



CODERS.BAY


HTML



CODERS.BAY

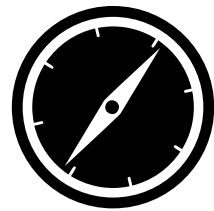
AUFBAU EINER SEITE UND DIE WICHTIGSTEN ELEMENTE

- drei wichtige Elemente für Webseiten – drei Sprachen
 - ▶ HTML
 - beschreibt die Struktur und den Inhalt der Seite
 - ▶ CSS
 - Cascading Style Sheets
 - sorgt dafür, dass alles gut aussieht bzw. so aussieht, wie man es haben möchte
 - ▶ JavaScript
 - ist für alles da, was etwas tun soll
 - z.B.: Rechnen, sich bewegen, ...

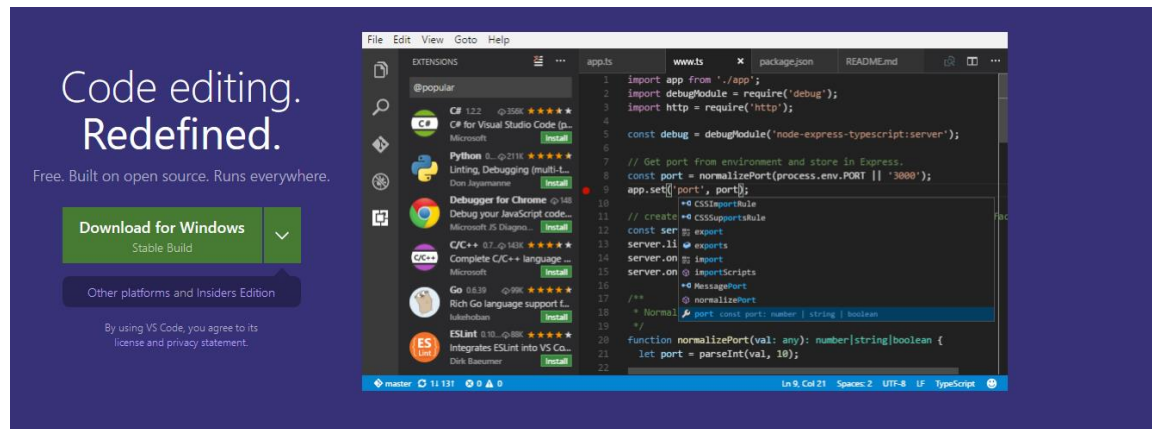
A yellow bookmark icon with a white outline, pointing downwards.

Der Browser ist das Programm, mit dem man Webseiten anschaut und im World Wide Web surft. Die bekanntesten sind Internet Explorer, Edge, Firefox, Safari und Chrome

- Webseiten sehen nicht in jedem Browser gleich aus
 - ▶ Safari
 - ▶ Chrome
 - ▶ Edge
 - ▶ Firefox



- ein guter Editor für HTML zeigt an, wenn das Dokument korrekt ist, hebt die Struktur hervor und erleichtert die Fehlersuche
 - ▶ Notepad++
notepad-plus-plus.org
 - ▶ Atom
atom.io
 - ▶ Brackets
brackets.io
 - ▶ **Visual Studio Code**
<https://code.visualstudio.com/>



WYSIGWYG – "what you see is what you get", also "was du siehst ist auch das, was rauskommt", und beschreibt alle Programme, bei denen das Dokument beim Bearbeiten genauso aussieht wie später für den Leser

- › alles was zwischen < und > steht, ist ein **Tag**
- › sind **geöffnete** Tags
 - <html>, <body>, <p>
- › mit / ist es ein **schließender** Tag
 - </html>, </body>, </p>
- › Alles zwischen einem öffnenden und dem schließenden Tag ist der **Tag-Body**, der Inhalt des Tags
- › Tags können und müssen ineinander verschachtelt werden
- › Tags dürfen beliebig tief ineinander verschachtelt werden, aber nicht jedes Tag darf an jeder Stelle stehen
- › Wenn das öffnende Tag innerhalb eines anderen Tags verschachtelt ist, dann **muss** das passende schließende Tag in dem gleichen Tag verschachtelt sein
- › Falsch verschachtelte Tags verursachen im Browser keine Fehlermeldung, aber die Seite kann falsch dargestellt werden
- › Mit CSS und JavaScript könnten falsche Verschachtelungen zu Fehlern führen, die nur schwer zu finden sind
- › Ein HTML-Dokument, in dem alle Tags korrekt geschlossen und verschachtelt sind, heißt **wohlgeformt** (well-formed)


```
<title>Meine erste Webseite</title>
```

öffnendes Tag Tag-Body schließendes Tag

Tag ist ein englisches Wort. Bedeutet "Markierung". Ein Tag markiert Text. HTML = Hyper Text Markup Language. Ist eine Markupsprache.

- Erstelle eine neue Datei und speichere sie mit der Endung .html
- Den HTML-Code einzurücken, wie gezeigt, ist für die Funktion der Seite nicht wichtig. Man könnte alles in eine Zeile schreiben und es sieht im Browser gleich aus
- Einrückungen machen es angenehmer die Datei zu bearbeiten

```
<!DOCTYPE html>
<html>
  <!-- Das ist wirklich meine erste Webseite -->
  <head>
    <title>Meine erste Webseite</title>
  </head>
  <body>
    <p>Hallo Webwelt!</p>
  </body>
</html>
```



Die meisten Editoren erkennen an der Dateiendung, dass es sich zum Beispiel um eine HTML-Datei handelt. Deshalb ist es besser, die Datei mit der richtigen Endung zu speichern, bevor man anfängt zu arbeiten.

STRUKTUR EINER SEITE

Ist kein echtes Tag, sondern die Doctype-Deklaration. Damit wird dem Browser mitgeteilt, dass jetzt wirklich ein HTML-Dokument beginnt – falls er der Dateiendung nicht vertraut.

`<!DOCTYPE html>`

Alles zwischen `<!-- -->` ist ein Kommentar und wird vom Browser ignoriert

`<html>`

`<!-- Das ist wirklich meine erste Webseite -->`

`<html>` ist ein Root-Tag. Muss bei jedem HTML-Dokument vorhanden sein. Davor darf nur der Doctype stehen

`<head>`

`<title>Meine erste Webseite</title>`

Hier gehört alles rein, was zwar mit der Seite zu tun hat, aber nicht zum Inhalt gehört.

`</head>`

`<body>`

Im `<body>` steht der wirkliche Inhalt der Seite, alles, was hier steht, wird im Browserfenster angezeigt

Seitentitel, wird nicht in der Seite angezeigt, sondern zum Beispiel in der Titelzeile (Tab) des Browserfensters

`<p>Hallo Webwelt!</p>`

`</body>`

`<p>` steht für Paragraph, also Textabsatz. Man muss Text nicht in `<p>`-Tags einpacken. Aber wenn man Fließtext schreibt, bietet es sich an.

Das Ende des HTML-Dokuments. Nach `</html>` darf nichts mehr kommen.

`</html>`


```
<!DOCTYPE html>
```

- kein HTML-Tag im strengen Sinne, sondern die Information für die Browser, um welche HTML-Version es sich bei der aktuellen Webseite handelt.
- !DOCTYPE ist kein HTML-Tag. Darum gibt es auch kein schließendes !DOCTYPE-Tag.
- !DOCTYPE ist das einzige, was vor dem <HTML>-Tag stehen kann.
- Das Ausrufezeichen "!" gehört dazu.
- DOCTYPE ist nicht case-sensitiv, darf also auch als Doctype oder doctype geschrieben werden.

➤ Doctype für ePub

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN" "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
```

➤ Doctype für SVG

```
<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.1//EN" "http://www.w3.org/Graphics/SVG/1.1/DTD/svg11.dtd">
```

➤ HTML 4.01 – Strict

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
```

➤ XHTML 1.0 – Strict

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
```

➤ XHTML 1.1 - DTD

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN" "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
```



CODERS.BAY

HEAD

<TITLE> & <LINK>

```
<!DOCTYPE html>
<html lang="de">
  <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,user-scalable=no" />
    <meta name="csrf-token" content="">
    <meta name="description" content="" />
    <meta name="author" content="" />

    <title> Irgendein Titel </title>

    <!-- Bootstrap core CSS -->
    <link href="vendor/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  </head>
</body>
```

Seitentitel, erscheint in der oberen Browserleiste

erzeugt einen Link zwischen dem aktuellen und einem verwandten Dokument, z.B. für eine CSS-Datei

```
<!DOCTYPE html>
<html lang="de">
  <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,user-scalable=no" />
    <meta name="csrf-token" content="">
    <meta name="description" content="" />
    <meta name="author" content="" />

    [...]

  </head>
<body>
```

- Meta-Tags enthalten Informationen über die Webseite, die im Browser nicht angezeigt werden, sondern sich eher an die Browser selbst, an den Server, die Suchmaschinen und spezielle Anwendungen richten.
- Sie sitzen meist im Kopf der Seite und als maschinenlesbare Informationen von der Kurzbeschreibung ([meta description](#)) über die [robots](#)-Anweisung bis zum Viewport der mobilen Geräte.

- Dokument muss in einem Unicode Encoding gespeichert sein und Browser muss das auch wissen
- Einstellung für Notepad++
 - ▶ Menü "Encoding", hier wählst du "Encode in UTF-8" aus
- Bei falscher Encoding werden im schlimmsten Fall nur wilde Sonderzeichen angezeigt.
- UTF-8 ist das verbreitetste der Unicode-Encodings

```
<meta charset="utf-8" />
```

- Character Entities fangen mit einem & an und hören mit einem ; auf
 - ▶ `©` = ©
 - ▶ <, >, & müssen IMMER als `<`, `>` und `&` codiert werden, sonst interpretiert sie der Browser als Teil eines Tags oder einer Entity

Ein Character Encoding, sie zu knechten,
alle Zeichen zu finden, zusammenzutreiben
und an Zahlen zu binden



CODERS.BAY

BODY / ELEMENTE

› Unterscheidung ob sie sich selbst schließen oder einen Inhalt haben

- ▶ Elemente mit Start und End-Tag
 - > `Google`
 - > `<p>Mein Paragraph</p>`
- ▶ Selbstschließende Elemente
 - > `
`
 - > ``

› Unterscheidung zwischen Block- oder Inline-Element

- ▶ Block Elemente
 - > `h1, h2, h3 ...`
 - > `div, p`
- ▶ Inline Elemente
 - > `a, span, i`

- HTML unterstützt sechs Ebenen von Überschriften
 - ▶ <h1> bis <h6>
- steht im Tag-Body

```
<!DOCTYPE html>
<html>
  <!-- Das ist wirklich meine erste Webseite -->
  <head>
    <title>Meine erste Webseite</title>
  </head>
  <body>
    <h1>Das ist eine Überschrift der ersten Ordnung</h1>
    <p>Hallo Webwelt!</p>
  </body>
</html>
```


- `` ist ein sogenanntes Void-Element, ein HTML-Tag, das niemals einen Tag-Body haben kann.




```

```

relative oder absolute URL
des Bildes

kurze, aussagekräftige
Beschreibung des Bildes

- Höhe und Breite des Bildes: `height` und `width`
- nicht verpflichtend



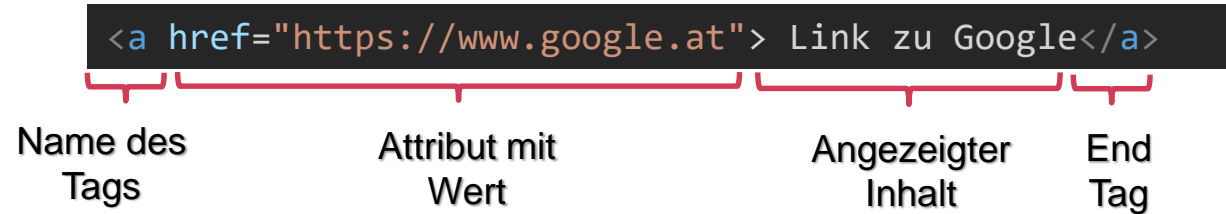
```

```

› Links

Textmarken, die man anklicken kann, um zu einer anderen Webseite oder einer anderen Stelle auf derselben Seite zu gelangen

- › Links werden auch Hyperlinks genannt
- › Tag: <a>



› Attribute

werden in ein öffnendes Tag geschrieben und haben einen Namen und meistens einen Wert

- › <a>-Tags, die das Attribut href haben, sind Links. Wenn man darauf klickt, gelangt man zu dem Ziel, das im Attribut href angegeben ist
- › `_blank`
Link wird in einem neuen Fenster bzw. Tab geöffnet.
Ob ein Fenster oder ein Tab geöffnet wird, liegt am Browser (default - `_self`)

```
<a href="..." target="_blank">Link</a>
```

Für jedes Tag gibt es eine Liste von gültigen Attributen.

In den meisten Browsern werden Links blau und unterstrichen angestellt

- Bei einem Link auf eine andere Seite wird kein Hash an den Anfang des `href` gestellt, die ist nur für Links innerhalb der Seite dar.
- Achte auf Groß- und Kleinschreibung. Es könnte sonst passieren, dass es auf dem Server nicht funktioniert.

➤ Diese Art von Link nennt man einen relativen Link, weil er Dateien relativ zu der Datei sucht, in der sich der Link befindet.

`Link zu einer anderen Seite`

➤ Die Zieldatei muss nicht im selben Verzeichnis liegen. In ein Unterverzeichnis kann man wie links angegeben verlinken.

`Link zu einer anderen Seite`

➤ In ein Verzeichnis weiter oben wie rechts angegeben

`Link zu einer anderen Seite`

➤ Dieser Wert von href ist technisch eine absolute URL

`Link zu einer anderen Webseite`

➤ `_blank`
Link wird in einem neuen Fenster bzw. Tab geöffnet. Ob ein Fenster oder ein Tab geöffnet wird, liegt am Browser



TABELLEN



- Früher hat man die ganze Webseite mit Tabellen gestaltet
 - ▶ Diente der einfachen Strukturieren von Elementen
 - ▶ Sehr starr und unflexibel
 - ▶ Nicht Responsive (auch für mobile Geräte schön darstellbar)

- Jetzt werden Tabellen nur noch im Content, zur Darstellung von Tabellen verwendet
 - ▶ `<table>`
 - ▶ `<tr>` - Table-Row
 - ▶ `<th>` - Table-Head
 - ▶ `<td>` - Table-Data

```
<table style="width:100%">
  <tr>
    <th>Firstname</th>
    <th>Lastname</th>
    <th>Age</th>
  </tr>
  <tr>
    <td>Jill</td>
    <td>Smith</td>
    <td>50</td>
  </tr>
  <tr>
    <td>Eve</td>
    <td>Jackson</td>
    <td>94</td>
  </tr>
</table>
```

- 2 verschiedene Arten von Listen:
 - ▶ `` - unordered List (Aufzählungszeichen)
 - Aufzählungszeichen kann man variieren
 - ▶ `` - ordered List (Nummerierung)
- In diesen Listen befinden sich Listen-Elemente
 - ▶ ``

```
<ul>  
  <li>Coffee</li>  
  <li>Tea</li>  
  <li>Milk</li>  
</ul>
```

- Ein HTML-Dokument ist valide, wenn es allen Regeln für HTML entspricht:
 - ▶ nur offizielle Tags und Attribute verwenden (an Stellen an denen sie erlaubt sind)
 - ▶ alle Tags korrekt schließen
 - ▶ verpflichtende Tags und Attribute setzen

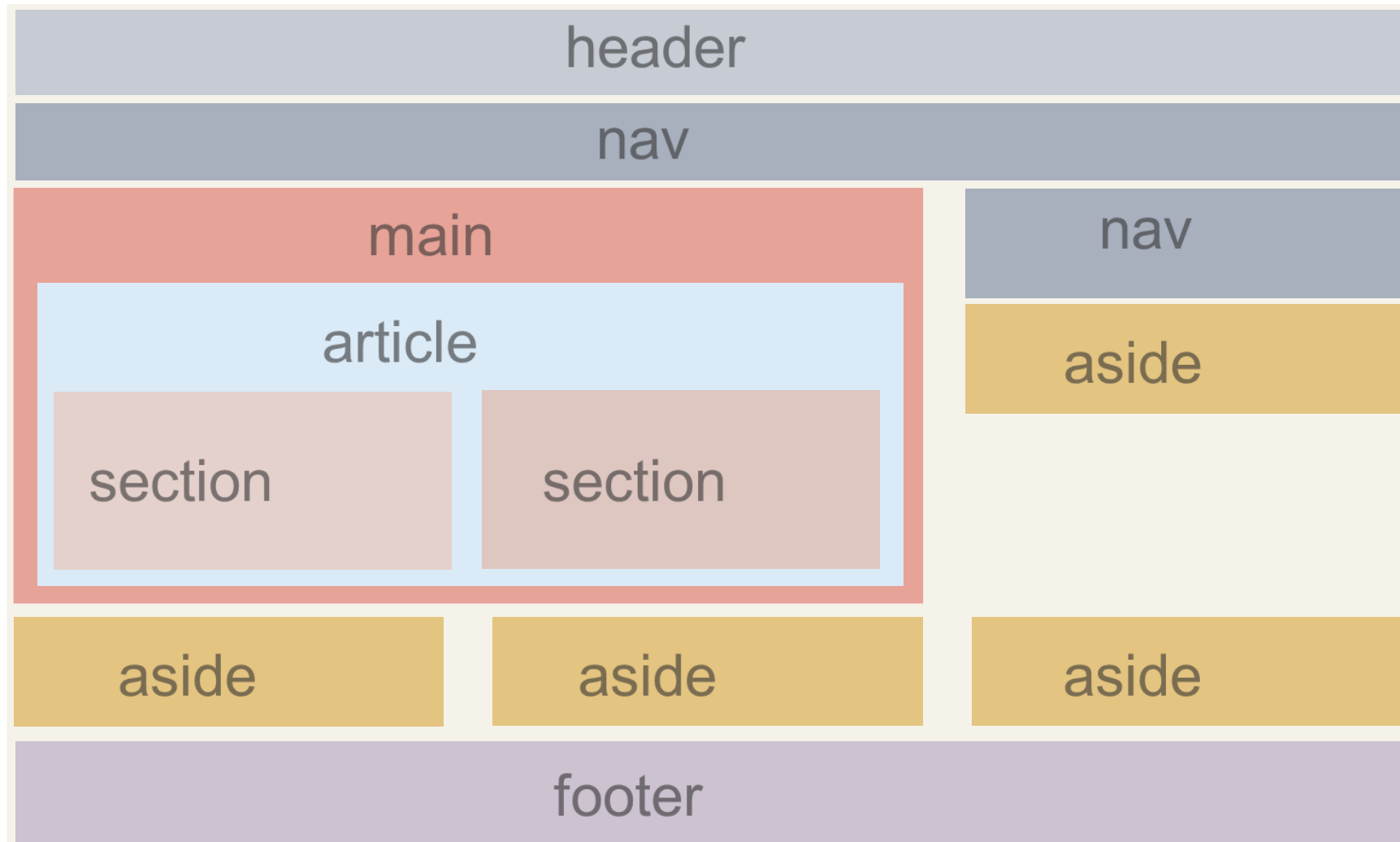
HTML-Dokument validieren:

- <https://validator.w3.org/>



CODERS.BAY

SEITENSTRUKTUR





NEUE ELEMENTE HELFEN BEI STRUKTURIERUNG



- Früher hat man alles in DIVs (**<div>** – division: Abteil, Bereich) getrennt
 - ▶ Container in dem man für gewöhnlich Elemente schachteln und strukturieren kann
 - ▶ Block-Element

- Jetzt gibt es neue Elemente, die uns zur Gliederung der Seite dienen:
 - ▶ `<header>` (nicht mit `<head>` verwechseln)
 - ▶ `<nav>`
 - ▶ `<main>`
 - ▶ `<article>`
 - ▶ `<section>`
 - ▶ `<aside>`
 - ▶ `<footer>`



ÜBUNG...



➤ Erstelle ein **valides** Dokument mit einer schönen Seitenstruktur:

- ▶ **header** mit
 - Bild
 - **nav** mit 3 Links (Home, About, Contact)
- ▶ **main** mit
 - Überschriften (jeweils über die Elemente😊)
 - 2 Texten (Lorem Ipsum)
 - Tabelle
 - geordneten Liste
- ▶ **aside** mit
 - ungeordneten Liste
- ▶ **footer** mit
 - Bild (Logo)
 - **nav** (Imprint, Privacy Policy)



MEHR NACHLESEN?



<https://www.w3schools.com/html/default.asp>



CODERS.BAY

FORMULARE



CODERS.BAY

ENDE