Air Cargo Analysis

Course-end Project 2

Description

Air Cargo is an aviation company that provides air transportation services for passengers and freight. Air Cargo uses its aircraft to provide different services with the help of partnerships or alliances with other airlines. The company wants to prepare reports on regular passengers, busiest routes, ticket sales details, and other scenarios to improve the ease of travel and booking for customers.

Project Objective:

You, as a DBA expert, need to focus on identifying the regular customers to provide offers, analyze the busiest route which helps to increase the number of aircraft required and prepare an analysis to determine the ticket sales details. This will ensure that the company improves its operability and becomes more customer-centric and a favorable choice for air travel.

Note: You must download the dataset from the course resource section in the LMS and create the tables to perform the above objective.

Dataset description:

Customer: Contains the information of customers

- customer_id – ID of the customer
- first_name – First name of the customer
- last_name – Last name of the customer
- date_of_birth – Date of birth of the customer
- gender – Gender of the customer

passengers_on_flights: Contains information about the travel details

- aircraft_id – ID of each aircraft in a brand
- route_id – Route ID of from and to location
- customer_id – ID of the customer
- depart – Departure place from the airport
- arrival – Arrival place in the airport
- seat_num – Unique seat number for each passenger
- class_id – ID of travel class
- travel_date – Travel date of each passenger
- flight_num – Specific flight number for each route

ticket_details: Contains information about the ticket details

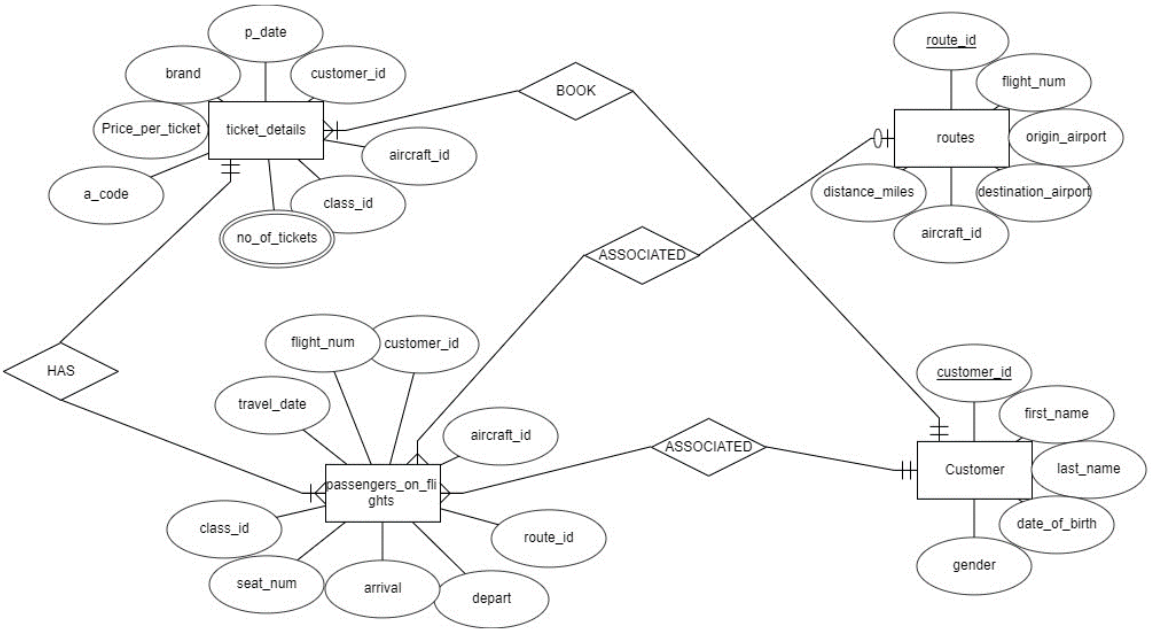
- p_date – Ticket purchase date
- customer_id – ID of the customer
- aircraft_id – ID of each aircraft in a brand
- class_id – ID of travel class
- no_of_tickets – Number of tickets purchased
- a_code – Code of each airport
- price_per_ticket – Price of a ticket
- brand – Aviation service provider for each aircraft

routes: Contains information about the route details

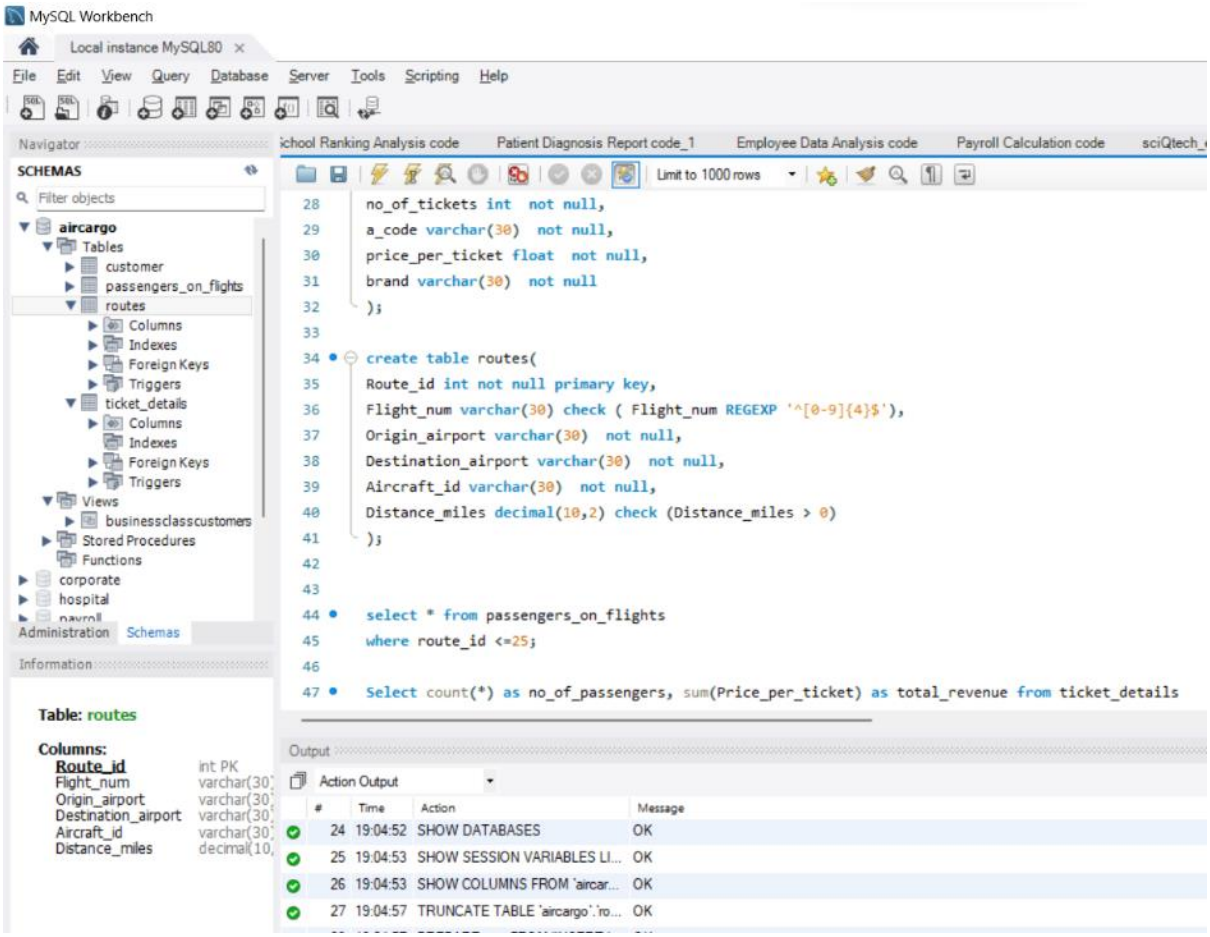
- Route_id – Route ID of from and to location
- Flight_num – Specific fight number for each route
- Origin_airport – Departure location
- Destination_airport – Arrival location
- Aircraft_id – ID of each aircraft in a brand
- Distance_miles – Distance between departure and arrival location

Following operations should be performed:

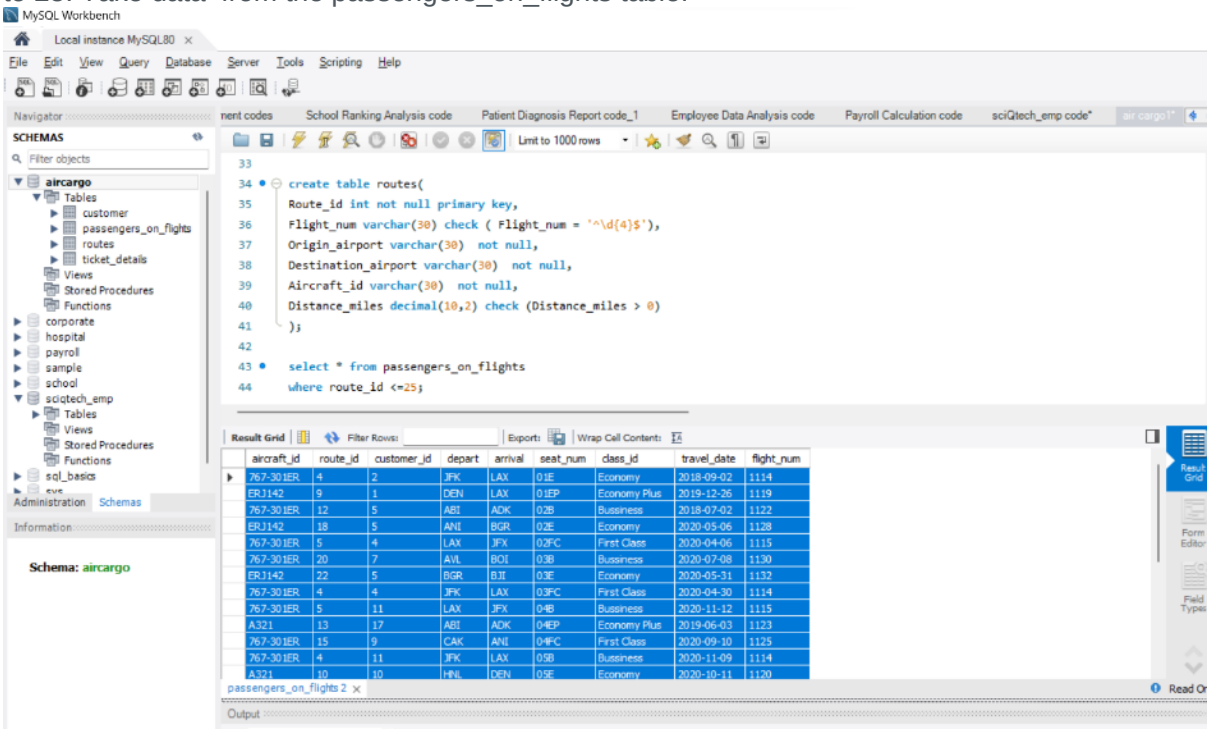
1. Create an ER diagram for the given airlines database.



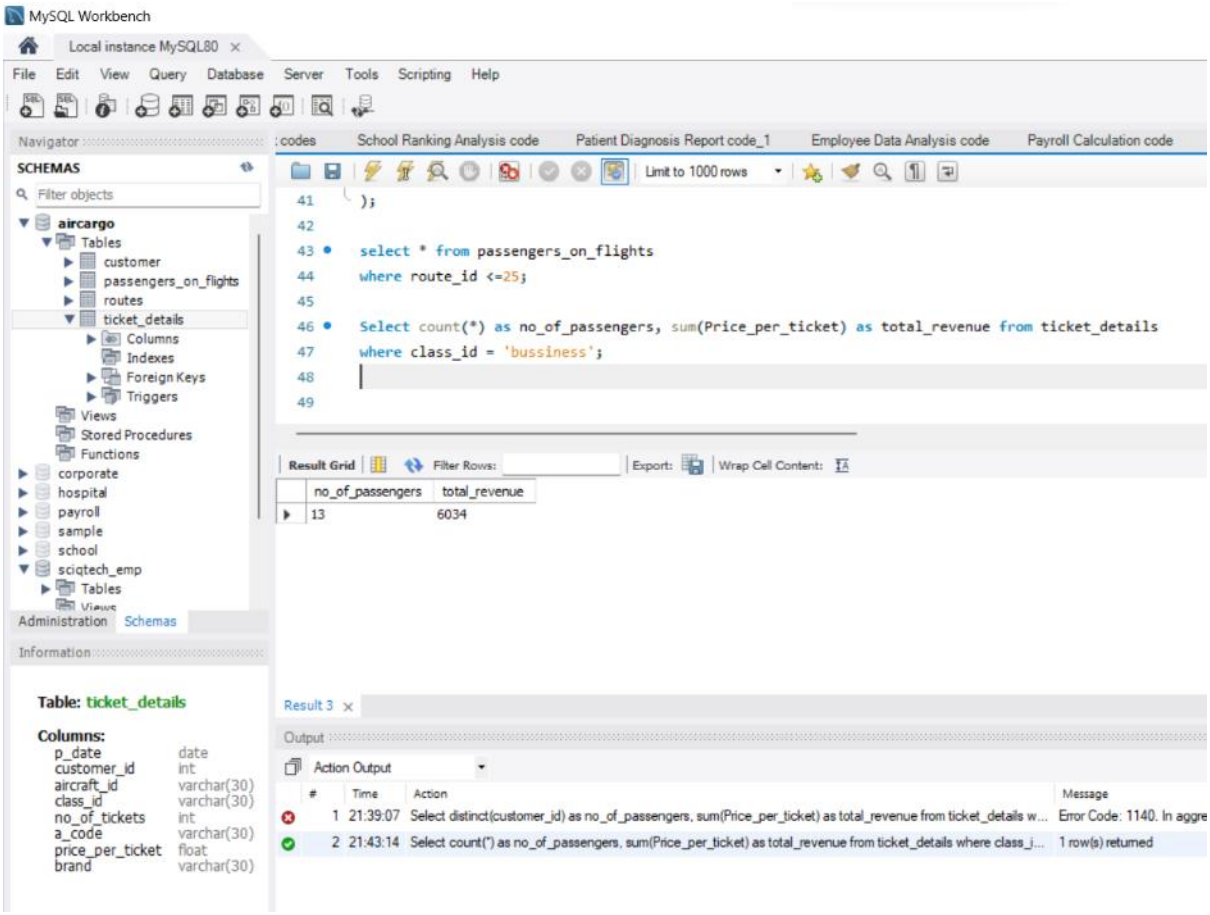
2. Write a query to create route_details table using suitable data types for the fields, such as route_id, flight_num, origin_airport, destination_airport, aircraft_id, and distance_miles. Implement the check constraint for the flight number and unique constraint for the route_id fields. Also, make sure that the distance miles field is greater than 0.



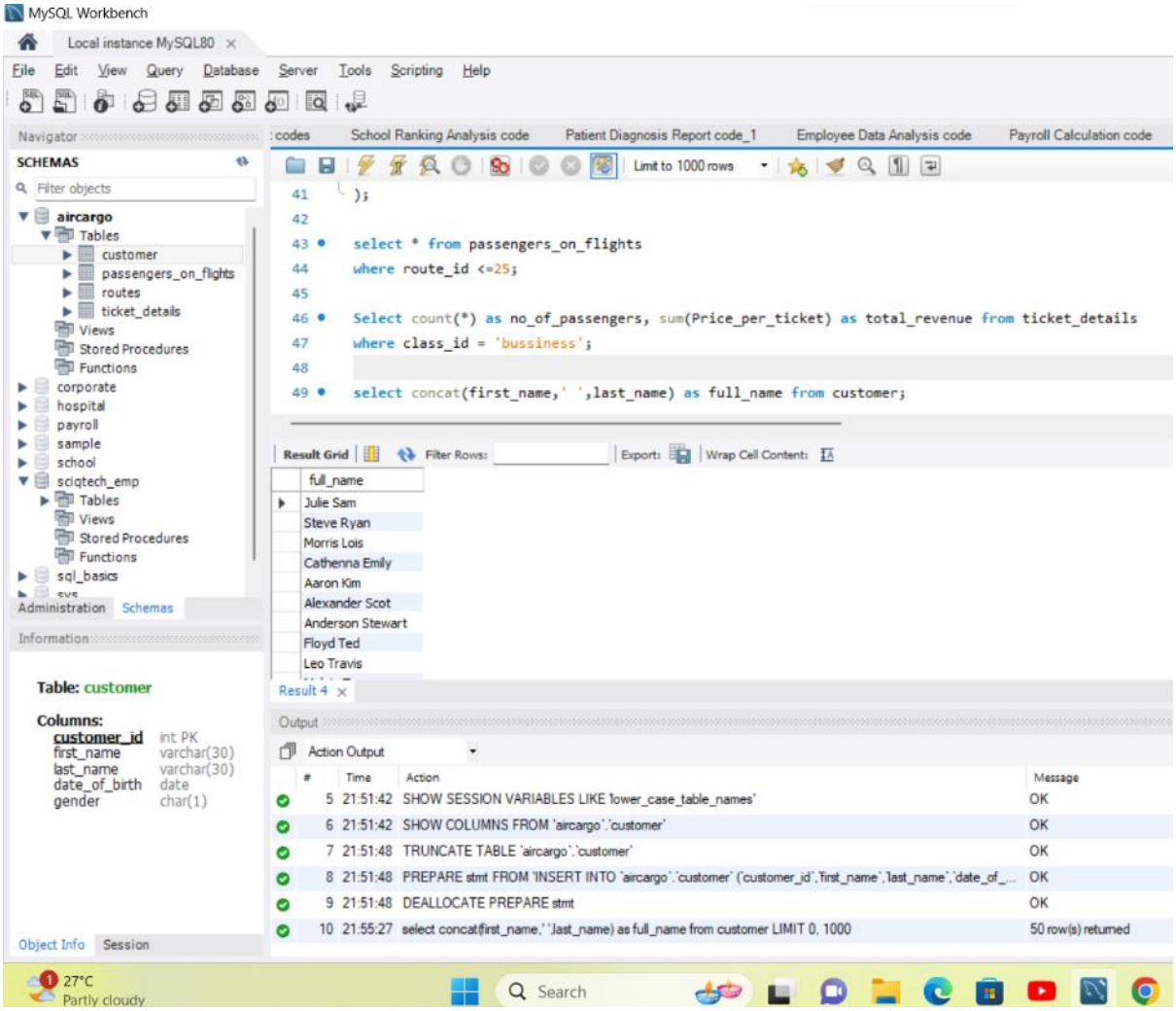
3. Write a query to display all the passengers (customers) who have travelled in routes 01 to 25. Take data from the passengers_on_flights table.



4. Write a query to identify the number of passengers and total revenue in business class from the ticket_details table.

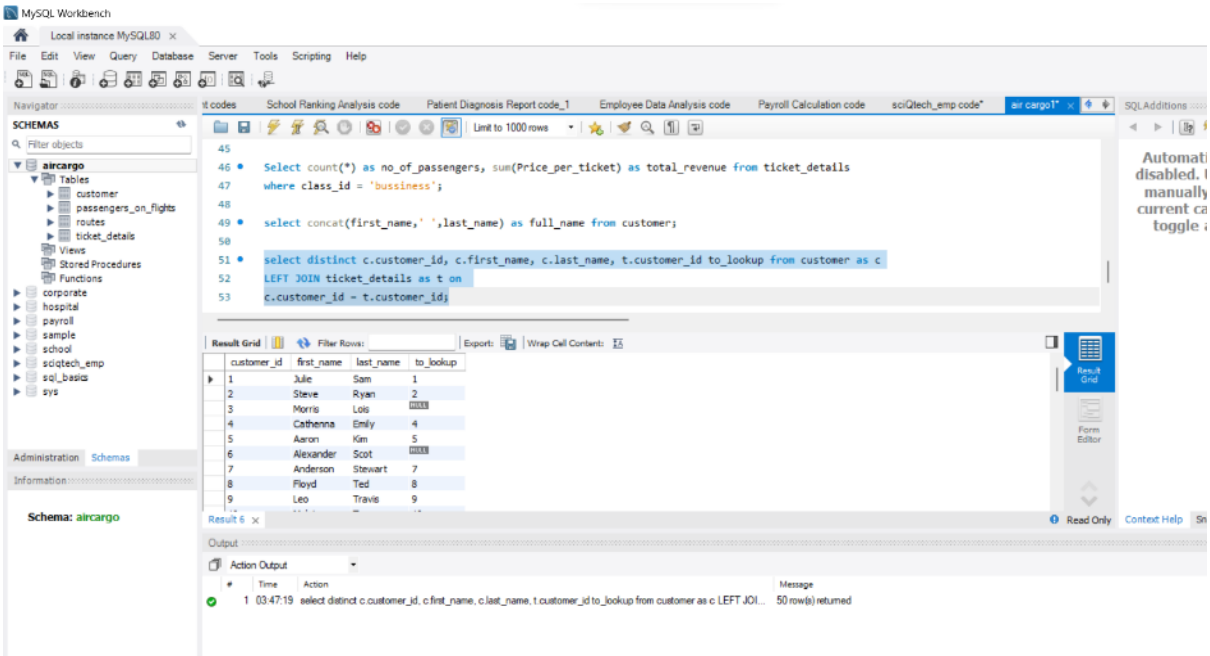


5. Write a query to display the full name of the customer by extracting the first name and last name from the customer table.

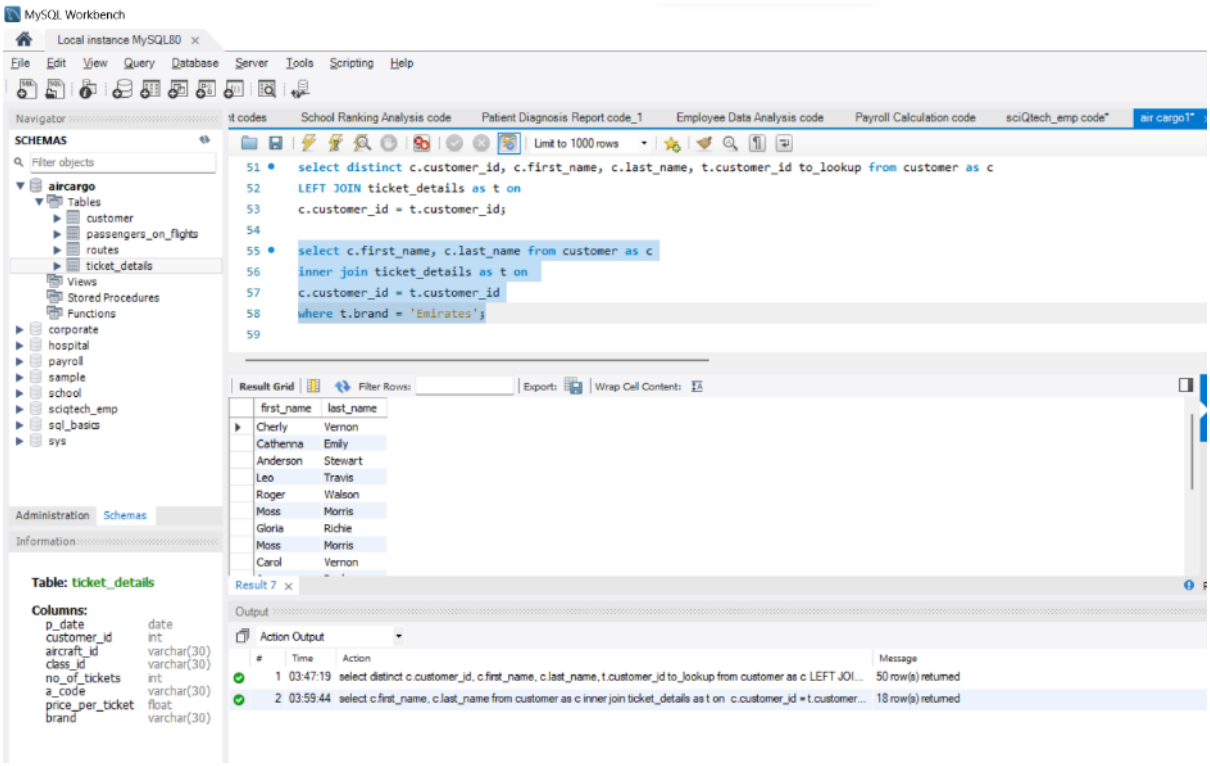


6. Write a query to extract the customers who have registered and booked a ticket. Use data from the customer and ticket_details tables.

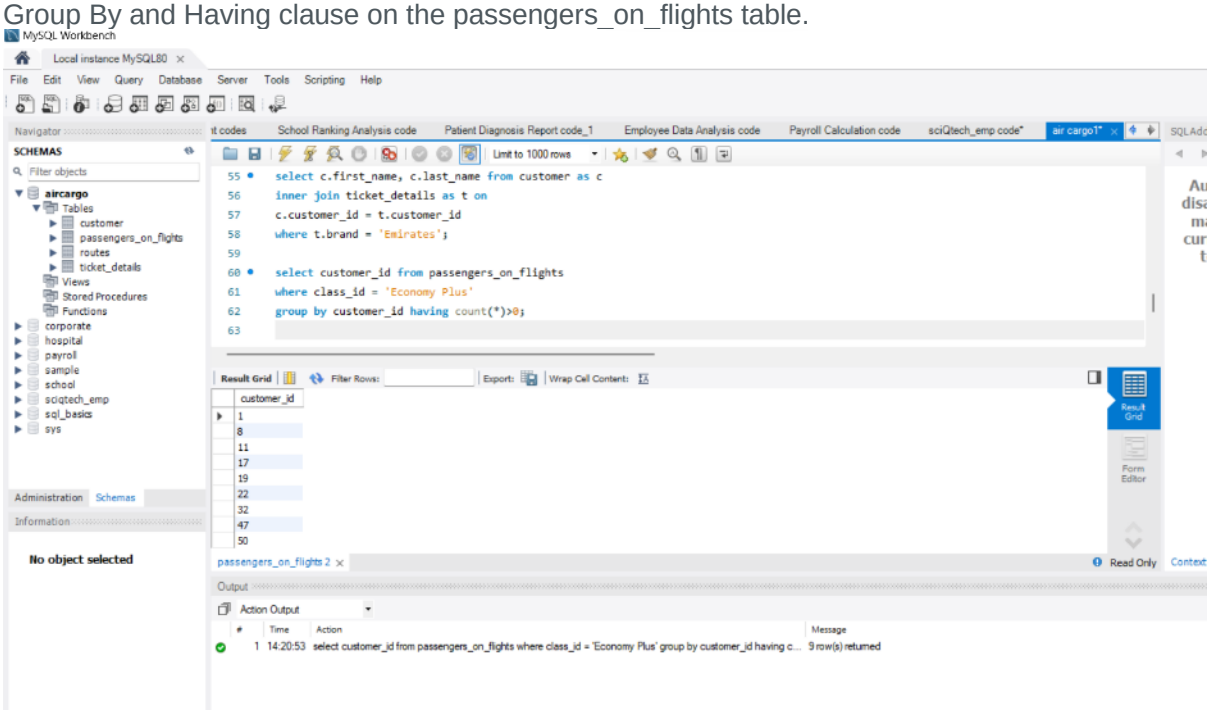
   The result tables has 17 null values in vlookup function.



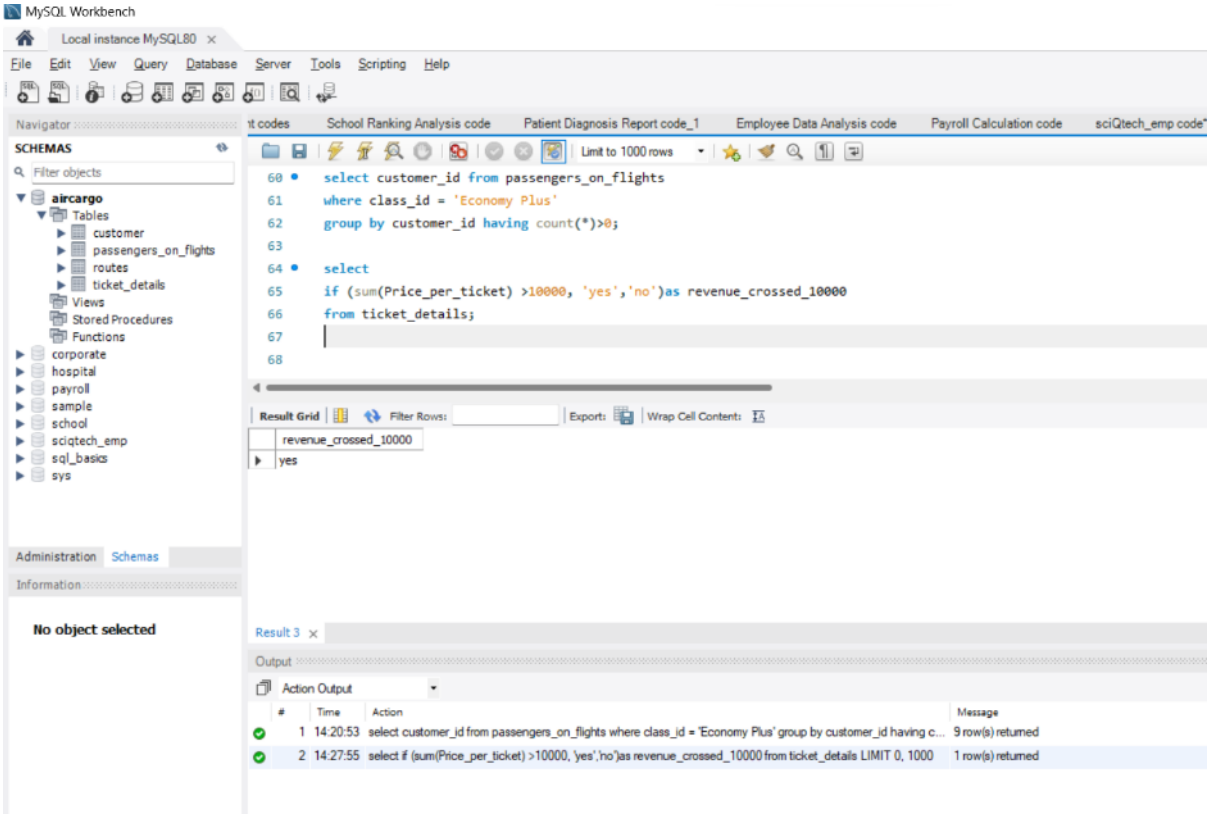
7. Write a query to identify the customer's first name and last name based on their customer ID and brand (Emirates) from the ticket_details table.



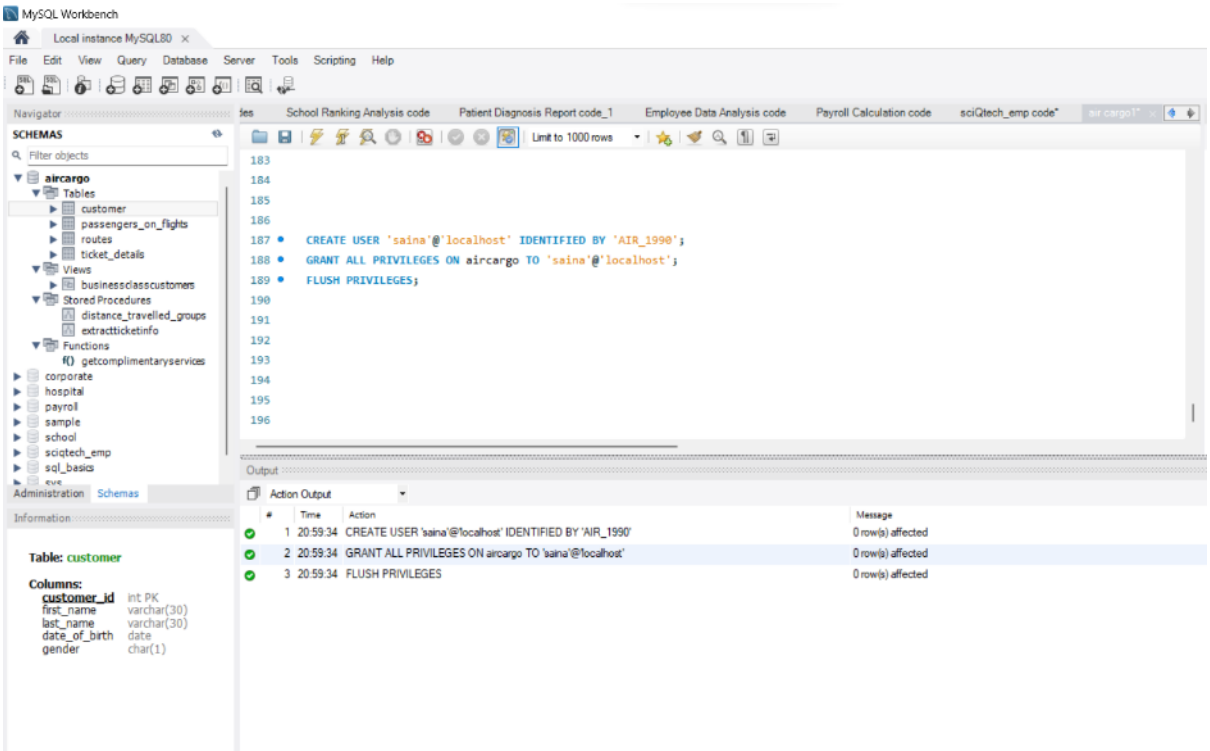
8. Write a query to identify the customers who have travelled by *Economy Plus* class using Group By and Having clause on the passengers_on_flights table.



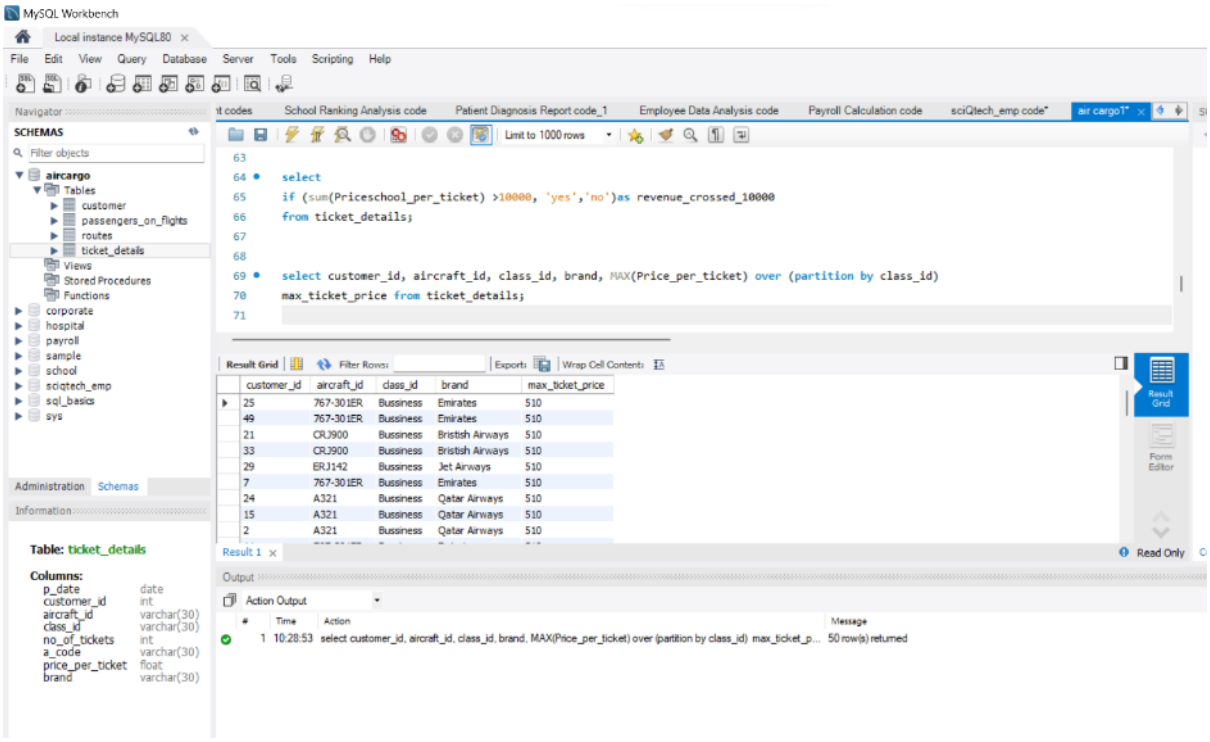
9. Write a query to identify whether the revenue has crossed 10000 using the IF clause on the ticket_details table.



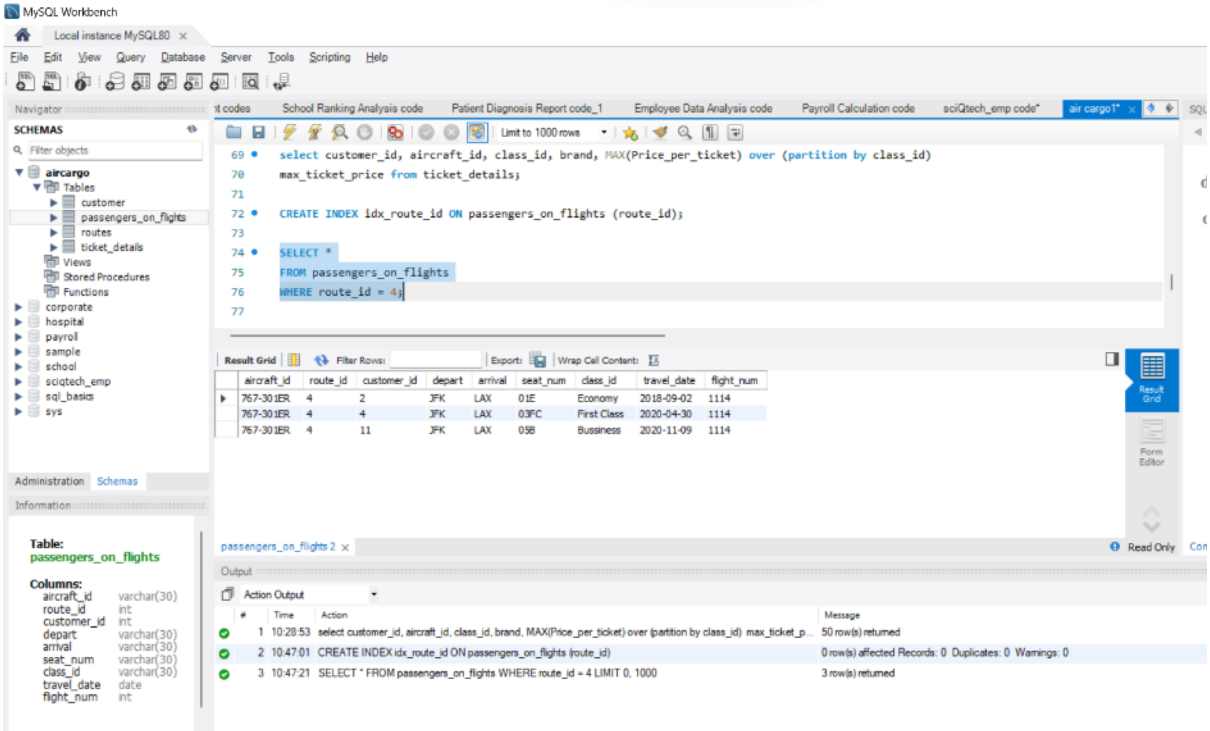
10. Write a query to create and grant access to a new user to perform operations on a database.



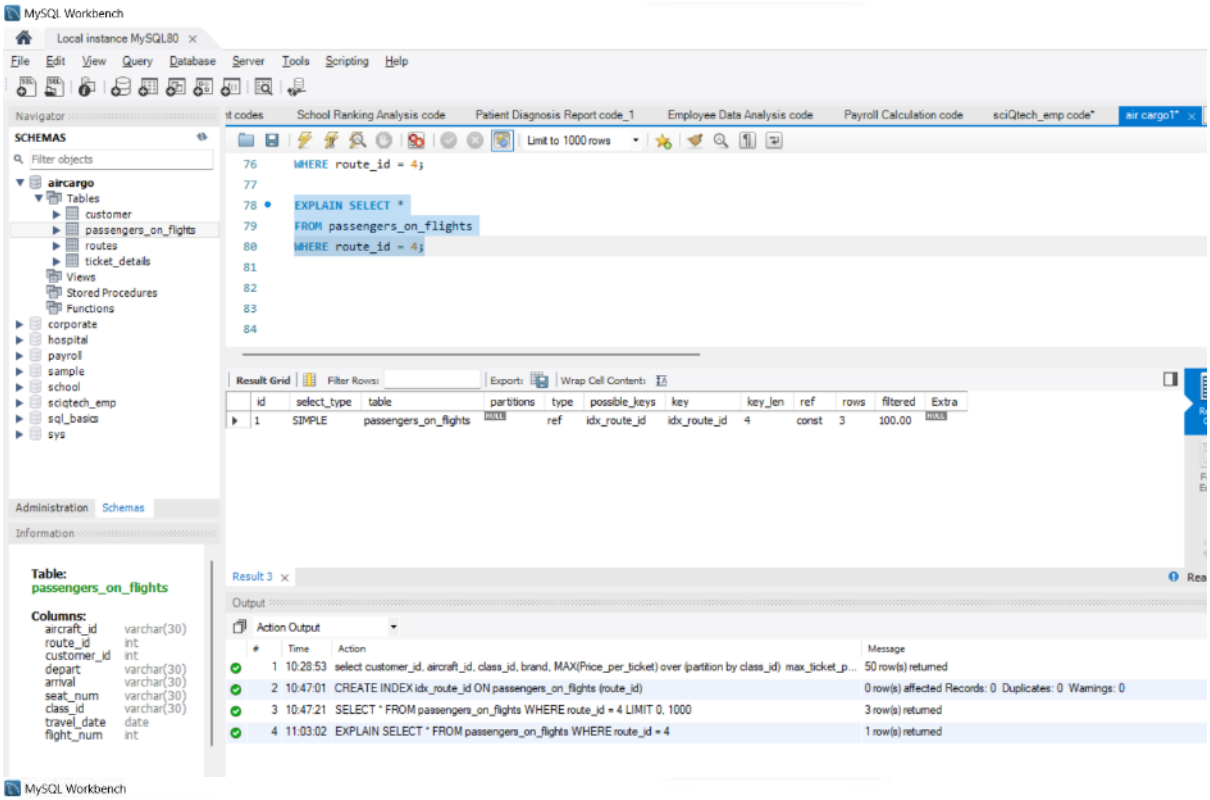
11. Write a query to find the maximum ticket price for each class using window functions on the ticket_details table.

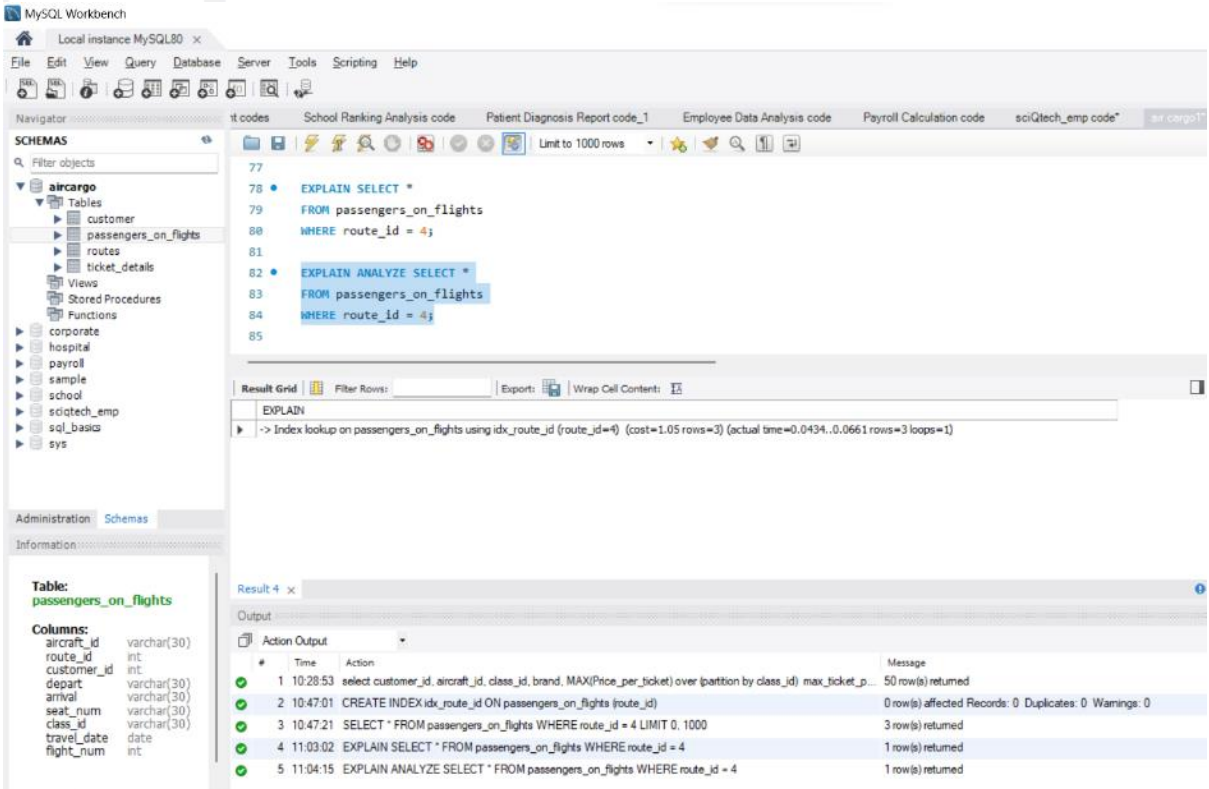


12. Write a query to extract the passengers whose route ID is 4 by improving the speed and performance of the passengers_on_flights table.

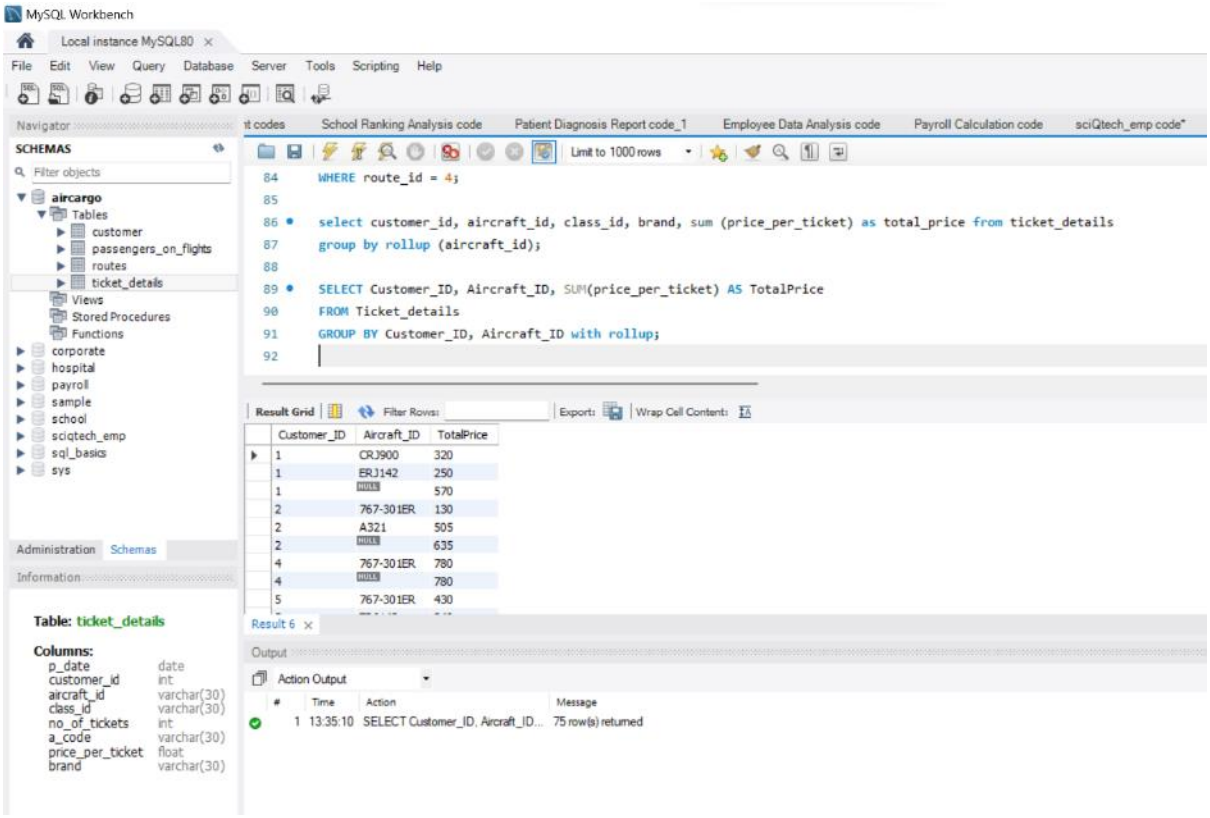


13. For the route ID 4, write a query to view the execution plan of the passengers_on_flights table.

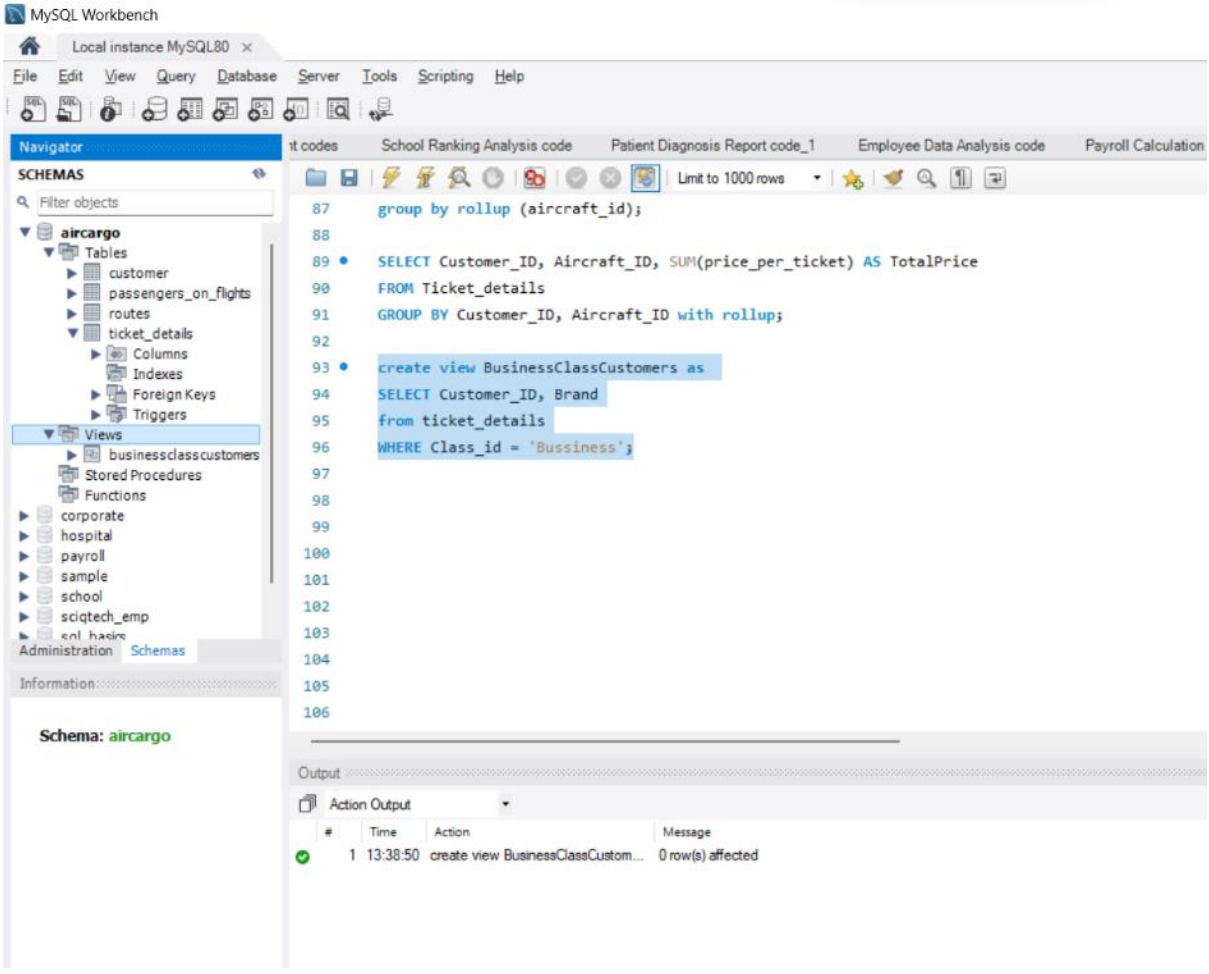




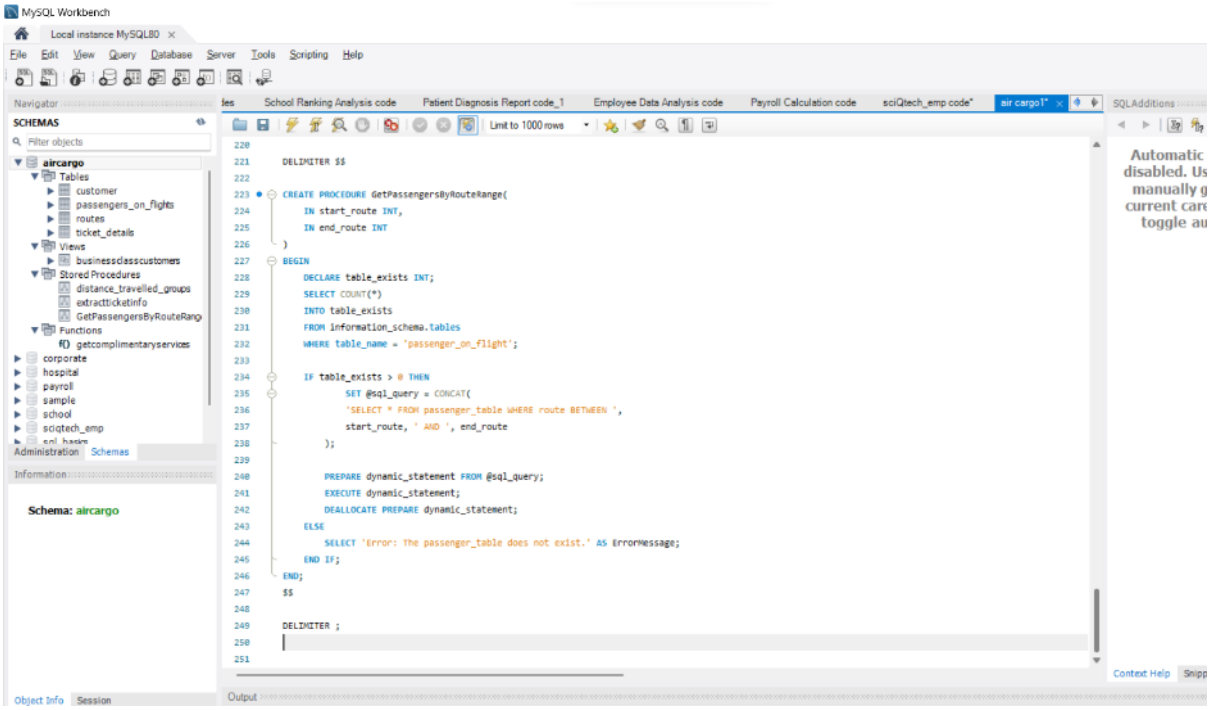
14. Write a query to calculate the total price of all tickets booked by a customer across different aircraft IDs using rollup function.



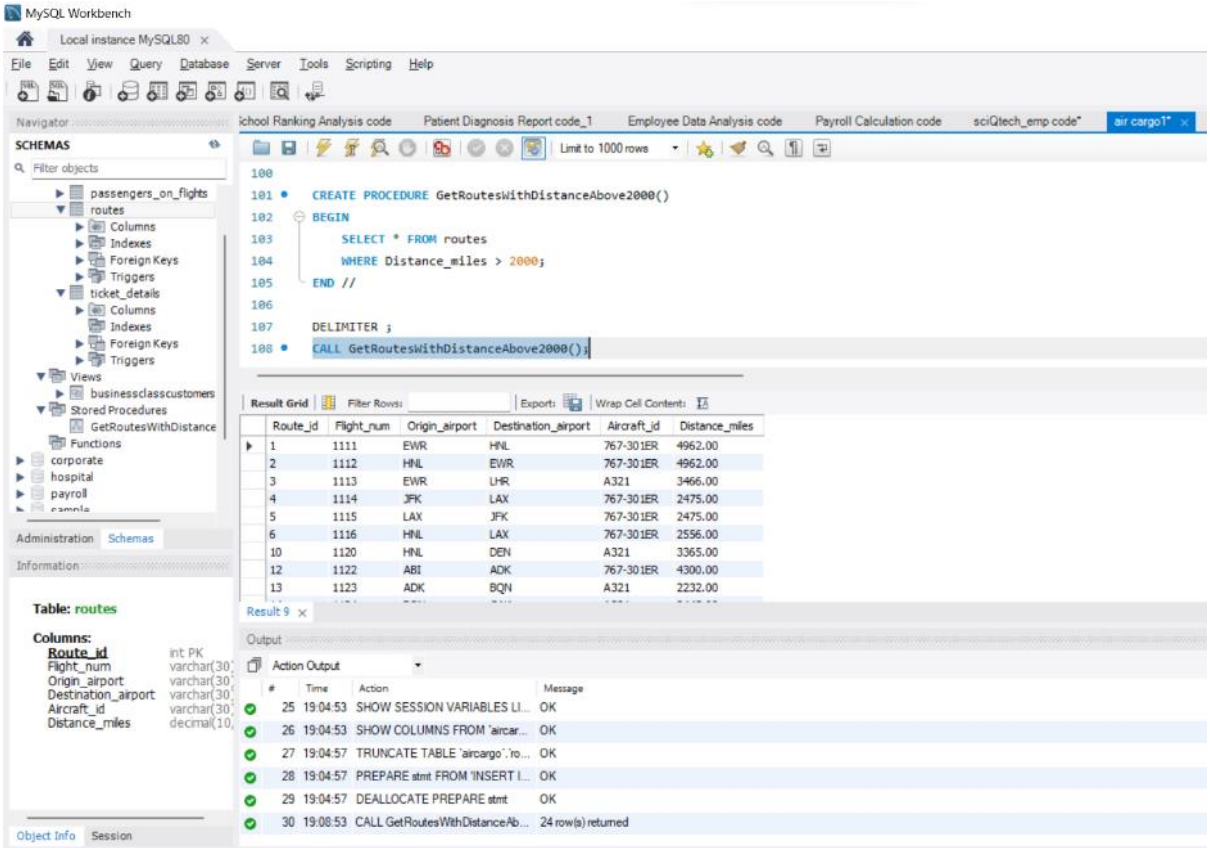
15. Write a query to create a view with only business class customers along with the brand of airlines.



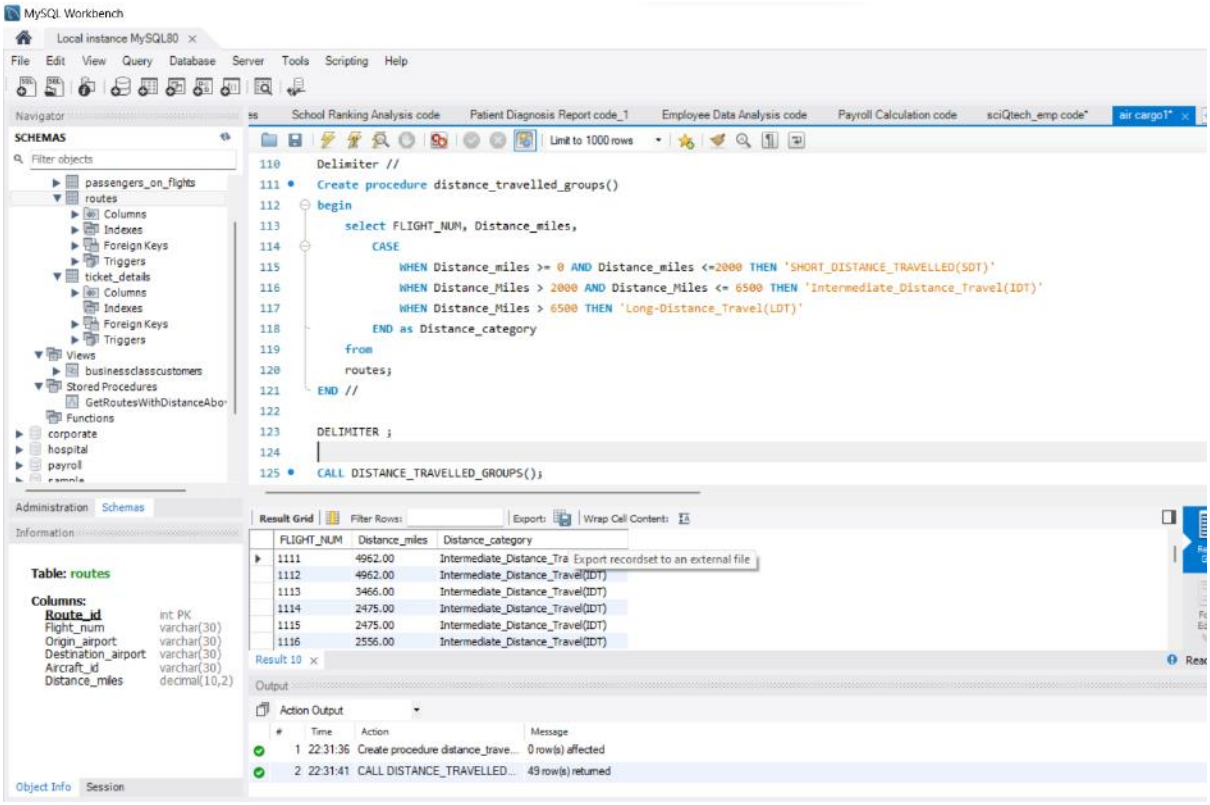
16. Write a query to create a stored procedure to get the details of all passengers flying between a range of routes defined in run time. Also, return an error message if the table doesn't exist.



17. Write a query to create a stored procedure that extracts all the details from the routes table where the travelled distance is more than 2000 miles.



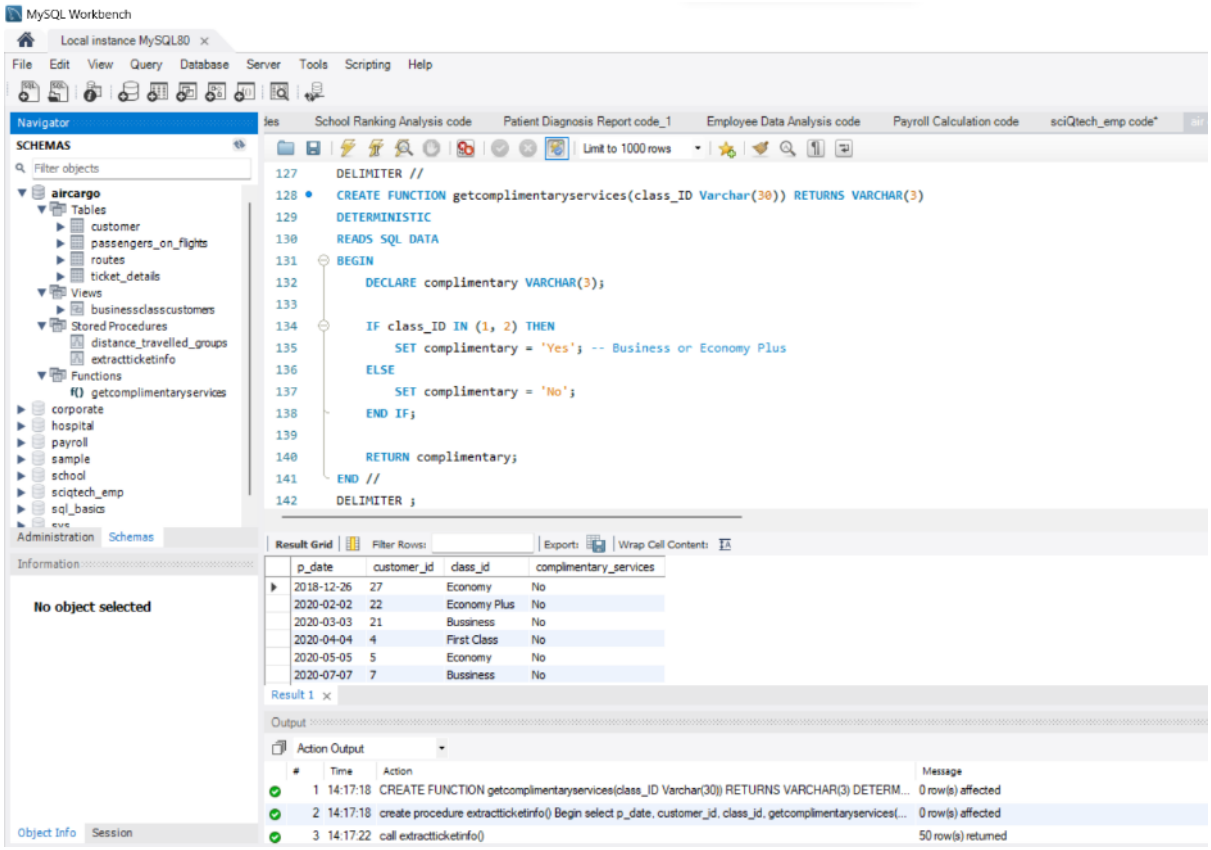
18. Write a query to create a stored procedure that groups the distance travelled by each flight into three categories. The categories are, short distance travel (SDT) for >=0 AND <= 2000 miles, intermediate distance travel (IDT) for >2000 AND <=6500, and long-distance travel (LDT) for >6500.

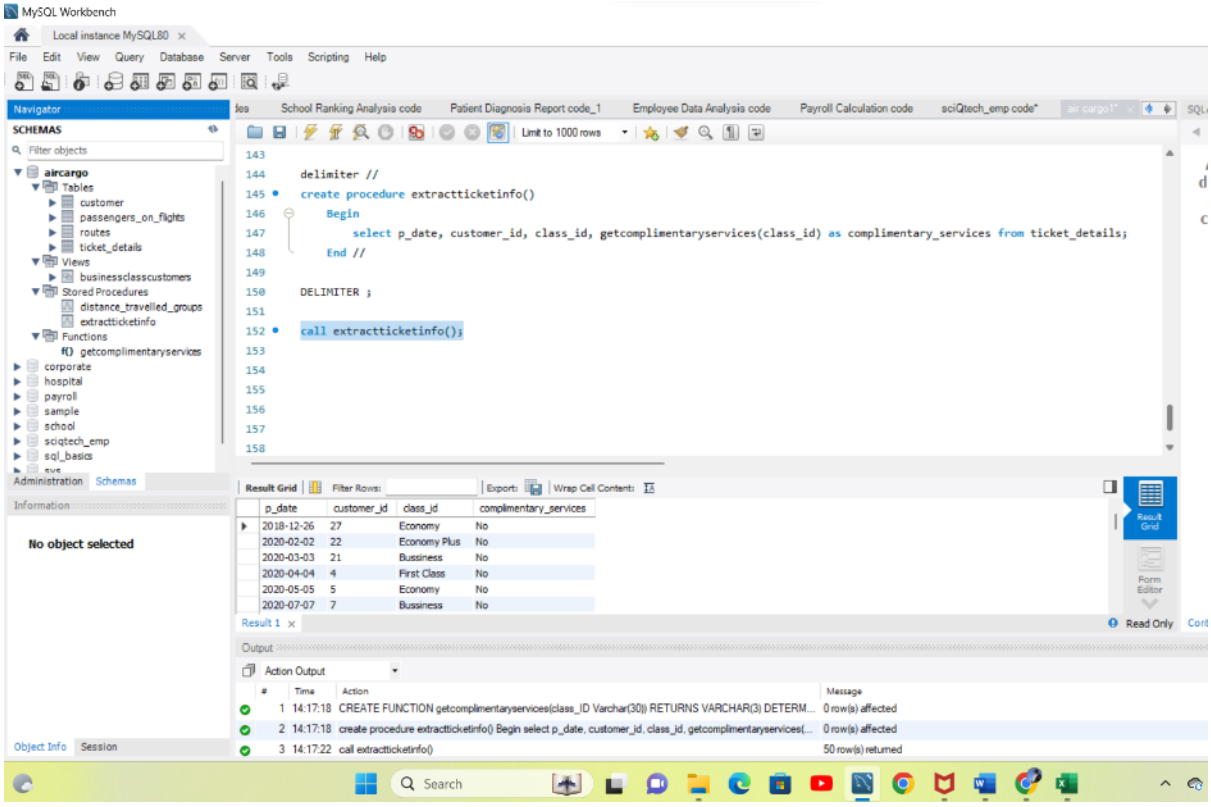


19. Write a query to extract ticket purchase date, customer ID, class ID and specify if the complimentary services are provided for the specific class using a stored function in stored procedure on the ticket_details table.

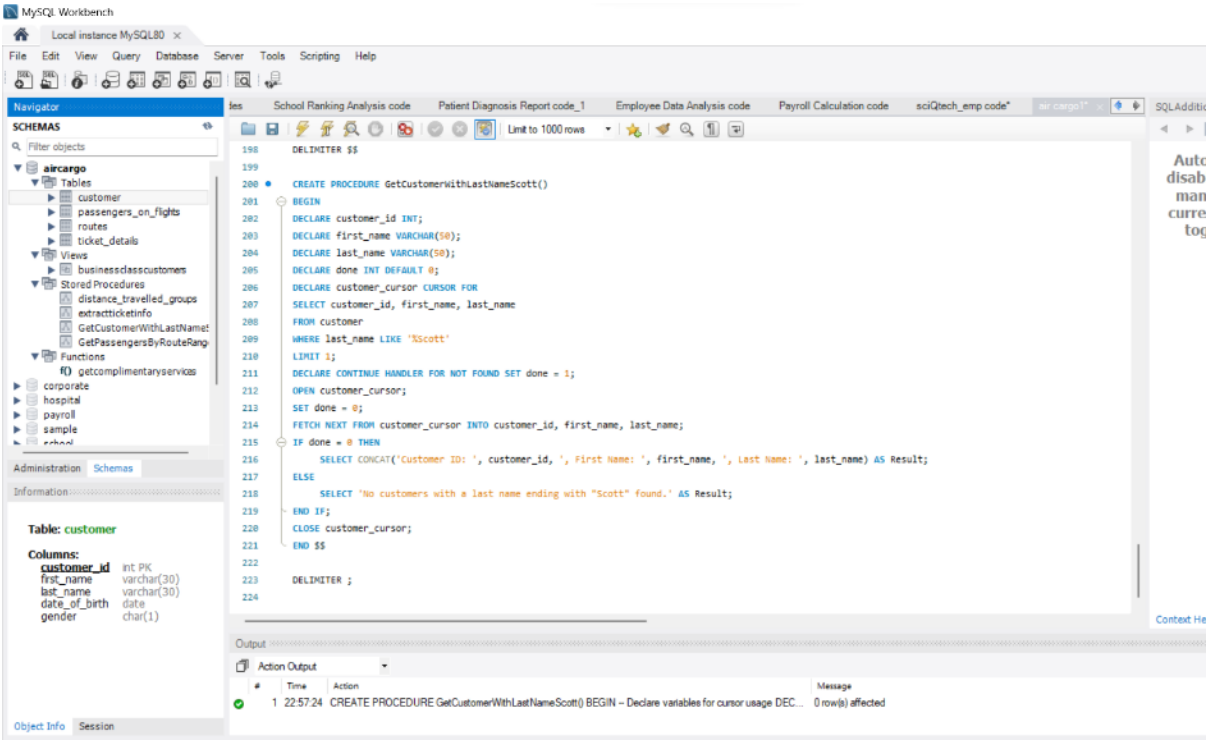
Condition:

- If the class is *Business* and *Economy Plus,* then complimentary services are given as *Yes,* else it is *No*





20. Write a query to extract the first record of the customer whose last name ends with Scott using a cursor from the customer table.


MySQL Workbench
Local instance MySQL80
File Edit View Query Database Server Tools Scripting Help
Navigator
SCHEMAS
Filter objects
aircargo
Tables
customer
passengers_on_flights
routes
ticket_details
Views
businessclasscustomers
Stored Procedures
distance_travelled_groups
extractticketinfo
GetCustomerWithLastNames
GetPassengersByRouteRang
Functions
getcomplimentaryservices
corporate
hospital
payroll
sample
Administration Schemas
Information
Table: customer
Columns:
customer_id int PK
first_name varchar(30)
last_name varchar(30)
date_of_birth date
gender char(1)
Object Info Session
School Ranking Analysis code
Patient Diagnosis Report code_1
Employee Data Analysis code
Payroll Calculation code
sciQtech_emp code*
Limit to 1000 rows
198 DELIMITER $$
199
200 CREATE PROCEDURE GetCustomerWithLastNameScott()
201 BEGIN
202 DECLARE customer_id INT;
203 DECLARE first_name VARCHAR(50);
204 DECLARE last_name VARCHAR(50);
205 DECLARE done INT DEFAULT 0;
206 DECLARE customer_cursor CURSOR FOR
207 SELECT customer_id, first_name, last_name
208 FROM customer
209 WHERE last_name LIKE '%Scott'
210 LIMIT 1;
211 DECLARE CONTINUE HANDLER FOR NOT FOUND SET done = 1;
212 OPEN customer_cursor;
213 SET done = 0;
214 FETCH NEXT FROM customer_cursor INTO customer_id, first_name, last_name;
215 IF done = 0 THEN
216 SELECT CONCAT('Customer ID: ', customer_id, ', First Name: ', first_name, ', Last Name: ', last_name) AS Result;
217 ELSE
218 SELECT 'No customers with a last name ending with "Scott" found.' AS Result;
219 END IF;
220 CLOSE customer_cursor;
221 END $$
222
223 DELIMITER ;
224
Output
Action Output
Time Action Message
1 22:57:24 CREATE PROCEDURE GetCustomerWithLastNameScott() BEGIN -- Declare variables for cursor usage DEC... 0 row(s) affected