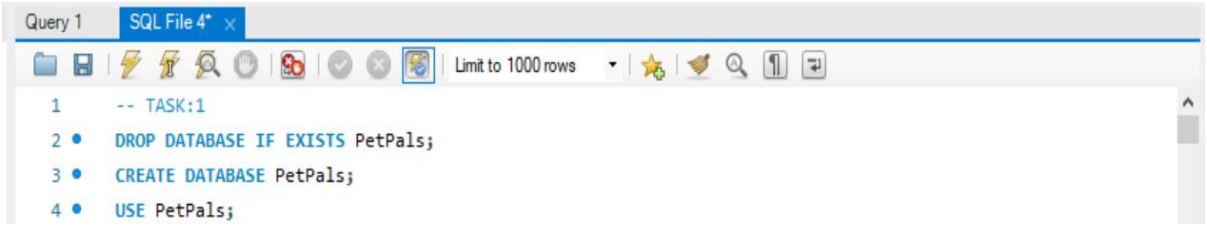
# CODING CHALLENGE- PETPALS, THE PET ADOPTION PLATFORM

# SUBMITTED BY: SRINIDHI.V

TASK 1 :**Provide a SQL script that initializes the database for the Pet Adoption Platform "PetPals".**



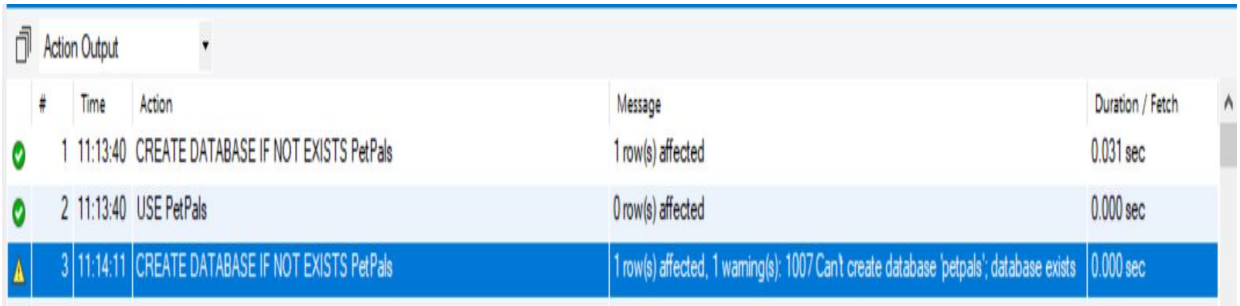

**TASK 2. Create tables for pets, shelters, donations, adoption events, and participants.**
**TASK 3. Define appropriate primary keys, foreign keys, and constraints.**

```
CREATE TABLE IF NOT EXISTS Shelters (
    ShelterID INT PRIMARY KEY,
    Name VARCHAR(100),
    Location VARCHAR(100)
);

CREATE TABLE IF NOT EXISTS Users (
    UserID INT PRIMARY KEY,
    Name VARCHAR(100)
);

CREATE TABLE IF NOT EXISTS Pets (
    PetID INT PRIMARY KEY,
    Name VARCHAR(100),
    Age INT,
    Breed VARCHAR(100),
    Type VARCHAR(50),
    AvailableForAdoption BIT,
    OwnerID INT,
    ShelterID INT,
    FOREIGN KEY (OwnerID) REFERENCES Users(UserID),
    FOREIGN KEY (ShelterID) REFERENCES Shelters(ShelterID)
);
```

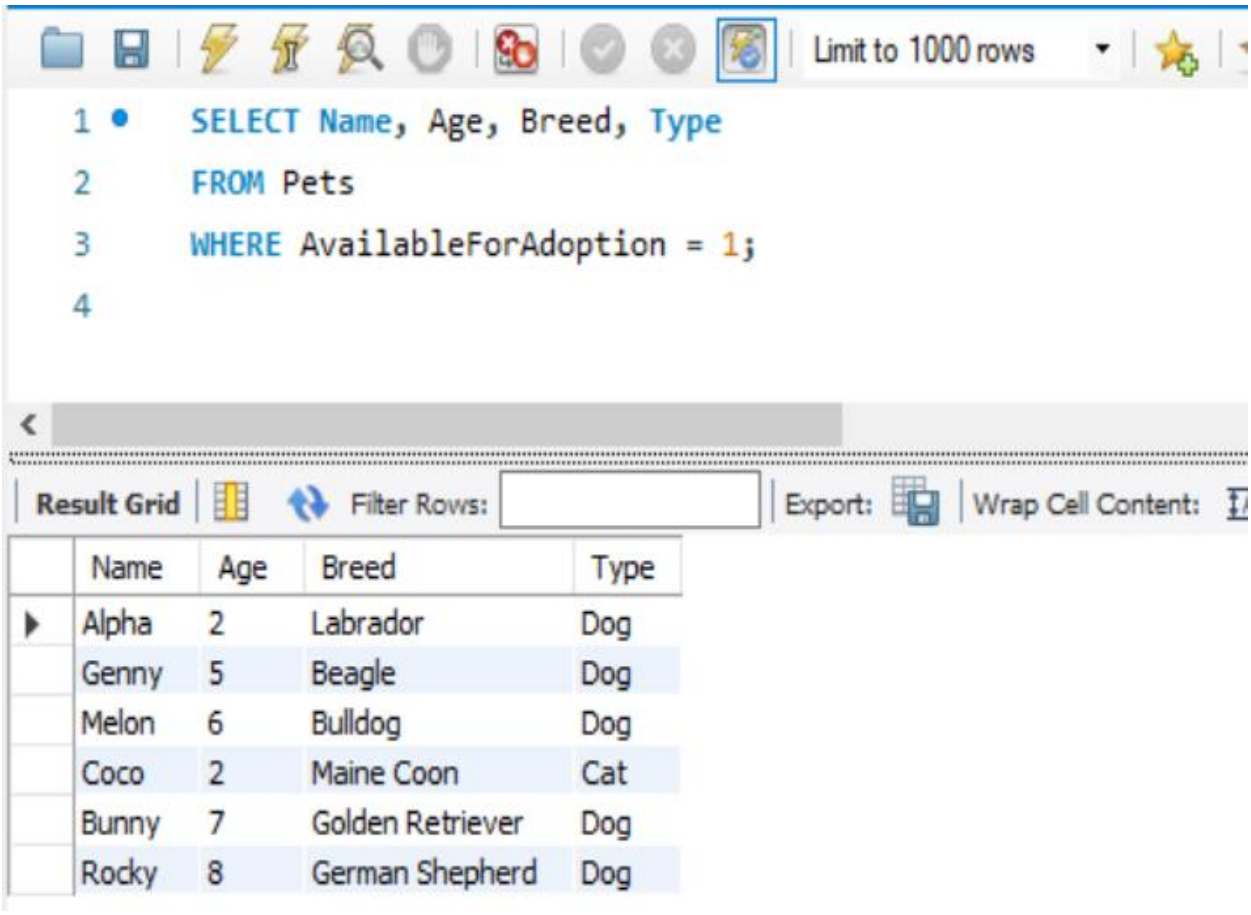
```
CREATE TABLE IF NOT EXISTS Donations (
   DonationID INT PRIMARY KEY,
   DonorName VARCHAR(100),
   DonationType VARCHAR(50),
   DonationAmount DECIMAL(10,2),
   DonationItem VARCHAR(100),
   DonationDate DATETIME,
   ShelterID INT,
   FOREIGN KEY (ShelterID) REFERENCES Shelters(ShelterID)
);

CREATE TABLE IF NOT EXISTS AdoptionEvents (
   EventID INT PRIMARY KEY,
   EventName VARCHAR(100),
   EventDate DATETIME,
   Location VARCHAR(100)
);


CREATE TABLE IF NOT EXISTS Participants (
   ParticipantID INT PRIMARY KEY,
   ParticipantName VARCHAR(100),
   ParticipantType VARCHAR(50),
   EventID INT,
   Location VARCHAR(100), --     Location added here
   FOREIGN KEY (EventID) REFERENCES AdoptionEvents(EventID)
);

INSERT INTO Shelters VALUES
(1, 'Happy Tails Shelter', 'Chennai'),
(2, 'Paws and Claws', 'Bangalore'),
(3, 'Furry Friends', 'Mumbai');


INSERT INTO Users VALUES
(1, 'Srinidhi'),
(2, 'Mickey'),
(3, 'Nidhi'),
(4, 'Sri');

INSERT INTO Pets VALUES
(1, 'Alpha', 2, 'Labrador', 'Dog', 1, NULL, 1),
(2, 'Genny', 5, 'Beagle', 'Dog', 1, NULL, 1),
(3, 'Luna', 1, 'Persian', 'Cat', 0, 1, 2),
(4, 'Melon', 6, 'Bulldog', 'Dog', 1, NULL, 2),
(5, 'Luca', 3, 'Pug', 'Dog', 0, 2, 2),
```

```
(6, 'Coco', 2, 'Maine Coon', 'Cat', 1, NULL, 3),
(7, 'Bunny', 7, 'Golden Retriever', 'Dog', 1, NULL, 3),
(8, 'Angle', 4, 'Siamese', 'Cat', 0, 3, 1),
(9, 'Rocky', 8, 'German Shepherd', 'Dog', 1, NULL, 1),
(10, 'Panda', 2, 'Persian', 'Cat', 0, 4, 2);

INSERT INTO Donations VALUES
(1, 'Ram', 'Cash', 5000.00, NULL, '2024-01-15', 1),
(2, 'Karun', 'Item', NULL, 'Pet Food', '2024-02-20', 1),
(3, 'Angelin', 'Cash', 3000.00, NULL, '2024-01-05', 2),
(4, 'Jokiee', 'Item', NULL, 'Blankets', '2024-03-01', 3);

INSERT INTO AdoptionEvents VALUES
(1, 'Adoptathon', '2024-03-10 10:00:00', 'Chennai'),
(2, 'Paws Fest', '2024-04-05 15:00:00', 'Bangalore');

INSERT INTO Participants VALUES
(1, 'Happy Tails Shelter', 'Shelter', 1, 'Chennai'),
(2, 'Srinidhi', 'Adopter', 1, 'Chennai'),
(3, 'Mickey', 'Adopter', 2, 'Bangalore');

DELETE FROM Shelters;

INSERT INTO Shelters VALUES
(1, 'Happy Tails Shelter', 'Chennai'),
(2, 'Paws and Claws', 'Bangalore'),
(3, 'Furry Friends', 'Mumbai');
```

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

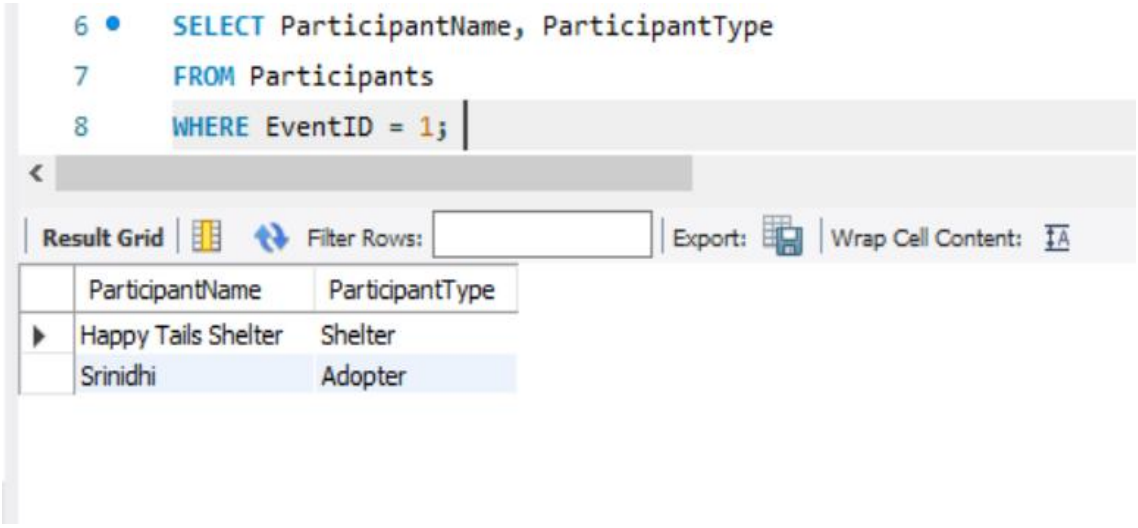


**TASK 5 :Write an SQL query that retrieves a list of available pets (those marked as available for adoption) from the "Pets" table. Include the pet's name, age, breed, and type in the result set. Ensure that the query filters out pets that are not available for adoption.**



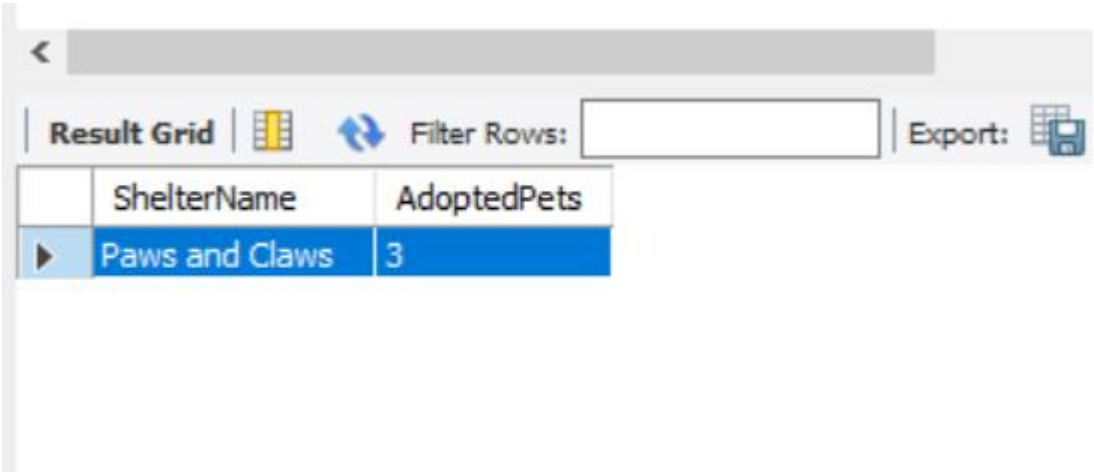

| Name | Age | Breed | Type |
|---|---|---|---|
| Alpha | 2 | Labrador | Dog |
| Genny | 5 | Beagle | Dog |
| Melon | 6 | Bulldog | Dog |
| Coco | 2 | Maine Coon | Cat |
| Bunny | 7 | Golden Retriever | Dog |
| Rocky | 8 | German Shepherd | Dog |

**TASK 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.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.**



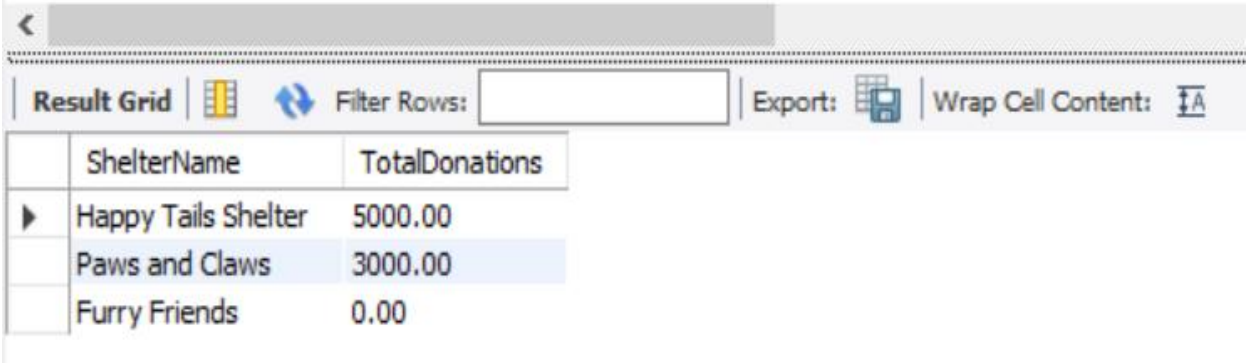

| ParticipantName | ParticipantType |
|---|---|
| Happy Tails Shelter | Shelter |
| Srinidhi | Adopter |

**TASK 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.**




**TASK 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.**

```
10 •   SELECT s.Name AS ShelterName,
11             IFNULL(SUM(d.DonationAmount), 0) AS TotalDonations
12     FROM Shelters s
13     LEFT JOIN Donations d ON s.ShelterID = d.ShelterID
14     GROUP BY s.ShelterID, s.Name;
15
16
```



**TASK 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.**

```
17 •  SELECT Name, Age, Breed, Type
18    FROM Pets
19    WHERE OwnerID IS NULL;
20
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

| Name | Age | Breed | Type |
|---|---|---|---|
| Alpha | 2 | Labrador | Dog |
| Genny | 5 | Beagle | Dog |
| Melon | 6 | Bulldog | Dog |
| Coco | 2 | Maine Coon | Cat |
| Bunny | 7 | Golden Retriever | Dog |
| Rocky | 8 | German Shepherd | Dog |

**TASK 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.**

```
32    ORDER BY
33        YEAR(DonationDate),
34        MONTH(DonationDate);
35
36
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

| Year | Month | MonthYear | TotalAmount |
|---|---|---|---|
| 2024 | 1 | January 2024 | 8000.00 |
| 2024 | 2 | February 2024 | 0.00 |
| 2024 | 3 | March 2024 | 0.00 |

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

```
36 •  SELECT DISTINCT Breed
37    FROM Pets
38    WHERE (Age BETWEEN 1 AND 3) OR Age > 5;
39
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

| Breed |
|---|
| Bulldog |
| German Shepherd |
| Golden Retriever |
| Labrador |
| Maine Coon |
| Persian |
| Pug |

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

```
41 •  SELECT p.Name AS PetName, s.Name AS ShelterName
42    FROM Pets p
43    JOIN Shelters s ON p.ShelterID = s.ShelterID
44    WHERE p.AvailableForAdoption = 1;
45
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

| PetName | ShelterName |
|---|---|
| Alpha | Happy Tails Shelter |
| Genny | Happy Tails Shelter |
| Rocky | Happy Tails Shelter |
| Melon | Paws and Claws |
| Coco | Furry Friends |
| Bunny | Furry Friends |

**TASK 13:Find the total number of participants in events organized by shelters located in specific city. Example: City=Chennai**

```
47      FROM Participants p
48      JOIN AdoptionEvents e ON p.EventID = e.EventID
49      JOIN Shelters s ON s.Name = p.ParticipantName AND p.ParticipantType = 'Shelter'
50      WHERE e.Location = 'Chennai';
51
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

| TotalParticipants |
|---|
| 1 |

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

```
53 •    SELECT DISTINCT Breed
54      FROM Pets
55      WHERE Age BETWEEN 1 AND 5;
56
```

Result Grid | Filter Rows: | Export: | Wrap Cell Conten

| Breed |
|---|
| Labrador |
| Beagle |
| Persian |
| Pug |
| Maine Coon |
| Siamese |

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

```
57 •    SELECT *
58      FROM Pets
59      WHERE AvailableForAdoption = 1;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

| PetID | Name | Age | Breed | Type | AvailableForAdoption | OwnerID | ShelterID |
|---|---|---|---|---|---|---|---|
| 1 | Alpha | 2 | Labrador | Dog | 1 | NULL | 1 |
| 2 | Genny | 5 | Beagle | Dog | 1 | NULL | 1 |
| 4 | Melon | 6 | Bulldog | Dog | 1 | NULL | 2 |
| 6 | Coco | 2 | Maine Coon | Cat | 1 | NULL | 3 |
| 7 | Bunny | 7 | Golden Retriever | Dog | 1 | NULL | 3 |
| 9 | Rocky | 8 | German Shepherd | Dog | 1 | NULL | 1 |
| NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL |

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

```
61 •  SELECT p.Name AS PetName, u.Name AS AdopterName
62    FROM Pets p
63    JOIN Users u ON p.OwnerID = u.UserID
64    WHERE p.OwnerID IS NOT NULL;
65
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

| PetName | AdopterName |
|---|---|
| Luna | Srinidhi |
| Luca | Mickey |
| Angle | Nidhi |
| Panda | Sri |

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

```
66 •  SELECT s.Name AS ShelterName, COUNT(p.PetID) AS AvailablePets
67    FROM Shelters s
68    LEFT JOIN Pets p ON s.ShelterID = p.ShelterID AND p.AvailableForAdoption = 1
69    GROUP BY s.ShelterID, s.Name;
70
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

| ShelterName | AvailablePets |
|---|---|
| Happy Tails Shelter | 3 |
| Paws and Claws | 1 |
| Furry Friends | 2 |

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

```
73    JOIN Pets p2 ON p1.ShelterID = p2.ShelterID
74                AND p1.Breed = p2.Breed
75                AND p1.PetID < p2.PetID
76    JOIN Shelters s ON p1.ShelterID = s.ShelterID;
77
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

| Pet1 | Pet2 | Breed | Shelter |
|---|---|---|---|
| Luna | Panda | Persian | Paws and Claws |

**TASK 19:List all possible combinations of shelters and adoption events.**

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

| ShelterName | EventName |
|---|---|
| Happy Tails Shelter | Paws Fest |
| Happy Tails Shelter | Adoptathon |
| Paws and Claws | Paws Fest |
| Paws and Claws | Adoptathon |
| Furry Friends | Paws Fest |
| Furry Friends | Adoptathon |

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

```
85      WHERE p.OwnerID IS NOT NULL
86      GROUP BY s.ShelterID, s.Name
87      ORDER BY AdoptedPets DESC
88      LIMIT 1;
89
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

| ShelterName | AdoptedPets |
|---|---|
| Paws and Claws | 3 |