



# HTML(5) Grundlegende Elemente



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



- Die Hypertext Markup Language (HTML), ist eine textbasierte Auszeichnungssprache zur Strukturierung digitaler Dokumente wie Texte mit Hyperlinks, Bildern und anderen Inhalten.
- HTML-Dokumente sind die Grundlage des World Wide Web und werden von Webbrowsern dargestellt.
- Neben den vom Browser angezeigten Inhalten können HTML-Dateien zusätzliche Angaben in Form von Metainformationen enthalten, z. B.
  - über die im Text verwendeten Sprachen,
  - den Autor oder
  - den zusammengefassten Inhalt des Textes.



- HTML dient als Auszeichnungssprache dazu, einen Text semantisch zu strukturieren, nicht aber zu formatieren.
  - HTML ist keine Programmiersprache!
- Die visuelle Darstellung ist nicht Teil der HTML-Spezifikationen und wird durch den Webbrowser und Gestaltungsvorlagen wie CSS bestimmt.
- Ein ähnliches Konzept zur logischen Beschreibung von Dokumenten steht hinter dem Satzsystem TeX/LaTeX, das im Unterschied zu HTML jedoch auf die Ausgabe per Drucker auf Papier zielt.



```
<!DOCTYPE html>
<html lang="de">
  <head>
    <meta charset="utf-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Titel</title>
  </head>
  <body>
    . . .
  </body>
</html>
```

- Dokumenttyp-Deklaration
- Head (<head>...</head>): Metadaten, Skript- und Style-Referenzen
- Body (<body>...</body>): Der sichtbare Inhalt der Webseite, also Text mit Überschriften, Verweisen, Grafikreferenzen usw.



```
<body>  
  <h1>Überschrift 1</h1>  
  <h2>Überschrift 2</h2>  
  <h3>Überschrift 3</h3>  
  <h4>Überschrift 4</h4>  
  <h5>Überschrift 5</h5>  
  <h6>Überschrift 6</h6>  
</body>
```

**Überschrift 1**

**Überschrift 2**

**Überschrift 3**

**Überschrift 4**

**Überschrift 5**

**Überschrift 6**



```
<body>  
  <h1>Überschrift 1</h1>  
  <p>Ich bin ein Absatz.</p>  
  <p>Und ich bin ein weiterer Absatz.</p>  
</body>
```

# Überschrift 1

Ich bin ein Absatz.

Und ich bin ein weiterer Absatz.

- Absätze dienen der Gliederung eines Textes.
- p umfasst einen Textblock - es steht nicht für den Zwischenraum zwischen zwei Absätzen.
- Absatz-Elemente dürfen keine anderen blockerzeugenden Elemente wie Überschriften, Textabsätze oder Listen enthalten.



```
<body>
<h1>Überschrift 1</h1>
<p>Ich bin ein Absatz. Mein Inhalt wird standardmäßig in einer Zeile angezeigt.</p>

<p>Um eine neue Zeile zu erzwingen, <br>wird das br-Element verwendet.</p>

<p>Je nachdem wie breit der Bildschirm ist, <br>
kann ein Absatz auch automatisch auf mehrere Zeilen aufgeteilt werden.</p>
</body>
```

## Überschrift 1

Ich bin ein Absatz. Mein Inhalt wird standardmäßig in einer Zeile angezeigt.

Um eine neue Zeile zu erzwingen,  
wird das br-Element verwendet.

Je nachdem wie breit der Bildschirm ist,  
kann ein Absatz auch automatisch auf mehrere Zeilen aufgeteilt werden.

- Das br-Element sollte ausschließlich für Zeilenumbrüche, die Bestandteil des Inhalts sind, genutzt werden. Zum Beispiel Gedichte oder Adressen.
- Dabei sind ausnahmsweise zwei Schreibweisen erlaubt:  
<br> und <br />



```
<body>
  <h1>Überschrift</h1>
  <p>Ich bin ein Absatz.</p>
  <hr>
  <p>Ich bin ein weiterer Absatz.</p>
</body>
```

## Überschrift

Ich bin ein Absatz.

---

Ich bin ein weiterer Absatz.

- Das Element hr kennzeichnet einen thematischen Bruch.
- Ursprünglich wurden solche Trenner als waagerechte Linien dargestellt.
- In HTML5 bekommt das hr-Element die Bedeutung eines Themenwechsels auf Absatz-Ebene: paragraph-level thematic break
- Sichtbare Trennlinien dienen auch visuell der Abgrenzung von nicht unmittelbar zusammengehörigen Textabschnitten.
- hr fügt also eine Trennlinie ein und erzeugt dabei einen eigenen Absatz.





```
<body>
  <h1>Lernliste</h1>
  <ol>
    <li>PR3 üben</li>
    <li>SE2 Übung abgeben</li>
    <li>Für DM-Klausur lernen</li>
  </ol>
</body>
```

## Lernliste

1. PR3 üben
2. SE2 Übung abgeben
3. Für DM-Klausur lernen

- Das Element ol bezeichnet eine geordnete oder nummerierte Liste; also eine Liste, bei der die Reihenfolge der Elemente von Bedeutung ist.
- Mit <li> beginnt ein neuer Punkt innerhalb der Liste.  
li = list item = Listeneintrag



- Das Verschachteln von nummerierten Listen ist möglich, bewirkt aber keine Gesamtnumerierung.
- Automatische Nummerierungshierarchien wie 1, 1.1, 1.1.1, sind mit HTML allein nicht möglich.
- Dazu bedarf es der Unterstützung von CSS, indem man Listen mit hybrider Nummerierung formatiert.
- Sie können aber in HTML den Startwert einer Liste festlegen und bei einem beliebigen Listenelement neu setzen.
  - Dazu nutzen Sie das Attribut start des ol-Elementes.
  - Erlaubte Werte für dieses Attribut sind ganze Zahlen:

```
<ol start="5">  
  <li>PR3 üben</li>  
  <li value="1">SE2 Übung abgeben</li>  
  <li>Für DM-Klausur lernen</li>  
</ol>
```

5. PR3 üben
1. SE2 Übung abgeben
2. Für DM-Klausur lernen



```
<body>
  <h1>Einkaufsliste</h1>
  <ul>
    <li>3 Äpfel</li>
    <li>2 Brötchen</li>
    <li>300 Gramm Rinderfilet</li>
  </ul>
</body>
```

## Einkaufsliste

- 3 Äpfel
- 2 Brötchen
- 300 Gramm Rinderfilet

- Das Element ul beschreibt eine Aufzählungsliste.
- Aufzählungslisten sind z.B. von Bedeutung, um Produkteigenschaften oder Argumente für eine These übersichtlich darzustellen.
- Bei einer Aufzählungsliste werden die Listeneinträge von grafischen Browsern standardmäßig mit einem Aufzählungszeichen (Bullet) versehen.
- Das Verschachteln von Listen ist ebenfalls möglich. Zwischen <li> und </li> darf eine komplette weitere Liste stehen.
  - Auch andere Listentypen sind dabei erlaubt.



```
<body>
  <h1>Abkürzungen</h1>
  <dl>
    <dt>allg.</dt>
    <dd>allgemein</dd>
    <dt>bez.</dt>
    <dd>bezüglich</dd>
    <dd>bezahlt</dd>
  </dl>
</body>
```

# Abkürzungen

allg.  
allgemein

bez.  
bezüglich  
bezahlt

- Definitionslisten, seit HTML5 Beschreibungslisten, sind beispielsweise für Glossare gedacht.
- Ein Glossar besteht meist aus einer Liste von Einträgen, die wiederum aus mindestens einem zu erklärenden Sachverhalt nebst mindestens einer Erklärung besteht.



- Eine Definitions- bzw. Beschreibungsliste wird mit `<dl>` eingeleitet und mit `</dl>` beendet.
- Der zu erläuternde Ausdruck steht zwischen `<dt>...</dt>` als description list term.
- `<dd>...</dd>` als description list description umschließt die Erläuterung.
- Die Abfolge von dt- und dd-Elementen innerhalb einer dl-Liste ist nicht streng geregelt:
  - Es dürfen auch mehrere dt- oder dd-Elemente hintereinander folgen.
  - Zum Beispiel könnten Begriffe mit einer Erklärung abgehandelt werden oder mehrere Bedeutungen für einen Begriff existieren.
- Das Verschachteln von Definitionslisten ist ebenfalls möglich.
  - Dadurch können Sie baumartige Strukturen im Text abbilden.



```

```

- Mit dem img-Element können Sie Grafikdateien in Ihre HTML-Dokumente einbinden.
- img ist ein sogenanntes Standalone-Tag.
- Folgende Angaben sind immer nötig:
  - src für die Referenz der Bilddatei. Diese kann wie in HTML üblich relativ oder absolut angegeben werden. Die Grafik selbst ist nicht Bestandteil des HTML-Dokuments, sondern wird in HTML an der Stelle referenziert, an der sie angezeigt werden soll.
  - alt gibt den Alternativtext an, der angezeigt wird, wenn das Bild nicht geladen werden kann oder soll. Screen Reader lesen den Text vor.



# Bilder, Figures, Länge und Höhe

## Image: img, figure, width, height



```
<figure>
  
  <figcaption>Die Hochschule Mannheim entstand im Jahr 2006 durch die Fusion...</figcaption>
</figure>
```



## hochschule mannheim

Die Hochschule Mannheim entstand im Jahr 2006 durch die Fusion der Fachhochschule Mannheim – Hochschule für Technik und Gestaltung und der Fachhochschule für Sozialwesen. Sie ist eine deutsche Fachhochschule mit Sitz in Mannheim.

- width bzw. height geben die Bildbreite bzw. -höhe an.
- Seit HTML5 sind nur noch Angaben in Pixeln erlaubt; es ist keine Einheit mehr anzugeben.



- Sie sollten jedoch stets die Breite und Höhe der Grafik im Quelltext mit angeben.
- Dadurch entnimmt der Web-Browser bereits der HTML-Datei, wie groß die Grafik ist, und muss nicht warten, bis er die entsprechende Header-Information der Grafikdatei ausgelesen hat.
- So kann er die gesamte Web-Seite bereits am Bildschirm aufbauen und bei noch nicht eingelesenen Grafiken erst mal eine entsprechend große Freifläche anzeigen.
- Ansonsten wartet der Browser mit der Anzeige der Web-Seite, bis er alle nötigen Größenangaben aus eingebundenen Grafikdateien eingelesen hat, oder er muss den Bildschirmaufbau korrigieren, was nicht sehr schön aussieht.





```
<p>  
  <a href="http://www.tagesschau.de/">ARD Tagesschau</a> - ein externer Verweis<br/>  
  <a href="./Beispiel/blabla.html">SelfHTML</a> - ein interner Verweis  
</p>
```

```
<a href="https://google.com">  
  <h2>Google.de</h2>  
  <p>Google ist die größte Suchmaschine im Internet.</p>  
</a>
```

- Der erste Link führt zu einem anderen Webangebot: extern.
- Der zweite Link führt zu einer anderen Seite der gleichen Domain: intern.
- Der dritte Link enthält mehrere Elemente.

[ARD Tagesschau](http://www.tagesschau.de/) - ein externer Verweis  
[SelfHTML](#) - ein interner Verweis

[Google.de](https://google.com)

[Google ist die größte Suchmaschine im Internet.](#)



- Mit einem a-Element können Sie einen Link zu einem anderen Dokument oder zu einem Anker (anchor) setzen.
- Aber erst das href-Attribut(hypertext reference) verwandelt das a-Element in einen Verweis, indem Sie ihm das gewünschte Verweisziel als Wert zuweisen.
- Hyperlinks sind ein entscheidender Bestandteil jedes Hypertext-Projekts und der „intelligente Mehrwert“ des World Wide Web.
- Ursprünglich war ein a-Element mit einem name-Attribut ein Sprungziel eines Verweises, dies ist jedoch mittlerweile obsolet.
- Heute kann jedes beliebige Element mit einem id-Attribut als Verweisziel fungieren.



- Als Inhalt des a-Elements, also zwischen `<a>` und `</a>`, notieren Sie den Text, der dem Anwender als Verweis angeboten wird.
- Dieser wird bei den meisten Web-Browsern andersfarbig und meist unterstrichen dargestellt.



- Als Anker bzw. Sprungziel ist jedes id-Attribut in jedem beliebigen Element geeignet, zum Beispiel

```
<h2 id="textmarke3">Kapitel 3</h2>
```

- Der Verweis zu dieser Stelle auf der gleichen Seite würde dann lauten

```
<a href="#textmarke3">zum Kapitel 3</a>
```



- Mit dem target-Attribut können Sie ein Zielfenster für Verweise festlegen, so dass ein Verweisziel im aktuellen oder einem neu geöffneten Browserfenster oder Tab angezeigt wird.
- Folgende Werte sind dabei möglich:
  - `_self`: öffnet im aktuellen Fenster (Standardwert)
  - `_blank`: öffnet in neuem Fenster
  - `_parent`: Elternfenster
  - `_top`: oberstes Fenster
  - `<name>`: eigener Fenstername für vorhandenes oder neues Fenster

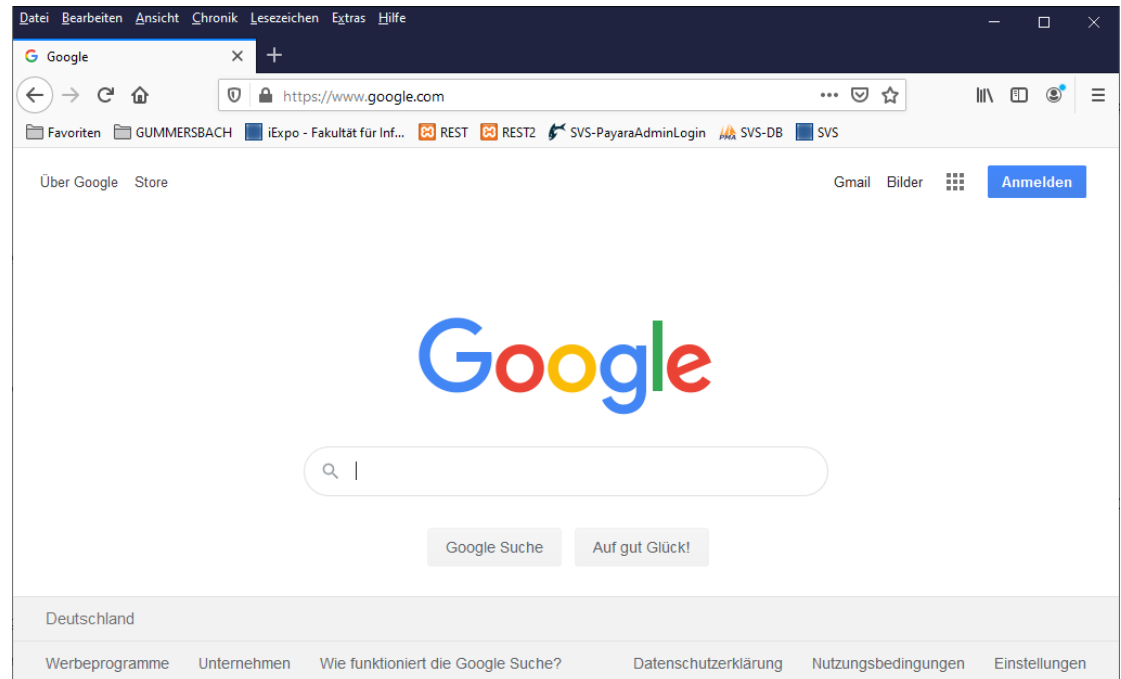


# Neues Fenster hinter dem Verweis Anchor: a target="\_blank"



```
<a href="https://google.com" target="_blank">Öffne Google</a>
```

Öffne Google





```
<a href="http://german.imdb.com/" tabindex="4">Movie Database</a>  
<a href="http://www.theonlineblues.com/" tabindex="2">Chart Lyrics</a>  
<a href="http://www.meinestadt.de/" tabindex="1">Meine Stadt</a>  
<a href="http://paperball.fireball.de/" tabindex="3">Paperball</a>
```

- Es ist möglich, mit Hilfe der Tabulator-Taste die Verweise einer HTML-Datei anzuspringen.
- Mit der Return-Taste ist der Verweis dann ausführbar.
- Normalerweise werden die Verweise dabei in der Reihenfolge angesprungen, in der sie in der Datei definiert sind.
- Sie können jedoch mit dem Universalattribut `tabindex` eine andere Reihenfolge festlegen.
- `tabindex`s beziehen sich stets auf das gesamte angezeigte Dokument. Dabei werden auch Formulare sowie Verweisbereiche in Grafiken und Objekten mit einbezogen.




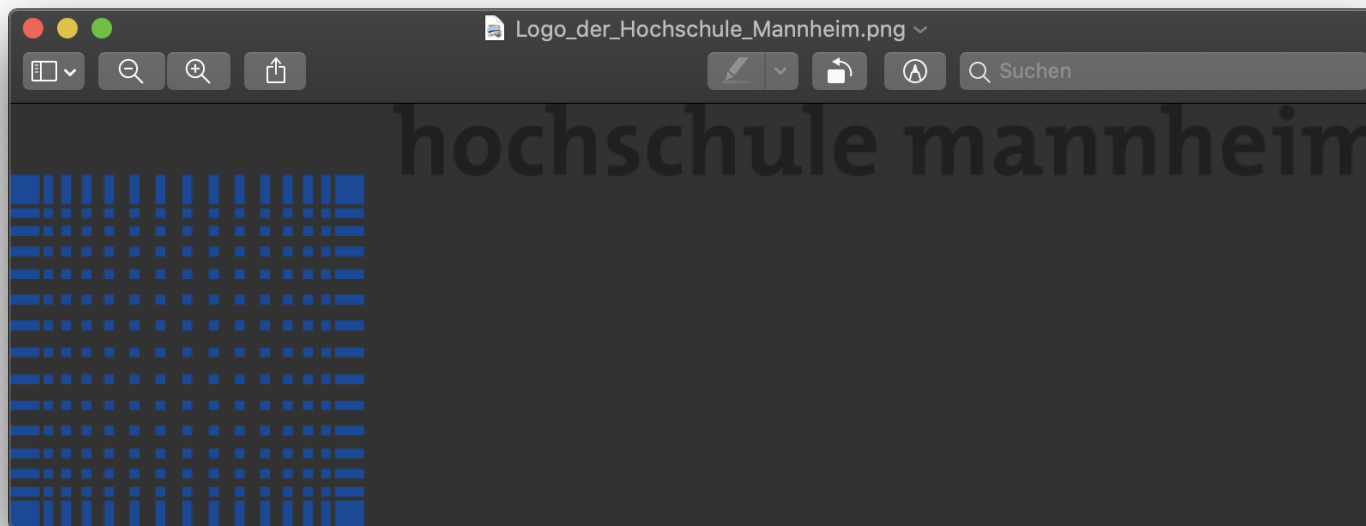
- Durch die Angabe von `download` öffnet sich ein Fenster, bei dem sich der Benutzer entscheiden kann, die Zieldatei im Browser mit einem vom Browser vorgeschlagenen Programm zu öffnen oder auf der Festplatte zu speichern.
- Beim Ausführen der Links wird in Abhängigkeit von den Browser-einstellungen die Ressource direkt gespeichert oder zum Download angeboten.
- Für den zweiten Link wird durch den Wert des `download`-Attributes ein Dateiname vorgeschlagen.
  - Bis auf die Microsoft-Browser Edge und Internet Explorer verhalten sich alle aktuellen Browser-Versionen spezifikationsgemäß, nur Edge ignoriert einen vorgeschlagenen Dateinamen
  - Der Internet Explorer kennt das `download`-Attribut gar nicht.





```
<a href="logo_hs.png" download="Logo_der_Hochschule_Mannheim">  
    
</a>
```

Name	Größe	Art	Hinzugefügt am
 Logo_der_Hochschule_Mannheim.png	21 KB	PNG-Bild	Heute, 18:12





- Eine Tabelle ist eine geordnete Zusammenstellung von Texten oder Daten.
- Die darzustellenden Inhalte werden dabei in Zeilen und Spalten gegliedert, die grafisch aneinander ausgerichtet werden.
- Sinnvolle Einsatzbereiche für Tabellen sind beispielsweise:
  - Kalender
  - Stundenpläne und Fahrpläne
  - TV-Programm
  - Statistiken
- Früher wurden Tabellen auch zur Layout-Gestaltung und Aufteilung von HTML-Seiten verwendet.
  - Dies ist jedoch strikt zu vermeiden, u.a. da Suchmaschinen HTML-Tabellen stets als Content interpretieren!
  - Statt dessen sollten zur Layout-Gestaltung div-Container mit CSS-Formatierung verwendet werden!



```
<table>
```

```
  <thead>
```

```
    <tr> <th> </th> <th> </th> <th> </th> </tr>
```

```
  </thead>
```

```
  <tbody>
```

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

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

```
  </tbody>
```

```
</table>
```



```
<table> <!-- Styles (z.B. border="1") nur mit CSS! -->
  <thead>
    <tr>
      <th>Kopf 1</th><th>Kopf 2</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>Inhalt 1</td><td>Inhalt 2</td>
    </tr>
  </tbody>
  <tfoot>
    <tr>
      <td>Fuss 1</td><td>Fuss 2</td>
    </tr>
  </tfoot>
</table>
```

<b>Kopf 1</b>	<b>Kopf 2</b>
Inhalt 1	Inhalt 2
Fuss 1	Fuss 2



- Dabei enthält der Tabellenkopf die Zeilen der einfachen Tabelle, die ausschließlich Kopfzellen bzw. Spaltenüberschriften enthalten.
- Der Tabellenkörper enthält die Zeilen, die ausschließlich Datenzellen enthalten.
- Der Tabellenfuß enthält zusammenfassende oder die Tabelle erläuternde Daten.
- Wenn die Tabelle im Ausdruck größer als eine Seite ist, werden die thead- und tfoot-Teile beim Drucken auf jeder Seite wiederholt.



- Das Attribut **colspan** erlaubt es, eine Tabellenzelle nach rechts über mehrere Spalten auszudehnen.
- Das Attribut **rowspan** erlaubt es, eine Tabellenzelle nach unten über mehrere Zeilen auszudehnen.
- Die beiden Attribute sind auch kombinierbar.



```
<table> <!-- Styles (z.B. border="1") nur mit CSS! -->
<tbody>
  <tr>
    <td>1. Spalte</td>
    <td>2. Spalte</td>
    <td>3. Spalte</td>
  </tr>
  <tr>
    <td rowspan="3" colspan="2">1. Spalte</td>
    <td>3. Spalte</td>
  </tr>
  <tr>
    <td>3. Spalte</td>
  </tr>
  <tr>
    <td>3. Spalte</td>
  </tr>
  <tr>
    <td>1. Spalte</td>
    <td>2. Spalte</td>
    <td>3. Spalte</td>
  </tr>
</tbody>
</table>
```

1. Spalte	2. Spalte	3. Spalte
1. Spalte		3. Spalte
		3. Spalte
		3. Spalte
1. Spalte	2. Spalte	3. Spalte



```
<h1>Hamlets Worte</h1>
```

```
<p>
```

```
    Wenn es bei einer Klausur um alles geht, so fallen oft Shakespeares  
    Worte <q cite="https://quelle.com">Sein oder Nichtsein, das ist hier die Frage</q>.
```

```
</p>
```

# Hamlets Worte

Wenn es bei einer Klausur um alles geht, so fallen oft Shakespeares Worte "Sein oder Nichtsein, das ist hier die Frage".

- Das Element q dient der Auszeichnung von Zitaten im Fließtext.
- Verwenden Sie das Attribut cite, um eine online-Quelle anzugeben.





```
<p>
```

```
Ein <em>Begriff</em> ist ein verbaler Griff nach einer losen Idee.
```

```
</p>
```

```
<p>
```

```
Bitte gehen Sie mit den Möglichkeiten des em-Elements <em>klug und bewusst</em> um.
```

```
</p>
```

Ein *Begriff* ist ein verbaler Griff nach einer losen Idee.

Bitte gehen Sie mit den Möglichkeiten des em-Elements *klug und bewusst* um.

- Wie genau der Browser die Elemente zur Textauszeichnung darstellt, darauf haben Sie mit HTML allein keinerlei Einfluss.
- Die Browser benutzen zwar default-Einstellungen, diese können sich aber von Browser zu Browser unterscheiden.
- Mit CSS können Sie Ihre Elemente jedoch nach Wunsch formatieren.



```
<p>
```

```
<strong>Beachten Sie:</strong>
```

```
strong-Elemente sind nicht nur visuell kräftig, sie wirken wie kleine Überschriften.
```

```
</p>
```

```
<ul>
```

```
<li><strong>Label 1:</strong> Mehrzeiliger <br/>Inhalt</li>
```

```
<li><strong>Label 2:</strong> Mehrzeiliger <br/>Inhalt</li>
```

```
</ul>
```

**Beachten Sie:** strong-Elemente sind nicht nur visuell kräftig, sie wirken wie kleine Überschriften.

- **Label 1:** Mehrzeiliger  
Inhalt
- **Label 2:** Mehrzeiliger  
Inhalt

- Wie em, nur stärker :-)



<p>

Herzofit - Jederzeit fit wie ein Turnschuh<br/>

<small>Zu Risiken und Nebenwirkungen lesen

Sie die Packungsbeilage und fragen Sie Ihren Arzt oder Apotheker.</small>

</p>

## Herzofit - Jederzeit fit wie ein Turnschuh

Zu Risiken und Nebenwirkungen lesen Sie die Packungsbeilage und fragen Sie Ihren Arzt oder Apotheker.

- Mit dem Element `small` wird ein Teil eines Fließtextes als weniger bedeutend ausgezeichnet; typisch für „das Kleingedruckte“.
- Das `small`-Element war ursprünglich ein rein präsentationsbezogenes Element ohne semantische Bedeutung für kleiner dargestellten Text.

```
<h1>Allgemeines Blockelement</h1>
<div>
  <h2>Listenüberschrift</h2>
  <ul>
    <li>Listenpunkt 1</li>
    <li>Listenpunkt 2</li>
    <li>Listenpunkt 3</li>
  </ul>
</div>
```

## Allgemeines Blockelement

### Listenüberschrift

- Listenpunkt 1
- Listenpunkt 2
- Listenpunkt 3

- Das div-Element ist dazu gedacht, mehrere Elemente wie Text, Grafiken, Tabellen usw., in einen gemeinsamen Bereich einzuschließen (hier: grau).
- Dieses allgemeine Element bewirkt nichts weiter als dass es in einer neuen Zeile des Fließtextes beginnt.
  - Divs blockieren standardmäßig die komplette Zeile, ansonsten haben sie keine Eigenschaften.
- Es ist dazu gedacht, Bereiche zu erzeugen, die mit Hilfe von CSS formatiert werden können.



```
<div>
```

```
  <p>Ein Container mit einem Absatz.</p>
```

```
  <p>Und einem <span id="test">weiteren</span> Absatz.</p>
```

```
</div>
```

Ein Container mit einem Absatz.

Und einem **weiteren** Absatz.

- Das span-Element ist ein Element, das Text und andere Inline-Elemente enthalten kann, selbst aber keine semantische Bedeutung hat und nichts bewirkt.
- Es ist nur dazu gedacht, um mit Hilfe von CSS formatiert zu werden.
- Wie das genau funktioniert, wird später behandelt.



```
<body>
  <header><h1>Willkommen!</h1></header>
  <nav>
    <h2>Navigation</h2>
    <ul>
      <li><a...> Seite 1</a></li>
      <li><a...> Seite 2</a></li>
      <li><a...> Seite 3</a></li>
    </ul>
  </nav>
  <main>
    . . .
  </main>
  <footer><p>Copyright Frank Dopatka</p></footer>
</body>
```

- header definiert den „Kopf“ der Webseite, oftmals mit einem Logo und dem Titel der Webseite.
- main definiert den Hauptteil der Webseite und darf nur einmal enthalten sein.
- footer definiert den „Fußteil“ der Webseite, oftmals mit einem Verweis auf das Impressum und Copyright.

- Das nav-Element umschließt Navigationsleisten und Menüs, wobei es neben einer ungeordneten Liste mit den Verweisen auch eine Überschrift oder ähnliches enthalten kann.



- Ein Sonderzeichen ist ein Schriftzeichen, das in einer Programmier- oder Auszeichnungssprache eine besondere Bedeutung hat und deshalb maskiert oder durch andere Zeichen oder Zeichenfolgen ersetzt werden muss, wenn es im normalen Text vorkommen soll.
- Ebenso kann es sein, dass besondere Zeichen nicht im Standard-Zeichensatz vorkommen und daher besonders codiert bzw. dargestellt werden müssen.

< &lt;	Ä &Auml;	<code>&lt;p&gt;Römer üben oft mit Älteren.&lt;/p&gt;</code>
> &gt;	ä &auml;	<code>&lt;p&gt;R&amp;ouml;mer &amp;uuml;ben oft mit &amp;Auml;lteren.&lt;/p&gt;</code>
& &amp;	Ö &Ouml;	
© &copy;	ö &ouml;	R?mer ?ben oft mit ?lteren.
® &reg;	Ü &Uuml;	Römer üben oft mit Älteren.
™ &trade;	ü &uuml;	