



# WEBBPROGRAMMERING MODUL 7 – DATABASHANTERING

Mahmud Al Hakim  
[mahmud@alhakim.se](mailto:mahmud@alhakim.se)

# VAD ÄR EN DATABAS?

”En databas (tidigare databank) är en samling information som är organiserad på ett sådant sätt att det är lätt att söka efter och hämta enskilda bitar information, samt ofta även att ändra informationen.

Ordet databas kan beteckna informationen som finns lagrad, eller den programvara (databashanterare) som förstår att tolka den ofta mycket komplexa datastrukturen som lagras på hårddisken.

Databaser nås via ett s.k. gränssnitt.”

Källa: <https://sv.wikipedia.org/wiki/Databas>

# PAGINAS IT-ORDBOK

[HTTP://ITORD.PAGINA.SE/DEFAULT.ASP?ID=1276](http://itord.pagina.se/default.asp?id=1276)

En databas är en samling information som organiserats så att ett dataprogram snabbt kan hitta önskade data.

En databas är ett elektroniskt kortregister.

Traditionella databaser organiseras i fält, poster och filer. Ett fält är en enskild datamängd, en post är en uppsättning sammanhörande fält och en fil är en samling poster.

En enkel telefonkatalog kan jämföras med en fil. Den innehåller en lista med poster som var och en innehåller tre fält: namn, adress och telefonnummer.

För att hantera informationen i en databas används ett databashanteringssystem (Database Management System, DBMS). Detta är en samling program för att lägga in, organisera och hämta ut data i en databas.

Uttrycket databas används numera oftast för att beteckna ett helt databashanteringssystem.

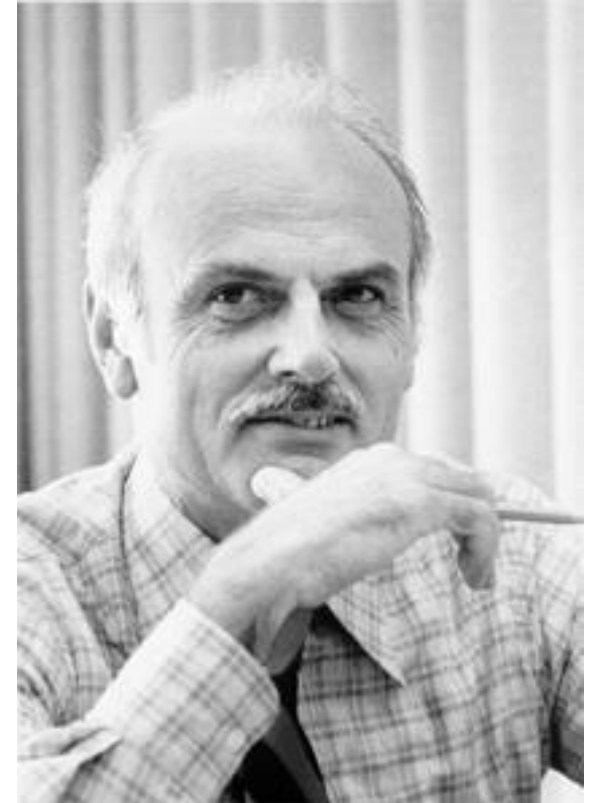
# RELATIONSDATABASES

En relationsdatabas är en databas där information ("data") är organiserad i **relationer** (även kallade **tabeller**).

Termen "relationsdatabas" definierades ursprungligen 1969 av Edgar Codd på IBM.

Källa: <https://sv.wikipedia.org/wiki/Relationsdatabas>

Bildkälla: [https://en.wikipedia.org/wiki/Edgar\\_F.\\_Codd](https://en.wikipedia.org/wiki/Edgar_F._Codd)



# EXEMPEL PÅ DATABASHANTERINGSSYSTEM (DBMS)

MS SQL Server

[www.microsoft.com/sql](http://www.microsoft.com/sql)

Oracle Database

[www.oracle.com](http://www.oracle.com)

IBM DB2

[www.ibm.com/db2](http://www.ibm.com/db2)

MySQL

[www.mysql.com](http://www.mysql.com)

MariaDB

<https://mariadb.org/>

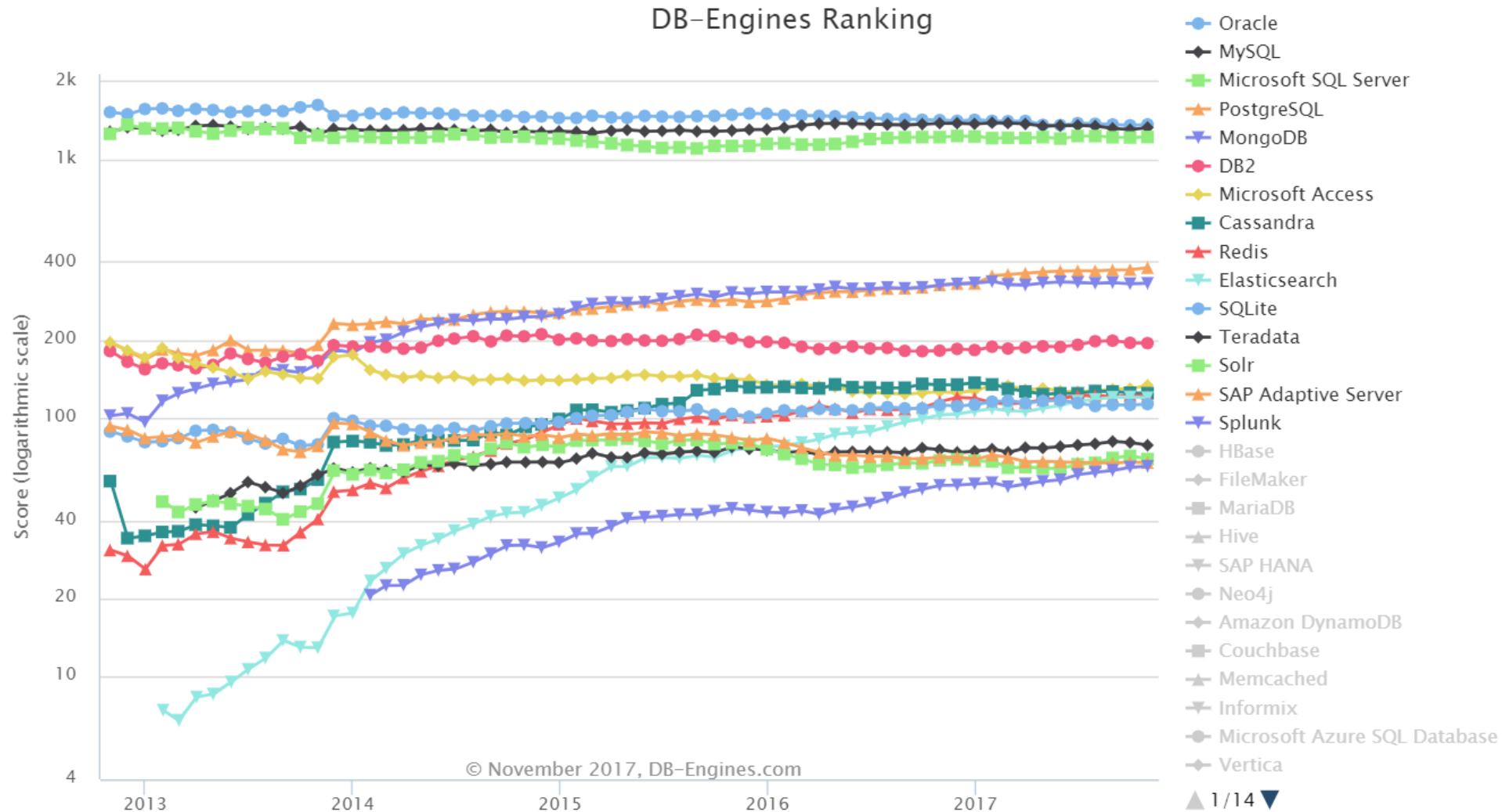
Tips:

[https://en.wikipedia.org/wiki/Comparison\\_of\\_relational\\_database\\_management\\_systems](https://en.wikipedia.org/wiki/Comparison_of_relational_database_management_systems)



# DB-ENGINES RANKING

[HTTP://DB-ENGINES.COM/EN/RANKING](http://db-engines.com/en/ranking) TREND



# VAD ÄR MYSQL?

MySQL är en databashanterare. Den använder sig av frågespråket SQL.

MySQL är fri programvara, licensierad under GNU General Public License.

Programmet skrevs och underhölls före 2008 av det svenska företaget MySQL AB i Uppsala. De sålde support och servicekontrakt såväl som kommersiella licensierade kopior av MySQL.

Programmets huvudsakliga utvecklare var finlandssvenske Michael Widenius och David Axmark. Grundarna arbetar numera med MariaDB.

2008 köptes MySQL AB av Sun Microsystems för en miljard dollar och den 27 januari 2010 köptes Sun upp av Oracle för 8,5 miljarder amerikanska dollar.

Källa <https://sv.wikipedia.org/wiki/MySQL>

[https://en.wikipedia.org/wiki/Michael\\_Widenius](https://en.wikipedia.org/wiki/Michael_Widenius)



# VAD ÄR MARIADB?

MariaDB är en fortsatt utvecklad gren av MySQL.

Programkoden går under GNU General Public License, i motsats till den numera Oracle-ägda MySQL, där uppdateringar skyddas i allt större utsträckning genom upphovsrätt.

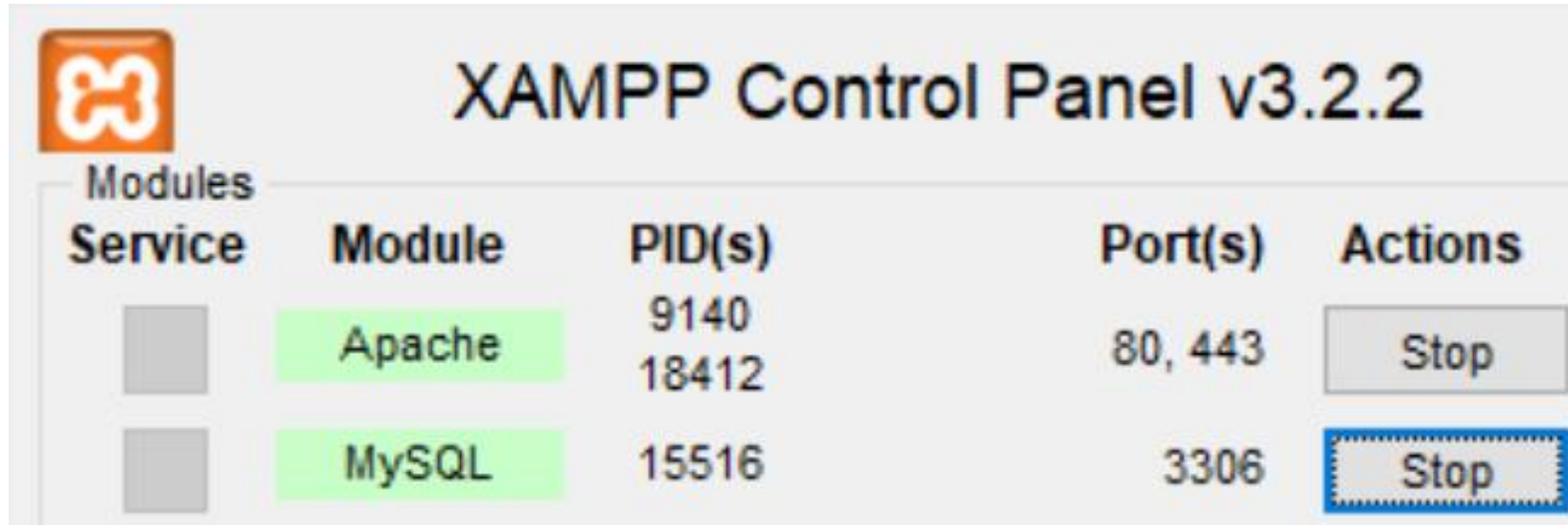
Bakgrunden är att Oracle inte verkar ha några planer för MySQL's framtida livscykel.

Bakom MariaDB står de ursprungliga skaparna till MySQL, Michael Widenius, David Axmark och Allan Larsson.

Källa: <https://sv.wikipedia.org/wiki/MariaDB>



# STARTA MYSQL/MARIADB



The image shows the XAMPP Control Panel v3.2.2 interface. It features a table with columns for Service, Module, PID(s), Port(s), and Actions. The MySQL service is listed with a green background, indicating it is running. The Actions column for MySQL has a 'Stop' button highlighted with a blue dashed border. A large blue arrow points to this 'Stop' button.

Service	Module	PID(s)	Port(s)	Actions
<input type="checkbox"/>	Apache	9140 18412	80, 443	Stop
<input type="checkbox"/>	MySQL	15516	3306	Stop

# VAD ÄR SQL?

SQL står för Structured Query Language.

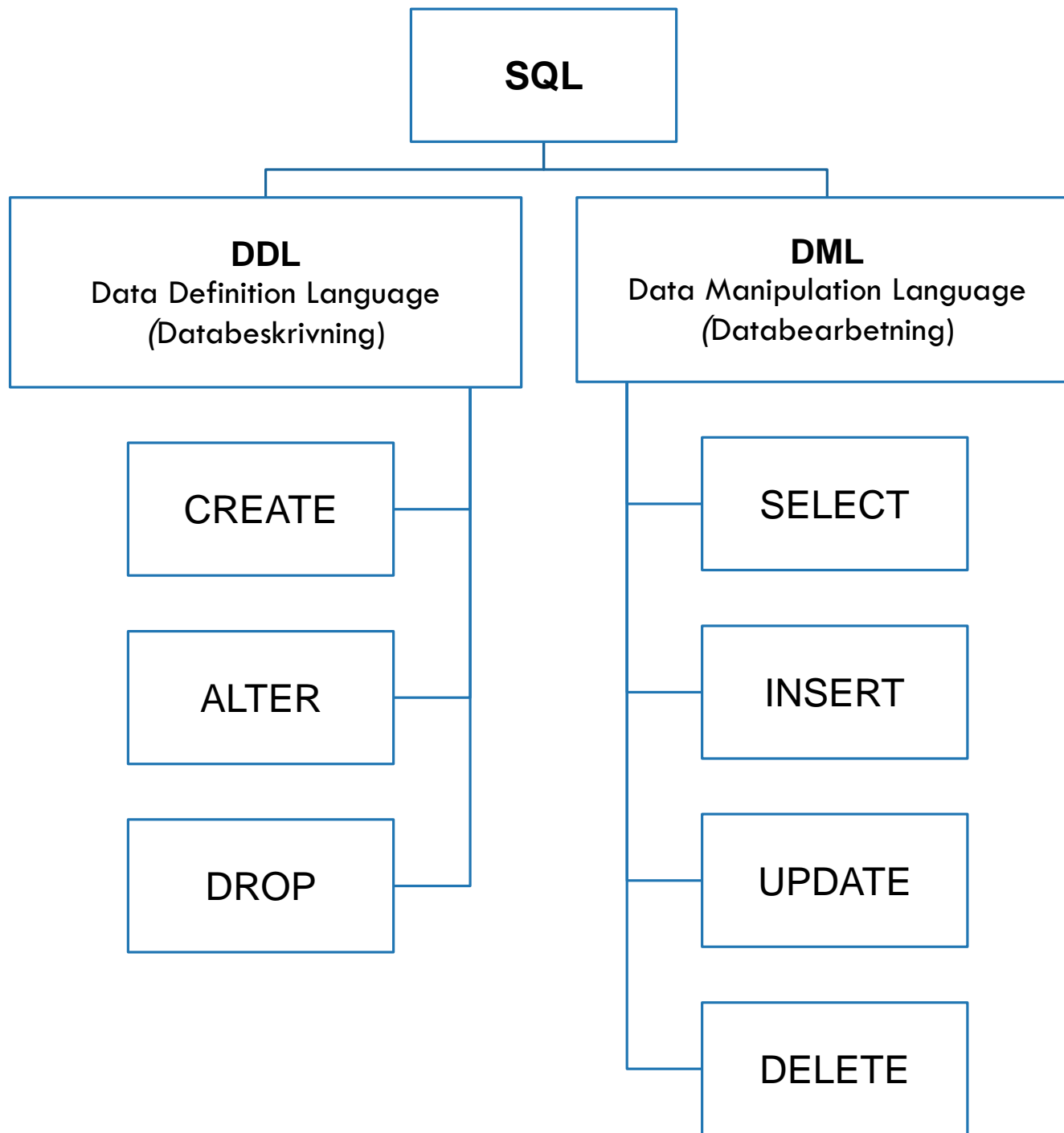
Under 1970 utvecklade IBM "Structured English Query Language" (SEQUEL).

SEQUEL byttes senare ut mot SQL eftersom 'SEQUEL' var ett varumärke.

SQL är ett standardiserat frågespråk för att hämta och modifiera data i en relationsdatabas.

Ett frågespråk är ett språk som man använder för att ställa frågor till en databashanterare, dvs göra sökningar i en databas.

SQL kan även användas för att ändra strukturen på databasen t.ex. lägga till och ta bort tabeller.



# SQL SYNTAX

SQL-frågor består av ett antal satser (statements).

SQL är INTE “Case Sensitive”.

SQL KOMMANDON skrivs vanligtvis i VERSALER.

# LOGGA IN I MYSQL CONSOLE (MONITOR)

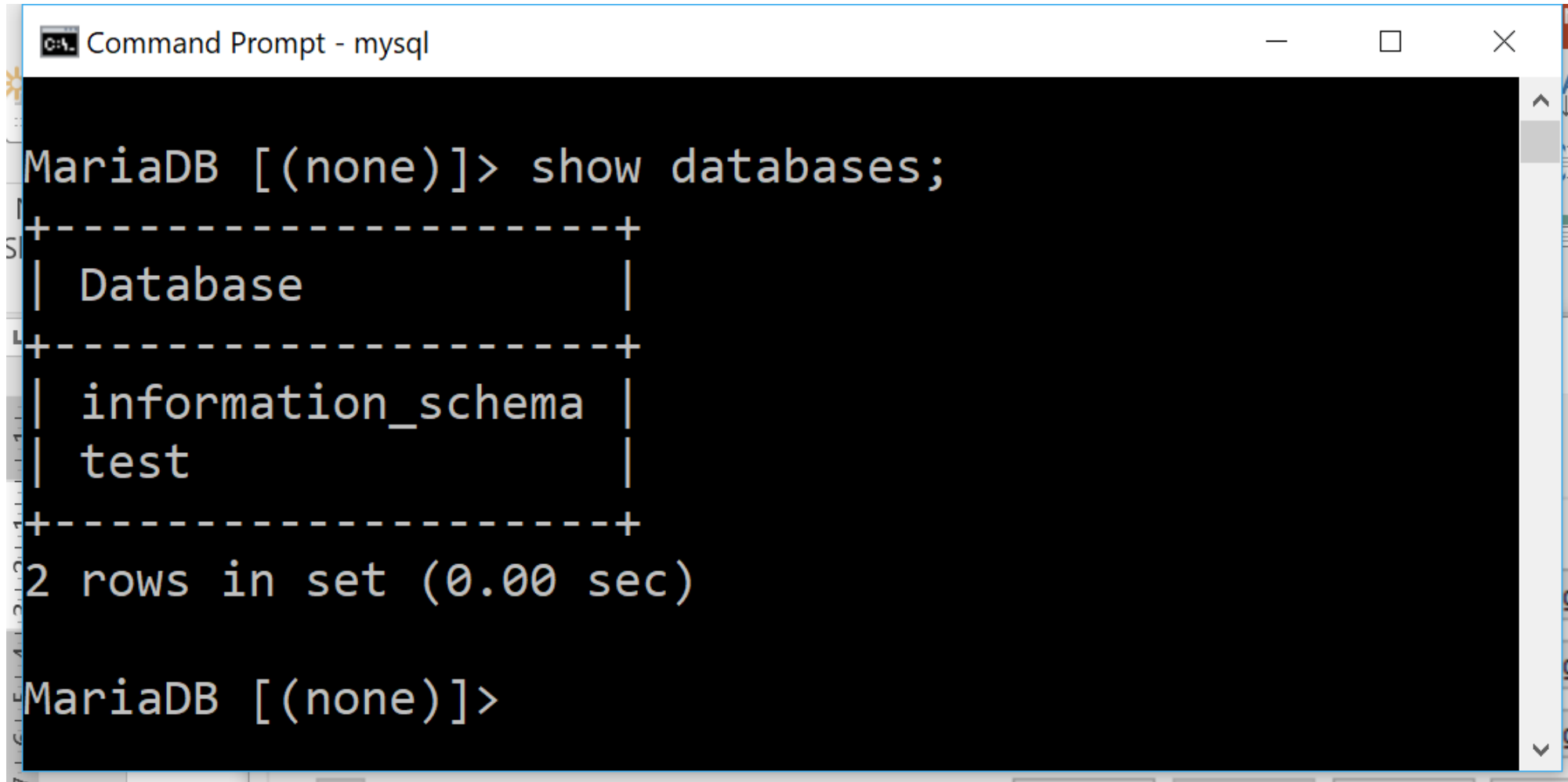


Command Prompt - mysql -u root -p



```
C:\xampp\mysql\bin>mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands
end with ; or \g.
```

# VISA ALLA DATABASER MED KOMMANDOT SHOW DATABASES;



```
Command Prompt - mysql

MariaDB [(none)]> show databases;
+-----+
| Database                |
+-----+
| information_schema       |
| test                     |
+-----+
2 rows in set (0.00 sec)

MariaDB [(none)]>
```

# NÅGRA VANLIGA KOMMANDON

```
CREATE DATABASE TestDatabas;
```

```
USE TestDB;
```

```
CREATE TABLE TestTabell(id INT);
```

```
SHOW TABLES;
```

```
INSERT INTO TestTabell VALUES(1), (2), (3);
```

```
SELECT * FROM TestTabell;
```

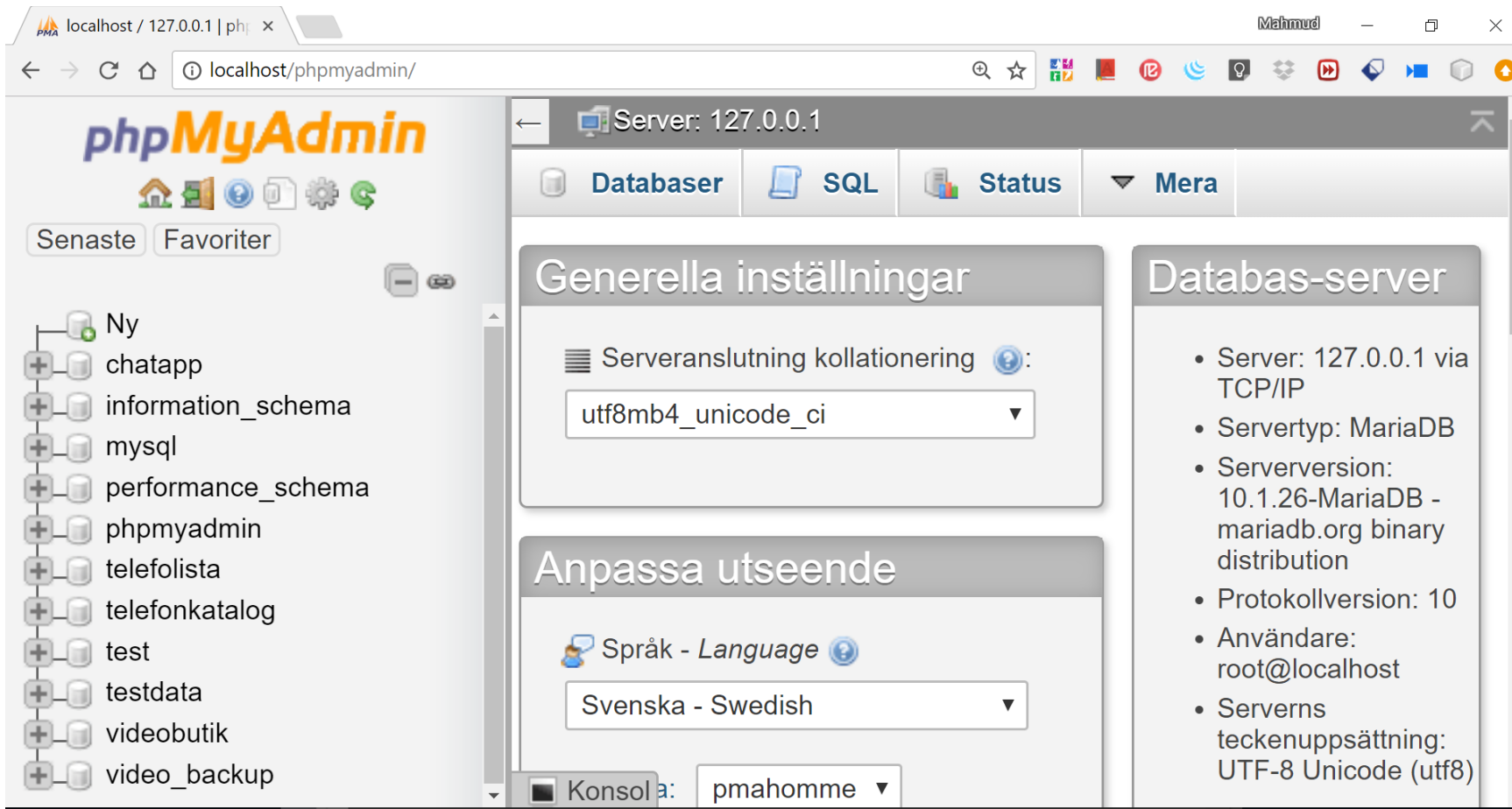
```
DROP DATABASE TestDatabas;
```

Testa dessa satser direkt  
i MySQL Console  
Mer SQL senare i kursen.

# phpMyAdmin

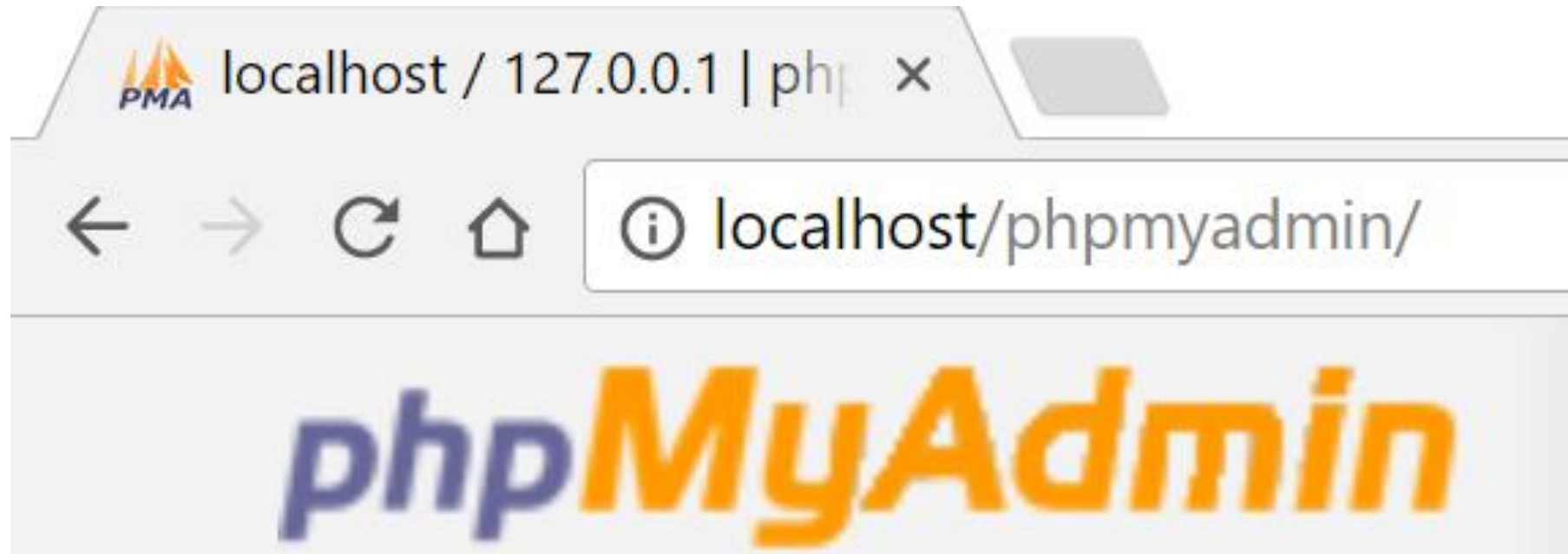
phpMyAdmin är en grafisk webbapplikation skrivet i PHP.

phpMyAdmin används för att administrera MySQL/MariaDB via en webbläsare.





# STARTA PHPMYADMIN LOKALT



# EN PRAKTISK ÖVNING

**Vi ska utveckla en databasbaserad telefonlista**

## **Steg 1**

Databasdesign (grunden till alla webbapplikationer)

## **Steg 2**

Användargränssnitt (Frontend)

## **Steg 3**

Databaskopplingar (Backend)

# STEG 1 – DATABASDESIGN

← Server: 127.0.0.1

Databaser SQL Status Användarkonton Exportera

## Databaser

Skapa databas ?

telefonlista utf8\_unicode\_ci ▼ Skapa

Ange  
Databasnamn

Välj  
Kollationering  
(Teckenkodning)

# SKAPA EN NY TABELL

← Server: 127.0.0.1 » Databas: telefonlista

Struktur SQL Sök Fråga Exportera Imp

⚠ Inga tabeller funna i databasen.

Skapa tabell

Namn:  Antal kolumner:

Skriv tabellnamn

Ange antal kolumner

# TABELLSTRUKTUR

Klicka här för mer  
info om datatyper

Namn

Typ ?

Längd/Värden ?

namn

VARCHAR ▼

50

Välj från huvudkolumn

telefon

VARCHAR ▼

50

Välj från huvudkolumn

Ange  
kolumnnamn

Välj Typ  
t.ex. VARCHAR

Ange längd







# PRIMÄRNYCKEL (PRIMARY KEY)


En primärnyckel är en kolumn, eller en kombination av kolumner, som alltid har ett unikt värde för varje rad i tabellen.

Om det finns flera möjliga primärnycklar säger man att man har flera kandidatnycklar, och man väljer en av dem som primärnyckel.

# LÄGG TILL EN PRIMÄRNYCKEL

	#	Namn	Typ	Kollationering
<input type="checkbox"/>	1	namn	varchar(50)	utf8_unicode_ci
<input type="checkbox"/>	2	telefon	varchar(50)	utf8_unicode_ci

Åtgärd
 Ändra  Radera  Primär
 Ändra  Radera  Primär

**Bekräfta** 

Vill du verkligen köra "ALTER TABLE `kontakt` ADD PRIMARY KEY(`telefon`);"?

OK

Avbryt

Klicka på **Primär** på  
fältet telefon

# LÄGG TILL DATA

Fyll i värden under  
fliken Lägg till



Bläddra



Struktur



SQL



Sök



Lägg till



Exportera



Importera



Mera

Kolumn	Typ	Funktion	nolläge	Värde
namn	varchar(50)	<input type="text"/>		Mahmud Al Hakim
telefon	varchar(50)	<input type="text"/>		070-1234567

Kör

✓ 1 rad infogad.

```
INSERT INTO `kontakt` (`namn`, `telefon`) VALUES ('Mahmud Al Hakim', '070-1234567');
```



# EXPORTERA

Bläddra Struktur SQL Sök Lägg till Exportera Importera

## Exportera rader från tabellen "kontakt"

**Exportera mallar:**

**Ny mall:**

Mallens namn

**Befintliga mallar:**

Mall:

**Exportmetod:**

☒ Snabb - visa endast minimala alternativ

☐ Anpassad - visa alla möjliga alternativ

**Format:**

Öppna fliken  
"Exportera"

Välj format

# SQL-FIL — EXEMPEL

```
CREATE TABLE `kontakt` (  
  `namn` varchar(50) COLLATE utf8_unicode_ci NOT NULL,  
  `telefon` varchar(50) COLLATE utf8_unicode_ci NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci;
```

```
INSERT INTO `kontakt` (`namn`, `telefon`) VALUES  
('Mahmud Al Hakim', '070-1234567');
```

```
ALTER TABLE `kontakt`  
ADD PRIMARY KEY (`telefon`);  
COMMIT;
```

# FLERA OPERATIONER

→ Server: 127.0.0.1 » Databas: telefonlista » Tabell: kontakt

BläddraStrukturSQLSökLägg tillExporteraImporteraPrivilegierOperationerSpårning

Flytta tabellen till (databas.tabell)

telefonlista . kontakt

☒ Lägg till AUTO\_INCREMENT-värde  
☒ Ändra privilegier

Kör

Tabellalternativ

Döp om tabellen till kontakt

☒ Ändra privilegier

Tabellkommentarer

Lagringsmotor InnoDB

Kollationering utf8\_unicode\_ci

☐ Ändra alla kolumners sortering

ROW\_FORMAT COMPACT

Kopiera tabellen till (databas.tabell)

telefonlista . kontakt

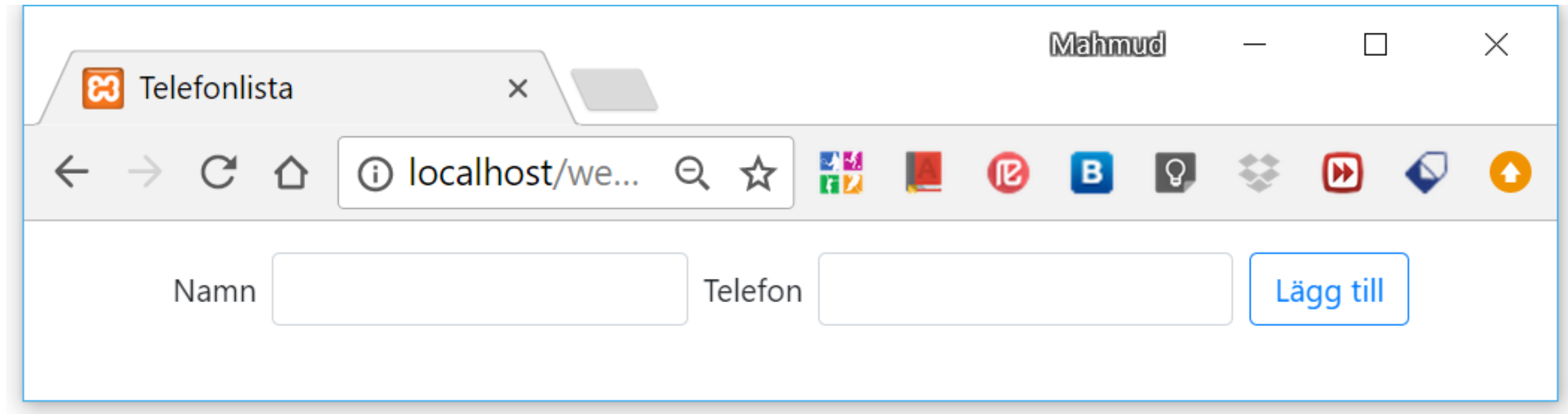
☐ Enbart struktur  
☒ Struktur och data  
☐ Enbart data

☐ Lägg till DROP TABLE  
☐ Lägg till AUTO\_INCREMENT-värde  
☒ Lägg till restriktioner

☒ Ändra privilegier  
☐ Byt till kopierad tabell

# STEG 2: ARBETA MED FRONTEND

## SKAPA ETT ANVÄNDARGRÄNSSNITT (GUI)



# FILEN INDEX.PHP

```
<body class="container">
<form class="form-inline my-3" method="post" action="insert.php" >
<label for="name">Namn </label>
<input type="text" class="form-control mx-2" id="name" name="name">
<label for="telefon">Telefon </label>
<input type="text" class="form-control mx-2" id="telefon" name="telefon">
<button type="submit" class="btn btn-outline-primary">Lägg till</button>
</form>
</body>
```

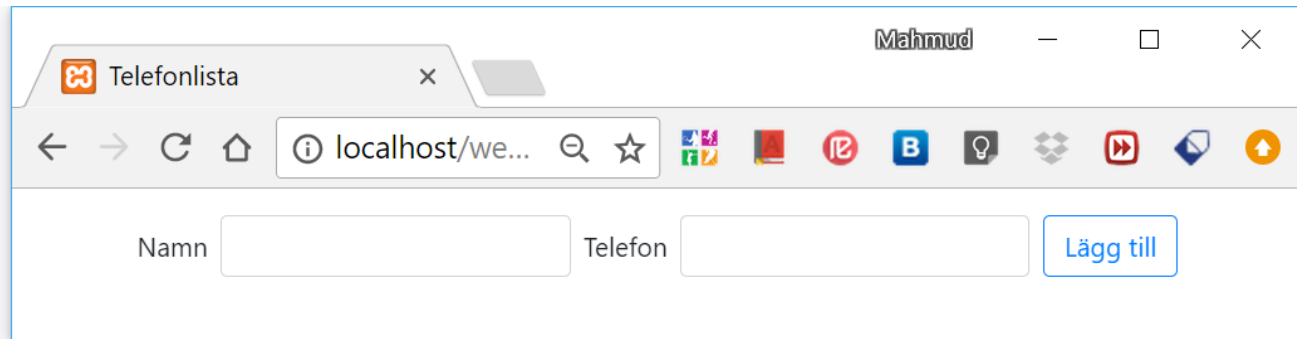
# STEG 3 – DATABASKOMMUNIKATION

Data paketeras med hjälp av metoden post och skickas vidare till filen insert.php

```
method="post" action="insert.php"
```

Filen `insert.php`

PHP-Skript loggar in i databasen och skickar data via SQL



The screenshot shows a web browser window with the title 'Mahmud'. The address bar shows 'localhost/we...'. The page content includes a form with two input fields: 'Namn' and 'Telefon', and a button labeled 'Lägg till'.

Databas

namn	telefon
Mahmud	0707556405

# FILEN INSERT.PHP — LOGGA IN I DATABASEN

```
<?php
```

```
$dbHost = "localhost" ;
```

```
$dbUser = "root" ;
```

```
$dbPwd = "" ;
```

```
$dbName = "telefonlista" ;
```

```
$connection =
```

```
mysqli_connect($dbHost, $dbUser, $dbPwd,$dbName)
```

```
or die("Fel vid connect");
```

```
// Lägg till denna rad för att lösa problem med svenska tecken.
```

```
mysqli_set_charset($connection, "utf8");
```

# FILEN INSERT.PHP — SAMLA IN DATA VIA POST OCH SKICKA VIDARE TILL DATABASEN MED SQL

```
$namn      = $_POST['name'];  
$telefon = $_POST['telefon'];  
$sql= "INSERT INTO kontakt VALUES ('$namn', '$telefon')";  
mysqli_query($connection, $sql)  
or die(mysqli_error($connection));  
echo "$namn har nu lagts till telefonlistan";  
?>
```



# ATT HÄMTA DATA FRÅN DATABASEN

```
$query = "SELECT * FROM kontakt";  
$table = mysqli_query($connection,$query)  
or die(mysqli_error($connection));  
echo "<table class='table'><tr>";  
echo "<th>Namn</th><th>Telefon</th></tr>";  
while($row = $table->fetch_assoc()){  
    echo "<tr><td>" . $row['namn'] . "</td>";  
    echo "<td>" . $row['telefon'] . "</td></tr>";  
}  
echo "</table>";  
?>
```

# ÖVNING

Lägg till flera fält i databasen, uppdatera gränssnittet och dina php-skript.

Exempel på extra fält

- E-post
- Adress

# WEBBTJÄNSTER

Webbtjänster (Web services) betecknar webbaserade datorprogram som kommunicerar och samarbetar dynamiskt med andra webbtjänster på samma vis som en människa kan surfa till olika webbsidor.

För att skicka data mellan applikationerna används protokollet HTTP.

För att beskriva applikationernas gränssnitt används XML eller JSON.

# XML — EXTENSIBLE MARKUP LANGUAGE







XML är ett märkspråk som är baserat på SGML "Standard Generalized Markup Language".

Med hjälp av XML kan du definiera egna taggar, vilket ger bättre sammanhang och struktur.

XML kan lagra data i textformat som är läsbart för människor samt ger stöd för att beskriva format för data som skall utbytas.

Ett välutformat XML-dokument kan tolkas av olika verktyg och applikationer.

# EXPORTERA DATA SOM XML VIA PHPMYADMIN

 Bläddra	 Struktur	 SQL	 Sök	 Lägg till	 Exportera
---	--	---	---	---	---

**Exportmetod:**

- ☒ Snabb - visa endast minimala alternativ
- ☐ Anpassad - visa alla möjliga alternativ

**Format:**

XML ▼

Kör

# XML — EXEMPEL

```
<!--
```

```
- Databas: 'telefonlista'
```

```
-->
```

```
<database name="telefonlista">
```

```
<!-- Tabell kontakt -->
```

```
<table name="kontakt">
```

```
<column name="namn">Mahmud Al Hakim</column>
```

```
<column name="telefon">070-1234567</column>
```

```
</table>
```

```
</database>
```

# JSON

JSON står för JavaScript Object Notation.




JSON är ett kompakt, textbaserat format som används för att utbyta data.







JSON är ett mindre utrymmeskrävande alternativ till XML.

OBS! JSON är enbart ett format för data. JSON säger inget om hur data ska presenteras för användaren.



# EXPORTERA DATA SOM JSON VIA PHPMYADMIN

→  Server: 127.0.0.1 »  Databas: telefonlista »  Tabell: kontakt

 Bläddra  Struktur  SQL  Sök  Lägg till  Exportera

**Format:**

JSON ▼

Kör

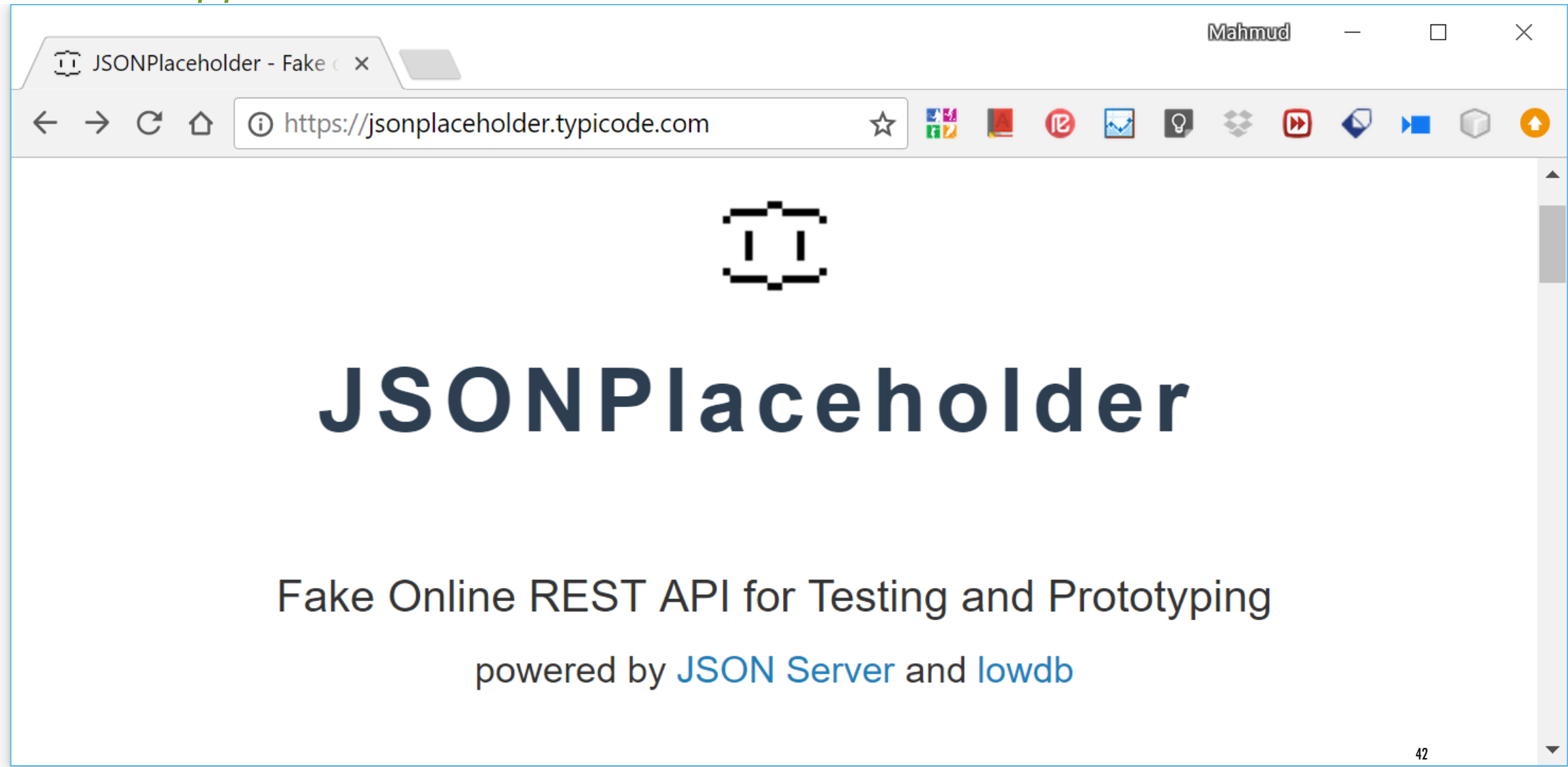


# JSON — EXEMPEL

```
{  
  "type": "table",  
  "name": "kontakt",  
  "database": "telefonlista",  
  "data": [  
    {  
      "namn": "Mahmud Al Hakim",  
      "telefon": "070-1234567"  
    }  
  ]  
}
```

# ARBETA MED JSON

[HTTPS://JSONPLACEHOLDER.TYPICODE.COM](https://jsonplaceholder.typicode.com)



# ARBETA MED JSON VIA PHP

```
<?php // Hämta en lista i JSON-format
$data = file_get_contents("https://jsonplaceholder.typicode.com/users");
echo "<pre>$data</pre>"; // Skriv ut allt

// Konvertera till en array med funktionen json_decode.
// Argumentet true skapar en associativ array (med key och value)
$array = json_decode($data, true);

// Välj data att visa
foreach ($array as $key => $value) {
    echo $value['name'] . ", " ;
    echo $value['email'] . "<br><strong>" ;
    echo $value['address']['city'] . "</strong><br>" ;
}
```

# SKAPA EGNA JSON-FILER FRÅN EN DATABAS

```
$query = "SELECT * FROM kontakt";  
$table = mysqli_query($connection,$query)  
or die(mysqli_error($connection));  
// Skapa en temporär array  
$tempArray = array();  
  
while($row = $table->fetch_assoc()){  
    $tempArray[] = $row;  
}  
// Skapa en JSON-string  
$json_string = json_encode($tempArray, JSON_PRETTY_PRINT);  
echo "<pre> $json_string </pre>";
```

# ARBETA MED FLERA TABELLER I EN RELATIONSDATABASES EN PRAKTISK ÖVNING

**Skapa en databas åt en liten videobutik som hyr ut filmer.**

Videobutiken har följande tabeller

Kund(**Kundnummer**, Namn, Gatadress, Postnummer, Ort)

Film(**FilmID**, Titel, Kategori, Huvudroll, Pris)

Uthyrning(*Film*, *Kund*, UtDatum, InDatum)

OBS! Kundnummer oh FilmID är primärnycklar.

Film och Kund i tabellen Uthyrning kallas främmande (sekundära) nycklar.

# DATABASDESIGN

## Databaser

 **Skapa databas** 

videobutik

utf8\_unicode\_ci ▼

**Skapa**

# SKAPA TABELLEN KUND



Skapa tabell

Namn:

kund

Antal kolumner:

5



# TABELLEN KUND – STRUKTUR

Tabellnamn:

kund

Lägg till

Namn

Typ ?

Längd/Värden ?

kundnummer

INT ▼

5

Välj från huvudkolumn

namn

VARCHAR ▼

50

Välj från huvudkolumn

gatuadress

VARCHAR ▼

50

Välj från huvudkolumn

postnummer

VARCHAR ▼

5

Välj från huvudkolumn

ort

VARCHAR ▼

50

Välj från huvudkolumn



# ANGE PRIMÄRNYCKEL OCH AUTO-INCREMENT

Index

A\_I



PRIMARY ▼

PRIMARY

☒

# TABELLEN FILM

Namn

Typ ?

Längd/Värden ?

Index

A I

filmID

INT ▼

5

Välj från huvudkolumn

PRIMARY ▼



PRIMARY

Titel

VARCHAR ▼

50

Välj från huvudkolumn

Kategori

VARCHAR ▼

50

Välj från huvudkolumn

Huvudroll

VARCHAR ▼

50

Välj från huvudkolumn

Pris

INT ▼

5

Välj från huvudkolumn

# TABELLEN UTHYRNING

	#	Namn	Typ	Kollationering	Attribut	nolläge	Standardvärde
<input type="checkbox"/>	1	Film	int(100)			Nej	Inget
<input type="checkbox"/>	2	Kund	int(100)			Nej	Inget
<input type="checkbox"/>	3	UtDatum	datetime			Nej	CURRENT_TIMESTAMP
<input type="checkbox"/>	4	InDatum	datetime			Nej	Inget

# IMPORTERA DATA FRÅN EN CSV-FIL

Bläddra Struktur SQL Sök Lägg till Exportera Importera

## Importera till tabellen "kund"

### Fil att importera:

Filen kan vara komprimerad (gzip, bzip2, zip) eller okomprimerad.

En komprimerad fils namn måste sluta med **[format].[komprimering]**. Exempel: **.sql.zip**

Gå igenom din dator:  kund.csv

Du kan också dra och släppa en fil på valfri sida.

Teckenuppsättning för filen:

### Partiell import:

☒ Tillåt avbrytande av import ifall skriptet upptäcker att det är nära till PHP timeout. (Detta är ett bra sätt att hantera stora transaktioner.)

Hoppa över detta antal frågor (för SQL) räknat från den första:

Välj en tabell först och öppna fliken Importera

Välj en CSV-fil

Hoppa över den första raden

Se nästa bild

CSV using LOAD DATA ▼

Välj  
CSV using LOAD DATA  
(Detta är ett MySQL-kommando)

**Format-specifika alternativ:**

☐ Uppdatera data när dublettnyckel hittas vid import (tillägg 'ON

Kolumnerna avslutas med:

;

Ändra till  
semikolon

Kolumner omges av:

"

Specialtecken i fält föregås av:

\

Rader avslutas med:

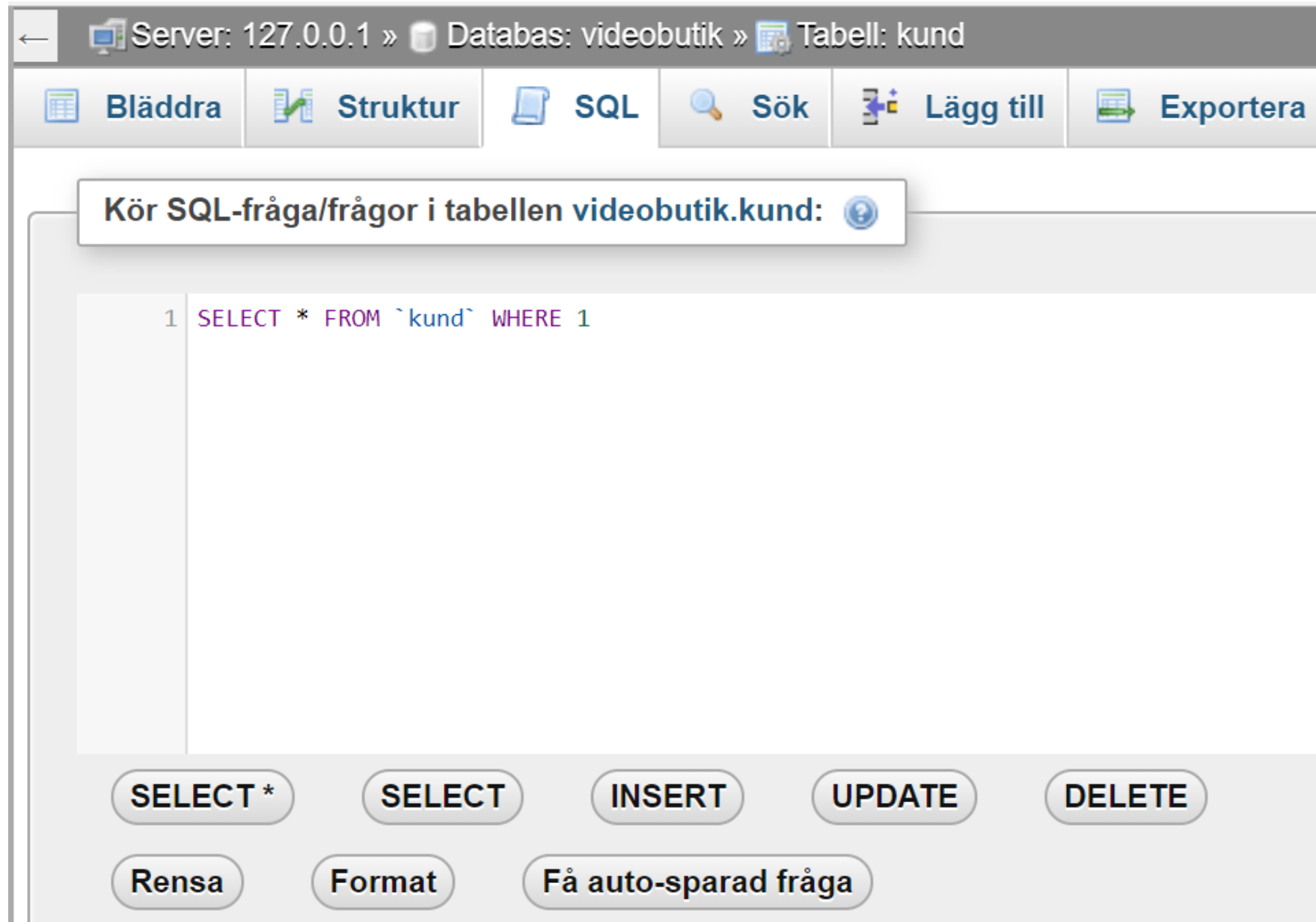
auto

Kolumnnamn:

☐ Avbryt inte vid INSERT-fel

☐ Använd nyckelordet LOCAL

# KÖR SQL-FRÅGOR VIA PHPMYADMIN



The screenshot shows the PHPMyAdmin interface for a MySQL database. The top navigation bar indicates the server is 127.0.0.1, the database is 'videobutik', and the selected table is 'kund'. Below this, there are six tabs: 'Bläddra', 'Struktur', 'SQL', 'Sök', 'Lägg till', and 'Exportera'. The 'SQL' tab is active, and a tooltip above the query area says 'Kör SQL-fråga/frågor i tabellen videobutik.kund:'. The query editor contains the following SQL statement:

```
1 SELECT * FROM `kund` WHERE 1
```

At the bottom of the interface, there are buttons for 'SELECT \*', 'SELECT', 'INSERT', 'UPDATE', and 'DELETE'. Below these, there are buttons for 'Rensa', 'Format', and 'Få auto-sparad fråga'.

# SQL – EXEMPEL 1

-- Enkla frågor mot enstaka tabeller

-- Visa alla kunder

```
SELECT * FROM Kund;
```

-- Visa alla filmer

```
SELECT * FROM Film;
```

-- Visa alla uthyrningar

```
SELECT * FROM Uthyrning;
```

# SQL – EXEMPEL 2

```
-- Vilka olika filmer med "Mel Gibson"  
-- i huvudrollen finns i databasen?
```

```
SELECT * FROM Film
```

```
WHERE Huvudroll = 'Mel Gibson'
```

```
-- Vilka alla filmer i kategorin Drama
```

```
SELECT * FROM Film
```

```
WHERE Kategori = 'Drama'
```



# SQL – EXEMPEL 3

-- Vilka kunder bor i Stockholm?

SELECT \* FROM Kund

WHERE Ort = 'Stockholm'

-- Vilka kunder bor inte i Stockholm?

SELECT \* FROM Kund

WHERE Ort != 'Stockholm'

# SQL – EXEMPEL 4

-- Vilka kunder bor på adresser som har  
postnummer som börjar på 1?

```
SELECT * FROM Kund
```

```
WHERE postnummer LIKE '1%'
```

-- Vilka kunder bor adresser som har  
postnummer som inte börjar på 1?

```
SELECT * FROM Kund
```

```
WHERE postnummer NOT LIKE '1%'
```

# SQL – EXEMPEL 5

```
-- Visa ej återlämnade filmer.  
-- Alltså, vilka filmer finns ute hos kunderna!
```

```
SELECT * FROM Uthyrning  
WHERE InDatum IS NULL;
```

# SQL – EXEMPEL 6

```
-- Lägg till en ny uthyrning  
-- Kundnummer 5 hyr ut film 10
```

```
INSERT INTO uthyrning  
VALUES ('10', '5', CURRENT_TIMESTAMP, '')
```

# SQL – EXEMPEL 7

-- Uppdatera kundnamn

```
UPDATE kund SET namn = 'Kalle Lindqvist'  
WHERE kundnummer = 2
```

# SQL – EXEMPEL 8

-- Ta bort en kund

```
DELETE FROM kund  
WHERE kundnummer = 1
```

# SQL – EXEMPEL 9

```
-- Att koppla flera tabeller  
-- Join (Inner Join)
```

```
SELECT * FROM Kund, Uthyrning  
WHERE kund.Kundnummer = Uthyrning.Kund
```

# SQL – EXEMPEL 10

-- Visa några fält från olika tabeller

SELECT

Kund.Namn,

Uthyrning.Film,

Uthyrning.UtDatum

FROM Kund, Uthyrning

WHERE kund.Kundnummer = Uthyrning.Kund



# SQL – EXEMPEL 11

```
-- Använd Alias  
-- OBS! AS kan utelämnas!
```

```
SELECT K.Namn, U.Film, U.UtDatum  
FROM    Kund AS K, Uthyrning U  
WHERE   K.Kundnummer = U.Kund
```

# SQL – EXEMPEL 12

```
-- Visa ej återlämnade filmer.  
-- Visa kundnamnet och filmID.
```

```
SELECT K.Namn, U.Film, U.UtDatum  
FROM Kund AS K, Uthyrning AS U  
WHERE K.Kundnummer = U.Kund  
AND U.InDatum IS NULL
```

# SQL – EXEMPEL 13

```
-- Visa ej återlämnade filmer.  
-- Visa Filmtitel och uthyrningsdatumet.  
-- Sortera efter uthyrningsdatumet
```

```
SELECT Film.Titel, Uthyrning.UtDatum  
FROM Uthyrning, Film  
WHERE InDatum IS NULL  
AND Film.FilmID = Uthyrning.Film  
ORDER BY UtDatum
```

# SQL – EXEMPEL 14

- Visa ej återlämnade filmer.
- Visa Kundnamn, Filmtitel och uthyrningsdatumet.

```
SELECT K.Namn, F.Titel, U.UtDatum
FROM Film F, Kund K, Uthyrning U
WHERE K.Kundnummer = U.Kund
AND F.FilmID = U.Film
AND InDatum IS NULL
```

# SQL – EXEMPEL 15

```
-- Visa en lista på alla filmer som är ute  
-- Visa kundnamn och filmtitel  
-- Sortera efter kundnamnet
```

```
SELECT K.Namn, F.Titel  
FROM Kund K, Film F, Uthyrning U  
WHERE U.InDatum IS NULL  
AND K.Kundnummer = U.Kund  
AND F.FilmID = U.Film  
ORDER BY K.Namn
```

# SKAPA EN WEBBSIDA SOM VISAR UTHYRDA FILMER

## DEL 1 — DATABASKOPPLINGAR OCH SQL

```
<body class="container">

<?php

$query = "SELECT K.Namn, F.Titel, U.UtDatum
FROM Kund K, Film F, Uthyrning U
WHERE U.InDatum IS NULL
AND K.Kundnummer = U.Kund
AND F.FilmID = U.Film
ORDER BY U.UtDatum";

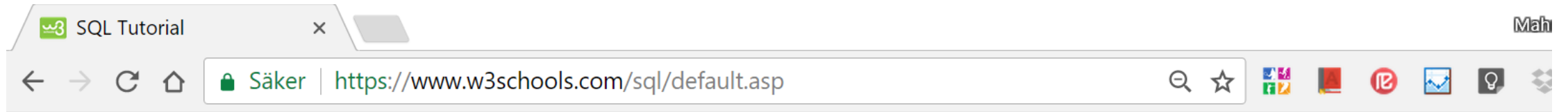
$table = mysqli_query($connection,$query)
or die(mysqli_error($connection));
```

# SKAPA EN WEBBSIDA SOM VISAR UTHYRDA FILMER

## DEL 2 – VISA INFO I EN TABELL

```
echo "<table class='table'><tr>";
echo "<th>Kund</th><th>Film</th><th>Uthyrningsdatum</th></tr>";
while($row = $table->fetch_assoc()):    ?>
    <tr>
    <td><?php echo $row['Namn'] ?> </td>
    <td><?php echo $row['Titel'] ?></td>
    <td><?php echo $row['UtDatum'] ?></td>
    </tr>
<?php endwhile; ?>
</table>
</body></html>
```

# LÄS SQL TUTORIAL PÅ <https://www.w3schools.com/sql>



w3schools.com

THE WORLD'S LARGEST WEB D



HTML

CSS

JAVASCRIPT

SQL

PHP

BOOTSTRAP

MORE ▾

REFERENCES ▾

EXAM

SQL Tutorial

SQL HOME

SQL Intro

SQL Syntax

SQL Select

SQL Select Distinct

SQL Where

SQL And, Or, Not

SQL Order By

SQL Insert Into

SQL Null Values

SQL Update

SQL Delete

Java, .net, node.js – koda dina appar på ditt språk.

Prova Azure  
kostnadsfritt

Microsoft

## SQL Tutorial

< Home

Next >

SQL is a standard language for storing, manipulating and retrieving data in databases.

Our SQL tutorial will teach you how to use SQL in: MySQL, SQL Server, MS Access, Oracle, Sybase, Informix, Postgres, and other database systems.