```sql
DROP database if exists petpals;
CREATE database PetPALS;
use petpals;

-- Creation of tables
DROP table if exists pets;
Create table Pets (
    PetID Int PRIMARY KEY,
    Name VARCHAR(255),
    Age INT,
    Breed VARCHAR(255),
    Type VARCHAR(50),
    AvailableForAdoption BIT
);

DROP table if exists shelters;
Create table Shelters (
    ShelterID Int PRIMARY KEY,
    Name VARCHAR(255),
    Location VARCHAR(255)
);


DROP table if exists Donations;
Create Table Donations (
    DonationID Int PRIMARY KEY,
    DonorName VARCHAR(255),
    DonationType VARCHAR(50),
    DonationAmount DECIMAL(10, 2),
    DonationItem VARCHAR(255),
    DonationDate DATETIME
);

DROP table if exists AdoptionEvents;
Create table AdoptionEvents (
    EventID Int PRIMARY KEY,
    EventName VARCHAR(255),
    EventDate DATETIME,
    Location VARCHAR(255)
);

DROP table if exists Participants;
Create Table Participants (
    ParticipantID Int PRIMARY KEY,
    ParticipantName VARCHAR(255),
    ParticipantType VARCHAR(50),
    EventID INT,
    FOREIGN KEY (EventID) REFERENCES AdoptionEvents(EventID)
);

-- Insertions of records

Insert Into Pets (PetID, Name, Age, Breed, Type, AvailableForAdoption)
Values
(1, 'Bailey', 2, 'Labrador Retriever', 'Dog', 1),
(2, 'Misty', 1, 'Persian', 'Cat', 1),
(3, 'Max', 3, 'German Shepherd', 'Dog', 1),
(4, 'Luna', 1, 'Siamese', 'Cat', 1),
(5, 'Buddy', 2, 'Golden Retriever', 'Dog', 1),
(6, 'Smokey', 3, 'Siamese', 'Cat', 0),
(7, 'Coco', 4, 'German Shepherd', 'Dog', 1),
(8, 'Whiskers', 2, 'Maine Coon', 'Cat', 1),
(9, 'Rocky', 3, 'Labrador Retriever', 'Dog', 1),
(10, 'Ginger', 1, 'British Shorthair', 'Cat', 0);
```

```sql
Insert Into Shelters (ShelterID, Name, Location)
Values
(1, 'Haven Animal Shelter', 'Delhi'),
(2, 'Love and Care Pet Rescue', 'Mumbai'),
(3, 'Newt Shelter', 'Bangalore'),
(4, 'ABC Animal Shelter', 'Jaipur'),
(5, 'Tails Adoption Center', 'Chennai'),
(6, 'Angels Animal Sanctuary', 'Hyderabad'),
(7, 'Pet Paradise Shelter', 'Pune'),
(8, 'Furry Friends Adoption Agency', 'Chennai'),
(9, 'Safe Animal Shelter', 'Jaipur'),
(10, 'Paws for Hope Rescue', 'Hyderabad');

Insert Into Donations (DonationID, DonorName, DonationType, DonationAmount,
DonationItem, DonationDate)
Values
(1, 'Hari', 'Cash', 5000.00, NULL, '2024-04-25 10:30:00'),
(2, 'Allan', 'Item', NUll, 'Pet Food', '2024-04-26 11:45:00'),
(3, 'Rahul', 'Both', 7500.00, 'Toys', '2024-04-27 09:15:00'),
(4, 'Sneha', 'Both', 1000.00, 'Toys', '2024-04-28 14:20:00'),
(5, 'Amit', 'Cash', 3000.00, NULL, '2024-04-29 16:50:00'),
(6, 'Priya', 'Cash', 6000.00, NULL, '2024-04-30 12:35:00'),
(7, 'Vikram', 'Item', NULL, 'Blankets', '2024-05-01 09:45:00'),
(8, 'Sai', 'Cash', 400.00, NULL, '2024-05-02 15:20:00'),
(9, 'Vishnu', 'Cash', 700.00, NULL, '2024-04-03 10:10:00'),
(10, 'Preeti', 'Both', 400.00, 'Leashes', '2024-04-04 13:55:00');

Insert Into AdoptionEvents (EventID, EventName, EventDate, Location)
Values
(1, 'Furry Friends Adoption Fair', '2024-05-10 12:00:00', 'Haven Animal
Shelter'),
(2, 'Paws for Love Adoption Drive', '2024-05-15 10:00:00', 'Love and Care Pet
Rescue'),
(3, 'Tails of Happiness Adoption Event', '2024-05-20 11:30:00', 'Newt Shelter'),
(4, 'Hearts for Paws Adoption Day', '2024-05-25 13:00:00', 'ABC Animal
Shelter'),
(5, 'Home for Every Paw Adoption Fair', '2024-05-30 09:00:00', 'Tails Adoption
Center'),
(6, 'Wagging Tails Adoption Camp', '2024-06-05 10:30:00', 'Angels Animal
Sanctuary'),
(7, 'Purrfect Homes Adoption Drive', '2024-06-10 12:00:00', 'Pet Paradise
Shelter'),
(8, 'Fur-ever Family Adoption Event', '2024-06-15 14:00:00', 'Furry Friends
Adoption Agency'),
(9, 'Safe Haven Adoption Festival', '2024-06-20 09:30:00', 'Safe Pet Refuge'),
(10, 'Paws in Need Adoption Fair', '2024-06-25 11:00:00', 'Paws for Hope
Rescue');

Insert Into Participants (ParticipantID, ParticipantName, ParticipantType,
EventID)
Values
(1, 'Aditi', 'Shelter', 1),
(2, 'Aryan', 'Adopter', 1),
(3, 'Neha', 'Shelter', 2),
(4, 'Kunal', 'Adopter', 2),
(5, 'Ishaan', 'Shelter', 3),
(6, 'Verma', 'Adopter', 3),
(7, 'Rohan', 'Shelter', 4),
(8, 'Divya', 'Adopter', 4),
(9, 'RiyaSharma', 'Shelter', 5),
(10, 'Manan Patel', 'Adopter', 5),
(11, 'Sanvi Singh', 'Shelter', 6),
(12, 'Parth', 'Adopter', 6),
```

```sql
(13, 'Aradhya Gupta', 'Shelter', 7),
(14, 'Aarav', 'Adopter', 7),
(15, 'Saumya', 'Shelter', 8),
(16, 'Aanya', 'Adopter', 8),
(17, 'Kabir', 'Shelter', 9),
(18, 'Ishika Patel', 'Adopter', 9),
(19, 'Vihaan Sharma', 'Shelter', 10),
(20, 'Ananya', 'Adopter', 10);


select * from pets;
select * from shelters;
select * from Donations;
select * from AdoptionEvents;
select * from Participants;

-- TASKS
-- 1. Provide a SQL script that initializes the database for the Pet Adoption
Platform â PetPalsâ .
    -- created
-- 2. Create tables for pets, shelters, donations, adoption events, and
participants.
        -- created
-- 3. Define appropriate primary keys, foreign keys, and constraints.
        -- created
-- 4. Ensure the script handles potential errors, such as if the database or
tables already exist.
        -- cretaed

-- 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.
select Name, Age, Breed, Type from Pets
where AvailableForAdoption = 1;

-- 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.
Select P.participantID , p.ParticipantName ,
p.participantType ,p.EventId ,ae.EventName,ae.eventDate,ae.Location
From Participants p inner join AdoptionEvents ae
on p.EventId = ae.EventId
where p.EventId = 4;


-- 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.
DELIMITER //

Create procedure UpdateShelterInfo (
    IN p_ShelterID INT,
    IN p_NewName VARCHAR(255),
    IN p_NewLocation VARCHAR(255)
)
begin
    declare v_RowCount INT;
    -- if exists
    select COUNT(*) into v_RowCount from Shelters where ShelterID = p_ShelterID;
    IF v_RowCount = 0 THEN
        SIGNAL SQLSTATE '45000'
```

```sql
            SET MESSAGE_TEXT = 'Invalid shelter ID';
    ELSE
        -- Update shelter information
        update Shelters
        SET Name = p_NewName, Location = p_NewLocation
        where ShelterID = p_ShelterID;
    END IF;
END//

DELIMITER ;



-- 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.
alter table donations add shelterID int;
update donations set shelterID=1 where donationID in (1,2,3,9);
update donations set shelterID=3 where donationID in(5,4,7,8);
update donations set shelterID=7  where donationID=6;
update donations set shelterID=9  where donationID=10;

select * from donations;
Select s.Name,
    SUM(d.DonationAmount) as TotalAmount
from Shelters s
inner join Donations d on s.ShelterID = d.ShelterID
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.
-- Write an SQL query to retrieve a list of electronic gadgets along with their
corresponding categories.
select * from pets;
alter table pets
add OwnerId int;
update pets
set Ownerid = case
    when petid = 1 then 10
    when petid = 2 then 11
    when petid = 3 then 12
    when petid = 4 then NULL
    when petid = 5 then 14
    when petid = 6 then 17
    when petid = 7 then 18
    when petid = 8 then 19
    when petid = 9 then 20
    when petid = 10 then NULL
    else NULL
end
where petid in (1,2,3,4,5,6,7,8,9,10);

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
```

```sql
month-year.
select
    DATE_FORMAT(DonationDate, '%Y-%m') AS YearMonth,
    SUM(DonationAmount) AS TotalDonationAmount
From Donations
group by DATE_FORMAT(DonationDate, '%Y-%m')
order by YearMonth;



-- 11. Retrieve a list of distinct breeds for all pets that are either aged
between 1 and 3 years or older than 5 years.
Select DISTINCT Breed
from Pets
where (Age BETWEEN 1 AND 3) OR (Age > 5);

-- 12. Retrieve a list of pets and their respective shelters where the pets are
currently available for adoption.
-- for joining two table there should be a reference in both table
-- but it does not contains ,so we are going to add shelters id in pets table
Alter Table Pets
add ShelterID INT,
add CONSTRAINT ShelterID FOREIGN KEY (ShelterID) REFERENCES Shelters(ShelterID);
select * from pets;

update pets
set ShelterID = case
    when petid = 1 then 1
    when petid = 2 then 2
    when petid = 3 then 3
    when petid = 4 then 4
    when petid = 5 then 5
    when petid = 6 then 6
    when petid = 7 then 7
    when petid = 8 then 8
    when petid = 9 then 9
    when petid = 10 then 10
    else NULL
end
where petid in (1,2,3,4,5,6,7,8,9,10);
Select
    p.Name AS PetName,
    p.Age AS PetAge,
    p.Breed AS PetBreed,
    p.Type AS PetType,
    s.Name AS ShelterName,
    s.Location AS ShelterLocation
from  Pets p inner 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 city. Example: City=Chennai
select * from shelters;
select * from participants;
select * from adoptionevents;

Select COUNT(*) as TotalParticipants from Participants p
inner join AdoptionEvents ae on p.EventID = ae.EventID
inner join Shelters s on p.ParticipantID = s.ShelterID
where s.Location = 'Chennai';


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

```
years.
select DISTINCT Breed from Pets
where Age BETWEEN 1 AND 5;


-- 15. Find the pets that have not been adopted by selecting their information
from the 'Pet' table.
select * from Pets
where OwnerID IS NULL;


-- 16. Retrieve the names of all adopted pets along with the adopter's name from
the 'Adoption' and 'User' tables.
-- we does not contains users and adoption table so we have to create it and
insert some values.


-- Create Users table
CREATE TABLE Users (
    UserID INT PRIMARY KEY,
    Name VARCHAR(255)
);


-- Create Adoption table
CREATE TABLE Adoption (
    AdoptionID INT PRIMARY KEY,
    UserID INT,
    PetID INT,
    FOREIGN KEY (UserID) REFERENCES Users(UserID),
    FOREIGN KEY (PetID) REFERENCES Pets(PetID)
);


INSERT INTO Users (UserID, Name) VALUES
(1, 'John Doe'),
(2, 'Jane Smith'),
(3, 'Michael Johnson'),
(4, 'Emily Johnson'),
(5, 'David Brown'),
(6, 'Emma Wilson'),
(7, 'Christopher Lee'),
(8, 'Taylor');

INSERT INTO Adoption (AdoptionID, UserID, PetID) VALUES
(1, 1, 1),
(2, 2, 2),
(3, 3, 3),
(4, 4, 4),
(5, 5, 5),
(6, 6, 6),
(7, 7, 7),
(8, 8, 8);

select p.Name AS PetName, u.Name AS AdopterName
from Adoption a
inner join Pets p ON a.PetID = p.PetID
inner 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.
select s.ShelterID,s.Name,s.Location,(
        select COUNT(*) from Pets p
        where p.ShelterID = s.ShelterID
        and p.AvailableForAdoption = 1
    ) as PetsAvailableForAdoption from Shelters s;
```

```
-- 18. Find pairs of pets from the same shelter that have the same breed.
select p1.PetID,p1.Name,p1.Breed,p2.PetID,p2.Name
from Pets p1,Pets p2
where p1.ShelterID = p2.ShelterID
    and p1.PetID < p2.PetID
    and p1.Breed = p2.Breed;


-- 19. List all possible combinations of shelters and adoption events.
select s.ShelterID,s.Name AS ShelterName,ae.EventID,ae.EventName
from Shelters s
cross join AdoptionEvents ae;


-- 20. Determine the shelter that has the highest number of adopted pets.
Select ShelterID,Name,
    (select COUNT(*) from Adoption
        where PetID in (select PetID from Pets where ShelterID = s.ShelterID)
    ) as AdoptedPetsCount
from Shelters s
order by AdoptedPetsCount DESC limit 1;
```