**CSC453 Homework 1 Due September 23rd, 2018 at 11pm**

Submit the HW in D2L Dropbox.

A. **Relational Schema**

You are given requirements for a database schema of the National Football League (NFL) as follows:

* the NFL has many teams,
* each team has a teamId, a name, a city, a coach, a captain, and a set of players,
* each player has a playerId, a name, a position (such as quarterback, wingback, etc), a skill level, and a set of injury records (injury records have a description and id),
* each player belongs to only one team,
* a team captain is also a player,
* a game is played between two teams (referred to as host team and guest team) and has a date (such as Aug 3rd, 2010), a host team score, and a guest team score.

Create a SQL database schema for the above requirements. Write a few Insert statements to enter dummy data into the schema. Submit a Q1.sql file.

Note:

1. Your database schema must include **ALL** primary and foreign key requirements.
2. Setting additional custom constraints is optional. Please state your assumptions (as a comment in .sql file) when you add custom constraints.
3. The database schema must be syntactically correct, i.e., we should be able to create all tables by running the Q1.sql in Oracle SQLDeveloper.
4. You must code this schema using ‘Create Table’ statements . You should not use any other tool except Oracle SQLDeveloper. **No points will be given for auto-generated schemas using other 3rd party tools or web-based editors.**
5. If you are unsure what is a .sql file or an auto-generated schema, check Resources FAQ.

**B. Creating a Database**

Download the Loan.csv file from D2L.

1. Identify the schema from this file. Write a ‘Create Table’ statement for loans. You must name the table as ‘Loan’. Attribute names must match the column names in the .csv file. Declare appropriate datatypes based on data and add a PK constraint.

Save the `Create Table` statement in a Q2.sql file.

2. Now, upload data in Loan.csv into Loan table. There are 149 rows to upload. Inserting by hand is a waste of time, unless you have extreme patience. Write a function in your favorite programming language to open and read this CSV file line by line and generate statements of the form

Insert Into Loan Values (Val1, ‘Val2’, ‘Val3’, Val4,…);

3. Use the program-generated ‘Insert Into’ statements to upload data into a table.

4. After inserting all data, copy/Paste the following query and run it:

select \* from user\_segments where segment\_name=’LOAN’;

5. Submit (i) Q2.sql, (ii) a Q2.pdf consisting of screenshot of your query and result.

**Bonus:** The Loan.csv considers all data values as strings. However, some values are indeed numbers and dates. Map a string to a NUMBER or DATE data type in your program and insert into a Loan table declared accordingly.

Note:

1. For this example, it is strongly recommended to go from a working-to-working condition. First, write the create table statement. Then use the first line of data values to manually write an `Insert Into ` statement in SQLDeveloper. From that determine a pattern, based on which you should write your program to generate several `Insert Into` statements.
2. Copy and paste the output from your program in SQLDeveloper to see if your `Insert Into` statements are correct. If correct, data will be inserted.
3. No attribute is missing a value, so your program need not check for NULLs.
4. For more information on screenshots read Resources FAQ on the website. Failure to comply with Screenshot instructions will result in loss of points.

**C.**  **Basic SQL Operators**

Dataset: Trips

The schema is as follows:

|  |  |  |
| --- | --- | --- |
| Trips | Trips | TID, TripState, TravelMode, Fare |
| ByCar | TID, Rental Company, Mileage |
| ByTrain | TID, Type, Coach, TrainSpeed, NumberofStops |
| ByPlane | TID, Airline, Class, LayoverTime |

The Trips table stores information about trips taken by various modes of travel and their respective fares. Each trip, owing to its mode has some relevant information specific to the mode. For instance, a car trip has the rental company and the mileage, where as train trip is characterized by the trainspeed, type, coach, and number of stops. The TravelCompany is interested in answering the following queries.

1. List car rental companies which have a mileage of at least 27 miles/gallon.
2. List trip IDs taken on train costing strictly more than $150.
3. Find trip IDs and their fare that are not taken in the US i.e., `Non-US` trips.
4. Find the business class plane trip IDs that are greater than $1000.
5. Find any car trip more expensive than a trip taken on a train.
6. List a pair of distinct trips that have exactly the same value of mileage.
7. List a pair of distinct train trips that do not have the same speed.
8. Find those pair of trips in the same state with the same mode of travel. List such pairs only once.
9. Find a state in which trips have been taken by all three modes of transportation:  train, plane, and car.
10. Find the details of a) the most costly trip, b) the cheapest trip taken by either the air, rail, or two separate queries.

**Bonus points for writing the last query as a self-join with basic SQL operators (Filter, Project, Rename, Join (cross-join, natural join), Union, Intersect, and Difference). Do not use ALL, ANY, DISTINCT, GROUP BY HAVING, MAX, MIN.**

Submit C.sql consisting of all SQL statements, and C.pdf consisting of screenshots of queries and results. Not following screenshot instructions will result in loss of points.