Multimedia im Netz
(Online Multimedia)
Wintersemester 2014/15

Übung 02 (Hauptfach)
Organization: Slides

- Slides for the tutorial (HF) in English from now on
- consistent with lecture...
## Organization: Topics

<table>
<thead>
<tr>
<th>#</th>
<th>Date</th>
<th>Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>20.10.2014</td>
<td>Database access, MySQL, PHP</td>
</tr>
<tr>
<td>3</td>
<td>27.10.2014</td>
<td>HTML5 &amp; JavaScript Introduction</td>
</tr>
<tr>
<td>4</td>
<td>03.11.2014</td>
<td>Web Interactivity: Drag &amp; Drop with HTML5</td>
</tr>
<tr>
<td>5</td>
<td>10.11.2014</td>
<td>jQuery: DOM Traversal and Manipulation, jQuery UI</td>
</tr>
<tr>
<td>6</td>
<td>17.11.2014</td>
<td>jQuery: AJAX with PHP backend</td>
</tr>
<tr>
<td>7</td>
<td>24.11.2014</td>
<td>NodeJS: Introduction, static web content</td>
</tr>
<tr>
<td>8</td>
<td>01.12.2014</td>
<td>NodeJS: Express, RESTful Web-Apps</td>
</tr>
<tr>
<td>9</td>
<td>08.12.2014</td>
<td>Excursus: Google Charts, Google Maps, Design Patterns</td>
</tr>
<tr>
<td>10</td>
<td>15.12.2014</td>
<td>Mashups with a multitude of technologies</td>
</tr>
<tr>
<td>--</td>
<td>22.12.2014</td>
<td>Christmas break / programming consultations</td>
</tr>
<tr>
<td>11</td>
<td>12.01.2014</td>
<td>Web apps with offline storage</td>
</tr>
<tr>
<td>12</td>
<td>19.01.2014</td>
<td>Questions</td>
</tr>
<tr>
<td>13</td>
<td>26.01.2014</td>
<td>Buffer</td>
</tr>
</tbody>
</table>
PHP & MySQL

• Multiple functions and APIs available for PHP to work with databases:
  – `mysql` („Deprecated“ since PHP 5.5.0)
  – `mysqli` (i is for „improved“)
  – PDO (PHP Data Objects)
• „mysql“ is still supported for older PHP versions
• It is highly recommendable to use mysqli or PDO
MySQL at the CIP-Pool

• Access “Datenbank Management” here: https://tools.rz.ifi.lmu.de/
• Create a new account (required)
• Create a new database (required)
• Connect to db2.cip.ifi.lmu.de
<?php
$c = mysql_connect("localhost", "user", "password");

if($c){
    echo "Connection to database has been established.";
} else {
    echo "Could not connect to database";
}
?>
Mysql (I)

- Establish connection
  ```php
  $c = mysql_connect("localhost", "user", "password");
  ```

- Create database
  ```php
  $query = "CREATE DATABASE mydb";
  mysql_query($query, $c);
  ```

- Select database for further calls
  ```php
  mysql_select_db("mydb");
  ```

- Create table
  ```php
  $query = "CREATE TABLE Personen(Name CHAR(30))";
  mysql_query($query, $c);
  ```

- Close connection
  ```php
  mysql_close($c)
  ```
Mysql (II)

• Exemplary queries:
  
  $query = "SELECT name FROM mydb";
  $query = "INSERT INTO mydb VALUES ('$name')";
  $query = "UPDATE mydb SET name='$name' WHERE pId=2";

• Further information: http://dev.mysql.com/doc/

• PHP-statement for a MySQL Query
  
  $results = mysql_query($query);

• Process the results
  mysql_fetch_array($result);
  mysql_fetch_array($result, MYSQL_ASSOC);
  mysql_fetch_array($result, MYSQL_NUM);
Mysqli

- Offers two interfaces
  - procedural
  - object-oriented
- Supports „prepared“ statements (recommendable)
- Supports multiple statements within one query
- Supports transactions
- Improved debugging tools
Mysqli (procedural)

• Establish connection
  
  ```php
  $c = mysqli_connect("localhost", "user", "password", "mydb");
  ```

• Select database
  
  ```php
  mysqli_select_db("mydb");
  ```

• Close connection
  
  ```php
  mysqli_close($c)
  ```

• PHP statement for MySQL query
  
  ```php
  $results = mysqli_query($c, $query);
  ```

• Process the results:
  
  ```php
  mysqli_fetch_array($result);
  mysqli_fetch_array($result, MYSQLI_NUM);
  mysqli_fetch_array($result, MYSQLI_ASSOC);
  ```
Object Oriented Programming in PHP (I)

• OOP paradigms / concept
  – classes
  – objects

• PHP class signature:

```php
<?php

class classMMN{

    //put members and methods here

}

?>
```
Object Oriented Programming in PHP (II)

Define member variables of a class:

```php
<?php
   class classMMN{
      var $name = "Multimedia im Netz";
      var $semester = "Wintersemester 2014/15";
      var $professor = "Prof. Dr. Heinrich Hussmann";
      var $termin = "Donnerstag: 10-13 Uhr";
   }
?>
```
Object Oriented Programming in PHP (III)

• Importing and instantiating a class

```php
<?php
    require_once("classMMN.php");
    $mmn = new classMMN();
?>
```

• Object access

```php
    echo "Veranstaltung: " . $mmn->name . "</br>";
    echo "Semester: " . $mmn->semester . "</br>";
    echo "Professor: " . $mmn->professor . "</br>";
```
Object Oriented Programming in PHP (III)

• Add methods:

```php
<?php

class classMMN{

    var $name = "Multimedia im Netz";
    var $semester = "Wintersemester 2014/15";
    var $professor = "Prof. Dr. Heinrich Hussmann";
    var $termin = "Donnerstag: 10-13 Uhr";

    function setTermin(var $termin){
        $this->termin = $termin;
    }

    function getTermin(){
        return $this->termin;
    }

}

?>
```
Object Oriented Programming in PHP (IV)

• Call methods

```php
<?php
    $mmn->setTermin("Mittwoch: 10-13 Uhr");
    echo $mmn->getTermin();
?>
```
Object Oriented Programming in PHP (V)

• Constructor: PHP’s constructors are methods with a special name: `__construct();`

```php
function __construct($name, $sem, $prof, $termin){
    $this->name = $name;
    $this->semester = $semester;
    $this->professor = $professor;
    $this->termin = $termin;
}
```

• Use constructor:

```php
$mmn = new classMMN("MMI2", "WS 2014/15", "Prof. Dr. Butz", "Freitag: 14-16 Uhr");
```
Mysqli (object oriented)

• Establish connection
  $c = \text{new} \ \text{mysqli}("\text{localhost}", \text{"root"}, \text{""}, \text{"mydb"});

• PHP statement for MySQL query
  $\text{results} = \text{c->query($query)};

• Process the results
  $\text{results->fetch_assoc}();
  $\text{results->fetch_row}();
  $\text{results->fetch_all}($\text{MYSQLI\_BOTH});
  $\text{results->fetch_all}($\text{MYSQLI\_ASSOC});
  $\text{results->fetch_all}($\text{MYSQLI\_NUM});

• Close the connection
  $\text{c->close}();
Mysqli: Prepared Statements (I)

• Separate structure and data through the use of wildcards. In the query we use „?“ as wildcards

• **Advantages:**
  – You can reuse the query with different parameters
  – More secure (cf. SQL Injections)

• **How to do it:**
  – „Prepare“: Prepare the query. The template is checked for errors an.
  – „Bind“: Bind the parameters to the wildcards
  – „Execute“: The query is executed with the passes parameters
Mysqli: Prepared Statements (II)

• Query with wildcards
  
  $q = "SELECT Nachname FROM myDB WHERE Vorname=?";

• Prepare the query
  
  $query = $c->prepare($q);

• Bind the parameters
  
  $name = "Isabell";
  
  $query->bind_param("s", $name);

• Execute the query
  
  $query->execute();
Mysqli: Prepared Statements (III)

- Bind result columns to variables
  $query->bind_result($nachname);

- Fetch results
  $query->fetch()
Example: Prepared Statements

```php
<?php
include_once 'db_info.php';

$c = new mysqli($host, $user, $password, $db);
$query = $c->prepare("SELECT Nachname FROM Uebung02 WHERE Vorname=?");

$name = "Isabell";
$query->bind_param("s", $name);
$query->execute();
$query->bind_result($nachname);

while($query->fetch()){
    echo $nachname;
    echo "<br/>";
}
?>
```
Cryptographic Hashes

• md5()
  – Message Digest Algorithm5
  – generates a 128bit hash value
  – fast, but vulnerable

• sha1()
  – generates a 160bit hash value
  – used in most (older) SSL certificates (they need to be replaced soon)
  – fast, but vulnerable
Password Hashing in PHP (I)

• PHP has built-in password hashing functions
  – password_hash()
  – password_verify()
  – ...

• Don’t store plain text passwords in databases.... Ever ;)

• Advantages:
  – More secure
  – easy to use

• Disadvantages:
  – only available with PHP >= 5.5.0
Password Hashing in PHP (II)

• Hashing a password:

```php
$pwHash = password_hash("password1234", PASSWORD_DEFAULT);
```

• Verifying a hash:

```php
if(password_verify("password1234", $pwHash)) {
    echo "Your password is correct";
}
```
Assignment 2

• Thema: Picture Gallery, Authentication
• Due in: 1 Week
• Due date: 27.10.2014 12:00
Thanks!
What are your questions or comments?