

Airways Management System



Team-06

Samip Shah

1001709873

Anushka Nagpure

1001764114

Airways Management System

Abstract

Airways Management Systems is a computerized system used to store and retrieve information and conduct transactions related to air travel. This project is aimed at exposing the importance of Airways Management Systems.

The purpose of this application is to reduce the manual work and get all the details under one roof. It's simple GUI makes it available for all the users and target audience can be increased. The combination of passengers-flight-airport details makes it beneficial for everyone and not for certain audience.

Contents

Abstract	I
Description	1
Data to be captured and displayed.....	2
Queries (This are the output and operations user will be able to perform on dashboard after successful login. They can be modified later)	3
Assumed entities with some of the attributes and relations (It may be modified in next phase according to requirements)	4
EER DIAGRAM.....	5
RELATIONAL MODEL	6
RELATIONAL ALGEBRA of some queries (May or may not be used in demo)	7
10 SQL Queries with Screenshot in demo	9
Conclusion.....	13

Description

Airways Management System incorporate airline schedules, fare tariffs, passenger personal details & reservations, ticket records, employee details and airport information. This project is a package using which you can confirm passenger reservations, have a look at list of passengers and on list of flights with their schedule, airport information along with details of flight employees. Its user-friendly graphical interface helps users to interact with the system and meet their requirements in just few clicks.

The login page would be having option to register the user for first time and then from next time credentials will be matched and according dashboard will be created. The passenger will be able to view all the flights with details. Employee login will also be similar with list of flights and locations. Both passengers and employees will be able to view airport details and location. Both Passenger and employee may be able to filter and have a look of flights for day or for flight type. They may also be able to filter, or sort following data based on price or type of ticket they want. Payment details are also present in the system.

Data to be captured and displayed

- Passenger's personal details (Name, Number, Address, SSN etc...);
- Passengers flight details (From where and to, dates of travel, price at what ticket is booked, class of seat booked etc...);
- Flight details (Flight number, from where and when it travels, capacity of flight, no of flight attendees etc...);
- Flight attendee's personal details (Name, Number, Address, SSN etc...);
- Airport Details (Airport ID, Location, # of Runways, # of employee)

Queries (This are the output and operations user will be able to perform on dashboard after successful login. They can be modified later)

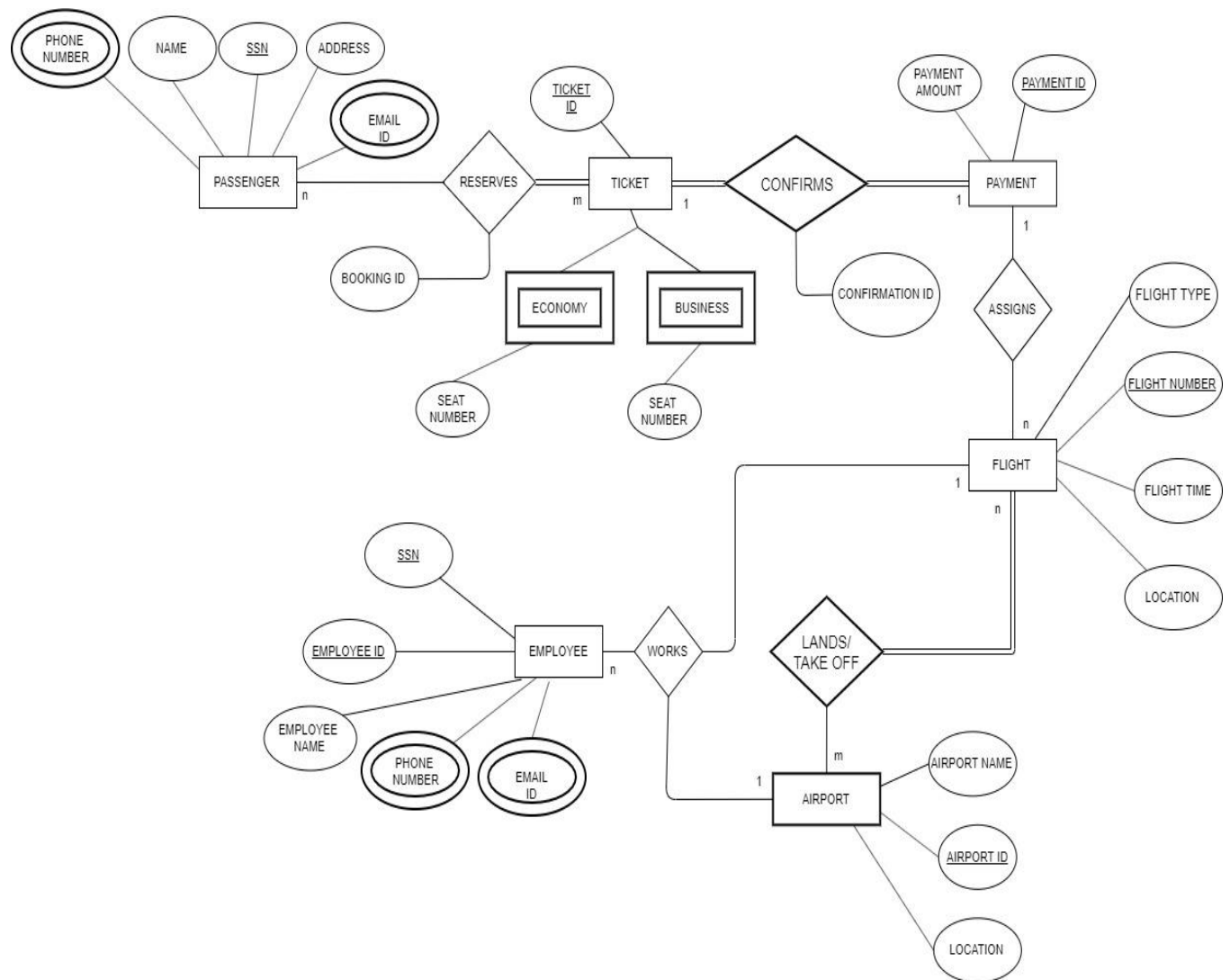
1. Display Complete Tables.
2. Display Ticket details with Payment details.
3. Display Airport details, Flight details and Schedule.
4. Display Employee details with Airport and flight schedules.
5. Display Passengers travel details.
6. Get total amount received from specific Passenger.
7. Get total Capacity of flight according to its type.
8. GUI to insert Data in table.
9. Delete Record From Table.
10. Filter Flight according to its type.

Assumed entities with some of the attributes and relations (It may be modified in next phase according to requirements)

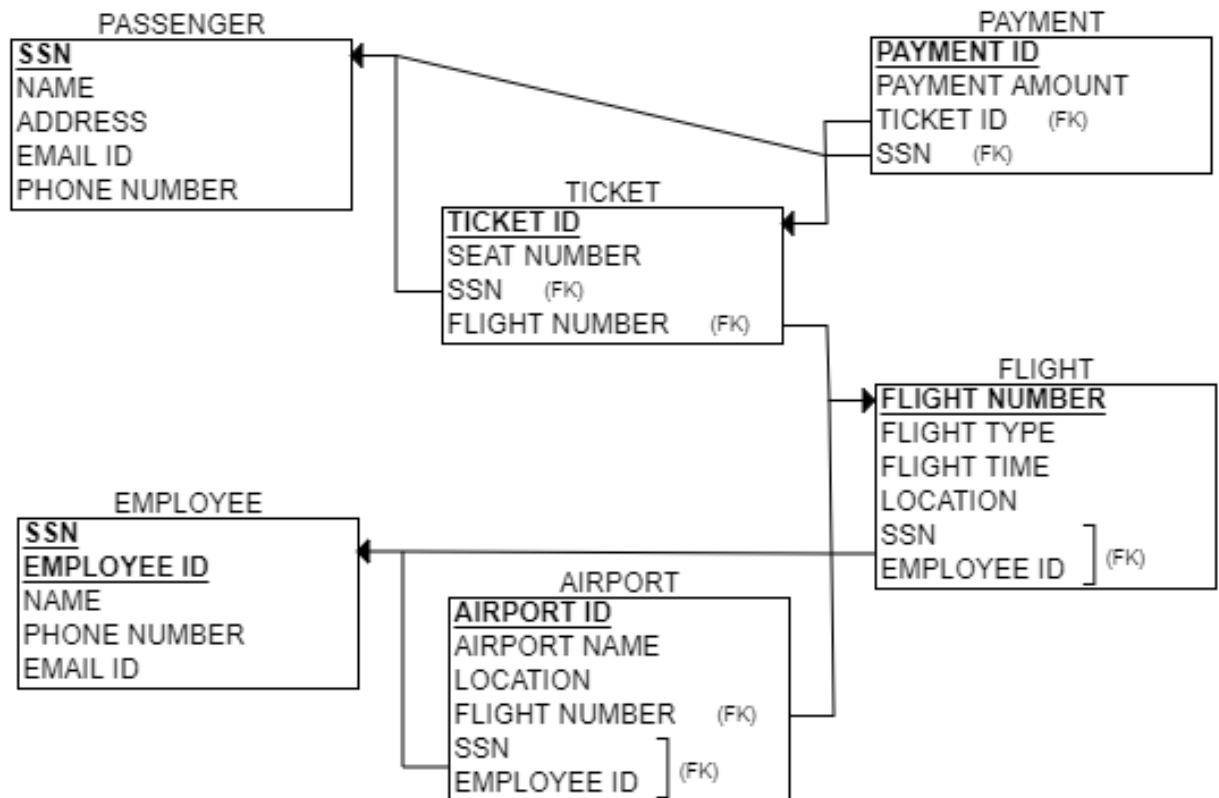
- Passenger personal (Name, SSN, Address, Phone number)
- Passenger booking (Booking ID, SSN, Flight number, Class, Price, Time)
- Flight attendee (Employee ID, Name, SSN, Phone number, Position, Flight number)
- Payment (Payment ID, Payment Amount,)
- Flight (Flight number, Flight time, Flight location)
- Airport (Airport ID, Location, # of runways)

Each SSN may be associated with specific booking ID and each booking ID may relate to specific flight number. There will be many to many relations between airport ID and Flight Number. Employee ID and flight number would have one to one relation. Employee ID may be associated with flight number.

EER DIAGRAM



RELATIONAL MODEL



RELATIONAL ALGEBRA of some queries (May or may not be used in demo)

1. Get Passengers Details;

$\sigma(\text{PASSENGERS})$

2. Get flight details of any particular passenger. (Assume get flight details of passenger with SSN: 123456789.)

$\Pi \text{ FLIGHT_ID, FLIGHT_TIME, FLIGHT_TYPE, LOCATION } (\sigma_{ssn="123456789"}(\text{FLIGHT}))$

3. Get Complete Flight Details; (All the flights)

$\sigma(\text{FLIGHTS})$

4. Sort Flight according to type (Get list of all Domestic Flights)

$\Pi \text{ FLIGHT_TYPE } (\sigma_{\text{FLIGHT_TYPE}="Domestic"}(\text{FLIGHT}))$

5. Filter flight according to its timings.

$\Pi \text{ FLIGHT_TIME } (\sigma_{\text{FLIGHT_TIME}="Morning"}(\text{FLIGHT}))$

6. Get flight attendant details

$\sigma(\text{EMPLOYEE})$

7. Filtering flight attendants according to their respective flights.

(i. Suppose we find all the employees for flight number 123)

(ii. Suppose we find all the employees for flight time: Morning)

$\text{GET_TEMP} \leftarrow \Pi_{\text{SSN}} (\sigma_{\text{FLIGHT_NUMBER}="123"}(\text{FLIGHT}))$

$\text{GET_ANSWER} \leftarrow \Pi_{(\text{EMPLOYEE ID, EMPLOYEE NAME, EMAIL ID})} (\sigma_{\text{SSN}=\text{GET-TEMP}(\text{EMPLOYEE}))}$

$\text{GET_TEMP} \leftarrow \Pi_{\text{SSN}} (\sigma_{\text{FLIGHT_TIME}="MORNING"}(\text{FLIGHT}))$

$\text{GET_ANSWER} \leftarrow \Pi_{(\text{EMPLOYEE ID, EMPLOYEE NAME, EMAIL ID})} (\sigma_{\text{SSN}=\text{GET-TEMP}(\text{EMPLOYEE}))}$

8. List of airports with its details

$\sigma(\text{AIRPORT})$

9. Filtering/ Sorting flight according to price range (Suppose we search for all the flight whose price is 1000)

$\text{GET_TEMP} \leftarrow \Pi_{\text{FLIGHT_NUMBER}} (\sigma_{\text{PAYMENT_AMOUNT}="10000"}(\text{PAYMENT}))$

$\text{GET_ANSWER} \leftarrow \Pi_{(\text{FLIGHT_NUMBER, FLIGHT_TYPE, LOCATION})} (\sigma_{\text{FLIGHT_NUMBER}=\text{GET-TEMP}(\text{FLIGHT}))}$

10. Filtering/ Sorting passengers according to their tickets (Economy/Business)

$GET_TEMP \leftarrow \Pi_{SSN} (\sigma_{(FLIGHT\ TYPE = "ECONOMY")}(FLIGHT))$

$GET_ANSWER \leftarrow \Pi_{(NAME,PHONE\ NUMBER,EMAIL\ ID,ADDRESS)} (\sigma_{(SSN = GET_TEMP(PASSENGER))})$

10 SQL Queries with Screenshot in demo

1. Display Tables.

```
"SELECT SSN,Name,EMAILID,PHONENUMBER,ADDRESS FROM PASSENGER";  
"SELECT TICKETID,SEATNUMBER,SSN,FLIGHTNUMBER FROM TICKET";  
"SELECT FLIGHTNUMBER,FLIGHTTYPE,CAPACITY FROM FLIGHT";  
"SELECT AIRPORTID,AIRPORTNAME,LOCATION FROM AIRPORT";  
"SELECT SSN,EMPLOYEEID,NAME,PHONENUMBER,EMAILID FROM EMPLOYEE";
```

DISPLAY TABLE

PASSENGER_DETAILS

TICKET_DETAILS

FLIGHT_DETAILS

AIRPORT_DETAILS

EMPLOYEE_DETAILS

SSN: 111111111 - Name: SAMIP - EMAILID: SAMIP@GMAIL.COM - PHONENUMBER: 1234567890 - ADDRESS: ARLINGTON
SSN: 222222222 - Name: NIRMIT - EMAILID: NIRMIT@GMAIL.COM - PHONENUMBER: 2345678901 - ADDRESS: ARLINGTON
SSN: 333333333 - Name: AASHU - EMAILID: AASHU@GMAIL.COM - PHONENUMBER: 3456789012 - ADDRESS: DALLAS
SSN: 444444444 - Name: ABHINN - EMAILID: ABHINN@GMAIL.COM - PHONENUMBER: 4567890123 - ADDRESS: AHMEDABAD
SSN: 555555555 - Name: AKSHIL - EMAILID: AKHIL@GMAIL.COM - PHONENUMBER: 5678901234 - ADDRESS: LONDON

2. /*GET TICKET ID WITH PAYMENT DETAILS*/

```
SELECT  
AIRWAYS.TICKET.TICKETID,SEATNUMBER,AIRWAYS.PAYMENT.PAYMENTID,P  
AYMENTAMOUNT FROM AIRWAYS.TICKET INNER JOIN AIRWAYS.PAYMENT ON  
AIRWAYS.TICKET.SSN= AIRWAYS.PAYMENT.SSN ORDER BY  
AIRWAYS.TICKET.TICKETID ASC;
```

GET TICKET ID WITH PAYMENT DETAILS

Search1

TICKETID: 1234567 - SEATNUMBER: 123 - PAYMENTID: 1112223330 - PAYMENTAMOUNT: 1000
TICKETID: 2345678 - SEATNUMBER: 456 - PAYMENTID: 4455332211 - PAYMENTAMOUNT: 2000
TICKETID: 3456789 - SEATNUMBER: 789 - PAYMENTID: 2233558844 - PAYMENTAMOUNT: 1500
TICKETID: 4567890 - SEATNUMBER: 963 - PAYMENTID: 6644402114 - PAYMENTAMOUNT: 3000
TICKETID: 5678901 - SEATNUMBER: 741 - PAYMENTID: 7788998877 - PAYMENTAMOUNT: 2500

3. /*GET AIRPORT DETAILS,FLIGHT DETAILS, FLIGHT SCHEDULE*/

```
SELECT  
AIRWAYS.AIRPORT.AIRPORTID,AIRPORTNAME,LOCATION,AIRWAYS.FLIGHT.F  
LIGHTNUMBER,FLIGHTTYPE,CAPACITY,AIRWAYS.FLIGHTSCHEDULE.ARRIVAL  
TIME,DEPARTURETIME,ARRIVALLOCATION,DEPARTURELOCATION  
FROM AIRWAYS.AIRPORT INNER JOIN (AIRWAYS.FLIGHT INNER JOIN  
AIRWAYS.FLIGHTSCHEDULE ON AIRWAYS.FLIGHT.FLIGHTNUMBER =  
AIRWAYS.FLIGHTSCHEDULE.FLIGHTNUMBER)  
ON (AIRWAYS.AIRPORT.FLIGHTNUMBER = AIRWAYS.FLIGHT.FLIGHTNUMBER)  
ORDER BY AIRWAYS.FLIGHT.FLIGHTNUMBER ASC;
```

GET AIRPORT DETAILS, FLIGHT DETAILS, FLIGHT SCHEDULE

Search2

AIRPORTID: 5 - AIRPORTNAME: LMN - LOCATION: INDIA - FLIGHTNUMBER: 0 - FLIGHTTYPE: SMALL - CAPACITY: 500 - ARRIVALTIME: 12:00:00 - DEPARTURETIME: 14:00:00 - ARRIVALLOCATION: USA - DEPARTURELOCATION: INDIA
AIRPORTID: 1 - AIRPORTNAME: ABC - LOCATION: INDIA - FLIGHTNUMBER: 0 - FLIGHTTYPE: SMALL - CAPACITY: 500 - ARRIVALTIME: 12:00:00 - DEPARTURETIME: 14:00:00 - ARRIVALLOCATION: USA - DEPARTURELOCATION: INDIA
AIRPORTID: 2 - AIRPORTNAME: XYZ - LOCATION: USA - FLIGHTNUMBER: 0 - FLIGHTTYPE: SMALL - CAPACITY: 500 - ARRIVALTIME: 12:00:00 - DEPARTURETIME: 14:00:00 - ARRIVALLOCATION: USA - DEPARTURELOCATION: INDIA
AIRPORTID: 3 - AIRPORTNAME: MNO - LOCATION: USA - FLIGHTNUMBER: 0 - FLIGHTTYPE: SMALL - CAPACITY: 500 - ARRIVALTIME: 12:00:00 - DEPARTURETIME: 14:00:00 - ARRIVALLOCATION: USA - DEPARTURELOCATION: INDIA
AIRPORTID: 4 - AIRPORTNAME: PQR - LOCATION: CANADA - FLIGHTNUMBER: 0 - FLIGHTTYPE: SMALL - CAPACITY: 500 - ARRIVALTIME: 12:00:00 - DEPARTURETIME: 14:00:00 - ARRIVALLOCATION: USA - DEPARTURELOCATION: INDIA

```
4. /*GET EMPLOYEE DETAILS WITH FLIGHT DETAILS AND AIRPORT DETAILS*/
SELECT
    AIRWAYS.EMPLOYEE.EMPLOYEEID,NAME,
    AIRWAYS.AIRPORT.AIRPORTID,AIRPORTNAME,LOCATION,AIRWAYS.FLIGHT.F
    LIGHTNUMBER
FROM AIRWAYS.AIRPORT INNER JOIN (AIRWAYS.FLIGHT INNER JOIN
AIRWAYS.EMPLOYEE ON AIRWAYS.FLIGHT.EMPLOYEEID =
AIRWAYS.EMPLOYEE.EMPLOYEEID)
ON (AIRWAYS.AIRPORT.FLIGHTNUMBER = AIRWAYS.FLIGHT.FLIGHTNUMBER)
ORDER BY AIRWAYS.EMPLOYEE.EMPLOYEEID ASC;
```

GET EMPLOYEE DETAILS WITH FLIGHT DETAILS AND AIRPORT DETAILS

Search3

EMPLOYEEID: 7418529 - NAME: ASHUTOSH - AIRPORTID: 5 - AIRPORTNAME: LMN - LOCATION: INDIA - FLIGHTNUMBER: 0
EMPLOYEEID: 7418529 - NAME: ASHUTOSH - AIRPORTID: 1 - AIRPORTNAME: ABC - LOCATION: INDIA - FLIGHTNUMBER: 0
EMPLOYEEID: 7418529 - NAME: ASHUTOSH - AIRPORTID: 2 - AIRPORTNAME: XYZ - LOCATION: USA - FLIGHTNUMBER: 0
EMPLOYEEID: 7418529 - NAME: ASHUTOSH - AIRPORTID: 3 - AIRPORTNAME: MNO - LOCATION: USA - FLIGHTNUMBER: 0
EMPLOYEEID: 7418529 - NAME: ASHUTOSH - AIRPORTID: 4 - AIRPORTNAME: PQR - LOCATION: CANADA - FLIGHTNUMBER: 0

```
5. /*GET PASSENGER DETAILS WITH TICKET DETAILS FLIGHT NUMBER
PAYMENT ID AND AIRPORT NAME AND LOCATION*/
SELECT
    AIRWAYS.PASSENGER.NAME,ADDRESS,AIRWAYS.FLIGHT.FLIGHTNUMBER,
    AIRWAYS.PAYMENT.PAYMENTID,AIRWAYS.AIRPORT.AIRPORTNAME,LOCATIO
    N
FROM AIRWAYS.AIRPORT
INNER JOIN (AIRWAYS.FLIGHT
INNER JOIN (AIRWAYS.PASSENGER
INNER JOIN (AIRWAYS.TICKET INNER JOIN AIRWAYS.PAYMENT ON
AIRWAYS.TICKET.TICKETID = AIRWAYS.PAYMENT.TICKETID)
ON AIRWAYS.PASSENGER.SSN = AIRWAYS.TICKET.SSN)
ON AIRWAYS.FLIGHT.FLIGHTNUMBER = AIRWAYS.TICKET.FLIGHTNUMBER)
ORDER BY AIRWAYS.FLIGHT.FLIGHTNUMBER DESC;
```

GET PASSENGER DETAILS WITH TICKET DETAILS FLIGHT NUMBER PAYMENT ID AND AIRPORT NAME AND LOCATION

Search4

NAME: AKSHIL - ADDRESS: LONDON - FLIGHTNUMBER: 4 - PAYMENTID: 7788998877 - AIRPORTNAME: MNO - LOCATION: USA
NAME: AKSHIL - ADDRESS: LONDON - FLIGHTNUMBER: 4 - PAYMENTID: 7788998877 - AIRPORTNAME: LMN - LOCATION: INDIA
NAME: AKSHIL - ADDRESS: LONDON - FLIGHTNUMBER: 4 - PAYMENTID: 7788998877 - AIRPORTNAME: XYZ - LOCATION: USA
NAME: AKSHIL - ADDRESS: LONDON - FLIGHTNUMBER: 4 - PAYMENTID: 7788998877 - AIRPORTNAME: PQR - LOCATION: CANADA
NAME: AKSHIL - ADDRESS: LONDON - FLIGHTNUMBER: 4 - PAYMENTID: 7788998877 - AIRPORTNAME: ABC - LOCATION: INDIA
NAME: ABHINN - ADDRESS: AHMEDABAD - FLIGHTNUMBER: 3 - PAYMENTID: 6644402114 - AIRPORTNAME: MNO - LOCATION: USA
NAME: ABHINN - ADDRESS: AHMEDABAD - FLIGHTNUMBER: 3 - PAYMENTID: 6644402114 - AIRPORTNAME: LMN - LOCATION: INDIA
NAME: ABHINN - ADDRESS: AHMEDABAD - FLIGHTNUMBER: 3 - PAYMENTID: 6644402114 - AIRPORTNAME: XYZ - LOCATION: USA
NAME: ABHINN - ADDRESS: AHMEDABAD - FLIGHTNUMBER: 3 - PAYMENTID: 6644402114 - AIRPORTNAME: PQR - LOCATION: CANADA
NAME: ABHINN - ADDRESS: AHMEDABAD - FLIGHTNUMBER: 3 - PAYMENTID: 6644402114 - AIRPORTNAME: ABC - LOCATION: INDIA
NAME: AASHU - ADDRESS: DALLAS - FLIGHTNUMBER: 2 - PAYMENTID: 2233558844 - AIRPORTNAME: LMN - LOCATION: INDIA
NAME: AASHU - ADDRESS: DALLAS - FLIGHTNUMBER: 2 - PAYMENTID: 2233558844 - AIRPORTNAME: XYZ - LOCATION: USA
NAME: AASHU - ADDRESS: DALLAS - FLIGHTNUMBER: 2 - PAYMENTID: 2233558844 - AIRPORTNAME: PQR - LOCATION: CANADA
NAME: AASHU - ADDRESS: DALLAS - FLIGHTNUMBER: 2 - PAYMENTID: 2233558844 - AIRPORTNAME: ABC - LOCATION: INDIA
NAME: AASHU - ADDRESS: DALLAS - FLIGHTNUMBER: 2 - PAYMENTID: 2233558844 - AIRPORTNAME: MNO - LOCATION: USA
NAME: NIRMIT - ADDRESS: ARLINGTON - FLIGHTNUMBER: 1 - PAYMENTID: 4455332211 - AIRPORTNAME: LMN - LOCATION: INDIA
NAME: NIRMIT - ADDRESS: ARLINGTON - FLIGHTNUMBER: 1 - PAYMENTID: 4455332211 - AIRPORTNAME: XYZ - LOCATION: USA
NAME: NIRMIT - ADDRESS: ARLINGTON - FLIGHTNUMBER: 1 - PAYMENTID: 4455332211 - AIRPORTNAME: PQR - LOCATION: CANADA
NAME: NIRMIT - ADDRESS: ARLINGTON - FLIGHTNUMBER: 1 - PAYMENTID: 4455332211 - AIRPORTNAME: ABC - LOCATION: INDIA
NAME: NIRMIT - ADDRESS: ARLINGTON - FLIGHTNUMBER: 1 - PAYMENTID: 4455332211 - AIRPORTNAME: MNO - LOCATION: USA
NAME: SAMP - ADDRESS: ARLINGTON - FLIGHTNUMBER: 0 - PAYMENTID: 1112223330 - AIRPORTNAME: XYZ - LOCATION: USA
NAME: SAMP - ADDRESS: ARLINGTON - FLIGHTNUMBER: 0 - PAYMENTID: 1112223330 - AIRPORTNAME: PQR - LOCATION: CANADA
NAME: SAMP - ADDRESS: ARLINGTON - FLIGHTNUMBER: 0 - PAYMENTID: 1112223330 - AIRPORTNAME: ABC - LOCATION: INDIA
NAME: SAMP - ADDRESS: ARLINGTON - FLIGHTNUMBER: 0 - PAYMENTID: 1112223330 - AIRPORTNAME: MNO - LOCATION: USA
NAME: SAMP - ADDRESS: ARLINGTON - FLIGHTNUMBER: 0 - PAYMENTID: 1112223330 - AIRPORTNAME: LMN - LOCATION: INDIA

6. /*GET TOTAL AMOUNT RECIEVED FROM SPECIFIC PASSENGER*/

SELECT SUM(PAYMENTAMOUNT), SSN

FROM PAYMENT

GROUP BY PAYMENTID;

GET TOTAL AMOUNT RECIEVED FROM SPECIFIC PASSENGER

Search5

PAYMENTAMOUNT: 1000 - SSN: 1111111111
PAYMENTAMOUNT: 1500 - SSN: 3333333333
PAYMENTAMOUNT: 2000 - SSN: 2222222222
PAYMENTAMOUNT: 3000 - SSN: 4444444444
PAYMENTAMOUNT: 2500 - SSN: 5555555555

7. /*GET TOTAL CAPACITY OF FLIGHT ACCORDING TO ITS TYPE*/

SELECT SUM(CAPACITY),FLIGHTNUMBER, FLIGHTTYPE

FROM FLIGHT

GROUP BY FLIGHTTYPE;

GET TOTAL CAPACITY OF FLIGHT ACCORDING TO ITS TYPE

Search6

CAPACITY: 1800 - FLIGHTNUMBER: 2 - FLIGHTTYPE: BIG
CAPACITY: 700 - FLIGHTNUMBER: 1 - FLIGHTTYPE: MEDIUM
CAPACITY: 1000 - FLIGHTNUMBER: 0 - FLIGHTTYPE: SMALL

8. /* INSERT INTO PASSENGER TABLE*/

"INSERT INTO AIRWAYS.PASSENGER (SSN, NAME, EMAILID, PHONENUMBER, ADDRESS)

VALUES

```
("'.$_POST["SSN"]."',".$_POST["NAME"]."',".$_POST["EMAILID"]."',".$_POST["PHONENUMBER"]."',".$_POST["ADDRESS"].'");
```

INSERT RECORD IN PASSENGER

SSN:

NAME:

EMAILID:

PHONENUMBER:

ADDRESS:

INSERT

9. /* DELETE ENTRY*/

```
"DELETE FROM AIRWAYS.PASSENGER WHERE SSN='9898989898'";
```

DELETE RECORD IN PASSENGER

DELETE

10. /* SEARCH FLIGHT ACCORDING TO ITS TYPE*/

```
"SELECT FLIGHTNUMBER,FLIGHTTYPE,CAPACITY FROM FLIGHT WHERE FLIGHTTYPE='SMALL'";
```

```
"SELECT FLIGHTNUMBER,FLIGHTTYPE,CAPACITY FROM FLIGHT WHERE FLIGHTTYPE='MEDIUM'";
```

```
"SELECT FLIGHTNUMBER,FLIGHTTYPE,CAPACITY FROM FLIGHT WHERE FLIGHTTYPE='BIG'";
```

SEARCH FLIGHT ACCORDING TO ITS TYPE

SMALL ▼

SEARCH

FLIGHTNUMBER: 0 - FLIGHTTYPE: SMALL - CAPACITY: 500

FLIGHTNUMBER: 3 - FLIGHTTYPE: SMALL - CAPACITY: 500

Conclusion

From this project I would be able to learn about entities and what attributes would be possible for each of them. How there would be relation between each of them and for what attribute what could be the primary key and how they can be related to each other.

EER diagram gives the flow of the project and gives details about which entity has what attributes. It also gives details about which entity has what primary key. Relational Model (Relational Schema) gives details about what key is connected where and how the data will be retrieved when search query will be fired. Phase 2 also has relational queries which gives details about how SQL queries will work and what details from what table will be retrieved.

Phase 3 gives more highlight on DDL and DML which gives better grip on SQL queries and flow of project. DDL is short name of Data Definition Language, which deals with database schemas and descriptions, of how the data should reside in the database. DML is short name of Data Manipulation Language which deals with data manipulation, and includes most common SQL statements such SELECT, INSERT, UPDATE, DELETE etc, and it is used to store, modify, retrieve, delete and update data in database.

Phase 4, where actual implementation takes place not only highlighted on SQL queries but also helped in PHP development. It included providing a GUI to run the above-mentioned queries and have a related output.

For better output of the project, more entities need to be added and more complex relations need to be formed. GUI can also be made more attractive and user friendly with more complex queries.