Coding Challenges - PetPals, The Pet Adoption Platform

Tasks:

1. Provide a SQL script that initializes the database for the Pet Adoption Platform ”PetPals”.

```
mysql> create database PetPals;
Query OK, 1 row affected (0.01 sec)

mysql> use PetPals;
Database changed
```

2. Create tables for pets, shelters, donations, adoption events, and participants.

3. Define appropriate primary keys, foreign keys, and constraints.

```
mysql> create table Pets(
    -> PetID INT PRIMARY KEY,
    -> Name VARCHAR(100) NOT NULL,
    -> Age INT NOT NULL,
    -> Breed VARCHAR(100) NOT NULL,
    -> Type VARCHAR(100) NOT NULL,
    -> AvailableForAdoption BIT NOT NULL
    -> );
Query OK, 0 rows affected (0.02 sec)

mysql> CREATE TABLE Shelters (
    ->     ShelterID INT PRIMARY KEY,
    ->     Name VARCHAR(100) NOT NULL,
    ->     Location VARCHAR(255) NOT NULL
    -> );
Query OK, 0 rows affected (0.01 sec)
```

```
mysql> CREATE TABLE Donations(
    -> DonationID INT PRIMARY KEY,
    -> DonorName VARCHAR(100) NOT NULL,
    -> DonationType VARCHAR(100) NOT NULL,
    -> DonationAmount DECIMAL(10,2),
    ->     DonationItem VARCHAR(100),
    ->     DonationDate DATETIME NOT NULL
    -> );
Query OK, 0 rows affected (0.01 sec)

mysql>
mysql> CREATE TABLE AdoptionEvents (
    -> EventID INT PRIMARY KEY,
    -> EventName VARCHAR(100) NOT NULL,
    -> EventDate DATETIME NOT NULL,
    -> Location VARCHAR(100) NOT NULL
    -> );
Query OK, 0 rows affected (0.01 sec)

mysql> CREATE TABLE Participants (
    -> ParticipantID INT PRIMARY KEY,
    -> ParticipantName VARCHAR(100) NOT NULL,
    -> ParticipantType VARCHAR(100) NOT NULL,
    -> EventID INT,
    -> FOREIGN KEY (EventID) REFERENCES AdoptionEvents(EventID)
    -> );
Query OK, 0 rows affected (0.02 sec)
```

4. Ensure the script handles potential errors, such as if the database or tables already exist.

```
DROP TABLE IF EXISTS Pets;

CREATE TABLE Pets (...);
```

5. Write an SQL query that retrieves a list of available pets (those marked as available for adoption) from the "Pets" table. Include the pet's name, age, breed, and type in the result set. Ensure that the query filters out pets that are not available for adoption.

```
mysql> select * from pets where AvailableForAdoption = 1;

+-------+----------+-----+-----------------+------+-------------------------------------------+

| PetID | Name     | Age | Breed           | Type | AvailableForAdoption                      |

+-------+----------+-----+-----------------+------+-------------------------------------------+

|     1 | Buddy    |   2 | Labrador        | Dog  | 1
           |

|     3 | Whiskers |   3 | Persian         | Cat  | 1
           |

|     5 | Shadow   |   1 | German Shepherd | Dog  | 1
           |

|     6 | Luna     |   4 | Siamese         | Cat  | 1
           |

+-------+----------+-----+-----------------+------+-------------------------------------------+

4 rows in set (0.00 sec)
```

6. Write an SQL query that retrieves the names of participants (shelters and adopters) registered for a specific adoption event. Use a parameter to specify the event ID. Ensure that the query joins the necessary tables to retrieve the participant names and types.

```
mysql> SELECT ParticipantID, ParticipantName from Participants

    -> WHERE EventID = 2;

+---------------+-----------------+

| ParticipantID | ParticipantName |

+---------------+-----------------+

|             2 | Safe Haven      |
|           4 | Emily Smith     |
+---------------+-----------------+
2 rows in set (0.00 sec)
```

7. Create a stored procedure in SQL that allows a shelter to update its information (name and location) in the "Shelters" table. Use parameters to pass the shelter ID and the new information. Ensure that the procedure performs the update and handles potential errors, such as an invalid shelter ID.

```
mysql> UPDATE Shelters
    -> SET Name = 'paws', Location = 'Chennai'
    -> WHERE ShelterID = 3 ;
Query OK, 0 rows affected (0.00 sec)
Rows matched: 1  Changed: 0  Warnings: 0
```

```
ALTER TABLE Donations ADD ShelterID INT;
ALTER TABLE Donations ADD FOREIGN KEY (ShelterID) REFERENCES Shelters(ShelterID);
```

8. Write an SQL query that calculates and retrieves the total donation amount for each shelter (by shelter name) from the "Donations" table. The result should include the shelter name and the total donation amount. Ensure that the query handles cases where a shelter has received no donations.

```
mysql> select s.Name AS ShelterName, COALESCE(sum(d.DonationAmount), 0) as TotalDonations
    -> from Shelters s
    -> left join Donations d on s.ShelterID = d.ShelterID
    -> group by s.ShelterID, s.Name;
+-------------+----------------+
| ShelterName | TotalDonations |
+-------------+----------------+
| Happy Paws  |           0.00 |
| Safe Haven  |           0.00 |
| paws        |           0.00 |
+-------------+----------------+
3 rows in set (0.01 sec)
```

9. Write an SQL query that retrieves the names of pets from the "Pets" table that do not have an owner (i.e., where "OwnerID" is null). Include the pet's name, age, breed, and type in the result set.

```
ALTER TABLE Pets ADD COLUMN OwnerID INT;

ALTER TABLE Pets ADD CONSTRAINT fk_owner FOREIGN KEY (OwnerID) REFERENCES Users(UserID);

mysql> SELECT PetID, Name, Age, Breed, Type
    -> FROM Pets
    -> WHERE OwnerID IS NULL;
+-------+----------+-----+------------------+------+
| PetID | Name     | Age | Breed            | Type |
+-------+----------+-----+------------------+------+
|     1 | Buddy    |   2 | Labrador         | Dog  |
|     2 | Milo     |   5 | Beagle           | Dog  |
|     3 | Whiskers |   3 | Persian          | Cat  |
|     4 | Coco     |   7 | Golden Retriever | Dog  |
|     5 | Shadow   |   1 | German Shepherd  | Dog  |
|     6 | Luna     |   4 | Siamese          | Cat  |
+-------+----------+-----+------------------+------+
6 rows in set (0.00 sec)
```

10. Write an SQL query that retrieves the total donation amount for each month and year (e.g., January 2023) from the "Donations" table. The result should include the month-year and the corresponding total donation amount. Ensure that the query handles cases where no donations were made in a specific month-year.

```
mysql> select concat(DATE_FORMAT(DonationDate, '%m'),'-', DATE_FORMAT(DonationDate, '%Y')) as MONTH_YEAR, sum(DonationAmount) as MonthlyTotal

    -> from Donations

    -> group by YEAR(DonationDate), MONTH(DonationDate);

+------------+--------------+

| MONTH_YEAR | MonthlyTotal |

+------------+--------------+

| 01-2023    |      5000.00 |

| 02-2023    |         NULL |

| 03-2023    |      3000.00 |

| 03-2024    |      2000.00 |

+------------+--------------+

4 rows in set (0.01 sec)
```

11. Retrieve a list of distinct breeds for all pets that are either aged between 1 and 3 years or older than 5 years.

```
mysql> select name, breed, age from pets where age between 1 and 3 or age>5;


+----------+------------------+-----+

| name     | breed            | age |

+----------+------------------+-----+

| Buddy    | Labrador         |   2 |

| Whiskers | Persian          |   3 |

| Coco     | Golden Retriever |   7 |

| Shadow   | German Shepherd  |   1 |

+----------+------------------+-----+

4 rows in set (0.00 sec)
```

12. Retrieve a list of pets and their respective shelters where the pets are currently available for adoption.

```
mysql> select p.name, p.age, p.type, p.breed, s.name from pets p join shelters s on p.shelterID =
s.shelterID where AvailableForAdoption = 1;

+----------+-----+------+------------------+------------+

| name     | age | type | breed            | name       |

+----------+-----+------+------------------+------------+

| Buddy    |   3 | Dog  | Labrador         | Happy Paws |

| Whiskers |   2 | Cat  | Persian          | Safe Haven |

| Max      |   6 | Dog  | Golden Retriever | Safe Haven |

| Rocky    |   5 | Dog  | Bulldog          | paws       |

| Mittens  |   1 | Cat  | Maine Coon       | paws       |

+----------+-----+------+------------------+------------+

5 rows in set (0.00 sec)
```

13. Find the total number of participants in events organized by shelters located in specific city.

Example: City=Chennai

```
mysql> select location, count(*) as Number_of_participants from AdoptionEvents group by location;

+----------+------------------------+

| location | Number_of_participants |

+----------+------------------------+

| Chennai  |                      1 |

| Mumbai   |                      1 |

+----------+------------------------+

2 rows in set (0.01 sec)
```

(or)

```
mysql> select location, count(*) as Number_of_participants from AdoptionEvents where location =
'Chennai';

+----------+------------------------+

| location | Number_of_participants |

+----------+------------------------+
| Chennai  |                      1 |
+----------+------------------------+
1 row in set (0.00 sec)
```

14. Retrieve a list of unique breeds for pets with ages between 1 and 5 years.

```
mysql> select distinct breed from pets where age between 1 and 5;
+-----------------+
| breed           |
+-----------------+
| Labrador        |
| Beagle          |
| Persian         |
| German Shepherd |
| Siamese         |
| Bulldog         |
| Maine Coon      |
+-----------------+
7 rows in set (0.00 sec)
```

(or)

```
mysql> select name, age, breed from pets group by breed having age between 1 and 5;
+----------+-----+-----------------+
| name     | age | breed           |
+----------+-----+-----------------+
| Buddy    |   2 | Labrador        |
| Milo     |   5 | Beagle          |
| Whiskers |   3 | Persian         |
| Shadow   |   1 | German Shepherd |
| Luna     |   4 | Siamese         |
| Rocky    |   5 | Bulldog         |

| Mittens  |   1 | Maine Coon      |

+----------+-----+-----------------+

7 rows in set (0.00 sec)
```

15. Find the pets that have not been adopted by selecting their information from the 'Pet' table.

```
mysql> select * from pets where AvailableForAdoption = 1;

+-------+----------+-----+------------------+------+---------------------------------------+---------+-----------+

| PetID | Name     | Age | Breed            | Type | AvailableForAdoption                  | OwnerID |
ShelterID |

+-------+----------+-----+------------------+------+---------------------------------------+---------+-----------+

|     1 | Buddy    |   2 | Labrador         | Dog  | 1                                     |    NULL |      NULL |

|     3 | Whiskers |   3 | Persian          | Cat  | 1                                     |    NULL |      NULL |

|     5 | Shadow   |   1 | German Shepherd  | Dog  | 1                                     |    NULL |      NULL |

|     6 | Luna     |   4 | Siamese          | Cat  | 1                                     |    NULL |      NULL |

|    11 | Buddy    |   3 | Labrador         | Dog  | 1                                     |    NULL |         1 |

|    12 | Whiskers |   2 | Persian          | Cat  | 1                                     |    NULL |         2 |

|    13 | Rocky    |   5 | Bulldog          | Dog  | 1                                     |    NULL |         3 |

|    15 | Max      |   6 | Golden Retriever | Dog  | 1                                     |    NULL |         2 |

|    16 | Mittens  |   1 | Maine Coon       | Cat  | 1                                     |    NULL |         3 |

+-------+----------+-----+------------------+------+---------------------------------------+---------+-----------+

9 rows in set (0.00 sec)
```

16. Retrieve the names of all adopted pets along with the adopter's name from the 'Adoption' and 'User' tables.

```
mysql> select u.Name as AdopterName, p.Name as PetName

    -> from Adoption a

    -> join Pets p on a.PetID = p.PetID

    -> join Users u on a.UserID = u.UserID;

+-------------+---------+
| AdopterName | PetName |
+-------------+---------+
| John        | Buddy   |
| Alice       | Milo    |
+-------------+---------+
2 rows in set (0.00 sec)
```

17. Retrieve a list of all shelters along with the count of pets currently available for adoption in each shelter.

```
mysql> select s.Name, s.Location, count(p.PetID) as Available_Count
    -> from Shelters s
    -> join Pets p ON s.ShelterID = p.ShelterID
    -> where p.AvailableForAdoption = 1
    -> group by s.ShelterID, s.Name, s.Location;
+------------+-----------+-----------------+
| Name       | Location  | Available_Count |
+------------+-----------+-----------------+
| Happy Paws | Chennai   |               1 |
| Safe Haven | Bangalore |               2 |
| paws       | Chennai   |               2 |
+------------+-----------+-----------------+
3 rows in set (0.00 sec)
```

18. Find pairs of pets from the same shelter that have the same breed.

```
mysql> select p1.PetID AS Pet1_ID, p1.Name AS Pet1_Name,
    ->p2.PetID AS Pet2_ID, p2.Name AS Pet2_Name,
    ->p1.Breed, p1.ShelterID
    -> from Pets p1
    -> join Pets p2
    -> on p1.Breed = p2.Breed
    -> and p1.ShelterID = p2.ShelterID
    -> and p1.PetID < p2.PetID
    -> order by p1.ShelterID, p1.Breed;
Empty set (0.00 sec)
```

19. List all possible combinations of shelters and adoption events.

```
mysql> SELECT s.ShelterID, s.Name AS ShelterName,
    ->        e.EventID, e.EventName
    -> FROM Shelters s
    -> CROSS JOIN AdoptionEvents e
    -> ORDER BY s.ShelterID, e.EventID;
+-----------+-------------+---------+---------------------+
| ShelterID | ShelterName | EventID | EventName           |
+-----------+-------------+---------+---------------------+
|         1 | Happy Paws  |       1 | Pet Lovers Meet     |
|         1 | Happy Paws  |       2 | Furry Adoption Fair |
|         2 | Safe Haven  |       1 | Pet Lovers Meet     |
|         2 | Safe Haven  |       2 | Furry Adoption Fair |
|         3 | paws        |       1 | Pet Lovers Meet     |
|         3 | paws        |       2 | Furry Adoption Fair |
+-----------+-------------+---------+---------------------+
6 rows in set (0.00 sec)
```

20. Determine the shelter that has the highest number of adopted pets.

```
mysql> SELECT ShelterID, COUNT(*) AS TotalAdoptedPets
    -> FROM Pets
    -> WHERE AvailableForAdoption = 1 and ShelterID IS NOT NULL
    -> GROUP BY ShelterID
    -> ORDER BY TotalAdoptedPets DESC
    -> LIMIT 1;
+-----------+------------------+
| ShelterID | TotalAdoptedPets |
+-----------+------------------+
|         2 |                2 |
+-----------+------------------+
1 row in set (0.00 sec)
```