CHITKARA UNIVERSITY PUNJAB

Airport Management System

Pranav Arora (2010990537)
Vishal Kumar (2010990786)
Saksham Kaushal (2010990625)
Shivam Soni (2010990665)
Paranav Mahajan (2010990514)

Introduction

We made this project to make the work easier for the managing team of the airport. With this they can easily access the details of flights, details of passengers travelling, status of flights, Also it can help them Monitor the flight schedule.

Case study Informal Description

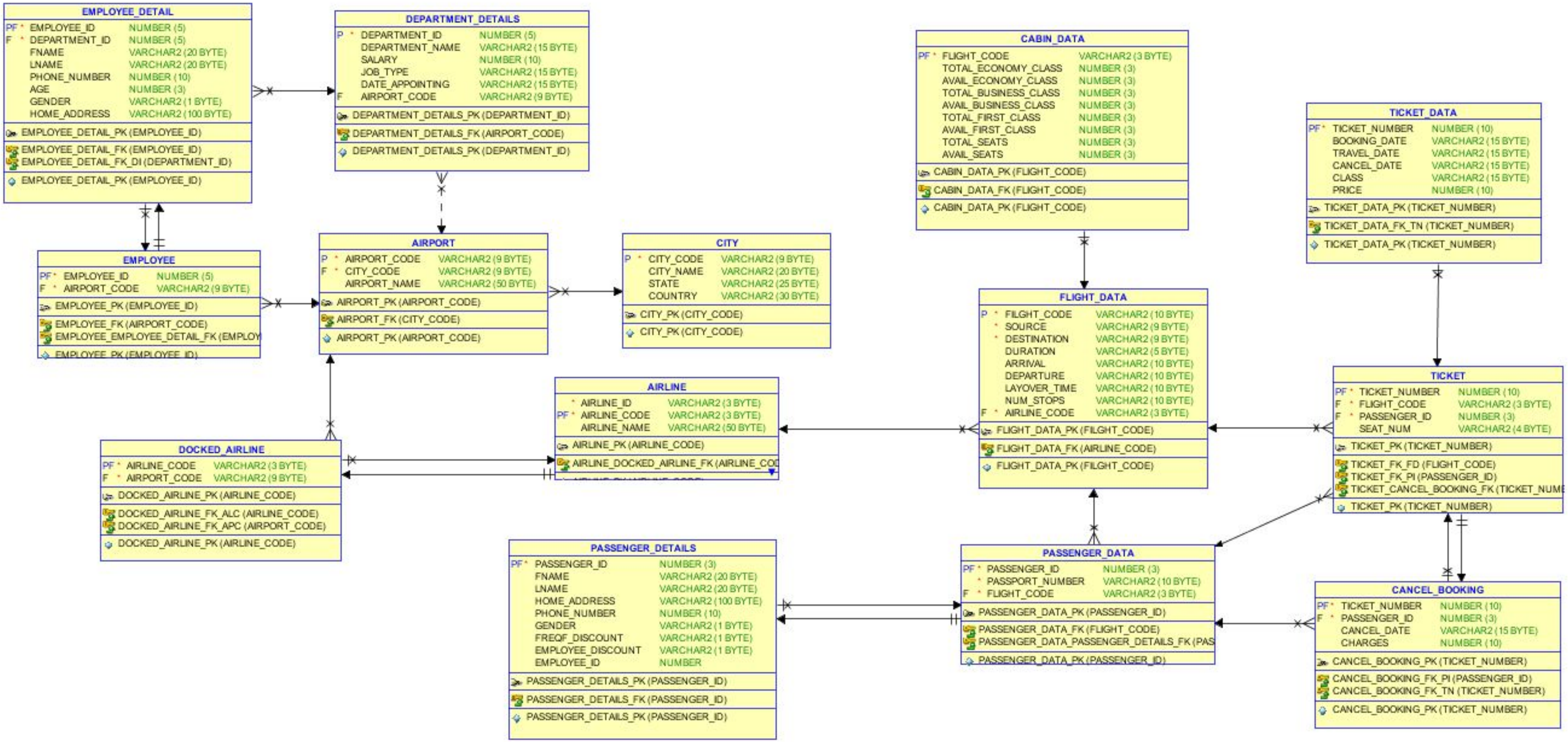
In this system a passenger can see how many seats are available in his particular flight, the flight status, get details of seats and check if he is eligible for discounts.

If an employee travels, he/she is also eligible for employee discounts

TABLES

- AIRLINE
- AIRPORT
- CABIN_DATA
- CANCEL_BOOKING
- CITY
- DEPARTMENT_DETAILS
- DOCKED_AIRLINE
- EMPLOYEE
- EMPLOYEE DETAIL
- FLIGHT_DATA
- PASSENGER_DATA
- PASSENGER_DETAILS
- TICKET
- TICKET_DATA

Case Study Logical Model



EMPLOYEE_DETAIL

PF *	EMPLOYEE_ID	NUMBER (5)
F *	DEPARTMENT_ID	NUMBER (5)
	FNAME	VARCHAR2 (20 BYTE)
	LNAME	VARCHAR2 (20 BYTE)
	PHONE_NUMBER	NUMBER (10)
	AGE	NUMBER (3)
	GENDER	VARCHAR2 (1 BYTE)
	HOME_ADDRESS	VARCHAR2 (100 BYTE)

EMPLOYEE_DETAIL_PK (EMPLOYEE_ID)
EMPLOYEE_DETAIL_FK (EMPLOYEE_ID)
EMPLOYEE_DETAIL_FK_DI (DEPARTMENT_ID)
EMPLOYEE_DETAIL_PK (EMPLOYEE_ID)

DEPARTMENT_DETAILS

P *	DEPARTMENT_ID	NUMBER (5)
	DEPARTMENT_NAME	VARCHAR2 (15 BYTE)
	SALARY	NUMBER (10)
	JOB_TYPE	VARCHAR2 (15 BYTE)
	DATE_APPOINTING	VARCHAR2 (15 BYTE)
F	AIRPORT_CODE	VARCHAR2 (9 BYTE)

DEPARTMENT_DETAILS_PK (DEPARTMENT_ID)
DEPARTMENT_DETAILS_FK (AIRPORT_CODE)
DEPARTMENT_DETAILS_PK (DEPARTMENT_ID)

CABIN_DATA

PF *	FLIGHT_CODE	VARCHAR2 (3 BYTE)
	TOTAL_ECONOMY_CLASS	NUMBER (3)
	AVAIL_ECONOMY_CLASS	NUMBER (3)
	TOTAL_BUSINESS_CLASS	NUMBER (3)
	AVAIL_BUSINESS_CLASS	NUMBER (3)
	TOTAL_FIRST_CLASS	NUMBER (3)
	AVAIL_FIRST_CLASS	NUMBER (3)
	TOTAL_SEATS	NUMBER (3)
	AVAIL_SEATS	NUMBER (3)

CABIN_DATA_PK (FLIGHT_CODE)
CABIN_DATA_FK (FLIGHT_CODE)
CABIN_DATA_PK (FLIGHT_CODE)

TICKET_DATA

PF *	TICKET_NUMBER	NUMBER (10)
	BOOKING_DATE	VARCHAR2 (15 BYTE)
	TRAVEL_DATE	VARCHAR2 (15 BYTE)
	CANCEL_DATE	VARCHAR2 (15 BYTE)
	CLASS	VARCHAR2 (15 BYTE)
	PRICE	NUMBER (10)

TICKET_DATA_PK (TICKET_NUMBER)
TICKET_DATA_FK_TN (TICKET_NUMBER)
TICKET_DATA_PK (TICKET_NUMBER)

EMPLOYEE

PF *	EMPLOYEE_ID	NUMBER (5)
F *	AIRPORT_CODE	VARCHAR2 (9 BYTE)

EMPLOYEE_PK (EMPLOYEE_ID)
EMPLOYEE_FK (AIRPORT_CODE)
EMPLOYEE_EMPLOYEE_DETAIL_FK (EMPLOY
EMPLOYEE_PK (EMPLOYEE_ID)

AIRPORT

P *	AIRPORT_CODE	VARCHAR2 (9 BYTE)
F *	CITY_CODE	VARCHAR2 (9 BYTE)
	AIRPORT_NAME	VARCHAR2 (50 BYTE)

AIRPORT_PK (AIRPORT_CODE)
AIRPORT_FK (CITY_CODE)
AIRPORT_PK (AIRPORT_CODE)

CITY

P *	CITY_CODE	VARCHAR2 (9 BYTE)
	CITY_NAME	VARCHAR2 (20 BYTE)
	STATE	VARCHAR2 (25 BYTE)
	COUNTRY	VARCHAR2 (30 BYTE)

CITY_PK (CITY_CODE)
CITY_PK (CITY_CODE)

FLIGHT_DATA

P *	FILGHT_CODE	VARCHAR2 (10 BYTE)
*	SOURCE	VARCHAR2 (9 BYTE)
*	DESTINATION	VARCHAR2 (9 BYTE)
	DURATION	VARCHAR2 (5 BYTE)
	ARRIVAL	VARCHAR2 (10 BYTE)
	DEPARTURE	VARCHAR2 (10 BYTE)
	LAYOVER_TIME	VARCHAR2 (10 BYTE)
	NUM_STOPS	VARCHAR2 (10 BYTE)
F *	AIRLINE_CODE	VARCHAR2 (3 BYTE)

FLIGHT_DATA_PK (FILGHT_CODE)
FLIGHT_DATA_FK (AIRLINE_CODE)
FLIGHT_DATA_PK (FILGHT_CODE)

TICKET

PF *	TICKET_NUMBER	NUMBER (10)
F *	FLIGHT_CODE	VARCHAR2 (3 BYTE)
F *	PASSENGER_ID	NUMBER (3)
	SEAT_NUM	VARCHAR2 (4 BYTE)

TICKET_PK (TICKET_NUMBER)
TICKET_FK_FD (FLIGHT_CODE)
TICKET_FK_PI (PASSENGER_ID)
TICKET_CANCEL_BOOKING_FK (TICKET_NUME
TICKET_PK (TICKET_NUMBER)

AIRLINE

*	AIRLINE_ID	VARCHAR2 (3 BYTE)
PF *	AIRLINE_CODE	VARCHAR2 (3 BYTE)
	AIRLINE_NAME	VARCHAR2 (50 BYTE)

AIRLINE_PK (AIRLINE_CODE)
AIRLINE_DOCKED_AIRLINE_FK (AIRLINE_COD

DOCKED_AIRLINE

PF *	AIRLINE_CODE	VARCHAR2 (3 BYTE)
F *	AIRPORT_CODE	VARCHAR2 (9 BYTE)

DOCKED_AIRLINE_PK (AIRLINE_CODE)
DOCKED_AIRLINE_FK_ALC (AIRLINE_CODE)
DOCKED_AIRLINE_FK_APC (AIRPORT_CODE)
DOCKED_AIRLINE_PK (AIRLINE_CODE)

PASSENGER_DETAILS

PF *	PASSENGER_ID	NUMBER (3)
	FNAME	VARCHAR2 (20 BYTE)
	LNAME	VARCHAR2 (20 BYTE)
	HOME_ADDRESS	VARCHAR2 (100 BYTE)
	PHONE_NUMBER	NUMBER (10)
	GENDER	VARCHAR2 (1 BYTE)
	FREQF_DISCOUNT	VARCHAR2 (1 BYTE)
	EMPLOYEE_DISCOUNT	VARCHAR2 (1 BYTE)
	EMPLOYEE_ID	NUMBER

PASSENGER_DETAILS_PK (PASSENGER_ID)
PASSENGER_DETAILS_FK (PASSENGER_ID)
PASSENGER_DETAILS_PK (PASSENGER_ID)

PASSENGER_DATA

PF *	PASSENGER_ID	NUMBER (3)
*	PASSPORT_NUMBER	VARCHAR2 (10 BYTE)
F *	FLIGHT_CODE	VARCHAR2 (3 BYTE)

PASSENGER_DATA_PK (PASSENGER_ID)
PASSENGER_DATA_FK (FLIGHT_CODE)
PASSENGER_DATA_PASSENGER_DETAILS_FK (PAS
PASSENGER_DATA_PK (PASSENGER_ID)

CANCEL_BOOKING

PF *	TICKET_NUMBER	NUMBER (10)
F *	PASSENGER_ID	NUMBER (3)
	CANCEL_DATE	VARCHAR2 (15 BYTE)
	CHARGES	NUMBER (10)

CANCEL_BOOKING_PK (TICKET_NUMBER)
CANCEL_BOOKING_FK_PI (PASSENGER_ID)
CANCEL_BOOKING_FK_TN (TICKET_NUMBER)
CANCEL_BOOKING_PK (TICKET_NUMBER)

Case Study Physical Model

CHITKARA
UNIVERSITY
PUNJAB

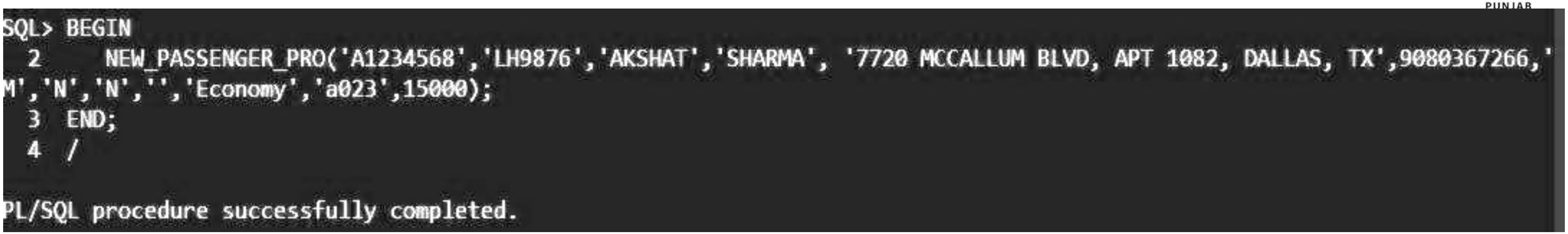
```
CREATE table "CITY" (
   "CITY_CODE"  VARCHAR2(9) NOT NULL,
   "CITY_NAME"  VARCHAR2(20),
   "STATE"     VARCHAR2(25),
   "COUNTRY"   VARCHAR2(30),
   constraint  "CITY_PK" primary key ("CITY_CODE"));

CREATE table "AIRPORT" (
   "AIRPORT_CODE" VARCHAR2(9) NOT NULL,
   "CITY_CODE"   VARCHAR2(9) NOT NULL,
   "AIRPORT_NAME" VARCHAR2(50),
   constraint  "AIRPORT_PK" primary key ("AIRPORT_CODE"));

CREATE table "TICKET" (
   "TICKET_NUMBER" NUMBER(10) NOT NULL,
   "FLIGHT_CODE"   VARCHAR2(10) NOT NULL,
   "PASSENGER_ID"  NUMBER(3) NOT NULL,
   "SEAT_NUM"     VARCHAR2(4),
   constraint  "TICKET_PK" primary key ("TICKET_NUMBER"));
```

```
CREATE table "AIRLINE" (
   "AIRLINE_ID"   VARCHAR2(3) NOT NULL,
   "AIRLINE_CODE" VARCHAR2(3) NOT NULL,
   "AIRLINE_NAME" VARCHAR2(50),
   constraint  "AIRLINE_PK" primary key ("AIRLINE_CODE"));

CREATE table "DOCKED_AIRLINE" (
   "AIRLINE_CODE" VARCHAR2(3) NOT NULL,
   "AIRPORT_CODE" VARCHAR2(9) NOT NULL,
   constraint  "DOCKED_AIRLINE_PK" primary key
("AIRLINE_CODE"));

CREATE table "PASSENGER_DATA" (
   "PASSENGER_ID"   NUMBER(3) NOT NULL,
   "PASSPORT_NUMBER" VARCHAR2(10) NOT NULL,
   "FLIGHT_CODE"    VARCHAR2(10) NOT NULL,
   constraint  "PASSENGER_DATA_PK" primary key
("PASSENGER_ID"));
```

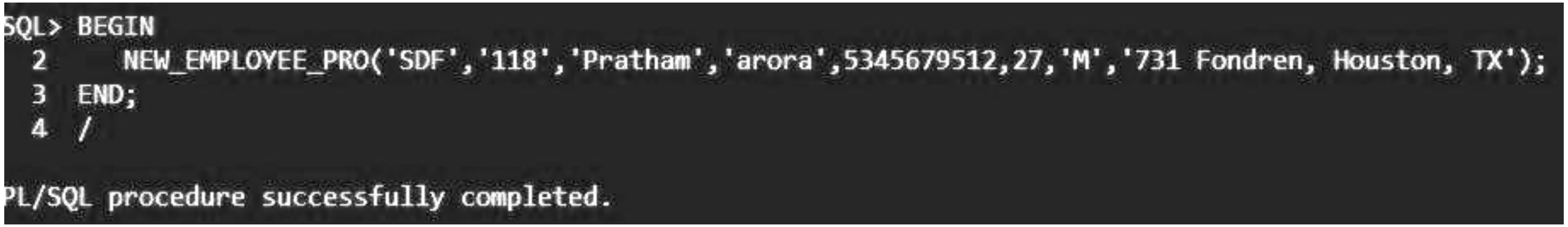
Case Study Interactive Queries

```
INSERT INTO CITY ("CITY_CODE","CITY_NAME","STATE","COUNTRY") VALUES('LKUS','Louisville','Kentucky','United States');
INSERT INTO AIRLINE ("AIRLINE_ID","AIRLINE_NAME","AIRLINE_CODE") VALUES('AI','Air India Limited','098');
INSERT INTO AIRPORT ("AIRPORT_NAME","AIRPORT_CODE","CITY_CODE") VALUES('Louisville International Airport','SDF','LKUS');
INSERT INTO DOCKED_AIRLINE ("AIRLINE_CODE","AIRPORT_CODE") VALUES('001','SDF');
INSERT INTO FLIGHT_DATA
( "FLIGHT_CODE","SOURCE","DESTINATION","DURATION","ARRIVAL","DEPARTURE","LAYOVER_TIME","NUM_STOPS","AIRLINE_CODE")
VALUES('AI2014','BOM','DFW','24hr','02:10','03:15' ,'3', '1','098');
BEGIN
  NEW_EMPLOYEE_PRO('SDF','118','Pratham','arora',5345679512,27,'M','731 Fondren, Houston, TX');
END;
BEGIN
  REM_PASSENGER_PRO(10,123);
END;
BEGIN
NEW_PASSENGER_PRO('A1234568','LH9876','AKSHAT','SHARMA', '7720 MCCALLUM BLVD, APT 1082, DALLAS,
TX',9080367266,'M','N','N','','Economy','a023',15000);
END;
```

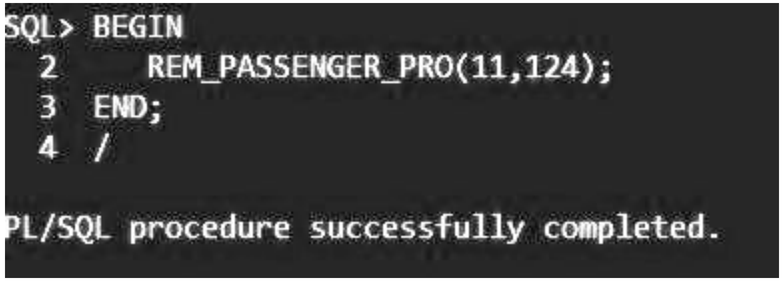
Output Screen Shots



```
SQL> BEGIN
  2     NEW_PASSENGER_PRO('A1234568','LH9876','AKSHAT','SHARMA', '7720 MCCALLUM BLVD, APT 1082, DALLAS, TX',9080367266,'
M','N','N','','Economy','a023',15000);
  3  END;
  4  /

PL/SQL procedure successfully completed.
```



```
SQL> BEGIN
  2     NEW_EMPLOYEE_PRO('SDF','118','Pratham','arora',5345679512,27,'M','731 Fondren, Houston, TX');
  3  END;
  4  /

PL/SQL procedure successfully completed.
```



```
SQL> BEGIN
  2     REM_PASSENGER_PRO(11,124);
  3  END;
  4  /

PL/SQL procedure successfully completed.
```

Conclusion and Future Work

While working on this project, we learnt a lot about creating a database and implementing all the queries related to creation of the database, modifying it and fetching the data from it. We have a good hold on the concepts related to ER diagrams and normalization now.

We also learned about procedures, sequences by this project and triggers.

Future Work

- We will be adding more features with Pl/Sql procedures.
- Also we will try to make it more interactive.

THANK YOU