A vertical blue decorative graphic on the left side of the slide, consisting of a series of overlapping, slightly curved lines that create a sense of depth and movement.

# Webbasierte Anwendungen

## CSS

Prof. Dr. Ludger Martin

# Gliederung

---

- ★ Grundlagen
- ★ Aufbau
- ★ Style-Selectoren
- ★ Styles

# Grundlagen

---

- ★ Cascading Style Sheets
- ★ Steuern das Layout einer Web-Seite bzw. Site
- ★ In HTML 2.0 wurden Tags benutzt (`<font>`)
- ★ Eigene Sprache, um Web-Seiten zu formatieren
  - ★ CSS1: Schriften, Farben, Anordnung, Ränder von Text und Bildern
  - ★ CSS2 (CSS2.1): Zusätzliche Unterstützung für mehr Medien-Typen
  - ★ CSS3: Erw. Farbmodell, Selectors, Med. Queries
- ★ Endung `.css` – Wird explizit einem/mehreren Dokument/-en zugeordnet

# Grundlagen

---

- ★ Layout einer Web-Seite bzw. Site kann durch Bearbeiten des CSS einfach geändert werden
- ★ Dokument wird für Entwickler lesbarer, da keine komplexen Formatanweisungen vorhanden sind.
- ★ Reduziert auch Größe einer Web-Seite
- ★ CSS kann für Ausgabegeräte (Drucker, ...) spezifisch angepasst werden
- ★ Formatregeln können für einen Tag oder für eine Reihe von Tags genutzt werden

# Grundlagen

- ★ CSS kann eingebettet (intern) oder durch Verweis (extern) eingebunden werden
- ★ Intern CSS
  - ★ Jedem Tag können direkt Styles zugeordnet werden (inline):

```
<... style="..."> ... </...>
```

```
<p style="font-style: italic">
```

```
Beispiel </p>
```

*Beispiel*

# Grundlagen

## ★ Intern CSS (Fortsetzung)

### ★ Mit

```
<span style="..."> ... </span>
```

kann für Teile von Text ein Format definiert werden.

`<span>` selbst verändert den Text nicht.

```
<span style="font-style: italic">
```

Beispiel

```
</span>
```

*Beispiel*

# Grundlagen

## ★ Intern CSS (Fortsetzung)

optional

- ★ Im head-Teil eines Dokuments können mittels `<style type="text/css">...</style>` Style-Regeln für ein ganzes Dokument angegeben werden (eingebettet):

```
<head>
  <style>
    p {font-style: italic}
  </style>
</head>
```

# Grundlagen

## ★ Externe CSS

- ★ Können zu mehreren Dokumenten zugeordnet werden

- ★ Verweis:

- ★ `<link rel="stylesheet" type="text/css" href="..." />`



optional

- ★ Mehrere Styles können geladen werden

- ★ Import:

- ★ `<style> @import url(...);  
... </style>`

- ★ Muss am Anfang einer Style-Definition stehen
  - ★ Ermöglicht verschachtelte Style Sheets
  - ★ Das letzte importierte Stylesheet überschreibt andere



# Grundlagen

## ★ Externe CSS

- ★ Können zu mehreren Dokumenten zugeordnet werden

- ★ Verweis:

★ `<link href="style.css" type="text/css">`

Zwecks Trennung von  
Inhalt und Layout, empfehle  
ich nur externe CSS!

★ `@import url(...);`

- ★ Muss am Anfang einer Style-Definition stehen
- ★ Ermöglicht verschachtelte Style Sheets
- ★ Das letzte importierte Stylesheet überschreibt andere

# Aufbau

## ★ Aufbau einer CSS-Formatanweisung

★ `StyleSelector{property:value;...}`

★ `StyleSelector` – Regelwerk für welche(s) Tag(s) das Format gilt

★ `property, value` – Formatangabe

★ Beispiel:

```
p{font-size:12px;font-family:Courier New}
```

## ★ Bei der Angabe innerhalb eines Tags ist nur `property:value;... notwendig`

```
<span style="font-size:12px;
font-family:'Courier New' ">...</span>
```

# Style-Selektoren

- ★ **Universell**: für Tags im gesamten Dokument;  
Angabe des Tag-Namen:

```
h1 {text-align: center}
```

```
h1, h2 {text-align: center}
```

## Beispiel:

```
<h1>Überschrift 1</h1>
```

```
<h2>Überschrift 2</h2>
```

**Überschrift 1**  
Überschrift 2

# Style-Selektoren

★ **Kontext:** für Tags in einer speziellen Abfolge;  
Angabe der Tag-Namen Abfolge:

```
ol li {list-style: upper-alpha}
ol ol li {list-style: upper-roman}
```

```
<ol>
  <li>Eintrag 1
    <ol>
      <li>Eintrag 1-1</li>
      <li>Eintrag 1-2</li>
    </ol></li>
</ol>
```

A) Eintrag 1  
I. Eintrag 1-1  
II. Eintrag 1-2

Notation auch mit **,** erlaubt: `ol li, ol ol li { }`

# Style-Selektoren

---

- ★ **Normale Klassen:** formatiert eine Sorte von speziell benannten Tags im Dokument; Klassen-Name wird durch **.** vom Tag-Namen getrennt:

```
h1.zentriert {text-align: center}
```

Aufruf der Klasse im Tag mittels `class="..."`

```
<h1 class="zentriert"> ... </h1>
```

# Style-Selektoren

## ★ Normale Klassen speziell h1 Name

Mehrere Klassen werden mit  
Leerzeichen getrennt:

```
class="zentriert fett"
```

```
h1.zentriert {text-align: center}
```

Aufruf der Klasse im Tag mittels `class="..."`

```
<h1 class="zentriert"> ... </h1>
```

# Style-Selektoren

---

- ★ **Allgemeine Klassen:** formatiert beliebige, speziell benannte Tags im Dokument; Klassen-Name wird nur ein **.** vorangestellt:

```
.zentriert {text-align: center}
```

Aufruf der Klasse im Tag mittels `class="..."`

```
<h1 class="zentriert"> ... </h1>  
<p class="zentriert"> ... </p>
```

# Style-Selektoren

---

- ★ **ID**: formatiert genau einen Tag mit einer festgelegten ID; ID wird durch # von Tag-Name getrennt:

```
#ueberschrift1 {text-align: center}
```

Angabe der ID mittels `id="..."`

```
<h1 id="ueberschrift1"> ... </h1>
```



# Style-Selektoren

---

★ **Pseudoklassen**: für bestimmte Tag-Zustände

★ **Verweise**: Ansicht von Verweis-Zustand

★ `active` – gerade gewählt

★ `visited` – schon mal besucht

★ `link` – noch nicht besucht

★ **Interaktionen**: Darstellung bei Interaktion

★ `hover` – Maus über Element

★ `focus` – Cursor Position auf Element

★ Mit **:** zwischen Tag-Namen und Pseudoklasse

`a:visited {color: green}`

# Style-Selektoren

---

- ★ Pseudoklassen und normale Klassen können miteinander kombiniert werden:

```
a.gruen:visited {color: green}
```

- ★ Klassen erben die Style-Eigenschaften des Basis-Tags

- ★ Zuordnungsreihenfolge: **id** vor **Klasse** vor **Kontext**

- ★ Für verschiedene Ausgabemedien können unterschiedliche Regeln definiert werden:

```
@media screen {a {color: green}}  
@media print  {a {color: black}}  
@media all    {a {font-style: italic}}
```

# Style-Selektoren



nur  
CSS3

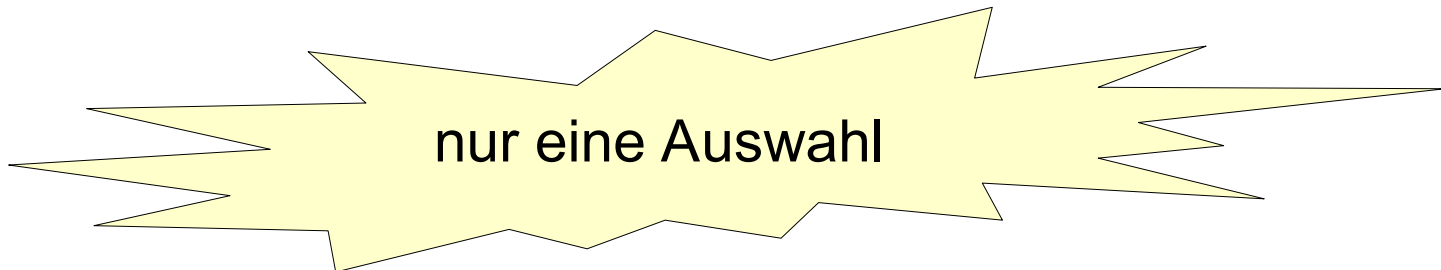
## ★ CSS3 bietet eine Reihe zusätzlicher Style-Selektoren

★ `:not(...)` – Negation

★ `:last-child` – letztes Kind

★ `:nth-child(x)` – Geschwisterelement  
z.B.  $2n+1$  bzw. `odd`

★ `[attr*="value"]` – Attribut enthält Wert




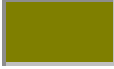






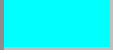



nur eine Auswahl

# Styles

## Farben und Ränder

- ★ Farben werden mit Farb-Namen oder in `#rrggbb` angegeben:

	<code>#000000</code> black		<code>#7f0000</code> maroon
	<code>#007f00</code> green		<code>#7f7f00</code> olive
	<code>#00007f</code> navy		<code>#7f007f</code> purple
	<code>#007f7f</code> teal		<code>#7f7f7f</code> gray
	<code>#bfbfbf</code> silver		<code>#ff0000</code> red
	<code>#00ff00</code> lime		<code>#ffff00</code> yellow
	<code>#0000ff</code> blue		<code>#ff00ff</code> fuchsia
	<code>#00ffff</code> aqua		<code>#ffffff</code> white

- ★ Alternativ

`rgb(rrr, ggg, bbb)`  
`rgba(rrr, ggg, bbb, alpha)`  
`hsl(hhh, sss%, lll%)`  
`hsla(hhh, sss%, lll%, alpha)`

bzw.



# Styles

## Farben und Ränder

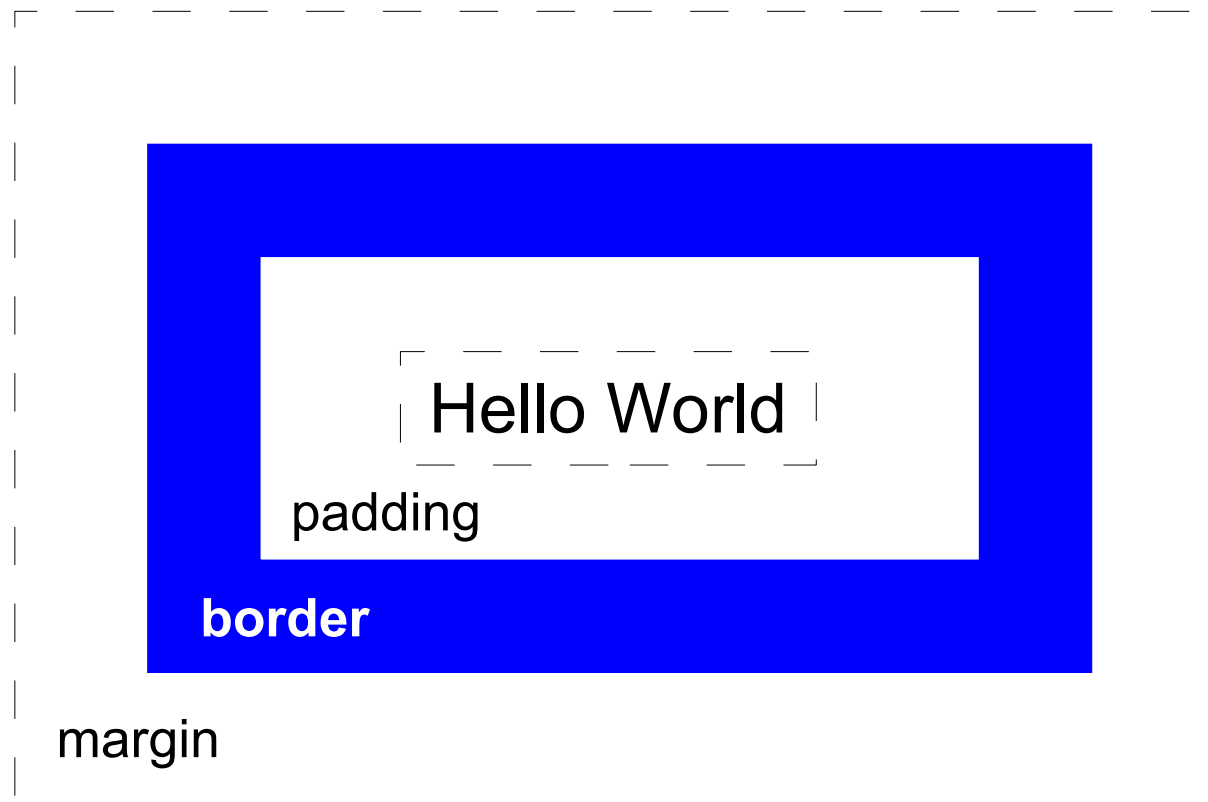
---

- ★ `color`: Farb-Namen oder Farb-Wert
- ★ `background-color`: Farb-Name oder Wert
- ★ `background-image`: `url(...)`
- ★ `border`: **Dicke** in px – **Farbe** – **Linientyp**  
(solid, dashed, dotted)  
**Bsp:** `border: 1px green solid`
- ★ `padding`: **Abstand** in px
- ★ `margin`: **Abstand** in px

# Styles

## Farben und Ränder

---



# Styles

## Farben und Ränder

---

★ Um einzelne Seiten der Ränder zu ändern:

★ `...-left`

★ `...-right`

★ `...-top`

★ `...-bottom`

★ Anwendbar für `border`, `margin`, `padding`

★ Beispiel:

`margin-left: 10px`

# Styles

## Positionierung

---

★ Positionierung kann u.a. mittels `<div>`-Tag vorgenommen werden:

★ Bezugspunkte:

★ `position: absolute` – gemessen am übergeordneten Element

★ `position: relative` – gemessen von der eigentlichen Position des Elements

★ `position: fixed` – gemessen am Browserfenster

★ `position: static` – keine Positionierung

★ Positionierung:

★ `top, bottom, left, right`



# Styles

## Positionierung

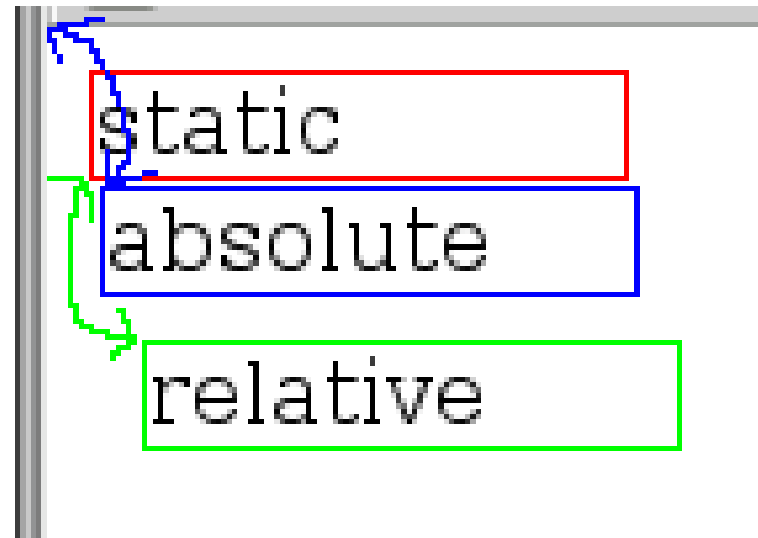
### ★ Positionierung Beispiel:

```
<body>
```

```
<div style="position:static;">static</div>
```

```
<div style="position:relative;  
  left:10px; top:30px;">relative</div>
```

```
<div style="position:absolute;  
  left:10px; top:30px;">absolute</div>
```



# Styles

## Positionierung

---

- ★ `height, width: px, ex`
- ★ `vertical-align: text-bottom (baseline),  
super (oberhalb), sub (unterhalb)`

# Styles

## Text

---

- ★ `text-align: right, left, center`
- ★ `text-decoration: none, blink, underline, line-through, overline`
- ★ `text-transform: none, capitalize, lowercase, uppercase`

# Styles

## Schriften

---

- ★ `font-family: font1, font2`
- ★ `font-size: z.B. in %, pt`
- ★ `letter-spacing: normal, ...em, ...ex`
- ★ `font-style: normal, italic, oblique`
- ★ `font-weight: normal, bold, bolder, lighter`



In CSS1 und CSS2 müssen  
die Schriften lokal installiert sein

# Styles Schriften

nur  
CSS3

## ★ In CSS3 Schriften vom Server laden

```
@font-face {  
  font-family: "...";  
  src: url(...) format("...");  
  font-weight: ...;  
  font-style: ...;  
}
```

Es können Alternativformate  
angegeben werden

optional

Es können für Formatierungen  
eigene Zeichensätze definiert werden

## ★ Derzeit gibt es kein Format, das alle Browser unterstützen (5 Formate möglich)

# Styles Schriften



nur  
CSS3

## ★ Beispiel mehrere Browser

```
@font-face {  
  font-family: "Caviar Dreams";  
  src: url(CaviarDreams.eot?)  
    format("ignoriert-mich"); /* IE 6-8 */  
  src: url(CaviarDreams.woff)  
    format("woff"); /* Moderne Browser */  
  src: url(CaviarDreams.ttf)  
    format("truetype"); /* Ältere Browser */  
  src: url(CaviarDreams.svg#webfont)  
    format("svg"); /* iPhone, iPad */  
}
```

★ **Nutzung:** `font-family: "Caviar Dreams";`

# Styles

## Schriften

---

- ★ Viele Schriften können Ladezeit extrem verlängern
- ★ Wenn Schrift noch nicht geladen, dann Standardschrift
- ★ Bitte Copyright beachten!
- ★ Nützliche Referenzen
  - ★ Google web fonts  
<https://fonts.google.com>
  - ★ Webfont Generator  
<https://www.fontsquirrel.com/tools/webfont-generator>

# Zusammenfassung

---

- ★ Vorgestellte Styles von CSS nicht vollständig, nur ein Auszug
- ★ Probieren Sie selbst aus, HTML-Dokumente mit CSS zu formatieren. Validieren Sie Ihre CSS-Dateien mit <http://jigsaw.w3.org/css-validator/>



# Literatur

---

- ★ Kröner, P.: HTML5 und CSS3, Video-Training, Galileo Computing, 2011
- ★ Lubkowitz, M: Webseiten programmieren und gestalten, 2. Auflage, Galileo Press, 2006
- ★ W3C: Cascading Style Sheets,  
<https://www.w3.org/Style/CSS/>