



- **Organisatorisches**
  - Ablauf der Übung
  - Themen
  - Abgabemodalitäten
  - Gruppenzuordnung
- **Aufgabenstellung**
  - Hinweise zur Umsetzung
- **Einstieg ins Thema**
  - XHTML
  - CSS
- **Links**

## ■ Terminplan

- **18.10.11 Übungseinführung 1A (Pflicht)**
- 25.10.11 Konsultation (Teilnahme freiwillig)
- 01.11.11 Konsultation (Teilnahme freiwillig)
- 08.11.11 Konsultation (Teilnahme freiwillig)
- **14.11.11 Abgabe der Ergebnisse (bis spätestens 13:00 Uhr)**
- 15.11.11 Übungseinführung 2A (Pflicht)
- ...
- 31.01.12 Abschlussveranstaltung und Präsentation (Pflicht)

## ■ Details siehe [Übungskalender](#)

## ■ Themenkomplex 1

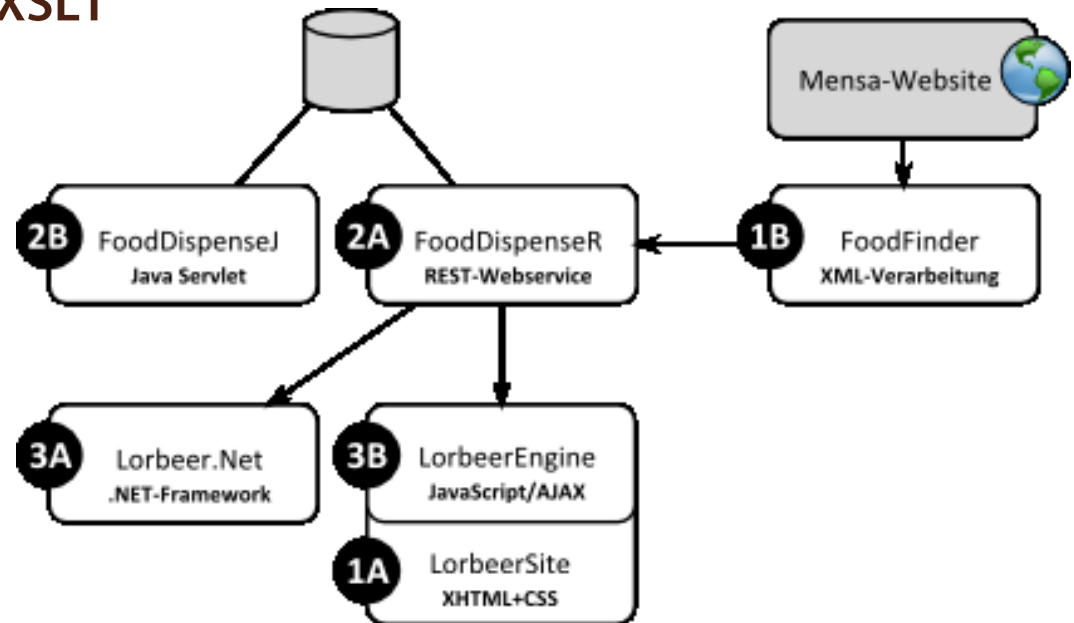
- 1A) XHTML und CSS
- 1B) XML-Verarbeitung und XSLT

## ■ Themenkomplex 2

- 2A) RESTful Webservice
- 2B) Java Servlets

## ■ Themenkomplex 3

- 3A) .NET/Silverlight
- 3B) JavaScript und AJAX



## ■ Informationen zu den Übungen

[http://www.mmt.inf.tu-dresden.de/Lehre/Wintersemester\\_11\\_12/WME/Uebung/index.xhtml](http://www.mmt.inf.tu-dresden.de/Lehre/Wintersemester_11_12/WME/Uebung/index.xhtml)

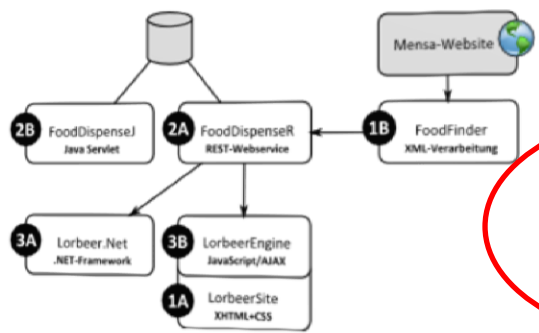
Übung zur Vorlesung  
**Web- und Multimedia-Engineering**

[Dipl.-Medieninf. Matthias Niederhausen](#)  
 siehe unten; zuerst in der Woche vom 17.10.2011  
 INF E069  
 0/2/0/0 SWS

**Kurzbeschreibung**

Zur Ergänzung und zum Verständnis des Vorlesungsstoffes werden drei Themenkomplexe mit jeweils zwei Aufgabenstellungen angeboten, die sich alle mit speziellen Teilproblemen einer übergeordneten Beispielanwendung beschäftigen. Jeder Student bearbeitet je eine Aufgabe aus den Themenkomplexen 1, 2 und 3 in einer Übungsgruppe (2 Studenten). Bei jedem Themenkomplex hat man die Wahl zwischen Thema A oder B. Man kann A und B bei jedem Themenkomplex neu wählen, also 1A, 2B und 3A sind durchaus möglich. Für die Bearbeitung einer Aufgabe haben die Gruppen etwa vier Wochen Zeit. Danach werden die Lösungen durch die Gruppen präsentiert (ca. fünfminütiger Vortrag). Wenn eine Gruppe gerne eine Projektplattform nutzen möchte, ist z.B. [assembla](#) zu empfehlen.

Außer bei den Einführungsveranstaltungen wird die Übung in selbstständiger Arbeit abgehalten; der Tutor steht bei eventuellen Fragen zur Seite. Teilweise werden wiederkehrende Probleme auch für alle Übungsteilnehmer erklärt. Die Einschreibung erfolgt für einen der vier angebotenen Termine, da die Räume eben nur begrenzt Platz bieten. Im Normalfall erfolgt die Einschreibung sich also für einen Termin, der auch mit dem Wahlthema übereinstimmt. Im Ausnahmefall ist es auch möglich, zu den anderen Terminen die Übungen besuchen. Dort wird man aber primär mit den Inhalten des anderen Themas konfrontiert, der Tutor wird natürlich trotzdem versuchen zu helfen.



**Inhalt**

- [Kurzbeschreibung](#)
- [Aktuelles](#)
- [Übungszeiten und Tutoren](#)
- [Terminplan](#)
- [Einschreibung](#)
- [Materialien](#)
- [Abgabe](#)
- [Bewertung](#)

**Nachgefragt**

Noch keine Fragen vorhanden...

[Alle Fragen ansehen](#)

[Frage stellen](#)

Forum für Nachfragen

**Nachgefragt**  
 Noch keine Fragen vorhanden...  
[Alle Fragen ansehen](#)  
[Frage stellen](#)

- via Upload auf **sFTP** Server:
  - Servername: **serv9.inf.tu-dresden.de**
  - Port **22**
  - Verzeichnis: **/zbv/WME/**
  - Authentifizierung mit **ZIH-Login**
  
- Abgabe der Lösung
  - Verzeichnis anlegen
    - Ergebnisse\_1\_A/<Gruppe>\_<Matrikel1>\_<Matrikel2>
  - Dateien hochladen
    - keine Umlaute in Dateinamen!
  - Zugriffsrechte nicht verändern!

- Lösung der Aufgabe
  - **zwei Studenten** bilden ein Team
    - auf der Einschreibliste eintragen!
  - Durchschleifen wird nicht toleriert
  - Abgabetermin ist verbindlich
    - Abgabeverzeichnisse werden nach Ablauf der Bearbeitungszeit gesperrt
  - jede Gruppe hat eine **eigenständige** Lösung abzugeben
  - ausgewählte Ergebnisse werden am letzten Übungstermin (Anfang Februar) von den Gruppen präsentiert

## ■ Aufgabe 1A: Gestaltung des Mensa-Bewertungsportals

### „Lorbeerblatt“

- Das Portal Lorbeerblatt soll Studenten eine Plattform bieten, auf der sie sich über das aktuelle Mensaessen austauschen können. Die Benutzer sollen Kommentare, Bewertungen und eigene Bilder der verschiedenen Speisen einstellen können. Sie sollen nun die Oberfläche dieses neuen Portals entwerfen.

#### Mensa-Speiseplan vom Freitag, den 14. Oktober 2011

Neue Mensa	Infos	Preise
Schinkenmakkaroni mit fruchtiger Tomatensoße und geriebenem Käse (auch ohne Schinken möglich)		1,66 € / 3,16 €
Schweinesteak mit Kräuterbutter dazu Pommes frites oder Risoleekartoffeln und Salat		ausverkauft
Wok: Pizza mit Zucchini und Hirtenkäse ---AUCH als halbe Pizza: 1,62€/ 2,62€-- -		2,43 € / 3,43 €

Weitere Angebote und Informationen anzeigen ▾

Alte Mensa	Infos	Preise
Germknödel mit Heidelbeerfüllung, dazu Vanillesoße		1,70 € / 3,20 €
Hausgemachte Schweinsröllchen mit Schinkenfüllung, dazu Möhren-Fenchelgemüse und Petersilienkartoffeln	  	2,20 € / 3,70 €
Gebratene Spätzle mit Gemüsestreifen und Käsesoße, dazu Salat	 	1,70 € / 3,20 €

Weitere Angebote und Informationen anzeigen ▾

- Entwicklung eines **Design-Mockups** für Lorbeerblatt
  - angepasste Benutzeroberflächen für **Standard-PCs** und **mobile Endgeräte**
  - Funktionen
    - Gerichte des aktuellen Tages mit Beschreibung, Preis, Zutaten und Bild einsehen
      - Filteroptionen für Mensa und Zutaten (Fleisch, Knoblauch, Alkohol, ...) sowie Sortieroptionen
    - selbst fotografierte Bilder zu Speisen hochladen
    - Gerichte bewerten (Kategorien Portionsgröße/Würzigkeit/Optik)
    - Gerichte mit kurzem Text (bis 255 Zeichen) kommentieren
  - **Der beste Entwurf wird im weiteren Verlauf der Übung weiterverwendet!**

## ■ Randbedingungen

- **gültiges XHTML**: einfach zu prüfen mit <http://validator.w3.org>
- lauffähig in **Chrome** (v13), **Firefox** (v6) und **Internet Explorer** (v8)
- Füllen des Entwurfs mit **realen** Daten der Mensawebsite
  - <http://www.studentenwerk-dresden.de/mensen/speiseplan/>
- Einsatz von **CSS Media Types/Queries** zur Zuordnung von PC- und Mobilversion
- **Lizenzrechte** bei verwendeten Fremdinhalten beachten!

## ■ Abgabe

- Quelldateien, evtl. als ZIP-Datei

## ■ XHTML

### ■ Seitengerüst:

```
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    <!-- ... -->
  </head>
  <body>
    <!-- ... -->
  </body>
</html>
```

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

## ■ XHTML-Body

### ■ ein paar Basics:

```
<h1>Überschrift ersten Grades</h1>
<p>Das ist ein Absatz</p>

<p class="subtitle">Logo des MMT</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://www.mmt.inf.tu-dresden.de/Lehre/](#) ist absolut

### ■ **class** als universelles Attribut, um **CSS-Eigenschaften** zuzuordnen

- nützlich, um verschiedene Elemente gleichartig zu formatieren
- auch mehrere Klassen für ein Element möglich

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

## ■ Cascading Style Sheets (CSS)

### ■ **Selektor** gibt an, welche Elemente formatiert werden sollen:

- 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

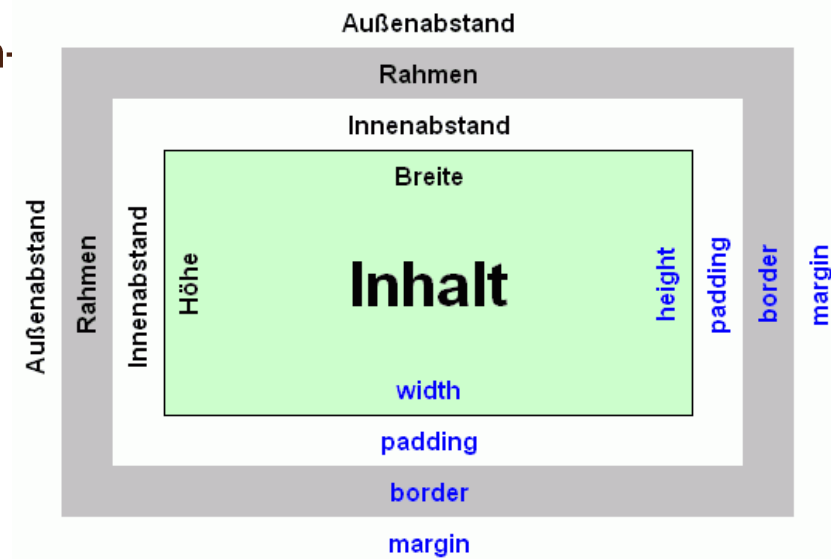
### ■ spezifischere Regeln überschreiben allgemeinere

- Klasse ist spezifischer als Element
- ID ist spezifischer als Klasse
- betrifft jedoch nur **widersprüchliche** Attribut-Wert-Paare!

## ■ 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(255,255,0)**
- Transparenzen: **opacity:0.2;**
- Schrift
  - Größe: **font-size:12pt;**
  - Lage: **font-style: italic;**
  - Stärke: **font-weight:bold;**
  - Extras: **text-decoration:underline;**



CSS Box Model, aus: SelfHTML

[http://de.selfhtml.org/css/formate/box\\_modell.htm](http://de.selfhtml.org/css/formate/box_modell.htm)

## ■ Cascading Style Sheets (CSS)

### ■ **Media Types** geben Zielmedien für Regeln an

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

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

### ■ **CSS 3 Media Queries** als Erweiterung (von IE8 nicht unterstützt)

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

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

## ■ XHTML-Head

### ■ Einbinden von CSS

#### – inline

```
<style>
  p.subtitle {font-size:12pt;color:#0000FF;}
  a:hover    {text-decoration:none}
</style>
```

#### – als Datei

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

### ■ Einbinden von JavaScript

#### – inline

```
<script type="text/javascript">
function alarm() {
  alert("Der Alarm wurde ausgelöst.");
}
</script>
```

#### – als Datei

```
<script type="text/javascript" src="/js/jquery-1.5.2.min.js"></script>
```

## ■ Editoren

### ■ Notepad++

- Texteditor mit Syntax-Highlighting
- klein und schnell
- gut geeignet für HTML und CSS

### ■ phase 5

- HTML-, CSS- und Script-Editor

### ■ Eclipse

- komplexe Entwicklungsumgebung, ursprünglich für Java
- vergleichsweise schwergewichtig
- evtl. zur Einarbeitung für spätere Aufgabenkomplexe

- **SELFHTML: ausführliches Nachschlagewerk für HTML, CSS und JavaScript**  
<http://de.selfhtml.org/>
- **w3schools: knappe Referenzen zu den meisten Web-Technologien**  
<http://www.w3schools.com/>
- **Adaptive Gestaltung mit CSS Media Queries**  
<http://www.alistapart.com/articles/responsive-web-design>
- **HTML 4.01 Specification – W3C Recommendation 24 December 1999**  
<http://www.w3.org/TR/html4/>
- **XHTML™ 1.0: The Extensible HyperText Markup Language - A Reformulation of HTML 4 in XML 1.0 - W3C Recommendation 26 January 2000**  
<http://www.w3.org/TR/xhtml1/>
- **Cascading Style Sheets, level 2 - CSS2 Specification – W3C Recommendation 12-May-1998**  
<http://www.w3.org/TR/REC-CSS2/>