

SOFTWARE ENGINEERING 2

03 - HTML & CSS

Technologien auf dem Computer des Benutzers

Client Side

World Wide Web

Standards des WWW	
Name	Bedeutung
HTTP	Anwendungsprotokoll für die Kommunikation zwischen Client und Server
URL	Identifizierung von entfernten Ressourcen
HTML	Beschreibungssprache um verteilte Dokumente zu erstellen
CSS	Layoutdefinition von HTML-Dokumenten
DOM	Objektmodell eines HTML-Dokuments
JavaScript	Programmiersprache um HTML-Dokumente zur Laufzeit zu ändern

Grundstruktur

```
<!DOCTYPE html>
<html>
  <head>
    <title>Hochschule Furtwangen</title>
  </head>
  <body>
    Text, der auf der Webseite erscheint.
  </body>
</html>
```

- Dokumententyp, verwendete HTML Version
- Wurzelement: <html>
- Kopfelement: <head>
- Inhaltselement: <body>

Vorstellung der wichtigsten Elemente

HTML Einführung

Titel

- Jedes HTML-Dokument muss einen Titel enthalten
- Elementname "title"
- Kopfelement → wird nicht im Inhalt dargestellt
 - ▶ Titelzeile des Browserfensters
 - ▶ Beschriftung des Tabs
 - ▶ Lesezeichen
 - ▶ Suchmaschinen

Elemente zur Auszeichnung von Text

Text

Überschrift

- Auszeichnungselement für Überschriften
- Elementname "h..." von "header"
- sechs mögliche Verschachtelungsebenen (h1 bis h6), z.B.:
 - ▶ Kapitel
 - ▶ Abschnitt
 - ▶ Unterabschnitt

Überschrift

```
<h1>Überschrift</h1>
```

Text ohne weitere Auszeichnungen

Überschrift

Text ohne weitere Auszeichnungen

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

Überschrift 1

Überschrift 2

Überschrift 3

Überschrift 4

Überschrift 5

Überschrift 6

Textabsätze

- Browser ignorieren Formatierungen im Quelltext
 - Leerzeichen, Tabs, neue Zeile werden zu einem einzelnen Leerzeichen zusammengefasst
- Absätze müssen mit einem Element ausgezeichnet werden
- Elementname "p" von "paragraph"

Textabsätze

Formatierung im Quelltext
funktioniert so nicht

Formatierung im Quelltext funktioniert so nicht

`<p>Formatierung im Quelltext</p>`
`<p>funktioniert so besser</p>`

Formatierung im Quelltext
funktioniert so besser

Zeilenumbbruch

- Browser brechen Absätze automatisch um
- HTML-Author kann Zeilen manuell umbrechen
 - ▶ Elementname "br" von "line break"
- Alternativ: sichtbare Trennlinie über gesamte Textbreite
 - ▶ Elementname "hr" von "horizontal rule"
- Leere Elemente

Zeilenumbruch

langer Text `
` Mit manuellem Umbruch

langer Text
Mit manuellem Umbruch

Textblock der durch eine `<hr/>` Trennlinie unterbrochen wird.

Textblock der durch eine

Trennlinie unterbrochen wird.

Aufzählungen

- Aufzählungen können als Liste ausgezeichnet werden
 - ▶ Elementname "ul" von "unordered list" → Symbole
 - ▶ Elementname "ol" von "ordered list" → Ziffern
- Innerhalb der Liste Einträge als Unterelemente
 - ▶ Elementname "li" von "list item"
- Verschachtelungen sind möglich

Aufzählungen

```
<ul>
  <li>erster Eintrag</li>
  <li>zweiter Eintrag</li>
</ul>
```

- erster Eintrag
- zweiter Eintrag

```
<ol>
  <li>erster Eintrag</li>
  <li>zweiter Eintrag</li>
</ol>
```

1. erster Eintrag
2. zweiter Eintrag

Logische Auszeichnungen

- Elemente um Text eine "Bedeutung" zu geben
- Elementname "em" von "emphasis"
→ Betonung des Inhalts → kursive Darstellung
- Elementname "strong" von "strong emphasis"
→ starke Betonung des Inhalts → fette Darstellung
- Elementname "code"
→ Inhalt ist Programmcode → Schreibmaschinenschrift
- Verschachtelungen sind möglich

Logische Auszeichnungen

Dieser Satz enthält `betonte` und `stark betonte` Wörter, sowie einen Codeschnipsel `<code>GOTO 10</code>.`

Dieser Satz enthält *betonte* und **stark betonte** Wörter, sowie einen Codeschnipsel `goto 10`.

Beispiel Textelemente

```
<h1>Spaghetti Aglio et Olio</h1>
```

```
<h2>Zutaten</h2>
```

```
<ul>
```

```
  <li>&frac12; kg Spaghetti</li>
```

```
  <li>3 Zehen Knoblauch</li>
```

```
  <li>50 ml Olivenöl</li>
```

```
</ul>
```

```
<hr/>
```

```
<h2>Zubereitung</h2>
```

```
<p>
```

```
Knoblauch in Scheiben schneiden.
```

```
Spaghetti <em>10 Minuten</em> kochen.
```

```
Knoblauch mit Öl anbraten und Spaghetti  
unterrühren.
```

```
</p>
```

Spaghetti Aglio et Olio

Zutaten

- ½ kg Spaghetti
- 3 Zehen Knoblauch
- 50 ml Olivenöl

Zubereitung

Knoblauch in Scheiben schneiden. Spaghetti *10 Minuten* kochen. Knoblauch mit Öl anbraten und Spaghetti unterrühren.

Elemente zur Auszeichnung von Verweisen

Hypertext

Verweise

- Erlauben die Verknüpfung von einem HTML-Dokument auf ein anderes Dokument
- Sind unidirektional (kein zurück)
- Bestehen aus:
 - ▶ Verweistext
 - ▶ Verweisziel

Verweise

- Elementname "a" von "anchor"
- Inhalt des Elements ist der Verweistext
- Attribut "href" von "hypertext reference" ist die URL des Verweisziels
 - ▶ relativ
 - ▶ absolut
 - ▶ extern
 - ▶ email

Verweise

`http://www.hs-furtwangen.de/web/kurs/dieses.html`

Link zum ``anderen Dokument``.

`http://www.hs-furtwangen.de/web/kurs/jenes.html`

Link zum ``anderen Dokument``.

Link zum [anderen Dokument](#).

Link zum [anderen Dokument](#).



`http://www.hs-furtwangen.de/web/kurs/dieses.html`

Link zum ``anderen Dokument``.

→ `http://www.hs-furtwangen.de/web/jenes.html`

Elemente ohne vordefinierte Semantik

Universalelemente

Allgemeine Auszeichnungen

- Allgemeine Auszeichnungen
 - umschließen Text
 - (noch) keine Semantik, nützlich für CSS und JavaScript
- Allgemeines Block-Element
 - Elementname "div" von "division"
- Allgemeines Inline-Element
 - Elementname "span"

Block / Inline

- Alle Elemente von HTML gehören zu einer Kategorie
 - ▶ Block → Erzeugt einen rechteckigen Bereich
100 % Breite
 - ▶ Inline → Erzeugt einen Bereich, der dem Textfluss folgt.
- Verschachtelungen sind möglich
- Aber: Inline-Elemente dürfen keine Block-Elemente enthalten

Block / Inline

Dieser Satz zeigt, `<div>`wie unterschiedlich sich Block- und Inline-Elemente`</div>` verhalten.

Dieser Satz zeigt, wie unterschiedlich sich Block- und Inline-Elemente verhalten.

Dieser Satz zeigt, ``wie unterschiedlich sich Block- und Inline-Elemente`` verhalten.

Dieser Satz zeigt, wie unterschiedlich sich Block- und Inline-Elemente verhalten.

Kombinationen

`<div>`
von Block- und Inline-Elementen
``sind in HTML-Dokumenten``
`</div>`
möglich.

Kombinationen von Block- und Inline-Elementen sind in HTML-Dokumenten möglich.

allgemeine Attribute für alle Elemente

Universalattribute

Universalattribute

- Attribut "id"
 - ▶ individueller Name des Elements
 - ▶ muss in der HTML-Seite eindeutig sein
- Attribut "title"
 - ▶ kurze individuelle Beschreibung des Elements
 - ▶ wird als Tooltip angezeigt
- Attribut "style" und "class"
 - ▶ CSS spezifische Attribute, Erklärung später

Universalattribute

```
<h1 id="erste" title="Dieser Text wird  
angezeigt, wenn der Mauszeiger über  
dem Element steht">  
  Überschrift  
</h1>
```

Überschrift

Dieser Text wird angezeigt, wenn der
Mauszeiger über dem Element steht

Elemente zur Auszeichnung von Tabellen (Selbststudium)

Tabellen

Definition von Tabellen

- Zweck:
 - ▶ Darstellung tabellarischer Daten
 - ▶ Layoutdefinition (mehrspaltiger Text, Menüstruktur, ...)
→ Besser mittels CSS
- Mit oder ohne Gitternetz

Standards des WWW	
Name	Bedeutung
HTTP	Anwendungsprotokoll für die Kommunikation zwischen Client und Server
URL	Identifizierung von entfernten Ressourcen
HTML	Beschreibungssprache um verteilte Dokumente zu erstellen

Tabellen

- Elementname "table"
→ Definition einer Tabelle → kein Text, nur Unterelemente
- Elementname "tr" von "table row"
→ Definition einer Tabellenzeile → nur Unterelemente
- Elementname "td" von "table data"
→ Definition einer Tabellenzelle → Text
- Elementname "th" von "table header"
→ Titelzelle → Spezialform von "td" → zentriert und fett

Tabellen

<table>

<tr> <th> <th>

<tr> <td> <td>

<tr> <td> <td>



```
<table>
  <tr>
    <th>1</th>
    <th>2</th>
  </tr>
  <tr>
    <td>3</td>
    <td>4</td>
  </tr>
  <tr>
    <td>5</td>
    <td>6</td>
  </tr>
</table>
```

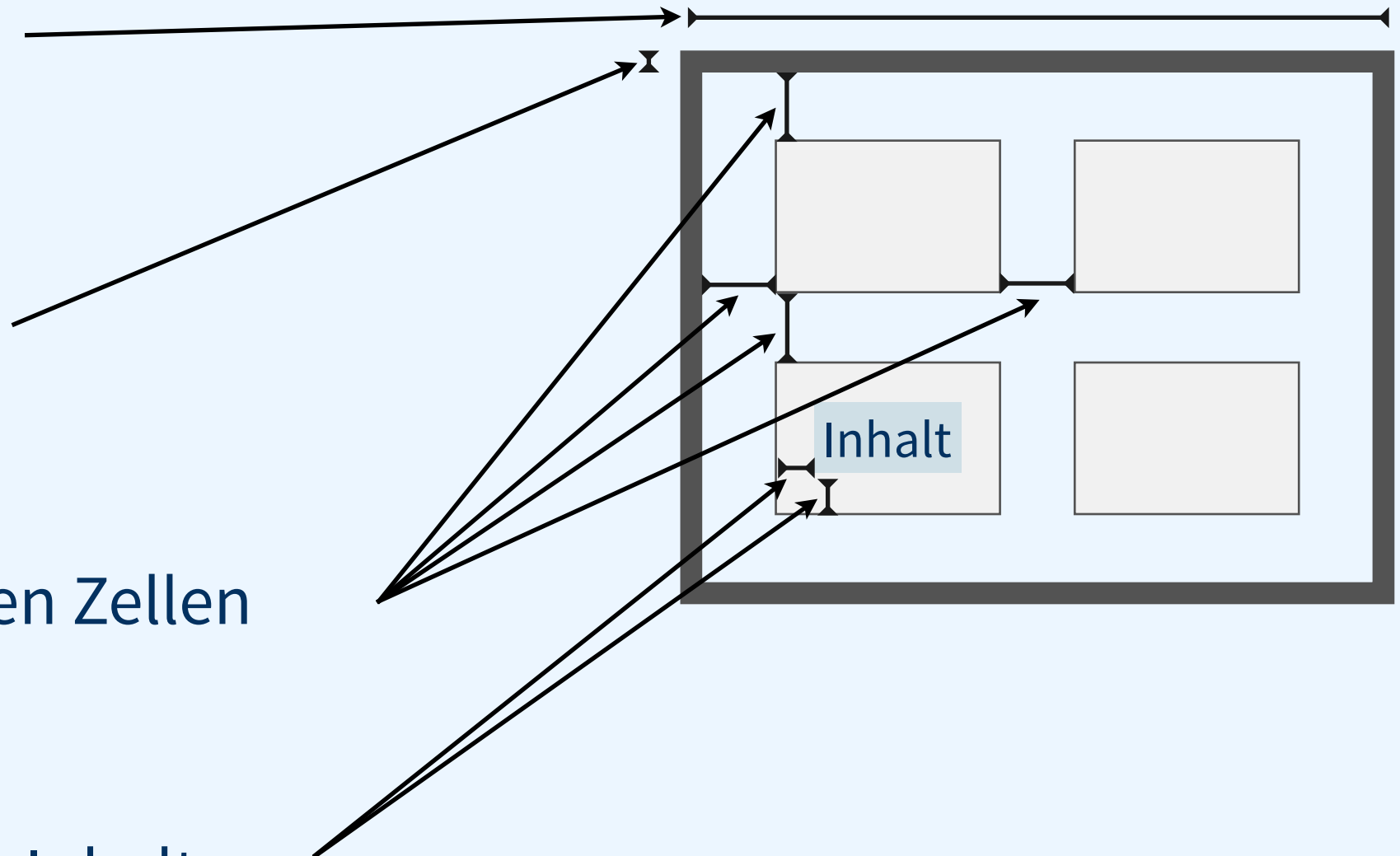


1	2
3	4
5	6

Element table

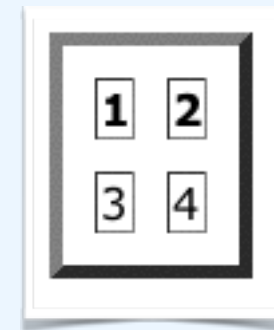
- Attribute:

- ▶ width
Tabellenbreite
- ▶ border
Rahmenbreite
- ▶ cellspacing
Abstand zwischen Zellen
- ▶ cellpadding
Abstand Zelle zu Inhalt



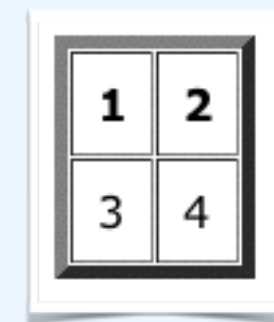
Element table

```
<table border="5"
        cellspacing="12"
        cellpadding="1">
  <tr>
    <th>1</th>
    <th>2</th>
  </tr>
  <tr>
    <td>3</td>
    <td>4</td>
  </tr>
</table>
```



1	2
3	4

```
<table border="5"
        cellspacing="1"
        cellpadding="9">
  <tr>
    <th>1</th>
    <th>2</th>
  </tr>
  <tr>
    <td>3</td>
    <td>4</td>
  </tr>
</table>
```



1	2
3	4

Element tr

- Attribute:

- ▶ `align = left | center | right`
Richtet Zellen einer Zeile horizontal aus
- ▶ `valign = top | middle | bottom`
Richtet Zellen einer Zeile vertikal aus

Element tr

```
<table border="1"
        width="200">
  <tr align="right">
    <th>1</th>
    <th>2</th>
  </tr>
  <tr align="center">
    <td>3</td>
    <td>4</td>
  </tr>
</table>
```

1	2
3	4

Element td

- Attribute:

- ▶ `align = left | center | right`
Richtet Zelle horizontal aus
- ▶ `valign = top | middle | bottom`
Richtet Zelle vertikal aus
- ▶ `width`
Breite der Zelle (und damit der Spalte) in Pixel
- ▶ `colspan`
Anzahl der Spalten, über die sich die Zelle erstreckt
- ▶ `rowspan`
Anzahl der Zeilen, über die sich die Zelle erstreckt

Element td

```
<table border="1" width="200">
  <tr>
    <th align="right">1</th>
    <th align="left">2</th>
  </tr>
  <tr>
    <td colspan="2" align="center">3</td>
  </tr>
</table>
```

1	2
3	

```
<table border="1" width="200">
  <tr>
    <th colspan="2" rowspan="2">1</th>
    <th>2</th>
  </tr>
  <tr>
    <td>3</td>
  </tr>
  <tr>
    <td>4</td>
    <td>5</td>
    <td>6</td>
  </tr>
</table>
```

1		2
		3
4	5	6

Beispiel Tabellenelemente

```
<table border="1">
  <tr>
    <th colspan="2">Standards des WWW</th>
  </tr>
  <tr>
    <th>Name</th>
    <th>Bedeutung</th>
  </tr>
  <tr>
    <td>HTTP</td>
    <td>Anwendungsprotokoll...</td>
  </tr>
  <tr>
    <td>URL</td>
    <td>Identifizierung...</td>
  </tr>
  <tr>
    <td>HTML</td>
    <td>Beschreibungssprache...</td>
  </tr>
</table>
```

Standards des WWW	
Name	Bedeutung
HTTP	Anwendungsprotokoll für die Kommunikation zwischen Client und Server
URL	Identifizierung von entfernten Ressourcen
HTML	Beschreibungssprache um verteilte Dokumente zu erstellen

Tabellen als Layoutmedium

```
<table border="0" cellpadding="0"
        cellspacing="20" width="400">
  <tr>
    <td colspan="2">
      <h1>Überschrift</h1></td>
    </td>
  </tr>
  <tr>
    <td valign="top">
      <h4>Navigation</h4>
      <p>Suche<br>
        Blog<br>
        Wiki<br>
        Mail</p>
      <h4>Aktuell</h4>
      <p>...</p>
    </td>
    <td valign="top">
      <h3>HTML Dokumente</h3>
      <p>Sind...</p>
    </td>
  </tr>
</table>
```

Überschrift

Navigation HTML Dokumente

Suche
Blog
Wiki
Mail

Aktuell

...

Sind reine Textdateien, die einer HTML-Konvention entsprechen. Sie bestehen aus darzustellenden Text und Strukturinformationen (Markup).

Elemente zur Auszeichnung von Bildern (Selbststudium)

Grafiken

Grafikelement

- Elementname "img" von "image"
- Attribut "src" von "source" definiert Bild-URL
- Attribut "alt" von "alternative" definiert Alternativtext
 - ▶ Falls Bild nicht geladen werden kann
 - ▶ Für barrierefreie Webseiten
- HTML definiert keine Grafikformate
 - ▶ unterstützt werden meist GIF, PNG, JPG

Grafikelement

```
<p>Monument Valley</p>  

```

Monument Valley



Grafikelement Attribute

- Größe mittels "width" und "height"
 - ▶ sollten immer angegeben werden
- Rahmen mittels "border"
- Ausrichten mittels "align"
 - ▶ "top", "middle" und "bottom"
 - ▶ "right" und "left"

Grafikelement

```
<p>Monument Valley</p>

```

```
<p>

Monument Valley
</p>
```

```
<p>
  
  Das Monument Valley...
</p>
```

Monument Valley



Monument Valley

Das Monument Valley ist eine Ebene auf dem Colorado Plateau an der südlichen Grenze des US-Bundesstaates Utah zu Arizona.



Design und Layout für HTML-Seiten

CSS

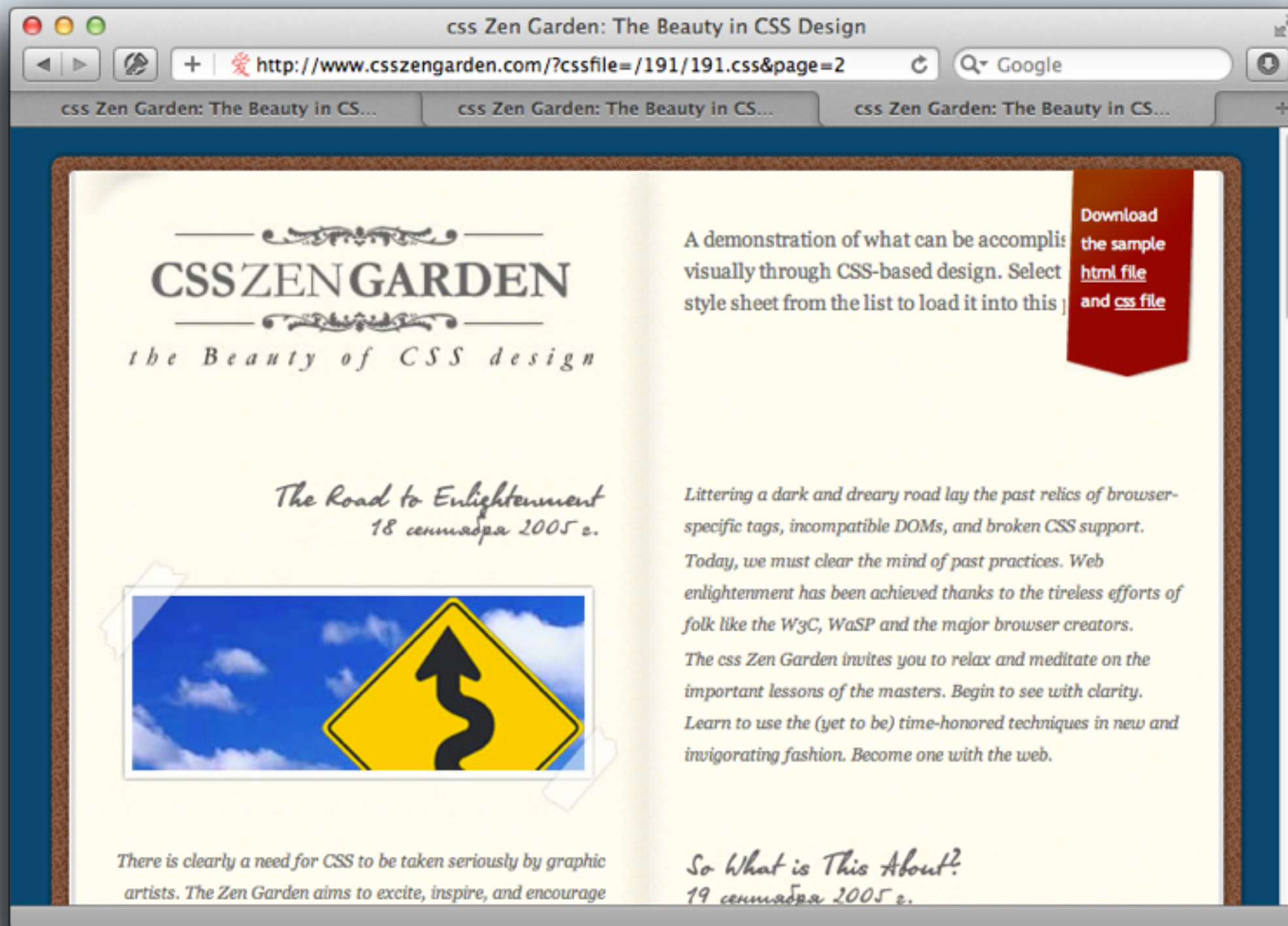
Motivation

- HTML beschreibt Inhalt und Struktur von Texten
- Keine/Kaum Aussagen über Aussehen und Gestaltung
- Redundante Definition in HTML
 - ▶ Beispiel Schriftart: muss in jedem HTML-Element definiert werden
 - ▶ häufige Wiederholung auf vielen Seiten
- Ziel: Auslagerung von Gestaltungsdefinitionen

Vorteile

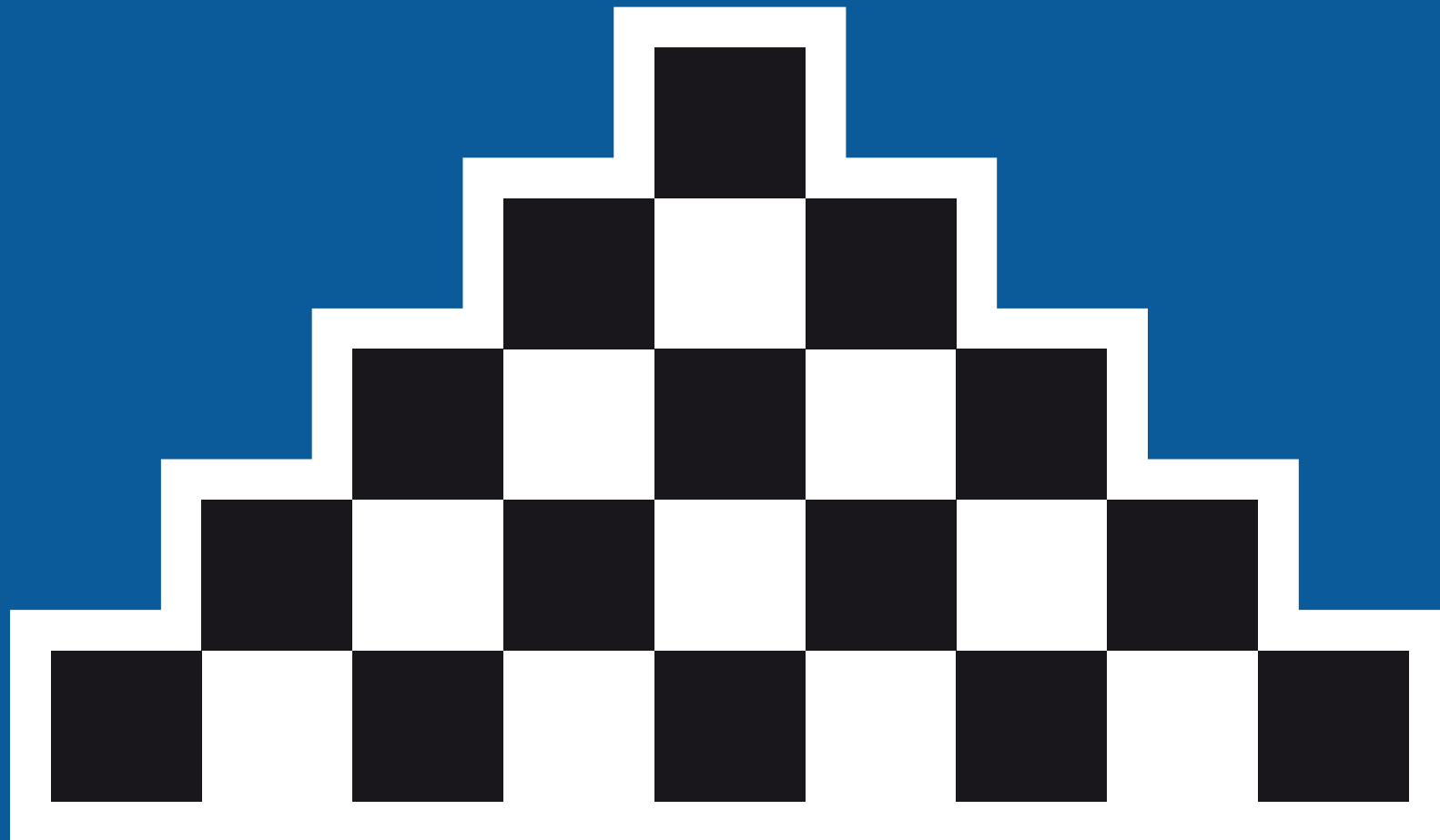
- Eine Gestaltungsdefinition für alle Seiten
- Zeit- und Platzersparnis:
 - ▶ muss nur einmal geladen werden
 - ▶ HTML-Seiten werden kompakter
- Gleicher Inhalt, viele Gestaltungen
 - ▶ <http://csszengarden.com>

CSS Zen Garden



Möglichkeiten

- Schrifteigenschaften (Schriftart, fett, kursiv, etc.)
- Farben (Vorder-, Hintergrund, Rahmenfarbe)
- Position und Abstände von Elementen
- Hintergrundbilder
- Mouse-Over-Effekte (z.B. für Links)



GRUNDLAGEN

CSS

- Cascading Style Sheets - kaskadierende Formatvorlagen
 - ▶ klingt einschüchternd
 - ▶ ist aber ein einfaches Konzept
- Andere Syntax als HTML, keine Elemente
- Wie nähern wir uns dem Begriff?
 - ▶ Formatvorlage jetzt
 - ▶ kaskadierende später

CSS - Begriffe

- Stylesheets sind eine Sammlung von (Gestaltungs-)Regeln
- Jede Regel besitzt die Form:
`Selektor { Eigenschaft: Wert; }`
- Eigenschaft definiert eine Gestaltungsmöglichkeit
 - z.B.: Hintergrundfarbe → `"background-color"`
- Wert definiert eine Zuweisung für die Eigenschaft
 - z.B.: die Farbe Gelb → `"yellow"`
- `background-color: yellow;`

Selektoren

- Auswahl der Elemente, auf die die Eigenschaften angewendet werden
- Im einfachsten Fall: Der Elementname
 - ▶ z.B.: h1 für das Element `<h1>Überschrift</h1>`
 - ▶ `h1 { background-color: yellow; }`
- Eine Selektor kann mehrere Elemente auswählen
- Viele weitere Selektor-Definitionen

Wie können CSS Regeln in HTML verwendet werden

Einbindung von CSS

Einbindung - Inline

- Im HTML-Dokument (Inline)
- Element "style" im "head" der HTML-Seite
 - ▶ Sammlung der Regeln als Inhalt dieses Elements
- Attribut "type" muss auf "text/css" gesetzt werden
 - ▶ potenziell verschiedene Sprachen für die Gestaltung
 - ▶ CSS ist aber die einzig verbreitete

Einbindung (Inline)

```
<!DOCTYPE html>
<html>
  <head>
    <style type="text/css">
      h1 { background-color: yellow; }
    </style>
  </head>
  <body>

    <h1>Überschrift</h1>
    <p>Absatz</p>

  </body>
</html>
```

Überschrift

Absatz

Einbindung - Extern

- Als externe CSS-Datei
- Element "link" im "head" der HTML-Seite
 - um externe Dateien logisch mit der Seite zu verknüpfen
- Attribut "rel" von "relationship" definiert Art der Beziehung
 - muss auf "stylesheet" gesetzt werden
- Attribut "type" muss auf "text/css" gesetzt werden
- Attribut "href" enthält URL der CSS Datei

Einbindung (Extern)

<http://www.hs-furtwangen.de/web/kurs/seite.html>

```
<!DOCTYPE html>
<html>
  <head>
    <link rel="stylesheet"
          type="text/css"
          href="formate.css">
  </head>
  <body>
    <h1>Überschrift</h1>
    <p>Absatz</p>
  </body>
</html>
```

Überschrift

Absatz

<http://www.hs-furtwangen.de/web/kurs/formate.css>

```
h1 {
  background-color: yellow;
}
```

Einbindung - Element

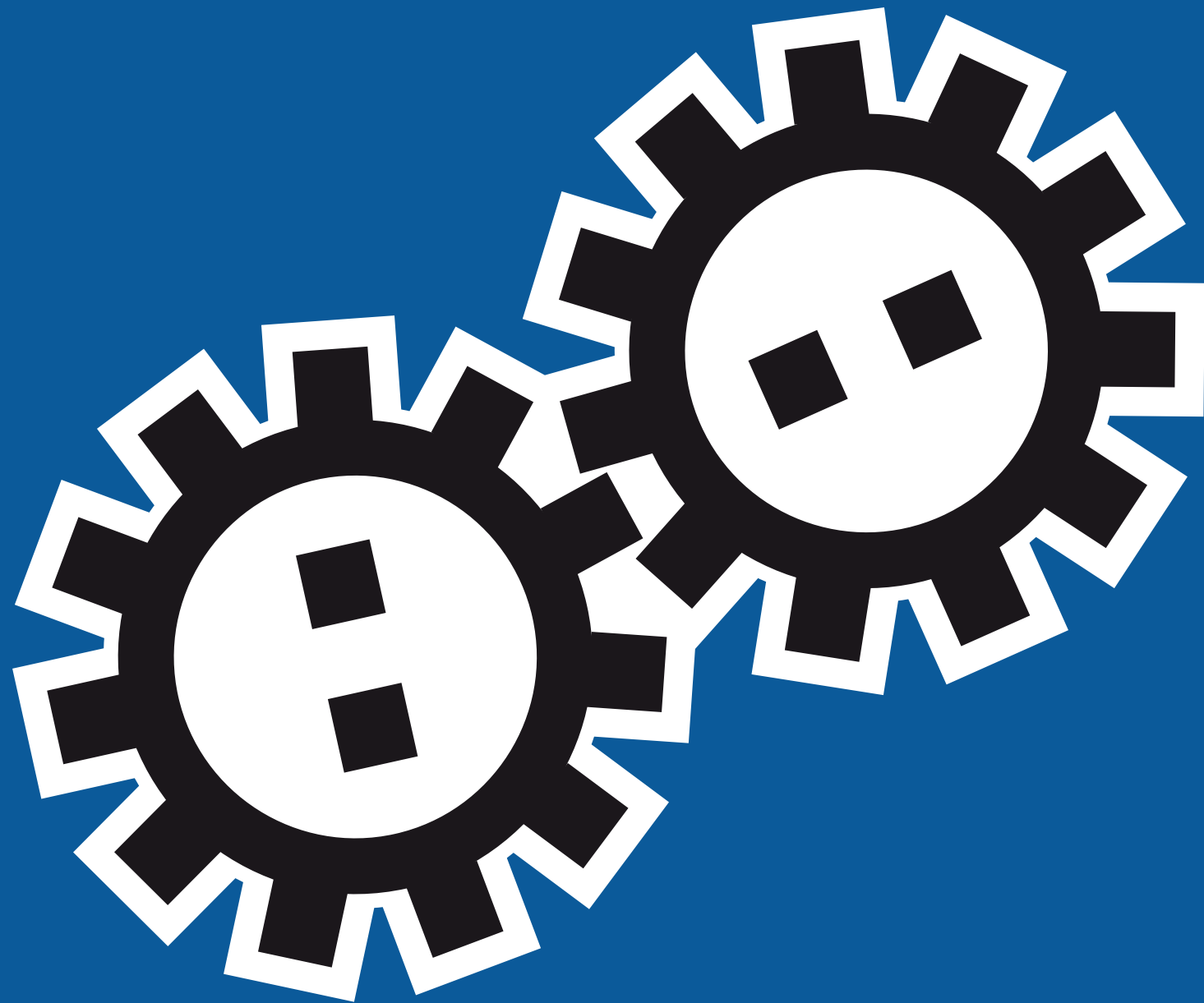
- Direkt an den Elementen
 - ▶ mit dem Universalattribut "style"
 - ▶ Teil eines Elements → deshalb ohne Selektor
`style="background-color: yellow;"`
- Gefahr von Redundanzen
 - ▶ nur in Ausnahmen den CSS-Dateien vorzuziehen

Einbindung (Element)

```
<!DOCTYPE html>
<html>
  <body>
    <h1 style="background-color: yellow;">
      Überschrift
    </h1>
    <p>Absatz</p>
  </body>
</html>
```

Überschrift

Absatz



TECHNIKEN

Allgemeine Hinweise

- Leerzeichen, Leerzeilen und Tabulatoren sind erlaubt
 - ▶ Strukturierung der Regeln
- Kommentare werden wie folgt definiert:
`/* Kommentar in CSS */`
 - ▶ dies unterscheidet sich von HTML:
`<!-- Kommentar in HTML -->`
 - ▶ einzeilige Kommentare funktionieren *nicht*:
`// kein Kommentar in CSS`

Auswahl der zu gestaltenden Elemente

Selektoren

Selektoren

- Universal-Selektor: *
- ▶ wählt alle Elemente einer Seite aus
- Element-Typ-Selektor: `ElementName`
- ▶ wählt alle Elemente eines Typs aus, z.B. `h1`
- ID-Selektor: `#IDName`
- ▶ wählt das Element mit dem entspr. ID-Attribut aus

Selektoren

```
<!DOCTYPE html>
<html>
  <head>
    <style type="text/css">
      *      { background-color: grey; }
      h1     { background-color: yellow; }
      #unten { background-color: aqua; }
    </style>
  </head>
  <body>
    <h1>Überschrift</h1>
    <p>oberer Absatz</p>
    <p id="unten">unterer Absatz</p>
  </body>
</html>
```

Überschrift

oberer Absatz

unterer Absatz

Selektoren

- Klassen-Selektor: `.Klassen-Name`
 - ▶ wählt alle Elemente einer Klasse aus
 - ▶ Klassen werden im Universal-Attribut `class` definiert
 - ▶ Im Gegensatz zur ID können mehrere Elemente die gleiche Klasse haben
 - ▶ Ein Element kann zu mehreren Klassen gehören
 - » durch Leerzeichen getrennt

Selektoren

```
<!DOCTYPE html>
<html>
  <head>
    <style type="text/css">
      .wichtig { background-color: red; }
      .info    { color          : blue; }
    </style>
  </head>
  <body>
    <h1 class="wichtig">Überschrift</h1>
    <p class="info">oberer Absatz</p>
    <p class="wichtig info">
      unterer Absatz
    </p>
  </body>
</html>
```

Überschrift

oberer Absatz

unterer Absatz

Selektoren

- ID oder Klasse?
- ID darf nur einmal je Seite auftauchen
 - ▶ Klassen beliebig oft
- Ein Element kann nur eine ID besitzen
 - ▶ aber beliebig viele Klassen
- Entscheidend: Ist die Gestaltung des Elements einmalig?
 - ▶ z.B. Navigationsmenü, Titelbereich, Fußzeile

Selektoren

- Pseudo-Klassen :Klassen-Name
 - wählt Elemente, die sich in einem bestimmten Zustand befinden
- :hover → Eigenschaften eines Elements, über das sich die Maus befindet
- :link → Eigenschaften eines Hyperlinks
- :visited → Eigenschaften eines besuchten Hyperlinks

Selektoren

```
<!DOCTYPE html>
<html>
  <head>
    <style type="text/css">
      a:link      { background-color: red; }
      a:visited   { background-color: green; }
      p:hover     { background-color: yellow; }
    </style>
  </head>
  <body>
    <a href="jenes.html">Hyperlink</a>
    <a href="weiteres.html">besuchter Hyperlink</a>
    <p>oberer Absatz</p>
    <p>unterer Absatz</p>
  </body>
</html>
```

Hyperlink ~~Hyperlink~~

oberer Absatz

unterer Absatz

Kombinationen

- Kombinationen sind möglich, z.B.:
 - ▶ `h1.info` → Alle h1 Elemente mit der Klasse "info"
 - ▶ `p.wichtig: hover` → Mouse-Over-Effekt für alle p Elemente mit der Klasse "wichtig"
- Wenn nur eine Möglichkeit offen ist, sollte diese auch angegeben werden:
 - ▶ schlecht: `#header`
 - ▶ besser: `div#header`

Mehrfachdeklaration

- Selektoren dürfen beliebig oft aufgeführt werden:

```
h1 { background-color: red;}  
h1 { color: blue;          }
```

- Können aber auch zusammengefasst werden:

```
h1 {  
    background-color: red;  
    color: blue;  
}
```

Gruppierungen

- Wenn mehrere Selektoren die gleichen Eigenschaften bekommen sollen → Gruppierung mittels Komma

- Statt:

```
h1      { color: red;}  
div     { color: red;}  
.info  { color: red;}
```

- geht auch:

```
h1, div, .info { color: red;}
```

Kaskaden

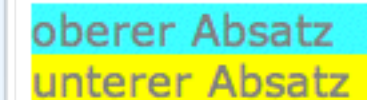
- Was passiert bei Widersprüchen?
- Regeln werden gewichtet (Auswahl):
 1. style-Attribute (`style="..."`)
 2. ID-Selektoren (`#unten {...}`)
 3. Klassen-Selektoren (`.wichtig {...}`)
 4. Element-Typ-Selektoren (`h1 {...}`)

Kaskaden

- Eigenschaften können mittels Kaskaden geschickt gemischt werden
- z.B.: Vorder- und Hintergrundfarbe für alle div-Elemente festlegen → Element-Typ-Selektor
- Hintergrundfarbe für ein einzelnes Element mit höherwertigem Selektor überschreiben, z.B. ID-Selektor
- Vordergrundfarbe bleibt unangetastet

Kaskaden

```
<!DOCTYPE html>
<html>
  <head>
    <style type="text/css">
      div {
        background-color: aqua;
        color           : grey;
      }
      #wichtig {
        background-color: yellow;
      }
    </style>
  </head>
  <body>
    <div>
      oberer Absatz
    </div>
    <div id="wichtig">
      unterer Absatz
    </div>
  </body>
</html>
```



oberer Absatz
unterer Absatz

Eine Auswahl der wichtigsten Gestaltungseigenschaften
(Selbststudium)

Eigenschaften

Übersicht

- Farben
- Text
- Ausrichtung
- Schrift
- (Hintergrundbild)
- (Listen)
- (Rahmen)
- (Abstände)
- (Positionierung)

Farben

- `color` (Textfarbe)
 - Name der Farbe
 - numerische Angabe der Farbe
- `background-color` (Hintergrundfarbe)
 - Name der Farbe
 - numerische Angabe der Farbe

Farbangaben

- Hexadezimal in der Form #RRGGBB (additiv)
ein Byte (0-ff hex, 0-255 dezimal) für Rot, Blau und Grün
#000000 → schwarz #ffffff → weiß #777777 → grau
#ff0000 → rot #00ff00 → grün #0000ff → blau
- Kurzschreibweise #RGB
#f00 → #ff0000 #123 → #112233 #a03 → #aa0033
- oder Farbnamen (nächste Folie)

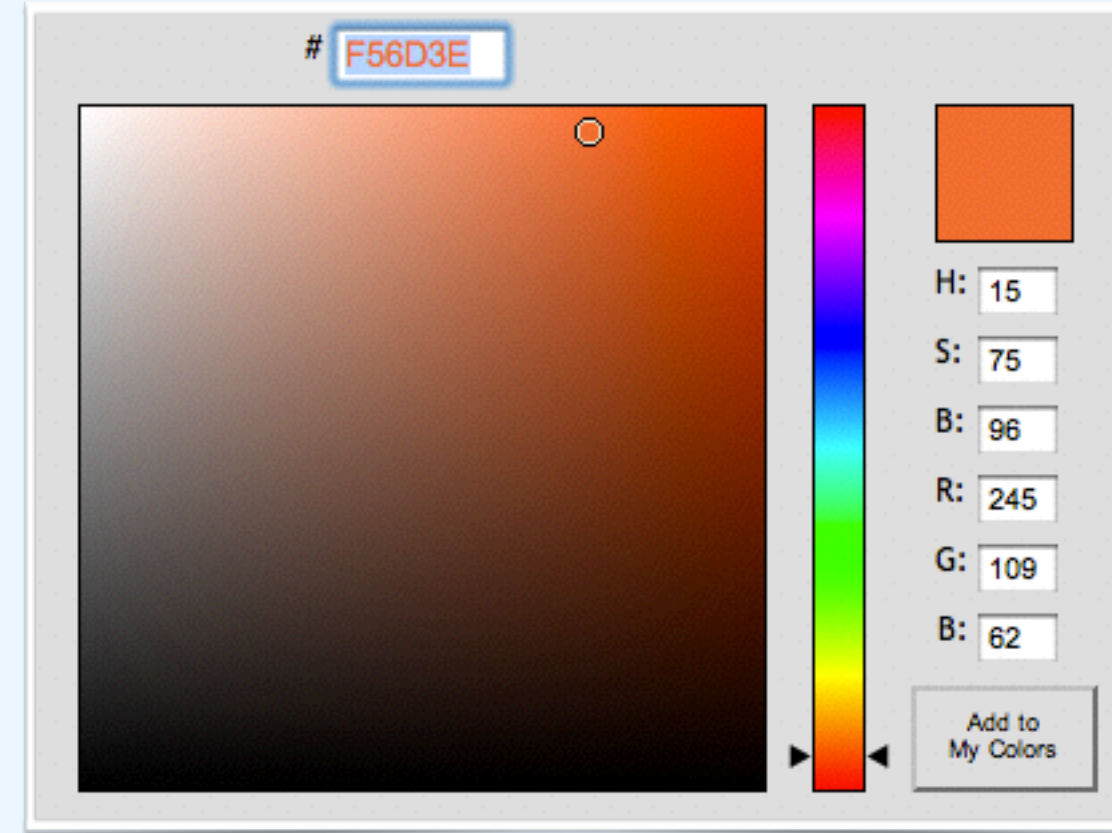
Farbnamen (Auswahl)

Name	deutscher Name	Hexadezimalcode	Farbe
black	schwarz	#000000	
maroon	bordeauxrot	#800000	
green	grün	#008000	
navy	dunkelblau	#000080	
gray	grau	#808080	
red	rot	#ff0000	
yellow	gelb	#ffff00	
blue	blau	#0000ff	
aqua	türkis	#00ffff	
white	weiß	#ffffff	

Farben

```
<!DOCTYPE html>
<html>
  <head>
    <style type="text/css">
      .wichtig {
        color : #f56d3e;
      }
      .info {
        color : #00f;
      }
    </style>
  </head>
  <body>
    <p>
      Absatz mit
      <span class="wichtig">wichtigen</span>
      und
      <span class="info">informativen</span>
      Elementen!
    </p>
  </body>
</html>
```

Absatz mit **wichtigen** und **informativen** Elementen!



Text

- `text-decoration` (Textdekoration)
 - `underline` (unterstrichen)
 - `overline` (überstrichen)
 - `line-through` (durchgestrichen)
- `text-transform` (Text-Transformation)
 - `capitalize` (Wortanfänge als Großbuchstaben)
 - `uppercase` (Nur Großbuchstaben)
 - `lowercase` (Nur Kleinbuchstaben)
- `letter-spacing` (Zeichenabstand)
 - numerischer Wert (Abstand zwischen den Buchstaben)

Exkurs: Numerische Werte

Name	CSS Einheit	Relation	Bedeutung
Pixel	px	absolut	Angabe in Bildschirmpixel
Millimeter	mm	absolut	Angabe in Millimetern
Zentimeter	cm	absolut	Angabe in Zentimetern
Prozent	%	relativ	Angabe relativ zu einer anderen Größe abhängig von der verwendeten Eigenschaft, z.B. zum Elternelement

Text

```
<!DOCTYPE html>
<html>
  <head>
    <style type="text/css">
      .wichtig {
        text-transform : uppercase;
        letter-spacing : 200%;
      }
      .info {
        text-decoration : underline;
      }
    </style>
  </head>
  <body>
    <p>
      Absatz mit
      <span class="wichtig">wichtigen</span>
      und
      <span class="info">informativen</span>
      Elementen!
    </p>
  </body>
</html>
```

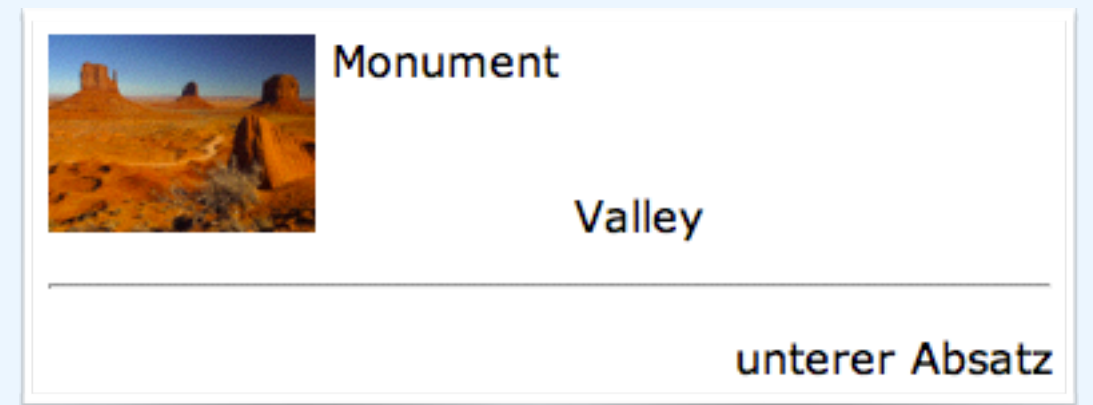
Absatz mit
W I C H T I G E N
und informativen Elementen!

Ausrichtung

- `line-height` (Zeilenhöhe)
numerischer Wert (Abstand zwischen den Zeilen)
- `vertical-align` (Vertikale Ausrichtung)
`top` (obenbündig ausrichten)
`middle` (mittig ausrichten)
`bottom` (untenbündig ausrichten)
- `text-align` (Horizontale Ausrichtung)
`left` (linksbündig ausrichten)
`center` (zentriert ausrichten)
`right` (rechtsbündig ausrichten)
`justify` (als Blocksatz ausrichten)

Ausrichtung

```
<!DOCTYPE html>
<html>
  <head>
    <style type="text/css">
      .oben    { vertical-align : top;    }
      .unten   { vertical-align : bottom; }
      .rechts  { text-align      : right; }
    </style>
  </head>
  <body>
    <p>
      
      <span class="oben">Monument</span>
      <span class="unten">Valley </span>
    </p>
    <hr/>
    <p class="rechts">
      unterer Absatz
    </p>
  </body>
</html>
```



Schrift

- `font-family` (Schriftart)
 - Name der Schriftart (muss vorinstalliert sein, z.B. Times)
 - `serif` (Schriftart mit Serifen)
 - `sans-serif` (Schriftart ohne Serifen)
 - `monospace` (Schriftart mit gleichbreiten Zeichen)
- `font-style` (Schriftstil)
 - `italic` (kursiver Schriftstil)

Schrift

- `font-weight` (Schriftgewicht)
`bold` (fett)
- `font-size` (Schriftgröße)
numerischer Wert
`small` (klein)
`medium` (mittel)
`large` (groß)

Schrift

```
<!DOCTYPE html>
<html>
  <head>
    <style type="text/css">
      .wichtig {
        font-family : Times;
        font-style   : italic;
        font-weight  : bold;
        font-size    : 30px;
      }
    </style>
  </head>
  <body>
    <p>oberer Absatz</p>
    <p class="wichtig">unterer Absatz</p>
  </body>
</html>
```

oberer Absatz

unterer Absatz



ZUSAMMENFASSUNG

Zusammenfassung

- Struktur von HTML
- Einführung in HTML
 - ▶ Text
 - ▶ Hypertext
 - ▶ Tabellen
- Grundlagen CSS
 - ▶ Möglichkeiten der Einbindung
- Techniken CSS
 - ▶ Selektoren, Vererbung und Verschachtelung
 - ▶ Eigenschaften

DANKE