DATABASE MANAGEMENT SYSTEM

AIRLINES RESERVATION SYSTEM

TEAM ID: 05

RITHIKA PAI
PES1UG19CS388

RITHIKA SHANKAR
PES1UG19CS387

RAMYA N PRABHU
PES1UG19CS380

User Interface

Using psycopg2 to connect the frontend GUI to our PostgreSQL database-

Psycopg2 is used to facilitate connectivity between the frontend and backend. Psycopg is the most popular PostgreSQL database adapter for the Python programming language. Its main features are the complete implementation of the Python DB API 2.0 specification and the thread safety (several threads can share the same connection)

- Straightforward and concise syntax
- Useful built-in methods
- Good error handling
- Supports all necessary SQL query types
- Familiarity with and ease of use of Python

Installation: ***`pip install psycopg2`***

Version 2.9.2

Using tkinter as our GUI-

● Comes bundled with initial Python installation

● Efficient integration with psycopg2 statements and PostgreSQL back-end database

● Fast compared to other GUI toolkits

● Flexible and stable

● Simple and concise syntax

● Many customisation options to format interactive display window

Installation: ***`pip install tk`***

`Version 8.6`

Dependencies installed for database connectivity

Our front-end has been implemented using the following libraries in Python, the reasons for which have been listed above-

● tkinter

● psycopg2

● PIL (for images)

Some of the specific modules we have used for implementation are-

● From tkinter

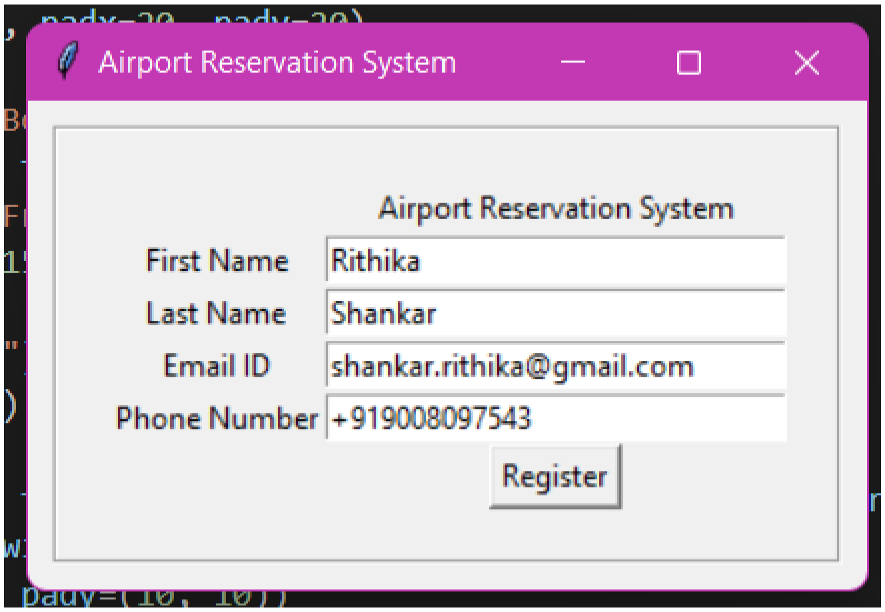
○ Tk ○ title ○ iconbitmap ○ Label ○ pack ○ grid

- From psycopg2

○ connect ○ cursor ○ execute ○ fetchall

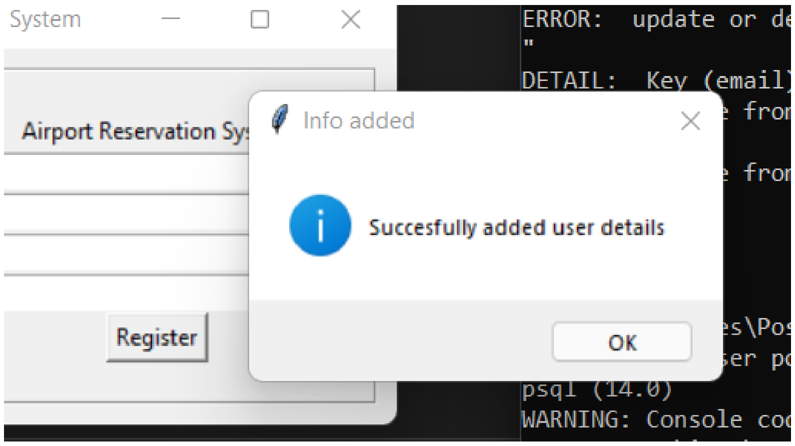
Sample queries executed from front-end

1. Registering a User



Airport Reservation System

First Name	Rithika
Last Name	Shankar
Email ID	shankar.rithika@gmail.com
Phone Number	+919008097543

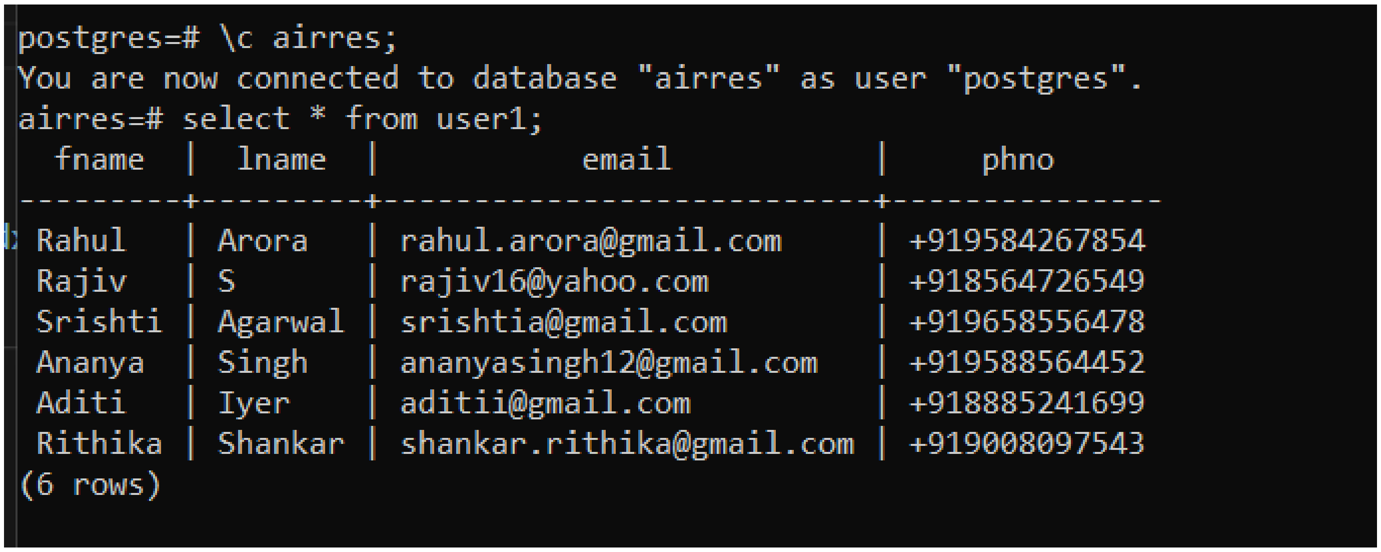


Info added

Succesfully added user details

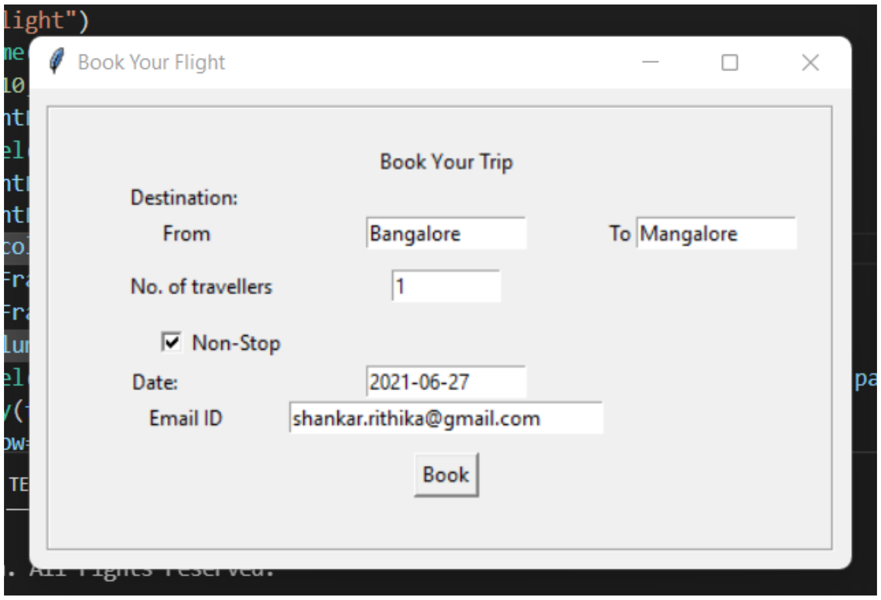
```
ERROR:  update or de
"
DETAIL:  Key (email)
psql (14.0)
WARNING: Console coo
```

Update in the back-end:



```
postgres=# \c airres;
You are now connected to database "airres" as user "postgres".
airres=# select * from user1;
  fname  |  lname  |           email           |     phno
---------+---------+---------------------------+---------------
 Rahul   | Arora   | rahul.arora@gmail.com     | +919584267854
 Rajiv   | S       | rajiv16@yahoo.com         | +918564726549
 Srishti | Agarwal | srishtia@gmail.com        | +919658556478
 Ananya  | Singh   | ananyasingh12@gmail.com   | +919588564452
 Aditi   | Iyer    | aditii@gmail.com          | +918885241699
 Rithika | Shankar | shankar.rithika@gmail.com | +919008097543
(6 rows)
```

2. Booking a flight



Book Your Flight

Book Your Trip

Destination:

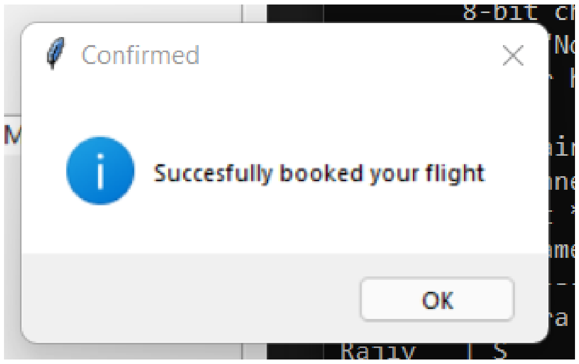
From: Bangalore   To: Mangalore

No. of travellers: 1

☑ Non-Stop

Date: 2021-06-27

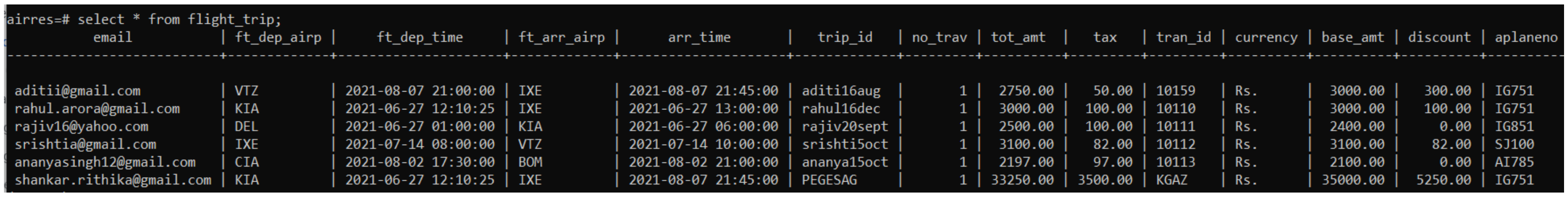
Email ID: shankar.rithika@gmail.com


Confirmed

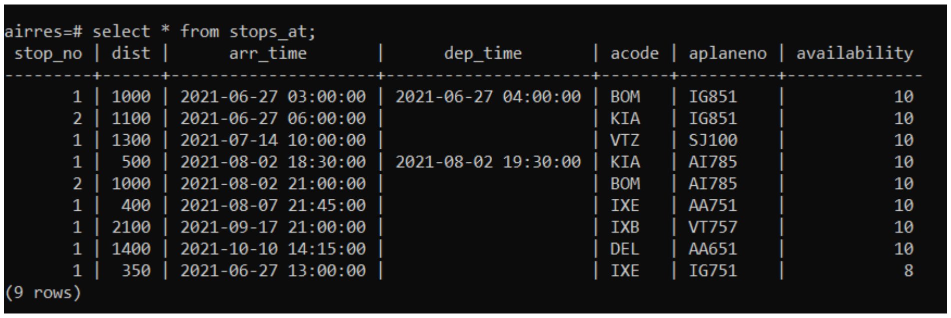
Succesfully booked your flight

Update in the back-end

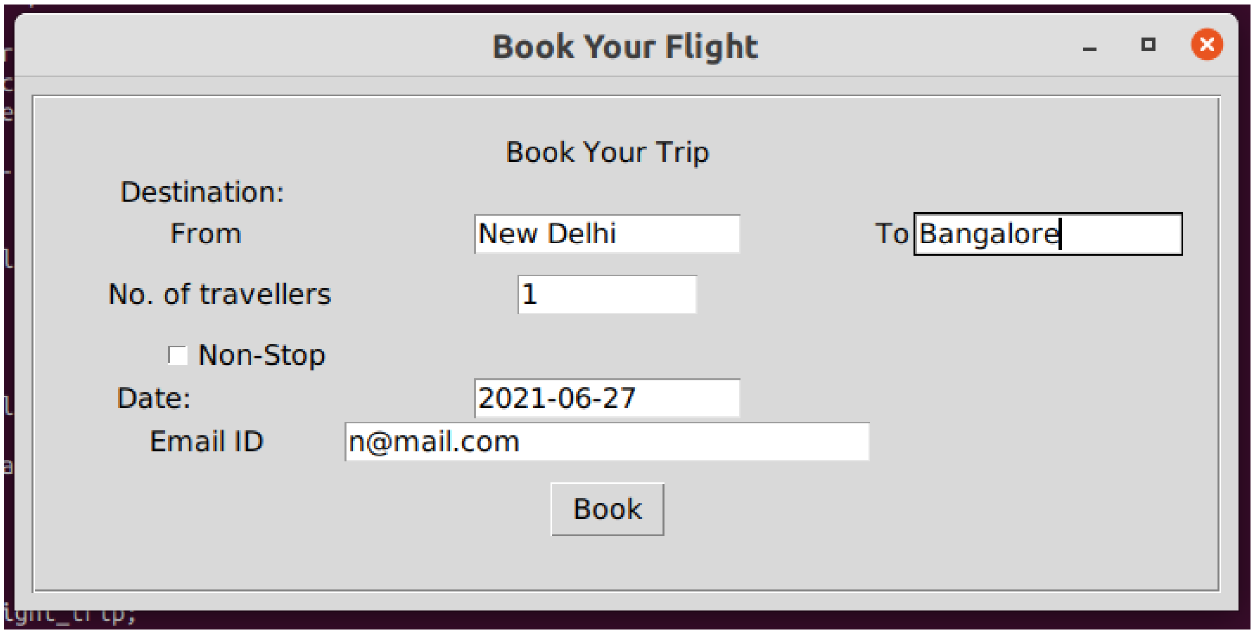
Adds details to the flight trip table:


```
airres=# select * from flight_trip;
          email            | ft_dep_airp |     ft_dep_time      | ft_arr_airp |       arr_time       |   trip_id   | no_trav | tot_amt  |   tax    | tran_id | currency | base_amt | discount | aplaneno
---------------------------+-------------+----------------------+-------------+----------------------+-------------+---------+----------+----------+---------+----------+----------+----------+---------


 aditii@gmail.com          | VTZ         | 2021-08-07 21:00:00 | IXE         | 2021-08-07 21:45:00 | aditi16aug  |       1 |  2750.00 |    50.00 | 10159   | Rs.      |  3000.00 |   300.00 | IG751
 rahul.arora@gmail.com     | KIA         | 2021-06-27 12:10:25 | IXE         | 2021-06-27 13:00:00 | rahul16dec  |       1 |  3000.00 |   100.00 | 10110   | Rs.      |  3000.00 |   100.00 | IG751
 rajiv16@yahoo.com         | DEL         | 2021-06-27 01:00:00 | KIA         | 2021-06-27 06:00:00 | rajiv20sept |       1 |  2500.00 |   100.00 | 10111   | Rs.      |  2400.00 |     0.00 | IG851
 srishtia@gmail.com        | IXE         | 2021-07-14 08:00:00 | VTZ         | 2021-07-14 10:00:00 | srishti5oct |       1 |  3100.00 |    82.00 | 10112   | Rs.      |  3100.00 |    82.00 | SJ100
 ananyasingh12@gmail.com   | CIA         | 2021-08-02 17:30:00 | BOM         | 2021-08-02 21:00:00 | ananya15oct |       1 |  2197.00 |    97.00 | 10113   | Rs.      |  2100.00 |     0.00 | AI785
 shankar.rithika@gmail.com | KIA         | 2021-06-27 12:10:25 | IXE         | 2021-08-07 21:45:00 | PEGESAG     |       1 | 33250.00 |  3500.00 | KGAZ    | Rs.      | 35000.00 |  5250.00 | IG751
```

Also, reduces the availability of seats in the stops at table:


```
airres=# select * from stops_at;
 stop_no | dist |       arr_time       |       dep_time       | acode | aplaneno | availability
---------+------+----------------------+----------------------+-------+----------+-------------
       1 | 1000 | 2021-06-27 03:00:00 | 2021-06-27 04:00:00 | BOM   | IG851    |           10
       2 | 1100 | 2021-06-27 06:00:00 |                      | KIA   | IG851    |           10
       1 | 1300 | 2021-07-14 10:00:00 |                      | VTZ   | SJ100    |           10
       1 |  500 | 2021-08-02 18:30:00 | 2021-08-02 19:30:00 | KIA   | AI785    |           10
       2 | 1000 | 2021-08-02 21:00:00 |                      | BOM   | AI785    |           10
       1 |  400 | 2021-08-07 21:45:00 |                      | IXE   | AA751    |           10
       1 | 2100 | 2021-09-17 21:00:00 |                      | IXB   | VT757    |           10
       1 | 1400 | 2021-10-10 14:15:00 |                      | DEL   | AA651    |           10
       1 |  350 | 2021-06-27 13:00:00 |                      | IXE   | IG751    |            8
(9 rows)
```

Booking a flight that has a layover



Book Your Trip

Destination:

From New Delhi To Bangalore

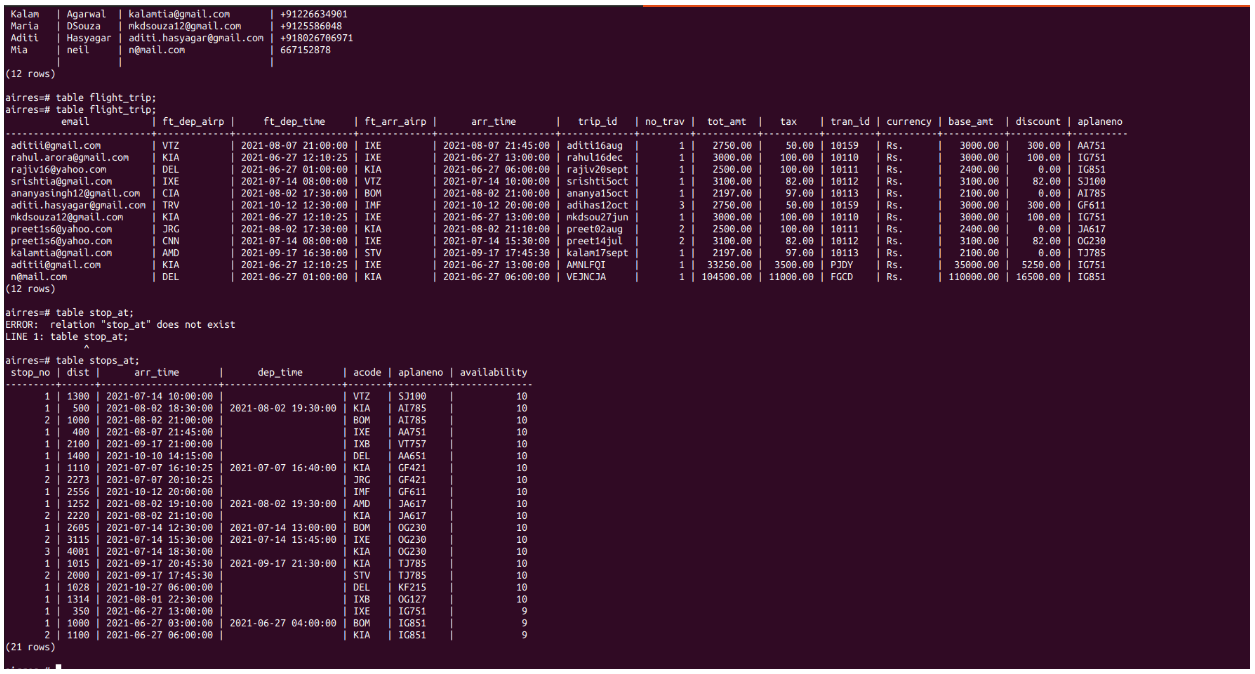
No. of travellers 1

☐ Non-Stop

Date: 2021-06-27

Email ID n@mail.com

Changes to the DB



```
 Kalam    | Agarwal  | kalamtia@gmail.com         | +91226634901
 Maria    | DSouza   | mkdsouza12@gmail.com       | +9125586048
 Aditi    | Hasyagar | aditi.hasyagar@gmail.com   | +918026706971
 Mia      | neil     | n@mail.com                 | 667152878
          |          |                            |
(12 rows)

airres=# table flight_trip;
airres=# table flight_trip;
          email           | ft_dep_airp |     ft_dep_time     | ft_arr_airp |      arr_time       |    trip_id     | no_trav |  tot_amt  |    tax    | tran_id | currency | base_amt  | discount | aplaneno
--------------------------+-------------+---------------------+-------------+---------------------+----------------+---------+-----------+-----------+---------+----------+-----------+----------+----------
 aditii@gmail.com         | VTZ         | 2021-08-07 21:00:00 | IXE         | 2021-08-07 21:45:00 | aditi16aug     |       1 |   2750.00 |     50.00 | 10159   | Rs.      |   3000.00 |   300.00 | AA751
 rahul.arora@gmail.com    | KIA         | 2021-06-27 12:10:25 | IXE         | 2021-06-27 13:00:00 | rahul16dec     |       1 |   3000.00 |    100.00 | 10110   | Rs.      |   3000.00 |   100.00 | IG751
 rajiv16@yahoo.com        | DEL         | 2021-06-27 01:00:00 | KIA         | 2021-06-27 06:00:00 | rajiv20sept    |       1 |   2500.00 |    100.00 | 10111   | Rs.      |   2400.00 |     0.00 | IG851
 srishtia@gmail.com       | IXE         | 2021-07-14 08:00:00 | VTZ         | 2021-07-14 10:00:00 | srishti5oct    |       1 |   3100.00 |     82.00 | 10112   | Rs.      |   3100.00 |    82.00 | SJ100
 ananyasingh12@gmail.com  | CIA         | 2021-08-02 17:30:00 | BOM         | 2021-08-02 21:00:00 | ananya15oct    |       1 |   2197.00 |     97.00 | 10113   | Rs.      |   2100.00 |     0.00 | AI785
 aditi.hasyagar@gmail.com | TRV         | 2021-10-12 12:30:00 | IMF         | 2021-10-12 20:00:00 | adihas12oct    |       3 |   2750.00 |     50.00 | 10159   | Rs.      |   3000.00 |   300.00 | GF611
 mkdsouza12@gmail.com     | KIA         | 2021-06-27 12:10:25 | IXE         | 2021-06-27 13:00:00 | mkdsou27jun    |       1 |   3000.00 |    100.00 | 10110   | Rs.      |   3000.00 |   100.00 | IG751
 preet1s6@yahoo.com       | JRG         | 2021-08-02 17:30:00 | KIA         | 2021-08-02 21:10:00 | preet02aug     |       2 |   2500.00 |    100.00 | 10111   | Rs.      |   2400.00 |     0.00 | JA617
 preet1s6@yahoo.com       | CNN         | 2021-07-14 08:00:00 | IXE         | 2021-07-14 15:30:00 | preet14jul     |       2 |   3100.00 |     82.00 | 10112   | Rs.      |   3100.00 |    82.00 | OG230
 kalamtia@gmail.com       | AMD         | 2021-09-17 16:30:00 | STV         | 2021-09-17 17:45:30 | kalam17sept    |       1 |   2197.00 |     97.00 | 10113   | Rs.      |   2100.00 |     0.00 | TJ785
 aditii@gmail.com         | KIA         | 2021-06-27 12:10:25 | IXE         | 2021-06-27 13:00:00 | AMNLFQI        |       1 |  33250.00 |   3500.00 | PJDY    | Rs.      |  35000.00 |  5250.00 | IG751
 n@mail.com               | DEL         | 2021-06-27 01:00:00 | KIA         | 2021-06-27 06:00:00 | VEJNCJA        |       1 | 104500.00 |  11000.00 | FGCD    | Rs.      | 110000.00 | 16500.00 | IG851
(12 rows)

airres=# table stop_at;
ERROR:  relation "stop_at" does not exist
LINE 1: table stop_at;
              ^
airres=# table stops_at;
 stop_no | dist |      arr_time       |      dep_time       | acode | aplaneno | availability
---------+------+---------------------+---------------------+-------+----------+--------------
       1 | 1300 | 2021-07-14 10:00:00 |                     | VTZ   | SJ100    |           10
       1 |  500 | 2021-08-02 18:30:00 | 2021-08-02 19:30:00 | KIA   | AI785    |           10
       2 | 1000 | 2021-08-02 21:00:00 |                     | BOM   | AI785    |           10
       1 |  400 | 2021-08-07 21:45:00 |                     | IXE   | AA751    |           10
       1 | 2100 | 2021-09-17 21:00:00 |                     | IXB   | VT757    |           10
       1 | 1400 | 2021-10-10 14:15:00 |                     | DEL   | AA651    |           10
       1 | 1110 | 2021-07-07 16:10:25 | 2021-07-07 16:40:00 | KIA   | GF421    |           10
       2 | 2273 | 2021-07-07 20:10:25 |                     | JRG   | GF421    |           10
       1 | 2556 | 2021-10-12 20:00:00 |                     | IMF   | GF611    |           10
       1 | 1252 | 2021-08-02 19:10:00 | 2021-08-02 19:30:00 | AMD   | JA617    |           10
       2 | 2220 | 2021-08-02 21:10:00 |                     | KIA   | JA617    |           10
       1 | 2605 | 2021-07-14 12:30:00 | 2021-07-14 13:00:00 | BOM   | OG230    |           10
       2 | 3115 | 2021-07-14 15:30:00 | 2021-07-14 15:45:00 | IXE   | OG230    |           10
       3 | 4001 | 2021-07-14 18:30:00 |                     | KIA   | OG230    |           10
       1 | 1015 | 2021-09-17 20:45:30 | 2021-09-17 21:30:00 | KIA   | TJ785    |           10
       2 | 2000 | 2021-09-17 17:45:30 |                     | STV   | TJ785    |           10
       1 | 1028 | 2021-10-27 06:00:00 |                     | DEL   | KF215    |           10
       1 | 1314 | 2021-08-01 22:30:00 |                     | IXB   | OG127    |           10
       1 |  350 | 2021-06-27 13:00:00 |                     | IXE   | IG751    |            9
       1 | 1000 | 2021-06-27 03:00:00 | 2021-06-27 04:00:00 | BOM   | IG851    |            9
       2 | 1100 | 2021-06-27 06:00:00 |                     | KIA   | IG851    |            9
(21 rows)
```

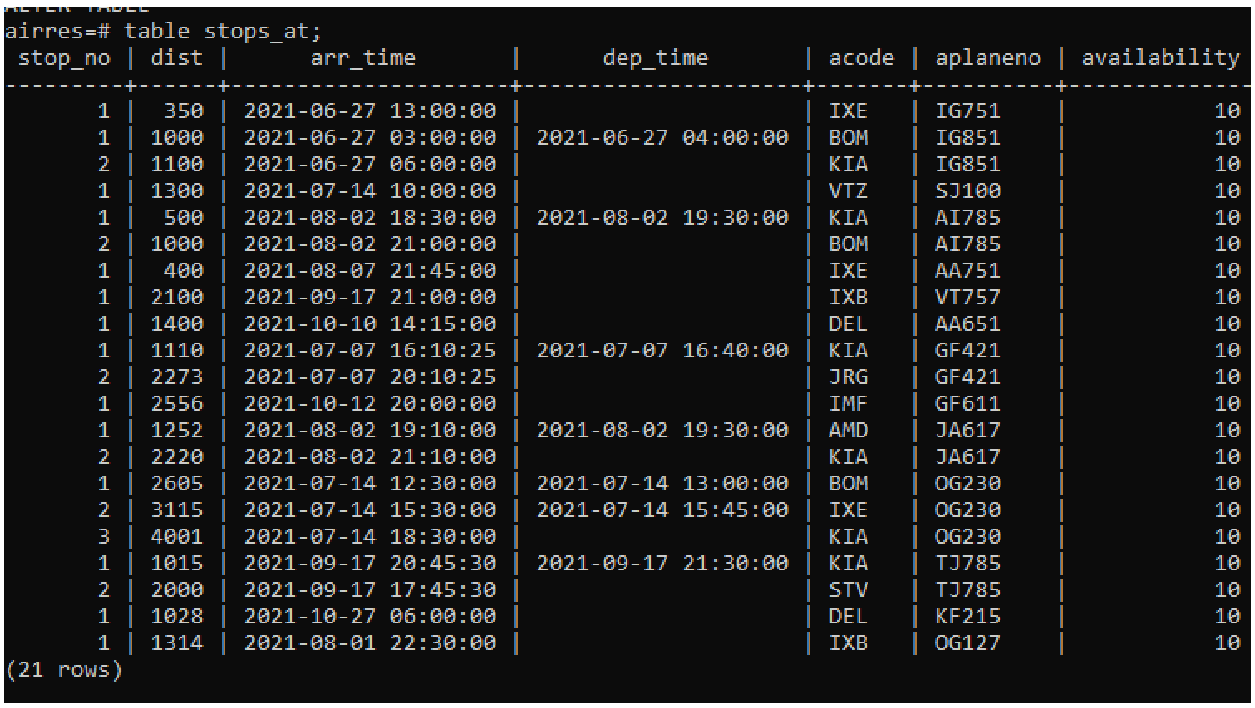
Schema Changes

- Adding an extra column for the seats available on the flight leg

To indicate the number of seats available, a column names number of seats is added to the stops_at table

Statements:

```
alter table stops_at

add column availability int;

update stops_at

set availability= 10;

alter table stops_at

alter column availability set not null;
```



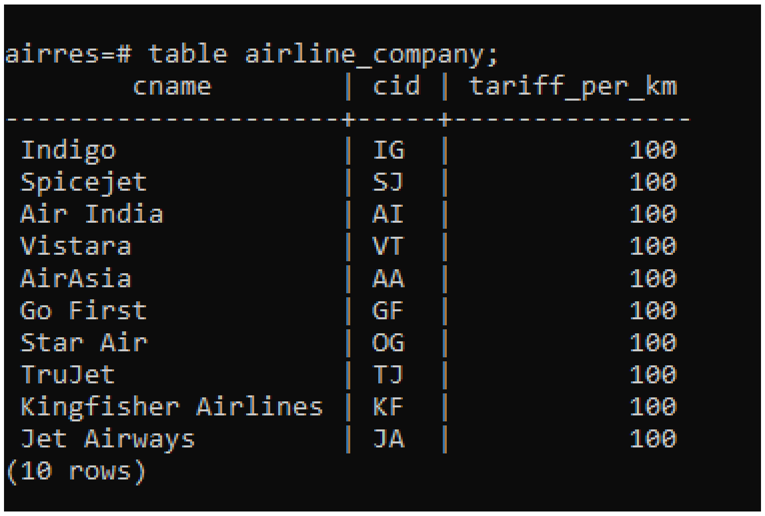
```
ALTER TABLE
airres=# table stops_at;
 stop_no | dist |      arr_time       |      dep_time       | acode | aplaneno | availability
---------+------+---------------------+---------------------+-------+----------+--------------
       1 |  350 | 2021-06-27 13:00:00 |                     | IXE   | IG751    |           10
       1 | 1000 | 2021-06-27 03:00:00 | 2021-06-27 04:00:00 | BOM   | IG851    |           10
       2 | 1100 | 2021-06-27 06:00:00 |                     | KIA   | IG851    |           10
       1 | 1300 | 2021-07-14 10:00:00 |                     | VTZ   | SJ100    |           10
       1 |  500 | 2021-08-02 18:30:00 | 2021-08-02 19:30:00 | KIA   | AI785    |           10
       2 | 1000 | 2021-08-02 21:00:00 |                     | BOM   | AI785    |           10
       1 |  400 | 2021-08-07 21:45:00 |                     | IXE   | AA751    |           10
       1 | 2100 | 2021-09-17 21:00:00 |                     | IXB   | VT757    |           10
       1 | 1400 | 2021-10-10 14:15:00 |                     | DEL   | AA651    |           10
       1 | 1110 | 2021-07-07 16:10:25 | 2021-07-07 16:40:00 | KIA   | GF421    |           10
       2 | 2273 | 2021-07-07 20:10:25 |                     | JRG   | GF421    |           10
       1 | 2556 | 2021-10-12 20:00:00 |                     | IMF   | GF611    |           10
       1 | 1252 | 2021-08-02 19:10:00 | 2021-08-02 19:30:00 | AMD   | JA617    |           10
       2 | 2220 | 2021-08-02 21:10:00 |                     | KIA   | JA617    |           10
       1 | 2605 | 2021-07-14 12:30:00 | 2021-07-14 13:00:00 | BOM   | OG230    |           10
       2 | 3115 | 2021-07-14 15:30:00 | 2021-07-14 15:45:00 | IXE   | OG230    |           10
       3 | 4001 | 2021-07-14 18:30:00 |                     | KIA   | OG230    |           10
       1 | 1015 | 2021-09-17 20:45:30 | 2021-09-17 21:30:00 | KIA   | TJ785    |           10
       2 | 2000 | 2021-09-17 17:45:30 |                     | STV   | TJ785    |           10
       1 | 1028 | 2021-10-27 06:00:00 |                     | DEL   | KF215    |           10
       1 | 1314 | 2021-08-01 22:30:00 |                     | IXB   | OG127    |           10
(21 rows)
```

- Adding a column for the tariff

A column called Tariff [rate per km] is added to the airline_companies table for each airline company in the table

Statement:

```
alter table airline_company

add column tariff_per_km int;



update airline_company

set tariff_per_km=100;



alter table airline_company

alter column tariff_per_km set not null;
```



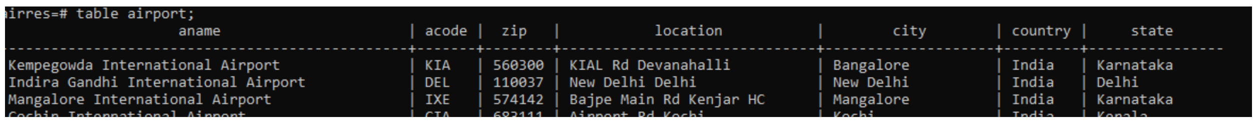
```
airres=# table airline_company;
        cname        | cid | tariff_per_km
---------------------+-----+---------------
 Indigo              | IG  |           100
 Spicejet            | SJ  |           100
 Air India           | AI  |           100
 Vistara             | VT  |           100
 AirAsia             | AA  |           100
 Go First            | GF  |           100
 Star Air            | OG  |           100
 TruJet              | TJ  |           100
 Kingfisher Airlines | KF  |           100
 Jet Airways         | JA  |           100
(10 rows)
```

- Changing the name of the column 'staete'

'staete' in the airport column is changed to 'state'

Statement:

```
alter table airport rename staete to state;
```



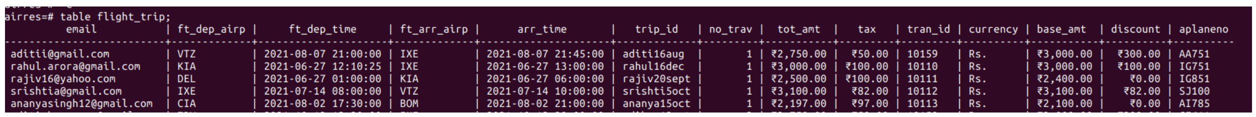
```
irres=# table airport;
                    aname                      | acode |  zip   |          location          |        city        | country |     state
-----------------------------------------------+-------+--------+----------------------------+--------------------+---------+---------------
 Kempegowda International Airport              | KIA   | 560300 | KIAL Rd Devanahalli        | Bangalore          | India   | Karnataka
 Indira Gandhi International Airport           | DEL   | 110037 | New Delhi Delhi            | New Delhi          | India   | Delhi
 Mangalore International Airport               | IXE   | 574142 | Bajpe Main Rd Kenjar HC    | Mangalore          | India   | Karnataka
```

- Changing the datatype of the column's whose data type was money

```
alter table flight_trip

alter column tot_amt type decimal(10,2),

alter column tax type decimal(10,2),

alter column base_amt type decimal(10,2),

alter column discount type decimal(10,2);
```

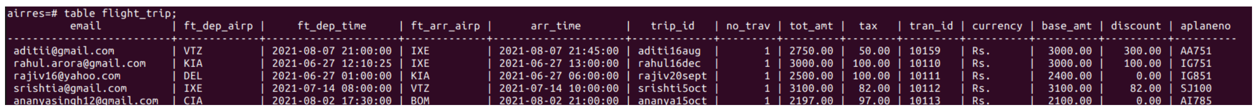
[Why? Ease of insertion]

Before:



```
airres=# table flight_trip;
         email          | ft_dep_airp |     ft_dep_time     | ft_arr_airp |      arr_time       |   trip_id   | no_trav |  tot_amt   |   tax    | tran_id | currency | base_amt  | discount | aplaneno
------------------------+-------------+---------------------+-------------+---------------------+-------------+---------+------------+----------+---------+----------+-----------+----------+----------
 aditii@gmail.com       | VTZ         | 2021-08-07 21:00:00 | IXE         | 2021-08-07 21:45:00 | aditi16aug  |       1 | ₹2,750.00  |   ₹50.00 | 10159   | Rs.      | ₹3,000.00 |  ₹300.00 | AA751
 rahul.arora@gmail.com  | KIA         | 2021-06-27 12:10:25 | IXE         | 2021-06-27 13:00:00 | rahul16dec  |       1 | ₹3,000.00  |  ₹100.00 | 10110   | Rs.      | ₹3,000.00 |  ₹100.00 | IG751
 rajiv16@yahoo.com      | DEL         | 2021-06-27 01:00:00 | KIA         | 2021-06-27 06:00:00 | rajiv20sept |       1 | ₹2,500.00  |  ₹100.00 | 10111   | Rs.      | ₹2,400.00 |    ₹0.00 | IG851
 srishtia@gmail.com     | IXE         | 2021-07-14 08:00:00 | VTZ         | 2021-07-14 10:00:00 | srishti5oct |       1 | ₹3,100.00  |   ₹82.00 | 10112   | Rs.      | ₹3,100.00 |   ₹82.00 | SJ100
 ananyasingh12@gmail.com | CIA        | 2021-08-02 17:30:00 | BOM         | 2021-08-02 21:00:00 | ananya15oct |       1 | ₹2,197.00  |   ₹97.00 | 10113   | Rs.      | ₹2,100.00 |    ₹0.00 | AI785
```

After:



```
airres=# table flight_trip;
         email           | ft_dep_airp |     ft_dep_time     | ft_arr_airp |      arr_time       |   trip_id   | no_trav | tot_amt |  tax   | tran_id | currency | base_amt | discount | aplaneno
-------------------------+-------------+---------------------+-------------+---------------------+-------------+---------+---------+--------+---------+----------+----------+----------+----------
 aditii@gmail.com        | VTZ         | 2021-08-07 21:00:00 | IXE         | 2021-08-07 21:45:00 | aditi16aug  |       1 | 2750.00 |  50.00 | 10159   | Rs.      |  3000.00 |   300.00 | AA751
 rahul.arora@gmail.com   | KIA         | 2021-06-27 12:10:25 | IXE         | 2021-06-27 13:00:00 | rahul16dec  |       1 | 3000.00 | 100.00 | 10110   | Rs.      |  3000.00 |   100.00 | IG751
 rajiv16@yahoo.com       | DEL         | 2021-06-27 01:00:00 | KIA         | 2021-06-27 06:00:00 | rajiv20sept |       1 | 2500.00 | 100.00 | 10111   | Rs.      |  2400.00 |     0.00 | IG851
 srishtia@gmail.com      | IXE         | 2021-07-14 08:00:00 | VTZ         | 2021-07-14 10:00:00 | srishti5oct |       1 | 3100.00 |  82.00 | 10112   | Rs.      |  3100.00 |    82.00 | SJ100
 ananyasingh12@gmail.com | CIA         | 2021-08-02 17:30:00 | BOM         | 2021-08-02 21:00:00 | ananya15oct |       1 | 2197.00 |  97.00 | 10113   | Rs.      |  2100.00 |     0.00 | AI785
```

Database Migration and Support

Overview

- Changes in the business model, application and clientele would definitely lead to changes in schema, constraints and maybe even the dbms
- Changes like a rapid increase in the number of customers would require the use of efficient scalable and low cost dbms. This essential calls for the use of NoSQL databases because these databases are designed to be used with low-cost commodity hardware. Moreover, support for big data applications, with NoSQL databases able to handle massive volumes of data and support for auto-sharding, allowing NoSQL databases to natively and automatically spread data across an arbitrary number of servers, without needing the application to be aware of the server pool composition
- Changes like providing a variety of services would call for more number of tables and difference in constraints in the same. However this also brings with it additional complexity
- If the company changes to an agile environment, the data model and schema are likely to be subjected to frequent change and in such case, it is best to switch to a NoSQL database

Changes Specific to our enterprise that might lead to changes in DB Management

Business Expansion

On expansion of our clientele, we would like to be able to allow them to pre book their meals, save their payment details . We would also wish to accommodate passengers with special needs, senior citizens, armed forces and students. Airlines have different tariff schemes that we would like to bring to our customer base.

Schema and constraint Changes

- **Pre-booked Meals:** a column for meal choice have to be added to the travellers relation. A column in the flight trip id to indicate that the trip in question has booked a meal. And new table with information about the meal options provided by each airline The new tables for meals would have two foreign key which would be the trip id and the traveller id which together will be used as a primary key
- **Save payment details:** A new table called payment would have to be created which would have to contain the preferred mode of payment, and the respective address to identify the appropriate payment interface for the user. The user's email would be the primary key here [it is also the foreign key].
- **Fare for students, senior citizens, armed forces and special needs:** A new table called special fares should be created. The schema for this table would be type [ex student, senior citizen, veterans, etc], airline company [this would be a foreign key and a primary key], and the amount of fare reduction.
- **Tariff Schemes:** Different tariff schemes can be accommodated by adding a new table, whose schema would be comany_id [foreign key, primary key], tariff name, tariff criteria, tariff rate.

Migrating Databases from this expansion

NoSQL databases were developed during the Internet era in response to the inability of SQL databases to address the needs of web scale applications that handled huge amounts of data and traffic.

Companies are finding that they can apply NoSQL technology to a growing list of use cases while saving money in comparison to operating a relational database. NoSQL databases invariably incorporate a flexible schema model and are designed to scale horizontally across many servers, which makes them appealing for large data volumes or application loads that exceed the capacity of a single server.

So, business is likely to migrate to a NoSQL database when:

- The pace of development with NoSQL databases can be much faster than with a SQL database.
- The structure of many different forms of data is more easily handled and evolved with a NoSQL database.
- The amount of data in many applications cannot be served affordably by a SQL database.
- The scale of traffic and need for zero downtime cannot be handled by SQL.
- New application paradigms can be more easily supported.

Database Migration: MongoDB (NoSQL)- Why should we use MongoDB?

● MongoDB is a document oriented database that enables you to manage data of any structure, not just tabular structures defined in advance and developers can reshape the data on their own.

- Scaling in terms of volume of traffic or size of data (or both) needs to be distributed across regions can be handled by MongoDB's scale-out architecture.

- Multi-cloud database that works the same way in every public cloud, can store customer data in specific geographic regions, and support the latest serverless and mobile development paradigms.

- Query performance in MongoDB can be accelerated by creating indexes on fields in documents and subdocuments. MongoDB allows any field of a document, including those deeply nested in arrays and subdocuments, to be indexed and efficiently queried.

- The downside of PostgreSQL compared to MongoDB is that it relies on relational data models that are unfriendly to the data structures developers work with in code, and that must be defined in advance, slowing progress whenever requirements change.

- In MongoDB scalability is built-in through native sharding, enabling a horizontal scale-out approach. MongoDB Atlas has a broad multi-cloud, globally aware platform at the ready, all fully managed for you.

- PostgreSQL uses a scale-up strategy. This means that at some point, for high performance use cases, you may hit a wall or have to divert resources to finding other ways to scale via caching or denormalizing data or using other strategies.

- PostgreSQL can support replication but more advanced features such as automatic failover must be supported by third-party products developed independently of the database. Such an approach is more complex and can work slower and less seamlessly than MongoDB's in-built self-healing capabilities.

Steps to migrate from PostgreSQL to MongoDB

1. To migrate data, you'll extract it from PostgreSQL and then import it to MongoDB using the mongoimport tool. Two different ways to extract the data are: returning queries as tab-separated values (TSV) or as JSON.

2. Prepare your application for connecting to MongoDB., MongoDB has support for all of the major programming languages as well as many popular frameworks.

3. Consider the schema changes that would be best for your data, while keeping in mind MongoDB schema best practices and avoiding anti-patterns.

4. Export the data from your PostgreSQL databases by piping the result of an SQL query into a COPY command, outputting the result either as JSON or TSV.

5. Restructure the data to fit your MongoDB schema by using mongoimport (or as an alternative: use bulkWrite operations to load the data).

Contributions

Rithika Pai

-

Rithika Shankar

- Frontend development and backend queries
- Database Migration (NoSQL)
- Report

7-8 hours

Ramya Prabhu

- Schema Changes

- Queries for the backend
- Database migration in case of business expansion
- [Report for the same]

4 hours