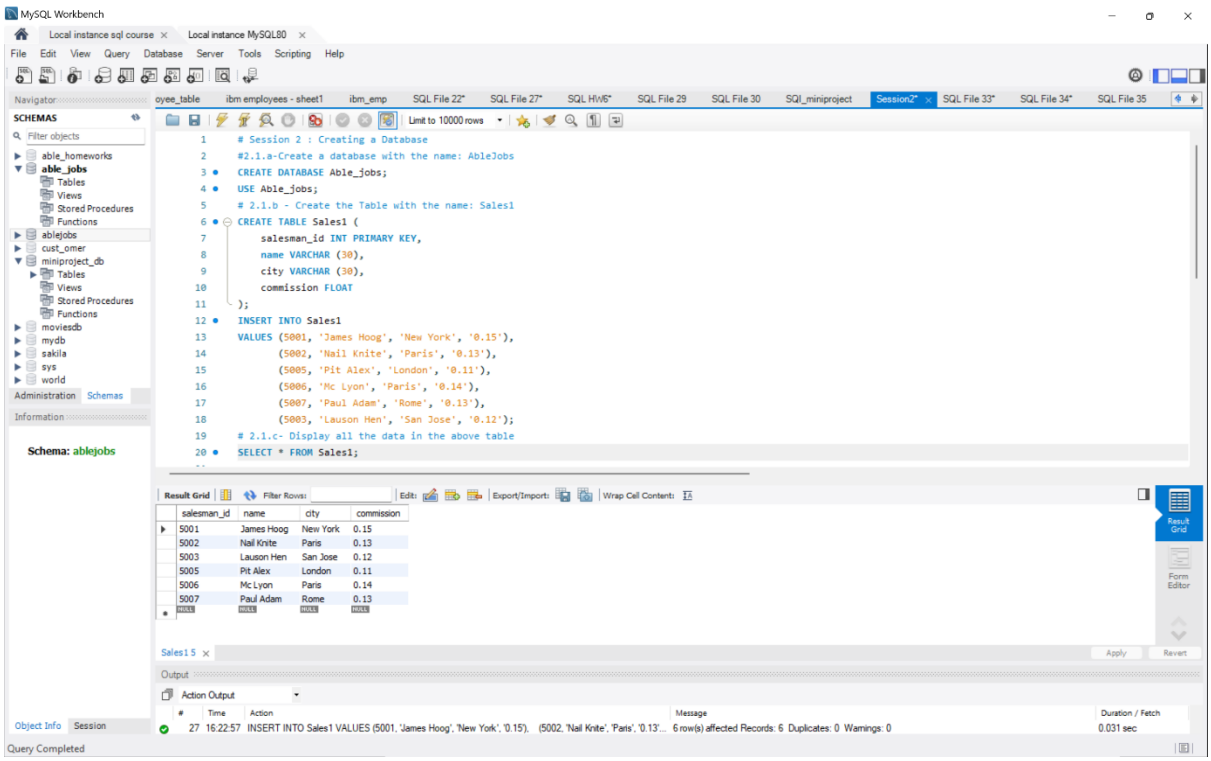
SQL: Capstone Project

Session 2: Creating a Database

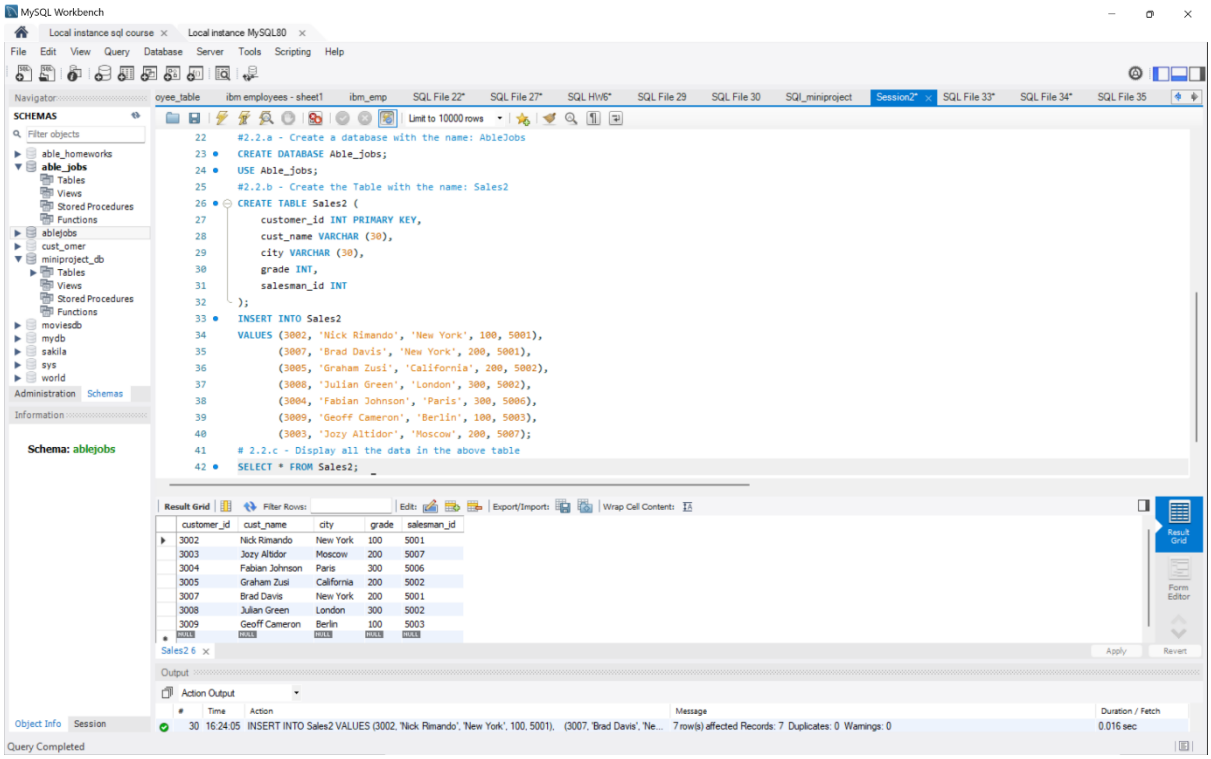
1.

a. Create a database with the name: AbleJobs
b. Create the following Table with the name: Sales1
c. Display all the data in the above table



2.

a. Create a database with the name: AbleJobs
b. Create the following Table with the name: Sales
c. Display all the data in the above table



Session 3: Other Basic Queries

1.

a. Create a database with the name: AbleJobs

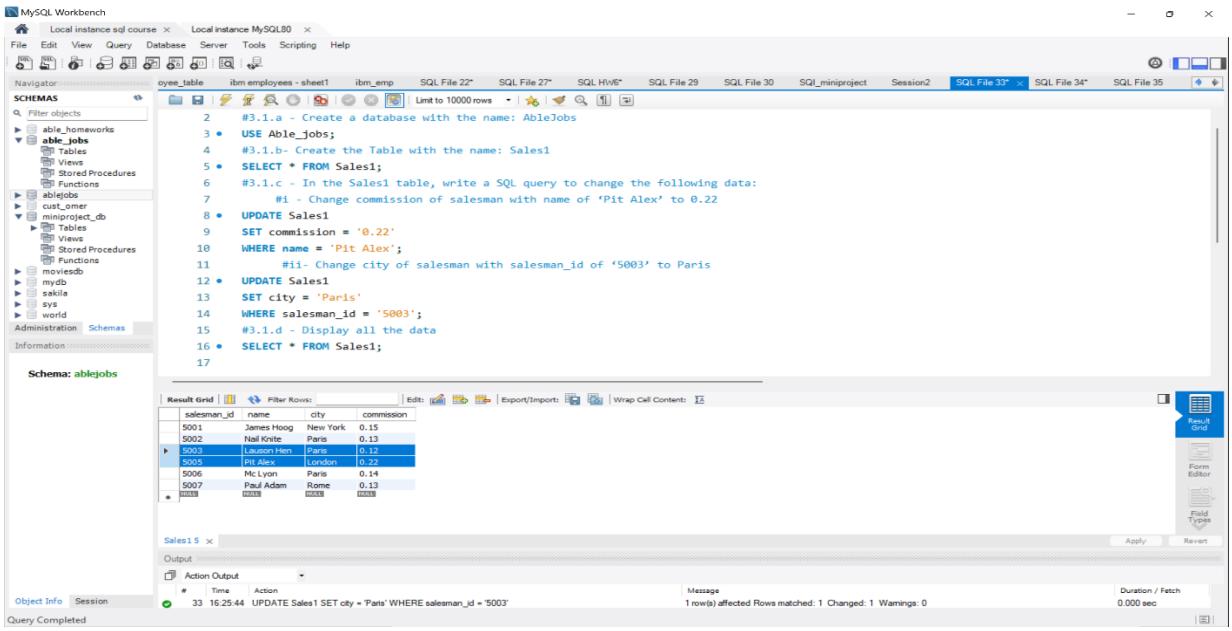
b. Create the following Table with the name: Sales1

c. In the above table, write a SQL query to change the following data:

i. Change commission of salesman with name of ‘Pit Alex’ to 0.22

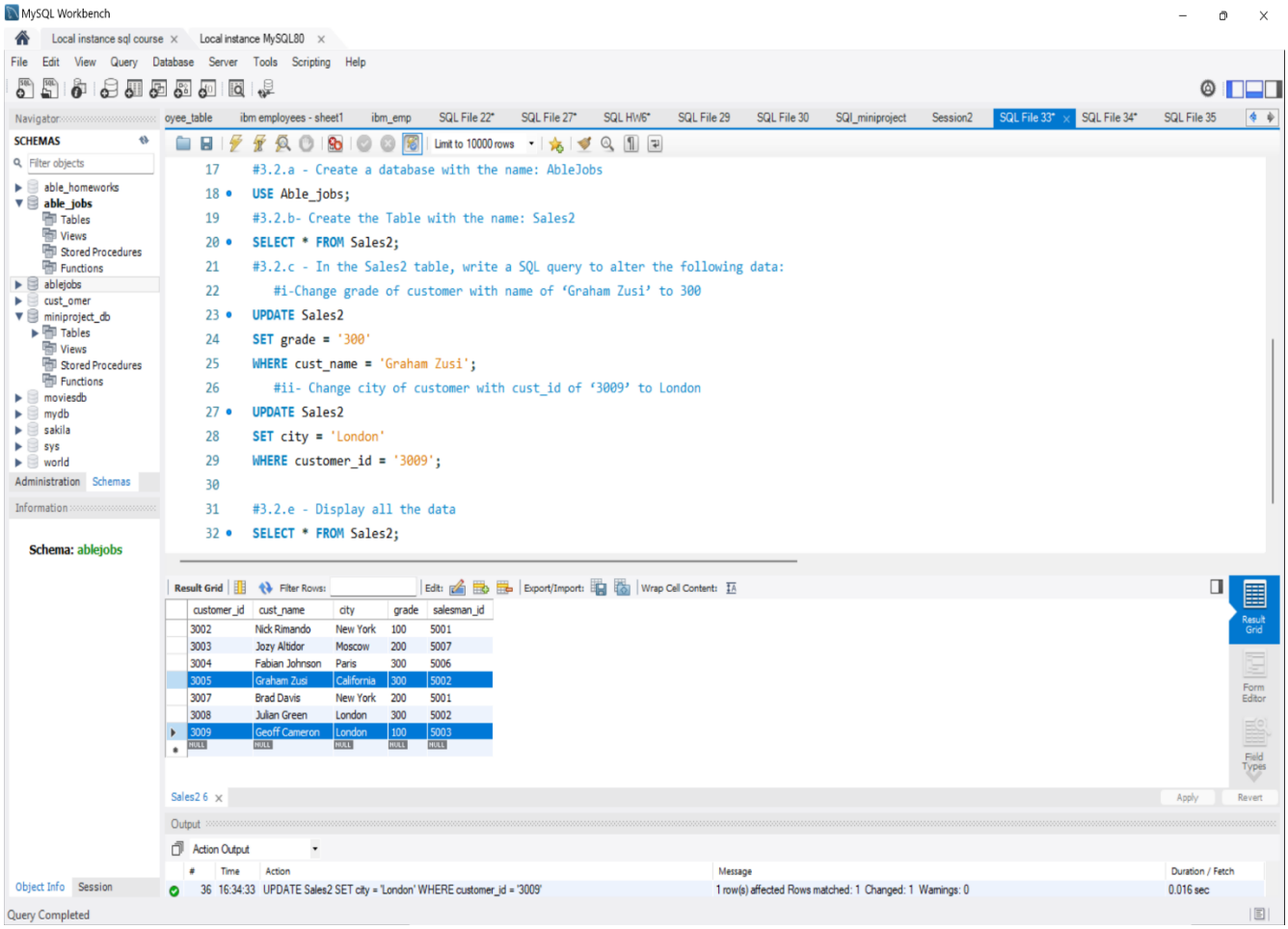
ii. Change city of salesman with salesman_id of ‘5003’ to Paris

d. Display all the data in the above table



2.

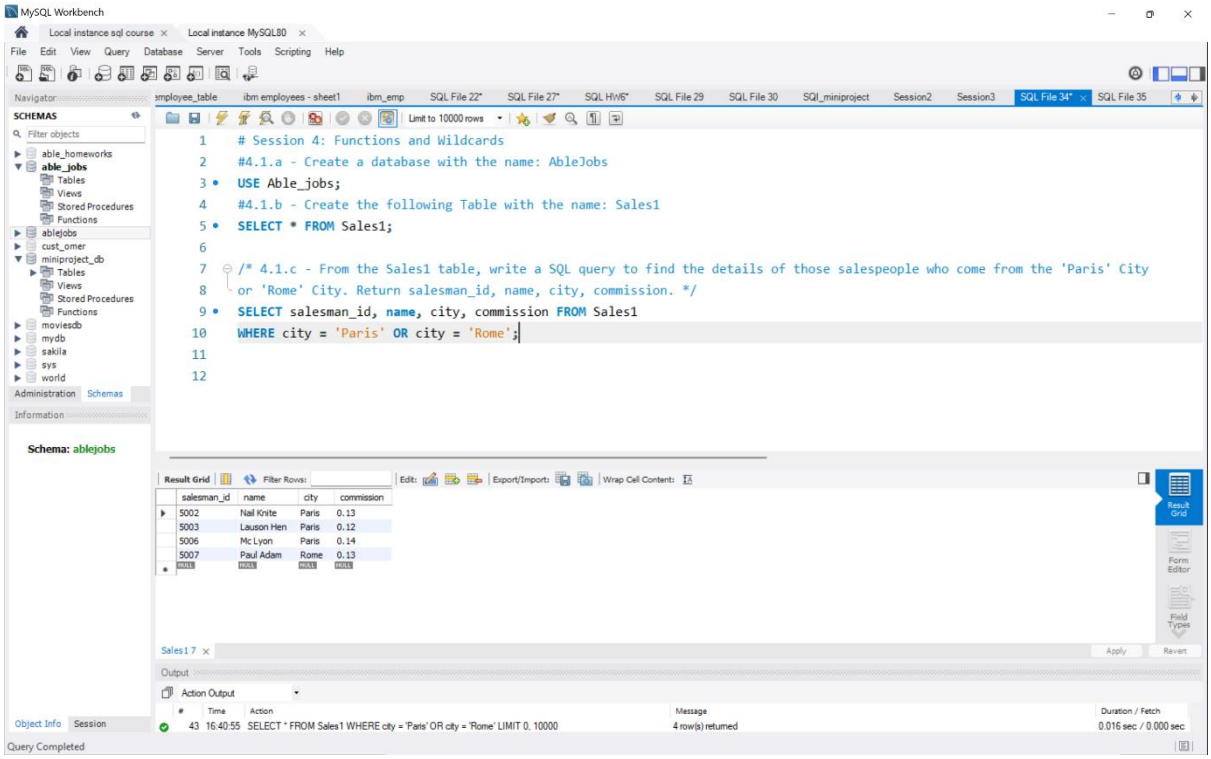
a. Create a database with the name: AbleJobs
b. Create the following Table with the name: Sales2
c. In the above table, write a SQL query to alter the following data:
i.Change grade of customer with name of ‘Graham Zusi’ to 300
ii. Change city of customer with cust_id of ‘3009’ to London
d.Display all the data in the above table



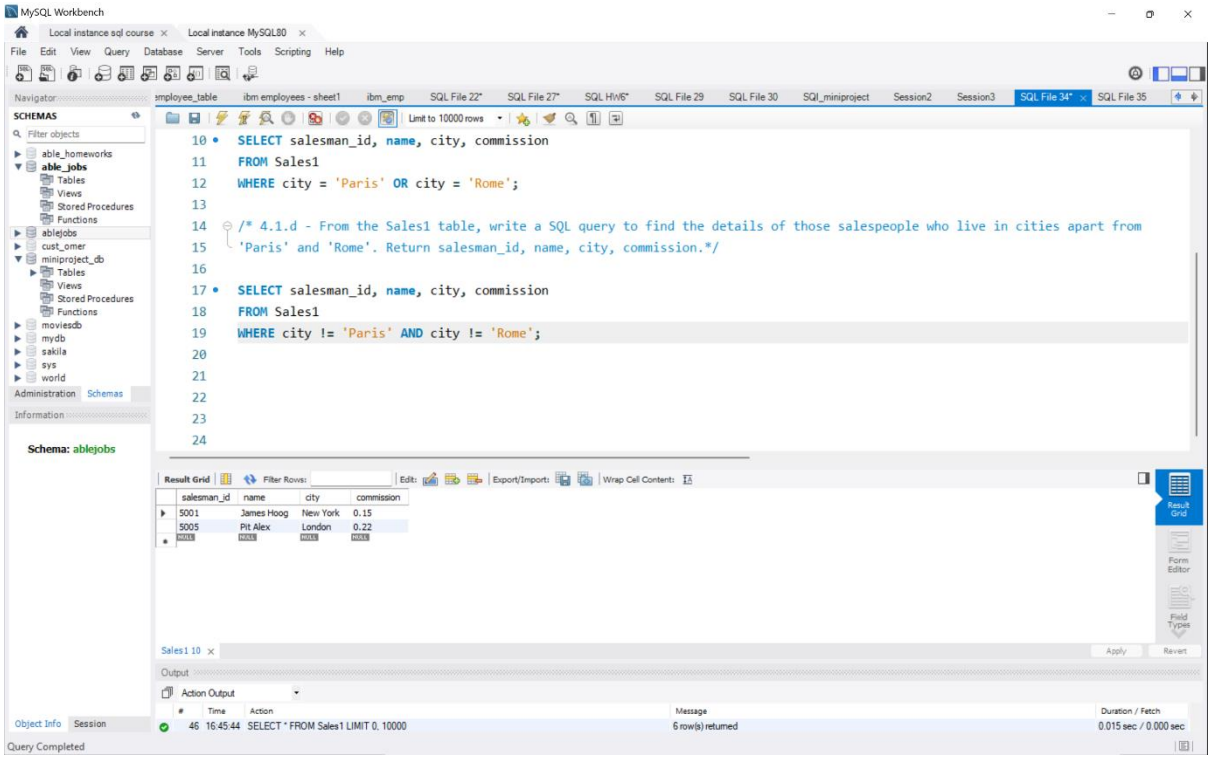
Session 4: Functions and Wildcards

1.

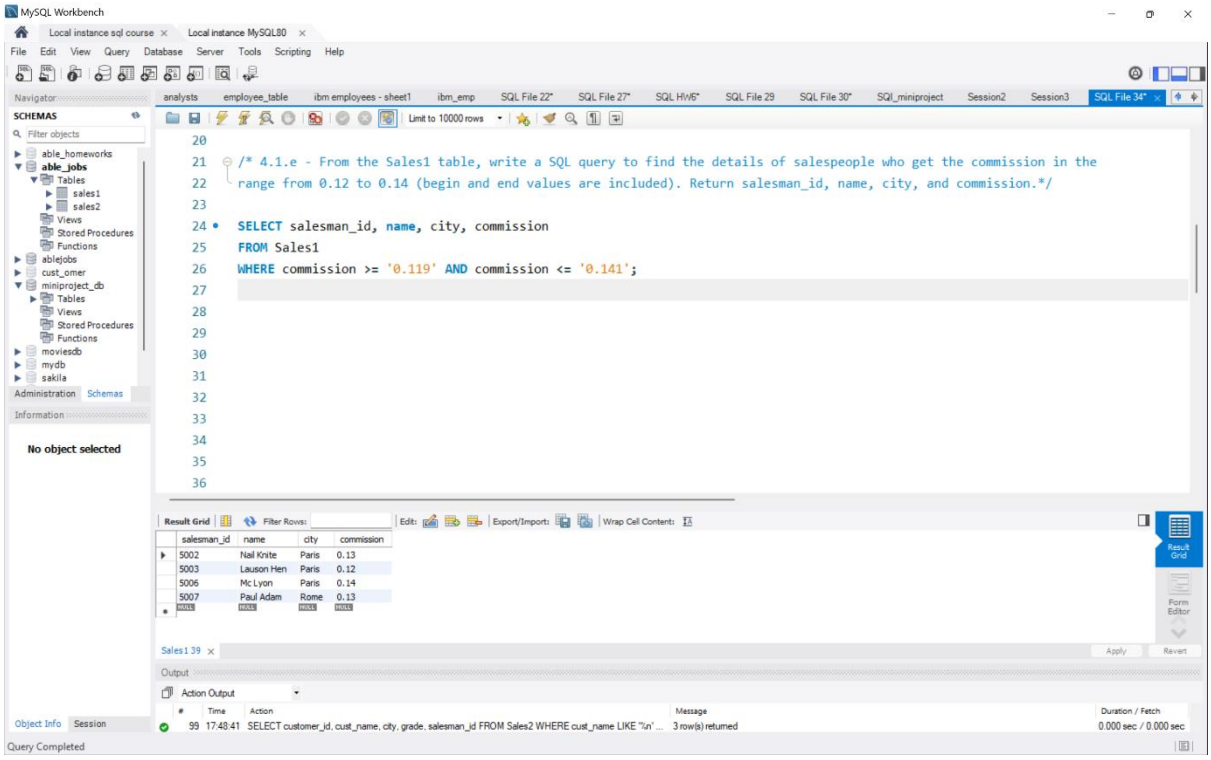
a. Create a database with the name: AbleJobs
b. Create the following Table with the name: Sales1
c. From the above table, write a SQL query to find the details of those salespeople who come from the 'Paris' City or 'Rome' City. Return salesman_id, name, city, commission.



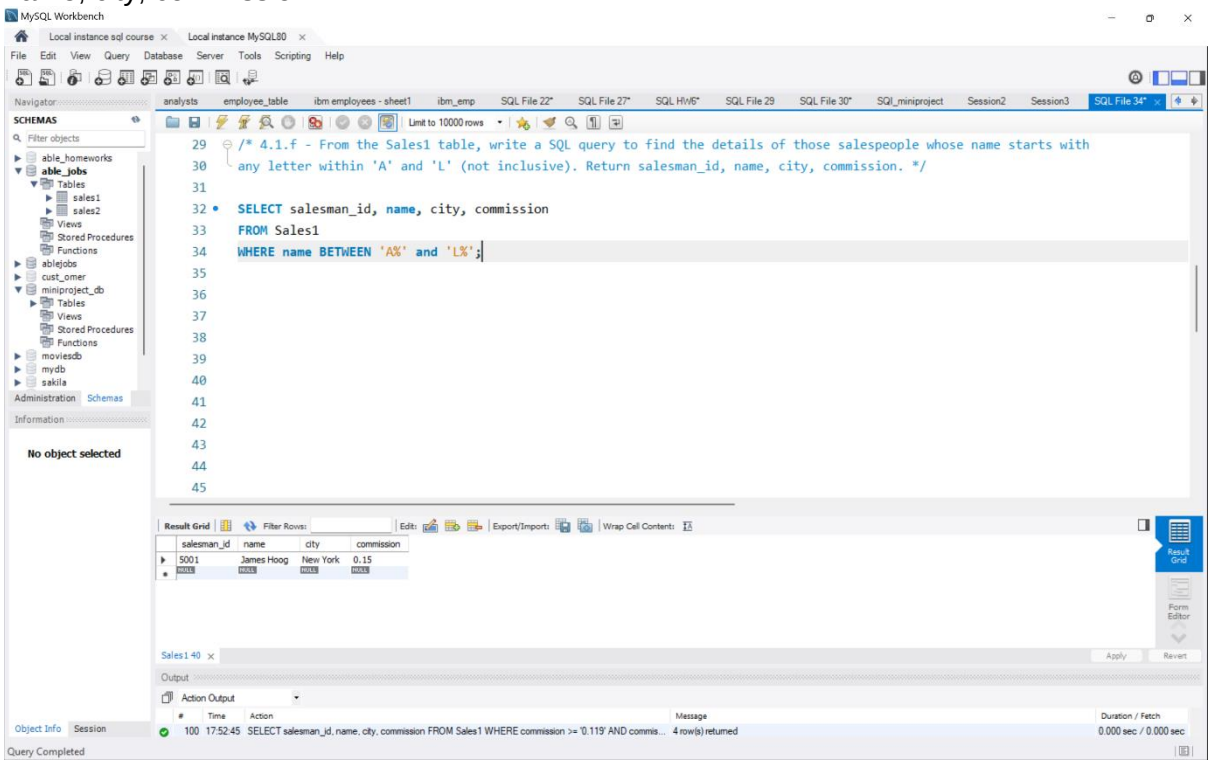
d. From the following table, write a SQL query to find the details of those salespeople who live in cities apart from 'Paris' and 'Rome'. Return salesman_id, name, city, commission.



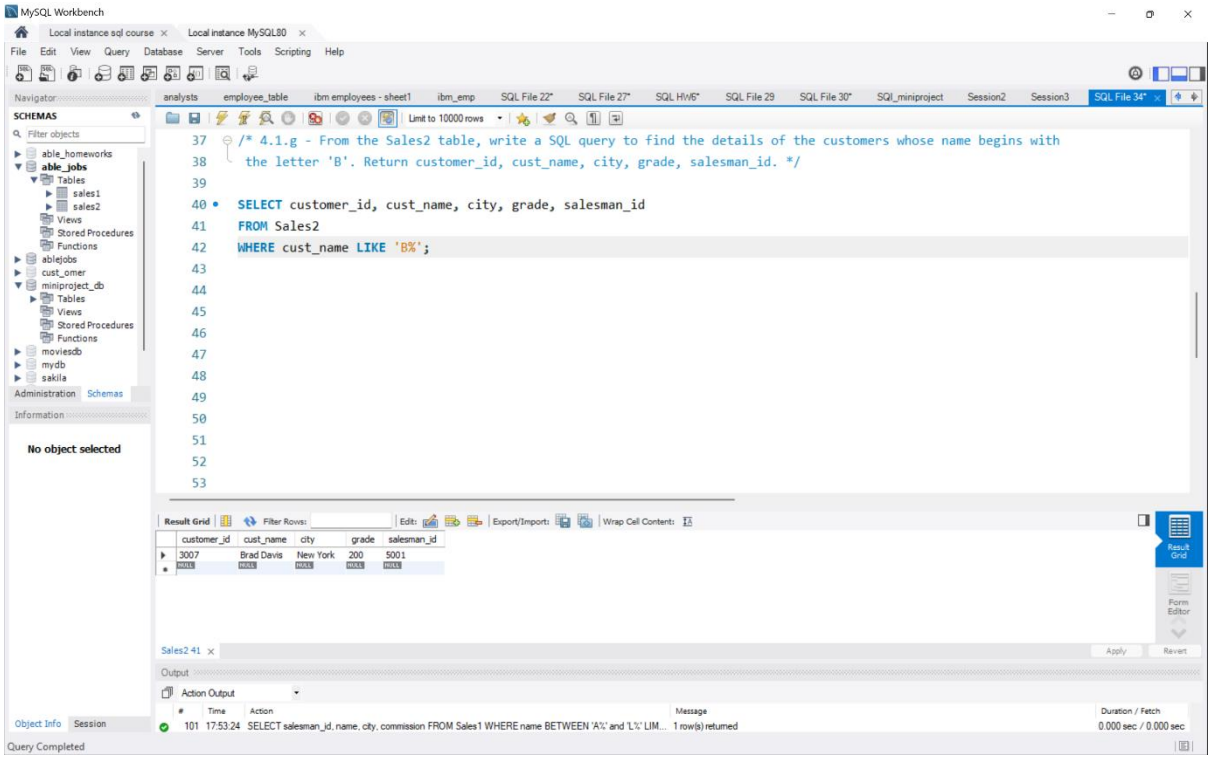
e. From the following table, write a SQL query to find the details of salespeople who get the commission in the range from 0.12 to 0.14 (begin and end values are included). Return salesman_id, name, city, and commission.



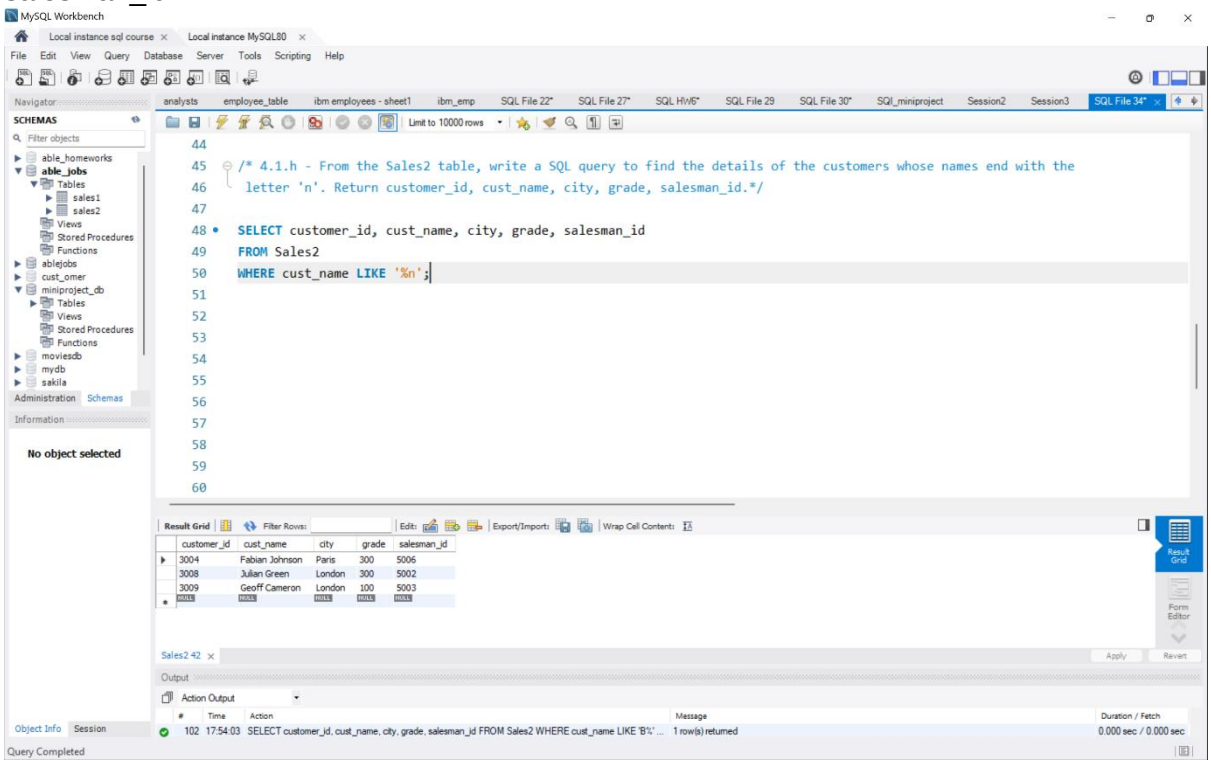
f. From the following table, write a SQL query to find the details of those salespeople whose name starts with any letter within 'A' and 'L' (not inclusive). Return salesman_id, name, city, commission.



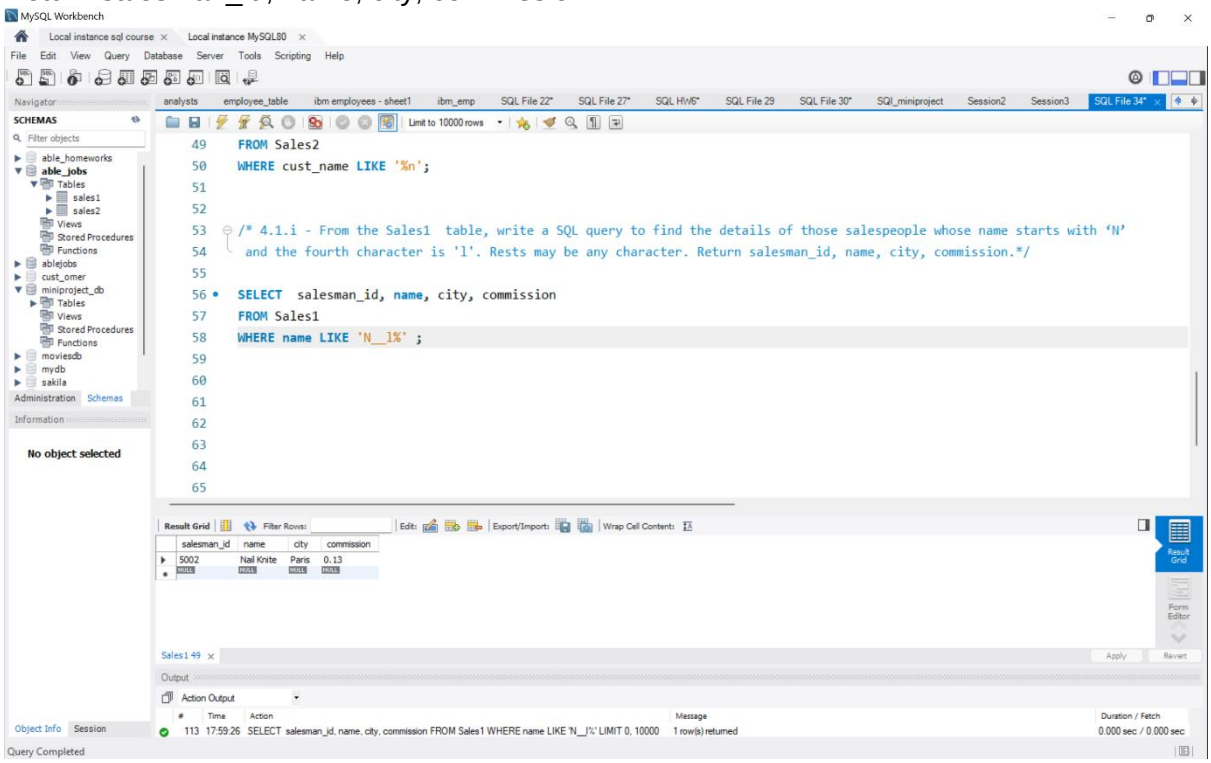
g. From the following table, write a SQL query to find the details of the customers whose name begins with the letter 'B'. Return customer_id, cust_name, city, grade, salesman_id.



h. From the following table, write a SQL query to find the details of the customers whose names end with the letter 'n'. Return customer_id, cust_name, city, grade, salesman_id.



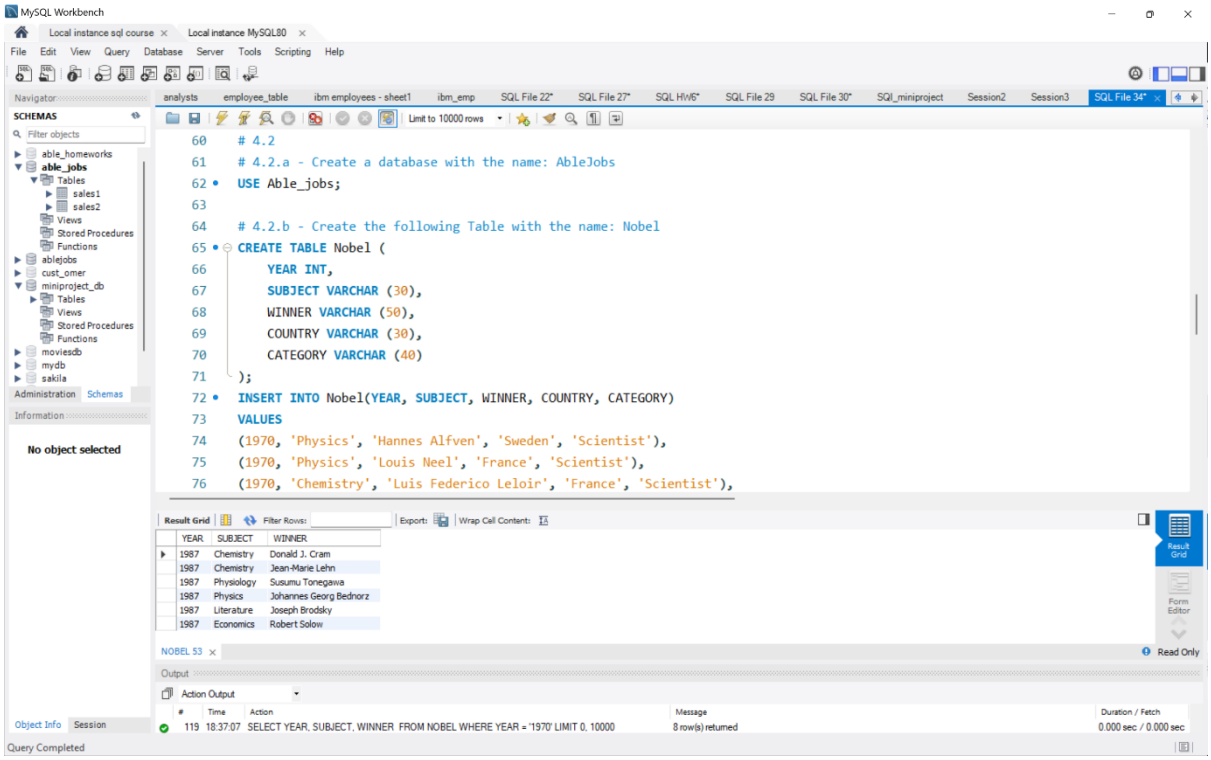
i. From the following table, write a SQL query to find the details of those salespeople whose name starts with ‘N’ and the fourth character is 'l'. Rests may be any character. Return salesman_id, name, city, commission.

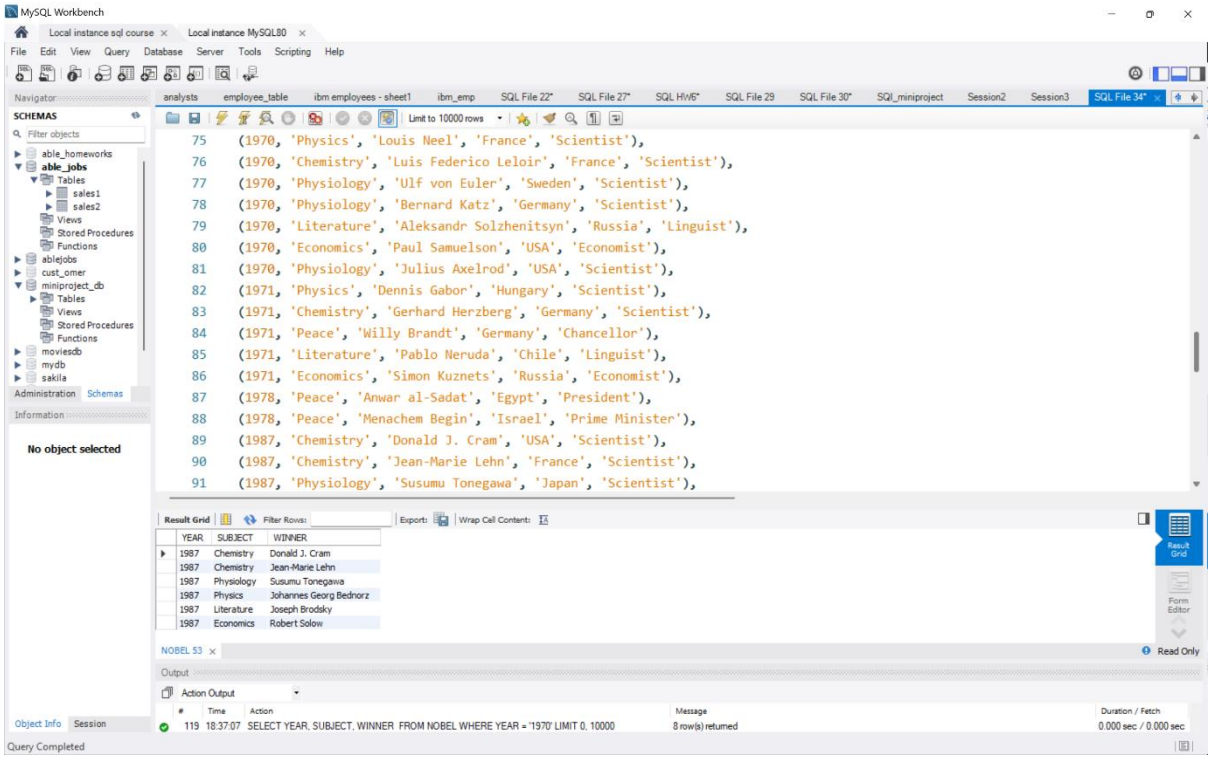


2.

a. Create a database with the name: AbleJobs

b. Create the following Table with the name: Nobel

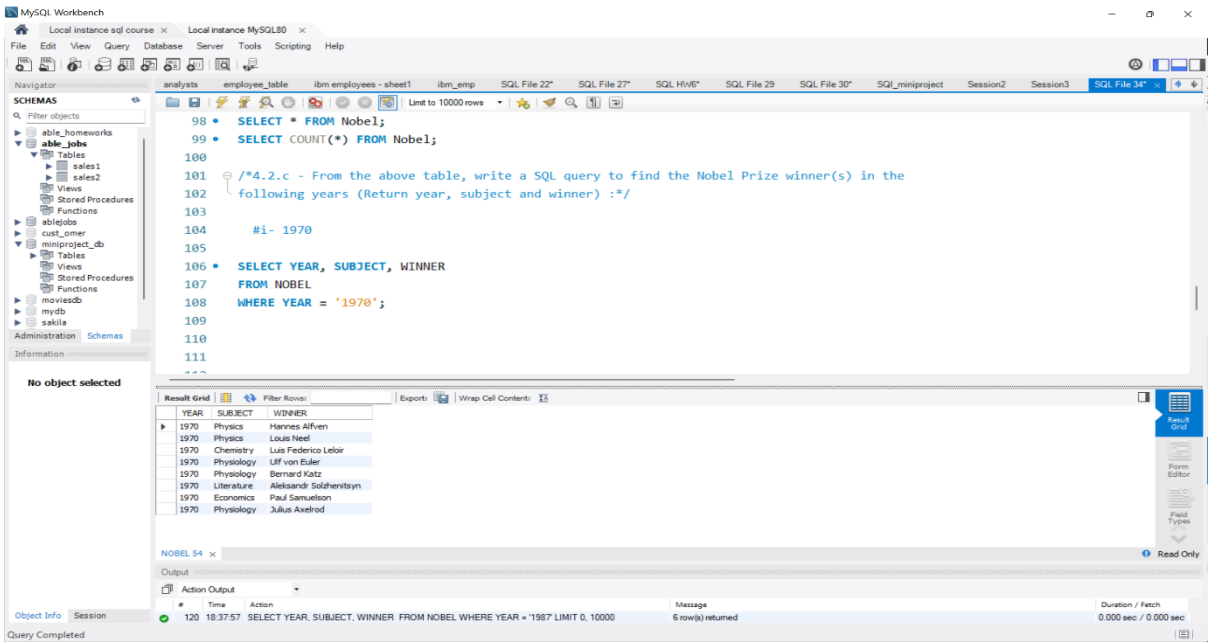


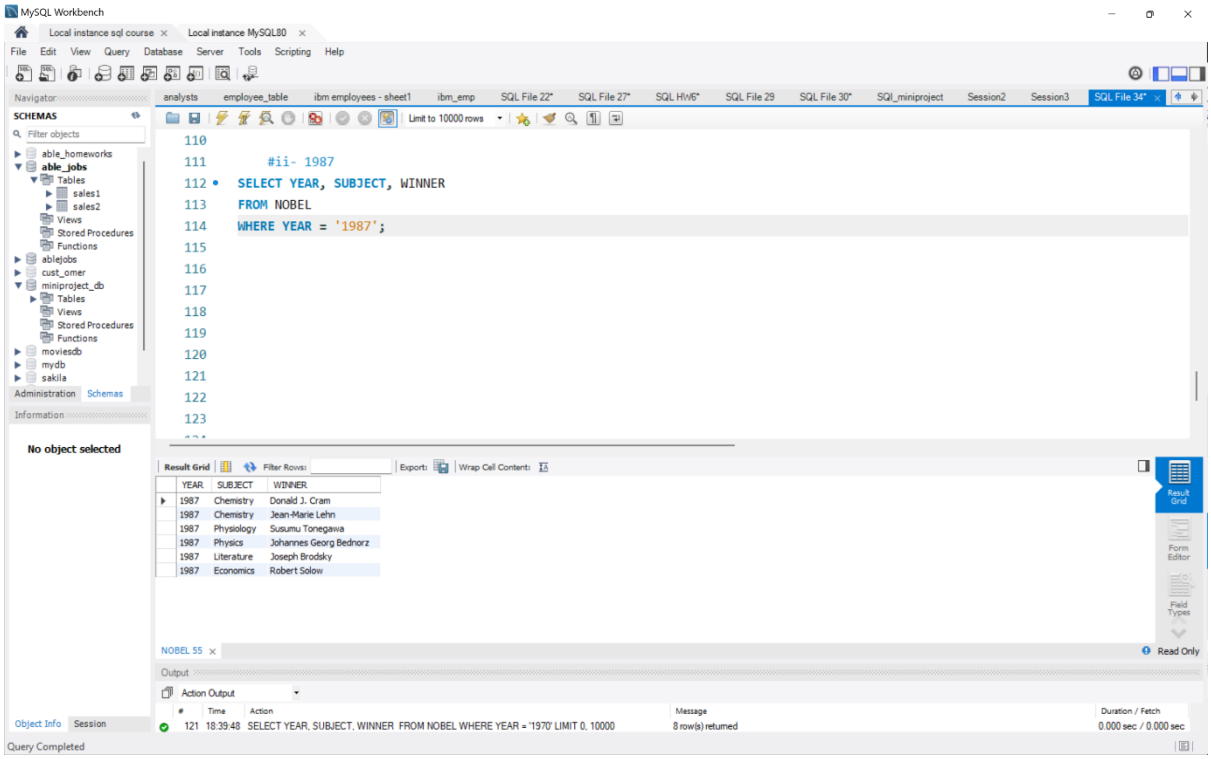


c. From the above table, write a SQL query to find the Nobel Prize winner(s) in the following years (Return year, subject and winner) :

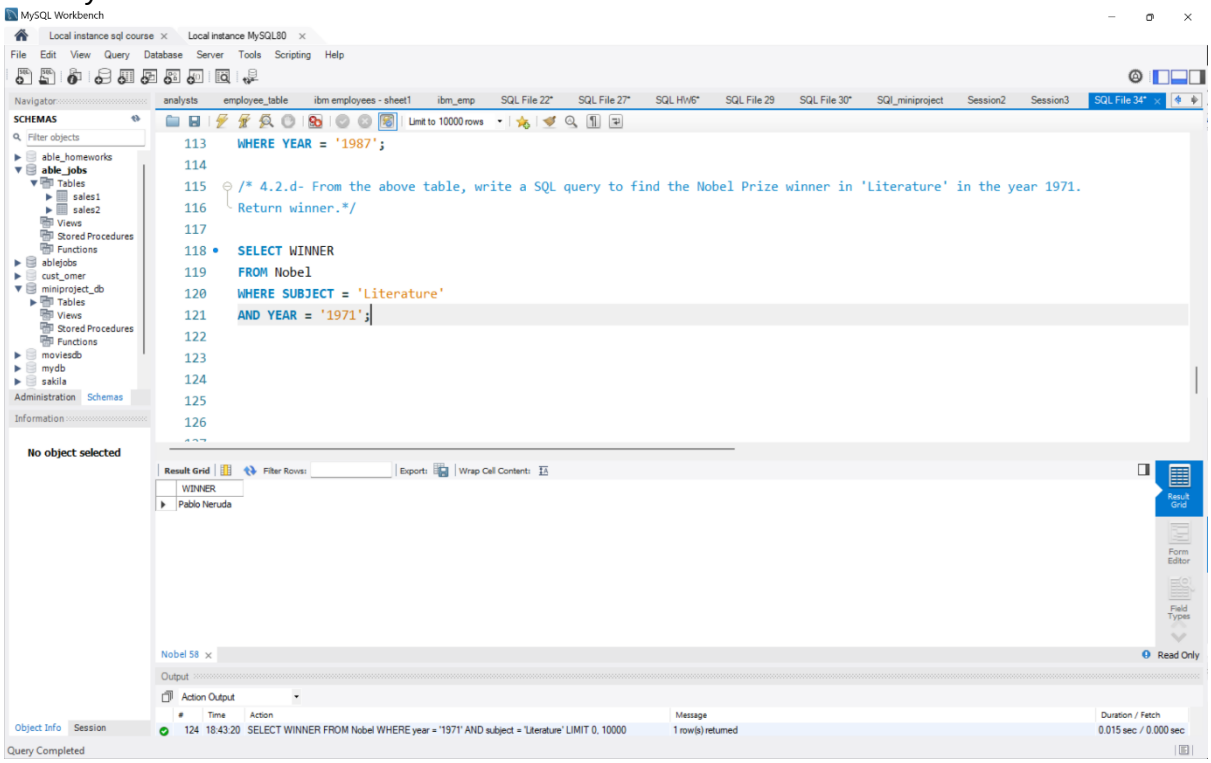
i.1970

ii.1987

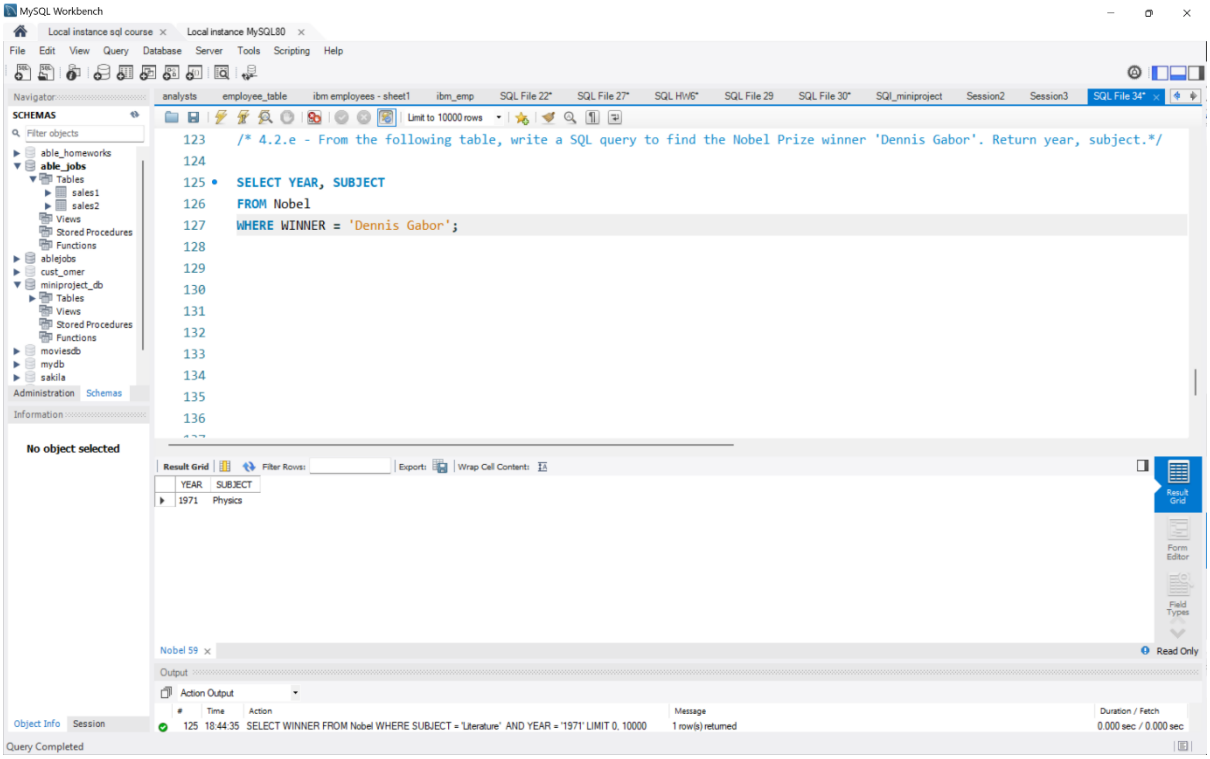




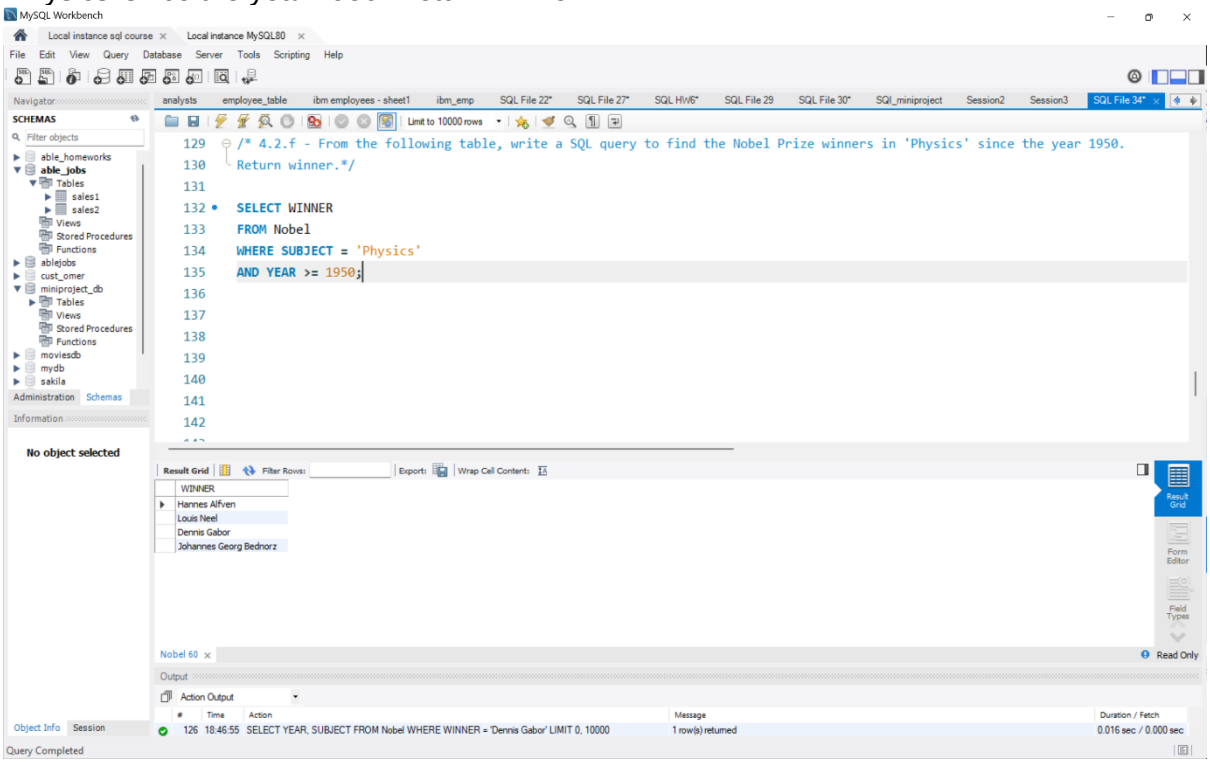
d. From the above table, write a SQL query to find the Nobel Prize winner in 'Literature' in the year 1971. Return winner.



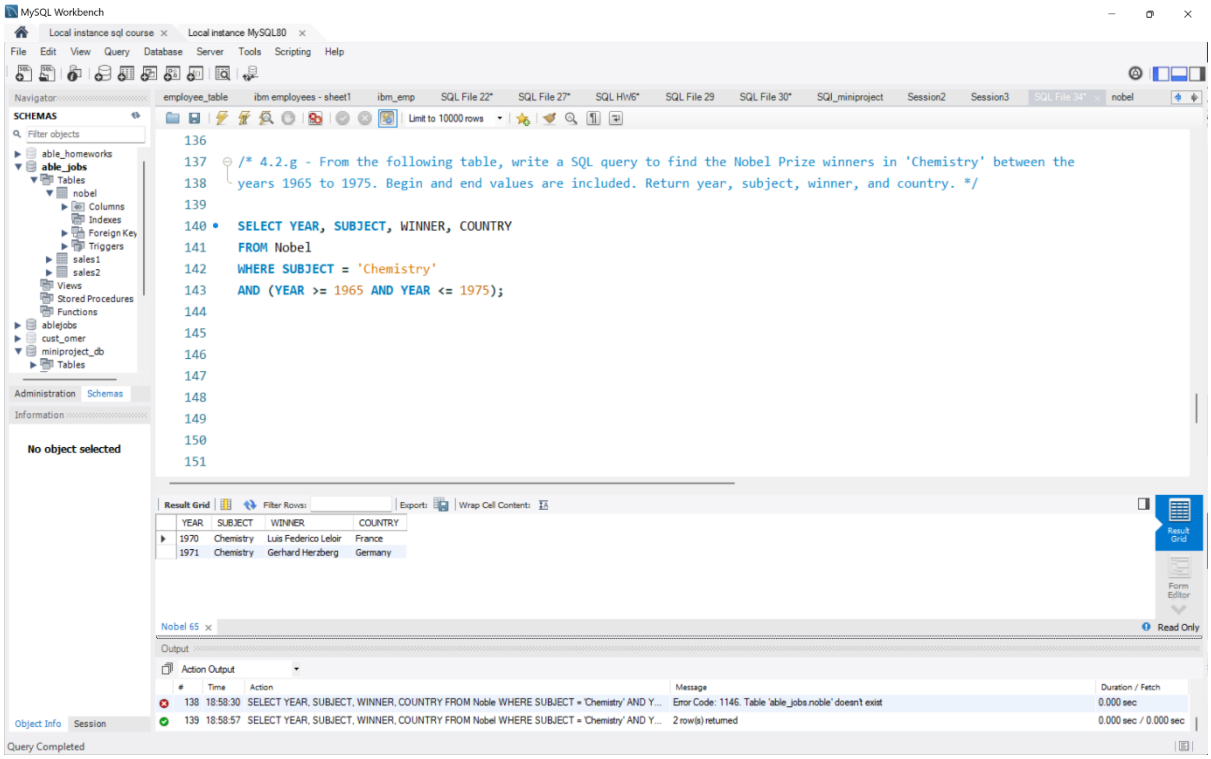
e. From the following table, write a SQL query to find the Nobel Prize winner 'Dennis Gabor'. Return year, subject.



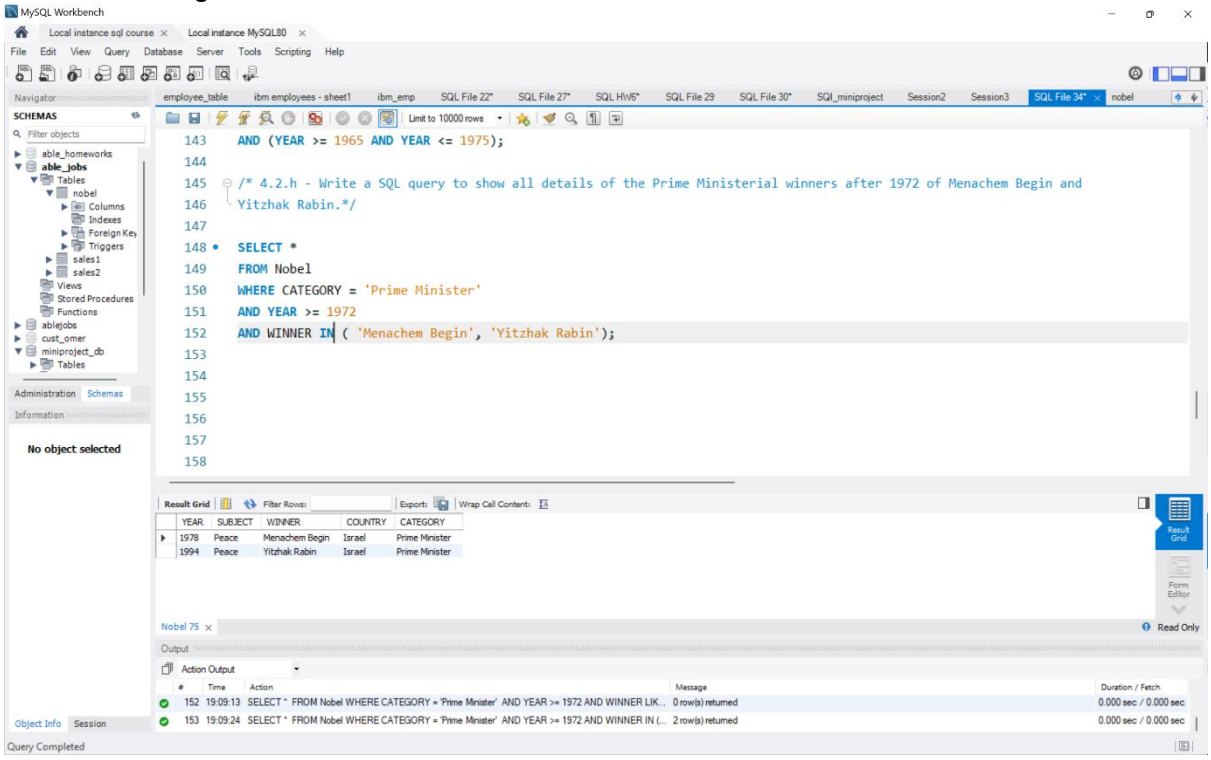
f. From the following table, write a SQL query to find the Nobel Prize winners in 'Physics' since the year 1950. Return winner.



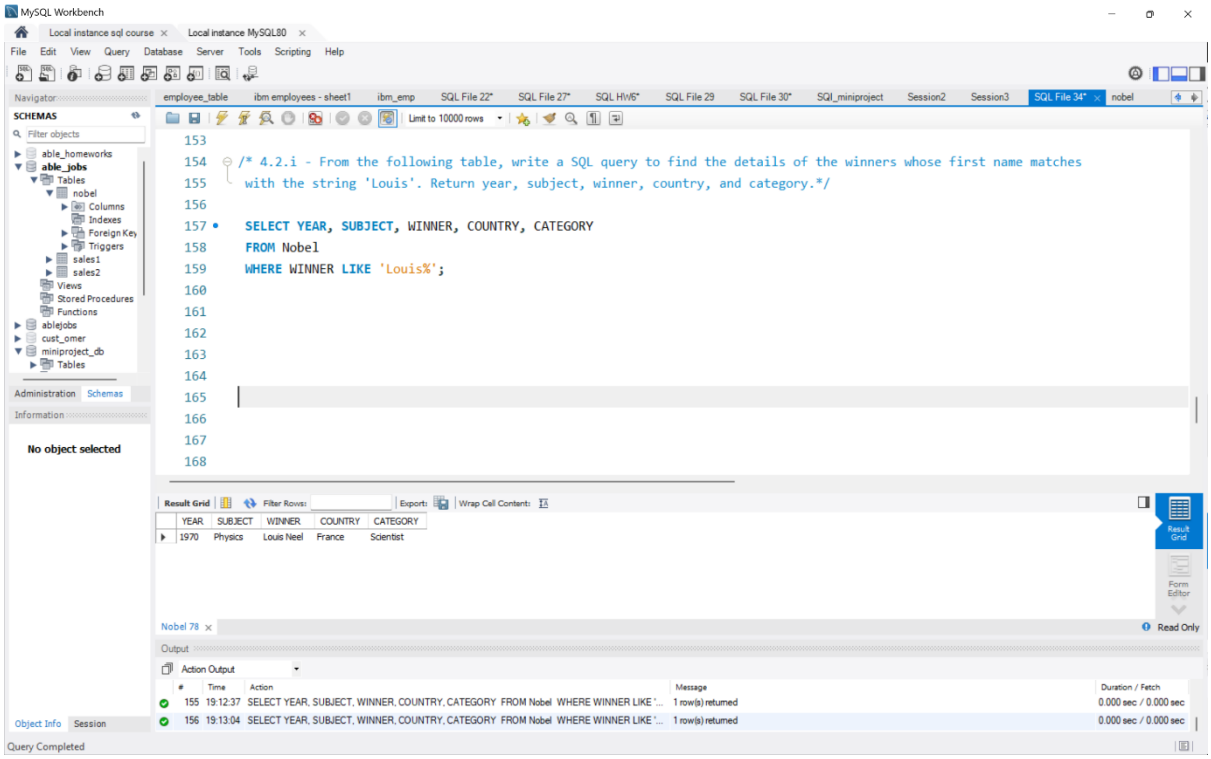
g. From the following table, write a SQL query to find the Nobel Prize winners in 'Chemistry' between the years 1965 to 1975. Begin and end values are included. Return year, subject, winner, and country



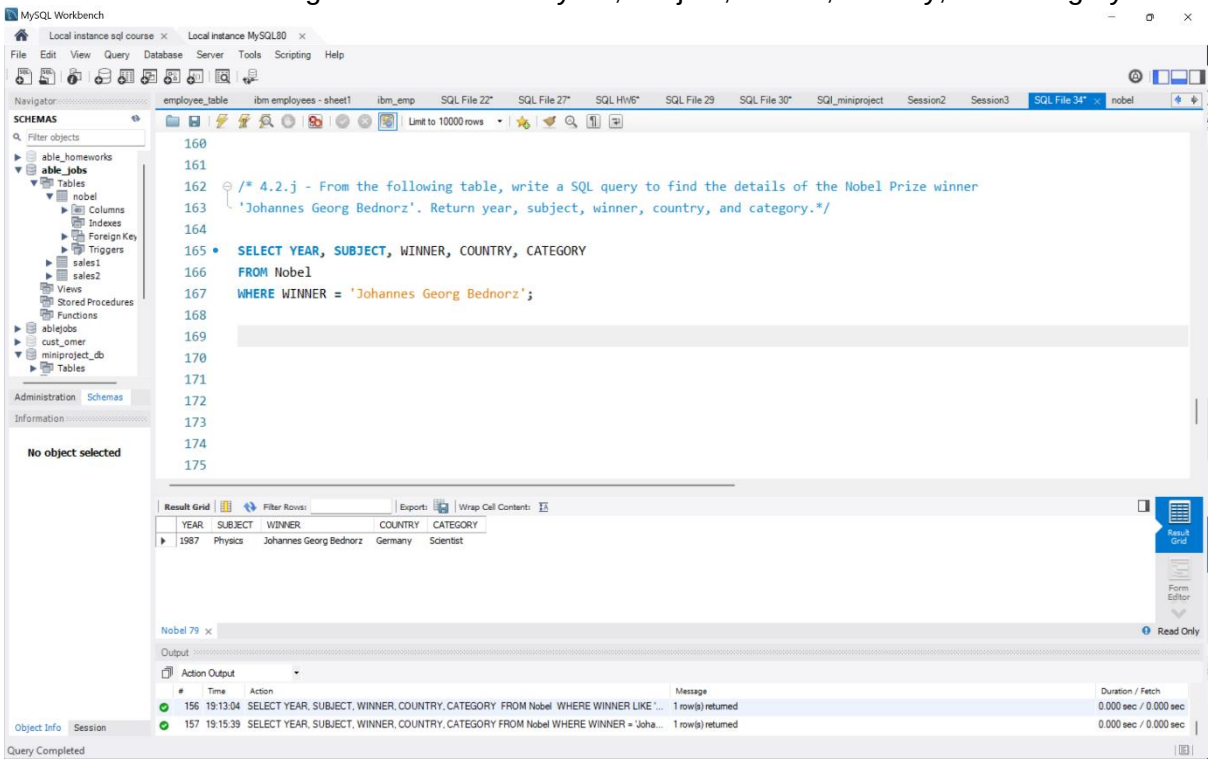
h. Write a SQL query to show all details of the Prime Ministerial winners after 1972 of Menachem Begin and Yitzhak Rabin.



i. From the following table, write a SQL query to find the details of the winners whose first name matches with the string 'Louis'. Return year, subject, winner, country, and category.



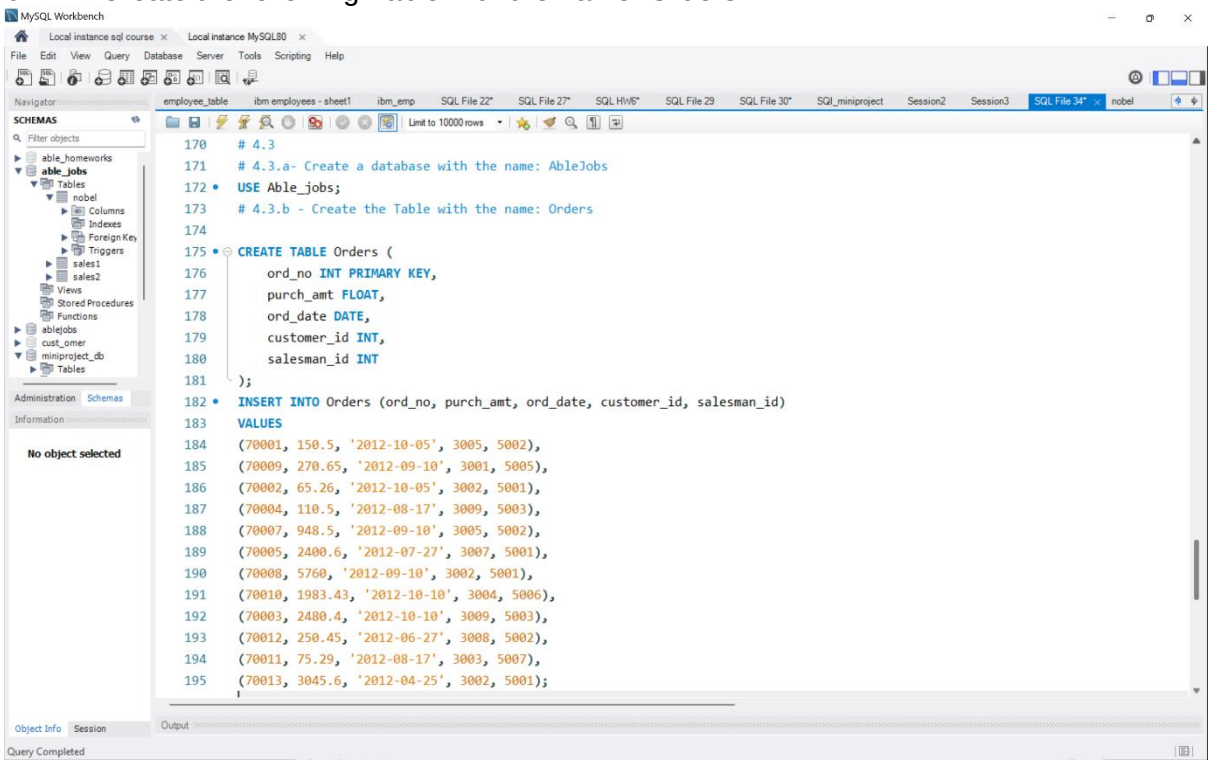
j. From the following table, write a SQL query to find the details of the Nobel Prize winner 'Johannes Georg Bednorz'. Return year, subject, winner, country, and category.



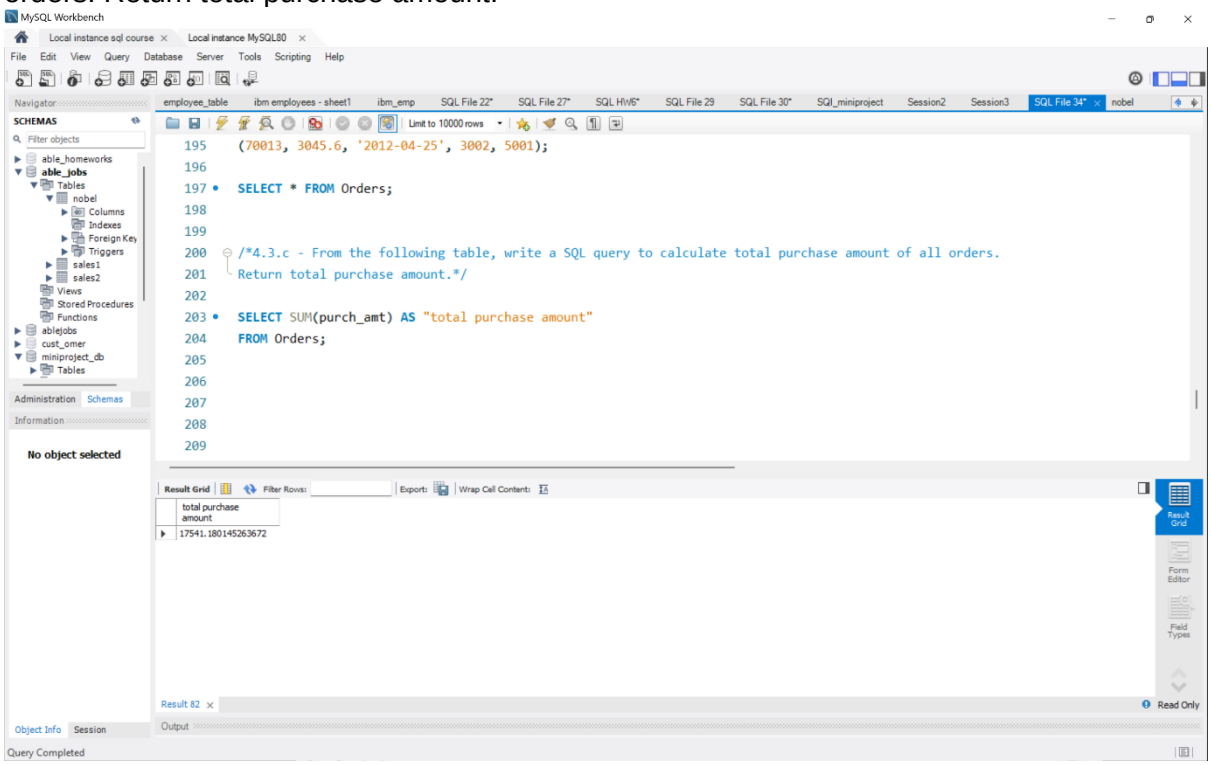
3.

a. Create a database with the name: AbleJobs

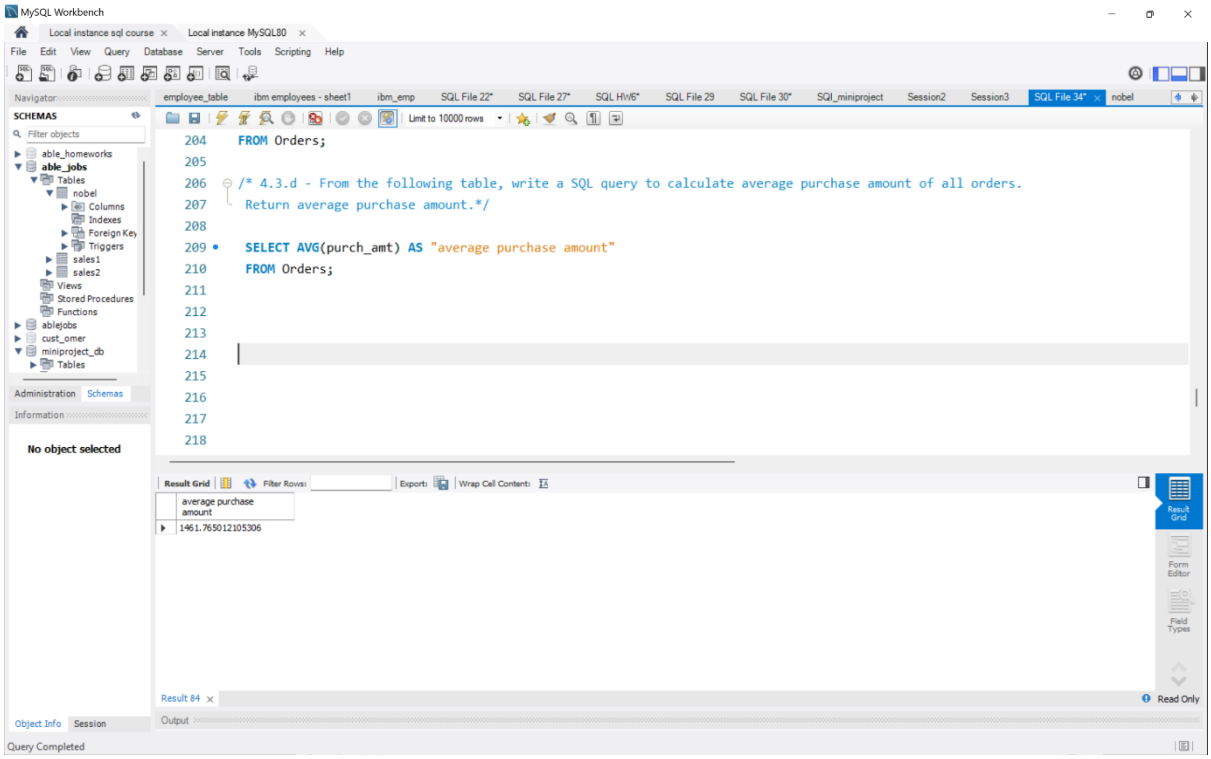
b. Create the following Table with the name: Orders



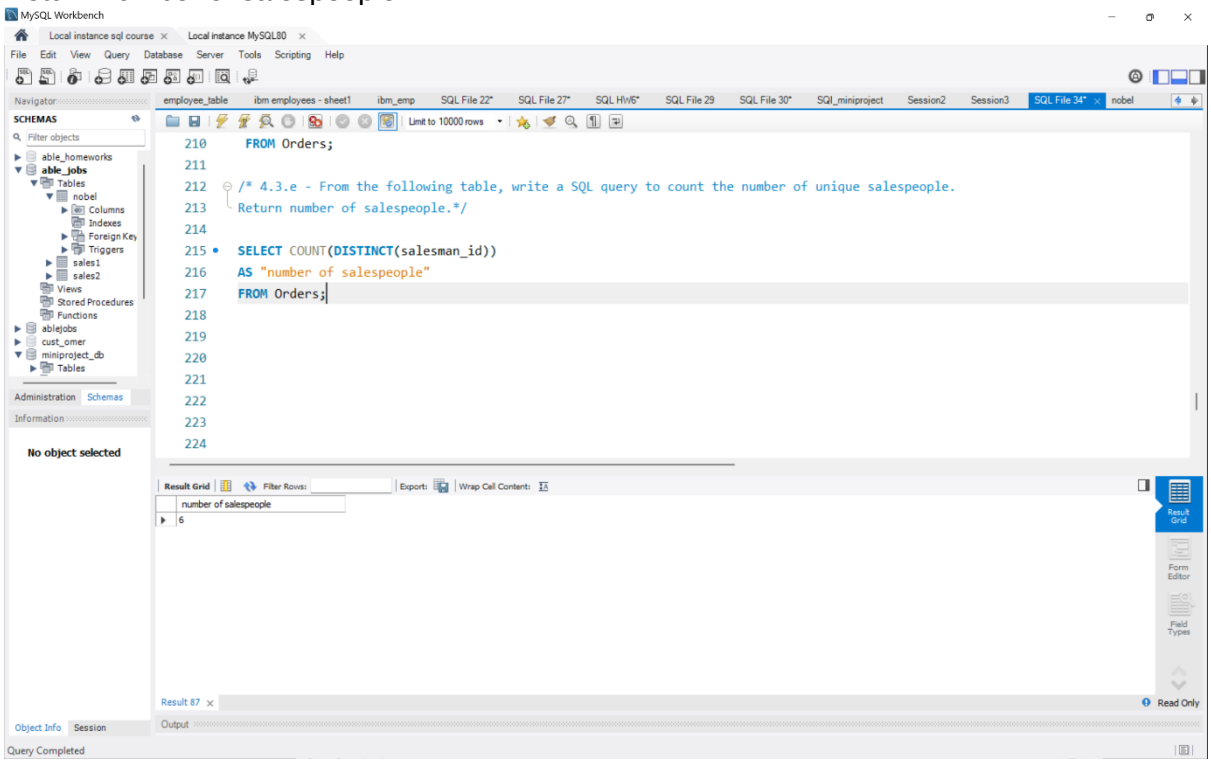
c. From the following table, write a SQL query to calculate total purchase amount of all orders. Return total purchase amount.



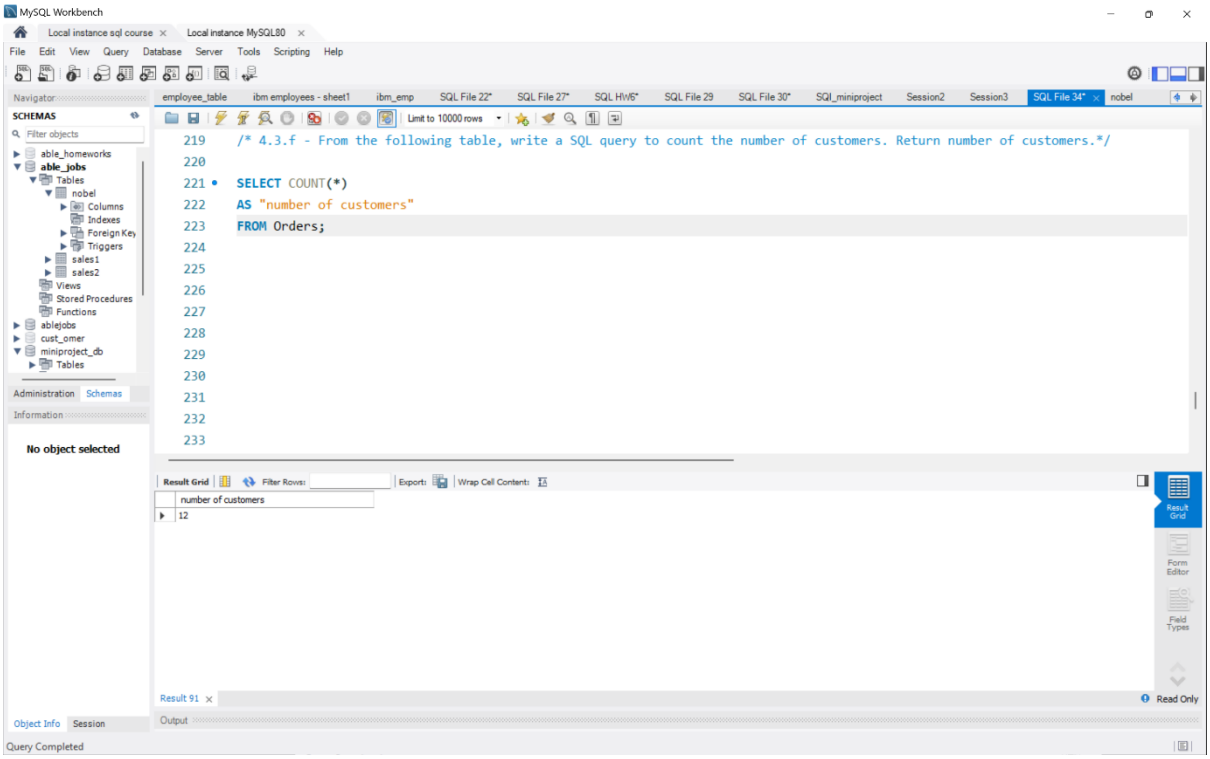
d. From the following table, write a SQL query to calculate average purchase amount of all orders. Return average purchase amount.



e.From the following table, write a SQL query to count the number of unique salespeople. Return number of salespeople.



f. From the following table, write a SQL query to count the number of customers. Return number of customers.



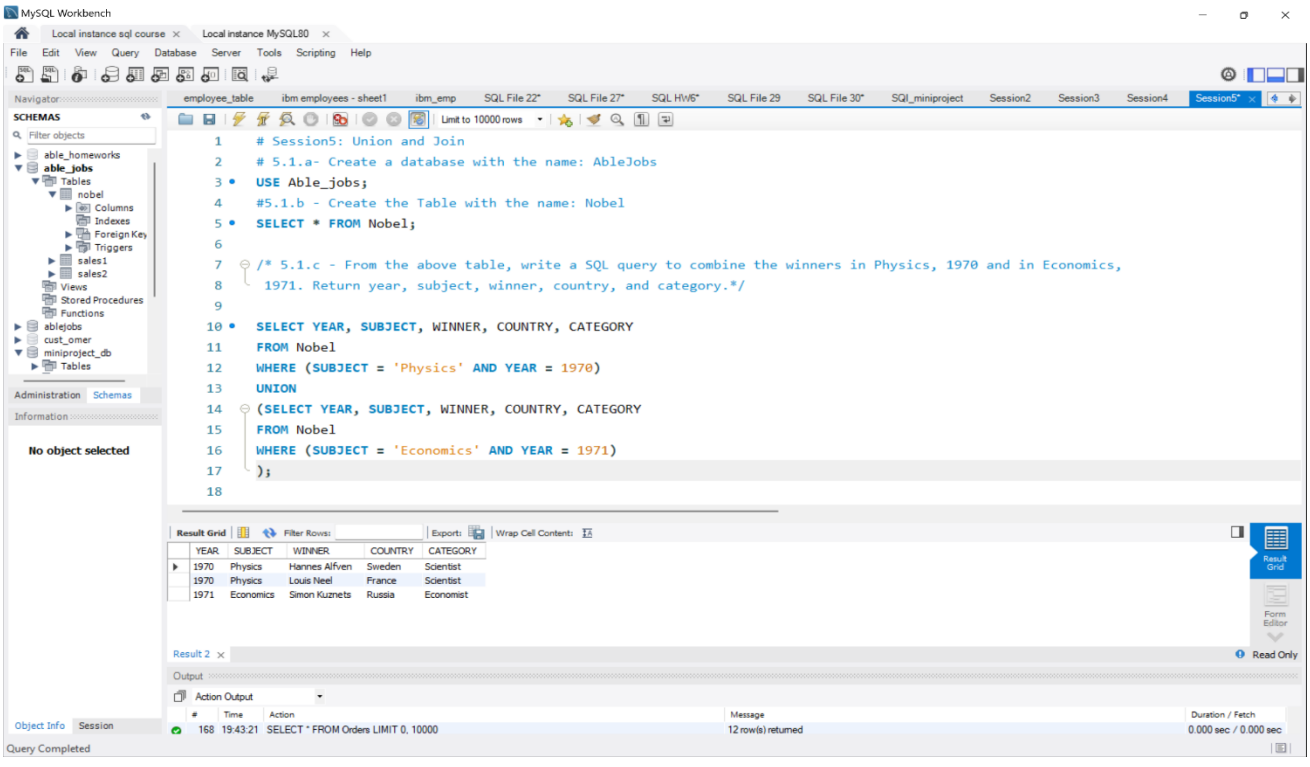
Session 5: Union and Join

1.

a. Create a database with the name: AbleJobs

b. Create the following Table with the name: Nobel

c. From the above table, write a SQL query to combine the winners in Physics, 1970 and in Economics, 1971. Return year, subject, winner, country, and category.



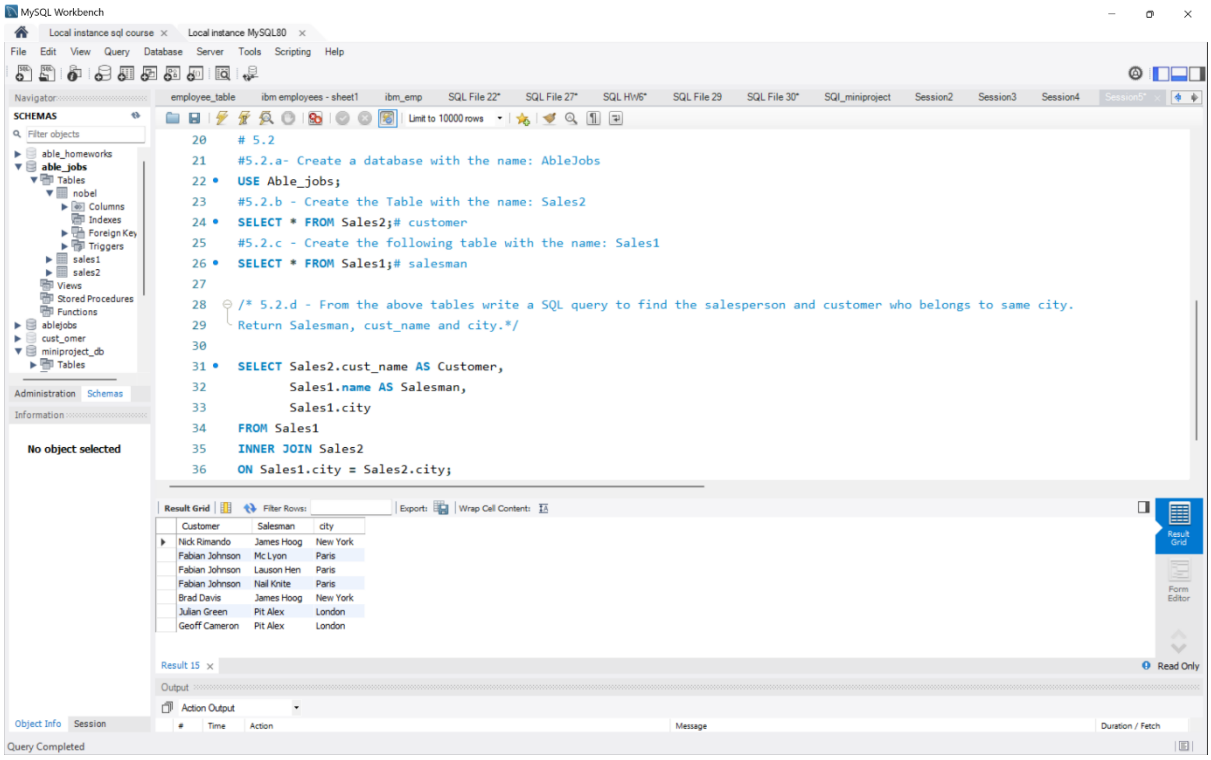
2.

a. Create a database with the name: AbleJobs

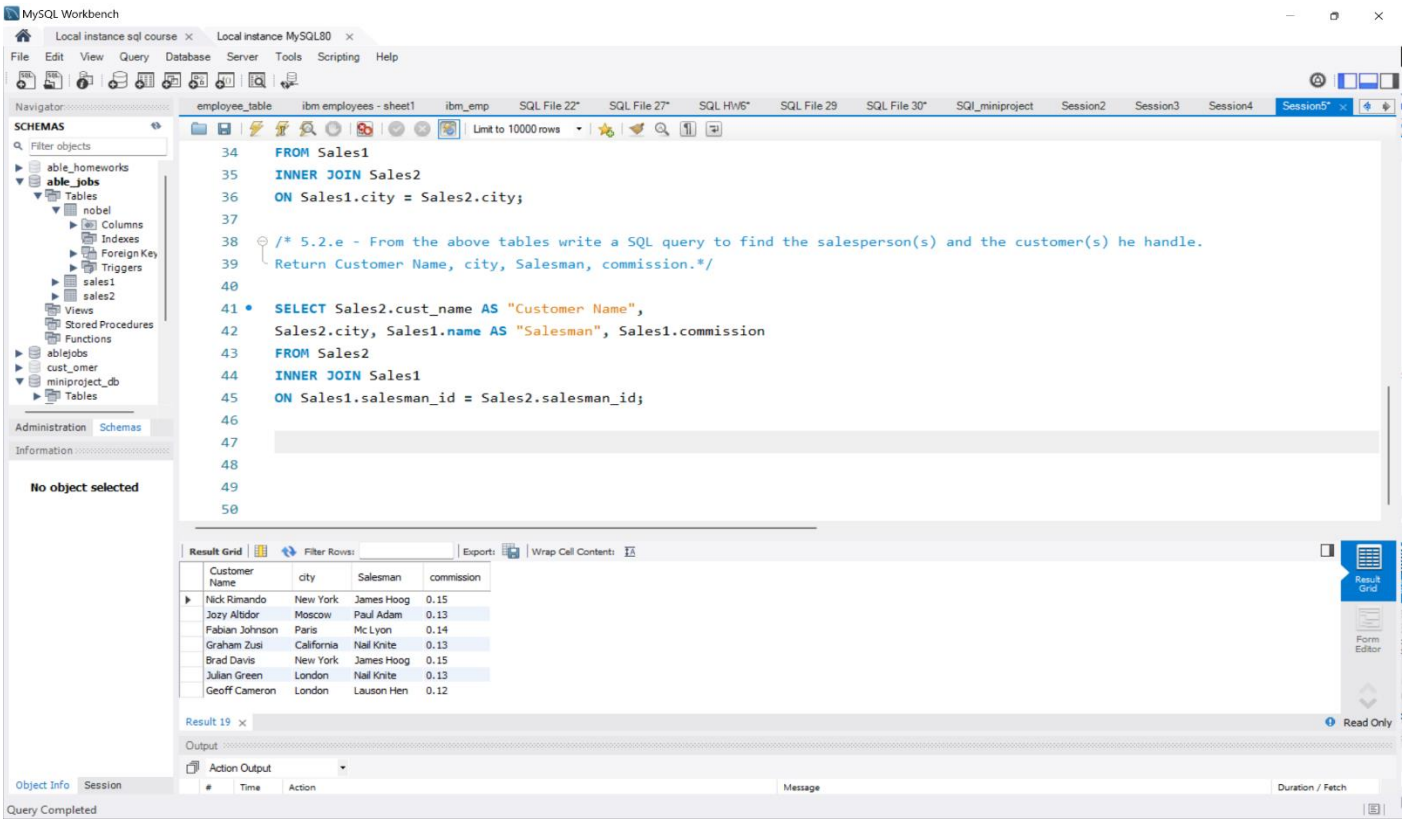
b. Create the following Table with the name: Sales2

c. Create the following table with the name: Sales1

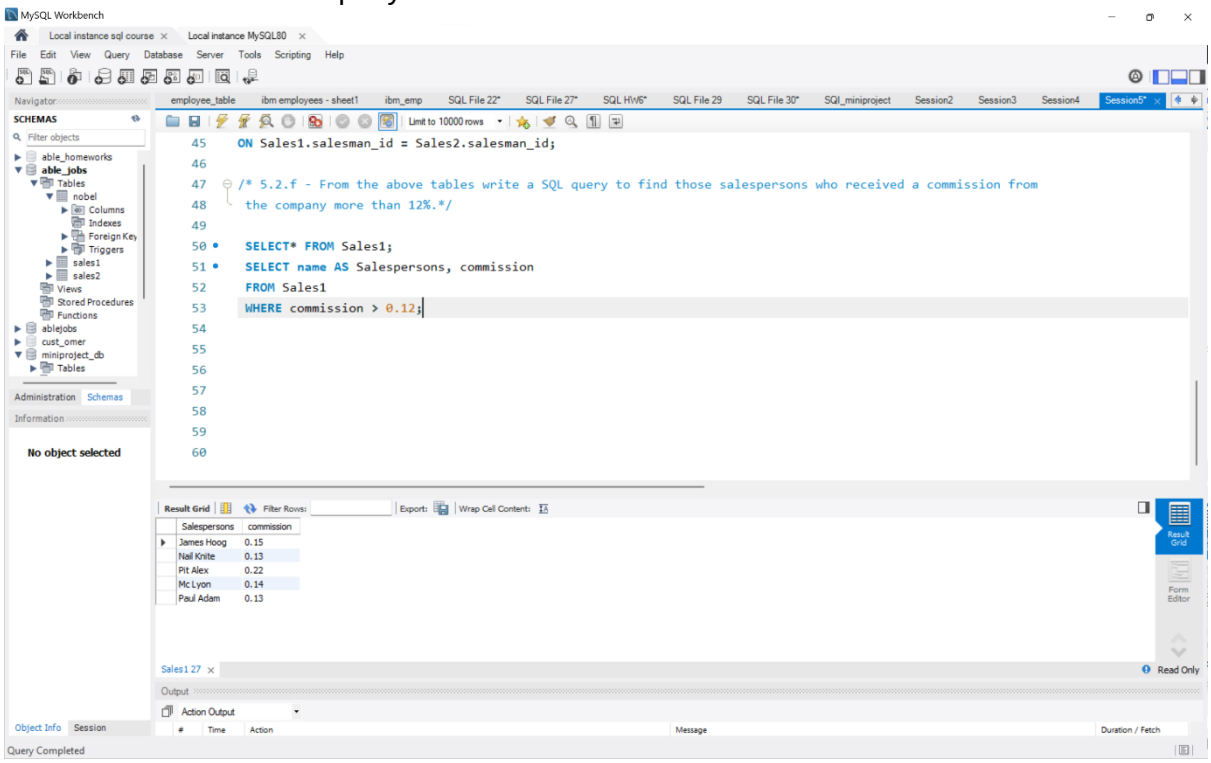
d. From the above tables write a SQL query to find the salesperson and customer who belongs to same city. Return Salesman, cust_name and city.



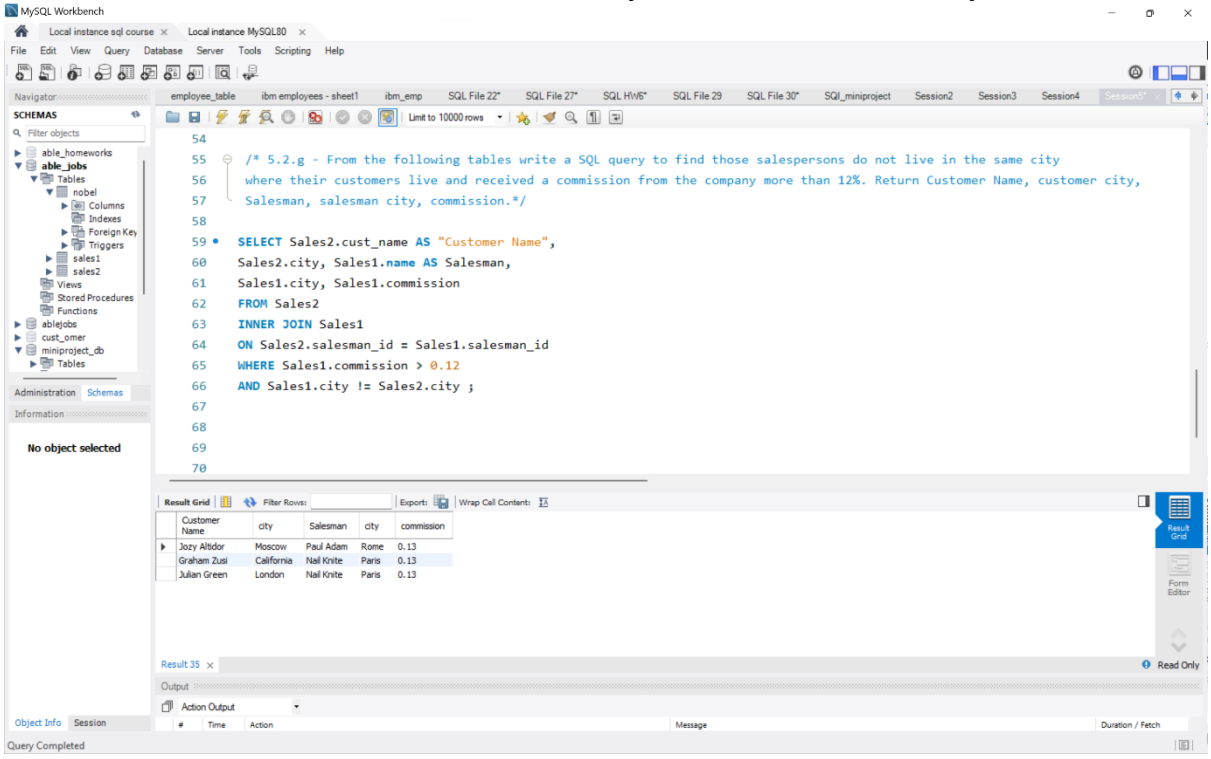
e. From the above tables write a SQL query to find the salesperson(s) and the customer(s) he handle. Return Customer Name, city, Salesman, commission.



f. From the above tables write a SQL query to find those salespersons who received a commission from the company more than 12%.

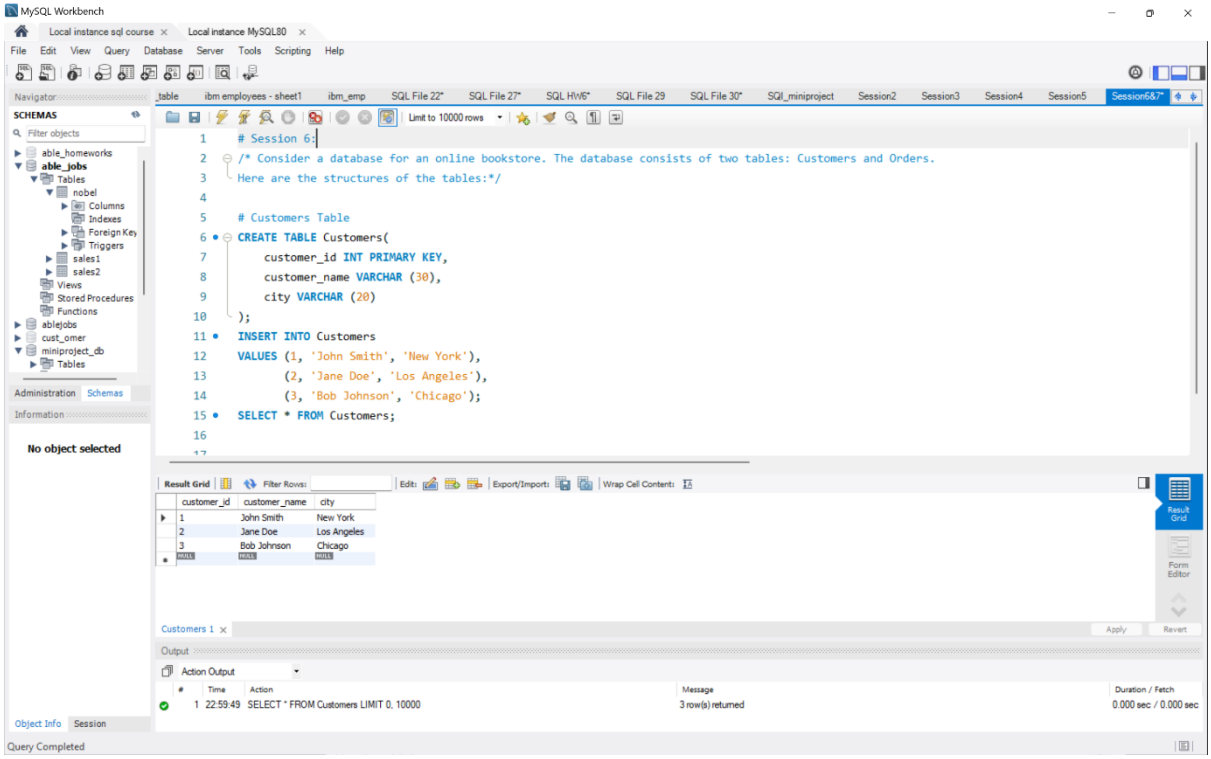


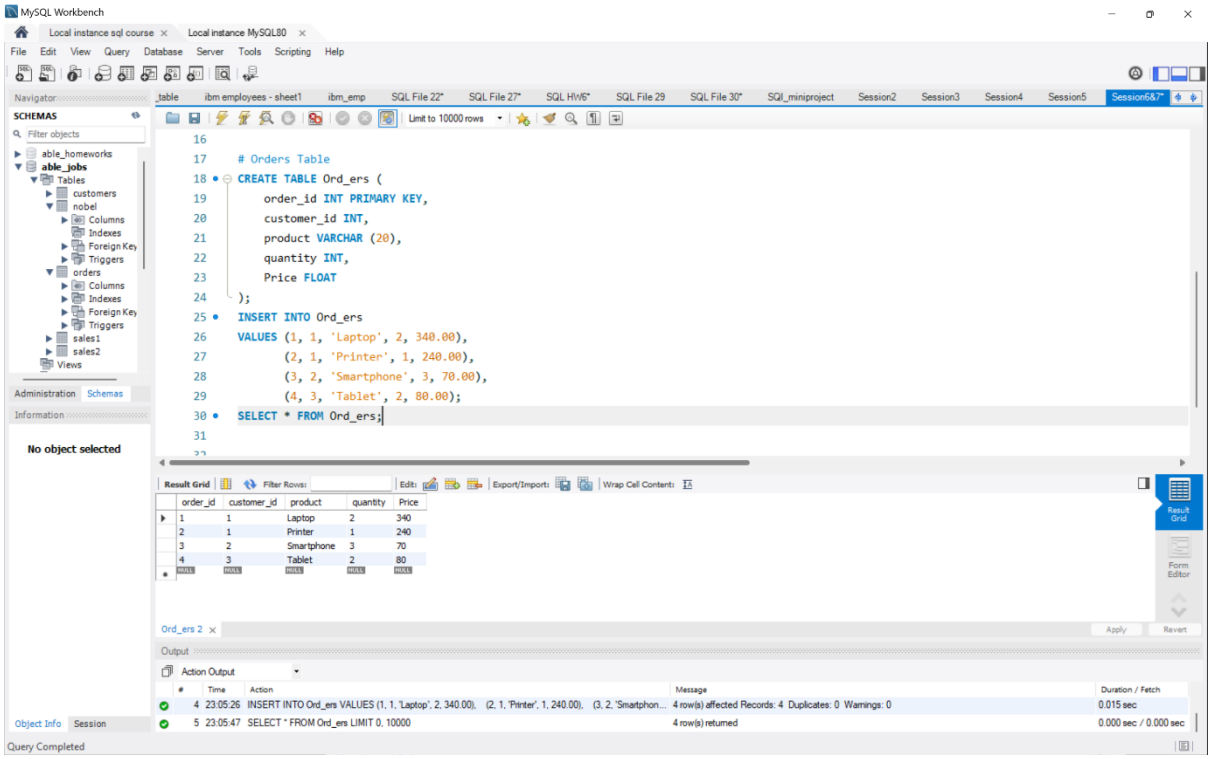
g. From the following tables write a SQL query to find those salespersons do not live in the same city where their customers live and received a commission from the company more than 12%. Return Customer Name, customer city, Salesman, salesman city, commission.



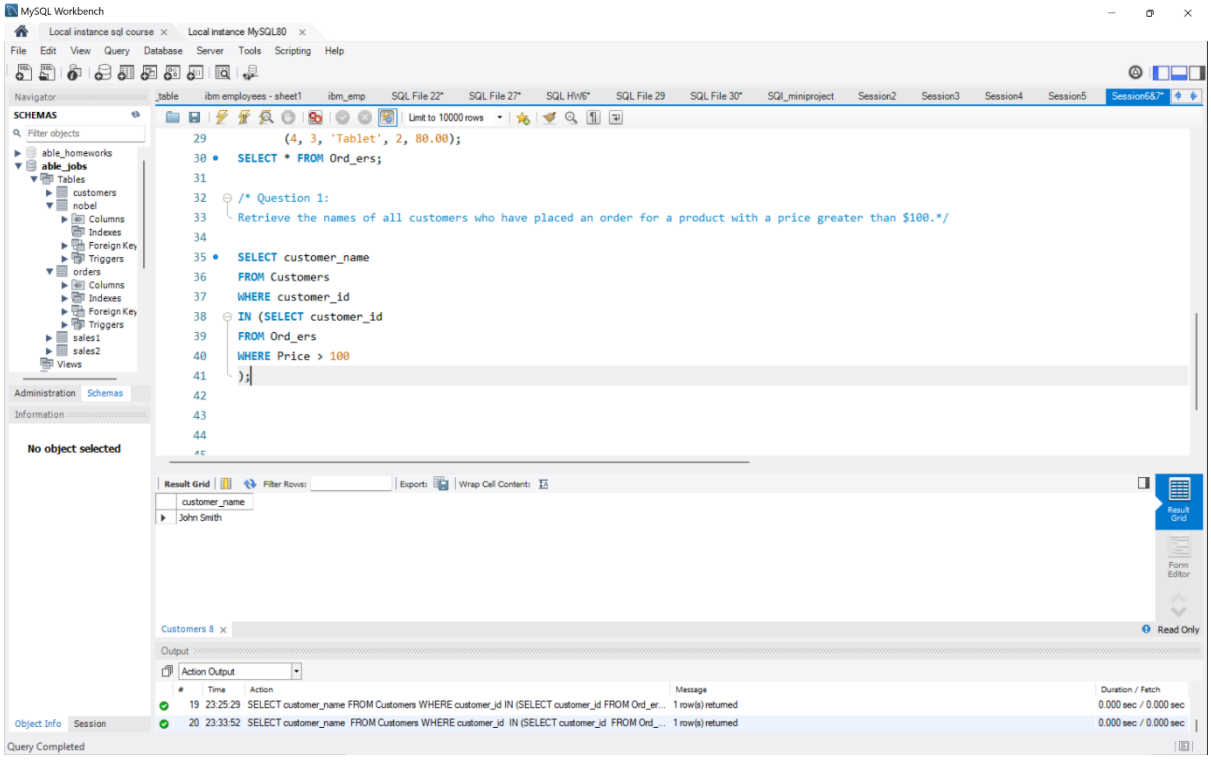
Session 6 & 7: Nested Queries & Normalization

Consider a database for an online bookstore. The database consists of two tables: Customers and Orders. Here are the structures of the tables:

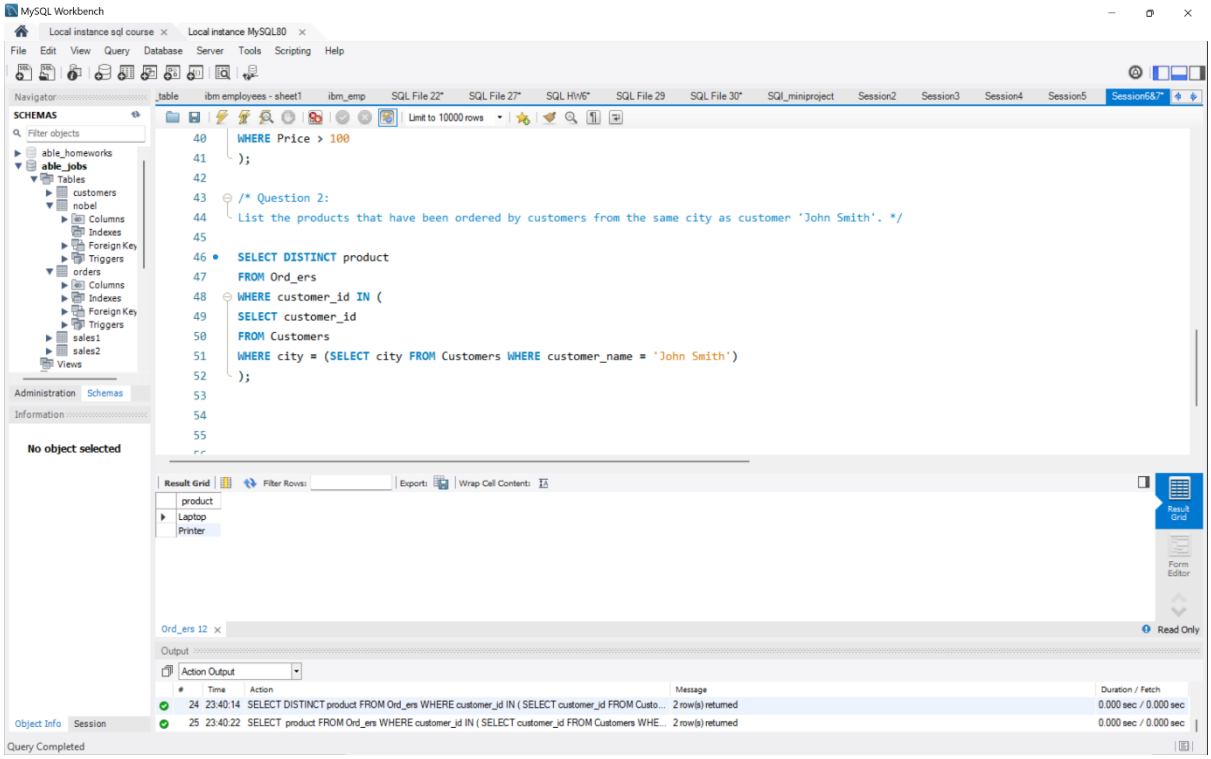




Question 1:
Retrieve the names of all customers who have placed an order for a product with a price greater than $100.



Question 2:
List the products that have been ordered by customers from the same city as customer 'John Smith'.



Question 3:
Find the order IDs and total order amounts for orders that contain at least one product with a quantity greater than 2.