SQL

STRUCTURED QUERY LANGUAGE

Prepared by:

DEVAKI N

PROJECT TITLE

"AIR CARGO ANALYSIS"

SQL and its Importance

SQL is the database tool that is used to create and access the database to support software applications.

- SQL is important for managing big data, allowing users to easily search, sort and analyze large datasets.
- Universal language
- Open-Source - Simple to use and learn
- Manage Millions of Rows of Data
- Technology Evolution and Increasing Demand

PROBLEM STATEMENT

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.

PROBLEM OBJECTIVE

As a DBA expert, we 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.

TABLES 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:

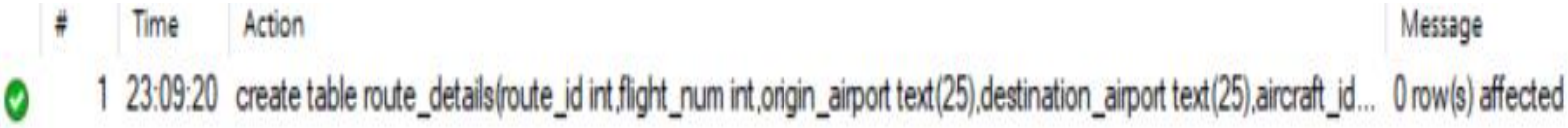
Contains information about the route details.

- 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

1. 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.

```
CREATE TABLE route_details
(route_id int, flight_num int, origin_airport text(25),
destination_airport text(25), aircraft_id text,
distance_miles int,
CHECK(flight_num>0),
UNIQUE(route_id),
CHECK(distance_miles>0));
```

Result:

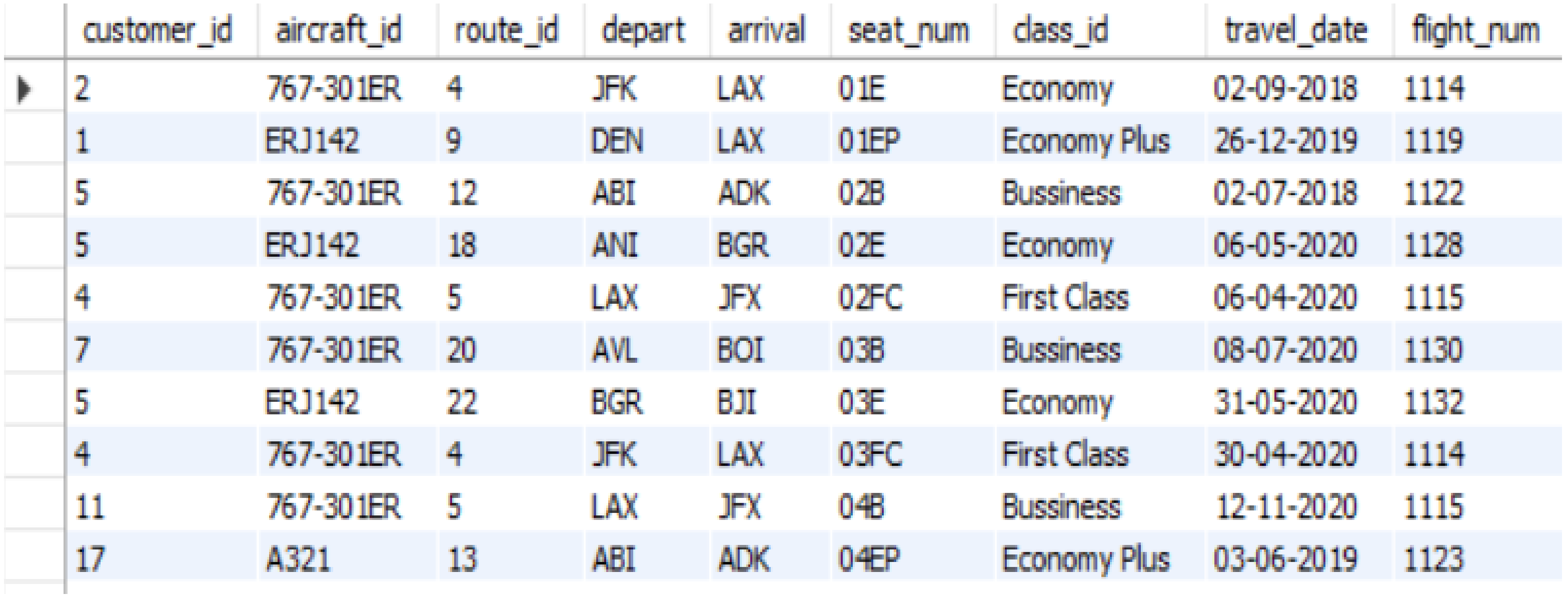


```
#   Time      Action                                                                                                   Message
1   23:09:20  create table route_details(route_id int,flight_num int,origin_airport text(25),destination_airport text(25),aircraft_id...  0 row(s) affected
```

2. 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.

```
SELECT * FROM passengers_on_flights
WHERE route_id
BETWEEN 1 AND 25;
```

Result:

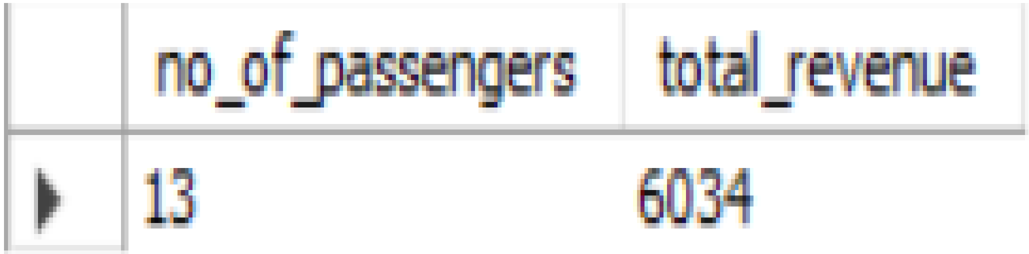


customer_id	aircraft_id	route_id	depart	arrival	seat_num	class_id	travel_date	flight_num
2	767-301ER	4	JFK	LAX	01E	Economy	02-09-2018	1114
1	ERJ142	9	DEN	LAX	01EP	Economy Plus	26-12-2019	1119
5	767-301ER	12	ABI	ADK	02B	Bussiness	02-07-2018	1122
5	ERJ142	18	ANI	BGR	02E	Economy	06-05-2020	1128
4	767-301ER	5	LAX	JFX	02FC	First Class	06-04-2020	1115
7	767-301ER	20	AVL	BOI	03B	Bussiness	08-07-2020	1130
5	ERJ142	22	BGR	BJI	03E	Economy	31-05-2020	1132
4	767-301ER	4	JFK	LAX	03FC	First Class	30-04-2020	1114
11	767-301ER	5	LAX	JFX	04B	Bussiness	12-11-2020	1115
17	A321	13	ABI	ADK	04EP	Economy Plus	03-06-2019	1123

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

```
SELECT COUNT(customer_id) as no_of_passengers,
SUM(price_per_ticket) as total_revenue
FROM ticket_details WHERE class_id=‘Bussiness’;
```

Result:

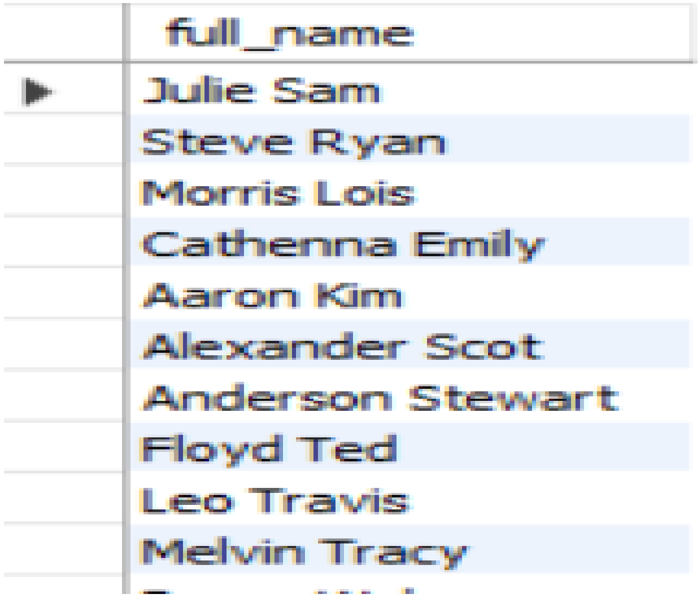


	no_of_passengers	total_revenue
▶	13	6034

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

```
SELECT CONCAT (first_name," ",last_name) as
full_name
FROM customer;
```

Result:

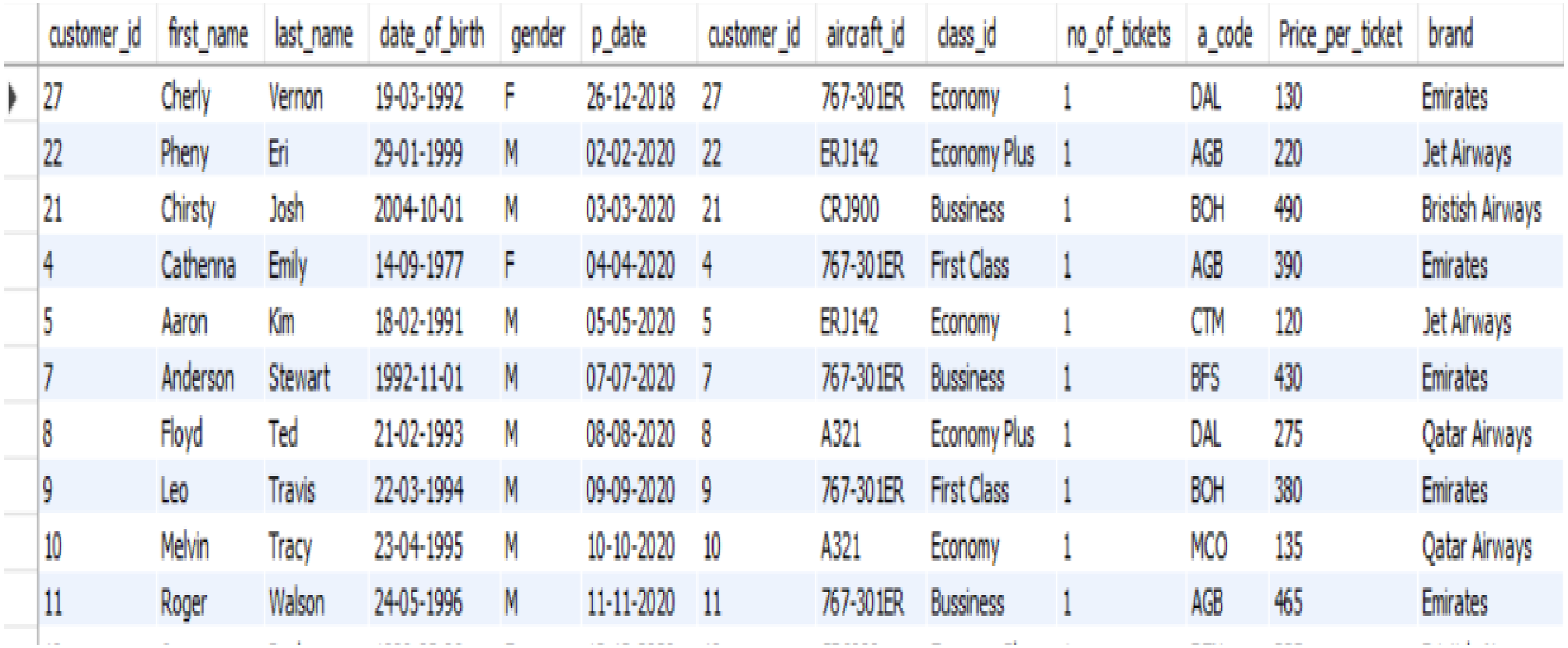


full_name
Julie Sam
Steve Ryan
Morris Lois
Cathenna Emily
Aaron Kim
Alexander Scot
Anderson Stewart
Floyd Ted
Leo Travis
Melvin Tracy

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

```
SELECT * FROM customer
INNER  JOIN ticket_details ON
customer.customer_id = ticket_details.customer_id;
```

Result:

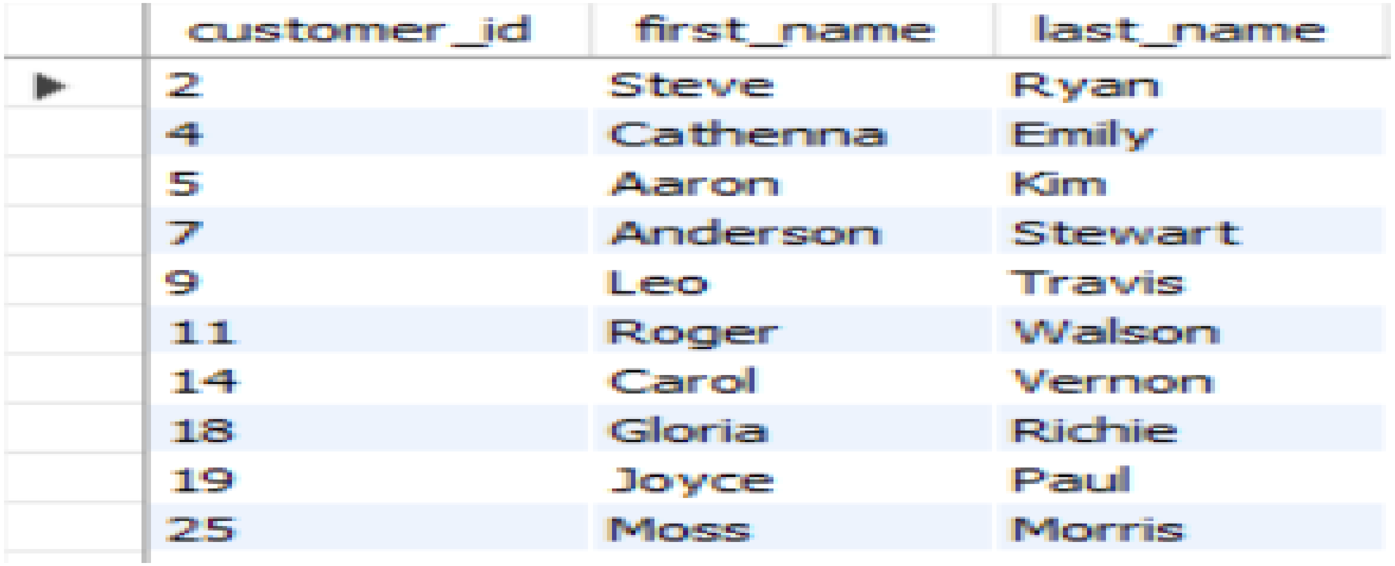


customer_id	first_name	last_name	date_of_birth	gender	p_date	customer_id	aircraft_id	class_id	no_of_tickets	a_code	Price_per_ticket	brand
27	Cherly	Vernon	19-03-1992	F	26-12-2018	27	767-301ER	Economy	1	DAL	130	Emirates
22	Pheny	Eri	29-01-1999	M	02-02-2020	22	ERJ142	Economy Plus	1	AGB	220	Jet Airways
21	Chirsty	Josh	2004-10-01	M	03-03-2020	21	CRJ900	Bussiness	1	BOH	490	Bristish Airways
4	Cathenna	Emily	14-09-1977	F	04-04-2020	4	767-301ER	First Class	1	AGB	390	Emirates
5	Aaron	Kim	18-02-1991	M	05-05-2020	5	ERJ142	Economy	1	CTM	120	Jet Airways
7	Anderson	Stewart	1992-11-01	M	07-07-2020	7	767-301ER	Bussiness	1	BFS	430	Emirates
8	Floyd	Ted	21-02-1993	M	08-08-2020	8	A321	Economy Plus	1	DAL	275	Qatar Airways
9	Leo	Travis	22-03-1994	M	09-09-2020	9	767-301ER	First Class	1	BOH	380	Emirates
10	Melvin	Tracy	23-04-1995	M	10-10-2020	10	A321	Economy	1	MCO	135	Qatar Airways
11	Roger	Walson	24-05-1996	M	11-11-2020	11	767-301ER	Bussiness	1	AGB	465	Emirates

6. 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.

SELECT DISTINCT (a.customer_id), a.first_name, a.last_name FROM customer a, ticket_details b WHERE a.customer_id = b.customer_id AND brand='emirates';

Result:

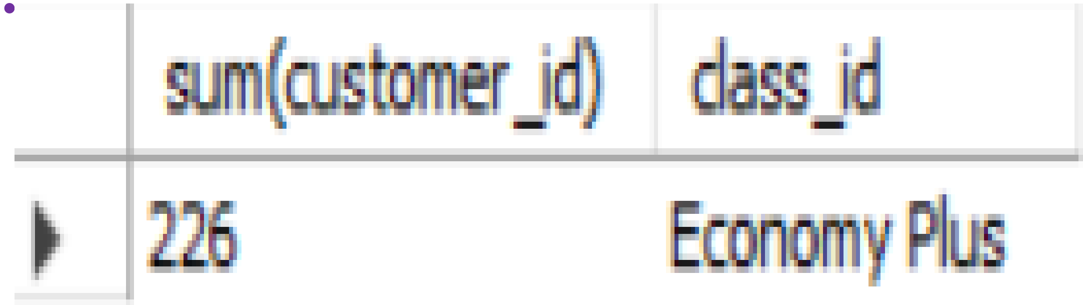


	customer_id	first_name	last_name
▶	2	Steve	Ryan
	4	Cathenna	Emily
	5	Aaron	Kim
	7	Anderson	Stewart
	9	Leo	Travis
	11	Roger	Walson
	14	Carol	Vernon
	18	Gloria	Richie
	19	Joyce	Paul
	25	Moss	Morris

7. 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.

```
SELECT SUM(customer_id), class_id FROM
passengers_on_flights
GROUP BY class_id
HAVING class_id="Economy plus";
```

Result:

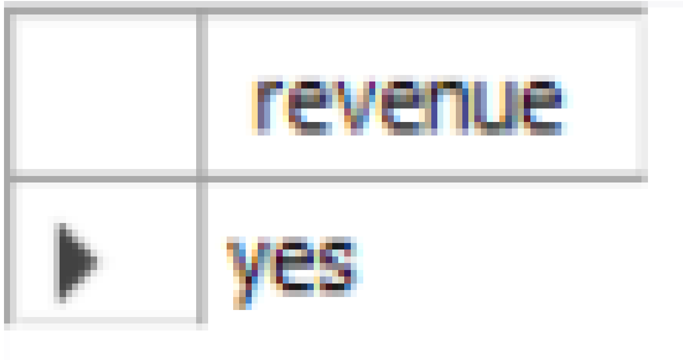


sum(customer_id)	class_id
226	Economy Plus

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

```
SELECT IF(sum(price_per_ticket)>10000,'yes','no')
as revenue FROM ticket_details;
```

Result:

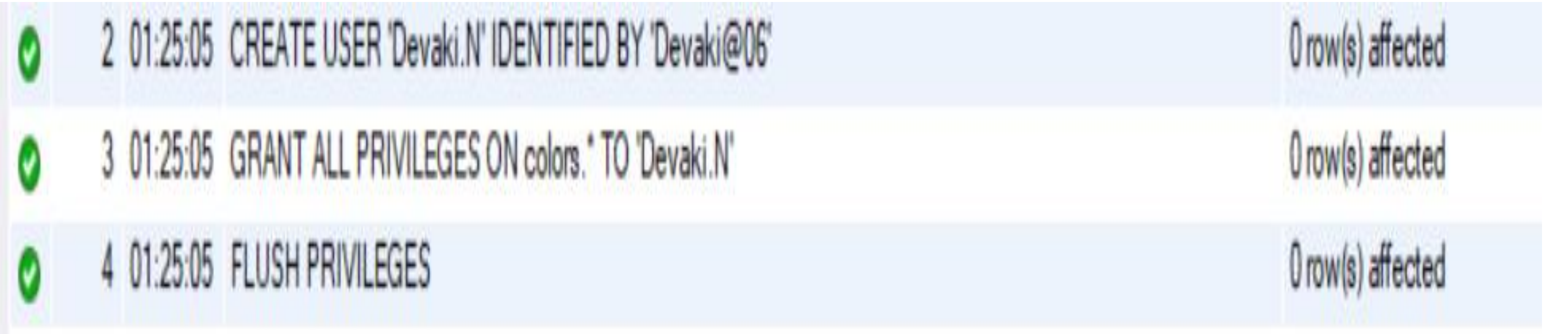


	revenue
▶	yes

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

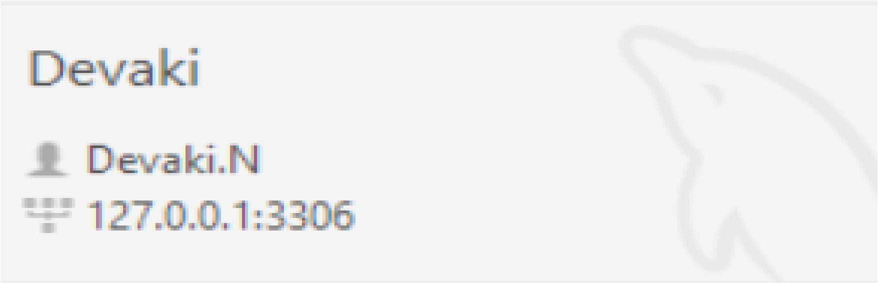
```
CREATE USER 'Devaki.N' IDENTIFIED BY 'Devaki@06';
GRANT ALL PRIVILEGES ON colors.* TO 'Devaki.N';
FLUSH PRIVILEGES;
```

Result:



```
✓  2  01:25:05  CREATE USER 'Devaki.N' IDENTIFIED BY 'Devaki@06'     0 row(s) affected
✓  3  01:25:05  GRANT ALL PRIVILEGES ON colors.* TO 'Devaki.N'       0 row(s) affected
✓  4  01:25:05  FLUSH PRIVILEGES                                     0 row(s) affected
```

Here the user called Devaki is created in the MYSQL connection

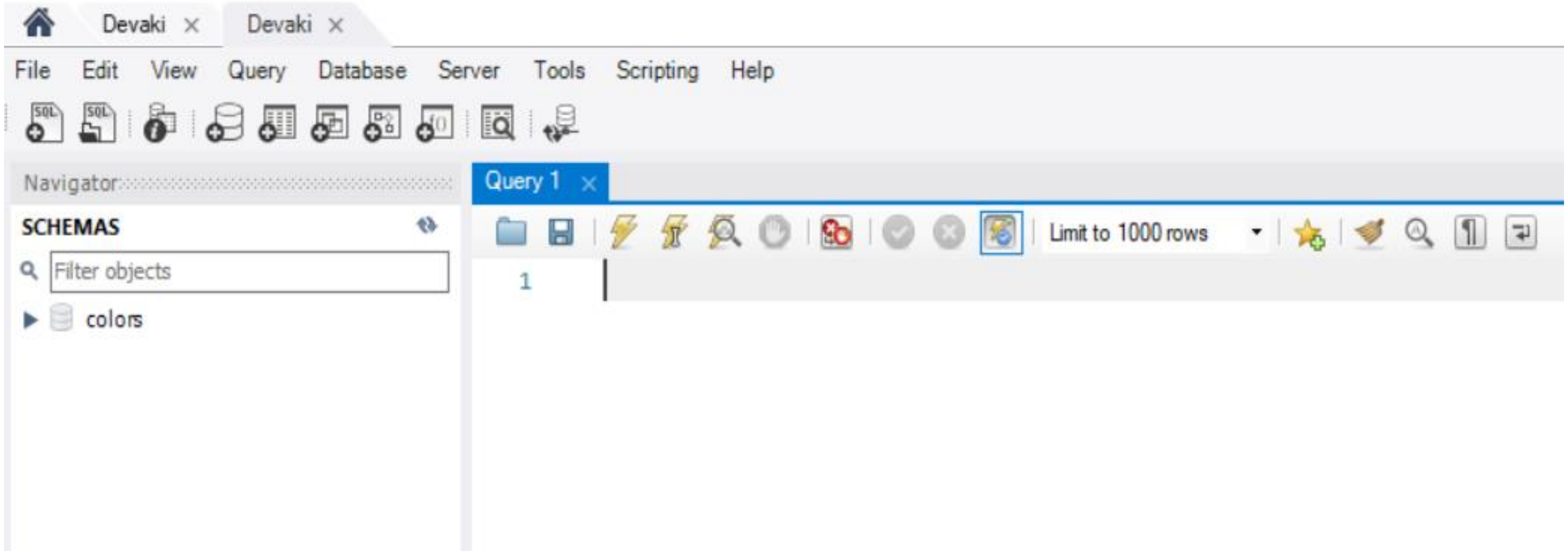


Devaki

Devaki.N

127.0.0.1:3306

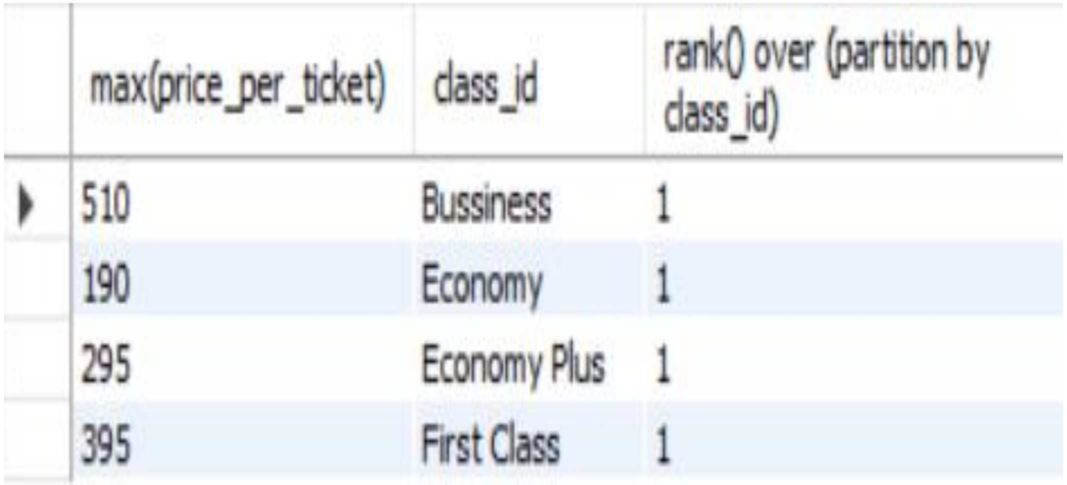
The username Devaki has privilage to the colors database.



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

```
SELECT MAX(price_per_ticket), class_id, RANK()
OVER (PARTITION BY  class_id) FROM
ticket_details GROUP BY class_id;
```

Result:

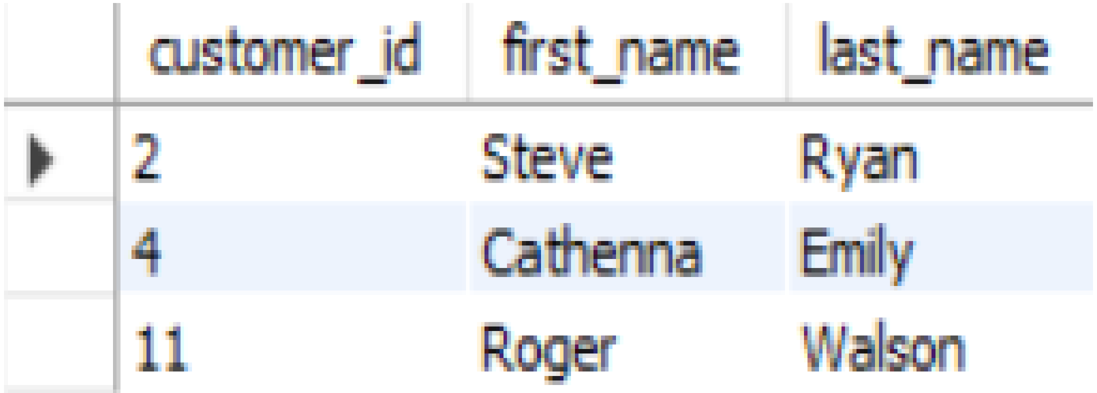


max(price_per_ticket)	class_id	rank() over (partition by class_id)
510	Bussiness	1
190	Economy	1
295	Economy Plus	1
395	First Class	1

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.

```
SELECT a.customer_id, a.first_name, a.last_name
FROM customer a, passengers_on_flights b
WHERE a.customer_id = b.customer_id
AND route_id=4;
```

Result:

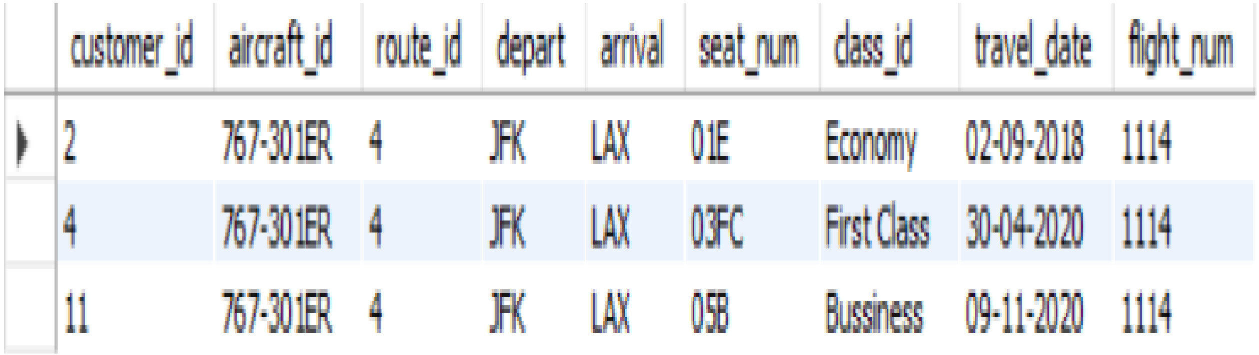


	customer_id	first_name	last_name
▶	2	Steve	Ryan
	4	Cathenna	Emily
	11	Roger	Walson

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

```
SELECT * FROM passengers_on_flights
WHERE route_id=4;
```

Result:

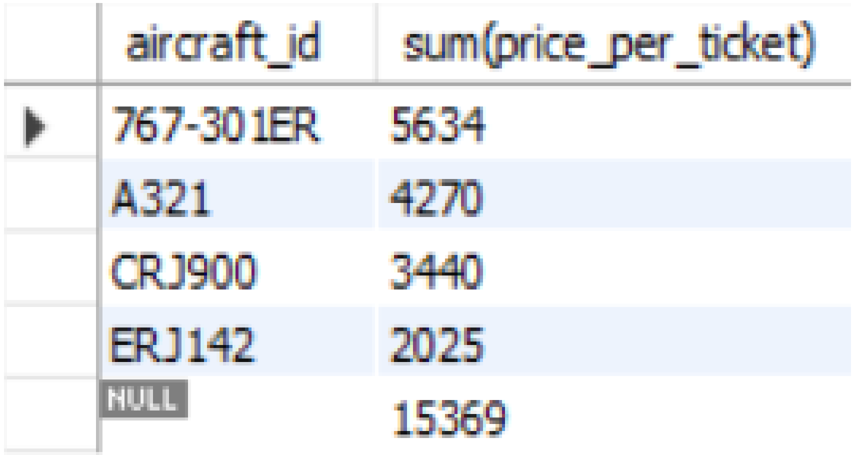


	customer_id	aircraft_id	route_id	depart	arrival	seat_num	class_id	travel_date	flight_num
▶	2	767-301ER	4	JFK	LAX	01E	Economy	02-09-2018	1114
	4	767-301ER	4	JFK	LAX	03FC	First Class	30-04-2020	1114
	11	767-301ER	4	JFK	LAX	05B	Bussiness	09-11-2020	1114

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

```
SELECT aircraft_id, SUM(price_per_ticket)
FROM ticket_details GROUP BY(aircraft_id)
WITH ROLLUP;
```

Result:



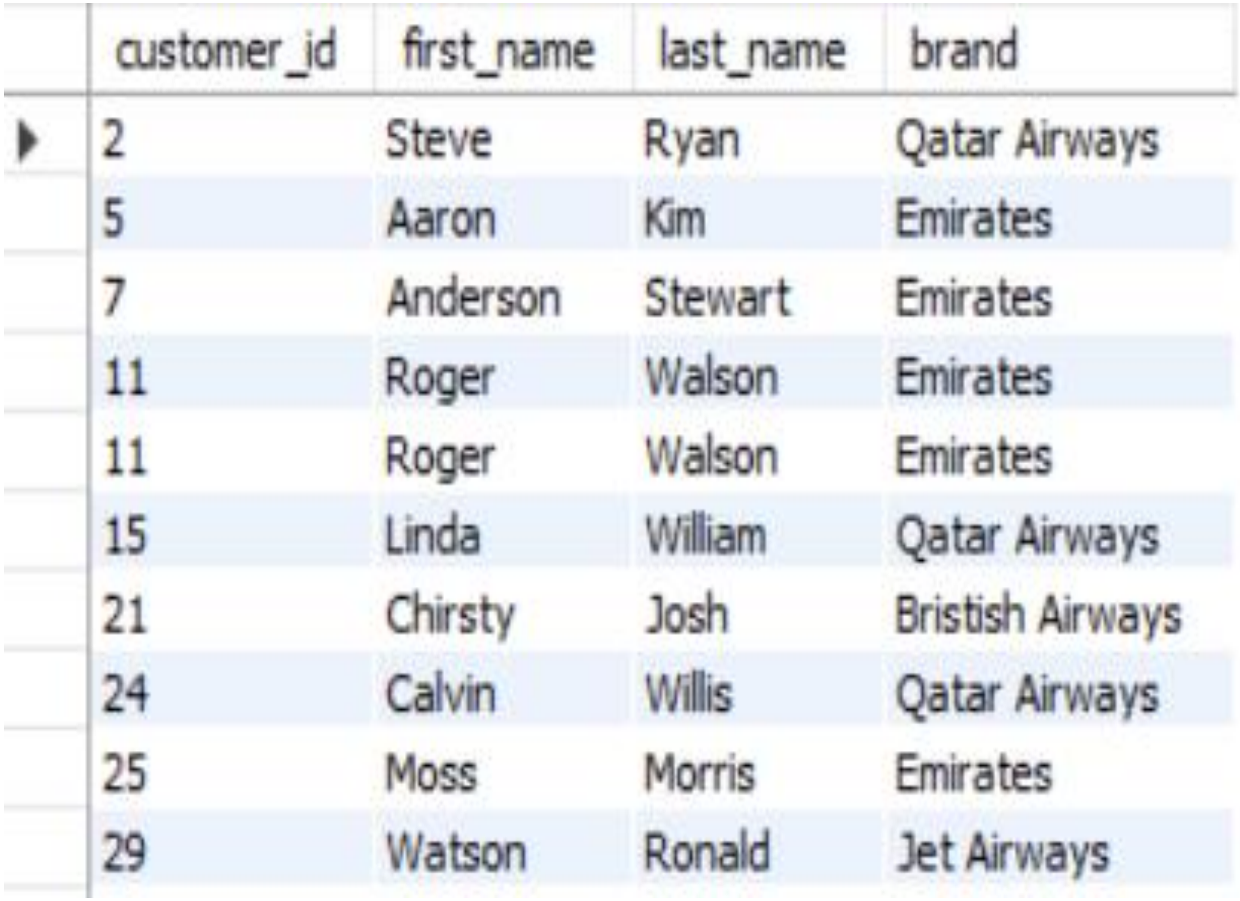
aircraft_id	sum(price_per_ticket)
767-301ER	5634
A321	4270
CRJ900	3440
ERJ142	2025
NULL	15369

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

```
CREATE VIEW ‘bussiness_class_view’ AS SELECT
a.customer_id,a.first_name, a.last_name, b.brand
FROM customer a, ticket_details b WHERE
a.customer_id= b.customer_id AND
class_id=‘Bussiness’;
```

```
SELECT * FROM ‘bussiness_class_view’
```

Result:



customer_id	first_name	last_name	brand
2	Steve	Ryan	Qatar Airways
5	Aaron	Kim	Emirates
7	Anderson	Stewart	Emirates
11	Roger	Walson	Emirates
11	Roger	Walson	Emirates
15	Linda	William	Qatar Airways
21	Chirsty	Josh	Bristish Airways
24	Calvin	Willis	Qatar Airways
25	Moss	Morris	Emirates
29	Watson	Ronald	Jet Airways

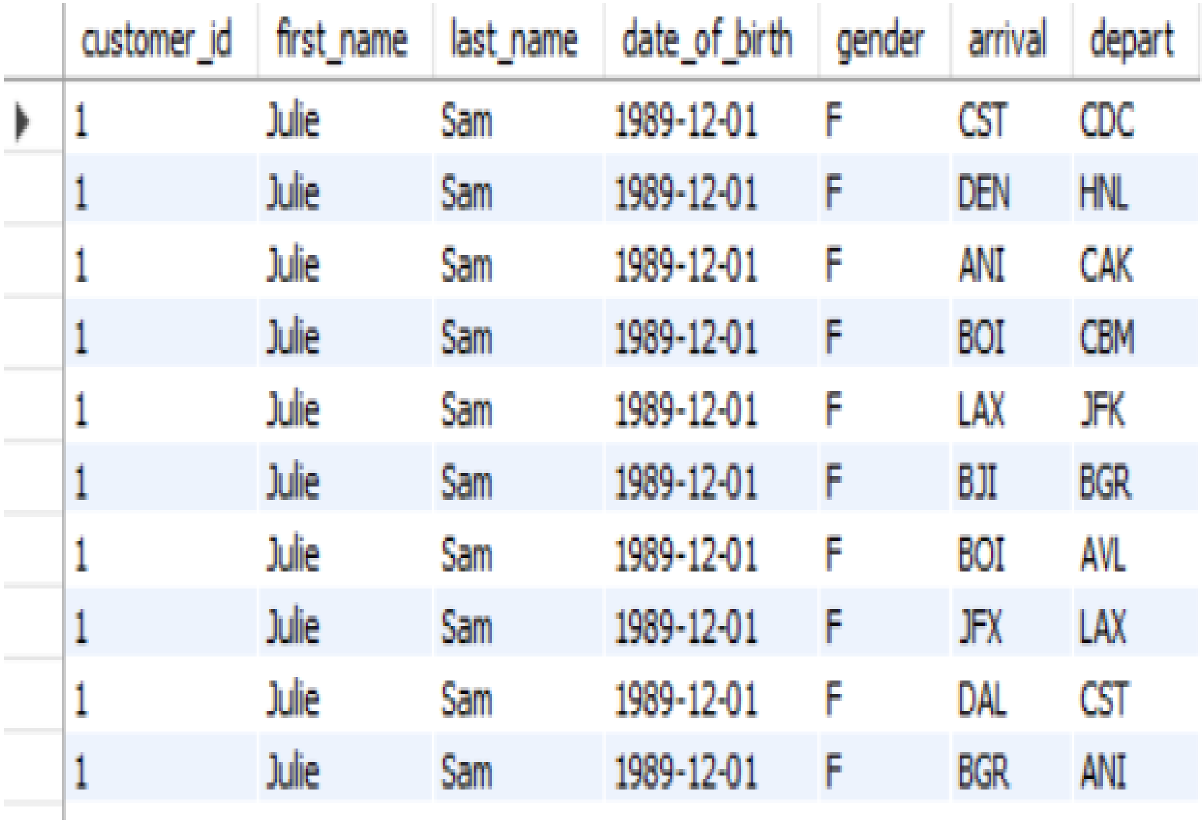
15. Write a query to create a stored procedure to get the details of all passengers flying between a range of routes defined in run time. Also, return an error message if the table doesn't exist.

```
CREATE DEFINER=`root`@`localhost`
PROCEDURE `new_procedure`(    IN route_from
VARCHAR(100),    IN route_to VARCHAR(100))
BEGIN
DECLARE table_exists INT;
--Check if the table exists
SELECT COUNT(*) INTO table_exists FROM
information_schema.tables
```

```
WHERE table_schema = DATABASE()
AND table_name = 'passengers_on_flights';
IF table_exists = 0 THEN
SELECT 'Error: Table passengers_on_flights does
not exist';
ELSE        -- Table exists, fetch passengers within
the specified route range
SELECT a.*,b.arrival, b.depart FROM customer a,
passengers_on_flights b
WHERE b.route_id AND
a.customer_id=b.customer_id
BETWEEN route_from AND route_to;
END IF;
END
```

call new_procedure(4,10);

Result:



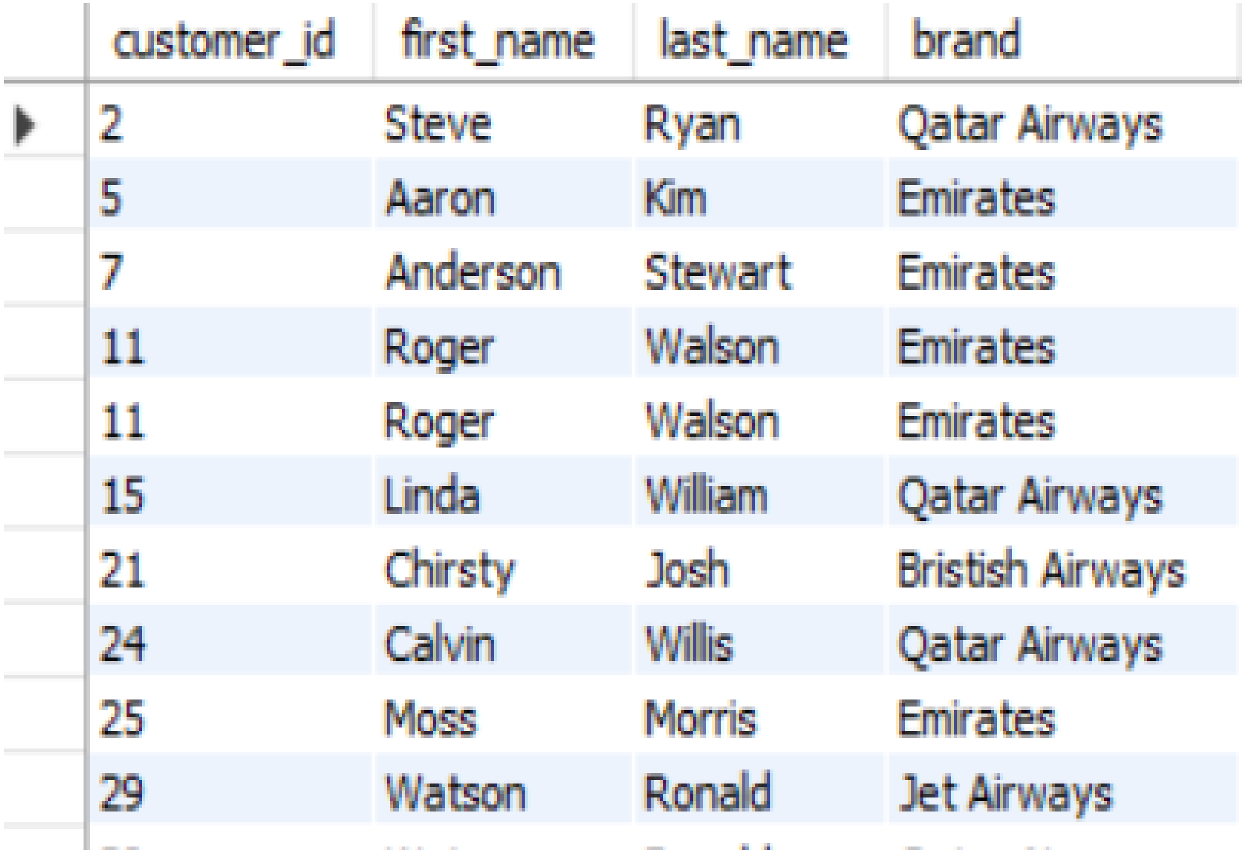
customer_id	first_name	last_name	date_of_birth	gender	arrival	depart
1	Julie	Sam	1989-12-01	F	CST	CDC
1	Julie	Sam	1989-12-01	F	DEN	HNL
1	Julie	Sam	1989-12-01	F	ANI	CAK
1	Julie	Sam	1989-12-01	F	BOI	CBM
1	Julie	Sam	1989-12-01	F	LAX	JFK
1	Julie	Sam	1989-12-01	F	BJI	BGR
1	Julie	Sam	1989-12-01	F	BOI	AVL
1	Julie	Sam	1989-12-01	F	JFX	LAX
1	Julie	Sam	1989-12-01	F	DAL	CST
1	Julie	Sam	1989-12-01	F	BGR	ANI

16. 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.

```
CREATE DEFINER=`root`@`localhost`
PROCEDURE `route_details_procedure`()
BEGIN
select * from routes where distance_miles>2000;
END

SELECT * FROM  route_details_procedure;
```

Result:



customer_id	first_name	last_name	brand
2	Steve	Ryan	Qatar Airways
5	Aaron	Kim	Emirates
7	Anderson	Stewart	Emirates
11	Roger	Walson	Emirates
11	Roger	Walson	Emirates
15	Linda	William	Qatar Airways
21	Chirsty	Josh	Bristish Airways
24	Calvin	Willis	Qatar Airways
25	Moss	Morris	Emirates
29	Watson	Ronald	Jet Airways

THANK YOU