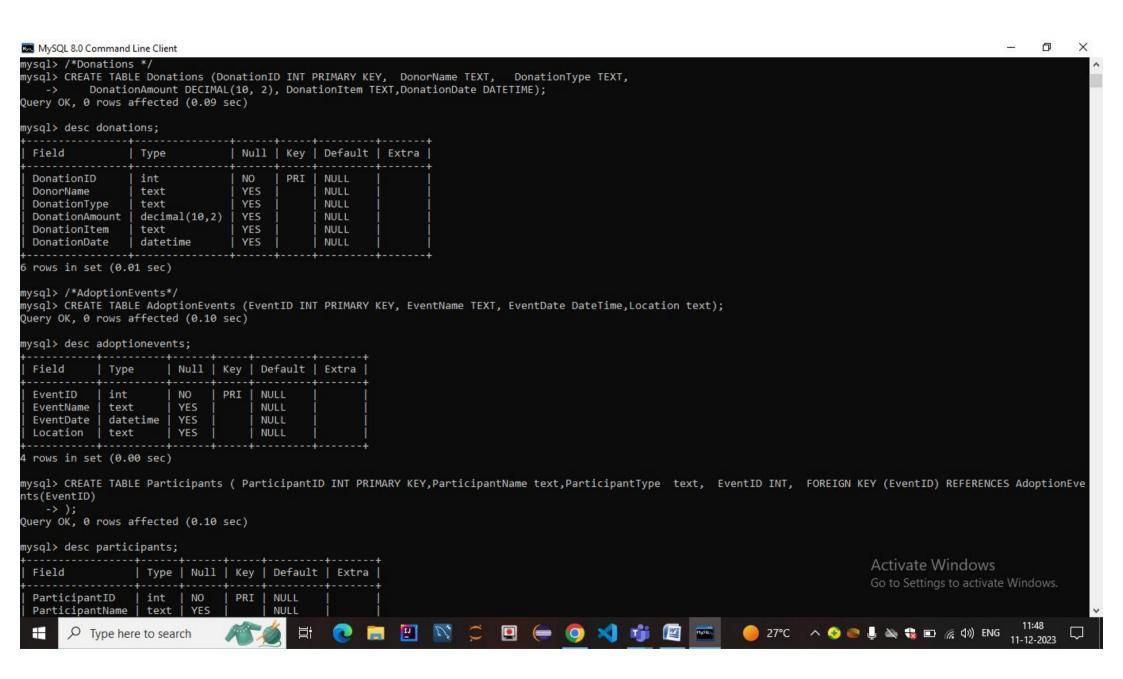MySQL 8.0 Command Line Client

```
mysql> /* CODING CHALLENGES */
mysql> /* PetPals, The Pet Adpotion Platform */
mysql> /*TASKS :
   /*> 1. Provide a SQL Script that initializes the database for the Pet Adoption Platform "PetPals"*/
mysql> Create database if not exists PetPals;
Query OK, 1 row affected (0.08 sec)

mysql> Use Petpals;
Database changed
mysql> /*2.Create tables for pets,shelters,donations,adoption events and participants*/
mysql> /*PETs*/
mysql> CREATE TABLE Pets( PetID INT PRIMARY KEY, Name TEXT, Age INT,Breed TEXT,Type text,AvailableForAdoption  BIT);
Query OK, 0 rows affected (0.23 sec)

mysql> desc pets;
+---------------------+--------+------+-----+---------+-------+
| Field               | Type   | Null | Key | Default | Extra |
+---------------------+--------+------+-----+---------+-------+
| PetID               | int    | NO   | PRI | NULL    |       |
| Name                | text   | YES  |     | NULL    |       |
| Age                 | int    | YES  |     | NULL    |       |
| Breed               | text   | YES  |     | NULL    |       |
| Type                | text   | YES  |     | NULL    |       |
| AvailableForAdoption | bit(1) | YES  |     | NULL    |       |
+---------------------+--------+------+-----+---------+-------+
6 rows in set (0.08 sec)

mysql> /*Shelters*/
mysql> CREATE TABLE Shelters(ShelterID INT PRIMARY KEY, Name TEXT,Location TEXT);
Query OK, 0 rows affected (0.04 sec)

mysql> desc Shelters;
+-----------+------+------+-----+---------+-------+
| Field     | Type | Null | Key | Default | Extra |
+-----------+------+------+-----+---------+-------+
| ShelterID | int  | NO   | PRI | NULL    |       |
| Name      | text | YES  |     | NULL    |       |
| Location  | text | YES  |     | NULL    |       |
+-----------+------+------+-----+---------+-------+
3 rows in set (0.00 sec)

mysql> _
```

```
mysql> /*Donations */
mysql> CREATE TABLE Donations (DonationID INT PRIMARY KEY,  DonorName TEXT,   DonationType TEXT,
    ->     DonationAmount DECIMAL(10, 2), DonationItem TEXT,DonationDate DATETIME);
Query OK, 0 rows affected (0.09 sec)

mysql> desc donations;
+----------------+---------------+------+-----+---------+-------+
| Field          | Type          | Null | Key | Default | Extra |
+----------------+---------------+------+-----+---------+-------+
| DonationID     | int           | NO   | PRI | NULL    |       |
| DonorName      | text          | YES  |     | NULL    |       |
| DonationType   | text          | YES  |     | NULL    |       |
| DonationAmount | decimal(10,2) | YES  |     | NULL    |       |
| DonationItem   | text          | YES  |     | NULL    |       |
| DonationDate   | datetime      | YES  |     | NULL    |       |
+----------------+---------------+------+-----+---------+-------+
6 rows in set (0.01 sec)

mysql> /*AdoptionEvents*/
mysql> CREATE TABLE AdoptionEvents (EventID INT PRIMARY KEY, EventName TEXT, EventDate DateTime,Location text);
Query OK, 0 rows affected (0.10 sec)

mysql> desc adoptionevents;
+-----------+----------+------+-----+---------+-------+
| Field     | Type     | Null | Key | Default | Extra |
+-----------+----------+------+-----+---------+-------+
| EventID   | int      | NO   | PRI | NULL    |       |
| EventName | text     | YES  |     | NULL    |       |
| EventDate | datetime | YES  |     | NULL    |       |
| Location  | text     | YES  |     | NULL    |       |
+-----------+----------+------+-----+---------+-------+
4 rows in set (0.00 sec)

mysql> CREATE TABLE Participants ( ParticipantID INT PRIMARY KEY,ParticipantName text,ParticipantType  text,  EventID INT,  FOREIGN KEY (EventID) REFERENCES AdoptionEve
nts(EventID)
    -> );
Query OK, 0 rows affected (0.10 sec)

mysql> desc participants;
+-----------------+------+------+-----+---------+-------+
| Field           | Type | Null | Key | Default | Extra |
+-----------------+------+------+-----+---------+-------+
| ParticipantID   | int  | NO   | PRI | NULL    |       |
| ParticipantName | text | YES  |     | NULL    |       |
```

```
MySQL 8.0 Command Line Client                                                    —    □    ×

mysql> /*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;
Empty set (0.12 sec)

mysql> /*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 p.ParticipantName, p.ParticipantType
    -> FROM Participants p
    -> JOIN AdoptionEvents ae ON p.EventID = ae.EventID
    -> WHERE ae.EventID = 101;
Empty set (0.05 sec)

mysql> /* 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>
mysql> CREATE PROCEDURE UpdateShelterInformation (
    ->     IN p_ShelterID INT,
    ->     IN p_NewName VARCHAR(255),
    ->     IN p_NewLocation VARCHAR(255)
    -> )
    -> BEGIN
    ->     DECLARE shelterCount INT;
    ->     SELECT COUNT(*) INTO shelterCount
    ->     FROM Shelters
    ->     WHERE ShelterID = p_ShelterID;
    ->
    ->     IF shelterCount > 0 THEN
    ->         UPDATE Shelters
    ->         SET Name = p_NewName, Location = p_NewLocation
    ->         WHERE ShelterID = p_ShelterID;
    ->
    ->         SELECT 'Shelter information updated successfully.' AS Result;
    ->     ELSE
    ->         SELECT 'Invalid Shelter ID. No updates performed.' AS Result;
    ->     END IF;
    -> END //
```

```
MySQL 8.0 Command Line Client                                                              —

affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> use petpals;
Database changed
mysql>
mysql> /*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
    ->      shelters.name,sum(Donations.donationAmount) as totaldonationamount
    ->      from donations
    ->      RIGHT JOIN shelters ON donations.donationId=shelters.shelterId
    ->      GROUP BY Shelters.Name;
Empty set (0.00 sec)

mysql> /*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 petid not in (SELECT petid from adoptionevents);
Empty set (0.01 sec)
```

MySQL 8.0 Command Line Client

```
mysql> /*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
    ->     DATE_FORMAT(DonationDate, '%M %Y');
Empty set (0.00 sec)

mysql> /*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);
Empty set (0.00 sec)
```

```
mysql> /*12. Retrieve a list of pets and their respective shelters where the pets are currently available for
    /*> adoption*/
mysql> SELECT P.Name AS PetName, P.Type AS PetType, P.Breed, S.Name AS ShelterName
    -> FROM Pets P
    -> JOIN Shelters S ON P.ShelterID = S.ShelterID
    -> WHERE P.AvailableForAdoption = 1;
ERROR 1054 (42S22): Unknown column 'P.ShelterID' in 'on clause'
mysql>
mysql> /*12. Retrieve a list of pets and their respective shelters where the pets are currently available for
    /*> adoption*/
mysql> SELECT Pets.*,Shelters.Name as ShelterName
    -> FROM Pets
    -> JOIN Shelters ON Pets.PetID=Shelters.ShelterID
    -> WHERE AvailableForAdoption=1;
Empty set (0.00 sec)

mysql> /*13. Find the total number of participants in events organized by shelters located in specific city.
    /*> Example: City=Chennai*/
mysql> SELECT COUNT(DISTINCT P.ParticipantID) AS TotalParticipants
    -> FROM Participants P
    -> JOIN Shelters S ON P.ParticipantID = S.ShelterID
    -> JOIN AdoptionEvents AE ON P.EventID = AE.EventID
    -> WHERE S.Location = 'Chennai' AND P.ParticipantType = 'Shelter';
+-------------------+
| TotalParticipants |
+-------------------+
|                 0 |
+-------------------+
1 row in set (0.01 sec)

mysql> /*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;
Empty 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;
Empty 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>
mysql> /*17. Retrieve a list of all shelters along with the count of pets currently available for adoption in each
    /*> shelter*/
mysql> SELECT
    ->     S.Name AS ShelterName,
    ->     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.Name;
ERROR 1054 (42S22): Unknown column 'P.ShelterID' in 'on clause'
mysql>
mysql>
mysql> /*17. Retrieve a list of all shelters along with the count of pets currently available for adoption in each
    /*> shelter*/
mysql> SELECT
    ->     S.Name AS ShelterName,
    ->     COUNT(P.PetID) AS AvailablePetsCount
    -> FROM
    ->     Shelters S
    -> LEFT JOIN
    ->     Pets P ON S.ShelterID = s.ShelterID
    -> WHERE
    ->     AvailableForAdoption = 1
    -> GROUP BY
    ->     S.shelterid;
Empty 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,
    ->     S.Name AS ShelterName
    -> FROM
    ->     Pets P1
    -> JOIN
    ->     Pets P2 ON P1.Breed = P2.Breed AND P1.PetID < P2.PetID
    -> JOIN
    ->     Shelters S ON P1.ShelterID = S.ShelterID
    -> WHERE
```

```
MySQL 8.0 Command Line Client                                                    —    ☐

mysql>
mysql>
mysql> /*19. List all possible combinations of shelters and adoption events.*/
mysql> SELECT Shelters.Name as shelterName,AdoptionEvents.EventName
    -> FROM Shelters
    -> CROSS JOIN AdoptionEvents;
Empty set (0.00 sec)

mysql> /*20.Determine the shelter that has the highest number of adopted pets*/
mysql> SELECT Shelters.name as sheltername,count(*) as AdoptedPetsCount
    -> FROM Shelters
    -> JOIN Pets ON Shelters.shelterID=Pets.shelterID
    -> JOIN AdoptionEvents a ON pets.petid=AdoptionEvents.petid
    -> GROUP BY Shelters.shelterId
    -> ORDER BY AdoptedPetsCount DESC;
```