



Databasteknik

Workbench, modellering, SQL

Utbildare: Mikael Olsson

mikael.olsson@emmio.se

076-174 90 43

NACKADEMIN

Objekttyper

- Självständiga objekt (strong entities)
 - Oberoende av andra objekt
 - Egen lagringsnyckel
- Beroende objekt (weak entities)
 - Ägs av ett eller flera överordnade objekt
 - Kan t ex ha sammansatt nyckel av ägarens pk + egen del
 - Ex: telnr registreras sällan utan ägare

Objekttyper



FAKTURA

- Existerar en fakturarad utan en faktura?
- Vilken objekttyp?

Fakturanr	5/2011
Datum	2011-02-24
Orderdatum	-
Referens	Niklas Nilsson
Betalningsvillkor	20 dagar netto
Förfallodatum	2011-03-16
Dräjsmärsinta	10 %

Fakturamottagare
Mobilgraven
Brunnsjögatan 10
775 35 Hedemora

Beskrivning	Summa
Webbutveckling del 1, se specifikation.	9 183 kr
Iframe till Faludäck, 1 timme	650 kr
Illustration omröstning	2 000 kr
Illustration Maskot	2 000 kr

Sammansatt nyckel



FakturalD	RadID	Benämning	Pris
1	1	1 Webbutveckling del 1, se specifikation.	9183
1	2	2 Iframe till Faludäck, 1 timme	650

Server vs Workbench

- Client / server
- Browser / web server
- Var finns servern?

Numeriska datatyper, exempel

Namn	Storlek signed	Storlek unsigned
INT	-2147483648 till 2147483647	0 till 4294967295
TINYINT	-128 to 127	0 till 255
SMALLINT	-32768 till 32767	0 till 65535

- Vad är skillnaden mellan *signed* och *unsigned*?
- Välja en så anpassad typ till dina behov som möjligt för att göra databasen effektiv och spara lagringsutrymme.

Datum-datatyper, exempel

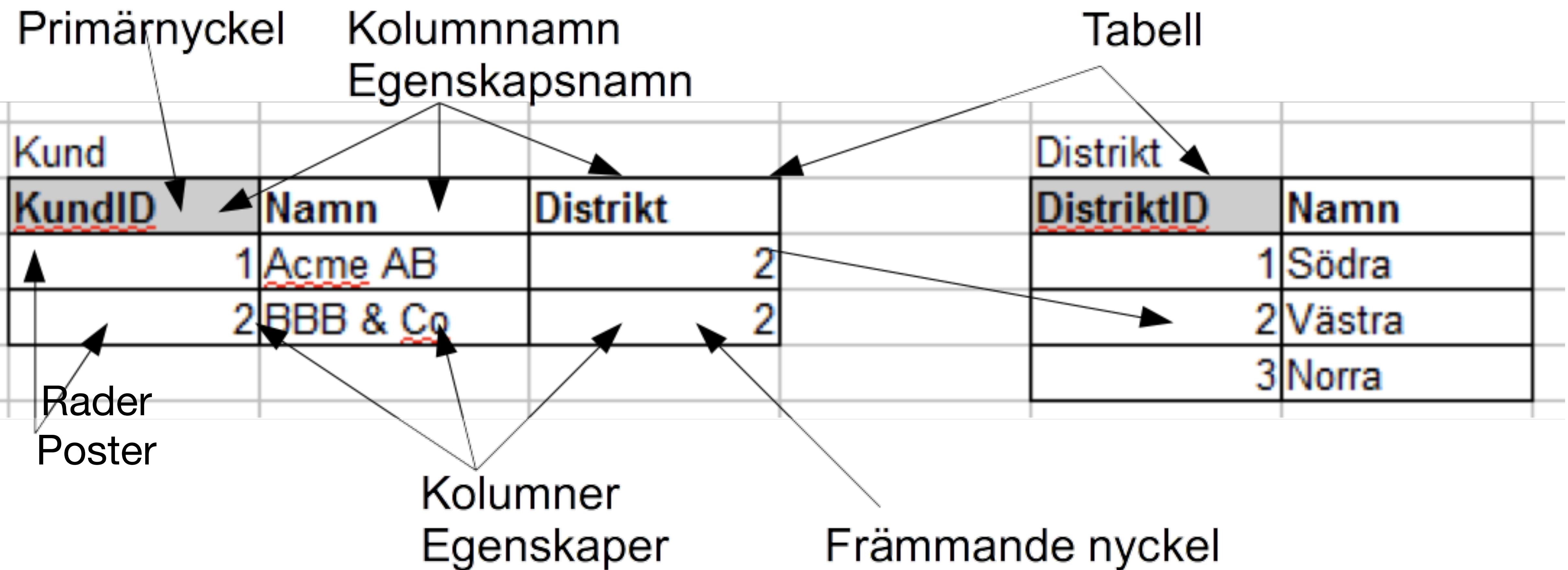
Namn	Format
DATE	YYYY-MM-DD
DATETIME	YYYY-MM-DD HH:MM:SS
TIME	HH:MM:SS

Sträng-datatyper, exempel

Namn	Storlek
VARCHAR	1 till 255 tecken
TEXT	max 65535 texken
ENUM	Lista

<https://www.tutorialspoint.com/mysql/mysql-data-types.htm>

Tabeller



Mer om nycklar

"The word key is one of the most overworked in the entire database field"

Primary key.....	Identifierar en rad unikt
Candidate key.....	Kolumner som kan fungera som pk
Secondary key.....	Kandidatnycklar som inte är pk
Surrogate key.....	Extra kolumn (autonumber) för pk
Alternate key.....	De kandidatnycklar som inte valdes till pk
Search key.....	Ett index avsett för att underlätta sökning
Foreign key.....	Primärnyckel från en annan tabell
Index key.....	Index på ett eller flera kolumner
Parent key.....	Pk på första sidan
Super key.....	En överbestämd pk
Composite key.....	Nyckel som består av flera kolumner
Ordering key.....	Ett index för att underlätta sorterad visning

Primärnyckel

- Huvudsyfte är att märka en post unikt
 - Bör väljas stabil. Svårt (=dyrt) att ändra.
 - Undvik talande nycklar som artnr, persnr, namn osv
 - Undvik stora pk – ger stora index och dålig prestanda
 - Autonummerade nycklar är bra
 - Numeriska nycklar ger bäst prestanda.

Local by Flywheel

- Utvecklingsmiljö för WordPress.
- Innehåller webbserver och databasserver.
- Lättaste sättet att sätta upp en miljö som funkar på både Mac och Windows, men det finns andra stackar som XAMPP, MAMP, WAMP osv.
- Sätter upp WordPress-siter, men vi kommer att strunta i filer och tömma databaser och bara använda databasservern.

Stack

- Ni läser till "Webbutvecklare fullstack open source", men vad betyder egentligen fullstack? Vad betyder stack?
- Stack är en uppsättning program som krävs för att tillhandahålla en tjänst.
- Programmen kan variera, men består (för webb) ofta av:
 - Ett operativsystem
 - En databas
 - Ett programmeringsspråk (för backend)
 - En webserver

Fullstack

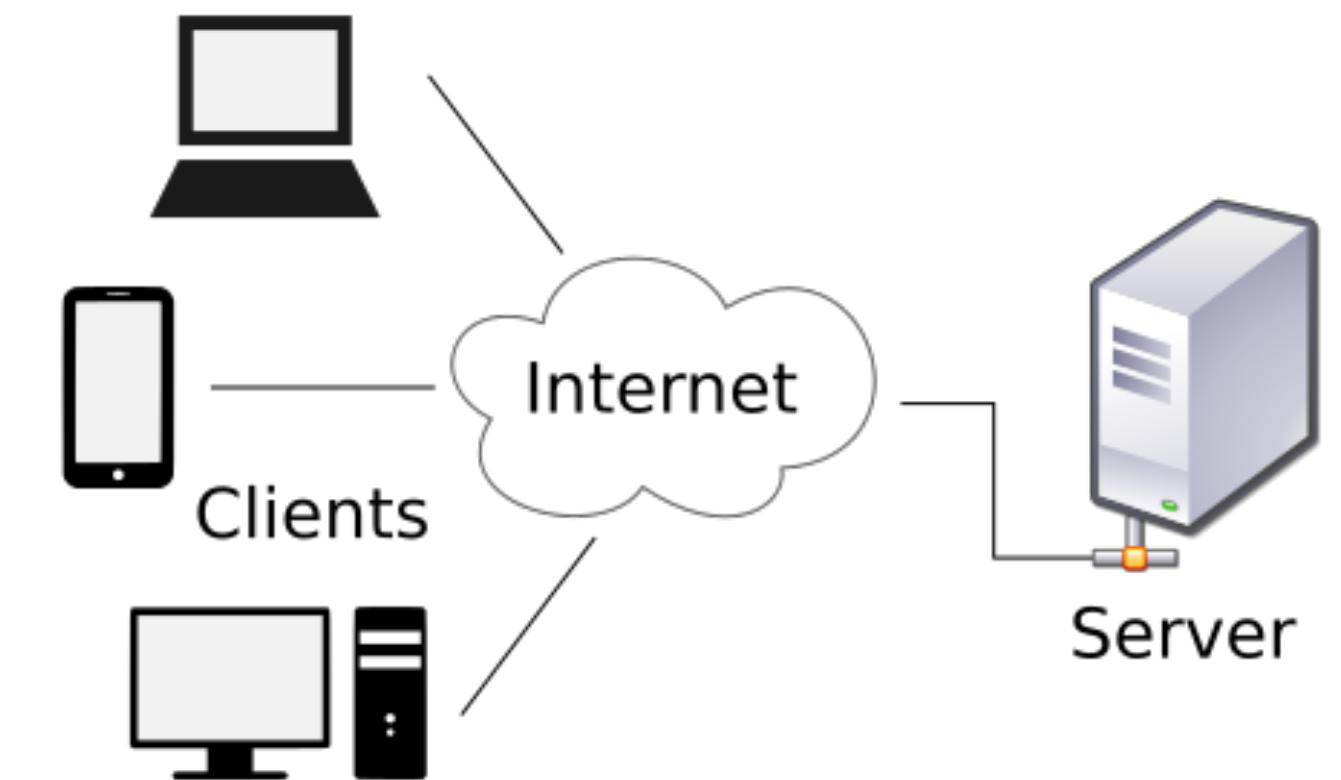
- En fullstackutvecklare brukar betyda en utvecklare som förstår hela kedjan; operativsystem, databas, backend och frontend.
- En del avser att man jobbar med frontend och backend (inkl databaser) och kanske, men inte nödvändigtvis, kan hantera operativsystem. (Dvs installera en webbserver, konfigurera en brandvägg och liknande.)

Vanliga stackar

- LAMP - "originalstacken" - Linux (OS), Apache (webbserver), MySQL/MariaDB (db), PHP/Perl/Python (backendspråk)
- MERN - MongoDB (db), Express.js (node framework), React.js (klient js framework), Node.js (webbserver)
- NMP - Nginx (webbserver), MySQL, PHP
- https://en.wikipedia.org/wiki/Solution_stack

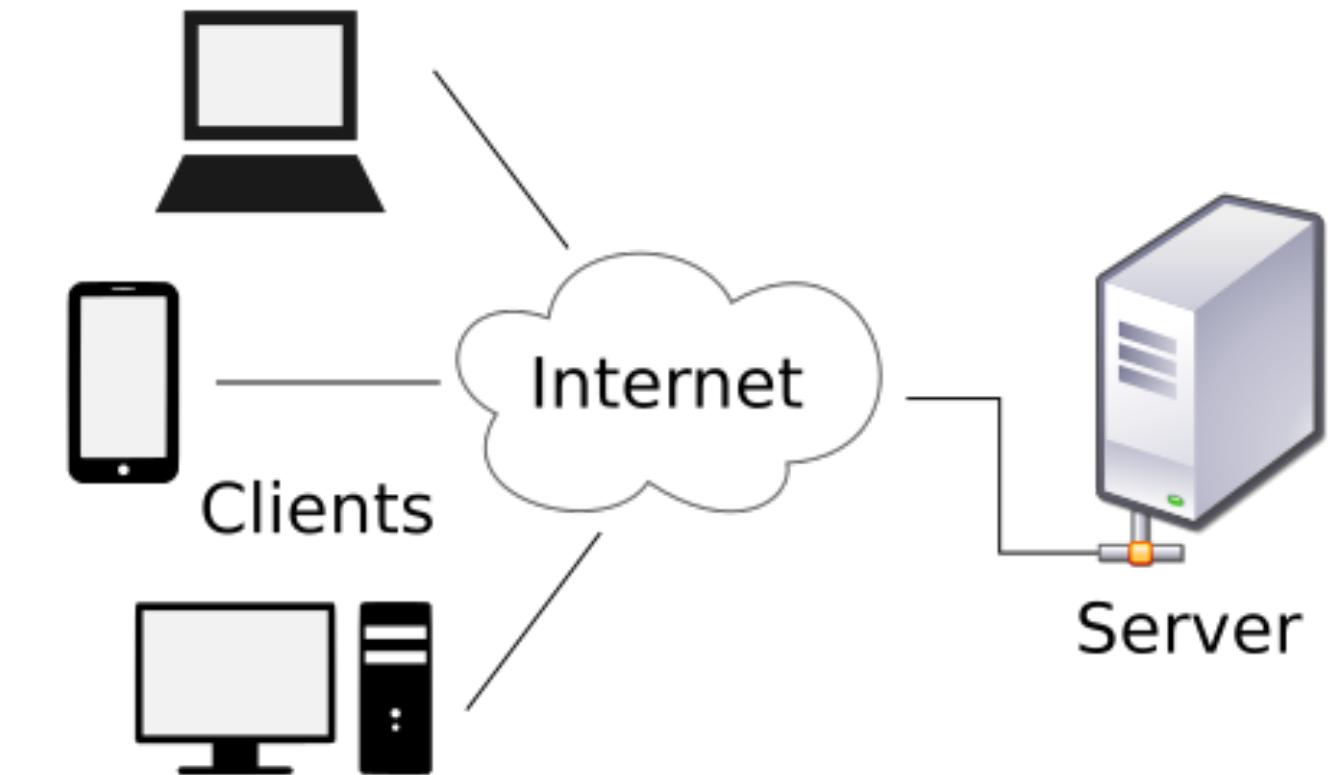
Klient/server-modellen

- Server utför något slags arbete, delar resurser
- Klient är en beställare.
- Ofta över ett nät, men inte alltid.
- Klienten delar inte med sig av sina resurser, den begär en tjänst.
- Klienten initierar kommunikation, servern inväntar den.
- Ex: mail, nätverksskrivare, www.



Klient/server-modellen

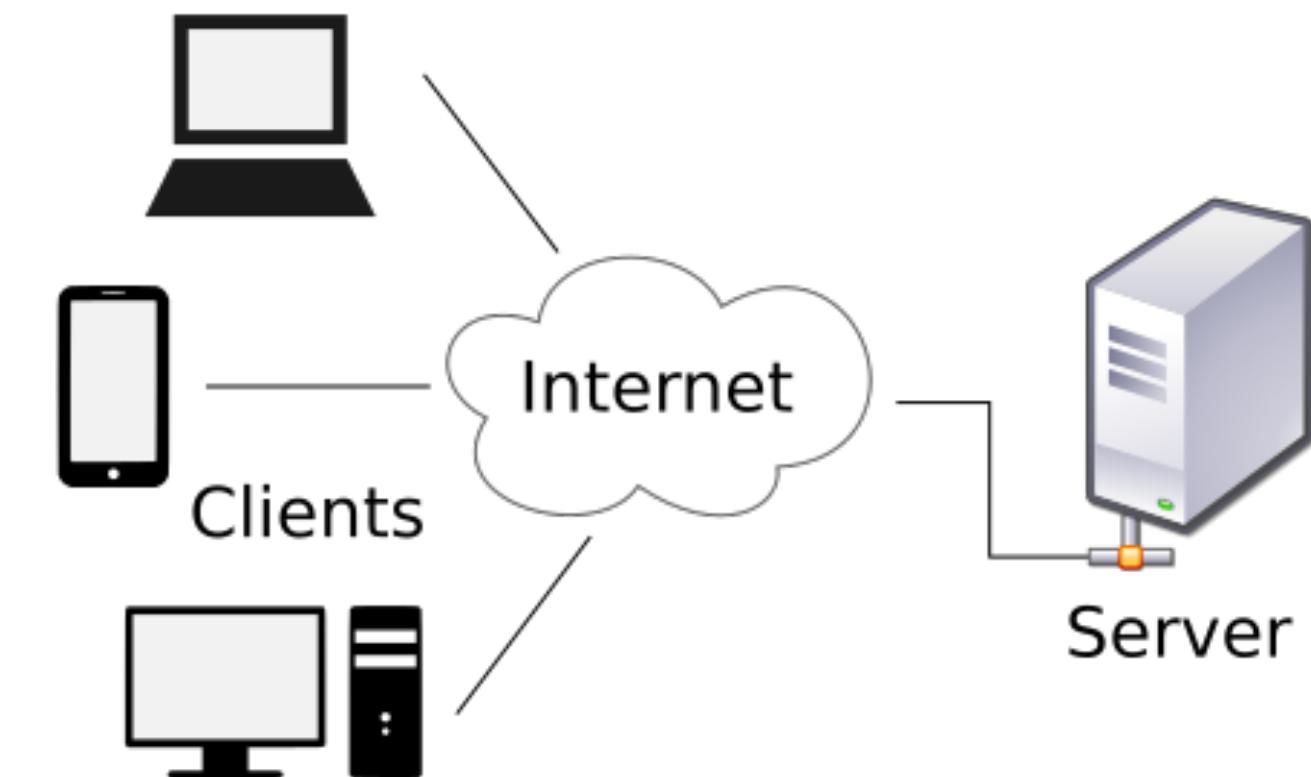
- Server är inte en slags dator.
- Server är ett program.
 - Lyssnar på requests
 - Kan i princip köras på vilken dator som helst
 - Datorn som kör server-programmet kallas ofta för server.
 - Man kan köra en server på sin egen dator. Då agerar samma dator både server och klient.



Förvirrande?!

Klient/server-modellen

- Vad finns det för klienter?
 - Browsers
 - Mailklienter
 - Ftp-klient
 - SSH-klient



Internet - webserver

www.aftonbladet.se

104.20.55.70



Webbserver

- Lyssnar på anrop
- Avgör vilken typ av resurs (fil) som efterfrågas.
- Statiska filer (html, css, js, bilder) skickas tillbaka direkt.
- PHP-filer och andra backendfiler måste *parsas* eller *exekveras* först.

Databasserver

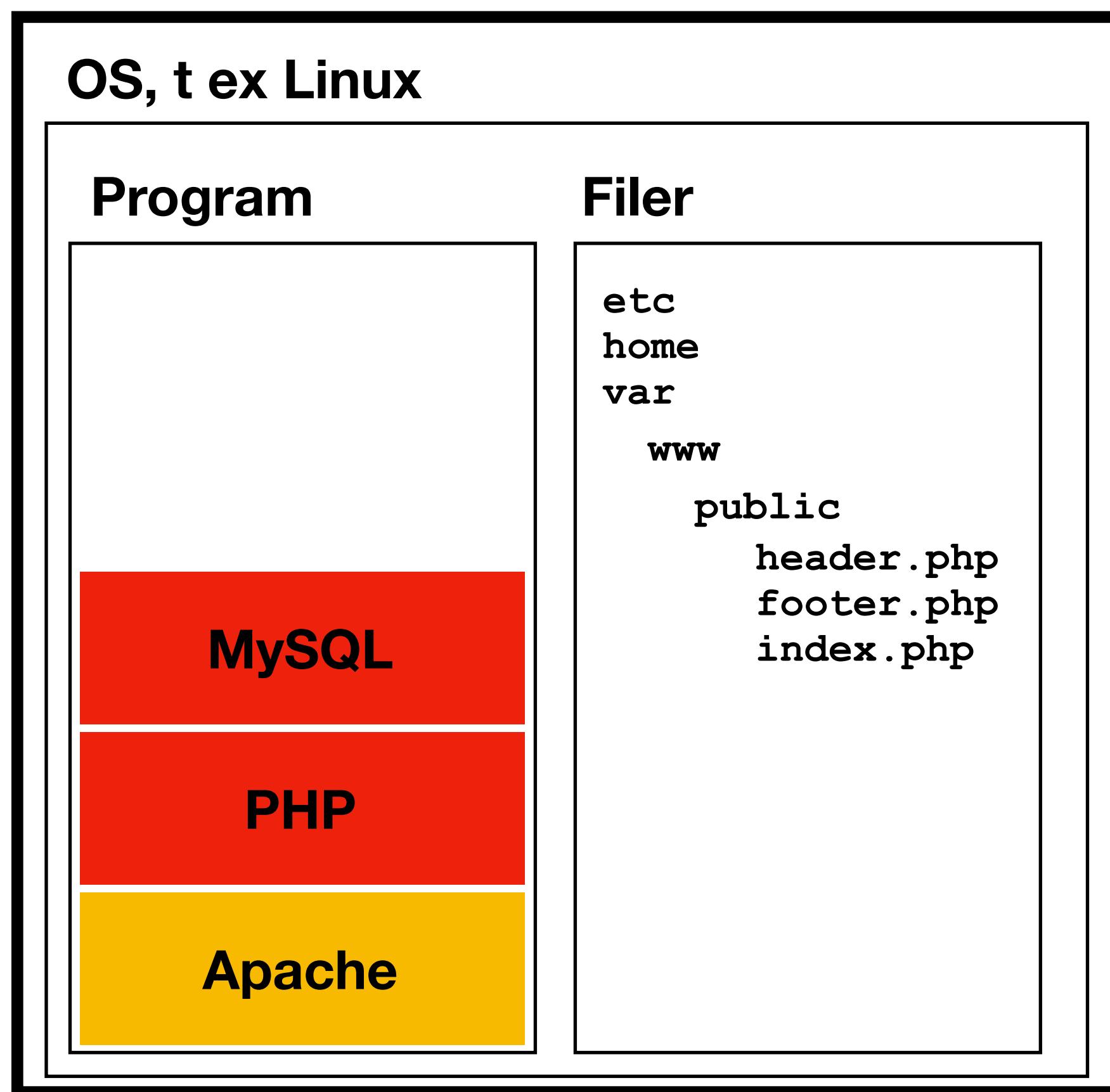
- Hanterar databas.
- Kan oftast kommunicera med webserver.

Lite om hur internet fungerar

I detta exempel använder vi PHP, men principen är densamma för Node.

Att köra php-filer

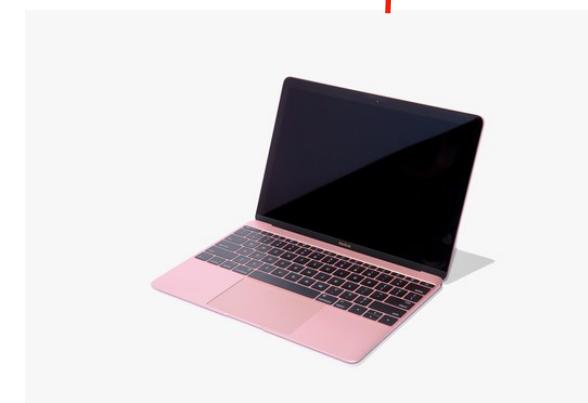
Dator



Internet

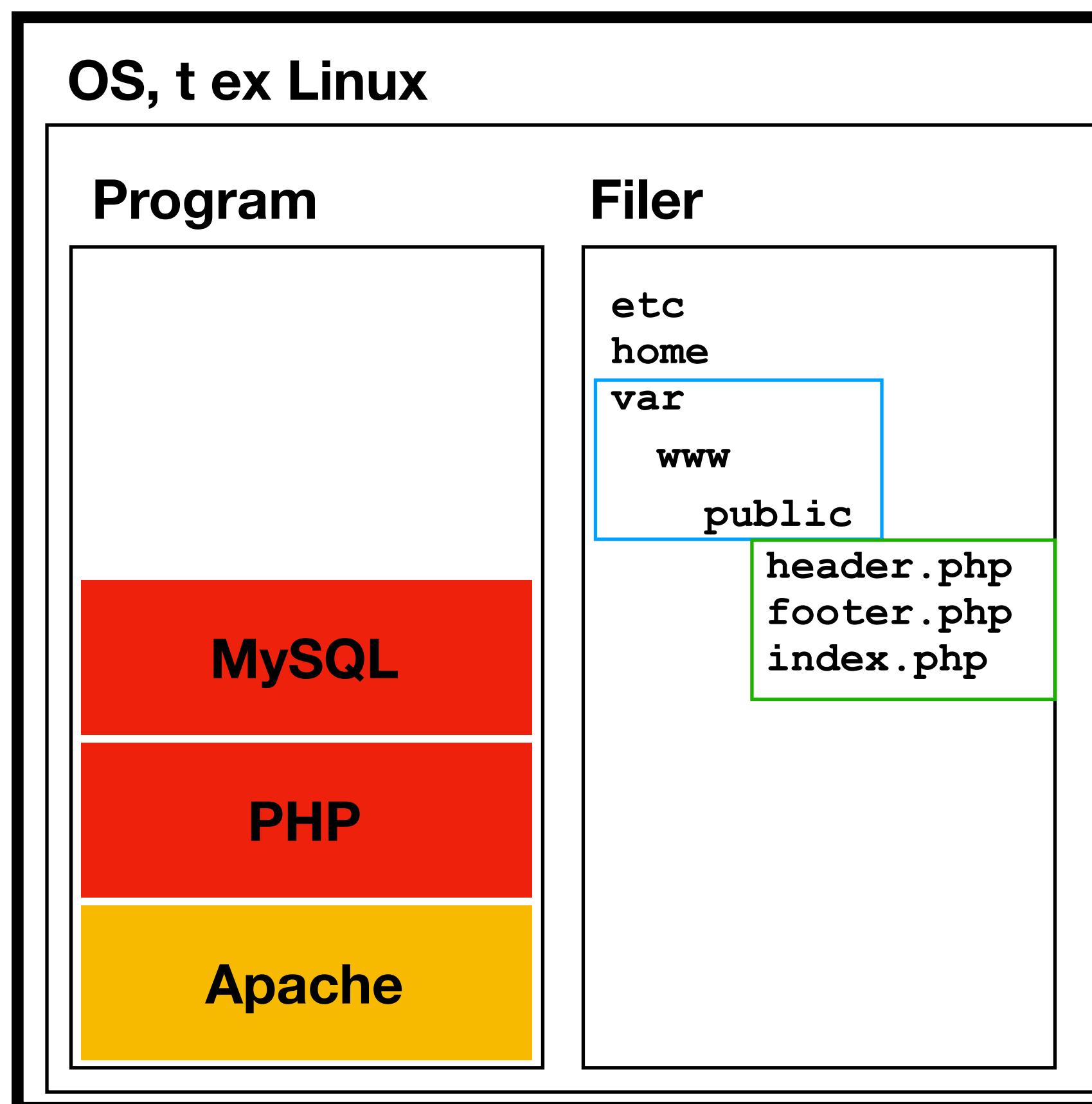


← → ⌂ example.com/



Att köra php-filer

Dator



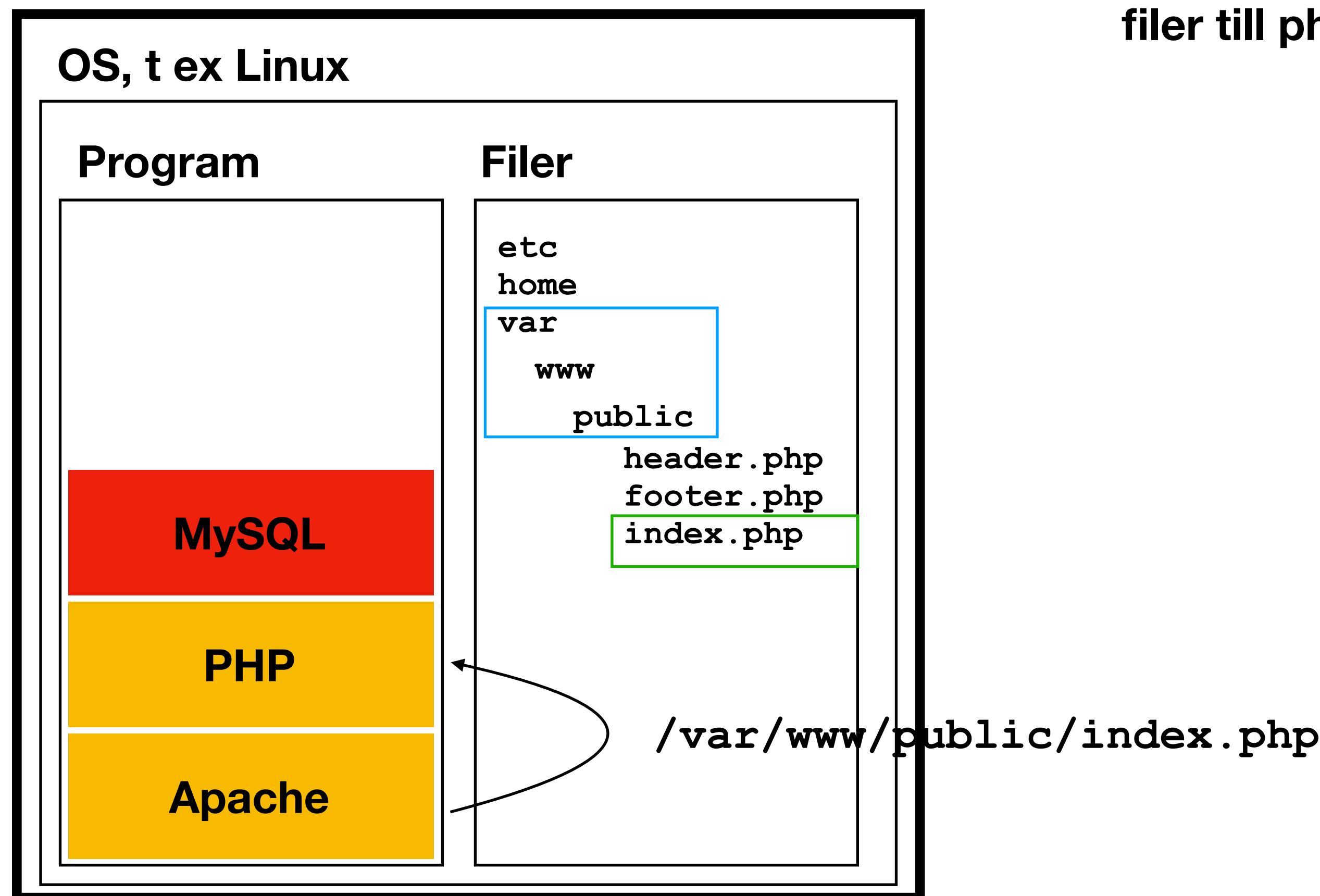
Apache kollar konfigurationen

```
<VirtualHost *:80>
    DocumentRoot "/var/www/public/"
    ServerName example.com
    DirectoryIndex index.html index.php
</VirtualHost>
```

- Den här regeln ska gälla om besökaren går in på domänen example.com
- Filerna för den här webbplatsen ligger i mappen /var/www/public
- DirectoryIndex visar vilken fil som automatiskt ska visas.
 - Finns det någon index.html?
 - Finns det någon index.php?

Att köra php-filer

Dator



Apache är konfigurerat för att skicka php-filer till php-applikationen.

Att köra php-filer

PHP

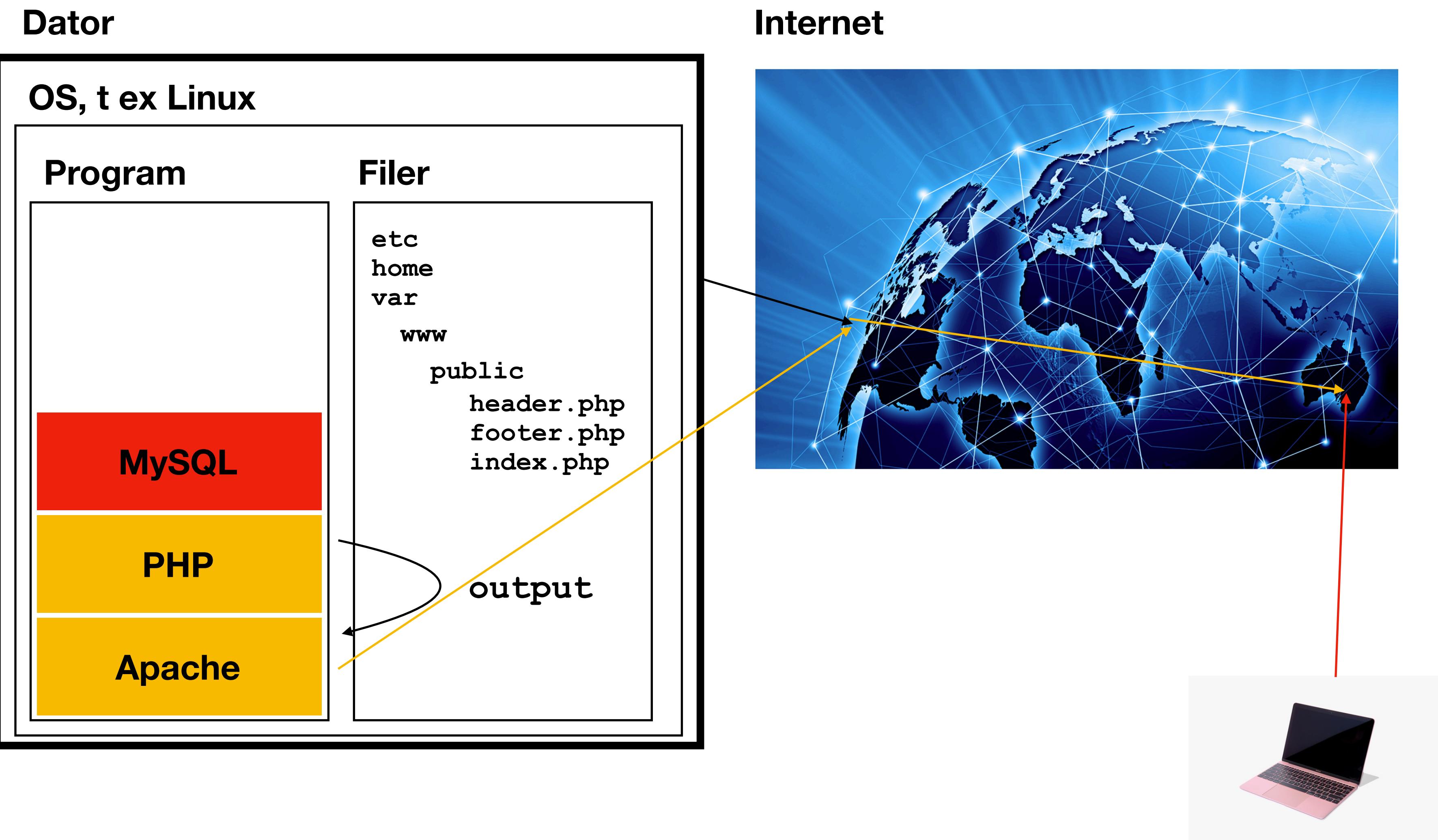
index.php

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <title>Min shop</title>
    <meta name="viewport" content="width=device-width,
initial-scale=1">
    <link rel="stylesheet" type="text/css" media="screen"
href="styles/main.css">
    <link rel="stylesheet" type="text/css" media="screen"
href="styles/styles/style01.css">
</head>
<body>
    <?php
        include 'functions.php';
        hello();
    ?>
</body>
</html>
```

Output till apache

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <title>Min shop</title>
    <meta name="viewport" content="width=device-width,
initial-scale=1">
    <link rel="stylesheet" type="text/css" media="screen"
href="styles/main.css">
    <link rel="stylesheet" type="text/css" media="screen"
href="styles/styles/style01.css">
</head>
<body>
    Hello world
</body>
</html>
```

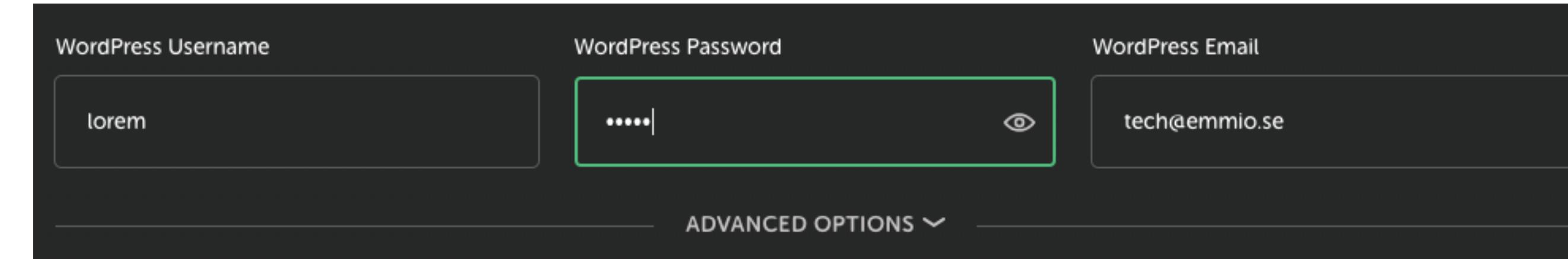
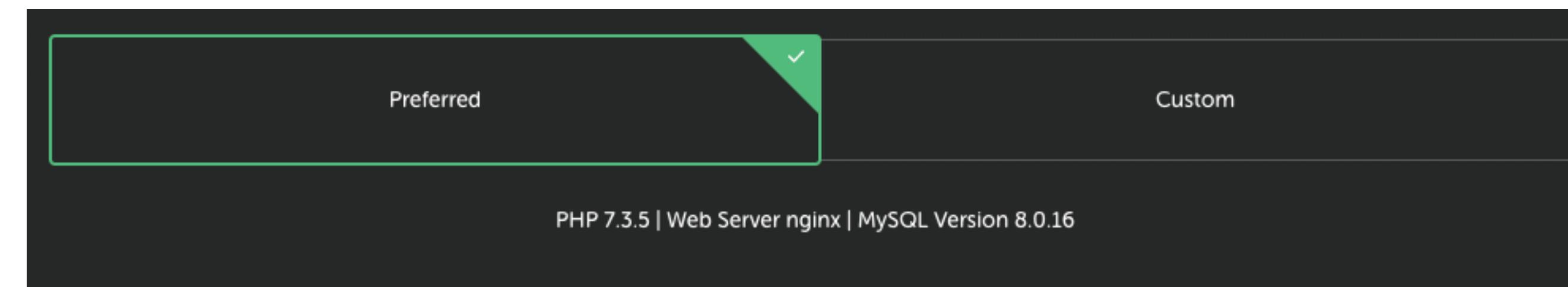
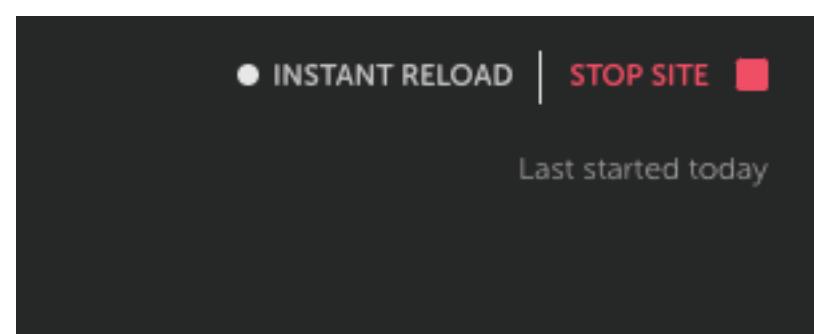
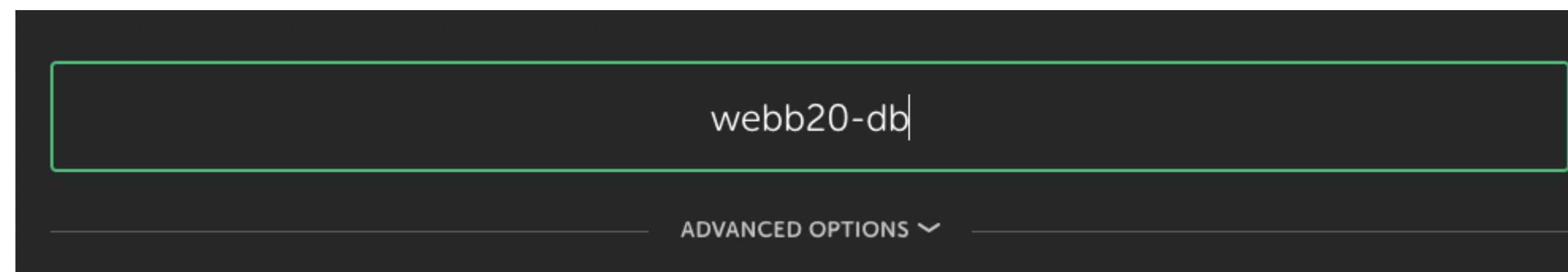
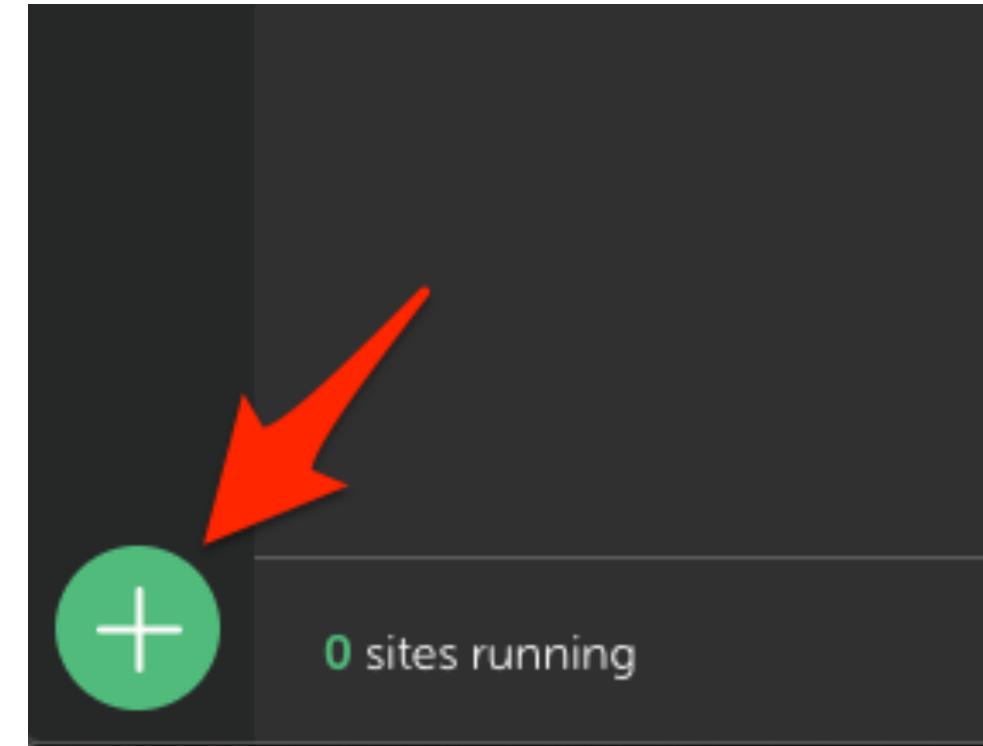
Att köra php-filer



Local by Flywheel

Sätta upp en ny site

- Skapa en ny site och gå igenom guiden. Fyll i vad som helst i WordPress-rutorna.
- Starta siten om den inte körs.



Local by Flywheel

Sätta upp en ny site

- Notera uppgifterna under fliken *Database*.
- Det kan finnas olika sätt att ansluta till databasen.
 - Socket
 - Host + port

webb20-db			
/Users/micke/www/projects/webb20db ⓘ			
OVERVIEW	DATABASE	UTILITIES	TOOLS
Connect	OPEN ADMINER	DOWNLOAD SEQUEL PRO	
Host	localhost		
Socket	/Users/micke/Library/Application Support/Local/run/d5uuTh7we/mysql/mysqld.sock		
Database Name	local		
Username	root		
Password	root		

MySQL Workbench

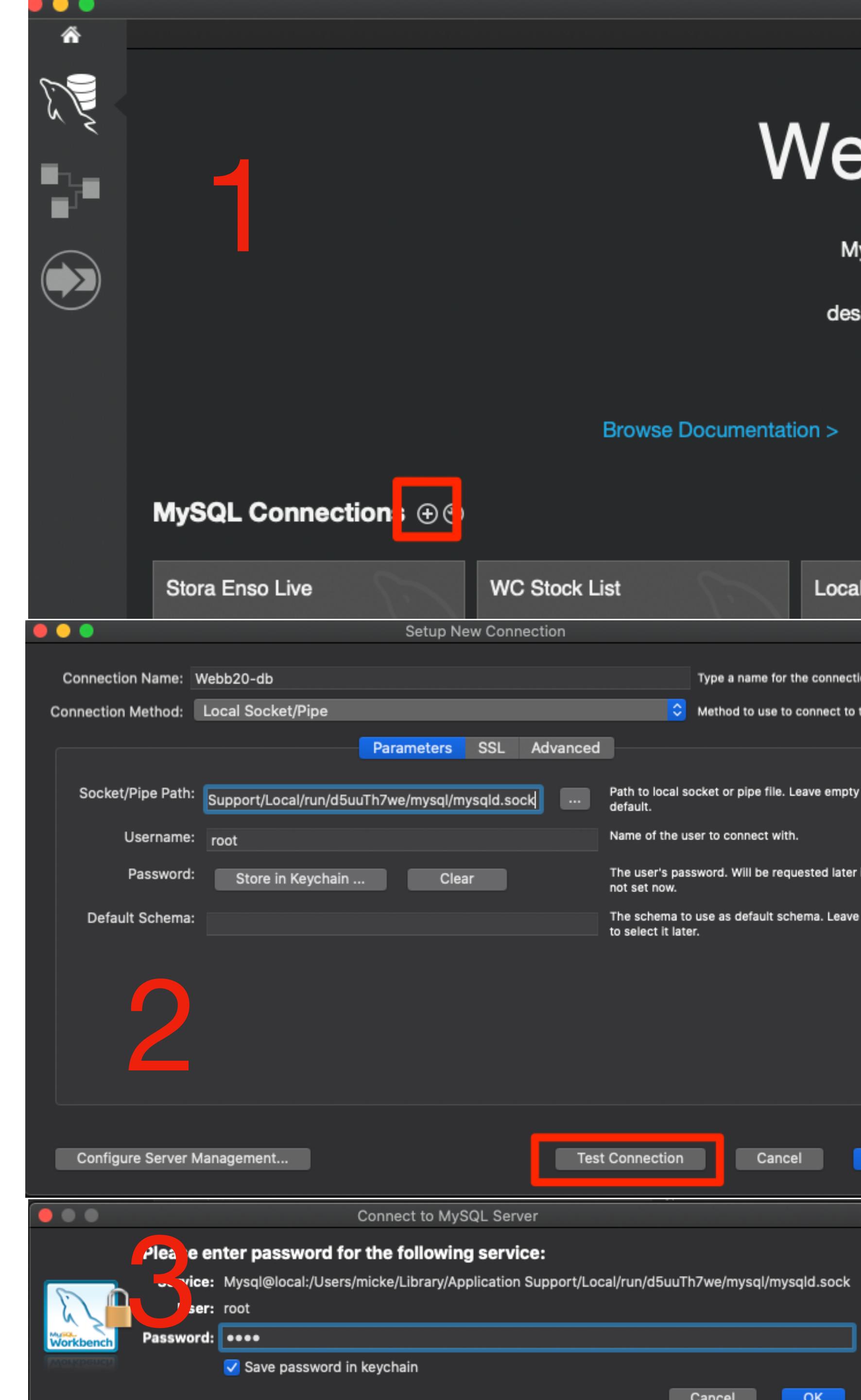
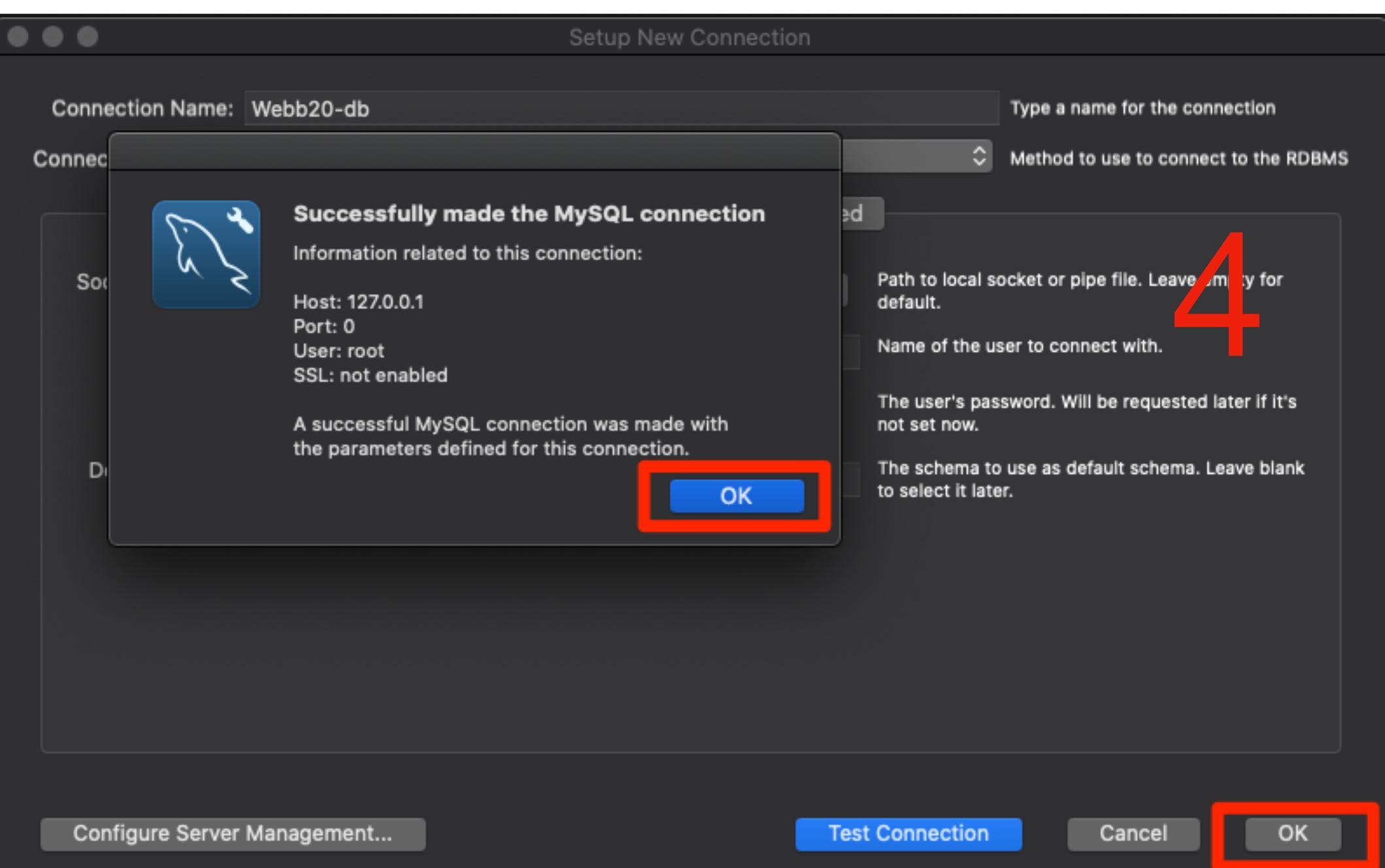
Databasklient

- Klient / GUI (Graphical User Interface) för att ansluta till databasserver.
- Man kan göra jämförelsen att MySQL Workbench är för en databasserver vad en browser är för en webbserver.

MySQL Workbench

Ansluta

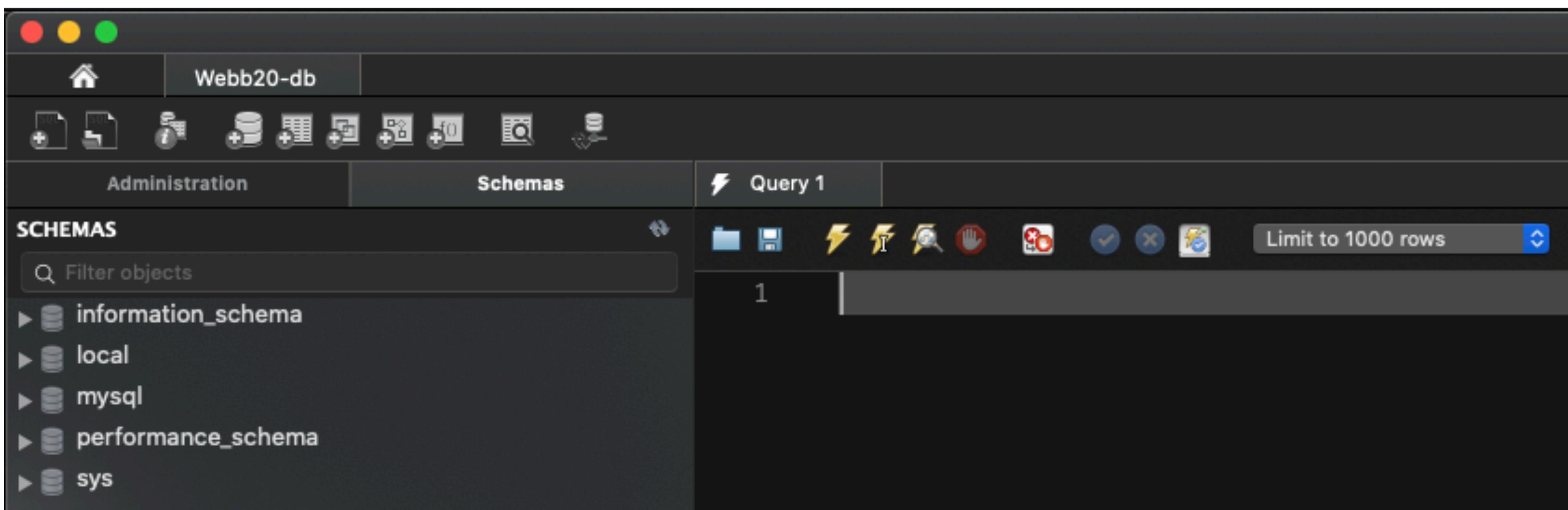
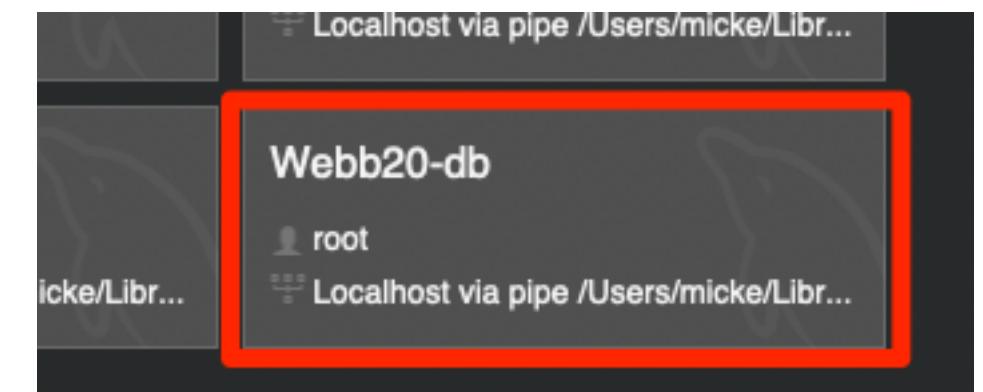
- Fyll i uppgifterna från databasfliken på local.
- Ange *Standard (TCP/IP)* eller *Local Socket / Pipe* beroende på vilken anslutningstyp databasfliken i local visar.



MySQL Workbench

Ansluta

- Öppna anslutningen genom att klicka på den.
- Nu är du redo! Vi kommer att komma tillbaka hit senare.



Kardinalitet

- 1:1 - En till En
 - En film kan ha en handling (inte så vanlig koppling, används möjligen om det finns ett stort fält som inte alla poster har)
- 1:N - En till Många
 - En person kan äga flera bilar
- N:1 - Många till En
 - Flera personer kan äga ett hus ihop eller bo på samma adress
- N:N (eller N:M) - Många till Många
 - En författare kan skriva flera böcker och en bok kan ha flera författare

Relation

Kardinalitet

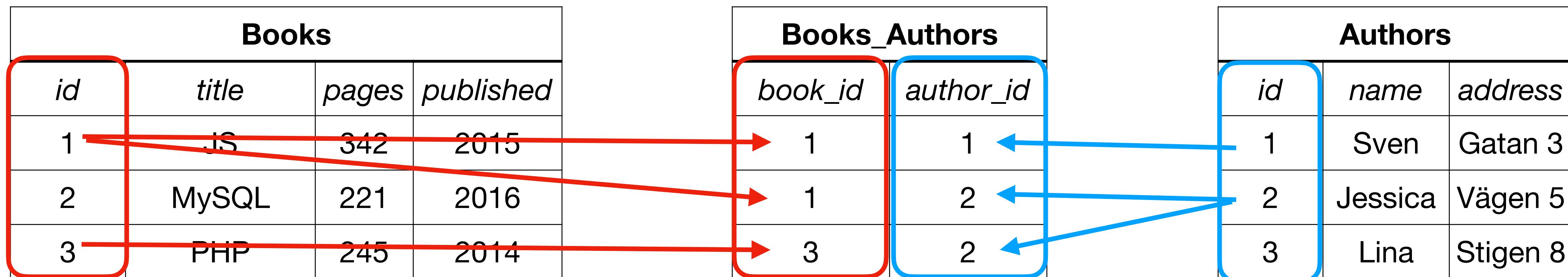
- Vad är det för kardinalitet mellan Person och Car
- Mellan Car och Person?
- Person och Address?
- Address och Person?

PersonID	FirstName	LastName	AddressID
1	Eva	Vik	1
2	Stina	Nilsson	2
3	Lars	Nilsson	2
AddressID	Address	Rooms	
1	Vägen 1	3	
2	Gatan 3	1	
CarID	Car	Registration	OwnerID
1	Volvo V70	KXC122	1
2	Ford Ka	GRE479	1
3	Ford Ka	ASD542	2

Kardinalitet

N:N

- Hur gör vi om vi vill kunna koppla ihop många till många?
- Vi får göra en *uppslagstabell*!!



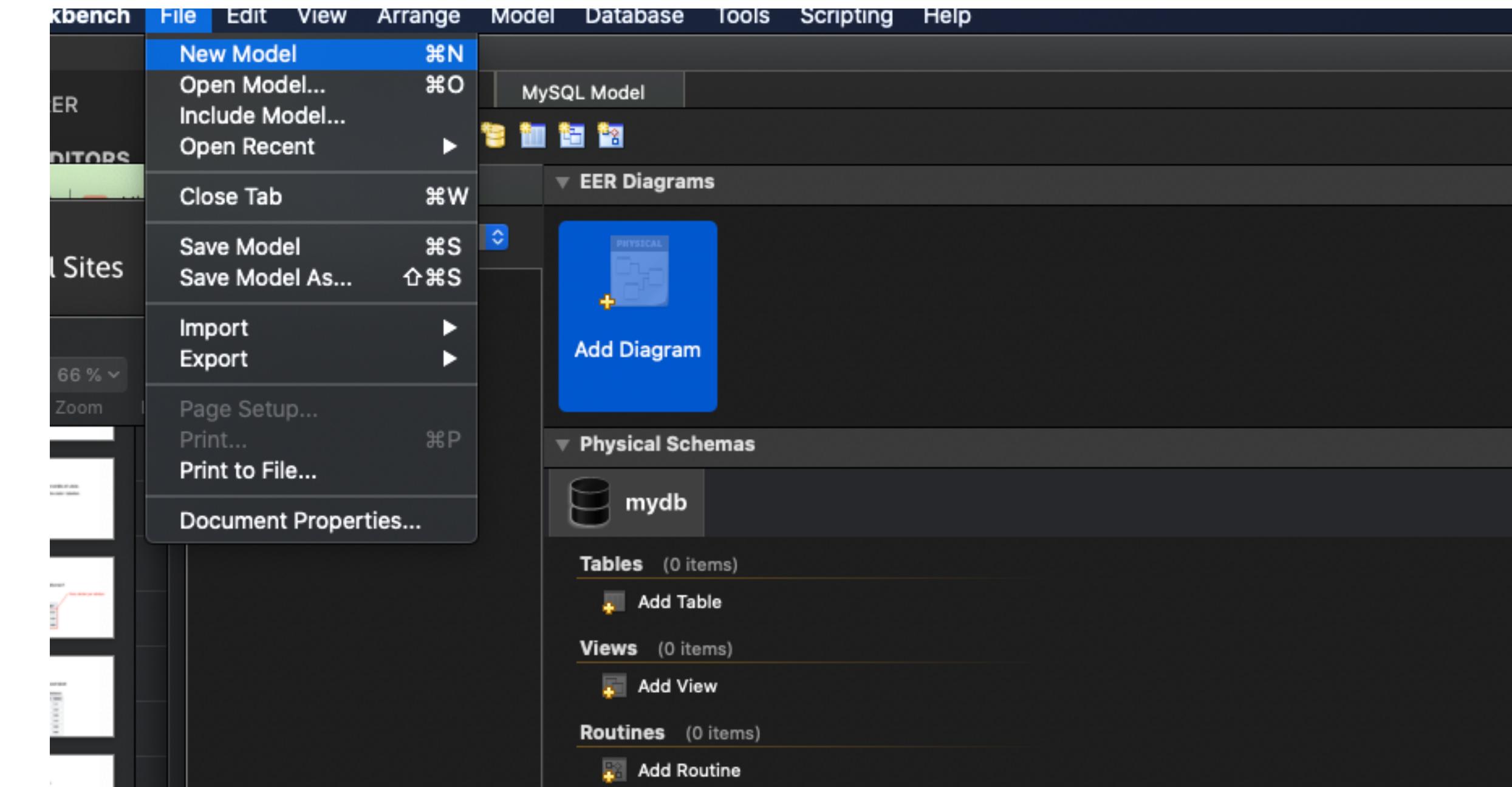
MySQL Workbench

Modellverktyget

- Modellverktyget hjälper oss att bygga ett schema över vår databas.
- Det gör det lättare för oss att få en översiktig bild över databasen.
- Vi kan även använda modellen för att skapa tabeller och liknande åt oss.

- Demo

- <https://www.youtube.com/watch?v=yq9iCb2JRHk>



MySQL Workbench

Modellverktyget

- Vad är det för skillnad mellan relationer med streckad linje och heldragen linje?
 - Streckad linje: non-identifying relationship
 - Heldragen linje: identifying relationship
 - Använd den heldragna och läs på nedanstående om du vill veta mer!
 - <https://vertabelo.com/blog/defining-identifying-and-non-identifying-relationships-in-vertabelo/>



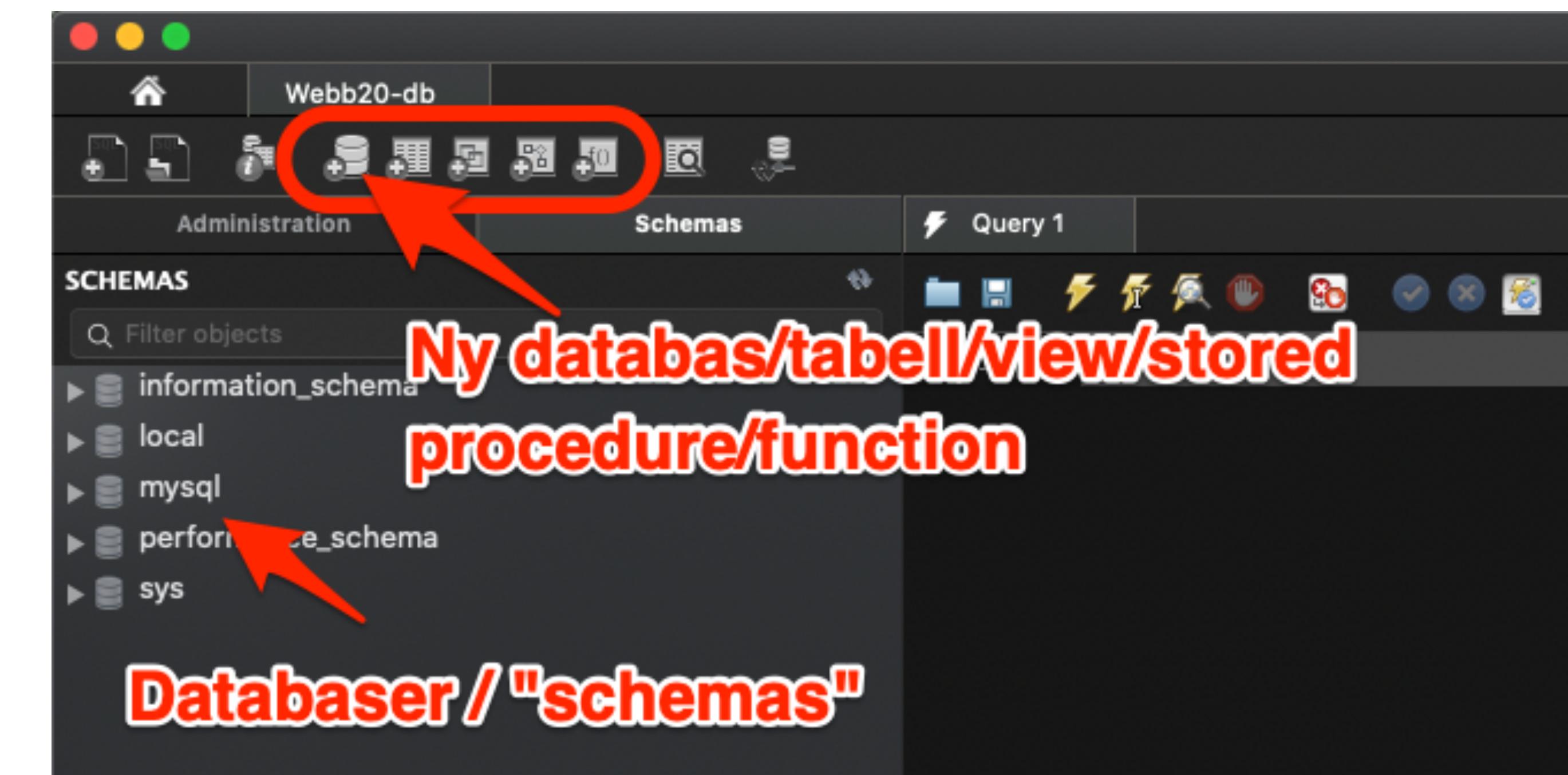
Lab MySQL Workbench

- Skapa en modell för orderhanteringssystemet (eller någon annan modell ni gjort).

MySQL Workbench

Skapa databas

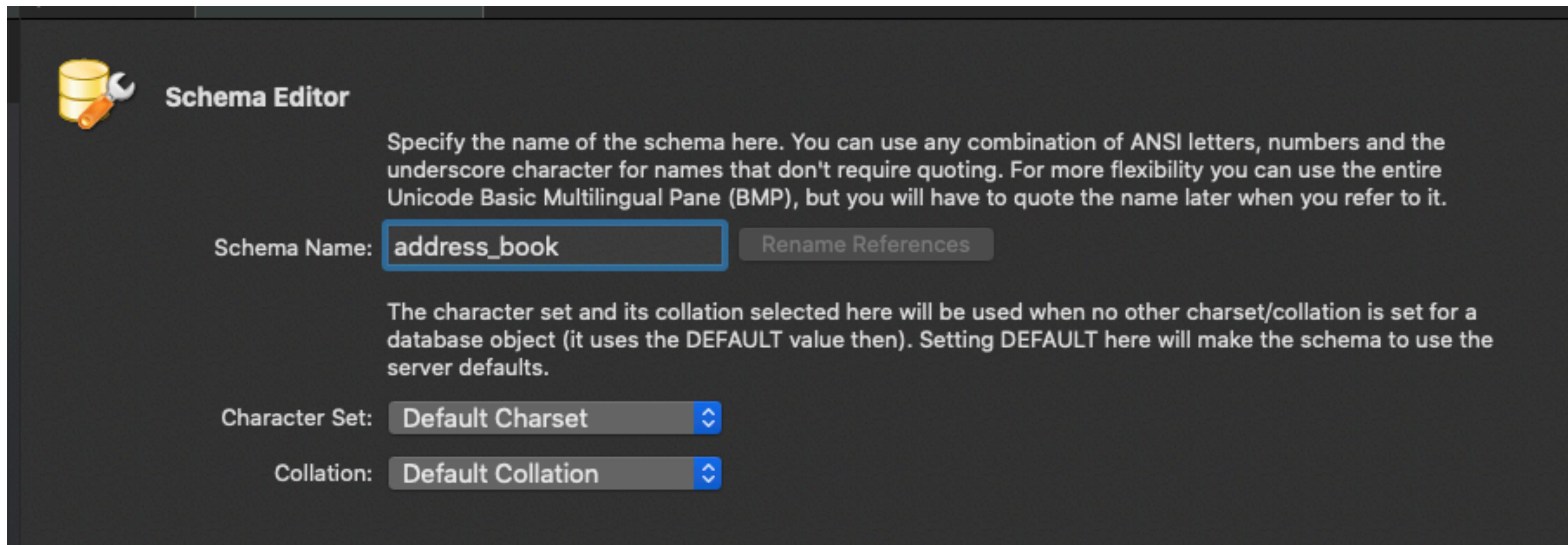
- Alla kommandon i SQL är egentligen text-baserade.
- Ett GUI är egentligen ett hjälpprogram. När vi klickar på en knapp för att skapa en databas så kommer programmet att skicka text-kommandon till databasen i bakgrunden.



MySQL Workbench

Skapa databas

- Vi ska skapa en adressbok med lämpliga egenskaper.



Charset

- Tabell som representerar tecken
- Ju fler bitar ett tecken tar upp, desto fler tecken kan uppsättningen ha, men desto mer plats tar varje tecken upp också.
- ASCII, ANSI
- Olika ISO-uppsättningar
- UTF8, UTF8MB4, UTF16, UTF32

Charset

UTF8MB4

- MySQLs UTF8 sparar max 3 bytes per tecken.
- Motsvarar “Basic Multilanguage Plane”
- UTF8MB4 ger utökat stöd, t ex emojis.

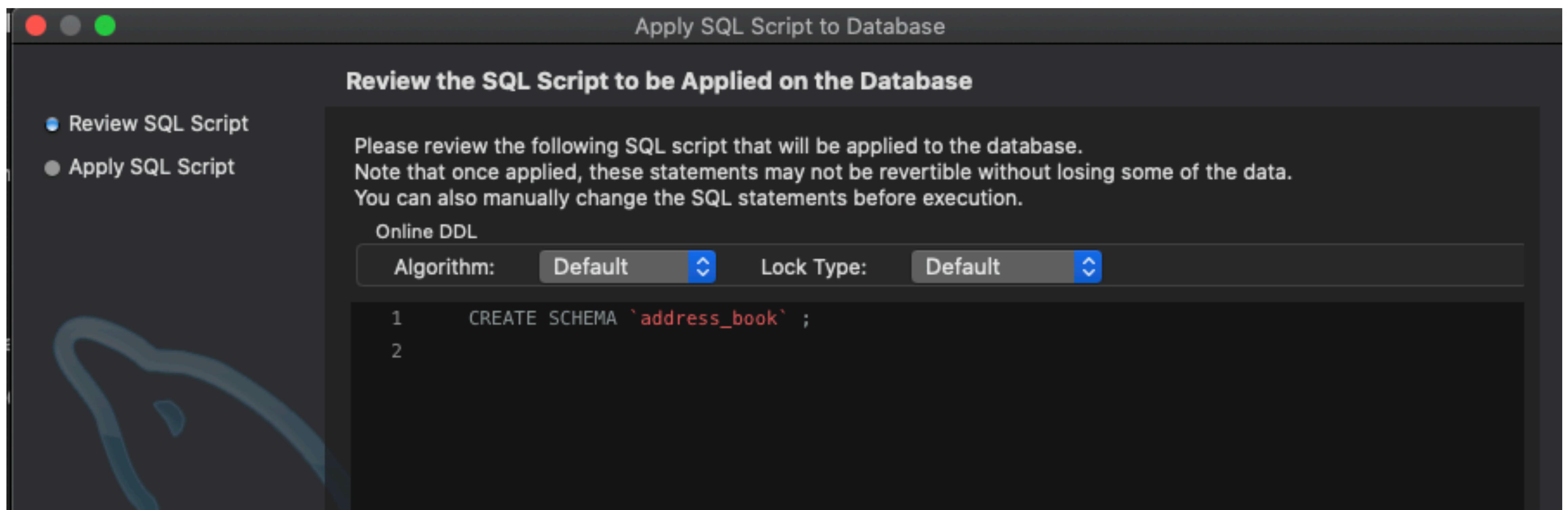
Collation

- Regler för sorteringsordning
- Ex: utf8mb4_swedish_ci
 - Reglerna gäller för utf8mb4
 - Svenska sorteringsregler, t ex att å, ä, ö kommer sist
 - ci = Case insensitive
- Man kan sätta standard charset och collation per server, databas och tabell.

MySQL Workbench

Skapa databas

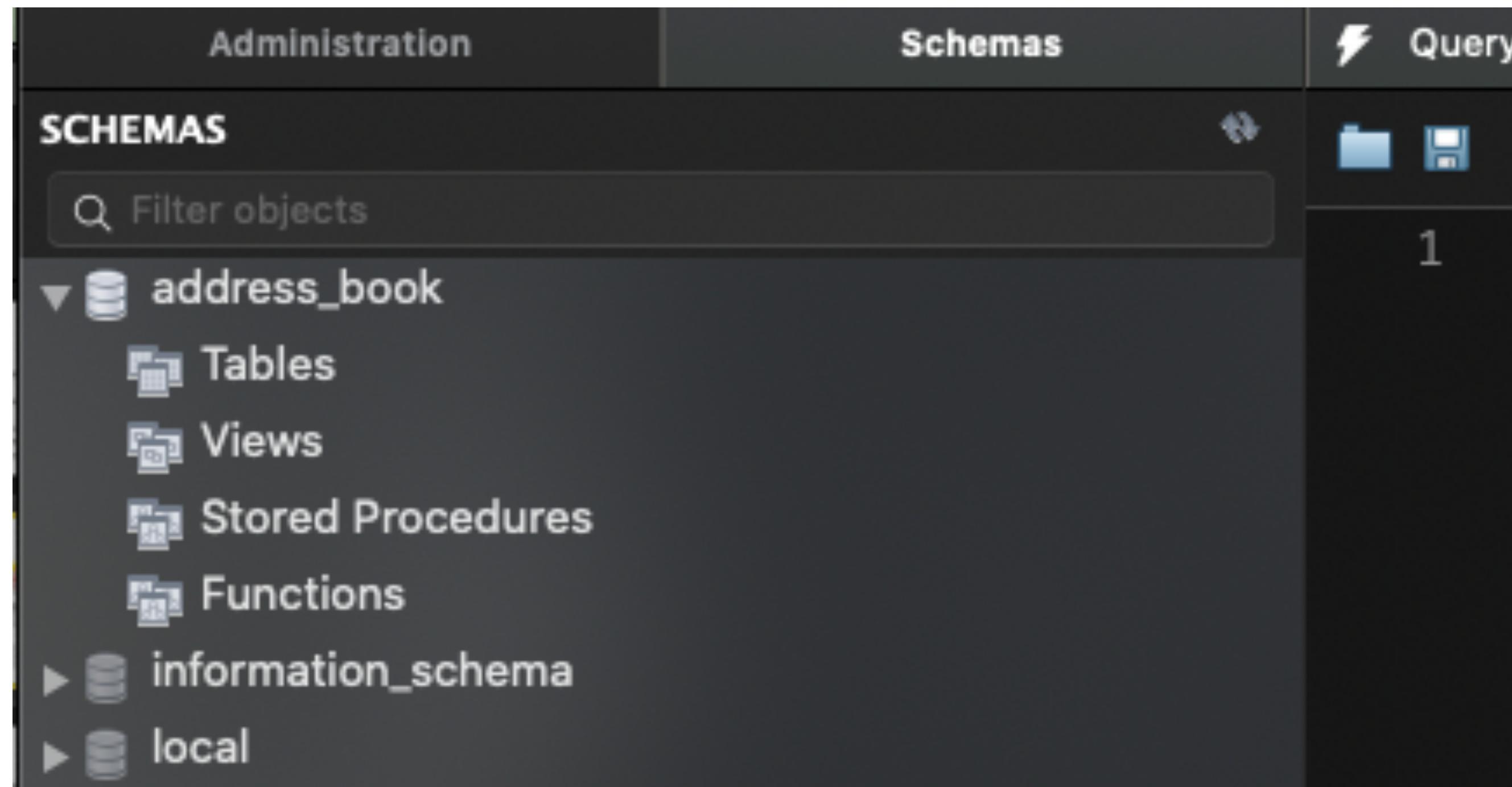
- Detta är koden som kommer att köras mot servern.



MySQL Workbench

Skapa databas

- Nu ska den finnas under Schemas till vänster.



MySQL Workbench

Skapa tabell / kolumner - flaggor

- PK - Primary Key
- NN - Not Null
- UQ - Unique
- BIN - Binary
- UN - Unsigned
- ZF - ZeroFill
- AI - Auto Increment
- G - Generated

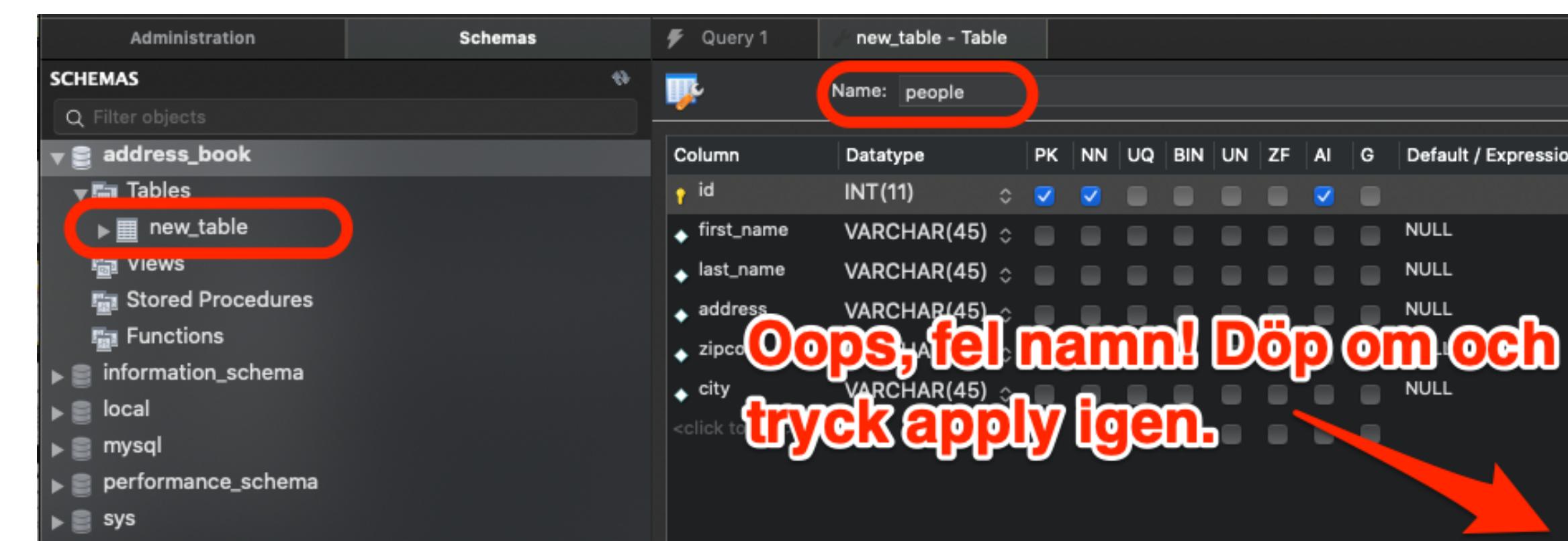
Name:	new_table	Column	Datatype	PK	NN	UQ	BIN	UN	ZF	AI	G	Def.	
		<input checked="" type="checkbox"/> id	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>						
		<input type="checkbox"/> first_name	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
		<input type="checkbox"/> last_name	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
		<input type="checkbox"/> address	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
		<input type="checkbox"/> zipcode	VARCHAR(10)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
		<input type="checkbox"/> city	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
		<click to edit>											

Columns Indexes Foreign Keys Triggers Partitioning Options

MySQL Workbench

Skapa tabell / kolumner - flaggor

- Skapa tabellen.



Redigera data

- Öppna tabellen.
- Lägg till några rader.
- Spara med Apply.

Fyll inte i id.

id	first_name	last_name	address	zipcode	city
HULL	Mikael	Olsson	Sandgatan 15	60221	Norrköping
HULL	Jessica	Persson	Gatan 4	12345	Staden
HULL	HULL	HULL	HULL	HULL	HULL

SCHEMAS

Filter objects

address_book

Tables

people

Views

Stored P

Function

information

local

Select Rows - Limit 1000

Table Inspector

Copy to Clipboard

Table Data Export Wizard

Table Data Import Wizard

Column

id

first_name

last_name

address

zipcode

city

<click to edit>

Apply SQL Script to Database

Review the SQL Script to be Applied on the Database

Please review the following SQL script that will be applied to the database.
Note that once applied, these statements may not be revertible without losing some of the data.
You can also manually change the SQL statements before execution.

```
1 INSERT INTO `address_book`.`people` (`first_name`, `last_name`, `address`, `zipcode`, `city`)
2 INSERT INTO `address_book`.`people` (`first_name`, `last_name`, `address`, `zipcode`, `city`)
3
```

id	first_name	last_name
1	Mikael	Olsson
2	Jessica	Persson

Lab MySQL Workbench

- Skapa tabeller och kolumner för era modeller.
- Lägg till ett par rader i varje tabell.

Manipulera data

CRUD

- Create
- Read
- Update
- Delete

Manipulera data

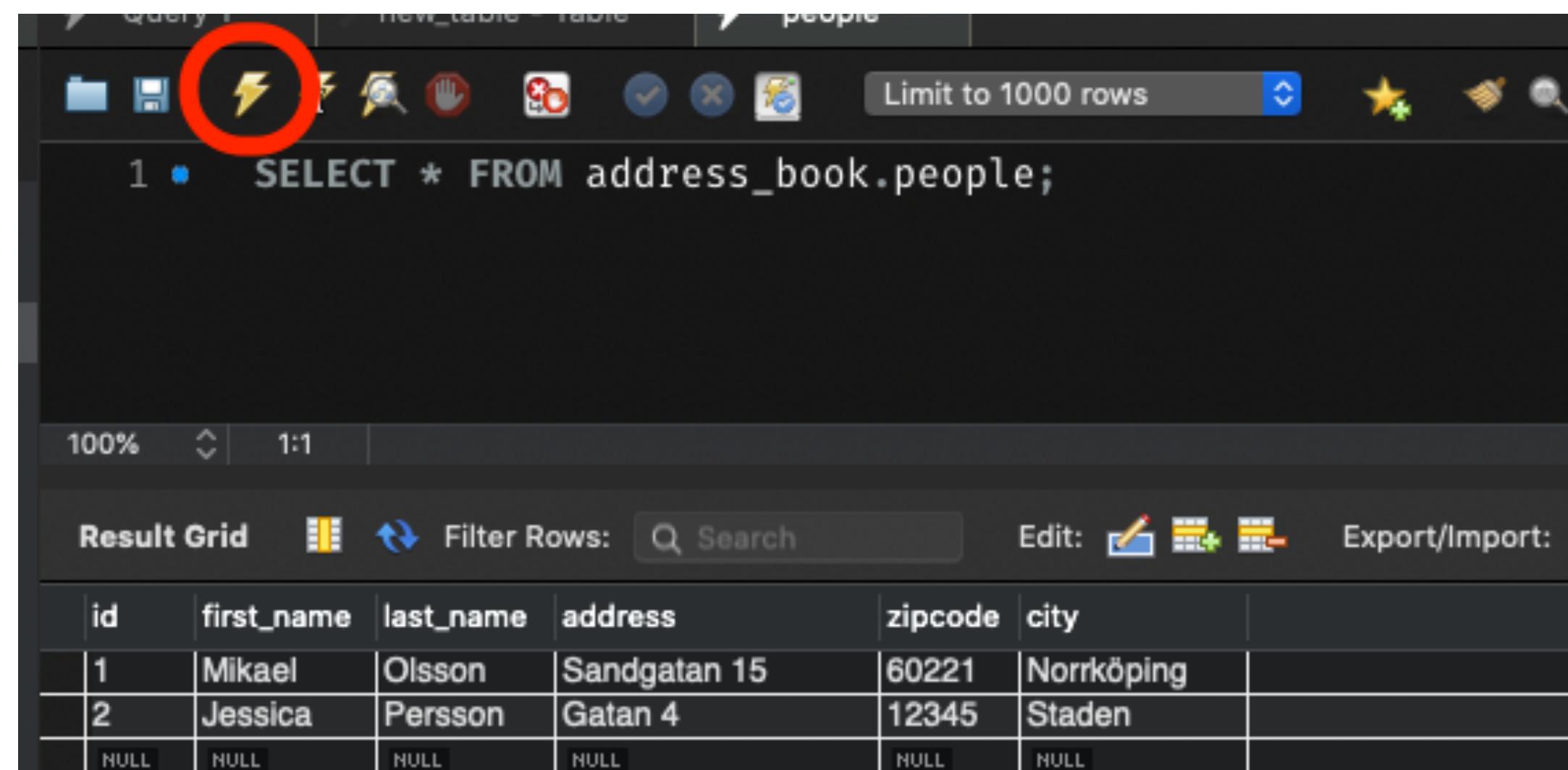
SQL - Structured Query Language

- INSERT INTO - Create
- SELECT - Read
- UPDATE - Update
- DELETE - Delete
- -- Kommentar
- /* Också kommentar */

Manipulera data

SELECT - Hämta/visa data

- `SELECT [fields] FROM [table];`
- `SELECT FirstName, LastName FROM Person;`
- `SELECT * FROM Person; -- Obs, prestandakrävande!`



The screenshot shows the MySQL Workbench interface. At the top, there's a toolbar with various icons, one of which is circled in red. Below the toolbar, a query editor window displays the following SQL command:

```
1 •  SELECT * FROM address_book.people;
```

Below the query editor is a result grid showing the data from the query. The columns are labeled: id, first_name, last_name, address, zipcode, and city. There are two rows of data:

	id	first_name	last_name	address	zipcode	city
1	1	Mikael	Olsson	Sandgatan 15	60221	Norrköping
2	2	Jessica	Persson	Gatan 4	12345	Staden
	NULL	NULL	NULL	HULL	NULL	NULL

Manipulera data

- `INSERT INTO [table] ([fields]) VALUES ([values]);`
- `INSERT INTO Person (`FirstName`, `LastName`) VALUES ('Mikael', 'Olsson');`
- Lägg märke till att backtick ` inte är detsamma som apostrof '. Vilket används var?
- Det finns varianter på `INSERT INTO`.

Manipulera data - Sortera

- `SELECT [fields] FROM [table] ORDER BY [field];`
- `SELECT FirstName, LastName
FROM Person
ORDER BY LastName DESC, FirstName;`

Manipulera data - Avgränsa

- `SELECT [fields] FROM [table] LIMIT [number], [offset];`
- `SELECT FirstName, LastName
FROM Person
LIMIT 3, 0; -- Ger de tre första träffarna`
- MS SQL Server: `SELECT TOP 3 FROM [...]`

Manipulera specifik data - WHERE

- `SELECT [fields] FROM [table] WHERE [condition];`
- `SELECT FirstName, LastName FROM Person WHERE id = 23;`
- `SELECT * FROM Person WHERE FirstName = 'Mikael';`

Manipulera specifik data

- Operatorer
 - Relationsoperatorer
 - <, >, !=, >=, <=, =
 - Logiska operatorer
 - AND
 - OR
 - NOT
 - BETWEEN

Manipulera specifik data - WHERE

- ```
SELECT FirstName, LastName
FROM Person
WHERE age >= 23;
```
- ```
SELECT FirstName, LastName
FROM Person
WHERE age BETWEEN 20 AND 65
OR NoOfCars > 5;
```

Manipulera specifik data - LIKE

- ```
SELECT FirstName, LastName
FROM Person
WHERE LastName LIKE "Ols%";
```
- % = jokertecken

# Manipulera data

- UPDATE [table] SET [field1] = '[value1]' WHERE [condition];
- UPDATE Person SET  
    FirstName = 'Mikael',  
    LastName = 'Olsson'  
WHERE id = 23;
- Varför tittade vi på WHERE innan vi började uppdatera?

# Manipulera data

- Varför är WHERE viktigt här?
  - `DELETE FROM [table] WHERE [condition];`
  - `DELETE FROM Person WHERE id = 23;`

# Lab

- Allt i labben ska göras med SQL, inte med GUI (dvs skriva queries, inte använda knappar och formulär).
- Lägg till 10 personer i adressboken. Låt minst två personer ha namn som börjar på J.
- Ge alla personer adresser.
- Ge några personer 0 bilar, några 1 bil och några 2 bilar.
- Uppdatera 2 personers adress och telnr.
- Ta bort 2 personer.
- Visa alla personer som börjar på J.

# Förberedelser inför nästa tillfälle

- Gör en modell över era TV-tablåer i Workbench.
- Skapa tabellerna med hjälp av Workbench -> Database -> Forward Engineer.