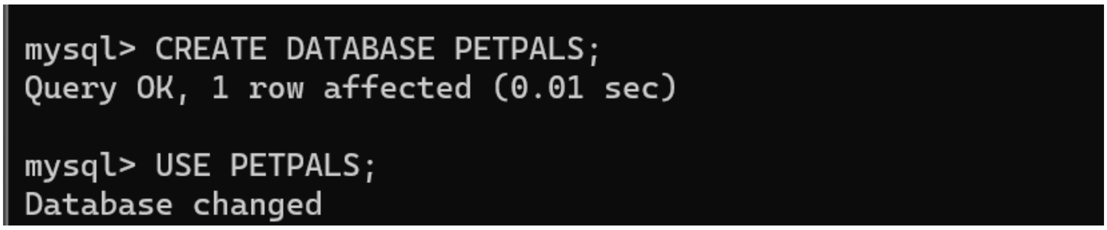
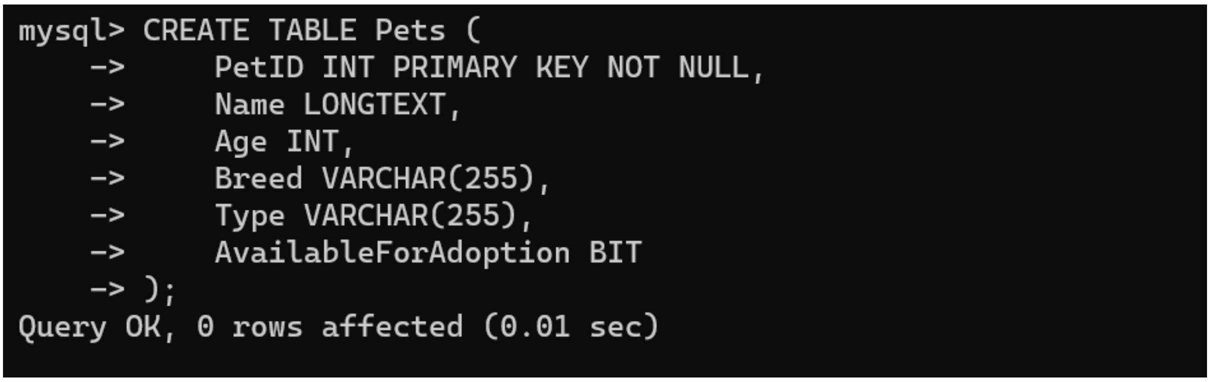
Problem Statement:

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

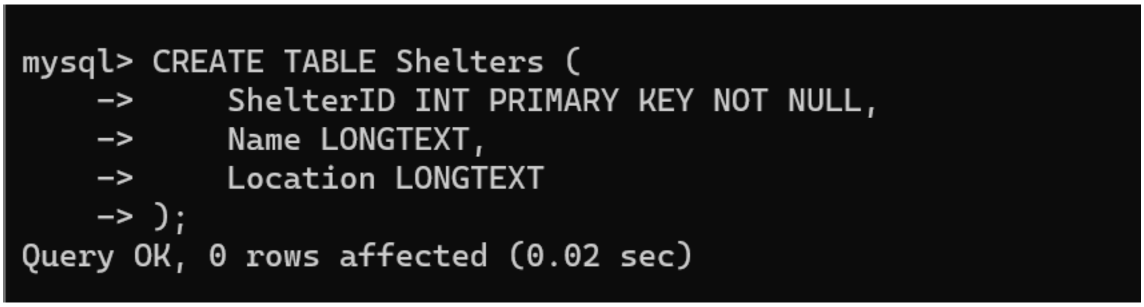


```
mysql> CREATE DATABASE PETPALS;
Query OK, 1 row affected (0.01 sec)

mysql> USE PETPALS;
Database changed
```

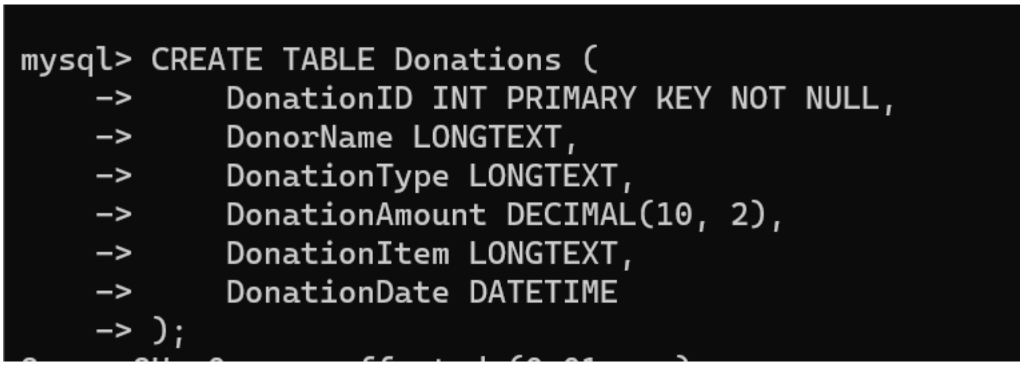
2. Create tables for pets, shelters, donations, adoption events, and participants.



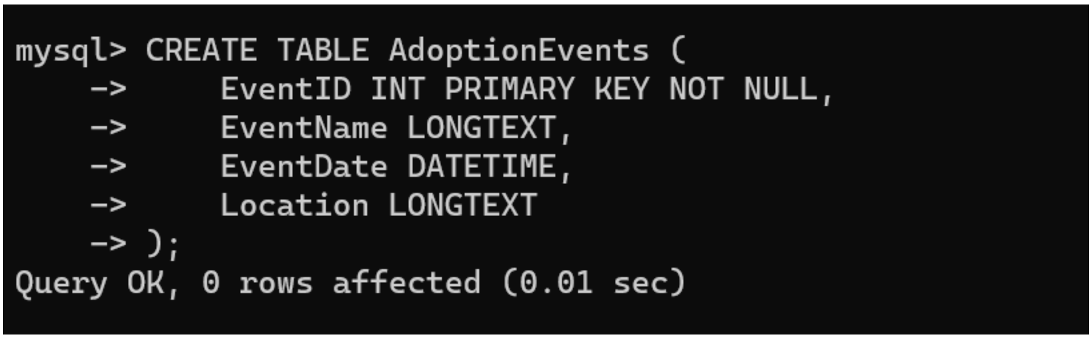
```
mysql> CREATE TABLE Pets (
    ->     PetID INT PRIMARY KEY NOT NULL,
    ->     Name LONGTEXT,
    ->     Age INT,
    ->     Breed VARCHAR(255),
    ->     Type VARCHAR(255),
    ->     AvailableForAdoption BIT
    -> );
Query OK, 0 rows affected (0.01 sec)
```



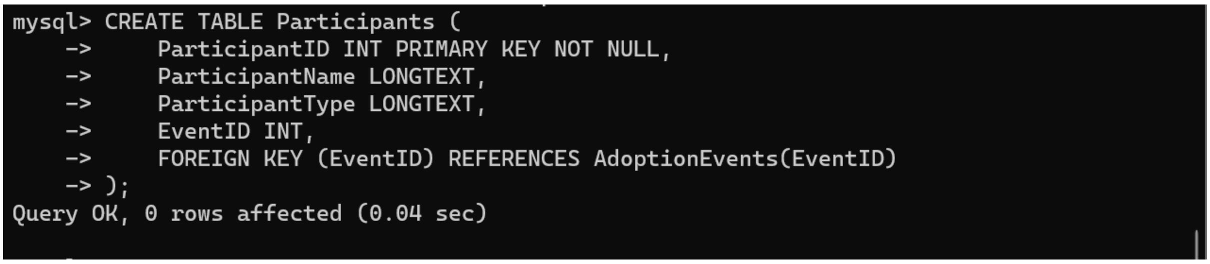
```
mysql> CREATE TABLE Shelters (
    ->     ShelterID INT PRIMARY KEY NOT NULL,
    ->     Name LONGTEXT,
    ->     Location LONGTEXT
    -> );
Query OK, 0 rows affected (0.02 sec)
```



```
mysql> CREATE TABLE Donations (
    ->     DonationID INT PRIMARY KEY NOT NULL,
    ->     DonorName LONGTEXT,
    ->     DonationType LONGTEXT,
    ->     DonationAmount DECIMAL(10, 2),
    ->     DonationItem LONGTEXT,
    ->     DonationDate DATETIME
    -> );
```



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



```
mysql> CREATE TABLE Participants (
    ->     ParticipantID INT PRIMARY KEY NOT NULL,
    ->     ParticipantName LONGTEXT,
    ->     ParticipantType LONGTEXT,
    ->     EventID INT,
    ->     FOREIGN KEY (EventID) REFERENCES AdoptionEvents(EventID)
    -> );
Query OK, 0 rows affected (0.04 sec)
```

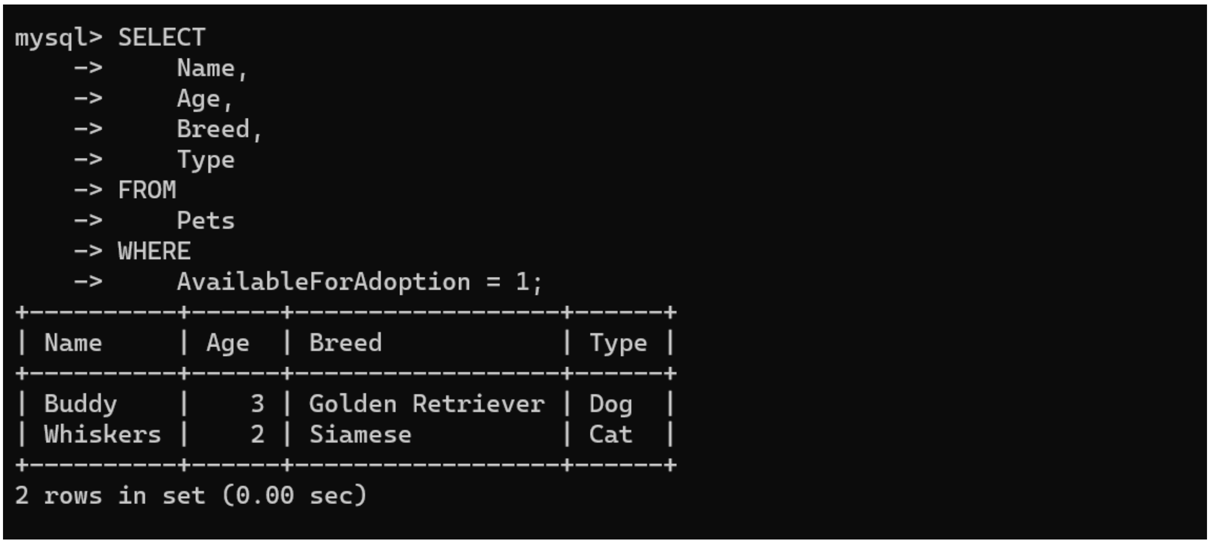
3. Define appropriate primary keys, foreign keys, and constraints.

DONE

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

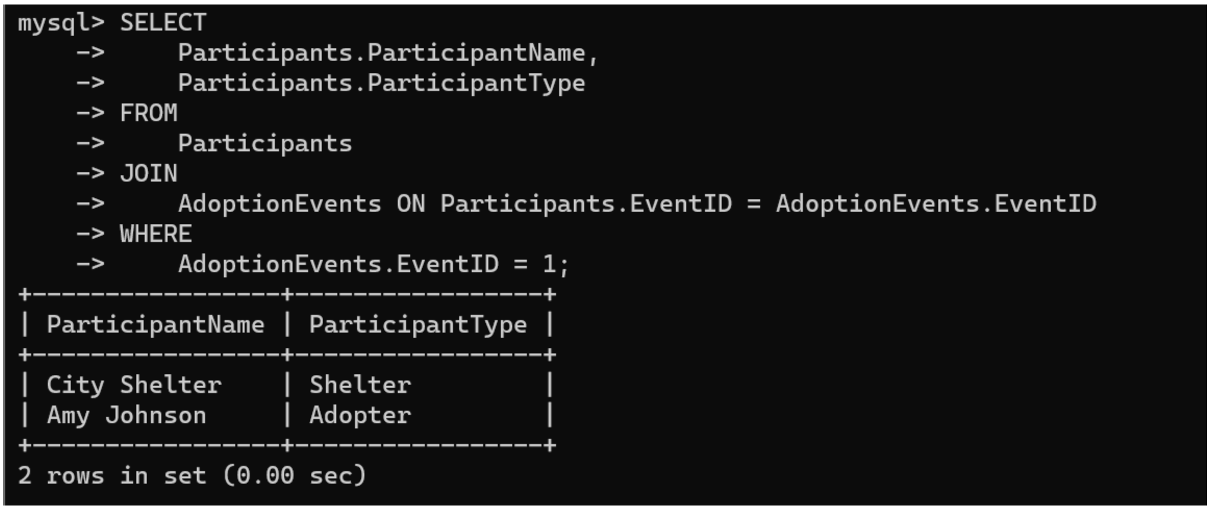
DONE

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.



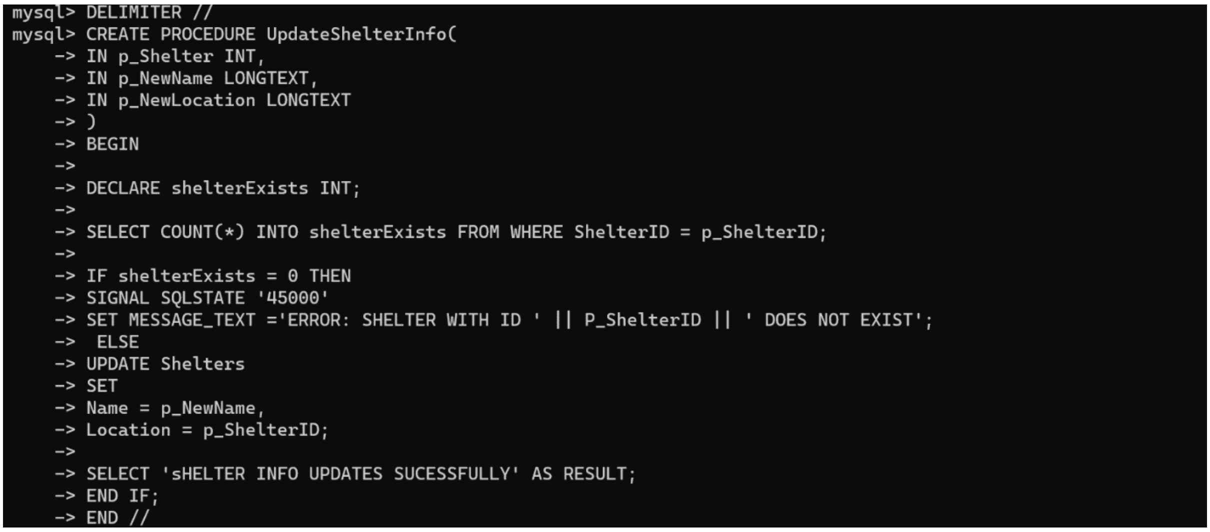
```
mysql> SELECT
    ->     Name,
    ->     Age,
    ->     Breed,
    ->     Type
    -> FROM
    ->     Pets
    -> WHERE
    ->     AvailableForAdoption = 1;
+----------+------+------------------+------+
| Name     | Age  | Breed            | Type |
+----------+------+------------------+------+
| Buddy    |    3 | Golden Retriever | Dog  |
| Whiskers |    2 | Siamese          | Cat  |
+----------+------+------------------+------+
2 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.



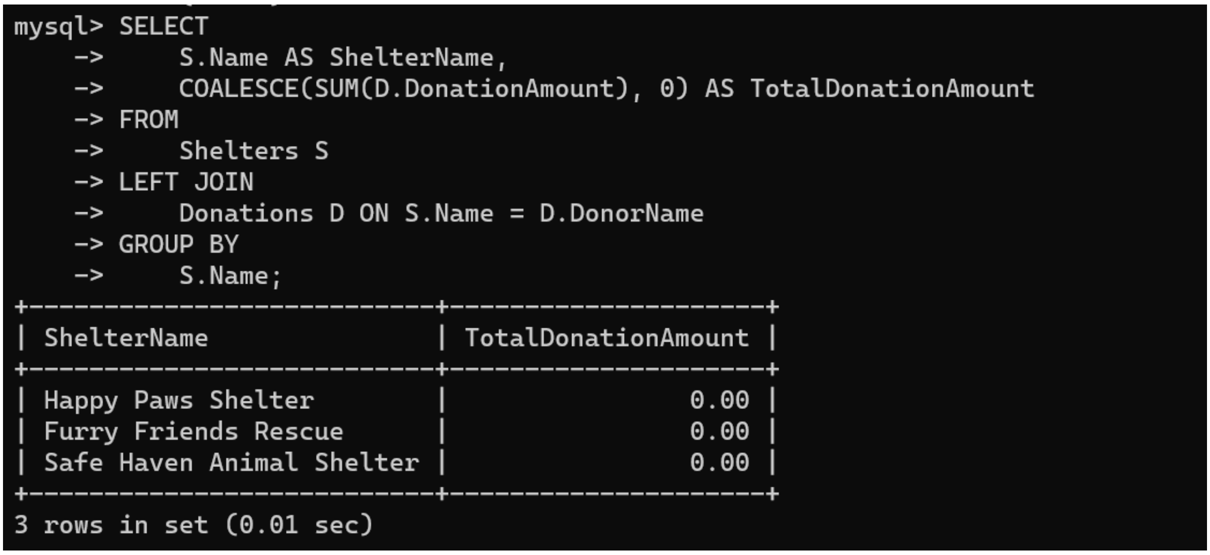
```
mysql> SELECT
    ->     Participants.ParticipantName,
    ->     Participants.ParticipantType
    -> FROM
    ->     Participants
    -> JOIN
    ->     AdoptionEvents ON Participants.EventID = AdoptionEvents.EventID
    -> WHERE
    ->     AdoptionEvents.EventID = 1;
+-----------------+-----------------+
| ParticipantName | ParticipantType |
+-----------------+-----------------+
| City Shelter    | Shelter         |
| Amy Johnson     | Adopter         |
+-----------------+-----------------+
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.



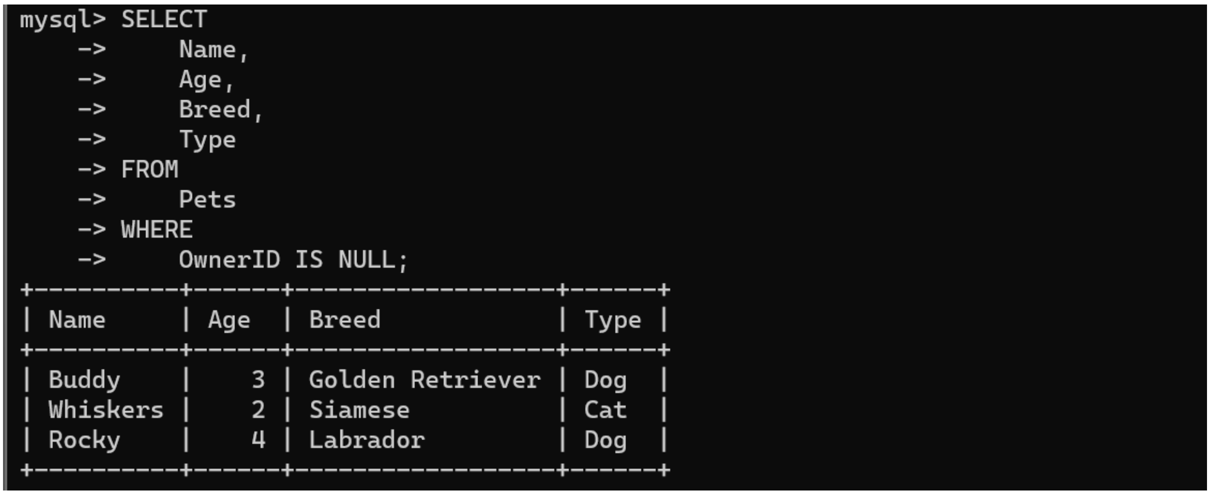
```
mysql> DELIMITER //
mysql> CREATE PROCEDURE UpdateShelterInfo(
    -> IN p_Shelter INT,
    -> IN p_NewName LONGTEXT,
    -> IN p_NewLocation LONGTEXT
    -> )
    -> BEGIN
    ->
    -> DECLARE shelterExists INT;
    ->
    -> SELECT COUNT(*) INTO shelterExists FROM WHERE ShelterID = p_ShelterID;
    ->
    -> IF shelterExists = 0 THEN
    -> SIGNAL SQLSTATE '45000'
    -> SET MESSAGE_TEXT ='ERROR: SHELTER WITH ID ' || P_ShelterID || ' DOES NOT EXIST';
    ->  ELSE
    -> UPDATE Shelters
    -> SET
    -> Name = p_NewName,
    -> Location = p_ShelterID;
    ->
    -> SELECT 'sHELTER INFO UPDATES SUCESSFULLY' AS RESULT;
    -> END IF;
    -> END //
```

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.



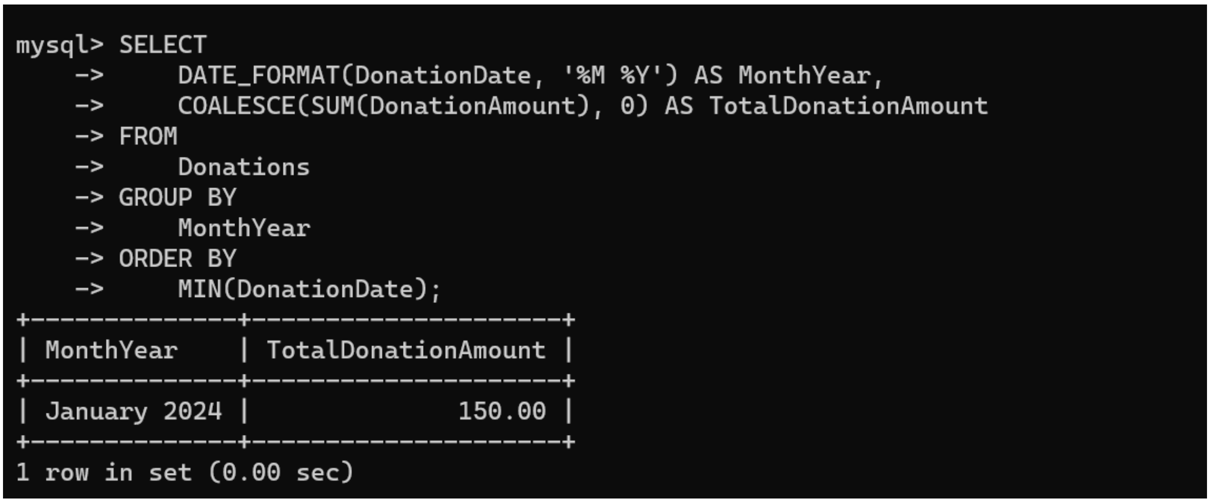
```
mysql> SELECT
    ->     S.Name AS ShelterName,
    ->     COALESCE(SUM(D.DonationAmount), 0) AS TotalDonationAmount
    -> FROM
    ->     Shelters S
    -> LEFT JOIN
    ->     Donations D ON S.Name = D.DonorName
    -> GROUP BY
    ->     S.Name;
+---------------------------+---------------------+
| ShelterName               | TotalDonationAmount |
+---------------------------+---------------------+
| Happy Paws Shelter        |                0.00 |
| Furry Friends Rescue      |                0.00 |
| Safe Haven Animal Shelter |                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.



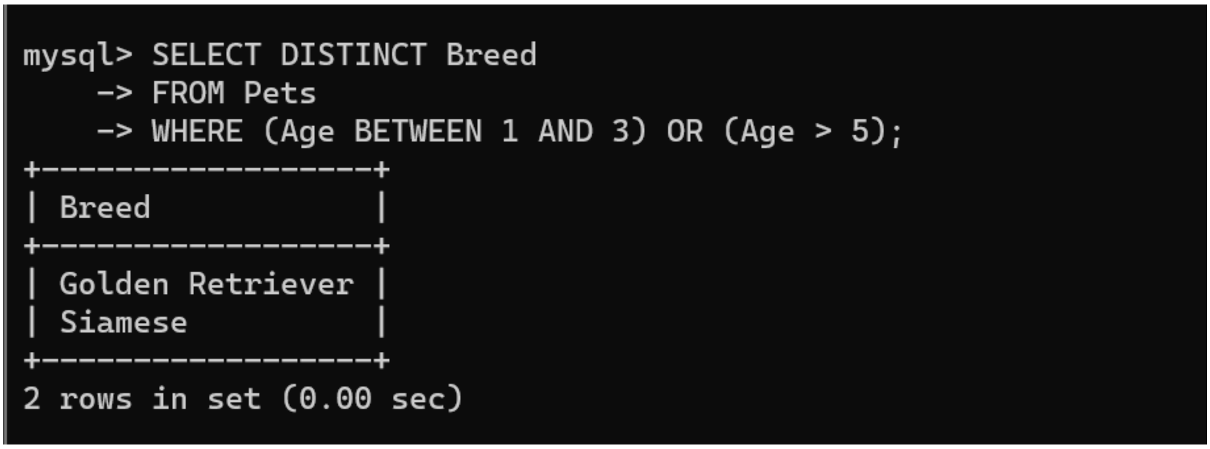
```
mysql> SELECT
    ->     Name,
    ->     Age,
    ->     Breed,
    ->     Type
    -> FROM
    ->     Pets
    -> WHERE
    ->     OwnerID IS NULL;
+----------+-----+------------------+------+
| Name     | Age | Breed            | Type |
+----------+-----+------------------+------+
| Buddy    |   3 | Golden Retriever | Dog  |
| Whiskers |   2 | Siamese          | Cat  |
| Rocky    |   4 | Labrador         | Dog  |
+----------+-----+------------------+------+
```

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.



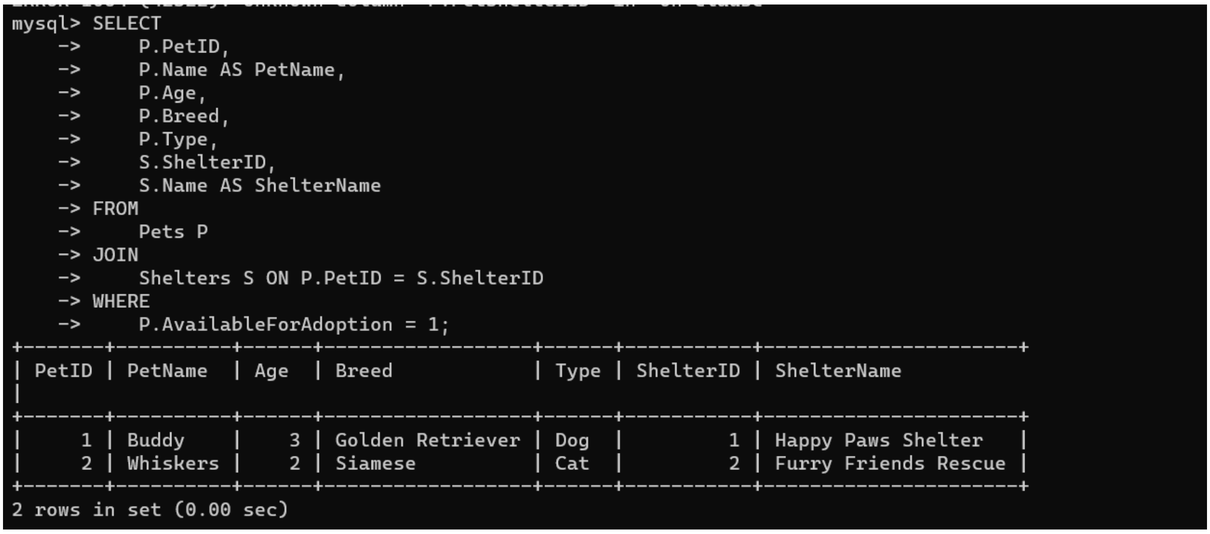
```
mysql> SELECT
    ->     DATE_FORMAT(DonationDate, '%M %Y') AS MonthYear,
    ->     COALESCE(SUM(DonationAmount), 0) AS TotalDonationAmount
    -> FROM
    ->     Donations
    -> GROUP BY
    ->     MonthYear
    -> ORDER BY
    ->     MIN(DonationDate);
+--------------+---------------------+
| MonthYear    | TotalDonationAmount |
+--------------+---------------------+
| January 2024 |              150.00 |
+--------------+---------------------+
1 row in set (0.00 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.



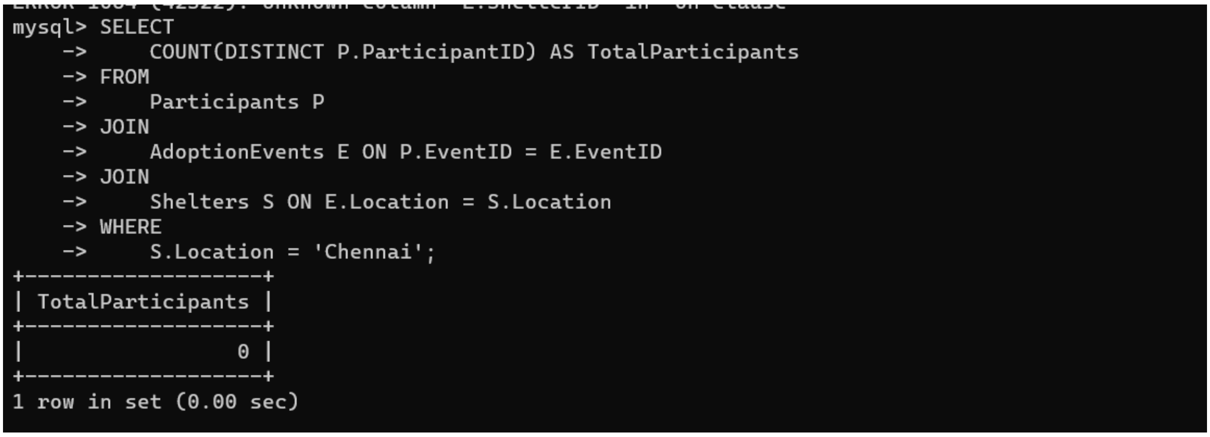
```
mysql> SELECT DISTINCT Breed
    -> FROM Pets
    -> WHERE (Age BETWEEN 1 AND 3) OR (Age > 5);
+------------------+
| Breed            |
+------------------+
| Golden Retriever |
| Siamese          |
+------------------+
2 rows in set (0.00 sec)
```

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



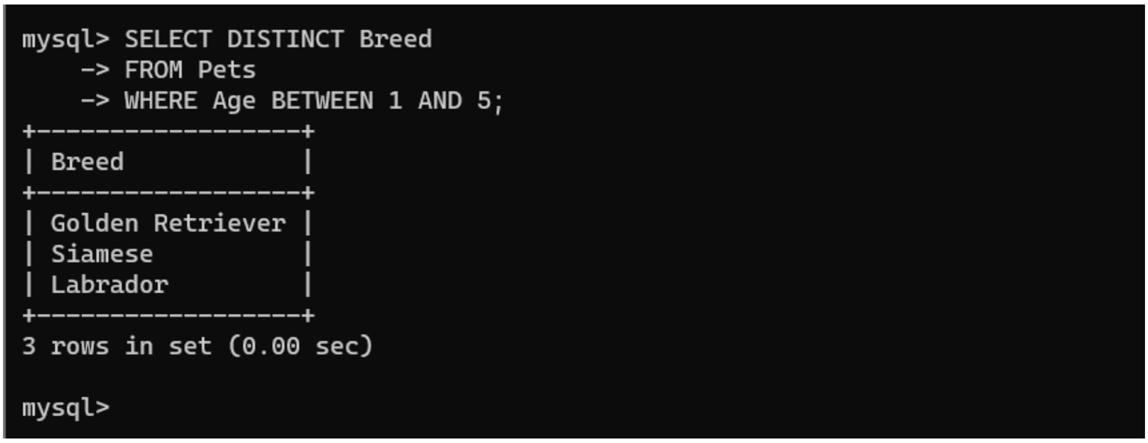
```
ERROR 1054 (42S22): Unknown column 'P.PetShelterID' in 'on clause'
mysql> SELECT
    ->     P.PetID,
    ->     P.Name AS PetName,
    ->     P.Age,
    ->     P.Breed,
    ->     P.Type,
    ->     S.ShelterID,
    ->     S.Name AS ShelterName
    -> FROM
    ->     Pets P
    -> JOIN
    ->     Shelters S ON P.PetID = S.ShelterID
    -> WHERE
    ->     P.AvailableForAdoption = 1;
+-------+----------+-----+------------------+------+-----------+----------------------+
| PetID | PetName  | Age | Breed            | Type | ShelterID | ShelterName
|
+-------+----------+-----+------------------+------+-----------+----------------------+
|     1 | Buddy    |   3 | Golden Retriever | Dog  |         1 | Happy Paws Shelter   |
|     2 | Whiskers |   2 | Siamese          | Cat  |         2 | Furry Friends Rescue |
+-------+----------+-----+------------------+------+-----------+----------------------+
2 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



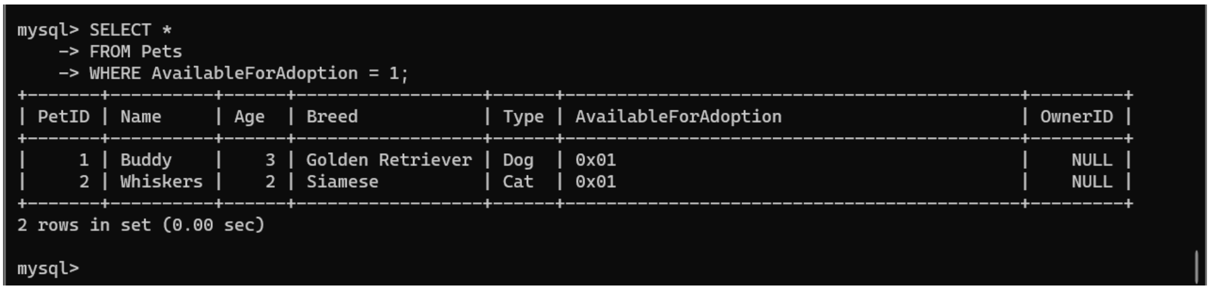
```
ERROR 1054 (42S22): Unknown column 'E.ShelterID' in 'on clause'
mysql> SELECT
    ->     COUNT(DISTINCT P.ParticipantID) AS TotalParticipants
    -> FROM
    ->     Participants P
    -> JOIN
    ->     AdoptionEvents E ON P.EventID = E.EventID
    -> JOIN
    ->     Shelters S ON E.Location = S.Location
    -> WHERE
    ->     S.Location = 'Chennai';
+-------------------+
| TotalParticipants |
+-------------------+
|                 0 |
+-------------------+
1 row in set (0.00 sec)
```

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



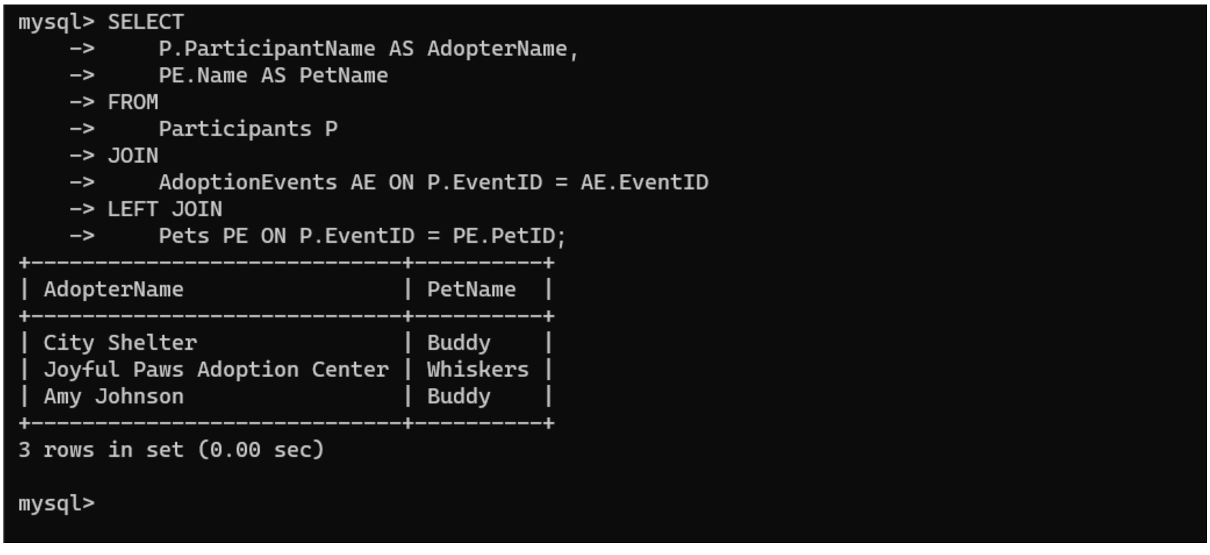
```
mysql> SELECT DISTINCT Breed
    -> FROM Pets
    -> WHERE Age BETWEEN 1 AND 5;
+------------------+
| Breed            |
+------------------+
| Golden Retriever |
| Siamese          |
| Labrador         |
+------------------+
3 rows in set (0.00 sec)

mysql>
```

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



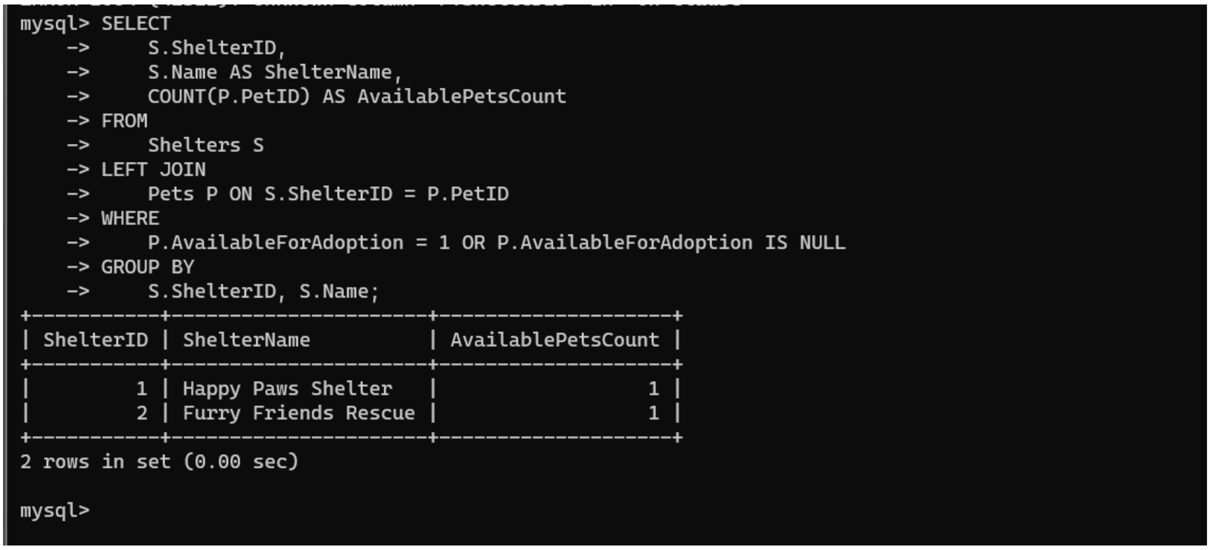
```
mysql> SELECT *
    -> FROM Pets
    -> WHERE AvailableForAdoption = 1;
+-------+----------+------+------------------+------+--------------------------------------------+---------+
| PetID | Name     | Age  | Breed            | Type | AvailableForAdoption                       | OwnerID |
+-------+----------+------+------------------+------+--------------------------------------------+---------+
|     1 | Buddy    |    3 | Golden Retriever | Dog  | 0x01                                       |    NULL |
|     2 | Whiskers |    2 | Siamese          | Cat  | 0x01                                       |    NULL |
+-------+----------+------+------------------+------+--------------------------------------------+---------+
2 rows in set (0.00 sec)

mysql>
```

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



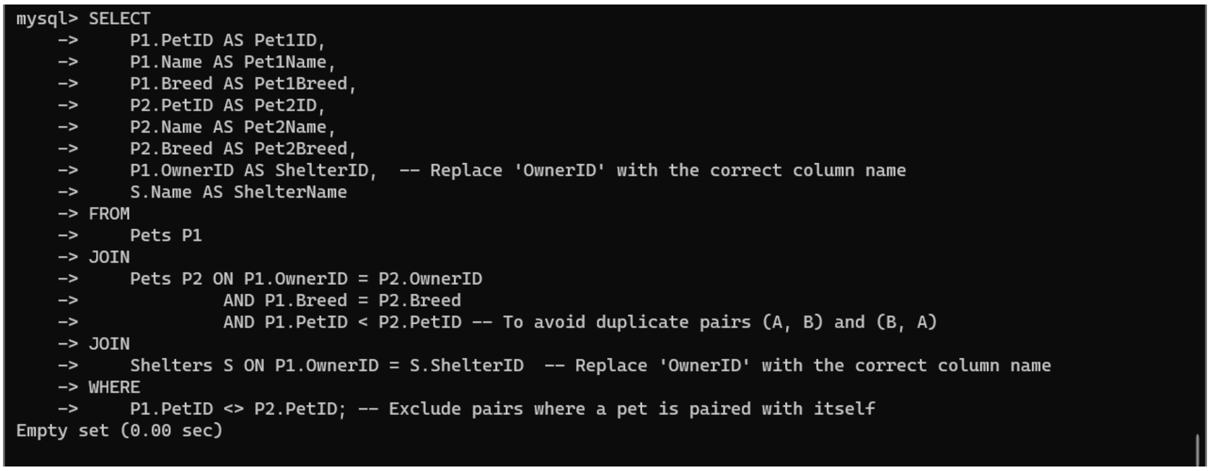
```
mysql> SELECT
    ->     P.ParticipantName AS AdopterName,
    ->     PE.Name AS PetName
    -> FROM
    ->     Participants P
    -> JOIN
    ->     AdoptionEvents AE ON P.EventID = AE.EventID
    -> LEFT JOIN
    ->     Pets PE ON P.EventID = PE.PetID;
+-----------------------------+----------+
| AdopterName                 | PetName  |
+-----------------------------+----------+
| City Shelter                | Buddy    |
| Joyful Paws Adoption Center | Whiskers |
| Amy Johnson                 | Buddy    |
+-----------------------------+----------+
3 rows in set (0.00 sec)

mysql>
```

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



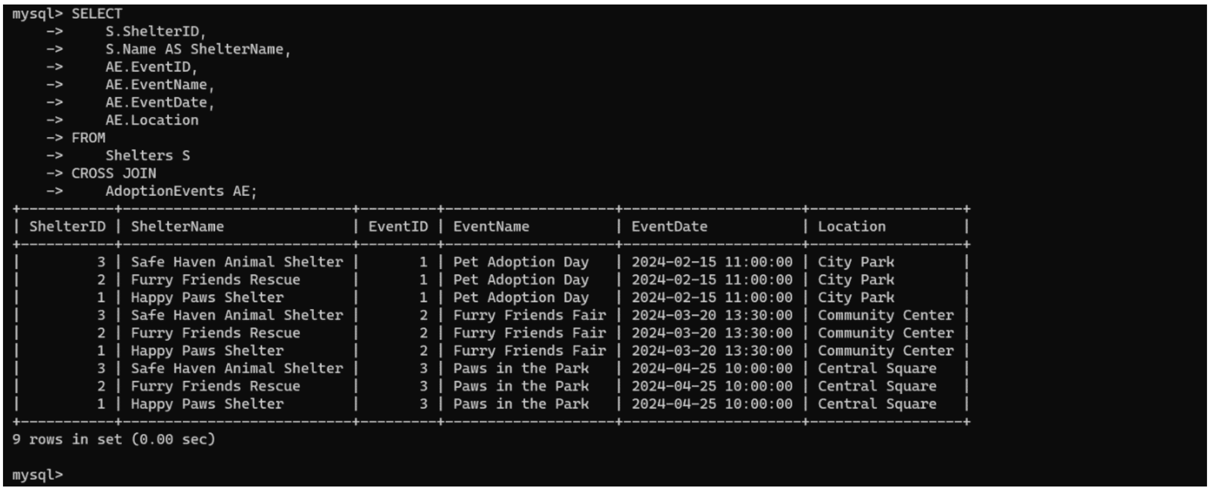
```
mysql> SELECT
    ->     S.ShelterID,
    ->     S.Name AS ShelterName,
    ->     COUNT(P.PetID) AS AvailablePetsCount
    -> FROM
    ->     Shelters S
    -> LEFT JOIN
    ->     Pets P ON S.ShelterID = P.PetID
    -> WHERE
    ->     P.AvailableForAdoption = 1 OR P.AvailableForAdoption IS NULL
    -> GROUP BY
    ->     S.ShelterID, S.Name;
+-----------+---------------------+--------------------+
| ShelterID | ShelterName         | AvailablePetsCount |
+-----------+---------------------+--------------------+
|         1 | Happy Paws Shelter  |                  1 |
|         2 | Furry Friends Rescue |                 1 |
+-----------+---------------------+--------------------+
2 rows in set (0.00 sec)

mysql>
```

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



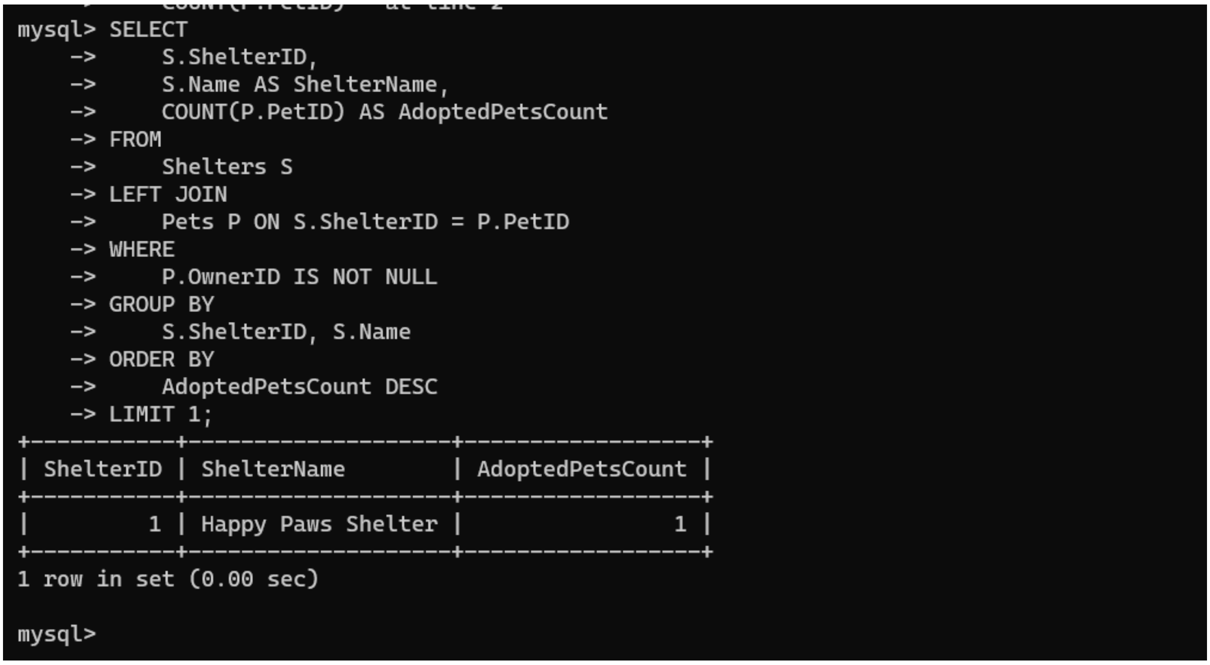
```
mysql> SELECT
    ->     P1.PetID AS Pet1ID,
    ->     P1.Name AS Pet1Name,
    ->     P1.Breed AS Pet1Breed,
    ->     P2.PetID AS Pet2ID,
    ->     P2.Name AS Pet2Name,
    ->     P2.Breed AS Pet2Breed,
    ->     P1.OwnerID AS ShelterID,  -- Replace 'OwnerID' with the correct column name
    ->     S.Name AS ShelterName
    -> FROM
    ->     Pets P1
    -> JOIN
    ->     Pets P2 ON P1.OwnerID = P2.OwnerID
    ->             AND P1.Breed = P2.Breed
    ->             AND P1.PetID < P2.PetID -- To avoid duplicate pairs (A, B) and (B, A)
    -> JOIN
    ->     Shelters S ON P1.OwnerID = S.ShelterID  -- Replace 'OwnerID' with the correct column name
    -> WHERE
    ->     P1.PetID <> P2.PetID; -- Exclude pairs where a pet is paired with itself
Empty set (0.00 sec)
```

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



```
mysql> SELECT
    ->     S.ShelterID,
    ->     S.Name AS ShelterName,
    ->     AE.EventID,
    ->     AE.EventName,
    ->     AE.EventDate,
    ->     AE.Location
    -> FROM
    ->     Shelters S
    -> CROSS JOIN
    ->     AdoptionEvents AE;
+-----------+-----------------------------+---------+--------------------+---------------------+------------------+
| ShelterID | ShelterName                 | EventID | EventName          | EventDate           | Location         |
+-----------+-----------------------------+---------+--------------------+---------------------+------------------+
|         3 | Safe Haven Animal Shelter   |       1 | Pet Adoption Day   | 2024-02-15 11:00:00 | City Park        |
|         2 | Furry Friends Rescue        |       1 | Pet Adoption Day   | 2024-02-15 11:00:00 | City Park        |
|         1 | Happy Paws Shelter          |       1 | Pet Adoption Day   | 2024-02-15 11:00:00 | City Park        |
|         3 | Safe Haven Animal Shelter   |       2 | Furry Friends Fair | 2024-03-20 13:30:00 | Community Center |
|         2 | Furry Friends Rescue        |       2 | Furry Friends Fair | 2024-03-20 13:30:00 | Community Center |
|         1 | Happy Paws Shelter          |       2 | Furry Friends Fair | 2024-03-20 13:30:00 | Community Center |
|         3 | Safe Haven Animal Shelter   |       3 | Paws in the Park   | 2024-04-25 10:00:00 | Central Square   |
|         2 | Furry Friends Rescue        |       3 | Paws in the Park   | 2024-04-25 10:00:00 | Central Square   |
|         1 | Happy Paws Shelter          |       3 | Paws in the Park   | 2024-04-25 10:00:00 | Central Square   |
+-----------+-----------------------------+---------+--------------------+---------------------+------------------+
9 rows in set (0.00 sec)

mysql>
```

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



```
    ->     COUNT(P.PetID)' at line 2
mysql> SELECT
    ->     S.ShelterID,
    ->     S.Name AS ShelterName,
    ->     COUNT(P.PetID) AS AdoptedPetsCount
    -> FROM
    ->     Shelters S
    -> LEFT JOIN
    ->     Pets P ON S.ShelterID = P.PetID
    -> WHERE
    ->     P.OwnerID IS NOT NULL
    -> GROUP BY
    ->     S.ShelterID, S.Name
    -> ORDER BY
    ->     AdoptedPetsCount DESC
    -> LIMIT 1;
+-----------+--------------------+------------------+
| ShelterID | ShelterName        | AdoptedPetsCount |
+-----------+--------------------+------------------+
|         1 | Happy Paws Shelter |                1 |
+-----------+--------------------+------------------+
1 row in set (0.00 sec)

mysql>
```