Assessment 7 - SQL

Objective

Use MySQL to create a relational database representing an org chart. Submit your answers in a SQL file titles answers.sql

Build Specifications

Your naming conventions must match to be graded.

First, create the below tables and insert the following entries.

Make sure you identify primary keys and/or foreign keys as necessary.

persons (20 pts)

1. person_id (int)
   The unique personal id of a team member.
2. first_name (varchar(30))
   A person's first name; this is a required field.
3. last_name (varchar(30))
   A person's last name; this is a required field.
4. city (varchar(30))
   A person's city; this is an optional field.
5. leader_id
   The personal ID of a person's leader. If not null should match a person in the table.

person_id	first_name	last_name	city	leader_id
1	Riley	Shirk	Ann Arbor	5
2	Tia	Kansas	Detroit	5
4	Hemory	Phifer	Detroit	5
5	Russ	Thomas		
6	Montana	Greyerbiehl	New York	

Note: *The relationship between TM and leader is a many-to-one relationship, as a leader can have multiple TMs but a TM may only have one direct leader.*

roles (10 pts)

1. role_id (int)

   The unique id for a role at the FoC.
2. role_title (varchar(100))

   The full role title; this is a required field.

role_id	role_title
1	Senior Business Program Manager
2	Associate Business Program Manager
3	Team Leader
4	Director

Tasks

Now that we have our basic database setup, complete the following tasks:

1. It appears someone made a mistake in spelling Tiia's name. Update the **persons** table to the correct spelling. **(10 pts)**
2. We have a roles table, but we're currently not making use of it. Add a column to persons called role_id (int) and update the existing entries with their correct job title. You should not be able to add a role_id that does not exist in **roles** into **persons**. **(15 pts)**
3. We would like to know which cities the people in our database live in. Create a view called **view_cities** that retrieves all the *distinct* cities within persons while excluding null entries. **(15 pts)**
4. We would also know what job title each TM holds. Create a view called **view_titles** that retrieves the full name of each team member along with their title. **(20 pts)**
5. Add yourself, in addition to any necessary fields to the database (i.e. your role, person, etc.) **(10 pts)**

For review of SQL views: https://www.w3schools.com/sql/sql_view.asp

Rubric

Build Spec	Points Possible	Points Scored
persons table	20	
roles table	10	
Task 1	10	
Task 2	15	
Task 3	15	
Task 4	20	
Task 5	10	
Total	100	