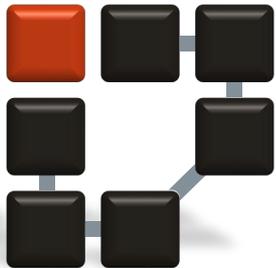


Informatik 1 für Nebenfachstudierende Grundmodul

HTML – Cascading Style Sheets

Kai-Steffen Hielscher
Folienversion: 22. November 2019



Informatik 7
Rechnernetze und
Kommunikationssysteme



FRIEDRICH-ALEXANDER
UNIVERSITÄT
ERLANGEN-NÜRNBERG
TECHNISCHE FAKULTÄT

Inhaltsübersicht

- Kapitel 2 - HTML
 - Einführung
 - Übersicht
 - HTML - Grundbegriffe
 - HTML - Texte und Verweise
 - HTML - Fortgeschrittene Techniken
 - **Cascading Style Sheets CSS**
 - Skripting

Cascading Style Sheets

- Einführung
- Bezug zu HTML
- CSS in HTML einbinden
- Genereller Formatdefinitionsaufbau
- weitere CSS-Eigenschaften
- HTML5

Einführung

- CSS ist eine Abkürzung für "Cascading Style Sheets"
- Erweiterung zu HTML, die vom W3C entwickelt wurde
- Ziel: Trennung von Inhalt und Gestaltung beim Web Design (Rückkehr zu den Wurzeln von HTML)
- Anwendung von CSS zur optischen Gestaltung der Inhalte, während HTML diese nur kennzeichnet und zur Verfügung stellt
- Formatvorlagen für ganze HTML-Dokumente (Logische Formatierung von Inhalten der HTML-Dokumente)
- "Make-up fürs Web"

Einführung

- Sprache zur Definition von Formateigenschaften einzelner HTML-Elemente.
 - Beispiel: Überschriften 1. Ordnung: Schriftgröße von 18 Punkt, Schriftart Helvetica, aber nicht fett, Abstand zum darauffolgenden Absatz von 1,75 Zentimeter ...
 - mit reinem HTML nicht möglich

Einführung

■ Weitere Möglichkeiten

- eigene Hintergrundfarbe, eigenes Hintergrundbild (Wallpaper), diverse Rahmen für nahezu beliebige HTML-Elemente
- pixelgenaue Positionierung von Elementen im Anzeigefenster des WWW-Browsers
- für Print-Layouts Möglichkeiten zur Definition von Seitenlayout und Textflusskontrolle
- akustische Wiedergabe von Web-Seiten: ganzes Arsenal an Befehlen zur Fernsteuerung der künstlichen Sprachausgabe
- spezielle Filter erlauben Grafik-Effekte bei Elementen, wie sie aus Grafikprogrammen wie Photoshop bekannt sind

Einführung

- **Definition zentraler Formate**
 - beispielsweise im Kopf einer HTML-Datei zentrale Definitionen zum Aussehen einer Tabellenzelle: Alle Tabellenzellen in der HTML-Datei erhalten diese Formateigenschaften, Reduktion von Codierarbeit und Größe der HTML-Dateien
- **Stylesheet-Definitionen in separaten Dateien möglich.**
 - die Stylesheet-Dateien sind in beliebig vielen HTML-Dateien referenzierbar
 - einheitliche Layouts für große Projekte
 - ein paar kleine Änderungen in einer zentralen Stylesheet-Datei erzeugen für Hunderte von HTML-Dateien ein anderes Aussehen
- **CSS unterstützt professionelle Gestaltung beim Web-Design und helfen beim Corporate Design für große Projekte oder für unternehmensspezifische Layouts**

Bezug zu HTML

- CSS Stylesheets erlauben Definition von Formaten
 - für bestimmte HTML-Elemente
 - für eine bestimmte Auswahl von HTML-Elementen
 - für Auswahl einer bestimmten Gruppe von HTML-Elementen, für die ein Format gelten soll, durch so genannte **Selektoren**
- CSS-Formate: eine oder mehrere Eigenschaften und Wertzuweisungen an diese Eigenschaften

Bezug zu HTML

- Drei verschiedene Möglichkeiten der Zuordnung von CSS-Formatierungen zu HTML-Dateien:
 - Direkte Formatierung
 - Zentrale Formatierung
 - Externe Formatierung

CSS in HTML einbinden

■ Direkte Formatierung

- Diese Formatierungen gelten nur für das speziell gewählte HTML-Elemente (ohne Bedeutung für die anderen Elemente)
- Verwendung des Attributs `style` im Start-Tag des entsprechenden HTML-Elements
`<element ... style="formatierung">...</element>`
- Parameter von `style` können die verschiedenen Formatierungen sein, notiert als
Eigenschaft:Wert oder
Eigenschaft:Wert1; Eigenschaft:Wert2; ...
- Beispiel:
`<h2 style="color:#00FF00">Überschrift in blau</h2>`
`<h2 style="color:black; background-color:yellow">Text</h2>`

CSS in HTML einbinden

■ Direkte Formatierung

```
<html>  
  <head>  
    <title>Titel</title>  
  </head>  
  <body>  
    <h1 style="[element-spezifische Formate]">text</h1>  
  </body>  
</html>
```

CSS-Definition als Attribut zum gerade aktuellen Tag, hier: h1. Wirkung ausschließlich auf diese Überschrift der Ebene 1, lokal beschränkt

text erhält
Formatierung gemäß
style

Beispiel: `style="background-color: white; color: red;"`

CSS in HTML einbinden

■ Direkte Formatierung

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
  <head>
    <title>Erstes CSS-Beispiel: Direkte Formatierung</title>
  </head>
  <body>
    <h1 style="color:#0000FF">Eine blaue &Uuml;berschrift
      der 1. Ordnung</h1>
    <h1 style="font-family:sans-serif">Eine &Uuml;berschrift
      der 1. Ordnung mit einer serifenlosen Schrift</h1>
    <h1 style="font-size:16px">Eine &Uuml;berschrift
      der 1. Ordnung mit einer 16 Pixel gro&szlig;en Schrift</h1>
  </body>
</html>
```

CSS in HTML einbinden

- Direkte Formatierung

Eine blaue Überschrift der 1. Ordnung

Eine Überschrift der 1. Ordnung mit einer serifenlosen Schrift

Eine Überschrift der 1. Ordnung mit einer 16 Pixel großen Schrift

CSS in HTML einbinden

■ Zentrale Formatierung

- Zentrale Formatierung unter Verwendung des `style`-Elements innerhalb des Gültigkeitsbereiches des `head`-Elements
- Außerdem Angabe des MIME-Typs, hier `text/css` als Parameter für das Attribut `type`
`<style type =“text/css“> ... </style>`

- Eigentliche Formatbeschreibungen werden mit Hilfe des Selector-Elements notiert:

```
Selector { Eigenschaft1:Wert1;  
          Eigenschaft2:Wert2;  
          EigenschaftN:WertN; }
```

- Beispiel:

```
h1 { color:#0000FF;  
     font-family:sans-serif;  
     font-size:16px; }
```

CSS in HTML einbinden

■ Zentrale Formatierung

■ Beispiel

```
<head>
```

```
...
```

```
<style type="text/css">  
  h1 { color:#0000FF;  
        font-family:sans-serif;  
        font-size:16px; }  
</style>
```

```
...
```

```
</head>
```

■ Bedeutung:

Jedem h1-Element innerhalb des HTML-Dokuments ist

- Farbe Blau (#0000FF),
- der Schrifttyp `sans-serif` und
- Schriftgröße 16 Pixel zugeordnet

CSS in HTML einbinden

- Zentrale Formatierung

- Ältere Browser verstanden kein CSS

- Inhalt des `style`-Elements in HTML-Kommentare einschließen:

- `<!-- ... -->`

CSS in HTML einbinden

■ Zentrale Formatierung

```
<html>
  <head>
    <title>Titel der Datei</title>
    <style type="text/css">
      <!--
        Formate
      -->
    </style>
  </head>
  <body>
    ...
  </body>
</html>
```

MIME-Typ

Bereich für CSS-
Formate

Kommentar für
alte Browser

CSS in HTML einbinden

■ Zentrale Formatierung

■ MIME: Multipurpose Internet Mail Extensions

- Datentypisierung für Mails und Web-Inhalte
- teilt Web-Browser bzw. Webserver mit, um welche Art von Daten es sich handelt
- adäquate Verarbeitung möglich

CSS in HTML einbinden

■ Zentrale Formatierung

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
  <head>
    <title>Zweites Stylesheet: Zentrale Formatierung </title>
    <style type="text/css">
      h1 { color:#336699;
          font-family:sans-serif; }
      p { font-size:14px;
          font-family:sans-serif; }
    </style>
  </head>
  <body>
    <h1>Zentrale Formatierung</h1>
    <p>Die Elemente, die in diesem HTML-Dokument verwendet werden,
      wurden durch CSS zentral formatiert.</p>
    <p>Dies kann besonders viel Arbeit sparen, wenn man dabei
      geschickt vorgeht.</p>
  </body>
</html>
```

CSS in HTML einbinden

- Zentrale Formatierung

Zentrale Formatierung

Die Elemente, die in diesem HTML-Dokument verwendet werden, wurden durch CSS zentral formatiert.

Dies kann besonders viel Arbeit sparen, wenn man dabei geschickt vorgeht.

CSS in HTML einbinden

■ Externe Formatierung

- Motivation: man möchte dieselben CSS-Formatierungen in mehreren HTML-Dokumenten verwenden
- Das Auslagern der Formatierungen wird als externe Formatierung bezeichnet und geschieht in Form einer gesonderten Datei mit Dateiendung `.css`
- Aufbau einer solchen CSS-Formatdatei ist völlig identisch zur zentralen Formatierung innerhalb der `style`-Tags
- Verbindung zwischen HTML-Datei und CSS-Formatdatei mittels `link`-Element im Gültigkeitsbereich des `head`-Elements

```
<head>
```

```
...
```

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

```
...
```

```
</head>
```

- Die Angaben `rel="stylesheet"` und `type="text/css"` geben die einzubindende Datei als Stylesheet des Typs `css` an (*rel* = relation)

CSS in HTML einbinden

■ Externe Formatierung

- Im Attribut href wird der Name der entsprechenden CSS-Datei notiert (ggf. relativen oder absoluten Pfad angeben)
- bei Pfadangaben gleiche Regeln wie bei Verweiszielen im a-Element der HTML-Webseitenerstellung

```
<html>  
  <head>  
    <title>Titel der Datei</title>  
    <link rel="stylesheet" type="text/css" href="formate.css">  
  </head>  
  <body>  
  </body>  
</html>
```

link: Verweis auf andere Datei, rel und type Pflichtangaben bei CSS, href wie bei Links

Definition der Formate in Datei **formate.css** im gleichen Verzeichnis

CSS in HTML einbinden

■ Externe Formatierung

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
  <head>
    <title>Drittes CSS-Beispiel: Externe Formatierung</title>
    <link rel="stylesheet" type="text/css" href="css-extern.css">
  </head>
  <body>
    <h1>Externe Formatierung</h1>
    <p>Die Elemente, die in diesem HTML-Dokument verwendet werden,
      wurden durch CSS zentral formatiert.</p>
    <p>Diese Formatierungen wurden aber in einer externen Datei
      gespeichert, um sie auch in anderen HTML-Dokumenten verwenden
      zu können.</p>
  </body>
</html>
```

CSS in HTML einbinden

■ Externe Formatierung

```
h1 { color: red;
      font-size:30px; }
p  { font-size:14px;
      font-family:sans-serif; }
```

Externe Formatierung

Die Elemente, die in diesem HTML-Dokument verwendet werden, wurden durch CSS zentral formatiert.

Diese Formatierungen wurden aber in einer externen Datei gespeichert, um sie auch in anderen HTML-Dokumenten verwenden zu können.

CSS in HTML einbinden

- Alle drei Formatierungsarten können auch gemeinsam auftreten
 - Regel: innere überschreibt bzw. ergänzt äußere Formatdefinition

```
<html>
```

```
<head>
```

```
<title>Titel der Datei</title>
```

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

```
<style type="text/css">
```

```
<!--
```

```
...hier sind datei-spezifische Zentrale Formate...
```

```
-->
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<h1 style="[element-spezifische Formate]">Text</h1>
```

```
</body>
```

```
</html>
```

externe Formate: lokal nicht beschränkt

zentrale Formate: lokal nicht beschränkt

direkte Formate: lokal beschränkt

Genereller Formatdefinitionsaufbau

- Formatdefinitionen folgen alle demselben Grundmuster

- Formate für HTML-Elemente

- Aufbau: Liste von Selektoren { [Eigenschaft:Wert;]+}

[*:*]+ bedeutet: Klammerinhalt wird mindestens einmal wiederholt, linke und rechte Seite gemäß Vorgabe auswählen

- Beispiel:

```
body {font-family: Trebuchet MS, Helvetica, Verdana, Arial, sans-serif;}
p, div, table, li, form {font-size: 12pt;}
p {text-align: justify;}
h1 {background-color: gray; color: white}
h2, h3 {color: #000099;}
a:link {color: #000099;}
a:hover {text-decoration: none; font-weight: 600;}
a:active {color: black;}
a:visited {color: #112222;}
```

Genereller Formatdefinitionsaufbau

■ Formate für verschachtelte HTML-Elemente

```
<html>
  <head>
    <title>Titel der Datei</title>
    <style type="text/css">
      <!--
      h1 { color:red; }
      h1 i { color:blue; font-style:normal; }
      -->
    </style>
  </head>
  <body>
    <h1>Wir lernen <i>Stylesheets</i></h1>
    <p>Wir lernen <i>Stylesheets</i></p>
  </body>
</html>
```

bedingte Formatierung:
falls *i* in *h1*, dann Formatierung wie angegeben,
ansonsten nach Voreinstellung

Genereller Formatdefinitionsaufbau

- Formate für verschachtelte HTML-Elemente

Wir lernen Stylesheets

Wir lernen *Stylesheets*

Genereller Formatdefinitionsaufbau

■ Pseudo-Formate

```
<style type="text/css">  
  <!--  
  a:link { color:#EE0000; text-decoration:none; font-weight:bold; }  
  a:visited { color:#EEAAAA; text-decoration:none; font-weight:bold; }  
  a:hover { color:#EE0000; text-decoration:none;  
           background-color:# FFFF99; font-weight:bold; }  
  a:active { color:#0000EE; background-color:#FFFF99;  
            font-weight: bold; }  
  body { font-family: serif; font-size:12pt; }  
  -->  
</style>
```

a: Anchor-Tag

:link = für Verweise zu noch nicht besuchten Seiten
:visited = für Verweise zu bereits besuchten Seiten
:hover = für Verweise, während der Anwender mit der Maus darüber fährt
:active = für gerade angeklickte Verweise
:focus = für Verweise, die den Fokus erhalten, z.B. durch "Durchsteppen" mit der Tabulatortaste

Weitere CSS-Eigenschaften

■ Schriftformatierung

■ Schriftart

`font-family`:*Schriftart1*,*Schriftart2*,...;

- Für *Schriftart1* den Namen gewünschten Schriftart notieren
- Für *Schriftart2* und weitere die Namen von Alternativschriftarten notieren

■ Schriftstil

`font-style`:*Wert*;

- *Wert* italic = Schriftstil kursiv
- *Wert* oblique = Schriftstil kursiv
- *Wert* normal = normaler Schriftstil

■ Schriftvariante

`font-variant`:*Wert*;

- *Wert* small-caps = Kapitälchen.
- *Wert* normal = normale Schriftvariante

Weitere CSS-Eigenschaften

■ Schriftformatierung

■ Schriftgröße

`font-size:Wert;`

- *Wert* als numerische Angabe wie 12pt, 14px oder 1.2em
- *Wert* xx-small = winzig bzw. x-small = sehr klein bzw. small = klein
- *Wert* medium = mittel
- *Wert* large = groß bzw. x-large = sehr groß bzw. xx-large = riesig
- *Wert* smaller = sichtbar kleiner als normal
- *Wert* larger = sichtbar größer als normal

■ Schriftgewicht

`font-weight:Wert;`

- *Wert* bold = fett
- *Wert* bolder = extra-fett
- *Wert* lighter = dünner
- *Wert* 100, 200, 300, 400, 500, 600, 700, 800, 900 = extra-dünn (100) bis extra-fett (900)
- *Wert* normal = normales Schriftgewicht

Weitere CSS-Eigenschaften

■ Schriftformatierung

■ Textdekoration

`text-decoration:Wert;`

- *Wert* underline = unterstrichen
- *Wert* overline = überstrichen
- *Wert* line-through = durchgestrichen
- *Wert* blink = blinkend
- *Wert* none = normal (keine Text-Dekoration)

■ Textfarbe

`color:Wert;`

- *Wert* als Farbangabe in RGB-Notation wie #FFCC99 oder rgb(96,255,0)
- *Wert* als Farbname aus CSS-Farben, z.B. red, blue, green

Weitere CSS-Eigenschaften

■ Hintergrundfarbe

`background-color: Wert;`

- *Wert* als Angabe wie #0000CC, fuchsia oder rgb(50%,60%,80%)

■ Hintergrundbild

`background-image: url(URI);`

- *URI* als Web-Adresse oder ein Ziel mit oder ohne Pfad angeben, wo sich die gewünschte Grafikdatei (GIF oder JPEG) befindet

■ Wiederholungs-Effekt

`background-image: url(URI); backgroundrepeat: Wert;`

- *Wert* repeat = wiederholen (Voreinstellung)
- *Wert* repeat-x = nur "eine Zeile lang" waagrecht wiederholen
- *Wert* repeat-y = nur "eine Spalte lang" senkrecht wiederholen
- *Wert* no-repeat = nicht wiederholen, nur als Einzelbild anzeigen

Weitere CSS-Eigenschaften

■ Wasserzeichen-Effekt

`background-image:url(URI); backgroundattachment:Wert;`

- *Wert* scroll = mitscrollen (Voreinstellung)
- *Wert* fixed = Hintergrundbild bleibt stehen

Weitere CSS-Eigenschaften

- viele weitere Formatierungsmöglichkeiten
 - siehe Dokumentation, z.B. selfhtml
 - Schriftformatierung
 - Ausrichtung/Absatzkontrolle
 - Außenrand/Abstand
 - Innenabstand
 - Rahmen
 - Hintergrundfarben und -bilder
 - Listenformatierung
 - Tabellenformatierung
 - Positionierung und Anzeige von Elementen
 - Anzeigefenster

HTML5

- HTML5 ist die fünfte Fassung der Hypertext Markup Language
- Das W3C hat am 28. Oktober 2014 die fertige HTML5-Spezifikation („W3C Recommendation“) vorgelegt
- Neue Funktionen wie Video, Audio, lokale Speicher und dynamische 2D- und 3D-Grafiken, die von HTML4 nicht direkt unterstützt werden
- Spezifische Multimedia-Elemente: Unterstützung nicht nur von Clients wie Desktop-PCs und Laptops, sondern jeder Art von mobilen Usern wie Smartphones, Tablets,...
- Bisher konsequenteste Separierung von Dokument-Beschreibung (HTML) und Layout (CSS)

HTML5

■ Neue Tags wie

- `<canvas>` (Container zur Erstellung von Grafiken mittels JavaScript)
- `<article>` (Eigenständiger logischer Teil innerhalb einer Webseite)
- `<main>` (Auszeichnung eines spezifisch inhaltstragenden Bereichs)
- `<mark>` (Auszeichnung eines markierten Texts auf HTML-Ebene)
- ...

■ Media-Elemente zur Erstellung multimedialer Layout-Inhalte

- `<audio>` (HTML-Container für webgerechte Audio-Ressourcen)
- `<embed>` (Standardkonforme Einbindung von Flashmovies)
- `<source>` (Standalone-Tag zur Quellenangabe)
- `<track>` (Standalone-Tag innerhalb video- oder audio-Elemente)
- `<video>` (HTML-Container für webgerechte Video-Ressourcen)
- ...

HTML5

- viele weitere Eigenschaften
 - umfangreiche Dokumentationen, z.B. auch in selfhtml