

Multimedia im Netz

Wintersemester 2013/14

Übung 11 (Nebenfach)

Klausur MMN

- **Termin:** **19.02.2014 (10-12 Uhr)**
- **Ort:** **B138 und B139** in der Theresienstr.
- Aufteilung auf die beiden Räume wird vor der Klausur noch bekannt gegeben

- **Anmeldung:** bis **12.02.2014 (23:00 Uhr)**
- **Abmeldung :** bis **17.02.2014 (12:00 Uhr)**

- Nur Leute die angemeldet sind, können an der Klausur teilnehmen.
- Sollte man nicht an der Klausur teilnehmen (obwohl man sich angemeldet hat) muss sich wieder von der Klausur abmelden!

Weitere Informationen

- Zur Teilnahme an der Klausur müssen Studierende einen gültigen Lichtbild- und den Studentenausweis mitbringen
- Die Klausur ist Closed-Book. Für Programmieraufgaben gibt es einen Anhang in der Klausur mit Auszügen aus API's der einzelnen Programmiersprachen.

Nachholprüfung

- Es wird eine Möglichkeit zur Nachholung angeboten werden. In welcher Form (schriftlich oder mündlich) wird je nach Bedarf nach der Klausur entschieden.

Themen

Mögliche Schwerpunkte

- Interaktive Web Applikationen (PHP, MySQLi, jQuery, ...)
- Digital Rights Management
- Multimedia Content Description
- Streaming-Architekturen

Ausgeschlossene Kapitel

- Kapitel 4: Web Programming with Java
- Kapitel 5: Communities, the Web and Multimedia
- Kapitel 7: Cryptographic Techniques

Wiederholung: JavaScript (1)

- JavaScript ist eine eigene Programmiersprache
- Code wird vom Browser interpretiert
- Code kann in HTML eingebettet werden:

```
<html>
  <head><title>Example</title>
    <script type="text/javascript">
      alert("Hallo Welt");
    </script>
  </head>
  <body></body>
</html>
```

Wiederholung: JavaScript (2)

- Wie auch in PHP, hat JS verschiedene Daten- und Kontrollstrukturen:
 - Variablen
 - Arrays
 - Funktionen
 - For-Schleife
 - If-Then-Else
 - Etc.

Wiederholung: JavaScript (3)

- **Geltungsbereiche:**

Je nachdem wie und wo Variablen in JS deklariert werden, haben Sie unterschiedliche Geltungsbereiche.

- **Aufgabe:**

Der Code auf der nächsten Seite kann zu verschiedenen Fehlern führen. Kommentieren Sie problematischen Zeilen des Codes aus und zeigen Sie dann auf, welche Alert-Ausgaben nacheinander gemacht werden.

Aufgabe: Geltungsbereiche

```
<html><head><title>Example</title>
<script type="text/javascript">
  var var1 = "A";

  function fun1(){
    var var2 = "B";
    var3 = "C";
    var var1 = "D";
  }

  alert(var1);
  alert(var2);
  alert(var3);
  fun1();
  alert(var1);
  alert(var2);
  alert(var3);
</script>
</head><body></body></html>
```


Wiederholung: Interaktion mit JavaScript

- **DOM:** Mit DOM kann jedes Element und dessen Inhalt in einem HTML / XML Dokument referenziert werden
- **Events:** JavaScript kann auf verschiedene Events reagieren (wichtig, z.B. bei Formularelementen)

Wiederholung: Interaktion mit JavaScript

- **DOM:** Mit DOM kann jedes Element und dessen Inhalt in einem HTML / XML Dokument referenziert werden
- **Events:** JavaScript kann auf verschiedene Events reagieren (wichtig, z.B. bei Formularelementen)
 - Mouse Events (onclick, onmousedown,...)
 - Keyboard Events (onkeydown, onkeypress, ...)
 - ...
- Es gibt in JS verschiedene Methoden um auf Events zu reagieren, z.B.: **addEventListener()**

Aufgabe: Interaktion mit JavaScript

```
<html><head><title>Example</title>
<script type="text/javascript">

</script>
</head>
<body>
  <input type="text" />
  <h3>Output</h3>
  <div id="output"></div>
</body>
</html>
```

Wiederholung: HTML 5

- Mit dem HTML-5 Standard kamen viele neue Features

Wiederholung: HTML 5

- Mit dem HTML-5 Standard kamen viele neue Features
 - HTML5 Formular-Validierung
 - Neue Input Types (E-Mail, URL, ...)
 - Neue Attribute (Platzhalter, Required, Pattern, ...)
 - Im Zusammenhang mit der Form-Validierung gibt es noch die Constraint Validation API (zum anpassen der Fehlermeldungen)
 - Neue Elemente
 - `<video></video>`,
 - `<audio></audio>`,
 - `<canvas></canvas>`
 - ...

Aufgabe: Formular-Validierung

- Gegeben sei das folgende Codegerüst (siehe nächste Folie). Es besteht aus einem Textfeld und einem Button. In das Textfeld kann der Nutzer Groß- und Kleinbuchstaben, sowie Zahlen eingeben. Das Feld ist nicht optional.

Versucht der Nutzer eine Eingabe zu machen, die der obigen Beschreibung nicht entspricht, erhält der Nutzer eine angepasste Fehlermeldung.

- Ergänzen Sie den Code auf der nächsten Folie so, damit die oben beschriebene Funktionalität umgesetzt wird.

Aufgabe: Formular-Validierung

```
<!DOCTYPE html><html lang="de">
<head><title>HTML5</title></head>
<body><form><div>
<label for="text">Text: </label>
<input id="text" type="text" name="text"/>

<input type="submit" id="submit" />
</div></form>

<script>

</script>
</body> </html>
```

Wiederholung: Audio/Video Element

- Kann verwendet werden, um Video- und Audio-Elemente einzubetten

Wiederholung: Audio/Video Element

- Kann verwendet werden, um Video- und Audio-Elemente einzubetten
- Hat den Vorteil, dass man keine zusätzlichen Player installieren muss (z.B. Flash)
- Hat aber auch den Nachteil, dass nicht alle Browser dieselben Formate unterstützen
- Man kann jedoch mehrere Medienquellen angeben und als allerletzte Option (wenn keine der Dateien funktioniert) dann doch auf ein Plugin zurückgreifen.

Aufgabe: Video-Element

- Sie haben ein Video in den Formaten movie.mp4 und movie.ogv vorliegen, welche Sie in einer Webseite mit Hilfe des HTML5 video-tags einbetten möchten. Wie sieht der entsprechende Code dafür aus?

Aufgabe: Video-Element

- Sie haben ein Video in den Formaten movie.mp4 und movie.ogv vorliegen, welche Sie in einer Webseite mit Hilfe des HTML5 video-tags einbetten möchten. Wie sieht der entsprechende Code dafür aus?

```
<video width="320" height="240" controls="controls">  
  <source src="movie.mp4" type="video/mp4">  
  <source src="movie.ogv" type="video/ogg">
```

Your browser does not support the video tag.

```
</video>
```

Danke! Fragen?