INT205 – Database Management System
1/2564

LAB02: SQL Review

Submission:

- Submit a lab file named “int205_lab02_xxxxxxxxxxx.docx/.pdf” into the LEB2 system. xxxxxxxxxxx = your student id
- **<u>Do NOT submit</u>** both a PNG file and a MWB file.

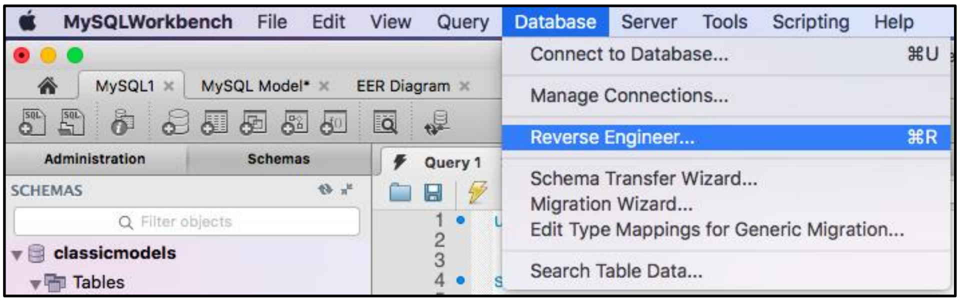
Due Date & Time:

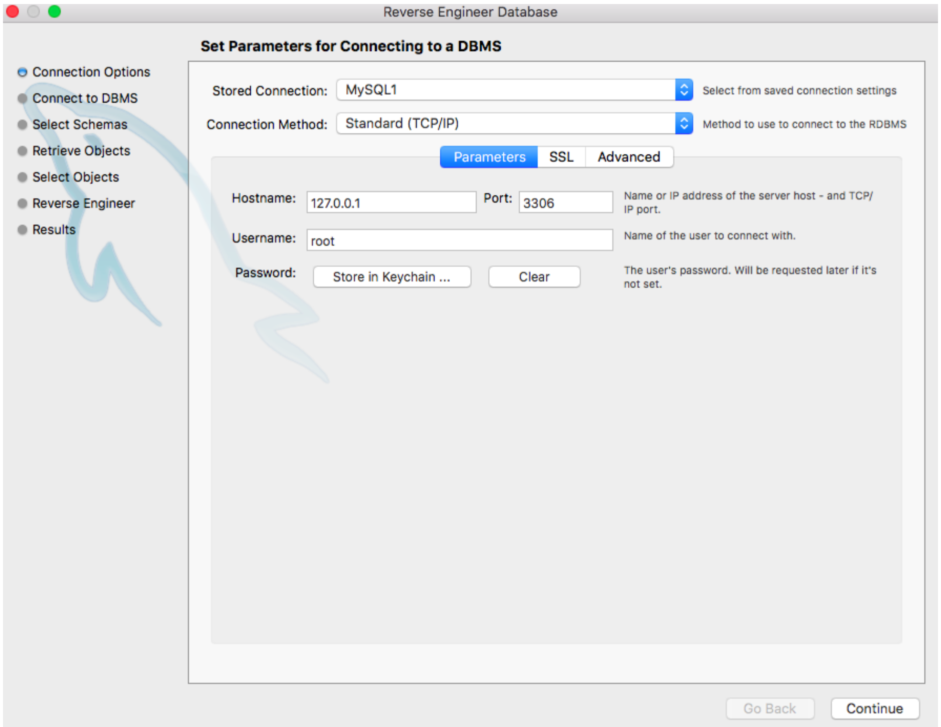
- Lecturer will inform the LAB02 due date and time in lab class.

Task 1: Using MySQL Workbench to create an ER-diagram for the “classicmodels”.

1.1 Use the script named “classicmodels” to generate ER diagram (Reverse Engineering technique).

- Open the MySQL Workbench program
- Select the menu “Database” => “Reverse Engineer…”.
- Please follow the step by step instructions below:





Set Parameters for Connecting to a DBMS

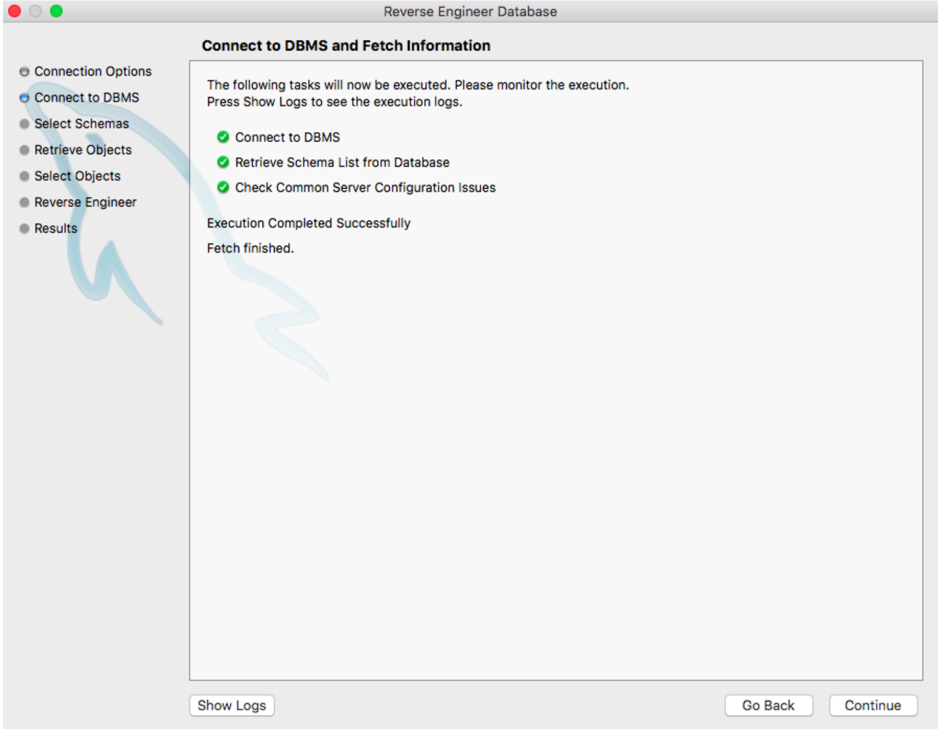
Stored Connection: MySQL1 — Select from saved connection settings

Connection Method: Standard (TCP/IP) — Method to use to connect to the RDBMS

Hostname: 127.0.0.1 Port: 3306 — Name or IP address of the server host - and TCP/IP port.

Username: root — Name of the user to connect with.

Password: — The user's password. Will be requested later if it's not set.



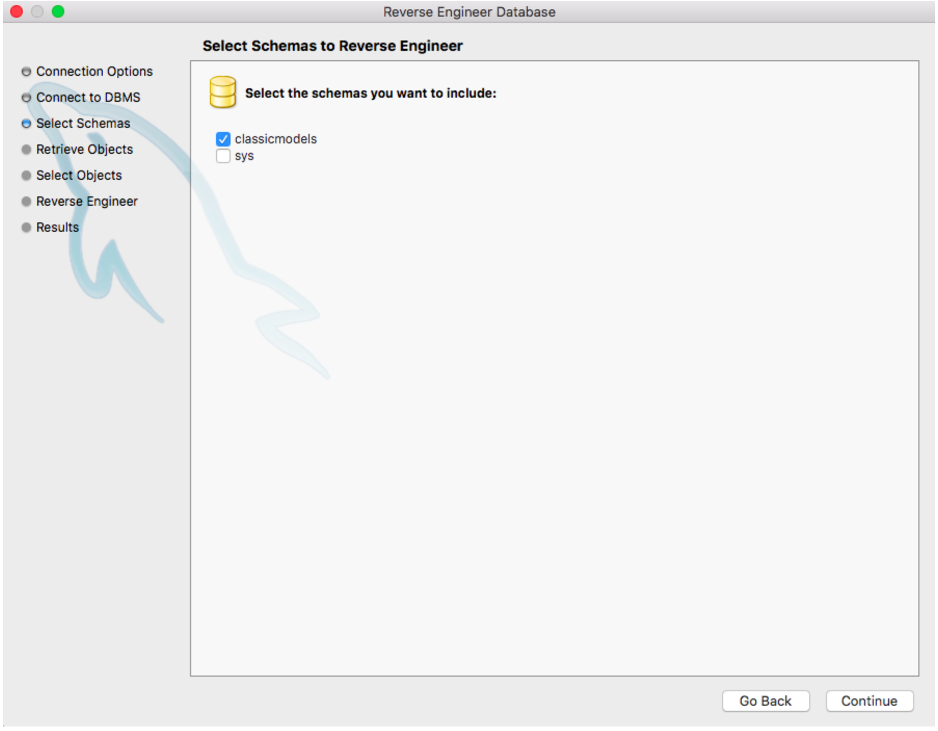
Connect to DBMS and Fetch Information

The following tasks will now be executed. Please monitor the execution.
Press Show Logs to see the execution logs.

- ✓ Connect to DBMS
- ✓ Retrieve Schema List from Database
- ✓ Check Common Server Configuration Issues

Execution Completed Successfully

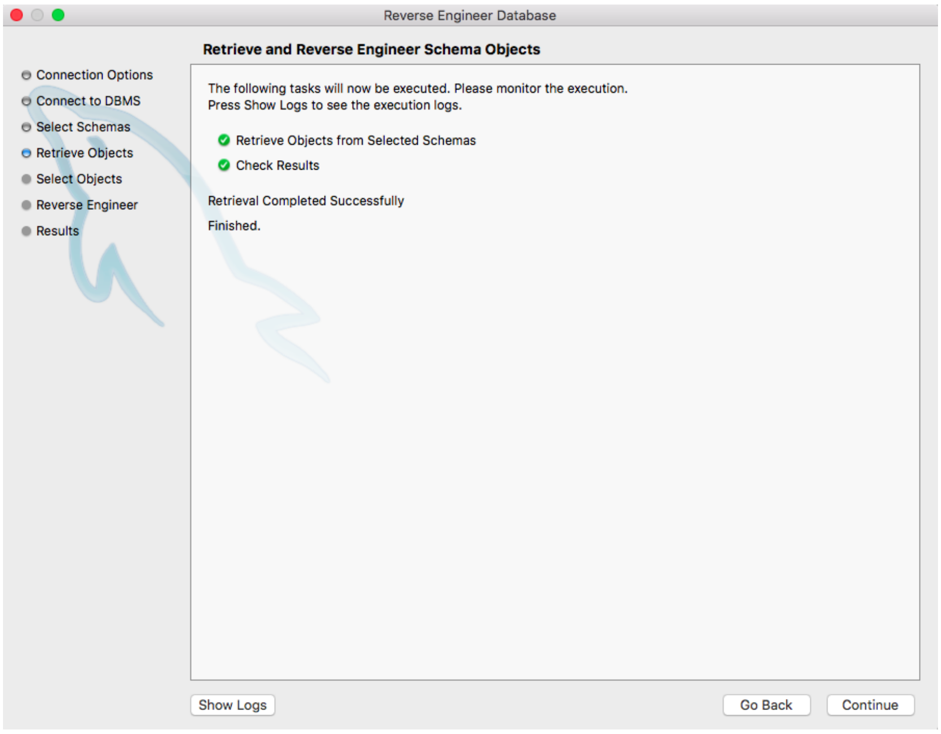
Fetch finished.



Select Schemas to Reverse Engineer

Select the schemas you want to include:

☑ classicmodels
☐ sys



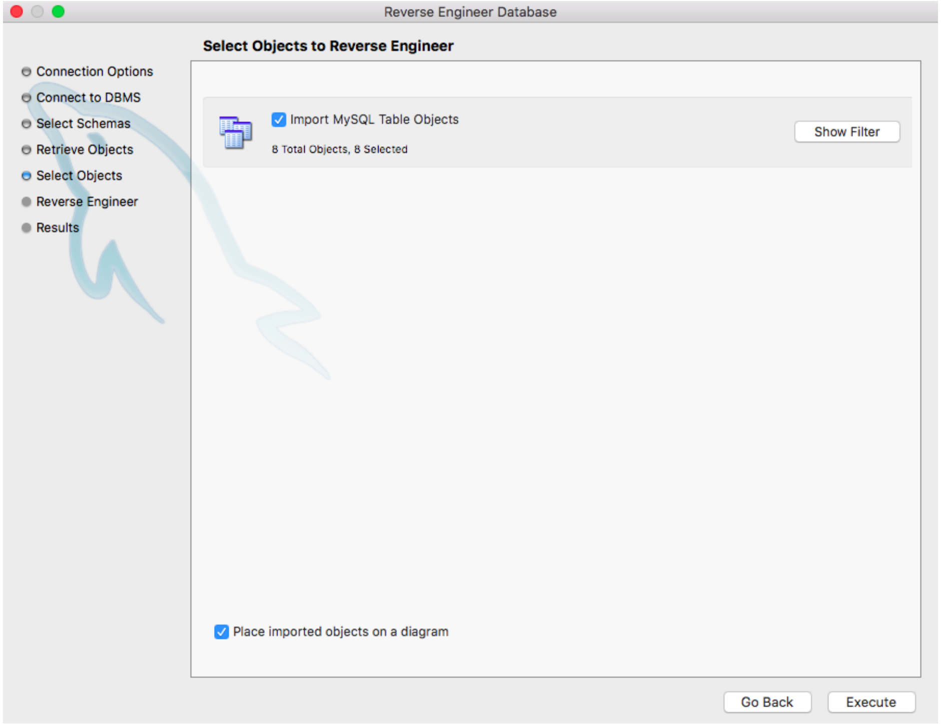
Retrieve and Reverse Engineer Schema Objects

The following tasks will now be executed. Please monitor the execution.
Press Show Logs to see the execution logs.

✓ Retrieve Objects from Selected Schemas
✓ Check Results

Retrieval Completed Successfully

Finished.

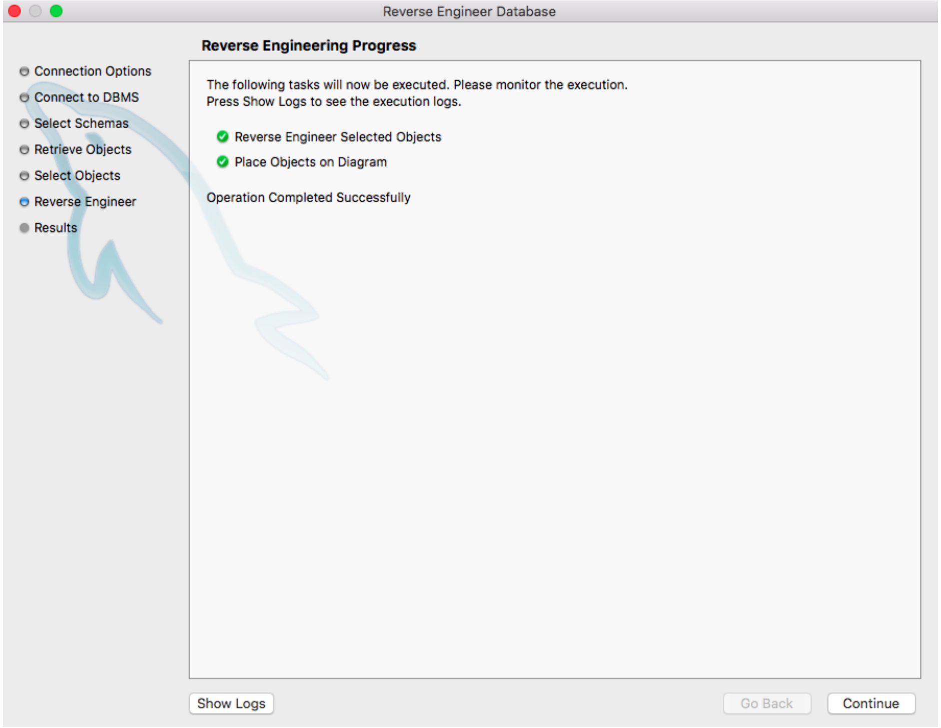


Select Objects to Reverse Engineer

☑ Import MySQL Table Objects

8 Total Objects, 8 Selected

☑ Place imported objects on a diagram



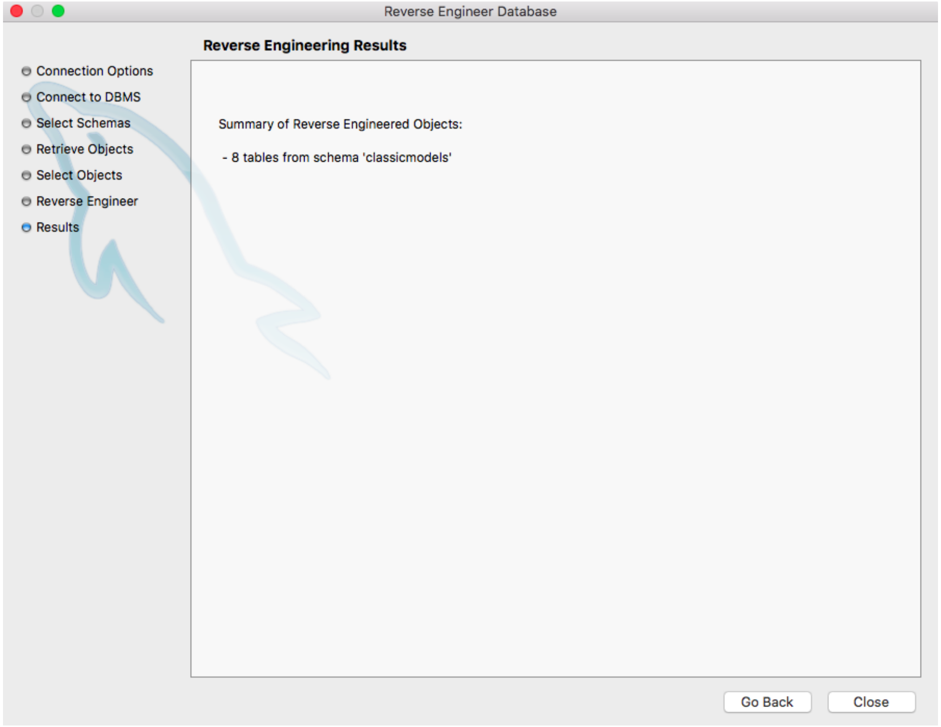
Reverse Engineering Progress

The following tasks will now be executed. Please monitor the execution.
Press Show Logs to see the execution logs.

✓ Reverse Engineer Selected Objects

✓ Place Objects on Diagram

Operation Completed Successfully



Reverse Engineer Database

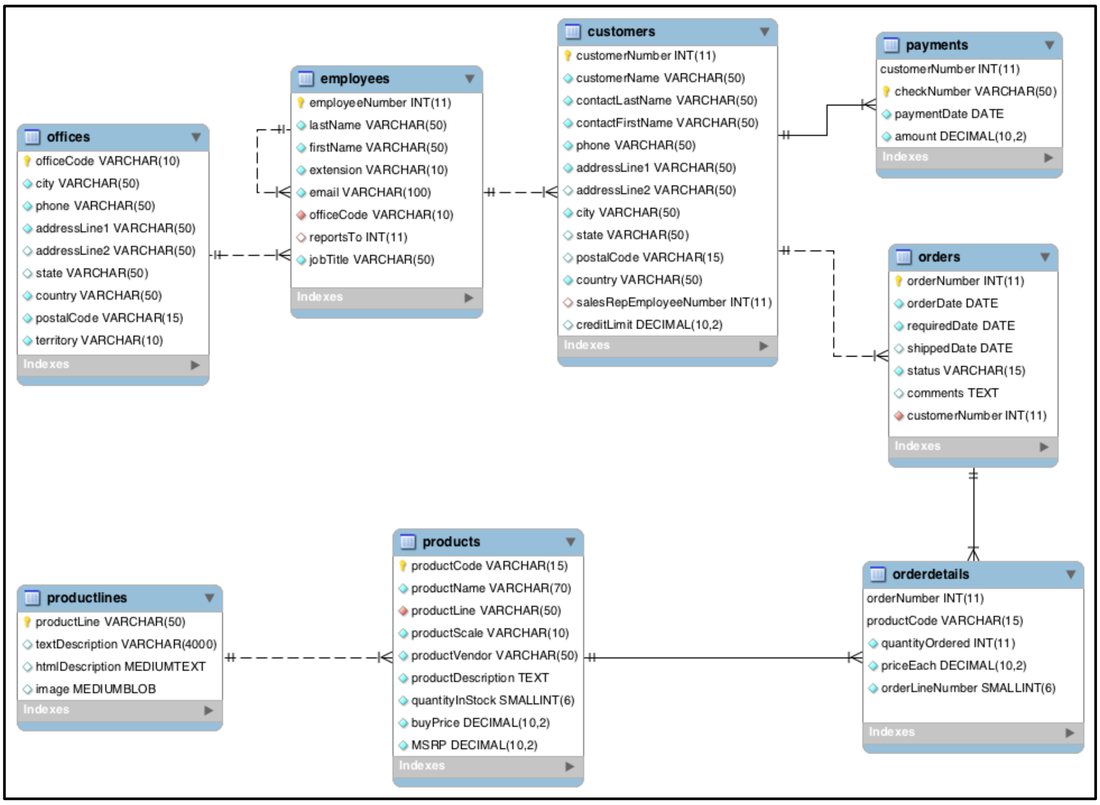
Reverse Engineering Results

- Connection Options
- Connect to DBMS
- Select Schemas
- Retrieve Objects
- Select Objects
- Reverse Engineer
- Results

Summary of Reverse Engineered Objects:

- 8 tables from schema 'classicmodels'

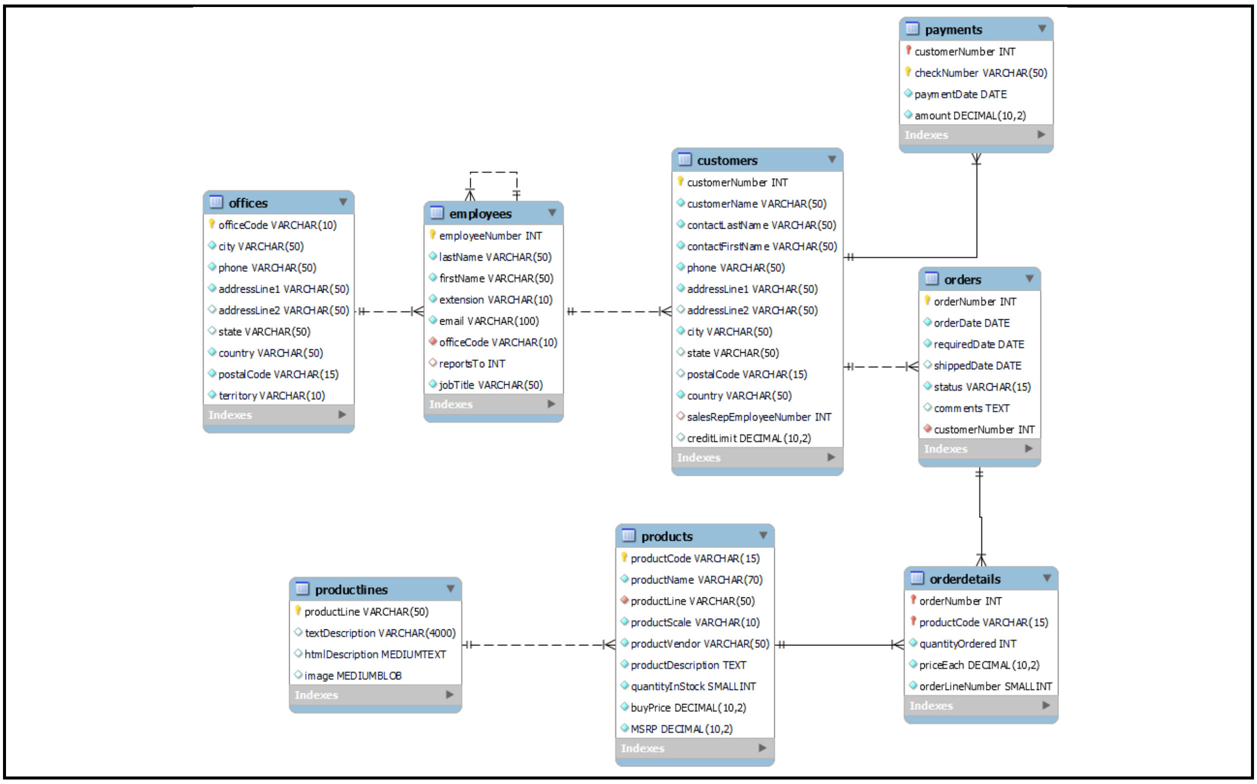
The example ER diagram for the classicmodels.



1.2 Re-arrange the ER diagram for readability and review the tables, columns, data types and relationships among the tables.

- Save a model named “classicmodels.mwb” (the default extension file is .mwb) and export a model as a .png file named “classicmodels_er.png” stored in your computer.

- Place your ER image (the png file) here.



Note: The MSRP is “Manufacturer's suggested retail price” (ราคาขายปลีกแนะนำของผู้ผลิต).

Task 2: Using the “classicmodels” schema and write SQL statements to answer the following questions.

The Syntax of SELECT statement:
Documentation: https://dev.mysql.com/doc/refman/8.0/en/select.html

Note: The MySQL error code 1064 is a syntax error. This means the reason there’s a problem is because MySQL doesn’t understand what you’re asking it to do.

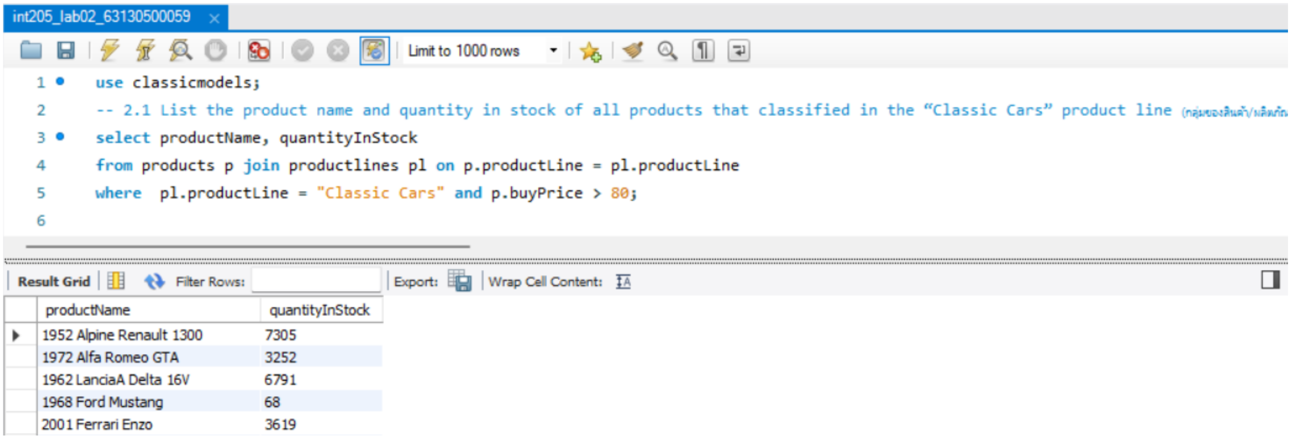
Switch to SQL Editor

- You should specify the classicmodels database before writing SQL statements using the following command:

```
USE db_name;
```

The USE statement tells MySQL to use the named database as the default (current) database for subsequent statements. This statement requires some privilege for the database or some object within it.

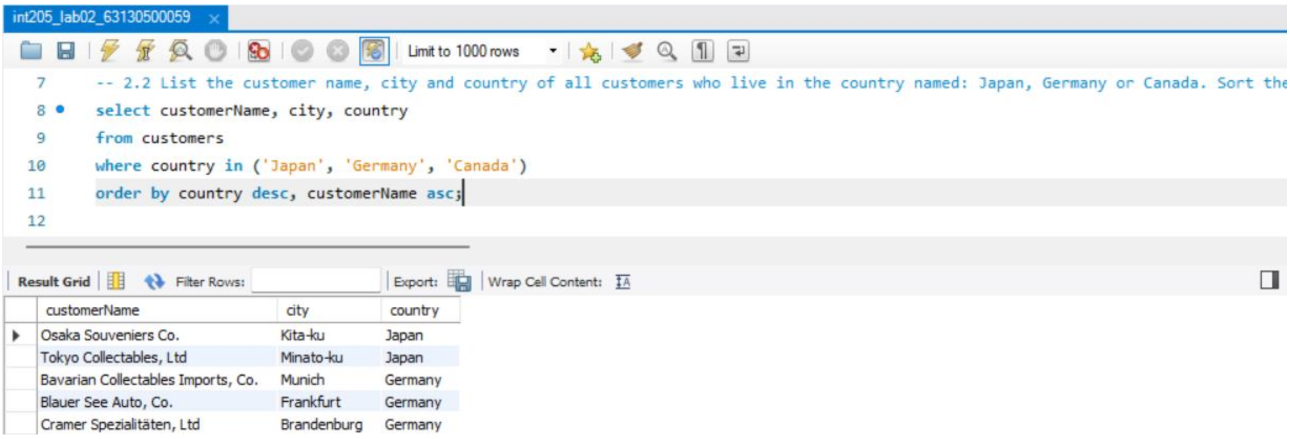
2.1 List the product name and quantity in stock of all products that classified in the “Classic Cars” product line (กลุ่มของสินค้า/ผลิตภัณฑ์) and their buy prices are more than 80.



```
use classicmodels;
-- 2.1 List the product name and quantity in stock of all products that classified in the “Classic Cars” product line (กลุ่มของสินค้า/ผลิตภั
select productName, quantityInStock
from products p join productlines pl on p.productLine = pl.productLine
where  pl.productLine = "Classic Cars" and p.buyPrice > 80;
```

productName	quantityInStock
1952 Alpine Renault 1300	7305
1972 Alfa Romeo GTA	3252
1962 LanciaA Delta 16V	6791
1968 Ford Mustang	68
2001 Ferrari Enzo	3619

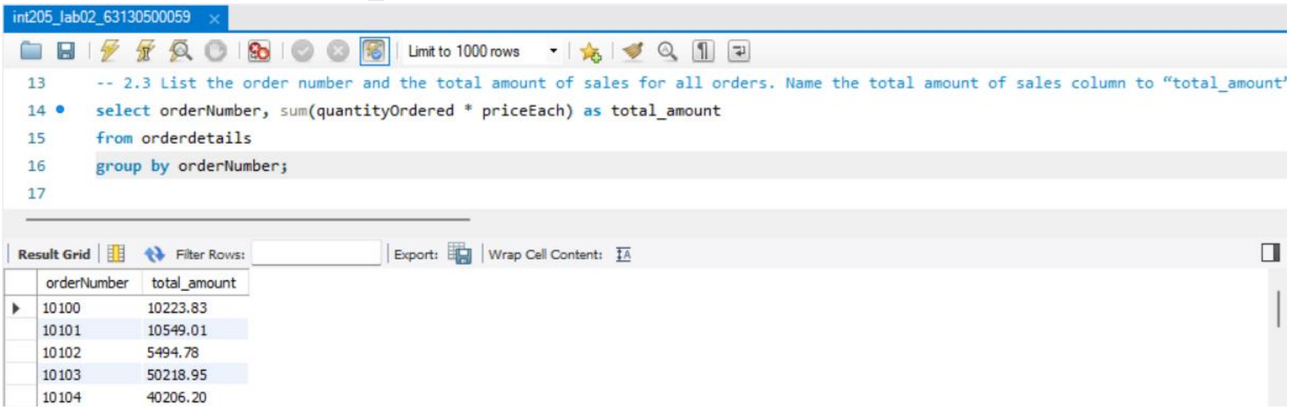
2.2 List the customer name, city and country of all customers who live in the country named: Japan, Germany or Canada. Sort the results in descending order by country and ascending order by customer name.



```
-- 2.2 List the customer name, city and country of all customers who live in the country named: Japan, Germany or Canada. Sort the
select customerName, city, country
from customers
where country in ('Japan', 'Germany', 'Canada')
order by country desc, customerName asc;
```

customerName	city	country
Osaka Souveniers Co.	Kita-ku	Japan
Tokyo Collectables, Ltd	Minato-ku	Japan
Bavarian Collectables Imports, Co.	Munich	Germany
Blauer See Auto, Co.	Frankfurt	Germany
Cramer Spezialitäten, Ltd	Brandenburg	Germany

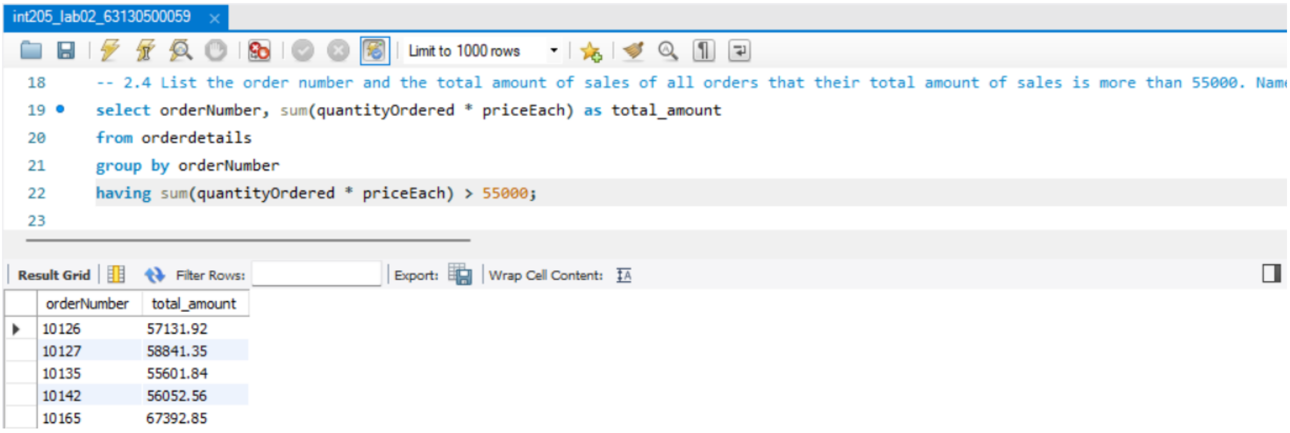
2.3 List the order number and the total amount of sales for all orders. Name the total amount of sales column to “total_amount”.



```
-- 2.3 List the order number and the total amount of sales for all orders. Name the total amount of sales column to “total_amount”
select orderNumber, sum(quantityOrdered * priceEach) as total_amount
from orderdetails
group by orderNumber;
```

orderNumber	total_amount
10100	10223.83
10101	10549.01
10102	5494.78
10103	50218.95
10104	40206.20

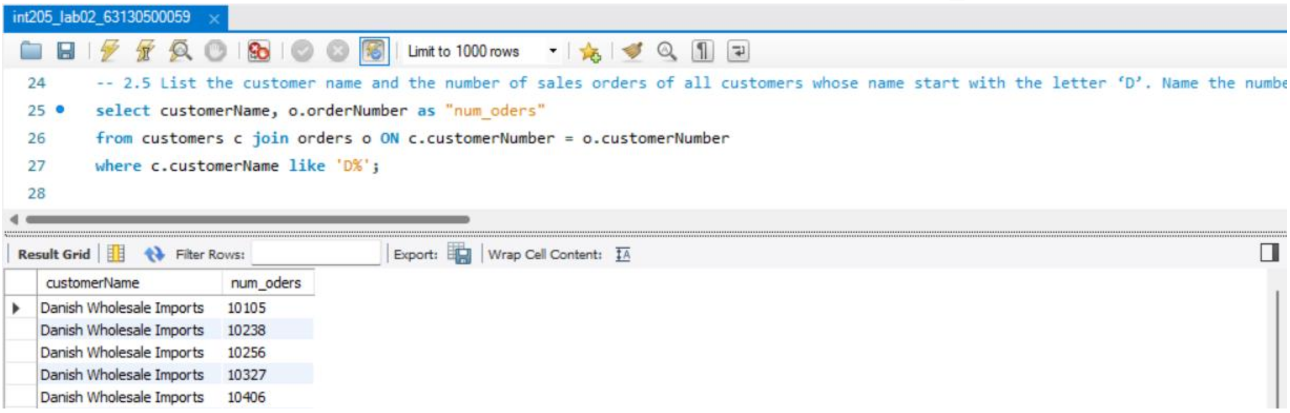
2.4 List the order number and the total amount of sales of all orders that their total amount of sales is more than 55000. Name the total amount of sales column to “total_amount”.



```
-- 2.4 List the order number and the total amount of sales of all orders that their total amount of sales is more than 55000. Nam
select orderNumber, sum(quantityOrdered * priceEach) as total_amount
from orderdetails
group by orderNumber
having sum(quantityOrdered * priceEach) > 55000;
```

orderNumber	total_amount
10126	57131.92
10127	58841.35
10135	55601.84
10142	56052.56
10165	67392.85

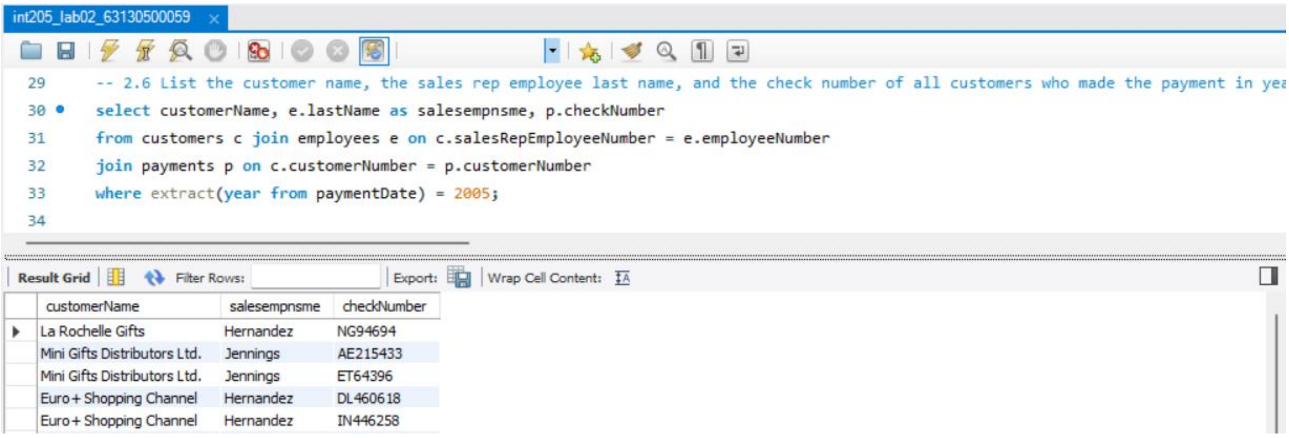
2.5 List the customer name and the number of sales orders of all customers whose name start with the letter ‘D’. Name the number of sales orders column to “num_orders”.



```
-- 2.5 List the customer name and the number of sales orders of all customers whose name start with the letter ‘D’. Name the numbe
select customerName, o.orderNumber as "num_oders"
from customers c join orders o ON c.customerNumber = o.customerNumber
where c.customerName like 'D%';
```

customerName	num_oders
Danish Wholesale Imports	10105
Danish Wholesale Imports	10238
Danish Wholesale Imports	10256
Danish Wholesale Imports	10327
Danish Wholesale Imports	10406

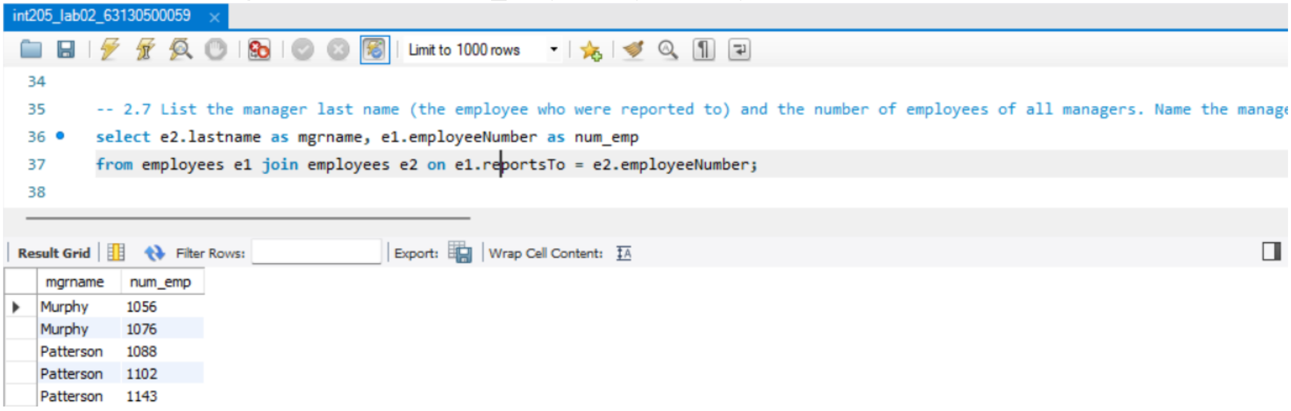
2.6 List the customer name, the sales rep employee last name, and the check number of all customers who made the payment in year 2005. Name the sales rep employee last name column to “salesempname”.



```
-- 2.6 List the customer name, the sales rep employee last name, and the check number of all customers who made the payment in yea
select customerName, e.lastName as salesempnsme, p.checkNumber
from customers c join employees e on c.salesRepEmployeeNumber = e.employeeNumber
join payments p on c.customerNumber = p.customerNumber
where extract(year from paymentDate) = 2005;
```

customerName	salesempnsme	checkNumber
La Rochelle Gifts	Hernandez	NG94694
Mini Gifts Distributors Ltd.	Jennings	AE215433
Mini Gifts Distributors Ltd.	Jennings	ET64396
Euro+ Shopping Channel	Hernandez	DL460618
Euro+ Shopping Channel	Hernandez	IN446258

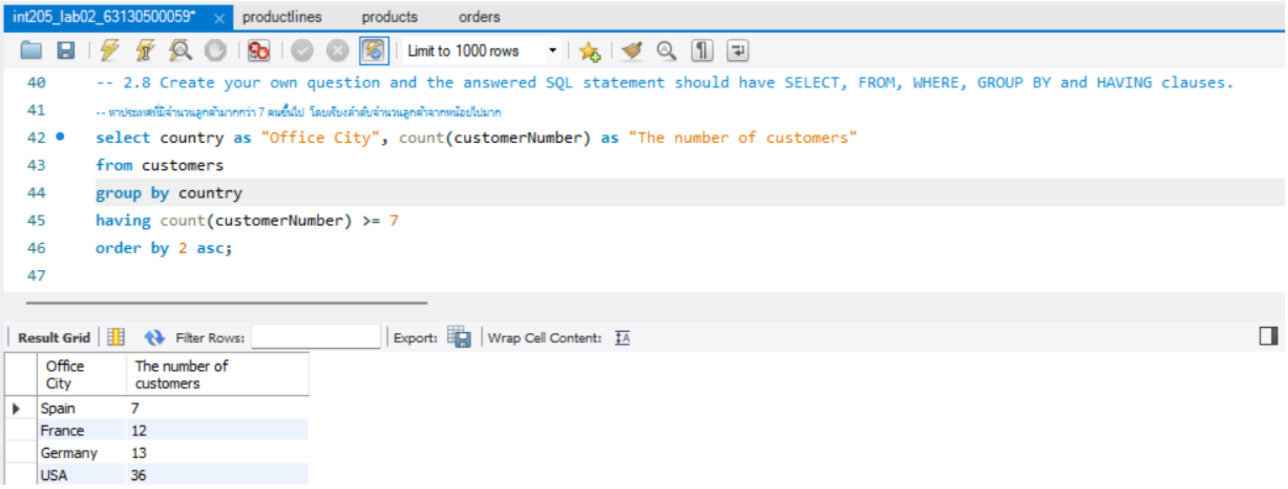
2.7 List the manager last name (the employee who were reported to) and the number of employees of all managers. Name the manager last name and the number of employees columns to “mgrname” and “num_emp”, respectively.



```
-- 2.7 List the manager last name (the employee who were reported to) and the number of employees of all managers. Name the manage
select e2.lastname as mgrname, e1.employeeNumber as num_emp
from employees e1 join employees e2 on e1.reportsTo = e2.employeeNumber;
```

mgrname	num_emp
Murphy	1056
Murphy	1076
Patterson	1088
Patterson	1102
Patterson	1143

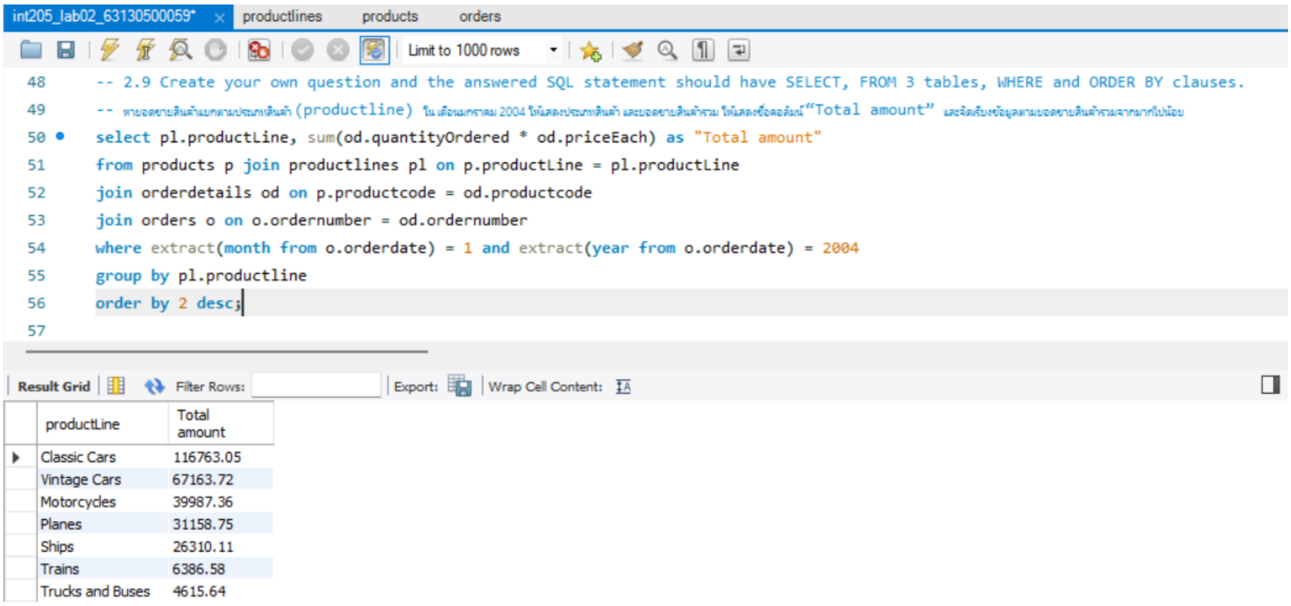
2.8 หาประเทศที่มีจำนวนลูกค้ามากกว่า 7 คนขึ้นไป โดยเรียงลำดับจำนวนลูกค้าจากหน้อยไปมาก



```
-- 2.8 Create your own question and the answered SQL statement should have SELECT, FROM, WHERE, GROUP BY and HAVING clauses.
-- หาประเทศที่มีจำนวนลูกค้ามากกว่า 7 คนขึ้นไป โดยเรียงลำดับจำนวนลูกค้าจากหน้อยไปมาก
select country as "Office City", count(customerNumber) as "The number of customers"
from customers
group by country
having count(customerNumber) >= 7
order by 2 asc;
```

Office City	The number of customers
Spain	7
France	12
Germany	13
USA	36

2.9 หายอดขายสินค้าแยกตามประเภทสินค้า (productline) ใน เดือนมกราคม 2004 ให้แสดงประเภทสินค้า และยอดขายสินค้ารวม ให้แสดงชื่อคอลัมน์ “Total amount” และจัดเรียงข้อมูลตามยอดขายสินค้ารวมจากมากไปน้อย



```
-- 2.9 Create your own question and the answered SQL statement should have SELECT, FROM 3 tables, WHERE and ORDER BY clauses.
-- หายอดขายสินค้าแยกตามประเภทสินค้า (productline) ใน เดือนมกราคม 2004 ให้แสดงประเภทสินค้า และยอดขายสินค้ารวม ให้แสดงชื่อคอลัมน์“Total amount” และจัดเรียงข้อมูลตามยอดขายสินค้ารวมจากมากไปน้อย
select pl.productLine, sum(od.quantityOrdered * od.priceEach) as "Total amount"
from products p join productlines pl on p.productLine = pl.productLine
join orderdetails od on p.productcode = od.productcode
join orders o on o.ordernumber = od.ordernumber
where extract(month from o.orderdate) = 1 and extract(year from o.orderdate) = 2004
group by pl.productline
order by 2 desc;
```

productLine	Total amount
Classic Cars	116763.05
Vintage Cars	67163.72
Motorcycles	39987.36
Planes	31158.75
Ships	26310.11
Trains	6386.58
Trucks and Buses	4615.64