DBMS - MINI PROJECT REPORT

Airline Ticket Booking

Submitted By:

Name: Jampani Gnana Karthik

SRN: PES1UG20CS174

V Semester Section C

ABSTRACT

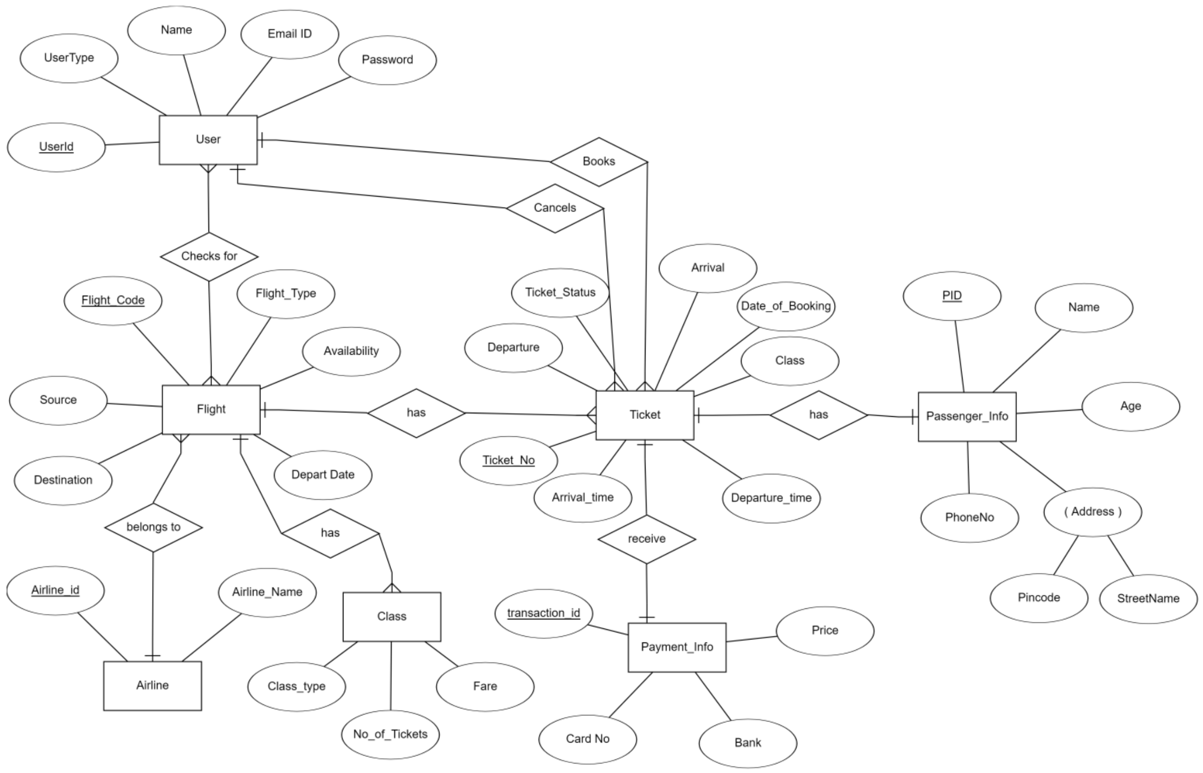
Airline ticket booking system incorporate airline schedules, fare, passenger reservations ticket records and payment details.Here in this project I have created a Mini Airline Ticket Booking database which describes all these features.

and minimal front end for executing queries on flight table and query box for querying all the tables in the database

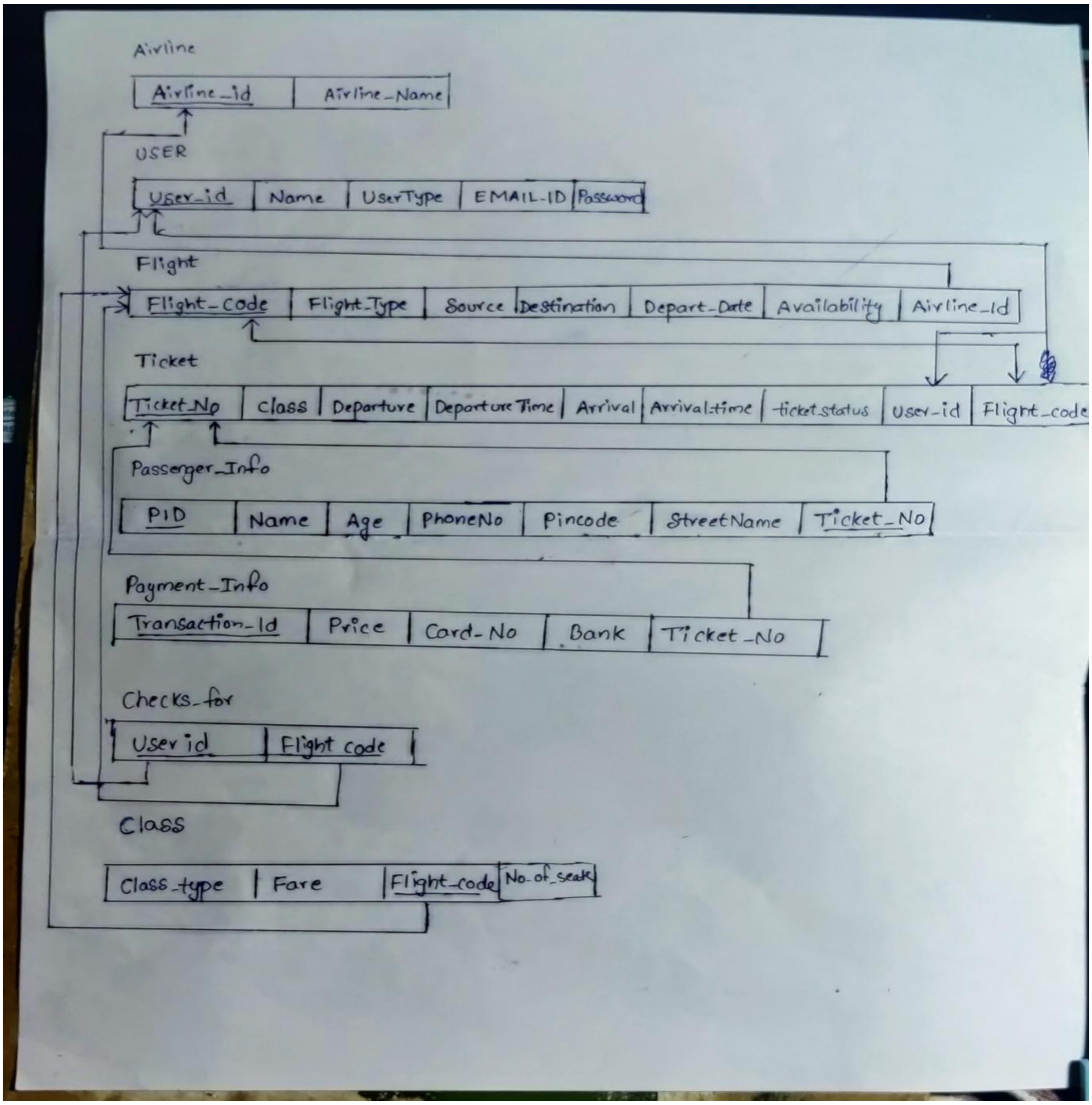
Use Case:

Airline ticket booking databases are used by many Airline companies for booking tickets in an easier way for users using their devices

ER Diagram

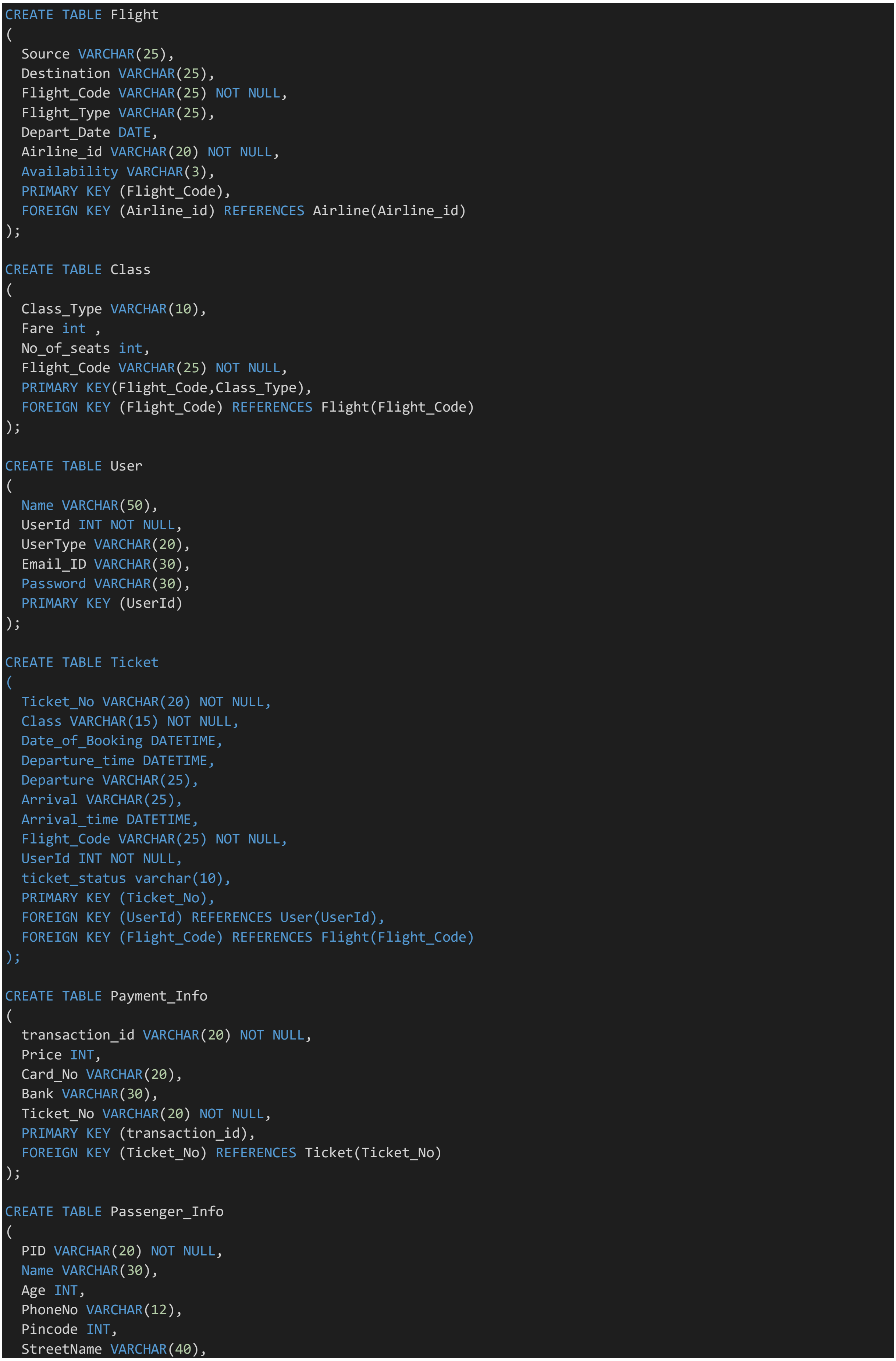


Relational Schema


Airline
Airline_id | Airline_Name
USER
User_id | Name | UserType | EMAIL-ID | Password
Flight
Flight_Code | Flight_Type | Source | Destination | Depart_Date | Availability | Airline_Id
Ticket
Ticket_No | class | Departure | Departure Time | Arrival | Arrival-time | ticket_status | User_id | Flight_code
Passenger_Info
PID | Name | Age | PhoneNo | Pincode | StreetName | Ticket_No
Payment_Info
Transaction_Id | Price | Card-No | Bank | Ticket_No
Checks_for
User id | Flight code
Class
Class_type | Fare | Flight_code | No-of-seats

DDL statements - Building the database

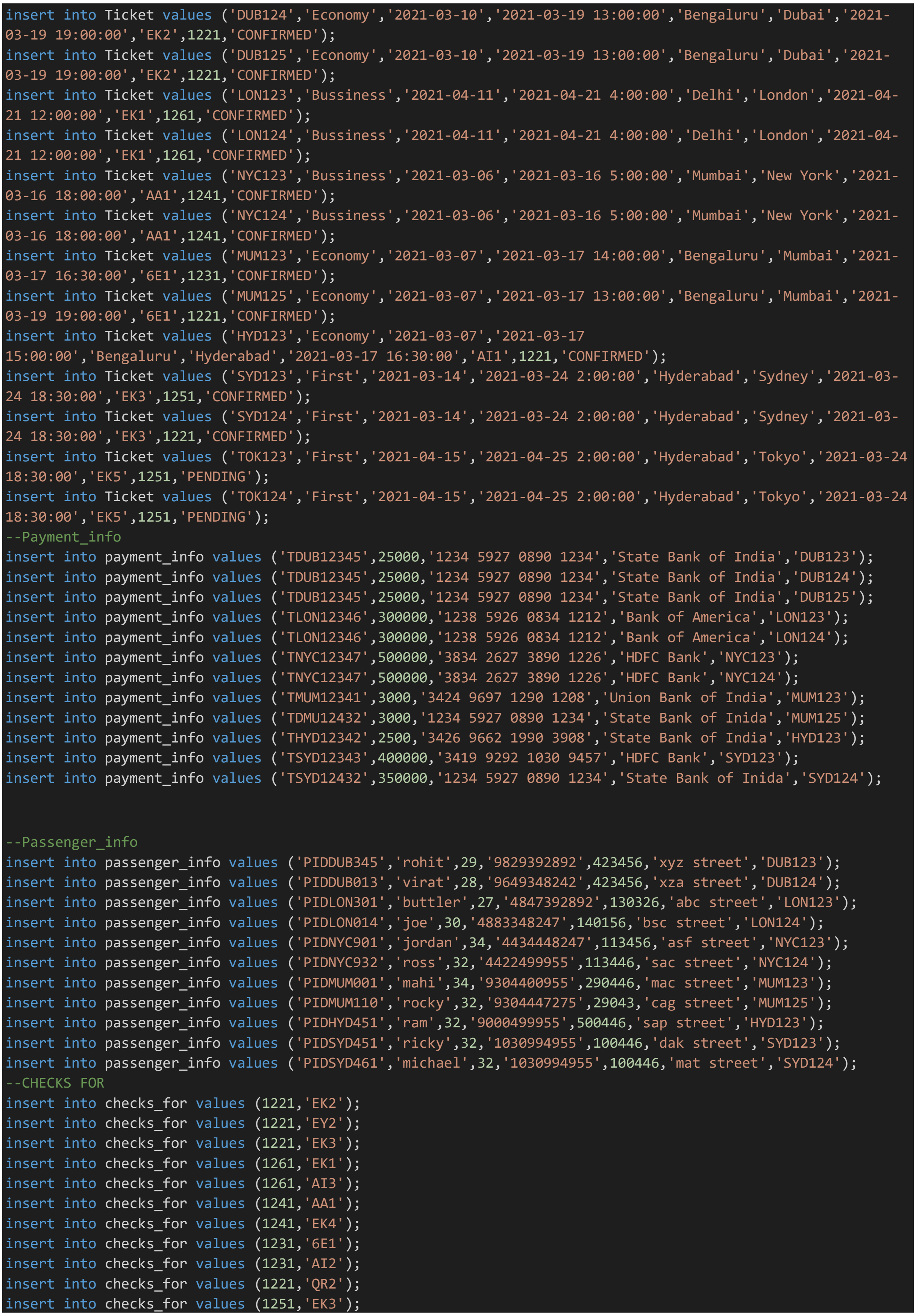
```
CREATE TABLE Airline
(
  Airline_id VARCHAR(10) NOT NULL,
  Airline_Name VARCHAR(50),
  PRIMARY KEY (Airline_id)
);
```



```
CREATE TABLE Flight
(
  Source VARCHAR(25),
  Destination VARCHAR(25),
  Flight_Code VARCHAR(25) NOT NULL,
  Flight_Type VARCHAR(25),
  Depart_Date DATE,
  Airline_id VARCHAR(20) NOT NULL,
  Availability VARCHAR(3),
  PRIMARY KEY (Flight_Code),
  FOREIGN KEY (Airline_id) REFERENCES Airline(Airline_id)
);

CREATE TABLE Class
(
  Class_Type VARCHAR(10),
  Fare int ,
  No_of_seats int,
  Flight_Code VARCHAR(25) NOT NULL,
  PRIMARY KEY(Flight_Code,Class_Type),
  FOREIGN KEY (Flight_Code) REFERENCES Flight(Flight_Code)
);

CREATE TABLE User
(
  Name VARCHAR(50),
  UserId INT NOT NULL,
  UserType VARCHAR(20),
  Email_ID VARCHAR(30),
  Password VARCHAR(30),
  PRIMARY KEY (UserId)
);

CREATE TABLE Ticket
(
  Ticket_No VARCHAR(20) NOT NULL,
  Class VARCHAR(15) NOT NULL,
  Date_of_Booking DATETIME,
  Departure_time DATETIME,
  Departure VARCHAR(25),
  Arrival VARCHAR(25),
  Arrival_time DATETIME,
  Flight_Code VARCHAR(25) NOT NULL,
  UserId INT NOT NULL,
  ticket_status varchar(10),
  PRIMARY KEY (Ticket_No),
  FOREIGN KEY (UserId) REFERENCES User(UserId),
  FOREIGN KEY (Flight_Code) REFERENCES Flight(Flight_Code)
);

CREATE TABLE Payment_Info
(
  transaction_id VARCHAR(20) NOT NULL,
  Price INT,
  Card_No VARCHAR(20),
  Bank VARCHAR(30),
  Ticket_No VARCHAR(20) NOT NULL,
  PRIMARY KEY (transaction_id),
  FOREIGN KEY (Ticket_No) REFERENCES Ticket(Ticket_No)
);

CREATE TABLE Passenger_Info
(
  PID VARCHAR(20) NOT NULL,
  Name VARCHAR(30),
  Age INT,
  PhoneNo VARCHAR(12),
  Pincode INT,
  StreetName VARCHAR(40),
  Ticket_No VARCHAR(20) NOT NULL,
  PRIMARY KEY (PID),
  FOREIGN KEY (Ticket_No) REFERENCES Ticket(Ticket_No)
);

CREATE TABLE Checks_for
(
  UserId INT NOT NULL,
  Flight_Code VARCHAR(25) NOT NULL,
  PRIMARY KEY (UserId, Flight_Code),
  FOREIGN KEY (UserId) REFERENCES User(UserId),
  FOREIGN KEY (Flight_Code) REFERENCES Flight(Flight_Code)
);
```

Populating the Database

```
--Airlines
insert into airline values ('6E','Indigo'),('AA','American Airlines'),('AI','Air India'),('EK','Emirates'),('EY','Etihad Airways'),('QR','Qatar Airways');

--Flight
insert into flight values ('Bengaluru','Mumbai','6E1','Nonstop','2021-03-17','6E','yes');
insert into flight values ('Mumbai','New York','AA1','Connecting','2021-03-16','AA','yes');
insert into flight values ('Bengaluru','Hyderabad','AI1','Nonstop','2021-03-17','AI','yes');
insert into flight values ('Delhi','London','EK1','Nonstop','2021-04-21','EK','yes');
insert into flight values ('Bengaluru','Dubai','EK2','Nonstop','2021-03-19','EK','yes');
insert into flight values ('Hyderabad','Sydney','EK3','Connecting','2021-03-24','EK','yes');
insert into flight values ('Kolkata','Chennai','EY1','Nonstop','2021-04-25','EY','yes');
insert into flight values ('Hyderabad','Qatar','QR1','Nonstop','2021-04-19','QR','yes');
insert into flight values ('Bengaluru','Mumbai','AI2','Nonstop','2021-03-17','AI','yes');
insert into flight values ('Mumbai','New York','EK4','Connecting','2021-03-16','EK','yes');
insert into flight values ('Bengaluru','Hyderabad','QR2','Nonstop','2021-03-17','QR','yes');
insert into flight values ('Delhi','London','AI3','Nonstop','2021-04-21','AI','yes');
insert into flight values ('Bengaluru','Dubai','EY2','Nonstop','2021-03-19','EY','yes');
insert into flight values ('Hyderabad','Tokyo','EK5','Nonstop','2021-04-25','EK','yes');

--CLASS
insert into class values ('ECONOMY',3000,250,'6E1');
insert into class values ('BUSSINESS',7000,100,'6E1');
insert into class values ('ECONOMY',25000,350,'EK2');
insert into class values ('BUSSINESS',50000,100,'EK2');
insert into class values ('ECONOMY',100000,300,'EK1');
insert into class values ('BUSSINESS',300000,150,'EK1');
insert into class values ('ECONOMY',200000,400,'AA1');
insert into class values ('BUSSINESS',500000,150,'AA1');
insert into class values ('ECONOMY',2500,300,'AI1');
insert into class values ('BUSSINESS',7000,100,'AI1');
insert into class values ('ECONOMY',200000,300,'EK3');
insert into class values ('FIRST',400000,50,'EK3');
insert into class values ('BUSSINESS',300000,50,'EK3');
insert into class values ('ECONOMY',80000,300,'EK5');
insert into class values ('FIRST',250000,50,'EK5');
insert into class values ('BUSSINESS',150000,50,'EK5');



--User
insert into user values ('ram',1221,'ADMN','ram@gmail.com','ram@123');
insert into user values ('raju',1231,'NADMN','raju@gmail.com','raju@123');
insert into user values ('musk',1241,'ADMN','musk@gmail.com','musk@123');
insert into user values ('jeff',1251,'NADMN','jeff@gmail.com','jeff@123');
insert into user values ('bill',1261,'NADMN','bill@gmail.com','bill@123');

--Ticket
insert into Ticket values ('DUB123','Economy','2021-03-10','2022-03-19 13:00:00','Bengaluru','Dubai','2021-03-19 19:00:00','EK2',1221,'CONFIRMED');
```



```
insert into Ticket values ('DUB124','Economy','2021-03-10','2021-03-19 13:00:00','Bengaluru','Dubai','2021-
03-19 19:00:00','EK2',1221,'CONFIRMED');
insert into Ticket values ('DUB125','Economy','2021-03-10','2021-03-19 13:00:00','Bengaluru','Dubai','2021-
03-19 19:00:00','EK2',1221,'CONFIRMED');
insert into Ticket values ('LON123','Bussiness','2021-04-11','2021-04-21 4:00:00','Delhi','London','2021-04-
21 12:00:00','EK1',1261,'CONFIRMED');
insert into Ticket values ('LON124','Bussiness','2021-04-11','2021-04-21 4:00:00','Delhi','London','2021-04-
21 12:00:00','EK1',1261,'CONFIRMED');
insert into Ticket values ('NYC123','Bussiness','2021-03-06','2021-03-16 5:00:00','Mumbai','New York','2021-
03-16 18:00:00','AA1',1241,'CONFIRMED');
insert into Ticket values ('NYC124','Bussiness','2021-03-06','2021-03-16 5:00:00','Mumbai','New York','2021-
03-16 18:00:00','AA1',1241,'CONFIRMED');
insert into Ticket values ('MUM123','Economy','2021-03-07','2021-03-17 14:00:00','Bengaluru','Mumbai','2021-
03-17 16:30:00','6E1',1231,'CONFIRMED');
insert into Ticket values ('MUM125','Economy','2021-03-07','2021-03-17 13:00:00','Bengaluru','Mumbai','2021-
03-19 19:00:00','6E1',1221,'CONFIRMED');
insert into Ticket values ('HYD123','Economy','2021-03-07','2021-03-17
15:00:00','Bengaluru','Hyderabad','2021-03-17 16:30:00','AI1',1221,'CONFIRMED');
insert into Ticket values ('SYD123','First','2021-03-14','2021-03-24 2:00:00','Hyderabad','Sydney','2021-03-
24 18:30:00','EK3',1251,'CONFIRMED');
insert into Ticket values ('SYD124','First','2021-03-14','2021-03-24 2:00:00','Hyderabad','Sydney','2021-03-
24 18:30:00','EK3',1221,'CONFIRMED');
insert into Ticket values ('TOK123','First','2021-04-15','2021-04-25 2:00:00','Hyderabad','Tokyo','2021-03-24
18:30:00','EK5',1251,'PENDING');
insert into Ticket values ('TOK124','First','2021-04-15','2021-04-25 2:00:00','Hyderabad','Tokyo','2021-03-24
18:30:00','EK5',1251,'PENDING');
--Payment_info
insert into payment_info values ('TDUB12345',25000,'1234 5927 0890 1234','State Bank of India','DUB123');
insert into payment_info values ('TDUB12345',25000,'1234 5927 0890 1234','State Bank of India','DUB124');
insert into payment_info values ('TDUB12345',25000,'1234 5927 0890 1234','State Bank of India','DUB125');
insert into payment_info values ('TLON12346',300000,'1238 5926 0834 1212','Bank of America','LON123');
insert into payment_info values ('TLON12346',300000,'1238 5926 0834 1212','Bank of America','LON124');
insert into payment_info values ('TNYC12347',500000,'3834 2627 3890 1226','HDFC Bank','NYC123');
insert into payment_info values ('TNYC12347',500000,'3834 2627 3890 1226','HDFC Bank','NYC124');
insert into payment_info values ('TMUM12341',3000,'3424 9697 1290 1208','Union Bank of India','MUM123');
insert into payment_info values ('TDMU12432',3000,'1234 5927 0890 1234','State Bank of Inida','MUM125');
insert into payment_info values ('THYD12342',2500,'3426 9662 1990 3908','State Bank of India','HYD123');
insert into payment_info values ('TSYD12343',400000,'3419 9292 1030 9457','HDFC Bank','SYD123');
insert into payment_info values ('TSYD12432',350000,'1234 5927 0890 1234','State Bank of Inida','SYD124');


--Passenger_info
insert into passenger_info values ('PIDDUB345','rohit',29,'9829392892',423456,'xyz street','DUB123');
insert into passenger_info values ('PIDDUB013','virat',28,'9649348242',423456,'xza street','DUB124');
insert into passenger_info values ('PIDLON301','buttler',27,'4847392892',130326,'abc street','LON123');
insert into passenger_info values ('PIDLON014','joe',30,'4883348247',140156,'bsc street','LON124');
insert into passenger_info values ('PIDNYC901','jordan',34,'4434448247',113456,'asf street','NYC123');
insert into passenger_info values ('PIDNYC932','ross',32,'4422499955',113446,'sac street','NYC124');
insert into passenger_info values ('PIDMUM001','mahi',34,'9304400955',290446,'mac street','MUM123');
insert into passenger_info values ('PIDMUM110','rocky',32,'9304447275',29043,'cag street','MUM125');
insert into passenger_info values ('PIDHYD451','ram',32,'9000499955',500446,'sap street','HYD123');
insert into passenger_info values ('PIDSYD451','ricky',32,'1030994955',100446,'dak street','SYD123');
insert into passenger_info values ('PIDSYD461','michael',32,'1030994955',100446,'mat street','SYD124');
--CHECKS FOR
insert into checks_for values (1221,'EK2');
insert into checks_for values (1221,'EY2');
insert into checks_for values (1221,'EK3');
insert into checks_for values (1261,'EK1');
insert into checks_for values (1261,'AI3');
insert into checks_for values (1241,'AA1');
insert into checks_for values (1241,'EK4');
insert into checks_for values (1231,'6E1');
insert into checks_for values (1231,'AI2');
insert into checks_for values (1221,'QR2');
insert into checks_for values (1251,'EK3');
```

Tool Used

Front End: Streamlit Backend - Mysql

Queries

Join queries (at least 6)

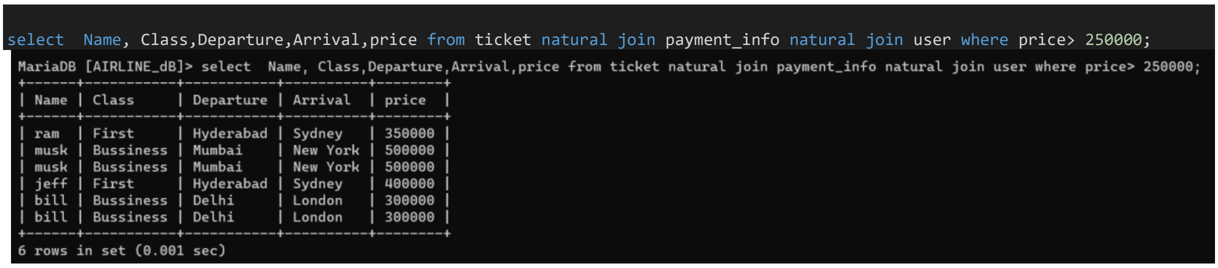
Write the query in English Language, Show the equivalent SQL statement and also screenshot of the query and the results.

Include 2 regular join, 2 co-related and 2 nested queries

Regular joins

--Getting all the users who has booked tickets for price more than 25000

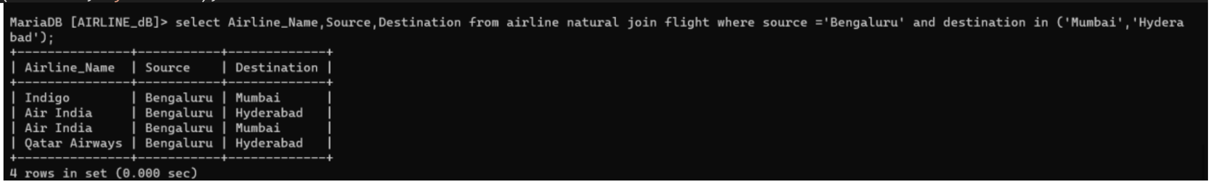
```
select  Name, Class,Departure,Arrival,price from ticket natural join payment_info natural join user where price> 250000;
```



```
MariaDB [AIRLINE_dB]> select  Name, Class,Departure,Arrival,price from ticket natural join payment_info natural join user where price> 250000;
+------+-----------+-----------+----------+--------+
| Name | Class     | Departure | Arrival  | price  |
+------+-----------+-----------+----------+--------+
| ram  | First     | Hyderabad | Sydney   | 350000 |
| musk | Bussiness | Mumbai    | New York | 500000 |
| musk | Bussiness | Mumbai    | New York | 500000 |
| jeff | First     | Hyderabad | Sydney   | 400000 |
| bill | Bussiness | Delhi     | London   | 300000 |
| bill | Bussiness | Delhi     | London   | 300000 |
+------+-----------+-----------+----------+--------+
6 rows in set (0.001 sec)
```

--Getting all the airline names of flights going from Bengaluru to Mumbai or Hyderabad

```
select Airline_Name,Source,Destination from airline natural join flight where source ='Bengaluru' and destination in
('Mumbai','Hyderabad');
```

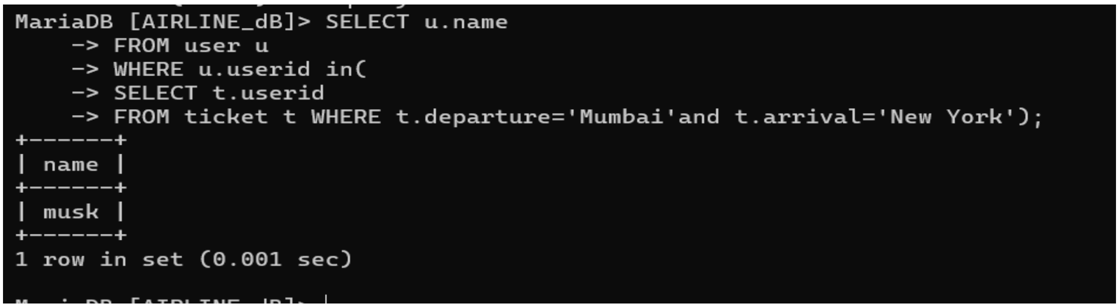


```
MariaDB [AIRLINE_dB]> select Airline_Name,Source,Destination from airline natural join flight where source ='Bengaluru' and destination in ('Mumbai','Hydera
bad');
+---------------+-----------+-------------+
| Airline_Name  | Source    | Destination |
+---------------+-----------+-------------+
| Indigo        | Bengaluru | Mumbai      |
| Air India     | Bengaluru | Hyderabad   |
| Air India     | Bengaluru | Mumbai      |
| Qatar Airways | Bengaluru | Hyderabad   |
+---------------+-----------+-------------+
4 rows in set (0.000 sec)
```

Co-related queries

--Getting uname if user has ticket from Mumbai to New York

```
SELECT u.name
FROM user u
WHERE u.userid in(
SELECT t.userid
FROM ticket t WHERE t.departure='Mumbai'and t.arrival='New York');
```

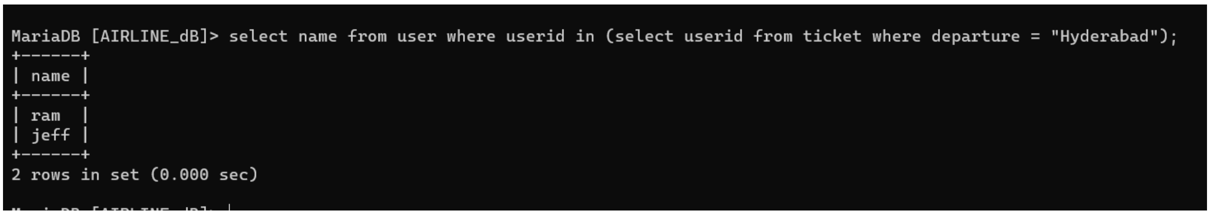


```
MariaDB [AIRLINE_dB]> SELECT u.name
    -> FROM user u
    -> WHERE u.userid in(
    -> SELECT t.userid
    -> FROM ticket t WHERE t.departure='Mumbai'and t.arrival='New York');
+-------+
| name  |
+-------+
| musk  |
+-------+
1 row in set (0.001 sec)
```

Nested Queries

--getting username who booked tickets with departure as Hyderabad

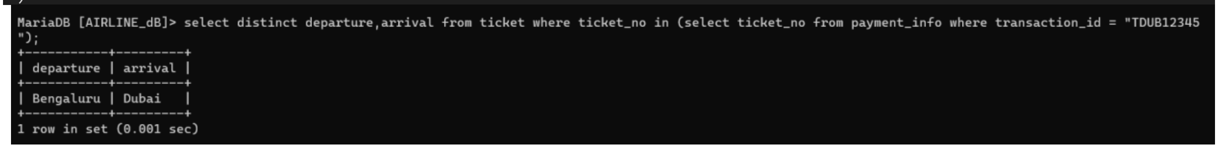
```
select name from user where userid in (select userid from ticket where departure = "Hyderabad")
```



```
MariaDB [AIRLINE_dB]> select name from user where userid in (select userid from ticket where departure = "Hyderabad");
+------+
| name |
+------+
| ram  |
| jeff |
+------+
2 rows in set (0.000 sec)
```

--getting departure,arrival by transaction_id

```
select departure,arrival from ticket where ticket_no in (select ticket_no from payment_info where transaction_id = "
TDUB12345
")
```



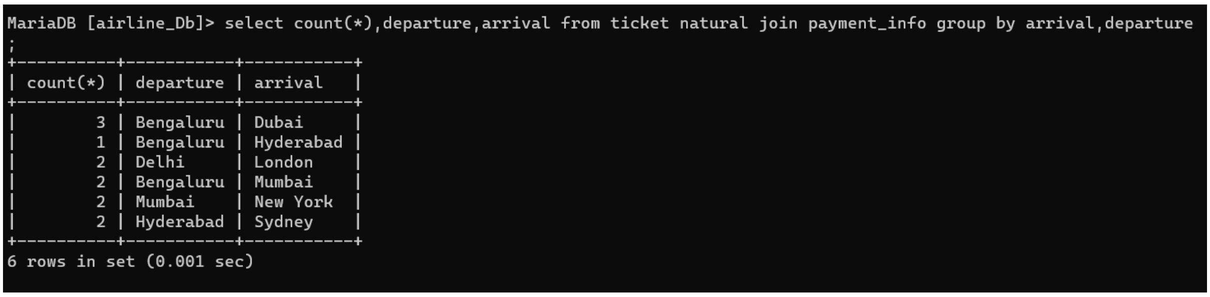
```
MariaDB [AIRLINE_dB]> select distinct departure,arrival from ticket where ticket_no in (select ticket_no from payment_info where transaction_id = "TDUB12345
");
+-----------+---------+
| departure | arrival |
+-----------+---------+
| Bengaluru | Dubai   |
+-----------+---------+
1 row in set (0.001 sec)
```

Aggregate Functions (at least 2)

Showcase at least 2 Aggregate function queries. Write the query in English Language, Show the equivalent SQL statement and also screenshot of the query and the results

--getting count of all the tickets booked for the different flights

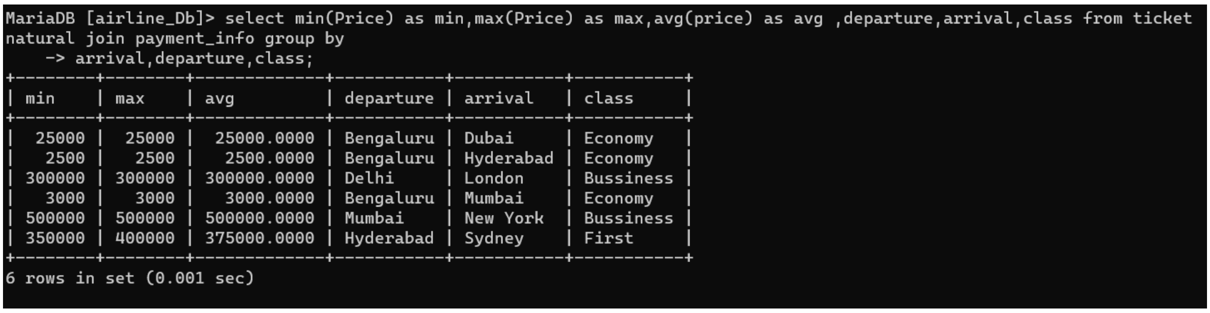
```
select count(*),departure,arrival from ticket natural join payment_info group by arrival,departure;
```



```
MariaDB [airline_Db]> select count(*),departure,arrival from ticket natural join payment_info group by arrival,departure
;
+----------+-----------+-----------+
| count(*) | departure | arrival   |
+----------+-----------+-----------+
|        3 | Bengaluru | Dubai     |
|        1 | Bengaluru | Hyderabad |
|        2 | Delhi     | London    |
|        2 | Bengaluru | Mumbai    |
|        2 | Mumbai    | New York  |
|        2 | Hyderabad | Sydney    |
+----------+-----------+-----------+
6 rows in set (0.001 sec)
```

--getting min,max,avg prices of tickets for each flight

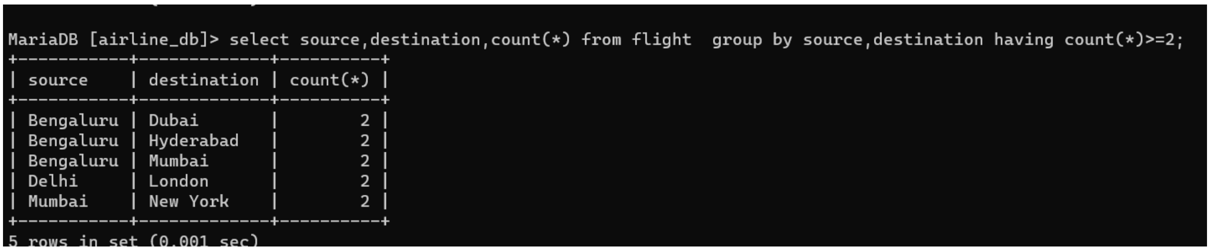
```
select min(Price) as min,max(Price) as max,avg(price) as avg ,departure,arrival,class from ticket natural
join payment_info group by
arrival,departure,class;
```



```
MariaDB [airline_Db]> select min(Price) as min,max(Price) as max,avg(price) as avg ,departure,arrival,class from ticket
natural join payment_info group by
    -> arrival,departure,class;
+--------+--------+-------------+-----------+-----------+-----------+
| min    | max    | avg         | departure | arrival   | class     |
+--------+--------+-------------+-----------+-----------+-----------+
|  25000 |  25000 |  25000.0000 | Bengaluru | Dubai     | Economy   |
|   2500 |   2500 |   2500.0000 | Bengaluru | Hyderabad | Economy   |
| 300000 | 300000 | 300000.0000 | Delhi     | London    | Bussiness |
|   3000 |   3000 |   3000.0000 | Bengaluru | Mumbai    | Economy   |
| 500000 | 500000 | 500000.0000 | Mumbai    | New York  | Bussiness |
| 350000 | 400000 | 375000.0000 | Hyderabad | Sydney    | First     |
+--------+--------+-------------+-----------+-----------+-----------+
6 rows in set (0.001 sec)
```

--getting count of flights having same source and destination having count>=2

```
select source,destination,count(*) from flight  group by source,destination having count(*)>=2;
```



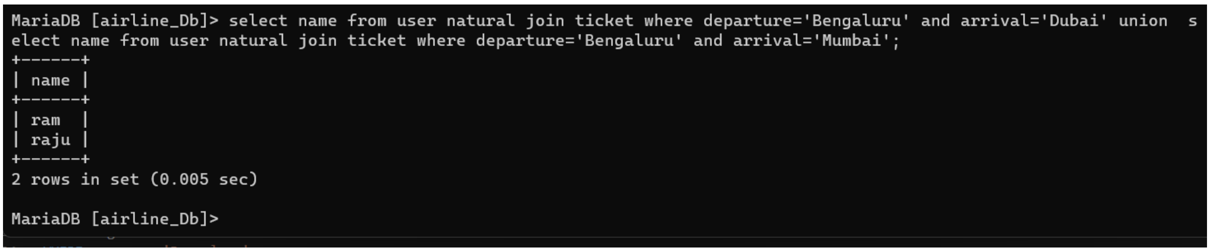
```
MariaDB [airline_db]> select source,destination,count(*) from flight  group by source,destination having count(*)>=2;
+-----------+-------------+----------+
| source    | destination | count(*) |
+-----------+-------------+----------+
| Bengaluru | Dubai       |        2 |
| Bengaluru | Hyderabad   |        2 |
| Bengaluru | Mumbai      |        2 |
| Delhi     | London      |        2 |
| Mumbai    | New York    |        2 |
+-----------+-------------+----------+
5 rows in set (0.001 sec)
```

Set Operations (at least 2)

Showcase at least 2 Set Operations queries . Write the query in English Language, Show the equivalent SQL statement and also screenshot of the query and the results

--getting users who booked tickets from Bengaluru to dubai or Bengaluru to Mumbai

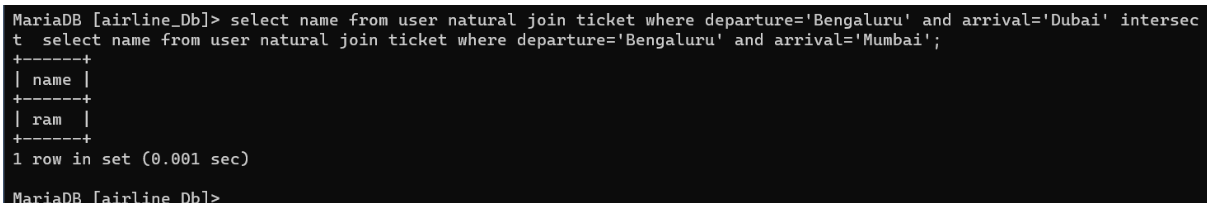
```
select name from user natural join ticket where departure='Bengaluru' and arrival='Dubai' union  select name
from user natural join ticket where departure='Bengaluru' and arrival='Mumbai';
```



```
MariaDB [airline_Db]> select name from user natural join ticket where departure='Bengaluru' and arrival='Dubai' union  s
elect name from user natural join ticket where departure='Bengaluru' and arrival='Mumbai';
+------+
| name |
+------+
| ram  |
| raju |
+------+
2 rows in set (0.005 sec)

MariaDB [airline_Db]>
```

--getting users who booked tickets from Bengaluru to dubai and Bengaluru to Mumbai

```
select name from user natural join ticket where departure='Bengaluru' and arrival='Dubai' intersect  select
name from user natural join ticket where departure='Bengaluru' and arrival='Mumbai';
```



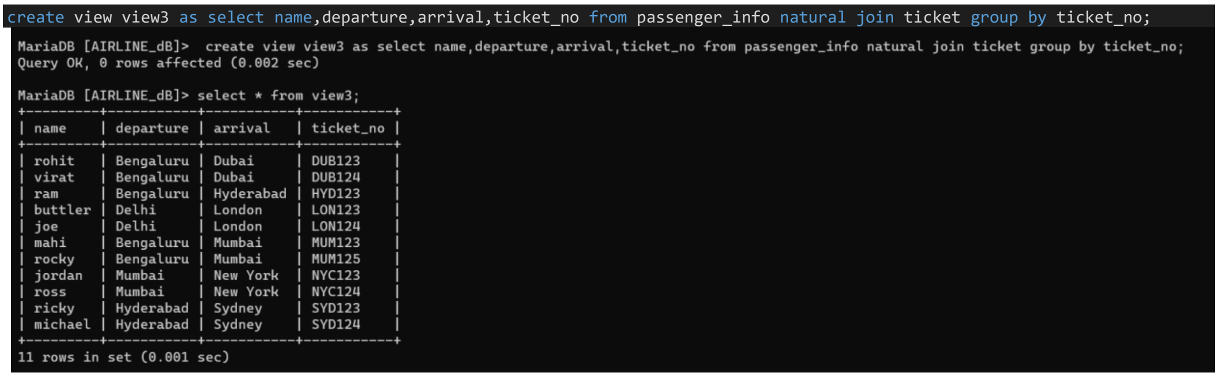
```
MariaDB [airline_Db]> select name from user natural join ticket where departure='Bengaluru' and arrival='Dubai' intersec
t  select name from user natural join ticket where departure='Bengaluru' and arrival='Mumbai';
+------+
| name |
+------+
| ram  |
+------+
1 row in set (0.001 sec)

MariaDB [airline_Db]>
```

View (atleast 1)

Demonstrate creation and querying one view

This view contains ticket_no , departure,arrival and passenger details

```
create view view3 as select name,departure,arrival,ticket_no from passenger_info natural join ticket group by ticket_no;
```



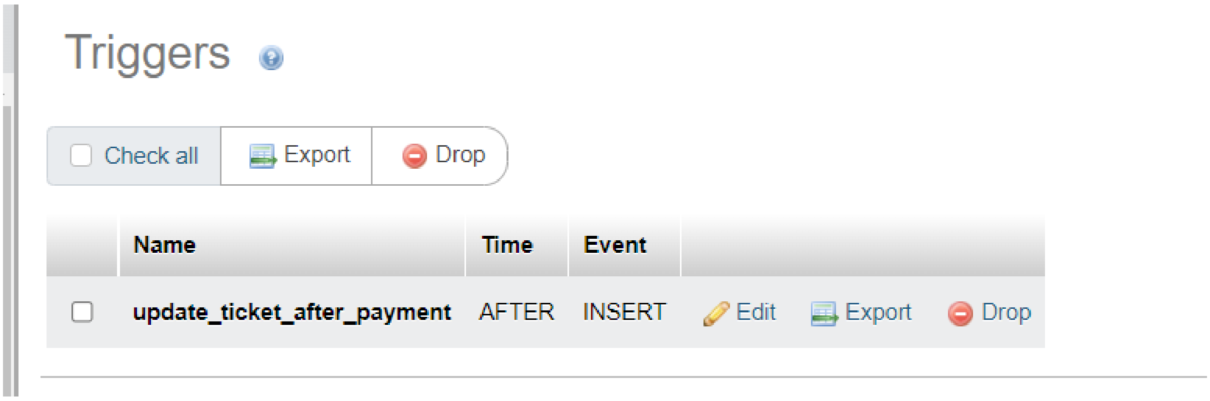
```
MariaDB [AIRLINE_dB]>  create view view3 as select name,departure,arrival,ticket_no from passenger_info natural join ticket group by ticket_no;
Query OK, 0 rows affected (0.002 sec)

MariaDB [AIRLINE_dB]> select * from view3;
+---------+-----------+-----------+-----------+
| name    | departure | arrival   | ticket_no |
+---------+-----------+-----------+-----------+
| rohit   | Bengaluru | Dubai     | DUB123    |
| virat   | Bengaluru | Dubai     | DUB124    |
| ram     | Bengaluru | Hyderabad | HYD123    |
| buttler | Delhi     | London    | LON123    |
| joe     | Delhi     | London    | LON124    |
| mahi    | Bengaluru | Mumbai    | MUM123    |
| rocky   | Bengaluru | Mumbai    | MUM125    |
| jordan  | Mumbai    | New York  | NYC123    |
| ross    | Mumbai    | New York  | NYC124    |
| ricky   | Hyderabad | Sydney    | SYD123    |
| michael | Hyderabad | Sydney    | SYD124    |
+---------+-----------+-----------+-----------+
11 rows in set (0.001 sec)
```

Triggers (Functions or Procedures)

Create a Function or a Procedure. State the objective of the function / Procedure. Run and display the results.

Trigger 1

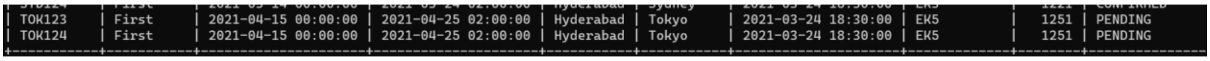
```
--trigger

DELIMITER $$
CREATE TRIGGER `update_ticket_after_payment` AFTER INSERT ON `payment_info`
 FOR EACH ROW UPDATE ticket
     SET ticket_status='CONFIRMED'
   WHERE ticket_no = NEW.ticket_no
DELIMITER ;
```



	Name	Time	Event			
☐	update_ticket_after_payment	AFTER	INSERT	Edit	Export	Drop

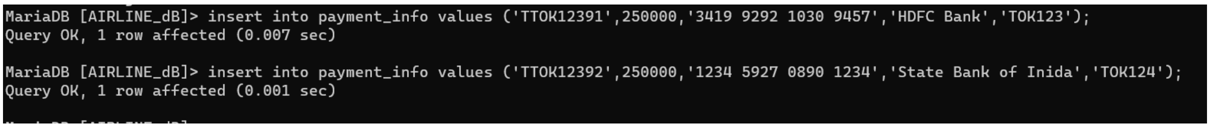
After adding details to payment table this trigger changes ticket_status from PENDING to CONFIRMED.

Before adding into payment

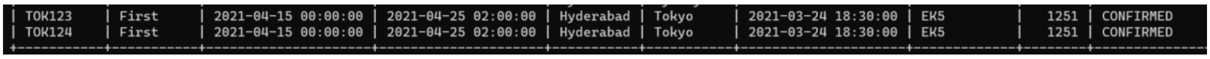


```
| TOK123     | First     | 2021-04-15 00:00:00 | 2021-04-25 02:00:00 | Hyderabad | Tokyo     | 2021-03-24 18:30:00 | EK5          |     1251 | PENDING
| TOK124     | First     | 2021-04-15 00:00:00 | 2021-04-25 02:00:00 | Hyderabad | Tokyo     | 2021-03-24 18:30:00 | EK5          |     1251 | PENDING
+-----------+----------+--------------------+--------------------+-----------+----------+--------------------+-------------+---------+------------
```

After adding into payment

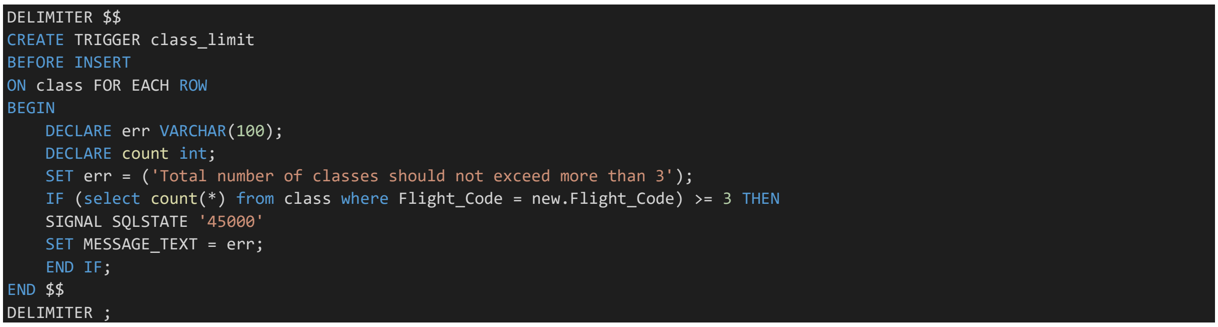


```
MariaDB [AIRLINE_dB]> insert into payment_info values ('TTOK12391',250000,'3419 9292 1030 9457','HDFC Bank','TOK123');
Query OK, 1 row affected (0.007 sec)

MariaDB [AIRLINE_dB]> insert into payment_info values ('TTOK12392',250000,'1234 5927 0890 1234','State Bank of Inida','TOK124');
Query OK, 1 row affected (0.001 sec)
```



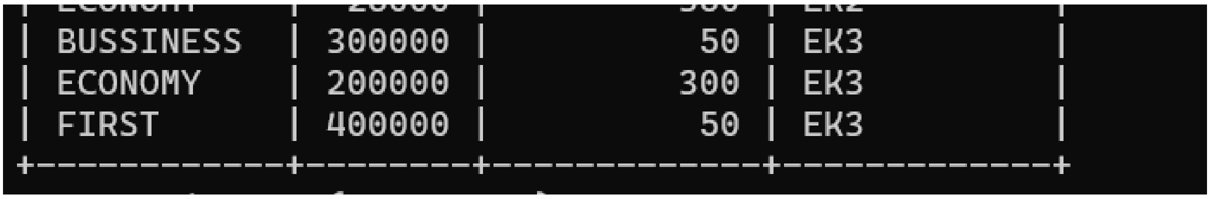
```
| TOK123     | First     | 2021-04-15 00:00:00 | 2021-04-25 02:00:00 | Hyderabad | Tokyo     | 2021-03-24 18:30:00 | EK5          |     1251 | CONFIRMED
| TOK124     | First     | 2021-04-15 00:00:00 | 2021-04-25 02:00:00 | Hyderabad | Tokyo     | 2021-03-24 18:30:00 | EK5          |     1251 | CONFIRMED
+-----------+----------+--------------------+--------------------+-----------+----------+--------------------+-------------+---------+------------
```

Trigger 2

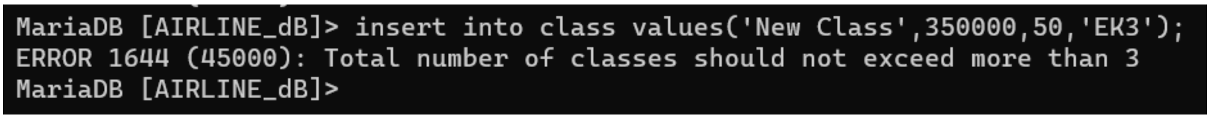


```
DELIMITER $$
CREATE TRIGGER class_limit
BEFORE INSERT
ON class FOR EACH ROW
BEGIN
    DECLARE err VARCHAR(100);
    DECLARE count int;
    SET err = ('Total number of classes should not exceed more than 3');
    IF (select count(*) from class where Flight_Code = new.Flight_Code) >= 3 THEN
    SIGNAL SQLSTATE '45000'
    SET MESSAGE_TEXT = err;
    END IF;
END $$
DELIMITER ;
```

This Trigger wouldn't allow adding more than 3 classes for a flight



```
| BUSSINESS   | 300000 |           50 | EK3          |
| ECONOMY     | 200000 |          300 | EK3          |
| FIRST       | 400000 |           50 | EK3          |
+-------------+--------+--------------+--------------+
```

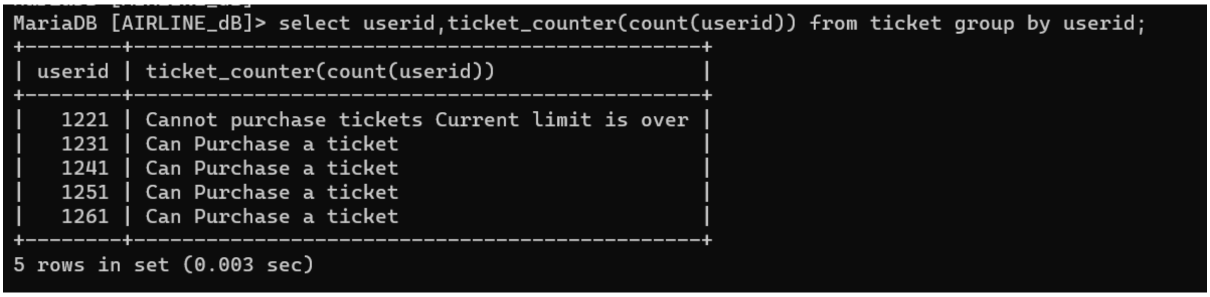


```
MariaDB [AIRLINE_dB]> insert into class values('New Class',350000,50,'EK3');
ERROR 1644 (45000): Total number of classes should not exceed more than 3
MariaDB [AIRLINE_dB]>
```

Function

```
DELIMITER $$
CREATE FUNCTION ticket_counter(count int)
RETURNS VARCHAR(50)
BEGIN
DECLARE message varchar(50);
IF count>=5 then
set message="Cannot purchase tickets Current limit is over";
ELSE
set message ="Can Purchase a ticket";
end if;
return message;
END $$
DELIMITER;
```

This function shows whether user can buy a ticket or not if he purchased more than 5 tickets this function returns you can't purchase any more or else it'll show you can purchase

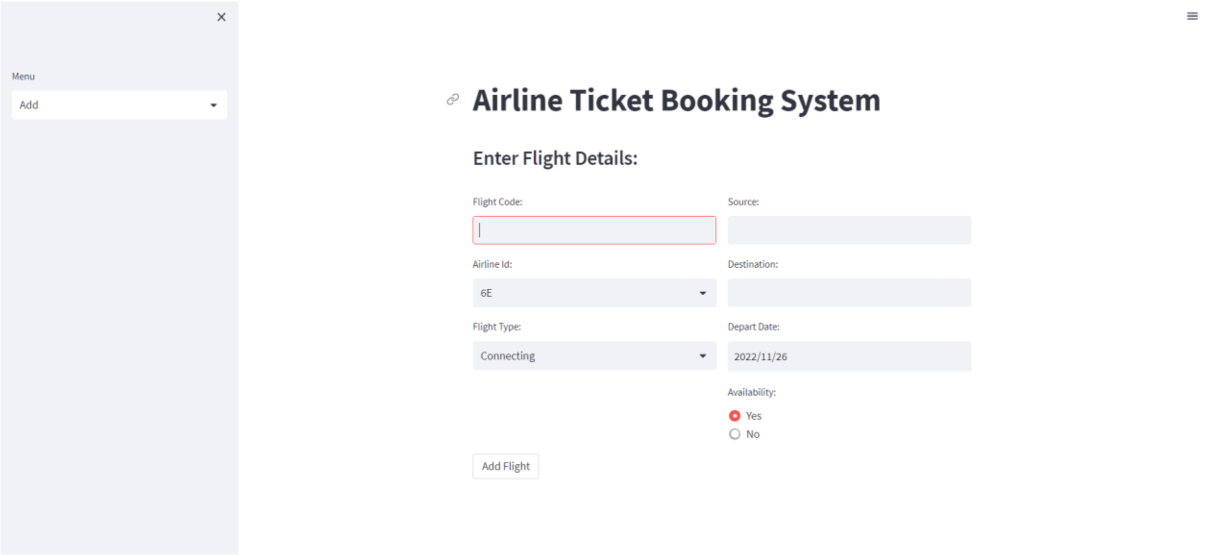


```
MariaDB [AIRLINE_dB]> select userid,ticket_counter(count(userid)) from ticket group by userid;
+--------+-----------------------------------------------+
| userid | ticket_counter(count(userid))                 |
+--------+-----------------------------------------------+
|   1221 | Cannot purchase tickets Current limit is over |
|   1231 | Can Purchase a ticket                         |
|   1241 | Can Purchase a ticket                         |
|   1251 | Can Purchase a ticket                         |
|   1261 | Can Purchase a ticket                         |
+--------+-----------------------------------------------+
5 rows in set (0.003 sec)
```

Developing a Frontend

The frontend should support

1. Addition, Modification and Deletion of records from any chosen table



Menu: Add

Airline Ticket Booking System

Enter Flight Details:

Flight Code:

Source:

Airline Id: 6E

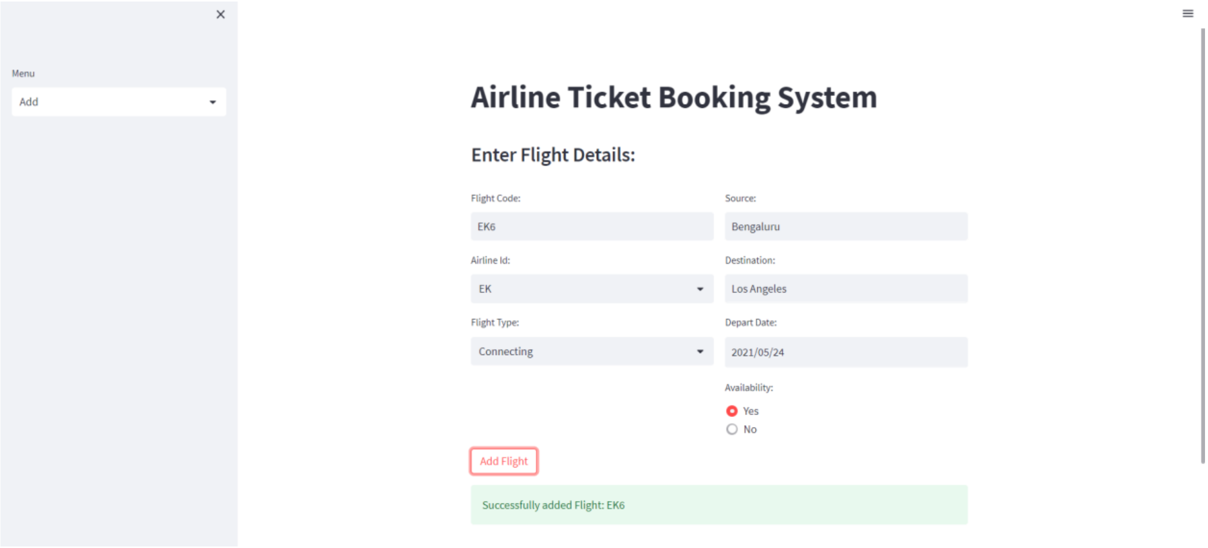
Destination:

Flight Type: Connecting

Depart Date: 2022/11/26

Availability:
● Yes
○ No

Insertion



Airline Ticket Booking System

Enter Flight Details:

Flight Code: EK6

Source: Bengaluru

Airline Id: EK

Destination: Los Angeles

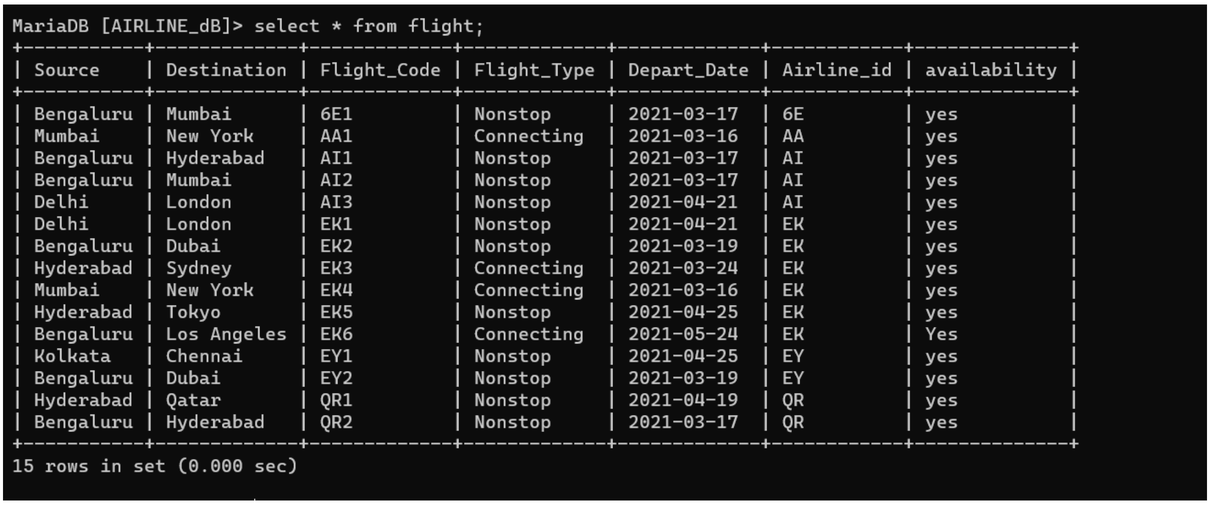
Flight Type: Connecting

Depart Date: 2021/05/24

Availability:
● Yes
○ No

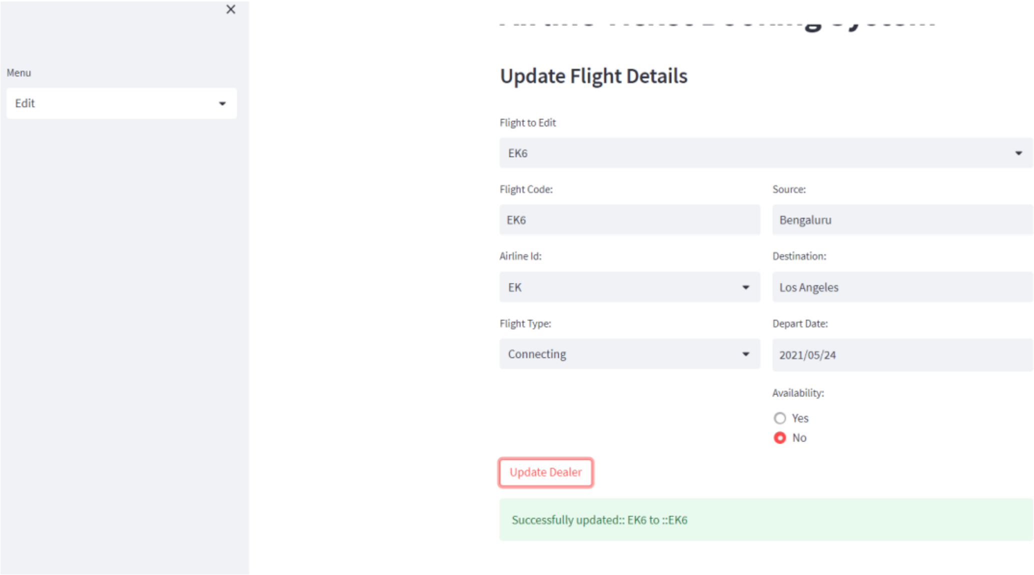
Successfully added Flight: EK6

We can see Bengaluru to Los Angeles flight has been added to the table



```
MariaDB [AIRLINE_dB]> select * from flight;
+-----------+-------------+-------------+-------------+-------------+------------+--------------+
| Source    | Destination | Flight_Code | Flight_Type | Depart_Date | Airline_id | availability |
+-----------+-------------+-------------+-------------+-------------+------------+--------------+
| Bengaluru | Mumbai      | 6E1         | Nonstop     | 2021-03-17  | 6E         | yes          |
| Mumbai    | New York    | AA1         | Connecting  | 2021-03-16  | AA         | yes          |
| Bengaluru | Hyderabad   | AI1         | Nonstop     | 2021-03-17  | AI         | yes          |
| Bengaluru | Mumbai      | AI2         | Nonstop     | 2021-03-17  | AI         | yes          |
| Delhi     | London      | AI3         | Nonstop     | 2021-04-21  | AI         | yes          |
| Delhi     | London      | EK1         | Nonstop     | 2021-04-21  | EK         | yes          |
| Bengaluru | Dubai       | EK2         | Nonstop     | 2021-03-19  | EK         | yes          |
| Hyderabad | Sydney      | EK3         | Connecting  | 2021-03-24  | EK         | yes          |
| Mumbai    | New York    | EK4         | Connecting  | 2021-03-16  | EK         | yes          |
| Hyderabad | Tokyo       | EK5         | Nonstop     | 2021-04-25  | EK         | yes          |
| Bengaluru | Los Angeles | EK6         | Connecting  | 2021-05-24  | EK         | Yes          |
| Kolkata   | Chennai     | EY1         | Nonstop     | 2021-04-25  | EY         | yes          |
| Bengaluru | Dubai       | EY2         | Nonstop     | 2021-03-19  | EY         | yes          |
| Hyderabad | Qatar       | QR1         | Nonstop     | 2021-04-19  | QR         | yes          |
| Bengaluru | Hyderabad   | QR2         | Nonstop     | 2021-03-17  | QR         | yes          |
+-----------+-------------+-------------+-------------+-------------+------------+--------------+
15 rows in set (0.000 sec)
```

Updation



Update Flight Details

Flight to Edit: EK6

Flight Code: EK6

Source: Bengaluru

Airline Id: EK

Destination: Los Angeles

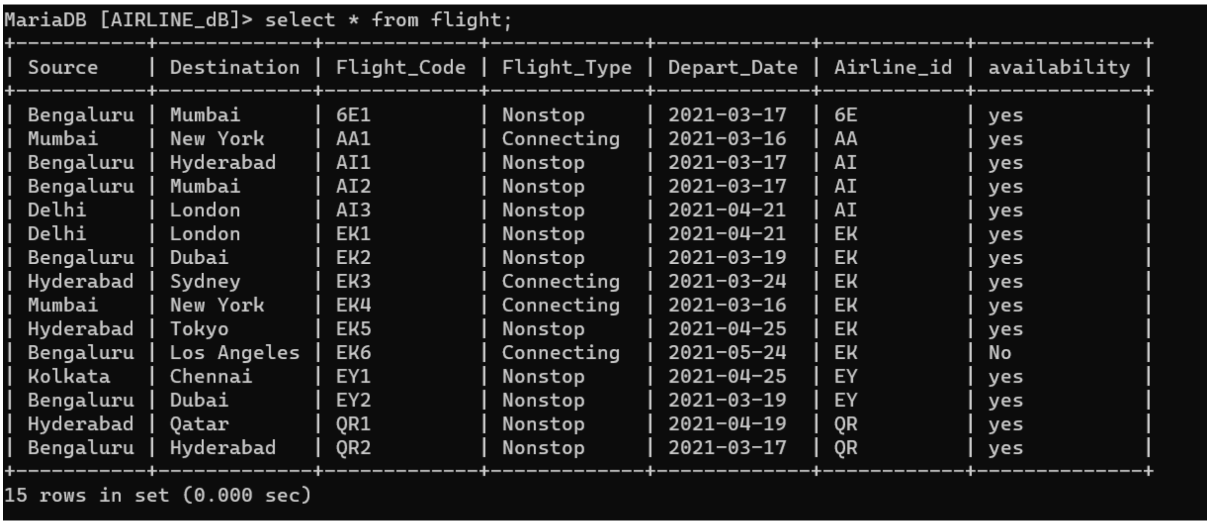
Flight Type: Connecting

Depart Date: 2021/05/24

Availability:
○ Yes
● No

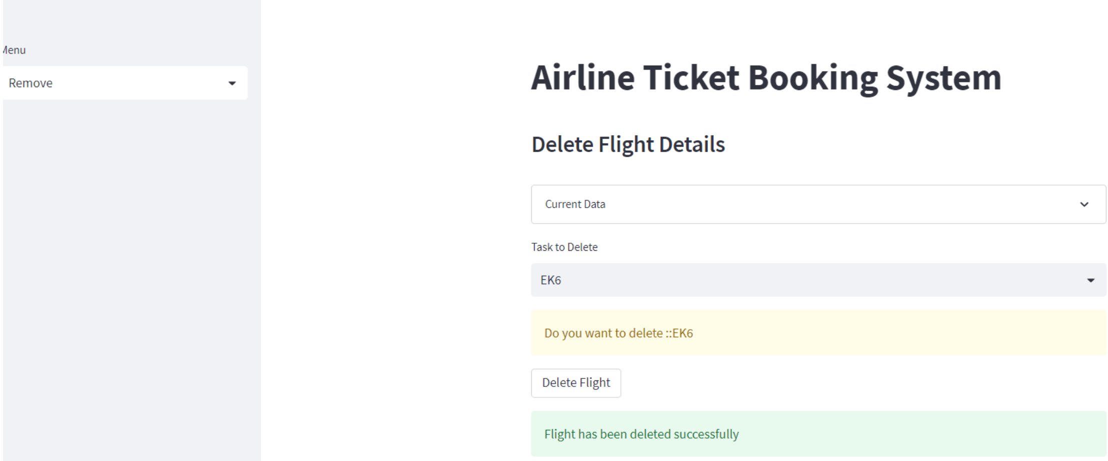
Successfully updated:: EK6 to ::EK6

We can see availability of flight from Bengaluru to Los Angeles has been updated to No



```
MariaDB [AIRLINE_dB]> select * from flight;
+-----------+-------------+-------------+-------------+-------------+------------+--------------+
| Source    | Destination | Flight_Code | Flight_Type | Depart_Date | Airline_id | availability |
+-----------+-------------+-------------+-------------+-------------+------------+--------------+
| Bengaluru | Mumbai      | 6E1         | Nonstop     | 2021-03-17  | 6E         | yes          |
| Mumbai    | New York    | AA1         | Connecting  | 2021-03-16  | AA         | yes          |
| Bengaluru | Hyderabad   | AI1         | Nonstop     | 2021-03-17  | AI         | yes          |
| Bengaluru | Mumbai      | AI2         | Nonstop     | 2021-03-17  | AI         | yes          |
| Delhi     | London      | AI3         | Nonstop     | 2021-04-21  | AI         | yes          |
| Delhi     | London      | EK1         | Nonstop     | 2021-04-21  | EK         | yes          |
| Bengaluru | Dubai       | EK2         | Nonstop     | 2021-03-19  | EK         | yes          |
| Hyderabad | Sydney      | EK3         | Connecting  | 2021-03-24  | EK         | yes          |
| Mumbai    | New York    | EK4         | Connecting  | 2021-03-16  | EK         | yes          |
| Hyderabad | Tokyo       | EK5         | Nonstop     | 2021-04-25  | EK         | yes          |
| Bengaluru | Los Angeles | EK6         | Connecting  | 2021-05-24  | EK         | No           |
| Kolkata   | Chennai     | EY1         | Nonstop     | 2021-04-25  | EY         | yes          |
| Bengaluru | Dubai       | EY2         | Nonstop     | 2021-03-19  | EY         | yes          |
| Hyderabad | Qatar       | QR1         | Nonstop     | 2021-04-19  | QR         | yes          |
| Bengaluru | Hyderabad   | QR2         | Nonstop     | 2021-03-17  | QR         | yes          |
+-----------+-------------+-------------+-------------+-------------+------------+--------------+
15 rows in set (0.000 sec)
```

Deletion



Menu: Remove

Airline Ticket Booking System

Delete Flight Details

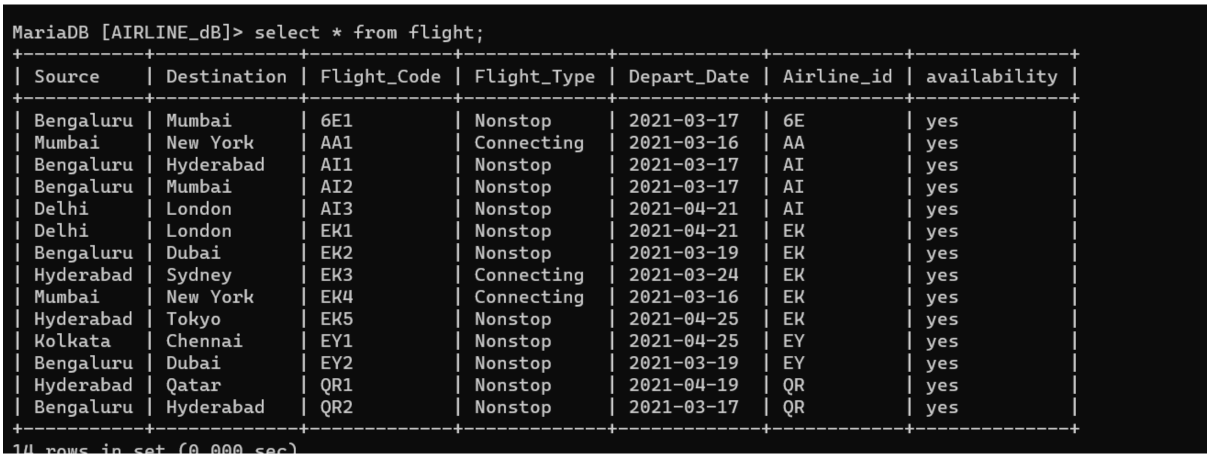
Current Data

Task to Delete: EK6

Do you want to delete ::EK6

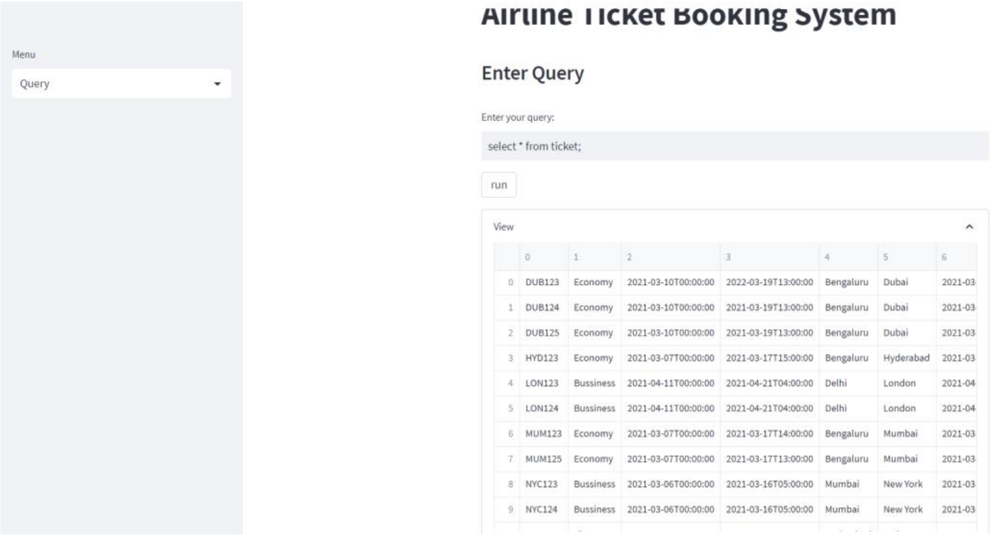
Flight has been deleted successfully

We can see Bengaluru to Los Angeles flight has been deleted in the database



```
MariaDB [AIRLINE_dB]> select * from flight;
+-----------+-------------+-------------+-------------+-------------+------------+--------------+
| Source    | Destination | Flight_Code | Flight_Type | Depart_Date | Airline_id | availability |
+-----------+-------------+-------------+-------------+-------------+------------+--------------+
| Bengaluru | Mumbai      | 6E1         | Nonstop     | 2021-03-17  | 6E         | yes          |
| Mumbai    | New York    | AA1         | Connecting  | 2021-03-16  | AA         | yes          |
| Bengaluru | Hyderabad   | AI1         | Nonstop     | 2021-03-17  | AI         | yes          |
| Bengaluru | Mumbai      | AI2         | Nonstop     | 2021-03-17  | AI         | yes          |
| Delhi     | London      | AI3         | Nonstop     | 2021-04-21  | AI         | yes          |
| Delhi     | London      | EK1         | Nonstop     | 2021-04-21  | EK         | yes          |
| Bengaluru | Dubai       | EK2         | Nonstop     | 2021-03-19  | EK         | yes          |
| Hyderabad | Sydney      | EK3         | Connecting  | 2021-03-24  | EK         | yes          |
| Mumbai    | New York    | EK4         | Connecting  | 2021-03-16  | EK         | yes          |
| Hyderabad | Tokyo       | EK5         | Nonstop     | 2021-04-25  | EK         | yes          |
| Kolkata   | Chennai     | EY1         | Nonstop     | 2021-04-25  | EY         | yes          |
| Bengaluru | Dubai       | EY2         | Nonstop     | 2021-03-19  | EY         | yes          |
| Hyderabad | Qatar       | QR1         | Nonstop     | 2021-04-19  | QR         | yes          |
| Bengaluru | Hyderabad   | QR2         | Nonstop     | 2021-03-17  | QR         | yes          |
+-----------+-------------+-------------+-------------+-------------+------------+--------------+
14 rows in set (0.000 sec)
```

2. There should be a window to accept and run any SQL statement and display the result



Menu: Query

Airline Ticket Booking System

Enter Query

Enter your query:

```
select * from ticket;
```

	0	1	2	3	4	5	6
0	DUB123	Economy	2021-03-10T00:00:00	2022-03-19T13:00:00	Bengaluru	Dubai	2021-03
1	DUB124	Economy	2021-03-10T00:00:00	2021-03-19T13:00:00	Bengaluru	Dubai	2021-03
2	DUB125	Economy	2021-03-10T00:00:00	2021-03-19T13:00:00	Bengaluru	Dubai	2021-03
3	HYD123	Economy	2021-03-07T00:00:00	2021-03-17T15:00:00	Bengaluru	Hyderabad	2021-03
4	LON123	Bussiness	2021-04-11T00:00:00	2021-04-21T04:00:00	Delhi	London	2021-04
5	LON124	Bussiness	2021-04-11T00:00:00	2021-04-21T04:00:00	Delhi	London	2021-04
6	MUM123	Economy	2021-03-07T00:00:00	2021-03-17T14:00:00	Bengaluru	Mumbai	2021-03
7	MUM125	Economy	2021-03-07T00:00:00	2021-03-17T13:00:00	Bengaluru	Mumbai	2021-03
8	NYC123	Bussiness	2021-03-06T00:00:00	2021-03-16T05:00:00	Mumbai	New York	2021-03
9	NYC124	Bussiness	2021-03-06T00:00:00	2021-03-16T05:00:00	Mumbai	New York	2021-03