



FH Bielefeld
University of
Applied Sciences

Campus Minden

Webbasierte Anwendungen

SS 2018

CSS

Dozent: B. Sc. Florian Fehring
mailto: florian.fehring@fh-bielefeld.de

Studiengang Informatik

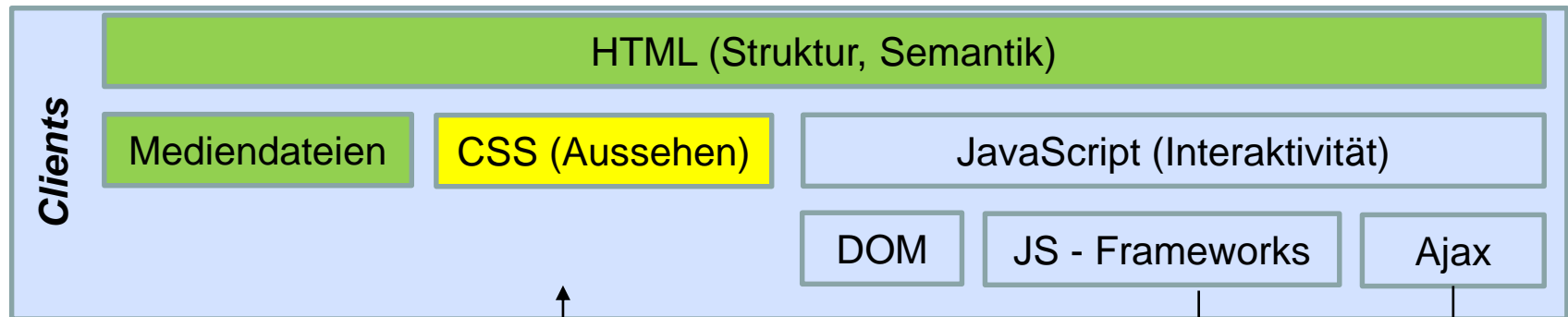
CSS

- 1. Kontext und Motivation**
2. CSS Notation und Einbindung
3. CSS Selektoren
4. CSS Eigenschaften
5. CSS Layout
6. CSS Animationen
7. Darüber hinaus
8. Projekt

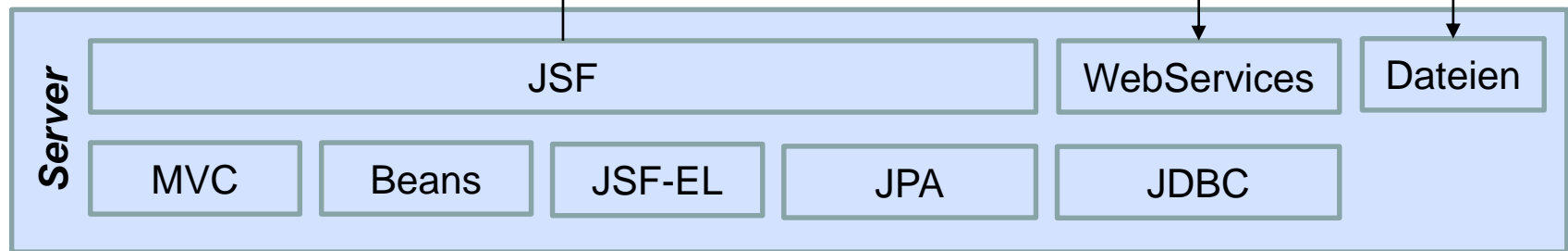
Problemfelder

Web-Anwendung

Mensch-Maschine-Kommunikation



Maschine-Maschine-Kommunikation



Anforderungen

Welche Anforderungen werden als nächstes bearbeitet?

TODO

- Schickes Design für die Seite
- Mediendateien einbinden
- Animationen
- Mehrsprachen-Fähigkeit
- (lokales) Speichern von Artikeln
- Client-Position anzeigen
- Offline-Verwendung ermöglichen
- Inhaltsverzeichnisse
- Medien bearbeiten
- Formlareingaben in Seite einfügen
- Navigation über Tastaturkürzel
- Externe Inhalte einbinden
- Medien hochladen / runterladen
- Kommentare hochladen / runterladen
- Kommentare speichern
- Kommunikation untereinander

DONE

- Technologische Grundlagen erarbeiten
- Was ist eine Web-Anwendung?
- News darstellen
- Projekte vorstellen
- Aufgaben darstellen
- Formular für Kommentare

CSS

1. Kontext und Motivation
- 2. CSS Notation und Einbindung**
3. CSS Selektoren
4. CSS Eigenschaften
5. CSS Layout
6. CSS Animationen
7. Darüber hinaus
8. Projekt

CSS

Definition: CSS (Cascading Style Sheets) ist die Standard - Stylesheetsprache für Webseiten. CSS formatieren und positionieren HTML-Elemente

Stylesheetsprache:

- Beschreibt das Aussehen eines Dokuments
 - Legt das Erscheinungsbild der Inhalte fest
 - Kann dynamisch geändert werden
- Dient nicht dem Implementieren von Algorithmen

CSS: (<https://www.w3.org/css/>)

- Definiert Selektoren mit denen HTML-Elemente erfasst werden können
- Legt mögliche Style-Attribute fest
- Legt die möglichen Werte für Style-Attribute fest
- Legt die Interpretation von Positionierungsangaben fest
- Beschreibt wie die Style-Angaben vom Browser zu interpretieren sind

CSS

Browser:

1. Interpretieren Style-Anweisungen
2. Suchen die zu stylenden Elemente über Selektoren
3. Wenden die Style-Anweisungen an

Geschichte von CSS:

Idee Idee von Hakon Wium Lie und Bert Bos

CSS L1 CSS Level 1, Text-Auszeichnungen, einfache Selektoren

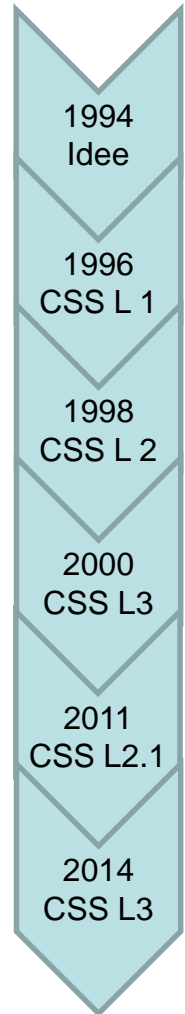
CSS L2 bis 2010 kaum Umsetzung

CSS L3 Modularer Aufbau, komplexe Selektoren, MediaQueries

CSS L2.1 Überarbeitung von L2

CSS L3 Seit 2014 breite Umsetzung von CSS L3 in Browsern

Hinweis: CSS L3 gilt noch als Draft, wenn es final ist, wird es eine W3C Recommendation



CSS I – Rule-Set

Definition: Ein CSS Rule-Set ist ein formeller Regelsatz, der das Aussehen unter festgelegten Bedingungen eines bestimmten HTML Elements festlegt.

```
Selector { Property: Value; Property: Value}
```

Selector:

- Wählt die Elemente aus, deren Eigenschaften gesetzt werden sollen

Property:

- Ist eine stylebare Eigenschaft eines Elements (hängt vom Element ab)

Value:

- Ist ein Wert, den die Eigenschaft annehmen kann

Mögliche Werte:

- Farbangaben: rgb(255,0,0), #ff0000, red
- Längen: 10px, 10pt, 10em
- Prozentuale Angaben: 10%
- Schlüsselwörter: Property-abhängige Schlüsselwörter

Weitere Informationen: Welche Werte bei welcher Property zulässig sind am besten aus einer CSS Dokumentation entnehmen. Z.B. <https://www.w3schools.com/css/default.asp>

CSS II - Einbindung

```
<head>
  <link rel="stylesheet" type="text/css" href="datei.css">
  <style>...</style>
</head>
<body>
  <p style="...">...</p>
</body>
```

Style-Einbindungen:

- **External:** Aus einer anderen Datei, link im Header
- **Internal:** In einem style-Tag im Header
- **Inline:** In einem style-Attribut bei einem Element

Media-Types:

```
<link rel="stylesheet" media="print" href="datei.css">
```

- Erlaubte Medien-Typen:
 - all, print, screen, speach

Media-Queries:

- Erlauben Einschränkungen nach Geräte-Eigenschaften

```
media="screen and (min-width: 480px) "
```

CSS III - Kaskadierung

Definition: Kaskadierung (Cascading) bezeichnet die Eigenschaft von CSS, dass mehrere Stylesheets zu einem gemeinsamen Stylesheet zusammengefasst werden.

Kaskadierungs-Reihenfolge:

1. Browser
2. Extern und Intern (in Deklarations-Reihenfolge)
3. Inline

```
<head>
  <style>
    p {
      color: orange;
    }
  </style>
  <link rel="stylesheet" href="stylestack.css"> //color: blue;
</head>
<body>
  <p style="color: green;">Essen 2.0 in der Mensa!</p>
</body>
```

Dies ist mein erster Artikel
In diesem Bereich stehen die News

Jetzt Essen 2.0 in der Mensa!

CSS IV – Style-Stack

Die Festlegung eines Style-Stack legt den Aufbau und die Verteilung der Style-Anweisungen fest und sollte in der Planungs-Phase einer Webanwendung erfolgen.

Beispiel Style-Stack:

1. Extern (global): Für die gesamte Anwendung (Look-And-Feel)
2. Extern (area): Für Elemente, die nicht überall, aber auf mehreren Seiten zu finden sind.
3. Intern: Für Elemente, die nur auf einer Seite vorkommen
4. Inline: Falls ein Element ein spezielles Aussehen benötigt

Allgemeine Regel: Je tiefer im Stack, desto weniger soll dort geregelt sein

Vorteile:

- Leichte Wiederverwendung, gute Übersichtlichkeit
- Redundanzfreie Formatierung
- Eine Änderung für alle Elemente gleicher Verwendung
- Selektives überdecken wo es notwendig ist
- Kleinere (schnellere) Web-Anwendungen
- Formatierungsanweisungen können schneller gefunden werden
- Trennung sorgt für mehr Übersichtlichkeit in HTML Dokumenten

CSS IV – Style-Stack

Browser-Entwickler-Tools erlauben den Einblick in die Kaskadierung.

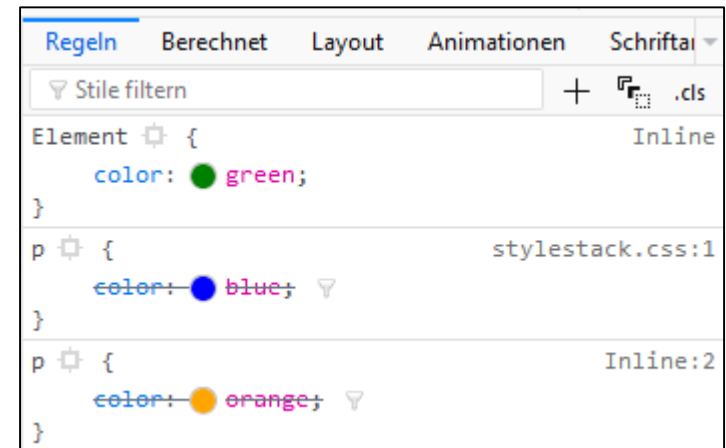
Vorgehen: (bspw. Firefox)

1. Rechtsklick auf ein Element
2. Element untersuchen
3. Entwickler-Konsole wird angezeigt
4. Regeln / Style zeigt die Kaskadierung

Dies ist mein erster Artikel
In diesem Bereich stehen die News

Jetzt Essen 2.0 in der Mensa!

```
<head>
<style>
  p {
    color: orange;
  }
</style>
<link rel="stylesheet" href="stylestack.css">
//color: blue;
</head>
<body>
  <p style="color: green;">Essen 2.0 in
der Mensa!</p>
</body>
```



CSS

1. Kontext und Motivation
2. CSS Notation und Einbindung
- 3. CSS Selektoren**
4. CSS Eigenschaften
5. CSS Layout
6. CSS Animationen
7. Darüber hinaus
8. Projekt

CSS Selektoren I – Standard Selektoren

```
.class { ... }  
#id { ... }  
* { ... }  
element { ... }
```

.class	Alle Elemente die diese Klasse haben (HTML-Attribut class="...") Ein Element kann mehrere Klassen besitzen class="a b c"
#id	Das Element mit der ID (HTML-Attribut id="...")
*	Alle Elemente
element	Das Element mit dem angegebenen Namen

```
<article id="b">  
  Dies ist mein erster Artikel  
</article>  
<section class="a">  
  In diesem Bereich stehen die News  
  <article class="a">  
    <p>Jetzt Essen 2.0 in der Mensa!</p>  
  </article>  
</section>
```

```
#b {  
  color: blue;  
}  
.a {  
  color: green;  
}  
article {  
  border: 1px solid red;  
}
```

Dies ist mein erster Artikel

In diesem Bereich stehen die News

Jetzt Essen 2.0 in der Mensa!

CSS Selektoren II – Attribut Selektoren

[attribute]

[attribute=wert]

...

[attribute]

Alle Elemente die ein Attribut mit diesem Namen haben

[attribute=wert]

Alle Elemente bei denen Das Attribut diesen Wert hat

Weitere Varianten unter: https://www.w3schools.com/cssref/css_selectors.asp

```
<article origin="general">
  Dies ist mein erster Artikel
</article>
<section>
  In diesem Bereich stehen die News
  <article origin="mensa">
    <p>Jetzt Essen 2.0 in der Mensa!</p>
  </article>
</section>
```

```
[origin] {
  color: blue;
}

article[origin=mensa] {
  border: 1px solid red;
}
```

Dies ist mein erster Artikel

In diesem Bereich stehen die News

Jetzt Essen 2.0 in der Mensa!

CSS Selektoren III – Pseudo-Klassen

```
:pseudo-class { ... }
```

pseudo-class:

- Sind Klassen, die ein Element durch den Browser aufgrund seiner Position im Dokument, seines Verhältnisses zum Benutzer oder dergleichen hat.

:hover	Alle Elemente, über die der Benutzer gerade seine Maus hat
:focus	Wenn ein Eingabeelement den Fokus hat
:link	Alle noch nicht besuchten Links
:first-child	Das erste Kind-Element (eines davor angegebenen Elements)

Weitere Varianten unter: https://www.w3schools.com/Css/css_pseudo_classes.asp

```
<article origin="general">
  Dies ist mein erster Artikel
</article>
<section>
  In diesem Bereich stehen die News
  <article origin="mensa">
    <p>Jetzt Essen 2.0 in der Mensa!</p>
  </article>
</section>
```

```
article:hover {
  color: blue;
}
article:only-child {
  border: 1px solid red;
}
```

Dies ist mein erster Artikel

In diesem Bereich stehen die News

Jetzt Essen 2.0 in der Mensa!

CSS Selektoren IV – Pseudo-Elemente

```
::pseudo-element { ... }
```

pseudo-element:

- Sind Elemente die implizit existieren.

::first-line	Die erste Zeile eines Textes
::first-letter	Der erste Buchstabe eines Textes
::before	Der Bereich vor einem Element

Weitere Varianten: https://www.w3schools.com/Css/css_pseudo_elements.asp

```
<article>
  Dies ist mein erster Artikel
</article>
<section>
  In diesem Bereich stehen die News
  <article>
    <p>Jetzt Essen 2.0 in der
Mensa!<br />Mit mehr Geschmack</p>
  </article>
</section>
```

```
::first-letter {
  color: blue;
}
p::first-line {
  color: red;
}
```

Dies ist mein erster Artikel
In diesem Bereich stehen die News

Jetzt Essen 2.0 in der Mensa!
Mit mehr Geschmack

CSS Selektoren V – Media-Queries

```
@media not|only mediatype and (mediafeature and|or|not  
mediafeature) {  
    CSS-Code;  
}
```

mediatype:

- all, print, screen, speech

mediafeature:

- Gerätespezifische-Eigenschaften

min-width Minimal benötigte darstellbare Breite

max-height Maximal benötigte darstellbare Höhe

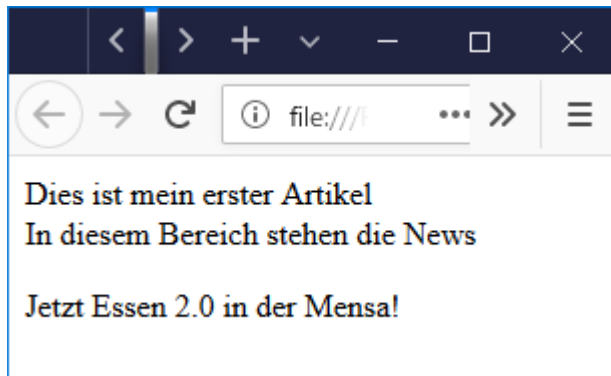
aspect-ratio Benötigtes Seitenverhältnis für die Anweisungen

Weitere Informationen: https://www.w3schools.com/cssref/css3_pr_mediaquery.asp

CSS Selektoren V – Media-Queries

```
<article>
  Dies ist mein erster Artikel
</article>
<section>
  In diesem Bereich stehen die News
  <article>
    <p>Jetzt Essen 2.0 in der
Mensa!<br />Mit mehr Geschmack</p>
  </article>
</section>
```

```
@media screen and (min-width: 400px) {
  p {
    color: blue;
  }
}
```



CSS Selektoren VI – Kombinationen

selector1, selector2, selector n
selector1 selector2

Kombinations-Operatoren:

, (**Kommata**) Wählt Selector1 oder Selector2 oder Selector n
(**Leerzeichen**) Alle Elemente von selector2 unter selector1

ACHTUNG! Speziellere Regeln haben immer Vorrang vor allgemeineren!

```
<article>
  Dies ist mein erster Artikel
</article>
<section>
  In diesem Bereich stehen die News
  <article>
    <p>Jetzt Essen 2.0 in der Mensa!</p>
  </article>
</section>
```

```
section article {
  color: blue;
}
section, article {
  color: red;
}
```

Dies ist mein erster Artikel
In diesem Bereich stehen die News
Jetzt Essen 2.0 in der Mensa!

Weitere Varianten: https://www.w3schools.com/cssref/css_selectors.asp

CSS

1. Kontext und Motivation
2. CSS Notation und Einbindung
3. CSS Selektoren
- 4. CSS Eigenschaften**
5. CSS Layout
6. CSS Animationen
7. Darüber hinaus
8. Projekt

CSS Eigenschaften I – Farben, Style

Darstellungs-Eigenschaften:

<code>color</code>	Schriftfarbe
<code>background-color</code>	Hintergrundfarbe
<code>font</code>	Eigenschaften der Schrift, wie Schriftart, Style
<code>text-shadow</code>	Schattenwurf für Texte
<code>text-overflow</code>	Verhalten bei überlangen Texten
<code>list-style-type</code>	Typ des Zeichens in Listen
<code>border-collapse</code>	Zusammenlegen der Ränder einer Tabellenspalte
<code>text-align</code>	Ausrichtung des Texts

Informationen zu Varianten und möglichen Werten:

color:	https://www.w3schools.com/Css/css_text.asp
font:	https://www.w3schools.com/Css/css_font.asp
Texteffekte:	https://www.w3schools.com/Css/css3_text_effects.asp
Listen-Design:	https://www.w3schools.com/Css/css_list.asp
Tabellen-Desing:	https://www.w3schools.com/Css/css_table.asp

CSS Eigenschaften II – Hintergründe, Rahmen

Einstellungen für Hintergründe:

<code>background-color</code>	Hintergrundfarbe
<code>background-image</code>	Hintergrundbilder
<code>background-repeat</code>	Wiederholung von Hintergrundbildern
<code>background-position</code>	Positionierung von Hintergründen

Rahmen:

<code>border-style</code>	Typ eines Rahmens (z.B. gepunktet, gestrichelt,...)
<code>border-color</code>	Farbe eines Rahmens
<code>border-radius</code>	Abrundung von Rahmen
<code>border</code>	Kombinierte Angaben zu Typ, Farbe, Dicke,...

Hintergründe und Rahmen sind für viele Elemente einstellbar.

Informationen zu Varianten und möglichen Werten:

Hintergründe: https://www.w3schools.com/Css/css_background.asp

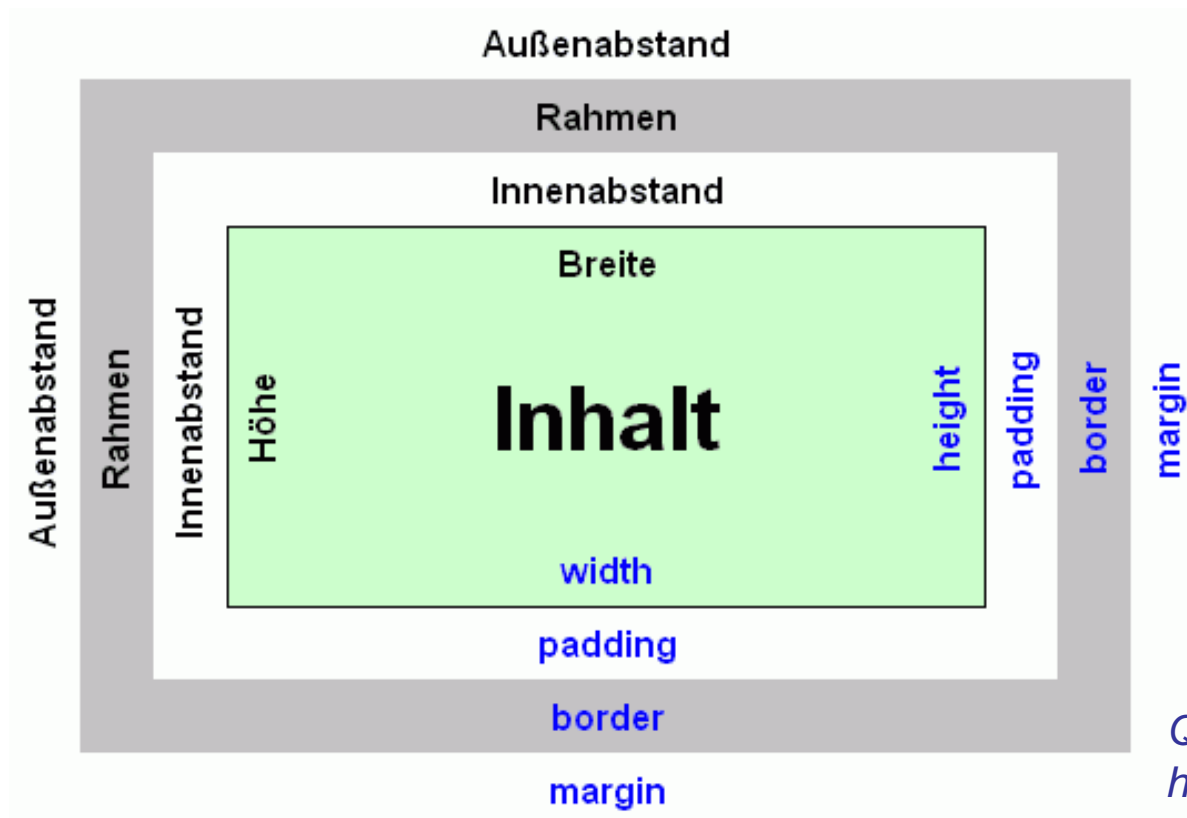
Rahmen: https://www.w3schools.com/Css/css_border.asp

CSS

1. Kontext und Motivation
2. CSS Notation und Einbindung
3. CSS Selektoren
4. CSS Eigenschaften
- 5. CSS Layout**
6. CSS Animationen
7. Darüber hinaus
8. Projekt

CSS-Layout I - Boxmodel

HTML-Auszeichnungen, **die einen Block oder Absatz bilden** werden in eine Box eingebettet. Die Box wird umgeben von einer **Polsterung (padding)**, dem **Rahmen (border)** und **dem Rand (margin)**. Die Gesamtausmaße der Box ergeben sich immer aus einer Addition dieser Größen.



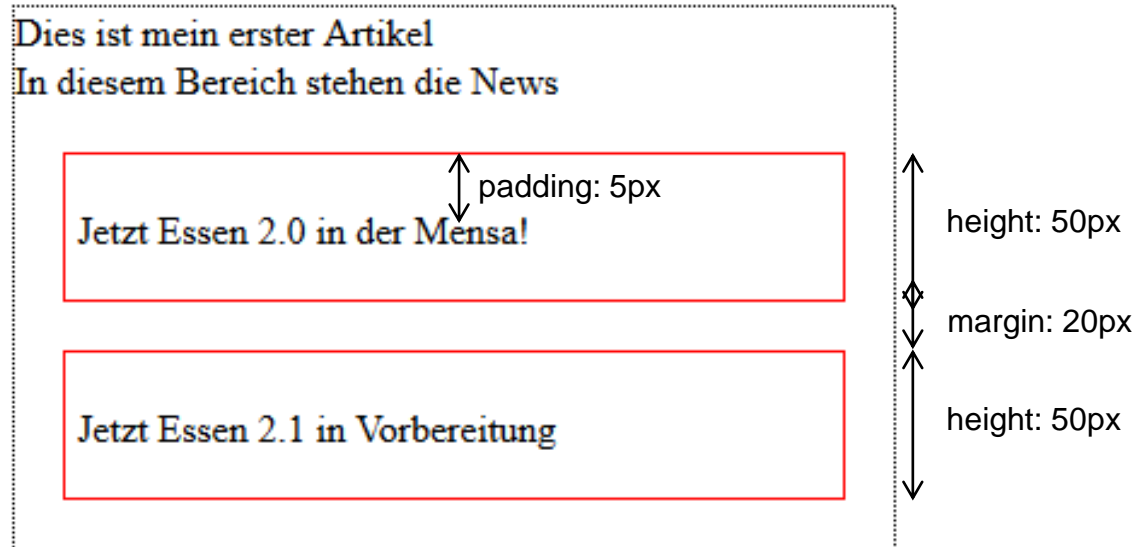
Quelle:
<http://de.selfhtml.org>

CSS-Layout I - Boxmodel

```
<body>
  <article>
    Dies ist mein erster Artikel
  </article>
  <section>
    In diesem Bereich stehen die News
    <article>
      <p>Jetzt Essen 2.0 in der Mensa!</p>
    </article>
    <article>
      <p>Jetzt Essen 2.1 in Vorbereitung</p>
    </article>
  </section>
</body>
```

```
body {
  border: 1px dotted;
}

section article {
  height: 50px;
  padding: 5px;
  border: 1px solid red;
  margin: 20px;
}
```



CSS-Layout II - Darstellung

Definition: Die Display-Eigenschaft stellt die Art der Darstellung ein.

display:

none	Das Element wird so dargestellt, als sei es nicht existent
inline	Kein Zeilenumbruch vor und hinter dem Element
block	Das Element wird wie ein Block-Element dargestellt
table	Zweidimensionale Tabellen-ähnliche Designs
positioned	Fix positionierte Elemente
flex	Flexible Positionierung, gleichmäßige Aufteilung, Stacking,...

visibility:

hidden Das Element wird nicht dargestellt, aber der Platz bleibt reserviert

Weitere Informationen: https://www.w3schools.com/Css/css3_flexbox.asp

CSS-Layout II - Darstellung

Display block, inline und none

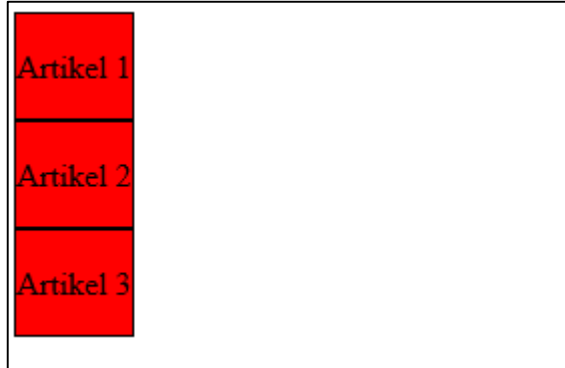
```
<body>
  <article>
    Dies ist mein erster Artikel
  </article>
  <section>
    In diesem Bereich stehen die News
    <article>
      <p>Jetzt Essen 2.0 in der Mensa!</p>
    </article>
    <article>
      <p>Jetzt Essen 2.1 in Vorbereitung</p>
    </article>
  </section>
</body>
```

```
article {
  display: none;
}
section article {
  display: inline;
}
section article p {
  display: inline;
}
```

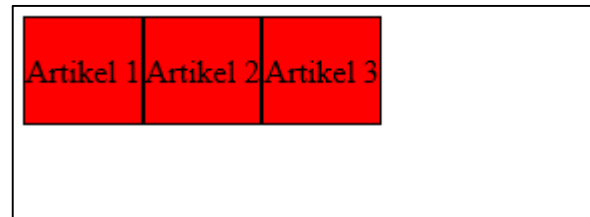
In diesem Bereich stehen die News Jetzt Essen 2.0 in der Mensa! Jetzt Essen 2.1 in Vorbereitung

CSS-Layout II - Darstellung

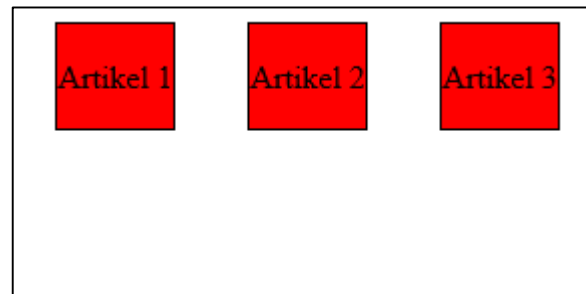
```
section {  
  display: flex;  
  flex-direction: column;  
}  
article {  
  background: red;  
  border: 1px solid black;  
  width: 20%;  
}
```



```
section {  
  display: flex;  
  flex-direction: row;  
}  
article {  
  background: red;  
  border: 1px solid black;  
  width: 20%;  
}
```



```
section {  
  display: flex;  
  flex-direction: row;  
  justify-content: space-around;  
}  
article {  
  background: red;  
  border: 1px solid black;  
  width: 20%;  
}
```



CSS-Layout III - Positionierung

position:

static

Normale Platzierung innerhalb des Seiteninhalts

relative

Positionsangaben gelten relativ zur normalen Platzierung

absolute

Positionierung absolut zum umgebenden Element

fixed

Absolut positioniert und bleibt beim Scrollen

sticky

Abhängig von scroll-Position. Bleibt sichtbar hängen

top, bottom:

Abstand vom oberen oder unteren Rand

left, right:

Abstand vom linken oder rechten Rand

```
<body>
  <section>
    <div>&nbsp;</div>
    <article id="art1">
      <p>Absolute</p>
    </article>
    <article id="art2">
      <p>Relative</p>
    </article>
    <article id="art3">
      <p>Fixed</p>
    </article>
  </section>
</body>
```

```
#art1 {
  position: absolute;
  top: 0px;
  left: 0px;
}
#art2 {
  position: relative;
  top: 0px;
  left: 0px;
}
#art3 {
  position: fixed;
  bottom: 0;
  right: 0;
}
```

Absolut

Relative

Fixed

CSS-Layout III - Positionierung

Weitere Positionierungs-Eigenschaften:

z-index

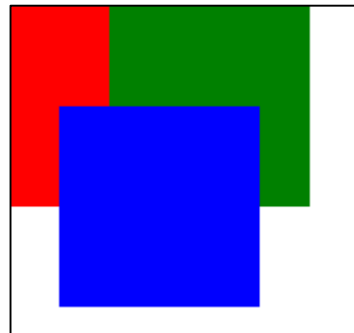
Ebene auf der das Element positioniert werden soll

Funktioniert nur für positionierte Elemente

box-sizing

Den Rahmen zur Höhe eines Elements rechnen

```
<body>
  <section>
    <article id="art1">
    </article>
    <article id="art2">
    </article>
    <article id="art3">
    </article>
  </section>
</body>
```



```
#art1 {
  position: absolute;
  top: 0px;
  left: 0px;
  z-index: 1;
  height: 100px;
  width: 100px;
  background: red;
}
#art2 {
  position: absolute;
  top: 0px;
  left: 50px;
  z-index: 2;
  height: 100px;
  width: 100px;
  background: green;
}
#art3 {
  position: absolute;
  top: 50px;
  left: 25px;
  z-index: 3;
  height: 100px;
  width: 100px;
  background: blue;
}
```

CSS

1. Kontext und Motivation
2. CSS Notation und Einbindung
3. CSS Selektoren
4. CSS Eigenschaften
5. CSS Layout
- 6. CSS Animationen**
7. Darüber hinaus
8. Projekt

CSS-Animationen I - Transitions

CSS Transitions erlauben es Änderungen an **CSS-Eigenschaften** animiert auszuführen, indem die Werte schrittweise geändert werden.

`transition: [eigenschaft] [dauer]`

[eigenschaft] Name der Eigenschaft, die Übergeleitet werden soll
Funktioniert nur für „animateable“ Eigenschaften

[dauer] Zeit in Sekunden, die der Effekt in Anspruch nehmen soll

```
<body>
  <a href="#">Ein Link</a>
</body>
```

```
a {
  font-size: 12pt;
  border-width: 1px;
  color: black;
}
a:hover {
  font-size: 30pt;
  color: red;
  transition: font-size 2s, color 5s;
}
```

Weitere Informationen: https://www.w3schools.com/Css/css3_transitions.asp

Animateable: https://www.w3schools.com/CSSref/css_animatable.asp

CSS-Animationen II - Transforms

CSS Transform erlauben es Elemente in ihrer Ausrichtung zu transformieren. Es gibt 2D und 3D Transformationen

`transform: [transformfunction]`

<code>translate()</code>	An der x und y Achse verschieben
<code>scale()</code>	Skalierung des Elements
<code>rotate()</code>	Rotation
<code>skew()</code>	Scherung
<code>matrix()</code>	Freie Transformation anhand einer Matrix

```
<body>
  <div>Container</div>
</body>
```

```
div {
  width: 100px;
  height: 100px;
  background-color: blue;
}
div:hover {
  transform: rotate(20deg)
}
```

Weitere Informationen: https://www.w3schools.com/Css/css3_2dtransforms.asp
https://www.w3schools.com/Css/css3_3dtransforms.asp

CSS-Animationen III - Animations

CSS Animations erlauben das Definieren eines komplexeren Verlaufes von Werteänderungen zum Erzeugen einer Animation.

```
@keyframes animationname {  
  from {eigenschaft: wert;}  
  to {eigenschaft: wert;}  
  x % {eigenschaft: wert;}  
}  
  
element {  
  animation-name: example;  
  animation-duration: 4s;  
}
```

@keyframes	Schlüsselbegriff zur Definition einer Animation
from	Startwerte der Eigenschaft(en)
to	Endwerte der Eigenschaft(en)
animation-name	Name der Animation, wie bei @keyframes angegeben
animation-duration	Dauer der Animation
x %	Eigenschaften zum Zeitpunkt x% der Animation

Weitere Informationen: https://www.w3schools.com/Css/css3_animations.asp

CSS-Animationen III - Animations

```
<body>
  <div>Container</div>
</body>
```

```
div {
  width: 100px;
  height: 100px;
  position: relative;
  background-color: red;
  animation-name: example;
  animation-duration: 4s;
}

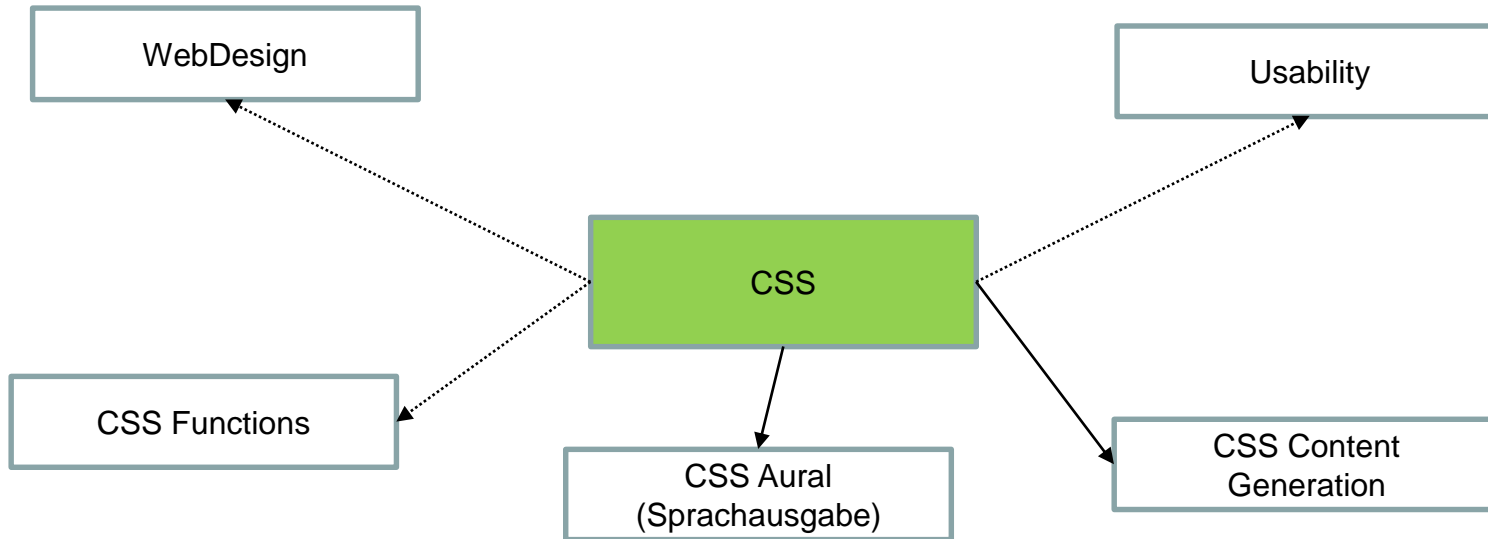
@keyframes example {
  0%   {background-color:red; left:0px; top:0px;}
  25%  {background-color:yellow; left:200px; top:0px;}
  50%  {background-color:blue; left:200px; top:200px;}
  75%  {background-color:green; left:0px; top:200px;}
  100% {background-color:red; left:0px; top:0px;}
}
```

Quelle: https://www.w3schools.com/Css/css3_animations.asp

Internet

1. Kontext und Motivation
2. CSS Notation und Einbindung
3. CSS Selektoren
4. CSS Eigenschaften
5. CSS Layout
6. CSS Animationen
- 7. Darüber hinaus**
8. Projekt

Darüber hinaus



Links:

CSS Functions:

CSS Content-Generation:

CSS Aural:

- https://www.w3schools.com/cssref/css_functions.asp
- https://www.w3schools.com/CSSref/pr_gen_content.asp
- https://www.w3schools.com/CSSref/css_ref_aural.asp

CSS: Einige Gestaltungshinweise

- **Usabilityanforderungen** sollten im Vordergrund stehen
- Die Seiten müssen leserlich sein z.B. in Farbauswahl, Kontrast, Schriftgröße
- Angleichung an spezielle Vorlagen in Farben oder komplettes Design einer Firma
- Browsernutzung im Kundenkreis
- Welche Hardware ? (kleinere Bildschirme auf MiniPC 's und Mobiles beachten)
- Design ist nicht im Mittelpunkt dieser Veranstaltung; sondern die **Befähigung zum Umsetzen eines vorgegebenen Designs** sowie Verständnis für Anforderungen von Gestaltern

CSS3 und HTML5 : Mobile Designprinzipien

Vorteile von HTML5 für Design mobiler Anwendungen

- Website auf Mobilem Gerät immer mit dabei
--> **Schnittstellen zur Kommunikation** anbieten wie z.B.
 - Telefonanrufe initiieren
 - E-Mails generieren
 - Inhalte in Soziale Netzwerke posten
- HTML5 nutzt Location-Info über Benutzer –
--> lokationsbasierte passgenaue Dienste anbieten
- HTML5 (+ Javascript) nutzen Nachladen von Teilen von Seiten

Besonderheiten Mobiler Anwendungen

- kleinerer Bildschirm
- Bedienung mit Finger
- Texteingabe unkomfortabel
- Unterwegs andere Anforderungen
- Verbindungsausfälle

Internet

1. Kontext und Motivation
2. CSS Notation und Einbindung
3. CSS Selektoren
4. CSS Eigenschaften
5. CSS Layout
6. CSS Animationen
7. Darüber hinaus
- 8. Projekt**

Anforderungen

Welche Anforderungen werden als nächstes bearbeitet?

TODO

- Schickes Design für die Seite
- Mediendateien einbinden
- Animationen
- Mehrsprachen-Fähigkeit
- (lokales) Speichern von Artikeln
- Client-Position anzeigen
- Offline-Verwendung ermöglichen
- Inhaltsverzeichnisse
- Medien bearbeiten
- Formlareingaben in Seite einfügen
- Navigation über Tastaturkürzel
- Externe Inhalte einbinden
- Medien hochladen / runterladen
- Kommentare hochladen / runterladen
- Kommentare speichern
- Kommunikation untereinander

DONE

- Technologische Grundlagen erarbeiten
- Was ist eine Web-Anwendung?
- News darstellen
- Projekte vorstellen
- Aufgaben darstellen
- Formular für Kommentare