HTML Merkblatt

# Zweck

Mit Hypertext Markup Language (dt. Hypertext Auszeichnungssprache) oder kurz HTML, lassen sich typische Dokumentenelemente, wie z.B. Überschriften, Kapitel, Textabsätze, Tabellen oder Grafiken bezeichnen und strukturieren, um Inhalte in Browsern aufbereitet darstellen zu können. Die Sprache fungiert somit zur vereinheitlichten Strukturierung der im World Wide Web grundlegenden dargestellten Dokumente, da sie von allen Webbrowsern plattformunabhängig und weltweit verstanden wird.

# Links

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

# Aufbau der HTML Befehle

## Tags

HTML basiert auf Befehlen, den sogenannten Tags. Die darzustellenden Inhalte werden von solchen Tags immer umschlossen, also zwischen einem öffnenden Tag und einem schliessenden Tag platziert.

🡪 html, head, body, title, p, a, …

Um die HTML Tags von dem übrigen Inhalt eines Dokumentes zu unterscheiden, werden sie immer durch die Zeichen < (kleiner als) und > (grösser als) umschlossen.

|  |
| --- |
| **<p>**Dies ist ein Tag**</p>** |

Dabei ist zu beachten, dass der schliessende Tag durch einen Slash (Schrägstrich) gekennzeichnet ist.

🡪 **/**

### Tags ohne abschliessenden Befehl

Neben Elementen mit Start- und Schlusstag gibt es in HTML auch wenige Elemente, die keinen abschliessenden Befehl besitzen. Dazu zählt z.B. der Zeilenumbruch: <br />.   
In einem solchen Fall wird der Slash vor das abschliessende > platziert.

|  |
| --- |
| Eine Zeile des Textes**<br />**Eine neue Zeile |

Das Ergebnis im Browser:

|  |
| --- |
| Eine Zeile des Textes Eine neue Zeile |

### Tags mit Attributen

Einige der HTML Befehle beinhalten noch zusätzliche Informationen, die sogenannten Attribute. Die Attribute werden immer nur in dem öffnenden Tag angegeben und zwar nach folgendem Muster:

🡪 <tag attribut="wert">

Möglich sind auch mehrere Attribut-Wert Paare, die dann durch Leerzeichen voneinander getrennt werden:

🡪 <tag attribut1="wert1" attribut1="wert2">

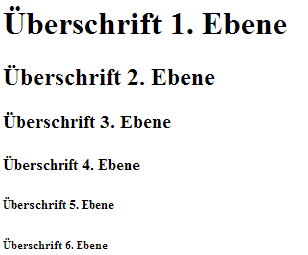
Wichtig ist, dass in standardkonformen HTML die Attribute immer einen Wert besitzen und die Werte durch Anführungsstriche umschlossen sind.

|  |
| --- |
| <a **href="http://www.google.ch">**Das ist ein Link</a> |

### Überschriften

Eine Überschrift umschreibt mit aussagekräftigen Begriffen den nachfolgenden Inhalt. Der Leser oder auch eine Suchmaschine entscheidet somit, ob dieser Beitrag interessante bzw. relevante Informationen enthält. Mit Hilfe von HTML können verschiedene Überschriften mit den Tags *<h1>* bis *<h6>* definiert werden. Dabei nimmt *h1* die höchste und *h6* die niedrigste Ebene ein.

|  |
| --- |
| <h1>Überschrift 1. Ebene</h1>  <h2>Überschrift 2. Ebene</h2>  <h3>Überschrift 3. Ebene</h3>  … |



# Grundgerüst aller HTML Dokumente

## Dokumenttyp (Doctype)

Am Dokumentenanfang steht die Dokumenttypangabe, welche dem Browser Informationen über die verwendete Sprache (HTML oder XHTML) und Variante (Strict, Transitional oder Frameset) mitteilt. Der Browser benötigt diese Informationen für die korrekte Darstellung der Webseite.

|  |
| --- |
| **<!DOCTYPE** HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd"**>** |

Minimal benötigte Information:

|  |
| --- |
| **<!DOCTYPE** HTML**>** |

## Wurzelelement (html)

Direkt nach der Dokumenttypangabe folgt das HTML Wurzelelement, welches man sich als einen grossen Container oder einen Rahmen vorstellen kann. Zwischen dem Start Tag und dem End Tag werden alle weiteren Elemente umschlossen.

|  |
| --- |
| **<html>**Alle weiteren Elemente**</html>** |

## Kopfelement (head)

Das head Element beinhaltet wichtige Informationen über das Dokument, welche durch den Browser und die Suchmaschinenrobots gelesen und abgearbeitet werden. Es befindet sich im Container des Wurzelelement (html). Hier können z.B. ein Titel, die Beschreibung der Webseite, der Name des Autors, entsprechende Suchbegriffe (Keywords) etc. hinterlegt werden.

|  |
| --- |
| **<head>**  <title>Dies ist ein Titel</title>  **</head>** |

### Metatagangaben

Die Metatagangaben werden im Kopf Bereich (head) der Webseite hinterlegt. Die Funktion der Metatagangaben beschränkt sich auf die Bereitstellung von Informationen für den Webserver und die Suchmaschinen. Unteranderem auch auf die Darstellung von Umlauten.

|  |
| --- |
| <meta charset="utf 8"> |

## Seitenkörper (body)

Nach dem Seitenkopf (head) Element folgt, im Container ders Wurzelelements(html), der Seitenkörper (body). Hier liegen alle Inhalte (Texte, Bilder etc.), welche letztendlich im Browserfenster angezeigt werden.

|  |
| --- |
| **<body>**  <p>Dieser Text wird im Browser angezeigt</p>  **</body>** |

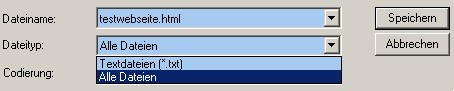
# Beispiel einer funktionierenden Webseite

Um den Code übersichtlicher zu halten, werden die Inhalte in den Container jeweils einmal eingerückt.

|  |
| --- |
| <!DOCTYPE HTML>  <html>  <head>  <meta charset="utf-8">  <title>Testwebseite</title>  </head>  <body>  <p>Meine erste Webseite</p>  </body>  </html> |

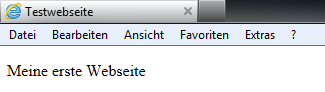
## Webseite im Browser anzeigen lassen

Der einfachste Weg, um eine Webseite zu erstellen, ist mit einem Texteditor. Der HTML-Code wird in den Editor geschrieben und als HTML-Dokument abgespeichert (.html).



Um die Webseite betrachten zu können, muss das Dokument im Browser geöffnet werden. Dies kann entweder mit einem Doppelklick auf das gespeicherte Dokument erreicht werden oder direkt über den Browser durch das Auswählen des Menüpunkts "datei/öffnen".

Das Ergebnis sieht dann etwa so aus:



# Tabellen

## Bestandteile einer Tabelle

Tabellen bestehen aus Reihen (row) und Spalten (column). Für eine Tabelle werden drei verschiedene HTML-TAGs benötigt. Diese HTML-Befehle sind:

<table> 🡪 Grundlegender Tag, jede Tabelle beginnt und endet damit.

<tr> 🡪 Neue Zeile anlegen. Das **tr** steht für **t**able **r**ow = Tabellen Reihe.

<td> 🡪 Neue Tabellenzelle anlegen. Das **td** steht für **t**able **d**ata = Tabellen Daten.

## Aufbau einer Tabelle

Dies ist der Aufbau einer Tabelle, allerdings wird so noch nichts angezeigt.

|  |
| --- |
| <table>  <tr>  <td>  </td>  </tr> </table> |

## Rahmen und Inhalt der Tabelle

Um die Tabelle sichtbar zu machen wird eine **Rahmenbreite** benötigt. Diese kann direkt im Anfang-Tag der Tabelle gesetzt werden. Der **Inhalt**, der in der Tabelle angezeigt werden soll, wird in den Tag <td> geschrieben.

|  |
| --- |
| <table **border="1"**>  <tr>  <td>  **INHALT 1**  </td>  </tr> </table> |

Nun können der Tabelle noch mehr Zeilen und Spalten hinzugefügt werden. l

|  |
| --- |
| **D:\userdata\hasa\Downloads\Table OMR - New Page (1).png**<**table** border="1">  <**tr**>  <**td**> INHALT1 </**td**>  <**td**> INHALT2 </**td**>  </**tr**>  <**tr**>  <**td**> INHALT3 </**td**>  <**td**> INHALT4 </**td**>  </**tr**>  </**table**> |

## Tabellenzellen verbinden

Wenn wir nun Tabellenzellen miteinander verbinden möchten, dass z.B. anstatt 2 Zellen nur eine vorhanden ist, funktioniert das über colspan und rowspan.

Zu beachten ist, dass die Zellen, die miteinander verbunden werden, im HTML-Code nicht angegeben werden. Nur die Zelle, die den Befehl beinhaltet, wird angegeben.

Die Anzahl Zellen, die miteinander verbunden werden sollen, muss jeweils mitgegeben werden.

### Rowspan

Mit dem Befehl rowspan, kann man die Zellen senkrecht verbinden.

Will man also die Zelle INHALT1 mit der Zelle INHALT3 verbinden, wird nur die Zelle INHALT1 mit dem Befehl angegeben.

|  |
| --- |
| <table border="1">  <tr>  <td **rowspan="2"**>  INHALT1  </td>  <td> INHALT2 </td>  </tr>  <tr>  <td> INHALT4 </td>  </tr>  </table> |

### Colspan

Mit dem Befehl colspan kann man die Zellen waagrecht verbinden.

Will man also die Zelle INHALT1 mit der Zelle INHALT2 verbinden, wird nur die Zelle INHALT1 mit dem Befehl angegeben.

|  |
| --- |
| <table border="1">  <tr>  <td **colspan="2"**>  INHALT1  </td>  </tr>  <tr>  <td> INHALT3 </td>  <td> INHALT4 </td>  </tr>  </table> |

# Formulare

Unter Formularen versteht man die Kontaktaufnahme und den Datenaustausch durch HTML.

Dies findet durch sogenannte Steuerelemente wie Button, Textfeld oder Radio Button statt.

## Definition von Formularen

Ein Formular wird mit **<FORM>** geöffnet und mit **</FORM>** wieder geschlossen. Dazwischen stehen dann die weiteren Anweisungen. Dies kann zum Austesten auf dem PC auch erst einmal genau so eingegeben werden. Um das Formular aber wirklich online nutzen zu können, muss später eine Adresse (wohin soll das Formular zurückgeschickt werden?) und eine Methode (wie wird es zurückgeschickt?) hinzugefügt werden.

|  |
| --- |
| **<FORM ACTION="urladresse" METHOD="POST">**  Hier können weitere Formularelemente stehen und auch ganz normale HTML-Befehle dürfen eingebaut werden.  **</FORM>** |

### Eingabefelder

Die Angabe hinter NAME darf keine deutschen Umlaute und auch keine anderen Sonderzeichen enthalten. Der NAME ist für die Formularverarbeitung wichtig und sollte daher nur einmal pro Formular verwendet werden.

Mit der Angabe von TYPE können verschiedene Typen von Eingabefeldern definiert werden.

**Textfelder**:

|  |
| --- |
| <INPUT TYPE=Textfeld NAME=" feldname"> |

**Passwortfelder**:

|  |
| --- |
| <INPUT TYPE=PASSWORD NAME="feldname"> |

Dateien:

|  |
| --- |
| <INPUT TYPE=**FILE** NAME="feldname"> |

**Textarea**:

|  |
| --- |
| <TEXTAREA NAME="feldname">  Mein Kommentar:  </TEXTAREA> |

### Auswahlfelder

**Auswahlliste**:

Mit dem Attribut SELECTED kann man angeben welche Option standardmässig ausgewählt ist.

|  |
| --- |
| <SELECT NAME="listenname">  <OPTION>Auswahl 1</OPTION>  <OPTION SELECTED>Auswahl 2</OPTION>  <OPTION>Auswahl 3</OPTION>  </SELECT> |

**Radiobuttons**:

Mit dem Attribut CHECKED kann man angeben, welche Option standardmässig ausgewählt ist.

Zu beachten ist, dass zusammengehörende Radiobuttons IMMER den gleichen NAMEN haben müssen.

|  |
| --- |
| <INPUT TYPE=RADIO NAME="feldname" VALUE="name1">Button 1 <INPUT TYPE=RADIO NAME="feldname" VALUE="name2" CHECKED>Button 2 <INPUT TYPE=RADIO NAME="feldname" VALUE="name3">Button 3 |



**Checkbuttons**:

Mit dem Attribut CHECKED kann man angeben, welche Option standardmässig ausgewählt ist.

Zu beachten ist, dass zusammengehörende Checkboxen IMMER den gleichen NAMEN haben müssen.

|  |
| --- |
| <INPUT TYPE=CHECKBOX NAME="feldname" VALUE="name1">Button 1 <INPUT TYPE=CHECKBOX NAME="feldname" VALUE="name2" CHECKED >Button 2 <INPUT TYPE=CHECKBOX NAME="feldname" VALUE="name3">Button 3 |



### Befehls-Schaltflächen

**Submit:**

Beim Drücken der Schaltfläche und einer bestehenden Internetverbindung wird das Formular mit allen getätigten Eingaben verschickt.

Für eine korrekte Weiterleitung und -verarbeitung ist die im FORM genannte URL-Adresse zuständig.

|  |
| --- |
| <INPUT TYPE=SUBMIT VALUE="Senden"> |



**Reset**:

Beim Drücken der Schaltfläche werden die Eingaben des Formulars gelöscht.

|  |
| --- |
| <INPUT TYPE=RESET VALUE="Löschen"> |

