

Multimedia im Netz
Online Multimedia
Winter semester 2015/16

Tutorial 13 – Minor Subject



Today's Agenda

- Announcements about the Exam
- Repetition
 - HTML5 + JavaScript
 - jQuery
 - AJAX
- Q&A

Announcements

- Remaining tutorial dates / topics
 - Only 1 tutorial left: **February 03!**
 - Very important: Prepare questions beforehand, send them via email!
- Repetition Sheet (assignment 13)
 - for your individual preparation
 - cannot be submitted to UniWorX
 - Try to complete the tasks on paper
 - Do not look up things on the Internet, while you do the tasks. Use your lecture/tutorial material and notes

Exam (1)

- The exam is open book
- Bring:
 - handwritten notes, printed slides, code, books (if necessary)
 - valid student card
 - valid personal ID card
- Leave at home:
 - any electronic device. That includes calculators of any kind, phones, computers, smart watches, etc.

Exam (2)

- You need to register on UniWorX to participate:
 - choose “Minor Subject (Nebenfach)”
 - Registration hard deadline: 08.02.2016 10:00
 - De-Registration deadline: 09.02.2016 10:00
- If you do not de-register and not show up, the attempt will be counted as “failed”.
- Date: **11.02.2016 10:00 – 12:00** (90 minutes writing time)
- Location: A240 Hauptgebäude, Geschwister-Scholl-Platz 1
- Retry exam probably early April, but we’d prefer if you passed the first exam 😊
- Grades will take a while. Don’t write hasty emails.

Repetition

JavaScript (1)

- JavaScript is a programming / scripting language
- Code is usually interpreted by the browser
- Code is embeddable into HTML:

```
<html>
<head>
  <title>Example</title>

  <script type="text/javascript">
    alert("Hello World!");
  </script>

</head>
<body></body>
</html>
```

JavaScript (2)

- As programming language, JavaScript naturally has a number of data- and control structures:
 - Variables
 - Arrays
 - Functions
 - For- loops / while-loops
 - If ... else
 - Etc.
- You should know that by know 😊

Breakout 1: Essential JavaScript

- Extend 01-loops.html
- Update the content of the #output <div>
- Take the beatles array and produce generate this output:

The Beatles are: John, Paul, George, Ringo

- Homework:
 - before the last entry, there should be an “and”.

The Beatles are: John, Paul, George, and Ringo

JavaScript: Event Listeners

- Quick Question: What does “DOM” stand for? What is this?
- Each element in the DOM of an HTML / XML file is accessible through JavaScript
- Events: JavaScript can handle numerous events, e.g. when the user types into a text field or submits a form
 - Mouse events (click, mousedown, ...)
 - Keyboard events (input, keydown, keypress, ...)
 -
- There are multiple ways to handle events in JS, e.g. with `addEventListener()`

Breakout 2: Interaction with JavaScript

- Extend 02-inputlistener.html
- Attach an appropriate event listener to the input field
- When the user types inside the input field, the output shows the exact content, only in **uppercase** letters:

`.toUpperCase()` is useful

I GUESS `.TOUPPERCASE()` IS USEFUL

- **Homework:**
 - Add a “save” button.
 - Upon clicking the button, the text is saved to a separate div and the field is emptied

HTML5

- The HTML5 standard brought many new features:
 - Form validation
 - New **attributes** for <input> (placeholder, required, pattern, type="email"...)
 - Constraint Validation API: allows you to customize user dialogs (not part of the tutorial)
 - New **elements**
 - <canvas></canvas>
 - <video></video>
 - <audio></audio>

Multimedia Elements

- The `<video>` and `<audio>` elements embed multimedia content into a web page
- Greatest advantage: Playback does not require plug-ins, e.g. Flash / QuickTime / VLC
- Shortcoming: Not all browsers support all file formats and encodings.
 - You can provide alternatives and the browser picks the one that it can play back.
 - Ultimately, you could fall back to a plug in.

Breakout 3: Video Element

- You want to embed a video into a web page
 - You have the video as movie.mp4 and movie.ogv
 - You want to use the `<video>` tag.
- How do you do that with HTML5?
Extend 03-video.html
- Homework:
 - Add a button that can play/pause the video

jQuery

jQuery



- JavaScript library
- Goals:
 - Easy DOM access / modification
 - Shorthand event handling
 - Simple animations
 - Convenient AJAX
 - Cross browser compatibility
- Included as a single JavaScript file
 - Either downloaded / locally
 - Or through CDN

jQuery: Basics (Reminder)

- Global jQuery function: `$ ()`
- Parameters:
 - any **CSS Selector** ("div", ".someClass", "#someID")
 - HTMLElements
 - Arrays
 - jQuery Objects
- Returns: jQuery-object that has many useful methods.

- Examples:
 - `$("#myDiv")`
 - `$(".div.container")`
 - `$("input[type='text']")`

- `$` is also an object that offers additional methods, e.g. `$.isArray(value, array)`

DOM Manipulations

- Access existing elements with selectors:

```
var output = $( '#output' );
```

- Create elements:

```
var heading = $( '<h2>' );
```

```
var paragraph = $( '<p>' );
```

- Modify elements:

```
heading.html( 'jQuery Example' );
```

```
paragraph.html( 'This is convenient' );
```

- Insert elements by appending them to another element:

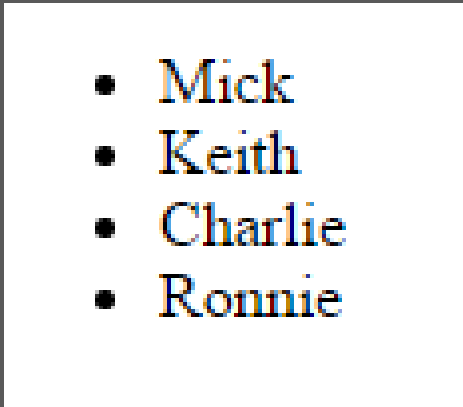
```
output.append( heading );
```

```
output.append( paragraph );
```

02-jq-create-append.html

Breakout 4: DOM Manipulation

- Extend 04-dom.html
- Use jQuery to:
 - Create four list items (``) with jQuery for Mick, Keith, Charlie and Ronnie
 - Create a list `` with variable name stones
 - Append the list items to the list
 - Append the list to the output `<div>`.
- Homework:
 - Increase font size of a list item, when the user clicks on it.
Use the `.css()` method

- 
- Mick
 - Keith
 - Charlie
 - Ronnie

AJAX

- Asynchronous Javascript and XML.
- Data-exchange without page refresh
- Usage:
 - Search engines
 - Web-apps (e.g. MS Office online)
 - Search-functionality in websites (e.g. Netflix)
- XML is now more and more replaced by JSON
 - Smaller file size
 - Easy usage in JavaScript

AJAX with jQuery

- Generic AJAX method: `$.ajax()`
- GET: `$.get()`
 - Parameters are attached to URL
 - Example:

```
$.get('03-thedoors.json', function (data) {  
    // do something with data.  
});
```
 - Use this to **retrieve** information from the server
- POST: `$.post()`
 - Parameters are transmitted in HTTP message body (less visible)
 - Use this to **modify** things on the server.

Breakout 5: Handling Data

- Extend 05-ajax.html
- Load the file 05-thedoors.json with an AJAX request.
- Create a heading containing the band name
- Create a list containing the band members
 - `$(...).each()` is helpful here.
- Test if your solution still works if you use another JSON file.
We provided 05-arcadefire.json
- Homework:
 - Exercise working with parameters
 - Exercise working with the `$.post()` method

The Doors

- Jim Morrison
- Ray Manzarek
- John Densmore
- Robby Krieger

What are your questions? Q & A Time!

Off Topic: Tutors wanted!

- If you enjoyed this course and are excited about the topic, become a tutor!
- Responsibilities:
 - run tutorials (at least one)
 - help with corrections
- Feel free to contact us for further details 😊
tobias.seitz@ifi.lmu.de



Thanks!

**Don't forget to prepare
questions until next time.**