

Hochschule Darmstadt

Fachbereich Informatik

Entwicklung webbasierter Anwendungen



h_da

HOCHSCHULE DARMSTADT
UNIVERSITY OF APPLIED SCIENCES

fbi

FACHBEREICH INFORMATIK

Hochschule Darmstadt

Fachbereich Informatik

Wiederholung



h_da

HOCHSCHULE DARMSTADT
UNIVERSITY OF APPLIED SCIENCES

fbi

FACHBEREICH INFORMATIK

Tags & Attribute



A diagram illustrating an HTML element with an attribute. The text `<p class="editor-note">My cat is very grumpy</p>` is shown on a dark background. A bracket and the label 'Attribute' point to the `class="editor-note"` part of the opening tag.

Source: https://developer.mozilla.org/en-US/Learn/Getting_started_with_the_web/HTML_basics



Grundgerüst einer (ordentlichen) HTML5-Datei

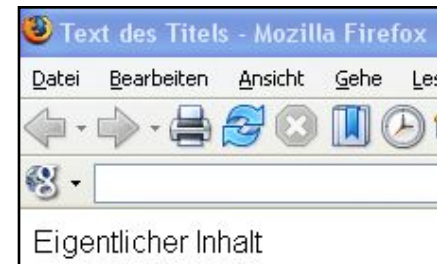
```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8" />
    <title>Text des Titels</title>
  </head>
  <body>
    <p>Eigentlicher Inhalt</p>
  </body>
</html>
```



Dokumenttyp

Zeichensatz

Titel für
Browserfenster



Strukturierung von Text

alle außer
 und

erzeugen einen Block

■ Überschriften

<h1> Überschrift der höchsten Gliederungsebene

<h6> Überschrift der niedrigsten Gliederungsebene

heading1 ... heading6

■ Abschnitte

<p> Textabsatz

<div> allgemeiner Block

div = division = Bereich

 Inline-Element

kein Block

"Aufhänger" für CSS

■ Aufzählungen (nummeriert oder auch nicht)

, ,

ordered list, unordered list, list item

■ Zeilenumbruch erzwingen und verhindern

 expliziter Zeilenumbruch (standalone tag)

kein Block

 geschütztes Leerzeichen – verhindert Zeilenumbruch

­ soft hyphen – Bindestrich bei Bedarf

z.B. 3.
Kapitel

Strukturierung von Webseiten



■ Der Text innerhalb des `<body>`-Tags kann gegliedert werden

- ⇒ `<section>` Abschnitt - ein logischer Bereich einer Webseite (z.B. der News Bereich)
- ⇒ `<article>` Artikel
ein Textabschnitt, der eigenständig einen Inhalt abdeckt (z.B. eine einzelne News)
- ⇒ `<body>` kann mehrere `<section>`s und `<article>`s enthalten – auch verschachtelt
- ⇒ `<nav>` Navigationsbereich – enthält Verknüpfungen zur Navigation
- ⇒ `<header>` und `<footer>` Kopf- / Fußzeilenbereich
ein Bereich mit Überschriften bzw. Logos, Datum usw. für das Gesamtdokument (im `<body>`) oder `<section>`s und `<article>`s
- ⇒ `<aside>` Blöcke an der Seite
z.B. für Seitenleisten (außer Navigation), Zitate, Anmerkungen

`<section>`s und `<article>`s haben eine Überschrift (`<h2>...<h6>`)!



Universalattribute

■ können zu jedem Tag hinzugefügt werden

- ⇒ `id` dateiweit eindeutiger Bezeichner für Skripte
- ⇒ `class` Name der zugehörigen Style Sheet Klasse
- ⇒ `title` Erläuterung zum Element, erscheint als Tooltip
- ⇒ `style` eingebettete Style Sheet Attribute (siehe CSS)
- ⇒ `lang, dir` Landessprache und Textlaufrichtung

```
<h2 id="JB007" class="mycssstyleclass" title="mytooltip" style="color:red" lang="de" dir="ltr">
```

Hallo

```
</h2>
```



Struktur einer Tabelle (2)

nur zur Darstellung
tabellarischer Daten

Tabellen-Überschrift

zeilenweise (tr = table row)

```
<table>
  <caption>Meine Tabelle</caption>
  <tr>
    <th>Kopfzelle: 1. Zeile, 1. Spalte</th>
    <th>Kopfzelle: 1. Zeile, 2. Spalte</th>
  </tr>
  <tr>
    <td>Datenzelle: 2. Zeile, 1. Spalte</td>
    <td>Datenzelle: 2. Zeile, 2. Spalte</td>
  </tr>
  <tr>
    <td>Datenzelle: 3. Zeile, 1. Spalte</td>
    <td>Datenzelle: 3. Zeile, 2. Spalte</td>
  </tr>
</table>
```

beliebig viele Zeilen und Spalten

Meta-Angaben

- Anweisungen für WWW-Server, WWW-Browser und automatische Suchprogramme ("Robots")

- eine kleine Auswahl von Meta-Angaben:

```
<meta name="description" content="Autovermietung" />
```

```
<meta name="author" content="B. Kreling" />
```

```
<meta name="keywords" content="Hotel,Urlaub,Meer" />
```

```
<meta name="robots" content="noindex" />
```

```
<meta name="date" content="2001-02-06" />
```

```
<meta name="language" content="de" />
```

```
<meta http-equiv="refresh" content="5">
```

lädt die aktuelle
Seite nach 5
Sekunden erneut

- Setzen des verwendeten Zeichensatzes

```
<meta charset="UTF-8">
```

Agenda

- 2. Webclient
 - 2.1 HTML
 - 2.1.2 Hyperlinks
 - 2.1.3 HTML Formulare

Hochschule Darmstadt

Fachbereich Informatik

2.1.2 Hyperlinks



h_da

HOCHSCHULE DARMSTADT
UNIVERSITY OF APPLIED SCIENCES

fbi

FACHBEREICH INFORMATIK

Anwendungsfälle für Hyperlinks

- Beispiele für Einsatzmöglichkeiten
 - ⇒ Querverweis (vgl. Lexikon, Literaturstelle)
 - ⇒ Blättern (nächste Seite / vorige Seite)
 - ⇒ Inhaltsverzeichnis (Unterkapitel / Oberthema)
 - ⇒ Stichwortverzeichnis
 - ⇒ freie Navigation, neue Dokumentstrukturen ⇒ Hypermedia
 - ⇒ Download einer Datei
 - ⇒ sonstiger Dienst
- "Hyperlink" ist lediglich eine technische Realisierung !
- Einsatzgebiet klären und gestalterisch unterscheiden

Gestaltungstipps für Verweise

- ein Verweis ist ein Blickfang
 - ⇒ nur bedeutungstragende Begriffe mit Hyperlink hinterlegen
- Verweistext soll das Verweisziel deutlich machen
 - ⇒ vorzugsweise immer derselbe Text für dasselbe Ziel
 - ⇒ nicht zu viele Verweise auf dieselbe Stelle innerhalb einer Seite
- Verweis sollte unmittelbar erkennbar sein
 - ⇒ nicht erst nach "Abtasten" mit der Maus
- alle Seiten vollständig verlinken
 - ⇒ "Zurück"-Button des Browsers sollte innerhalb einer Website überflüssig sein - aber er sollte möglichst auch funktionieren

Ziele von Verweisen

- eine Datei, die der Browser als Seite darstellen kann
 - ⇒ meistens HTML, aber auch anderes möglich
 - ⇒ im Internet oder lokal
- bestimmte Position ("Anker") innerhalb einer darstellbaren Datei
- eine Datei, die der Browser selbst nicht darstellen kann
 - ⇒ diese wird zum Download angeboten oder mit einer Hilfsanwendung geöffnet
- andere Dienste neben WWW
 - ⇒ mailto, gopher, ftp, telnet, news

```
<a href="mailto:j.bond@fbi.h-da.de">J. Bond</a>  
<a href="ftp://www.xyz.de/setup.zip">Download</a>  
<a href="file:///c:/lokal.htm">lokale Datei</a>
```

Verweise

Der Verweistext sollte eine klare Information über das Ziel des Verweises geben !

■ Allgemeine Form

```
<a href="Dienst://Server:Port/Verz/Datei#Anker">  
  Text</a>
```

Teile davon können weggelassen werden

■ Datei im selben / unter- / übergeordneten Verzeichnis

```
<a href="start.htm">Text</a>
```

```
<a href="sub/datei.htm">Text</a>
```

```
<a href="../inhalt.htm">Text</a>
```

```
<a href="/inhalt.htm">Text</a>
```

relativ

auch: localhost

■ Datei auf anderem Server

```
<a href="http://www.xyz.de/datei.htm">Text</a>
```

absolut

■ Groß-/Kleinschreibung beachten

Absolute und relative Verweise

ohne Angabe von Server und Verzeichnispfad

- relative Verweise innerhalb der eigenen Website (projekt-intern) sind vorteilhaft für
 - ⇒ Migration auf anderen Server oder in anderes Verzeichnis
 - ⇒ Entwicklung auf lokaler Festplatte mit späterem Upload
 - ⇒ Download als ZIP und lokale Installation

- absolute Verweise sind vorteilhaft für
 - ⇒ Versenden von Seiten per eMail (z.B. Werbung, Stundenplan; sofern der Leser online ist wird er direkt auf den Webserver weitergeleitet)
 - ⇒ Verweise auf fremde Websites (projekt-extern)

Verweise innerhalb einer Datei ("Anker")

- wird häufig eingesetzt für "Inhaltsverzeichnis" am Anfang einer Datei
 - ⇒ z.B. bei FAQ
- Verweisziel definieren per id in beliebigem Tag

```
<h2 id="Erl">Erläuterung</h2>
```
- Verweis definieren

```
siehe die <a href="#Erl">Erläuterung</a> unten
```
- der Verweis kann auch zu einer bestimmten Position in einer anderen Datei zeigen

```
<a href="datei.htm#Erl">Erläuterung</a>
<a href="http://www.xyz.de/datei.htm#Anker">...</a>
```
- der Browser scrollt die Seite so, dass der Anker an der Oberkante des Fensters erscheint

veraltet:
`...`

Zusammenfassung

- Grundgerüst: DOCTYPE, <html>, <head>, <body>, <title>, charset...
- Schreibregeln: Zeilenumbruch, Kommentare und Sonderzeichen
- Tags und Attribute
- Tabellen
- Logische Formatierung und verpönte Formatierung
- Einbinden von Grafiken, Audio, Video...
- Meta-Angaben
- Verwendung von Hyperlinks
- Verweise innerhalb einer Seite (Anker)

Jetzt können Sie eine einfache HTML-Seite schreiben!

Hochschule Darmstadt

Fachbereich Informatik

2.1.3 HTML Formulare



h_da

HOCHSCHULE DARMSTADT
UNIVERSITY OF APPLIED SCIENCES

fbi

FACHBEREICH INFORMATIK

Formular - Beispiel

Steuerelemente für Formulare

Bitte machen Sie Ihre Eingaben

Einzeiliges Textfeld

Ihre Eingabe

Textfeld mit Scrollbalken

Viiiiiel Text

Bitte wählen Sie aus

List-Box

1. Möglichkeit

2. Möglichkeit

3. Möglichkeit

Combo-Box

1. Möglichkeit

Radiobuttons

Ja ☒ Naja ☐ Nein ☐

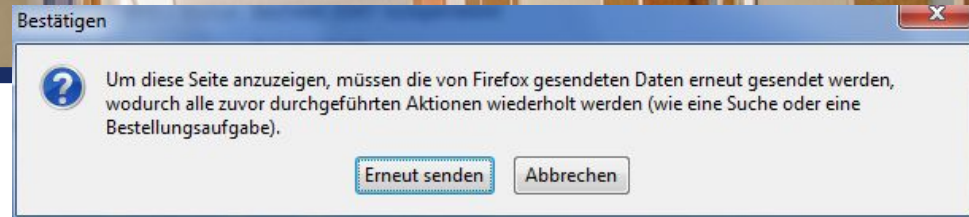
Checkboxen

Flag1 ☒ Flag2 ☐

Schaltflächen

Abschicken

Zurücksetzen



2.1.3 HTML Formulare

Funktion von Formularen

■ Formulare dienen der Eingabe von Daten

- ⇒ eingegebene Daten werden an Server übermittelt und dort ausgewertet
- ⇒ es gibt 2 Möglichkeiten der Datenübertragung
 - `get` übermittelt Parameter für Abfrage (z.B. Suchmaschine)
 - `post` übermittelt Daten zwecks Speicherung (z.B. Bestellung)

vgl. Reload
im Browser

■ Bereich mit Eingabeelementen im HTML-Body markieren

```
<form action="/cgi-bin/Echo.pl" id="form1"  
      accept-charset="UTF-8" method="get">
```

Steuerelemente (Eingabefelder, Auswahllisten, Buttons...) und sonstige HTML-Tags und CSS-Formatierung

hier: Übergabe der
Daten an Perl-Skript

```
</form>
```

- ⇒ `accept-charset` zur Sicherheit gegen willkürliche Benutzereinstellung
- ⇒ falls das Steuerelement außerhalb des Formulars liegt, kann der Bezug über `form="form1"` hergestellt werden

aber nicht mit
Internet Explorer

■ Alternative Aktion: Formulardaten per eMail verschicken

- ⇒ `action="mailto:Meier@xyz.de"`
- ⇒ unsicher, weil von der Installation beim Surfer abhängig

2.1.3 HTML Formulare

Eingabefelder

placeholder erscheint
nur ohne value



■ einzeilige Textbox

```
<input type="text" name="zuname" value="Muster" size="30"
    maxlength="40" placeholder="Ihre Eingabe"
    readonly />
```

standalone tag

- ⇒ `name` und `value` wird an Server übermittelt
- ⇒ `value` kann vorgelegt sein
- ⇒ `size` und `maxlength` für Anzeigelänge und Maximalgröße
- ⇒ `placeholder` wird angezeigt, bevor man eine Eingabe macht
- ⇒ mit `readonly` reine Anzeige (ausgegraut)

■ Variante: Passwortfeld mit *-Anzeige

- ⇒ wie oben, jedoch `type="password"`
- ⇒ keine verschlüsselte Übertragung!

■ mehrzeiliges Textfeld (bei Bedarf mit Scrollbalken)

```
<textarea name="feedback" cols="50" rows="10"
    placeholder="Viiiiiel Text"></textarea>
```



2.1.3 HTML Formulare

Auswahllisten

■ Listbox

```
<select name="top4[]" size="3" multiple>
  <option selected>      1. Möglichkeit</option>
  <option>                2. Möglichkeit</option>
  <option value="3">      3. Möglichkeit</option>
  <option>                4. Möglichkeit</option>
</select>
```

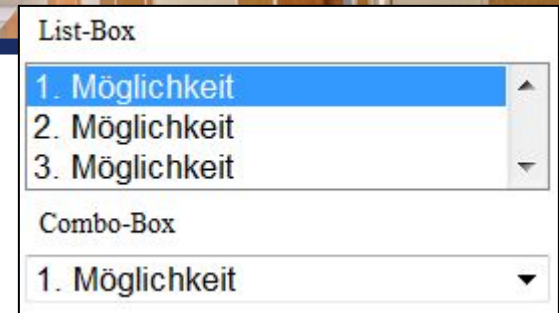
- ⇒ `size` bestimmt die Höhe in Zeilen
- ⇒ Vorauswahl ggfs. mit `<option selected>`
- ⇒ angezeigter Text wird als ausgewählter Wert übertragen, sofern kein `<option value="xyz">` definiert ist

■ Combobox

- ⇒ Eine Listbox mit `size="1"` liefert eine aufklappende Liste mit Optionen

■ Mehrfachauswahl mit zusätzlichem Attribut `multiple`

- ⇒ Bei erlaubter Mehrfachauswahl eckige Klammern an den Namen hängen (z.B. `name="top4[]"`)! PHP macht dann daraus ein Array!



Schaltflächen und verborgene Felder

Schaltflächen

- allgemeine Schaltflächen für JavaScript-Ereignisse
`<input type="button" name="Start" value="Startseite" onclick="window.location.href='http://www.xyz.de/'" />`
- Schaltfläche zum Absenden der Formulardaten
 - ⇒ wie oben, jedoch `type="submit"` ; `onclick` nicht nötig
- Schaltfläche zum Löschen der Formulardaten
 - ⇒ wie oben, jedoch `type="reset"` ; `onclick` nicht nötig
- verborgenes Datenfeld (z.B. für Sessionverwaltung)
 - ⇒ `<input type="hidden" name="sessionID" value="4711" />`
- in HTML5 gibt es noch diverse andere Typen für das Input-Tag
 - ⇒ `<input type="email"... />` oder auch `date`, `number`, `color` uvm.
 - ⇒ diese Typen überprüfen automatisch die Eingabe



Daten werden nur dann übertragen, wenn die Felder ein name-Attribut haben!

Radiobuttons und Checkboxes

- Radiobuttons als Gruppe von Knöpfen, die sich gegenseitig auslösen (Auswahl 1 aus n)

- ⇒ Gruppierung erfolgt durch identischen `name`
- ⇒ der `value` wird als Wert der Gruppe übertragen

```
<input type="radio" name="OK" value="1" />  
<input type="radio" name="OK" value="2" />  
<input type="radio" name="OK" value="3" checked />
```

- ⇒ Vorauswahl durch Attribut `checked`
- ⇒ Zur Beschriftung ist `<label>` geeignet (nächste Folie)

- Checkboxes für Boole'sche Eingabe

```
<input type="checkbox" name="zutat" value="salami" />
```

- ⇒ übermittelt wird der `value` nur für angekreuzte Checkboxes
- ⇒ Vorauswahl durch Attribut `checked`
- ⇒ Beschriftung erfolgt mit Labels

The screenshot shows a web form with two sections. The first section, titled 'Radiobuttons', contains three radio buttons labeled 'Ja', 'Naja', and 'Nein'. The 'Ja' button is selected. The second section, titled 'Checkboxen', contains two checkboxes labeled 'Flag1' and 'Flag2'. The 'Flag1' checkbox is checked. A yellow box with the text 'Die Beschriftungen sind <label>' has a line pointing to the labels of the radio buttons.

Die Beschriftungen
sind `<label>`

"on" wenn value fehlt

Beschriftung von Formularelementen

Vorname:	<input type="text" value="Ihr Vorname"/>
Zuname:	<input type="text" value="Ihr Nachname"/>
Auswahl:	<input checked="" type="checkbox"/>

- Formularelemente haben kein Attribut für Text
 - ⇒ Der Text "Vorname" und das Eingabefeld "Ihr Vorname" im Beispiel haben (für den Browser) keinen Zusammenhang
- Mit Hilfe von "Labels" wird ein logischer Bezug zwischen Formularelement und Beschriftungstext hergestellt
 - ⇒ `<label>Zuname:`
`<input type="text" id="nachname" name="Zuname"/>`
`</label>`
 - ⇒ Das `<label>` umschließt "sein" Formularelement
 - ⇒ Alternativ wird der Bezug über eine `id` und das Attribut `for` hergestellt:
`<label for="nachname"></label>`
 - ⇒ anwendbar für `<input>`, `<select>` und `<textarea>`
- Vorteile bei der Verwendung
 - ⇒ Beim Klicken auf den (zugeordneten) Text wird das Eingabefeld selektiert bzw. die Checkbox selektiert

Gruppierung von Formularelementen

- Größere Formulare bestehen häufig aus *Gruppen* von Elementen. Ein typisches Bestellformular besteht beispielsweise aus Elementgruppen wie "Absender", "bestellte Produkte" und "Formular absenden"
 - ⇒ Eine zusammengehörige Gruppe von Formularelementen wird durch `<fieldset>...</fieldset>` umrahmt
 - ⇒ Dazwischen können Sie beliebige Teile Ihres Formulars definieren.
- Unterhalb des einleitenden `<fieldset>`-Tags und vor den ersten Formularinhalten der Gruppe sollte eine Gruppenüberschrift (z.B. Formular) für die Elementgruppe vergeben werden.
 - ⇒ Schließen Sie den Gruppenüberschriftentext in die Tags `<legend>...</legend>` ein
- Vorteil bei der Verwendung
 - ⇒ Formatierung nach Wunsch über HTML/CSS
 - ⇒ Web-Browser kann Elementgruppen durch Linien oder ähnliche Effekte optisch sichtbar machen

Bitte machen Sie Ihre Eingaben

Vorname:

Zuname:

Auswahl: ☒

Attribute für Eingabefelder

■ Hinweistexte

- ⇒ werden bei Eingabefeldern angezeigt, bevor man eine Eingabe macht
- ⇒ `placeholder="Ihr Nachname"`



■ Tabulatorreihenfolge

- ⇒ normalerweise entsprechend der Reihenfolge in der HTML-Datei
- ⇒ oder explizit setzen mit Attribut `tabindex="1"` usw.

■ Tastaturkürzel definieren

- ⇒ `accesskey="x"` springt mit alt+x sofort in das entsprechende Eingabefeld
- ⇒ Dies ist im Browser nicht erkennbar und muss beschriftet werden !



2.1.3 HTML Formulare

Attribute zur Validierung von Eingabefeldern

■ required

```
<input type="email"
      required />
```

E-Mail

Bitte füllen Sie dieses Feld aus.

■ pattern

```
<input pattern="[0-9]{5}" name="plz"
      title="Fünfstellige Postleitzahl in Deutschland." />
```

Ohne required, darf das Feld trotz pattern auch leer bleiben!

PLZ

Bitte halten Sie sich an das vorgegebene Format:
Fünfstellige Postleitzahl in Deutschland.

■ min..max

```
<input name="bday" type="date" max="1994-12-31" />
```

Zusammenfassung

Jetzt wissen Sie alles um eine (statische) HTML-Seite zu entwickeln!

- Grundidee Formulare (Übertragung von Daten an den Web Server)
- Aufbau von Formularen

- ⇒ 1- und mehrzeiliges Textfeld (`<input type="text"... />` bzw. `<textarea> ... </textarea>`)
- ⇒ Listbox und Combobox (`<select...><option>...`)
- ⇒ Radiobuttons und Checkboxes
`<input type="radio" name="x"... />` bzw.
`<input type="checkbox"... />`)
- ⇒ Schaltflächen und verborgene Felder
`<input type="button" ... onclick... />` bzw.
`<input type="hidden"... />`
- ⇒ Abschicken von Formularen
`<input type="submit"... />`
- ⇒ Beschriftung von Formularelementen
`<label>`, `<fieldset>` und `<legend>`

Daten werden nur für Formularelemente übertragen, die innerhalb eines `<form>`s liegen und ein name-Attribut haben!
Wohin die Übertragung geht, legt das action-Attribut des `<form>`s fest.