

## Airlines Database Management System

**Group Name:** ChauhanCChavaR

**Team Members:** Chetan Chauhan  
Renusree Chava

### **Abstract:**

The Airlines database management system allows for managing the expanding operations and meeting customer demands. To ensure flexible and efficient services for customers, it is recommended to implement an Airlines Reservation System, which will computerize all information related to flights, passengers, reservations, transactions, and payments. The passenger can check the fare and book the ticket. There are multiple airlines with multiple aircrafts flying different routes. This database would help the airline sort their data efficiently and help their transactions. Airlines management systems are best suitable for tracking ticket bookings in an effortless and effective manner with minimal effort and time so that it increases revenue.

### **Technical Specifications:**

Each Airline has a unique identification number, name, employees who work for it, and multiple aircrafts that belong to different models.

An Employee can only work for a single Airline. Each employee member has a name, email, and address. An Employee can either be a ground staff. Each employee is uniquely identified by an ID. Each aircraft is uniquely identified by its ID. An Aircraft also has a start date of flying and a capacity. An Employee can also manage to add the new aircraft details in the system with all necessary details and the pilots associated with it. Each pilot has a Pilot License number, and rankings with a start date. Pilots fly the aircraft while ground staff assists the passengers.

Countries consist of cities that are connected by flights that are specified in the routes. A route can be uniquely identified by its ID and consists of the city of departure, the city of arrival along with the fare. It also consists of the seats left and is updated as the passenger books the flight. Each Route has a specific fare associated with it.

A registered passenger is uniquely identified by email and has the name, SSN, date of the birth, and city. A passenger can view the routes and schedules and book multiple schedules.

For "booking the ticket" the database keeps track of airline bookings made by passengers. Each booking has a unique booking\_id that is automatically generated for each booking made. The "schedule\_id" and "passenger\_id" columns store the IDs of the specific schedule and passenger associated with each booking, respectively which also ensures that the data in these columns reference the correct IDs in the "schedules" and "passengers". So, a database allows the organization and track passenger bookings, schedules, and associated passenger information.

### **Passenger functionalities overview :**

- Check routes and availability of seats.
- All passengers are required to submit valid contact details.
- All passengers must reserve a flight to travel. No on-the-gate sales are available. Also, customers can't buy tickets at airports, instead, they need to visit an online store or a sales counter to get a ticket or reserve for future plans.
- Passengers can book a flight by selecting a particular route.
- Passengers can view their booking history.
- Passengers can view current bookings.

- Passengers can cancel their existing bookings.
- Passengers can update their current bookings to any other schedules available for the same origin and destination.
- Passengers can also update their own details such as name, email, password, and city
- Passengers can only book or update the ticket only if the tickets left on the new schedule is greater than zero.

#### **Employee functionalities overview:**

- Can view all the bookings of all the passengers with their email.
- Can create and add a new schedule.
- Can update an existing schedule.
- Can delete a schedule that deletes all the bookings related to it.
- Can add new aircraft details.
- Can check if a passenger exists
- Can find all the details of a particular flight
- Can find all the passengers in a particular schedule

#### **Technical Specifications:**

**Database Type:** SQL (MySQL)

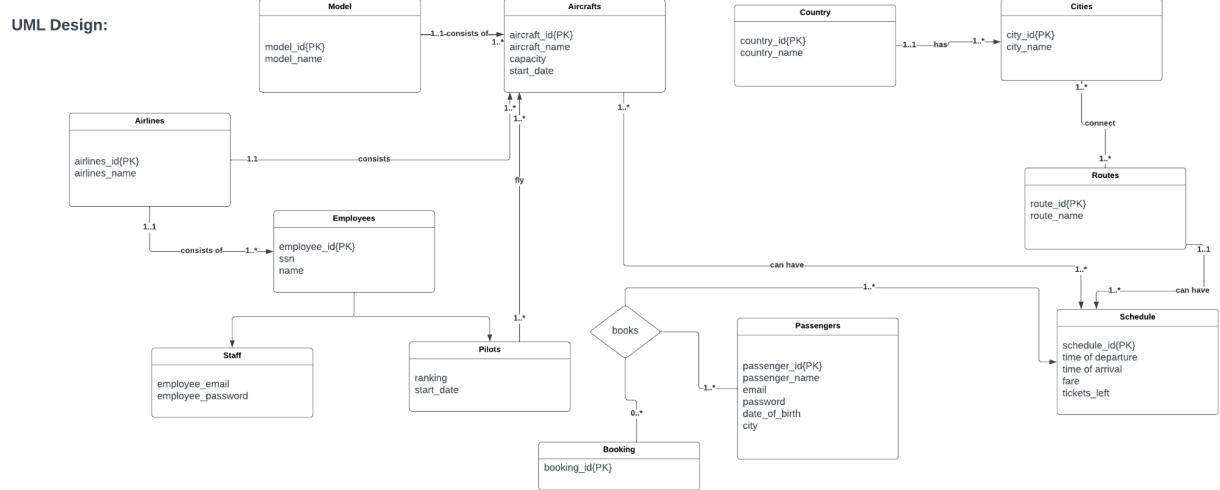
**Technologies used:** MySQL, Python, MYSQL Workbench for Scripts.

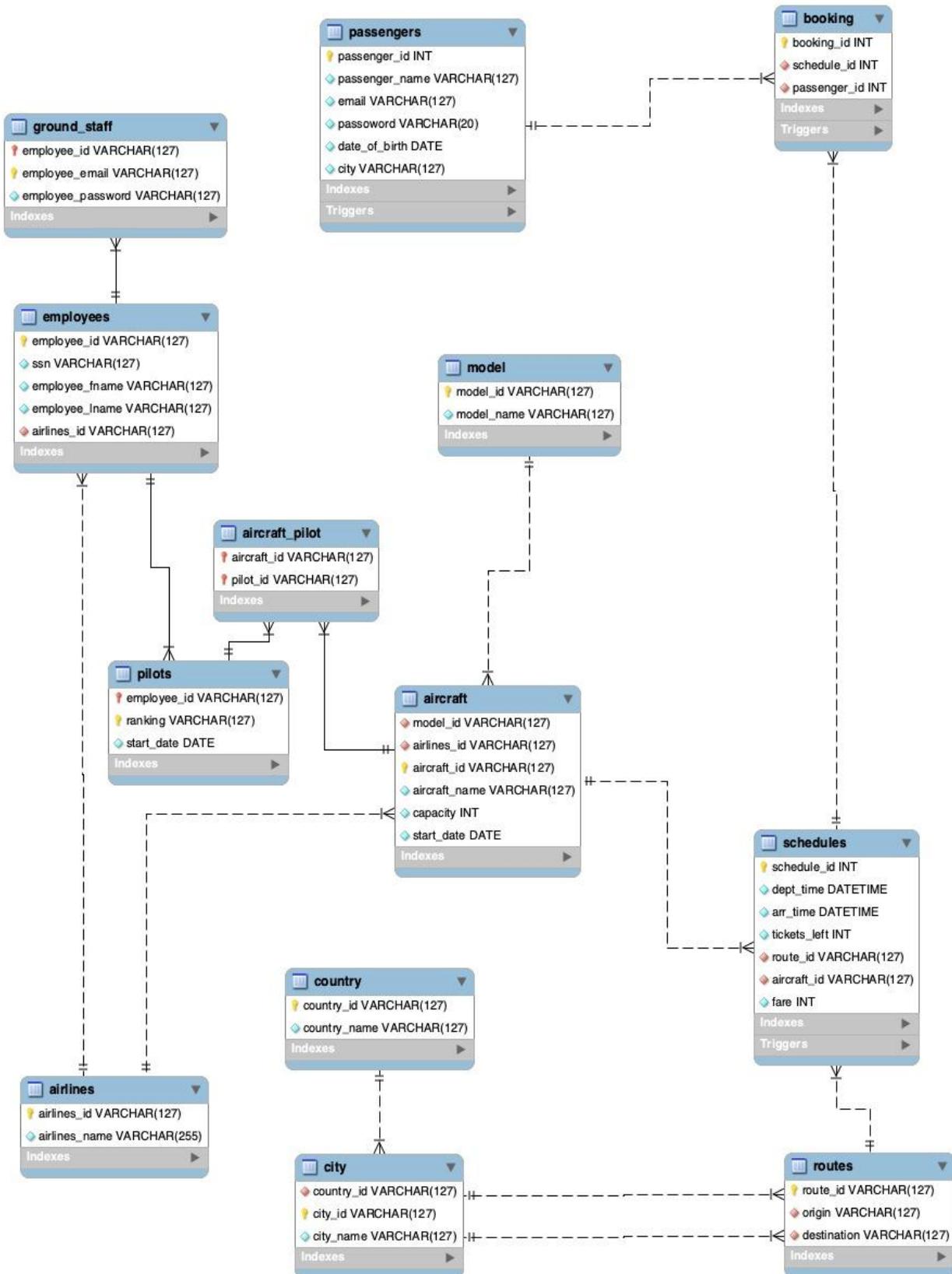
No machine restrictions for the project.

#### **Why does this project or this data domain interest you?**

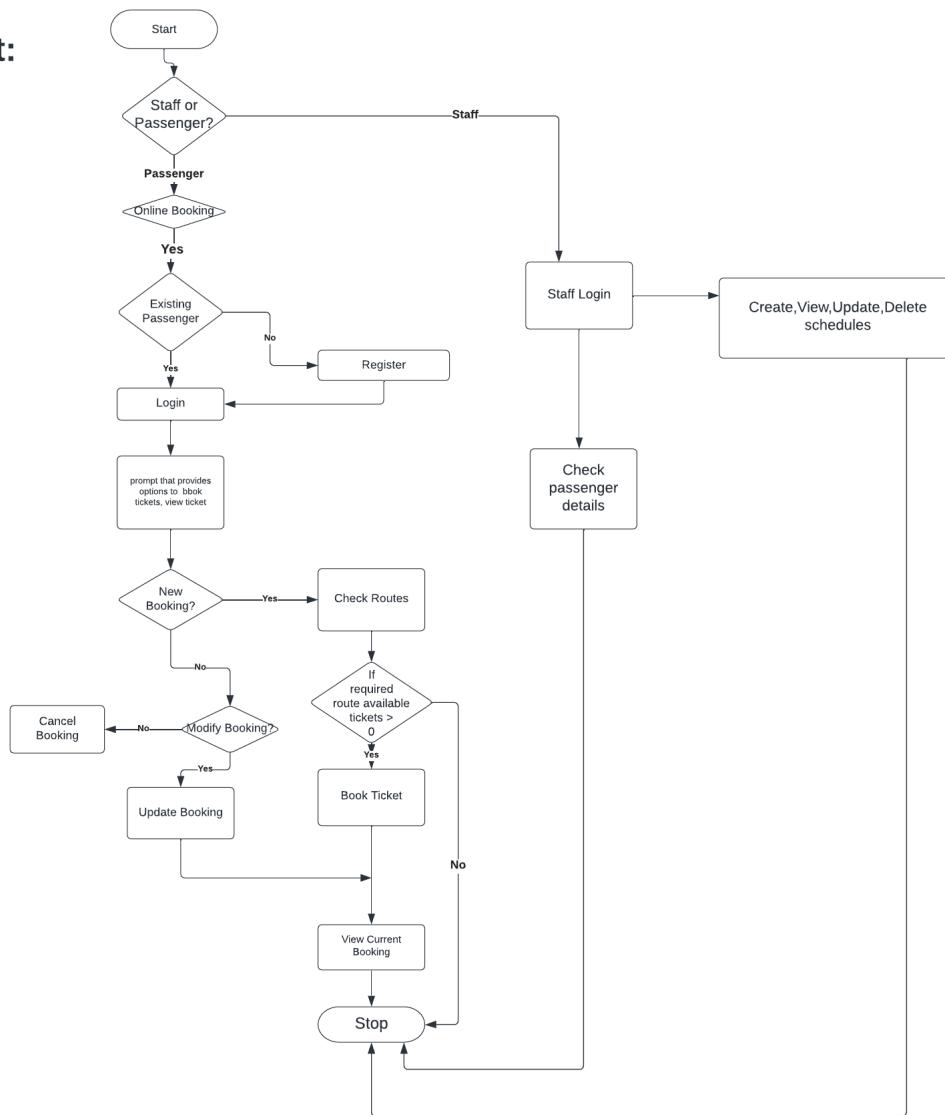
Flight management plays a vital role in the aviation industry, which is a major contributor to the global economy. It is crucial for airlines, airports, and other aviation-related businesses to manage flights efficiently and safely in order to succeed. This involves dealing with extensive amounts of data and using a database management system (DBMS) to maintain data consistency and security. The flight management process is complex and includes tasks such as scheduling flights, sales, transactions, and ticketing. Reliable data is essential for making high-quality decisions, and a DBMS provides a framework for ensuring data accuracy, leading to better decision-making. So, we saw great potential in working on this project and want to do it from scratch as it is really helpful to us and has vast data coverage.

## UML DESIGN





## Flow Chart:



## README:

This main code is written entirely in python and requires python to run and execute.

- Download the provided zip file to the local directory.
- Download and Install MYSQL
- Extract the downloaded project zip.

There are two SQL files note that chavachauhan.sql is our dump file and chauhanchava.sql is our script file.

- Now install all the dependencies using the following command :

python -m pip (only if pip is not installed)

**Please make sure you have prettytable and pymysql installed before you run the program**

**pip install prettytable**

**pip install pymysql**

- Once all the dependencies are installed run the files <project-root>/chavachauhan.sql or <project-root>/chauhanchava.sql to create the database and the related procedures, triggers and functions and insert the data.

You can also run the script in the SQL file in MYSQL workbench. Use chavachauhan.sql as the dump or the chauhanchava.sql as the main sql file and run it in the workbench before you start. Here you will find the ‘airport’ database.

- Then, run the app using the following command from the project root directory in the terminal.

`python3 app.py`

Follow the instruction from the prompts from there.

You can also use Jupyter Notebook or PyCharm Professional to run our program.

#### **STRUCTURE:**

- There are two types of users:
  - Staff: Staff can view the details of a particular flight, the passengers of a particular flight, Create a schedule, Update a schedule, Delete a schedule, View all the bookings of a particular passenger, View all the details of a passenger.
  - Passengers: Passengers can book a new schedule of flight, view their booking history, can cancel their booking, can update their booking to a different schedule on the same date, can update their own details.

#### **For Passengers:**

Please find the attached images for the sample run at the end of the report.

Register (For new Passenger):

Enter your name: Jonathan

Enter your email (should end with @gmail.com only): jonathan@gmail.com

Enter your password (must be at least 4 character): jon1234

Enter your date of birth in YYYY-MM-DD format: 1996-02-23

Enter your city: BOS

Login (For existing Passenger)

- You can login to the app with

Enter your email: jonathan@gmail.com

Enter your password: jon1234

■ To book the tickets --> 1

Enter the date in YYYY-MM-DD format: 2023-06-01

Enter the origin:NYC

Enter the destination:BEI

Enter the schedule id you want to book: 20

Do you want to confirm booking the ticket: Yes/No:Yes

View Your Bookings : Yes/No : Yes

■ To view Your Bookings ----> 2

What would you like to do?

- 1.Update a booking ---> 1
- 2.Cancel a booking ---> 2
- 3.Go back to Dashboard ---->3

Choose the option to update the ticket to any other flight or cancel the ticket.

- Update the passenger details---> 3

1.Name ----->1

2.City ----->2

3. Email ----->3

4.Password ----->4

Choose any details you want to update

### **For Staff User:**

Welcome to the staff dashboard!. What would you like to do?

- 1.Search a particular passenger--->1
- 2.Find all bookings of a particular passenger----> 2
- 3.Find all passengers in a particular flight---->3
- 4.Add New Aircraft details---->4
- 5.Delete a particular schedule----> 5
- 6.Create a particular schedule----> 6
- 7.Update a particular schedule----> 7
- 8. Get all details of the flight ---->8
- 9.Logout ---->9

Please Enter your choice:

- Search for a particular passenger's details

Enter the email of the passenger you want to search: [jonathan@gmail.com](mailto:jonathan@gmail.com)

- Find all bookings of a particular passenger.

Enter the email of the passenger you want to search: [jonathan@gmail.com](mailto:jonathan@gmail.com)

- Find all passengers on a particular flight

Enter the email of the passenger you want to search: [jonathan@gmail.com](mailto:jonathan@gmail.com)|

- Add New Aircraft details.

Enter model ID: 737

Enter model Name: Boeing 737

Enter airlines ID: UA

Enter airlines Name: United Airlines

Enter aircraft ID: N12345

Enter aircraft name: Spirit of America

Enter capacity: 344

Enter start date (YYYY-MM-DD): 2014-03-02

- Create a particular schedule

Select an aircraft for the new schedule: EK4001

Enter the departure date and time in the format YYYY-MM-DD HHH:MM:SS: 2023-04-05 23:00:00

Enter the arrival date and time in the format YYYY-MM-DD HHH:MM:SS: 2023-04-06 23:00:00

Enter the number of tickets left: 356

Enter the fare of each ticket: 700

■ Delete a particular schedule

Enter the schedule you want to delete: 1

■ Update a particular schedule

Enter the schedule you want to update: 2

Enter the new departure time in YYYY-MM-DD HH:MM:SS format: 2023-05-02 08:00:00

Enter the new arrival time in YYYY-MM-DD HH:MM:SS format: 2023-05-02 10:00:00

## LESSON LEARNED

❖ Time Management:

We spent considerable time writing all the required procedures, triggers, and functions.

We made sure that there is very little SQL code in the front end (only for staff login credentials match and passenger credentials match). So due to this, we had to write a lot of procedures.

In our airline database management project, our main concern was to make sure user finds the system easy to use. So we focused more on what the user might think when they start running the code. We had to make sure that all the prompts give enough hints for the user to understand what is happening and what they have to type for them to reach the next step.

To address this challenge, we prioritized the backend tasks and allocated more time to them. We first outlined the essential backend tasks, such as designing the database schema, creating tables, writing SQL queries, and implementing triggers, stored procedures, and error-handling mechanisms. We then set up a schedule with time slots for each task and tracked our progress to ensure that we met the project timeline.

Despite the time-consuming nature of writing Python scripts for the backend, we made sure to prioritize it over the scripting. We recognized that the backend was crucial for the proper functioning and management of the data within the airline database system. By dedicating more time and effort to the backend tasks, we were able to deliver a functional and robust airline database system within the project deadline.

❖ Technical expertise gained:

Python

MYSQL programming objects.

## FUTURE WORK

❖ Enhanced booking updates: In future work, the airline database management system could be enhanced to provide more updated options for flight bookings, such as changing flight routes, seats, or class upgrades. This could make the booking process more flexible for the customers and improve the overall user experience.

- ❖ Automated seat allocation: The airline database management system could be further improved by adding an automated seat allocation feature. This feature would allow passengers to choose seats based on their preferences, such as window or aisle seats, or it could automatically assign seats based on the passenger's preference or booking history.
- ❖ Staff access: The airline database management system could also be improved by providing staff members with access to update existing flight information. For example, staff could update the cost of flights, change the number of available seats or update the flight schedules in real-time. This would help airlines manage their flights more efficiently and ensure that the information provided to customers is accurate and up-to-date.
- ❖ Real-time analytics and reporting: The airline database management system could be further enhanced by adding real-time analytics and reporting capabilities. This would allow airlines to track flight and passenger data in real-time, analyze trends, and make informed decisions. For example, airlines could use analytics to optimize flight routes, improve pricing strategies, or personalize the customer experience.
- ❖ Improved security: In future work, the airline database management system could be improved by implementing more robust security measures. This would help protect sensitive data such as passenger information, flight schedules, and financial transactions from unauthorized access or cyber-attacks.

#### **Sample Run:**

##### **Register new passenger:**

```
[(base) renuchava@Renusrees-MacBook-Air ChauhanCChavaR_project % python3 app.py
Enter the username:root
Enter the password:Saila*2571
Welcome to the database. What would you like to do?
1. Login--->1
2. Staff Login--->2
3. Register---->3
Enter your choice: 3
Enter your name: Jonathan
Enter your email (should end with @gmail.com only): jonathan@gmail.com
Enter your password (must be atleast 4 character): pass1
Enter your date of birth in the format YYYY-MM-DD1996-05-27
Enter your city: Boston
Registration successful!
```

## Passenger Login and book tickets

```
Welcome to the database. What would you like to do?
1. Login--->1
2. Staff Login--->2
3. Register--->3
Enter your choice: 1
Enter your email: jonathan@gmail.com
Enter your password: pass1
Login successful!
Welcome, to the dashboard Jonathan!. What would you like to do?
1. Book Tickets ---> 1
2. View Your Bookings ----> 2
3. Update user details---> 3
4. Logout---> 4
Enter your choice: 1
Enter the date in YYYY-MM-DD format: 2023-06-01
Select origin and destination from the list
['NYC', 'NYC']
['BEI', 'BEI']
Enter the origin:NYC
Enter the destination:BEI
+-----+
| schedule_id | Time_of_Departure | Time_of_Arrival | route_id | origin | destination | fare | tickets_left |
+-----+
|    19       | 2023-06-01 09:00:00 | 2023-06-01 11:00:00 |   R10    | NYC    | BEI      | 180  |      1      |
|    20       | 2023-06-01 11:00:00 | 2023-06-01 13:00:00 |   R10    | NYC    | BEI      | 800  |      2      |
+-----+
Enter the schedule id you want to book: 19
Do you want to confirm booking the ticket: Yes/No:yes
Ticket booked successfully!
View Your Bookings : Yes/No
Enter your choice: yes
+-----+
| passenger_id | passenger_name |   email    | booking_id | schedule_id | dept_time | arr_time | origin | destination | fare |
+-----+
|     4        |   Jonathan    | jonathan@gmail.com |      3      |      19     | 2023-06-01 09:00:00 | 2023-06-01 11:00:00 | NYC    | BEI      | 180  |
+-----+
Login successful!
Welcome, to the dashboard Jonathan!. What would you like to do?
1. Book Tickets ---> 1
2. View Your Bookings ----> 2
3. Update user details---> 3
4. Logout---> 4
Enter your choice: ■
```

```
Welcome, to the dashboard Jonathan!. What would you like to do?
1. Book Tickets ---> 1
2. View Your Bookings ----> 2
3. Update user details---> 3
4. Logout---> 4
Enter your choice: 1
Enter the date in YYYY-MM-DD format: 2023-05-1
Select origin and destination from the list
['NYC', 'NYC']
['TOR', 'TOR']
Enter the origin:NYC
Enter the destination:TOR
+-----+
| schedule_id | Time_of_Departure | Time_of_Arrival | route_id | origin | destination | fare | tickets_left |
+-----+
|    1       | 2023-05-01 10:00:00 | 2023-05-01 12:00:00 |   R01    | NYC    | TOR      | 200  |      1      |
|    2       | 2023-05-01 14:00:00 | 2023-05-01 16:00:00 |   R01    | NYC    | TOR      | 180  |     149      |
+-----+
Enter the schedule id you want to book: 1
Do you want to confirm booking the ticket: Yes/No:yes
Ticket booked successfully!
View Your Bookings : Yes/No
Enter your choice: yes
+-----+
| passenger_id | passenger_name |   email    | booking_id | schedule_id | dept_time | arr_time | origin | destination | fare |
+-----+
|     4        |   Jonathan    | jonathan@gmail.com |      3      |      19     | 2023-05-01 10:00:00 | 2023-05-01 12:00:00 | NYC    | TOR      | 200  |
|     4        |   Jonathan    | jonathan@gmail.com |      4      |      1       | 2023-05-01 14:00:00 | 2023-05-01 16:00:00 | NYC    | TOR      | 180  |
+-----+
Login successful!
Welcome, to the dashboard Jonathan!. What would you like to do?
1. Book Tickets ---> 1
2. View Your Bookings ----> 2
3. Update user details---> 3
4. Logout---> 4
Enter your choice: ■
```

## Update booking:

```
Welcome, to the dashboard Jonathan!. What would you like to do?
1. Book Tickets ---> 1
2. View Your Bookings ---> 2
3. Update user details---> 3
4. Logout---> 4
Enter your choice: 2
+-----+-----+-----+-----+-----+-----+-----+-----+
| passenger_id | passenger_name | email | booking_id | schedule_id | dept_time | arr_time | origin | destination | fare |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 4 | Jonathan | jonathan@gmail.com | 3 | 19 | 2023-06-01 09:00:00 | 2023-06-01 11:00:00 | NYC | BEI | 180 |
| 4 | Jonathan | jonathan@gmail.com | 4 | 1 | 2023-05-01 10:00:00 | 2023-05-01 12:00:00 | NYC | TOR | 200 |
+-----+-----+-----+-----+-----+-----+-----+-----+
What would you like to do?
1.Update a booking ---> 1
2.Cancel a booking ---> 2
3.Go back to Dashboard --->3
Please enter your choice of action :1
Please retype the booking id that you want to update3
+-----+-----+-----+-----+-----+-----+-----+
| schedule_id | Time_of_Departure | Time_of_Arrival | route_id | origin | destination | fare | tickets_left |
+-----+-----+-----+-----+-----+-----+-----+
| 19 | 2023-06-01 09:00:00 | 2023-06-01 11:00:00 | R10 | NYC | BEI | 180 | 0 |
| 20 | 2023-06-01 11:00:00 | 2023-06-01 13:00:00 | R10 | NYC | BEI | 800 | 2 |
+-----+-----+-----+-----+-----+-----+-----+
Enter the schedule id you want to book: 20
Booking updated
+-----+-----+-----+-----+-----+-----+-----+
| passenger_id | passenger_name | email | booking_id | schedule_id | dept_time | arr_time | origin | destination | fare |
+-----+-----+-----+-----+-----+-----+-----+
| 4 | Jonathan | jonathan@gmail.com | 3 | 20 | 2023-06-01 11:00:00 | 2023-06-01 13:00:00 | NYC | BEI | 800 |
| 4 | Jonathan | jonathan@gmail.com | 4 | 1 | 2023-05-01 10:00:00 | 2023-05-01 12:00:00 | NYC | TOR | 200 |
+-----+-----+-----+-----+-----+-----+-----+
What would you like to do?
1.Update a booking ---> 1
2.Cancel a booking ---> 2
3.Go back to Dashboard --->3
Please enter your choice of action :■
```

## Delete booking:

```
Enter your choice: 2
+-----+-----+-----+-----+-----+-----+-----+
| passenger_id | passenger_name | email | booking_id | schedule_id | dept_time | arr_time | origin | destination | fare |
+-----+-----+-----+-----+-----+-----+-----+
| 4 | Jonathan | jonathan@gmail.com | 3 | 19 | 2023-06-01 09:00:00 | 2023-06-01 11:00:00 | NYC | BEI | 180 |
| 4 | Jonathan | jonathan@gmail.com | 4 | 1 | 2023-05-01 10:00:00 | 2023-05-01 12:00:00 | NYC | TOR | 200 |
+-----+-----+-----+-----+-----+-----+-----+
What would you like to do?
1.Update a booking ---> 1
2.Cancel a booking ---> 2
3.Go back to Dashboard --->3
Please enter your choice of action :1
Please retype the booking id that you want to update3
+-----+-----+-----+-----+-----+-----+-----+
| schedule_id | Time_of_Departure | Time_of_Arrival | route_id | origin | destination | fare | tickets_left |
+-----+-----+-----+-----+-----+-----+-----+
| 19 | 2023-06-01 09:00:00 | 2023-06-01 11:00:00 | R10 | NYC | BEI | 180 | 0 |
| 20 | 2023-06-01 11:00:00 | 2023-06-01 13:00:00 | R10 | NYC | BEI | 800 | 2 |
+-----+-----+-----+-----+-----+-----+-----+
Enter the schedule id you want to book: 20
Booking updated
+-----+-----+-----+-----+-----+-----+-----+
| passenger_id | passenger_name | email | booking_id | schedule_id | dept_time | arr_time | origin | destination | fare |
+-----+-----+-----+-----+-----+-----+-----+
| 4 | Jonathan | jonathan@gmail.com | 3 | 20 | 2023-06-01 11:00:00 | 2023-06-01 13:00:00 | NYC | BEI | 800 |
| 4 | Jonathan | jonathan@gmail.com | 4 | 1 | 2023-05-01 10:00:00 | 2023-05-01 12:00:00 | NYC | TOR | 200 |
+-----+-----+-----+-----+-----+-----+-----+
What would you like to do?
1.Update a booking ---> 1
2.Cancel a booking ---> 2
3.Go back to Dashboard --->3
```

## Update Passenger Details:

```
Welcome, to the dashboard Jonathan King!. What would you like to do?
1. Book Tickets ----> 1
2. View Your Bookings ----> 2
3. Update user details----> 3
4. Logout---> 4
Enter your choice: 3
Select from the following to update
1.Name ----->1
2.Password ----->2
3.Email ----->3
4.City ----->4
Select the details you want to update:2
Enter the new passwordpass2
Updated Password Succesfully
Login successful!
Welcome, to the dashboard Jonathan King!. What would you like to do?
1. Book Tickets ----> 1
2. View Your Bookings ----> 2
3. Update user details----> 3
4. Logout---> 4
Enter your choice: ■
```

## Staff Login and find passenger's bookings:

```
Welcome to the database. What would you like to do?
1. Login--->1
2. Staff Login--->2
3. Register---->3
Enter your choice: 2
Enter your email: sanjana@gmail.com
Enter your password: 1234
Login Succesful
Welcome to the staff dashboard!.What would you like to do?
1.Search a particular passenger---->1
2.Find all bookings of a particular passenger----> 2
3.Find all passengers in a particular flight---->3
4.Add New Aircraft details---->4
5.Delete a particular schedule----> 5
6.Create a particular schedule----> 6
7.Update a particular schedule----> 7
8. Get all details of the flight ---->8
9.Logout ---->9
Please Enter your choice:
Enter the email of the passenger you want to search:jonathan@gmail.com
+-----+
| passenger_id | passenger_name |   email      | booking_id | schedule_id |      dept_time    |      arr_time     | origin | destination | fare |
+-----+
|      4       | Jonathan King | jonathan@gmail.com |      4       |        1      | 2023-05-01 10:00:00 | 2023-05-01 12:00:00 | NYC   | TOR       | 200  |
+-----+
```

## Find all details of a schedule:

```
1.Search a particular passenger---->1
2.Find all bookings of a particular passenger----> 2
3.Find all passengers in a particular flight---->3
4.Add New Aircraft details---->4
5.Delete a particular schedule----> 5
6.Create a particular schedule----> 6
7.Update a particular schedule----> 7
8. Get all details of the flight ---->8
9.Logout ---->9
Please Enter your choice:
[1, 2, 15, 3, 4, 16, 5, 6, 17, 7, 8, 18, 9, 10, 11, 12, 13, 14, 19, 20]
Enter the schedule you want to update: 1
+-----+
| schedule_id | dept_time      | arr_time      | tickets_left | fare | route_id | origin | destination | model_id | model_name | aircraft_id | aircraft_name | capacity | start_date | airlines_id | airlines_name |
+-----+
|      1       | 2023-05-01 10:00:00 | 2023-05-01 12:00:00 |        0      | 200  | R01     | NYC    | TOR       | AA1001  | Boeing 787  | Boeing 737-800 | 160    | 2021-09-01 | AA          |
+-----+
```

## Find all passengers in a particular schedule:

```
Login Successful
Welcome to the staff dashboard!.What would you like to do?
1.Search a particular passenger---->1
2.Find all bookings of a particular passenger----> 2
3.Find all passengers in a particular flight---->3
4.Add New Aircraft details---->4
5.Delete a particular schedule----> 5
6.Create a particular schedule----> 6
7.Update a particular schedule----> 7
8. Get all details of the flight ---->8
9.Logout ---->9
Please Enter your choice:
[1, 2, 15, 3, 4, 16, 5, 6, 17, 7, 8, 18, 9, 10, 11, 12, 13, 14, 19, 20]
Enter the schedule_id of the flight: 1
+-----+
|      email      | passenger_name | booking_id | schedule_id |
+-----+
| renusrechava@gmail.com | Rensuree Chava |      1       |        1      |
|  jonathan@gmail.com  | Jonathan King  |      4       |        1      |
+-----+
```

## Update a schedule:

```
Welcome to the staff dashboard! What would you like to do?  
1.Search a particular passenger---->1  
2.Find all bookings of a particular passenger---->2  
3.Find all passengers in a particular flight---->3  
4.Add New Aircraft details---->4  
5.Delete a particular schedule---->5  
6.Create a particular schedule----> 6  
7.Update a particular schedule----> 7  
8. Get all details of the flight ---->8  
9.Logout ---->9  
Please Enter your choice:  
[1, 2, 15, 3, 4, 16, 5, 6, 17, 7, 8, 18, 9, 10, 11, 12, 13, 14, 19, 20]  
Enter the schedule you want to update: 1  
Enter the new departure time in YYYY-MM-DD HH:MM:SS format: 2023-06-01 20:00:00  
Enter the new arrival time in YYYY-MM-DD HH:MM:SS format: 2023-06-01 22:00:00  
Schedule Updated
```

## Delete a Schedule:

```
Please Enter your choice:  
[1, 2, 15, 3, 4, 16, 5, 6, 17, 7, 8, 18, 9, 10, 11, 12, 13, 14, 19, 20]  
Enter the schedule you want to delete:1  
Schedule deleted  
Login Success  
Welcome to the staff dashboard! What would you like to do?  
1.Search a particular passenger---->1  
2.Find all bookings of a particular passenger----> 2  
3.Find all passengers in a particular flight---->3  
4.Add New Aircraft details---->4  
5.Delete a particular schedule---->5  
6.Create a particular schedule----> 6  
7.Update a particular schedule----> 7  
8. Get all details of the flight ---->8  
9.Logout ---->9  
Please Enter your choice:  
Enter the email of the passenger you want to search:jonathan@gmail.com  
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+  
| passenger_id | passenger_name | email | booking_id | schedule_id | dept_time | arr_time | origin | destination | fare |  
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+  
| 4 | Jonathan King | jonathan@gmail.com | None |
```

Here Jonathan had a seat on a flight with schedule 1. Now since schedule 1 was canceled by a staff member his booking has also been canceled.