

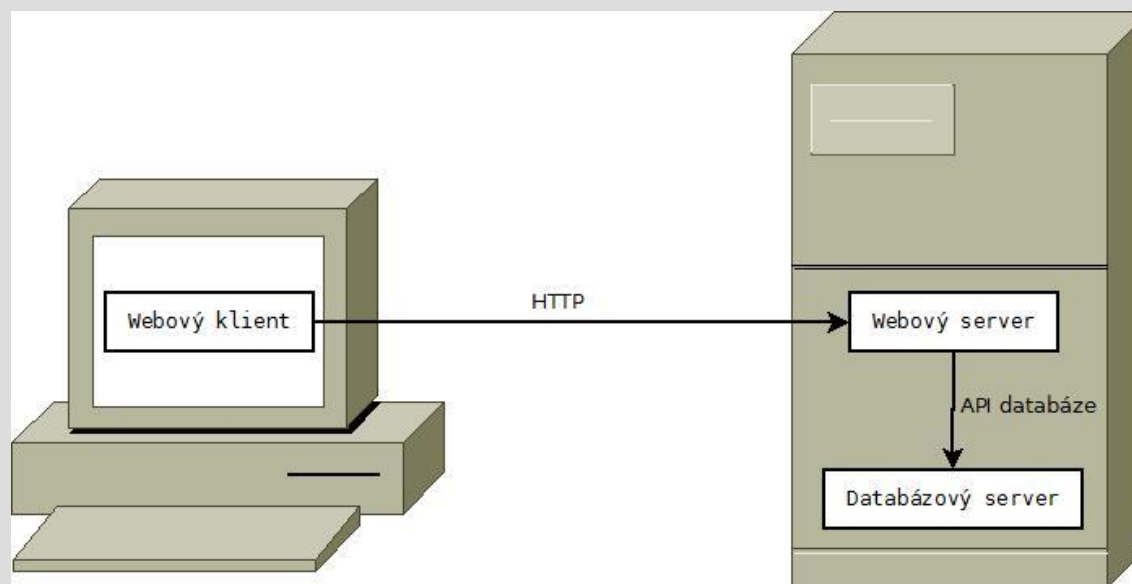


## Přednáška 5: Zpřístupnění databází přes webové rozhraní pomocí PHP

### *Architektura webové databázové aplikace*

#### **Webová databázová aplikace je realizována na třech úrovních:**

- klient, tj. www prohlížeč uživatele
- server, webová aplikace běžící na webovém serveru s podporou pro konektivitu do zvolené databáze (např. aplikace v prostředí PHP běžící na serveru Apache)
- databázový server (např. MySQL)



#### **Podpora konektivity k databázím v PHP**

PHP podporuje různá databázová rozhraní (API). Tato API umožňují přístup k velkému množství databází (např. MySQL, PostgreSQL, MSSQL, apod.).



## Přednáška 5: Zpřístupnění databází přes webové rozhraní pomocí PHP

### *Podpora API MySQL v PHP (1/4)*

#### MySQL funkce

*resource* **mysql\_connect** (*[string server]*, *[string uživatel]*, *[string heslo]*)

**Vytváří spojení** s MySQL databází.

parametry:

**server** – název databázového serveru

**uživatel, heslo** – jméno a heslo uživatele s přístupovými právy k databázi

např.

```
<?php
```

```
$link = mysql_connect('localhost', 'uživatel', 'heslo');
```

```
if (!$link) {
```

```
    die('Spojeni neuskutecneno: ' . mysql_error());
```

```
}
```

```
echo 'Uspesne spojeno';
```

```
mysql_close($link);
```

```
?>
```

Funkce vrací v případě úspěchu identifikátor spojení (kladné celé číslo) jinak vrací false. Tento identifikátor spojení se použije ve všech voláních funkcí, které odesílají požadavky na server MySQL. Spojení je automaticky uzavřeno po dokončení běhu skriptu, pokud není explicitně uzavřeno dříve funkcí `mysql_close()`.



## Přednáška 5: Zpřístupnění databází přes webové rozhraní pomocí PHP

### Podpora API MySQL v PHP (2/4)

*boolean* **mysql\_close** (*[resource link]*)

**Ukončí spojení** s MySQL databází.

parametry:

**link** – identifikátor spojení

*boolean* **mysql\_select\_db**(*string databáze* [, *resource link*])

**Vybírá databázi**, která má být **aktivní**

parametry:

**databáze** – název databáze, která se má stát aktivní

**link** – identifikátor spojení

*resource* **mysql\_query** (*string dotaz* [, *resource link*])

Odesílá **SQL dotaz** na server.

parametry:

**dotaz** – **SQL dotaz**, který má být proveden nad dříve zvolenou databází

**link** – identifikátor spojení

*mysql\_escape\_string* (*string \$unesescaped\_string*): *string*

Provede tzv. **escapování** řetězce.

*mysql\_real\_escape\_string* (*string \$unesescaped\_string*, *resource \$link\_identifier = NULL*): *string*

Alternativně k předchozí funkci s identifikátorem spojení.

U příkazů **SELECT, SHOW, DESCRIBE, EXPLAIN** a dalších (vracejících *vysledek*), funkce vrátí *resource vysledek* v případě úspěchu nebo *false* v případě chyby.

U ostatních typů SQL příkazů, **INSERT, UPDATE, DELETE, DROP**, atd., funkce vrátí *true* v případě úspěchu nebo *false* v případě chyby.



## Přednáška 5: Zpřístupnění databází přes webové rozhraní pomocí PHP

### *Podpora API MySQL v PHP (3/4)*

*int **mysql\_num\_rows** (resource **vysledek**)*

Vrací **počet řádků** v identifikátoru výsledku.

parametry:

**vysledek** – identifikátor výsledku vrácený funkcí **mysql\_query**

*int **mysql\_insert\_id** ([resource **link**])*

Vrací **ID sloupce** s nastaveným **AUTO\_INCREMENT** atributem po provedení posledního SQL dotazu typu **INSERT**.

parametry:

**link** – identifikátor spojení

*resource **mysql\_fetch\_row** (resource **vysledek**)*

Vrací **řádek** z identifikátoru výsledku jako **číslované pole**.

parametry:

**vysledek** – identifikátor výsledku vrácený funkcí **mysql\_query**

*resource **mysql\_fetch\_assoc** (resource **vysledek**)*

Vrací **řádek** z identifikátoru výsledku jako **asociativní pole**.

parametry:

**vysledek** – identifikátor výsledku vrácený funkcí **mysql\_query**



## Přednáška 5: Zpřístupnění databází přes webové rozhraní pomocí PHP

### *Podpora API MySQL v PHP (4/4)*

*resource* **mysql\_fetch\_array** (*resource* **vysledek** [, *int* **typ\_vysledku**])

Vrací **řádek** z identifikátoru výsledku jako **asociativní pole**, **číslované pole** nebo **obojí** (výchozí stav).  
parametry:

**vysledek** – identifikátor výsledku vrácený funkcí **mysql\_query**

**typ\_vysledku** – nepovinný parametr, nabývá těchto hodnot:

- MYSQL\_NUM: číslované pole (obdoba `mysql_fetch_row`)
- MYSQL\_ASSOC: asociativní pole (obdoba `mysql_fetch_assoc`)
- MYSQL\_BOTH: obojí typy indexů (výchozí stav)

*boolean* **mysql\_data\_seek** (*resource* **vysledek**, *int* **cislo\_radku**)

**Posune** interní **ukazatel** na řádek identifikátoru výsledku.

parametry:

**vysledek** – identifikátor výsledku vrácený funkcí **mysql\_query**

**cislo\_radku** – požadované číslo řádku, na který se chceme posunout

*int* **mysql\_errno** (*[resource* **link**])

Vrací **číslo chyby** poslední operace s MySQL databází.

*string* **mysql\_error** (*[resource* **link**])

Vrací **chybovou zprávu** poslední operace s MySQL databází.

parametry:

**link** – identifikátor spojení



## Přednáška 5: Zpřístupnění databází přes webové rozhraní pomocí PHP

### *Komentované příklady a techniky práce s MySQL funkcemi (1/3)*

Následující příklad ilustruje práci s daty v tabulce **studenti**:

```
<?php
// vytvorime spojeni k databazi
$link = mysql_connect('localhost', 'uzivatel', 'heslo');

// vybereme konkretni databazi
mysql_select_db('databaze');

// vytvorime dotaz
$query = '
    SELECT *
    FROM studenti;
';

// odesleme dotaz do databaze
$result = mysql_query($query);

// projdeme vysledek dotazu v cyklu
while ($row = mysql_fetch_array($result)) {

    echo $row['jmeno'] . ' ' . $row['prijmeni'] . '<br>';

}
// uzavreme spojeni s databazi
mysql_close($link);
?>
```



## Přednáška 5: Zpřístupnění databází přes webové rozhraní pomocí PHP

### *Komentované příklady a techniky práce s MySQL funkcemi (2/3)*

Část předchozího příkladu **s ošetřením chyb:**

```
<?php
...
// odesleme dotaz do databaze
$result = mysql_query($query);
if (mysql_errno()) {
    echo "MySQL error " . mysql_errno() . ": " . mysql_error() . "<br>Behem provadeni SQL dotazu:<br>$query<br>";
} else {

    // projdeme vysledek dotazu v cyklu
    while ($row = mysql_fetch_array($result)) {

        echo $row['jmeno'] . ' ' . $row['prijmeni'] . '<br>';

    }

}

?>
```



## Přednáška 5: Zpřístupnění databází přes webové rozhraní pomocí PHP

### *Komentované příklady a techniky práce s MySQL funkcemi (3/3)*

Opakované vytváření databázového spojení lze řešit např. samostatným skriptem:

```
<?php
$server = 'localhost';
$uzivatel = 'uzivatel';
$heslo = 'heslo';
$databaze = 'databaze';

// vytvorime spojeni k databazi
$link = mysql_connect($server, $uzivatel,
$heslo);
if (!$link) {
    die('Spojeni neuskuteceno: ' . mysql_error());
}

// vybereme konkretni databazi
$db_selected = mysql_select_db($databaze);
if (!$db_selected) {
    die('Databaze nenalezena: ' . mysql_error());
}
mysql_query('SET CHARACTER SET UTF8;'); //
komunikace mezi klientem a serverem bude
probihat v utf-8
?>
```

Vložení pomocného skriptu do hlavního skriptu

```
<?php
require('db.php');
...
mysql_close();
?>
```

Případně funkcí, která např. vrací ident. spojení:

```
<?php
function db() {
    $server = 'localhost';
    ...
    // vytvorime spojeni k databazi
    $link = mysql_connect($server, $uzivatel, $heslo);
    ...
    return $link;
}

<?php
require('db.php');
$link = db();
...
mysql_close($link);
?>
```