

(X)HTML:

<...> beginnender HTML Tag
</...> schließender HTML Tag
<... /> leere HTML Elemente
<tag attribut ="wert"> Attribute und Werte
&....; Sonderzeichen, HTML Entities

Aufbau einer Datei

Ein HTML-Dokument besteht aus drei Bereichen:

1. der Dokumenttypdeklaration ganz am Anfang der Datei, die die verwendete DTD angibt, z. B. HTML 4.01 Strict,
2. dem HTML-Kopf (HEAD), der hauptsächlich technische oder dokumentarische Informationen enthält, die nicht direkt im Browser sichtbar sind und
3. dem HTML-Körper (BODY), der anzuzeigende Informationen enthält.

Überprüfung der einzelnen Seiten:

<http://www.w3.org/>

HTML4.01 Transitional

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
<html>
  <head>
    <title>Titel der Webseite</title>
    <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
    <meta name="keywords" content="Wifi, Wien, max. 1000 Zeichen, Schlüsselwörter durch Komma getrennt ">
    <meta name="description" content="Dies ist der Beschreibungstext max. 200 Zeichen">
  </head>
  <body>
    Inhalt der Webseite
  </body>
</html>
```

HTML4.01 Strict

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
<html>
  <head>
    <title>Titel der Webseite</title>
    <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
    <meta name="keywords" content="Wifi, Wien, max. 1000 Zeichen, Schlüsselwörter durch Komma getrennt ">
    <meta name="description" content="Dies ist der Beschreibungstext max. 200 Zeichen">
  </head>
  <body>
    Inhalt der Webseite
  </body>
</html>
```

XHTML 1.0 Transitional

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    <title>Titel der Webseite</title>
    <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
    <meta name="keywords" content="Wifi, Wien, max. 1000 Zeichen, Schlüsselwörter durch Komma getrennt " />
    <meta name="description" content="Dies ist der Beschreibungstext max. 200 Zeichen" />
  </head>
  <body>
    Inhalt der Webseite
  </body>
</html>
```

XHTML 1.0 Strict

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    <title>Titel der Webseite</title>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
    <meta name="keywords" content="Wifi, Wien, max. 1000 Zeichen, Schlüsselwörter durch Komma getrennt " />
    <meta name="description" content="Dies ist der Beschreibungstext max. 200 Zeichen" />
  </head>
  <body>
    Inhalt der Webseite
  </body>
</html>
```

Blockelemente

Nach den Vorgaben des W3C muss zumindest in den Strict-Varianten von (X)HTML jeglicher Inhalt - sei es einfach nur Text oder seinen es Inline-Elemente - innerhalb so genannter Blockelemente stehen. Blockelemente definieren Bereiche, die an ihrem Ende automatisch einen Zeilenumbruch erzielen und so eine Art Block erzeugen.

Bei der Verschachtelung spricht man auch von Eltern- und Kindelementen. Eltern sind immer die äußeren Elemente, die darin befindlichen sind die jeweiligen Kind-Elemente. Allen Elementen können Eigenschaften zugewiesen werden.

DIV (<div>...</div>)

Im Grunde werden DIV-Elemente selbst erst dann sichtbar, wenn sie via CSS anders positioniert oder mit Rahmen oder anderen Hintergrundfarben versehen werden.

```
<div>
  Das ist ein DIV-Element.
</div>
```

P (<p>...</p>) Absatz

Das P-Element wird auch als Absatz bezeichnet. Im Gegensatz zum DIV-Element erzeugt das P-Element zum vorhergehenden einen zusätzlichen Abstand.

```
<p>
  Das ist ein P-Element
</p>
```

H1 – H6 (<h1>...</h1>) Überschriften

Es stehen für Überschriften unterschiedlicher Größe und Wichtigkeit gleich sechs verschiedene Tags zur Verfügung: H1, H2, H3, H4, H5 und H6. Die Ziffer hinter dem H gibt die Ordnung der Überschrift an. H1 wäre also eine Überschrift erster Ordnung, also die oberste, H6 demnach die kleinste.

```
<h1>Eine ganz wichtige Überschrift</h1>
<h2>Diese ist schon nicht mehr ganz so wichtig</h2>
<h3>Und diese noch weniger</h3>
<h4>So langsam werden die Überschriften immer kleiner</h4>
<h5>Und noch kleiner</h5>
<h6>Bis zur kleinsten</h6>
```

UL und LI (....) ungeordnete Listen

Das Element *ul* zeichnet eine ungeordnete Liste aus, das heißt, deren Listenelemente *li* unterliegen keiner bestimmten Ordnung, sind also gleichwertig.

```
<ul>
  <li>HTML</li>
  <li>CSS</li>
  <li>Javascript</li>
  <li>PHP</li>
  <li>MYSQL</li>
</ul>
```

OL und LI (`<ol>`
 `<li>....</li>`
 `</ol>`)

geordnete Listen

Das *Element* `ol` zeichnet eine geordnete Liste aus, deren Listenelemente `li` einer bestimmten Reihenfolge unterliegen.

```
<ol>
  <li>HTML</li>
  <li>CSS</li>
  <li>Javascript</li>
  <li>PHP</li>
  <li>MYSQL</li>
</ol>
```

Verschachtelung

Untergeordnete Listen müssen korrekt **innerhalb der einzelnen Listenelemente** aufgeführt werden, wie folgendes Beispiel zeigt.

```
<ol>
  <li>HTML
    <ul>
      <li>Blockelement</li>
      <li>Inline-Elemente</li>
    </ul>
  </li>
  <li>CSS</li>
</ol>
```

Inline Elemente:

Inline Elemente können quasi überall stehen, nur nicht direkt innerhalb des BODY-Tags und des FORM-Tags. Um dort auftreten zu können, müssen sie mit einem Block-Element umschlossen werden. Inline-Elemente dürfen fast beliebig ineinander verschachtelt werden. Jedoch darf innerhalb eines Inline-Elementes nie ein Blockelement auftreten. Einige Inline-Elemente dienen zur Formatierung von Text, andere übernehmen echte Aufgaben.

SPAN (`<span>...</span>`) Formatierung

Das span-Element hat ohne CSS absolut keine Funktion. Er umschließt Text, dem via CSS eine bestimmte zusätzliche oder geänderte Formatierung zugewiesen werden soll, um ihn in irgendeiner Form hervorzuheben.

`<span>`Dies ist ein Text welcher über CSS formatiert werden kann`</span>`

EM (`<em>...</em>`)

Das Element em kennzeichnet Text als betont. (Kursiv)

`<em>`Der Text wird betont`</em>`

STRONG (`<strong>...</strong>`)

Das Element strong kennzeichnet Text als stark betont. (Fett)

`<strong>`Der Text wird stark betont`</strong>`

A (`<a ...>...</a>`) Anker

Das Element `a` definiert

- einen Zielanker für beliebig viele Links, wenn ein Attribut `name` definiert wird (`<a name="bezeichner">Ankertext</a>`), oder
- einen Quellanker von genau einem Link, wenn es ein Attribut `href` enthält (`<a href="uri">Ankertext</a>`). Der Wert des `href`-Attributs ist ein URI.

Der Einfachheit halber wird ein Quellanker im Allgemeinen auch als Link bezeichnet.

Ein Link auf die Website vom Institut WIFI Wien:

```
<a href="http://www.wifiwien.at">Wifi Wien</a>
```

Um ein Dokument zum Download anzubieten, setzen Sie folglich einen einfachen Link.

```
<a href="http://www.wifiwien.at/file.zip">Download</a>
```

PFADANGABE

Absoluter Pfad:

`http://www.wifiwien.at`
`http://www.wifiwien.at/test/`
`http://www.wifiwien.at/test/index.html`
`http://www.wifiwien.at/bilder/grafik.gif`

Relativer Pfad:

`bilder/grafik.gif`
`html/test.html`

`../bilder/grafik.gif`
`../html/test.html`

`../../bilder/grafik.gif`
`../../html/test.html`

`/bilder/grafik.gif`
`/html/test.html`

IMG ()

Das Element img bettet ein Bild an der Stelle in das aktuelle Dokument ein.

img-Elemente verfügen üblicherweise über die Attribute **src**, **width**, **height**, **alt** und **title**.

Das Attribut **src** gibt den Ort der Bildquelle über einen URI an.

Die Attribute **width** und **height** in Pixel oder Prozent weisen Benutzerprogramme an, die Originalbildgröße zu überschreiben; das Bild wird entsprechend skaliert.

Das Attribut **alt** gibt alternativen Text für Benutzerprogramme an, die Bilder nicht darstellen können, beispielsweise Screenreader oder Textbrowser, aber auch Suchmaschinen.

Das alt-Attribut muss für jedes img-Element angegeben werden !!!!

Das Attribut **title** bewirkt einen Tooltipp. (wie bei den andern Elementen)

```

```

```

```