Pet Pals, The PetAdoption Platform

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

```sql
CREATE DATABASE PetPals;
USE PetPals;
```

Output

Action Output

#	Time	Action	Message
1	19:17:49	CREATE DATABASE PetPals	1 row(s) affected
2	19:17:49	USE PetPals	0 row(s) affected

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

```sql
CREATE TABLE Pets ( PetID INT PRIMARY KEY,
    Name VARCHAR(255),
    Age INT,
    Breed VARCHAR(255),
    Type VARCHAR(255),
    AvailableForAdoption BIT );

CREATE TABLE Shelters ( ShelterID INT PRIMARY KEY,
    Name VARCHAR(255),
    Location VARCHAR(255) );

CREATE TABLE Donations ( DonationID INT PRIMARY KEY,
    DonorName VARCHAR(255),
    DonationType VARCHAR(255),
    DonationAmount DECIMAL,
    DonationItem VARCHAR(255),
    DonationDate DATETIME );

CREATE TABLE AdoptionEvents ( EventID INT PRIMARY KEY,
    EventName VARCHAR(255),
    EventDate DATETIME,
    Location VARCHAR(255) );

CREATE TABLE Participants ( ParticipantID INT PRIMARY KEY,
    ParticipantName VARCHAR(255),
    ParticipantType VARCHAR(255),
    EventID INT,
    FOREIGN KEY (EventID) REFERENCES AdoptionEvents(EventID) );
```

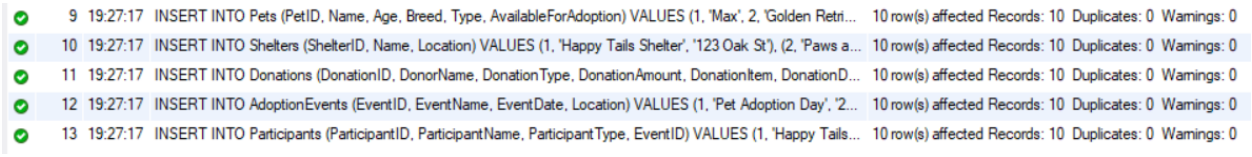
3. Define appropriate primary keys, foreign keys, and constraints.

   Already did it in 2nd Question

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

   Already did it in 2nd Question

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.



```
9  19:27:17  INSERT INTO Pets (PetID, Name, Age, Breed, Type, AvailableForAdoption) VALUES (1, 'Max', 2, 'Golden Retri...  10 row(s) affected Records: 10  Duplicates: 0  Warnings: 0
10 19:27:17  INSERT INTO Shelters (ShelterID, Name, Location) VALUES (1, 'Happy Tails Shelter', '123 Oak St'), (2, 'Paws a...  10 row(s) affected Records: 10  Duplicates: 0  Warnings: 0
11 19:27:17  INSERT INTO Donations (DonationID, DonorName, DonationType, DonationAmount, DonationItem, DonationD...  10 row(s) affected Records: 10  Duplicates: 0  Warnings: 0
12 19:27:17  INSERT INTO AdoptionEvents (EventID, EventName, EventDate, Location) VALUES (1, 'Pet Adoption Day', '2...  10 row(s) affected Records: 10  Duplicates: 0  Warnings: 0
13 19:27:17  INSERT INTO Participants (ParticipantID, ParticipantName, ParticipantType, EventID) VALUES (1, 'Happy Tails...  10 row(s) affected Records: 10  Duplicates: 0  Warnings: 0
```

```
1 •  SELECT Name, Age, Breed, Type
2    FROM Pets
3    WHERE AvailableForAdoption = 1;
4
```

Result Grid | Filter Rows: | Exp

Name	Age	Breed	Type
Max	2	Golden Retriever	Dog
Luna	1	Persian	Cat
Charlie	3	Beagle	Dog
Lucy	4	Dachshund	Dog
Oliver	1	Bengal	Cat
Leo	3	Maine Coon	Cat
Cooper	2	Poodle	Dog
Zoe	1	Ragdoll	Cat

```sql
INSERT INTO Pets (PetID, Name, Age, Breed, Type, AvailableForAdoption)
VALUES
(1, 'Max', 2, 'Golden Retriever', 'Dog', 1),
(2, 'Luna', 1, 'Persian', 'Cat', 1),
(3, 'Charlie', 3, 'Beagle', 'Dog', 1),
(4, 'Milo', 2, 'Siamese', 'Cat', 0),
(5, 'Lucy', 4, 'Dachshund', 'Dog', 1),
(6, 'Oliver', 1, 'Bengal', 'Cat', 1),
(7, 'Bella', 2, 'Labrador', 'Dog', 0),
(8, 'Leo', 3, 'Maine Coon', 'Cat', 1),
(9, 'Cooper', 2, 'Poodle', 'Dog', 1),
(10, 'Zoe', 1, 'Ragdoll', 'Cat', 1);


INSERT INTO Shelters (ShelterID, Name, Location)
VALUES
(1, 'Happy Tails Shelter', '123 Oak St'),
(2, 'Paws and Whiskers Rescue', '456 Pine St'),
(3, 'Heavenly Creatures Shelter', '789 Maple St'),
(4, 'Safe Haven Animal Rescue', '321 Elm St'),
(5, 'Rescue Me Now Shelter', '567 Birch St'),
(6, 'Forever Friends Sanctuary', '987 Cedar St'),
(7, 'Second Chance Pet Adoption', '654 Walnut St'),
(8, 'Furry Paws Haven', '876 Spruce St'),
(9, 'Purrfect Homes Shelter', '234 Sycamore St'),
(10, 'Canine Companions Rescue', '432 Chestnut St');


INSERT INTO Donations (DonationID, DonorName, DonationType, DonationAmount, DonationItem, DonationDate)
VALUES
(1, 'John Smith', 'Cash', 50.00, NULL, '2023-01-05'),
(2, 'Emily Davis', 'Item', NULL, 'Pet Toys', '2023-02-10'),
(3, 'Michael Johnson', 'Cash', 75.00, NULL, '2023-03-15'),
(4, 'Sophia Brown', 'Item', NULL, 'Pet Bed', '2023-04-20'),
(5, 'William Taylor', 'Cash', 100.00, NULL, '2023-05-25'),
(6, 'Ava Martinez', 'Item', NULL, 'Cat Litter', '2023-06-30'),
(7, 'James Anderson', 'Cash', 60.00, NULL, '2023-07-05'),
(8, 'Emma Garcia', 'Item', NULL, 'Dog Food', '2023-08-10'),
(9, 'Daniel Robinson', 'Cash', 90.00, NULL, '2023-09-15'),
(10, 'Olivia White', 'Item', NULL, 'Scratching Post', '2023-10-20');

INSERT INTO AdoptionEvents (EventID, EventName, EventDate, Location)
VALUES
(1, 'Pet Adoption Day', '2023-04-05 10:00:00', 'City Park'),
(2, 'Rescue Fest', '2023-05-15 12:00:00', 'Community Center'),
(3, 'Furry Friends Fair', '2023-06-20 14:00:00', 'Fairgrounds'),
(4, 'Paws in the Park', '2023-07-10 11:00:00', 'City Square'),
(5, 'Home for the Holidays', '2023-12-15 15:00:00', 'Shopping Mall'),
(6, 'Adopt-a-Pet Expo', '2023-08-25 13:00:00', 'Convention Center'),
(7, 'Fall Fur Festival', '2023-09-30 16:00:00', 'Park Pavilion'),
(8, 'Winter Whisker Wonderland', '2023-11-05 14:00:00', 'Winter Garden'),
(9, 'Spring Fling Adoption Fair', '2023-03-25 12:00:00', 'Botanical Gardens'),
(10, 'Summer Splash Adoption Event', '2023-07-30 15:00:00', 'Aquatic Center');

INSERT INTO Participants (ParticipantID, ParticipantName, ParticipantType, EventID)
VALUES
(1, 'Happy Tails Shelter', 'Shelter', 1),
(2, 'Emily Adams', 'Adopter', 1),
(3, 'Paws and Whiskers Rescue', 'Shelter', 2),
(4, 'Michael White', 'Adopter', 2),
(5, 'Heavenly Creatures Shelter', 'Shelter', 3),
(6, 'Sophia Garcia', 'Adopter', 3),
(7, 'Safe Haven Animal Rescue', 'Shelter', 4),
(8, 'William Martinez', 'Adopter', 4),
(9, 'Rescue Me Now Shelter', 'Shelter', 5),
(10, 'Ava Johnson', 'Adopter', 5);
```

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.

```
1
2 •  SELECT Participants.ParticipantName, Participants.ParticipantType
3    FROM Participants
4    JOIN AdoptionEvents ON Participants.EventID = AdoptionEvents.EventID
5    WHERE AdoptionEvents.EventID = 1;
6
```

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

ParticipantName	ParticipantType
Happy Tails Shelter	Shelter
Emily Adams	Adopter

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.

---

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.

```
1 •  SELECT S.Name AS ShelterName, SUM(D.DonationAmount)
2    AS TotalDonationAmount
3    FROM Shelters S LEFT JOIN Donations D ON S.ShelterID = D.ShelterID
4    GROUP BY S.Name;
```

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.

```sql
SELECT Name, Age, Breed, Type
FROM Pets
WHERE OwnerID IS NULL;
```

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 year.

```sql
SELECT
    DonationDateNew,
    COALESCE(SUM(DonationAmount), 0) AS TotalDonationAmount
FROM
(SELECT Donations.*,  DATE_FORMAT(DonationDate, '%Y-%M') as DonationDateNew FROM Donations) as Donations
GROUP BY DonationDateNew;
```

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

DonationDateNew	TotalDonationAmount
2023-January	50
2023-February	0
2023-March	75
2023-April	0
2023-May	100
2023-June	0
2023-July	60
2023-August	0
2023-September	90
2023-October	0

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

```sql
SELECT DISTINCT(Breed)
FROM PETS
WHERE Age BETWEEN 1 AND 3;
```

Result Grid | Filter Rows:

Breed
Golden Retriever
Persian
Beagle
Siamese
Bengal
Labrador
Maine Coon
Poodle
Ragdoll

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

```sql
SELECT P.Name AS PetName, P.Age, P.Breed, P.Type, S.Name AS ShelterName
FROM Pets P JOIN Shelters S ON P.ShelterID = S.ShelterID
WHERE P.AvailableForAdoption = 1;
```

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

```sql
SELECT COUNT(P.ParticipantID) AS TotalParticipants
FROM Participants P
JOIN AdoptionEvents AE ON P.EventID = AE.EventID
JOIN Shelters S ON AE.ShelterID = S.ShelterID
WHERE S.Location = 'Chennai';
```

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

```sql
SELECT DISTINCT(Breed)
FROM PETS
WHERE Age BETWEEN 1 AND 5;
```

	Breed
▶	Golden Retriever
	Persian
	Beagle
	Siamese
	Dachshund
	Bengal
	Labrador
	Maine Coon
	Poodle
	Ragdoll

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

```sql
SELECT PetID, Name, Age, Breed, Type, AvailableForAdoption
FROM Pets
WHERE AvailableForAdoption = 1 AND OwnerID IS NULL;
```

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

```sql
SELECT P.Name AS PetName, U.UserName AS AdopterName
FROM Adoptions A
JOIN Pets P ON A.PetID = P.PetID
JOIN Users U ON A.UserID = U.UserID;
```

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

```sql
SELECT
    S.ShelterID,
    S.Name AS ShelterName,
    S.Location,
    COUNT(P.PetID) AS AvailablePetsCount
FROM Shelters S
LEFT JOIN Pets P ON S.ShelterID = P.ShelterID
WHERE P.AvailableForAdoption = 1 OR P.AvailableForAdoption IS NULL
GROUP BY S.ShelterID, S.Name, S.Location;
```

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

```sql
SELECT A.PetID AS Pet1ID, A.Name AS Pet1Name, A.Breed AS SharedBreed, B.PetID AS Pet2ID, B.Name AS Pet2Name
FROM Pets A
JOIN Pets B ON A.Breed = B.Breed AND A.PetID < B.PetID;
```

	Pet1ID	Pet1Name	SharedBreed	Pet2ID	Pet2Name

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

```sql
SELECT S.ShelterID AS ShelterID, S.Name AS ShelterName, AE.EventID AS EventID, AE.EventName AS EventName
FROM Shelters S
CROSS JOIN AdoptionEvents AE;
```

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

```sql
SELECT S.ShelterID, S.Name AS ShelterName, COUNT(A.PetID) AS AdoptedPetsCount
FROM Shelters S LEFT JOIN Pets P ON S.ShelterID = P.ShelterID
LEFT JOIN Adoptions A ON P.PetID = A.PetID
GROUP BY S.ShelterID, S.Name
ORDER BY AdoptedPetsCount DESC LIMIT 1;
```