**Graphical user interface, website

Description automatically generated**

**Introduction:**

One of the most common medium of traveling is by air, so that Airports have become one of the most crowded platform, having a load of passengers, going in and out of it, flying via airplanes of different airlines, and traveling to and from different countries. So that a sophisticated management system is required in order to manage every operation happening inside a airport effectively and efficiently.

**Description:**

Since we need to keep record of data such as which flight is taking off or landing at what date and time, and from or to what city and airport, containing how much number of passengers , and belonging to which airlines, so that an effective database system is needed to manage and keep record of everything efficiently.

**Database Schema:**

Table: **Airlines**

|  |  |  |  |
| --- | --- | --- | --- |
| Entity |  | Attribute | Constraint |
| Airline ID | Number |  | Primary Key |
| Airline Name | Varchar2 |  | Not Null |
| Airline Alias | Varchar2 |  | Not Null |
| Country | Varchar2 |  | Not Null |

Table: **Department**

|  |  |  |  |
| --- | --- | --- | --- |
| Entity |  | Attribute | Constraint |
| Department ID | Number |  | Primary Key |
| Department Name | Varchar2 |  | Not Null |

Table: **Employees**

|  |  |  |  |
| --- | --- | --- | --- |
| Entity |  | Attribute | Constraint |
| Employee ID | Number |  | Primary Key |
| Employee Name | Varchar2 |  | Not Null |
| Department ID | Number |  | Foreign Key (Department) |
| Salary | Number |  |  |

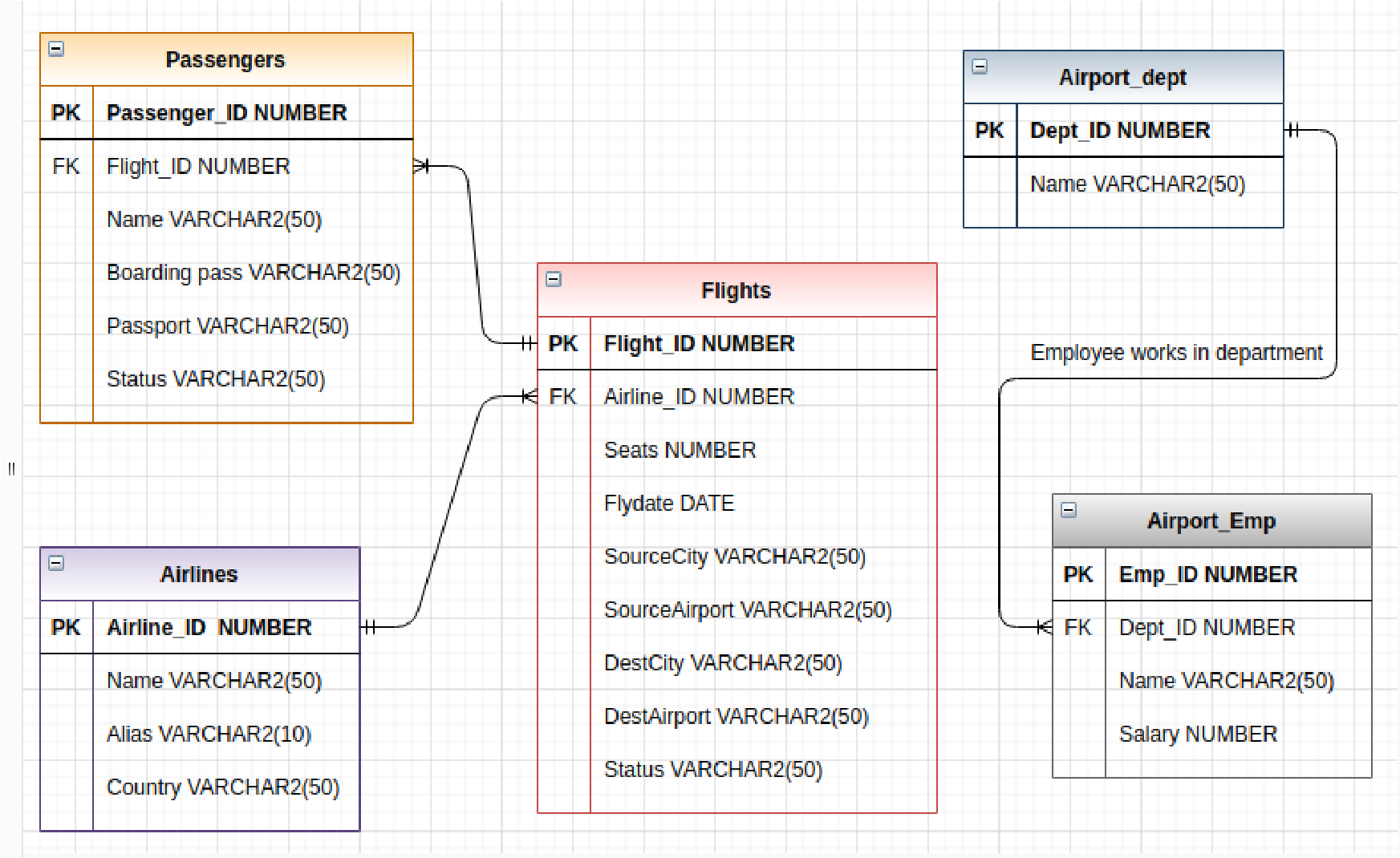
Table: **Flights**

|  |  |  |  |
| --- | --- | --- | --- |
| Entity |  | Attribute | Constraint |
| Flight ID | Number |  | Primary Key |
| Airline ID | Number |  | Foreign Key (Airlines) |
| Fly Date | Date | | Not Null |
| Seats | Number | | Not Null |
| Source City | Varchar2 | | Not Null |
| Source Airport | Varchar2 | | Not Null |
| Destination City | Varchar2 | | Not Null |
| Destination Airport | Varchar2 | | Not Null |
| Status | Varchar2 | | Not Null (In/Out) |

Table: **Passengers**

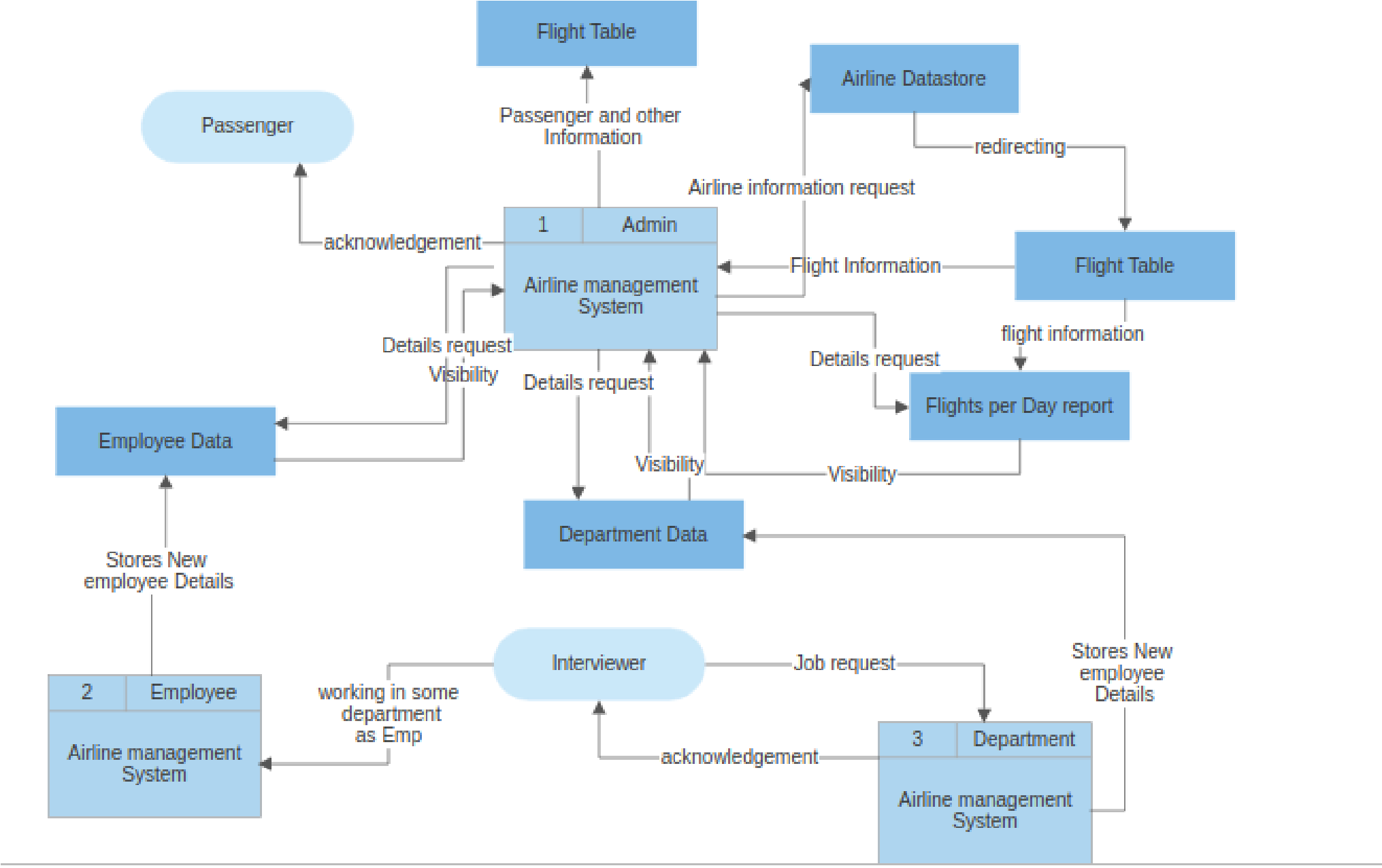
|  |  |  |  |
| --- | --- | --- | --- |
| Entity |  | Attribute | Constraint |
| Passenger ID | Number |  | Primary Key |
| Passenger Name | Varchar2 |  | Not Null |
| Passport | Varchar2 |  | Not Null |
| Boarding Pass | Varchar2 |  | Not Null |
| Flight ID | Number |  | Foreign Key (Flights) |
| Status | Varchar2 |  | Not Null (In/Out) |

**Entity Relationship Diagram (Crow’s Foot Notation):**



Tool Used: **Draw.io**

**Data Flow Diagram (Level-0):**

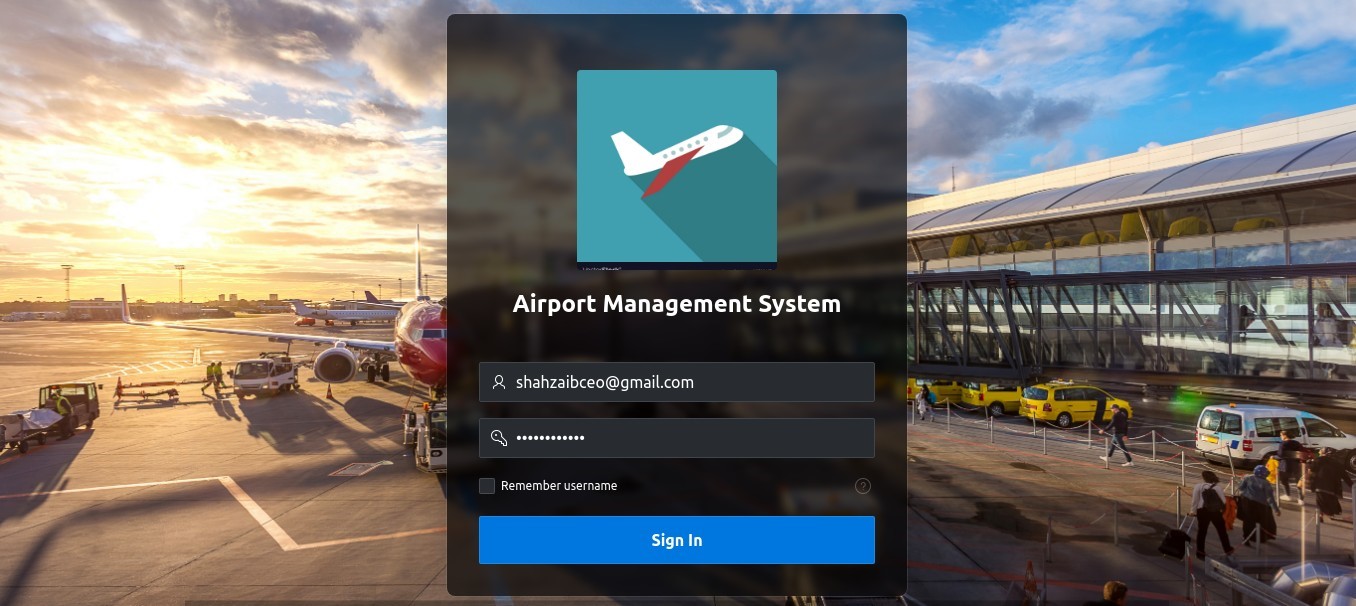


Tool Used: **Smart Draw**

**Functionality and Features of App:**

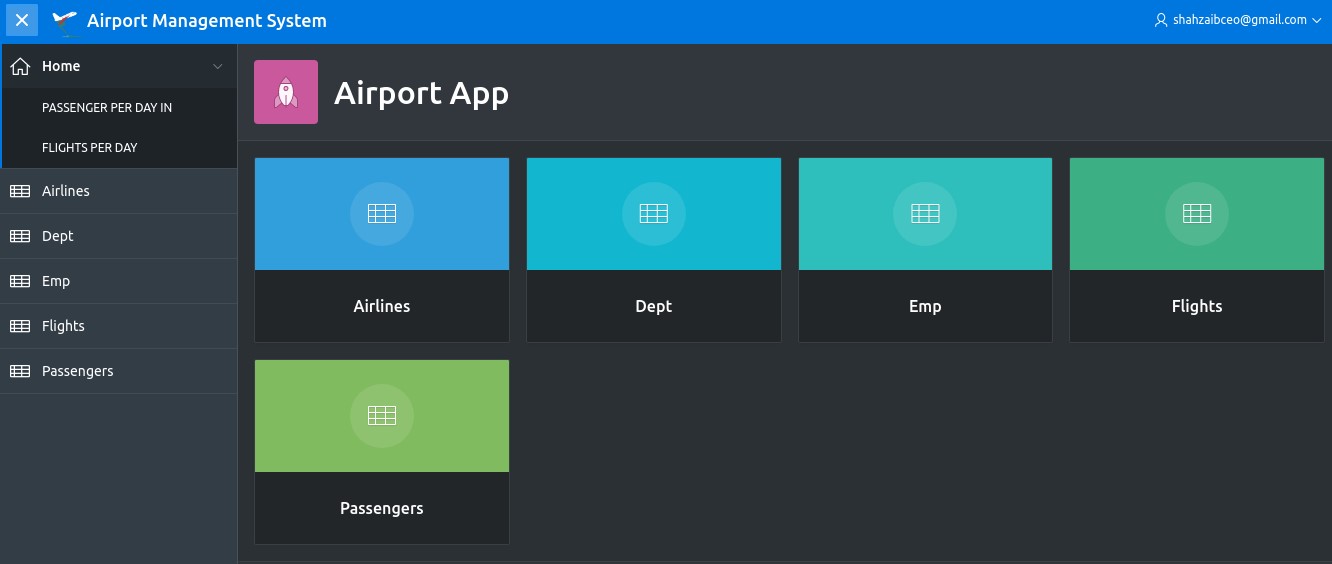
**1. Login Page:**

Login Screen lets the admin get logged in to the Airport Management System App, and view and modify all the available features. Once logged in, he/she can use the entire Airport Management System App. He can perform CRUD (Create Read Update Delete) operations on all the tables provided in the database. He can view multiple reports available, he can also view analysis of data by means of multiple charts



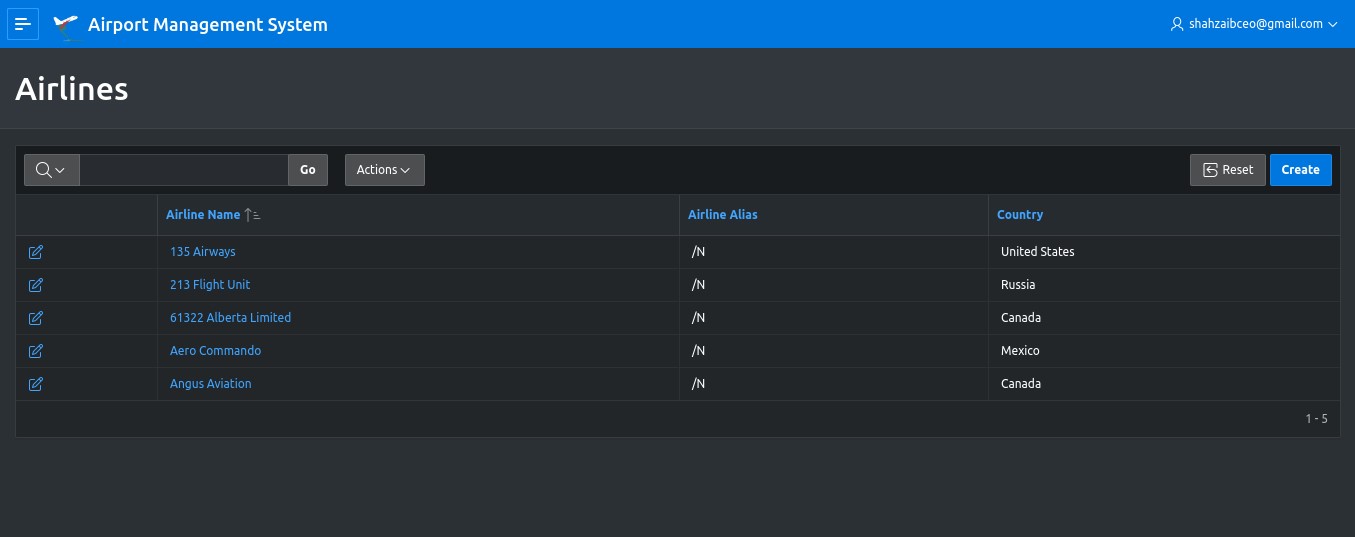
Login Screen of AMS

# 2. Home Page

After Logging In, the App will be redirected to the Home Page, from where admin can redirect to multiple different pages, offering different features, and serving different purpose in the Airport Management System App. From here Admin can either redirect into Airlines Page, Department Page, Employees Page, Flights Page, Passengers Page, and reports like Passengers per day and Flights per day.

# 3. Airlines Page

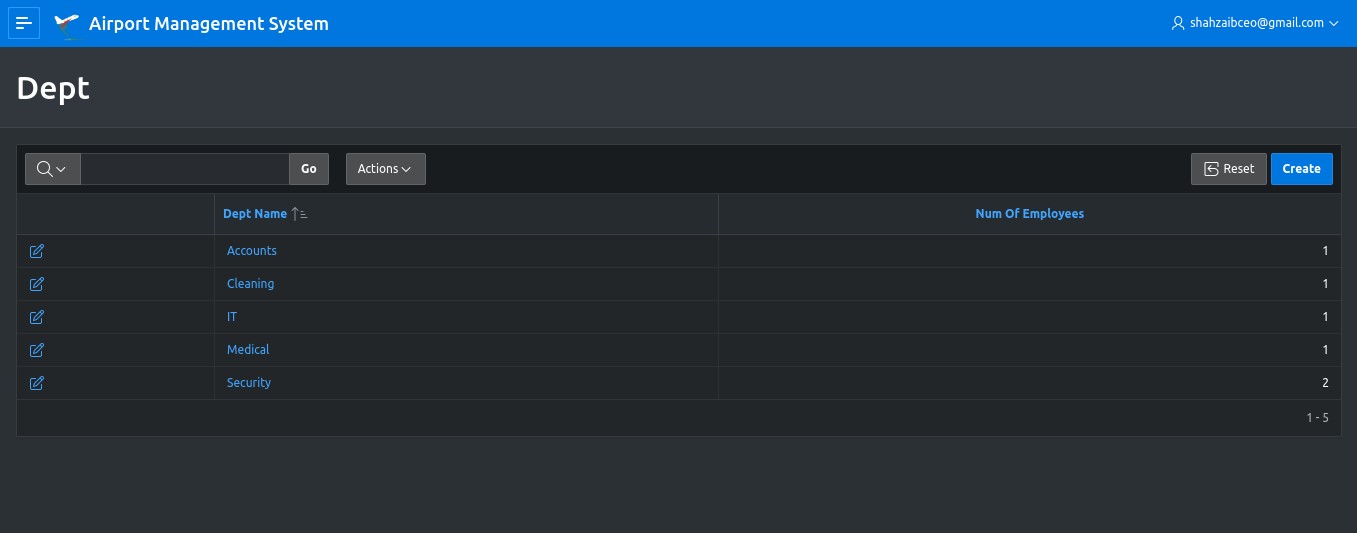
Airlines Page offers the CRUD(Create Read Update Delete) operations upon the Airlines Table. It also offers searching a particular row, via GUI, also admin can filter the columns upon different conditions.



Airlines Page of AMS

**4. Departments Page:**

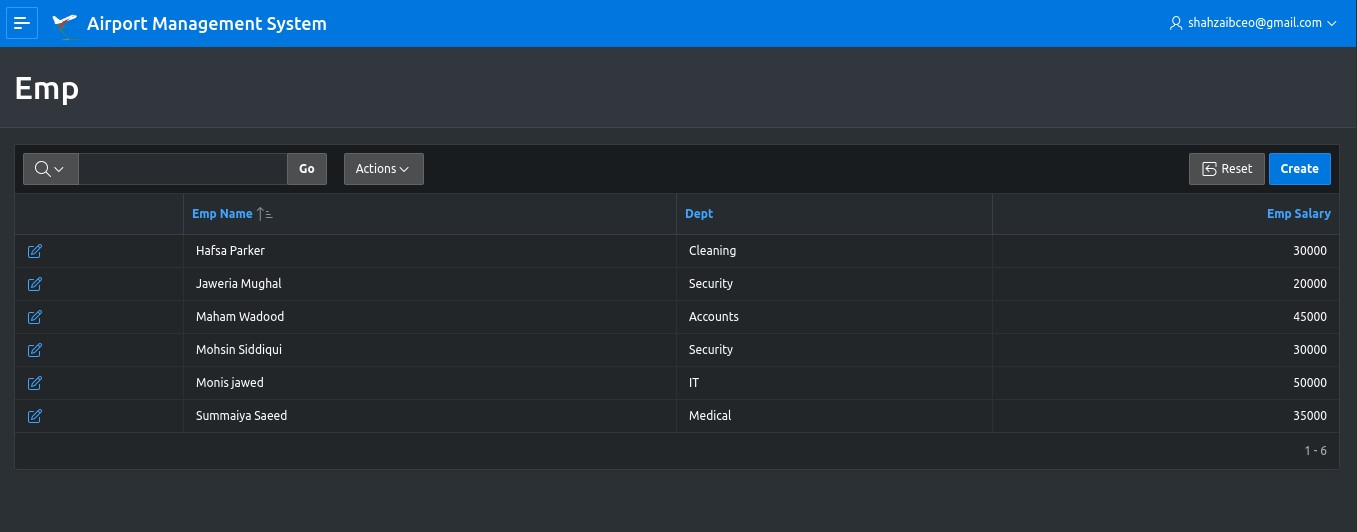
Department Page offers the CRUD(Create Read Update Delete) operations upon the Departments Table. It also offers searching a particular row, via GUI, also admin can filter the columns upon different conditions. Further It also shows the Number of Employees in each department (means count of employees having each Department’s ID as a Foreign key in their row).



Department Page of AMS

# 5. Employees Page

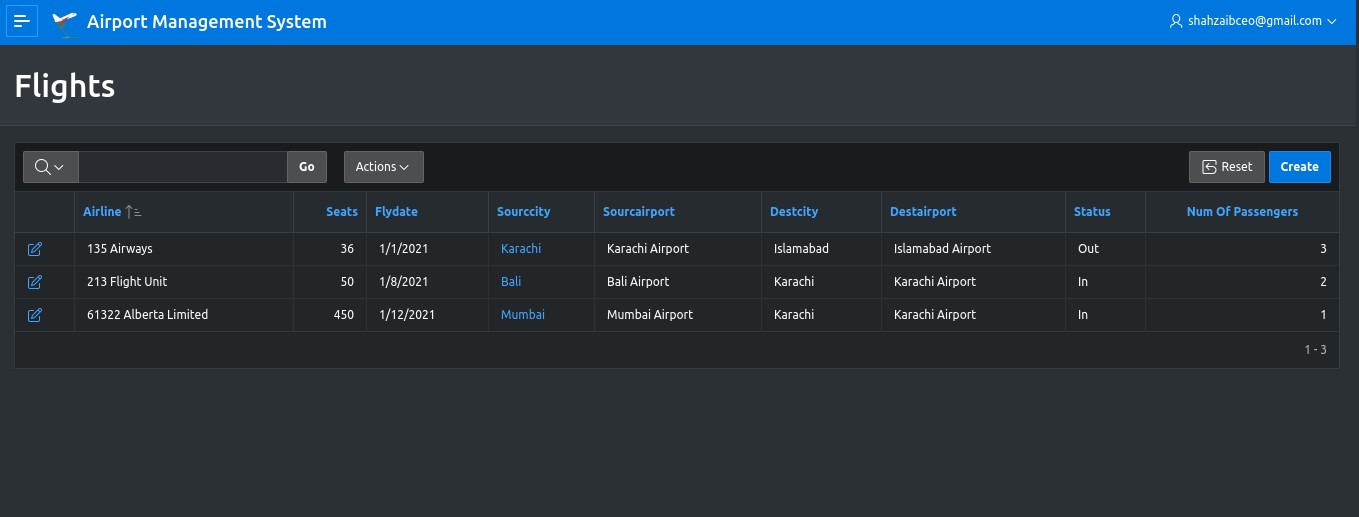
Employees Page offers the CRUD(Create Read Update Delete) operations upon the Employees Table. It also offers searching a particular row, via GUI, also admin can filter the columns upon different conditions. It also has Department name in which each employee works or belongs to. The department ID is fed in the Employees table as a Foreign key of Department Table, and against that ID the name of Department is being retrieved from the Department Table for each of the employee.



Employees Page of AMS

# 6. Flights Table

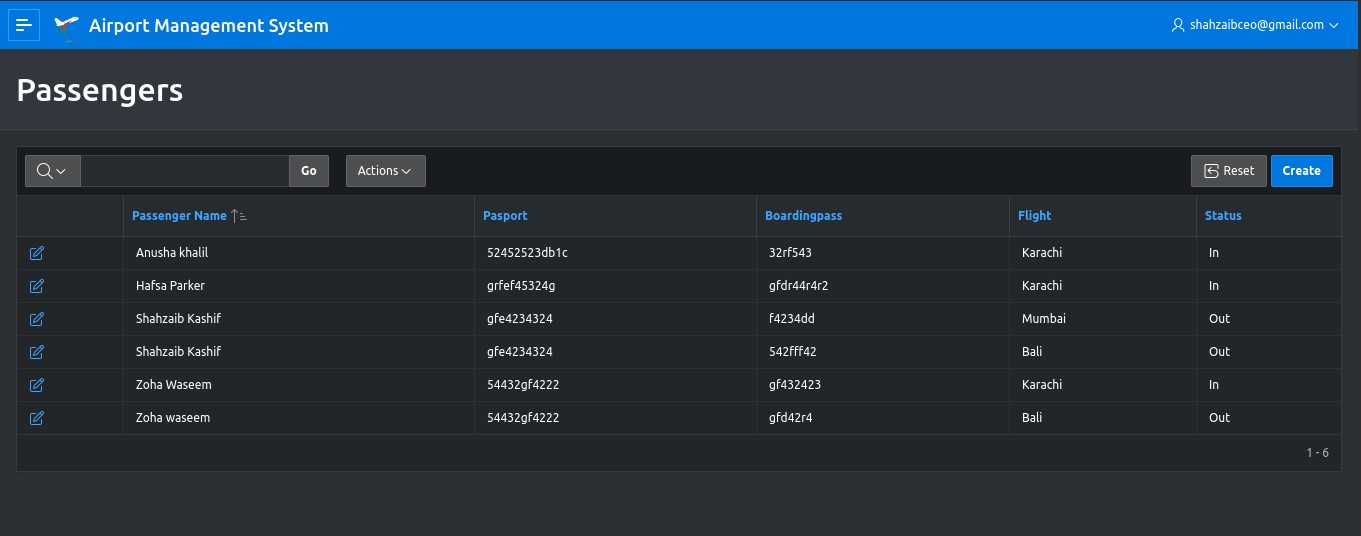
Flights table offers the CRUD(Create Read Update Delete) operations upon the Flights Table. It also offers searching a particular row, via GUI, also admin can filter the columns upon different conditions. Further It also shows the Number of Passengers in each Flight (means count of passengers having each Flight’s ID as a Foreign key in their row).



Flights Page of AMS

# 7. Passengers Page

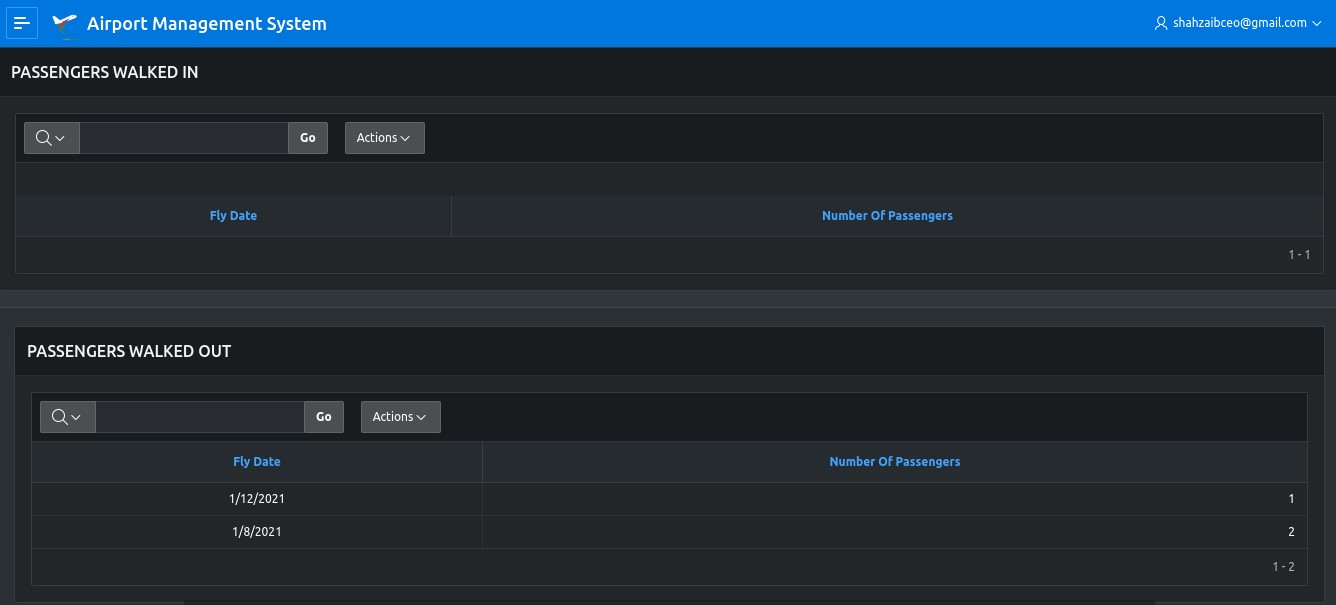
Airlines Page offers the CRUD(Create Read Update Delete) operations upon the Airlines Table. It also offers searching a particular row, via GUI, also admin can filter the columns upon different conditions. It further has a Flight column, that shows the source city from the Flights table, against the Flight ID foreign key in each record of Passenger.



Passengers Page of AMS

# 8. Passengers Per Day Report Page

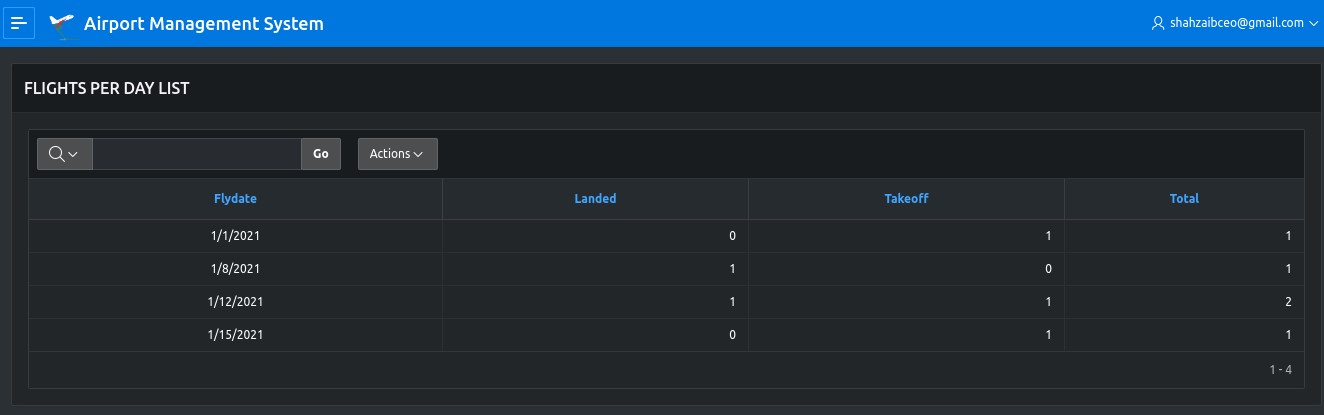
Passengers Per Day Report offers the Number of Passengers walked in and out of the Airport for each Fly Date in the Flights record. It basically counts the Passengers ID for each Fly Date, with a Status of ‘In’, in the Passengers record, for counting the Number of Passengers walked in the Airport. Similarly it counts the Passengers ID for each Fly Date, with a Status of ‘Out’, in the Passengers record, for counting the Number of Passengers walked out of the Airport.



Passengers Per Day Report Page of AMS

# 9. Flights Per Day Report Page

Flights per day report page has two sections. First section offers the Report of Number of Flights landed in the airport and took off from the airport, also total flights that occurred in the airport for each of the Fly Date in the Flights Record. It basically counts the Flights ID for each Fly Date, with a Status of ‘In’, in the Flights record, for counting the Number of Flights landed in the Airport. Similarly it counts the Flight ID for each Fly Date, with a Status of ‘Out’, in the Flights record, for counting the Number of Flights took off from the Airport. And for the total flights, it simply counts the Flight IDs for each of the Fly Date in the Flights Record.



Flights Per Day Report Page (Section-1) of AMS

In the second section, the report, containing the information of total Flights took place in the Airport, for each of the Fly Date in the Flights Record, is represented in form of Bar Chart.



Flights Per Day Report Page (Section-2) of AMS

**Queries:**

1. If the admin wants to view how much passengers entered the Airport, on each Fly Date in Flights Records.

SELECT F.FLYDATE AS FLYDATE, COUNT(P.PASSENGER\_ID) AS NUMBER\_OF\_PASSENGERS

FROM PASSENGERS P INNER JOIN FLIGHTS F

ON P.FLIGHT\_ID = F.FLIGHT\_ID

WHERE P.STATUS = 'In'

GROUP BY F.FLYDATE



1. If the admin wants to view how much passengers exited the Airport, on each Fly Date in Flights Records.

SELECT F.FLYDATE AS FLYDATE, COUNT(P.PASSENGER\_ID) AS NUMBER\_OF\_PASSENGERS

FROM PASSENGERS P INNER JOIN FLIGHTS F

ON P.FLIGHT\_ID = F.FLIGHT\_ID

WHERE P.STATUS = 'Out'

GROUP BY F.FLYDATE



1. If the admin wants to see how much flights landed and how much flights took off on each of the Fly Date in Flights Record.

SELECT F.FlyDate, COUNT(

(

SELECT Flight\_ID FROM Flights WHERE Flight\_ID = (SELECT Flight\_ID FROM

Flights WHERE Status = 'In' AND Flight\_ID=F.Flight\_ID)

)

) AS Landed, COUNT(

(

SELECT Flight\_ID FROM Flights WHERE Flight\_ID = (SELECT Flight\_ID FROM

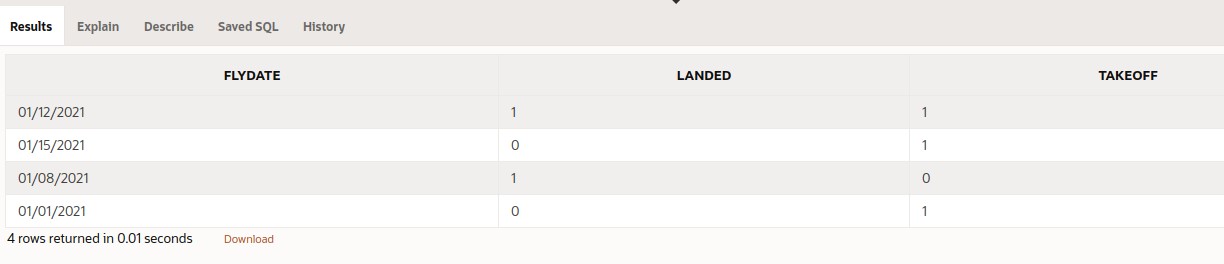
Flights WHERE Status = 'Out' AND Flight\_ID=F.Flight\_ID)

)

) AS takeoff

FROM Flights F

GROUP BY F.FlyDate

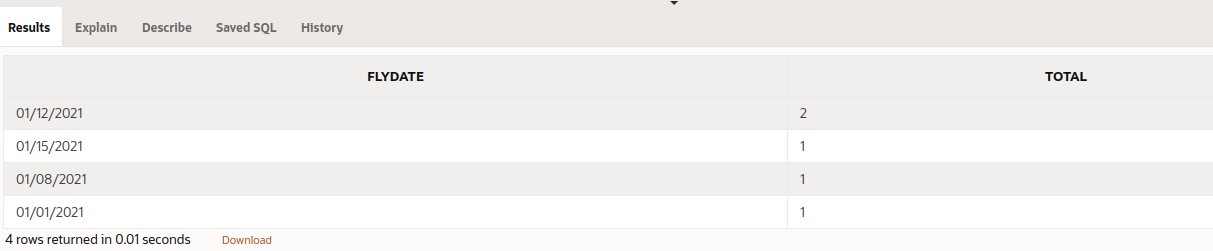


1. If the admin wants to view how much flights took place (either landed or took off) one each of the Fly Date on Flights Record.

SELECT F.FlyDate, COUNT(F.FlyDate) AS Total

FROM Flights F

GROUP BY F.FlyDate



1. If the admin wants to view the number of employees in each of the department in departments records.

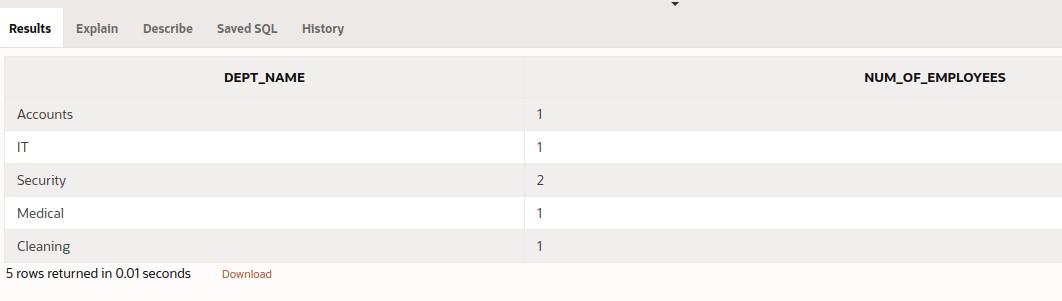
SELECT D.DEPT\_NAME, COUNT(E.EMP\_ID) AS NUM\_OF\_EMPLOYEES

FROM AIRPORT\_DEPT D, AIRPORT\_EMP E

WHERE D.DEPT\_ID = E.DEPT\_ID

GROUP BY D.DEPT\_ID, D.DEPT\_NAME

ORDER BY D.DEPT\_ID



1. If the admin wants to view how many passengers traveled from which flight, going from where to where, on which date.

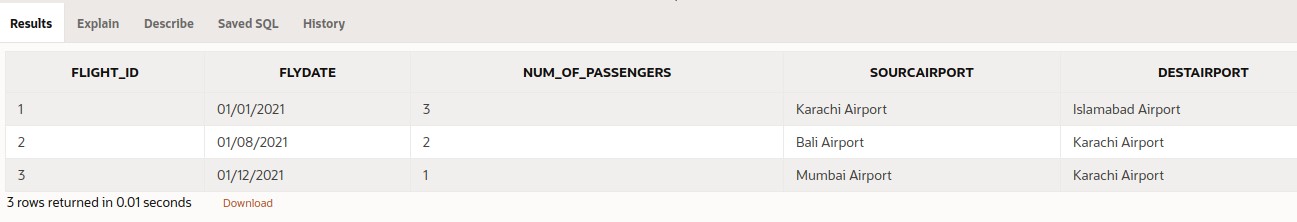
SELECT F.FLIGHT\_ID , F.FLYDATE, COUNT(P.PASSENGER\_ID) AS NUM\_OF\_PASSENGERS,

F.SOURCAIRPORT, F.DESTAIRPORT FROM FLIGHTS F, PASSENGERS P

WHERE F.FLIGHT\_ID = P.FLIGHT\_ID

GROUP BY F.FLIGHT\_ID,AIRLINE\_ID, F.FLYDATE, F.SOURCAIRPORT, F.DESTAIRPORT

ORDER BY F.FLIGHT\_ID



1. If the admin wants to view how many passengers traveled from which flight, of which airline, going from where to where, on which date.

SELECT F.FLIGHT\_ID ,AR.AIRLINE\_NAME, F.FLYDATE, COUNT(P.PASSENGER\_ID) AS

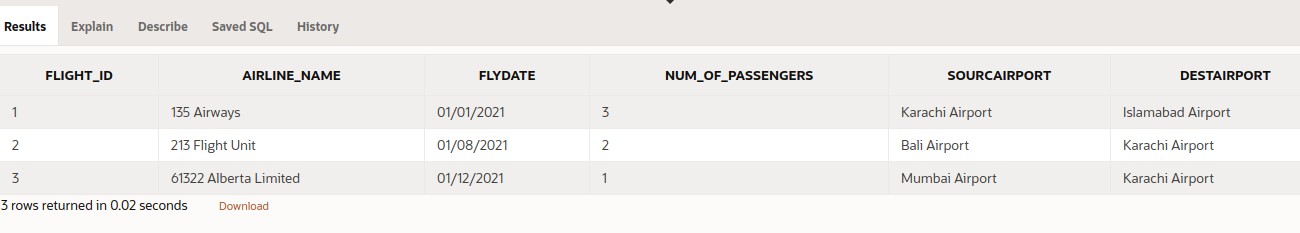
NUM\_OF\_PASSENGERS, F.SOURCAIRPORT, F.DESTAIRPORT

FROM FLIGHTS F, PASSENGERS P, AIRLINES AR

WHERE F.FLIGHT\_ID = P.FLIGHT\_ID AND F.AIRLINE\_ID=AR.AIRLINE\_ID

GROUP BY F.FLIGHT\_ID,F.AIRLINE\_ID, F.FLYDATE, F.SOURCAIRPORT,

F.DESTAIRPORT , AR.AIRLINE\_NAME ORDER BY F.FLIGHT\_ID



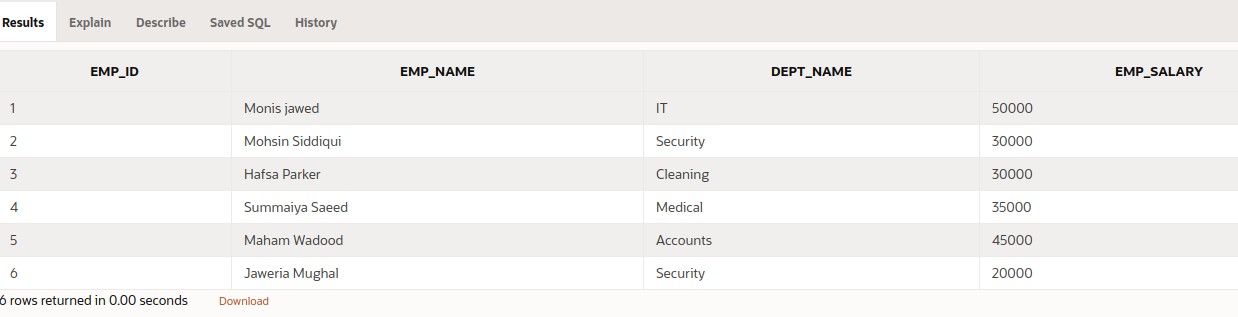
1. If admin wants to view name of each employee, along with its department name, and salary.

SELECT E.EMP\_ID, E.EMP\_NAME, D.DEPT\_NAME, E.EMP\_SALARY

FROM AIRPORT\_EMP E, AIRPORT\_DEPT D

WHERE E.DEPT\_ID=D.DEPT\_ID

ORDER BY E.EMP\_ID



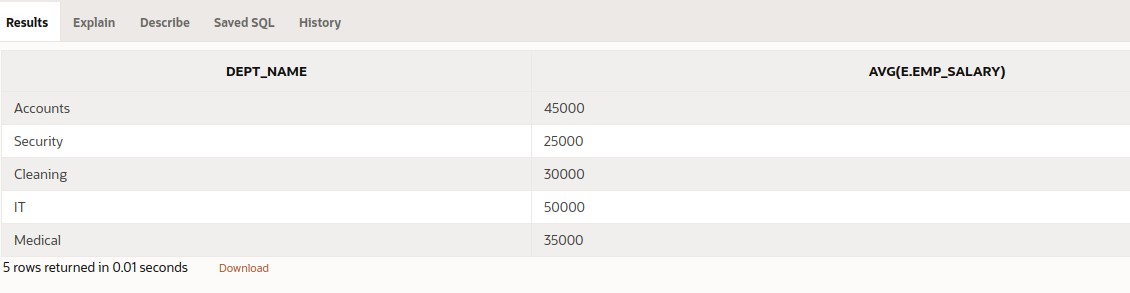
1. If admin wants to view average salary of employees working in each of the department.

SELECT D.DEPT\_NAME, AVG(E.EMP\_SALARY)

FROM AIRPORT\_DEPT D, AIRPORT\_EMP E

WHERE D.DEPT\_ID = E.DEPT\_ID

GROUP BY D.DEPT\_NAME



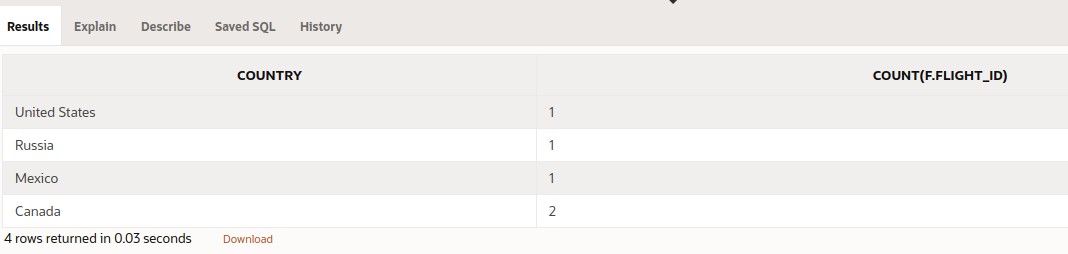
1. If the admin wants to view how many flights occurred of which country’s airlines.

SELECT AR.COUNTRY, COUNT(F.FLIGHT\_ID)

FROM AIRLINES AR, FLIGHTS F

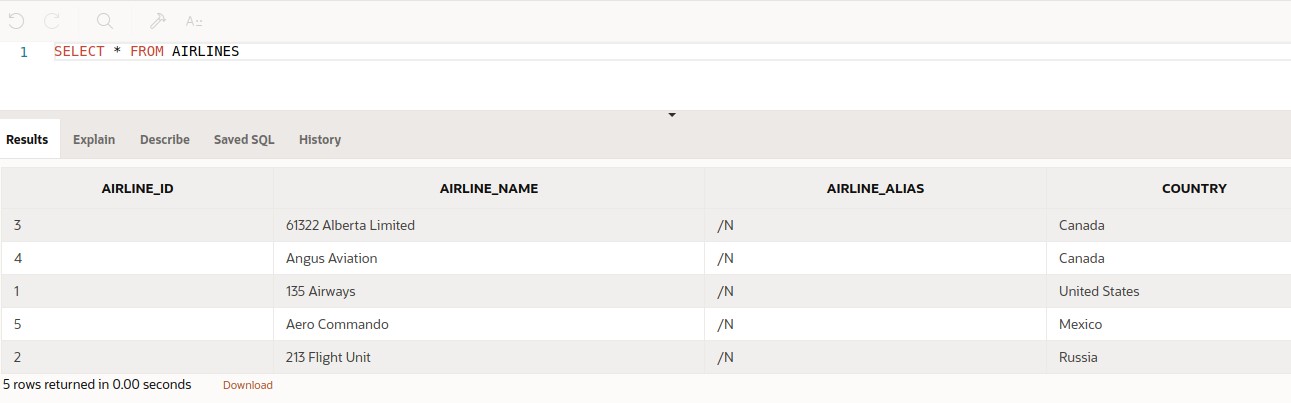
WHERE AR.AIRLINE\_ID=F.FLIGHT\_ID

GROUP BY AR.COUNTRY

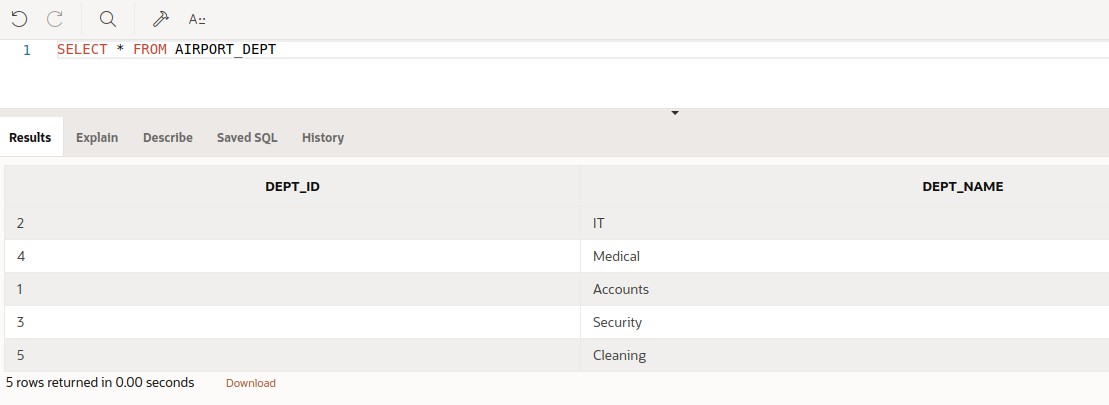


**Tables:**

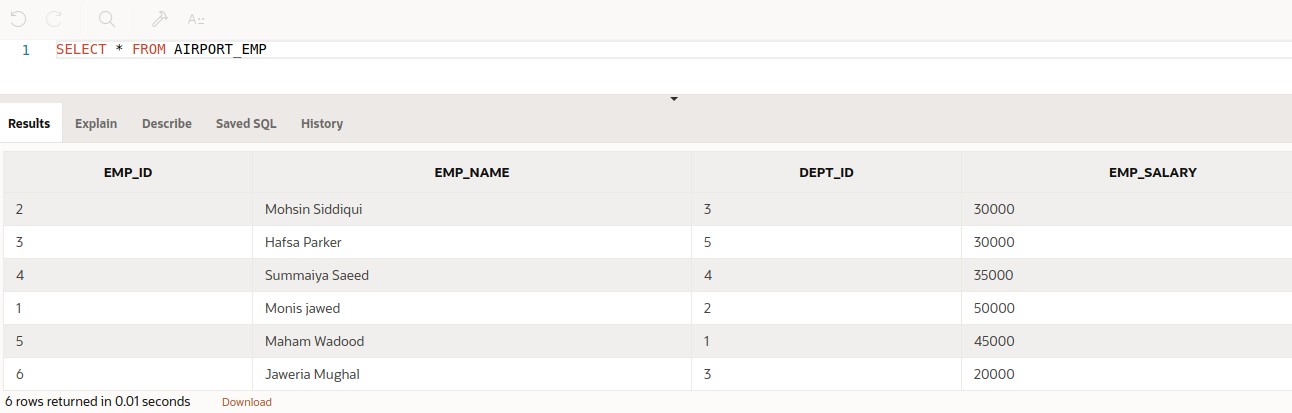
# 1 Airlines



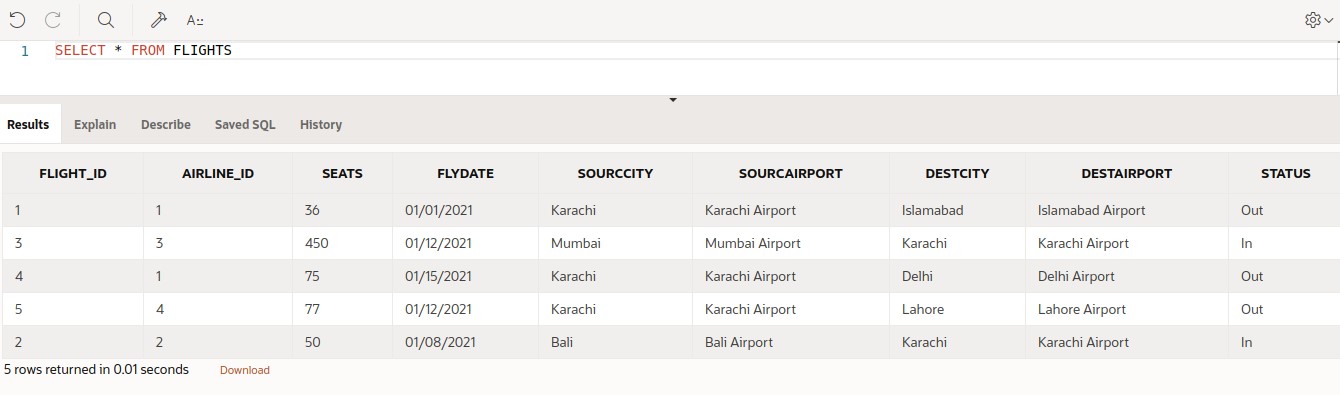
# 2 Departments



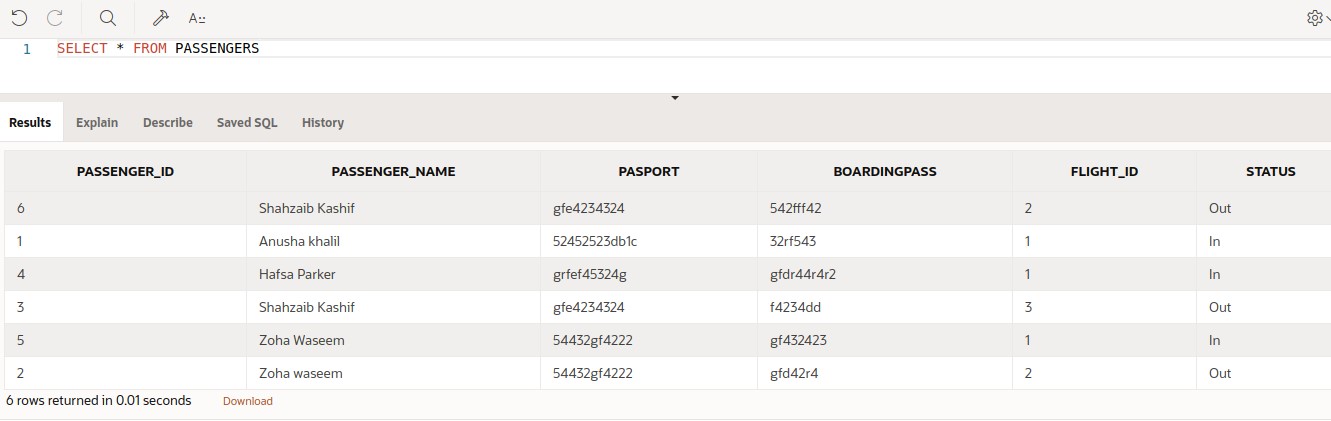
# 3 Employees



# 4 Flights

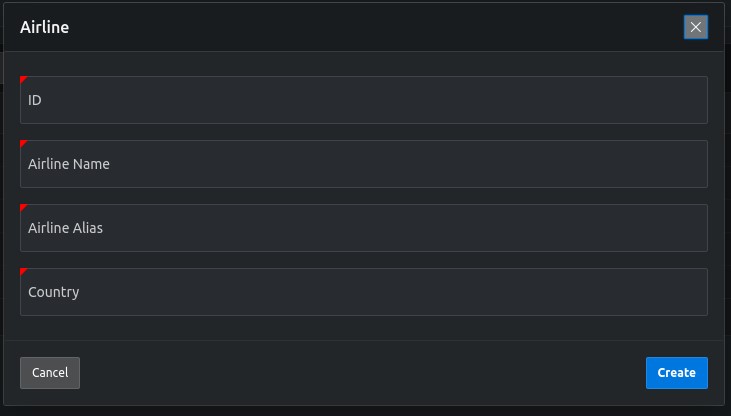


# 5 Passengers

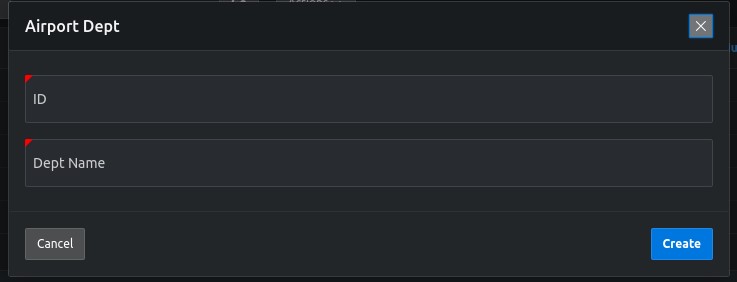


**Forms:**

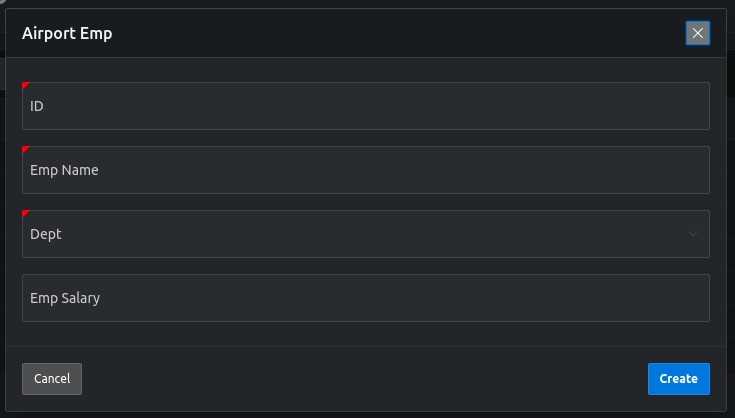
# 1 Airlines Record Insertion Form



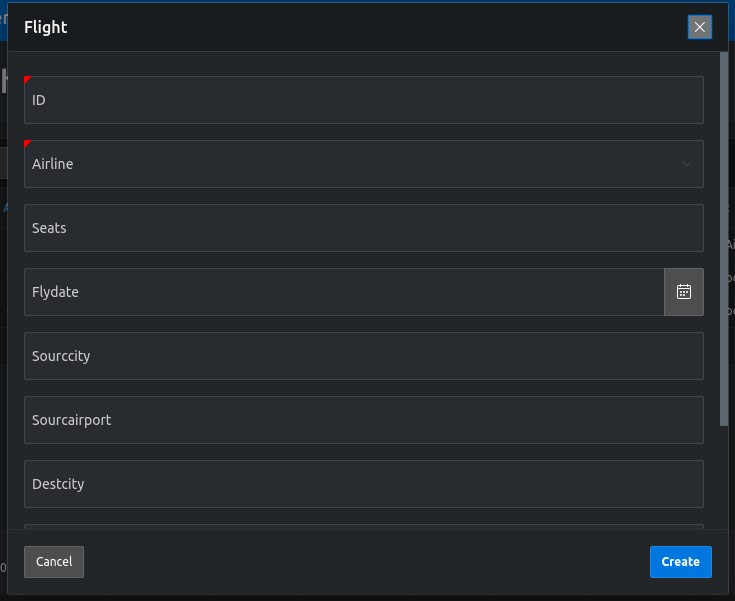
# 2 Departments Record Insertion Form



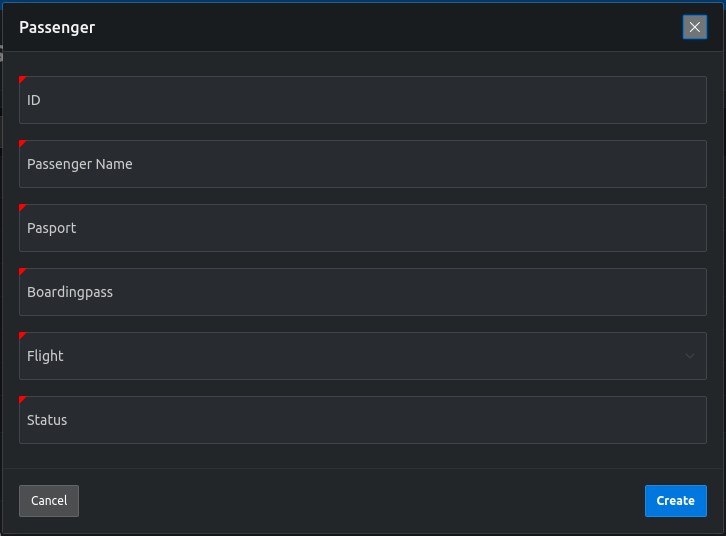
# 3 Employees Record Insertion Form



# 4 Flights Record Insertion Form



# 5 Passengers Record Insertion Form



**URL for the App:**

<https://apex.oracle.com/pls/apex/my-worspace/r/airport-app2/home?session=128523461562632>

Username: Shahzaibceo@gmail.com

Password: Airport23@$

**User Manual**

* First Login to the Airport Management System App by the User Credentials.
* From the Home Page Re-direct into any of the table’s page.
* There is a list icon on the top left of the screen which shows the list of all the pages of app.
* Use the create button to open up a form, to add up new entries/records in the table
* The pen tool on the extreme left of each entry is used to edit/update each entry individually.
* In the List of pages, under the home tab, are 2 separate pages for Viewing Reports.