**Gliederung des Lehrbriefes – Web Design - HTML**

[1.Das Internet und seine Dienste](#_1._Das_Internet)

[2. WEB Editoren](#_2._WEB_Editoren)

3. HTML Programmierung

[3.1. Grundbegriffe](#_3.1._HTML_Grundbegriffe)

* Eigenschaften
* Konzept der Tags, HTML Grundgerüst

[3.2. Text- und Absatzformatierung](#_3.2._Text_und)

* Textabsatz und Zeilenumbruch
* Absatzausrichtung
* Überschriften, Kommentare einfügen
* Zeichen formatieren
* Schrift ändern - Größe, Art, Farbe
* Sonderzeichen, Trennlinien
* Listen

[3.3. Farben in HTML](#_3.3._Farben_in)

[3.4. Grafiken](#_3.4._Grafiken)

* Kurzbeschreibung
* Größenangaben
* Grafik umranden, Hintergrundgrafik
* Beschriftung von Grafiken
* Abstand zwischen Grafik und Text

[3.5. Hyperlinks (Verknüpfungen)](#_3.5._Hyperlinks)

[3.6. Tabellen](#_3.6._Tabellen)

* Überblick und Aufbau einer Tabelle
* Breite einer Tabelle
* Breite von Spalten
* Ausrichtungen
* Gestaltung von Tabellen
* Spalten und Zeilen verbinden
* Rahmen und Abstand
* Tabellenlinien
* Farben, Hintergrund und Hintergrundbilder

[3.7. Frames](#_3.7._Frames)

* Grundlagen
* Aufteilung des Browser Fensters
* Frames anpassen
* Hyperlinks setzen
* Frames beenden

[3.8. Formulare](#_3.8._Formulare)

* Elemente von Formularen
* Eingabefelder
* Schaltflächen und Auswahllisten
* Kontrollkästchen und Optionsfelder

[4. Java Script](#_4._Java_Script)

[5. Nützliche Tools](#_Nützliche_Tools_für)

[6. Links zum Selbst-Lernen](#_Zusätzliches_zum_Selbstlernen)

# 1. Das Internet und seine Dienste

*Erarbeiten Sie sich folgende Begriffe, die bei der Arbeit und zum Verständnis des Internet notwendig sind*:

|  |  |
| --- | --- |
| *Begriff* | *Definition/ Erläuterung* |
| Internet | Weltgrößter Computerverbund  - entstanden in  -  -  - |
| TCP/ IP |  |
| Browser | browser |
| Dienste des  Internet  → www  → E- Mail  → FTP  → IP Telefonie  → Internetradio | world wide web (Standard zur Übermittlung  von Multimedia Dokumenten in HTML) |
|  |
|  |
|  |
|  |
| URL | Abkürzung für:  Aufbau: <http://www.help.de/hilfe.html>  https:// |
| Hyperlink | Verweis auf eine |
| HTML |  |

# 2. WEB Editoren

Ein **HTML-Editor** ist eine Anwendungssoftware, mit der man Webseiten mit HTML-Code erstellen und bearbeiten kann. Einige Editoren können auch mit auf Webservern eingesetzten Skriptsprachen, wie zum Beispiel PHP und JavaScript, oder Stylesheet-Sprachen wie zum Beispiel CSS umgehen.

Dabei unterscheidet man zwischen textbasierten Programmen, bei denen direkt der Quellcode editiert wird, und sogenannten WYSIWYG-Editoren, bei denen die Seite beim Bearbeiten so angezeigt wird, wie sie der Browser später darstellt. Bei Letzteren wird meist keine Kenntnis der HTML-Sprache benötigt. Häufig ermöglichen WYSIWYG-Editoren zusätzlich, den Quellcode auch direkt zu bearbeiten. Viele HTML-Editoren unterstützen Autovervollständigung und besitzen eine so genannte Syntaxhervorhebung, welche Tags im Quellcode farblich kenntlich macht.

HTML-Code kann aber auch in jedem normalen Texteditor, (z.B. Notepad) geschrieben werden.

**Quelltext Editoren unter WINDOWS**

* HTML-Editor Phase 5
* CoffeeCup HTML Editor
* CSE HTML Validator
* MarkUp Webeditor
* Macromedia HomeSite

**WYSIWYG-Editoren**

* [**SeaMonkey**](https://de.wikipedia.org/wiki/SeaMonkey)**Composer**
* [**BlueGriffon**](https://de.wikipedia.org/wiki/BlueGriffon)
* Dreamweaver
* SharePoint Designer (ausschließlich für SharePoint-Seiten)
* Visual Studio / Microsoft WebMatrix
* Microsoft FrontPage
* Microsoft Expression Web

# 3. HTML Programmierung

# 3.1. HTML Grundbegriffe

**3.1.1. Eigenschaften**

# Elemente auszeichnen

HTML bedeutet **H**yper**T**ext **M**arkup **L**anguage. Es handelt sich dabei um eine Sprache, die mit Hilfe von SGML (Standard Generalized Markup Language) definiert wird. SGML ist als ISO-Norm 8879 festgeschrieben.

HTML ist eine sogenannte Auszeichnungssprache (Markup Language). Sie hat die Aufgabe, die logischen Bestandteile eines Dokuments zu beschreiben. Als Auszeichnungssprache enthält HTML daher Befehle zum Markieren typischer Elemente eines Dokuments, wie Überschriften, Textabsätze, Listen, Tabellen oder Grafikreferenzen.

Das Beschreibungsschema von HTML geht von einer hierarchischen Gliederung aus. HTML beschreibt Dokumente. Dokumente haben globale Eigenschaften wie zum Beispiel einen Titel oder eine Hintergrundfarbe. Der eigentliche Inhalt besteht aus Elementen, zum Beispiel einer Überschrift 1. Ordnung. Einige dieser Elemente haben wiederum Unterelemente. So enthält ein Textabsatz zum Beispiel eine als fett markierte Textstelle, eine Aufzählungsliste besteht aus einzelnen Listenpunkten, und eine Tabelle gliedert sich in einzelne Tabellenzellen.

Die meisten dieser Elemente haben einen fest definierbaren Erstreckungsraum. So geht eine Überschrift vom ersten bis zum letzten Zeichen, eine Aufzählungsliste vom ersten bis zum letzten Listenpunkt, oder eine Tabelle von der ersten bis zur letzten Zelle. Auszeichnungen markieren Anfang und Ende von Elementen.

WWW-Browser, die HTML-Dateien am Bildschirm anzeigen, lösen die Auszeichnungsbefehle auf und stellen die Elemente dann in optisch gut erkennbarer Form am Bildschirm dar.

HTML ist ein sogenanntes Klartext-Format. HTML-Dateien können Sie mit jedem beliebigen Texteditor bearbeiten, der Daten als reine Textdateien abspeichern kann. Es gibt also keine bestimmte Software, die man zum Erstellen von HTML-Dateien benötigt. Zwar gibt es längst mächtige Programme, die auf das Editieren von HTML spezialisiert sind, doch das ändert nichts an der entscheidenden Eigenschaft: **HTML ist nicht an irgendein bestimmtes, kommerzielles Software-Produkt gebunden**. Diese wichtige, vielleicht sogar wichtigste Eigenschaft von HTML sollten Sie immer im Auge behalten, wenn man Ihnen erzählen will, dass Web Publishing nur mit bestimmten Software-Produkten möglich sei.

Die Klartext-Befehle von HTML sind für Maschinen **und** Menschen gedacht. Wer keine sinnlosen Vorurteile gegenüber sichtbaren Befehlen am Bildschirm hat, wird in HTML eine überraschend einfache Befehlssprache finden. Die Sprache ist Englisch, doch da die Anzahl der Befehle begrenzt ist, ist es auch ohne tiefere Kenntnisse der englischen Sprache möglich, sich in HTML hineinzudenken.

**3.1.2. Konzept der Tags**

Ein HTML-Dokument besteht einerseits aus den Textteilen, die letztendlich am Bildschirm zu sehen sein sollten und andererseits aus den Befehlen, die die obengenannten Elemente definieren bzw. formatieren.

HTML-Befehle werden als "Tags" bezeichnet. Diese Tags werden zwischen zwei spitze Klammern (Größer- und Kleiner - Zeichen, also < und >) gesetzt. Dies könnte z.B. für den Befehl "den nachfolgenden Text fett darstellen" folgend ausschauen: <B>. Die Befehle kann man sich leicht merken, wenn man sich die englischen Wörter, die "hinter" den Befehlen stecken vergegenwärtigt, so steht z.B. das "B" des Befehls <B> für "Bold".   
Es gibt nun Tags, die für eine einmalige Aktion stehen (z.B. eine Grafik einbinden oder ein Zeilenumbruch) und solche, die einen nachfolgenden Text oder Abschnitt definieren. Diese werden durch einen sogenannten Abschluss-Tag wieder deaktiviert. Ein Abschluss-Tag unterscheidet sich vom Start-Tag nur insofern, als dass zwischen dem Kleiner -Zeichen (<) und dem Befehl ein Schrägstrich (/) eingefügt wird. Für das obige Beispiel würde der Abschluss-Tag also folgend ausschauen:  
</B> (was soviel bedeuten würde wie "ab jetzt den Text nicht mehr fett darstellen").

Die Sprache HTML ist nicht "case-insensitive", d.h. dass es nicht von Bedeutung ist, ob Befehle groß oder klein geschrieben sind. So hat auch eine unterschiedliche Groß- und Kleinschreibung innerhalb eines Start- und Abschlusstags keinerlei Bedeutung.  
Sie sollten sich jedoch trotzdem eine einheitlich Schreibweise angewöhnen, denn diese erhöht die Lesbarkeit des Quellcodes erheblich.

**3.1.3. HTML Grundgerüst**

Jedes Dokument besteht in seiner Grobform aus zwei Elementen, dem Kopf-(head) und dem Körperteil (body). So würde das Grobgerüst einer HTML-Seite folgendes Aussehen haben:

<HTML>

<HEAD>

<TITLE></TITLE>

</HEAD>

<BODY>….

…….

</BODY>

</HTML>

Tags können ineinander ***verschachtelt*** werden, z.B.

<B><I>Dieser Text wird im Browserfenster fett und kursiv dargestellt</I></B>

# 

# 3.2. Text und Absatzformatierung

**3.2.1. Textabsatz und Zeilenumbruch**

Zeilenumbrüche, die man im Editor eingegeben hat, werden vom Browser ignoriert und der gesamte Text wird hintereinander dargestellt. (siehe Übung 2)

|  |  |
| --- | --- |
| <br> | Zeilenumbruch mit einem sehr geringen Abstand zur vorhergehenden Seite |
| <p>....</p> | Einleitung eines Absatzes. Dieser wird dann mit einem etwas größeren Abstand zum  vorherigen und nächsten Absatz dargestellt |

Es gibt Textpassagen, bei denen ein Zeilenumbruch nicht wünschenswert ist, z.B. „ 3 Millionen Menschen..“.

|  |  |
| --- | --- |
| <nobr>...</nobr> | Die Textpassage, die nicht getrennt werden soll, wird innerhalb der <nobr> Tags gesetzt –  no break |

**3.2.2. Absatzausrichtung**

Text muss nicht nur linksbündig ausgerichtet dargestellt werden. Er kann auch rechtbündig, zentriert oder als Blocksatz geschrieben werde.

|  |  |
| --- | --- |
| align=“Wert“  Wert:  left (Standard)  center  right  justify | Ausrichtung eines nachfolgenden Textes  linksbündig <p align=“left“> Der Zauberlehrling </p>  zentriert <p align=“center“> Der Zauberlehrling </p>  rechtbündig <p align=“right“> Der Zauberlehrling </p>  Blocksatz <p align=“justify“> Der Zauberlehrling..... </p> |

Gestalten Sie eine web Seite, in der viel Text einfließt, ist es mühsam, jeden einzelnen Textabsatz ausrichten zu müssen. Benutzen Sie in diesem Fall einen definierten Bereich!

|  |  |
| --- | --- |
| <div>...</div> | Den mit <div> umschlossenen Bereich können Sie als Ganzes ausrichten! |

**3.2.3. Überschriften**

Zur Darstellung von Überschriften erlaubt die HTML Sprache die Verwendung von sechs Überschriftstypen.

|  |  |
| --- | --- |
| <h1>...</h1>  ...  ...  <h6>...</h6> | Formatierung eines Textes als Überschrift (heading). <h1> ist dabei die höchste Ebene.  Die Überschriften können zusätzlich mit dem align Attribut ausgerichtet werden, z.B.  <h1 align=“center>........ |

**3.2.4. Kommentare einfügen**

|  |  |
| --- | --- |
| <!...Kommentar  ...> | Kommentare dienen zum Hinzufügen von zusätzlichen Informationen und sind nur im  Quelltext lesbar. Die Browser zeigen diese Abschnitte nicht an! |

**3.2.5. Zeichen formatieren**

|  |  |
| --- | --- |
| <b>...</b> | = bold – Formatierung **fett** |
| <i>...</i> | =italic – Formatierung *kursiv* |
| <u>…</u> | = underline – Formatierung untersteichen |
| <tt>…</tt> | = teletyper – Formatierung Schreibmaschineneffekt |
| <sub>…</sub> | = Formatierung tiefgestellter Text CO2 |
| <sup>...</sup> | = Formatierung hochgestellter Text 10-4 |

Um bestimmte Effekte zu erreichen, können Sie die Befehle miteinander kombinieren und schachteln.

**3.2.6. Schrift ändern (Größe, Art, Farbe)**

Die Größe des dargestellten Textes kann individuell angepasst werden. Der Befehl zum Ändern der Schrift (Schriftgröße, Schriftart, Schriftfarbe lautet „font“.

|  |  |
| --- | --- |
| <font>...</font> | = Schrift – leitet eine Änderung der Schrift des Textes ein |
|  |  |
| <font size=“Wert“>  Wert: 1...7  Wert: -2...+4  Standard: 3 | = size (Größe) – Änderung der Schriftgröße des Textes, der im Tag eingeschlossen  ist. Das Attribut erwartet einen absoluten Wert zwischen 1 und 7 oder einen relativen Wert gegenüber der Standardschriftgröße mit den Angaben +Wert oder –Wert.  <font size=“6“> Das ist ein Test </font> |
| <font face=“Schriftart“> | =face (Art) – Festlegung der Schriftart eines Textes  Die Angabe der Schriftart muss immer in Anführungszeichen erfolgen. Mehrere  Schriftarten werden durch Kommata voneinander getrennt. Der Browser versucht  die erste angegebene Schriftart zu laden. Ist diese nicht vorhanden, sucht er die  nächste im System usw.  <font face=“Bolognese,Arial,Helvetica“>Hallo </font> |
| <font color=”Wert”>  Wert: Farbname oder Farbcode (siehe Ab-  schnitt 3.3) | = Farbe – Gestaltung des Textes durch den englischen Farbnamen oder eine  6-stellige Hexadezimalzahl, die den RGB- Wert der Farbe darstellt  <font color=“black“>Schwarze Schrift</font>  Die 16 wichtigsten Farbnamen lauten: *black, navy, silver, blue, maroon, purple, red,*  *fuchsia, green, teal, lime, aqua, olive, gray, yellow, white* |

Die drei verschiedenen Textformatierungen lassen sich in beliebiger Weise miteinander kombinieren.

<font size=“+2“ face=“Arial, Helvetica”> Text in Schriftart Arial und der Schriftgröße 5</font>

<font size=“4“ color=“blue“ face=“Gothic“>Testtext</font>

**Sonderzeichen im HTML Format**

Dies ist eine weitere Variante, die Sonderzeichen darzustellen, ohne dass Sie davon abhängig sind, ob Sie den richtigen Zeichensatz eingestellt haben. Der Browser übersetzt den Zeichencode in das entsprechende Zeichen.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| HTML Code | Zeichen |  | HTML Code | Zeichen |
| &auml; | ä |  | &frac12; | ½  …Anf**&auml**;ngergl**&uuml**;ck….. - Anfängerglück |
| &ouml; | ö |  | &plusmn; | ± |
| &uuml; | ü |  | &lt; | > |
| &Auml; | Ä |  | &gt; | > |
| &Ouml; | Ö |  | &micro; | µ |
| &Uuml; | Ü |  |  |  |
| &szlig; | ß |  |  |  |
| &nbsp; | Leerzeichen |  | &copy; | © |
| &sect; | § |  | &deg; | ° |
| &frac14; | ¼ |  | &reg; | ® |

**3.2.7. Trennlinien**

Trennlinien dienen der optischen Abgrenzung von Textabschnitten, die nicht unmittelbar

zusammengehören. Eine Trennlinie erzeugt einen Absatz.

Zur individuellen Darstellung der horizontalen Trennlinie haben Sie die Möglichkeit, über verschiedene

Attribute das Aussehen zu ändern.

|  |  |
| --- | --- |
| <hr> | =horizontal ruler – Querlinie mit Schattenverlauf über die gesamte Breite des  Bildschirmes |
| width=“Wert“ | Bestimmung der Breite der Trennlinie in Prozent des Bildschirmfensters oder  pixelgenau |
| size=“Wert“ | Festlegung der Höhe der Trennlinie in Pixel. |
| noshade | Standardmäßig besitzt die Linie einen Schatten. Dieser Schatten kann (nur in  Netscape) ausgeschaltet werden. |
| align=“Art“ | Standardmäßig wird die Trennlinie zentriert ausgerichtet. Durch dieses Attribut  kann man die Linie links- oder rechtbündig ausrichten. |
| color=“Wert“ | Die Farbe der Trennlinie ist standardmäßig grau. Dies können Sie (im Internet  Explorer) Ihren Bedürfnissen anpassen. |

Beispiel:

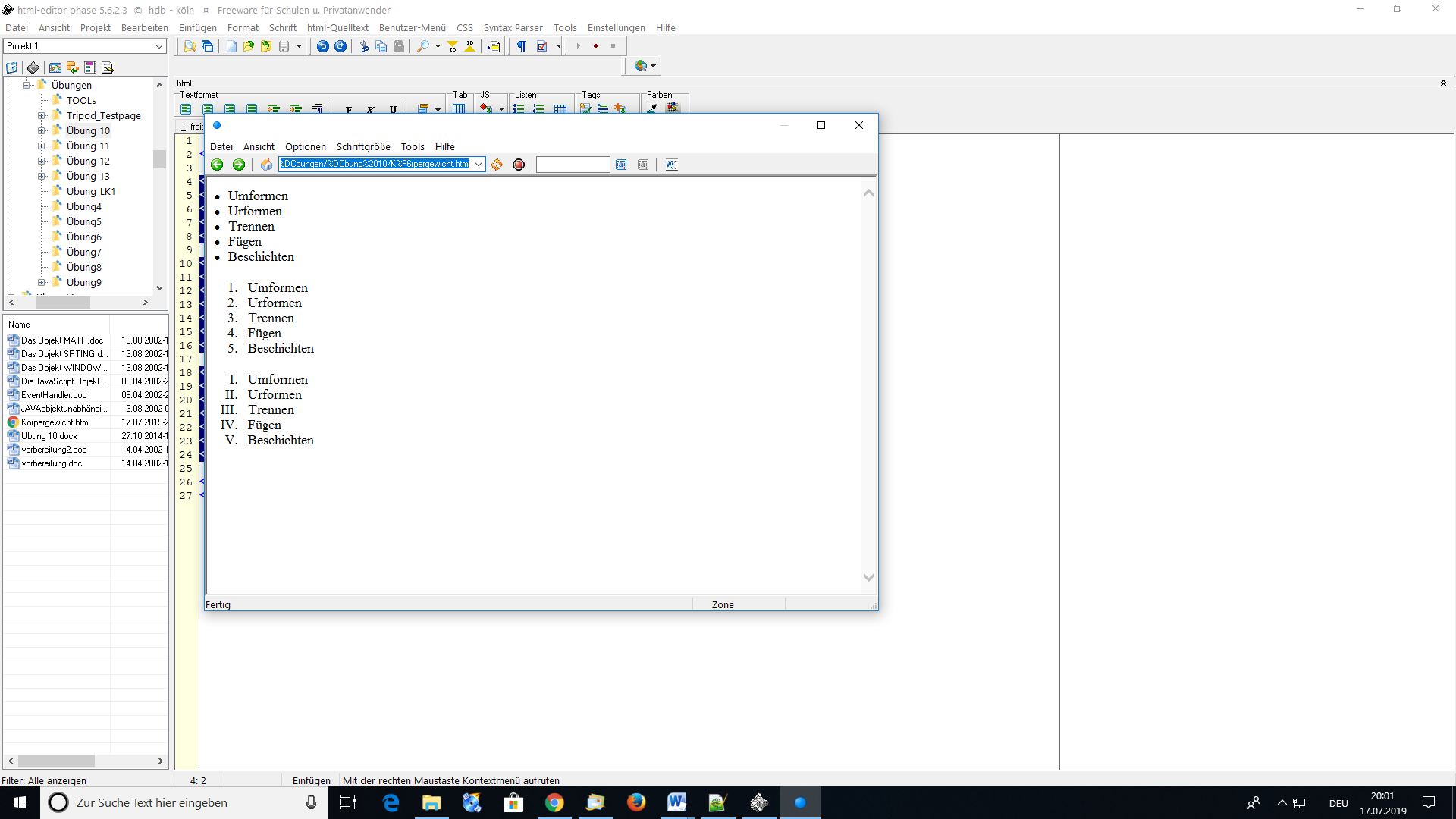
<hr>

<hr noshade width=“50%“ size=“5“ align=“right“>

**3.2.8. Listen**

**Jeder Punkt in Auflistung wird von dem HTML-TAG <li> umschlossen. Dabei steht <li> für das englische**

**„list item“ = Element der Liste. Man unterscheidet sortierte Listen <ol> und unsortierte Listen <ul>**



<li>Umformen</li>

<li>Urformen</li>

<li>Trennen</li>

<li>Fügen</li>

<li>Beschichten</li>

<ol start="1" type="1">

<li>Umformen</li>

<li>Urformen</li>

<li>Trennen</li>

<li>Fügen</li>

<li>Beschichten</li>

</ol>

<ol start="1" type="I">

<li>Umformen</li>

<li>Urformen</li>

<li>Trennen</li>

<li>Fügen</li>

<li>Beschichten</li>

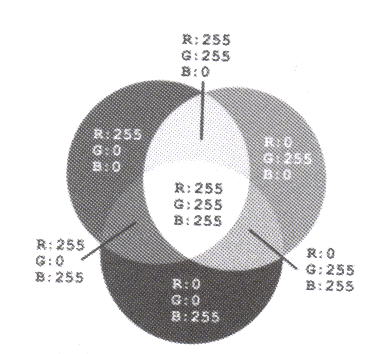
</ol>

# 3.3. Farben in HTML

Die Buchstabe **R**, **G** und **B** stehen für die englisch bezeichneten Grundfarben Reg, Green und Blue. Aus deren Mischung können bis zu 16 Mio. verschiedene Farbvarianten für die Darstellung am Bildschirm gebildet werden.

Jeder Bildpunkt am Bildschirm besteht aus drei Bytes, den Informationen über die grundfarben. Der Inhalt dieses Bytes kann Werte zwischen **0 und 255** annehmen. Dabei entsprechen diese Werte der Farbintensität (je höher, desto heller) Die Farbkombination 255,255,255 entspricht der Farbe weiß, die Kombination 0,0,0 entspricht der Farbe schwarz.

Mit Hilfe der drei Grundfarben sind demzufolge 256 x 256 x 256 = 16.777.216 Farbkombinationen möglich.

Die drei Grundfarben:

R:255 G: 0 B: 0 = rot

R: 0 G:255 B: 0 = grün

R: 0 G: 0 B:255 = blau

Mischfarben:

R: 0 G:255 B:255 = aqua

R:255 G:255: B: 0 = gelb

R:255 G: 0 B:255 = magenta

R:255 G:255 B:255 = weiß

In HTML werden die Werte in hexadezimaler

Form angegeben. Jede hexadezimale Farb-

definition hat ein Aussehen nach dem Schema:

#xxxxxx

**3F**

**C0**

**A0**

#

Rot Grün Blau

*Informieren Sie sich über das Hexadezimalsystem und die Umwandlung einer Dezimalzahl in eine Hex- Zahl!*

*Wandeln Sie die Farbe R:102 G:153 B:255 in den Hex – Code um!*

Im Zusammenhang mit dem Befehl <font color …….können Sie nun diesen Wert direkt verwenden und ermöglicht Ihnen eine viel größere Farbauswahl als bisher beschrieben!

Geben Sie die Farbwerte hexadezimal an, sind Sie außerdem browserunabhängig und haben eine freie Auswahl.

Beispiele:

<font color=“#6699FF“>Dieser Text wird farbig dargestellt</font>

<font color=“#FFD700“>Goldfarbiger Text</font>

Natürlich gelten Diese Farbinformationen auch für Hintergrundfarbe des Dokumentes und für die Farben der Links, die bereits besucht sind, die gerade angeklickt werden oder nachdem Sie den Link besucht haben.

In folgender Übersicht finden Sie die Attribute zum Ändern der Farben eines Dokumentes.

|  |  |
| --- | --- |
| <body> | Werden keine Attribute angegeben, werden die Standardfarben des Browsers  Benutzt |
|  |  |
| bgcolor=“Farbe“ | = Hintergrundfarbe des Dokumentes |
| text=“Farbe“ | bestimmt die standardmäßige Textfarbe des Dokumentes |
| link=“Farbe“ | Definition der Farbe für Verweise zu den noch nicht besuchten Seiten |
| vlink=“Farbe“ | = visited link  Definition der Farbe für Verweise zu den bereits besuchten Seiten |
| alink=“Farbe“ | = aktivated link  Definition der Farbe für Verweise die z.Z. angeklickt werden |

Beispiel:

<body bgcolor=“blue“ text=“white“ link=“#669966“ vlink=”#FFFF66“ alink=“#FFFFFF”>

(Hintergrund= blau, Textfarbe= weiß, Link= grüner Farbton, besuchter Link= gelblich, Link beim Anklicken= weiß)

# 3.4. Grafiken

Bisher haben Sie gelernt, wie Sie einen Text in einem HTML Dokument farbig gestalten können. Eine weitere Möglichkeit, Ihre web Seiten anschaulich und interessant zu gestalten, ist der Einsatz von Fotos, Bildern, Diagrammen und dekorativen Hintergründen!

|  |  |
| --- | --- |
| <img> | Einbinden einer Grafik |
| src=“Dateiname“ | Attribut src (source= Quelle) zur Angabe des Dateinamens |

Beispiel:

<img src=“flasche.jpg“>

..\..\kk.bmpDas genannte Beispiel funktioniert nur, wenn die Bilder **im gleichen Verzeichnis** stehen wie das HTML Dokument. Findet der Browser die angegebenen Daten nicht, erscheint je nach Browser folgendes Symbol:

Man kann zwar einen Grafikverweis (Pfad) in den <img> Tag einbinden, nur gibt es dann beim Finden der Daten im Online- betrieb eventuell Probleme: z.B. <img src=“Grafik/Bilder/bild5.gif“>

**Empfehlung:** alle Grafiken und sonstigen Objekte in das aktuelle Verzeichnis kopieren!!!!

**3.4.1. Kurzbeschreibung**

Binden Sie Grafiken in Ihr Dokument ein, dann sollten Sie ebenfalls eine kleine Kurzbeschreibung angeben. Der angegebene Text wird angezeigt, wenn die Grafik aus irgend einem Grund nicht gezeigt wird. Dies ist beispielsweise möglich, wenn der Besucher Ihrer Seite einen langsamen Internetzugang besitzt und die Darstellung der Grafik ausgeschaltet hat, dann sieht er statt der Grafik die Kurzbeschreibung als alternativen Text. Außerdem zeigen die Browser den alternativen Text als „Hilfe“ an, wenn man die Maus über die Grafik bewegt.

|  |  |
| --- | --- |
| alt=“Kurzbeschreibung“ | Das Attribut alt stet für alternativ und wird innerhalb des Befehls <img...>  angegeben |

Beispiel:

<img src=“flasche.jpg“ alt=“Bierflaschensammlung“>

**3.4.2. Größenangaben**

Beim Einbinden von Grafiken sollten Sie immer deren Höhe und Breite mit angeben. Dies hat den Vorteil, dass der Browser beim Anzeigen Ihrer Web- Seiten bereits weiß, wie groß die zu ladende Grafik sein wird. Er kann bereits die benötigte Fläche „reservieren“.

Benutzen Sie ggf. ein Grafikprogramm um die Originalgröße zu ermitteln. Interessante Effekte erzielen Sie, wenn Sie Größen der Grafik angeben, die sich von den tatsächlichen Proportionen unterscheiden.

|  |  |
| --- | --- |
| height=“Höhe“  width=“Breite“ | Angabe von Höhe und Breite einer Grafik in Pixel oder Prozent |

Beispiel:

<img src=“flasche.jpg“ alt=“Bierflaschensammlung“ height=“200“ width=“55“>

<img src=“flasche.jpg“ alt=“Bierflaschensammlung“ height=“50%“ width=“50%“>

**3.4.3. Grafik umranden**

|  |  |
| --- | --- |
| border=“Breite“ | Mit dem Attribut border (Rand) bestimmen Sie, ob und wie breit ein Rand  um die Grafik angezeigt werden soll.  Geben Sie dieses Attribut nicht an, oder setzen Sie border=“0“, wird kein Rand  um die Grafik gezeichnet. |

Beispiel:

<img src=“berg.jpg“ alt=“Berglandschaft“ heigth=“120“ width=“160“ border=“5“>

**3.4.4. Hintergrundgrafik**

Wenn Sie Grafiken als Hintergrund einsetzen wollen, müssen Sie dem bereits bekannten HTML Befehl <body> ein weiteres Attribut angeben:

|  |  |
| --- | --- |
| background=“Datei“  bgproperties=“fixed“ | Mit dem Attribut background ersetzen sie eine einfache Hintergrundfarbe durch  die angegebene Grafik.  Dieses Attribut bewirkt, dass der Hintergrund beim Scollen nicht verschoben wird.  Dieser Effekt nennt sich Wasserzeichen-Effekt und lässt den Text über den  Hintergrund gleiten. |

Beispiel:

<body background=“hgrund23.jpg“ bgproperties=“fixed“>

**3.4.5. Beschriftung von Grafiken**

|  |  |
| --- | --- |
| align=”Ausrichtung”  top, middle, bottom | Eine Grafik können Sie mit Hilfe des Attributes align (Ausrichtung) beschriften.  Die möglichen Werte sind: top (oben), middle (mittig), bottom (unten)  Der Text wird jeweils rechts neben der Grafik dargestellt |

Beispiel:

<img src=“berg.jpg“ align=“top“ alt=“Berglandschaft“ height=“60“ width=“80“ border=“2“> Die Alpen

**3.4.6. Abstand zwischen Grafik und Text**

Das Attribut align besitzt noch zwei weitere Werte, die jedoch nicht zur Grafikbeschriftung, sondern zum bestimmen des Abstandes zwischen Grafik und text eingesetzt werden.

|  |  |
| --- | --- |
| hspace=”Breite”  vspace=”Höhe” | Der Abstand einer Grafik zu dem Text wird in Pixel angegeben. |
| align=“Ausrichtung“  left, right | Zusätzlich ist die Ausrichtung der Grafik mit dem Attribut align unbedingt notwendig.  Für das Umfließen eines Textes um eine Grafik sind nur die Werte left (links)  sowie right (rechts) zulässig.  Lassen Sie die Attributwerte weg, wird der Text unterhalb der Grafik dargestellt. |

Beispiel:

<img src=”flasche.jpg“ height=“100“ width=“55“ align=“right“ vspace=“15“ hspace=“20“> Text

Um das Umfließen eines Textes zu beenden, wird eine Abwandlung des Befehls <br> angewendet:

<br clear=”all“>

# 

# 3.5. Hyperlinks

Querverweise im Internet, bekannt unter dem begriff „Hyperlinks“ oder „Links“ dienen dazu, mehrere Web- Seiten miteinander zu verbinden. Diese Möglichkeit der Verknüpfung von verschiedenen Angeboten und Informationen brachte den riesigen Erfolg des Internet mit sich.

Hyperlinks sind Querverweise auf andere Seiten, die viele tausend Kilometer entfernt sein können oder es sind Querverweise innerhalb eines Dokumentes.

Einen Hyperlink erkennen Sie immer daran, dass der Pfeil der Maus zu einer Hand wird, wenn er sich über einem Link befindet.

In diesem Abschnitt beschäftigen wir uns in Theorie und Praxis mit

* Hyperlinks innerhalb einer Datei
* Arten der Hyperlinks (relativ, absolut)
* Ziele von Hyperlinks
* Verweis zu einer E- Mail Adresse
* Verweise auf Elemente (Grafiken, Sound, Video)

**3.5.1. Hyperlinks innerhalb einer Datei**

Wenn Sie ein sehr umfangreiches Textdokument erstellen, erweist sich eine Übersicht zum Inhalt am Beginn sehr günstig. Von dort aus sollte man an die verschiedenen zutreffenden Stellen „springen“ können!

Sie müssen dafür zwei Arbeitsschritte durchführen:

1. ein Verweisziel (Anker) definieren
2. einen Querverweis zu dem definierten Verweisziel setzen

|  |  |
| --- | --- |
| <a name=“Bezeichnung“> | An die Stelle, zu der ein Verweis führen soll (Markierung für den Verweis) wird  dieser Befehl gesetzt. Man sagt dazu auch, einen Anker setzen. |
| <a ...>Text</a> | Der Befehl a steht für Anker und leitet einen Verweis ein. Der Text gibt nähere  Informationen darüber, was den Benutzer erwartet, wenn er auf den Hyperlink  klickt. Der Text des Hyperlinks erscheint unterstrichen auf dem Bildschirm |
| <href=“#Ziel“> | Das Attribut verweist auf einen Anker innerhalb der angezeigten web- Seite, die  bei einen Mausklick aufgerufen werden soll.  Der Querverweis muss mit <a name=...> markiert worden sein. |

Beispiel:

<a href=“#sonnenblume“>Verweis auf den Text über die Sonnenblumen</a>

........

<a href name=“sonnenblume“><h3>Die Sonnenblume – Zier- und Futterpflanze</h3></a>

**3.5.2. Verweis zu einer anderen Web- Seite**

Bieten Sie zu viele Informationen auf einer Seite, so dass diese eine Länge von mehreren geschriebenen Seiten erreicht, dauert das Laden der Seite sehr lange und der Besucher wird genervt Ihr Internet Angebot verlassen. Es ist also nötig, die Informationen auf mehrer Seiten zu verteilen oder zu anderen Web- Seiten zu „springen“.

Der Befehl, den Sie einfügen müssen, um auf eine andere Seite zu gelangen, lautet:

|  |  |
| --- | --- |
| <a href=“Ziel“> | Verweis auf eine Web- Seite, die bei Mausklick auf den Hyperlink angezeigt  werden soll. Das „Ziel“ ist der Name der Datei, die aufgerufen werden soll.  Das Ziel kann dabei eine absolute Adresse (URL) sein, eine Datei in einem  anderen Verzeichnis innerhalb der Homepage oder eine andere Datei innerhalb  des gleichen Verzeichnisses |

Beispiele:

<a href=”<http://www.tu-chemnitz.de/informatik/mueller/grafik.htm>“>Hyperlinktext</a>

<a href=“arbeitsgemeinschaften/musik.htm”>Hyperlinktext</a>

<a href=“seite2.htm“>Link zu Seite 2</a>

Verlinkte Dokumente müssen nicht unbedingt in dem Browser Fenster geöffnet werden, in dem sie aufgerufen werden.

|  |  |
| --- | --- |
| target=“Fenstername“ | Das angeforderte Dokument wird in dem Browser- Fenster mit dem angegebe-  nen Namen angezeigt. Existiert das Fenster mit dem Namen noch nicht, wird  ein neues Browser- Fenster geöffnet und es erhält den Namen |
| target=”\_blank” | Das Dokument wird in einem neuen leeren Fenster geöffnet |

Beispiele:

<a href=“<http://www.lkg.de.cx>“ target=“schule“> Homepage der Schule </a>

<a href=“<http://www.quelle.de>“ target=“\_blank“>Versandhaus Quelle </a>

**3.5.3 Verweis an eine bestimmte Stelle einer anderen Web- Seite**

Genauso wie Sie innerhalb einer Seite auf eine bestimmte Stelle verweisen können, ist dies auch bei verweisen auf eine andere Web- Seite möglich.

|  |  |
| --- | --- |
| <a href=“datei.htm#marker“>Hyperlinktext</a> | Lade die Seite *datei.htm*, suche die Stelle *marker* und  springe dorthin |

Beispiel:

|  |  |
| --- | --- |
| Code der seite1.htm  ...  <a href=“seite2.htm#eins!“>1. Verweis</a>  <br>  <a href=“seite2.htm#zwei“>2. Verweis</a> | Code der seite2.htm  <a name=“eins“>Beispieltext:Fortune plango vulnera stilantibus ocrellis, quod sua michi munera subtrahit rebellis...</a><br>  <a name=“zwei“>Beispieltext: ..........</a> |

**3.5.4. Verweis zu einer E- Mail Adresse**

Möchten Sie, dass die Besucher der Web-Seite Ihnen eine E-Mail schicken können, so benutzen Sie das Attribut *mailto!*

|  |  |
| --- | --- |
| mailto:Adresse | Wenn der Besucher der Seite auf solch einen Hyperlink klickt, öffnet sich das Standard E-Mail Programm und er kann eine E-Mail an den angegebenen Empfänger schicken. |

Beispiel:

<a href=“<mailto:lkg_annaberg@t-online.de>“> Nehmen Sie mit uns Kontakt auf</a>

**3.5.5. Grafiken als Hyperlink**

Sie werden schon festgestellt haben, dass im Internet häufig Grafiken als Links benutzt werden- treten somit als wichtiges Navigationsmittel in Erscheinung. Das Vorgehen ist einfach.

|  |  |
| --- | --- |
| <a href=“Datei des Verweisdokumentes“><img src=“Bildname“></a> | der Hyperlink wird einfach um eine eingebettete Grafik gelegt. |

Beispiel:

<a href=“datei.htm“><img src=“bild.jpg“></a>

Mit Hilfe des Attributes *border* müssen Sie auch beim Einsatz der Grafik als Hyperlink die Umrandung abschalten!

<a href=“grafik.htm“>

<img src=“flasche.jpg“ alt=“Hyperlink“ height=“100“ width=“28“ align=“left“ vspace=“5“ ***border=“0“***>

</a>

**3.5.6. Verweis auf Grafiken und andere Elemente**

Mit einem Hyperlink müssen Sie nicht unbedingt nur auf andere Web- Seiten verweisen. Auch ein Link auf Textdateien, Grafiken, Videos, Musikdateien oder Programme sind möglich. Der Browser erkennt dabei an der Endung der aufgerufenen Datei, welche Aktion er ausführen soll.

Existieren die entsprechenden Programme zum Öffnen der Dateien auf Ihrem Rechner, so werden die Dateien *automatisch* gestartet z.B. Musik abgespielt, ein Download gemacht, ein Video gezeigt.

Beispiele:

.txt - .docx -

.xls - .pdf -

.jpg - .gif -

.mpg - .avi -

.wav - .mp4 -

.mp3 - .zip -

.exe -

# 3.6. Tabellen

**3.6.1. Überblick und Aufbau einer Tabelle**

Tabellen sind ein wichtiges Hilfs- und Gestaltungsmittel, um Daten in einer Gitterstruktur übersichtlich darzustellen. Die Tabellenlinien können auch ausgeblendet werden, um z.B. mehrere Bilder oder Textelemente nebeneinander zu platzieren.

Zum Erstellen von Tabellen ist eine Vielzahl von Tags nötig

|  |  |
| --- | --- |
| <table>......</table> | Befehl zum Einleiten und Abschließen einer Tabelle |
| border=”wert” | Umrandung der Tabelle in Pixel.  Der Wert 0 erstellt eine Tabelle ohne Rand. |
| <tr>  <td>  </td>  </tr> | <tr> definiert eine Zeile (tr=table row)  <td> legt die Zelle fest, die in der definierten Zeile dargestellt werden soll  (td=table data) |
| <th>....</th> | Der Inhalt dieser Zelle wird als Spaltenüberschrift deklariert. Man kann diesen  Tag anstelle <td> verwenden, der Text wird dann fett hervorgehoben |
| <caption>  </caption> | Erstellt eine Tabellenüberschrift.  Mit dem bekannten Attribut *align* ist eine Ausrichtung *left, right, top, bottom*  möglich |

<table>

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| <tr> | <td>Inhalt</td> | <td>Inhalt</td> | <td>Inhalt</td> | </tr> |
| <tr> | <td>Inhalt</td> | <td>Inhalt</td> | <td>Inhalt</td> | </tr> |

</table>

Die Anzahl der Zellen in der ersten Zeile muss unbedingt mit der Anzahl der Zellen in der nächsten Zeile übereinstimmen.

Es ist hier besonders wichtig, dass Sie jedem geöffneten Tag einen schließenden Tag zuweisen!

Einfache Tabellen haben die Eigenschaft, die Größe ihrer Zellen automatisch dem Text, der in ihnen steht, anzupassen. Ein Umbruch des Textes erfolgt erst, wenn die Gesamtbreite der Tabelle die Breite des Browserfensters überschreitet. Eine einfache Tabelle kann maximal die Breite des Fensters annehmen. Dies können Sie einfach testen, indem Sie die Breite Ihres Browser Fensters verkleinern. Die Tabelle wird neu formatiert und ihr Inhalt umgebrochen.

<table align=”center“ border=“1“ width=90%>

<caption align=”left”><b>Das bin ich</caption>

<tr>

<th>&nbsp;</th>

<th>Daten</th>

</tr>

<tr>

<td>Nachname></td>

<td>Mittermeier</td>

</tr>

<tr>

<td>Vorname</td>

<td>Hansi></td>

</tr>

.....

....

</table>

Tabellenzellen dürfen auch leer sein. Sie können einfach schreiben <td></td>. Bei einigen Browsern kommt es dann jedoch zu Fehlinterpretationen- besser wäre, die geben wenigstens ein Leerzeichen ein! **<td>&nbsp;</td>**

|  |  |
| --- | --- |
|  | **Daten** |
| Nachname | Mittermeier |
| Vorname | Hansi |
| Geburtsdatum | 29.02.1952 |
| Wohnort | Auerbach |

**3.6.2. Breite einer Tabelle**

|  |  |
| --- | --- |
| width=“Wert“  Angabe in Pixel oder  Prozent | Die Angabe *width* innerhalb des <table> Tags setzt die Breite der Tabelle fest,  ohne den Inhalt der Tabelle zu beachten |

Beispiel:

<table border=”1” width=”30%”> oder <table border=”1” width=”1000”>

Es empfiehlt sich, die Breite einer Tabelle grundsätzlich in % anzugeben, da die Internetnutzer unterschiedliche Bildauflösungen benutzen. Der Nachteil ist, dass sich das Grundlayout der web- Seite ändert, sobald die Größe des Browserfensters geändert wird.

**3.6.3. Breite von Spalten**

Wie die Breite der Tabelle können Sie auch gezielt die Breite jeder einzelnen Zelle beeinflussen. Dabei besitzen die Zellen einer Spalte dieselbe Breite, d.h. man braucht die Breite einer Spalte nur in der ersten Zeile anzugeben.

|  |  |
| --- | --- |
| width=“Wert“  Angabe in Pixel, \* oder  Prozent | Die Angabe *width* innerhalb der <td> oder <th> Tags setzt die Breite der Spalte fest,  ohne den Inhalt der Zelle zu beachten.  Das Zeichen \* ist ein Spezialfall und bedeutet: Rest der übrigbleibt |

**• Bei prozentualer Angabe der Breite muss die Summe aller Spaltenbreiten 100% sein!**

**• Die Breitenangaben sind nur wirksam, wenn der Inhalt einer Zelle die Breitenangabe nicht überschreitet!**

Beispiele:

<table align=“center“ border=“1“ width=“75%“>

<tr><td width=”20%”>20%</td>

<td width=”30%”>30%</td>

<td width=”50%”>50%</td>

</tr>

</table>

|  |  |  |
| --- | --- | --- |
| 20% | 30% | 50% |

<table align=“center“ border=“1“ width=“50%“>

<tr><td width=”20%”>20%</td>

<td width=”\*”>Rest</td>

<td width=”40%”>40%</td>

</tr>

</table>

|  |  |  |
| --- | --- | --- |
| 20% | Rest | 40% |

**3.6.4. Ausrichtung**

Tabellen sind standardmäßig links ausgerichtet.

|  |  |
| --- | --- |
| align=“Ausrichtung“  *left, right, center,*  *justify* | Horizontales Ausrichten des Textes innerhalb der Zelle  Sie erweitern den Tag <td> um das Attribut *align*. Es kann dabei die vier  verschiedenen Ausrichtungen annehmen |
| valign=“Ausrichtung“  *top, middle, bottom* | Vertikales Ausrichten des Textes innerhalb der Zelle  Sie erweitern den Tag <td> um das Attribut v*align*. Es kann dabei die vier  verschiedenen Ausrichtungen annehmen |

Beispiel:

Möchten Sie alle Zellen einer Zeile gleich ausrichten, reicht die Angabe innerhalb des <tr>Tags! Ohne jegliche Angabe zur Positionierung des Textes in der Zelle wird dieser linksbündig und vertikal mittig dargestellt!

<tr align=“right“ valign=”middle”>In diesem Beispiel ......</tr>

**3.6.5. Gestaltung von Tabellen**

Zur anspruchsvollen Darstellung von Datenmengen oder zur Darstellung besonderer Merkmale kann man Tabellen nach bestimmten Merkmalen gruppieren.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | **Verbundene Zeilen und Spalten** | | | | | |  | |
|  |  |  |  |
|  |  |  |  |  |  |  |  |  | |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  | |  | | | | | |  | |
|  | |  | | | | | |  | |

**3.6.5.1. Spalten und Zeilen verbinden**

Tabellen mit verbundenen Zellen sind sehr fehleranfällig, d.h. Fehler sind schwer zu lokalisieren. Es ist sinnvoll, eine Tabelle mit allen Zellen zu erstellen und **danach** erst die Spalten und Zeilen zu verbinden!

|  |  |
| --- | --- |
| colspan=“Anzahl“ | Das Attribut *colspan* (column span= Spalte umfassen) gibt die Breite der  jeweiligen Spalte an. Die Breite bezieht sich dabei auf die Anzahl der Zellen,  über die sich die Spalte erstrecken soll!  Dieses Attribut kann mit dem <th> und dem <td> Tag angegeben werden |
| rowspan=”Anzahl” | Das Attribut *rowspan* (row span= Zeile umfassen) gibt die Höhe der jeweiligen  Spalte an. Die Höhe bezieht sich dabei auf die Anzahl der Zellen,  über die sich die Zeile erstrecken soll!  Dieses Attribut kann mit dem <th> und dem <td> Tag angegeben werden |

Beispiel:

<table border=“1“ width=”200“> <table border=“1“ width=”200“>

<tr> <tr>

<td width=”100”>Zelle 1</td> <td width=”100”>Zelle 1</td>

<td width=”100”>Zelle 2</td> <td width=”100” rowspan=“2“>Zelle 2 und 4</td>

</tr> </tr>

<tr> <tr>

<td colspan=“2“>Zelle 3 und 4</td> <td> Zelle 3</td>

</tr> </tr>

|  |  |
| --- | --- |
| Zelle 1 | Zelle 2 |
| Zelle 3 und 4 | |

|  |  |
| --- | --- |
| Zelle 1 | Zelle 2 und 4 |
| Zelle 3 |

**3.6.5.2. Rahmen und Abstand**

Standardmäßig werden Texte oder Objekte innerhalb von Textbegrenzungen dargestellt. Mit zusätzlichen Tabellenattributen lässt sich das Aussehen von Tabellen verändern!

cellspacing

border

cellpadding

Die folgenden Angaben (Attribute) müssen im Befehl <table> erfolgen!

### Zellinhalt

|  |  |
| --- | --- |
| cellspacing=“Pixel“ | Der Zellzwischenraum (*cellspacing*) vergrößert den Abstand der einzelnen Zellen. |
| cellpadding=“Pixel“ | Der Zellinhaltsabstand (*cellpadding)*setzt einen Zwischenraum zwischen dem  Inhalt und der Begrenzung der Zelle. |
| border=“Pixel“ | Dieses Attribut gibt der Tabelle einen Rahmen. Setzt man border=“0“, wird kein Rahmen gezeichnet |
| frame=“Art“  *viod, box, above,, below, hsides, vsides, lhs, rhs* | Das Attribut erlaubt speziellere Angaben zum Zeichnen eines Rahmens um die  Tabelle, z.B. *lhs*- Rahmen an der linken Seite der Zelle |

Beispiele: siehe Übung 10

**3.6.5.3. Tabellenlinien**

|  |  |
| --- | --- |
| rules=“Art“  *none, rows, cols, all* | Dieses Attribut erlaubt spezielle Angaben zum optischen Trennen von Zeilen und Spalten z.B. bei *rows* Trennlinien zwischen jeder Zeile |

**3.6.5.4. Farben, Hintergrund und Hintergrundbilder**

|  |  |
| --- | --- |
| bordercolor=“Farbe“  bordercolorlight=“Farbe“  bordercolordark=”Farbe” | Innerhalb der Anweisung <table> erzwingen diese Angaben die angegebene  Umrandungsfarbe um die gesamte Tabelle. Die Breite des Rahmens muss dabei  mindestens 1 Pixel sein!  In den Tags <tr>, <th> und <td> werden die entsprechenden Zeilen oder Zellen  mit der entsprechenden Farbe umrandet.  Bei der Wahl von hellen und dunklen Bestandteilen einer Farbe lassen *border-*  *colorlight* und *bordercolordark* den Rahmen dreidimensional erscheinen! |
| bgcolor=”Farbe” | Die Farbe der gesamten Tabelle <table>, einer Zeile <tr> oder einer einzelnen  Zelle <th> oder <td> setzen Sie mit diesem Attribut |
| background=“Dateiname“ | Das Hintergrundbild einer Tabelle <table>, einer Zeile <tr> oder einer einzelnen  Zelle <th> oder <td> setzen Sie mit diesem Attribut |

Tipps für die Erstellung von Tabellen:

• Für die horizontale Ausrichtung gilt: Zellen haben Vorrang gegenüber Zeilen <td> vor <tr>

• Für die vertikale Ausrichtung gilt: Zeilen haben Vorrang gegenüber Zellen <tr> vor <td>

• Die Eigenschaften von einzelnen Zellen haben Vorrang vor den höheren Elementen <td> vor <table>

• Eigenschaften von Zellinhalten haben Vorrang vor den Eigenschaften der Zellen

# 3.7. Frames

**3.7.1. Grundlagen**

Ein Frame teilt das Browser Fenster physisch in verschiedene rechteckige Bereiche- sie bieten damit die Möglichkeit, statt einer Seite gleichzeitig mehrer Seiten anzuzeigen und diese unabhängig voneinander zu steuern. Nachteilig könnte sich die längere Ladezeit der Web Seite auswirken.

Der Entwurf des Layout für die Web Seite auf dem Papier steht natürlich am Anfang jeder Homepagegestaltung. Jede „gute“ Web Seite enthält Frames in irgendeiner Form!!!

Titel

Navi-

gation

Textbereich

Kontakt

Titel

Navi-

gation

Textbereich

Die Definition eines Frame erfolgt in einer separaten Datei. Diese Datei enthält Informationen zu Größe und Inhalt der Framefenster.

Grundgerüst: (Ein Frameset enthält kein <body>!)

<html>

<head>

<title>Ein Frameset- Test</title>

</head>

**<frameset………..>**

<! Definition des Frame z.B. rows=”20%,\*”>

**</frameset>**

</html>

**3.7.2. Aufteilung des Browser Fensters**

Das Browserfenster wird in Spalten oder Zeilen oder Beides geteilt:

|  |  |
| --- | --- |
| cols=“Aufteilung“  rows=“Aufteilung“ | Die beiden Attribute *cols* (columns = Spalten) und *rows* (Zeilen) legen fest,  in welche Gruppen das Browser- Fenster aufgeteilt werden soll.  Die Werte, die auch hier entweder in Prozent in Pixel oder mit \* angegeben  werden können, müssen durch ein Komma voneinander getrennt werden.  Die Anzahl der angegebenen Werte bestimmt dabei die Anzahl der Spalten  und Zeilen. Die Angabe der Werte in % ist empfehlenswert, um flexibel zu bleiben. |

Beispiele:

<frameset cols=“100,\*,200“>

<frameset rows=“2\*,\*“ > (Fenster 1 ist doppelt so groß wie Fenster 2)

<frameset rows=“25%,45%,30%“>

<frameset cols=“160,\*”>

<frameset rows=”10%,90%”>

</frameset>

</frameset>

Nun muss festgelegt werden, welche Dateien innerhalb der Frames gezeigt werden. Beachten Sie bitte genau Groß- und Kleinschreibung!!!

|  |  |
| --- | --- |
| <frame> | Mit diesem Befehl wird das Fenster in die entsprechenden Bereiche aufgeteilt.  Die jeweiligen Bereiche sind in diesem Moment noch leer. |
| src=“Dateiname“ | Das Attribut src /source)= Quelle) mit der Angabe einer HTML Datei sorgt dafür,  dass das Frame mit Informationen gefüllt wird. |
| name=“xyz“ | Das Frame **muss** einen Namen besitzen, um beispielsweise in einem linken  Frame eine web Seite aufzurufen, die im rechten Fenster angezeigt werden soll. |

Beispiel:

<html>

ueberschrift.htm

aus-

wahl.htm

hauptteil.htm

<head>

<title> Frametest</title>

</head>

<frameset cols=”20%,\*”>

<frame name=”Links” src=”auswahl.htm”>

<frameset rows=“90,\*“>

<frame name=“Oben“ src=“ueberschrift.htm“>

<frame name=“Unten“ src=“hauptteil.htm“>

</frameset>

</frameset>

</html>

**3.7.3. Frames anpassen**

Frames können in Breite und Höhe „manuell“ verändert werden. Dies führt jedoch oft zu Überschneidungen mit dem Inhalt.

Sie können solche Möglichkeiten unterbinden durch:

1. Bildlaufleisten b) unveränderbare Frames c) Anlegen von Umrandungen
2. Bestimmung des Frameabstandes e)Festlegung der Umrandungsbreite
3. Bestimmung der Umrandungsfarbe g) Erstellung von Framerändern

|  |  |
| --- | --- |
| scrolling=“yes“/ „no“/  “auto“ | Mit diesem Attribut können Sie das Einschalten der Bildlaufleiste für das  jeweilige Frame beeinflussen. (Bei *auto* entscheidet der Browser selbst, wann  die BLL angezeigt werden muss) |
| noresize | Diese Angabe unterbindet die Größenänderung des angegebenen Frames. |
| frameborder=“Wert“ | Ein- (Wert=1) und Ausschalten (Wert=0) der Umrandung eines Frames |
| framespacing=“Pixel“ | Die Angabe des Frame- Abstandes muss auch innerhalb des Frameset |
| border=“Pixel“ | Die Angabe der Breite der Umrandung in Pixel als 2- dimensionale Umrandung  ab dem Wert 1 muss auch innerhalb des Framesets erfolgen. |
| bordercolor=“Farbe“ | Die Farbe der sichtbaren Umrandung kann jede beliebige vordefinierte Farb-bezeichnung oder hexadezimale Farbangabe sein. |
| marginheight=“Pixel“  marginwidth=“Pixel“ | Mit diesen Attributen kann man die Randhöhe und die Randbreite zum linken  und oberen Framerand bestimmen. (Für jedes Frame einzeln!) |

Beispiel:

<frameset name=“Mitte“ src=“hauptteil.htm“ scrolling=“yes“ noresize frameborder=“1“ framespacing=“5“ bordercolor=“yellow“>

**3.7.4. Hyperlinks setzen**

Für das Setzen von Hyperlinks zu einem andern Frame gilt das gleiche Schema wie für Hyperlinks in HTML. Beachten Sie bitte in diesem Zusammenhang die Benennung der einzelnen Framesets im Punkt 3.7.2.!

|  |  |
| --- | --- |
| target=“Name“ | Das Verweisziel target enthält den Namen des Frames, in dem der Hyperlink  geöffnet werden soll. Das Ziel ist dabei innerhalb des Hyperlink- Befehls  <a href=“Dateiname, die im Frame geöffnet werden soll“ ***target***=“Framename“>...  </a> anzugeben.  Existiert der Verweisname nicht, wird der Hyperlink im aktiven Frame angezeigt |

**3.7.5. Frames beenden**

Zum Schließen eines Frames gibt es vier verschiedene Zielfenster:

|  |  |
| --- | --- |
| target=“\_blank“  target=“\_self“  target=”\_parent”  target=”\_top” | Das Dokument wird in einem neuen, leeren Fenster geöffnet  Das Dokument wird in dem Fenster geöffnet, in dem der verweis steht  Das Dokument wird in das direkt übergeordnete Frameset geladen  Das gesamte Frameset wird geschlossen und das Dokument im gesamten  Browserfenster angezeigt. |

# 3.8. Formulare

Mit Hilfe eines Formulars können Sie dem Besucher Ihrer web Seite mehr bieten als eine Rückantwort per E- Mail, nämlich Auswahlmöglichkeiten oder ein Feedback (Lob oder Kritik).

Formulare bestehen aus mehreren Elementen, müssen aber zunächst als zusammengehöriges Objekt definiert werden. Dann können Attribute bestimmt werden, wie das Formular verarbeitet werden soll.

|  |  |
| --- | --- |
| <form …..>  </form> | Der Befehl *<form>* (Formular) leitet ein Formular ein. Die späteren Schaltflächen, Eingabefelder und Auswahllisten innerhalb eines Tags sind die Elemente dieses Formulars. |

Formulare können nicht ineinander verschachtelt werden, man muss sie ggf. nacheinander anlegen.

Zur Definition eines Formulars gehören noch einige Attribute, die bestimmen, wie das Formular verarbeitet werden soll:

|  |  |
| --- | --- |
| method=”art”  art: get, post | Das Attribut *method* bestimmt, ob die Daten des Formulars nur per E- Mail  versendet (*post*) oder versendet und mit einem Programm, das auf dem Server  läuft, weiterverarbeitet wird (*get*) |
| action=”ausführung”  mailto:Programmaufruf | Mit *action* geben Sie an, welches Programm auf dem Server, auf der Ihre Web  Seite liegt, nach abschicken der Daten ausgeführt werden soll, z.B. Weitergabe  an eine Suchmaschine. |
| enctype=”format” | Mit *enctype=“text/plain“* werden die Daten des Formulars formatiert und Sie  können Sie selbst besser lesen (empfehlenswert!!!) |

Beispiel:

<form action=”<mailto:lkg_annaberg@t-online.de>“ method=“post“ enctype=“text/plain“>……………

… </form>

**3.8.1. Elemente von Formularen**

Alle folgende Elemente müssen zwischen <form> und </form> platziert werden, um beim versenden übergeben zu werden.

Wir behandeln im folgenden Eingabefelder, Schaltflächen, Auswahllisten, Kontrollkästchen, Optionsfelder und grafische Schaltflächen:

**3.8.1.1. Eingabefelder**

Eingabefelder dienen zur Eingabe von text, Zahlen, Datumsangaben, Postleitzahlen etc. im Textformat

|  |  |
| --- | --- |
| <input type=”text”….>  <input type=“password“> | Dieser Befehl leitet das Feld ein und initialisiert ein einzeiliges Eingabefeld  Dieses Attribut stellt die Eingaben als Sternchen dar und ist somit für die  Eingabe von Passworten geeignet. |
| name=“Bezeichnung“ | Jedes Formularelement muss einen internen Bezeichnernamen besitzen, um  eine spätere Zuordnung zu ermöglichen. Benutzen Sie treffende Bezeichner, um selbst die Übersicht zu behalten, wenn die Daten an Sie gemailt werden.  Verwenden Sie ggf. mehrere Wörter, bitte aber keine Sonderzeichen! |
| size=“Zeichen“  maxlength=“Zeichen“ | Dieses Attribut bestimmt die Breite der Anzeige für das Eingabefeld.  Mit *maxlength* geben Sie an, wie viele Zeichen maximal in das Feld eingegeben  werden dürfen, z.B. 5 für PLZ.  Ist die maximale Länge größer als die Breite des Eingabefeldes, wird automa-  tisch gescrollt. |
| value=“inhalt“  readonly | Das Eingabefeld kann mit einem „Vorgabeinhalt“ versehen werden  Dieses Attribut sperrt das Eingabefeld vor evtl. editieren, das ist sinnvoll, wenn  Sie einen Wert vorgeben wollen, der nicht verändert werden soll. |
| <textarea …..></textarea>  rows=“zeilen“  cols=“spalten“ | Für die Eingabe längerer Texteingaben stehen Ihnen mehrzeilige Eingabefelder zur Verfügung.  Mit diesen Attributen können Sie die sichtbare Zeilenanzahl (*rows*) und die Breite  des Feldes (*cols*) festlegen |

Beispiel:

<form>

<p> Name : <input type=“text“ name=“name“ size=“20“></p>

<p> Wohnort : <input type=“text“ name=“ort“ size=“15“></p>

<p> URL : <input type=“text“ name=“url“ size=“25“ value=”http://www”></p>

<p> Beitrag : <input type=“text“ name=“beitrag“ size=“8“ value=”15,00 €” readonly></p>

<p> Passwort: <input type=“password“ name=“pwd“ size=“8“></p>

<p> Bemerkungen: <textarea name=“bemerkungen“ rows=”5” cols=”10”></textarea></p>

</form>

**3.8.1.2. Schaltflächen**

In HTML stehen standardmäßig zwei Schaltflächen zum Versenden bzw. Zurücksetzen von Formulardaten zur Verfügung, die je nach Wunsch auch beschriftet werden können.

|  |  |
| --- | --- |
| <input type=”submit”>  <input type=“reset“> | Dieser Befehl erzeugt eine Schaltfläche zum Versenden der Daten. Die Adresse,  an welche die Daten versendet werden, ist am beginn des Formulars festzulegen!    Diese Schaltfläche dient zum Zurücksetzen der Daten. Die bisher eingegebenen  Daten werden nicht versendet. |
| value=”Beschriftung” | Die Beschriftung der Schaltflächen übernimmt der Browser, z.B. im Internet  Explorer sind es „Anfrage senden“ und „Zurücksetzen“.  Mit dem Attribut *value* haben Sie selbst Einfluss auf die Beschriftung der beiden Schaltflächen. |

Beispiel:

<input type=“submit“ value=“Daten bitte absenden“>

<input type=“reset“ value=“Eingaben löschen“>

Möchten Sie die Schaltflächen zusätzlich mit einer Grafik versehen, so stehen Ihnen folgende Möglichkeiten zur Verfügung:

|  |  |
| --- | --- |
| <input type=”image” name=”bezeichnung” src=”grafik”> | Die standardmäßige Schaltfläche zum Absenden wird als grafische Darstellung  definiert |



Beispiel:

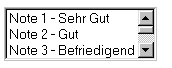
<input type=“image“ name=“Versenden“ src=“grafik1.gif“>

**3.8.1.3. Auswahllisten**

Sie können dem Besucher eine Liste mit festgelegten Werten anbieten, aus der er den Eintrag auswählt, den er für angemessen hält.

|  |  |
| --- | --- |
| <select ....>  </select> | *select* leitet eine Auswahlliste ein. Alle Daten der Liste müssen sich innerhalb  der Tags befinden. |
| name=“Bezeichnung“ | Die Auswahlliste muss einen Namen besitzen, damit der ausgewählte Eintrag  An sie übergeben werden kann. |
| <option>Eintrag</option> | Mit *<option>...*füllen Sie die Auswahlliste mit Daten. Beim Verschicken der  Formulardaten wird nur der Text des Eintrages übertragen, den der Anwender  ausgewählt hat. |
| size=“höhe“ | Die Höhe der Anzeige wird in Zeilen angegeben. Ist die Menge der Einträge grö-  ßer als die Höhe der Anzeige, werden die vertikalen Bildlaufleisten eingeschaltet. |
| selected | Beim Erstellen und Anzeigen einer Auswahllist ist immer der erste Eintrag ausgewählt. Möchten Sie dies ändern, setzen Sie *<option selected>....* |
| multiple | Multiple (mehrfach) erlaubt die Auswahl mehrerer Einträge innerhalb des *select*  Tags. Um mehrer Einträge auswählen zu können müssen die Tasten SHIFT  oder STRG betätigt werden! |

Beispiele:

size=“3“ <select multiple....

<form>

<p><select name="liste" size=”3”>

<option>Note 1 - Sehr Gut</option>

<option>Note 2 - Gut</option>

<option>Note 3 - Befriedigend</option>

<option>Note 4 - Ausreichend</option>

<option>Note 5 - Mangelhaft</option>

<option>Note 6 - Ungenügend</option>

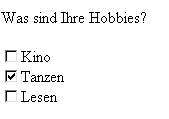
</select> </p>

</form>

**3.8.1.4. Kontrollkästchen und Optionsfelder**

|  |  |
| --- | --- |
| <input type=“checkbox“  name=”Bezeichnung”  <input type=“radio“  name=”Bezeichnung”  value=”wert”  checked | Es wird ein rechteckiges Feld mit einer Kotrollauswahl im Dokument eingefügt.  So kann der Anwender keine, eine oder mehrere Möglichkeiten einer Auswahl  selektieren.  Es wird ein Optionsfeld im Dokument eingefügt.  So kann der Anwender **nur eine** Möglichkeit einer Auswahl selektieren, z.B.  Zahlungsweise  Mit *value* **müssen** Sie einen Wert angeben, der bei der Auswahl vom Formular  übergeben wird. Geben Sie keinen Wert an, erhalten Sie nur die Rückmeldung,  ob ein Kotrollkästchen aktiviert wurde (yes/no)  Dieses Attribut gibt eine Auswahl standardmäßig vor. |

Beispiele:



<form>

<p>Was sind Ihre Hobbies? </p>

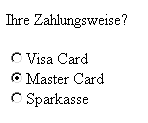
<p><input type="checkbox" name="hobby" value="kino">Kino<br>

<input type="checkbox" name="hobby" value="tanzen" checked>Tanzen<br>

<input type="checkbox" name="hobby" value="lesen">Lesen<br>

</p>

</form>



<form>

<p>Ihre Zahlungsweise? </p>

<p><input type="radio" name="zahlungsweise" value="visa">Visa Card<br>

<input type="radio" name="zahlungsweise" value="master" checked>Master Card<br>

<input type="radio" name="zahlungsweise" value="spark">Sparkasse<br>

</p>

</form>

# 4. Java Script

Script ist kein direkter Bestandteil von HTML, sondern eine ***eigene Programmiersprache***, um HTML- Autoren ein Werkzeug in die Hand zu geben, mit dessen Hilfe sich Web Seiten optimieren lassen.

Java Scripts werden wahlweise direkt in der HTML Datei (im head oder im body) oder in einer separaten Datei notiert. Sie werden erst zur Laufzeit des Browsers interpretiert, wenn der Browser entsprechende Interpreter Software besitzt.

Eine dritte Möglichkeit ist die Verwendung von JavaScript Elementen innerhalb von herkömmlichen HTML Tags. Das ist dann kein komplexer Programmcode, sondern in der Regel nur der Aufruf bestimmter Methoden, Funktionen oder Eigenschaften- man nennt sie Event Handler!

In der Programmiersprache Java Script gib es für Anfänger viele verwirrende Elemente.

In JAVA SCRIPT gibt es - ein Variablenkonzept - Funktionen

* Abfragen (if, if ... else, switch) - Methoden
* Vergleichsoperatoren (==, < ,> ,<=, ...) - Formulare
* logische Opratoren (&&, ||, ! ) - Event Handler
* Zuweisungsoperatoren (++a, --a ) - Arrays
* Schleifen (for, while, do while)

Nur ein langwieriger Lernprozess und viel, viel Übung lassen sinnvolle, und dynamisch ansprechende Programme entstehen.

Am Anfang ist es sogar ratsam, sich im Web umzusehen, ob es nicht schon frei verfügbare JavaScript Beispiele gibt, die genau ihr Problem lösen. In vielen Fällen genügt es, vorhandenen Code den eigenen Erfordernissen anzupassen.

Dazu einige Links:

<http://www.jrp.at>

<http://www.internet-partner.de/webmaster/java-script/javascript.html>

<http://www.mywebaid.de/index.shtml>

<http://www.scriptindex.de>

**Grundlagen:**

|  |  |
| --- | --- |
| <script language=”JavaScript”>  < !-- ………..//-->  </script> | Beginnmarkierung eines Scriptes  Inhalt eines Scriptes  Endemarkierung eines Scriptes |

Beispiel:

....

<body>...

...

<script language="Javascript">

<!--

gr=window.prompt("Bitte deine Körpergröße in cm angeben: "," ");

tatgew=window.prompt("Bitte dein tatsächliches Gewicht angeben: "," ");

if (gr-100<tatgew) {text="Du bist zu dick";};

if (gr-100>tatgew) {text="Du bist zu dünn"};

if (gr-100==tatgew) {text="Du bist zu genau richtig"};

window.document.write(text);

//-->

</script>

</body> **{;** Semikolon muss am Ende jedes Befehles stehen }

**Die Hierarchie der Objekte: (Ausschnitt)**

window Weiter Standartobjekte:

frames

math

document

all date

anchors

applets array

forms

images string

layers

links

event

location

Die Objekte haben eine Vielzahl von Methoden (Funktionen, die Aktionen ausführen) und Eigenschaften (siehe Übersicht).

Um eine Eigenschaft oder Methode aufzurufen, wird ihnen das entsprechende Objekt vorangestellt:

**Objekt.Eigenschaft** oder **Objekt.Methode**

**Ausgewählte Befehle:**

window.document.write(„....“); -

window.alert() ; -

window.prompt(); -

window.open(“Dateiname”,”Fenstername”); -

date.getDate(); - Auslesen des aktuellen Tages

date.getMonth(); - Auslesen des aktuellen Monats

if(Bedingung wahr) {Aktionen;}

else {Aktionen;} - Alternative

onMouseOver...; - Event Handler –

onMouseOut.....; - Event Handler –

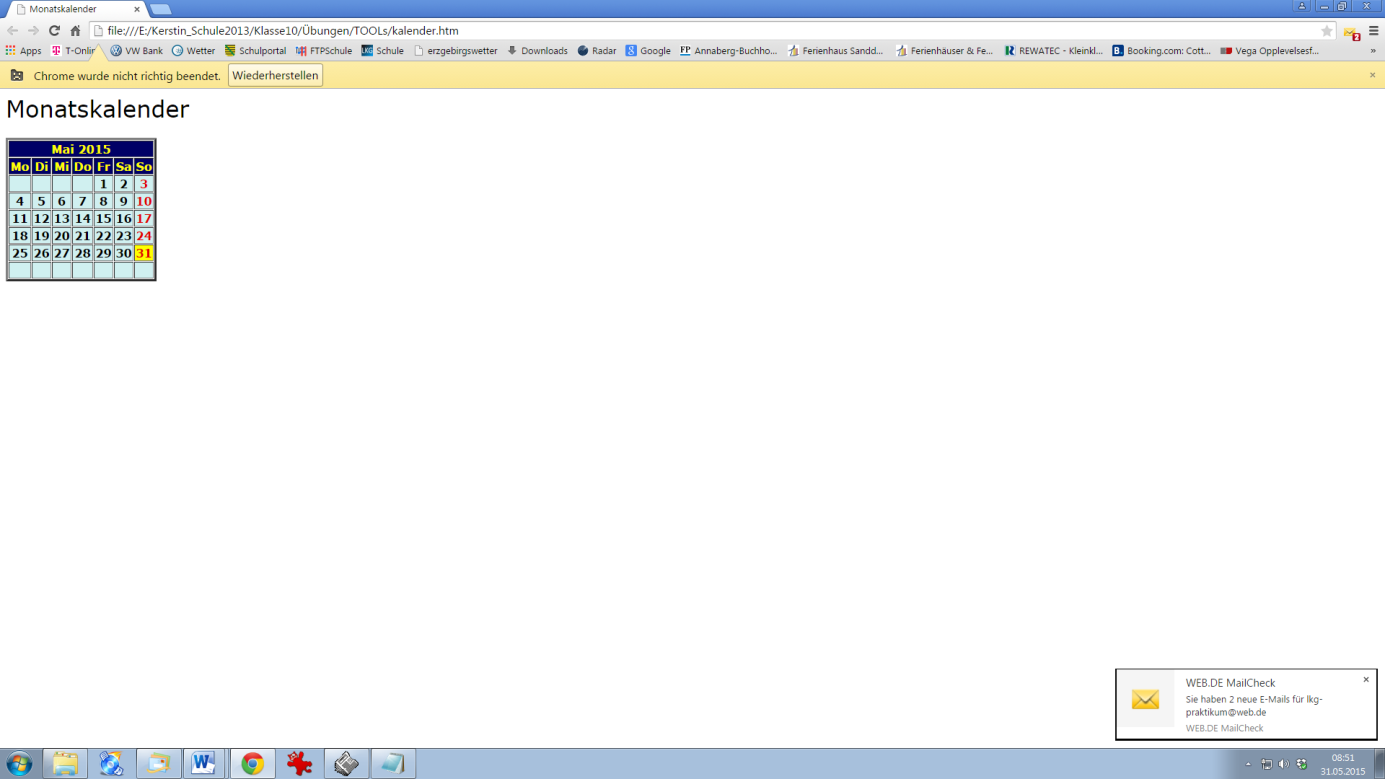
Ausgewählte Beispiele:

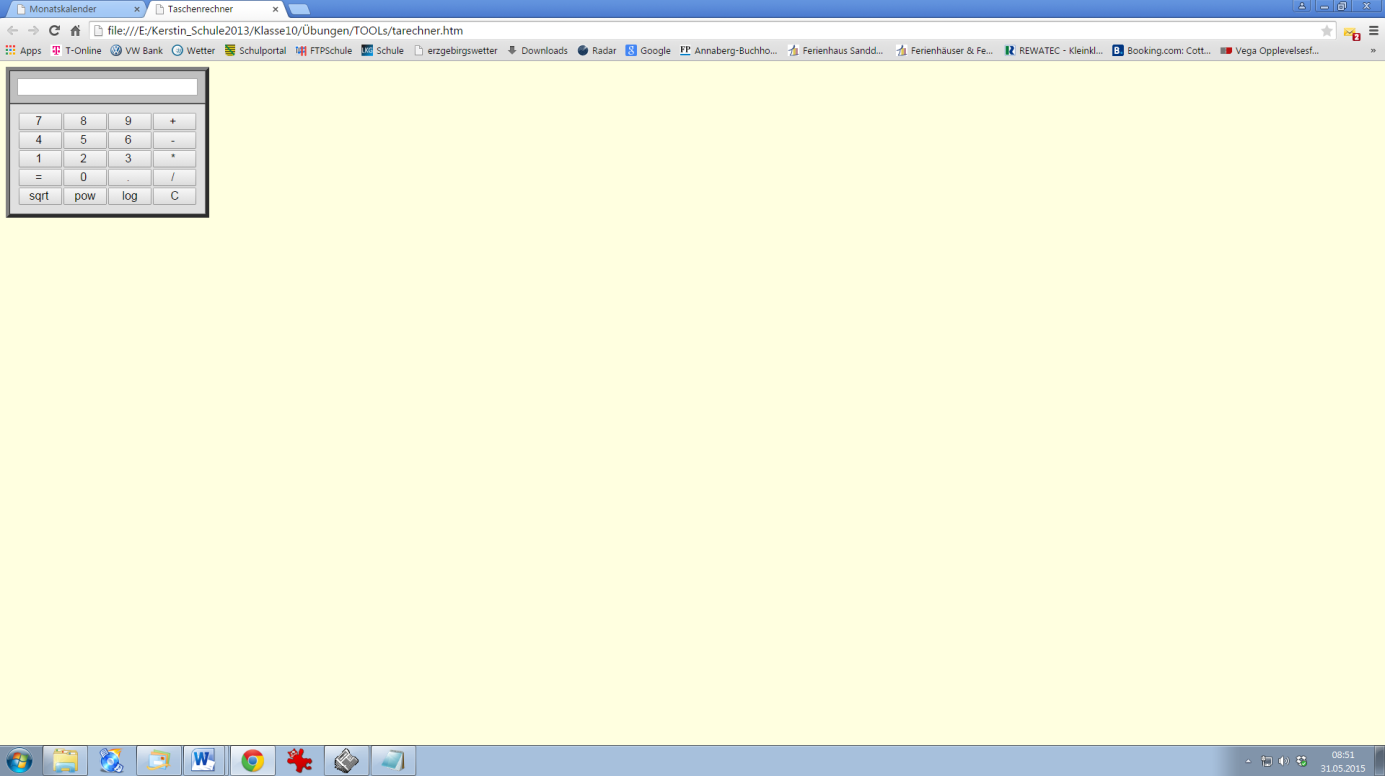
1. Begrüßung des Benutzers mit einer Dialogbox:
2. Anzeige des aktuellen Datums:
3. Beim Öffnen der Homepage soll automatisch ein Fenster mit Werbung angezeigt werden:
4. Beim „Überfahren“ eines Bildes mit der Maus wird ein

anderes Bild gezeigt:

# 

# 6. Nützliche Tools für die Homepage

**Datum und Uhrzeit einblenden (datum\_zeit.htm)**



**Kalender einblenden (kalender.htm)**

**Taschenrechner einblenden (tarechner.htm)  
  
Laufschrift**

<p><marquee border="0">Herzlich Willkommen</marquee></p>

<marquee direction="left" behavior="alternate" scrollamount="10" scrolldelay="5" style="color:#F75014;font-size:16px;font-family:Arial;" > Herzlich Willkommen </marquee>

**Hintergrundmusik**

Muss in den Head eingefügt werden, z.B.

<bgsound src=“horse.wav“ loop=“3“>

**Mouseover/Mouseout Effekt**

<img src="anin.gif" name="bild" onMouseOver="document.bild.src='sport.gif'"   
onMouseOut="document.bild.src='anin.gif'">  **Counter (VORSICHT Werbung!!!)**Man sucht sich dort einen passenden Counter aus und bestellt diesen. Man erhält dann eine ID und ein Passwort, sowie einen HTML-Code, den man an die entsprechende Stelle der Page kopiert! <https://www.andyhoppe.com/counter/>

**Tipps & Fundgrube zum Web Design**

[**http://www.dauerstress.de**](http://www.dauerstress.de)

http://www.buttongenerator.com/temp/temp598624300up.png<http://www.kommdesign.de>  
<http://www.computerbild.de/downloads/webdesign/>

<https://dabuttonfactory.com/> oder  [<https://www.freegifs.de/buttongenerator/buttongenerator.html> /](http://www.buttongenerator.com/) (Buttons zum Download)

# Fundgrube für Bilder und Grafiken

<http://www.clip-art.com> <http://www.gifsammlung.de>

<http://www.grafikarchiv.de> <http://www.barrysclipart.com>

# 7. Zusätzliches zum Selbstlernen

Grundlagen:

<https://www.w3schools.com/html/default.asp>

<https://wiki.selfhtml.org/>

<https://selfhtml.de/>

<https://www.heise.de/download/product/selfhtml-696>

<https://www.html-seminar.de/>