Create an Airline management application on the Command Line/Simple Web App which will follow the following protocol.

User is first presented the following options:

MAIN:
Welcome to the Utopia Airlines Management System. Which category of a user are you

1) Employee/Agent
2) Administrator
3) Traveler

   <take input>

Based on what the user selects, the following prompts will appear to fit his role:

Employee

EMP1:
1) Enter Flights You Manage
2) Quit to previous **(should take you menu MAIN)**

   <take input>

For Option 1, Give a list of Flights using the names of the Origin and Destination Airports as well as the Cities they are located in:

EMP2:
1) LAX, Los Angeles → JFK, New York
2) DFW, Dallas → IAD, Washington DC
3) SEA, Seattle → MIA, Miami
4) etc.
5) Quit to previous **(should take you menu EMP1)**

   <take input>

The user will have a minimum of 10 options to select from.

The user will only pick the number in the above list and you should figure out which flight he is referring to.

Based on the selection, the next menu would be:

EMP3:
1) View more details about the flight
2) Update the details of the Flight
3) Add Seats to Flight
4) Quit to previous **(should take you menu EMP2)**

<take input>

Option 1 should retrieve flight details for the Flight the user selected. This should be like:

You have chosen to view the Flight with Flight Id: X and Departure Airport: A and Arrival Airport: B.

Departure Airport: A | Arrival Airport: B |
Departure Date: C | Departure Time: D |
Arrival Date: E | Arrival Time: F

Available Seats by Class:
1) First → X
2) Business → Y
3) Economy → Z
4) Return to previous menu **(should take you menu EMP3)**
<take input>

Then go back to menu **EMP3,** to start over again.

Option 2 should update **flights** table for the Flight he/she had picked before. This should be like:

You have chosen to update the Flight with Flight Id: X and Flight Origin: A and Flight Destination: B.
Enter 'quit' at any prompt to cancel operation.

Please enter new Origin Airport and City or enter N/A for no change:
….
….
<take input>
Please enter new Destination Airport and City or enter N/A for no change:
<take input>
Please enter new Departure Date or enter N/A for no change:
<take input>
Please enter new Departure Time or enter N/A for no change:
<take input>
Please enter new Arrival Date or enter N/A for no change:
<take input>

Then update the values for this flight and say successfully updated. Then go back to menu **EMP3,** to start over again.

Option 3 should give further options like this:

Then you should update the **flight_seats** table with the new values. Then take the user back to **EMP3.**

TRAVELER

Enter the your Membership Number:
<take input>

Don't let the traveler proceed until he gives a valid membership number.

TRAV1:
1) Book a Ticket
2) Cancel an Upcoming Trip
3) Quit to Previous **(should take you to menu MAIN)**

Option 1 should show the following options

Pick the Flight you want to book a ticket for:
1) LAX, Los Angeles → JFK, New York
2) ......
3) ......
4) ......
5) Quit to previous **(should take you menu TRAV1)**

Pick the Seat you want to book a ticket for **(make sure you only show seats that have at least one available in Seat Class in the flight picked)**

1) View Flight Details
2) First
3) Business
4) Economy
5) Quit to cancel operation **(should take you menu TRAV1)**

Option 1 should retrieve flight details for the Flight the user selected.

Then add entry into your tickets table.

Implement the cancel trip functionality the same way as above.

ADMINISTRATOR

Implement the following ADMINISTRATOR functions: (CRUD Operations)

1) **Add/Update/Delete/Read Flights**
2) **Add/Update/Delete/Read Seats**
3) **Add/Update/Delete/Read Tickets and Passengers**
4) **Add/Update/Delete/Read Airports**
5) **Add/Update/Delete/Read Travelers**
6) **Add/Update/Delete/Read Employees**
7) **Over-ride Trip Cancellation for a ticket.**

Add Flights
a. **Origin/Departure Airports**
   1.
   2.
   …
b. **Arrival/Destination Airports**
   1.
   2.
   …
   User picks an Airport or creates a new one.
c. **List of Departure Dates**
   1.
   2.
   …
   User picks one or more date.
d. **List of Departure Times**
   Users picks one departure time or creates a new one

Use Transactions