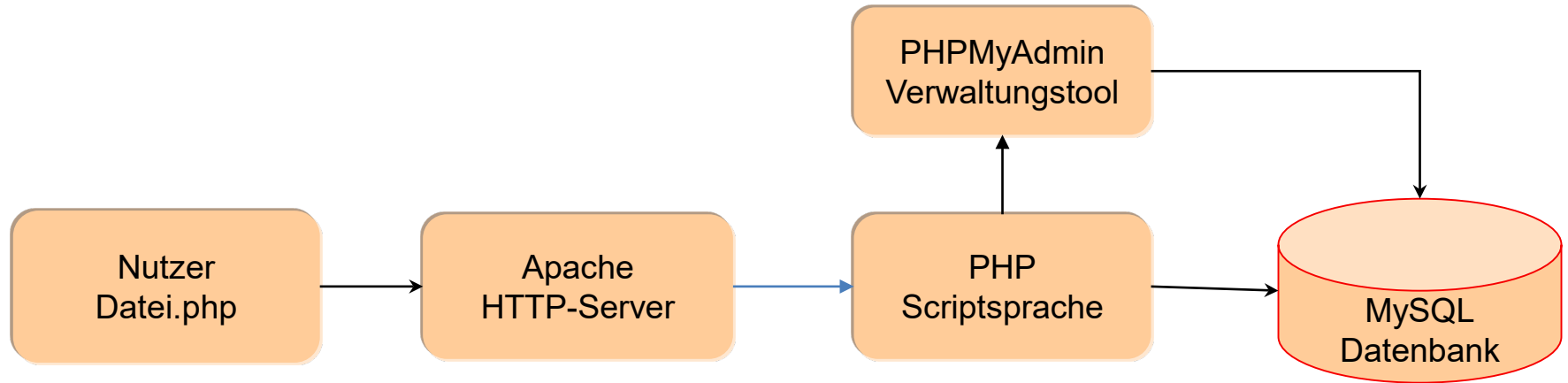


PHP und MySQL (MariaDB)

Zusammenspiel PHP + MySQL



Warum eine Datenbank?

- Unabhängigkeit von Anwendung und Datenbank
- Standardisierte Abfragesprache (SQL = Structured Query Language)
- Sehr hohe Performance auch bei komplizierten Datenbank-Abfragen
- Möglichkeit Beziehungen und Abhängigkeiten zwischen Daten herzustellen
- Wahrung der Datenkonsistenz: Vermeidung doppelter Datensätze, Vermeidung widersprüchlicher Datensätze
- Leichte Sortierung der Daten möglich
- Vielfältige Backupstrategien verfügbar/möglich
- Leichter Abgleich mehrerer Datenbanken möglich

Die MySQL-Datenbank

Ein System zur Strukturierung, Speicherung und Wiedergewinnung von Daten.

MySQL ist ein sehr kompaktes und leistungsfähiges relationales Datenbanksystem.

Wichtige Leistungsmerkmale von MySQL sind:

- SQL-92-Konformität,
- Transaktionen,
- Fremdschlüssel,
- Volltextsuche,
- Unterabfragen,
- gespeicherte Prozeduren und Funktionen.

MySQL stellt fünf Tabellentypen für unterschiedliche Anwendungsbereiche zur Verfügung:

- MyISAM
- InnoDB,
- BDB,
- MERGE
- HEAP

Technische Funktionalität

MySQL ist ein RDBMS (Relational-Database-Management-System).

In einer relationalen Datenbank werden Daten in Tabellen und Spalten abgelegt.

Relationales DB-System:

RDBS → Wenn einzelne Tabellen einer Datenbank in Verbindung zueinander stehen.

Primärschlüssel

id	vorname	Nachname
1	Tobias	Hauser
2	Christian	Wenz
3	Melville	Herman

Tabelle autor

Primärschlüssel

id	Titel	autor
1	Das Website-Handuch	1
2	Javascript & AJAX	2
3	Moby Dick	3

Fremdschlüssel

Tabelle title

Wie funktioniert MySQL?

MySQL verwendet als Sprache zur Abfrage/Manipulation von Daten SQL.

Auch zur Manipulation der Datenbankstrukturen (z. B. Tabellen-Definitionen) wird SQL verwendet.

MySQL stellt Grundsätzlich zwei Möglichkeiten zur Verfügung SQL Befehle an die Datenbank zu abzusetzen:

- Über das Kommandozeilenprogramm "mysql"
- Über die Schnittstelle der jeweiligen Programmiersprache (**mysql_-Befehle**)
- in PHP (mysqli - MySQL Improved Extension)
- MySQL speichert die Daten in einem binären Format auf der Festplatte oder im Hauptspeicher

Handwerkszeug – Klassen und Objekte

Klassen:

Typen von Objekten

abstraktes Modell oder Bauplan für viele ähnliche Objekte

Objekte:

Objekte können mit `new` erstellt werden.

Eigenschaften, Methoden:

Hängen am Objekt und können mit dem Pfeil abgerufen werden

```
class Hund {  
    public $rasse = "Pudel";  
  
    public function rede() {  
        echo "wuff";  
    }  
}
```

```
$bello = new Hund();
```

```
$bello->rede();  
echo $bello->rasse;
```

Mysqli

Moderne Objektorientierte Lösung für MySQL und PHP

Objektorientiert bedeutet, dass es Klassen für mysql zugriff gibt aus welchen wir uns Objekte erstellen.

Wichtigste Klasse ist `mysqli`

Legitimation am Datenbanksystem

```
<?php

// new mysqli($Server, $User, $Password, $Datenbank);

$db = new mysqli("localhost", "root", "", "studis");

if ($db->connect_errno > 0){
    die("Unable to connect to database: " . $db->connect_error );
}

echo "Datenbankzugriff erfolgreich!";

$db->close();

?>
```

Datenbank erstellen

```
<?php
$db = new mysqli("localhost", "root", "sesam");

if($db->connect_errno > 0){
    die('Unable to connect to database [' . $db->connect_error . '] <br />');
}
else {
    echo "<br />DB-Server wurde erreicht!";
}
$result = $db->query("CREATE DATABASE studis");

if(!$result){
    die('<br />There was an error running the query [' . $db->error . ']');
}
else {
    echo "<br />studis wurde angelegt!";
}

$db->close();
?>
```

Tabelle erstellen

```
<?php
$db = new mysqli("localhost", "root", "sesam", "studis");
if($db->connect_errno > 0){
    die('Unable to connect to database [' . $db->connect_error . ']');
}
$sql = ("CREATE TABLE adressen (
id INT(6) NOT NULL AUTO_INCREMENT PRIMARY KEY,
    Name VARCHAR(15),
    VorName VARCHAR(15),
    Strasse VARCHAR(20),
    Hausnummer INT(3),
    Postleitzahl INT(5),
    Stadt VARCHAR(20)) "
);
if(!$result = $db->query($sql)){
    die('There was an error running the query [' . $db->error . ']');
}
?>
```

Tabelle löschen

```
<?php
$db = new mysqli("localhost", "root", "sesam", "studis");
if($db->connect_errno > 0){
    die('Unable to connect to database [' . $db->connect_error . ']);
}
$sql = "DROP TABLE adressen";

if(!$result = $db->query($sql)){
    die('There was an error running the query [' . $db->error . ']);
}
?>
```

Datensatz in Tabelle einlesen

```
<?php
$db = new mysqli("localhost", "root", "sesam", "studis");
if($db->connect_errno > 0){
    die('Unable to connect to database [' . $db->connect_error . ']);
}

$sql =
    "INSERT INTO adressen (id, Name, VorName, Strasse, Hausnummer, Postleitzahl, Stadt)
    VALUES (NULL, 'Hans', 'Wurst', 'Amalienstrasse', '81', '76137', 'Karlsruhe)";

if(!$result = $db->query($sql)){
    die('There was an error running the query [' . $db->error . ']);
}

$db->close();
?>
```

Mehrere Datensätze in eine Tabelle einfügen

```
<?php
$db = new mysqli("localhost", "root", "sesam", "studis");
if($db->connect_errno > 0){
    die('Unable to connect to database [' . $db->connect_error . ']);
}
$namen=array("name1"=>array("1","Maier","Fritz","Moltkestr","23","76133","Karlsruhe"),
             "name2"=>array("2","Kunz","Ute","Karlstr","104","76136","Karlsruhe"),
             "name3"=>array("3","Gans","Gustav","Ottostr","12","77116","Leonberg"),
             "name4"=>array("4","Kasper","Lena","Kantstr","122","78378","Kurberg"));

foreach ($namen as $wert)
{ $sql = "INSERT INTO adressen (id,Name,VorName,Strasse,Hausnummer,Postleitzahl,Stadt)
  VALUES ('NULL','$wert[1]','$wert[2]','$wert[3]','$wert[4]','$wert[5]','$wert[6]')";

if(!$result = $db->query($sql)){
    die('There was an error running the query [' . $db->error . ']);
}}
$db->close();
?>
```

Datensätze aus `$_POST` in eine Tabelle einfügen (Variante 1)

```
<?php $db = new mysqli("localhost", "root", "sesam", "studis");

if($db->connect_errno > 0){
    die('Unable to connect to database [' . $db->connect_error . ']);
}

$sql = "insert into Leute (id, Vorname, Nachname, Stadt, Age, Nationalitaet)
values (NULL, '". $_POST["Vorname"]."',
        '". $_POST['Name']."',
        '". $_POST['Stadt']."',
        '". $_POST['Age']."',
        '". $_POST['Nationalitaet']. "')";

if(!$result = $db->query($sql)){
    die('There was an error running the query [' . $db->error . ']);
}$db->close();
?>
```

Datensätze aus `$_POST` in eine Tabelle einfügen (Variante 2)

```
<?php    $db = new mysqli("localhost", "root", "sesam", "ex");
if($db->connect_errno > 0){
    die('Unable to connect to database [' . $db->connect_error . ']);
}
// Übergabe der Array-Einträge an Variablen.
$VN = $_POST['Vorname']; $NN = $_POST['Name']; $ST = $_POST['Stadt']; $AL = $_POST['Age']; $NT =
    $_POST['Nationalitaet'];

$sql = "insert into Leute (id, Vorname, Nachname, Stadt, Age, Nationalitaet)
    values (NULL, '$VN', '$NN', '$ST', '$AL', '$NT)";

If (!$result = $db->query($sql)) {
    die('There was an error running the query [' . $db->error . ']);
}
$db->close();
?>
```


Datensätze aus `$_POST` in eine Tabelle einfügen (Variante 3)

```
$conn = new mysqli("localhost", "root", "sesam", "ex");
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}

if (isset($_POST["Marke"])) {
    $result = mysqli_query($conn, "INSERT INTO KFZ (Marke, Typ, Leistung, Verbrauch)
    VALUES ('$_POST[Marke]', '$_POST[Typ]', '$_POST[Leistung]', '$_POST[Verbrauch]')");

    if ($result) {
        print "Die Tabelle KFZ wurde befüllt!<br /><br />";
    }
    else {
        echo "Fehler mit KFZ!!!<br /><br />";
    }
}
```

Spalte aus Datensatz auslesen

```
<?php
$db = new mysqli("localhost", "root", "sesam", "studis");
if($db->connect_errno > 0){
    die('Unable to connect to database [' . $db->connect_error . ']');
}
$sql = ("SELECT Name FROM adressen");
if(!$result = $db->query($sql)){
    die('There was an error running the query [' . $db->error . ']');
}
while($row = $result->fetch_assoc()){
    echo "$row[Name] <br />";
}

$db->close();
?>
```

Aus einer Datenbanktabelle Daten auslesen

```
<?php
$db = new mysqli("localhost", "root", "sesam", "studis");
if($db->connect_errno > 0){
    die('Unable to connect to database [' . $db->connect_error . ']);
}
$sql = ("SELECT * FROM adressen");
if(!$result = $db->query($sql)){
    die('There was an error running the query [' . $db->error . ']);
}
while($row = $result->fetch_assoc()){
    echo $row['Name'] . '<br />';
    echo $row['VorName'] . '<br />';
    echo $row['Strasse'] . '<br />';
    echo $row['Hausnummer'] . '<br />';
    echo $row['Postleitzahl'] . '<br />';
    echo $row['Stadt'] . '<br />';
}
$db->close();
?>
```

Aus einer Datenbanktabelle Daten auslesen mit **foreach**

```
<?php
$db = new mysqli("localhost", "root", "sesam", "ex");
if($db->connect_errno > 0){
    die('Unable to connect to database [' . $db->connect_error . ']);
}

// Using iterators (support was added with PHP 5.4)
foreach ( $db->query('SELECT * FROM gruppe_a3 WHERE Alt > 40 and Geschlecht = "w"') as $row ) {
    echo "$row[Vorname] <br /> $row[Name] <br /> $row[Alt] <br /> $row[Geschlecht] <br /><br />";
}

$db->close();
?>
```

PHP/SQL-Übung

1. Erstellen Sie eine Tabelle zur Speicherung der folgenden Daten in der Datenbank „Menschen“ mit Phpmyadmin.
 1. id (Primärschlüssel)
 2. Vorname
 3. Nachname
 4. Alter (**Alter ist ein SQL-Befehl. Zeichenfolge „Alter“ im SQL-Teil vermeiden!**)
 5. Stadt
 6. Nationalität
2. Erstellen Sie ein PHP-Programm das über ein Formular die Personendaten entgegennehmen kann.
3. Erstellen Sie die entsprechenden PHP- und SQL-Anweisungen, damit die Eingabedaten in die unter Punkt 1 angelegte Tabelle, durch einen Klick auf eine Schaltfläche, eingetragen werden.
4. Erstellen Sie ein weiteres PHP-Programm, mit dem Sie die vorhandenen Einträge in Ihrer Tabelle auslesen und visualisieren können. Das zweite PHP-Programm soll durch einen Link unter dem Eingabeformular gestartet werden können.

Dateien hochladen 1

HTML-Seite:

```
<form enctype="multipart/form-data" action="upload.php" method="POST">  
<input type="hidden" name="MAX_FILE_SIZE" value="30000">  
Send this file: <input name="userfile" type="file">  
<input type="submit" value="Send File">  
</form>
```

Dateien hochladen 2

PHP-Programm upload.php:

```
<?php
$uploadDir = '/var/www/web1/html/upload_aufgabe/';
$uploadFile = $uploadDir . $_FILES['userfile']['name'];
echo "<pre>";
if (move_uploaded_file($_FILES['userfile']['tmp_name'], $uploadFile))
{ echo "Datei ist in Ordnung und Sie wurde erfolgreich hochgeladen.";
echo "\n";
print_r($_FILES);
}
else
{
echo "Es wurde ein Fehler gemeldet!\nHier sind die Fehler informationen:\n";
print_r($_FILES);
}
echo "</pre>";
?>
```