REPORT

AIRLINE RESERVATION SYSTEM

SUBMITTED BY :

MARYAM BIBI(FA21-BCS-047)

ISHA EMAN (FA21-BCS-046)

SUPERVISED BY:

MR. WASEEM KHAN

6/JUNE/2023

Contents

Chapter 1

Introduction

Introduction of Project

The domain is airline industry .The motivation for domain analysis is to develop a new software we want to develop a software " airline reservation system ".The **purpose** is:

• To provide an easy and efficient way for customers to search and book flights online.

• To provide airlines with a streamlined system to manage flight schedules, seat availability, and customer reservations.

1.2 Project Background:

The domain is airline reservation. The reason for the problem is that traditional methods of flight booking can be time-consuming and inconvenient for customers. The motivation for domain analysis is to develop a new software solution that simplifies the flight booking process. The purpose is to provide customers with a user-friendly platform to search for flights, view available seats, and make reservations online. Additionally, the software aims to assist airlines in managing their flight operations more effectively.

1.2.1 Stakeholders:

➢ Customers: Individuals who wish to book flights for personal or business travel.

➢ Airlines: Airline companies that offer flight services.

➢ System Administrators: Responsible for maintaining and managing the airline reservation system.

Scope And Vision of Project:

1.3.1 Vision Statement:

The purpose is:

• To provide customers with a user-friendly online platform to search for flights, view available seats, and make reservations conveniently.

• To provide airlines with an efficient system to manage flight schedules, seat availability, and customer reservations, leading to improved customer satisfaction and operational effectiveness.

The project is expected to allow customers to easily search for flights, compare prices, select seats, make payments online, and receive confirmation of their reservations.

1.4 Features of System:

Following are the features provided by our project:

➢ **Customer Features:**

- Search for flights based on criteria such as *date, destination, and preferred airline.*
- View available flights, along with details of departure and arrival times, available seats in the flight etc.
- Select preferred seats.
- Receive e-tickets and booking confirmation via email.
- Cancel or modify reservations within a specified timeframe.

➢ **Airline Features:**

- Manage flight schedules, including adding, modifying, and canceling flights.
- Manage seat availability and assign seats to passengers.
- Process customer reservations and payments.
- Generate reports on flight occupancy, revenue, and other relevant statistics.
- Handle cancellations and refunds.

CHAPTER 2:

Requirement Specification

2.1 Functional Requirements:

2.1.1 Customer Requirements:

➢ **User Registration:**

Customers should be able to create an account by providing their personal information.
The registration process should include email verification or account activation.

➢ **Flight Search and Booking:**

Customers should be able to search for flights based on criteria such as departure date, destination, and preferred airline.
The system should display a list of available flights with relevant details.
Customers should be able to select seats, specify any special requirements, and make payments securely.

➢ **Reservation Management:**

Customers should be able to view their reservations, including flight details, seat assignments, and payment status.
Customers should have the option to modify or cancel their reservations within a specified timeframe.

2.1.2 Airline Requirements:

➢ **Flight Management:**

Airlines should be able to add, modify, or cancel flights.
The system should allow airlines to specify flight schedules, seat availability, and ticket prices.

➢ **Reservation Management:**

Airlines should be able to view and manage customer reservations, including seat assignments and payment status.
The system should provide functionality for processing cancellations and refunds.

2.2 Non-Functional Requirements:

➢ **Security:**

- ✓ The system should ensure the confidentiality and integrity of customer data.
- ✓ Secure protocols should be used for user authentication and data transmission.

➢ **Performance:**

- ✓ The system should be able to handle a large number of concurrent users without significant delays or performance issues.
- ✓ Search and booking processes should be fast and efficient.

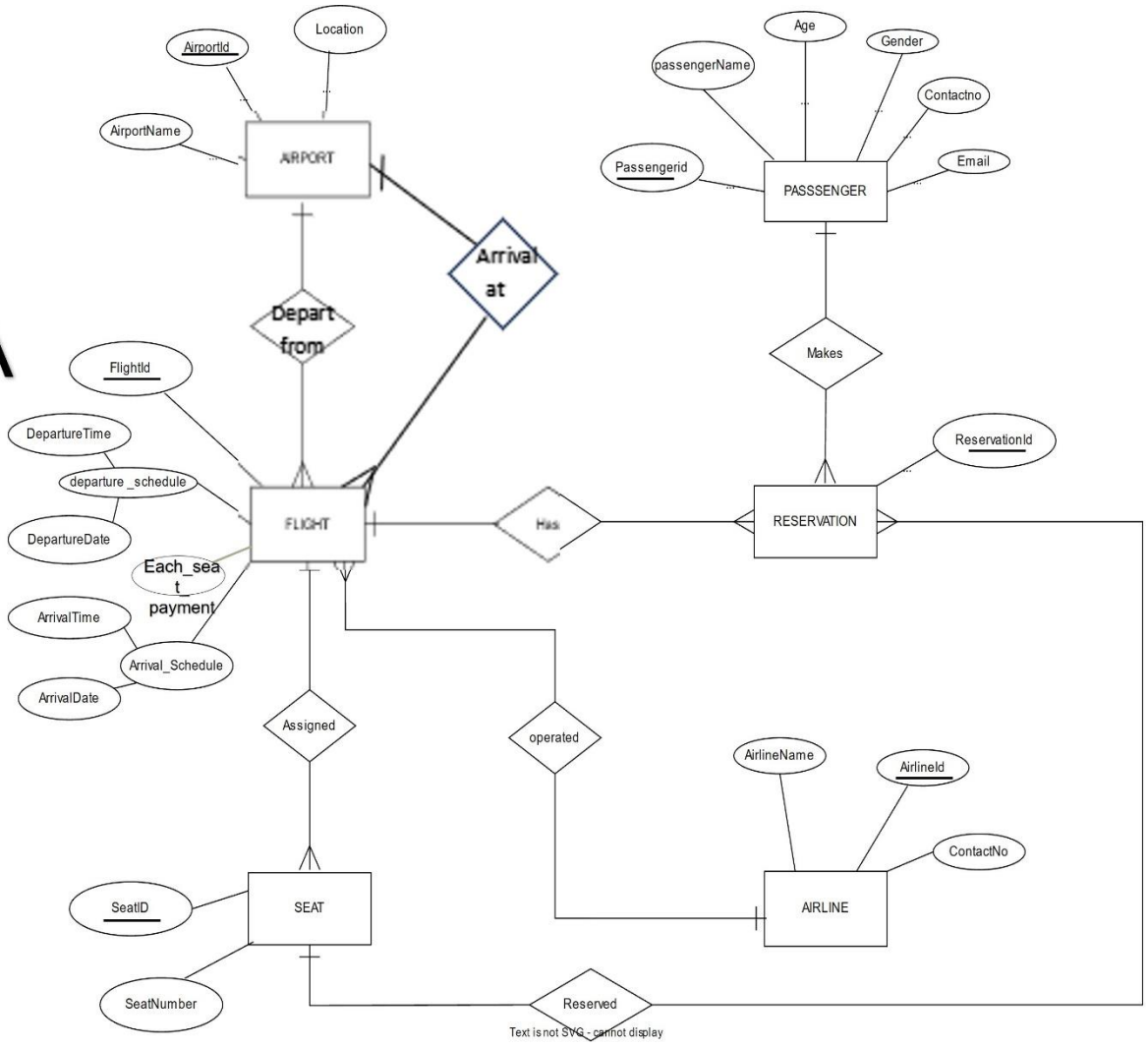
➢ **User-Friendly Interface:**

- ✓ The system should have an intuitive and user-friendly interface for customers to search and book flights easily.
- ✓ Clear instructions and error messages should be provided to guide users throughout the process.

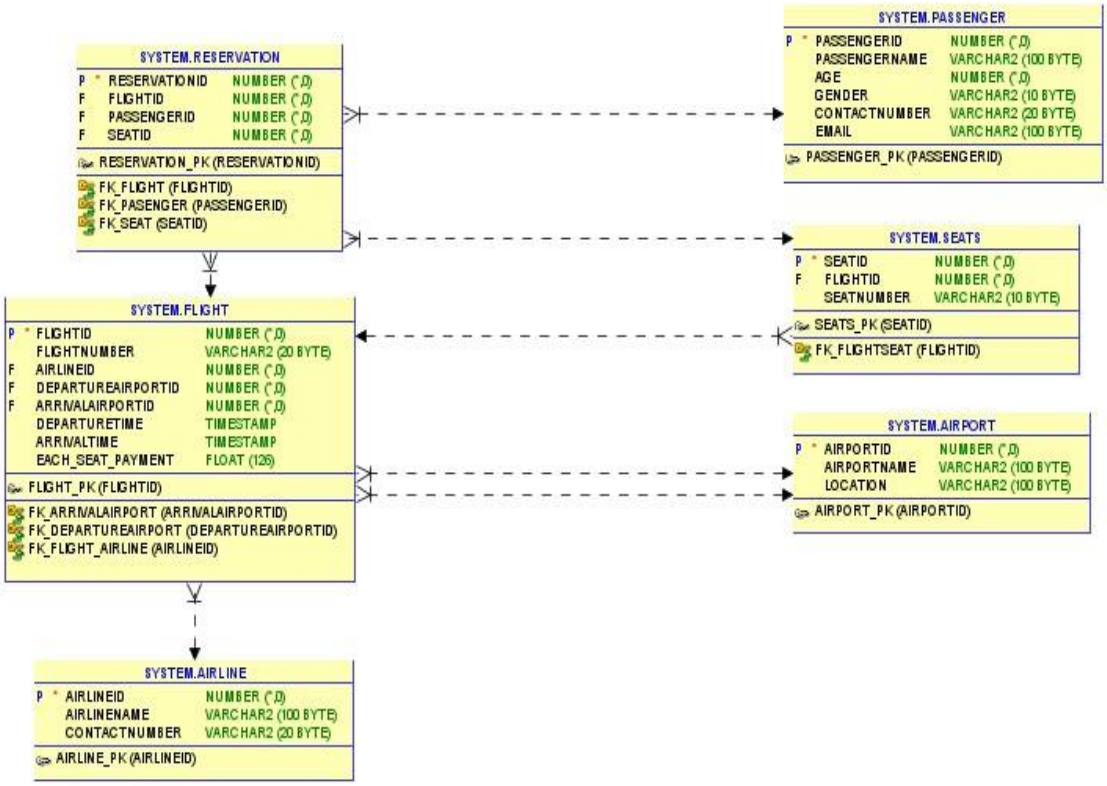
CHAPTER 3:

ERD DIAGRAM:

ERD DIAGRAM



CLASS DIAGRAM:



CHAPTER 4

TABLES :

PASSENGER TABLE:

```
CREATE TABLE passenger (
   passengerid   INT PRIMARY KEY,
   passengername VARCHAR(100),
   age           INT,
   gender        VARCHAR(10),
   contactnumber VARCHAR(20),
   email         VARCHAR(100)
);
```

Insert Passengers in Data Base:

```
INSERT INTO Passenger  VALUES  (1, 'Ali Khan', 30, 'Male', '1234567890', 'alikhan@gmail.com');
```

INSERT INTO Passenger VALUES (2, 'Sara Ahmed', 25, 'Female', '9876543210', 'saraahmed@gmail.com');

INSERT INTO Passenger VALUES (3, 'Ahmed Malik', 40, 'Male', '5555555555', 'ahmedmalik@gmail.com');

INSERT INTO Passenger VALUES (4, 'Aisha Khan', 28, 'Female', '1111111111', 'aishakhan@gmail.com');

INSERT INTO Passenger VALUES (5, 'Muhammad Hassan', 32, 'Male', '9999999999', 'hassanmuhammad@gmail.com');

INSERT INTO Passenger VALUES (6, 'Fatima Ali', 35, 'Female', '7777777777', 'fatimaali@gmail.com');

INSERT INTO Passenger VALUES (7, 'Abdullah Khan', 27, 'Male', '4444444444', 'abdullahkhan@gmail.com');

INSERT INTO Passenger VALUES (8, 'Zainab Ahmed', 29, 'Female', '2222222222', 'zainabahmed@gmail.com');

INSERT INTO Passenger VALUES (9, 'Usman Ali', 33, 'Male', '6666666666', 'usmanali@gmail.com');

INSERT INTO Passenger VALUES (10, 'Maryam Malik', 26, 'Female', '8888888888', 'maryammalik@gmail.com');

SELECT ALL QUERY:

select * from passenger;

```
select * from passenger;
```

Query Result

SQL | All Rows Fetched: 10 in 0.114 seconds

	PASSENGERID	PASSENGERNAME	AGE	GENDER	CONTACTNUMBER	EMAIL
1	1	Ali Khan	30	Male	1234567890	alikhan@gmail.com
2	2	Sara Ahmed	25	Female	9876543210	sonia@gmail.com
3	3	Ahmed Malik	40	Male	5555555555	ahmedmalik@gmail.com
4	4	Aisha Khan	28	Female	1111111111	aishakhan@gmail.com
5	5	Muhammad Hassan	32	Male	9999999999	hassanmuhammad@gmail.com
6	6	Fatima Ali	35	Female	7777777777	fatimaali@gmail.com
7	7	Abdullah Khan	27	Male	4444444444	abdullahkhan@gmail.com
8	8	Zainab Ahmed	29	Female	2222222222	zainabahmed@gmail.com
9	9	Usman Ali	33	Male	6666666666	usmanali@gmail.com
10	10	Maryam Malik	26	Female	8888888888	maryammalik@gmail.com

UPDATE QUERY:

update Passenger set email='Mona@gmail.com' where passengerId=2;

```
update  Passenger set email='Mona@gmail.com' where passengerId=2;
select * from passenger;
```

cript Output | Query Result

SQL | All Rows Fetched: 10 in 0.004 seconds

	PASSENGERID	PASSENGERNAME	AGE	GENDER	CONTACTNUMBER	EMAIL
1	1	Ali Khan	30	Male	1234567890	alikhan@gmail.com
2	2	Sara Ahmed	25	Female	9876543210	Mona@gmail.com
3	3	Ahmed Malik	40	Male	5555555555	ahmedmalik@gmail.com
4	4	Aisha Khan	28	Female	1111111111	aishakhan@gmail.com
5	5	Muhammad Hassan	32	Male	9999999999	hassanmuhammad@gmail.com
6	6	Fatima Ali	35	Female	7777777777	fatimaali@gmail.com
7	7	Abdullah Khan	27	Male	4444444444	abdullahkhan@gmail.com
8	8	Zainab Ahmed	29	Female	2222222222	zainabahmed@gmail.com
9	9	Usman Ali	33	Male	6666666666	usmanali@gmail.com
10	10	Maryam Malik	26	Female	8888888888	maryammalik@gmail.com

Search Specific Query:

select * from passenger where passengerid=4;

```
select  * from passenger where passengerid=4;
```

cript Output | Query Result

SQL | All Rows Fetched: 1 in 0.016 seconds

	PASSENGERID	PASSENGERNAME	AGE	GENDER	CONTACTNUMBER	EMAIL
1	4	Aisha Khan	28	Female	1111111111	aishakhan@gmail.com

CREATE TABLE :

```
CREATE TABLE airport
 (airportID int PRIMARY KEY,
   airportName varchar(100),
  city varchar(100));
```

Insertion Query:

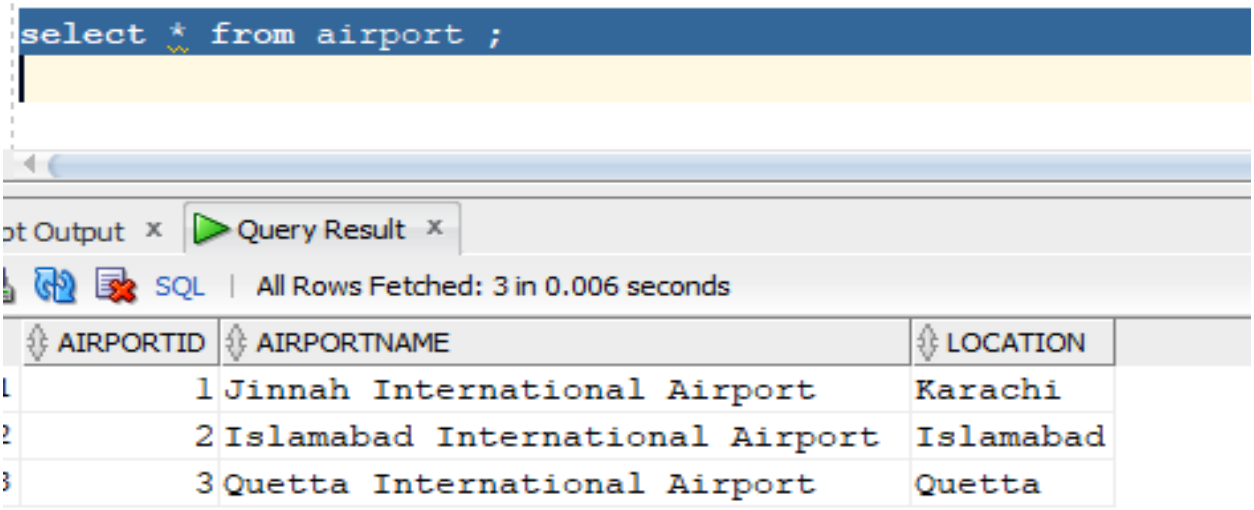
INSERT INTO airport VALUES(1, 'Jinnah International Airport', 'Karachi');I

NSERT INTO airport VALUES(2, 'Islamabad International Airport ', 'Islamabad');

INSERT INTO airport VALUES(3, 'Quetta International Airport', 'Quetta');

Select All Query:

select * from airport;



AIRPORTID	AIRPORTNAME	LOCATION
1	Jinnah International Airport	Karachi
2	Islamabad International Airport	Islamabad
3	Quetta International Airport	Quetta

Create Table Query:

CREATE TABLE airline (

airlineId int PRIMARY KEY,

name varchar(100),

contactNumber varchar(20));

Insertion Query:

INSERT INTO airline VALUES(101, 'Air Blue Limited','1234567890');

INSERT INTO airline VALUES(102, 'Pakistan International Airlines', '9876543210');

INSERT INTO airline VALUES(103, 'Serene Air', '5555555555');

```
select * from airline;
```

ipt Output | Query Result

SQL | All Rows Fetched: 3 in 0.002 seconds

	AIRLINEID	AIRLINENAME	CONTACTNUMBER
1	101	Air Blue Limited	1234567890
2	102	Pakistan International Airlines	9876543210
3	103	Serene Air	5555555555

Create Table Query:

```
CREATE TABLE seats (
seatID int PRIMARY KEY,
  flightid  int,
  seatNumber varchar(10),
   CONSTRAINT fk_flightSeatss_1 FOREIGN KEY (flightid) REFERENCES flights(flightID));
```

Insertion Query:

```
INSERT INTO seats VALUES (111, 001, 'A1');
INSERT INTO seats VALUES (112, 001, 'B1');I
NSERT INTO seats VALUES (113, 001, 'A2');
INSERT INTO seats VALUES (114, 001, 'B2');
INSERT INTO seats VALUES (115, 001, 'A3');
INSERT INTO seats VALUES (116, 001, 'B3');
NSERT INTO seats VALUES (117, 002, 'A1');
NSERT INTO seats VALUES (118, 002, 'B1');
NSERT INTO seats VALUES (119, 002, 'A2');
NSERT INTO seats VALUES (120, 002, 'B2');
```

```
select * from seats;
```

ript Output × | Query Result ×

SQL | All Rows Fetched: 18 in 0.0

	SEATID	FLIGHTID	SEATNUMBER
1	111	1	A1
2	112	1	B1
3	113	1	A2
4	114	1	B2
5	115	1	A3
6	116	1	B3
7	117	2	A1
8	118	2	B1
9	119	2	A2
10	120	2	B2
11	121	2	A3
12	122	2	B3
13	123	3	A1
14	124	3	B1
15	125	3	A2

Create Table :

```
CREATE TABLE flights (
Flightid int PRIMARY KEY,
  flightnumber varchar(20),
    Airlineid int,
    Departureairportid int,
   arrivalairportid int,
 departuretime timestamp,
  arrivaltime timestamp,
Each_seat_payment float,
   CONSTRAINT fk_departureairport_1 FOREIGN KEY (departureairportid) REFERENCES
airports(airportid),
  CONSTRAINT fk_arrivalairport_1 FOREIGN KEY (arrivalairportid) REFERENCES airports(airportid),
   CONSTRAINT fk_flight_airline_1 FOREIGN KEY (airlineid) REFERENCES airrline(airlineid));
```

Insertion Query:

INSERT INTO Flights VALUES(001,'FL001', 101, 1, 2, TIMESTAMP '2023-06-01 09:00:00', TIMESTAMP '2023-06-01 11:00:00');I

NSERT INTO Flights VALUES (002,'FL002', 102, 2, 3, TIMESTAMP '2023-06-01 12:00:00', TIMESTAMP '2023-06-01 14:00:00');I

NSERT INTO FlightsVALUES (003,'FL003', 103, 3, 1, TIMESTAMP '2023-06-02 10:00:00', TIMESTAMP '2023-06-02 12:00:00');

```
select * from flight ;
```

ipt Output | Query Result

SQL | All Rows Fetched: 3 in 0.007 seconds

	FLIGHTID	FLIGHTNUMBER	AIRLINEID	DEPARTUREAIRPORTID	ARRIVALAIRPORTID	DEPARTURETIME	ARRIVALTIME
1	1	FL001	101	1	2	01-JUN-23 09.00.00.000000000 AM	01-JUN-23 11.00.00.000000000 AM
2	2	FL002	102	2	3	01-JUN-23 12.00.00.000000000 PM	01-JUN-23 02.00.00.000000000 PM
3	3	FL003	103	3	1	02-JUN-23 10.00.00.000000000 AM	02-JUN-23 12.00.00.000000000 PM

```
select flightid , flightnumber , to_Char(departuretime,'DD-MM-YY HH.MI.SS') departurture_schedule,
to_Char(arrivaltime,'DD-MM-YY HH.MI.SS')  arrival_schedule
from flight ;
```

Script Output | Query Result

SQL | All Rows Fetched: 3 in 0.005 seconds

	FLIGHTID	FLIGHTNUMBER	DEPARTURTURE_SCHEDULE	ARRIVAL_SCHEDULE
1	1	FL001	01-06-23 09.00.00	01-06-23 11.00.00
2	2	FL002	01-06-23 12.00.00	01-06-23 02.00.00
3	3	FL003	02-06-23 10.00.00	02-06-23 12.00.00

Create Table :

```
CREATE TABLE reservation (
   reservationid INT PRIMARY KEY,
   flightid     INT,
   passengerid   INT,
   seatid       INT,
   CONSTRAINT fk_flight FOREIGN KEY ( flightid )
      REFERENCES flight ( flightid ),
   CONSTRAINT fk_pasenger FOREIGN KEY ( passengerid )
      REFERENCES passenger ( passengerid ),
```

```
    CONSTRAINT fk_seat FOREIGN KEY ( seatid )
        REFERENCES seats ( seatid )
);
```

Insertion Query:

```
insert into reservation values(211,001,1,111);
insert into reservation values(212,001,1,112);
insert into reservation values(213,002,3,118);
insert into reservation values(214,003,7,125);
insert into  reservation values(215,001,4,115);
```

```
select * from reservation;
```

cript Output | Query Result

SQL | All Rows Fetched: 5 in 0.01 seconds

	RESERVATIONID	FLIGHTID	PASSENGERID	SEATID
1	211	1	1	111
2	212	1	1	112
3	213	2	3	118
4	214	3	7	125
5	215	1	4	115

CHAPTER 5:

QUERIES

DISTINCT:

```
--->>>>>>>>>>>>>>------- DISTINCT

SELECT DISTINCT  flightid , passengerid FROM reservation;
```

Query Result

SQL | All Rows Fetched: 4 in 0.442 seconds

	FLIGHTID	PASSENGERID
1	2	3
2	3	7
3	2	4
4	1	1

LIKE:

```
--->>>>>>>>>>>>>>------- LIKE

SELECT passengername , email FROM passenger WHERE passengername LIKE '%Ali';
```

Script Output | Query Result

SQL | All Rows Fetched: 2 in 0.011 seconds

	PASSENGERNAME	EMAIL
1	Fatima Ali	fatimaali@gmail.com
2	Usman Ali	usmanali@gmail.com

CONCAT:

```
--->>>>>>>>>>>>>>------- CONCATENATION

SELECT passengername  || '   ' ||  email || '  ' ||   gender AS passengerdetails FROM passenger;
SELECT DISTINCT flightid FROM reservation;
```

Script Output | Query Result

SQL | All Rows Fetched: 2 in 0.011 seconds

	PASSENGERNAME	EMAIL
1	Fatima Ali	fatimaali@gmail.com
2	Usman Ali	usmanali@gmail.com

WHERE:

```
--->>>>>>>>>>>>>>>------- WHERE

SELECT flightid , flightnumber , airlineid FROM flight WHERE flightid> 1;
```

Script Output | Query Result | SQL | All Rows Fetched: 2 in 0.017 seconds

	FLIGHTID	FLIGHTNUMBER	AIRLINEID
1	2	FL002	102
2	3	FL003	103

BETWEEN :

```
--->>>>>>>>>>>>>>>------- BETWEEN

SELECT seatid , seatnumber FROM seats WHERE seatid BETWEEN 113 AND 117 ;
```

Script Output | Query Result | SQL | All Rows Fetched: 2 in 0.017 seconds

	FLIGHTID	FLIGHTNUMBER	AIRLINEID
1	2	FL002	102
2	3	FL003	103

```
SELECT * FROM seats WHERE flightid='&FlightIds' ORDER BY &columnname;

--->>>>>>>>>>>>>>>------- TO CHAR
```

Script Output | Query Result | SQL | All Rows Fetched: 6 in 0.02 seconds

	SEATID	FLIGHTID	SEATNUMBER
1	111	1	A1
2	112	1	B1
3	113	1	A2
4	114	1	B2
5	115	1	A3
6	116	1	B3

JOINS:

```
--natural join
SELECT flightid,flightnumber,seatid ,reservationid FROM reservation NATURAL JOIN  flight;
```

Script Output | Query Result

SQL | All Rows Fetched: 6 in 0.024 seconds

	FLIGHTID	FLIGHTNUMBER	SEATID	RESERVATIONID
1	1	FL001	111	211
2	1	FL001	112	212
3	2	FL002	118	213
4	3	FL003	125	214
5	2	FL002	115	215
6	2	FL002	119	216

```
---equijoin
SELECT P.passengername, passengerid , R.flightid , R.reservationid FROM passenger P JOIN reservation R
USING (passengerid) ORDER BY flightid ;
```

cript Output | Query Result

SQL | All Rows Fetched: 6 in 0.024 seconds

	FLIGHTID	FLIGHTNUMBER	SEATID	RESERVATIONID
1	1	FL001	111	211
2	1	FL001	112	212
3	2	FL002	118	213
4	3	FL003	125	214
5	2	FL002	115	215
6	2	FL002	119	216

```
---genrerate the bill of the passenger
SELECT P.passengerid, P.passengername,F.flightnumber , SUM(F.each_seat_payment)AS total_payment,COUNT(R.seatid)
FROM passenger P
JOIN reservation R ON P.passengerid = R.passengerid
JOIN flight F ON R.flightid = F.flightid
GROUP BY P.passengerid, P.passengername,F.flightnumber
;
```

Script Output | Query Result

SQL | All Rows Fetched: 4 in 0.032 seconds

	PASSENGERID	PASSENGERNAME	FLIGHTNUMBER	TOTAL_PAYMENT	COUNT(R.SEATID)
1	4	Aisha Khan	FL002	40000	2
2	3	Ahmed Malik	FL002	20000	1
3	7	Abdullah Khan	FL003	15000	1
4	1	Ali Khan	FL001	20000	2

```sql
--full outer join
SELECT passenger.passengername, flight.flightnumber
FROM passenger
FULL OUTER JOIN reservation ON passenger.passengerid = reservation.passengerid
FULL OUTER JOIN flight ON reservation.flightid = flight.flightid;
```

Script Output | Query Result

SQL | All Rows Fetched: 12 in 0.01 seconds

	PASSENGERNAME	FLIGHTNUMBER
1	Ali Khan	FL001
2	Ali Khan	FL001
3	Sara Ahmed	(null)
4	Ahmed Malik	FL002
5	Aisha Khan	FL002
6	Aisha Khan	FL002
7	Muhammad Hassan	(null)
8	Fatima Ali	(null)
9	Abdullah Khan	FL003

```sql
----right outer join
SELECT   R.reservationid,P.passengername FROM reservation R RIGHT OUTER JOIN passenger P ON P.passengerid=R.passengerid,
```

Query Result

SQL | All Rows Fetched: 12 in 0.011 seconds

	RESERVATIONID	PASSENGERNAME
1	211	Ali Khan
2	212	Ali Khan
3	213	Ahmed Malik
4	214	Abdullah Khan
5	215	Aisha Khan
6	216	Aisha Khan
7	(null)	Fatima Ali
8	(null)	Sara Ahmed
9	(null)	Zainab Ahmed

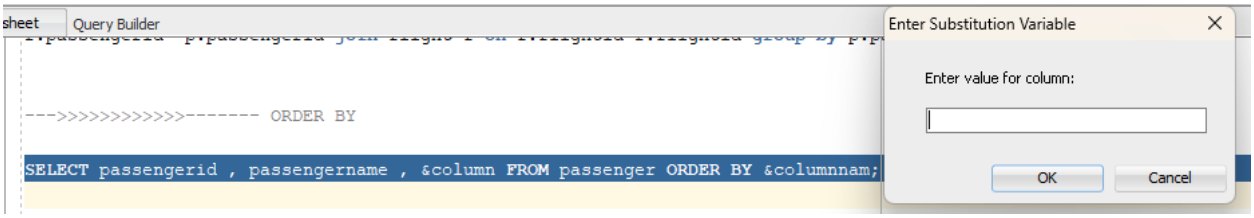
```sql
-----MULTIPLE JOINS
SELECT flight.flightid, flight.departureairportid, flight.flightnumber, reservation.reservationid, seats.seatid, seats.seatnumber
FROM flight
JOIN seats ON flight.flightid = seats.flightid
JOIN reservation ON seats.seatid = reservation.seatid;

-----------------
```

Script Output | Query Result

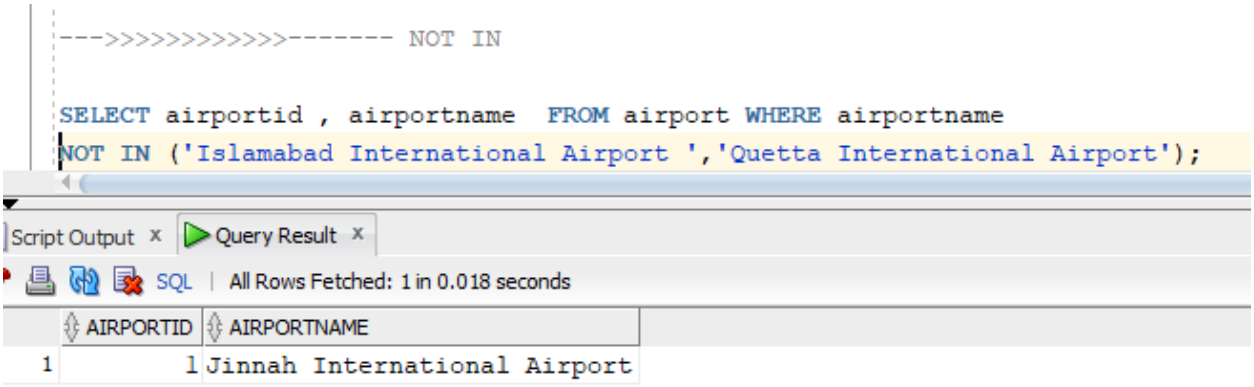
SQL | All Rows Fetched: 6 in 0.009 seconds

	FLIGHTID	DEPARTUREAIRPORTID	FLIGHTNUMBER	RESERVATIONID	SEATID	SEATNUMBER
1	1	1	FL001	211	111	A1
2	1	1	FL001	212	112	B1
3	1	1	FL001	215	115	A3
4	2	2	FL002	213	118	B1
5	2	2	FL002	216	119	A2
6	3	3	FL003	214	125	A2

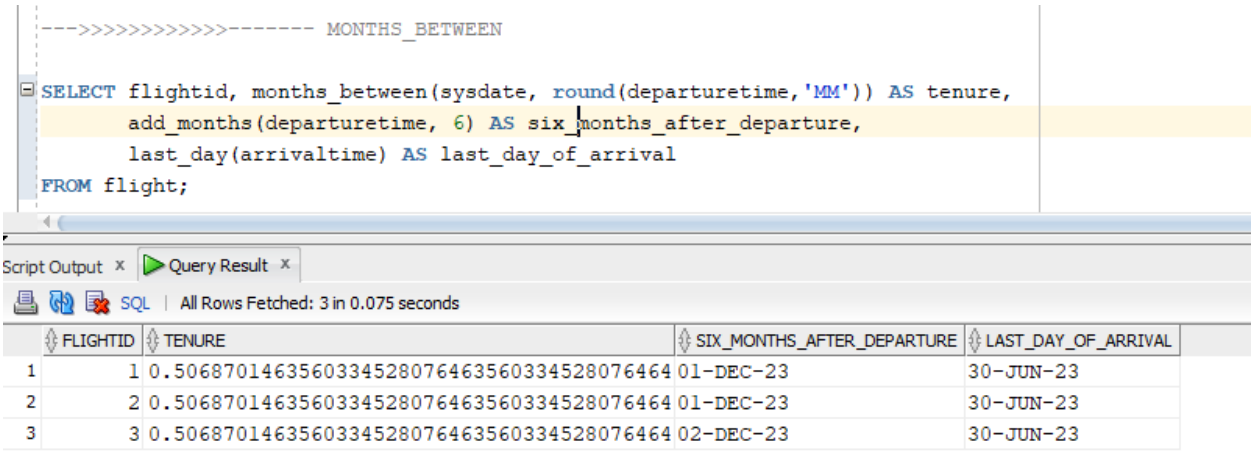
ORDER BY:



NOT IN:



MONTHS BETWEEN :



GROUP BY :

```
-- select the passenger id of the person who have reserved seats for more than one flight
SELECT passengerid , COUNT(flightid ) FROM reservation GROUP BY passengerid HAVING COUNT(flightid) >1  ;
```

Script Output | Query Result

SQL | All Rows Fetched: 2 in 0.012 seconds

	PASSENGERID	COUNT(FLIGHTID)
1	1	2
2	4	2

SET OPERATOR:

```
--->>>>>>>>>>>>>------- SET OPERATOR

--- select he seat number in the seat table which is not reserved

SELECT seatid ,  flightid FROM seats MINUS SELECT seatid ,flightid FROM reservation;
```

ript Output | Query Result

SQL | All Rows Fetched: 13 in 0.015 seconds

	SEATID	FLIGHTID
1	113	1
2	114	1
3	115	1
4	116	1
5	117	2
6	120	2
7	121	2
8	122	2
9	123	3

CHAPTER 6 :

CONCLUSION:

Overall, the Airline Management System offers numerous benefits, including improved operational efficiency, accurate data management, enhanced customer service, and effective financial management. The system plays a vital role in optimizing airline operations, improving passenger experience, and driving business growth.