Multimedia im Netz

Online Multimedia

Winter semester 2015/16

Tutorial 01 – Minor Subject

Welcome!

Today's Agenda

- Organization & modalities
- Client side scripting: JavaScript (repetition)
  - Repetition of the basics
  - Event handling
- Quiz

Organization & Modalities

Dates

- **For major subject students** (Master)
  Medien-/Informatik, Mensch-Computer-Interaktion:

Day	Time	Tutor
Monday	16 – 18 h	Peter Juras
Monday	18 – 20 h	André Schmidt
Wednesday	14 – 16 h	Tobias Stockinger
Wednesday	18 – 20 h	Thomas Weber

- **For minor subject students** (Bachelor)
  Kunst und Multimedia, Pädagogik, Statistik, Lehramt:

Day	Time	Tutor
Wednesday	16 – 18 h	Thomas Weber

Programming Trainings

- Depending on the demand, we offer programming trainings instead of regular tutorials

- Individual consultation is optional. Please contact your tutor to arrange an appointment.

Tutorials – Why are we doing this?

- Application and immersion of lecture content
- Hands-on activities and discussion
- Opportunity to ask questions
- Preparation of the upcoming assignment
- Discussion of the solutions to exercises

Procedure – Part 1

- Slides and assignment online prior to tutorial
- Due dates for assignments: one **or** two weeks.
  Wednesday to Wednesday.
- News, updates, and important announcements on the official website:
  http://www.medien.ifi.lmu.de/lehre/ws1516/mmn/

Procedure – Part 2

- Doing the assignments is completely **voluntary**.
- We recommend you do the assignments, because they …
  - are fun and challenging.
  - go beyond the lecture content.
  - prepare you to pass the exam.
- Assignments are turned in via UniWorX
  - Make sure to check the due date
  - You can't hand in an assignment after the deadline.
  - Individual- or group submission
  - Make sure to do the right assignment:
    - Assignment 01 (HF) = Hauptfach, major subject students
    - Assignment 02 (NF) = Nebenfach, minor subject students

Exam

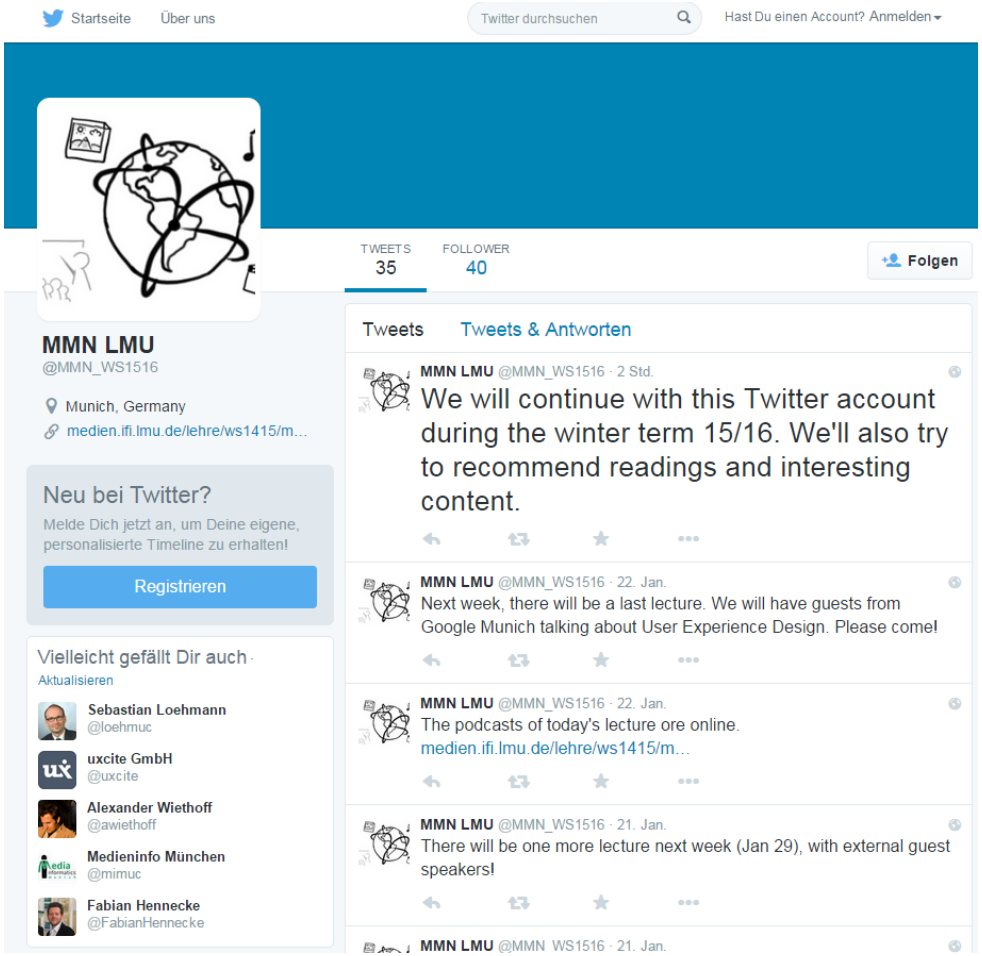
- Date and time: **11.02.2016** between **10-12 a.m**.
- Location: Main building, lecture halls **M118** and **A240**
- Most likely open-book.
- The exam includes tasks from both the **lecture and tutorial**!
- Retry exam probably in April 2016.

Semester Plan (subject to change)

Dates	Topics
21.10.	Organization, JavaScript Basics and Repetition
28.10.	Server-side scripting with PHP – Basics
04.11.	PHP: Sessions
11.11.	MySQL
18.11.	PHP with MySQL
25.11.	jQuery: DOM Manipulation, Animations
02.12.	jQuery: AJAX
09.12.	HTML5 Form validation
16.12.	HTML5 Canvas
23.12.	Christmas Tutorial – Programming Consultation
13.01.	Multimedia Content Description
20.01.	HTML5 Multimedia
27.01.	Repetition
03.02.	Repetition

News, Readings, Q&A via Twitter

https://twitter.com/MMN_WS1516



Client Side Scripting: JavaScript + HTML5 = ♥

Where do you see it in action every day?

HTML5

HTML5 introduced a bunch of new features:

- New Elements:
  - `<canvas></canvas>`
  - `<audio></audio>`
  - `<video></video>`
  - More: http://www.w3schools.com/html/html5_new_elements.asp
- Form features (examples):
  - Wildcards
  - Validation
- Drag and Drop

Editing HTML Files

- HTML files are regular **text-files** with special **mark-up**
- Programs installed on CIP-pool machines:
  - gedit
  - BlueFish
- Other HTML editors:
  - Aptana (IDE, open source)
  - Sublime (powerful text editor, commercial, free to try)
  - Webstorm (commercial IDE, free for students)

HTML5: Document Structure

```
<!DOCTYPE html>
<html lang="de">

<head>
    <meta charset="UTF-8" />
    <title>HTML5 Structure<title>
</head>

<body>
</body>

</html>
```

html_skeleton.html

JavaScript

- JavaScript is a dynamic scripting / programming language
- Code is interpreted by the web browser
- Code can be embedded into HTML

```
<script>
/*
    Here goes your script!
 */
</script>
```

- Alternatively, the code can be included anywhere in the <head> or <body> from a separate file

```
<script src="myScript.js"></script>
```

Variables

- Values (numbers, strings…) can be stored in variables
- The values are changeable (hence the name „variable“)
- JavaScript does not require a type declaration, you only use the keyword **`var`** (dynamic type system)

- Declare variables:

```
var letterCount = 10; // explicit
wordCount = 20;       // implicit
```

Functions

- Functions perform specific tasks as a unit
- Used to avoid redundant code
- In JavaScript, we can define functions in two ways:

```
function sayHello(){ // option A
    alert("Hello!");
}

var sayHelloAgain = function(){ // option B
    alert("Hello again!");
}
```

- Difference: `sayHello` can be called in code above its definition, `sayHelloAgain` only works in code below its definition. http://www.w3schools.com/js/js_function_definition.asp

Scopes

- Variables and functions have different visibilities (**scopes**) depending on **where** and **how** they were declared.

```
<script>
    var letterCount = 10;

    function alertCounts(){
        var wordCount = 2;
    }

</script>
```

scopes_1.html

Advanced “Scope” Example

```
var var1 = 10;
function fun1(){
    var var2 = 20;
    var3 = 30;
    var var1 = 40;
    alert("Fun1 Var 1: " + var1);
}


alert("Var 1: " + var1);
alert("Var 2: " + var2);
alert("Var 3: " + var3);

fun1();

alert("Var 1: " + var1);
alert("Var 2: " + var2);
alert("Var 3: " + var3);
```

scopes_2.html

Loops

- For loop:

```
for(var i=0; i<10; i++){
    console.log("I'm counting");
}
```

- While loop

```
var nagCount=0;
while(nagCount<10){
    console.log("Mommy!");
    nagCount++;
}
```

I'm counting
I'm counting
I'm counting
I'm counting
I'm counting
I'm counting
I'm counting
I'm counting
I'm counting
I'm counting
Mommy!
Mommy!
Mommy!
Mommy!
Mommy!
Mommy!
Mommy!
Mommy!
Mommy!
Mommy!

loops_1.html

Example: For Loops, Arrays, Concatenation

```
<script>
    var fruits = new Array("Oranges", "Apples", "Pears");
    var output = "";
    for(var i=0; i<fruits.length; i++){
        output = output + fruits[i] + ", ";
    }

    alert("I like eating: " + output);
</script>
```

loops_2.html

User Interaction with Forms

```
<input type="text"/>
```


```
<button>Send</button>
```

- **Mouse Events**
  - onclick
  - onmouseover
  - …
- **Keyboard Events**
  - onkeypress
  - onkeyup
  - …
- **…**

User Interaction - EventListener

- Basically all HTML elements can trigger certain events
- JavaScript can listen for and consequently handle such events
- EventListener attribute:

```
 body>

<script>
    function showTime(){
        alert(new Date().toString());
    }
</script>

<button onclick="showTime()">Show Time!</button>

</body>
```

date_popup.html

DOM (Document Object Model)

- The DOM references every element and its content in an HTML (or XML) document.
- Elements, contents and structure can be modified:
  - **`document`**: Content of the browser window
  - **`getElementById()`**: gets an HTML element with a unique identifier
  - **`getElementsByTagName()`**: gets all elements by a specific tag
  - **`querySelector()`**: Find first node that matches a CSS selector
  - **`Node.firstChild`**: returns the first child node
  - **`Node.nodeValue`**: gets or sets the value of a node

- http://wiki.selfhtml.org/wiki/JavaScript
  http://de.selfhtml.org/javascript/index.htm

Example: Say Hello

```
<body>
<input type="text" id="nameInput" />
<button onclick="sayHello()" />

<script>
    function sayHello(){
        var nameInput = document.getElementById('nameInput');
        var name = nameInput.value;
        if(name){
            alert("Hello "+name+"!");
        }
        else{
            alert("Why don't you tell me your name first?")
        }
    }

</script>

</body>
```

whats_your_name.html

Break Out Task

- Create two user input fields
  - one for text input
  - the other is for a number
- Create a dynamic output
  - display the text input as often as the number indicates.
  - update the output when any of the two fields change

- Take 25 Minutes time

Break Out Task – Demo Video

Quiz Part 1

1. Name 3 elements that are ‘new’ in HTML5!
2. Which document type is correct for HTML5:
   a) <!DOCTYPE html>
   b) <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 5.0//EN" "http://www.w3.org/TR/html5/strict.dtd">
   c) <!DOCTYPE HTML5>
3. Is JavaScript usually used on the client or the server side?
4. Which attribute of <script> is no longer required?
   rel | href | src | type

http://www.w3schools.com/quiztest/quiztest.asp

Quiz Part 2

1. What does `document.querySelector('input');` do?
2. What is the difference between these two declarations:
   ```
   testCount = 2;
   var testCases = 10;
   ```
3. What event is fired when the user types in an input field?
4. onclick is an…
   a) element
   b) event attribute
   c) style attribute

Link Collection

- http://caniuse.com/
- http://www.w3schools.com/js/default.asp
- https://stackoverflow.com/
- https://www.coursera.org/learn/html-css-javascript
- http://www.html5rocks.com/

Thanks!

What are your questions?

Let’s begin with the Assignment!

- Download the assignment sheet
- Start with task 1
- You can collaborate with your neighbor

- Turn in the assignment by October 28, 12:00 noon via UniWorX