

Informatikerinnen EFZ und Informatiker EFZ

Lehrbeginn 2024

Lernziele

- Ihr könnt eine mehrteilige Website mit **HTML**-Befehlen erstellen und validieren.
- Ihr könnt Webseiten mit **CSS**-Befehlen gestalten und validieren.
- Ihr könnt mit HTML und CSS einfache **Formulare** erstellen und Benutzereingaben mithilfe von **regulären Ausdrücken** validieren.
- Ihr könnt eine Website so gestalten, dass sie sowohl auf Smartphones und grossen Bildschirmen gleich gut benutzbar sind (**responsive design**).
- Ihr könnt eure Website mit einem geeigneten Tool **veröffentlichen**, sodass sie im Internet angeschaut werden kann. ◆

Grundlagen HTML

Reguläre Ausdrücke

Grundlagen CSS

Responsive Design

Webseite veröffentlichen



Was bedeutet die Abkürzung HTML?

- HTML steht für **H**ypertext **M**arkup **L**anguage.
- Das Wort **Hypertext** steht für eine Menge von Texten, die durch Querverweise (wie z.B. Hyperlinks oder Links) miteinander verbunden sind.
- Mit **Markup Language** ist gemeint, dass der darzustellende Text nicht nur den Inhalt, sondern auch eine Beschreibung der Darstellung enthält.
- Markup Language kann man auf Deutsch mit "Auszeichnungssprache" übersetzen. ◆

HTML

- HTML wurde im Jahr 1990 vom Wissenschaftler Tim Berners-Lee am CERN in Genf entwickelt.
- Sinn und Zweck war eigentlich, dass Wissenschaftler verschiedener Universitäten so einfacher auf die Forschungsergebnisse anderer zugreifen können sollten.
- Das Projekt wurde ein grösserer Erfolg, als Tim Berners-Lee je gedacht hatte. Mit der Erfindung von HTML legte er den Grundstein für das Internet, wie wir es heute kennen. ◆


Auszeichnungssprachen

- Eine **Auszeichnungssprache** ist eine maschinenlesbare Sprache für die Gliederung und Formatierung von Texten und anderen Daten. Dazu müssen jeweils **Syntax** und **Semantik** angegeben werden.
- Unter **Syntax** versteht man allgemein ein Regelsystem zur Kombination elementarer Zeichen zu zusammengesetzten Zeichen in natürlichen oder künstlichen Zeichensystemen.
- **Semantik**, auch Bedeutungslehre, nennt man die Theorie oder Wissenschaft von der Bedeutung der Zeichen. Zeichen können in diesem Fall Wörter, Phrasen oder Symbole sein. Die Semantik beschäftigt sich typischerweise mit den Beziehungen zwischen den Zeichen und deren Bedeutungen. ◆

Syntax vs. Semantik

- Der Satz „**Nachts ist es kälter als draussen.**“ ist grammatikalisch vollkommen korrekt.
- Das heisst, die Grammatik der deutschen Sprache wurde fehlerfrei umgesetzt – das entspricht einer fehlerfreien Syntax.
- Allerdings ist dieser Satz ziemlich sinnlos. Er hat keine Bedeutung. ◆

Auszeichnungssprachen: Beispiele

- HTML (Hypertext Markup Language)
- XML (Extended Markup Language)
- LaTeX (Textsatzsystem)
- Wikitext
- ... 

Website vs. Webseite

- "Website" und "Webseite" klingen sehr ähnlich, bedeuten aber etwas völlig verschiedenes:
Während mit Webseite eine HTML-Datei gemeint ist, die eine Seite im Browser darstellt, ist eine Website (englisch "site" = Standort, Lage, Platz) eine Sammlung von meist mehreren Webseiten.
- Eine Website ist somit die Präsenz einer Firma im World Wide Web (www), eine Webseite ist eine Datei, in der eine Webseite als HTML-Code gespeichert ist. ◆

Homepage

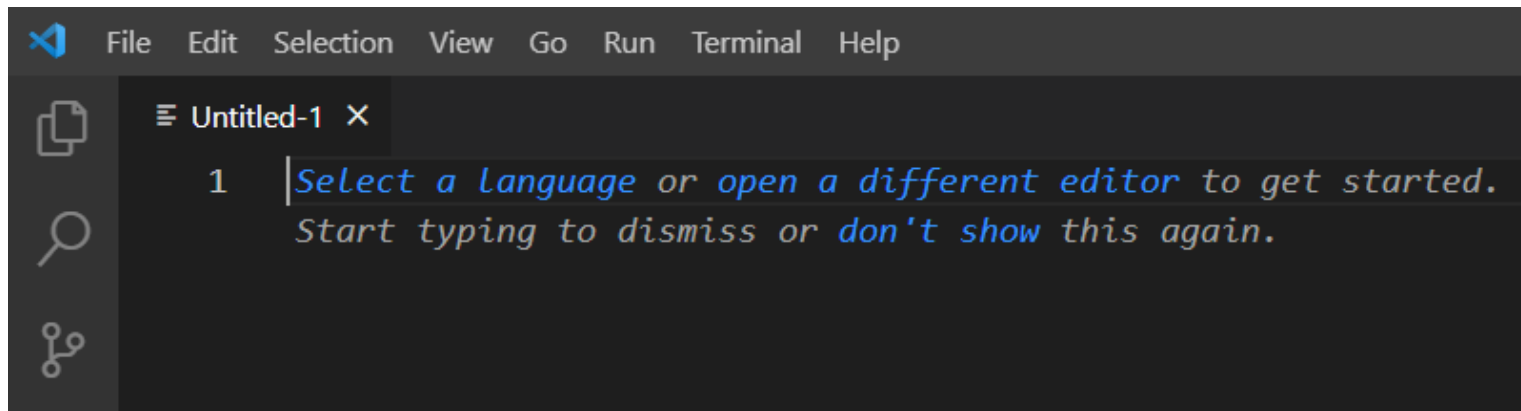
- Der Begriff "Homepage" wird auch oft falsch benutzt.
- In seiner ursprünglichen Bedeutung war es die Webseite einer Website, bei der man gestartet ist.
- Wenn man sich verläuft, kann man mit einem Link oder Button wieder zur "home page" (auf Deutsch etwa "Startseite") zurückgehen. ◆

HTML-Editoren und andere Tools

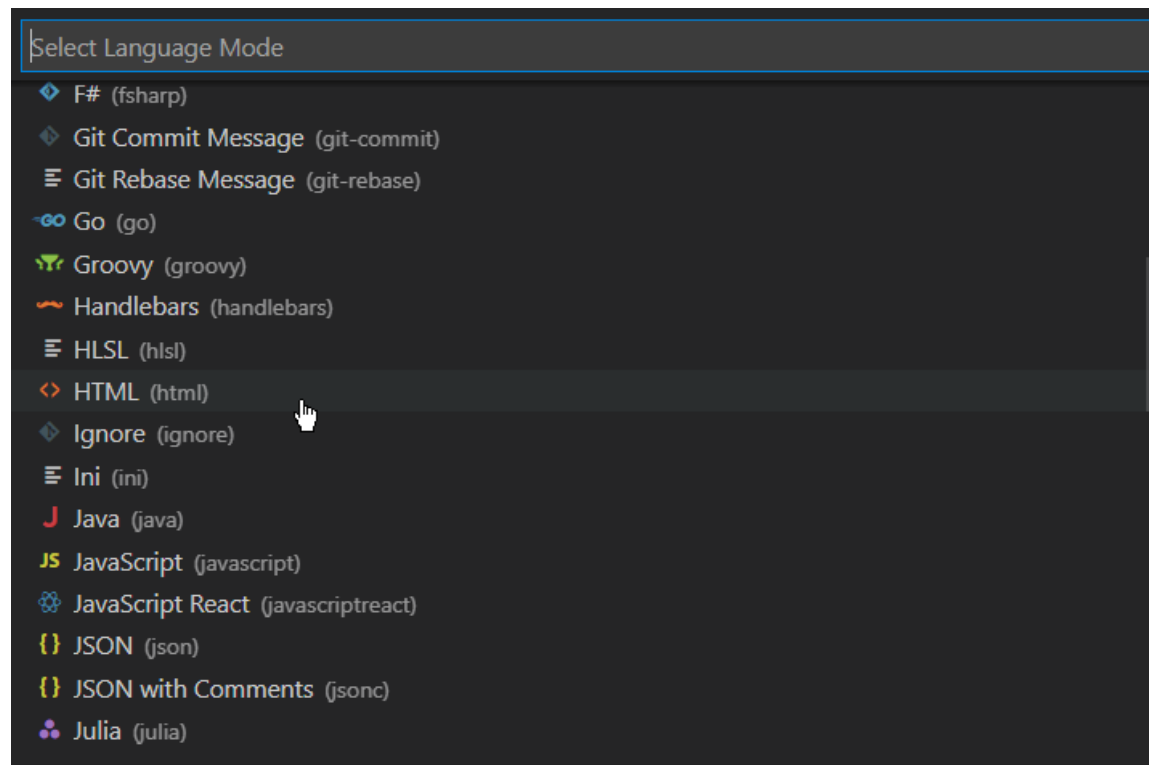
- Visual Studio Code
<https://code.visualstudio.com/download>
- Notepad++
<https://notepad-plus-plus.org/downloads>
- HTML Beautifier and Formatter
<https://www.cleancss.com/html-beautify>
- CSS Formatter
<https://www.cleancss.com/css-beautify> ◆

Eine HTML-Datei anlegen

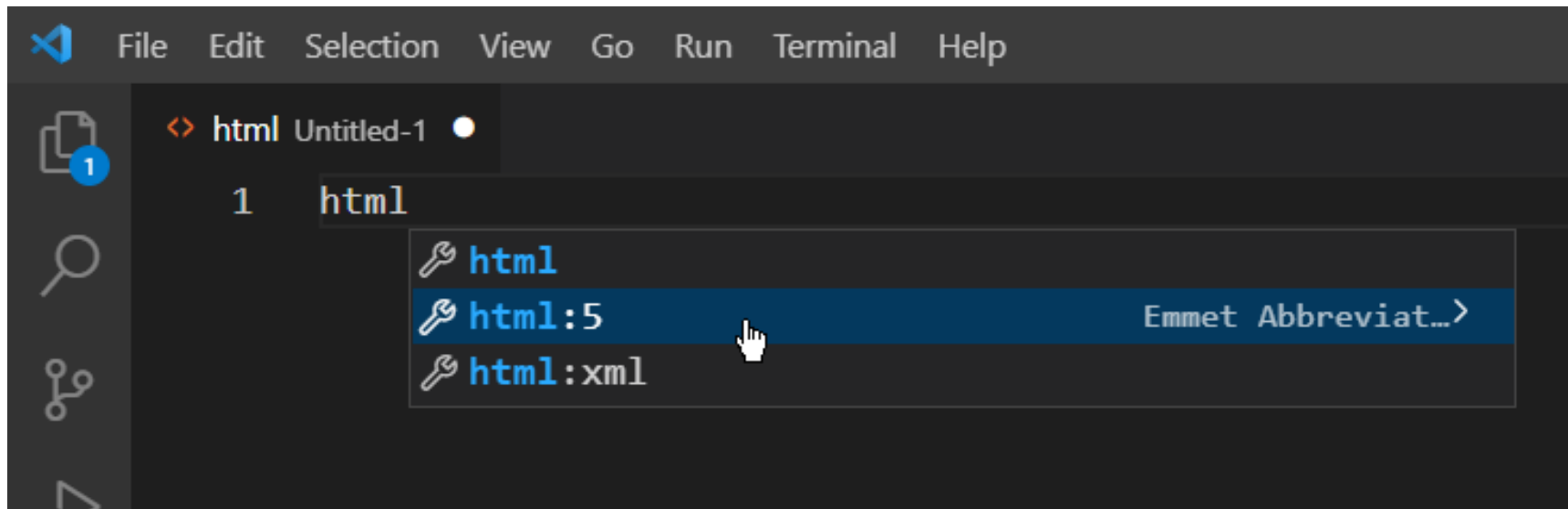
- Hier am Beispiel von Visual Studio Code.



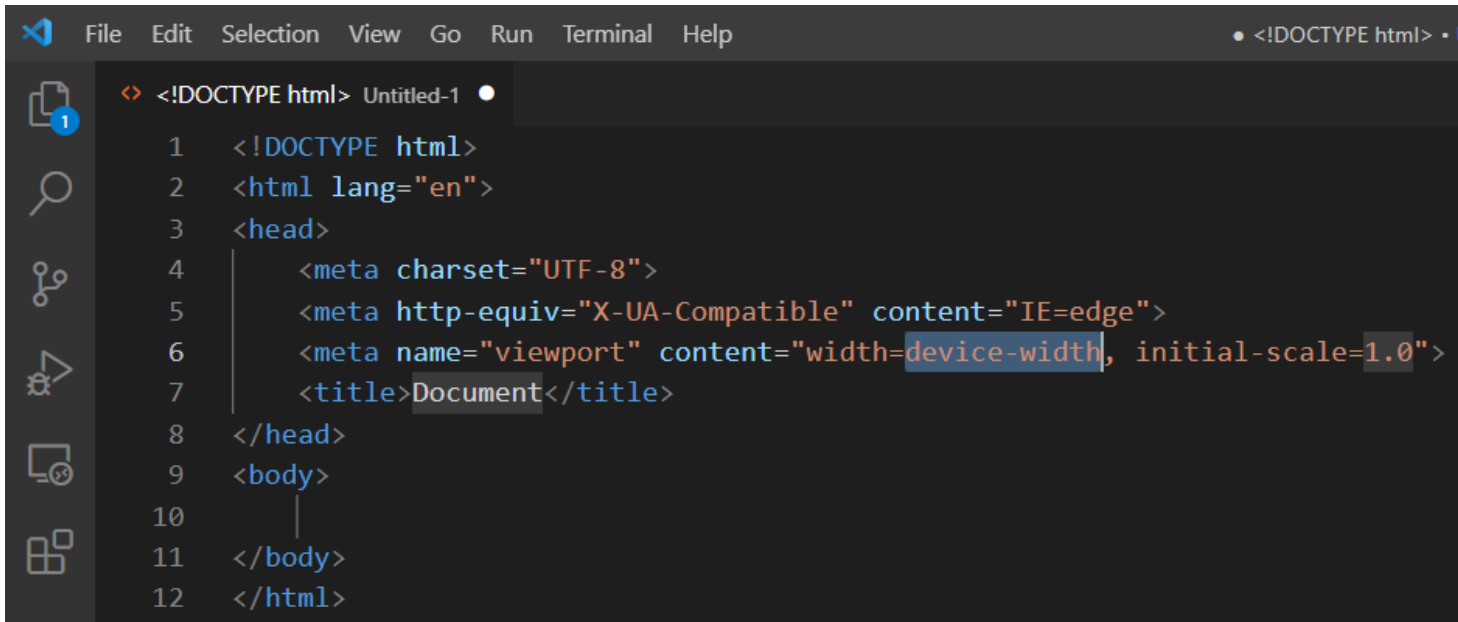
Eine HTML-Datei anlegen



Eine HTML-Datei anlegen



Eine HTML-Datei anlegen



```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
</head>
<body>
</body>
</html>
```

Die Elemente des HTML-Codes

- **HTML** besteht aus Elementen
- **Elemente** starten und enden (meist) mit Tags.
- Ein **Element** ist ein Teil des HTML-Codes, Elemente können ineinander verschachtelt werden.
- Ein **Tag** (auf Deutsch "Markierung") markiert den Beginn und das Ende eines Elements.
- Alle **Tags** haben das gleiche Format: Sie beginnen mit einem Kleiner-als-Zeichen "<" und enden mit einem Größer-als-Zeichen ">". ◆

Die Elemente des HTML-Codes

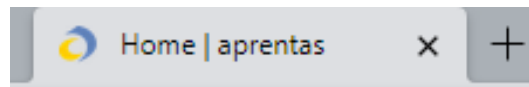
- Das End-Tag unterscheidet sich vom Start-Tag durch einen Schrägstrich: `<html>` und `</html>`. Der so zu markierende Inhalt wird zwischen das Start-Tag und das End-Tag geschrieben. ◆

Das Element "html"

- Das Element `<html>...</html>` kennzeichnet den Beginn und das Ende des in HTML codierten Inhaltes einer Datei.
- Eine HTML-Datei hat meistens die Endung ".html" oder ".htm".
- Innerhalb des html-Elements kommen die Elemente `<head>...</head>` und `<body>...</body>`. ◆

Das Element "head"

- Das Element `<head>...</head>` kennzeichnet den Kopf einer HTML-Seite.
- Hier werden Informationen über die HTML-Seite zentral abgelegt.
- Innerhalb des Kopfes wird zum Beispiel der Titel mit dem Element `<title>...</title>` angegeben. Der Titel ist das, was in der Lasche angezeigt wird:



Das Element "body"

- Das Element `<body>...</body>` kennzeichnet den eigentlichen Inhalt einer HTML-Seite. ◆

Die Elemente "b" und "i"

- Das Element `...` wird im Browser für Fettschrift verwendet. Das "b" steht dabei für das englische Wort "bold" (fett)...
- ... während das Element `<i>...</i>` im Browser für kursive Schrift verwendet wird. Das "i" steht dabei für das englische Wort "italic" (kursiv).
- Es ist auch möglich, beide Tags zu verschachteln: `<i>...</i>`; man beachte dabei aber die unterschiedliche Reihenfolge der End-Tags! (Tags sind wie Klammern in der Mathematik anzusehen.)
- Statt `...` wird heute eher `...` verwendet. ◆

Heading-Elemente

- Die Elemente `<h1>...</h1>`, `<h2>...</h2>`, `<h3>...</h3>` bis `<h6>...</h6>` werden zum Kennzeichnen von Überschriften verwendet.
- "h" steht dabei für "heading", englisch für Überschrift. ◆

Das Element "p"

- Das Element `<p>...</p>` stellt einen Absatz dar.
- Es ist vergleichbar mit einem Textteil in MS Word, der mit dem ¶-Zeichen endet.
◆

leere Elemente

- Bei einigen Elementen gibt es kein End-Tag.
- Diese so genannten **leeren Elemente** sind nicht mit einer speziellen Textpassage verbunden, sondern eher eigenständige Markierungen.
- Beispiele für leere Elemente sind:
 -
 für einen Zeilenumbruch
 - <hr> für eine horizontale Linie. ◆

unsortierte Liste

- Eine unsortierte Liste (Liste mit Aufzählungspunkten) sieht im HTML-Code so aus:

```
<ul>  
  <li>Ein Eintrag</li>  
  <li>Ein weiterer Eintrag</li>  
</ul>
```
- Dabei steht "ul" für "*unordered list*", also eine Liste ohne Reihenfolge, damit ist fehlende Aufzählung gemeint. ◆

sortierte Liste

- Eine sortierte Liste (Liste mit Aufzählungen) sieht im HTML-Code so aus:

```
<ol>  
  <li>Ein Eintrag</li>  
  <li>Ein weiterer Eintrag</li>  
</ol>
```

- Dabei steht „ol“ für "*ordered list*", also eine Liste mit Reihenfolge. ◆

Definitionsliste

- Das Element `<dl>...</dl>` wird folgendermassen eingesetzt:
`<dl>`
 `<dt>Term 1</dt>`
 `<dd>Definition of term 1</dd>`
 `<dt>Term 2</dt>`
 `<dd>Definition of term 2</dd>`
`</dl>`
- "dl" steht für "*definition list*", "dt" für "*definition term*" und "dd" für "*definition*".
- Es war ursprünglich gedacht für das Abbilden eines Glossars oder Definitionen, weil der Text zwischen `<dd>` und `</dd>` eingerückt gegenüber dem Text zwischen `<dt>` und `</dt>`. ◆

Schreibweise der Tags

- Es spielt keine Rolle, ob die Tags in Gross-, Klein- oder gar gemischten Buchstaben geschrieben werden.
- Sowohl `<HTML>`, als auch `<html>` oder `<HtMl>` werden normalerweise identisch angezeigt.
- Es hat sich jedoch die Schreibweise in Kleinbuchstaben durchgesetzt. ◆

(Hyper-)Links

- Ein (Hyper-)Link wird mit dem Element "a" (englisch "*anchor*") dargestellt:
``
Hier ist ein Link zu HTML.net
``
- Hier sehen wir zum ersten Mal ein Element mit einem **Attribut** ("href"). "href" steht für *hypertext reference* und gibt an, wohin der Link führen soll. Meist ist dies eine Internetseite oder der Dateiname zu einer anderen Webseite.
- Der Satz "Hier ist ein Link zu HTML.net" ist der Text, der vom Browser als Link dargestellt wird.
- Das Element endet mit einem ``. ◆

(Hyper-)Links in neuem Tab öffnen

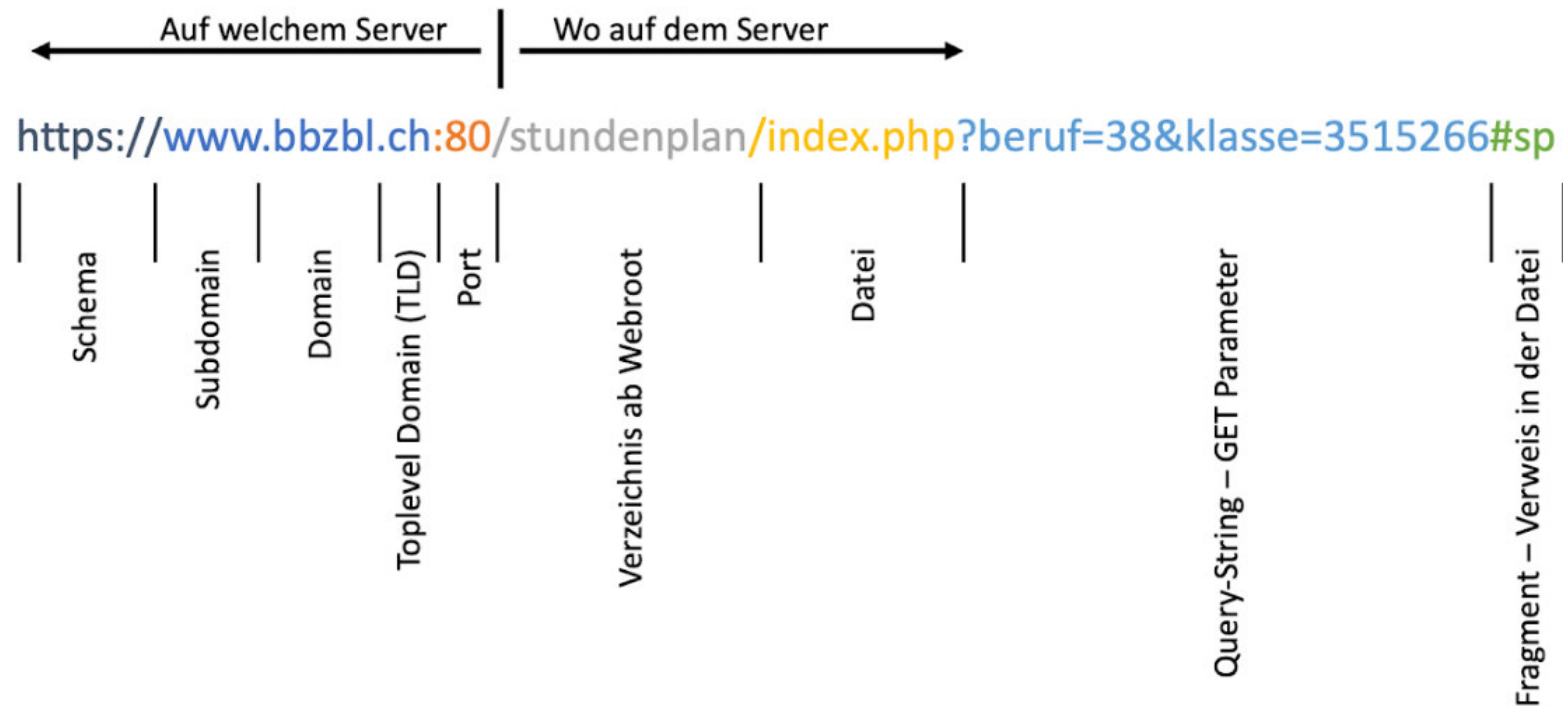
- Soll mit einem (Hyper-)Link die andere Seite oder ein Dokument in einem neuen Tab geöffnet werden, dann muss noch ein weiteres Attribut hinzugefügt werden:

```
<a href="https://www.html.net/" target="_blank">  
  Hier ist ein Link zu HTML.net  
</a> ◆
```

URL

- Im vorherigen Beispiel hat das Attribut "href" den Wert "https://www.html.net", welches die voll ausgeschriebene Adresse von **html.net** ist.
- Diese werden auch URL (Uniform Resource Locator) genannt.
- Bitte beachten Sie, dass "https://" immer in den URLs angegeben werden muss. ◆

Aufbau einer URL



lokale Links

Downloads ➤ webseite

Name

 `seite-1.html`

 `seite-2.html`

- Wenn Sie zwischen den Seiten der gleichen Webseite einen Link setzen möchten, müssen Sie nicht die ganze Adresse (URL) ausschreiben.
- Wenn Sie z.B. zwei Seiten ("seite-1.htm" und "seite-2.htm") erstellt haben und diese im selben Ordner gespeichert sind, dann ist im Link nur die Angabe des Dateinamens nötig.
- Unter diesen Umständen sähe ein Link dann so aus:
`Zur Seite 2` ◆

lokale Links

Downloads > webseite

Name

 ordner

 seite-1.html

- Wenn "seite-2.htm" in einen Unterordner mit dem Namen "ordner" liegen würde, könnte der Link von seite-1 nach seite-2 so aussehen:
`Zur Seite 2`
- Der Link von seite-2 nach seite-1 sähe so aus:
`Zur Seite 1`
- Das "../" zeigt auf den Ordner eine Ebene über der Position der Datei in der der Link eingefügt wurde.
- Mit dem gleichen System können Sie auch zwei oder noch mehr Ebenen überspringen. Für zwei Ebenen würde dies so aussehen: "../..". ♦

Sprungmarken

- Diese sehen so aus:

[Link zur Überschrift 1](#)

[Link zur Überschrift 2](#)

Überschrift 1

Text text text text

Überschrift 2

Text text text text

Sprungmarken

- Diese Sprungmarken werden folgendermassen realisiert:

```
<html>
  <head>
  </head>
  <body>
    <p><a href="#heading1">Link zur Überschrift 1</a></p>
    <p><a href="#heading2">Link zur Überschrift 2</a></p>
    <h1 id="heading1">Überschrift 1</h1>
    <p>Text text text text</p>
    <h1 id="heading2">Überschrift 2</h1>
    <p>Text text text text</p>
  </body>
</html> ◆
```

Link auf eine E-Mail-Adresse

- Ein Hyperlink auf eine E-Mail-Adresse sieht so aus:
- `niemand@html.net`
- Klickt man ihn an, so öffnet sich meistens das installierte E-Mail-Programm, es wird ein leeres E-Mail angezeigt, bei dem der Empfänger jedoch schon eingetragen ist.
- Man kann den Hyperlink aber auch so erweitern, dass das Subject-Feld (die Betreffzeile) auch schon ausgefüllt ist:
- `niemand@html.net` ◆

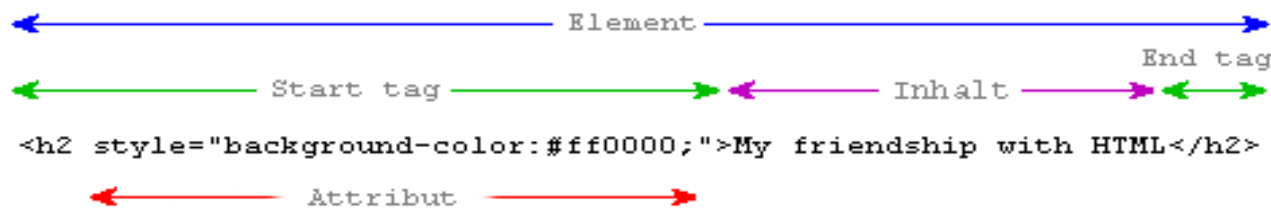
Link mit Tooltip

- Um einen Link zu erstellen, müssen Sie immer das href-Attribut verwenden. Zusätzlich dazu können Sie dem Link mit dem Attribut title noch einen Titel geben:
- `HTML.net`
- Das sieht im Browser so aus: 
- Das Attribut "title" wird dazu benutzt, eine kurze Beschreibung des Links zu geben. Wenn Sie den Mauszeiger – ohne zu klicken – auf dem Link platzieren, erscheint der Text "HTML.net besuchen und HTML lernen". ♦

Attribute

- **Elemente** verleihen dem HTML-Code eine Struktur und geben dem Browser vor, wie Sie Ihre Webseite angezeigt bekommen möchten.
- Einigen Elementen kann man aber noch weitere Informationen zuweisen.
- Eine solche Information nennt man **Attribut**.
`<h2 style="background-color:#ff0000;">`
 Meine Freundschaft mit HTML
`</h2>`. ◆

Attribute



- Attribute werden **immer in das Start-Tag** des Elementes geschrieben und werden von einem Gleichheitszeichen gefolgt.
- Die Attributwerte werden in Anführungszeichen ("..." oder '...') geschrieben.
- Das Semikolon nach dem Attribut im Beispiel dient der Trennung verschiedener Style-Angaben. ◆

Farben

- Im obigen Beispiel haben wir die Hintergrundfarbe mit dem Code `#ff0000` angegeben; dies ist der Farbcode für die Farbe rot in der sogenannten Hexadezimalschreibweise. Jede Farbe hat ihren eigenen hexadezimalen Farbwert.
- Hier einige Beispiele:

weiss:	<code>#ffffff</code>	blau:	<code>#0000ff</code>
schwarz:	<code>#000000</code>	grün:	<code>#00ff00</code>
rot:	<code>#ff0000</code>	gelb:	<code>#ffff00</code>
- Ein hexadezimaler Farbcode besteht aus der Raute "#" und einer sechsstelligen hexadezimalen Zahl. ◆

Farben

- Für die gebräuchlichsten Farben können Sie auch die englischen Farbnamen verwenden: `white`, `black`, `red`, `blue`, `green` und `yellow`.
- Das Beispiel von oben hätte man also auch so schreiben können:

```
<h2 style="background-color:red;">  
  Meine Freundschaft mit HTML  
</h2>
```

 ◆

Bilder

- Sie müssen dem Browser zuerst mitteilen, dass Sie ein Bild ("img", kurz für "image") einfügen möchten. Dann müssen Sie ihm noch die Quelle ("src", kurz für "source") nennen.
- Bitte beachten Sie, dass das img-Element kein End-Tag hat.
- `` ◆

Bilder

- Bilder können sich auch in anderen Ordnern der gleichen Website oder in anderen Websites befinden:

```

```

```

```

- Bilder können auch Links sein:

```
<a href="https://www.html.net"></a> ◆
```

Bilder

- Das Attribut "alt" gibt eine alternative Beschreibung des Bildes an. Diese Beschreibung wird angezeigt, wenn – aus welchen Gründen auch immer – das Bild nicht angezeigt wird. Besonders wichtig ist "alt" z.B. für sehbehinderte oder blinde Internetnutzer.
``
- Einige der älteren Browser lassen den Inhalt des alt-Attributes in einem kleinen sog. Tooltip (ein kleines Fenster neben dem Mauszeiger) erscheinen, wenn man den Mauszeiger über das Bild bewegt.
- Bitte beachten Sie, dass eine Beschreibung des Inhaltes die Aufgabe dieses Attributes ist und nicht Informationen zum Bild zu geben, da sehbehinderte Menschen diesen Text hören, ohne zu wissen, dass es sich um ein Bild handelt. ◆

Bilder

- Das Attribut "title" hingegen kann dazu benutzt werden, Informationen zum Bild zu geben:
``
- Zwei weitere wichtige Attribute sind "width" und "height":
``
- Die wirkliche Dateigrösse des Bildes bleibt jedoch erhalten!
- **Achtung:**
Eine solche Verkleinerung hat keinen Einfluss auf die Ladegeschwindigkeit Ihrer Webseite, auch wenn das Bild kleiner aussieht. Deswegen sollten Sie nie versuchen, die Grösse Ihres Bildes mit "width" und "height" zu verkleinern. Dies sollte eher mit einem Bildbearbeitungsprogramm geschehen. ◆

Tabellen

- Das Grundgerüst einer HTML-Tabelle sieht folgendermassen aus:

```
<table>
  <tr>
    <td>Zeile 1, Spalte 1</td>
    <td>Zeile 1, Spalte 2</td>
  </tr>
  <tr>
    <td>Zeile 2, Spalte 1</td>
    <td>Zeile 2, Spalte 2</td>
  </tr>
</table>
```

- Dabei steht "tr" für "table row", also Tabellenzeile, während "td" für "table data" steht. ◆

Tabellen

- Theoretisch können Sie alles in Tabellen einfügen: Text, Links und Bilder.
- Tabellen sind jedoch dafür gedacht "Tabellendaten" anzuzeigen (also Informationen, welche in Zeilen und Spalten präsentiert werden sollten).
- Früher hat man Tabellen dazu missbraucht, verschiedene Inhalte entsprechend zu platzieren; in HTML 5 gibt es dafür ganz andere Möglichkeiten. ◆

Tabellen-Attribute

- `<table border="1">`
- Wird verwendet, um die Dicke der Rahmen (Begrenzungslinien) der Tabelle anzugeben; die Dicke der Begrenzungslinien wird in Pixeln angegeben.
- `<table width="30%">`
- Die Breite der Tabelle wird entweder in Pixeln oder in Prozent (der Gesamtbreite) angegeben. ◆

Tabellen-Attribute

- `<td align="right" valign="top">Zelle 1</td>`
- Das Attribut "align" gibt die horizontale Ausrichtung (englisch "alignment") des Inhaltes in der Tabelle, einer Tabellenspalte oder einer einzelnen Zelle an.
- Werte für dieses Attribut sind "left" (links), "center" (zentriert) oder "right" (rechts).
- Das Attribut "valign" gibt die vertikale Ausrichtung ("vertical alignment") des Inhaltes in einer Tabellenzelle.
- Werte für dieses Attribut sind z.B. "top" (oben), "middle" (in der Mitte) oder "bottom" (unten). ◆

Tabellen-Attribute

- ```
<table border="1">
 <tr>
 <td colspan="3">Zelle 1</td>
 </tr>
 <tr>
 <td>Zelle 2</td>
 <td>Zelle 3</td>
 <td>Zelle 4</td>
 </tr>
</table>
```
- Die beiden Attribute "colspan" und "rowspan" werden dazu verwendet, um besondere Tabellen zu erstellen.
- "colspan" bedeutet "column span", welches man mit "Spalten-Spannweite" übersetzen kann.
- Beide Attribute werden im <td>-Tag benutzt und geben an, über wie viele Spalten die Zelle aufgespannt (zusammengefasst) werden soll. ◆

# Kommentare

- Um den eigenen HTML-Code mit Anmerkungen zu bereichern, die auf das Aussehen der Webseite keinen Einfluss haben sollen, muss man Kommentare verwenden.
- Diese werden folgendermassen erstellt:
- `<!-- hier kommt der Kommentar -->` ◆

# Formulare

- Zu Beginn war HTML lediglich als "Einbahnstrasse" konzipiert: Es sollten Information von einem Webserver auf einen Webclient (Browser) transportiert werden.
- Lediglich der Aufruf einer anderen Webseite war Information, die vom Webclient in Richtung des Webserver ging.
- Recht früh hat man HTML dann um Formulare erweitert, um auch andere Daten in diese Richtung zu übertragen. ♦

# Formulare

First name:

Last name:

- Solche Eingabefelder (einschliesslich Labels) werden folgendermassen realisiert:
- ```
<form>  
  <label for="fname">First name:</label><br>  
  <input type="text" id="fname" name="fname"><br>  
  <label for="lname">Last name:</label><br>  
  <input type="text" id="lname" name="lname">  
</form>
```
- ◆

Formulare

- Es gibt folgende *input types* (siehe auch https://www.w3schools.com/html/html_form_input_types.asp):
 - **text** zeigt ein einzeliges Eingabefeld an
 - **radio** zeigt *Radiobuttons* an (zum Selektieren einer Option aus vielen)
 - **checkbox** zeigt *Checkboxes* an (zum Selektieren keiner, mehrerer oder alle Optionen aus vielen)
 - **submit** zeigt einen submit button an, mit dem man die Information an den Server überträgt
 - **button** Displays a clickable button ◆

Formulare



- Für mehrzeilige Textfelder steht `textarea` zur Verfügung:
- `<form>`
 `<textarea id="..." name="..."`
 `rows="4" cols="50">...</textarea>`
`</form>` ◆

Formulare

- Dem Formular muss über `action` noch mitgeteilt werden, was mit diesen Daten, die man verschickt, geschehen soll.
- Und damit man es abschicken kann, wird ein *Submit Button* benötigt.
- ```
<form action="danke.html">
 <label for="fname">First name:</label>

 <input type="text" id="fname" name="fname" value="John">

 <label for="lname">Last name:</label>

 <input type="text" id="lname" name="lname"
value="Doe">

 <input type="submit" value="senden">
</form>
```

 ◆

# Standards und Validierung

- Der heutige HTML-Standard heisst HTML5; dies wird ausgedrückt durch `<!DOCTYPE html>` in der ersten Zeile.
- In HTML-Seiten bzw. dessen Code von älteren Standards findet man Zeilen wie:  
`<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"  
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">`
- Es gibt Webseiten, wo man seinen HTML-Code prüfen lassen kann:  
<https://validator.w3.org> ◆

# guter HTML-Stil

- Webseiten werden von modernen Browsern deutlich schneller und sauberer aufgebaut, wenn bei jeder referenzierten Grafik die genauen Angaben zu Breite und Höhe der Grafik notieren.
- Ausserdem lassen die Browser entsprechend grosse Platzhalter frei, wenn die Grafik gar nicht oder nicht vollständig übertragen wird, und die übrige Seite wird dadurch nicht "zerhauen".
- Bei Grafiken sollten Sie sich auch angewöhnen, Alternativtexte zu notieren. Es gibt zwar keine nennenswerten nicht-grafikfähige Browser mehr, doch darf man die blinden Websurfer nicht vergessen. ◆

# Favicons

- Ebenfalls guter Stil ist heute das Favicon.
- Das ist ein kleines Symbol – meistens das Logo der Firma – auf dem Tab der geöffneten Webseite.
- Das Favicon wird folgendermassen eingebunden:  
`<link rel="icon" href= "../pictures/favicon.ico" />` ◆

