

DCGI

DEPARTMENT OF COMPUTER GRAPHICS AND INTERACTION

WA1 (TW1)

cvičení 6

Přístup k databázi, PDO

1. Organizace, seznámení s prostředím, HTTP, HTML
2. CSS
3. Skriptování na straně klienta
4. Jazyk PHP
5. Obsluha fomulářů, udržení stavu aplikace (sezení)
- 6. Přístup k databázi, PDO**
7. OOP v PHP
8. MVC, Smarty
9. MVC - pokračování
10. Ajax a PHP
11. Autentizace a autorizace
12. Pear a Zend
13. Odevzdávání semestrálních úloh, zápočet

- **Organizace, seznámení s prostředím, HTTP, HTML**
- **CSS**
- **Skriptování na straně klienta**
- **Jazyk PHP**
- **Obsluha formulářů, udržení stavu aplikace (sezení)**
- **Přístup k databázi, PDO**
- **OOP v PHP**
- MVC, Smarty
- MVC - pokračování
- Ajax a PHP
- Autentizace a autorizace
- Pear a Zend

- <http://www.php.net/manual/en/>
- <http://dev.mysql.com/doc/refman/5.1/en/index.html>

Domácí úkol

- Kontrola

1. Co je to relace? Základní typy relací v relačních databázích.
2. První tři normální formy? Příklad.
3. Co znamená SQL a na co se dělí?
4. Co jsou to transakce? ACID
5. Co je to SQL injection?

- nulová NF: alespoň jeden sloupec (atribut), který může obsahovat více druhů hodnot.
- první NF: pokud všechny sloupce (atributy) nelze dále dělit na části nesoucí nějakou informaci neboli prvky musí být atomické.
- druhá NF: obsahuje pouze atributy (sloupce), které jsou závislé na celém klíči.
- třetí NF: neexistují žádné závislosti mezi neklíčovými atributy (sloupci).

Normální formy nultá NF

Číslo letu	Odkud	Kam	Rozvrh	
OK 101	Praha	Bratislava	Út	09:00
			Ne	16:10
OK 101	Praha	Bratislava	St	14:00
OK 101	Praha	Bratislava	So	18:00
OK 800	K.V	Moskva	Čt	08:00
OK 800	K.V	Moskva	Út	09:00

Normální formy první NF

Číslo	Odkud	Kam	Den zkr.	Den	Čas
OK 101	Praha	Bratislava	Út	Úterý	09:00
OK 101	Praha	Bratislava	Ne	Neděle	16:00
OK 101	Praha	Bratislava	St	Středa	14:00
OK 101	Praha	Bratislava	So	Sobota	18:00
OK 800	K.V	Moskva	Čt	Čtvrtek	08:00
OK 800	K.V	Moskva	Út	Úterý	09:00

Normální formy druhá NF

Číslo letu	Den zkr.	Den	Čas
OK 101	Út	Úterý	09:00
OK 101	Ne	Neděle	16:00
OK 101	St	Středa	14:00
OK 101	So	Sobota	18:00
OK 800	Čt	Čtvrtek	08:00
OK 800	Út	Úterý	09:00

I:N

Číslo letu	Odkud	Kam
OK 101	Praha	Bratislava
OK 800	K.V	Moskva

Normální formy třetí NF

Číslo letu	Den zkr.	čas
OK 101	Út	09:00
OK 101	Ne	16:00
OK 101	St	14:00
OK 101	So	18:00
OK 800	Čt	08:00
OK 800	Út	09:00

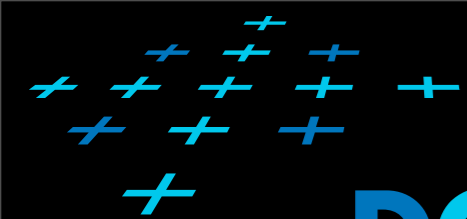
Číslo letu	Odkud	Kam
OK 101	Praha	Bratislava
OK 800	K.V	Moskva

Den zkr.	Den
Po	Pondělí
Út	Úterý
St	Středa
Čt	Čtvrtek
Pá	Pátek
So	Úterý
Ne	Neděle

I:N

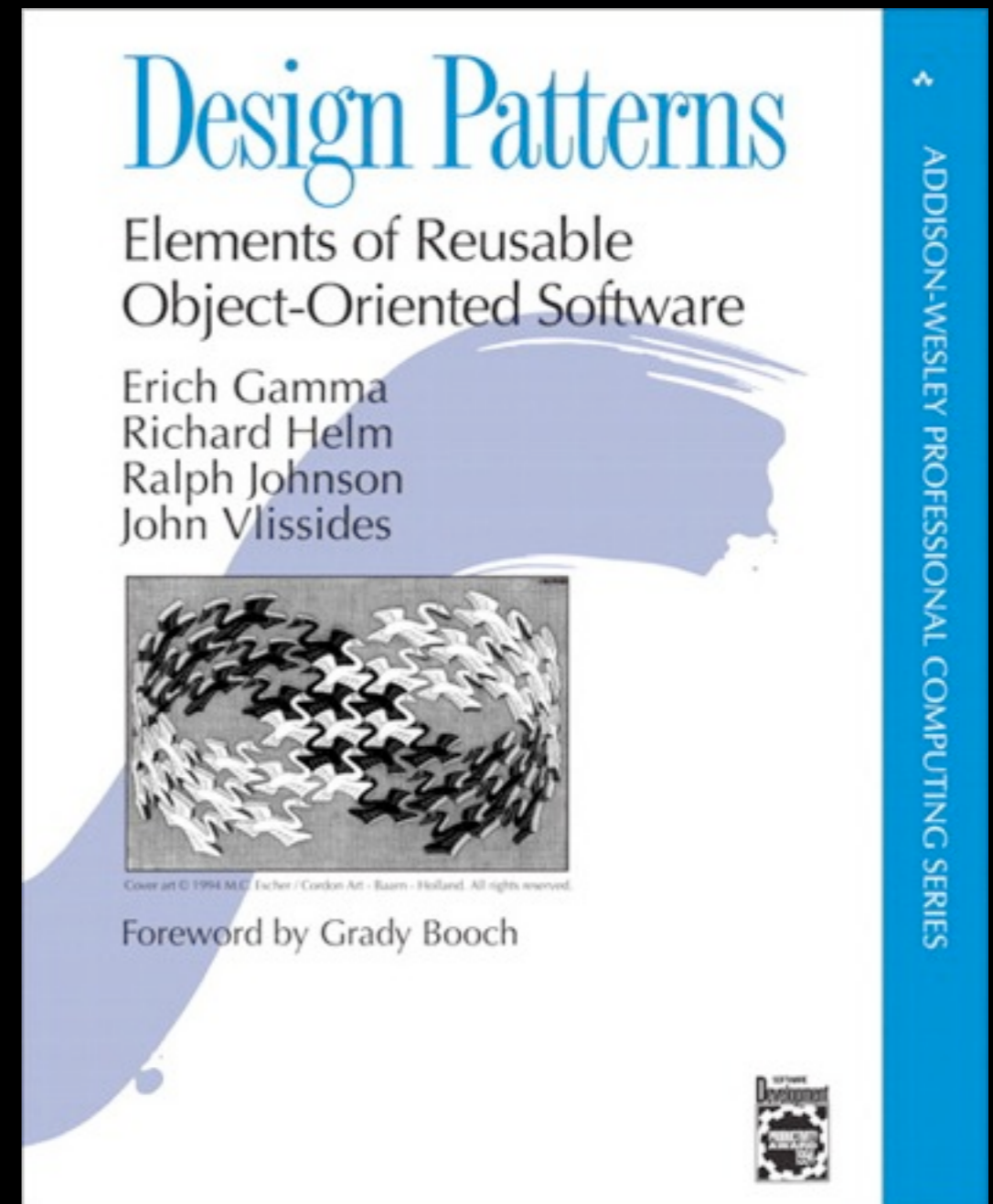
I:N

- Co je to PDO?
- Co jsou to návrhové vzory?
- Co je SINGLETON?
- Co je FACTORY?



- PHP Data Objects
- Abstraktní přístup k DB
- Implementace pro nejběžnější DB
- Podpora transakcí
- Objektový přístup

- Obecně: znovupoužitelné obecné řešení pro nějaký často se vyskytující problém
- v kontextu OOP se myslí kniha Design Patterns: Elements of Reusable Object-Oriented Software
- Gamma, Helm, Johnson, Vlissades (gang od four)



Návrhový vzor Singleton

- V matematice “singleton” je množina, které má právě jeden prvek
- V OOP je o omezení tvoření instancí tak, že třídě odpovídá právě jeden objekt

```
public class Singleton {  
    private static final Singleton INSTANCE = new Singleton();  
    // Private constructor prevents instantiation from other classes  
    private Singleton() {}  
    public static Singleton getInstance() {  
        return INSTANCE;  
    }  
}
```

Návrhový vzor Factory

- Koncept továrny pro vytváření objektů
- Když chceme získat instanci, ale nechceme řešit, jak se vytvoří

```
class Complex
{
    public static Complex fromCartesian(double real, double imag)
    {
        return new Complex(real, imag);
    }
    public static Complex fromPolar(double modulus, double angle)
    {
        return new Complex(modulus * cos(angle), modulus * sin
(angle));
    }
    private Complex(double a, double b) {}
}
Complex c = Complex.fromPolar(1, pi);
```



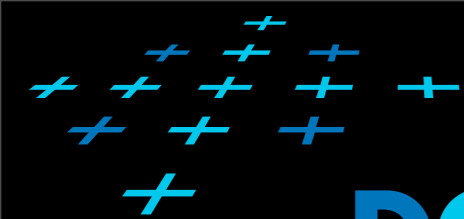

Připojení k MySQL

- viz: <http://webdev.felk.cvut.cz/userinfo/start#mysql>

- implementujte nástěnku, které bude jako úložiště zpráv sloužit MySQL databáze

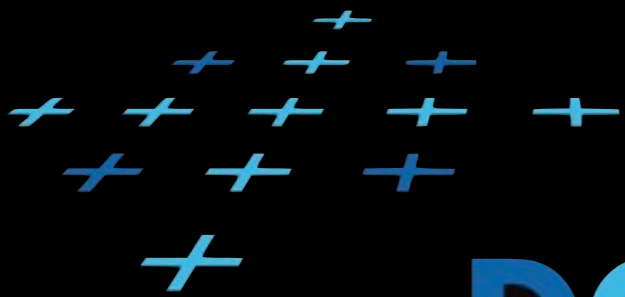
```
DROP TABLE IF EXISTS `x36tv1_record`;  
  
CREATE TABLE IF NOT EXISTS `x36tv1_record` (  
  `id` INT NOT NULL AUTO_INCREMENT ,  
  `title` VARCHAR(100) NOT NULL ,  
  `author` VARCHAR(100) NULL ,  
  `date` DATETIME NOT NULL ,  
  `text` TEXT NULL ,  
  PRIMARY KEY (`id`) )  
ENGINE = InnoDB  
DEFAULT CHARACTER SET = utf8  
COLLATE = utf8_czech_ci;
```

- Každý příspěvek na nástěnku má
 - titulek (povinný),
 - jméno autora (nepovinné),
 - datum vzniku (povinné),
 - text (nepovinný).
- Implementujte nejprve pomocí rozšíření mysqli (nebo mysql) a následně pomocí PDO následující funkce, nezapomeňte ošetřit možné chyby.



1. spojení se s databázovým serverem, volba databáze
2. zobrazení seznamu záznamů,
`SELECT * FROM `x36www_record``
3. zobrazení seznamu záznamů se stejným titulkem,
`SELECT * FROM `x36www_record` WHERE `title` = 'titulek'`
4. přidání příspěvku pomocí formuláře.
`INSERT INTO `x36www_record` VALUES(0, 'titulek', 'autor', casova_znamka, 'text')`
5. odpojení se od databázového serveru
6. Ošetřete situace, kde by mohlo docházet k útoku SQL injection.

1. Pro každé rozšíření zapouzdřete funkce pro navázání a ukončení spojení do třídy.
2. Spojení s databází bude možné navázat jen jedenkrát (návrhový vzor singleton).
3. Nemožnost navázání spojení indikujte vyhozením vlastní výjimky `DatabaseMysqlException`, která bude obsahovat chybovou zprávu a chybový kód vrácený databázovým strojem.



DCGI

DEPARTMENT OF COMPUTER GRAPHICS AND INTERACTION

Q&A

macikmir@fel.cvut.cz