Tradewater Tech Challenge

Good luck!

The first eight questions will test your basic knowledge of SQL

1. Please write a query to return the number of rows in a table called "TestTable".

```
SELECT COUNT(*) FROM TestTable
```

2. The table Person contains the following columns: PersonID, Firstname, Lastname, Email, Address, City, State, Phone. Write a query to return the first and last names for every row in the table.

```
SELECT Firstname, Lastname FROM Person
```

3. From the same Person table, pull each unique first name as well as a count of how many times that name appears in the table.

```
SELECT DISTINCT Firstname FROM Person
SELECT COUNT(DISTINCT Firstname) FROM Person
```

4. In a separate database, there are two tables: "Person" and "PersonAddress". "Person" contains the columns: PersonID, Firstname, Lastname, while "PersonAddress" contains: PersonAddressID, PersonID, Address1, Address2, City, State, Zip. Write a query to return the first and last names as well as the full address information for each person with an entry in both Person and PersonAddress.

```
SELECT Person.Firstname, Person.Lastname, PersonAddress.Address1,
PersonAddress.Address2, PersonAddress.City, PersonAddress.State, PersonAddress.Zip
FROM Person
INNER JOIN PersonAddress ON Person.PersonID=PersonAddress.PersonID;
```

5. In the same database as question 4, there are sometimes people in the Person table that we don't have address information for. Write a query that will return the same information as above, except that if a person does not have an entry in PersonAddress, each field from PersonAddress with instead be filled with the string "PLACEHOLDER DATA".

```
SELECT Person.Firstname, Person.Lastname, IFNULL(PersonAddress.Address1,”
PLACEHOLDER DATA"), IFNULL(PersonAddress.Address2,”PLACEHOLDER DATA"),
PersonAddress.City, PersonAddress.State, PersonAddress.Zip
FROM Person
INNER JOIN PersonAddress ON Person.PersonID=PersonAddress.PersonID;
```

6. Now we want to take the data pulled in question 5 and put it into a temporary table. Please modify your existing query to input the data into a temporary table called "Person_Info_Tmp". (Note: Depending on how you approach this problem, this may or may not be an issue, but you can assume that all strings are of data type varchar(100).)

```
CREATE TEMPORARY TABLE IF NOT EXISTS Person_Info_Tmp AS (
SELECT Person.Firstname, Person.Lastname, IFNULL(PersonAddress.Address1,”
PLACEHOLDER DATA"), IFNULL(PersonAddress.Address2,”PLACEHOLDER DATA"),
PersonAddress.City, PersonAddress.State, PersonAddress.Zip
FROM Person
INNER JOIN PersonAddress ON Person.PersonID=PersonAddress.PersonID
);
```

7. In this question, you will define a small database schema. This schema should have three tables: Person, PersonAddress, PersonEmail. They must meet the following criteria

- Person contains the following information:
    - First name
    - Last name
    - Date of Birth
    - Link to Person Info on mother
    - Link to Person Info on father
- PersonAddress contains the following information:
    - Address line 1
    - Address line 2
    - City
    - State
    - Zip code
    - Link to Person Info for the person whose address this is
- PersonEmail contains the following information:
- Person's email address
- field denoting whether the email address is the person's primary address or not (since a person might have more than one email address)
- Link to Person Info for the person whose email address this is.

Define the above tables in MySQL. Also define any keys or indexes as you feel is appropriate to make joins more efficient.

```
DROP TABLE IF EXISTS Persons;
CREATE TABLE Persons (
   PersonID int NOT NULL AUTO_INCREMENT,
   LastName varchar(255),
   FirstName varchar(255),
   MotherID int,
   FatherID int,
   PRIMARY KEY (PersonID)
);

DROP TABLE IF EXISTS PersonAddress;
CREATE TABLE PersonAddress (
   PersonID int,
   Address1 varchar(255),
   Address2 varchar(255),
         City varchar(255),
   State varchar(255),
   Zip int,
   PRIMARY KEY (PersonID)
);
```

```
DROP TABLE IF EXISTS PersonEmail;
CREATE TABLE PersonEmail (
   PersonID int,
   PersonEmail varchar(255),
   PersonEmailAdd varchar(255),
   IsMailPrimr bit,
   PRIMARY KEY (PersonID)
);
```

8. Now load some test data into your tables. It doesn't have to be a lot. No more than 5 rows per table is sufficient. You may use whatever technique you want to insert the data into the tables, but if you use an external file, be sure to attach the file you uploaded along with the rest of your answers.
Please turn in the commands you used to upload the data into the tables.

```
INSERT INTO Persons (PersonID, LastName, FirstName, MotherID, FatherID) VALUES
(1,'Mawyin','Jose', NULL, NULL);
INSERT INTO Persons (PersonID, LastName, FirstName, MotherID, FatherID) VALUES
(2,'Mawyin','Marlene', NULL, NULL);
INSERT INTO Persons (PersonID, LastName, FirstName, MotherID, FatherID) VALUES
(NULL,'Mawyin','Hugo', 2, 1);
INSERT INTO Persons (PersonID, LastName, FirstName, MotherID, FatherID) VALUES
(NULL,'Mawyin','Cristina', 2, 1);
INSERT INTO Persons (PersonID, LastName, FirstName, MotherID, FatherID) VALUES
(NULL,'Mawyin','Jose', 2, 1);

INSERT INTO PersonAddress (PersonID, Address1, Address2, City, State, Zip) VALUES (1,
'144-46', 'Blossom Avenue', 'Flushing', 'NY', 11655);
INSERT INTO PersonAddress (PersonID, Address1, Address2, City, State, Zip) VALUES (2,
'144-46', 'Blossom Avenue', 'Flushing', 'NY', 11655);
INSERT INTO PersonAddress (PersonID, Address1, Address2, City, State, Zip) VALUES (3,
'256', 'Avenue', 'Forest Hills', 'NY', 10456);
INSERT INTO PersonAddress (PersonID, Address1, Address2, City, State, Zip) VALUES (4,
'133', 'Street', 'Corona', 'NY', 90674);
INSERT INTO PersonAddress (PersonID, Address1, Address2, City, State, Zip) VALUES (5,
'1st', 'Main Street', 'Long Beach', 'NY', 47193);

INSERT INTO PersonEmail (PersonID, PersonEmail, IsMailPrimr) VALUES (1,
'Jose@rocket.com', 1);
INSERT INTO PersonEmail (PersonID, PersonEmail, IsMailPrimr) VALUES (2,
'Marlene@rocket.com', 1);
INSERT INTO PersonEmail (PersonID, PersonEmail, IsMailPrimr) VALUES (3,
'Hugo@rocket.com', 0);
INSERT INTO PersonEmail (PersonID, PersonEmail, IsMailPrimr) VALUES (4,
'Cristina@rocket.com', 1);
INSERT INTO PersonEmail (PersonID, PersonEmail, IsMailPrimr) VALUES (5,
'JoseJr@rocket.com', 0);
```

The following four questions will test your ability to analyze an existing dataset. Please use the MySQL database that has been provided. The database describes a test of three test emails sent to a fraction of an online mailing list attempting to convince recipients to take action by signing up through an online form. These questions call for you to analyze the performance of each mailing across various conditions.

9. For each of the three fundraising emails, please provide a topline summary of its performance as well as the SQL code you used to generate that summary. The summary should contain the following fields:
- Recipient count
- Open Count
- Opens per Recipient
- Click count
- Clicks per recipient
- Number of Actions
- Actions Per Recipient

- Recipient count:

Recipient count calculated from table “mailing_recipient”. Each distinct value in mailing_recipient_id is a different constituent that received an email.

```
SELECT COUNT(DISTINCT mailing_recipient_id) FROM mailing_recipient;
```

- Open Count:

Open Count calculated from filtering the mailing_recipient_click_type_id to only “open” records and counting the number of records.

```
SELECT mailing_recipient_click_type_id, count(mailing_recipient_click_type_id) FROM
mailing_recipient_click
WHERE mailing_recipient_click_type_id = 2;
```

- Opens per Recipient:

```
SELECT mailing_recipient_id, count(mailing_recipient_click_type_id)  FROM
mailing_recipient_click
WHERE mailing_recipient_click_type_id = 2
GROUP BY mailing_recipient_id
ORDER BY 1;
```

- Click count:

Click Count calculated from filtering the mailing_recipient_click_type_id to only “click” records and counting the number of records.

```
SELECT mailing_recipient_click_type_id, count(mailing_recipient_click_type_id) FROM
mailing_recipient_click
WHERE mailing_recipient_click_type_id = 1;
```

- Clicks per recipient:

```
SELECT mailing_recipient_id, count(mailing_recipient_click_type_id)  FROM
mailing_recipient_click
WHERE mailing_recipient_click_type_id = 1
GROUP BY mailing_recipient_id
ORDER BY 1;
```

- Number of Actions:

```
SELECT COUNT(DISTINCT cons_action_id) FROM cons_action;
```

- Actions Per Recipient:

```
SELECT cons_id, COUNT(cons_action_id) FROM cons_action
GROUP BY cons_id
ORDER BY 2 DESC;
```

10. Which of the three emails should be sent out to the larger list in order to maximize actions? Why do you think this is? Please limit your answer to one or two sentences.

A. First create a table that contains the unique mail recipients ID, the identifier if the received email was open or open AND clicked, and the ID for the type of email sent:

```
CREATE TABLE IF NOT EXISTS Sent_OpenClick
SELECT mailing_recipient_click.mailing_recipient_click_id,
mailing_recipient_click.mailing_recipient_click_type_id, mailing_recipient.mailing_id
FROM mailing_recipient_click
INNER JOIN mailing_recipient ON mailing_recipient_click. mailing_recipient_id =
mailing_recipient.mailing_recipient_id;
```

B. Then create a second table that aggregates the Open or Open AND Click actions per typer of email sent:

```
CREATE TABLE IF NOT EXISTS Opn_Clck_per_mail_AGG
SELECT *
FROM ( SELECT mailing_id AS Open_id, COUNT(*) AS OpenMailCount
FROM Sent_OpenClick
WHERE mailing_recipient_click_type_id = 2
GROUP BY mailing_id) AS A
JOIN ( SELECT mailing_id AS Close_id, COUNT(*) AS ClickMailCount
FROM Sent_OpenClick
WHERE mailing_recipient_click_type_id = 1
GROUP BY mailing_id) AS B
ON A.Open_id=B.Close_id;
```

C. Finally, we can compare which of the 3 emails led to more Open + Click results rather than just open by:

```
SELECT Open_id AS Email_ID, OpenMailCount, ClickMailCount, (ClickMailCount/
OpenMailCount) AS Action_Ratio
FROM Opn_Clck_per_mail_AGG
ORDER BY 4 DESC;
```

11. Provide a similar topline assessment of each email, but this time break each topline up into stats for male and female recipients of each mailing.

A. First create a new table containing the gender and age breakdown for every email that was received.

```
CREATE TABLE IF NOT EXISTS Age_Gender_MailID
SELECT cons.gender, cons.age_id, mailing_recipient.mailing_id
FROM cons
```

```
INNER JOIN mailing_recipient ON mailing_recipient.cons_id = cons.cons_id;
```

B. Finally count the per-gender id’s grouped by the mail type id.

```
SELECT mailing_id, gender, COUNT(gender) AS AGE_Groups_Count
FROM Age_Gender_MailID
GROUP BY mailing_id, gender;
```

12. Provide a topline assessment of each email, breaking each mailing into the following age groups:
- 18-35
- 36-60
- 60+

Using the same table from Q.11, count the per-age id’s grouped by the mail type id.

```
SELECT mailing_id, age_id, COUNT(age_id) AS AGE_Groups_Count
FROM Age_Gender_MailID
GROUP BY mailing_id, age_id;
```

13. In a scripting language of your choice, write a program that runs the SQL code you wrote for question 9 and writes the output into a comma separated file. Please provide the output file and your program.

The following question will test your proficiency with utilizing APIs to produce useful information.

Answer in Github depository.

14. Github, the software version control website maintains an API with which developers can interact with their services (https://docs.github.com/en/rest). Use the Github API to create a list of all repositories associated with an account of your choosing (e.g. Mozilla - https://github.com/mozilla). Please export the ID, Name, Description and URL fields for each repository into a human-readable CSV file. Please provide all code used (excluding any private or authenticating information from your own account) as well as the final CSV file. Please select an account with at least three repositories for export. You may use the language or toolset you are most familiar with.

Answer in Github depository.

The following question is a hypothetical situation we would like you to provide solutions for.

15. Tradewater wants to identify from among its new and existing sales regions, those areas which have the greatest tendency towards producing high numbers of sales so that they may be targeted with additional advertising and personnel resources. Your task is to use the information we already have about our sales regions which new regions will be the most likely to produce the highest number of sales.

In your response, please consider the following questions:
15.a How do you identify these sales regions?
There is strong regulation to deter further use of industrial or commercial products that are or may degrade into greenhouse gases (GHG). Locations where Tradewater can generate new sales is where new access to old stocks of GHG becomes available.

This may be because of new local regulations that incentivize removal of GHG or because a new plant that recycles industrial or commercial equipment opens up and needs the specialized know-how of Tradewater to get rid of GHG related downstream material.

15.b What information would you use to predict high value regions? (Feel free to assume we have some basic demographic data as well as a history of interactions we have had with clients and potential clients. You will not be penalized for assuming we have information that we don't actually have.)

High value regions are those where the greatest concentration of GHG materials are located with the lowest cost of capture/treatment per unit of product. For example, areas of high population density or areas with a high density of industrial or commercial zones.

Population density maps may be overlaid with those of industrial and commercial zones to map those ares where investment in GHG material collections would get the most return on investment.

15.c What techniques or tools would you use to confirm which pieces of data from (15.b) correlate with a regions sales, and how would you determine how much weight to give each data point?

I would use a dataset that contains the per county (or per city) average population density and industrial, commercial zone density statistics. The weighting would depend on if historically Tradewater has generated more revenue from GHG removal from domestic or commercial sources.

15.d How would you verify your prediction?

I can map the location of Tradewater collection centers across all its regions and see if there is a correlation between revenue per site and the density factors already mentioned.