

HTML-Programmieren leicht gemacht

v.3.5

Dies soll eine Hilfe und Referenz zum erstellen von HTML-Dokumenten sein.
Um HTML zu erlernen, brauchen wir gewisse Voraussetzungen:

1. Einen HTML Editor
2. Eine Idee
3. Einen Provider
4. Einen Browser
5. Webspaces

1. Ein guter HTML-Editor in deutscher Sprache ist unter <http://www.obernsoft.de> zu finden. Dort liegt eine 16bit sowie eine 32bit Version des Editors ObernSoft-HPE. Die Version die Sie downloaden möchten, erreichen Sie über die einfache Menüstruktur!

2. Grund für die Homepage. Werbung für die eigene Firma oder eine private Homepage.

3. Einen Provider der wie CompuServe, T-Online, oder AOL die Zugang zum Internet bieten.

4. Am besten hierfür ist ObernSoft-Offline Browser geeignet, da er sehr schnell geladen wird (auch auf langsamen Computern!) und komplett offline arbeitet, das heißt er ignoriert Links zu Bildern oder ähnlichem, die nicht auf der eigenen Festplatte liegen. Er kann ebenfalls unter <http://www.obernsoft.de> bezogen werden!

5. Webspaces brauchen wir, um unsere Homepage hochzuladen. Viele Provider bieten kostenlos Webspaces an, aber auch einige kleinere oder größere Firmen gegen fast keine Gegenleistung!

Sind diese 5 Punkte erfüllt, können wir beginnen. Sie sollten bevor wir anfangen erst einmal nachdenken, ob Sie genau wissen, was HTML ist. Wenn nicht, informieren Sie sich!

Jetzt werden erst einmal eine einfache Seite generieren, die wir im Laufe der Zeit noch erweitern und ergänzen wollen. Haben Sie dies erfüllt, können Sie sich eine Referenz zu HTML anschauen, oder sich über die Farben informieren.

Wenn Sie nun bescheid wissen, wie Sie HTML programmieren, lesen Sie sich bitte durch, wie Sie programmieren sollten.

Stichworte:

Definition von HTML

Erklärung eines HTML-Tags

Einführung und erste Schritte in HTML

Referenz der Befehle in der HTML-Sprache

Referenz der Farben und ihre Bildung

Grafikformate

gutes HTML

Kostenlose Web-Space Angebote

HTML Erweiterungen

Was hat sich seit der Version 3 geändert?

Programmtips vom Autor

ACHTUNG:

Beachten Sie bitte das Copyright dieses Dokumentes!

Autor:

Wiktor Wodecki

http://ourworld.compuserve.com/homepages/wiktor_wodecki/index.htm

(C) Wodecki 1995-1996 - Helpfile generated by VB HelpWriter.

Farben

Farben in HTML bilden sich nach einem einfachem Prinzip:

Sie werden durch ein # und drei **HEX**dezimalzahl Codes gebildet. Die drei HEX-Codes werden aus folgendermaßen gebildet: RR (dezimaler Rotwert) GG (dezimaler Grünwert) BB (dezimaler Blauwert).

Da kein Mensch alle Dezimalwerte der einzigen Farbstufen kennt, habe ich hier eine Liste von Farben und ihrem Code hingestellt.

Farben und ihr dazugehöriger Code:

Farbe: Code:

Weiß [#FFFFFF]
Rot [#FF0000]
Grün [#00FF00]
Blau [#0000FF]
Magenta [#FF00FF]
Cyan [#00FFFF]
Gelb [#FFFF00]
Schwarz [#000000]
Blau-Violet [#9F5F9F]
Braun [#A62A2A]
Bronze [#8C7853]
Bronze II [#A67D3D]
Dunkel-Braun [#5C4033]
Dunkel-Grün [#2F4F2F]
Dunkel-Oliv-Grünark [#4F4F2F]
Dunkel Orchidee [#9932CD]
Dunkle Sonnenbräune [#97694F]
Dunkles Holz [#855E42]
Waldes-Grün [#238E23]
Golden [#CD7F32]
Grau [#C0C0C0]
Grün-Gelb [#93DB70]
Jägers-Grün [#215E21]
Indianer-Rot [#4E2F2F]
Khaki [#9F9F5F]
Hell Blau [#C0D9D9]
Hell Grau [#A8A8A8]
Helles Stahl Blau [#8F8FBD]
Helles Holz [#E9C2A6]

Linden-Grün [#32CD32]
Mandarinen-Orange [#E47833]
Mittel-Blau [#3232CD]
Mittel-Wald-Grün [#6B8E23]
Mittel-Orchidee [#9370DB]
Mittel-See-Grün [#426F42]
Mittel-Frühlings-Grün [#7FFF00]
Mittel-Türkis [#70DBDB]
Mittel-Violet-Rot [#DB7093]
Mittel-holz [#A68064]
Mitternachts Blau [#2F2F4F]
Marine-Blau [#23238E]
Neon-Blau [#4D4DFF]
Neon-Pink [#FF6EC7]
Neues Mitternachts Blau [#00009C]
Neue Sonnenbräune [#EBC79E]
Alt-Golden [#CFB53B]
Orange [#FF7F00]
Orange-Rot [#FF2400]
Orchideen Farbig [#DB70DB]
Blasses Grün [#8FBC8F]
Pink [#BC8F8F]
Pflaumen Farbe [#EAADEA]
Quartz [#D9D9F3]
Helles Blau [#5959AB]
Lachs Fraben [#6F4242]
Scharlach [#8C1717]
See-Grün [#238E68]
Halb-Süße Schokolade [#6B4226]
Sienna [#8E6B23]
Silber [#E6E8FA]
Himmels-Blau [#3299CC]
Verrissenes Blau [#007FFF]
Pikantes Pink [#FF1CAE]
Frühlings-Grün [#00FF7F]
Stahl-Blau [#236B8E]
Sommer-Himmel [#38B0DE]
Sonnenbräune [#DB9370]
Diestel Farben [#D8BFD8]
Türkis [#ADEAEA]
Sehr dunkles Braun [#5C4033]
Sehr helles Grün [#CDCDCD]
Violet [#4F2F4F]
Violet-Rot [#CC3299]
Korn-Farbig [#D8D8BF]
Gelb-Grün [#99CC32]

siehe auch unter Grafikformate

Es gibt natürlich noch andere Farben, die ich aber nicht aufgenommen habe, da ich Sie nicht kenne, bzw. mir keine Bezeichnung eingefallen ist. Wenn du noch einen Code kennst, eMaile mir ihn bitte.

(C) Wodecki 1995-1996 - Helpfile generated by VB HelpWriter.

Definiton von HTML

Als das Internet 1989 immer mehr dem privatem Nutzer zugänglich wurde, wurde es Zeit das Internet benutzerfreundlicher zu gestalten. Die Stanford Universität in den USA entwickelte 1989 die Hyper Text Markup Language, kurz: HTML. Sie entwickelte sich schnell zum Standart im Internet, das World Wide Web war geboren. Seitdem wird HTML kontinuierlich ausgebaut, verbessert und ergänzt.

HTML ist mit ein paar Ausnahmen eine Seitenbeschreibungssprache. Sie können jede beliebige Textdatei zur HTML Datei umwandeln, indem Sie einfach an bestimmten Stellen dem Browser durch HTML Tags kenntlich machen, wie er den Text zu formatieren hat.

HTML wird durch das HTTP (Hyper Text Transfer Protocol) aus dem Internet übertragen.

Freier Speicherplatz!!

viele Firmen bieten gegen einen Link auf die eigene Homepage Usern freien Webespace an. Zum Beispiel www.yi.com, www.schlund.de, www.geocities.com und viele andere!

Einführung in die HTML-Sprache

Als erstes erstellen wir auf unserer Festplatte ein Verzeichnis, z.B. HOMEPAGE. In dieses Verzeichnis werden wir nachher alle Bilder und Dateien speichern, welche wir für unsere Homepage verwenden wollen. Wenn wir später unsere Homepage uploaden wollen, müssen wir natürlich alle Dateien die sich im Verzeichnis befinden uploaden!

Grundwissen über den Aufbau einer HTML Datei

Bevor wir Informationen in unserer HTML-Datei schreiben, müssen wir das Grundgerüst der HTML Datei aufbauen. Dieses sieht ungefähr so aus:

```
<HTML>
<HEAD>
<TITLE></Title>
</HEAD>
<body background="Grund1.gif" bgcolor="#000000" text="#00ff00" link="#80ffff"
vlink="#400000 alink="#00ff00">

</BODY>
</HTML>
```

Das Grundgerüst besteht aus dem <HTML> Tag, dem Datei Kopf <HEAD> und dem Bodygerüst, in das wir später alle Informationen unserer Homepage reinschreiben. Im <HEAD> Tag notieren wir alle browserspezifischen Informationen, wie Scripts, Metainformationen (Das sind Informationen, die z.B. der Server mit Hilfe von Javascript abrufen kann), und die Titelzeile. Die Titelzeile ist die Zeile in Windows die standardmäßig blau unterlegt ist, und immer im oberem Bereich eines Fensters liegt. Diese Information müssen wir jedoch noch mit dem Subtag <TITLE> einführen. Im <BODY> Bereich werden die für den HTML Inhalt verantwortlichen Grundfarben eingetragen. So kann mit *background=""* ein Hintergrundbild eingetragen werden, mit *bgcolor="#(Farbenwert)"* eine Hintergrundfarbe (Wenn beides eingetragen ist, gibt der Browser dem Hintergrundbild den Vorzug). Mit *text="#(Farbenwert)"*, *link="#(Farbenwert)"*, *vlink="#(Farbenwert)"* und *alink="#(Farbenwert)"* bestimmen Sie die Text, Link, besuchte Link und die aktive Linkfarbe. Das heißt Sie wählen die Texteigenschaften wie Farbe vor und nach dem anklicken aus! Im nachfolgenden Bereich wollen wir gleich unsere Informationen einbringen, die dem Surfer in seinem Browserfenster angezeigt werden sollen. Anschließend schließen wir die Tags <BODY> und <HTML> wieder.

erstes designen der Homepage

Als erstes wollen wir eine einfache zentrierte Überschrift mit einem durchgezogenem darunter Strich kreieren.

Wir verwenden hierzu den `<Hx>` und den `<center>` Tag. Der `<Hx>` Tag führt eine Überschrift ein die die Größe x hat. Für x müssen Sie einen Wert zwischen einschließlich 1 und 6 wählen. Der Wert 1 steht für die größte, und die 6 für die kleinste Überschrift. Eine Trennlinie erreichen wir mit dem Befehl `<hr>`.
Nun schreiben wir in die ersten Zeilen folgendes:

```
<HTML>
<HEAD>
<TITLE></Title>
</HEAD>
<body background="Grund1.gif" bgcolor="#000000" text="#00ff00" link="#80ffff"
vlink="#400000 alink="#00ff00">
<h1><center>Meine erste HTML Seite</center></h1>
<hr>

</BODY>
</HTML>
```

Am besten probieren Sie dies mal aus, und begutachten das Resultat Ihrer Arbeit. Als Nächstes wollen wir ein Bild von uns unter die Überschrift mit dem Strich setzen. Der entsprechende Tag heißt `` Tag.

Die Syntax für den `` Tag:

```
<IMG SRC="BILD1.GIF" align=center>
```

SRC steht für das englische Wort Source und gibt den Pfad an, wo das Bild liegt. Da wir unser Bild in unserem Ordner HOMEPAGE abgelegt haben, soll uns dieser nicht weiter interessieren. Da uns das Bild in seiner jetzigen Größe nicht gefällt, verkleinern wir es etwas. Dafür müssen wir unsere Syntax durch zwei Parameter ergänzen. Diese Parameter sind *height=x* und *width=x*. X steht für eine Größe die variieren können. wir nehmen als Beispiel für die Höhe (*height*) 63, und für die Breite(*width*) 501. Die Zahlangaben wrden immer in Pixeln gemacht. Der Parameter *ALIGN=CENTER* gibt die Ausrichtung des Bildes an. Anstatt *CENTER* könnte dort auch *LEFT* für Links und *RIGHT* für Rechts stehen.

Jetzt steht in unser Bildzeile folgendes:

```
<IMG src="Bild1.gif" height=63 width=501>
```

Allerdings wollen wir das Bild nicht so dicht unter den Trennstrich setzen, also fügen wir mit dem Tag `
` einen Zeilensprung ein.

Wir wollen ein zweites, weiteres Bild in unsere Homepage einfügen. Also springen wir mit Hilfe des Tags `
` in die nächste Zeile unseres Dokumentes und wieder holen die Syntax, mit der Ausnahme, das wir jetzt natürlich ein anderes Bild nehmen, nämlich BILD2.GIF.

```
<HTML>
<HEAD>
<TITLE></Title>
</HEAD>
<body background="Grund1.gif" bgcolor="#000000" text="#00ff00" link="#80ffff">
```

```
vlink="#400000 alink="#00ff00>
<h1><center>Meine erste HTML Seite</center></h1>
<hr>
<br>
<IMG src="Bild1.gif" height=63 width=501>
<br>
<IMG src="Bild2.gif" height=63 width=501>
</BODY>
</HTML>
```

Unter unsere beiden schönen Bilder setzen wir wieder eine Linie mit <hr>. Da diese Linie allerdings etwas dicker ausfallen sollte, ergänzen wir den <hr> Tag um den Parameter size=3. Eine normale Linie hat die Dicke 1. mit dem Parameter 3, verdreifachen wir die Dicke also.

Jetzt wollen wir etwas schreiben. Dazu aktivieren wir Zuerst den automatischen Zeilenumbruch mit dem Tag <P>. Hinter dem <P> schreiben wir jetzt unseren Text. Als Beispiel schreiben wir:

Dies ist mein erstes HTML-Dokument. Wie findest du es?? Ich habe es selbst geschrieben, und es macht riesen Spaß. Ich benutze dafür eine tolle Onlinehilfe und ein echt cooles Programm. Natürlich werde ich es bald registrieren!

Nach diesem Text setzen wir wie bei fast jedem HTML-Tag das </p>.

Unsere Htmlseite müßte jetzt so aussehen:

```
<HTML>
<HEAD>
<TITLE></Title>
</HEAD>
<body background="Grund1.gif" bgcolor="#000000" text="#00ff00" link="#80ffff"
vlink="#400000 alink="#00ff00>
<h1><center>Meine erste HTML Seite</center></h1>
<hr>
<br>
<IMG src="Bild1.gif" height=63 width=501>
<br>
<IMG src="Bild2.gif" height=63 width=501>
<hr size=3>
<p>Dies ist mein erstes HTML-Dokument. Wie findest du es?? Ich habe es selbst
geschrieben, und es macht riesen Spaß. Ich benutze dafür eine tolle Onlinehilfe und ein
echt cooles Programm. Natürlich werde ich es bald registrieren!</p>
</BODY>
</HTML>
```

An dieser Stelle möchte ich erst einmal erklären, wie ein "Link" funktioniert. Ein Link ist eine Verbindung innerhalb eines Dokumentes, dem Internet oder einer eMail Funktion. Die Syntax des Links sieht mit einer Ausnahme immer folgendermaßen aus:

`Text`

XXXXX steht in diesem Fall für eine der drei Optionen. Wir können anstatt XXXXX eine Seite im Internet angeben, also eine URL wie "http://www.polpotyle.com". Wenn wir eine eMailfunktion einbinden wollen, ersetzen wir XXXXX durch

"mailto:Mustermann@provider.com"(Wie Ihre Internetadresse ist, erfragen Sie am besten bei Ihrem Provider oder dem Sysop Ihrer Mailbox!). Wollen wir jedoch einen Link innerhalb einer Seite definieren, sieht die Syntax folgendermaßen aus:

`Text`

Nun muß jedoch an der Stelle wohin der Link führen soll folgendes stehen:

`Text`

XXXXX steht in diesem Fall für eine Sprungmarke, und Muß in beiden Fällen gleich bleiben!

Der "Text" steht in allen 3 Möglichkeiten für den Text, den der User anklicken soll, um die gewünschte Funktion auszuführen, sei es eMails oder einen Sprung zu vollführen. Die Funktion eines Links kann auch innerhalb eines zusammenhängenden Textes ausgeführt werden. So wollen wir unseren Text den wir schon implementiert haben, um einen Satz erweitern:

Wenn Dir diese Homepage gefällt, dann schicke mir doch eine `eMail!!`

Wenn Du jetzt deine Page in einem Browser anschaust, dann wirst du feststellen, das wenn Du auf das Wort E-MAIL klickst, sich ein eMail Fenster öffnen wird.

Jetzt wollen wir eine weitere Trennlinie mit `<hr>` einleiten. Anschließend machen wir noch eine kleine Liste mit Links. Dazu benutzen wir den Aufzählungstag ``. die Syntax für den `` Tag lautet folgendermaßen:

`(Text)`

Wir wollen jedoch 2 Links einbauen, und so setzen wir 2 `` Tags untereinander. Und hinter den einen schreiben wir `Hier ein Link zu einem Superserver`, hinter den anderen Hier ein Link zu einer `Multiplayerseite`

Nun müßte unser HTML Dokument so aussehen:

`<HTML>`

`<HEAD>`

`<TITLE></Title>`

`</HEAD>`

`<body background="Grund1.gif" bgcolor="#000000" text="#00ff00" link="#80ffff" vlink="#400000 alink="#00ff00">`

`<h1><center>Meine erste HTML Seite</center></h1>`

`<hr>`

`
`

``

`
`

``

`<hr size=3>`

`<p>Dies ist mein erstes HTML-Dokument. Wie findest du es?? Ich habe es selbst`

geschrieben, und es macht riesen Spaß. Ich benutze dafür eine tolle Onlinehilfe und ein echt cooles Programm. Natürlich werde ich es bald registrieren! Wenn Dir diese Homepage gefällt, dann schicke mir doch eine [eMail](mailto:Mustermann@provider.com)!!

*Hier ein Link zu einem Superserver
Hier ein Link zu einer [Multiplayerseite](http://www.geocities.com/SiliconValley/8068/)*

*</BODY>
</HTML>*

Jetzt schauen Sie sich Ihre Arbeit in einem Browser an, und Sie können stolz sein, Sie haben Ihre erste Webpage designed. Sie ist zwar nicht sehr anspruchsvoll, aber immerhin. Um mehr HTML Tags kennenzulernen, [klicken Sie hier](#). Viele der hier verwendeten Tags hat eine ganze Reihe von Parametern. Über diese können Sie sich am besten in der [Referenz](#) informieren.

(C) Wodecki 1995-1996 - Helpfile generated by VB HelpWriter.

Ein HTML Tag signalisiert dem Browser in welcher Art der eingeschlossene Text formatiert werden soll. Ein Tag sieht immer so aus:

<Befehl>eingeschlossener Text</Befehl>

Es können auch problemlos mehrere Tags einen Textabschnitt definieren, wie zum Beispiel

```
<center><blink>Text</blink></center>
```

Diese 2 Tags lassen das Wort "Text" in die Mitte rücken und Blinken. HTML unterscheidet nicht zwischen groß und kleinschreibung, so ist
 das gleiche wie
 oder
.

HTML Tags können auch Parameter haben. Statt zu schreiben

```
<h1><center>TEXT</center></h1>
```

könnten wir, um uns die Arbeit zu erleichtern, folgendes schreiben:

```
<h1 align=center>TEXT</h1>
```

Diese Methode funktioniert umgekehrt nicht. Auch tut sie`s nicht bei allen Tags. Jedoch bei den meißten, die eine Beschreibung ihrer Ausrichtung zulassen.

Referenz von HTML

Text Layout

`(text)`

zeigt den ausgewählten Text im Fettdruck

`<I>(text)</I>`

zeigt den ausgewählten Text im Kursivdruck

`<U>(text)</U>`

Unterstreicht den gewählten Text

`(text)`

Zeigt dem Browser an, dass der ausgewählte Text besonders wichtig ist. Wird von den meisten Browsern im Fettdruck interpretiert

`<tt>(text)</tt>`

lässt den gewählten Text diktengleich erscheinen

`<s>(text)</s>`

streicht den gewählten Text durch

`<small>(text)</small>`

verkleinert den gewählten Text

`<big>(text)</big>`

vergrößert den gewählten Text

`_(text)`

stellt den gewählten Text tief

`^(text)`

stellt den gewählten Text hoch

`<blink>(text)</blink>`

Lässt den gewählten Text in festen nicht veränderbaren Intervallen blinken.

`(text)`

Der Font Tag bestimmt Größe und Farbe des durch ihn erfassten Textabschnittes.

`(text)`

lässt den gewählten Text in beliebiger Farbe erscheinen. Der Farbcode (in diesem Fall #000080) gibt an, welche Farbe der Text haben soll. Siehe dazu die [Farbtabelle](#)

`<fontsize="+1">(text)`

läßt den gewählten Text in einer Größe von -2, -1 normal, +1 und +2 erscheinen

`<hx>(text)</hx>`

läßt den gewählten Text in der Überschriftsgröße X erscheinen. Für X können wir eine Zahl zwischen 1 und 6 eingeben, wobei 1 die größte und 6 die kleinste Variante ist.

`<blink>(text)</blink>`

läßt den ausgewählten Text blinken

`(text)`

läßt den ausgewählten Text emphatisch erscheinen

`<cite>(text)</cite>`

Zeigt dem Browser an, das der ausgewählte Text ein Zitat ist. Wird von den meisten Browsern in Kursivschrift interpretiert

`<code>(tag)</code>`

Zeigt dem Browser an, das der eingetagte Tag nicht interpretiert, sondern nur am Monitor abgebildet wird

`<adress>(text)</adress>`

Der Textbereich zeigt dem Browser den Autor und dessen email Adresse an. Wird von den meisten Browsern in Kursivschrift wiedergegeben.

Sonderzeichen

Gewisse Zeichen können von einigen Browsern fehlinterpretiert werden. Es gibt daher Sonderzeichen, die International von allen Browsern anerkannt werden. Dies sind für die deutsche Sprache folgende:

Zeichen	HTML-Ersatz
ä	ä
Ä	Ä
ü	ü
Ü	Ü
ö	ö
Ö	Ö
ß	ß
<	<
>	>
"	"
&	&

Seiten Layout

`<p>(text)</p>`

fügt einen automatischen Zeilenumbruch im Text ein

`
(text)</br>`

Fügt an dieser Stelle einen Zeilensprung ein

`<pre>(text)</pre>`

läßt den Text so erscheinen, wie man ihn im Editor definiert.

`<center>(tag)</center>`

Läßt den ausgewählten Tag in die Mitte rücken!

`<hr>`

Läßt eine Linie auf dem Browser erscheinen, die in ihrer Größe durch SIZE, in der Länge durch width= und in ihrer Ausrichtung durch ALIGN= verändert werden kann.

Listen Tags

`<U>`

`(text)`

...

`</U>`

Setzt einen dicken Markierungspunkt vor den Text

``

` (text)`

...

``

Durch den Tag OL wird die Liste numerisch. Das heißt, alle ``'s in diesem Tag werden durchnummeriert, un zwar von oben nach unten!

Frames

Mit Hilfe von Frames kann man das Browserfenster in verschiedene, voneinander unabhängige Teile aufteilen. Die Verschiedenen durch Frames definierten Fenster können verschiedene HTML-Dateien angezeigt werden.

Frames werden anstatt des BODY Bereichs in einer HTML Datei notiert. Das sieht dann etwa so aus:

`<html>`

```

<head>
<title>Frame-Test</title>
</head>
<frameset cols="50%,50%">
  <frame src="Zelle1.htm" name="1">
  <frame src="Zelle2.htm" name="2">
</frameset>
</html>

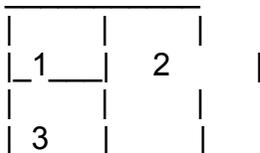
```

Im Frameset definieren sie ob der Bildschirm Horizont, oder vertikal aufgeteilt werden soll. Nach dem Frameset kommen die eigentlichen HTML Dateien, die im Fenster angezeigt werden sollen. Wollen sie nun das eine Fenster nun noch mal unterteilen, schreibt man anstatt der Framequelle einfach ein neues Frameset. Das sieht dann zum Beispiel so aus:

```

<html>
<head>
<title>Frame-Test</title>
</head>
<frameset cols="50%,50%">
  <frameset rows="50%,50%">
    <frame src="Zelle1.htm" name="1">
    <frame src="Zelle3.htm" name="3">
  </frameset>
  <frame src="Zelle2.htm" name="2">
</frameset>
</html>

```



Anhand der Skizze kann man erkennen, wie die Frames jetzt aufgebaut sind. Ich gebe zu die Skizze ist nicht besonders gut, aber das Programm mit dem ich arbeite ist in der Beziehung etwas eigensinnig.

Im Frameset definiert man per ROWS und COLS ob der Bildschirm horizontal oder vertikal aufgeteilt werden soll. Außerdem kann man in Pixel- oder in Prozentangaben angeben. Natürlich müssen sich diese Aufgaben sich nicht nur auf 2 Angaben beschränken. Es können beliebig viele Angaben dort stehen, Voraussetzung ist jedoch, daß die Prozent oder Pixel Angaben in sich stimmen, also nicht das im Endeffekt 103% entstehen oder sonst etwas in der Richtung. Man darf auch den Joker * benutzen, um dem Browser zu zeigen, das der Rest für das letzte Fenster reserviert ist, also würde so ein Frameset etwas so aussehen:

```

<frameset rows="135,650,99,*">

```

--Links in Frames--

Wenn man innerhalb eines Framefensters einen Link betätigt, so erscheint das Ergebnis des Links im Aktuellem Fenster!! Man kann jedoch mit Hilfe von TARGET="" das Framefenster wählen, in welchem der Link erscheinen soll. Dazu müssen wir jedoch die Framefenster bezeichnen, in unserem Beispiel weiter oben sind die Namen 1, 2 und 3. Ein Link der von Zelle1 betätigt wird, und in Fenster 2 erscheinen soll, sähe dann so aus:

```
(text)<a href="datei.htm" target="2">Link</a>(text)
```

Um aus dem Frameset auszubrechen, gibt man als target einfach "_parent" ein. Der Link wird dann im übergeordnetem Fenster dargestellt. Dies kann ein weiteres Framefenster sein, oder das standart Browser Fenster. Soll mit einem Link aus sämtlichen Framesets auf einmal ausgebrochen werden, gibt man als Target einfach "_top" an. Der Browser verläßt dann das Frameset, und baut die Seite neu auf. Der ursprüngliche Zustand kann dann jedoch nicht mehr hergestellt werden, es sei denn, man lädt das ganze Frameset noch einmal. Wenn man durch einen Link ein völlig neues, unabhängigen Browserfenster öffnen möchte, sollte man als Target Befehl "_blank" eingeben. Es öffnet sich dann ein neues Fenster das den Inhalt des Links anzeigt.

Wenn Sie eine reine Linkseite innerhalb eines Framesets aufbauen, ist es nicht besonders sinnvoll bei jedem Link ein target="_top" zu setzen. Dafür gibt es den Befehl <base>. Als Parameter geben Sie einfach target="_parent" ein. Natürlich kann hier auch ein von Ihnen definierter Fenstername stehen.

Wenn Sie so eine Lösung benützen wollen, achten Sie bitte darauf, das der <base> Tag im Dateikopf stehen muß!! Also im <head>(dateikopf)</head> Bereich!

Es ist leider nicht möglich mit einem Link 2 Fenster, und oder alle Framefenster neu zu laden. Diese Lösung ist jedoch mit JavaScript realisierbar.

Links

```
<a href="http://www.polopty.com">(text)</a>
```

Dies ist eine Linkform, die zu einer anderen Internetadresse führt.

```
<a href="mailto:Mustermann@provider.com">(text)</a>
```

Diese Linkform läßt ein eMail Eingabemenü im Browser erscheinen, indem eine eMail an die Adresse abgeschickt wird!

```
<a href="#XXX">(text)</a>
```

Dies ist eine Linkform. Dieser Link führt innerhalb einer Webseite zu einem spezifischem Ort, der mit (text) markiert wird!. XXX muß identisch mit dem Folgelink sein!

```
<a name="XXX">(text)</a>
```

Dies ist eine Sprungmarke. Der vorhergehende Link führt hierher. XXX muß identisch mit dem erstem Link sein!

Bilder und Graphiken

Wenn wir Bilder implementieren wollen, benutzen wir entweder das JPG oder das GIF Format, welche von allen gängigen Browsern unterstützt werden! Wenn wir andere Formate benutzen, kann es sein, das einige Browser Fehlermeldungen ausgeben werden!

```
</A>
```

Ein Bild wird in die gegebene Stelle eingesetzt. Sollte das Bild noch nicht geladen sein, erscheint ein Alternativtext auf dem Browser, den man hier editieren kann. zwei weitere Parameter kann man innerhalb der Bildsyntax noch einfügen, unzwär die Größe und Breite des Bildes in Pixeln. Die Parameter sind "width=(Pixelwert)" und "height=(Pixelwert)"

Achtung: als Quelle (SRC) eines Bildes können Sie auch eine Datei auf einem fremdem Rechner im Internet benutzen, sie müssen dann nur als Quelle die URL des Bildes angeben, zum Beispiel bei einem Counter!

```
<body background="Bild.gif" bgproperties=fixed>
```

Hintergrundgrafik mit Wasserzeichen

Ein Bild in Verbindung mit einem Link

```
<a href="XXX"></A>
```

XXX steht in diesem Fall für eine der drei uns bekannten Link Arten. Ansonsten siehe vorherige Erklärungen!

Es gibt noch eine zweite Variante, um ein Bild mit einem Link zu kombinieren. Diese Funktion heißt "Imagemaps" oder "clickable maps". Sie ist ungleich komplizierter, hat jedoch den Vorteil, das jedes Pixel in einem Bild einen Link zu einer anderen Stelle durchführen kann. Da ich zur Zeit mir überlege wie ich dieses Prinzip am deutlichstem machen kann, verweise ich vorläufig auf die Shareware Szene, die in diesem Bereich hervorragende Arbeit leistet.

Tabellen

```
<Table>
```

```
<Tr><Th></Th><Th></Th></Tr>
```

```
<Tr><Td></Td><Td></Td></Tr>
```

</Table>

Um eine Tabelle zu generieren, genügt ein simpler <Table> Tag. Der <Table> Tag erlaubt eine Vielzahl von Parametern:

align=(left, right oder center)
weist der Tabelle eine Ausrichtung auf der Seite zu

border=(wert)
gibt der Tabelle optional einen Rand

bordercolor=(farbwert)
legt die Farbe des Rahmens fest

cellspacing=(Pixelangabe)
gibt den Zellen innerhalb der Tabelle einen Rahmen

height=(Prozent oder Pixelangabe)
legt die Höhe der Tabelle fest. Bei Weglassen dieses Wertes wird die Tabelle so hoch, wie es die Werte der Zellen erfordern

width=(Prozent oder Pixelangabe)
legt die Breite der Tabelle fest. Siehe height

Innerhalb des <Table> Tags leiten sie mit <TR> eine Zeile ein, und beenden sie mit </TR> wieder. Innerhalb dieses Subtags ist die Anzahl und Art der Reihen angegeben. So wird mit dem <TH></TH> Tag der Kopf einer Tabelle notiert, und erscheint bei den meisten Browsern im Fettdruck. Der <TD> Tag notiert eine ganz normale Tabellenreihe. Ebenso wie der <TH> Tag notiert der <TD> Tag eine Reihe. Innerhalb dieser Subtags ist es möglich durch verschiedene Parameter das Aussehen der Zelle zu verändern:

bgcolor=(farbwert)
weist der Zelle eine bestimmte Hintergrundfarbe zu

align=(left, right oder center)
erlaubt den Zelleninhalt auszurichten

border=(Wert)
Optional kann die Zelle einen bestimmbaren Rand bekommen

height=(Pixelangabe)
weist der Zelle eine bestimmte Höhe in Pixeln zu

width=(Pixelangabe)

weist der Zelle eine bestimmte Breite in Pixeln zu

`valign=(top oder bottom)`
richtet den Zellentext oben oder unten aus

`colspan=(Zellenwert)`
Dieser Parameter erlaubt es Zelleninhalte miteinander zu verbinden. Jedoch muß wenn zum Beispiel 2 Zellen miteinander verbunden werden sollen, die nachfolgende Zelle gelöscht werden, da sonst eine weitere unerwünschte Zelle in Ihrer Tabelle auftaucht

`rowspan=(Zellenwert)`
Hier werden Zelleninhalte in der Senkrechte miteinander verbunden. Hier gilt die gleiche Regel wie bei `rowspan`

Bedenken Sie jedoch, das Sie bei einigen dieser Parameter in Konflikte geraten können, die im `<Table>` Tag notiert werden.

Multimedia

--Musik--

`<bgsound src="sound.au" alt="(text)" loop=infinite>`
Fügt Background Sound zur Seite hinzu. "loop=" gibt an, wie oft die Musik wiederholt werden soll.

--Videos--

``
einfaches Video wird mit dem Betriebssystem eigenem Player wiedergegeben! Vorsicht, das AVI (MPG, MOV, etc.) muß vorher komplett heruntergeladen werden! Auch hier wieder die Einstellungen für die wiederholungen, die Höhe und Breite sowie eine alternative Grafikanzeige!

Forms und Eingabe Masken

Forms oder auch Eingabemasken werden dazu gebraucht Informationen vom Betrachter ohne eMail an den Hersteller zu schicken. Diese Eingabemasken können beispielsweise wie ein Vertrag oder ein Fragebogen aufgebaut sein.

Mit dem `<FORM>` Tag signalisieren Sie dem Browser, das jetzt eine Eingabemaske kommt. In den selben Tag müssen jedoch noch Informationen bezüglich der Art des Zustellens. Dies sieht dann etwa so aus:

`<Form action="mailto:mustermann@provider.com" method="post">`
Anstatt „mailto:...“ könnten Sie auch ein CGI Script definieren, welches dann vom Browser ausgeführt wird.

Mit `</FORM>` beenden Sie die Eingabemaske wieder.

Um Informationen vom Benutzer zu erlangen gibt es verschiedene Eingabemöglichkeiten. Ich möchte sie hier aufzählen, und der Reihe nach besprechen:

- Einzeilige Texteingabe
- Mehrzeilige Texteingabe
- Radioeingaben
- Checkboxen
- Auswahlmenüs
- Aktions Knöpfe

Bevor ich jedoch in die Besprechung der einzelnen Arten eingehe, möchte ich noch den Aufbau eines Tags zur Erfassung der Daten:

```
<Input Type="" Name="" Size="" Maxlength="" Value=""></Input>
```

Der Tag heißt INPUT. `TYPE=""` signalisiert dem Browser um welche Eingabe es sich handelt. Mit `NAME=""` wird ein Name für die Eingabe vergeben. Besonders bei den Radioeingaben wird die Namenvergabe wichtig. `SIZE=""` und `MAXLENGTH=""` setzen die Länge der Eingabe in Buchstaben fest. `VALUE=""` setzt eine Vorgabe für Texteingaben fest. Bei Knöpfen hat der `VALUE` Wert jedoch eine spezielle Bedeutung, auf die ich später noch eingehen werde.

Einzellige Texteingaben `TYPE="TEXT"`

stellt ein vorher definiertes Textfeld zur Verfügung.

Mehrzeilige Texteingabe

```
<Textarea Name="" Value="" Rows="" Cols=""></Textarea>
```

Die mehrzeilige Texteingabe ist eine Eingabeform, die vom INPUT Tag abweicht. Sie generiert einen neuen Tag, dessen Spalten(Cols) und Reihen(Rows) in Buchstabenlängen wieder definiert werden müssen.

Radioeingaben `TYPE="RADIO"`

Radioeingaben stellen eine Vielzahl von möglichen Auswahlmöglichkeiten vor, wovon jedoch nur eine einzige angeklickt werden kann. Man muß sich das in etwa so vorstellen, das man einen kleinen Kreis hat, den man anklicken kann, also auf JA setzen kann. Um dem Browser zu signalisieren, das mehrere Radioeingaben zusammengehören, müssen diese denselben Namen tragen.

Checkboxen `TYPE="CHECKBOX"`

Checkboxen funktionieren im Prinzip genauso wie Radioeingaben, nur mit dem Unterschied, das sie nicht zusammengehören.

Auswahlmenüs

```
<Select name="">  
<Option>
```

```
<Option>
<Option>
<Option>
</select>
```

Auch die Auswahlmenüs weichen vom INPUT Tag ab. Hinter die <OPTION> Subtags, werden die einzelnen Auswahlmöglichkeiten dargestellt. Ebenso wie bei den Radioeingaben, kann hier nur eine Auswahl getroffen werden.

Aktions Knöpfe TYPE="BUTTON"

Der VALUE="" hat hier eine besondere Bedeutung. Er gibt an, was auf dem Knopf steht. Natürlich können Sie dieses Feld auch leerlassen, jedoch wird der Benutzer dann einen grauen Knopf sehen, auf dem nichts draufsteht, und dessen Funktion ihm bis zum drücken unerschlossen bleibt. Solche Knöpfe sind vorallem für Scriptsprachen wie Javascript interessant, da es da zum Beispiel bei Javascript den onClick Button gibt...

Es gibt besondere Knöpfe, die jedoch ein eigenen TYPE Wert haben; Dies sind die Knöpfe zum absenden und zum löschen der eingegebenen Daten:

```
<INPUT NAME="Abschicken" TYPE="submit" value="Abschicken">
```

sendet die Daten per ACTION (im Kopf des Formtags) dem Hersteller zu.

```
<INPUT NAME="Reset" TYPE="reset" value="Löschen">
```

löscht die durch den Benutzer eingegebenen Werte.

--Auswertung von Forms und Eingabemasken--

Sofern Sie nicht ein CGI Script zur Erfassung der Daten benutzen, werden Sie jedesmal wenn Sie eine Form zugesendet bekommen, verschiedene ASCII Zeichen sehen. Diese beruhen etwa nicht auf Fehleingaben des Benutzers sondern des Internets. Um die Daten wieder lesbar zumachen, brauchen Sie ein TOOL zum ENTPARSEN. Es gibt ein tolles Sharewaretool dazu, welches sich webparse nennt. Ich weiß nicht, wo man es bekommt, ich habe es mal auf einer Shareware CD gefunden.

*****Kommentare*****

```
<!--(text)-->
```

Fügt einen Kommentar in das Dokument ein, das vom Browser nicht gelesen, jedoch durch den User per Texteditor angesehen werden kann. Wird auch zur Einbindung von Scriptsprachen wie Javascript benutzt.

Javascript ist eine von Netscape (<http://www.netscape.com>) erfundene Programmiersprache. Sie macht HTML noch benutzerfreundlicher, und ist auch nicht schwer zu erlernen, da sie auf JAVA basiert, welches wiederum auf C+ basiert!

Copyright

Das Copyright© für dieses Dokument liegt beim Autor (Wodecki@obernsoft.de). Das Copyright© für das Programm das Ich für die Beispiele verwendet haben, liegt beim Autor Jörg Oberndorf (oberndorf@obernsoft.de).

Auszüge in jeglicher Form sind ohne Einverständnis des Autors verboten, und können strafrechtlich verfolgt werden.

Dieses Dokument darf und soll in eigenem vermessenen weiter verbreitet und vertrieben werden. Auch die plazierung dieses Produktes auf Shareware oder Heft-CD`s ist erlaubt. Ich würde nur in so einem Fall bitten mich per eMail zu benachrichtigen, da ich gerne wissen würde, was mit meinem Produkt passiert.

Die aktuelle Version dieser Hilfedatei ist im Internet unter http://ourworld.compuserve.com/homepages/wiktor_wodecki/html.zip zu finden. Sie liegt dort als gepacktes ZIP Archiv vor. Sollte Ihr Computer ZIP nicht kennen, besuchen Sie die Seite <http://www.winzip.com>!

HTML "gut" programmieren

Sie sollten, wann immer Sie HTML programmieren, sich nicht darauf einlassen jeden beliebigen Browser unterstützen zu wollen. Die Mehrheit im Internet surft mit Netscape herum (etwa 70%). Sie sollten sich daher auf die von Netscape unterstützten Tags stützen. Um herauszufinden, welche dies sind, sollten Sie einfach mal bei Netscape (<http://www.netscape.com>) vorbeischaun. Da auch in letzter Zeit Microsoft mit seinem Internet Explorer 3.0 den Browsermarkt stürmt, sollten Sie versuchen so zu programmieren, dass Ihre Seite von möglichst beiden Browsern unterstützt wird.

Viele der Tags die ich in meiner Referenz erwähne, arbeiten sowohl in Netscape wie auch im Internet Explorer von Microsoft. Sollte ein Tag nur bei einem bestimmten Browser funktionieren, so werde ich dies kenntlich machen.

(C) Wodecki 1995-1996 - Helpfile generated by VB HelpWriter.

HTML Erweiterungen

Einigen Leuten im Internet reichten die Möglichkeiten vonHTML nicht aus. Sie konstruierten eigene, in HTML eingebundene Programmiersprachen. Diese Script's müssen entweder vom Browser interpretiert werden, oder Sie nehmen dem Browser die Abreit ab, indem sie ihm nur die fetigen Programcodes zur Ausführung übermitteln.

Folgende Scriptsprachen sind zur Zeit im Internet weit verbreitet:

ActiveX

ActiveX ist eine neue Programmroutine von Microsoft.(<http://www.microsoft.com>). Sie ist eine für Windows95 und Windows NT programmierte Schnittstelle, die Programme stärker mit dem Internet kombinieren läst. Mit ActiveX lassen sich auch die typischen routinen wie mit Javascript herstellen.

CGI

CGI ist eine Serverroutine. CGI-Scripts dienen hauptsächlich dazu, Vorgänge wie das entgegennehmen von Formularen oder Webcountern auf Servern zu automatisieren. CGI kann in vielen Sprachen wie C oder Visual Basic programmiert werden. Es wird jedoch meistens die Sprache Pearl benutzt, da diese auf fast jedem System läuft.

Java

Java ist ein von SUN(<http://www.sun.com>) programmiertes Betriebssystem. Es erlaubt dem Webdesigner größere Interaktivität mit dem Anwender.

Javascript

Javascript ist eine von Netscape (<http://www.netscape.com>) entwickelte Programmiersprache. Sie macht HTML noch benutzerfreundlicher, und ist auch nicht schwer zu erlernen, da sie auf JAVA basiert, welches wiederum auf C+ basiert!

Als Link bezeichnet man einen Querverweis im Internet. Die vorzüge von HTML beruhen, darauf, das es Hypertext fähig ist, und dies beinhaltet eben die Eigenschaft, durch anklicken eines hervorgehobenen Wortes eine Sprungmarke zu erreichen. Das ganze WWW Funktioniert nach diesem Prinzip.

Änderungen zur vorherigen Version 3.0

Ich habe ein komplett neues Tutorial geschrieben. Außerdem wurden große Teile der Referenz und der Farbtabelle überarbeitet. Weiterhin gibt es jetzt Informationen zu HTML Erweiterungen wie Javascript, CGI und andere

(C) Wodecki 1995-1996 - Helpfile generated by VB HelpWriter.

Tips vom Autor

HTML zu programmieren ist viel tipp und schweißarbeit. Spätestens wenn Sie sich an Frames ranwagen, und die Frames nicht das machen was Sie wollen, werden Sie sich ärgern. Auch ich hatte dieses Problem.

Dann kam jedoch dieser wunderbare AHA-Effekt. Schauen Sie sich doch einfach mal in der Sharewareszene um, und schauen Sie was es da so gibt!

Einige besonders schöne Tools, die auch ich benutze um mir das Leben leichter zu machen, habe ich hier aufgelistet:

ObernSoft-Homepage Wizard

<http://www.obernsoft.de>

Dieser Wizzard erstellt in sekunden ein HTML Grundgerüst, und speichert dieses als Datei ab. Sie können interaktiv die Farben u.a. festlegen.

Frame it!

<http://www.iinet.net.au/~bwh/frame-it.html>

Dies ist ein sehr nützliches Tool zum anlegen und einrichten von Framesets. Es bietet eine komfortable Oberfläche, auf der Sie alle Eigenschaften wie Framename, Breite, Höhe etc. festlegen können.

Mapedit

<http://www.boutell.com/mapedit>

Mapedit ist ein Tool zum erstellen von "multilink" Bildern. D.h. in einem Bild können verscheide Links versteckt sein. Durch Angabe der HTML Datei, wird die Datei so umgeschrieben, das das Bild durch vorher definierte Bereiche anklickbar wird.

Grafikformate

im Internet haben sich 2 große Grafikformat Standarte durchgesetzt. Einmal ist dies das JPG und das GIF Format.

Definition und Verwendungszweck vom JPG Format

Das JPG oder auch JPEG Format genannt, wird hauptsächlich zur Anzeige von fotorealistischen und computer generierten Bildern zuständig. Es ist dafür hervorragend geeignet, da es bei großer Komprimierungsrate eine gute Bildqualität und kleine Dateien liefert.

Definition und Verwendungszweck vom GIF Format

Das GIF Format ist dazu da, kleinere Grafiken wie Buttons oder Logo`s darzustellen. Es hat dafür auch sehr viele Vorteile, die ich Ihnen weiter unten näherbringen möchte. Das GIF Format hat sogenannte Unterformate, daß heißt, es ist variabel in seiner Einsatzart:

GIF87

Dies ist das ursprüngliche, standard Format von GIF

GIF89a

GIF89a hat verschiedene Optionen, die man einstellen kann, das sind folgende:

Interlaced

Der Interlaced Modus ändert an der Bilddarstellung nichts. Jedoch beeinflusst er den Bildaufbau beim laden. Ein normales GIF wird immer von oben nach unten geladen. Bei einem GIF im Interlaced Modus, wird jedoch erst die Grundstruktur des GIF`s übertragen, dann der Rest. Folgender Effekt ergibt sich dadurch: Das Bild ändert seinen Aufbau vom verschwommenem ins klar sichtbare.

Transparent

Der Transparent Modus erlaubt es dem Programmierer eine Farbe in einem GIF unsichtbar zu machen, um zum Beispiel Hintergründe hervor zuheben, oder um einen Button nicht unbedingt eckig erscheinen zu lassen.

(C) Wodecki 1995-1996 - Helpfile generated by VB HelpWriter.

