

1 Übung Web- und Multimedia-Engineering

Organisatorisches und Allgemeines

Aufgabenstellung Ü1 und thematische Einführung

1. Organisatorisches und Allgemeines
 - Website, Zugang und Kontakt
 - Ablauf und Abgabe
 - Überblick Aufgabenstellungen
 - Terminplan
 - Teambildung und Vorbereitung
2. Aufgabenstellung Ü1: HTML5 und CSS3
3. Grundlagen Webentwicklung
 - HTML
 - CSS
4. Hilfreiches, Tipps und Links

Teil 1

Organisatorisches und Allgemeines


Die Übung zur Vorlesung


- Allgemeine Informationen, Neuigkeiten, Materialien, Ergebnisse uvm. immer unter:

http://mt.inf.tu-dresden.de/studium/lehre/ws_14-15/wme_14-15/wme-ueb_14-15/

- Benutzername: mtstudent, Passwort: *** bitte erfragen ***

- Ansprechpartner und Betreuung


-  Dipl.-Ing. Ricardo Langner
Büro: INF 2050
Kontakt: ricardo.langner@tu-dresden.de

-  Tom Horak
Kontakt: tom.horak@tu-dresden.de

Die Übung zur Vorlesung

- Versteht sich als **Ergänzung** zur Vorlesung
 - Keinesfalls Ersatz, nur zur Übung gehen = „keine gute Idee“
 - Kann nur einige Teilen des Vorlesungsinhalts abdecken
- Verständnis durch **praktische** Aufgaben- und Problemstellungen
- Unterstützung und Hilfestellungen während des Selbststudiums
- Austausch durch Gruppen-/Teamarbeit
- Qualifikation bzw. Zulassung zur Klausur
 - Mindestens 50% der Übungspunkte


Ablauf der Übung

- 3 Übungstermine jeweils im E067 (INF):
 - Mo. 2DS + Mo. 3DS + Di. 2DS
 - Wichtig: Einschreibung per jExam 
 - Insgesamt noch 13 Termine im Semester
- Teamarbeit während des gesamten Semesters möglich
 - Teams bleiben über Aufgaben hinweg bestehen
 - Gleiche Bewertung für alle Teammitglieder
 - 2 Studenten je Team
 - Aktuell 112 Einschreibungen: 60 Teams
- Übungsbewertung basiert auf
 - Erfüllung konkreter Teilaufgaben/Anforderungen
 - Quellcode, Kommentare, Fehler, Aufbau und Struktur, Grafik und Design

Ablauf der Übung: Abgabe

- **Termin für Abgabe:** immer Montag der entsprechenden Abgabe-/Einführungswoche bis 9.00 Uhr
- Abgabe der Ergebnisse über zentrales Abgabeverzeichnis (SFTP/SSH) des ZIH
- Ergebnisabgabe umfasst
 - Immer alle Quell-Dokumente
 - Eine Readme mit Teamnummer und Namen
 - Denken Sie an Materialien, wie z. B. Bilder
 - Weitere Details siehe Aufgabenstellung
- **Nachreichungen werden nicht akzeptiert!**
 - Server sperrt Zugang automatisch
 - Abgabe verpassen = „keine gute Idee“, es gibt nur 4 Aufgaben

Ablauf der Übung: Abgabe

- Zentrales Abgabeverzeichnis auf ZIH-Server
 - Zugriff erfolgt i.d.R. über SFTP/SSH/SCP mit ZIH-Login
 - Servername: `ganymed.inf.tu-dresden.de` Port: 22
 - Verzeichnis: `/zbv/mt/wme/`  Wichtig: **NICHT** `/zbv/WME`
 - Server speichert Informationen über Zugriffe und Änderungen im Verzeichnis `/zbv/mt/wme/`
 - Je Aufgabe bzw. Themenkomplex ein separates Verzeichnis
 - Ü1: `/zbv/mt/wme/ueb1_HTML-CSS/`
 - Ü2: `/zbv/mt/wme/ueb2_Javascript/`
 - Ü3: `/zbv/mt/wme/ueb3_PHP-XML/`
 - Ü4: `/zbv/mt/wme/ueb4_Android/`

Ablauf der Übung: Abgabe

- Vorgehensweise je Aufgabe bzw. Themenkomplex
 - *Regel 1:* Jedes Team erstellt ein separates Verzeichnis im Ordner der entsprechenden Aufgabe; bspw. für Team 13:
`/zbv/mt/wme/ueb1_HTML-CSS/Team_13/` bzw. `/zbv/mt/wme/ueb3_PHP-XML/Team_13/`
 - *Regel 2:* Jedes Team passt die Zugriffsrechte des eben erstellten Teamverzeichnis an, um Missbrauch/Betrug vorzubeugen
 - Automatisch: Eigentümer ist Ersteller (ZIH-Account), Gruppe ist `mt_wme`
 - Korrekte Zugriffsrechte: `750` bzw. `rwxr-x---`
 - Eigentümer: R W X; Gruppe: RX
 - *Regel 3:* Jedes Team erstellt ein **ZIP**-Archiv (**kein RAR**) mit Lösung und speichert es im Teamverzeichnis
 - Auf Benennung achten: `Ueb<NR>_Team<T_NR>.zip`
 - `<NR>` Nummer der Übung, 1-2-3-4
 - `<T_NR>` Teamnummer, 1-2-3-n

Ablauf der Übung: Abgabe


- Bewertung der Abgabe mit **0 Punkten**, wenn
 - Keine **fristgerechte** Abgabe
 - Abgabe nicht ausführbar (z.B. defektes ZIP-Archiv) oder nicht zu öffnen (z.B. falsche Zugriffsrechte)
 - Gravierender Verstoß gg. Regel 1 oder Regel 3
 - **Ergebnis mit einem Anderen identisch!**
- **Keine Ausnahmen!**

- Finales Beispiel für Team 13:
 - Je Aufgabe bzw. Themenkomplex ein ZIP-Archiv
 - Ü1: /zbv/mt/wme/ueb1_HTML-CSS/Team_13/Ueb1_Team13.zip
 - Ü2: /zbv/mt/wme/ueb2_Javascript/Team_13/Ueb2_Team13.zip
 - Ü3: /zbv/mt/wme/ueb3_PHP-XML/Team_13/Ueb3_Team13.zip
 - Ü4: /zbv/mt/wme/ueb4_Android/Team_13/Ueb4_Team13.zip

Ablauf der Übung: Abgabe

- ZIP-Archive umfassen immer alle Source-Dateien
 - Quellcode (HTML, CSS, JavaScript, PHP, XML, Java, ...)
 - Weitere zur Kontrolle notwendige Materialien (Assets) wie zum Beispiel Bilder
 - **README** mit Team, Namen der Mitglieder, ggf. Hinweise und Anmerkungen an Betreuer
- **Nachreichungen werden nicht akzeptiert!**
 - Server sperrt Zugang automatisch
 - Abgabe verpassen = „keine gute Idee“, es gibt nur 4 Aufgaben

Ablauf der Übung: Präsentation der Ergebnisse

- **Ergebnispräsentation:** immer Montag vor neuer Aufgabeneinführung
 - Ca. 5-6 Kurzpräsentationen ohne Folien
 - Jedes Team mindestens 1x im Semester
 - Anmeldung E-Mail an **beide** Betreuer 
ricardo.langner@tu-dresden.de und tom.horak@tu-dresden.de

Ablauf der Übung: Themenkomplexe

- 4 Aufgabenstellungen

- Jeweils ein Themenkomplex
- Jeweils unterschiedlich lange Bearbeitungszeit
- Gemeinsames Thema über alle Übungen: inf_box
- **Ü1:** Grundlagen client-seitige Technologien: HTML5 und CSS3
 - Grundgerüst für eine Website als Interface für inf_box
- **Ü2:** Dynamische Webseiten: JavaScript, Ajax und jQuery
 - Manipulation, Animationen und dynamische Inhalte für die Website
- **Ü3:** Grundlagen server-seitige Technologien: XML und PHP
 - XML-Schema, XSLT, Grundlagen PHP einer inf_box Serverkomponente
- **Ü4:** Mobile App Development: Android SDK
 - Einfache inf_box Android-Client-App



Woche	Datum	Übungsthema /-inhalt
KW 42	13./14.10.	keine Übung
KW 43	20./21.10.	Einführung Aufgabenstellung Ü1: HTML5 und CSS3 → Materialien für Übungsaufgabe 1: HTML5 und CSS3 → Abgabe / Einreichung von Ergebnissen
KW 44	27./28.10.	Konsultation
KW 45	03.11. 9.00 Uhr	Abgabe Ü1 HTML5 und CSS3
KW 45	03./04.11.	Einführung Aufgabenstellung Ü2: JavaScript, Ajax, JSON und jQuery → Materialien für Übungsaufgabe 2
KW 46	10./11.11.	Konsultation
KW 47	17./18.11.	keine Übung
KW 48	24.11. 9.00 Uhr	Abgabe Ü2 JavaScript, Ajax, JSON und jQuery
KW 48	24./25.11.	Einführung Aufgabenstellung Ü3: XML und PHP → Materialien für Übungsaufgabe 3
KW 49	01./02.12.	Konsultation
KW 50	08./09.12.	Konsultation
KW 51	15.12. 9:00 Uhr	Abgabe Ü3 PHP und XML
KW 51	15./16.12.	Einführung Aufgabenstellung Ü4: Mobile App Development → Materialien für Übungsaufgabe 4 Version 2 → Schnittstellenbeschreibung (API)
KW 52	22./23.12.	keine Übung
KW 1	29./30.12.	keine Übung
KW 2	06./07.01.	Android Netzwerkkommunikation und Einführung inf_box_lib
KW 3	12./13.01.	Konsultation
KW 4	19.01. 9:00 Uhr	Abgabe Ü4 Mobile App Development
KW 4	19./20.01.	keine Übung
KW 5	26./27.01.	Anschluss, Feedback und Fragen
KW 6	02./03.02.	keine Übung
Woche	Datum	Übungsthema /-inhalt

Teambildung

(je 2 Studenten)

Notieren Sie ihre Teamnummer 

Vorbereitung: Wichtige Schritte

1. Eintrag in Mailingliste

- Regel für E-Mail: *@mail.zih.tu-dresden.de notwendig (ZIH-E-Mail-Adresse)
- Sie werden automatisch eingetragen!

Abonnieren von Mt-wme

Abonnieren Sie Mt-wme, indem Sie das folgende Formular ausfüllen:

In Kürze erhalten Sie eine Bestätigungs-e-Mail, um sicherzustellen, dass es wirklich Sie sind, der abonnieren möchte. Administrator der Liste eingesehen werden kann.

Ihre e-Mailadresse:

Ihr Name (optional):

Sie können weiter unten ein Passwort eingeben. Dieses Passwort bietet nur eine geringe Sicherheit, sollte aber verhindern, dass andere **Verwenden Sie kein wertvolles Passwort**, da es ab und zu im Klartext an Sie geschickt wird!

Wenn Sie kein Passwort eingeben, wird für Sie ein Zufallspasswort generiert und Ihnen zugeschickt, sobald Sie Ihr Abonnement bestätigen. Passwort jederzeit per Email zuschicken lassen, wenn Sie weiter unten die Seite zum ändern Ihrer persönlichen Einstellungen aufrufen

Wählen Sie ein Passwort:

Erneute Eingabe zur Bestätigung:

Welche Sprache bevorzugen Sie zur Benutzerführung?


2. Codewort zur Anzeige von Ergebnissen

- Ergebnisse auf Website nur im geschützten Bereich (mtstudent)
- Zusätzlich: Ergebnisse nie mit Namen verknüpft, nur Codewort
- Ohne Codewort keine Ergebnisse im Web
- Beispiele:
 - **Gut:** Mondstern_453, \$1_postcard, sun-shiny
 - **Schlecht:** Melanie_87, Max_DD, Donald_P
- Verknüpfung Name und Codewort nur Betreuern bekannt

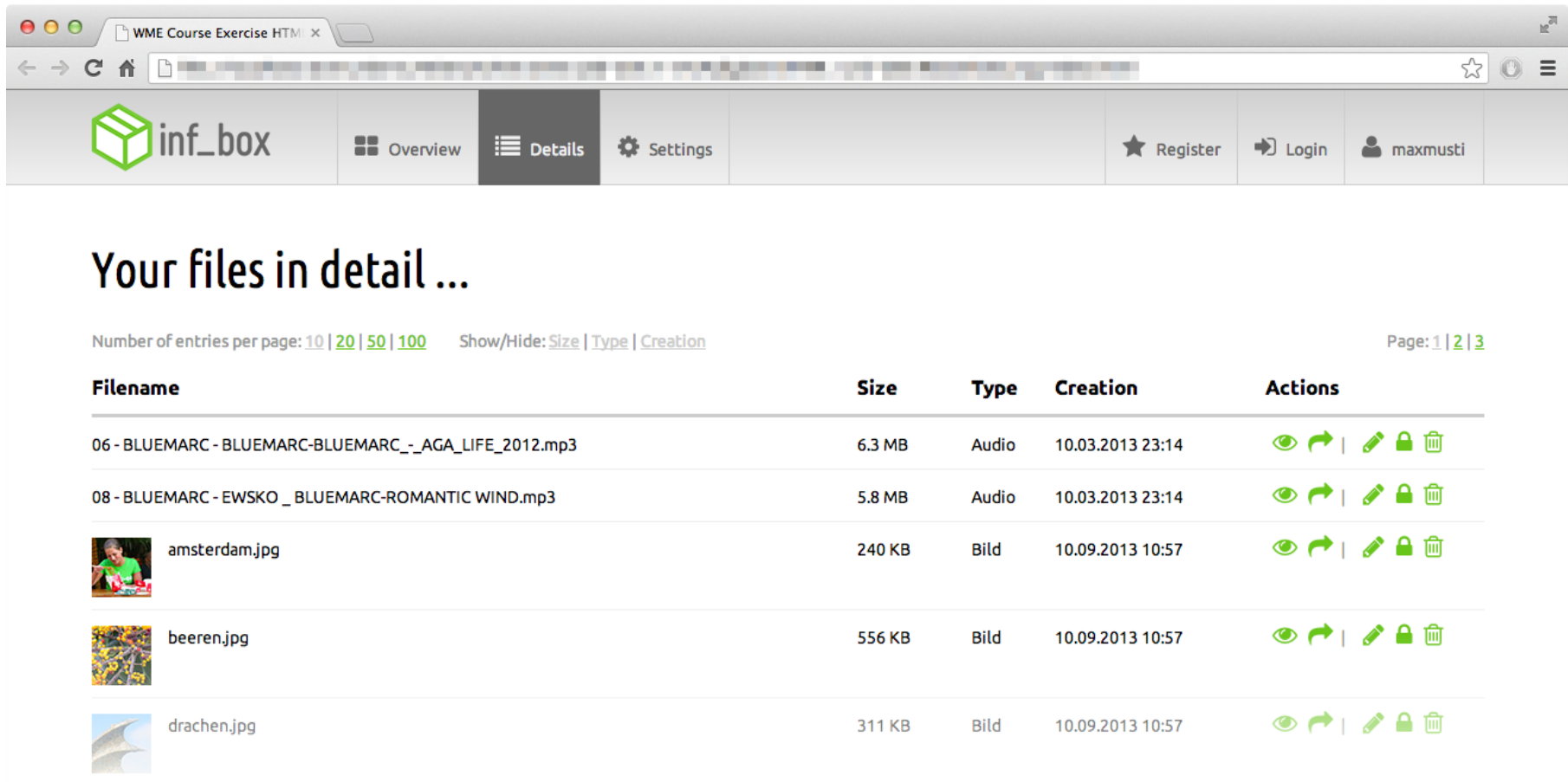


Teil 2

Aufgabenstellung Ü1: HTML5 und CSS3

- Einfaches Webinterface zur Anzeige der Inhalte
 - Navigationsleiste (Menü im Header)
 - Formular/Dialog für Registrierung, Anmeldung und Abmeldung
 - Listenansicht der Inhalte: Vorschaubild, Metadaten, Aktionen
 - Anzeigeeoptionen in Kontroll-Leiste
 - Fußbereich (Footer) mit Copyright und Namen
- Grundgerüst: Webseite ohne Funktionalität
- Statische Inhalte (1:1 übertragen)
- Testen: Chrome (aktuelle Version)
- Abgabe: Montag, **03.11.2014 bis 9.00 Uhr** 




- Erstellen (Rekonstruieren) Sie das folgende Webinterface
 - Screenshots der Webseite
 - Live-Demo



The screenshot shows a web browser window with the address bar displaying "WME Course Exercise HTML". The page features a navigation bar with the "inf_box" logo, tabs for "Overview", "Details" (selected), and "Settings", and user controls for "Register", "Login", and a user profile "maxmusti".

Your files in detail ...

Number of entries per page: [10](#) | [20](#) | [50](#) | [100](#) Show/Hide: [Size](#) | [Type](#) | [Creation](#) Page: [1](#) | [2](#) | [3](#)

Filename	Size	Type	Creation	Actions
06 - BLUEMARC - BLUEMARC-BUEMARC_-_AGA_LIFE_2012.mp3	6.3 MB	Audio	10.03.2013 23:14	
08 - BLUEMARC - EWSKO _ BLUEMARC-ROMANTIC WIND.mp3	5.8 MB	Audio	10.03.2013 23:14	
 amsterdam.jpg	240 KB	Bild	10.09.2013 10:57	
 beeren.jpg	556 KB	Bild	10.09.2013 10:57	
 drachen.jpg	311 KB	Bild	10.09.2013 10:57	

- Allgemeine Kriterien und Verwendung
 - ☐ Nur HTML und CSS, kein JavaScript, keine Frameworks, o. ä.
 - ☐ Dateikodierung (Encoding): **UTF-8** und *Unix-LF (Zeilenende)*
 - ☐ Valides HTML5 und CSS3
 - ☐ Notwendige Header-Angaben: title, description, author, keywords
 - ☐ CSS-Reset einbinden: [\[http://html5doctor.com/html-5-reset-stylesheet/\]](http://html5doctor.com/html-5-reset-stylesheet/)
 - ☐ Webfonts (Ubuntu, Ubuntu Condensed) und Icon-Font einbinden und verwenden
 - ☐ HTML5-Element für Datumsauswahl in Registrierungs-Formular
 - ☐ HTML5-Elemente für Header, Footer, Nav
 - ☐ Liste (Unordered List, ul) für Navigation
 - ☐ Tabellenelemente thead und tbody einsetzen
 - ☐ Fußzeile (Footer): Namen und Teamnummer

■ Design

- ☐ Basisschrift: "Ubuntu" [\[http://www.google.com/fonts#QuickUsePlace:quickUse/Family:Ubuntu\]](http://www.google.com/fonts#QuickUsePlace:quickUse/Family:Ubuntu)
 - Basisschriftgröße: 14px, Zeilenabstand: 145%, Style: Normal 400
- ☐ Iconschrift: "Font Awesome" [\[http://fontawesome.github.io/Font-Awesome/\]](http://fontawesome.github.io/Font-Awesome/)
- ☐ Überschrift: "Ubuntu Condensed" [\[http://www.google.com/fonts#QuickUsePlace:quickUse/Family:Ubuntu+Condensed\]](http://www.google.com/fonts#QuickUsePlace:quickUse/Family:Ubuntu+Condensed)
 - Schriftgröße: 3em, Zeilenabstand: 140%, Style: Normal 400
- ☐ Container-Breite (Wrapper): 1170px, links-rechts-zentriert
- ☐ Farb-Hover für Standard-Links (siehe Screenshots)
- ☐ Hover-Effekt für Logo
- ☐ Farbgradient im Header (Hinweis: geht auch ohne Bild)
- ☐ Formulare für Registrierung, Anmeldung und Account/Abmeldung als CSS-Dropdown (Popup)
- ☐ CSS-Transition für Hintergrundfarbe im Header (siehe Live-Demo)
- ☐ Vorschaubilder aus Screenshots extrahieren

Teil 3

Grundlagen Webentwicklung: Eine thematische Einführung



■ HTML5 – Grundstruktur

```
– <!DOCTYPE html>
  <html>
    <head>
      <!-- ... -->
    </head>
    <body>
      <!-- ... -->
    </body>
  </html>
```

- **head** enthält Seitentitel, CSS- und JS-Referenzen, etc.
- **body** enthält sichtbare Seitenelemente: Überschriften, Texte, Bilder, Tabellen, etc.



■ HTML5 – Basics

- `<h1>`Überschrift ersten Grades`</h1>`
- ``
- `<p>`Ich bin ein Absatz.`</p>`
- Bilder, Skripte etc. können **absolut** oder **relativ** adressiert werden
 - Start ohne Slash, z.B. „`bilder/logo.jpg`“ ist relativ zum aktuellen Verzeichnis
 - Start mit Slash, z.B. „`/css/layout.css`“ ist relativ zum Wurzelordner des Servers
 - Start mit `http://`, z.B. „`http://mt.inf.tu-dresden.de/lehre/wme`“ ist absolut
- **Semantische Tags** in HTML5 für SEO
 - Struktur: `<section>`, `<article>`, `<nav>`, ...
 - Textsemantik: `<time>`, `<meter>`, `<progress>`, ...



■ HTML5 – Strukturtags

- Statt wie bisher mit <div>-Tags werden Dokumente nun mittels neuer Tags strukturiert → verbessert die Maschinenlesbarkeit und macht den Code übersichtlicher

– `<!DOCTYPE html>`

`<html>`

`<head> ... </head>`

`<body>`

`<header><h1>HTML5 Demo</h1>`

`<nav> ... </nav>`

`</header>`

`<article>`

`<h2>Semantische Tags</h2>`

`<section> ... </section>`

`</article>`

`<footer> ... </footer>`

`</body>`

`</html>`

Kopfbereich eines
Dokumenteninhaltes

Navigationselemente

Textuell relevante
Inhalte

Untergliederung von
Textinhalten

Fußbereich eines
Dokumenteninhaltes



■ Cascading Style Sheets (CSS)

- Regeln zur Darstellung eines HTML-Dokuments
- Farben, Rahmen, Abstände, Schriften, ...
- bestehend aus
 - Selektor
 - Eigenschaft-Wert-Paaren
- Syntax
 - `Selektor {Eigenschaft:Wert;Eigenschaft:Wert;...}`
 - z.B.: `p{font-family:"Times New Roman";font-size:20px;}`
passt Schriftart und -größe aller Absätze an
- *class* und *id* um HTML-Elementen Eigenschaften zuzuordnen
 - class als Selektor für Vielzahl von Elementen
 - id für einzelnes selektieren von Elementen

■ Cascading Style Sheets (CSS)



– Einbinden von CSS

■ **direkt im Quellcode**

```
<h1 style=„color:red;“>Webdesign für Anfänger</h1>
```

■ **am Anfang der HTML-Datei**

```
<style>
```

```
    p.subtitle    {font-size:12pt;color:#0000FF;}
```

```
    a:hover       {text-decoration:none;}
```

```
</style>
```

■ **als externe Datei**

```
<link rel=„stylesheet“ type=„text/css“  
href=„/css/layout.css“>
```



■ Cascading Style Sheets (CSS)

- **Selektor** gibt zu formatierende Elemente an:
 - Global: * (alle Elemente des Dokuments)
 - Elementtyp: `h1` (alle Überschriften 1. Grades)
 - Klasse: `.subtitle` (alle Elemente mit class-Attribut „subtitle“)
 - ID: `#navigation` (Element mit der ID „navigation“)
 - Pseudo-Klasse: `a:hover` (beim Überfahren des Elements mit der Maus)
 - Liste: `h1, img, p` (alle 1. Überschriften, Bilder und Absätze)
 - Unterelemente: `p a` (alle Links innerhalb von Absätzen)
- nahezu beliebig kombinierbar
- spezifischere Regeln überschreiben allgemeinere
 - Klasse ist spezifischer als Element
 - ID ist spezifischer als Klasse
 - betrifft jedoch nur **widersprüchliche** Attribut-Wert-Paare!



■ CSS3 – Selektoren

- CSS3 definiert neue oder erweitert Selektortypen: Attribut-Selektoren, Pseudo-Klassen, Kombinatoren
- **Attribut-Selektoren:** erlauben Selektion von Elementen unter Prämissen der id-Benennung

<code><div id="nav-primary"/></code>	<code>div[id^="nav"] { background:#ff0; }</code>	Alle Elemente die mit „nav“ beginnen
<code><div id=„second-nav"/></code>	<code>div[id\$="nav"] { background:#ff0; }</code>	Alle Elemente die mit „nav“ enden
<code><div id=„supernavone"/></code>	<code>div[id*="nav"] { background:#ff0; }</code>	Alle Elemente die „nav“ enthalten

- **Strukturelle Pseudo-Klassen:** erlaubt die Selektion von Elementen abhängig von ihrer Position im DOM-Baum:
 - root, nth-child, first-child, last-child, only-child, only-of-type, empty, ...

■ CSS3 – Selektoren



- `nth-child()` – Funktion kann nach vielfältigen Formeln Kindelemente selektieren

<code>nth-child(3) { color:#f00; }</code>	Selektiert das dritte Kindelement
<code>nth-child(odd) { color:#f00; }</code>	Selektiert alle ungeraden Kindelemente (even für gerade)
<code>nth-child(2n+7) {color:#f00; }</code>	Selektiert alle Elemente die der Formel $an+b$ entsprechen (a - Zyklusgröße, n - Zählvariable (ab 0), b - Offset)

- **Beispiel:** abwechselnde Farben der Tabelle hier?
- `tr:nth-child(2n) { background-color: #D0D8E8; }`
`tr:nth-child(2n+1) { background-color: #E9EDF4; }`

■ CSS3 – Kombination von Selektoren [9]



X + Y	Selektiert <u>nur</u> das Y-Element nach einem X	ul + p { color: blue; }
X > Y	Selektiert <u>nur</u> die <u>direkten</u> Kinder Y von X	div#container > ul { border: 1px solid blue; }
X ~ Y	Selektiert <u>alle</u> Y nach einem X (weniger strikt als X + Y)	ul ~ p { color: blue; }

– **Beispiel:**

```
<div id="container">
  <ul>
    <li> List Item
      <ul>
        <li> Child </li>
      </ul>
    </li>
    <li> List Item </li>
  </ul>
  <p> Paragraph </p>
  <p> Paragraph </p>
</div>
```


■ Cascading Style Sheets (CSS)

– Stil gibt an, wie Elemente formatiert werden sollen

■ Rahmen und Abstände: **CSS Box Model**

■ Positionierung: immer relativ zum Eltern-element

– absolute, relative, inline, fixed

■ Farben

– als RGB-Hex-String: #FFFFFF

– als RGB-Tupel: rgb(0,0,0)

– Weitere: rgba, hsl, hsla

■ Transparenzen: opacity:0.2;

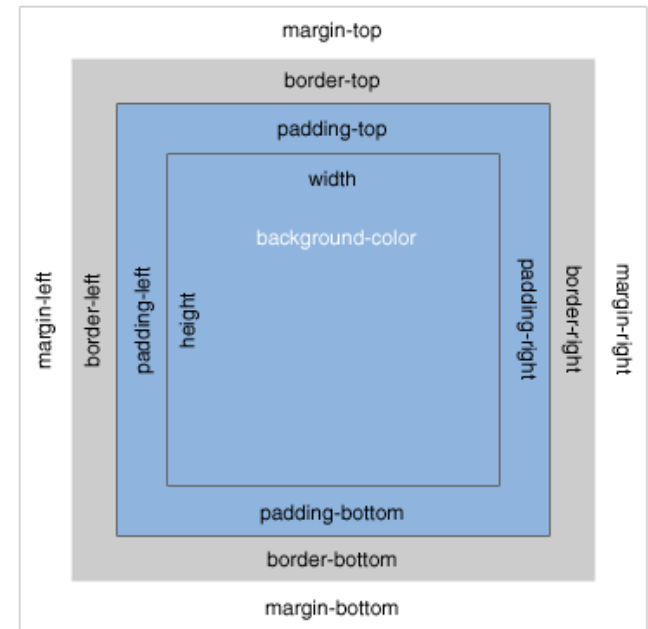
■ Schrift

– Größe: font-size:12pt;

– Lage: font-style:italic;

– Stärke: font-weight:bold;

– Extras: text-decoration:underline;

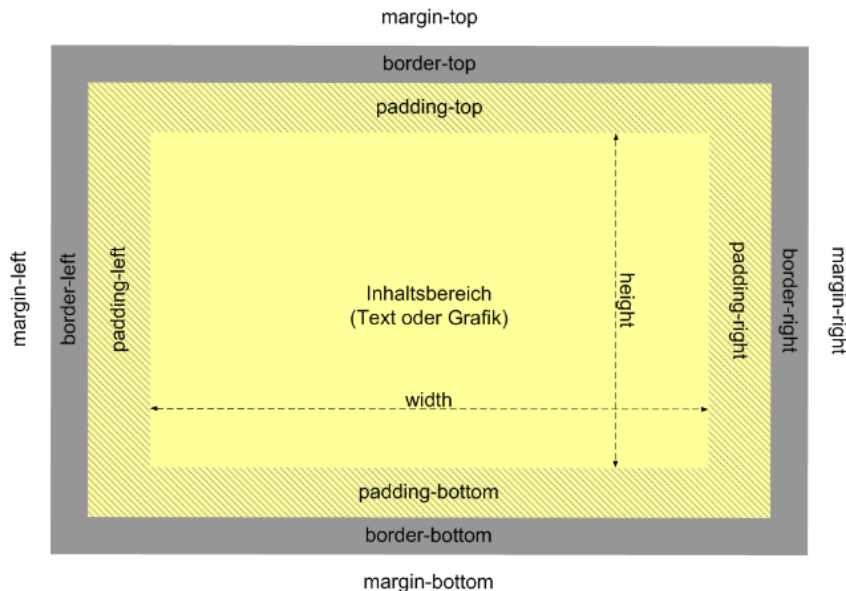


Quelle: [<http://www.mandalatv.net/itp/drivebys/css/>]

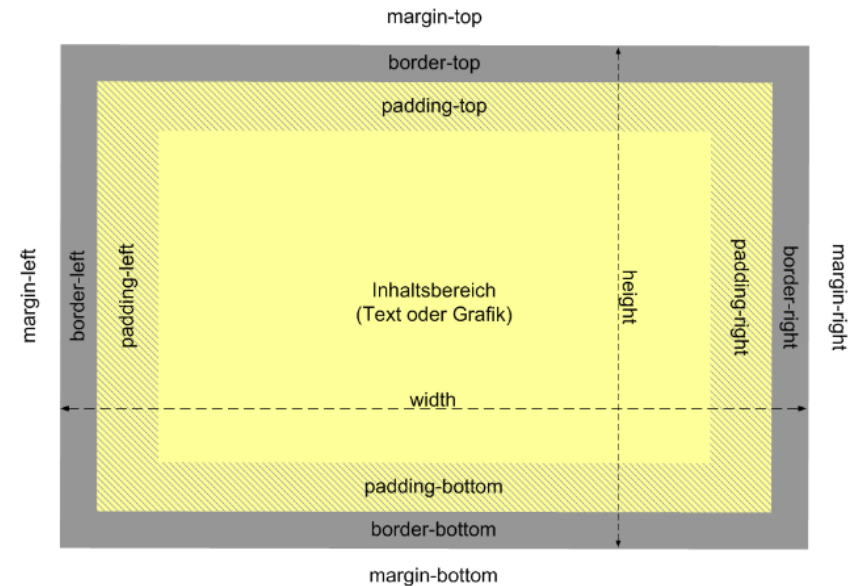


■ CSS3 – Neues Box-Modell [10]

- Bisher: {box-sizing: content-box} → Höhe und Breite definieren Inhaltsbereich; Padding, Border, Margin werden hinzugefügt
- Neu: {box-sizing: border-box} → alle Außenelemente sind in Höhe und Breite enthalten



Klassisches Box-Modell für Block-Elemente – { box-sizing: content-box } (Standard-Einstellung) – <http://title-boxes.de/>



CSS3 Box-Modell für Block-Elemente – { box-sizing: border-box } – <http://title-boxes.de/>



■ Cascading Style Sheets (CSS)

– Media Types geben Zielmedien für Regeln an

- @media print, handheld, projection, tv, speech, braille

– z.B.:

```
@media handheld, print {  
    p.subtitle { color: #000000; }  
}
```

– CSS 3 Media Queries als Erweiterung

- ermöglichen Adaptivität der Darstellung an Bildschirmauflösung etc.
- UND-verknüpft
- (max-/min-)width, color, aspect-ratio, ...

– z.B.:

```
@media screen and (max-width: 500px) {  
    p.subtitle { font-size: 14pt; }  
}
```

■ Browser-Support

- Nicht alle HTML5 & CSS3 Elemente werden von allen Browsern unterstützt

- Empfehlung: [\[http://caniuse.com/\]](http://caniuse.com/)



IE	Firefox	Chrome	Safari	Opera
8: 9%	30: 81%	35: 83%	5.1: 58%	
9: 24%	31: 81%	36: 83%	6.1: 65%	12.1: 71%
10: 53%	32: 81%	37: 86%	7: 65%	24: 85%
11: 63%	33: 81%	38: 87%	7.1: 69%	25: 86%
	34: 81%	39: 87%	8: 69%	26: 86%
	35: 83%	40: 87%		27: 86%
	36: 83%	41: 87%		

Teil 4

Hilfreiches, Tipps und Links

Hilfreiches, Tipps und Links

- Klassisches Thema für Selbststudium: „Step-by-Step learning“
- Erstellung eines Farb- und Designschemas mittels Paper-Prototyping [1, 2, 3]
- Hilfreich bei der Gestaltung sind Rastervorlagen [4]
- Umsetzung des Entwurfs in einem Editor der Wahl
 - Liste von HTML-Editoren [11] und Universelle Texteditoren [12]
- Validierung des kompletten Markups
 - HTML5 [5] und CSS3 [6]

Hilfreiches, Tipps und Links

- Responsive Webdesign [7]
 - Optimale Ansicht einer Website auf allen Arten von Geräten
 - Früher drei Versionen vs. Heute eine flexible Version
 - Einfache Navigation, Informationsfindung mit möglichst wenig Veränderungen an der Seite selbst (Resizing, Panning, Scrolling)
 - Hauptbestandteile: flexible Schrift, flexibles Grid, Media Queries



Quelle:
http://en.wikipedia.org/wiki/File:Boston_Globe_responsive_website.jpg

Quellen, Referenzen

- [1] „What is Paper Prototyping“, [\[http://www.paperprototyping.com/\]](http://www.paperprototyping.com/)
- [2] „Paper Prototyping“, [\[http://de.wikipedia.org/wiki/Paper_Prototyping\]](http://de.wikipedia.org/wiki/Paper_Prototyping)
- [3] „Six Signs That You Should Use Paper Prototyping“, [\[http://today.java.net/pub/a/today/2003/12/23/sixSigns.html\]](http://today.java.net/pub/a/today/2003/12/23/sixSigns.html)
- [4] „Free Printable Sketching, Wireframing and Note-Taking PDF Templates“, [\[http://www.smashingmagazine.com/2010/03/29/free-printable-sketching-wireframing-and-note-taking-pdf-templates/\]](http://www.smashingmagazine.com/2010/03/29/free-printable-sketching-wireframing-and-note-taking-pdf-templates/)
- [5] „W3C Markup Validation Service“, [\[http://validator.w3.org/\]](http://validator.w3.org/)
- [6] „W3C CSS Validation Service“, [\[http://jigsaw.w3.org/css-validator/\]](http://jigsaw.w3.org/css-validator/)
- [7] „Responsive Webdesign mit HTML5 und CSS3 – Grundlagen“, [\[http://t3n.de/news/responsive-webdesign-html5-css3-grundlagen-335305/\]](http://t3n.de/news/responsive-webdesign-html5-css3-grundlagen-335305/)
- [8] „Media Queries: Very Good Examples“, [\[http://mediaqueri.es/\]](http://mediaqueri.es/)
- [9] „The 30 CSS Selectors you Must Memorize“, [\[http://net.tutsplus.com/tutorials/html-css-techniques/the-30-css-selectors-you-must-memorize/\]](http://net.tutsplus.com/tutorials/html-css-techniques/the-30-css-selectors-you-must-memorize/)
- [10] CSS3: Das etwas andere Box-Modell mit {box-sizing:border-box}, [\[http://www.drweb.de/magazin/css3-das-etwas-andere-box-modell-mit-box-sizingborder-box-39078/\]](http://www.drweb.de/magazin/css3-das-etwas-andere-box-modell-mit-box-sizingborder-box-39078/)

Quellen, Referenzen (2)

- [11] Liste von HTML-Editoren, [http://de.wikipedia.org/wiki/Liste_von_HTML-Editoren]
- [12] Liste von Texteditoren, [http://de.wikipedia.org/wiki/Liste_von_Texteditoren]

Fragen?



Interactive Media Lab Dresden
Professur für Multimedia-Technologie

Kontakt:

Ricardo Langner (ricardo.langner@tu-dresden.de)

Tom Horak (tom.horak@tu-dresden.de)

Changelog

Datum / Zeit	Beschreibung
2014-10-20 9:00	▪ Initiale Downloadversion
2014-10-26 13:30	▪ Ergänzung Informationen zur Abgabe bzw. Einreichung von Ergebnissen