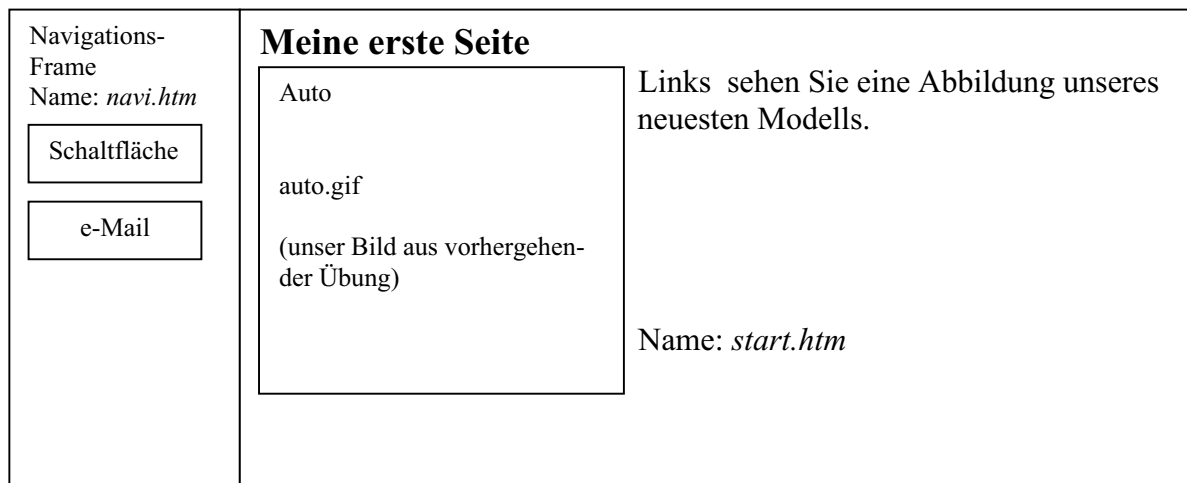


## Einsatz von Frames

Sicher ist es Ihnen beim Surfen im Internet bereits aufgefallen, dass wenn man links auf der Seite einen Link anklickt, sich nur der Inhalt auf der rechten Seite ändert. Das Banner oben und die linke Seite bleiben davon unberührt.

Diesen Effekt erreichen Sie durch Einsatz von sogenannten *Framesets*. *Framesets* setzen sich aus verschiedenen *Frames* (Rahmen) zusammen. Der Browser stellt die Internet-Seite nicht als einzelne Seite dar, sondern zeigt verschiedene Seiten in eigenen Rahmen gleichzeitig an. Am einfachsten erklärt sich das anhand eines Beispiels.

Wir möchten folgende Internet-Seite mit Hilfe von *Frames* darstellen:



Zuerst erstellen wir das sogenannte *Frameset*. Eröffnen Sie, wie eine ganz normale Seite. Beachten Sie jedoch, dass die Angaben der Tags `<frameset>` und `</frameset>` im `<head>`-Teil der Seite stehen müssen und nicht im `<body>`-Tag.

```
<!doctype html public "-//w3c//dtd html 4.0//en">
<html>
<head>
<title>Frameset</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
</head>
```

Wir nennen unsere Seite sinnigerweise *Frameset*.

Legen Sie die Rahmen fest:

Der linke Rahmen soll *navi* heißen (Ich könnte ihn auch *Manfred* oder *linkeseite* nennen) und hier z. B. 120 Pixel breit sein, der \* bedeutet, dass der Rest der Seite dem Frame „*inhalt*“ zur Verfügung gestellt wird. Der Rahmen des linken Frames soll nicht sichtbar sein und es soll keine Scrollbalken geben, da wir ja nur 2 Schaltflächen auf dieser Seite haben.

In diesem Frame soll die Seite *navi.htm* ihren Platz finden, die wir noch erstellen müssen.

```
<frameset cols="120,*" frameborder="NO" border="0" framespacing="0">
<frame name="navi" scrolling="NO" src="navi.htm">
```

Nun legen wir den Hauptframe an. Er soll den Namen *inhalt* bekommen und unsere bereits erstellte Seite *start.htm* aufnehmen.

```
<frame name="inhalt" src="start.htm">
```

Jetzt müssen wir das *Frameset* noch schließen.

```
</frameset>
```

Wir müssen Nutzern älterer Browser, die keine *Frames* darstellen können, dies auch mitteilen.

Das übernehmen die beiden Tags: `<noframes>` und `</noframes>`. Dazwischen Schreiben wir den `body`-Tag, setzen die Hintergrundfarbe standardmäßig auf weiß und schreiben: *Ihr Browser kann leider keine Frames darstellen.*, beenden den `body`-Tag und unsere `html`-Seite.– fertig.

```
<noframes><body bgcolor="#FFFFFF"> (Hintergrundfarbe: weiss)
Ihr Browser kann leider keine Frames darstellen.
</body></noframes>
</html>
```

Speichern Sie das *Frameset* als Ihre Startseite *index.htm* ab.

## Navigationsleiste

Wir haben unsere Navigationsleiste noch nicht erstellt. Es ist genau so eine Seite wie alle anderen, nur dass sie nur den Platz in unserem Frame "navi" auf der linken Browserseite zur Verfügung hat. Das holen wir jetzt nach und eröffnen unsere Seite wieder mit den bekannten Tags,

```
<!doctype html public "-//w3c//dtd html 4.0//en">
<html>
<head>
<title>Navigationsleiste</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
</head>
```

legen die Hintergrundfarbe, hier: weiss, fest,

```
<body bgcolor="#FFFFFF">
```

Um unsere Schaltflächen genau zu positionieren, bedienen wir uns wieder einer „unsichtbaren“ Tabelle mit zwei Reihen und in einer Spalte.

Sie soll 100 Pixel (*width="100"*) breit sein, keine Rahmen haben (*border="0"*) und die Zellen sollen zum inneren Rand der Tabelle einen Abstand von 5 Pixeln haben (*cellspacing="5"*), um die Tabelle etwas aufzuloockern. Außerdem soll die Tabelle einen vertikalen Abstand zu anderen Objekten (Text oder Bilder) von 20 Pixeln aufweisen (*vspace="20"*).

```
<table width="100" border="0" cellspacing="5" vspace="20">
<tr>
```

In die obere Zelle kommt unsere Schaltfläche *sf.gif*. Diese Schaltfläche soll einen *Link* auf unsere Zweite Seite (*zweite.htm*) enthalten und dieser Seite sagen, dass sie in unserem Frame-Fenster *inhalt* zu erscheinen hat.

Das sieht im Quelltext dann so aus:

```
<td><a href="zweite.htm" target="inhalt"></a></td>
</tr>
<tr>
```

Beim Anklicken der unteren Schaltfläche *email.gif* soll das E-Mail-Programm aufgerufen werden, damit unser Surfer uns eine Mail schicken kann. Die Empfänger-Adresse (unsere E-Mail-Adresse) soll bereits im E-Mailprogramm in dem Feld [AN:] stehen.

Das macht der Befehl: *mailto:* mit nachfolgender E-Mail-Adresse des Empfängers (Das sind wir.)

```
<td><a href="mailto:e-mail@t-online.de"></a></td>
</tr>
```

Wir beenden mit den abschließenden Befehlen

```
</table>
```

unsere Tabelle

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

und unsere Navigationsleiste bzw. HTML-Seite

Speichern Sie diese Seite als *navi.htm* in dem Verzeichnis für Ihre Homepage.

Zu guter Letzt fehlt uns noch unsere Zweite Seite (*zweite.htm*), auf die unsere erste Schaltfläche verweist. Den Anfang kennen Sie schon...

```
<!doctype html public "-//w3c//dtd html 4.0//en">
<html>
<head>
<title>Zweite Seite</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
</head>
```

Diesmal verwenden wir sogar eine Hintergrundgrafik mit dem Befehl `background=` gefolgt vom Namen unserer Hintergrundgrafik `gelbhint.gif` in Anführungszeichen. Da das Hintergrundbild eine strukturierte gelbe Optik hat und es Surfer gibt, die das Anzeigen aller Bilder in ihren Browsern abgestellt haben, ist es günstig hier auch eine Hintergrundfarbe in hellgelb (`#FFFFCC`) zuzuweisen.

```
<body bgcolor="#FFFFCC" background="gelbhint.gif">
```

Wir definieren eine Überschrift in Größe 3 mit der Schriftart *Arial* (Alternativ sollte man noch ähnliche Schriftarten mit angeben, falls auf dem Computer, auf dem unsere Seite angesurft wird, diese Schriftart nicht installiert ist.). Unsere Überschrift soll rot (`#CC0033`) aussehen und zentriert dargestellt sein: *Meine zweite Seite*.

```
<h3 align="center"><font face="Arial, Helvetica, sans-serif" color="#CC0033">Meine  
zweite Seite </font></h3>
```

Mit Tabellen kann man ohne größeren Aufwand sehr schöne Effekte erzielen.

Wir erzeugen eine Tabelle, die 100 Pixel hoch sein soll und die 50 % der Internet-Seite einnehmen soll. Sie hat einen Rand von 5 Pixeln und wird durch `cellspacing` und `cellpadding` mit jeweils 5 Pixeln aufgelockert.

Der helle Rahmen (`bordercolorlight=`) soll hellblau (`#CCCCFF`), der dunkle Rahmen (`bordercolordark=`) soll dunkelblau (`#3333FF`) sein. Dem Tabellenhintergrund (`bgcolor=`) soll hellgrün werden (`#CCFFCC`).

Die Tabelle soll zu anderen Objekten (Texte oder Bilder) keinen Abstand halten und auf der Seitenmitte zentriert dargestellt werden:

```
<table width="50%" border="5" cellspacing="5" cellpadding="5" height="100" bordercolorlight="#CCCCFF" bordercolordark="#000099" bordercolor="#3333FF" bgcolor="#CCFFCC" vspace="0" hspace="0" align="center">  
<tr>
```

In die erste Zelle geben wir folgenden Text ein: *Mit Tabellen kann man eine Seite sehr schön gestalten*.

Beachten Sie die Umschreibung der Umlaute bei „schön“= `sch &ouml;`; `n`

```
<td>Mit Tabellen kann man eine Seite sehr sch&ouml;n gestalten.</td>
```

In der zweiten Zelle soll der Text wieder in *Arial* geschrieben sein, die Schriftgröße 2 erhalten und in der Zelle mittig ausgerichtet sein.

```
<td align="center" valign="middle">  
<p><font face="Arial, Helvetica, sans-serif" size="2">Die Formatierung bei Text und Tabellen ist das A und O. Probieren Sie verschiedene Formatierungen aus.</font></p>  
</td>  
</tr>  
</table>
```

Wir beenden wieder die Tabelle und bieten unserem Surfer eine Schaltfläche, damit er wieder zurück zur vorhergehenden Seite in unserem Frame *inhalt* gelangt...

```
<a href="start.htm" target="inhalt"></a>
```

```
</body>
```

und beenden die Internet-Seite.

```
</html>
```

## **Herzlichen Glückwunsch!**

Sie haben Ihren ersten Internet-Auftritt in HTML erstellt.

Für die ersten Schritte sollte dies hier erst einmal reichen.

Für weitere und umfassendere Informationen über HTML, JavaScript, CSS und andere Programmiersprachen gibt es auf dem Markt ausreichend Literatur. Auch einschlägige Zeitschriften, wie z. B. die **HOMEPA-GE** oder **INTERNET Professionell** leisten gute Dienste und auf den beiliegenden Heft-CD's befinden sich oft fertige Scripte oder Java-Applets zur freien Verfügung, die Sie in Ihre Homepage einbinden können. Außerdem ist die html-programmierte HTML-Referenz „SELFHTML“ von Stefan Münz, was stets aktualisiert wird, dort ebenso zu finden.

## Programmierhilfen

Keiner würde heute noch auf die Idee kommen, seine Internet-Seiten mit dem Windows mitgelieferten Editor zu programmieren – der Programmieraufwand wäre dem Ergebnis gegenüber unverhältnismäßig hoch.

Der Markt wird überschwemmt von Authoring-Tools zur Programmierung von Internet-Seiten. Das Reicht von

- reinen Text-Editoren, die den Programmierer mit farbiger *Tag-Darstellung* und ähnlicher Programmiererleichterungen unterstützen (z. B. 1<sup>st</sup> Page 2000 2.0 [Freeware, Englisch], HotDog Pro 6.0 [Shareware, Englisch])

über

- Text-Editoren, die eine WYSIWYG-Vorschau integriert haben (z. B. HotMetal 6.0, Almighty Notepad 2000 2.01 [Shareware, Deutsch] oder NetObjects Fusion), in denen man seine geistigen Programmierergüsse sofort überprüfen kann.

zu

- echten Multimedia-Anwendungen (z. B. Adobe GoLive oder Macromedia Dreamweaver), die dem Anwender jedoch trotzdem noch erlauben, direkte Eingriffe in den HTML-Code vorzunehmen und allenfalls eine Fehlermeldung (je nach Version) fabrizieren.

Dies ist besonders interessant, wenn das Authoring-System kein Javascript, DHTML, XML, CGI, CSS, oder XHTML verstehen. Dies kann man dann einfach per Hand in den Quellcode eintragen.

Im Zeitalter, in dem auch immer mehr Vektororientierte bewegte Bilder (Flash-Animationen) im Web auftauchen wird auch der Funktionsumfang der Multimedia-Anwendungen immer ausgereifter – aber auch immer schwieriger zu bedienen und werden auch in Zukunft wohl nur echten Web-Designern vorbehalten bleiben (z. B. Adobe Live Motion 1.0, Macromedia Flash 4.0, Macromedia Director 8 Shockwave Studio).

Außerdem stellt eine finanzielle Ausgabe von mehreren Hundert bis mehreren Tausend Mark für die erforderliche Software wohl kaum eine Relation gegenüber der Pflege der eigenen kleinen Homepage, in der man als Privatperson allenfalls eine Bildershow einbaut oder Interessantes über die Katzenpflege zu berichten hat, dar.

bis zu

- reinen WYSIWYG-Anwendungen, die keine Möglichkeit zum Eingriff in den Quellcode bieten (z. B. *Sierra Web Artist* oder *Meine Homepage* von DATA BECKER.) Derartige Websites fallen meist etwas spartanisch aus. Wer aber unbedingt ohne Programmierkenntnisse auskommen möchte, eine willkommene Alternative.

Da ich mir bis an diese Stelle nicht umsonst die Arbeit gemacht haben möchte, Sie ein wenig in die Programmierung in HTML einzuführen, kommen die zuletzt genannten Anwendungen für Sie nun nicht mehr in Frage!

### Welcher Editor soll's nun sein?

Natürlich ist es jedem selbst überlassen, mit welchem HTML-Werkzeug er arbeitet. Aus eigener Erfahrung kann ich jedoch sagen, dass man sich erst einmal mit *Hot Metal* die Tiefgründe des HTML erarbeitet, um erst einmal zu erfahren, was man da eigentlich tut.

Hat man die Grundzüge erst einmal verstanden, rate ich zur Arbeit mit *Macromedia Dreamweaver 2.0* oder der neuesten Version 3.0 (Kostenpunkt ca. 350 DM).

Adobe – Fans könnten mich jetzt verteufeln und auf jeden Fall *Adobe GoLive* als Multimedia-Anwendung empfehlen.

Der Grund für die Entscheidung für Macromedias *Dreamweaver* ist, das Macromedia bereits in der 2-er Version die Integration der Produkte *Fireworks* (Bildbearbeitung und Vektorgrafik fürs Internet) und *Flash* (Vektor-Animationen) anbietet.

Bei Adobe GoLive ist dieses Zusammenspiel zwischen Internet-Bildbearbeitung und HTML-Editor wohl erst in der Version 5.0 zufriedenstellend gelöst.

## Webspace und Upload

Um eine ansehnliche Website ins Netz zu stellen, benötigen Sie eine Upload-Hilfe, sogenannte FTP-Clients. Hier empfehle ich *WS-FTP*, *CuteFTP* oder *LeechFTP*. Diese Programme sind von der Bedienung her fast selbsterklärend.

Außerdem brauchen Sie etwas Festplattenplatz (sogenannten Webspace) auf einem Webserver, den Sie entweder kostenlos bei entsprechenden Anbietern oder Ihrem Provider (z. B. AOL, T-online, Compuserve, World Online) oder ganz professionell mit eigener Domain wie z. B. *www.Ihr-name.de* (z. B. 1 & 1 Puretec, LOOMES, Strato) zu erhalten ist. Dies ist heute auch schon erschwinglich und kostet im Monat ab 0,49 DM (je nach Anbieter) zuzüglich der einmaligen Einrichtungspauschale und Anmeldung bei der DENIC von ca. 40 DM.

## Nachwort

Das Internet hält nicht an. Erweitern Sie Ihr Wissen über Programmierung. Das W3C-Consortium tendiert dahin, z. B. CSS bei der Textformatierung zum Standard zu machen, so dass die sogenannten *font*-Tags zukünftig wohl mehr und mehr in Vergessenheit geraten werden. Außerdem hält die Einbindung von *CSS-Befehlen* anstelle von *font*-Tags Ihren Quellcode schlank, was sich wieder positiv auf die Ladezeit Ihrer Site auswirkt.

Am Anfang unserer kleinen Einführung in HTML habe ich gesagt, dass Internet-Dateien stets mit der Dateiendung *.htm* oder *.html* enden.

Der Vollständigkeit halber füge ich hier jedoch an, dass mit der Einführung von DHTML, XML, XHTML, reinen Flash-Seiten, Shop-Seiten o. ä. dies auch anders sein kann. Beobachten Sie dies bei Ihrem nächsten Surfgang im Internet.

`<p><font face="Arial" size="2"><b>`  
**Viel Spass nun bei der Erstellung Ihrer ersten Homepage und beim weiteren Dazulernen.**  
`</b></font></p>`

Bleiben Sie am Ball, wenn Sie ansprechende Seiten gestalten wollen. Wenn Sie aufhören, auf diesem Gebiet hinzuzulernen, werden die von Ihnen erstellten Internet-Seiten, seien die Inhalte noch so interessant und informativ, nicht mehr angesurft, da ihre Site vielleicht veraltet programmiert wurde und von den Browsern in zwei oder drei Jahren nicht mehr entsprechend dargestellt werden kann. Große Schritte in die Richtung XHTML und XML werden zur Zeit getan.

Sollte sich der XML-Standard durchsetzen, gewöhnen Sie sich lieber gleich daran, alle Tags abzuschließen (auch `<hr>` mit `</hr>` und `<br>` mit `</br>` zum Beispiel), da dies Ihnen den Umstieg erleichtern wird. Übrigens verstehen die meisten Browser diese „neuen“ End-Tags bereits heute.

Ich möchte Ihnen an dieser Stelle keine Angst machen. Aber wenn Sie die Zeit vom Beginn der Zurverfügung-Stellung des Internets für die breite Masse bis heute beobachten, müssen Sie doch sicher auch zugeben, dass der Fortschritt ziemlich rasant vonstatten ging.

*Weiß man was die schnelle Entwicklung der Technik und die Möglichkeiten des Internets in Zukunft noch bringen werden und wie schnell das geht?*

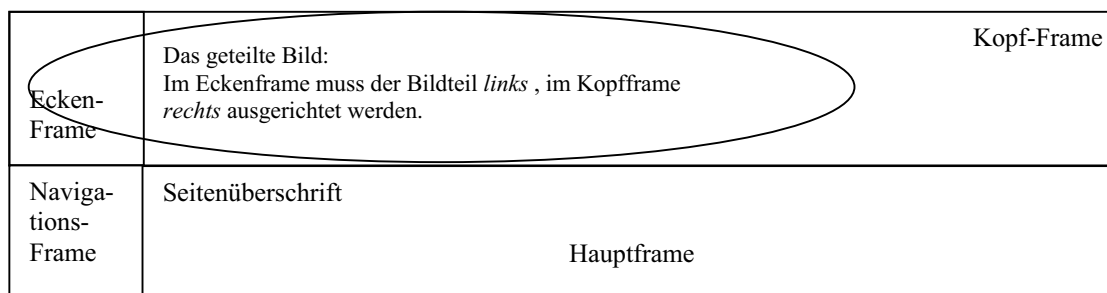


## Interessante Tipps

### Arbeiten mit Framesets und geteilten Bildern

Wenn Sie auf Ihrer Site Framesets zur Gestaltung einsetzen und damit geteilte Bilder genau positionieren möchten, teilen Sie den Browsern dies mit. *Left Margin* und *Top Margin* als auch *Margin Left* und *Margin Right* sollten dann auf „0“ gesetzt werden. Als Standardeinstellung hält eine Internetseite immer einen kleinen „Sicherheitsabstand“ zum Browserrand – und der, wie kann es anders sein – ist bei allen Browsern auch abhängig von der eingestellten Auflösungen, unterschiedlich. Sagen Sie auch stets *Border=“0“*.

Zur Veranschaulichung folgende Skizze:



Wählen Sie die Breite der Frames der Größe und Breite des Bildes angepaßt inklusiv der Befehle, um keine Rahmen zu erzeugen und das Scrollen zu vermeiden. Dann dürfte das geteilte Bild in allen Browsern ordnungsgemäß dargestellt werden.

Das Frameset für oben abgebildetes Beispiel könnte dann so aussehen:

```

<!doctype html public "-//w3c//dtd html 4.0/en">
<html>
<head>
<title>Untitled Document</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
</head>
<frameset rows="80,*" cols="80,*" frameborder="NO" border="0" framespacing="0">
  <frame name="Ecke" scrolling="NO" noresize src="ecke.htm" marginwidth="0" marginheight="0" framebor-
  der="NO">
  <frame name="oben" scrolling="NO" noresize src="kopf.htm" marginwidth="0" marginheight="0" framebor-
  der="NO">
  <frame name="navi" scrolling="NO" noresize src="navi.htm" marginwidth="0" marginheight="0" framebor-
  der="NO">
  <frame name="inhalt" src="start.htm" frameborder="NO">
</frameset>
<noframes><body bgcolor="#FFFFFF">
Ihr Browser kann leider keine Frames darstellen.
</body></noframes>
</html>

```

### Einfacheres Arbeiten mit einem CSS-Script

Obwohl ich CSS-Styles in dieser kurzen Anleitung nicht behandeln wollte, möchte ich Ihnen mit Hilfe von CSS eine kleine Arbeitserleichterung geben.

Beim Schreiben von längeren Texten ist es zwischen den Tags `<p>` und `</p>` stets erneut notwendig die Textformate anzugeben. Dies ist auf die Dauer nicht nur viel Arbeit, sondern bläht den Quellcode unnötig auf. Eine einfache, aber wirkungsvolle Alternative ist der Einsatz z. B. folgenden CSS-Scriptes im *head*-Teil der Seite, um den Eingaben zwischen `<p>` und `</p>` auf dieser Seite generell ein bestimmtes Format zuzuweisen.

## Das Ergebnis:

Die Schriftart (Hier: Arial – alternativ Helvetica oder Sans-Serif), die Schriftgröße (Hier: 12 pt), der Schriftstil (Hier: normal) und die Schriftfarbe (Hier: blau) wird auf dieser Seite für den Tag `<p></p>` fest vergeben. Überall, wo jetzt auf der Seite die Tags auftauchen, auch in Tabellen, werden diese Werte zugewiesen und müssen nicht bei jedem Absatz neu eingegeben werden.

Tippen Sie dieses Script in Ihren HTML-Editor:  
Der eigentliche CSS-Befehl ist hier **fett** dargestellt.

```
<html>
<head>
<title>Untitled Document</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
<style type="text/css">
<!--
p { font-family: Arial, Helvetica, sans-serif; font-size: 12pt; font-style: normal; color: #6666FF}
-->
</style>
</head>
<body bgcolor="#FFFFFF">
<p>Das ist ein Probetext mit CSS</p>
<table width="50%" border="3" bordercolor="#FF3333" height="100">
<tr>
<td bordercolor="#0000FF" width="50%">
<p>Auch in Tabellen funktioniert der Trick mit CSS, wenn man die &lt;p&gt;-Tags
nicht vergisst</p>
</td>
<td bordercolor="#FFFF00" width="50%">Schreibe ich allerdings einfachen Text,
erscheint dieser im Standard-Format</td>
</tr>
</table>
</body>
</html>
```

Schauen Sie sich diese Seite, die Sie vielleicht *css.htm* nennen in Ihrem Browser an und Sie werden staunen, wie man auch noch Tabellen aufpeppen kann.

■ Bedenken Sie auf jeden Fall, wenn Sie CSS-Scripte benutzen, dass diese durch ältere Browser nicht angezeigt werden können.

### **Kleiner Datums-Trick mit Java-Script**

Wenn Sie auf Ihre Seite ein Aktualisierungs-Datum geschrieben haben und diese ins Internet gestellt haben, sind Sie erst einmal froh und glücklich und denken nicht gleich an Aktualisierungen, wenn es nichts neues zu berichten gibt – das kann sich Monate hinziehen-. Wie wäre es da mit einem kleinen Script, das dem Surfer vorgaukelt, dass diese Seite zuletzt am selben Tag aktualisiert wurde, wo der Surfer darauf stösst?

Tragen Sie unten auf Ihrer Seite im *body-Tag* folgendes Java-Script ein:

```
<script language="JavaScript">
document.write("letzte Aktualisierung: " + document.lastModified);
</script>
```

Surfen Sie Ihre Seite am nächsten Tag im Internet an und schau Sie auf das Aktualisierungs-Datum.  
- Überraschung! -



## Noch ein kleiner Hinweis am Rande

- Verwenden Sie nur freie JPG- oder GIF- Dateien auf Ihrer Seite, um nicht rechtliche Konsequenzen tragen zu müssen.
- Stellen Sie eigene Fotos oder GIF-Dateien ins Netz, sichern Sie Ihre Seiten mit einem Copyright-Eintrag (©) ganz unten auf Ihrer Seite, um Ihre Bilder und Inhalte vor Fremdgebrauch zu schützen. Viel kostspieliger und aufwendiger ist das Versehen Ihrer Bilder mit einem Wasserzeichen.

Sicher haben Sie das auch schon gemacht:

Ein schönes Bild auf einer Internet-Seite sehen, linke Maustaste, Bild speichern unter...

Mit dem o. g. Eintrag wirken Sie diesem Tun etwas entgegen.

- Beantragen Sie eine *Domain* für Ihre Web-Präsenz, überprüfen Sie Ihren Wunschnamen für Ihre Internet-Präsenz, ob er nicht schon vorhanden ist. Falls dies der Fall ist, sitzen Sie viel zu schnell und auf Grund Unkenntnis auf einem unüberwindbaren Schuldenberg. Überprüfungsmöglichkeiten finden Sie bei vielen Suchmaschinen, wie z. B. bei *Altavista* oder *Fireball*. Bei *Fireball* finden Sie außerdem einen hervorragenden *Meta-Tag-Generator*, bei dem Sie einfach Ihre Suchbegriffe und Beschreibung für Ihre Homepage eingeben und den Code, den die Suchmaschine dann ausspuckt einfach über die Zwischenablage von Windows kopieren und in den `<head>`-Teil Ihrer *index.htm* einfügen.

## Hinweise zur ansprechenden Gestaltung

1. Strukturieren Sie Ihre Homepage klar und übersichtlich. Machen Sie sich von Ihrem Projekt vorab eine Skizze über Ihre Linkstruktur.
2. Es hat sich im Web eingebürgert, Navigationsleisten links oder oben (es geht auch beides) auf der Seite zu positionieren. Halten Sie sich möglichst an diese Netikette.
3. Achten Sie auf Kontrast zwischen Hintergrund und Schrift. Nichts ist schlimmer als schwarze Schrift auf dunkelblauem Hintergrund.
4. Halten Sie Hintergrundbilder in einem Farbton mit nicht zu krassen hell/dunkel-Unterschieden.
5. Vermeiden Sie das horizontale Scrollen. Es zählt zu einer Unart im Web.
6. Unterlassen Sie, wenn möglich speicherintensive Bilder auf Ihrer Eingangsseite. Ist es unumgänglich, teilen Sie die Bilder in mehrere kleine Teilbilder und setzen sie diese zusammen.
7. Nerven Sie ihre Surfer nicht mit Cookies. Viele akzeptieren Sie schon gar nicht.
8. Wenn Sie mit transparenten Gif-Dateien arbeiten, wählen Sie beim Erstellen derselben eine Hintergrundfarbe (die Sie dann transparent setzen), die ihrem späteren Seitenhintergrund nahe kommt. Dadurch erreichen Sie, dass die Ränder des Bildes glatter erscheinen.
9. Wählen Sie eine nicht zu kleine Schriftart, so dass man sie in jeder Bildschirmauflösung noch gut lesen kann.
10. Bringen Sie nicht zu viel Text auf eine Seite und lockern Sie Ihre Texte mit Absätzen zur besseren Lesbarkeit auf.
11. Lassen Sie keine fremden Seiten innerhalb ihres Framesets zu, sondern lassen Sie sie in einem neuen Browserfenster erscheinen (`target= "blank"`). Es ist rechtlich nicht einwandfrei.
12. Setzen Sie animierte Gif's sparsam ein. Eine zu „belebte“ und kunterbunte Seite irritiert und lenkt von den Inhalten ab.
13. Verleihen Sie Ihrem Projekt ein einheitliches Bild, damit jeder Surfer weiß, dass er sich auf Ihrer Homepage befindet und nicht bereits auf einer anderen.



14. Verwenden Sie zum genauen Ausrichten von Schrift oder Bildern ein sogenanntes Blind-GIF. Sie erstellen es am einfachsten mit Ihrem Bildbearbeitungsprogramm, legen die Größe eines neuen Bildes auf 1 x 1 Pixel fest, reduzieren auf eine Farbe und setzen diese Hintergrundfarbe auf transparent.
  15. Beim Einsatz von Tabellen achten Sie darauf, dass Sie in leere Felder `<td>&nbsp;</td>` eintragen. Das ist ein Leerzeichen und sichert die Tabellenstruktur. Alternativ können Sie auch vorher beschriebenes Blind-GIF einsetzen.
  16. Wenn Sie Programme oder Zip-Archive zum Download anbieten achten Sie auf die Schutzrechte. Versehen Sie den Link darauf mit einem Hinweis auf die Dateigröße, damit der Surfer etwa die Dauer des Downloades abschätzen kann.
  17. Bei der Ausrichtung von Bildern und/oder Schrift in einer Tabelle ist oft die genaue Pixelangabe hilfreich. Schreiben Sie also `width="250"` anstelle von `width="50%"`.
- 



Viel Freude beim Experimentieren!