**Case Study: Airline Reservation System**

**Problem Statement:**

Design and implement an Airline Reservation System using Oracle SQL and PL/SQL. The system will be used to manage flight bookings, track passenger information, and handle ticketing for an airline. Your task is to create the necessary database schema, populate the database with sample data, and develop PL/SQL procedures to handle reservation management, passenger information, and ticketing.

**Requirements:**

1. **Reservation Management**:
   * Implement the functionality to add, update, delete, and search for flight reservations.
   * Ensure that each reservation has attributes such as RESERVATION\_ID, FLIGHT\_ID, PASSENGER\_NAME, SEAT\_NUMBER, and STATUS.
2. **Passenger Information**:
   * Implement the functionality to manage passenger information.
   * Ensure that each passenger has attributes such as PASSENGER\_ID, FIRST\_NAME, LAST\_NAME, DOB, EMAIL, and PHONE\_NUMBER.
3. **Ticketing**:
   * Implement the functionality to handle ticketing.
   * Ensure that each ticket has attributes such as TICKET\_ID, RESERVATION\_ID, PASSENGER\_ID, FLIGHT\_ID, SEAT\_NUMBER, and TICKET\_PRICE.

**Tasks:**

1. **Design the Database Schema**:
   * Create the Reservations, Passengers, and Tickets tables with the appropriate fields and constraints.
   * Define primary keys and foreign keys to maintain data integrity.
2. **Populate the Database with Sample Data**:
   * Insert sample records into the Reservations, Passengers, and Tickets tables to facilitate testing of the system.
3. **Develop PL/SQL Procedures**:
   * Create a procedure to handle reservation management, including adding, updating, and deleting reservation records.
   * Create a procedure to manage passenger information, including adding, updating, and deleting passenger records.
   * Create a procedure to handle ticketing, including generating tickets and updating ticket information.

**Expected Outcomes:**

1. **Reservations Table**:
   * Contains all information about the flight reservations.
2. **Passengers Table**:
   * Stores details of all passengers who have booked flights.
3. **Tickets Table**:
   * Tracks all tickets issued for flight reservations.
4. **PL/SQL Procedures**:
   * Efficiently manage flight reservations, passenger information, and ticketing, maintaining accurate records in the database.

**Deliverables:**

1. SQL scripts to create the Reservations, Passengers, and Tickets tables.
2. SQL scripts to insert sample data into the tables.
3. PL/SQL scripts for the procedures to handle reservation management, passenger information, and ticketing.
4. Documentation explaining how to set up and use the system, including how to run the PL/SQL procedures.

**Database Schema:**

1. **Reservations Table**:
   * **RESERVATION\_ID**: Number, Primary Key
   * **FLIGHT\_ID**: Number, Foreign Key References Flights(FLIGHT\_ID)
   * **PASSENGER\_NAME**: Varchar2(100)
   * **SEAT\_NUMBER**: Varchar2(10)
   * **STATUS**: Varchar2(20) -- ('Confirmed', 'Cancelled')
2. **Passengers Table**:
   * **PASSENGER\_ID**: Number, Primary Key
   * **FIRST\_NAME**: Varchar2(50)
   * **LAST\_NAME**: Varchar2(50)
   * **DOB**: Date
   * **EMAIL**: Varchar2(100)
   * **PHONE\_NUMBER**: Varchar2(15)
3. **Tickets Table**:
   * **TICKET\_ID**: Number, Primary Key
   * **RESERVATION\_ID**: Number, Foreign Key References Reservations(RESERVATION\_ID)
   * **PASSENGER\_ID**: Number, Foreign Key References Passengers(PASSENGER\_ID)
   * **FLIGHT\_ID**: Number, Foreign Key References Flights(FLIGHT\_ID)
   * **SEAT\_NUMBER**: Varchar2(10)
   * **TICKET\_PRICE**: Number

**Case Study Task:**

* **Design**: Create the database schema as provided.
* **Implement**: Insert sample data into the Reservations, Passengers, and Tickets tables.
* **Develop**: Write PL/SQL procedures for handling reservation management, passenger information, and ticketing.
* **Test**: Test the procedures with various scenarios (e.g., managing reservations, updating passenger information, ticketing passengers, ensuring proper updates).