

# Topic 02 - Basic SQL

BDM3302: Data Management

# What is a Database?

- A collection of related information
  - Phone book
  - Name your 5 best friends
  - Facebook's user base
  - Shopping list
- Database can be stored in:
  - Paper
  - Your mind
  - Computer
  - Etc.

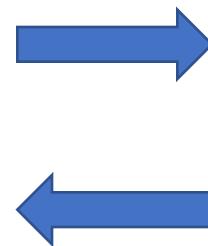
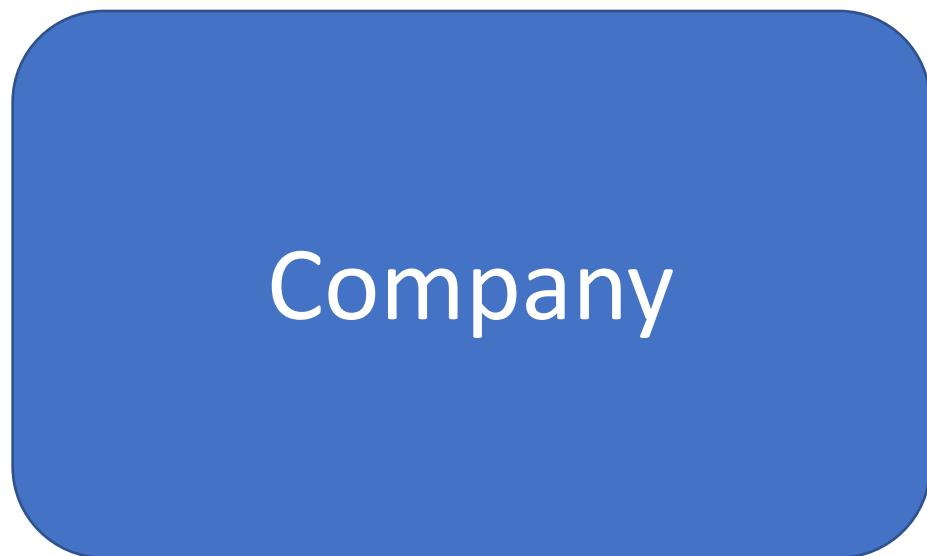
# Database Management Systems (DBMS)

- A special software program that help users create and maintain a database.
- Makes it easy to manage large amounts of information
- Handles security
- Backups
- Import and export data
- Concurrency
- Interacts with other software applications

# Database Management Systems (DBMS)

- Represent data as two-dimensional tables called relations or files
- Each table contains data on entity and attributes
- Table: grid of columns and rows
  - Rows (tuples): Records for different entities
  - Fields (columns): Represents attribute for entity
  - Key field: Field used to uniquely identify each record
  - Primary key: Field in table used for key fields
  - Foreign key: Primary key used in second table as look-up field to identify records from original table

# A Company Database Diagram



A company interact with DBMS in order to create, read, update and delete information.

\*\*\*\*\* C. R. U. D. operations \*\*\*\*\*

# Types of Databases

## Relational Database (SQL)

- Organize data into one or more tables.
  - Each table has columns and rows
  - A unique key identifies each row

## Non-Relational (noSQL / not just SQL)

- Organize data is anything but a traditional table
  - Key-value stores
  - Documents (JSON, XML, etc.)
  - Graphs
  - Flexible Tables

# Relational Database (SQL)

- Relational Database Management Systems (RDBMS)
  - Help users create and maintain a relational database
    - mySQL, Oracle, PostgreSQL, MariaDB, etc.
- Structured Query Language (SQL)
  - Standardized language for interacting with RDBMS
  - Used to perform C.R.U.D. operations, and other administrative tasks (user management, security, backup, etc.)
  - Used to define tables and structures
  - SQL code used on one RDBMS is not always portable to another without modification.

# RELATIONAL DATABASE TABLES

SUPPLIER

Columns (Attributes, Fields)					
Supplier_Number	Supplier_Name	Supplier_Street	Supplier_City	Supplier_State	Supplier_Zip
8259	CBM Inc.	74 5 <sup>th</sup> Avenue	Dayton	OH	45220
8261	B. R. Molds	1277 Gandolly Street	Cleveland	OH	49345
8263	Jackson Composites	8233 Micklin Street	Lexington	KY	56723
8444	Bryant Corporation	4315 Mill Drive	Rochester	NY	11344

Key Field  
(Primary Key)

A relational database organizes data in the form of two-dimensional tables. Illustrated here are tables for the entities SUPPLIER and PART showing how they represent each entity and its attributes. Supplier Number is a primary key for the SUPPLIER table and a foreign key for the PART table.

# RELATIONAL DATABASE TABLES

PART

Part_Number	Part_Name	Unit_Price	Supplier_Number
137	Door latch	22.00	8259
145	Side mirror	12.00	8444
150	Door molding	6.00	8263
152	Door lock	31.00	8259
155	Compressor	54.00	8261
178	Door handle	10.00	8259

Primary Key                      Foreign Key

A relational database organizes data in the form of two-dimensional tables. Illustrated here are tables for the entities SUPPLIER and PART showing how they represent each entity and its attributes. Supplier Number is a primary key for the SUPPLIER table and a foreign key for the PART table.

# Non-Relational Databases (noSQL/ not just SQL)

- Non-Relational Database Management Systems (NRDBMS)
  - Help users create and maintain a non-relational database
    - mongoDB, DynamoDB, Apache Cassandra, firebase, etc.
- Implementation Specific
  - Any non-relational database falls under this category, so there's no set language standard.
  - Most NRDBMS will implement their own language for performing C.R.U.D. and administrative operation on the database.

# Database Queries

- Queries are requests made to the database management system for specific information. (*basically, you are asking the DBMS for a specific information*)
- As the database's structure become more and more complex, it becomes more difficult to get the specific pieces of information we want.
- A google search is a query. (use English or other languages)
- SQL (use a specific language for database management systems)

# Structured Query Language (SQL)

- A language used for interacting with RDBMS
- You can use SQL to get the RDBMS to do things for you.
  - CRUD data
  - Create and manage databases
  - Design and create database tables
  - Perform administrative tasks (user management, security, import/export, etc)

# SQL

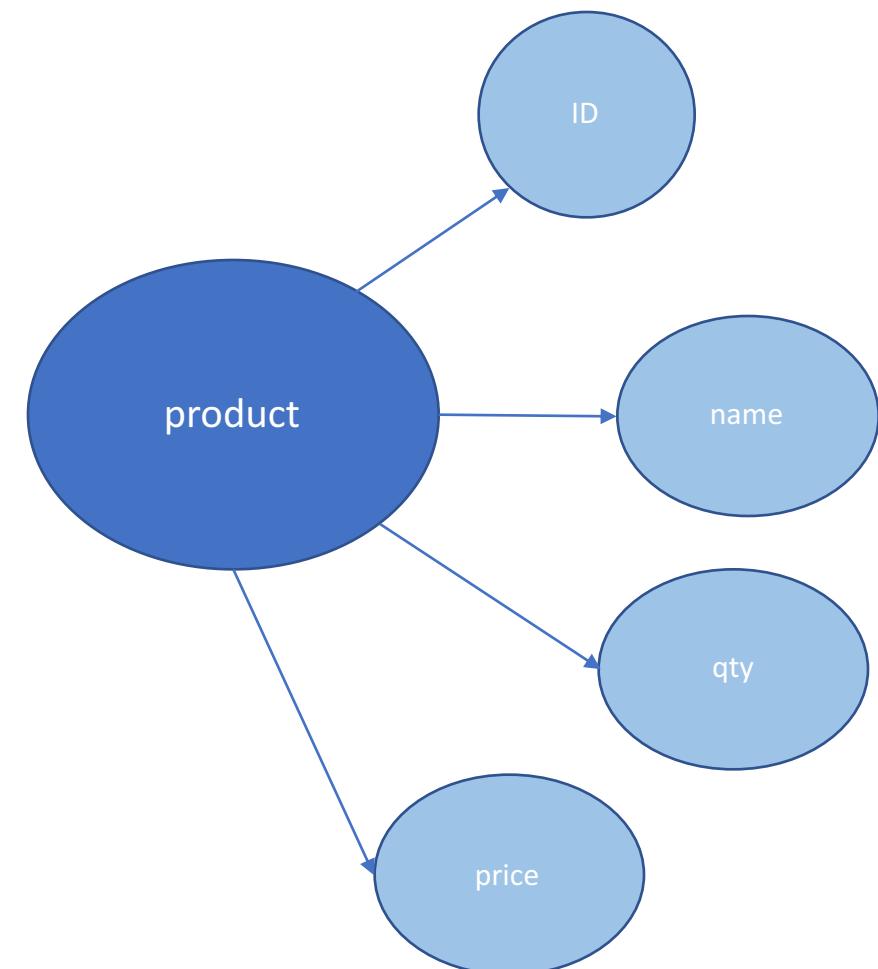
- Is actually a hybrid language, there are 3 types of languages in one.
- Data Definition Language (DDL)
  - Used for defining database schemas.
- Data Control Language (DCL)
  - Used for controlling access to the data in the database.
  - User and permissions management.
- Data Manipulation Language (DML)
  - Used for inserting, updating and deleting data from the database.

# Queries

- A query is a set of instructions given to the RDBMS (written in SQL) that tell the RDBMS what information you want it to retrieve for you.
- TONS of data in a DB
- Often hidden in a complex schema
- Goal is to only get the data you need
  1. **SELECT** employee.name, employee.age
  2. **FROM** employee
  3. **WHERE** employee.salary > 30000

Table: Product

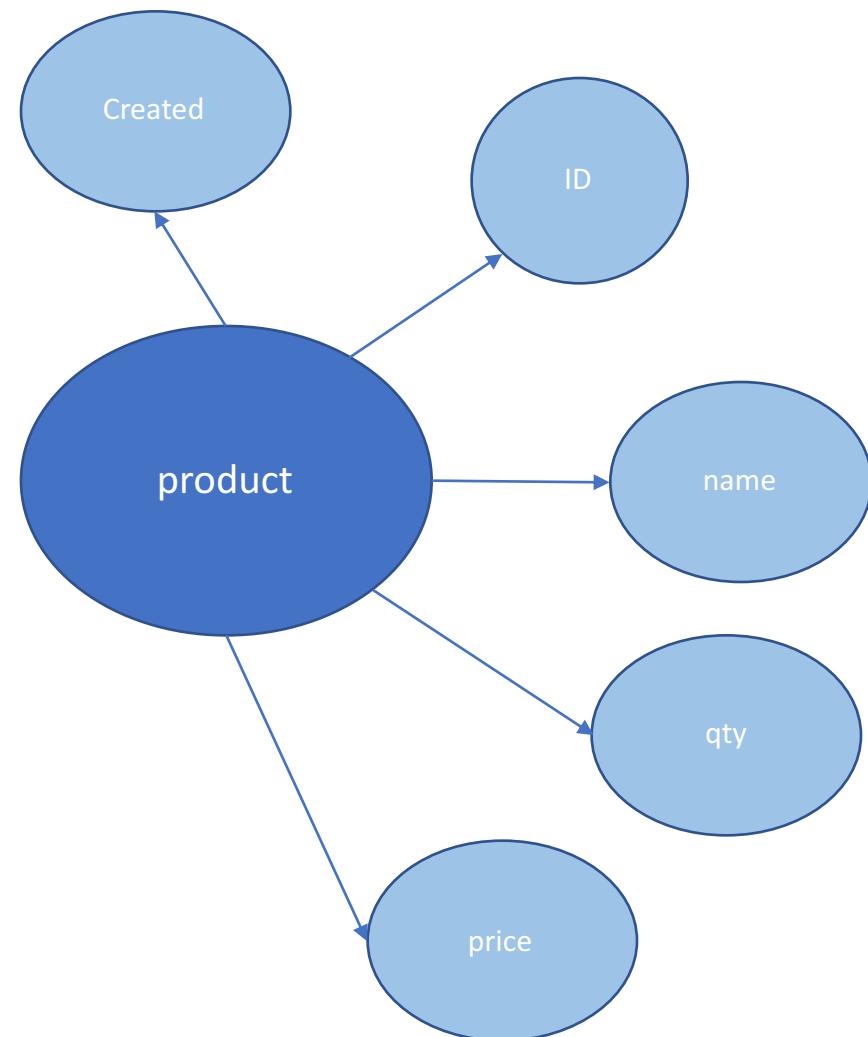
Row/Tuples/Records	Column/Data Field/Attribute			
	ID	name	qty	price
1	p1	10	100	
2	p2	20	200	
3	p3	30	300	



1. Define Primary Key
2. Define Column Schema

Users	Staff, Manager, Supplier, Account	
Process	Acquisition, Purchase, Inventory, Order	
Owner	Sale and Marketing Team	
ID	rule	Last reviewed on
id	Unique integer	10 Nov 2021
name	Max 200 character Unique name Lower case	20 Nov 2021
qty	Max 1000	10 Nov 2021
price	Numeric value	12 Nov 2021
Created	Date Time	14 Nov 2021

process	Data Query frequency	Backup policy
Acquisition	daily	daily
Order	sale transaction	daily
Inventory	daily	weekly
Purchase	daily	weekly



**data**

ID	name	qty	price
1	p1	10	100
2	p2	20	200
3	p3	30	300

```
create table product
(
    id      INT      NOT NULL PRIMARY KEY ,
    name   varchar(20) NOT NULL ,
    qty    integer    NOT NULL ,
    price  double    NOT NULL
)
```

**data**

<b>id</b>	<b>name</b>	<b>qty</b>	<b>price</b>
1	p1	10	100
2	p2	20	200
3	p3	30	300

**insert**      **into**  
**product**  
( id , name , qty , price )  
**value**  
( '1' , 'p1' , '10' , '100' )

**insert into product(id,name,qty,price) value('1','p1','10','100')**

**data**

<b>id</b>	<b>name</b>	<b>qty</b>	<b>price</b>
1	p1	10	100
2	p2	20	200
3	p3	10	300

**update product****set****qty = '20',****price='100'****where****name = 'p1'****update product set qty = '20', price='100' where name = 'p1'**

**data**

<b>id</b>	<b>name</b>	<b>qty</b>	<b>price</b>
1	p1	10	100
2	p2	20	200
3	p3	10	300

**delete from product**

**where**

**name = 'p1'**

**delete from product where name = 'p1'**

**Data: product**

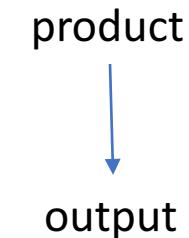
<b>id</b>	<b>name</b>	<b>qty</b>	<b>price</b>
1	p1	10	100
2	p2	20	200
3	p3	10	300

**select \* from product**

Relational Algebra

**output**

<b>id</b>	<b>name</b>	<b>qty</b>	<b>price</b>
1	p1	10	100
2	p2	20	200
3	p3	10	300



**Data: product**

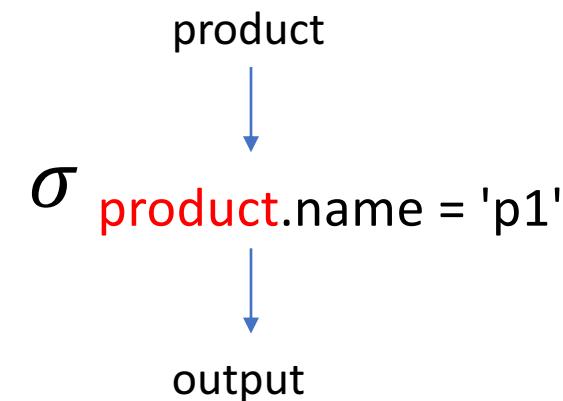
<b>id</b>	<b>name</b>	<b>qty</b>	<b>price</b>
1	p1	10	100
2	p2	20	200
3	p3	10	300

**select \* from product**  
**where**  
**name = 'p1'**

Relational Algebra

**output**

<b>id</b>	<b>name</b>	<b>qty</b>	<b>price</b>
1	p1	10	100



**Data: product**

<b>id</b>	<b>name</b>	<b>qty</b>	<b>price</b>
1	p1	10	100
2	p2	20	200
3	p3	10	300

**output**

<b>id</b>	<b>name</b>	<b>qty</b>	<b>price</b>
3	p3	10	300

**select \* from product**

**where**

**qty >= 10**

**and**

**price > 200**

Relational Algebra

product

$\sigma_{\text{product}.\text{qty} \geq 10 \text{ AND } \text{product}.\text{price} > 200}$

output

**Data: product**

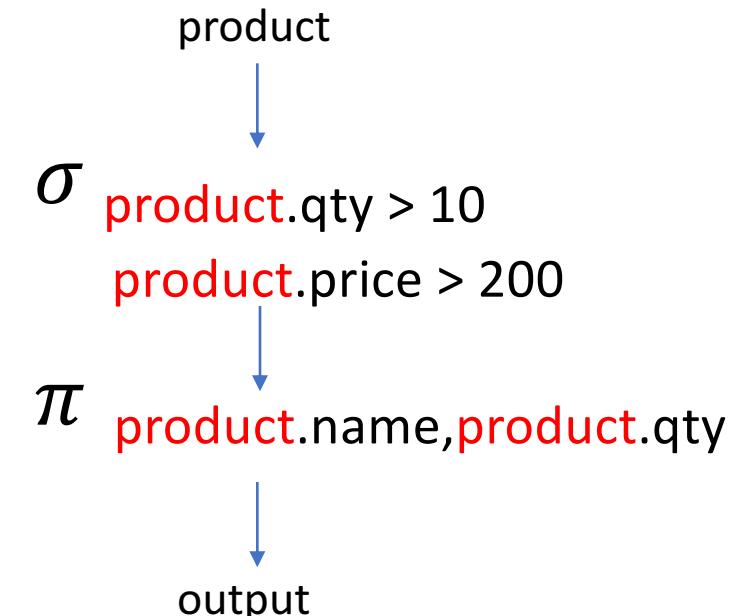
<b>id</b>	<b>name</b>	<b>qty</b>	<b>price</b>
1	p1	10	100
2	p2	20	200
3	p3	10	300

**select**    **name, qty**    **from**    **product**  
**where**  
**qty >= 10**  
**and**  
**price > 200**

Relational Algebra

**output**

<b>name</b>	<b>qty</b>
p3	10



**Data: product**

<b>id</b>	<b>name</b>	<b>qty</b>	<b>price</b>
1	p1	10	100
2	p2	20	200
3	p3	10	300

**output**

<b>product name</b>	<b>quantity</b>
p3	10

**select    name as `product name`, qty as quantity**

**from    product**

**where**

**qty >= 10**

**and**

**price > 200**

Relational Algebra

product

$\sigma$  **product.qty >= 10**  
**product.price > 200**

$\pi$  **product.name as `product name` ,**

**product.qty as quantity**

output

## Data: product

id	name	qty	price
1	p1	10	100
2	p2	20	200
3	p3	10	300

## output

product name	quantity	total
p3	10	3000

select    name as `product name`,  
           qty as quantity,  
           qty\*price as total

from    **product**

where

qty > 10

and

price > 200

Relational Algebra

product

$\sigma$  **product.qty >= 10**  
**product.price > 200**

$\pi$  **product.name as product name ,**

**product.qty as quantity,**  
**product.qty\*product.price as total**

output

**Data: product**

pid	name	qty	price
1	p1	10	100
2	p2	20	200
3	p3	10	300

**Data: Supplier virtual Stock**

no	sid	pid	qty	price
1	s1	1	10	50
2	s2	1	20	60
3	s1	2	30	70
4	s2	2	40	40
5	s1	3	0	100

**output**

name	qty	sqty	Total
p1	10	30	40
p2	20	70	90
P3	10	0	10

- Locate product with empty virtual stock
- Display total stock for each product

## Data Management > SQL > CRUD > Sub Query for a stock condition process

Data: **product**

pid	name	qty	price
1	p1	10	100
2	p2	20	200
3	p3	10	300

Data: **VirtualStock**

no	sid	pid	qty	price
1	s1	1	10	50
2	s2	1	20	60
3	s1	2	30	70
4	s2	2	40	40
5	s1	3	0	100

```

select
    p.name, p.qty
    ( select sum(v.qty)
        from VirtualStock as v
        where p.productId = v.productId
    ) as sqty
    ,
    p.qty + ( select sum(v.qty)
        from VirtualStock as v
        where v.productId = p.productId
    ) as Total
from Product as p

```

output

name	qty	sqty	Total
p1	10	30	40
p2	20	70	90
P3	10	0	10

```

select
    p.productName,
    p.qty,
    (select sum(vs.qty)
        from VirtualStock as vs
        where vs.productId = p.productId
    ) as sqty
    ,
    p.qty+(select sum(vs.qty)
        from VirtualStock as vs
        where vs.productId = p.productId
    ) as `total`
from
    Product as p

```

**Data: Product**

pid	name	qty	price
1	p1	10	100
2	p2	20	200
3	p3	10	300

**Data: VirtualStock**

no	sid	pid	qty	price
1	s1	1	10	50
2	s2	1	20	60
3	s1	2	30	70
4	s2	2	40	40
5	s1	3	0	100

**Data: Acquisition**

no	sid	pid	qty	price	total	status	commit
1	s1	1	10	50	500	yes	no
2	s1	1	10	60	600	yes	no
3	s3	3	20	100	2000	Back-order	no

- Locate product with qty <= 10
- Reorder with qty = 20,  
go with lower price first
- If virtual qty =0, put back-order status
- Create stock acquisition log
- Set review field as no

```

select
v.supplierId as sid,
p.productId as pid,
v.qty,
v.price,
"no" as commit

```

```

from
Product as p,
VirtualStock as v

```

```

where
p.productId = v.productId
and
p.qty <=10

```

sid	pid	qty	price	commit
s1	1	10	50	no
s2	1	20	60	no
s1	3	0	100	no

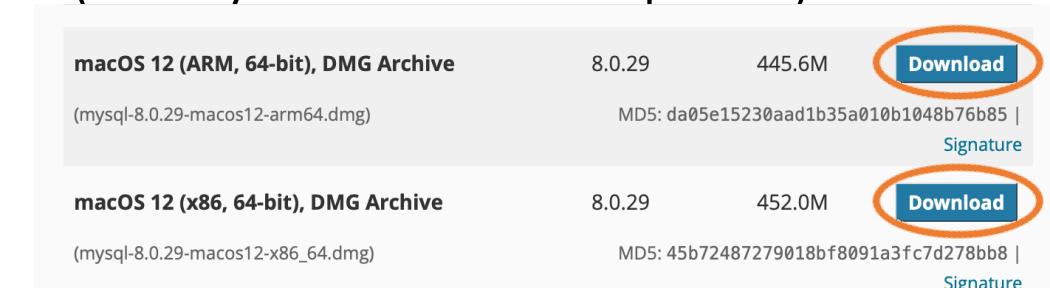
# SQL Lab Practical class – Let's try this !!!

- Tools & files requirements in our lab practical class
  - **MySQL** – The database management software that requires to import database
  - **PopSQL** – The ease-of-use application which manages database using SQL commands to display the result
  - **mysqlsampledatabase.sql** – The database file that uses to import in MySQL. This file is available in LMS

# SQL Lab Practical class – Let's try this !!!

- MySQL installations:
  - Windows
    - Google search: "mysql community server"
    - Click "[Download MySQL Community Server](#)"
    - Click this picture below
- MySQL installations:
  - macOS
    - Google search: "mysql community server"
    - Click "[Download MySQL Community Server](#)"
    - Click Operating System: macOS
    - Choose 1 download button from 2 of these (check your MacBook compatible):

The screenshot shows the MySQL Installer for Windows page. It features a large image of the Windows logo and a 'Go to Download Page' button. Below it, there are two download options: 'Windows (x86, 32 & 64-bit), MSI Installer' and 'Windows (x86, 64-bit), MSI Installer'. The '64-bit' option has a blue 'Download' button highlighted with a red oval.

      - Click Operating System: Microsoft Windows
      - Click "**Windows (x86, 64-bit), MSI Installer**" download button

The screenshot shows the MySQL download page for macOS. It lists two DMG Archive options: 'macOS 12 (ARM, 64-bit), DMG Archive' and 'macOS 12 (x86, 64-bit), DMG Archive'. Both download buttons are highlighted with red ovals.

      - Select "**No thanks, just start my download**" (See next page)

**Note for macOS - If your MacBook uses M1 chip, you have to choose macOS 12 (ARM, 64-bit), DMG Archive instead.**

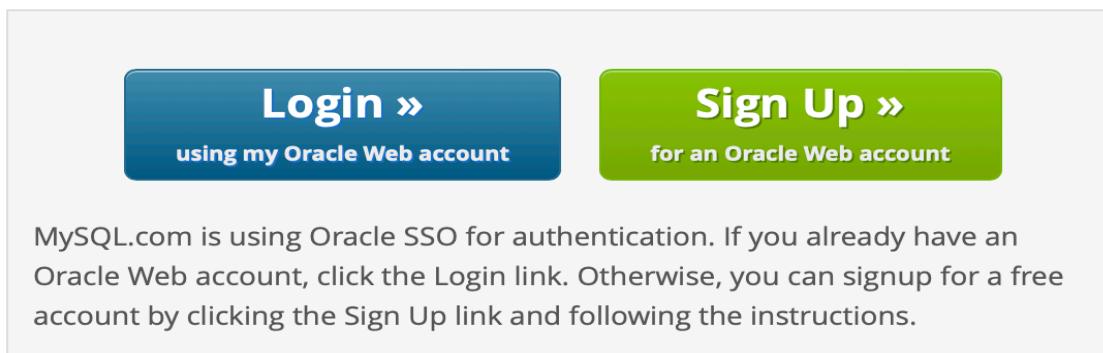
# SQL Lab Practical class – Let's try this !!!

## ④ MySQL Community Downloads

[Login Now](#) or [Sign Up](#) for a free account.

An Oracle Web Account provides you with the following advantages:

- Fast access to MySQL software downloads
- Download technical White Papers and Presentations
- Post messages in the MySQL Discussion Forums
- Report and track bugs in the MySQL bug system



The image shows a snippet of the MySQL Community Downloads landing page. It features two main buttons: a blue 'Login »' button with 'using my Oracle Web account' below it, and a green 'Sign Up »' button with 'for an Oracle Web account' below it. Below these buttons is a text box containing instructions about Oracle SSO authentication. At the bottom of the snippet, there is a link 'No thanks, just start my download.' which is circled in orange.

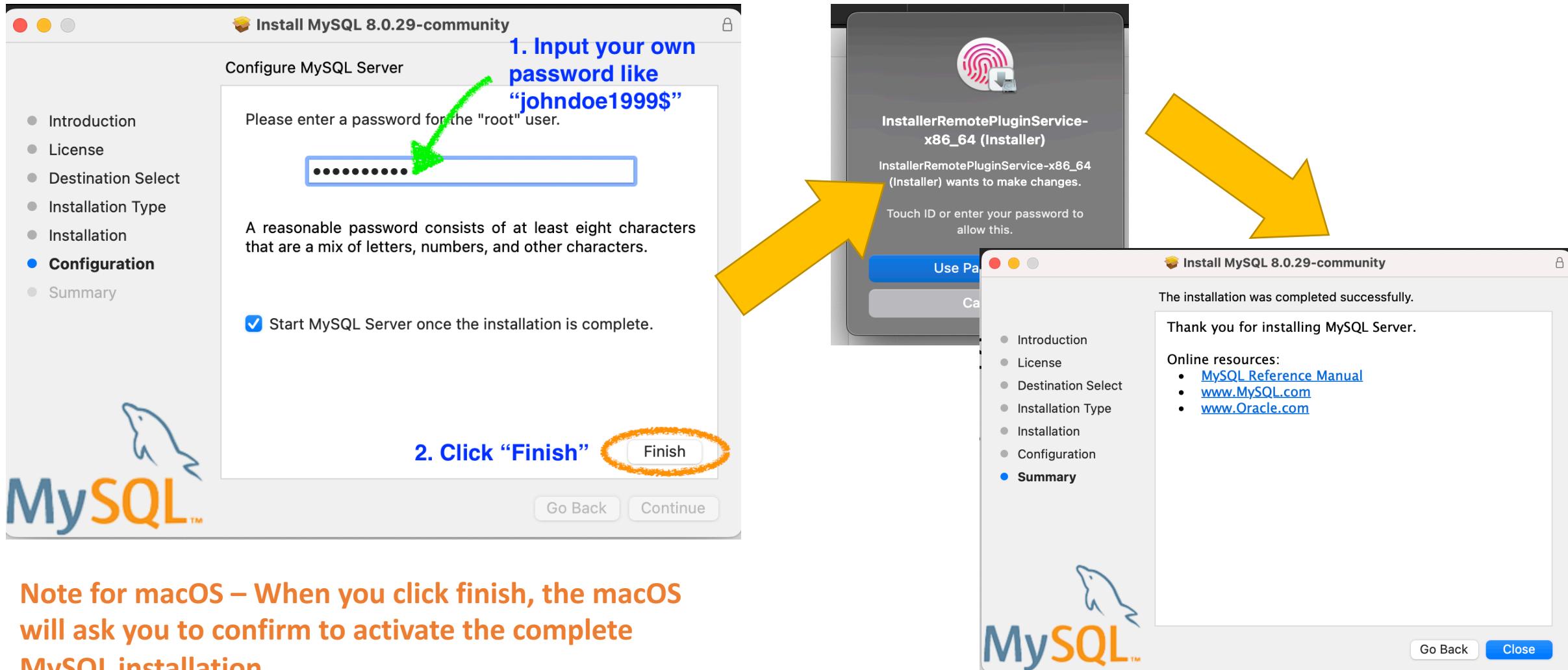
MySQL.com is using Oracle SSO for authentication. If you already have an Oracle Web account, click the Login link. Otherwise, you can signup for a free account by clicking the Sign Up link and following the instructions.

**When the MySQL file is complete download, please double click to install and follow the instructions.**

# SQL Lab Practical class – Let's try this !!!

- MySQL installation setup – macOS
  - Double click installation file “**mysql-8.0.29-macos12-x86\_64.dmg**”
  - Double click “**mysql-8.0.29-macos12-x86\_64.pkg**”
  - When the setup screen appear, Click “**Allow**” button
  - Click “**Continue**” button and then click “**Agree**” button
  - Click “**Install**” button and input your MacBook password or fingerprint to ready install
  - During installation, the screen will ask you to create the MySQL password for security. Choose “**Use Strong Password Encryption**” and then click “**Next**”
  - Input your MySQL appropriate password including alphabets, numbers, and symbols at least 8 characters (e.g., johndoe1999\$) (See next page)

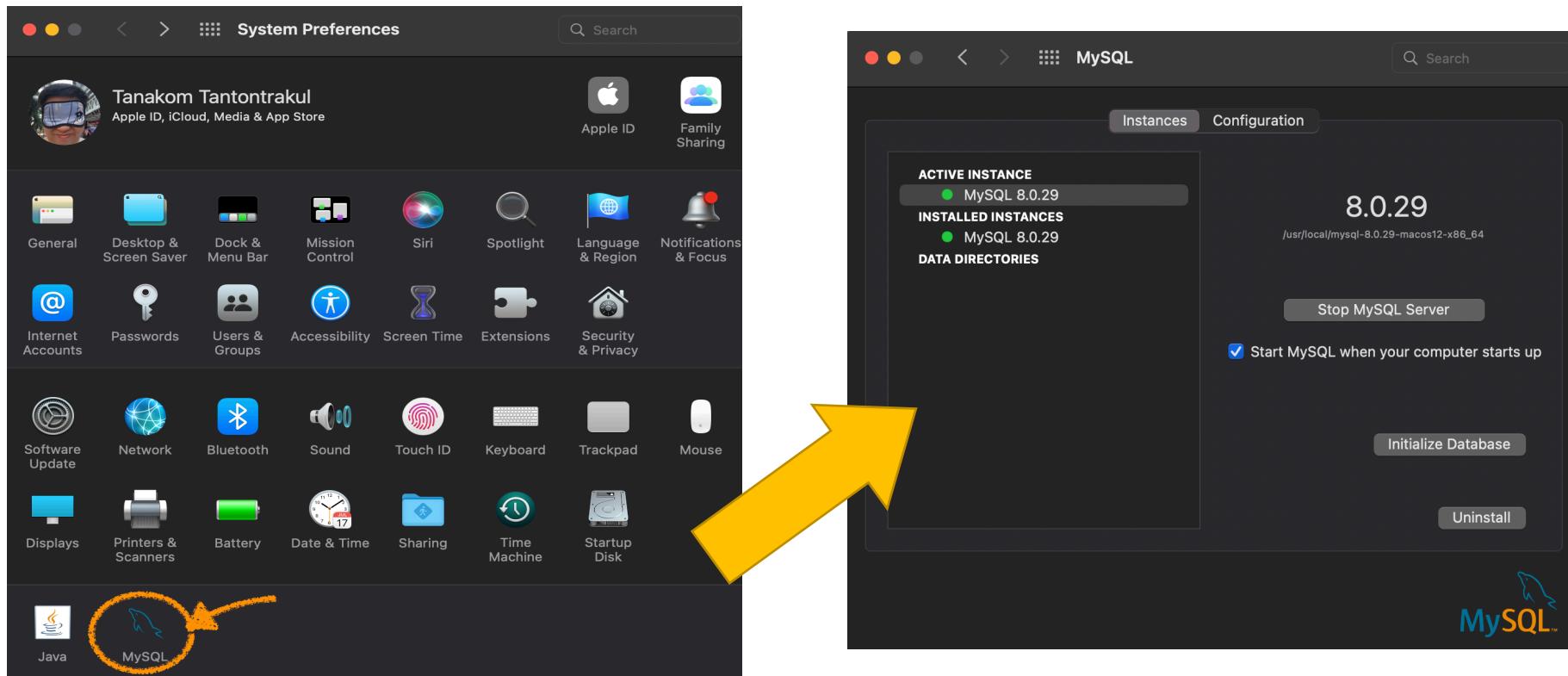
# SQL Lab Practical class – Let's try this !!!



# SQL Lab Practical class – Let's try this !!!

- macOS

- When you complete MySQL installation, you can check how MySQL activate by type and choose “System Preferences” in Spotlight Search to open. If you see like these pictures below, please close it and follow next step.



# SQL Lab Practical class – Let's try this !!!

- Import database into MySQL – macOS
  - Open “Terminal” app in your macOS by typing “Terminal” and choose in Spotlight Search
  - Need some unix commands that allow to enter “mysql” in Terminal by typing and press enter key following like these below:

```
Prompt$ echo 'export PATH=/usr/local/mysql/bin:$PATH' >> ~/.bash_profile  
Prompt$ . ~/.bash_profile
```

- Then you can run mysql by typing and press enter key following like these below (See next page):

```
Prompt$ mysql -u root -p
```

# SQL Lab Practical class – Let's try this !!!

```
Tanakoms-MacBook-Pro:~ tanakomtantontrakul$ mysql -u root -p
Enter password:  2. Enter your MySQL password like "johndoe1999$"
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 12
Server version: 8.0.29 MySQL Community Server - GPL

Copyright (c) 2000, 2022, Oracle and/or its affiliates.

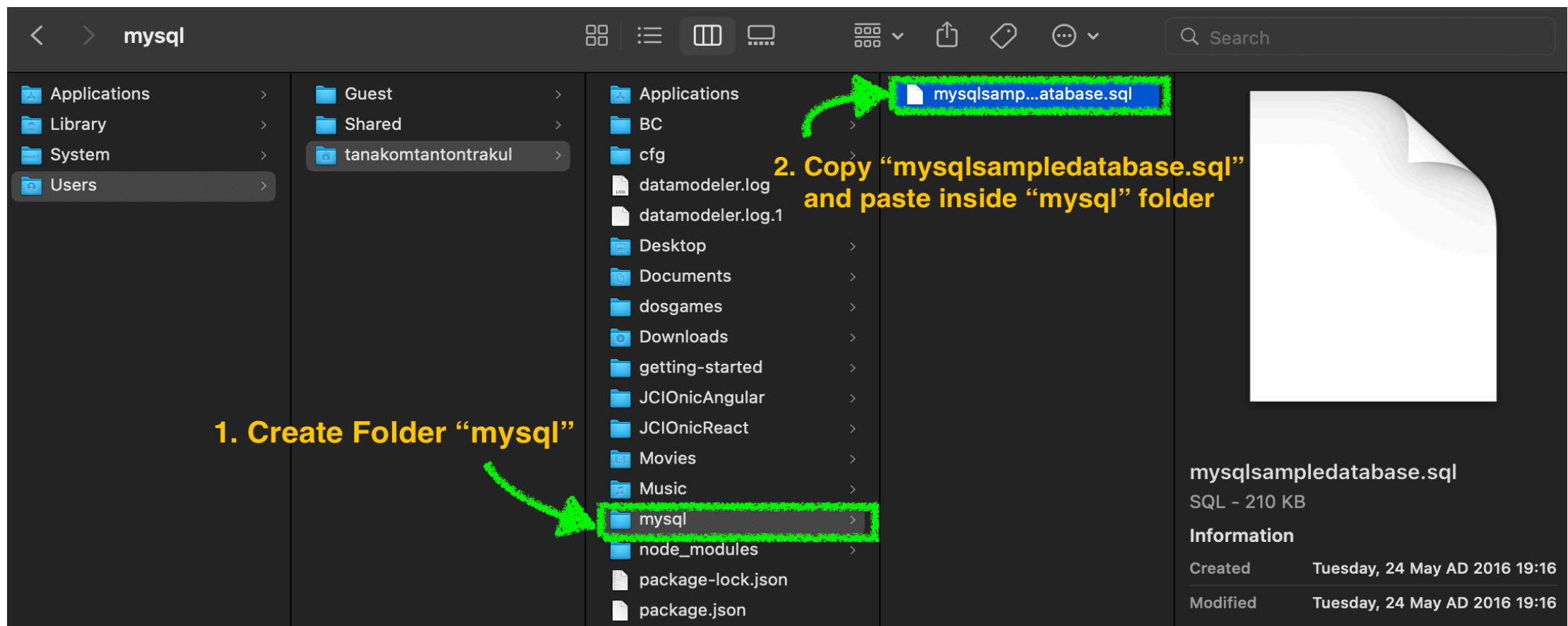
Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>  3. If you see this prompt, congratulations and ready to import database file
```

# SQL Lab Practical class – Let's try this !!!

- Importing database file – macOS



# SQL Lab Practical class – Let's try this !!!

- Importing database file – macOS
  - When you complete input database file into an appropriate folder, please follow these commands to import database file into MySQL

```
mysql> source ./mysql/mysqlsampledatabase.sql
```

← Need to put sql file inside mysql folder of root mac.

```
mysql> show databases;
```

← Check how many tables inside your database.

```
mysql> use classicmodels;
```

← To select database and ready for query.

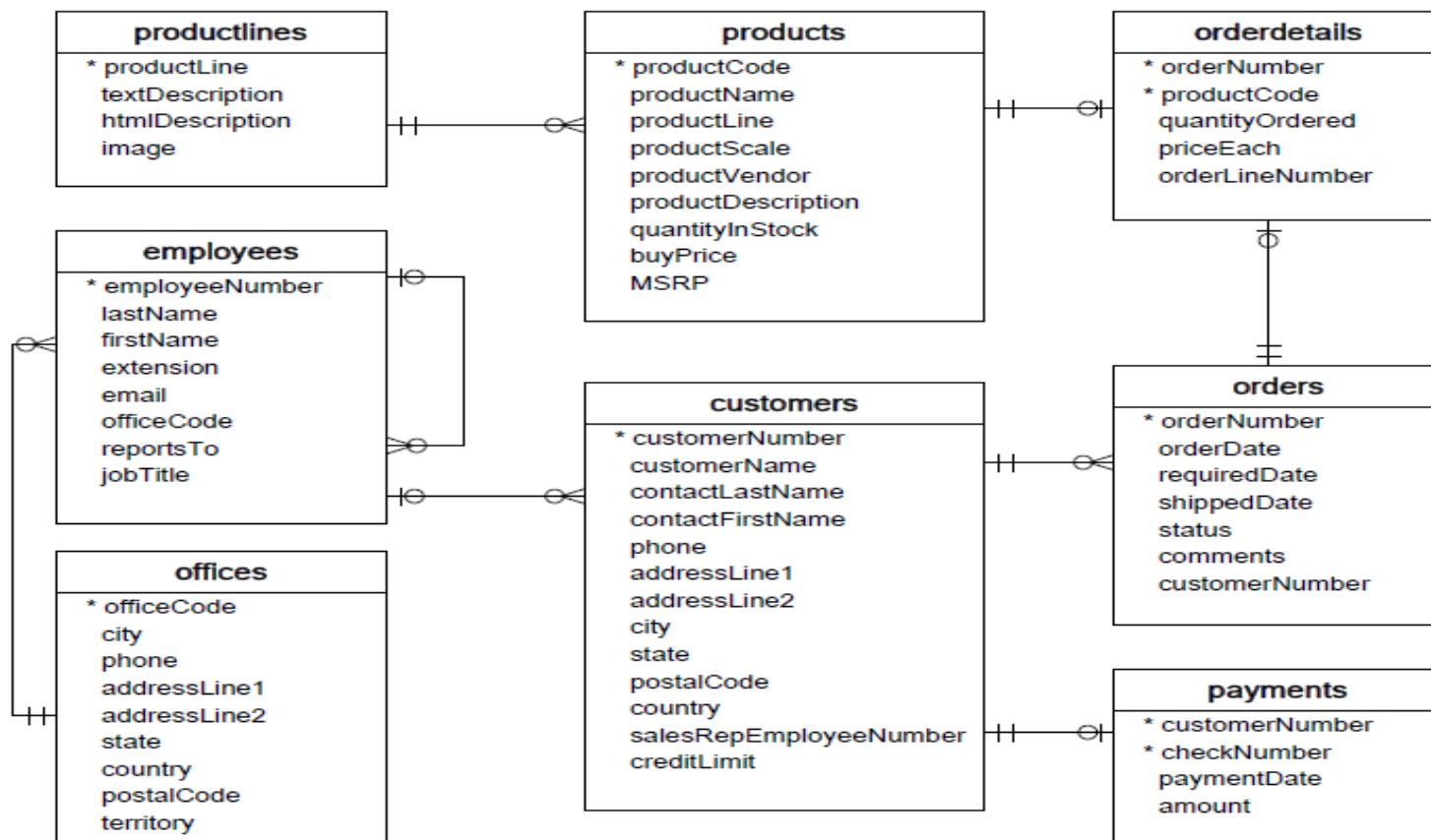
Please note that our database name is “classicmodels”. It is inside “mysqlsampledatabase.sql” file.

```
mysql> show databases;
+-----+
| Database |
+-----+
| classicmodels |
| information_schema |
| mysql |
| performance_schema |
| sys |
+-----+
5 rows in set (0.01 sec)
```

```
mysql> use classicmodels;
Database changed
mysql>
```

# SQL Lab Practical class – Let's try this !!!

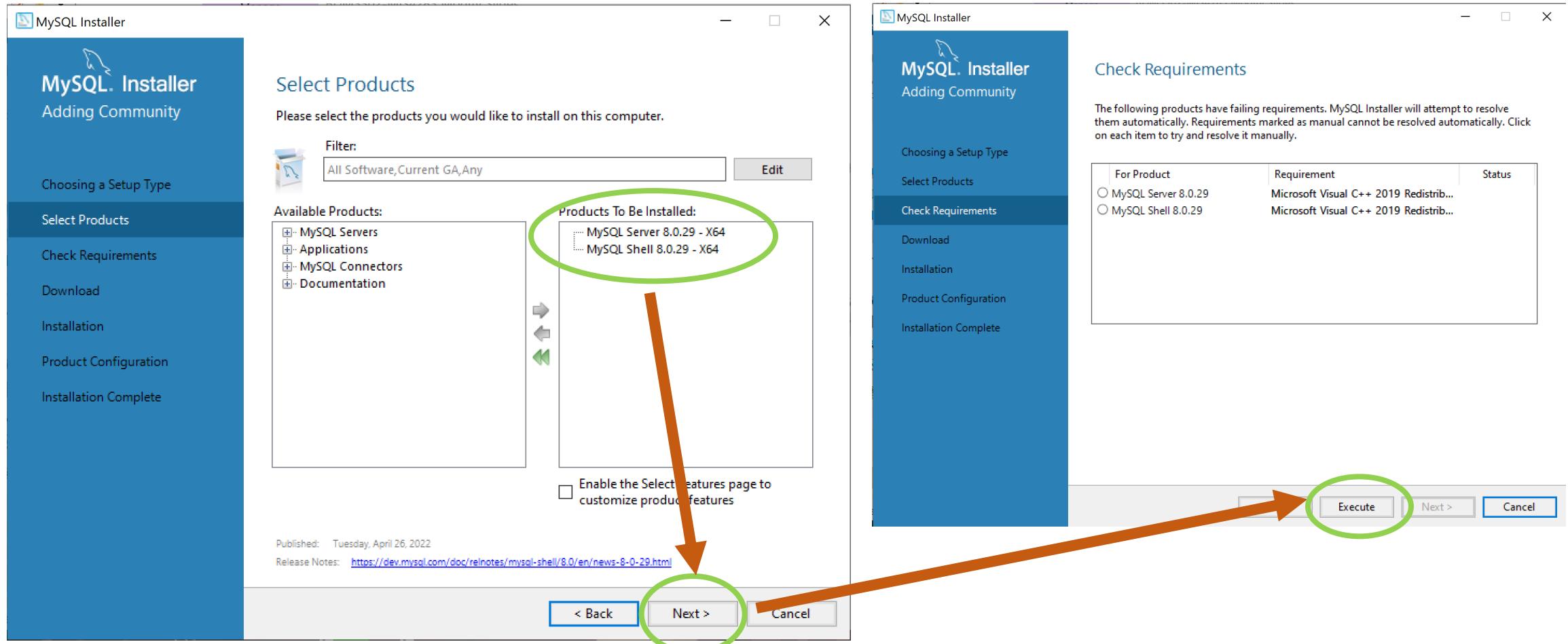
- “classicmodels” Entity Relationship Diagram



# SQL Lab Practical class – Let's try this !!!

- MySQL installation setup – Windows
  - Double click “**mysql-installer-web-community-8.0.29.0.msi**”
  - When the setup screen appear, choose “**Custom**” and then click “**Next**”
  - Choose and add **MySQL Server** → **MySQL Server 8.0** → **MySQL Server 8.0.29 – X64**
  - Choose and add **Applications** → **MySQL Shell** → **MySQL Shell 8.0**  
(See next page)

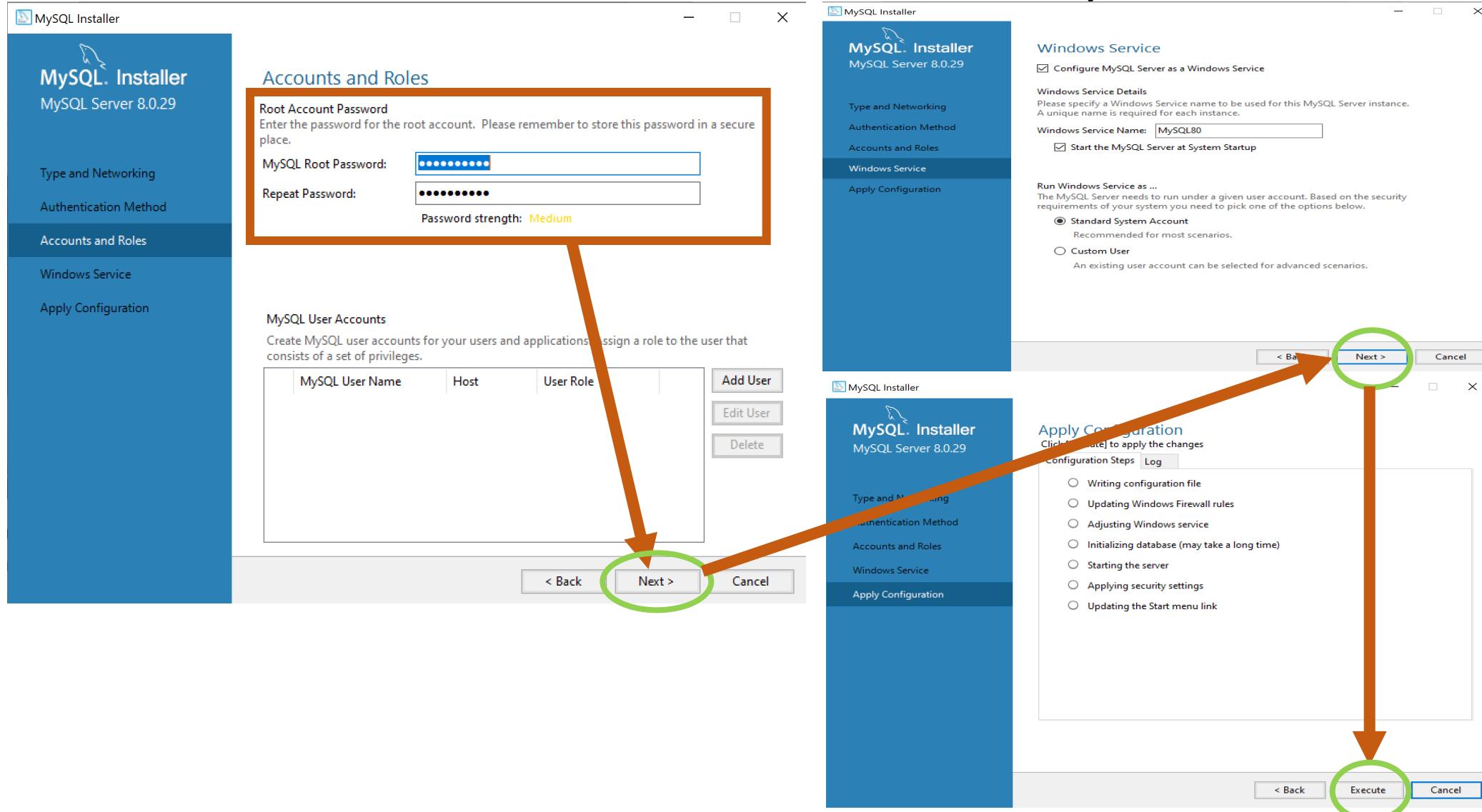
# SQL Lab Practical class – Let's try this !!!



# SQL Lab Practical class – Let's try this !!!

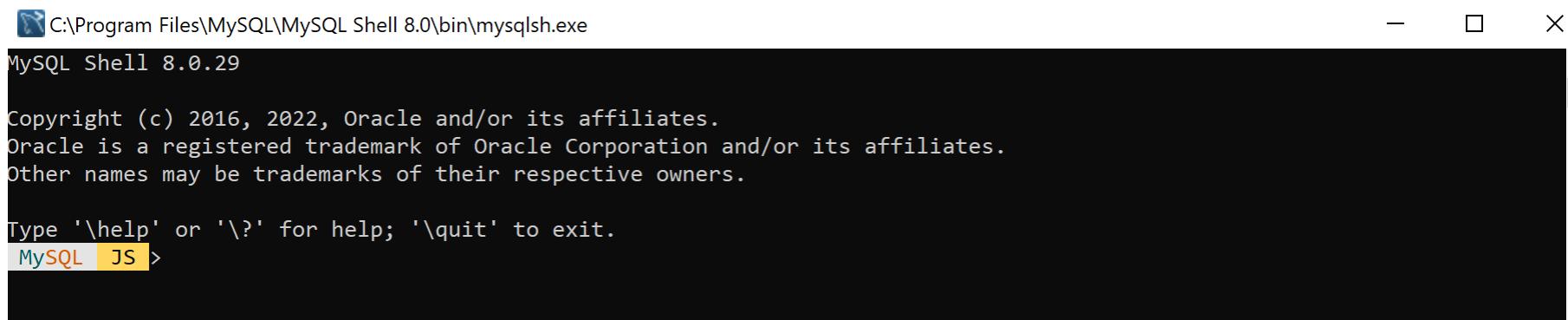
- MySQL installation setup – Windows
  - During download, there are something ask you to download some additional pre-requisite tools. Please allow them to install all also
  - When download is completed, click “**Next**” to continue
  - Click “**Execute**” again to install
  - When install is completed, click “**Next**” to continue
  - Click “**Next**” to continue for **Product Configuration**
  - Choose Config Type: “**Development Computer**” and then click “**Next**”
  - In Authentication Method, choose “**Use Strong Password Encryption for Authentication (RECOMMENDED)**” and then click “**Next**”
  - Input your MySQL appropriate password including alphabets, numbers, and symbols at least 8 characters (e.g., johndoe1999\$) and type repeat password, and then click next (See next page)

# SQL Lab Practical class – Let's try this !!!



# SQL Lab Practical class – Let's try this !!!

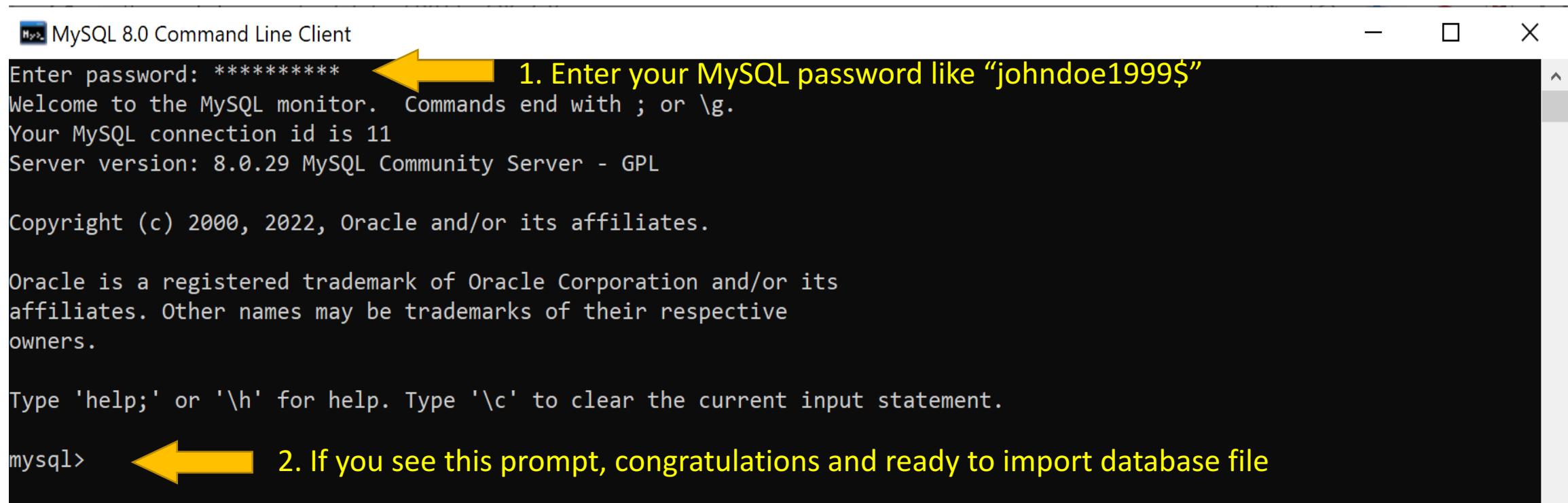
- Windows
  - When you complete MySQL instatallation, you can click “Finish” button at the rest. The **MySQL Shell** will auto open to display like this. **Please close it:**



A screenshot of the MySQL Shell 8.0.29 command-line interface. The window title is "C:\Program Files\MySQL\MySQL Shell 8.0\bin\mysqlsh.exe". The shell version is "MySQL Shell 8.0.29". The output area displays the MySQL copyright notice: "Copyright (c) 2016, 2022, Oracle and/or its affiliates. Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners." Below the notice, it says "Type '\help' or '\?' for help; '\quit' to exit." The prompt shows "MySQL JS >".

- Choose “**MySQL 8.0 Command Line Client**” directly from **Windows Start Menu**  by typing “**MySQL 8.0 Command Line Client**” and click (See next page)

# SQL Lab Practical class – Let's try this !!!



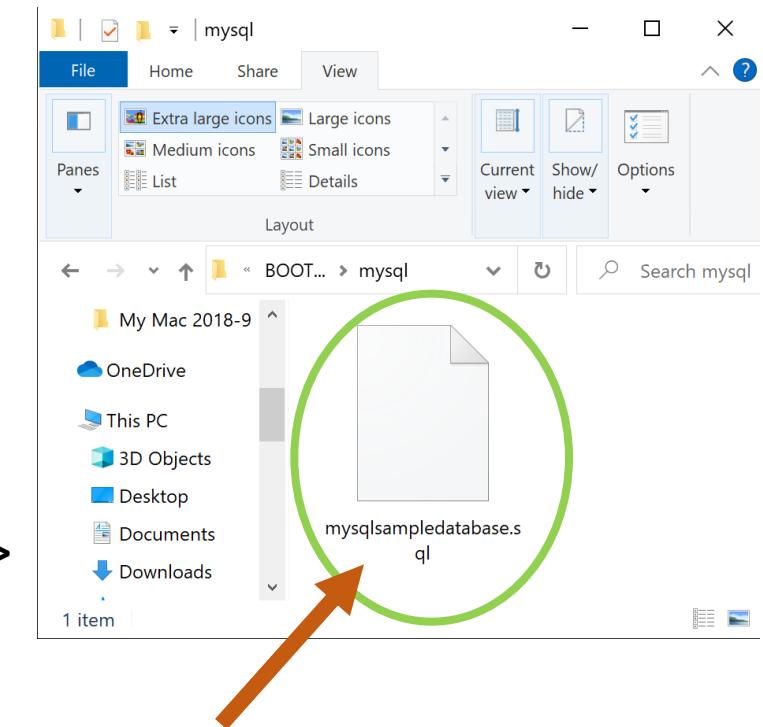
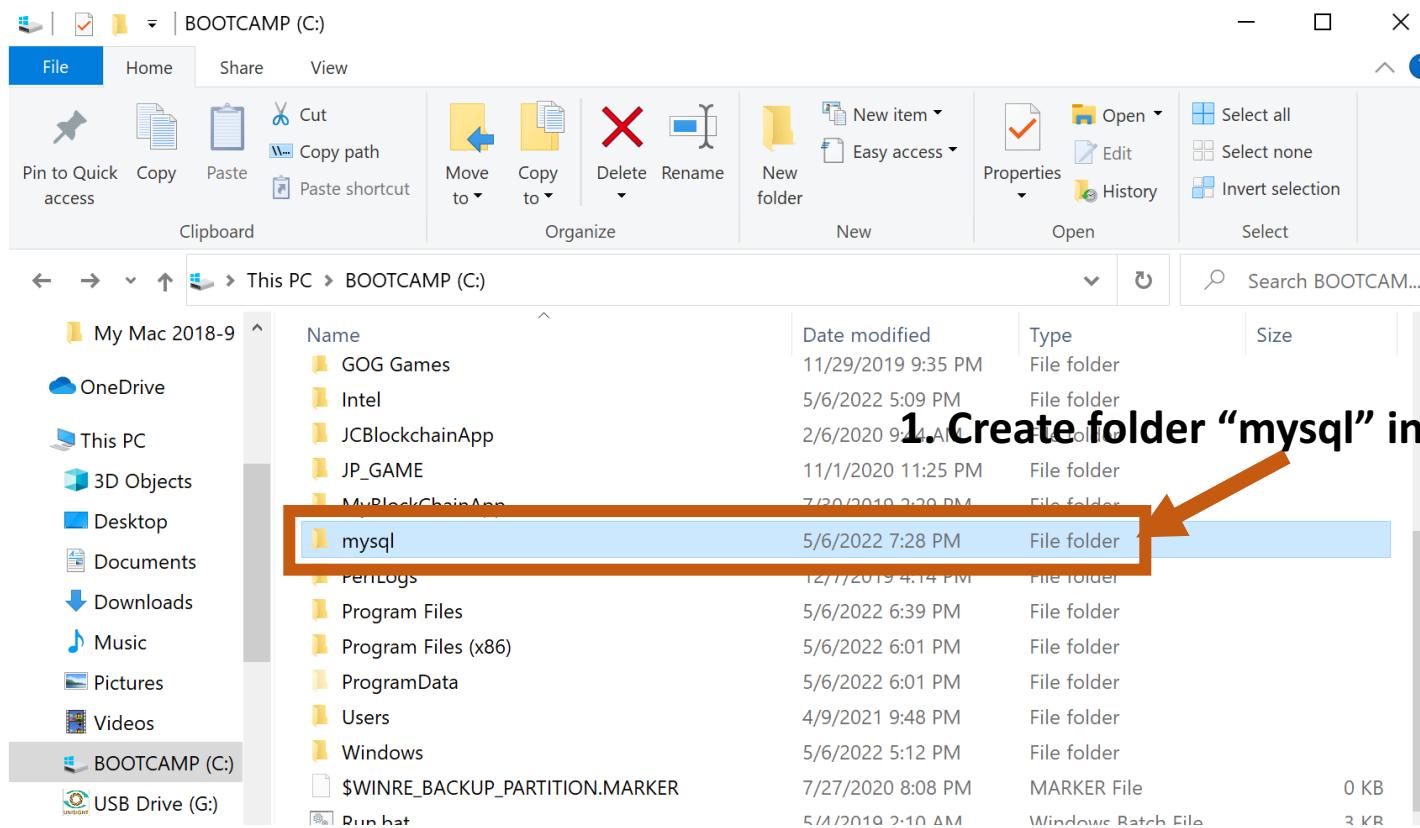
MySQL 8.0 Command Line Client

```
Enter password: *****  
1. Enter your MySQL password like "johndoe1999$"  
Welcome to the MySQL monitor. Commands end with ; or \g.  
Your MySQL connection id is 11  
Server version: 8.0.29 MySQL Community Server - GPL  
  
Copyright (c) 2000, 2022, Oracle and/or its affiliates.  
  
Oracle is a registered trademark of Oracle Corporation and/or its  
affiliates. Other names may be trademarks of their respective  
owners.  
  
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.  
mysql> 2. If you see this prompt, congratulations and ready to import database file
```

A screenshot of the MySQL 8.0 Command Line Client window. The title bar says "MySQL 8.0 Command Line Client". The main area shows the following text:  
Enter password: \*\*\*\*\*  
Welcome to the MySQL monitor. Commands end with ; or \g.  
Your MySQL connection id is 11  
Server version: 8.0.29 MySQL Community Server - GPL  
  
Copyright (c) 2000, 2022, Oracle and/or its affiliates.  
  
Oracle is a registered trademark of Oracle Corporation and/or its  
affiliates. Other names may be trademarks of their respective  
owners.  
  
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.  
mysql>  
A yellow arrow points to the password entry field with the text "1. Enter your MySQL password like \"johndoe1999\$\"". Another yellow arrow points to the mysql> prompt with the text "2. If you see this prompt, congratulations and ready to import database file".

# SQL Lab Practical class – Let's try this !!!

- Importing database file – Windows



# SQL Lab Practical class – Let's try this !!!

- Importing database file – Windows
  - When you complete input database file into an appropriate folder, please follow these commands to import database file into MySQL

```
mysql> source c:\mysql\mysqlsampledatabase.sql
```

← Need to put sql file inside mysql folder of root as C:\>.

```
mysql> show databases;
```

← Check how many tables inside your database.

```
mysql> use classicmodels;
```

← To select database and ready for query.

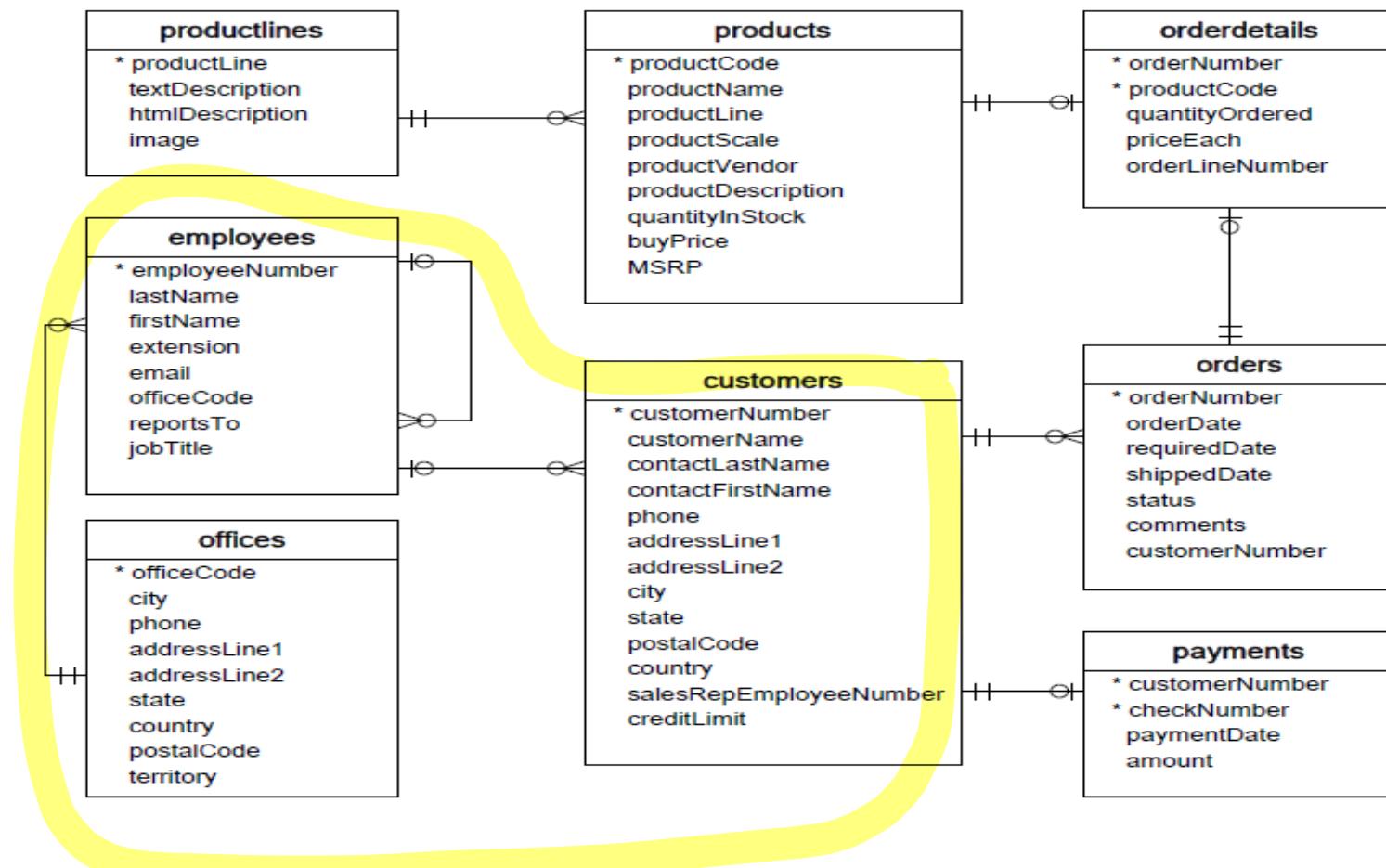
Please note that our database name is “classicmodels”. It is inside “mysqlsampledatabase.sql” file.

```
+-----+  
| Database          |  
+-----+  
| classicmodels    |  
| information_schema|  
| mysql             |  
| performance_schema|  
| sys               |  
+-----+  
5 rows in set (0.00 sec)  
  
mysql>
```

```
mysql> use classicmodels;  
Database changed  
mysql>
```

# SQL Lab Practical class – Let's try this !!!

- “classicmodels” Entity Relationship Diagram



# SQL Lab Assignment (Week 3-1)

- Form your group of 3 persons
- Provide the video clip of demonstration how to setup MySQL and import sql file from your computer
  - **Hint:** You can meet and discuss to do in your own **private MS Teams chat** and save video into **mp4** file format
- You can choose which OS you would like to do (Windows, macOS)
- Submit by post the video clip in MS Teams channel provided (Deadline also be noticed in LMS)
- Need one member per group to submit

# SQL Lab Practical class – Let's try this !!!

- PopSQL Installations and Setup

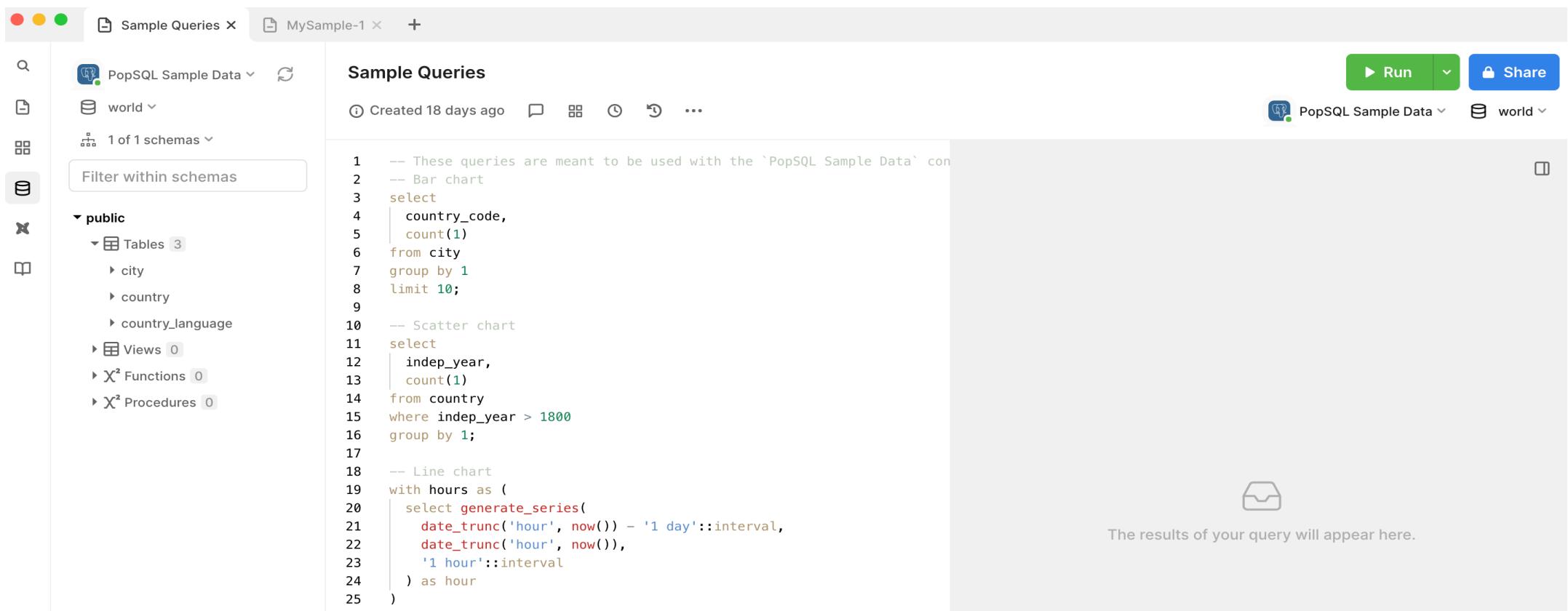
- Go to <https://popsql.com>
- Click “**Get started for Free**”
- Sign up your account and remember account to access any time
- Choose “**Download PopSQL for desktop**”



- Double click PopSQL to install and follow the instructions

# SQL Lab Practical class – Let's try this !!!

- When complete install PopSQL and open, the screen will display like this:



The screenshot shows the PopSQL application interface. The top navigation bar includes tabs for "Sample Queries" and "MySample-1". On the left, a sidebar displays the "PopSQL Sample Data" connection, showing the "world" schema with 1 schema and 3 tables: city, country, and country\_language. Below this, there are sections for Views (0), X<sup>2</sup> Functions (0), and X<sup>2</sup> Procedures (0). The main area is titled "Sample Queries" and shows a list of numbered SQL statements. The first few statements are:

```
1 -- These queries are meant to be used with the `PopSQL Sample Data` con
2 -- Bar chart
3 select
4   | country_code,
5   | count(1)
6   from city
7   group by 1
8   limit 10;
9
10 -- Scatter chart
11 select
12   | indep_year,
13   | count(1)
14   from country
15   where indep_year > 1800
16   group by 1;
17
18 -- Line chart
19 with hours as (
20   select generate_series(
21     date_trunc('hour', now()) - '1 day'::interval,
22     date_trunc('hour', now()),
23     '1 hour'::interval
24   ) as hour
25 )
```

To the right of the code editor, there is a large empty area with a placeholder message: "The results of your query will appear here." A small envelope icon is located in the bottom right corner of this area.

# SQL Lab Practical class – Let's try this !!!

- Setup imported MySQL database into PopSQL
    - Click “**PopSQL Sample Data**” and then choose “**Manage connections**”
    - Click “**Add new connection**” and then choose “**MySQL**”
    - Input information like this picture below, and click “**Test**” to make sure that your imported database is connected and then click “**Connect**”
- (See next page)

# SQL Lab Practical class – Let's try this !!!

<  Edit My Classic Model

Connection name: My Classic Model

Hostname/Port: localhost 3306

Database: classicmodels

Username/Password: root .....  
Credentials will be securely stored in your computer's keychain.

Connection Type:  Connect directly from my computer  
Direct: Use your computer's network to connect to your database.

Who has access?: Add users or teams  
Tanakom Tanto... Owner  
Grant granular permissions to teams and specific users.  
[Talk to sales](#) to add this to your team.

Advanced Options:  Connect over SSH  
Useful to connect to a private network.

SSL options  
SSL is used when available

**Test** **Save** **Connect**

**Looking for help?**

- [MySQL connection guide](#)
- [Connection Types](#)
- [Sharing Connections](#)

**Whitelist IPs**

If you maintain an IP whitelist, you'll need to whitelist PopSQL static IPs:

23.20.131.72,  
54.211.234.135

# SQL Lab Practical class – Let's try this !!!

- PopSQL Screen Overview

The image shows the PopSQL interface with three main sections:

- Left Panel:** Displays the database schema. A red box highlights the "classicmodels" schema under "Tables". Below it, a red box highlights the "MySample-1" tab in the top navigation bar.
- Middle Panel:** Shows a code editor with a series of SQL commands. A red box highlights the first few lines of code:

```
1 --Show all customer data
2 select *
3   from customers;
4
5 --Show some customer data
6 select customerNumber, customerName, phone, city, country
7   from customers
8   limit 50;
9
10 --Display total customers per country data
11 select country, count(1)
12   from customers
13   group by 1;
```

A red box also highlights the "Run" button in the top right corner of the code editor.
- Bottom Panel:** Shows the results of the executed query. A red box highlights the "Data result area" title. The results table contains 122 rows of customer data, with columns: customerNumber, customerName, and contactLastName. A red box highlights the "Run" button in the top right corner of the results panel.

**Annotations:**

- Display database table**: Points to the "classicmodels" schema in the left sidebar.
- Type SQL commands in this area, select one part, and click Run button**: Points to the code editor area.
- Click Run to show the data result**: Points to the "Run" button in the code editor and the "Run" button in the results panel.
- Data result area**: Points to the results table in the bottom panel.

customerNumber	customerName	contactLastName
103	Atelier graphique	Schmitt
112	Signal Gift Stores	King
114	Australian Collectors, Co.	Ferguson
119	La Rochelle Gifts	Labrune
121	Baane Mini Imports	Bergulfsen
124	Mini Gifts Distributors Ltd.	Nelson
125	Havel & Zbyszek Co	Piestrzaniewi
128	Blauer See Auto, Co.	Keitel
129	Mini Wheels Co.	Murphy
131	Land of Toys Inc.	Lee

# SQL Lab Practical class – Let's try this !!!

- Exporting file into CSV, JSON

The screenshot illustrates the export process for a dataset named 'MySample-1'. The main window shows a table with columns: customerNumber, customerName, contactLastN, contactFirstN, phone, addressLine1, addressLine2, city, state, postalCode, country, salesRepEmpl, and creditLimit. A success message indicates 'Success (122 rows)' in 0.2 seconds at 8:50 PM. The 'Export' button in the top right is highlighted with an orange box. A context menu is open, listing options: Copy TSV, Copy CSV, Copy JSON, Export CSV (which is selected and highlighted in blue), Export JSON, and Other apps. A second context menu is shown in a separate window titled 'Export MySample-1', where 'Save As:' is set to 'MySampleData.csv', 'Format:' is 'CSV', and the 'Export' button is also highlighted with an orange box.

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	customerNu	customerNa	contactLastN	contactFirstN	phone	addressLine1	addressLine2	city	state	postalCode	country	salesRepEmpl	creditLimit
2	103	Atelier graphique	Schmitt	Carine	40.32.2555	54, rue Royale		Nantes		44000	France	1370	21000
3	112	Signal Gift Stores	King	Jean	7025551838	8489 Strong St.		Las Vegas	NV	83030	USA	1166	71800
4	114	Australian Collectors, Co.	Ferguson	Peter	03 9520 4551	636 St Kilda	Level 3	Melbourne	Victoria	3004	Australia	1611	117300
5	119	La Rochelle Gifts	Labrune	Janine	40.67.8555	67, rue des Cinquante Ota	Nantes			44000	France	1370	118200
6	121	Baane Mini Imports	Bergulfesen	Jonas	07-98 9555	Erling Skakkes gate 78	Stavern			4110	Norway	1504	81700
7	124	Mini Gifts Distributors Ltd.	Di Nelson	Susan	4155551450	5677 Strong St.	San Rafael	CA		97562	USA	1165	210500
8	125	Havel & Zbyszek Co	Piestrzeniew	Zbyszek	(26) 642-755 ul. Filtrowa	68	Warszawa		01-012		Poland		0
9	128	Blauer See Auto, Co.	A Keitel	Roland	+49 69 66 90 Lyonerstr.	34	Frankfurt			60528	Germany	1504	59700
10	129	Mini Wheels Co.	Lee	Julie	6505555787	5557 North Pendale Street	San Francisco	CA		94217	USA	1165	64600
11	131	Land of Toys Inc.	Kwai		2125557818	897 Long Airport Avenue	NYC	NY		10022	USA	1323	114900
12	141	Euro+ Shop Freyre	Diego		(91) 555 94 4 C/ Moralzarjal,	86	Madrid			28034	Spain	1370	227600
13	144	Volvo Model Berglund	Christina		0921-12 355	Berguvsvv�gen 8	Lule�		S-958 22		Sweden	1504	53100
14	145	Danish Whol Petersen	Jytte		31 12 3555	Vinb�ltet 34	Kopenhagen			1734	Denmark	1401	83400
15	146	Saveley & He Saveley	Mary		78.32.5555	2, rue du Commerce	Lyon			69004	France	1337	123900
16	148	Dragon Souvenirs Natividad	Eric		+65 221 7551	Bronz Sok.	Bronz Apt. 3, Singapore			79903	Singapore	1621	103800

# Basic SQL Commands

- Try to input SQL commands in PopSQL
  - Display all fields in products table

```
select *  
from products;
```

- Display some fields in products table with limit by 10

```
select productCode, productName, quantityInStock, buyPrice  
from products  
limit 10;
```

- Try some more using “select” command in other tables

# Basic SQL Commands

- Try to input SQL commands in PopSQL
  - Sort all product name in an ascending order

```
select *  
from products  
order by productName asc;
```

- Sort all product name in a descending order

```
select *  
from products  
order by productName desc;
```

- Try some more using an enhance command in other tables (Check **classicmodels** entity relationship diagram carefully)

# Basic SQL Commands

- Try to input SQL commands in PopSQL
  - Display all fields where productCode is ‘S12\_1666’ in products table

```
select *  
from products  
where productCode = 'S12_1666';
```

- Display some fields where quantityInStock >= 300 and buyPrice > 50 in products table

```
select productCode, productName, quantityInStock, buyPrice  
from products  
where quantityInStock >= 300 and buyPrice > 50;
```

- Try some more using “select” command in other tables

# Basic SQL Commands

- Try to input SQL commands in PopSQL
  - Display some fields with adding “total” field where quantityInStock > 300 and buyPrice > 50 in products table

```
select productCode, productName, quantityInStock, buyPrice, quantityInStock * buyPrice as total  
from products  
where quantityInStock > 300 and buyPrice > 50;
```
  - Try some more using “select” command in other tables

# Basic SQL Commands

- Some more SQL commands with PopSQL

- Insert new item in products table

```
insert into products (productCode, productName, productLine, productScale, productVendor,  
productDescription, quantityInStock, buyPrice, MSRP)  
value ('S11_1050', '2022 Diamond Supra', 'Classic Cars', '1:15', 'Autocart Studio Design', 'Upgrade classic  
mode for enhancing', '1200', '75.25', '99.99');
```

- Update current item in products table

```
update products  
set buyPrice = '85.78', MSRP = '120.12'  
where productCode = 'S11_1050';
```

- Delete current item in products table

```
delete from products  
where productCode = 'S11_1050';
```

- Try some more using “insert”, “update”, “delete” command in other tables

# Basic SQL Commands

- Some more SQL commands with PopSQL
  - Enhancing two tables between “products” and “orderDetails”

```
select p.productCode as pid, o.orderNumber as oid, p.quantityInStock as qtyStock, o.quantityOrdered as qtyOrdered  
from products as p, orderDetails as o  
where p.productCode = o.productCode;
```

```
select p.productCode as pid, o.orderNumber as oid, p.quantityInStock as qtyStock, o.quantityOrdered as qtyOrdered  
from products as p, orderDetails as o  
where p.productCode = o.productCode and p.quantityInStock >= 300;
```

- Display all tables in database

```
show full tables;
```
- Try some more using an enhance command in other tables (Check **classicmodels** entity relationship diagram carefully)

# SQL Lab Assignment (Week 3-2)

- Form your group of 3 persons
- Download **world.sql** from LMS and import database using MySQL
- Setup **world** database that connect into PopSQL
- Create powerpoint to present the SQL and result follow the questions:
  - How many tables have you seen in world database? List all table names and fields.
  - List first 20 countries's name, continent, and population
  - List all cities's name, country code, and distinct only in Australia (**Hint:** check country code)
  - List the first 50 countries's name, surface area, population, and life expectancy with greater than 500,000 people of population and less than 60 years of life expectancy
  - List all city's name, country's continent, country's name, and country's population in the North America. Display city's name in an ascending order. How many city name in the list? (**Hint:** Check how to join country code)
- Submit into LMS through the link provided (Deadline also be noticed in LMS)
- Need one member per group to submit