

HTML:

Inhalte strukturieren

HTML-Gerüst

- HTML-Dateien bestehen aus
 - Tags (Auszeichnungen, Markups)
 - Inhalte (einfacher ASCII-Text oder Links)
- In der Regel besteht ein HTML-Grundgerüst aus einem Grund-Tag(<html>) in dem sich ein Kopfbereich (<head>) und ein Körperbereich (<body>) befinden

```
<!DOCTYPE html>
<html lang="de">
  <head>
    <meta charset="utf-8">
    <title> </title>
    <style> </style>
  </head>
  <body>

  </body>
</html>
```

HTML-Gerüst

- `<!DOCTYPE html>`: Gibt an, dass es sich um ein HTML5-Dokument handelt.
 - `<html lang="de">`: Beginn des HTML-Dokuments und Festlegung der Sprache auf Deutsch.
 - `<head>`: Beginn des Kopfbereichs, der Metadaten enthält.
 - `<meta charset="utf-8">`: Stellt sicher, dass das Dokument den UTF-8 Zeichensatz verwendet, der alle wichtigen Zeichen abdeckt.
 - `<title>Meine Webseite</title>`: Setzt den Titel des Dokuments, der im Browser-Tab angezeigt wird.
 - `<body>`: Beginn des Hauptinhaltsbereichs des Dokuments.
 - `<h1>Willkommen auf meiner Webseite!</h1>`: Eine Hauptüberschrift innerhalb des Körperbereichs.
-

Aufgabe: Quelltext inspizieren

- Schauen Sie sich über den entsprechenden Menüeintrag ihres Browsers den Quelltext einer Webseite an und identifizieren Sie die genannten Strukturelemente.



Faustregeln für Datei-/Verzeichnisnamen

- **Keine Umlaute oder ß:** Vermeiden Sie die Verwendung von Umlauten (ä, ö, ü) und dem scharfen S (ß).
- **Keine Akzente oder Sonderzeichen:** Verwenden Sie keine Akzente oder Sonderzeichen.
- **Keine Leerzeichen:** Verwenden Sie keine Leerzeichen in Datei- oder Verzeichnisnamen.
- **Nur Kleinbuchstaben:** Verwenden Sie nur Kleinbuchstaben von a bis z.
- **Ziffern und Bindestriche:** Verwenden Sie Ziffern (0-9) und Bindestriche (-) für die Benennung.
- **Beispiele:**
 - Richtig: meine-webseite.html
 - Falsch: Meine Webseite.html, meine-webseiteä.html, meine_webseite.html
- **Erklärung:**
 - Durch die Einhaltung dieser Regeln wird sichergestellt, dass die Dateien und Verzeichnisse auf allen Betriebssystemen und Servern problemlos funktionieren und keine Probleme mit der URL-Struktur auftreten.

Aufgabe: Die erste HTML-Seite

Schaffen sie mit einem Editor eine Datei *index.html* folgender Struktur:

```
<!DOCTYPE html>
<html lang="de">
  <head>
    <meta charset="utf-8">
    <title> </title>
    <style> </style>
  </head>
  <body>

  </body>
</html>
```

- Öffnen sie diese Datei mit einem Browser
-> lokal sichtbare Webseite
-

Dokumenttyp-Definition

- Die „Dokumenttyp-Definition“ (DTD) legt fest, welche Art von Auszeichnungssprache (z. B. HTML) und welche Version davon in einem Dokument verwendet wird. Das bedeutet, dass sie bestimmt, wie das Dokument strukturiert und geschrieben ist.
- HTML gibt es in verschiedenen Versionen, und die DTD hilft dem Browser oder einem anderen Programm zu verstehen, welche Version von HTML verwendet wird.
- Ein gültiges (valides) Dokument sollte oder muss eine solche Angabe enthalten, damit es korrekt interpretiert und angezeigt werden kann.

Dokumenttyp-Definition

Eine **Dokumenttyp-Definition (DTD)** enthält Regeln, die alle Dokumente erfüllen müssen.

Es wird u.a. geregelt

- Welche Elemente dürfen enthalten sein
- Welche Elemente dürfen innerhalb anderen Elemente enthalten sein.
- Welche Attribute können bei welchen Element verwendet werden
- Welche Angaben sind Pflicht, welche freiwillig

Dokumenttyp-Definition für HTML5: `<!DOCTYPE html>`

HTML - Tags

Beispiel mit Überschrift 1.Ordnung
(h = heading = Überschrift)

`<h1>Das ist meine Hauptüberschrift</h1>`

`<h1>` = Anfang der Überschrift

`</h1>` = Ende der Überschrift

HTML - Tags

"Standalone-Tags"

- Elemente, die keinen Inhalt haben und deshalb aus nur einem Tag bestehen könnten
- Beispiel Zeilenumbruch (br = break = Umbruch)
Alternative Schreibweisen:
 - `
`
 - `
</br>`
 - `
`

Verschachtelung von Elementen

Elemente können miteinander verschachtelt werden

- hierarchische Struktur

Komplexe HTML-Dateien enthalten sehr viele Verschachtelungen

```
<p>Das ist ein Absatz mit einer Liste:</p>
<ul>
  <li>Erster Eintrag</li>
  <li>Zweiter Eintrag
    <ul>
      <li>Unterpunkt 1</li>
      <li>Unterpunkt 2</li>
    </ul>
  </li>
  <li>Dritter Eintrag</li>
</ul>
```

Regeln beim Editieren von HTML

Vollständiges HTML-Gerüst

Lesbarkeit des Quelltexts:

- Verwenden Sie Zeilenumbrüche und Einrückungen, um den Quelltext lesbar zu gestalten.
- Gut formatierter Code erleichtert die Wartung und das Verständnis.

Einrückungen und Zeilenumbrüche:

- Einrückungen (Tabs oder Leerzeichen) haben keine Auswirkung auf die Darstellung im Browser, sondern dienen der besseren Lesbarkeit des Codes.

Erklärung:

- Die Einrückungen und Zeilenumbrüche im Code helfen Entwicklern, die Struktur des Dokuments zu verstehen und den Code leichter zu bearbeiten.

HTML: Strukturierung von Inhalten

```
<html lang="de">
  <head>
    <meta charset="utf-8">
    <title>Die erste Seite</title>
  </head>
  <body>
    <h2>Minnies Männer</h2>
    <ul>
      <li>Donald<br>
        14.3.68
      </li>
      <li>Goofy<br>
        15.11.66
      </li>
    </ul>
  </body>
</html>
```

Minnies Männer

- Donald
14.3.68
- Goofy
15.11.66

Aufgabe: Textstrukturierung und Links

- Verwenden sie weitere Tags wie
 - p, h1, h3, ...
- Fügen sie einen externen Link
`...`
ein und prüfen sie ihn.

Links (Verweise)

Inhalt:

- **Was sind Links (Verweise)?**
 - Links sind Verweise zu anderen Punkten oder Dokumenten.
 - Bei Klick auf einen Link wird normalerweise ein referenziertes Ziel angezeigt oder geöffnet.
 - Ziele können im HTML-Dokument selbst oder auch externe Dateien sein.

Erklärung:

- **Definition von Links:** Links sind wie Wegweiser im Internet. Sie zeigen uns, wo wir hingehen können, wenn wir auf sie klicken. Sie können zu anderen Seiten im selben Dokument, zu anderen Webseiten oder sogar zu Dateien wie Videos, PDF-Dokumenten oder Bildern führen.
- **Beispiel:** Stell dir vor, du bist auf einer Webseite über Pflanzen. Ein Link könnte dich zu einer anderen Seite mit Informationen über Rosen führen, oder zu einem PDF-Dokument mit Anleitungen zum Pflanzen von Samen

Nutzung von Links zur Navigation

Inhalt:

- **Wie werden Links zur Navigation genutzt?**
 - Links verknüpfen zusammenhängende HTML-Dokumente.
 - Navigation auf einer Website erfolgt mit Hilfe von Links.
 - Links können auch auf nicht-HTML-Dateien (z.B. Videos, Archive, Word-Dokumente, PDFs, Sounds) verweisen und einen Download einleiten, wenn kein Interpreter für die Datei bekannt ist.

Erklärung:

- **Navigation durch Links:** Links helfen uns, von einer Seite zur nächsten zu gelangen. Auf diese Weise können wir durch die gesamte Webseite navigieren.
- **Beispiel:** Auf einer Rezeptseite könntest du Links finden, die zu verschiedenen Rezepten führen. Ein Link könnte dich zu einem Video führen, das zeigt, wie man ein bestimmtes Gericht kocht. Ein anderer Link könnte dich zu einer PDF-Datei mit einer Einkaufsliste bringen.

Links (Verweise)

Link-Tag

`<a>`

Ziel definieren mit Attribut

`href="ziel"`

` HS-Hannover`

`Start`

Weitere Beispiele für Links

`Hochschule Hannover`

`Impressum`

`Über uns`

`Rezepte zum Nachlesen`

`Bild von mir`

`Unterlagen Download`

Weitere Beispiele für Links

Erklärung:

- **Links zu Webseiten und Dateien:** Diese Beispiele zeigen verschiedene Arten von Links:
 - Links zu anderen Webseiten (Hochschule Hannover).
 - Links zu internen Seiten (Impressum, Über uns).
 - Links zu Dateien (PDF-Dokumente, Bilder, ZIP-Archive).
 - **Beispiele:**
 - Der erste Link führt zu einer Service-Seite.
 - Der zweite Link führt zur Impressum-Seite der Webseite.
 - Der dritte Link führt zur "Über uns"-Seite.
 - Der vierte Link führt zu einem PDF-Dokument mit Rezepten.
 - Der fünfte Link zeigt ein Bild.
 - Der sechste Link lädt eine ZIP-Datei mit Unterlagen herunter.
-

Aufgabe: Links

Schaffen Sie drei Dateien, die untereinander vollständig verlinkt sind. Jede Datei soll ein "Hauptmenü" mit drei Einträgen enthalten.

z.B. auf S.2: [Seite 1](#)

Seite 2

[Seite 3](#)

Links (Verweise) - Zielfenster

Ziel im bestehenden Fenster oder neuem öffnen

Attribut **target**

- Wert **_self** = geöffnetes Fenster (Standard)
- Wert **_blank** = neues Fenster

```
<a href="https://www.hs-hannover.de" target="_blank">Zur HS</a>
```

Links (Verweise) – Seiteninterne Verweise

Um innerhalb eines Dokuments zu einer Stelle ("Anker") springen zu können, muss zunächst ein Ziel definiert sein

- Zu Elementen mit dem Attribut **id** oder **name** springen



Bei <a> Tags

Links (Verweise) – Seiteninterne Verweise

<p>

Zu Abschnitt 5

Zu Abschnitt 6

</p>

<h1 id="abschnittA">Abschnitt 1</h1>

.

.

.

<h1 id="abschnittE">Abschnitt 5</h1>

Abschnitt 6

Links (Verweise) – Seiteninterne Verweise

Auch seitenübergreifend kann direkt auf seiteninterne Verweise verwiesen werden

```
<a href="info.html#abschnitt3">Direkt zur Info</a>
```

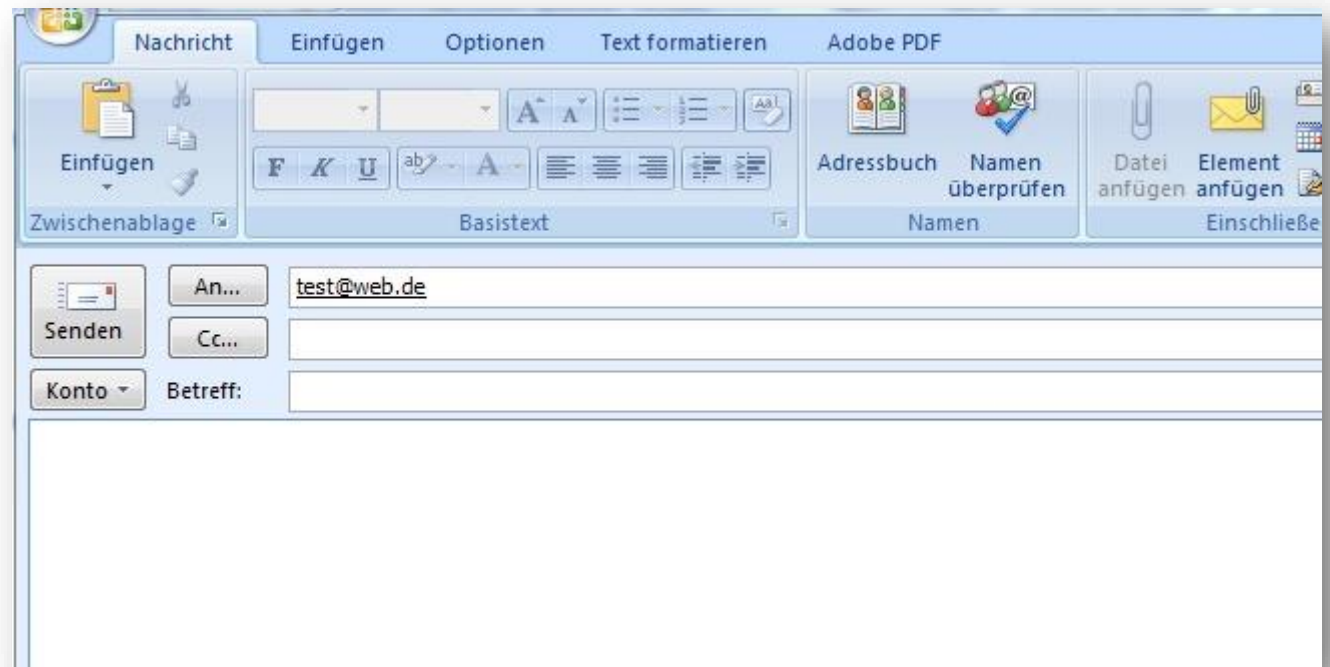

Links (Verweise) – E-Mail

Auch auf E-Mail-Adressen im Internet-Format können Verweise gesetzt werden

Bei Klick auf den Link wird das Standard-Email-Programm geöffnet und entsprechende Daten übergeben

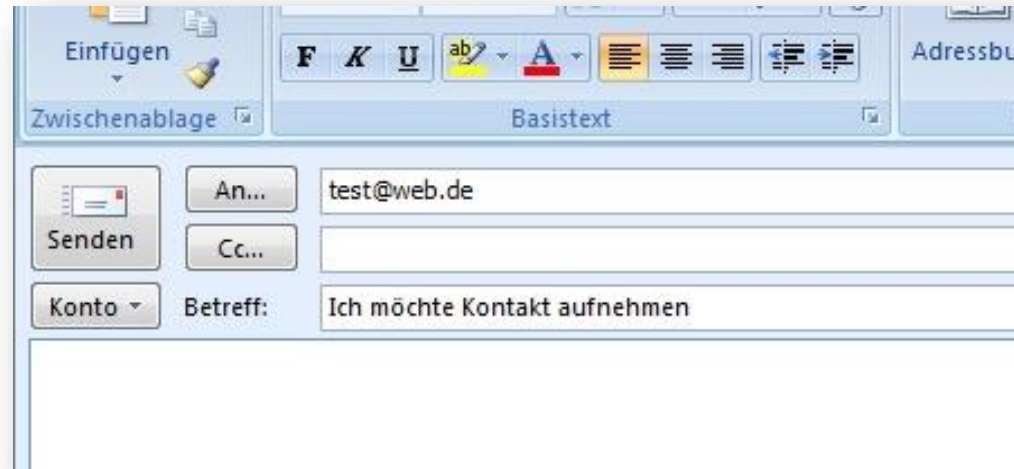
Links (Verweise) – E-Mail

`Mail`



Links (Verweise) – E-Mail

`Mail`



Links (Verweise) – E-Mail

Weitere Optionen:

- Weitere Empfänger: cc oder bcc
- vollständiges Empfängerschema
(Max Muster <test@web.de>)
- Subject: subject
- Body-Inhalt: body

Links (Verweise) – E-Mail

`Mail`





Hausaufgabe

- Überlegen Sie sich externe Links, auf die Sie in Ihrer Website verweisen können
- Planen Sie eine Menüstruktur für Ihre Inhalte

Kommentare

Innerhalb einer HTML-Datei können Kommentare eingefügt werden

Kommentare werden von Browsern ignoriert

Kommentare dienen für Anmerkungen im Text, Angaben zu Autor, Datum, etc. und Auskommentieren von Code

Kommentare sind für den Entwickler ein wichtiges Werkzeug und eine große Hilfe, besonders im Team.

Kommentare

<!--

Autor: Mr. X

Datum: 01.01.1970

-->

Angaben

<!-- Jetzt kommt eine Überschrift -->

<h1>Jetzt geht's los</h1>

Anmerkung, Notiz

<!--

<p>

Lorem ipsum lorem

</p>

-->

Code auskommentieren

Attribute

Tags können zusätzliche Angaben enthalten (Attribute)

Zugewiesene Werte sollen in Anführungsstrichen stehen

Beispiele:

```
<a href="http://www.google.de" target="_blank">Link</a>
```

```

```

Textstrukturierung

Überschriften

- h1 bis h6

Textabsätze

- p

Zeilenumbbruch

- br,

Zitate und Adressen

- q, blockquote, cite

Auszeichnung auf Textebene (sollten in der Regel durch CSS ersetzt werden)

- em, strong, small, dffn, abbr, code, var, samp, kbd, sub, sup, i, b, bdo, time, mark ,ruby, rt, rp

Allgemeine Elemente für Textbereiche

- div, span

Trennlinien

- hr

Überschriften - Attribute

align – Textausrichtung

~~<h2 align="right">Überschrift</h2>~~

Achtung: Solche "Formatierungen" sollten nicht mehr mit HTML durchgeführt werden, sondern mit CSS.

Textabsätze

Gliedern von Text in Absätzen

`<p>` (paragraph)

Automatische Umbrüche und Abstände

`<p>`

Das ist der Inhalt von einem Absatz

`</p>`

`<p>`

Nächster Absatz

`</p>`

Geschütztes Leerzeichen

Leerzeichen, das nicht automatisch umgebrochen werden kann

Beispiel

```
<!DOCTYPE html>
<html lang="de">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Beispiel für geschütztes Leerzeichen</title>
</head>
<body>
  <p>Hier ist ein Beispiel für ein geschütztes Leerzeichen:</p>
  <p>Die Telefonnummer lautet: 01234&nbsp;56789.</p>
  <p>Der Name des Autors ist: Max&nbsp;Mustermann.</p>
  <p>Der Preis beträgt: 49,&nbsp;99&nbsp;&euro;.</p>
</body>
</html>
```

Beispiel

In diesem Beispiel:

- 01234 56789 stellt sicher, dass die Telefonnummer nicht am Leerzeichen getrennt wird.
- Max Mustermann verhindert, dass der Name des Autors an der Stelle des Leerzeichens getrennt wird.
- 49, 99 € sorgt dafür, dass der Betrag und das Euro-Zeichen zusammenbleiben und nicht getrennt werden.

Das geschützte Leerzeichen wird durch den HTML-Code ` ` dargestellt und hilft, die Lesbarkeit und Formatierung des Textes zu verbessern

Zitate und Adressen

Zitate

<blockquote>

<p>Wir dürfen jetzt nur nicht den Sand in den Kopf stecken!</p>

</blockquote>

Zitate im Fließtext

Und er sagt <q>Wir dürfen...</q>


Werk, das zitiert oder erwähnt wird, auszeichnen

Und er sagt <q>Mailand oder Madrid – Hauptsache Italien
<cite>Andreas Möller</cite></q>

Adresse

<address>Max Mustermann, max@muster.de</address>

Aufgabe: Kapitel 2 im Web Design Playground

 WEB DESIGN PLAYGROUND

Run Code ▶ MENU ≡

2.1: Laying the Basic Page Structure

◀ Previous Page

Next Page ▶

Laying the Basic Page Structure

All your web page projects, from the simplest page to the most sophisticated business site, will begin with the same basic structure, which is shown in the HTML Editor.

HTML

```
1 <!DOCTYPE html>
2 <html lang="en">
3   <head>
4     <meta charset="utf-8">
5     <title></title>
6     <style></style>
7   </head>
8   <body>
9   </body>
10 </html>
```

CSS

```
1
```

Hide Editors New Sandbox

Allgemeine Elemente für Inhalte

Allgemeines Inline-Element, zum Formatieren mit CSS, schafft einen „Inline-Container“

```
<span> </span>
```

```
<span style="color:#ff0000; font-weight:bold">Hallo  
Welt</span>
```

Aufgabe

- Nutzen Sie `<div>` und `` im Body einer HTML-Datei und formatieren Sie diese Tags über Style-Regeln im Head. Trennen Sie die beiden Bereiche im Body mit einer horizontalen Linie (`<hr>`).

Listen

Liste mit Aufzählungszeichen und automatischer Ausrichtung

Eine Liste besteht aus einem List-Tag und ein oder mehreren Listpunkt-Tags

Listen

Punkt 1

Punkt 2

Punkt 3

- Punkt 1
- Punkt 2
- Punkt 3

Listen

Listen können beliebig verschachtelt werden

```
<ul>  
  <li>  
    Punkt 1  
    <ul>  
      <li>Unterpunkt 1</li>  
      <li>Unterpunkt 2</li>  
    </ul>  
  </li>  
  <li>Punkt 2</li>  
  <li>Punkt 3</li>  
</ul>
```

- Punkt 1
 - Unterpunkt 1
 - Unterpunkt 2
- Punkt 2
- Punkt 3

Nummerierte Liste

Punkt 1

Punkt 2

Punkt 3

1. Punkt 1
2. Punkt 2
3. Punkt 3

Definitionsliste

<dl>

<dt>bzw.</dt>

<dd>Beziehungsweise</dd>

<dt>u. a.</dt>

<dd>unter anderem, unter anderen</dd>

<dd>und andere, und anderes</dd>

</dl>

bzw.

Beziehungsweise

u. a.

unter anderem, unter anderen
und andere, und anderes

Liste

Listen können mit CSS formatiert werden (mehr in den nächsten Vorlesungen)

Listen werden häufig für die Navigation in Internetseiten verwendet (Menüs)

Aufgabe: Listen

- Schreiben sie eine verschachtelte ungeordnete Liste mit drei Ebenen
- Experimentieren Sie mit einer nummerierten Liste (ol) statt der ungeordneten (ul)

Präformatierter Text


Ausgabe wie im Code (Umbrüche, Einrückungen)

Wird z.B. für Programmlistings angewandt

```
<pre>  
  <code>  
    &lt;?php  
        echo "Hallo Welt"  
    &gt;  
  </code>  
</pre>
```

```
<?php  
    echo "Hallo Welt"  
>
```

Aufgabe: Kapitel 3 im Web Design Playground

 WEB DESIGN PLAYGROUND

Run Code ▶ MENU ≡

3.1: Working with Paragraphs

◀ Previous Page

Next Page ▶

Working with Paragraphs

Web browsers ignore white space, including carriage returns created by pressing Enter or Return, which is normally how you'd separate text into paragraphs in a text editor or word processor. The most common way to create a paragraph in HTML is to place a `<p>` (for paragraph) tag at the beginning of the text, and a closing `</p>` tag at the end of the text.

HTML

1

CSS

1

Hide Editors

New Sandbox



Hausaufgabe

- Bereiten Sie für jede geplante Seite eine Datei mit HTML-Grundgerüst vor
- Schaffen Sie Ihr "Hauptmenü" als Liste von Menüpunkten
- Passen Sie dieses Hauptmenü für jede Datei entsprechend an