

6. vaja

PHP IN MYSQL

PHP Superglobals

- PHP globalne spremenljivke
- dostopne v vseh datotekah php
- `$GLOBALS`, `$_SERVER`, `$_REQUEST`, `$_GET`, `$_POST`, `$_FILES`, `$_ENV`, `$_COOKIE`, `$_SESSION`

Organizacija SUPB MySQL

- vsak SUPB vsebuje več `shem` (oz. baz)
- vsaka shema vsebuje nič ali več `tebel`
- tabele pozamezne sheme so vsebinsko povezane:
 - npr. shema `information_schema` vsebuje metapodatke o podatkovni bazi
 - npr. shema `spletna_trgovina` vsebuje tabele `stranka`, `artikel`, `racun`
- nadzor dostopa se izvršuje na podlagi para uporabniško ime/geslo:
 - uporabnik `root` ima vse pravice nad celotno bazo
 - nove uporabnike se doda z ukazom SQL

Povezovanje na PB

- proces SUPB MySQL komunicira po protokolu TCP na vratih 3360
- za dostop uporabimo:
 - terminal in ukaz

```
$ mysql -u <username> -p
```

- namenski program (phpMyAdmin)

Primer: ukaz SQL

- kreiramo shemo `spletna_trgovina`
- kreiramo uporabnika `student` z geslom `geslo`
- dodelimo pravice uporabniku `student` nad shemo `spletna_trgovina`

```
CREATE DATABASE spletna_trgovina;  
CREATE USER 'student'@'localhost' IDENTIFIED BY 'geslo';  
GRANT ALL ON spletna_trgovina.* TO 'student'@'localhost';
```

PHP in MySQL: funkcije

mysql_*

- osnovni način predstavlja uporaba funkcij iz skupine mysql_
- pogosti ukazi za delo z bazo:
 - mysql_connect()
 - mysql_select_db()
 - mysql_query()
 - mysql_close()
- stari način, ki je specifičen za MySQL in ne podpira drugih SUPB
- velja za manj varnega (SQL injection)

PHP in MySQL: Primer

```
<?php
$link = mysql_connect('localhost', 'mysql_user', 'mysql_password');
if (!$link) {
    die('Could not connect: ' . mysql_error());
}
echo 'Connected successfully';

$db_selected = mysql_select_db('foo', $link);
if (!$db_selected) {
    die ('Can\'t use foo : ' . mysql_error());
}

$result = mysql_query('SELECT * FROM table');
if (!$result) {
    die('Invalid query: ' . mysql_error());
}

mysql_close($link);
?>
```

PHP in MySQL: PDO - PHP Data Object

- knjižnica za poenoten dostop do različnih SUPB:
 - PDO_CUBRID, PDO_DBLIB, PDO_FIREBIRD, PDO_IBM, PDO_INFORMIX, PDO_MYSQL, PDO_OCI, PDO_ODBC, PDO_PGSQL, PDO_SQLITE, PDO_SQLSRV, PDO_4D
- možen prehod med bazami
- objektno orientirana
- varnejša kot uporaba funkcij mysql_*

PDO Primer: povezovanje na bazo

```
<?php
    try {
        $db = new PDO("mysql:host=localhost;dbname=testdb;", "username",
"password");
    } catch(PDOException $ex) {
        echo "An Error occurred: $ex->getMessage()";
    }
?>
```

PDO Primer: izvajanje poizvedbe SELECT

```
<?php
    try {
        $query = $db->query("SELECT name, date FROM table");
        foreach($query as $row) {
            echo $row["name"] . " " . $row["date"] . "\n";
        } catch(PDOException $ex) {
            echo "An Error occurred: $ex->getMessage()";
        }
    }
?>
```

PDO Primer: izvajanje poizvedbe SELECT s parametrom

```
<?php
    try {
        $query = $db->prepare("SELECT name, date FROM table WHERE name = ?");
        $query->bindValue(1, "John");
        $query->execute();
        foreach($query->fetchAll() as $row) {
            echo $row["name"] . " " . $row["date"] . "\n";
        } catch(PDOException $ex) {
            echo "An Error occurred: $ex->getMessage()";
        }
    }
?>
```

PDO Primer: vstavljanje

```
<?php
    try {
        $query = $db->prepare("INSERT INTO table (name, date) VALUES (?, ?)");
        $query->bindValue(1, "Jane");
        $query->bindValue(2, "2015-11-26");
        $query->execute();
    } catch(PDOException $ex) {
        echo "An Error occurred: $ex->getMessage()";
    }
?>
```

PDO Primer: brisanje in parametri z oznakami

```
<?php
    try {
        $query = $db->prepare("DELETE FROM table WHERE id = :id");
        $query->bindValue(":id", 1, PDO::PARAM_INT);
        $query->execute();
        $affected_rows = $query->rowCount();
    } catch(PDOException $ex) {
        echo "An Error occurred: $ex->getMessage()";
    }
?>
```

PDO Primer: posodabljanje

```
<?php
    try {
        $query = $db->prepare("UPDATE table SET name = :name WHERE id = :id");
        $query->bindParam(":name", $new_name, PDO::PARAM_STR);
        $query->bindValue(":id", 1, PDO::PARAM_INT);
        $new_name = "John";
        $query->execute();
    } catch(PDOException $ex) {
        echo "An Error occurred: $ex->getMessage()";
    }
?>
```

XAMPP in MySQL

- MySQL nadomesti MariaDB
- del osnovne namestitve XAMPP paketa
- dodatno je nameščena tudi PHP aplikacija `phpMyAdmin`
 - spletna aplikacija, ki omogoča delo s podatkovno bazo
 - <http://www.phpmyadmin.net>
- lokalno dosegljiva na naslovu <http://localhost/phpmyadmin>
- prijavimo se kot uporabnik `root` brez gesla