***SQL***

***PROJECT-2***

***TOPIC: "Leveraging SQL for Air Cargo Optimization: A Comprehensive Analysis"***

WHAT IS SQL?

SQL (Structured Query Language) is a way to communicate with a relational database that lets you define, query, modify, and control the data. Using SQL syntax, you can construct a statement that extracts records according to criteria you specify.

***Why is sql important?***

It allows us to communicate with relational databases and gather information.

Course-End Project: Air Cargo Analysis Problem Statement Scenario:

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 favourable choice for air travel.

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

Write a query to create a 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.

CODE;

create database air\_cargo;

use air\_cargo;

create table route\_detail(

route\_id int auto\_increment primary key,

flight\_num varchar(20),

origin\_airport varchar (100),

destination\_airport varchar(100),

aircraft\_id varchar(20) ,

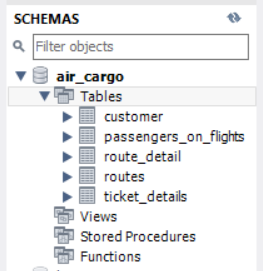
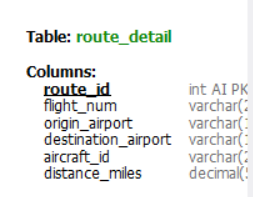
distance\_miles decimal(5,2),

constraint chk\_flight\_num check (flight\_num = '^[A-Z]{2,3}\d{1,4}$'),

constraint uq\_route\_id unique (route\_id),

constraint distance\_miles check(distance\_miles > 0));

OUTPUT;

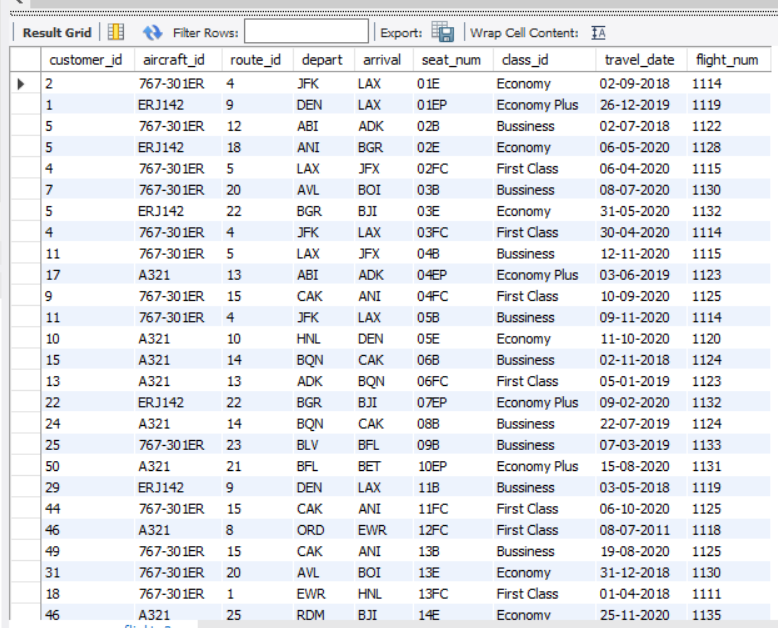
Write a query to display all the passengers (customers) who have travelled in routes 01 to 25. Take data from the passengers\_on\_flights table.

CODE;

select \* from passengers\_on\_flights

where route\_id between 1 and 25;

OUTPUT;



3. Write a query to identify the number of passengers and total revenue in business class from the ticket\_details table.

CODE;

SELECT

COUNT(customer\_id) AS number\_of\_passengers,

SUM(price\_per\_ticket) AS total\_revenue

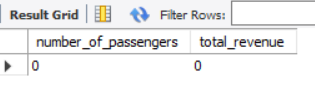
FROM

ticket\_details

WHERE

class\_id = 'Business';

OUTPUT;



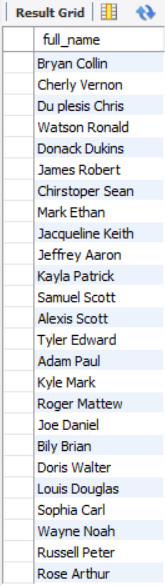
Write a query to display the full name of the customer by extracting the first name and last name from the customer table.

CODE;

select concat(first\_name," ",last\_name) as full\_name

from customer;

OUTPUT;



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.

CODE;

SELECT

c.first\_name,

c.last\_name,

c.customer\_id

FROM

customer c

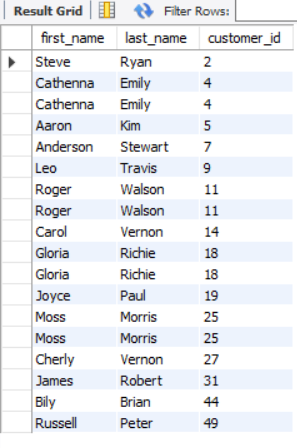
JOIN

ticket\_details t ON c.customer\_id = t.customer\_id

WHERE

t.brand = 'Emirates'

OUPUT;



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.

CODE;

select customer\_id

FROM

passengers\_on\_flights

WHERE

class\_id = 'Economy Plus'

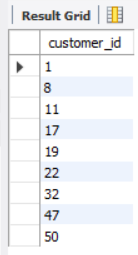
GROUP BY

customer\_id

HAVING

COUNT(customer\_id) > 0;

OUTPUT;

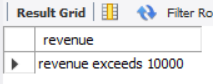


Write a query to identify whether the revenue has crossed 10000 using the IF clause on the ticket\_details table.

CODE;

select if(sum(price\_per\_ticket) > '10000', 'revenue exceeds 10000', 'revenue does not exceed 10000' ) as revenue from ticket\_details;

OUTPUT;



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

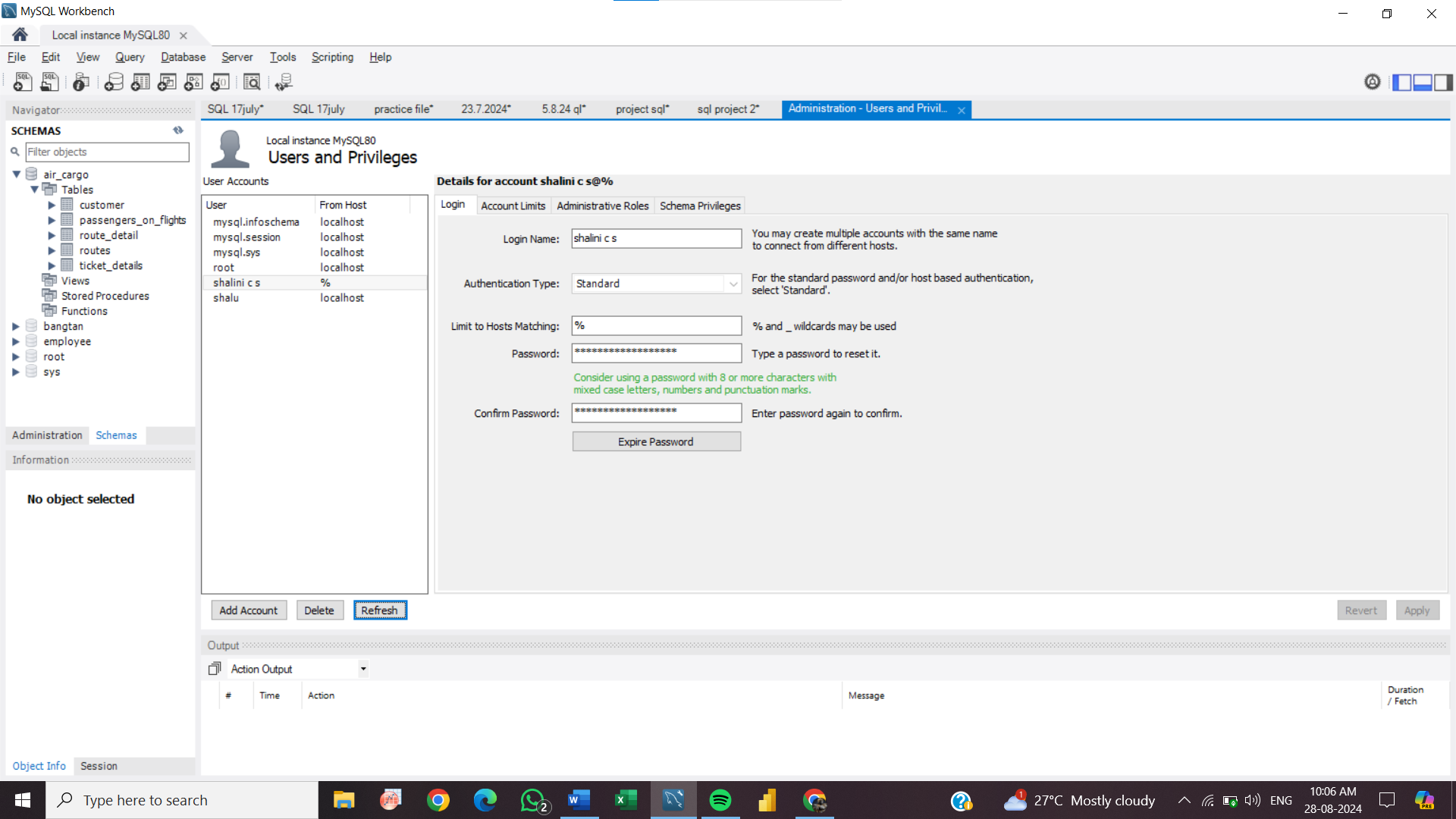
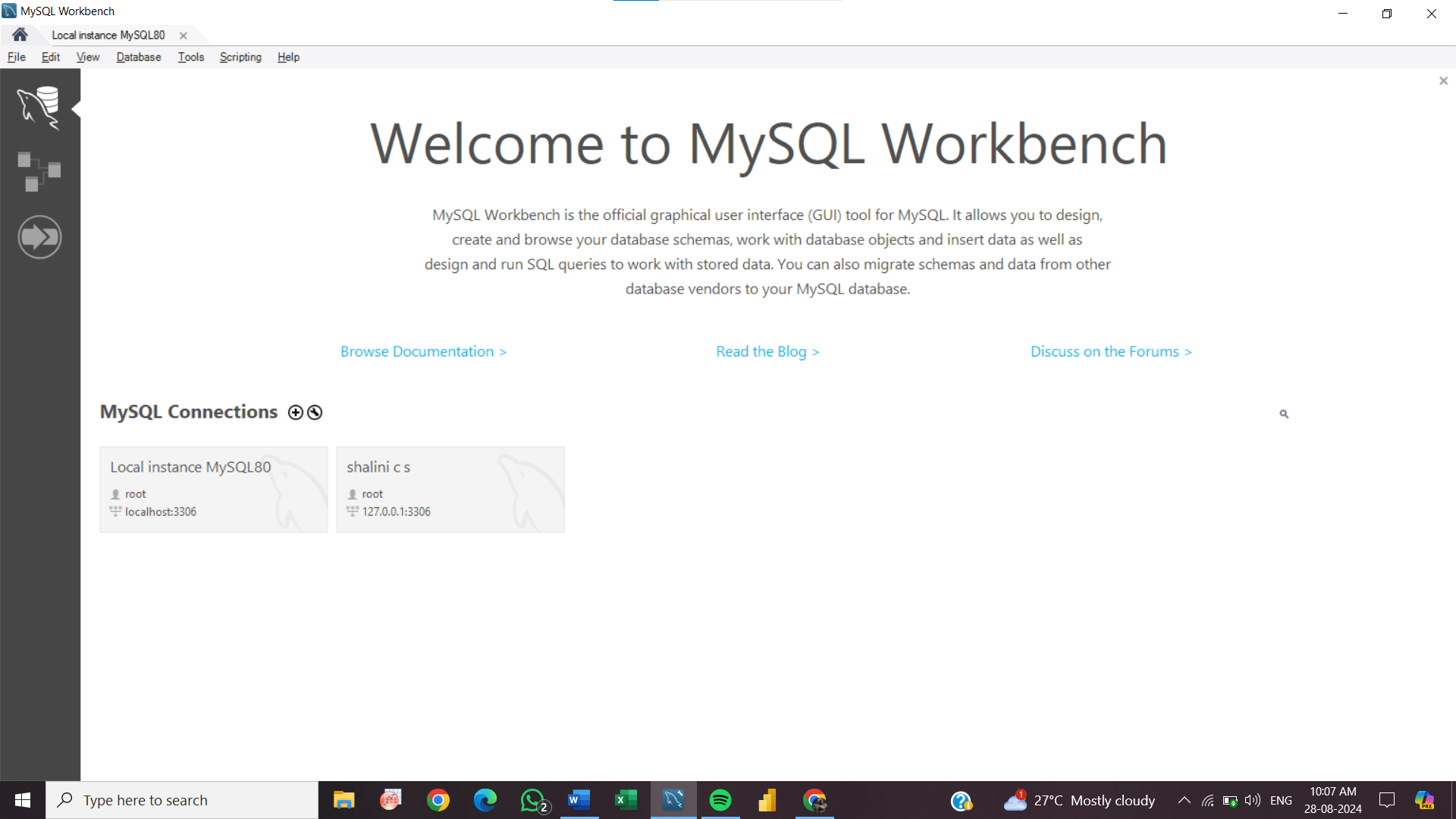
CODE;

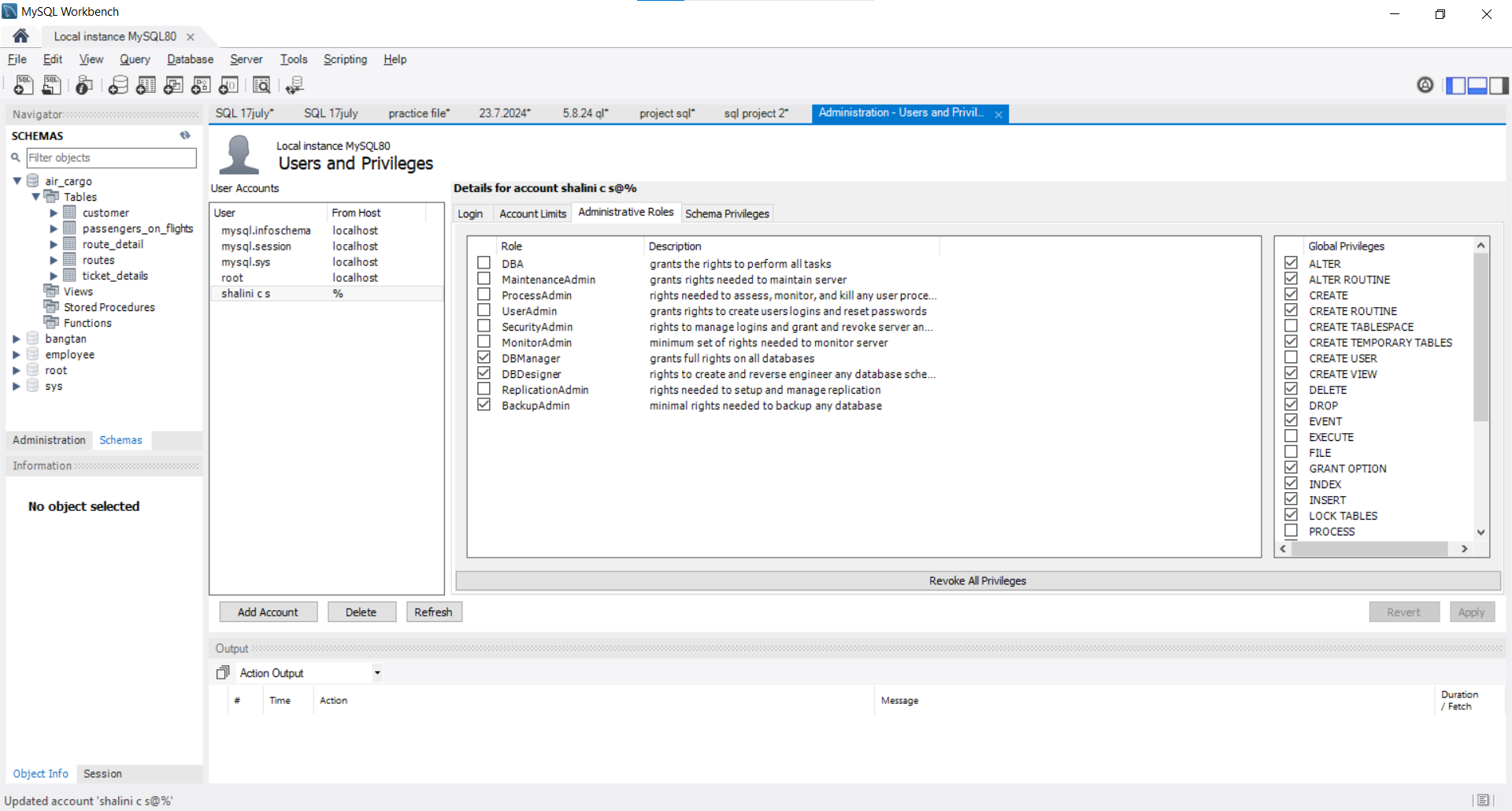
CREATE USER 'shalu'@'localhost' IDENTIFIED BY 'shalu123';

GRANT all PRIVILEGES ON air\_cargo TO 'shalu'@'localhost';

OUTPUT;

or



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

CODE;

SELECT

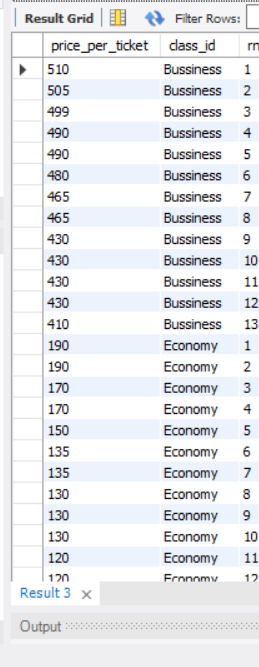
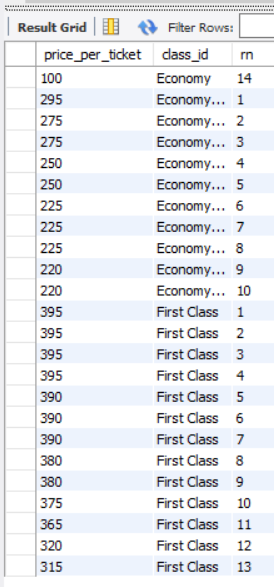
price\_per\_ticket,

class\_id,

ROW\_NUMBER() OVER (PARTITION BY class\_id ORDER BY price\_per\_ticket DESC) AS rn

FROM ticket\_details;

OUTPUT;

11. Write a query to extract the passengers whose route ID is 4 by improving the speed and performance of the passengers\_on\_flights table.

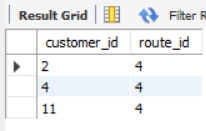
CODE;

SELECT customer\_id,route\_id

FROM passengers\_on\_flights

WHERE route\_id = 4;

OUTPUT



For the route ID 4, write a query to view the execution plan of the passengers\_on\_flights table.

CODE;

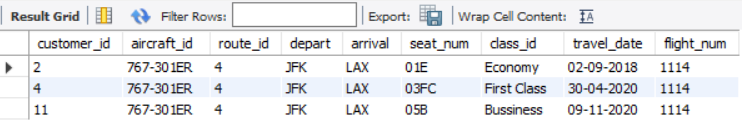
SELECT customer\_id,aircraft\_id,route\_id,depart, arrival,

seat\_num,class\_id,travel\_date,flight\_num

FROM passengers\_on\_flights

WHERE route\_id = 4;

OUTPUT;



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

CODE;

SELECT

distinct Customer\_ID,

aircraft\_ID,

SUM(Price\_per\_ticket) AS Total\_Price

FROM

Ticket\_details

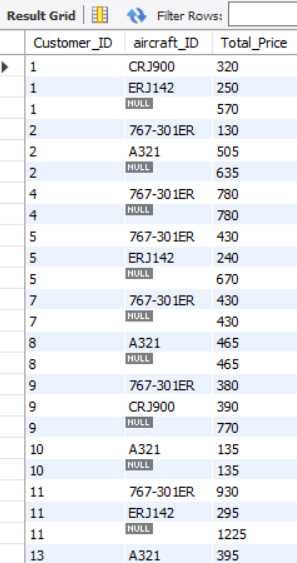
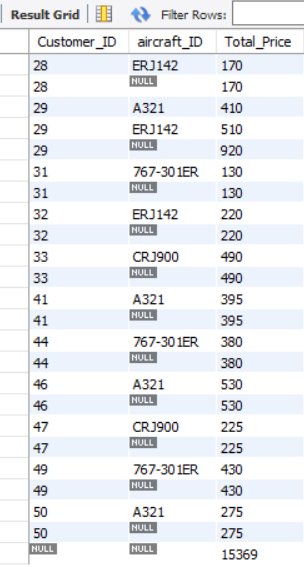
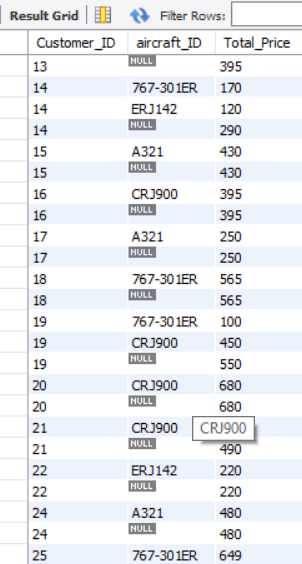
GROUP BY

Customer\_ID,

aircraft\_ID

WITH ROLLUP;

OUTPUT;

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

CODE;

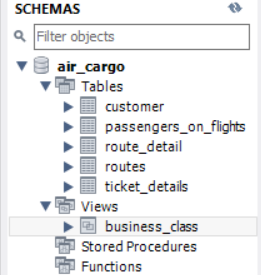
create view business\_class as

select customer\_id,class\_id, brand

from ticket\_details

where class\_id = 'business';

OUTPUT;



select \* from business\_class;

