

Multimedia im Netz

Wintersemester 2013/14

Übung 02 (Hauptfach)

PHP und MySQL

- Es werden mehrere APIs von PHP angeboten um mit Datenbankabfragen zu arbeiten
 - mysql („Deprecated“ seit PHP 5.5.0)
 - mysqli (i steht für „improved“)
 - PDO (PHP Data Objects)
- mysql um mit älteren MySQL Versionen zu arbeiten (< 4.1.3)
- Ansonsten wird empfohlen, mysqli oder PDO zu verwenden

MySQL im CIP-Pool

- Unter <https://tools.rz.ifi.lmu.de/> “Datenbank Management” aufrufen
- Neuen Datenbank Account anlegen (MUST DO)
- Neue Datenbank anlegen (MUST DO)
- Verbindung via db2.cip.ifi.lmu.de

Verbindung Testen

```
<?php
    $c = mysql_connect("localhost", "user", "password");

    if($c){
        echo "Die Verbindung zur DB wurde aufgebaut.";
    } else {
        echo "Es konnte keine Verbindung zur DB aufgebaut
            werden.";
    }
?>
```

Mysql (I)

- Verbindung aufbauen

```
$c = mysql_connect("localhost", "user", "password");
```

- Datenbank anlegen

```
$query = "CREATE DATABASE mydb";  
mysql_query($query, $c);
```

- Datenbank auswählen

```
mysql_select_db("mydb");
```

- Tabelle erstellen

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

- Verbindung beenden

```
mysql_close($c)
```

Mysql (II)

- Mögliche SQL- Anfragen

```
$query = "SELECT name FROM mydb" ;
```

```
$query = "INSERT INTO mydb VALUES ( '$name' )" ;
```

```
$query = "UPDATE mydb SET name=' $name' WHERE pId=2" ;
```

- Weitere Informationen: <http://dev.mysql.com/doc/>

- PHP-Befehl für MySQL-Query (Anfrage)

```
$results = mysql_query($query) ;
```

- Ergebnisse verarbeiten

```
mysql_fetch_array($result) ;
```

```
mysql_fetch_array($result, MYSQL_ASSOC) ;
```

```
mysql_fetch_array($result, MYSQL_NUM) ;
```

Mysqli

- Bietet zwei Schnittstellen an
 - Prozedural
 - Objektorientiert
- Unterstützt „prepared“ Statements
- Unterstützt mehrere Statements
- Unterstützt Transaktionen
- Verbesserte Debugging-Möglichkeiten

Mysqli (Prozedural)

- Verbindung aufbauen

```
$c = mysqli_connect("localhost", "user", "password",  
"mydb");
```

- Datenbank wechseln

```
mysqli_select_db("mydb");
```

- Verbindung beenden

```
mysqli_close($c)
```

- PHP-Befehl für MySQL-Query (Anfrage)

```
$results = mysqli_query($c, $query);
```

- Ergebnisse verarbeiten

```
mysqli_fetch_array($result);  
mysqli_fetch_array($result, MYSQLI_NUM);  
mysqli_fetch_array($result, MYSQLI_ASSOC);
```


Exkurs: Objektorientierung in PHP (I)

- Grundelemente der objektorientierten Programmierung:
 - Klassen
 - Objekte
- Grundgerüst einer PHP-Klasse:

classMMN.php

```
<?php
    class classMMN{

        //Hier kommen Attribute und Methoden einer
        //Klasse

    }
?>
```

Exkurs: Objektorientierung in PHP (II)

Attribute einer Klasse definieren:

classMMN.php

```
<?php
    class classMMN{
        var $name = "Multimedia im Netz";
        var $semester = "Wintersemester 2013/14";
        var $professor = "Prof. Dr. Heinrich Hussmann";
        var $termin = "Donnerstag: 10-13 Uhr";
    }
?>
```

Exkurs: Objektorientierung in PHP (III)

- Einbinden einer Klasse und Objekt erzeugen

mmn.php

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

?>
```

- Zugriff auf die Werte einer Klasse

mmn.php (Fortsetzung)

```
echo "Veranstaltung: " . $mmn->name . "</br>";
echo "Semester: " . $mmn->semester . "</br>";
echo "Professor: " . $mmn->professor . "</br>";
```

Exkurs: Objektorientierung in PHP (III)

- Methoden hinzufügen:

classMMN.php

```
<?php
class classMMN{
    var $name = "Multimedia im Netz";
    var $semester = "Wintersemester 2013/14";
    var $professor = "Prof. Dr. Heinrich Hussmann";
    var $termin = "Donnerstag: 10-13 Uhr";

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

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

Exkurs: Objektorientierung in PHP (IV)

- Methoden Aufrufen:

mmn.php

```
<?php
    $mmn->setTermin("Mittwoch: 10-13 Uhr");
    echo $mmn->getTermin();
?>
```

Exkurs: Objektorientierung in PHP (III)

- Konstruktor: In PHP ist der Konstruktor einer Klasse eine Methode mit einem besonderen Namen: **__construct()**;

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

- Konstruktor verwenden:

```
$mmn = new classMMN("MMI2", "WS 2013/14",  
                    "Prof. Dr. Butz", "Freitag: 14-16 Uhr");
```

Mysqli (Objektorientiert)

- Verbindung aufbauen

```
$c = new mysqli("localhost", "root", "", "mydb");
```

- PHP-Befehl für MySQL-Query (Anfrage)

```
$results = $c->query($query);
```

- Ergebnisse verarbeiten

```
$results->fetch_assoc();
```

```
$results->fetch_row();
```

```
$results->fetch_all(MYSQLI_BOTH);
```

```
$results->fetch_all(MYSQLI_ASSOC);
```

```
$results->fetch_all(MYSQLI_NUM);
```

- Verbindung beenden

```
$c->close();
```

Mysqli: Prepared Statements (I)

- Trennung von Logik und Daten, indem man mit Platzhaltern „?“ innerhalb der Anfrage arbeitet
- **Vorteile:**
 - Man kann dieselbe Anfrage mit verschiedenen Daten ausführen
 - Sicherer (z.B. gegenüber SQL Injections)
- **Ablauf:**
 - Prepare: Anfrage vorbereiten. Das Anfragentemplate wird zum Server gesendet und auf Syntaxfehler überprüft.
 - Execute: Die Anfrage wird mit den übergebenen Daten ausgeführt.

Mysqli: Prepared Statements (II)

- Anfrage mit Platzhalter
`$q = "SELECT Nachname FROM myDB WHERE Vorname=?";`
- Anfrage Vorbereiten
`$query = $c->prepare($q);`
- Parameter binden
`$name = "Isabell";`
`$query->bind_param("s", $name);`
- Ausführen der Anfrage
`$query->execute();`

Mysqli: Prepared Statements (III)

- Ergebnisspalten an Variablen binden
`$query->bind_result($nachname);`
- Ergebnisse holen
`$query->fetch();`

Beispiel: Prepared Statements

```
<?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/>";
    }
?>
```

Übungsblatt 2

- **Thema:**
- Bearbeitungszeit: 1 Woche
- Abgabe: 13.11.2013 23:00 Uhr

Danke! Fragen?