Project Topic: Design of a database system for a frequent flyer program

Deliver (1) the create_tables.sql script file containing your CREATE TABLE SQL statements, (2) the queries.sql script file containing your SQL queries

Project Component 2/3 Description

Task 1: Map the ER diagram to Relational Database Tables and Constraints (50%)

After the long discussions with the 550Airlines representatives, the group decided to use the ER diagram attached to the end of this file for representing the frequent flyer database. Your task 1 in this project component is to translate the ER diagram into a relational schema on the Oracle database. In other words you need to convert the ER entities and relationships into relational database tables satisfying the business rules and constraints. Apply the procedures we discussed in class to map the entities, attributes, and the Many-to-Many, 1-to-Many, and 1-to-1 relationships in the ER diagram to relational database tables. The following is the set of steps you need to apply in this project component part:

1. Study and understand the structure of the ER diagram. Stand on the significance of each entity, its attributes, and relationship(s) with other entities in the database. You should end up with **16** relational tables if you did the mapping correctly. If you find anything ambiguous in the diagram including the entities, attributes, and relationships, refer back to the 550Airlines representatives (The GTA and the Professor).
2. Translate the entities, attributes, and relationships in the ER diagram into their respective relational tables using the **CREATE TABLE SQL** statement. Make sure that your tables represent a valid mapping of the **Many-to-Many**, **1-to-Many**, and **1-to-1** relationships in the ER diagram and, accordingly, they contain the correct attributes, primary key, and foreign key constraints.
3. Store your CREATE TABLE SQL statements in a single .sql script using Oracle SQL Developer. Your .sql file should be named ***create_tables.sql***
4. Execute your .sql file on the GMU Oracle server and make sure that all the tables are created successfully.
5. The rubric for grading this task is very simple. For each correct CREATE TABLE SQL statement you will receive 2.5 pts (totalling to 16x2.5 = 40 pts). 10 points will be awarded for the successful execution of the .sql database creation script on the Oracle server.

   <u>Task 1 Hint and Naming Convention:</u>

   1. Use the same entity and attribute names as specified in the ER diagram in your relational table specifications.
   2. As you already know, Many-to-Many relationships between two entities in the ER diagram result in creating a third table to represent this relationship. To make it clearer with an example, the Many-to-Many relationship between the Flights and Promotions

entities in the ER diagram is realized using 3 relational tables: Promotions, Flights, and a third table linking the Promotions to Flights. Name this third table after the two entities participating in the Many-to-Many relationship. In our example, let it be named Promotions_Flights. Apply this naming convention when mapping the Many-to-Many relationships in the ER diagram to the relational tables on Oracle.

Task 2: SQL Query Implementation (50%)

Provide the SQL SELECT statements satisfying the following queries. Include your queries in a file named ***queries.sql***. **Number your queries using SQL Comments to make it easier on your GTA to grade your project. Not following this guideline would result in a deduction of 10%. Note that you may insert some dummy data to your tables to check the query execution; however the data insertions wouldn't be graded.**

1) *Select the promotion id, promotion action, and promotion period provided to a particular flight id.* (2.5%)

2) *Display all the flight ids, flight points, and the flight arrival dates for a particular passenger name.* (2.5%)

3) *Find the Flight ids and the number of promotions provided to each flight id.* (2.5%)

4) Find the passengers ids and names who arrived to Berlin between March 1st and March 15th 2023. (2.5%)

5) *Display for a particular flight id, the flight id, source, destination, the number of miles collected, and the trip ids and arrival dates included in the flight.* (2.5%)

6) *Find the number of expired cards available in the database*. (2.5%)

7) *Find the passenger with the maximum number of expired cards.* (2.5%)

8) *Find the redemption history of a particular passenger name. You should display the award ID, award description, passenger name,center id, and number of points redeemed.* (3%)

9) *Display the name and occupation of the passengers living in Fairfax.* (3%)

10) *Display the sum of points of the passengers living in Fairfax.* (3%)

11) *Display the passenger name with the maximum number of collected points*. (3%)

12) *Find the total number of points redeemed on a particular date.* (3%)

13) *Find the number of awards redeemed by a particular passenger id.* (3%)

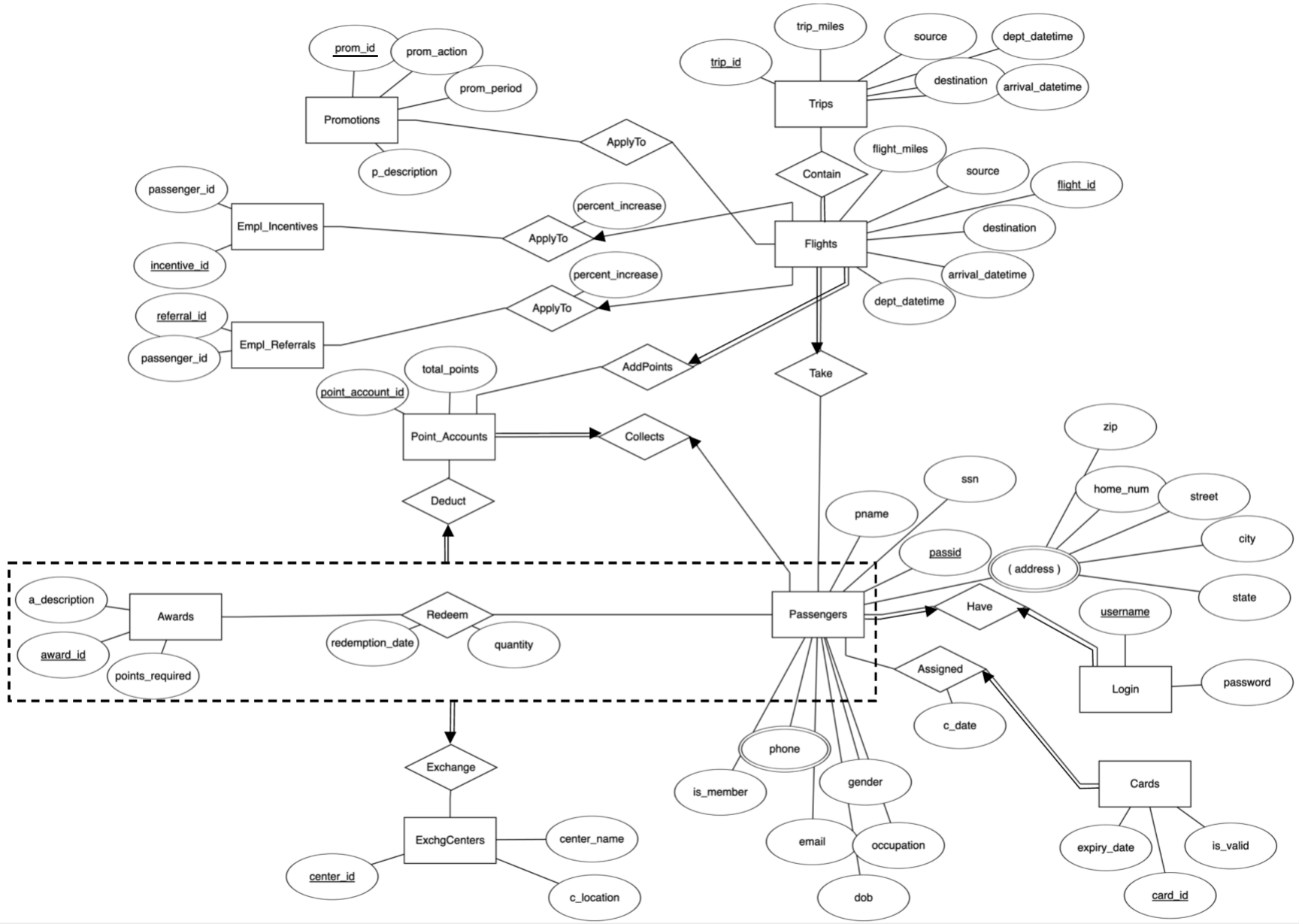
14) *Find the number of passengers who redeemed awards from a particular center id.* (3%)

15) *Find the total number of awards in the database.* (3%)

16) *Display a list of passenger names living in Fairfax and whose occupation is Engineer*. (2.5%)

17) *Find the list of trips not included in any flight.* (3%)

18) *Find the trip booked the most by passengers* (3%)


prom_id
prom_action
prom_period
Promotions
p_description
ApplyTo
trip_miles
source
dept_datetime
trip_id
destination
arrival_datetime
Trips
Contain
flight_miles
source
flight_id
destination
arrival_datetime
dept_datetime
Flights
passenger_id
Empl_Incentives
incentive_id
percent_increase
ApplyTo
referral_id
Empl_Referrals
passenger_id
percent_increase
ApplyTo
AddPoints
Take
total_points
point_account_id
Point_Accounts
Collects
Deduct
a_description
Awards
award_id
points_required
Redeem
redemption_date
quantity
Passengers
zip
ssn
home_num
street
pname
city
passid
(address)
state
Have
username
Login
password
Assigned
c_date
Cards
expiry_date
is_valid
card_id
phone
is_member
gender
email
occupation
dob
Exchange
ExchgCenters
center_id
center_name
c_location