

Praktikum

DBAE / WI

HTML

Pascal Reuss, M.Sc.

Raum A08b Spl.

Email: reusspa@uni-hildesheim.de

HTML

HyperText Markup Language

- Netzartige Organisation von Dokumenten mit Verweisen zwischen diesen Dokumenten (Hypertext / http)
- Beschreibung der Darstellung von Dokumenten (Markup Language)

Standards werden vom World Wide Web Consortium (W3C) entwickelt

- Aktuell: HTML 5 (seit 2014)

HTML-Dateien sind einfache „Textdateien“

- Können mit einfachen Texteditoren erzeugt werden (z.B. Editor, Notepad++, emacs)
- Oder auch mit HTML-Editoren (z.B. FrontPage, Mozilla Composer)

HTML-Dateien werden dann von einem HTML-Parser bearbeitet

- z.B. in einem Web-Browser

HTML Historie

1989	Sir Timothy Berners-Lee erfindet World Wide Web
1991	Sir Timothy Berners-Lee erfindet HTML
1993	HTML+
1995	HTML 2.0 von HTML Working Group
1997	HTML 3.2 vom W3C
1999	HTML 4.01 vom W3C
2000	XHTML 1.0 vom W3C
2008	Erste Draft-Version von HTML 5 (Web Hypertext Application Technology Working Group)
2012	HTML 5 Living Standard von WHATWG
2014	HTML 5 vom W3C

HTML Aufbau

HTML-Dokumente werden in Elementen (Tags) unterteilt

Alle Dokumente sollten mit einer Dokumenttyp-Deklaration (DTD) beginnen

Seit HTML 5 sehr simpel:

```
<!DOCTYPE html">
```

Der Inhalt der Seite befindet sich unter dem Element „html“

HTML Tags

Die Tags repräsentieren Elemente des HTML-Dokumentes

- Tags sagen dem Browser wie das Dokument dargestellt werden soll
- Tags enthalten Informationen über das Dokument / über die Darstellung des Dokumentes

Sie werden immer zwischen ,<' und ,>' geschrieben

- Meistens als Paar mit einem „Öffnenden“ Tag und einem „Schließenden“ Tag (z.B. <html> text </html>)
- Aber auch alleine (z.B.
 bzw.
)
- Kein Unterschied zwischen Groß-/Kleinschreibung (<HTML> ist dasselbe wie <html>)
- Kleinschreibung sollte aber bevorzugt werden

HTML Tags

Die Tags können Attribute haben

- Sie enthalten zusätzliche Informationen über das Element
- Mit oder ohne Wertzuweisung (<body bgcolor="blue"> vs. <hr noshade>)
- Jedem Attribut sollte aber ein Wert zugewiesen werden (<hr noshade="noshade">)

Tags können verschachtelt werden

Unbekannte Tags werden ignoriert

Beispiel

```
<!DOCTYPE html>
<html>
  <head>
    <title>Titel der Seite</title>
  </head>
  <body bgcolor="blue">
    Willkommen auf meiner Beispielseite
  </body>
</html>
```

// „Sagt“ Browser: HTML-Dokument, Beginn
// Header-Informationen, Beginn
// Header-Info: Titel der Seite
// Header-Info, Ende
// Körper = eigentlicher Inhalt, Beginn
// Inhalt
// Körper, Ende
// HTML-Dokument Ende

Grundlegende Tags

Überschriften (headline): `<h1>überschrift</h1>`, .. , `<h6>überschrift</h6>`

Paragraphen: `<p> text </p>`

Zeilenumbruch (break): `
` oder: `
`

Kommentare: zwischen: `<!-- Ich bin ein Kommentar -->`

Kursivschrift (italic): `<i>text</i>`

Fettdruck (bold): `text`

Präformatierter Text: `<pre>text</pre>`

Querverweise - Hyperlinks

Querverweise werden mittels der Tags `<a>` angesetzt

Mit dem Attribut ‚name‘ kann man Anker in einem Dokument definieren

z.B. Mit `Anfang einer langen Seite`

Mit dem Attribut ‚href‘ können verschiedene Verweise gesetzt werden

auf definierte Anker

z.B. `zum Anfang der Seite`

auf lokale Dateien bzw. definierte Anker in lokalen Dateien

z.B. `zur Datei`

bzw. `zum Anfang der anderen Datei`

auf URLs bzw. definierte Anker auf URLs

z.B. `AG IIS`

auf Emails

z.B. `E-mail`

Tabellen

Tabellen kann man mit dem Tag `<table>` definieren

Dem Attribut 'border' kann man ein Wert zuweisen, um die Dicke der Umrandungen zu definieren

- `<table border="0">`

Neue Zeilen werden mit `<tr>` eingefügt

Zellen in Zeilen werden mit `<th>` (für Kopfzeilen) oder `<td>` (für normale Zeilen) eingefügt

Zellen können zeilenweise (mit dem Attribut 'rowspan') oder spaltenweise (mit 'colspan') verbunden werden

Tabellen

Übersicht Tabellenaufbau mit HTML:

```
<table>
<tr>
<td>                </td> <td>                </td> <td>                </td>
</tr>
<tr>
<td>                </td> <td>                </td> <td>                </td>
</tr>
<tr>
<td>                </td> <td>                </td> <td>                </td>
</tr>
</table>
```

Listen

Aufzählungslisten werden mit `` definiert

Nummerierte Listen mit ``

`` wird benutzt, um Listeneinträge einzugeben

``

`Probieren geht über Studieren`

`Liebe geht über Triebe`

`Tante fährt über Kante`

``

Definitionslisten werden mit `<dl>` definiert

Der zu definierende Term mit dann mit `<dt>` eingegeben

und die Definition mit `<dd>`

Formulare

Formulare bestehen aus verschiedenen Elemente, die es dem Benutzer erlauben, Informationen einzugeben

Sie werden mit dem tag `<form>` definiert

Mit dem Attribut 'action' wird dem Browser gesagt, was mit den Eingaben geschehen soll:

- Email schicken
- an einem Server schicken, um bearbeitet zu werden (durch CGI-Skripte, Servlets, usw...)
- ...

Formulare Texteingabe

Einzeilige Textfelder werden mit `<input>` (ohne Endtag) definiert.

Dabei wird mit dem Attribut 'type' der Typ des Feldes festgelegt
"text" für normaler Text "password" für Passwörter

```
<input name="vorname" type="text" size="30" maxlength="30">
```

Mit `<textarea>` kann man mehrzeilige Textfelder definieren

Das Attribut 'rows' gibt die Anzahl der Zeilen an und 'cols' die Anzahl der Spalten (Zeichen)

Jedes Eingabefeld sollte mit dem Attribut 'name' einen Bezeichnernamen erhalten

```
<textarea name="user_eingabe" cols="50" rows="10"></textarea>
```

Formulare Auswahllisten

Auswahllisten definiert man mit `<select>`

Sie sollten auch mit `'name'` einen Bezeichnernamen erhalten

Mit `<option>` fügt man Listeneinträge ein

Man kann jedem Eintrag einen Wert mit dem Attribut `'value'` zuweisen.

Dieser wird dann z.B. dem CGI-Skript / Servlet geschickt.

```
<select name="top3" size="3" value="Musiker">  
  <option>Tom Waits</option>  
  <option>Nina Hagen</option>  
  <option>Marianne Rosenberg</option>  
</select>
```

Formulare Radio-Buttons

Radio-Buttons sind eine Gruppe von beschrifteten Knöpfen, von denen nur einen ausgewählt werden kann

Jeder Knopf wird durch `<input type="radio">` definiert

sollte auch einen Bezeichnernamen und einen Wert erhalten

Mit dem Attribut `'checked'` kann eine Auswahlmöglichkeit vorselektiert werden

```
<form action="input_radio.htm">  
  <p>Geben Sie Ihre Zahlungsweise an:</p>  
  <p>  
    <input type="radio" name="Zahlungsmethode" value="Mastercard"> Mastercard<br>  
    <input type="radio" name="Zahlungsmethode" value="Visa"> Visa<br>  
    <input type="radio" name="Zahlungsmethode" value="AmericanExpress"> American Express  
  </p>  
</form>
```


Formulare Checkboxes

Checkboxen sind eine Gruppe von ankreuzbaren Quadraten, bei denen der Anwender keine, eins oder mehrere auswählen kann

Jede Checkbox wird durch `<input type="checkbox">` definiert

Hier kann man auch mit dem Attribut `'checked'` Auswahlmöglichkeiten vorselektieren

```
<form action="input_checkbox.htm">
  <p>Kreuzen Sie die gewünschten Zutaten an:</p>
  <p>
    <input type="checkbox" name="zutat" value="salami"> Salami<br>
    <input type="checkbox" name="zutat" value="pilze"> Pilze<br>
    <input type="checkbox" name="zutat" value="sardellen"> Sardellen
  </p>
</form>
```

Formulare Send-Button

Absende-Buttons definiert man mit `<input type="submit">`

Abbrechen-Buttons können auch mit `<input type="reset">` definiert werden

Die Beschriftung der Buttons wird dem Attribut `'value'` bestimmt

```
<input type="submit" value="Absenden">
```

```
<input type="reset" value="Abbrechen">
```

Literaturhinweise

SELFHTML:

<http://selfhtml5.org>

W3schools HTML Tutorial

<http://www.w3schools.com/html>