

Lokal Databas

Förutsättning: Du har installerat till exempel XAMPP med MariaDB och vill koppla upp dig mot den. Webbserver och databasserver är startade via XAMPP:s kontrollpanel.

Förberedelse första gången

1. I webbläsaren skriv in adressen localhost/phpmyadmin
2. I phpmyadmin, om du inte gjort det så skapa upp en databas, kalla den **cms**
3. I phpmyadmin, skapa upp de tabeller du tänkt använda dig av eller importera från den sql-fil som finns i Canvas (d0019e_blog.sql)
 - a. För att importera markera databasen (cms) och välj import, bläddra fram filen som du laddat ner från canvas och klicka på Go
4. I phpmyadmin, kan man även skapa upp några rader i tabellerna för att testa med
 - a. Välj databas (cms) och gå till Insert
 - b. Skriv in de värden som krävs och klicka på Go

Inloggningsinformation

Nu är du redo att koppla upp dig till databasen med php-kod, den information som behövs för uppkopplingen ska fungera är:

- Databasserver: localhost
- Användarnamn är default: root
- Lösenord är default inget, så ange tom sträng
- Databasnamnet är det namn du valt i punkt 3 ovan

Skapa en fil db_credentials.php och definiera inloggningsuppgifterna på följande sätt:

```
<?php
define("DB_SERVER", "localhost");
define("DB_USER", "root");
define("DB_PASS", "");
define("DB_NAME", "cms");
```

I db.php inkluderas db_credentials.php och konstanterna används för att koppla upp mot databasen.

Skapa och kör frågor mot databasen

När man har en fråga som man vill kunna köra mot databasen så skapar man en funktion, låt oss säga att vi vill hämta användarnamn för alla användare som är sparade i databasen, vi väljer ut id och usernamn som de kolumner som vi vill ha tillbaka.

```
function get_users()
{
    global $connection;
    $sql = 'SELECT id, username FROM user ORDER BY created';
    $statement = mysqli_prepare($connection, $sql);
    mysqli_stmt_execute($statement);
    $result = get_result($statement);
    mysqli_stmt_close($statement);
    return $result;
}
```

Notera att `get_result` är en funktion som finns i `db.php`, det anropet kan ersättas med:

```
$result = mysqli_stmt_get_result($statement);
```

Använder man `get_result` så läggs resultatet över till en array som innehåller alla rader som kommer från databasen. Varje rad är i sin tur en array med kolumnens namn som nyckel och fältets värde som innehåll:

```
Array
(
    [0] => Array
        (
            [id] => 1
            [username] => Anna
        )
    [1] => Array
        (
            [id] => 2
            [username] => Britta
        )
    [2] => Array
        (
            [id] => 3
            [username] => Clara
        )
    [3] => Array
        (
            [id] => 4
            [username] => Doris
        )
)
```

Vill vi då göra om arrayen till en array som har `id` som nyckel och användarnamn som värde måste vi skapa upp en ny array på följande sätt:

```
$users = Array();
foreach($result as $value)
{
    $users[$value['id']] = $value['username'];
}
```

LTU:s databasserver

När du ska använda LTU:s databas så krävs det att du gör förändringar i `db_credentials.php`. De inloggningsuppgifter som behövs får du i din studentmejl efter att du begärt ett databaskonto via: https://staff.www.ltu.se/~susfah/db_users/

Mejlet kommer att se ut som följer:

```
Hej,
här kommer dina inloggningsuppgifter till MySQL-servern som du ska använda i inlämningsuppgift 4
på kursen D0019E, Webbutveckling II - Skriptspråk och databaser. Information om hur du praktiskt
går tillväga för att få det att fungera finns i Canvas.

# DB_SERVER: utbweb.its.ltu.se:3308
# DB_USER: qwerty0
# DB_PASS: zy5Lavun
# DB_NAME: D0019E_V22_qwerty0
```

Redigera `db_credentials.php` med de uppgifter som kommer i ditt mejl. Notera att mejlet kommer till **studentmejl** och inte till inboxen i Canvas.

Exportera databasen och importera till LTU-databasen

Exportera via phpmyadmin

1. Markera databasen
2. Välj Export i menyn
3. Klicka på Export.

Filen som skapas kommer få databasens namn, tex. cms.sql

OBS, jag fick nätverksfel när jag försökte exportera min databas. Det visade sig bero på min version av phpmyadmin (5.1.3), så stöter du på det problemet så uppdatera phpmyadmin. Jag följde denna instruktion som fungerade bra:

<https://stackoverflow.com/questions/27131029/how-to-upgrade-phpmyadmin-in-xampp-to-latest>

Importera med hjälp av db.php funktionen db_import()

1. Skapa en sida import.php som ser ut som nedan
2. Kontrollera att db_credentials.php innehåller inloggningsuppgifter till LTU-databasservern
3. Ladda upp sidorna till LTU:s webbserver
 - a. db.php
 - b. db_credentials.php
 - c. import.php
 - d. cms.sql (ska ligga i samma mapp som import.php)
4. Besök sidan import.php 1 gång
5. Går allt bra så skrivs endast "Importereringen är klar!" ut.
6. Nu kan du ta bort sidan import.php och cms.sql

```
<?php
require_once "db.php";
import('cms.sql');
```

Om du behöver göra om importeringen, se till så lägg till true som en andra parameter i anropet till import så tas de gamla tabellerna med innehåll bort: `import('$cms.sql', true);`

Vill eller kan man inte använda en lokal databasserver

Det går förstås bra att jobba direkt mot LTU:s databasserver. Då importerar du förslagsvis den sql-fil som finns i canvas så finns tabellerna där och du kan använda dem. Nackdelen är att det inte blir så enkelt att göra ändringar i tabellerna men de går ju att använda tabellerna som de är.

Det går förstås också att skapa upp tabellerna med hjälp av php-kod om man vill det.

Tänk på att webbsidorna som använder LTU:s databasserver måste ligga på LTU-nätet. Använder du VPN kan du ha sidorna på en lokal webbserver och ändå komma åt LTU:s databasserver.