Today’s Agenda

• Repetition:
  – Sessions: Powerpoint Karaoke
  – Discussion of Assignment 03

• Database access through PHP & MySQL

• Break Out: Music Library Management app

• Quiz
PHP Sessions: Powerpoint Karaoke
Sessions with PHP

• Sessions need to be started before any output occurs

• Create session ID cookie:
  session_start()

• Delete the session ID cookie:
  session_destroy()

• Read / write session values:
  – superglobal $_SESSION array
  – immediately reset session like this $_SESSION = array();
Example: Counting visits

```php
<?php
session_start();

<!DOCTYPE html>
<html>
<body>

<?php
if(!isset($_SESSION['count'])){
    $_SESSION['count'] = 1;
}
else{
    $_SESSION['count']++;}

echo '<p>Current count: '.$_SESSION['count'].'</p>,'

?>
</body></html>
```
Example: Destroying Sessions

```php
<?php
session_start();
<!DOCTYPE html>
<html>
[...]
<body>

<?php
if(isset($_POST['destroy'])){  
    session_destroy();  
    $_SESSION = array();
}

if(!isset($_SESSION['count'])){  
    $_SESSION['count'] = 1;
} else{  
    $_SESSION['count']++;
}

echo '<p>Current count: '.$_SESSION['count'].'</p>';

<form method="post">  
    <input type="submit" name="destroy" value="Reset"/>
</form>
</body>
</html>
```
You have 7 remaining guesses.

A B C D

A B C E

A B F E

You won.
The code was: A B F E
Persistent Data Storage: PHP & MySQL
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";
}
?>
Mysqli

• i stands for “improved”
• Offers two interfaces
  – Procedural (traditional)
  – Object-oriented (see lecture slides)
• Supports…:
  – „prepared“ statements
  – multiple statements within one query
  – transactions
• Improved debugging tools
Mysqli (procedural)

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

- Select database
  ```php
  mysqli_select_db($c,"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($results);
  mysqli_fetch_array($results, MYSQLI_NUM);
  mysqli_fetch_array($results, MYSQLI_ASSOC);
  ```
Mysqli (object oriented)

- Establish connection
  ```php
  $c = new mysqli("host","user","password","db");
  ```

- PHP statement for MySQL query
  ```php
  $results = $c->query($query);
  ```

- Process the results
  ```php
  $results->fetch_assoc();
  $results->fetch_row();
  ```

- Close the connection
  ```php
  $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
  $query = "SELECT lastName FROM people WHERE firstName=?";

• Create the statement
  $statement = $c->prepare($query);

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

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

- Bind result columns to variables
  $statement->bind_result($lastNameResults);

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

```php
<?php
include_once('connectionInfo.php');
$c = new mysqli($host,$user,$password,$db);
$query = "SELECT lastName FROM people WHERE firstName=?";
$statement = $c->prepare($query);
$name = "Sam";
$statement->bind_param("s", $name);
$statement->execute();
$statement->bind_result($lastNameResults);
while($statement->fetch()){
    print_r($lastNameResults);
    echo "<br />";
}
```
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 of `password_hash()` and `password_verify()`:
  - 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";
  }
  ```
Break-Out Task

- Create a music library management tool
- Allow users to create artist & album entries
- There is a code skeleton on GitHub: https://github.com/MIMUC-MMN/tutorials-15-16/tree/master/tutorial04/breakout
After an insert...

The album was successfully inserted into the database

<table>
<thead>
<tr>
<th>ID</th>
<th>Artist</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>The Rolling Stones</td>
<td>Exile on Main Street</td>
</tr>
</tbody>
</table>

**Album Artist:** ___________________  **Album Title:** ___________________  [Add Album]
If you have time...

• Allow deleting the entries
• Avoid duplicates
• Create additional optional fields for runtime, track count, buying link etc.
Round-up Quiz

1. Name an advantage of mysqli over the deprecated mysql.
2. What does mysqli->fetch_assoc() do?
3. Why is it recommendable to call mysqli->close?
4. What is an advantage of prepared statements?
5. What are the steps of working with prepared statements?
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 November 9th, 12:00 noon via UniWorX