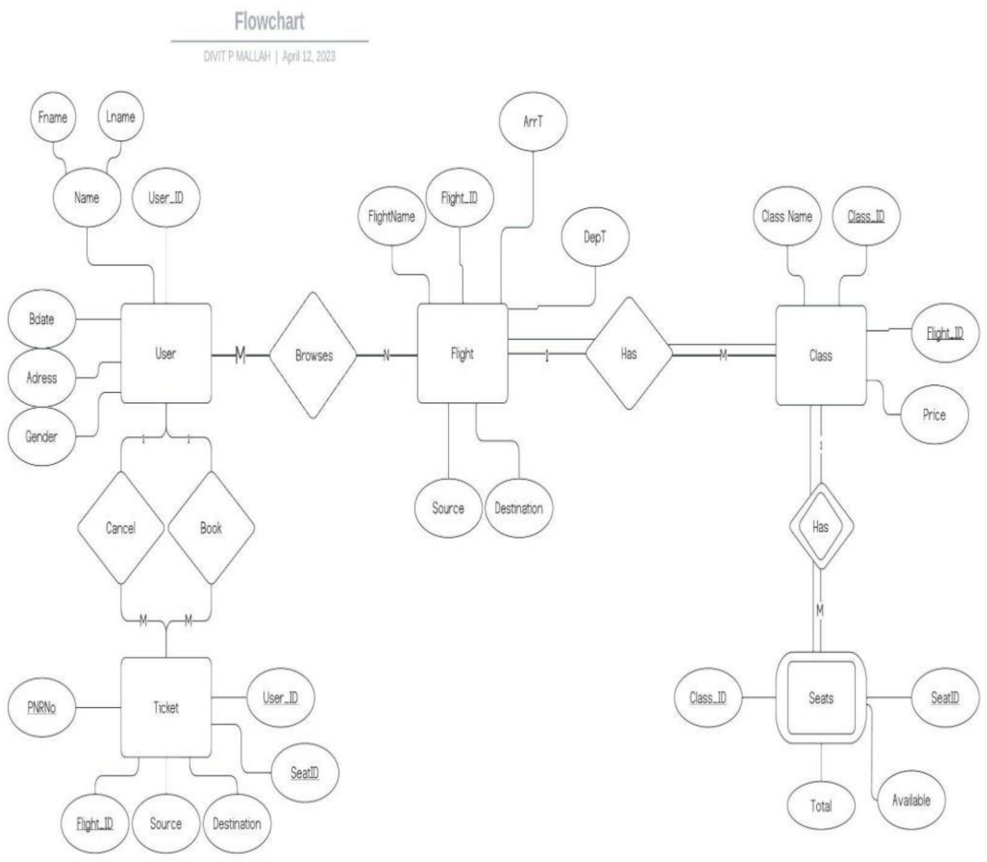
Database Systems Project
Airline Ticket Booking System

Documentation and SQL Code

1. E-R Diagram



In this E-R diagram, we have 5 entities namely User, Flight, Class, Ticket and Seats. The relationship between the entities are:

- Users-Browse-Flights, which is a many-to-many relationship, as multiple users can browse a flight and a user can browse multiple flights.

- Flight-Has-Class, which is a one-to-many relation, as class can only belong to one flight but a flight can have many classes. It is also a total participation relation fromboth ends.
- Class-Has-Seats, which is a one-to-many relationship, asa seat can only belong to one class, but a class can have many seats. It is also a weak relationship as seats dont have any meaning without class.
- User-Book-Ticket, which is a one-to-many relationship asa User can book many tickets, but a ticket can only belong to one user. Same is the case with User-Cancel-Ticket.

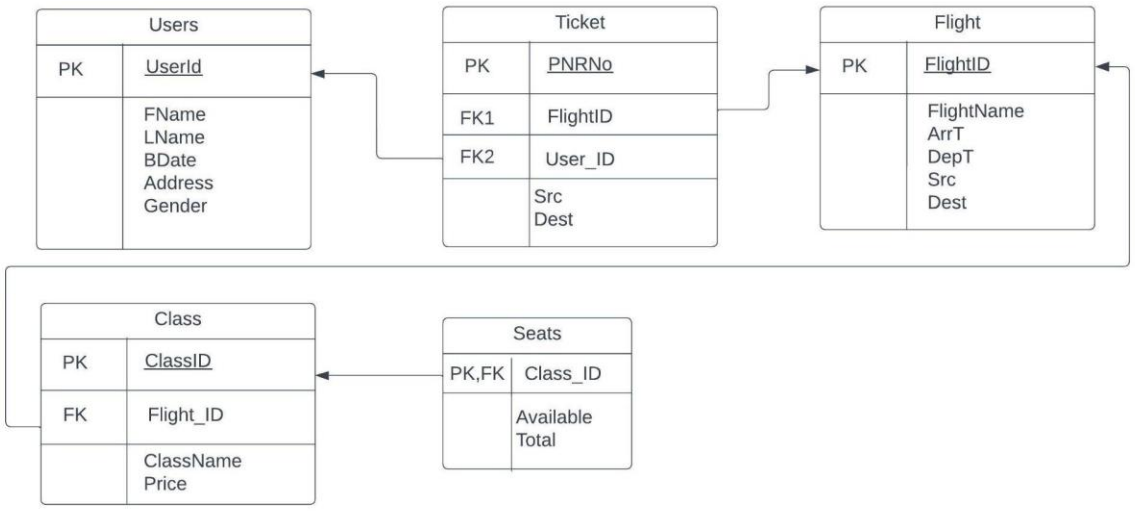
<u>Primary Keys:</u>

- User: User_ID
- Flight: Flight_ID
- Ticket: PNRNo.
- Class: Class_ID

<u>Foreign Keys:</u>

- Class has FK Flight_ID which corresponds to Flight's Primary Key.
- Seats has FK Class_ID which corresponds to Class's Primary Key.
- Ticket has FKs Flight_ID,Seat_ID,User_ID.

2. Conversion and normalization of relational schema



This is the Relational Schema corresponding to the E-Rdiagram.

As we can see the given table is already in 1NF as all the tables have primary keys and all attributes are atomic and non-multivalued.

This table is also in 2NF as there is no partial depenency and there are no combined primary keys (all are only single attributes).

To see 3NF, we need to see transitive dependencies. For the Users table, the first name, last name, bdate,address and

gender are not dependent on any other attributes or each other. In Ticket, User_ID is not dependent on Flight_ID, and source and destianation is dependent on PNRNo. Which is not dependent on any other attribute. So here also 3NF is satisfied. Same is the case with Flights table, where FlightName, Arrival, Departure Time, Source and Destination are not dependent on each other. Same for the Seats table where class_id is PK and Available, Total and Seat_Id are not dependent on each other. Finally, Class is also in 3NF as ClassName and Price are only dependent on ClassID but not each other. This leads us to believe that our Schema is Already in 3NF.
If we dig deep into our table, we realize that all the dependencies that exist are mostly related to the id itself, sothe tables are already in 3NF.

3. Functional dependencies

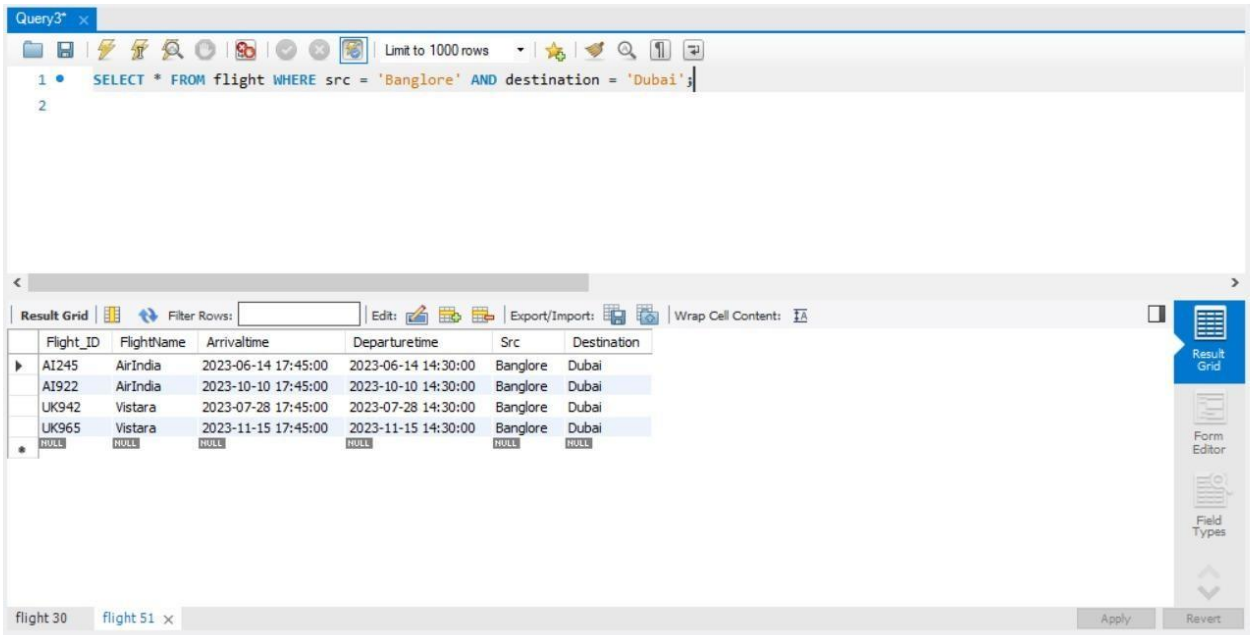
UserID ->Fname,Lname,Bdate,Address,Gender
PNRNo.->Src,Dest
FlightId->FlightName,ArrT,DepT,Src,Dest
ClassId->ClassName,Price

4.SQL Queries

We were asked to answer the following SQL queries:

1. Given the source and destination, what all flights will satisfy the conditon and will be travelling on that particular day?

SQL:**SELECT * FROM flight WHERE src = 'Banglore' AND destination = 'Dubai';**

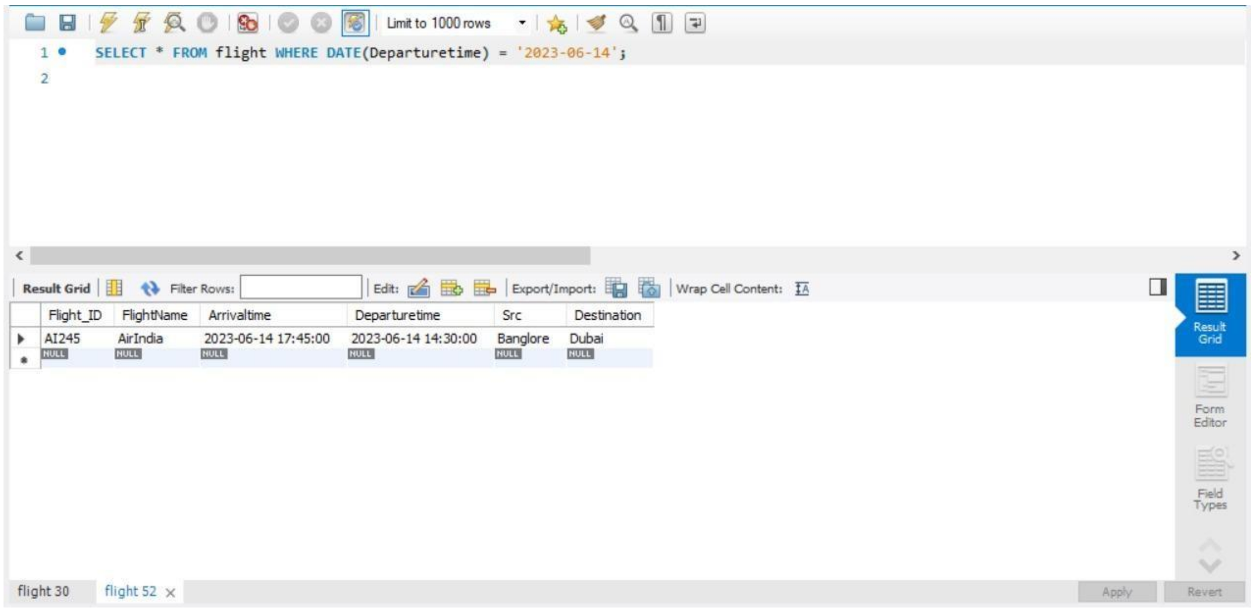


```
SELECT * FROM flight WHERE src = 'Banglore' AND destination = 'Dubai';
```

Flight_ID	FlightName	Arrivaltime	Departuretime	Src	Destination
AI245	AirIndia	2023-06-14 17:45:00	2023-06-14 14:30:00	Banglore	Dubai
AI922	AirIndia	2023-10-10 17:45:00	2023-10-10 14:30:00	Banglore	Dubai
UK942	Vistara	2023-07-28 17:45:00	2023-07-28 14:30:00	Banglore	Dubai
UK965	Vistara	2023-11-15 17:45:00	2023-11-15 14:30:00	Banglore	Dubai
NULL	NULL	NULL	NULL	NULL	NULL

2. Given the date, mention all the flights along with details flowing on that particular day?

SQL:**SELECT * FROM flight WHERE DATE(Departuretime) = '2023-06-14';**

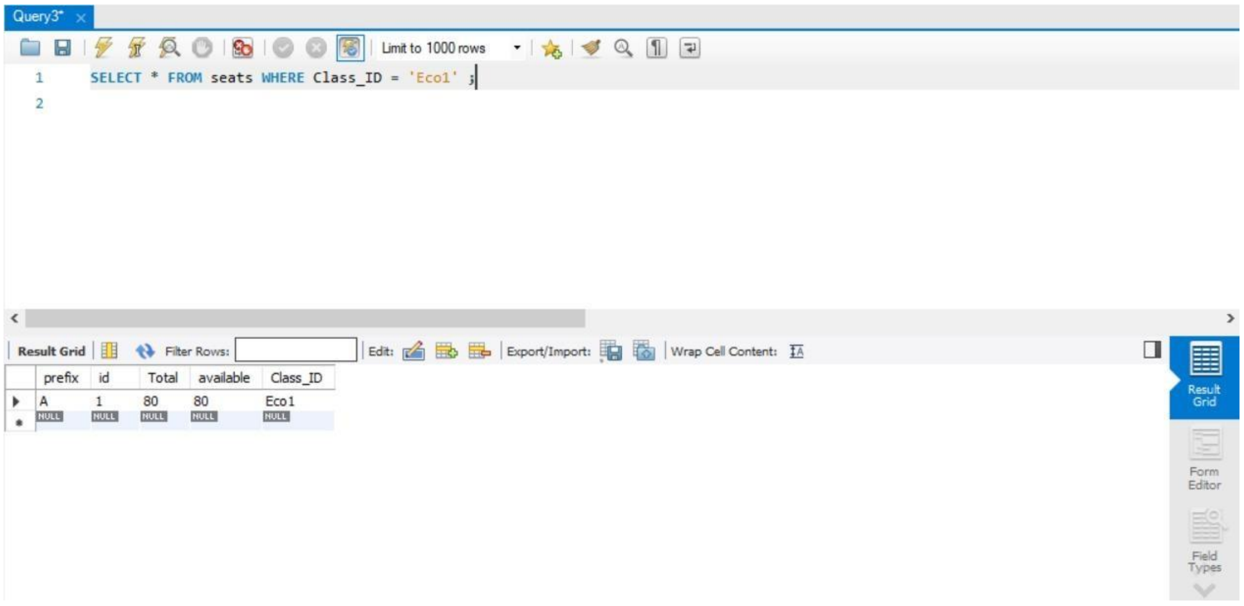


```
SELECT * FROM flight WHERE DATE(Departuretime) = '2023-06-14';
```

Flight_ID	FlightName	Arrivaltime	Departuretime	Src	Destination
AI245	AirIndia	2023-06-14 17:45:00	2023-06-14 14:30:00	Banglore	Dubai
NULL	NULL	NULL	NULL	NULL	NULL

3. A flight selected show all the vacant seats?

SQL:**SELECT * FROM seats WHERE Class_ID = 'Eco1' ;**

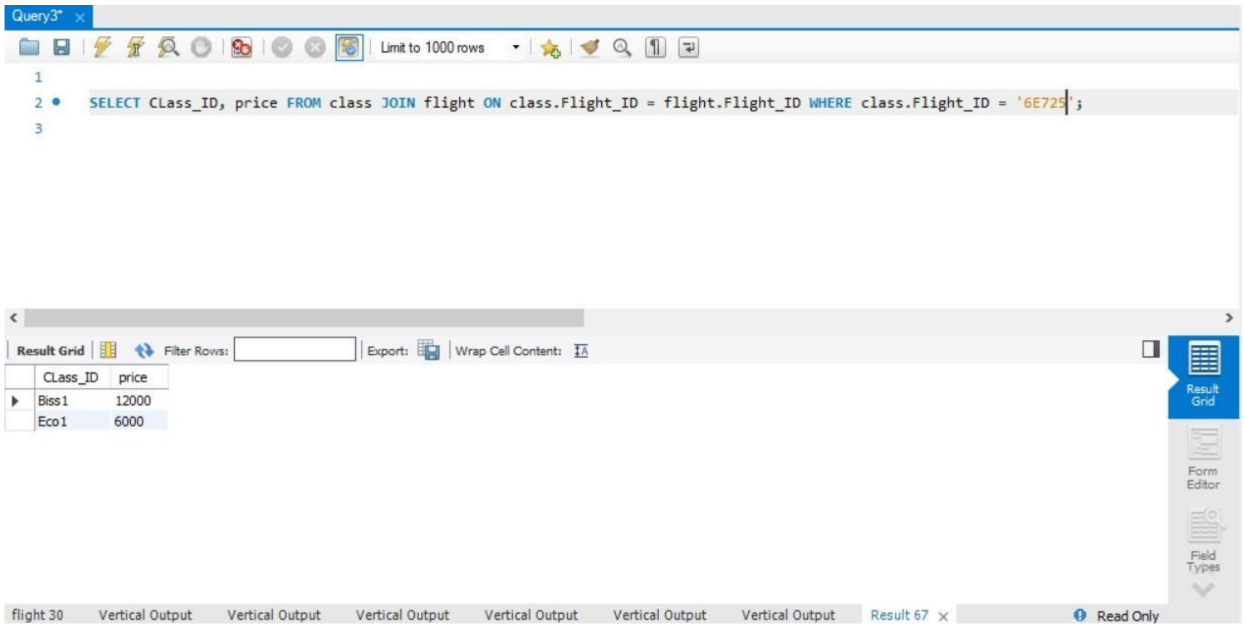


```
SELECT * FROM seats WHERE Class_ID = 'Eco1' ;
```

prefix	id	Total	available	Class_ID
A	1	80	80	Eco1
NULL	NULL	NULL	NULL	NULL

4. Display the price of each seat?

SQL:**SELECT CLass_ID, price FROM class JOIN flight ON class.Flight_ID = flight.Flight_ID WHERE class.Flight_ID = '6E725';**



```
SELECT CLass_ID, price FROM class JOIN flight ON class.Flight_ID = flight.Flight_ID WHERE class.Flight_ID = '6E725';
```

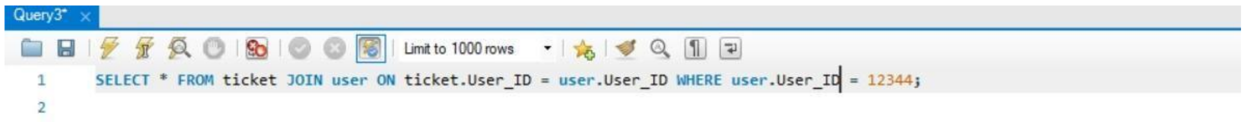
CLass_ID	price
Biss1	12000
Eco1	6000

5. Making sure of concurrency control.

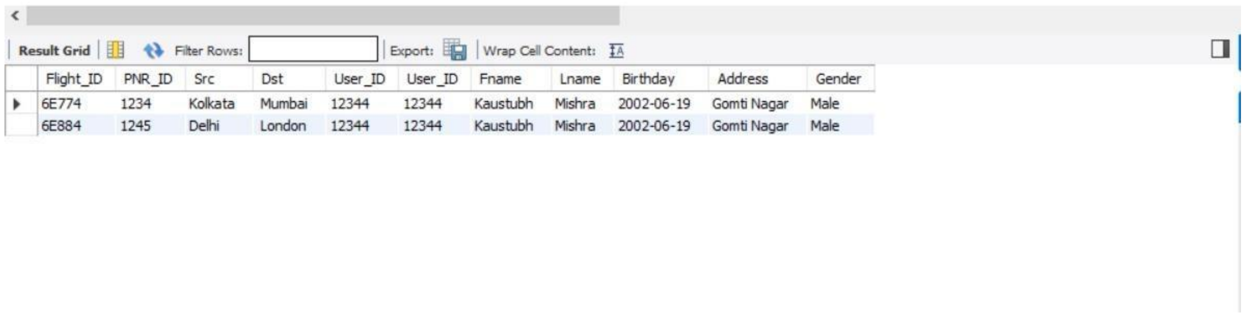
Concurrency control is generally maintained at the database levels by setting appropriate transaction isolation levels. This is done by setting the transaction isolation level to Serializable which ensures that the transactions are executed one after the other.

6. Showing the seats booked by a particular user?

SQL:**SELECT * FROM ticket JOIN user ON ticket.User_ID = user.User_ID WHERE user.User_ID = 12344;**



```
SELECT * FROM ticket JOIN user ON ticket.User_ID = user.User_ID WHERE user.User_ID = 12344;
```

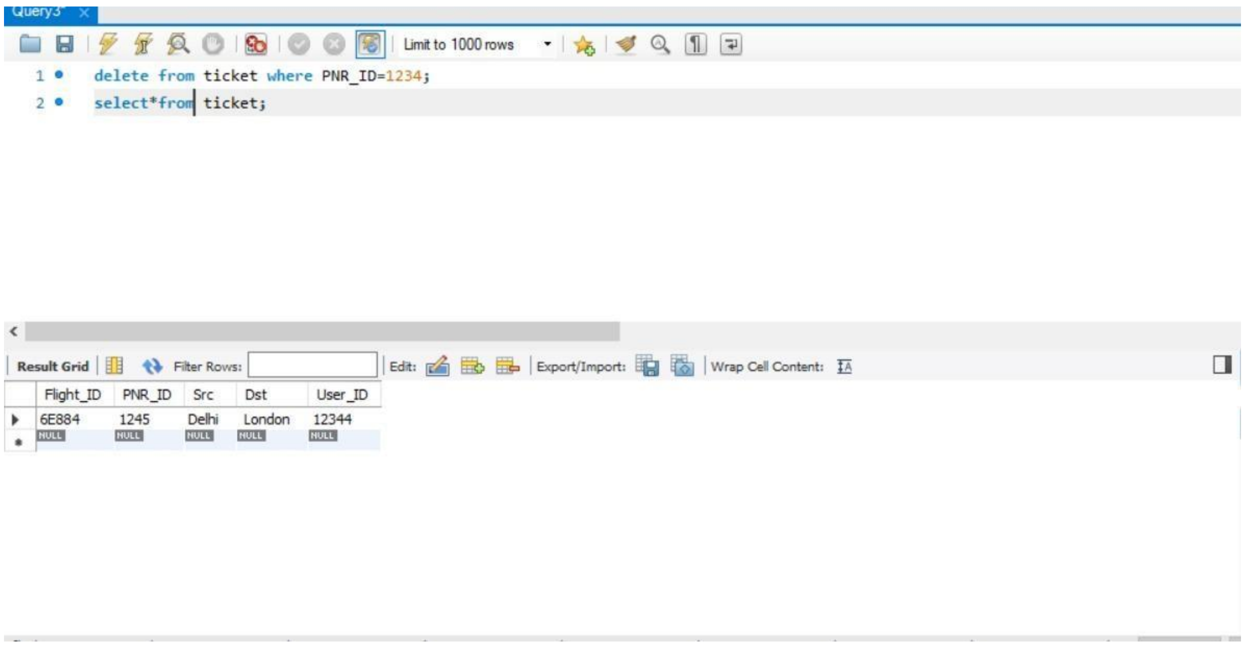


Flight_ID	PNR_ID	Src	Dst	User_ID	User_ID	Fname	Lname	Birthday	Address	Gender
6E774	1234	Kolkata	Mumbai	12344	12344	Kaustubh	Mishra	2002-06-19	Gomti Nagar	Male
6E884	1245	Delhi	London	12344	12344	Kaustubh	Mishra	2002-06-19	Gomti Nagar	Male

7. Cancelling a booked ticket.SQL:

delete from ticket where PNR_ID=1234;

select*from ticket;

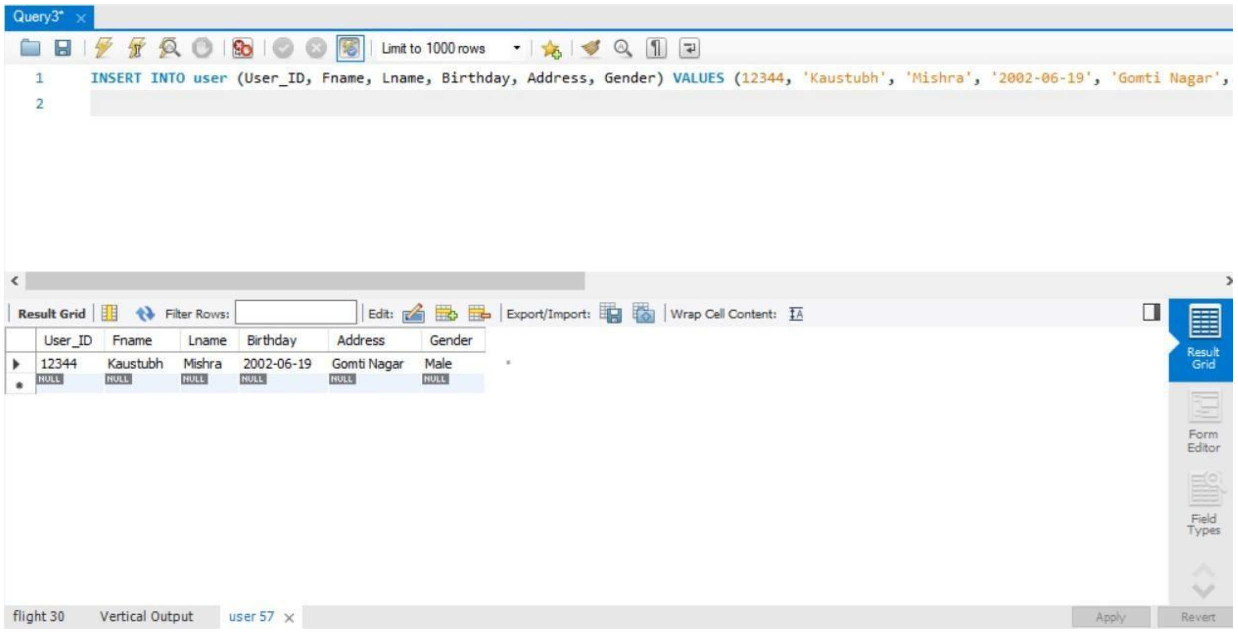


```
delete from ticket where PNR_ID=1234;
select*from ticket;
```

Flight_ID	PNR_ID	Src	Dst	User_ID
6E884	1245	Delhi	London	12344
NULL	NULL	NULL	NULL	NULL

8. Adding a user ID in the system.

SQL:**INSERT INTO user (User_ID, Fname, Lname, Birthday, Address, Gender) VALUES (12344, 'Kaustubh', 'Mishra', '2002-06-19', 'Gomti Nagar', 'Male');**

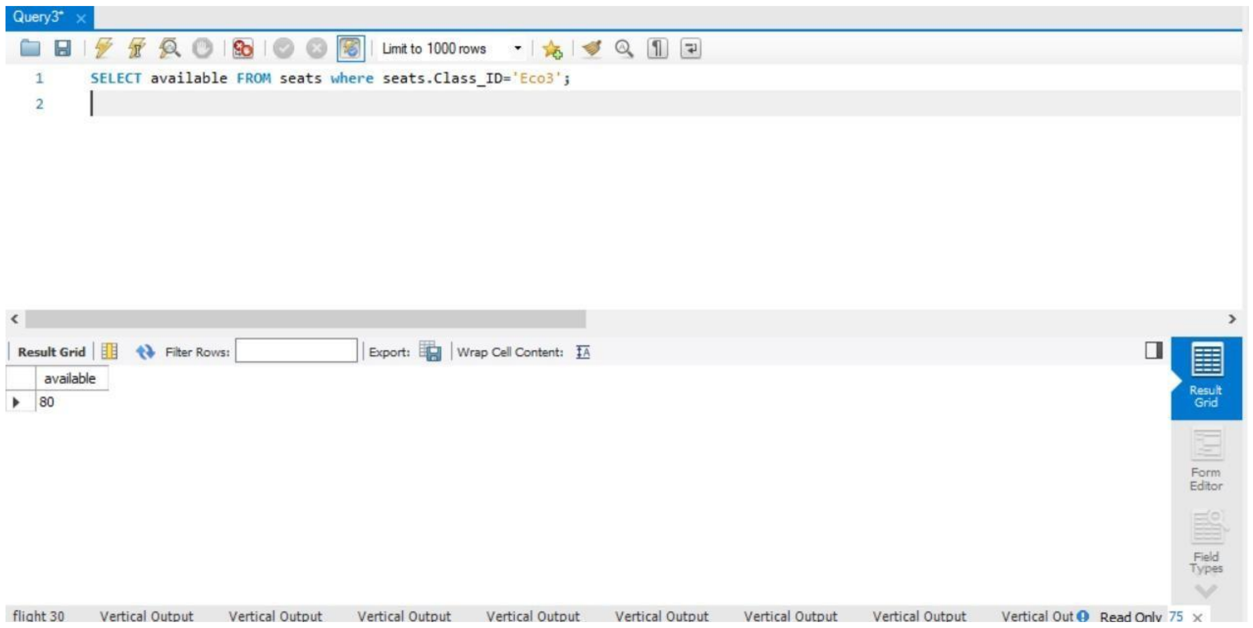


```
INSERT INTO user (User_ID, Fname, Lname, Birthday, Address, Gender) VALUES (12344, 'Kaustubh', 'Mishra', '2002-06-19', 'Gomti Nagar',
```

User_ID	Fname	Lname	Birthday	Address	Gender
12344	Kaustubh	Mishra	2002-06-19	Gomti Nagar	Male
NULL	NULL	NULL	NULL	NULL	NULL

9. Showing the number of available seats in a particular class eg. Number of seats available in XYZ flight in Business class are?

SQL:**SELECT available FROM seats where seats.Class_ID='Eco3';**

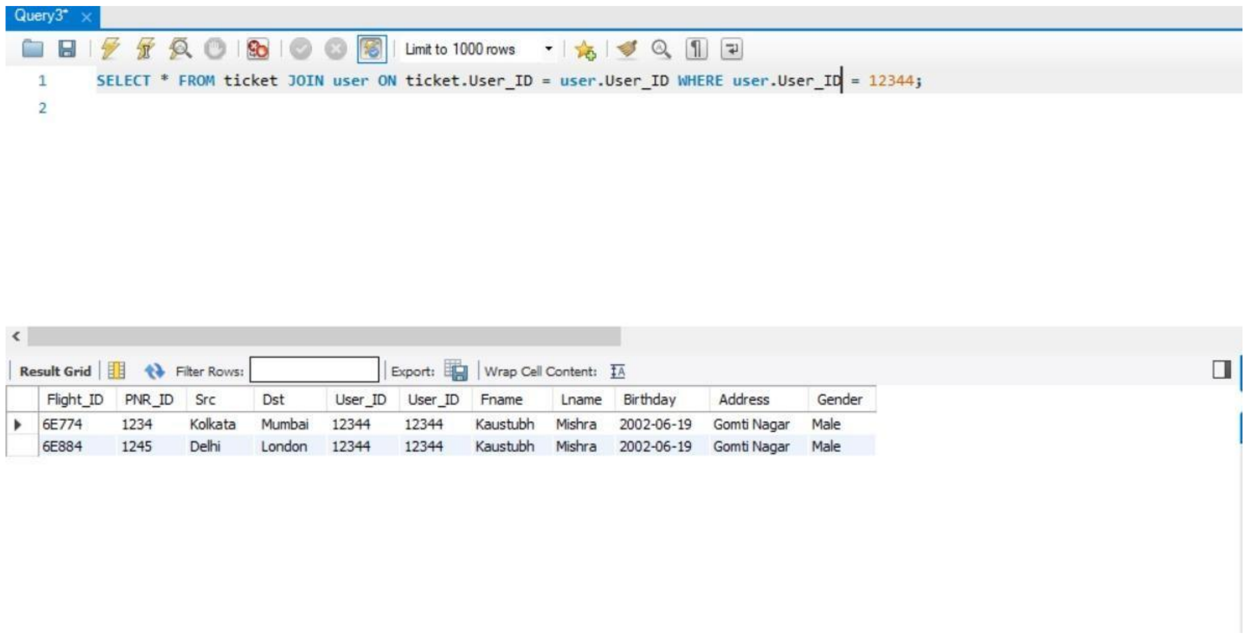


```
SELECT available FROM seats where seats.Class_ID='Eco3';
```

```
available
80
```

10. Confirming that the ticket has been successfully booked by the user.

SQL:**SELECT * FROM ticket JOIN user ON ticket.User_ID = user.User_ID WHERE user.User_ID = 12344;**

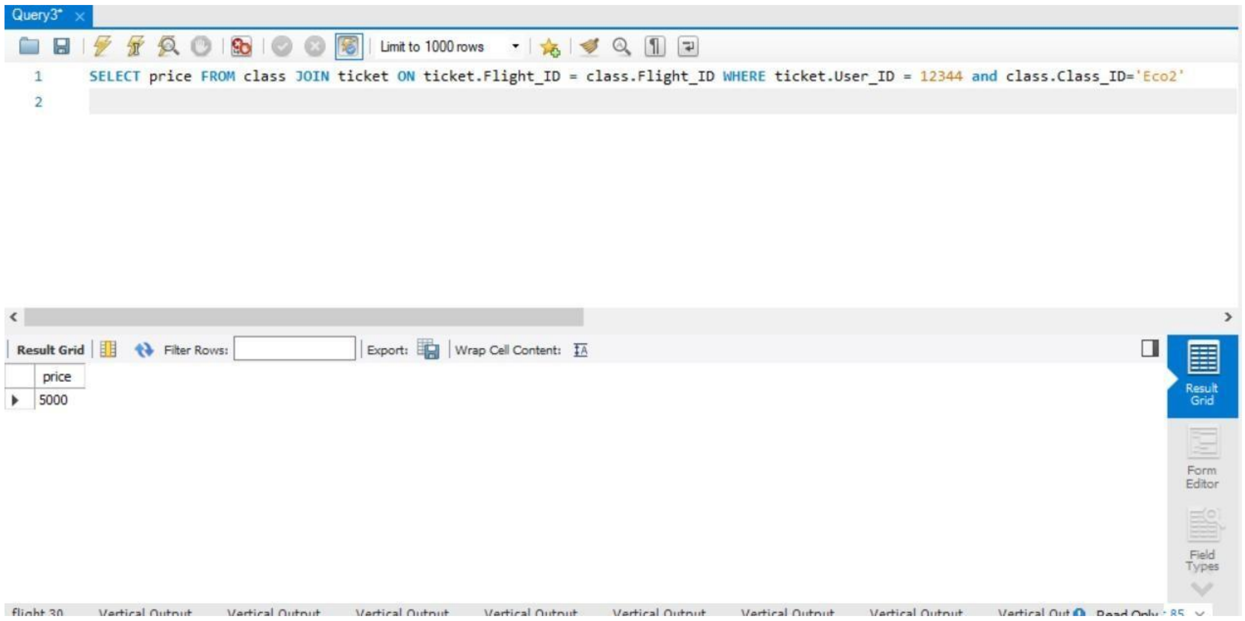


```
SELECT * FROM ticket JOIN user ON ticket.User_ID = user.User_ID WHERE user.User_ID = 12344;
```

Flight_ID	PNR_ID	Src	Dst	User_ID	User_ID	Fname	Lname	Birthday	Address	Gender
6E774	1234	Kolkata	Mumbai	12344	12344	Kaustubh	Mishra	2002-06-19	Gomti Nagar	Male
6E884	1245	Delhi	London	12344	12344	Kaustubh	Mishra	2002-06-19	Gomti Nagar	Male

11. Show the total price of all the seats selected by the user.

SQL:**SELECT price FROM class JOIN ticket ON ticket.Flight_ID = class.Flight_ID WHERE ticket.User_ID = 12344 and class.Class_ID='Eco2'**



```
SELECT price FROM class JOIN ticket ON ticket.Flight_ID = class.Flight_ID WHERE ticket.User_ID = 12344 and class.Class_ID='Eco2'
```

price
5000