



B.TECH. (CSE)

V SEMESTER

UE19CS301 – DATABASE MANAGEMENT SYSTEM

ASSIGNMENT3

TEAM NO: 13

TOPIC: AIRLINE RESERVATION SYSTEM

SUBMITTED BY

NAME

SRN

SAI JAYANTH IMMADISETTY

PES2UG19CS152

AMIT JITTA

PES2UG19CS169

KUNTAL GORAI

PES2UG19CS198

AUGUST– DECEMBER 2021

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

ELECTRONIC CITY CAMPUS,

BENGALURU – 560100, KARNATAKA, INDIA

SIMPLE QUERIES

1. Display all the users who are travelling

```
--Display all the users who are travelling  
SELECT * FROM TRAVELLER;
```

	traveller_id	fname	phone	lname
1	2	AISHA	9654876543	BHATT
2	3	ROHAN	6734568765	SINGH
3	4	ROSHNI	6543567288	SINGH
4	5	ROHIT	9023457890	SHETTY
5	6	MANISHA	3456789012	GUPTA
6	7	ALEX	2197834763	STANDHALL
7	8	SUMAN	3045832732	PAL
8	9	VISHAL	345672891	KUMAR
9	10	RAJ	897679230	KUMAR
10	11	DINESH	9289286555	KUMAR
11	12	GOPAL	9987817932	KUMAR
12	13	NEESHANTH	7654293978	GUPTA
13	14	SURYA	8765293978	GUPTA
14	15	RAHUL	2863564890	VERMA
15	1	ALIA	8129877654	BHATT

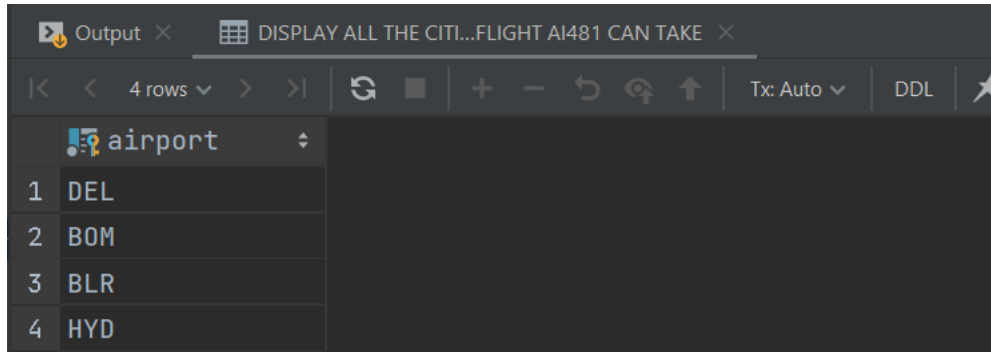
2. Display all names of end users who are agents

```
--DISPLAY ALL NAMES OF END USERS WHO ARE AGENTS  
SELECT FNAME,LNAME FROM END_USER WHERE TYPE_OF_USER='AGENT';
```

	fname	lname
1	ABHINAV	JAISWAL
2	VIJAY	AGGARWAL
3	VISHNU	SAHA
4	KUNAL	ARORA
5	ANJALI	GUPTA

3. Display all the cities where the flight AI481 can take

```
--DISPLAY ALL THE CITIES WHERE THE FLIGHT AI481 CAN TAKE  
SELECT AIRPORT FROM PLANE_PORT_ACCESS WHERE AIRPLANE='AI481';
```

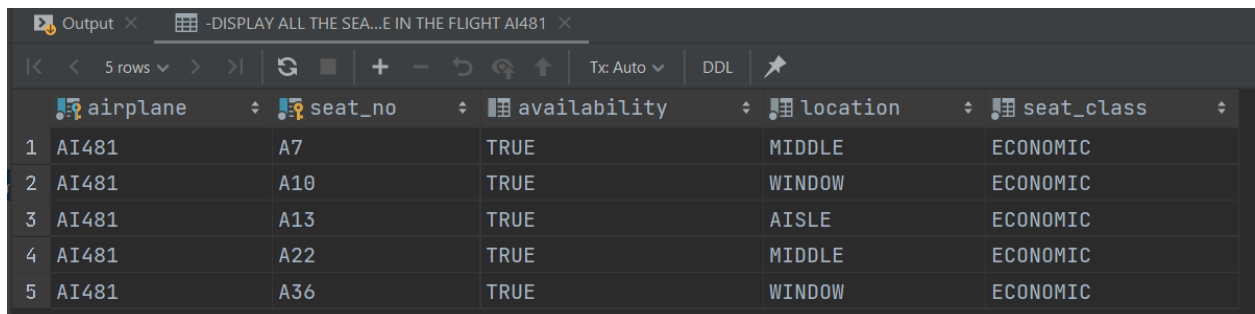


The screenshot shows a database query output window with the title "DISPLAY ALL THE CITIES WHERE THE FLIGHT AI481 CAN TAKE". The output is a table with one column named "airport" and four rows of data.

	airport
1	DEL
2	BOM
3	BLR
4	HYD

4. Display all the seats available in the flight AI481

```
---DISPLAY ALL THE SEATS AVAILABLE IN THE FLIGHT AI481  
SELECT * FROM SEAT WHERE AIRPLANE='AI481';
```



The screenshot shows a database query output window with the title "-DISPLAY ALL THE SEATS AVAILABLE IN THE FLIGHT AI481". The output is a table with six columns: "airplane", "seat_no", "availability", "location", and "seat_class". There are five rows of data.

	airplane	seat_no	availability	location	seat_class
1	AI481	A7	TRUE	MIDDLE	ECONOMIC
2	AI481	A10	TRUE	WINDOW	ECONOMIC
3	AI481	A13	TRUE	AISLE	ECONOMIC
4	AI481	A22	TRUE	MIDDLE	ECONOMIC
5	AI481	A36	TRUE	WINDOW	ECONOMIC

5. Update The ID of the traveller with first name alia with ALIA1708

```
--UPDATE THE ID OF THE TRAVELLER WITH FIRST NAME ALIA WITH ALIA1708  
UPDATE FLIGHT_TRIP  
SET FLIGHT_TRIP_ID='ALIA1708'  
WHERE FLIGHT_TRIP_ID IN  
  (SELECT FLIGHT FROM TRAVELLER_ITENARY WHERE ID IN  
   (SELECT TRAVELLER_ID FROM TRAVELLER WHERE FNAME='ALIA'));  
SELECT * FROM FLIGHT_TRIP;
```

flight_trip_id	no_of_travellers	arrival_airport	arrival_time	depart_airport
13 MANI1807	1	AMD	2021-02-02 08:00:00.000000	BOM
14 VISHAL0201	1	GOI	2021-01-17 22:00:00.000000	DEL
15 ALIA1708	1	BLR	2021-01-09 10:00:00.000000	CCU

Complex /Nested Queries

1. DISPLAY THE NAME OF TRAVELLER WHO HAVE AN ID WITH FLIGHT NAMED ALIA1708

```
--DISPLAY THE NAME OF TRAVELLER WHO HAVE AN ID WITH FLIGHT NAMED ALIA1708
SELECT FNAME, LNAME FROM TRAVELLER WHERE TRAVELLER_ID IN
(SELECT ID FROM TRAVELLER_ITENARY WHERE FLIGHT='ALIA1708');
```

fname	lname
1 ALIA	BHATT

2. DISPLAY ALL TRAVELLERS NAME WHO HAVE AN AGENT

```
--DISPLAY ALL TRAVELLERS NAME WHO HAVE AN AGENT
SELECT FNAME, LNAME FROM TRAVELLER WHERE TRAVELLER_ID IN
(SELECT ID FROM TRAVELLER_ITENARY WHERE FLIGHT IN
(SELECT FLIGHT_TRIP_ID FROM FLIGHT_TRIP WHERE USER_EMAIL IN
(SELECT EMAIL FROM END_USER WHERE TYPE_OF_USER='AGENT'))));
```

fname	lname
1 ALIA	BHATT
2 SURYA	GUPTA
3 VISHAL	KUMAR
4 ROSHNI	SINGH
5 SUMAN	PAL
6 ROHAN	SINGH
7 RAJ	KUMAR
8 AISHA	BHATT
9 ALEX	STANDHALL
10 NEESHANTH	GUPTA

3. DISPLAY THE FARE OF TRAVELLER WHOSE FIRST NAME IS ALIA

```
--DISPLAY THE FARE OF TRAVELLER WHOSE FIRST NAME IS ALIA
SELECT * FROM FARE WHERE FLIGHT_TRIP_ID IN
    (SELECT FLIGHT FROM TRAVELLER_ITENARY WHERE ID IN
        (SELECT TRAVELLER_ID FROM TRAVELLER WHERE FNAME='ALIA'));
```

flight_trip_id	final_amount	amount	currency	discount	tax
1 ALIA1708	5019.18	5121.45	Rupee	512.14	409.87

4. DISPLAY ALL AGENT EMAIL AND COUNT OF NO OF TRAVELLER THE AGENT HAS BOOKED

```
--DISPLAY ALL AGENT EMAIL AND COUNT OF NO OF TRAVELLER THE AGENT HAS BOOKED
SELECT USER_EMAIL, COUNT (NO_OF_TRAVELLERS) FROM FLIGHT_TRIP WHERE USER_EMAIL IN
    (SELECT EMAIL FROM END_USER WHERE TYPE_OF_USER ='AGENT') GROUP BY USER_EMAIL;
```

user_email	count
1 ABHI@GMAIL.COM	2
2 ANJALI@GMAIL.COM	2
3 KUNAL@GMAIL.COM	2
4 VIJAY@GMAIL.COM	2
5 VISHNU@GMAIL.COM	2

5. DISPLAY END USER DETAILS WHO HAS A HOP ID HOP7463

```
--DISPLAY END USER DETAILS WHO HAS A HOP ID HOP7463
SELECT * FROM END_USER WHERE EMAIL IN
    (SELECT USER_EMAIL FROM FLIGHT_TRIP WHERE FLIGHT_TRIP_ID IN
        (SELECT FLIGHT_TRIP_ID FROM FARE WHERE FLIGHT_TRIP_ID IN
            (SELECT FLIGHT_TRIP FROM HAS WHERE HOP='HOP7463')));
```

email	fname	lname	phone	type_of_user
1 VIJAY@GMAIL.COM	VIJAY	AGGARWAL	6734567288	AGENT

CREATION OF MULTIPLE USERS WITH DIFFERENT PRIVILEGES

```
--CREATE USER ADMINS WITH ALL PERMISSIONS  
CREATE USER admin WITH ENCRYPTED PASSWORD 'admin@airline';  
GRANT ALL ON ALL TABLES IN SCHEMA public TO admin;
```

```
--CREATE USER TESTER WITH SELECT,UPDATE,INSERT PERMISSIONS  
CREATE USER tester WITH ENCRYPTED PASSWORD 'tester@airline';  
GRANT SELECT,UPDATE,INSERT ON ALL TABLES IN SCHEMA public TO tester;
```

```
--CREATE USER DEVELOPER WITH SELECT,UPDATE,INSERT,DELETE,CREATE PERMISSIONS  
CREATE USER developer WITH ENCRYPTED PASSWORD 'developer@airline';  
GRANT SELECT,UPDATE,INSERT,DELETE ON ALL TABLES IN SCHEMA public TO developer;
```

```
--CREATE USER USER WITH SELECT PERMISSION  
CREATE USER airline_user WITH PASSWORD 'user@airline';  
GRANT SELECT ON ALL TABLES IN SCHEMA public TO airline_user;
```

CONCURRENCY CONTROL

```
SQL Shell (psql)
You are now connected to database "airline_reservation" as user "postgres".
airline_reservation=# BEGIN;
BEGIN
airline_reservation=#
airline_reservation=# UPDATE FLIGHT_TRIP SET FLIGHT_TRIP_ID='MANISHA2304' WHERE FLIGHT_TRIP_ID IN (SELECT FLIGHT FROM TRAVELLER_ITENARY WHERE ID IN (SELECT TRAVELLER_ID FROM TRAVELLER WHERE FNAME='MANISHA'));
UPDATE 1
airline_reservation=# SELECT * FROM FLIGHT_TRIP;
 flight_trip_id | no_of_travellers | arrival_airport | arrival_time | depart_
airport |      depart_time      |      user_email      |
-----+-----+-----+-----+-----+
AISHA4087 | 1 | BLR | 2021-01-09 10:00:00 | CCU
ROHIT0902 | 1 | BOM | 2021-01-28 10:00:00 | HYD
ALEX2010 | 1 | PAT | 2021-02-02 19:00:00 | COI
SUNAM5678 | 1 | PAT | 2021-02-02 19:00:00 | COI
RAJ2109 | 1 | DEL | 2021-02-02 19:00:00 | BLR
DINESH0306 | 1 | LKO | 2021-01-09 22:00:00 | DEL
GOPAL2512 | 1 | LKO | 2021-01-09 22:00:00 | DEL
NEESH0506 | 1 | BLR | 2021-02-12 23:00:00 | COI
SURYA2310 | 1 | BLR | 2021-02-12 23:00:00 | COI
RAHUL2312 | 1 | BOM | 2021-03-10 23:00:00 | CCU
ROHAN3210 | 1 | BOM | 2021-01-17 10:00:00 | MAA
ROSHNI1709 | 1 | BOM | 2021-01-17 10:00:00 | MAA
VISHAL0201 | 1 | GOI | 2021-01-17 22:00:00 | DEL
ALIA1708 | 1 | BLR | 2021-01-09 10:00:00 | CCU
MANISHA2304 | 1 | BOM | 2021-02-02 08:00:00 | BOM
(15 rows)
```

```
SQL Shell (psql)
airline_reservation=# BEGIN;
BEGIN
airline_reservation=#
airline_reservation=# UPDATE FLIGHT_TRIP SET FLIGHT_TRIP_ID='MANISHA2304' WHERE FLIGHT_TRIP_ID IN (SELECT FLIGHT FROM TRAVELLER_ITENARY WHERE ID IN (SELECT TRAVELLER_ID FROM TRAVELLER WHERE FNAME='MANISHA'));
UPDATE 1
airline_reservation=#
```

We see that the Update is occurred but has not been updated in the second terminal showing a concurrency control.

If we do a rollback the update happens in the second terminal showing that until a transaction is complete (either commit or a rollback) same transaction can't be done.

If we do commit we see update 0 in the other terminal showing that other terminal has updated improving the concurrent transaction.

```
SQL Shell (psql)
ALEX2010 | 1 | PAT | 2021-02-02 19:00:00 | COI
SUNAM5678 | 1 | PAT | 2021-02-02 19:00:00 | COI
RAJ2109 | 1 | DEL | 2021-02-02 19:00:00 | BLR
DINESH0306 | 1 | LKO | 2021-01-09 22:00:00 | DEL
GOPAL2512 | 1 | LKO | 2021-01-09 22:00:00 | DEL
NEESH0506 | 1 | BLR | 2021-02-12 23:00:00 | COI
SURYA2310 | 1 | BLR | 2021-02-12 23:00:00 | COI
RAHUL2312 | 1 | BOM | 2021-03-10 23:00:00 | CCU
ROHAN3210 | 1 | BOM | 2021-01-17 10:00:00 | MAA
ROSHNI1709 | 1 | BOM | 2021-01-17 10:00:00 | MAA
VISHAL0201 | 1 | GOI | 2021-01-17 22:00:00 | DEL
ALIA1708 | 1 | BLR | 2021-01-09 10:00:00 | CCU
MANISHA2304 | 1 | BOM | 2021-02-02 08:00:00 | BOM
(15 rows)

airline_reservation=# ROLLBACK;
ROLLBACK
airline_reservation=# BEGIN;
BEGIN
airline_reservation=# UPDATE FLIGHT_TRIP SET FLIGHT_TRIP_ID='MANISHA2304' WHERE FLIGHT_TRIP_ID IN (SELECT FLIGHT FROM TRAVELLER_ITENARY WHERE ID IN (SELECT TRAVELLER_ID FROM TRAVELLER WHERE FNAME='MANISHA'));
UPDATE 0
airline_reservation=#
```

```
SQL Shell (psql)
Role name | Attributes | Member of
-----+-----+-----
admin |  | {}
developer |  | {}
postgres | Superuser, Create role, Create DB, Replication, Bypass RLS | {}
tester |  | {}

airline_reservation=# ALTER DEFAULT PRIVILEGES IN SCHEMA public
airline_reservation=# GRANT SELECT ON TABLES TO PUBLIC;ALTER DEFAULT PRIVILEGES IN SCHEMA public
ALTER DEFAULT PRIVILEGES
airline_reservation=# GRANT SELECT ON TABLES TO DEVELOPER;
ALTER DEFAULT PRIVILEGES
airline_reservation=# \DU
invalid command \DU
Try \? for help.
airline_reservation=# \du

List of roles
Role name | Attributes | Member of
-----+-----+-----
admin |  | {}
developer |  | {}
postgres | Superuser, Create role, Create DB, Replication, Bypass RLS | {}
tester |  | {}

airline_reservation=# BEGIN;
BEGIN
airline_reservation=#
airline_reservation=# UPDATE FLIGHT_TRIP SET FLIGHT_TRIP_ID='MANISHA2304' WHERE FLIGHT_TRIP_ID IN (SELECT FLIGHT FROM TRAVELLER_ITENARY WHERE ID IN (SELECT TRAVELLER_ID FROM TRAVELLER WHERE FNAME='MANISHA'));
UPDATE 1
airline_reservation=# COMMIT;
COMMIT
airline_reservation=#
```

```
airline_reservation=# BEGIN;
BEGIN
airline_reservation=# UPDATE FLIGHT_TRIP SET FLIGHT_TRIP_ID='MANISHA2305' WHERE FLI
GHT_TRIP_ID IN (SELECT FLIGHT FROM TRAVELLER_ITENARY WHERE ID IN (SELECT TRAVELLER_I
D FROM TRAVELLER WHERE FNAME='MANISHA'));
UPDATE 1
airline_reservation=# COMMIT;
COMMIT
```

```
airline_reservation=# BEGIN;
BEGIN
airline_reservation=# UPDATE FLIGHT_TRIP SET FLIGHT_TRIP_ID='MANISHA2306' WHERE FLI
GHT_TRIP_ID IN (SELECT FLIGHT FROM TRAVELLER_ITENARY WHERE ID IN (SELECT TRAVELLER_I
D FROM TRAVELLER WHERE FNAME='MANISHA'));
UPDATE 0
airline_reservation=# COMMIT;
COMMIT
```

Another example of concurrency control in the PostgreSQL.

CONTRIBUTION

TOPIC	CONTRIBUTION BY	TIME TAKEN
SIMPLE QUERIES	SAI JAYANTH IMMADISETTY(PES2UG19CS152) AMIT JITTA(PES2UG19CS169)	30 MINUTES
NESTED/COMPLEX QUERIES	KUNTAL GORAI(PES2UG19CS198)	30 MINUTES
CREATING USERS	KUNTAL GORAI(PES2UG19CS198)	30 MINUTES
CONCURRENCY	KUNTAL GORAI(PES2UG19CS198)	30 MINUTES