Java Foundation Training

Coding Challenge Batch-5 : MySQL

Name : Aayush Kamalakar Vadagaonakar

Superset_ID : 5277023

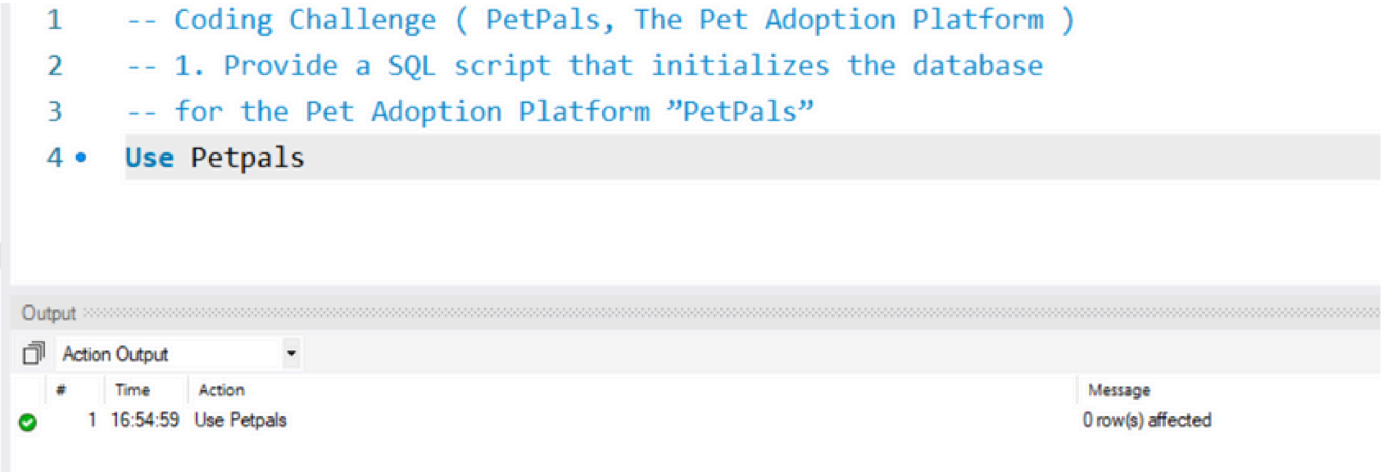
Email : aayushvadagaonkar22@gmail.com

GitHub Link : https://github.com/Aayush220503/Hexaware_Training-/tree/main/AayushCodingChallenge

Topic : PetPals, The Pet Adoption Platform

Task 1: Database Design:

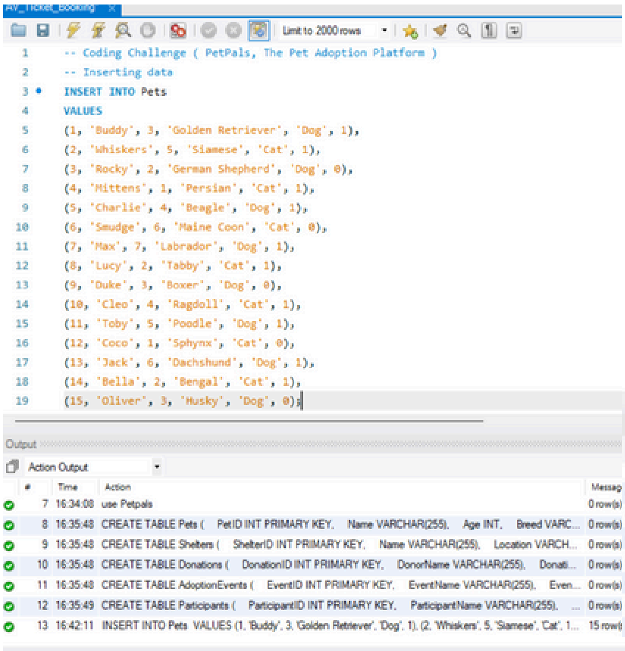
1. Provide a SQL script that initializes the database for the Pet Adoption Platform ”PetPals”.



```sql
-- Coding Challenge ( PetPals, The Pet Adoption Platform )
-- 1. Provide a SQL script that initializes the database
-- for the Pet Adoption Platform ”PetPals”
Use Petpals
```

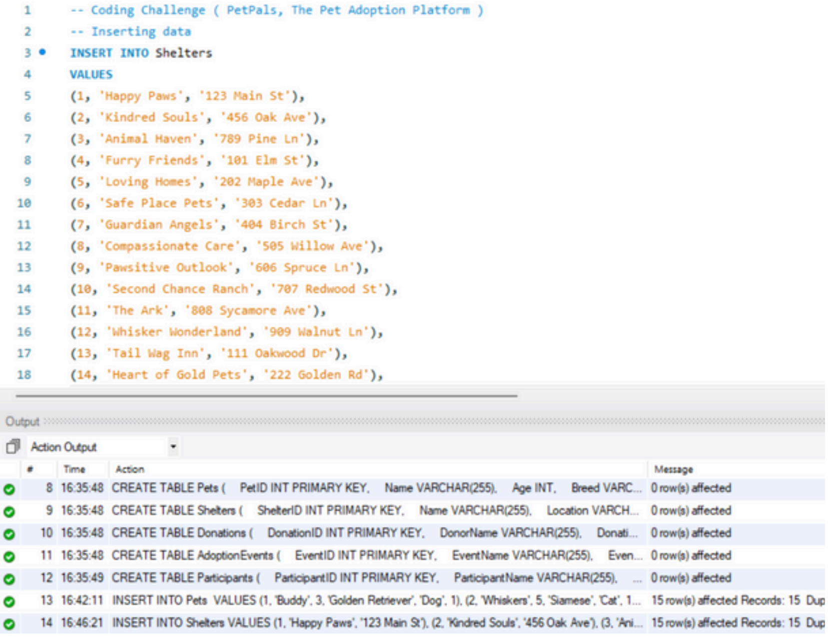
```
#   Time      Action        Message
1   16:54:59  Use Petpals   0 row(s) affected
```

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



```sql
-- Coding Challenge ( PetPals, The Pet Adoption Platform )
-- Inserting data
INSERT INTO Pets
VALUES
(1, 'Buddy', 3, 'Golden Retriever', 'Dog', 1),
(2, 'Whiskers', 5, 'Siamese', 'Cat', 1),
(3, 'Rocky', 2, 'German Shepherd', 'Dog', 0),
(4, 'Mittens', 1, 'Persian', 'Cat', 1),
(5, 'Charlie', 4, 'Beagle', 'Dog', 1),
(6, 'Smudge', 6, 'Maine Coon', 'Cat', 0),
(7, 'Max', 7, 'Labrador', 'Dog', 1),
(8, 'Lucy', 2, 'Tabby', 'Cat', 1),
(9, 'Duke', 3, 'Boxer', 'Dog', 0),
(10, 'Cleo', 4, 'Ragdoll', 'Cat', 1),
(11, 'Toby', 5, 'Poodle', 'Dog', 1),
(12, 'Coco', 1, 'Sphynx', 'Cat', 0),
(13, 'Jack', 6, 'Dachshund', 'Dog', 1),
(14, 'Bella', 2, 'Bengal', 'Cat', 1),
(15, 'Oliver', 3, 'Husky', 'Dog', 0);
```

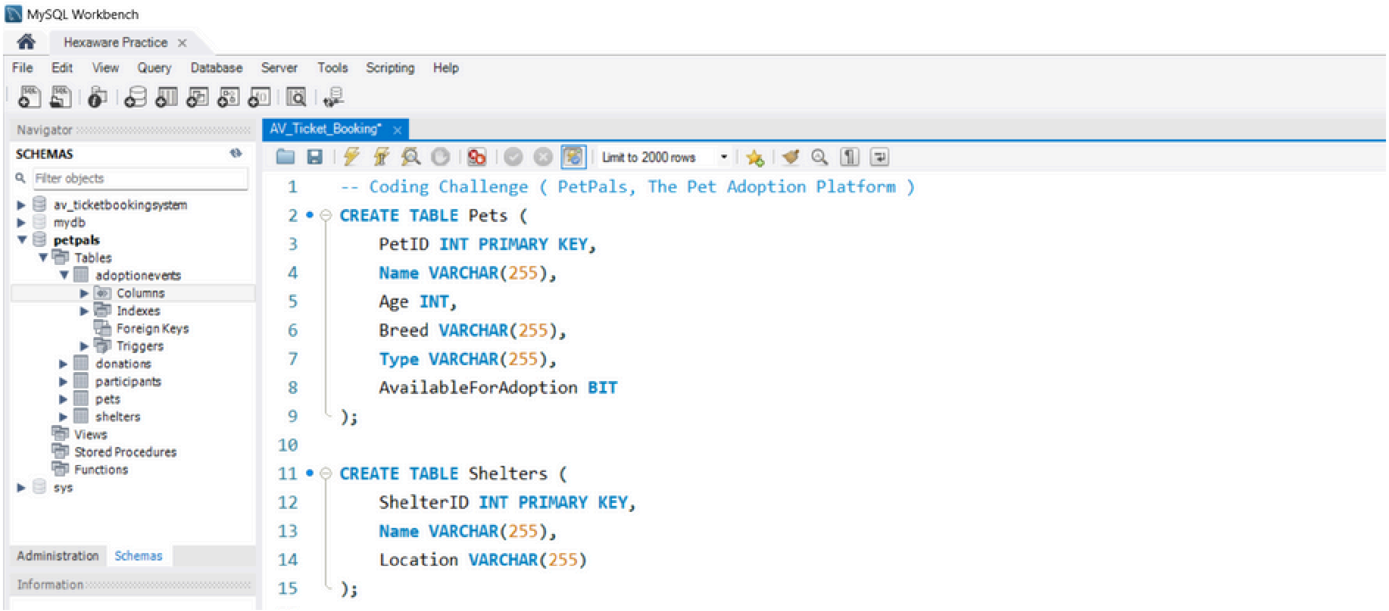
```
#   Time      Action                                                                                          Messag
7   16:34:08  use Petpals                                                                                     0 row(s)
8   16:35:48  CREATE TABLE Pets (   PetID INT PRIMARY KEY,   Name VARCHAR(255),   Age INT,   Breed VARC...    0 row(s)
9   16:35:48  CREATE TABLE Shelters (   ShelterID INT PRIMARY KEY,   Name VARCHAR(255),   Location VARCH...   0 row(s)
10  16:35:48  CREATE TABLE Donations (   DonationID INT PRIMARY KEY,   DonorName VARCHAR(255),   Donati...    0 row(s)
11  16:35:48  CREATE TABLE AdoptionEvents (   EventID INT PRIMARY KEY,   EventName VARCHAR(255),   Even...    0 row(s)
12  16:35:49  CREATE TABLE Participants (   ParticipantID INT PRIMARY KEY,   ParticipantName VARCHAR(255),   ...   0 row(s)
13  16:42:11  INSERT INTO Pets  VALUES (1, 'Buddy', 3, 'Golden Retriever', 'Dog', 1), (2, 'Whiskers', 5, 'Siamese', 'Cat', 1...   15 row(
```



```sql
-- Coding Challenge ( PetPals, The Pet Adoption Platform )
-- Inserting data
INSERT INTO Shelters
VALUES
(1, 'Happy Paws', '123 Main St'),
(2, 'Kindred Souls', '456 Oak Ave'),
(3, 'Animal Haven', '789 Pine Ln'),
(4, 'Furry Friends', '101 Elm St'),
(5, 'Loving Homes', '202 Maple Ave'),
(6, 'Safe Place Pets', '303 Cedar Ln'),
(7, 'Guardian Angels', '404 Birch St'),
(8, 'Compassionate Care', '505 Willow Ave'),
(9, 'Pawsitive Outlook', '606 Spruce Ln'),
(10, 'Second Chance Ranch', '707 Redwood St'),
(11, 'The Ark', '808 Sycamore Ave'),
(12, 'Whisker Wonderland', '909 Walnut Ln'),
(13, 'Tail Wag Inn', '111 Oakwood Dr'),
(14, 'Heart of Gold Pets', '222 Golden Rd'),
```

```
#   Time      Action                                                                                          Message
8   16:35:48  CREATE TABLE Pets (   PetID INT PRIMARY KEY,   Name VARCHAR(255),   Age INT,   Breed VARC...    0 row(s) affected
9   16:35:48  CREATE TABLE Shelters (   ShelterID INT PRIMARY KEY,   Name VARCHAR(255),   Location VARCH...   0 row(s) affected
10  16:35:48  CREATE TABLE Donations (   DonationID INT PRIMARY KEY,   DonorName VARCHAR(255),   Donati...    0 row(s) affected
11  16:35:48  CREATE TABLE AdoptionEvents (   EventID INT PRIMARY KEY,   EventName VARCHAR(255),   Even...    0 row(s) affected
12  16:35:49  CREATE TABLE Participants (   ParticipantID INT PRIMARY KEY,   ParticipantName VARCHAR(255),   ...   0 row(s) affected
13  16:42:11  INSERT INTO Pets  VALUES (1, 'Buddy', 3, 'Golden Retriever', 'Dog', 1), (2, 'Whiskers', 5, 'Siamese', 'Cat', 1...   15 row(s) affected Records: 15  Dup
14  16:46:21  INSERT INTO Shelters VALUES (1, 'Happy Paws', '123 Main St'), (2, 'Kindred Souls', '456 Oak Ave'), (3, 'Ani...   15 row(s) affected Records: 15  Dup
```

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



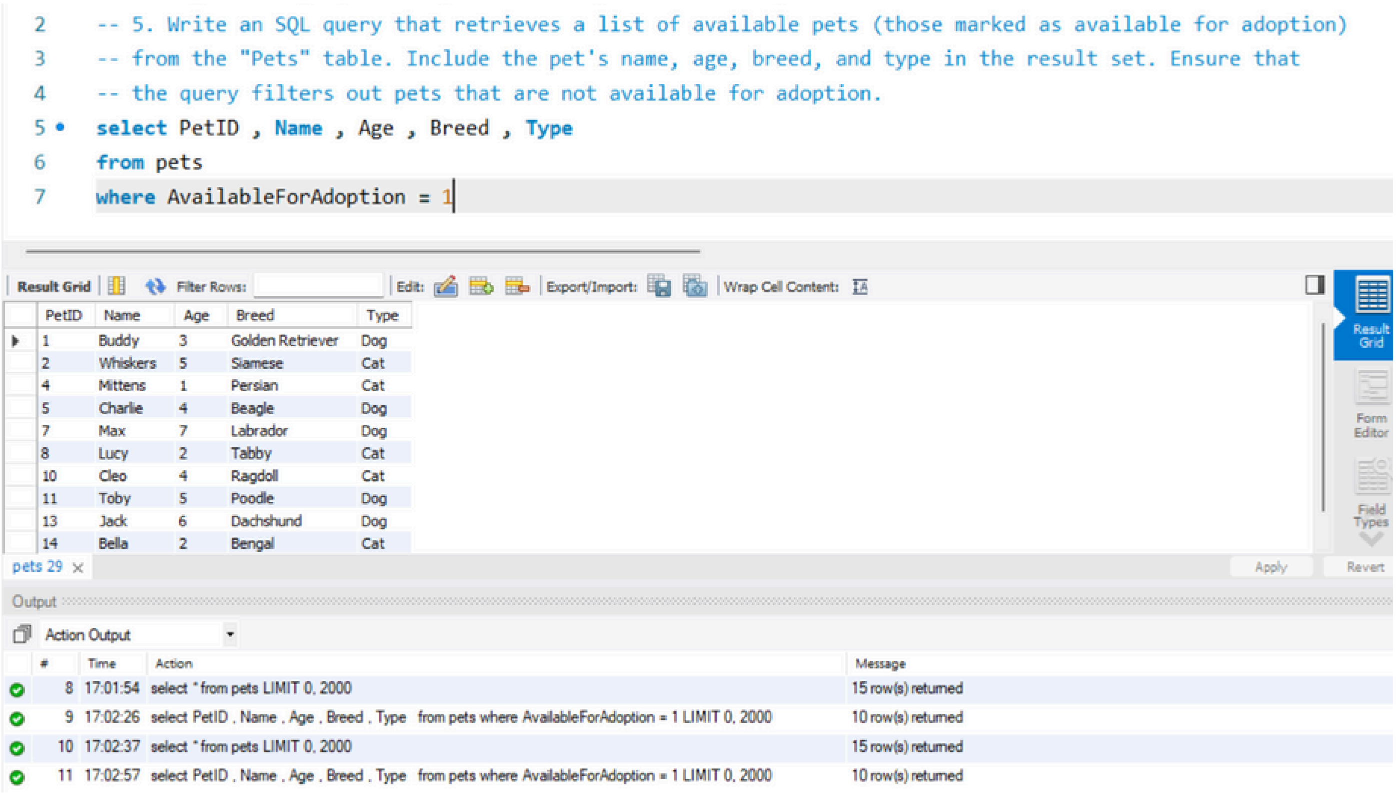
```sql
-- Coding Challenge ( PetPals, The Pet Adoption Platform )
CREATE TABLE Pets (
    PetID INT PRIMARY KEY,
    Name VARCHAR(255),
    Age INT,
    Breed VARCHAR(255),
    Type VARCHAR(255),
    AvailableForAdoption BIT
);

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

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

-> Completed Initially while creating DB;

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.

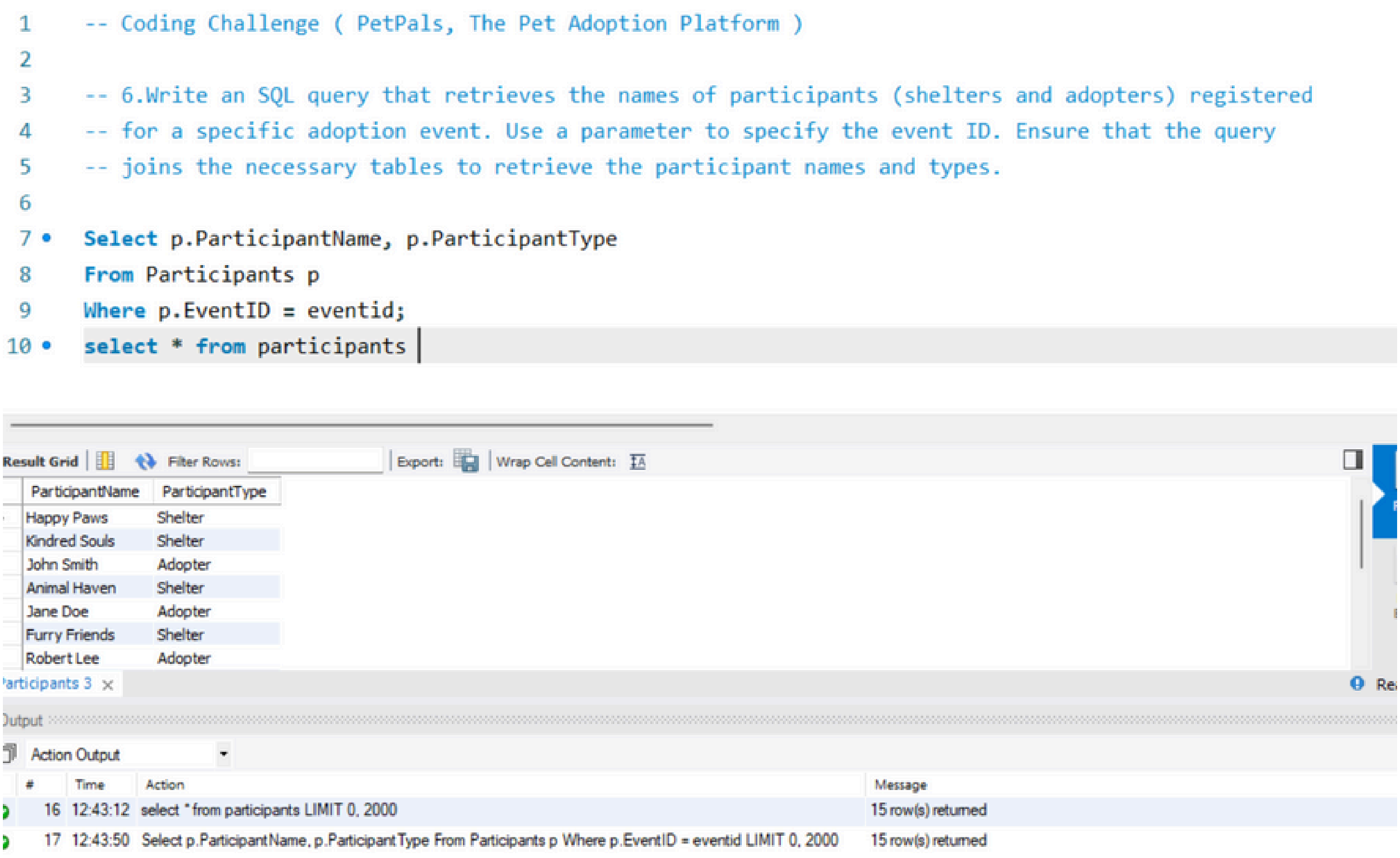


```sql
-- 5. Write an SQL query that retrieves a list of available pets (those marked as available for adoption)
-- from the "Pets" table. Include the pet's name, age, breed, and type in the result set. Ensure that
-- the query filters out pets that are not available for adoption.
select PetID , Name , Age , Breed , Type
from pets
where AvailableForAdoption = 1
```

PetID	Name	Age	Breed	Type
1	Buddy	3	Golden Retriever	Dog
2	Whiskers	5	Siamese	Cat
4	Mittens	1	Persian	Cat
5	Charlie	4	Beagle	Dog
7	Max	7	Labrador	Dog
8	Lucy	2	Tabby	Cat
10	Cleo	4	Ragdoll	Cat
11	Toby	5	Poodle	Dog
13	Jack	6	Dachshund	Dog
14	Bella	2	Bengal	Cat

#	Time	Action	Message
8	17:01:54	select * from pets LIMIT 0, 2000	15 row(s) returned
9	17:02:26	select PetID , Name , Age , Breed , Type from pets where AvailableForAdoption = 1 LIMIT 0, 2000	10 row(s) returned
10	17:02:37	select * from pets LIMIT 0, 2000	15 row(s) returned
11	17:02:57	select PetID , Name , Age , Breed , Type from pets where AvailableForAdoption = 1 LIMIT 0, 2000	10 row(s) returned

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.



```
-- Coding Challenge ( PetPals, The Pet Adoption Platform )

-- 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.ParticipantName, p.ParticipantType
From Participants p
Where p.EventID = eventid;
select * from participants
```

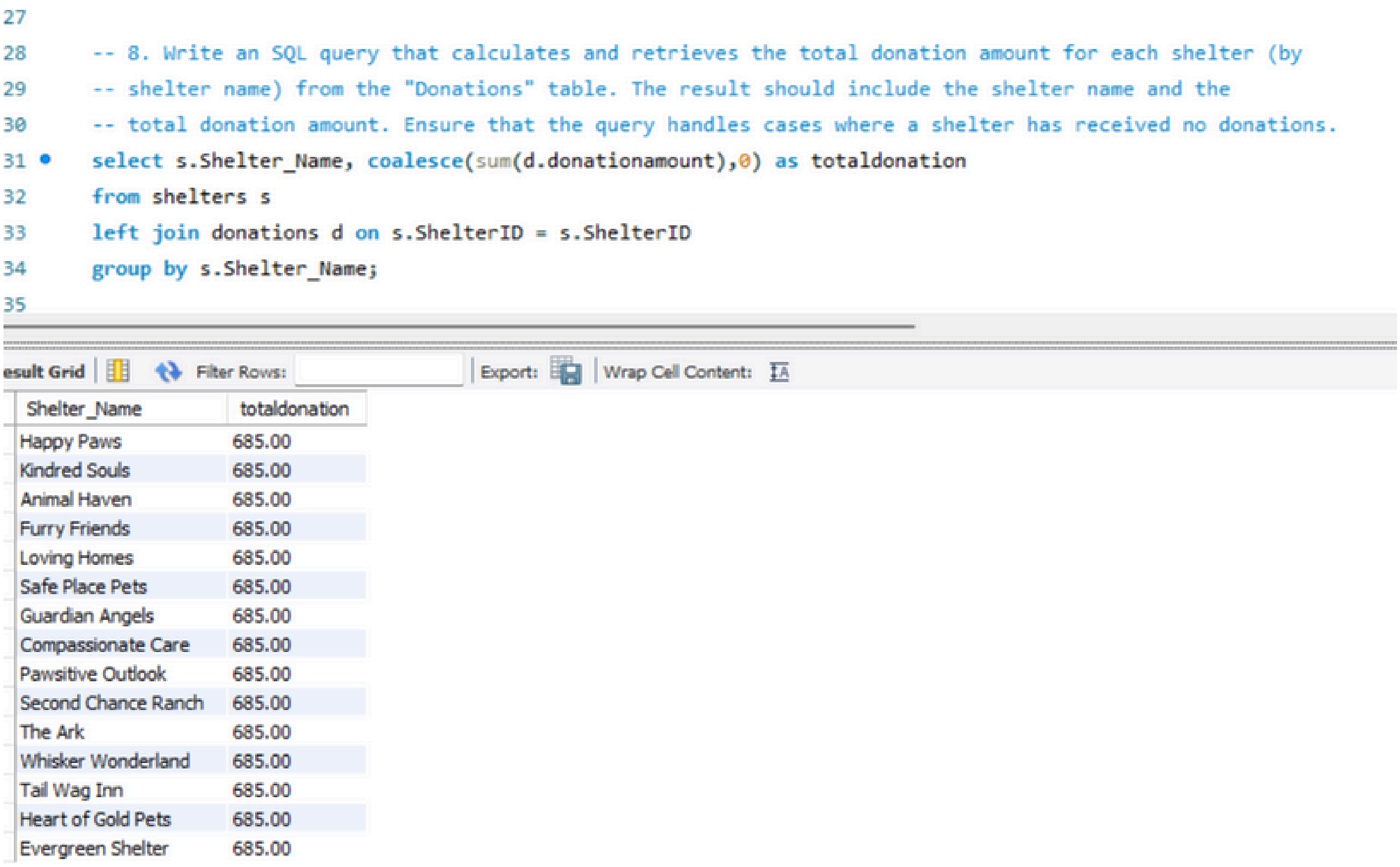
ParticipantName	ParticipantType
Happy Paws	Shelter
Kindred Souls	Shelter
John Smith	Adopter
Animal Haven	Shelter
Jane Doe	Adopter
Furry Friends	Shelter
Robert Lee	Adopter

#	Time	Action	Message
16	12:43:12	select * from participants LIMIT 0, 2000	15 row(s) returned
17	12:43:50	Select p.ParticipantName, p.ParticipantType From Participants p Where p.EventID = eventid LIMIT 0, 2000	15 row(s) returned

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.

-> Didn't got the question and the procedure concept is not clear to me .

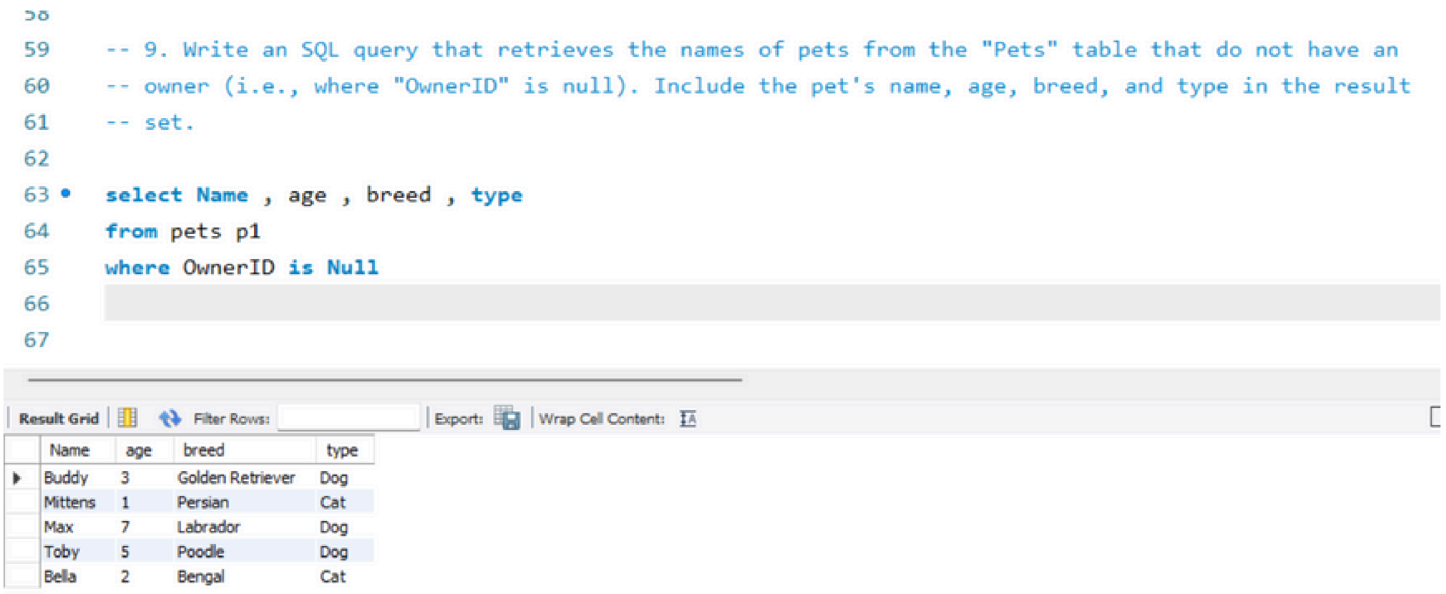
8. Write an SQL query that calculates and retrieves the total donation amount for each shelter (by shelter name) from the "Donations" table. The result should include the shelter name and the total donation amount. Ensure that the query handles cases where a shelter has received no donations.



```
-- 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.
select s.Shelter_Name, coalesce(sum(d.donationamount),0) as totaldonation
from shelters s
left join donations d on s.ShelterID = s.ShelterID
group by s.Shelter_Name;
```

Shelter_Name	totaldonation
Happy Paws	685.00
Kindred Souls	685.00
Animal Haven	685.00
Furry Friends	685.00
Loving Homes	685.00
Safe Place Pets	685.00
Guardian Angels	685.00
Compassionate Care	685.00
Pawsitive Outlook	685.00
Second Chance Ranch	685.00
The Ark	685.00
Whisker Wonderland	685.00
Tail Wag Inn	685.00
Heart of Gold Pets	685.00
Evergreen Shelter	685.00

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.

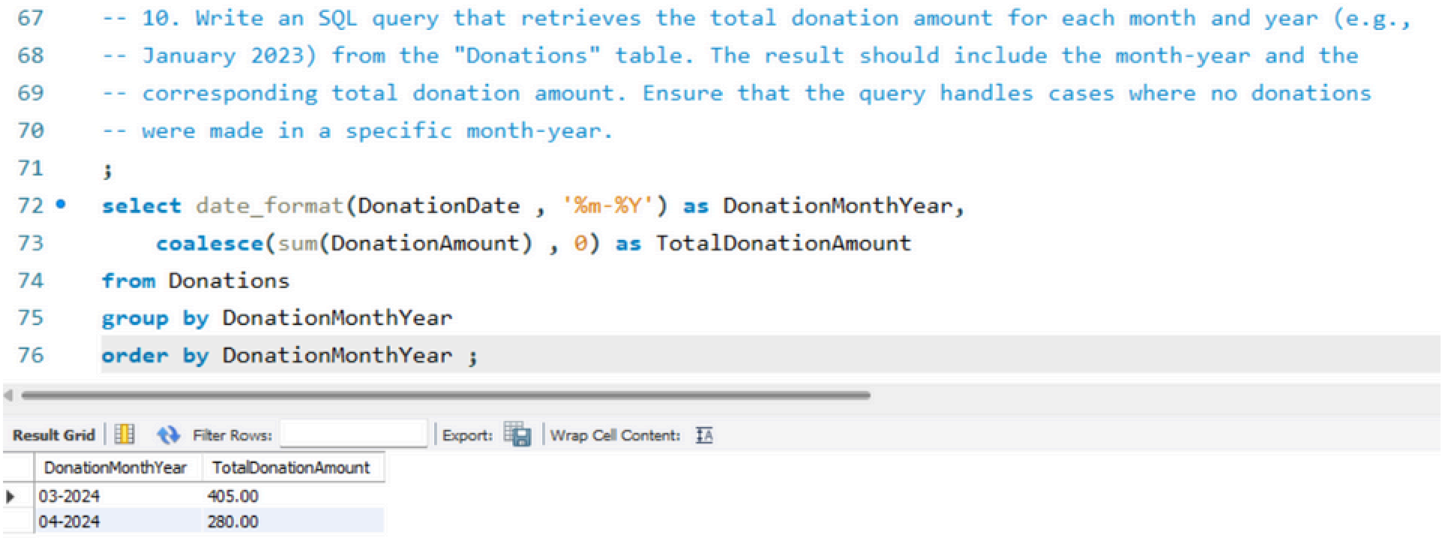


```
-- 9. Write an SQL query that retrieves the names of pets from the "Pets" table that do not have an
-- owner (i.e., where "OwnerID" is null). Include the pet's name, age, breed, and type in the result
-- set.

select Name , age , breed , type
from pets p1
where OwnerID is Null
```

Name	age	breed	type
Buddy	3	Golden Retriever	Dog
Mittens	1	Persian	Cat
Max	7	Labrador	Dog
Toby	5	Poodle	Dog
Bella	2	Bengal	Cat

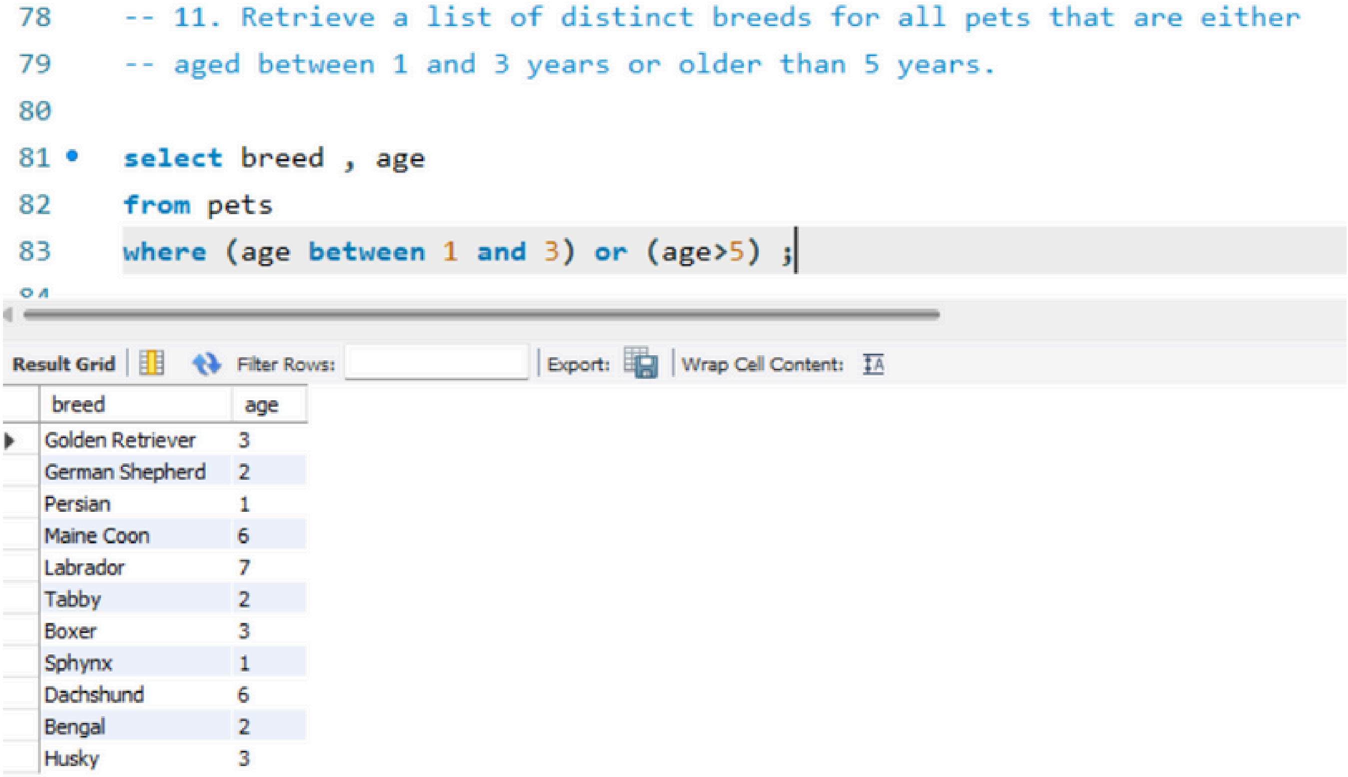
10. Write an SQL query that retrieves the total donation amount for each month and year (e.g., January 2023) from the "Donations" table. The result should include the month-year and the corresponding total donation amount. Ensure that the query handles cases where no donations were made in a specific month-year.



```
-- 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.
;
select date_format(DonationDate , '%m-%Y') as DonationMonthYear,
    coalesce(sum(DonationAmount) , 0) as TotalDonationAmount
from Donations
group by DonationMonthYear
order by DonationMonthYear ;
```

DonationMonthYear	TotalDonationAmount
03-2024	405.00
04-2024	280.00

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

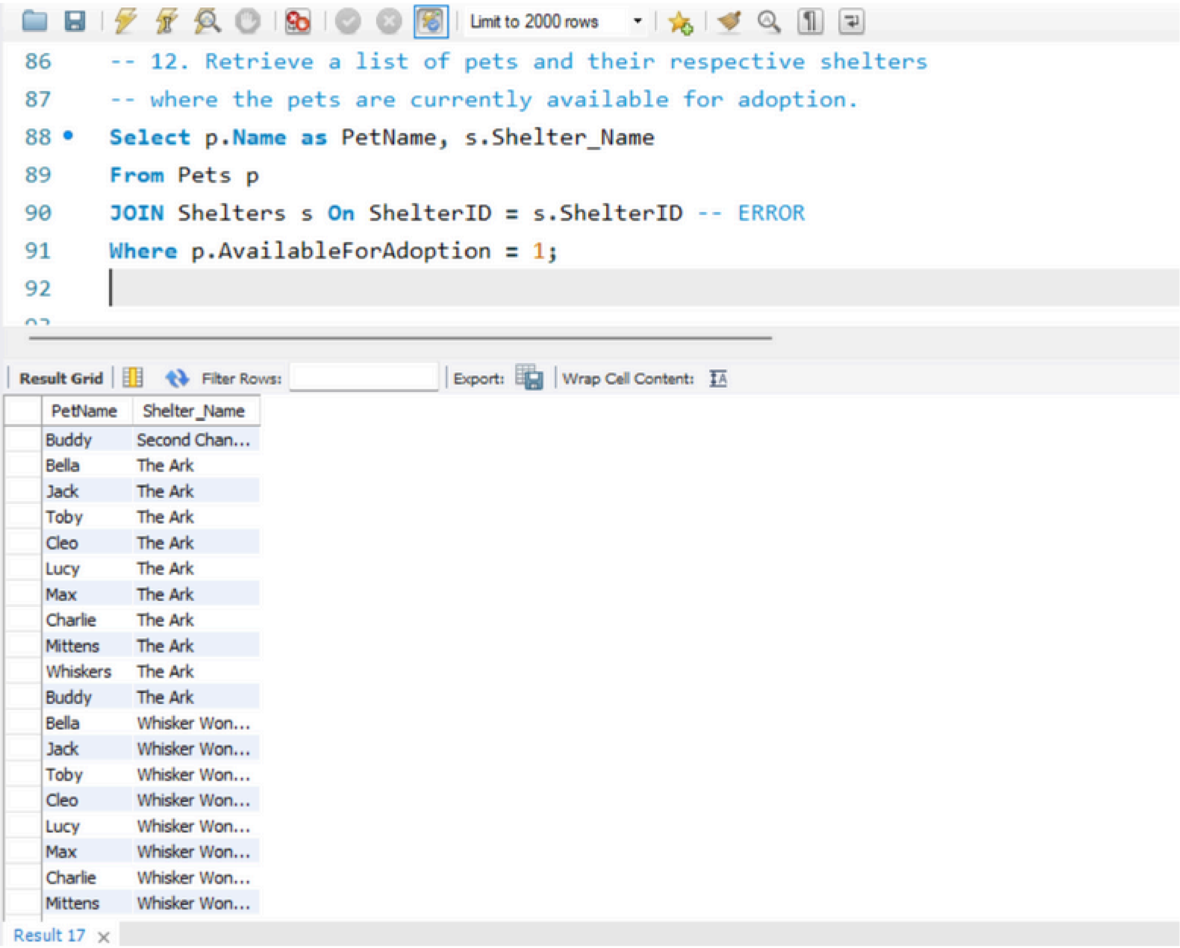


```
-- 11. Retrieve a list of distinct breeds for all pets that are either
-- aged between 1 and 3 years or older than 5 years.

select breed , age
from pets
where (age between 1 and 3) or (age>5) ;
```

breed	age
Golden Retriever	3
German Shepherd	2
Persian	1
Maine Coon	6
Labrador	7
Tabby	2
Boxer	3
Sphynx	1
Dachshund	6
Bengal	2
Husky	3

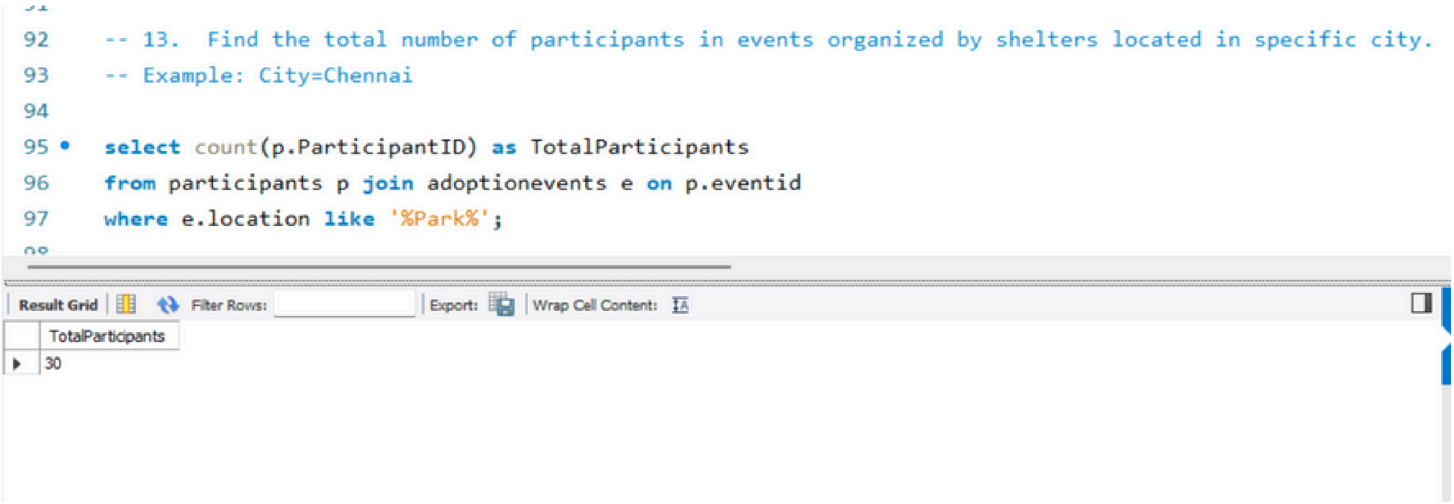
12. Retrieve a list of pets and their respective shelters where the pets are currently available for adoption.



```
-- 12. Retrieve a list of pets and their respective shelters
-- where the pets are currently available for adoption.
Select p.Name as PetName, s.Shelter_Name
From Pets p
JOIN Shelters s On ShelterID = s.ShelterID -- ERROR
Where p.AvailableForAdoption = 1;
```

PetName	Shelter_Name
Buddy	Second Chan...
Bella	The Ark
Jack	The Ark
Toby	The Ark
Cleo	The Ark
Lucy	The Ark
Max	The Ark
Charlie	The Ark
Mittens	The Ark
Whiskers	The Ark
Buddy	The Ark
Bella	Whisker Won...
Jack	Whisker Won...
Toby	Whisker Won...
Cleo	Whisker Won...
Lucy	Whisker Won...
Max	Whisker Won...
Charlie	Whisker Won...
Mittens	Whisker Won...

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

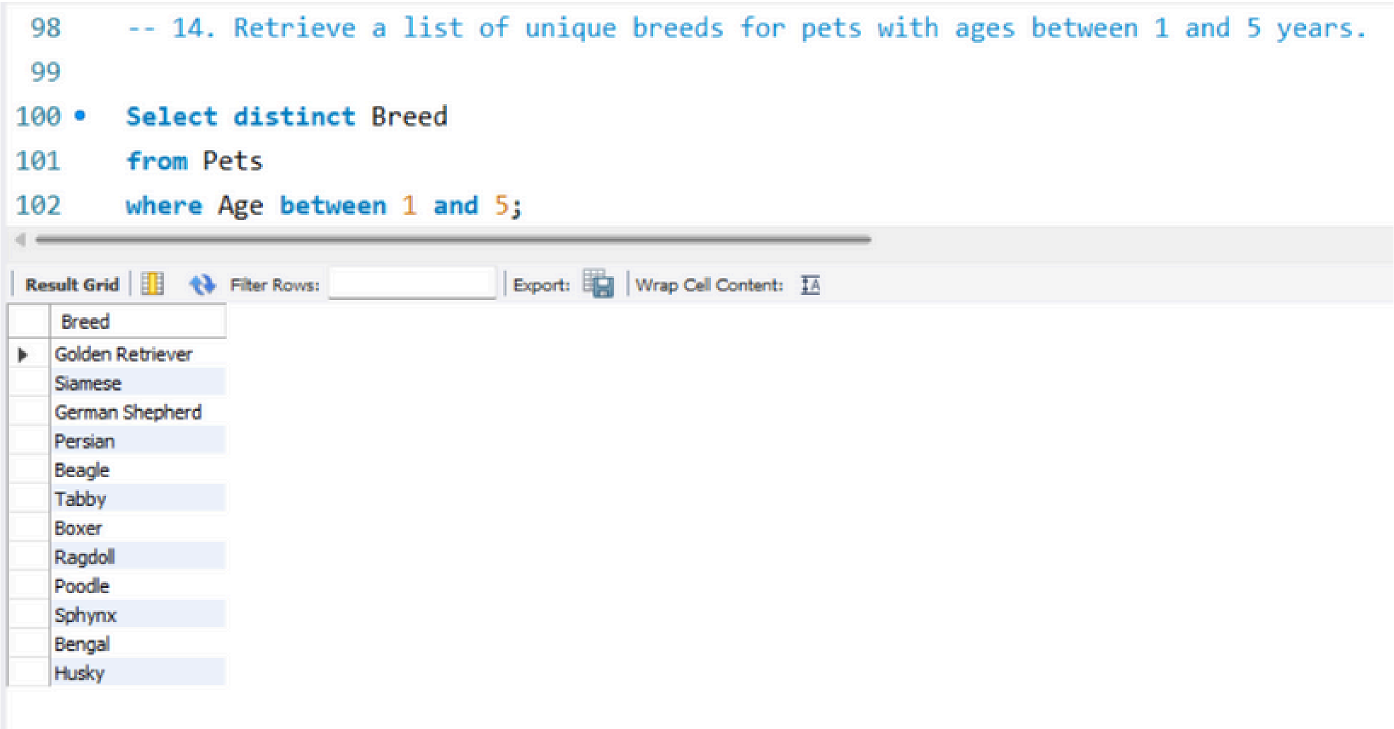


```
-- 13.  Find the total number of participants in events organized by shelters located in specific city.
-- Example: City=Chennai

select count(p.ParticipantID) as TotalParticipants
from participants p join adoptionevents e on p.eventid
where e.location like '%Park%';
```

TotalParticipants
30

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

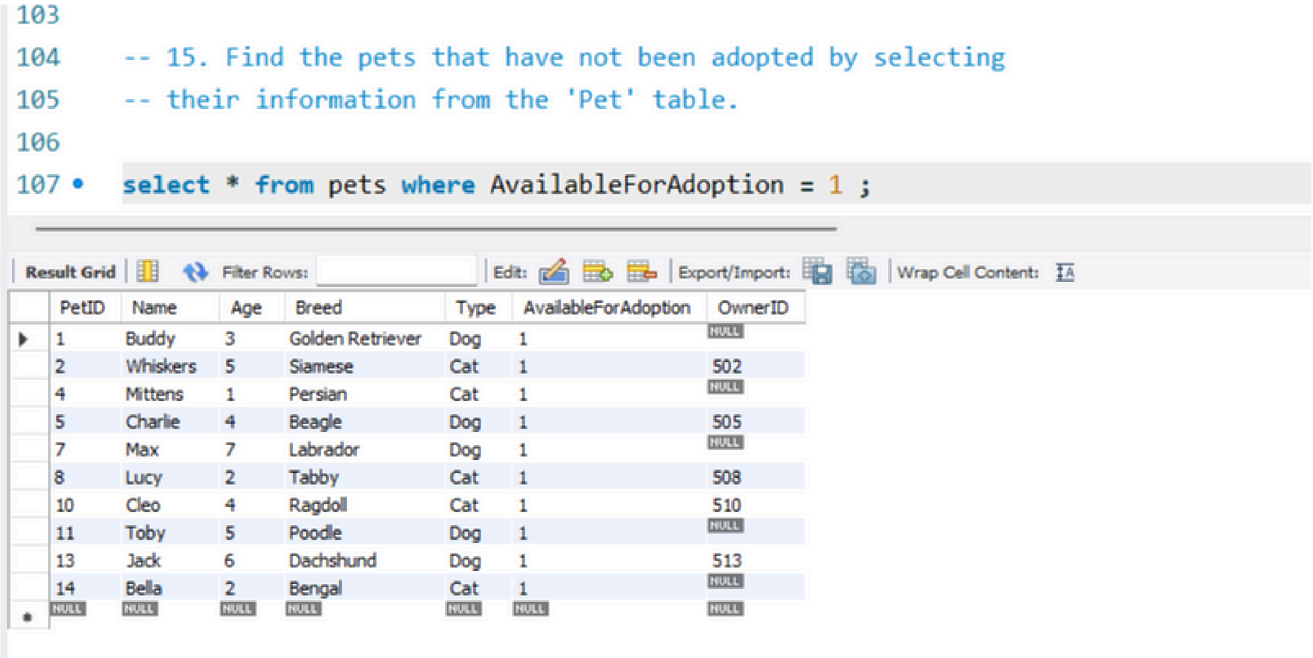


```sql
-- 14. Retrieve a list of unique breeds for pets with ages between 1 and 5 years.

Select distinct Breed
from Pets
where Age between 1 and 5;
```

Breed
Golden Retriever
Siamese
German Shepherd
Persian
Beagle
Tabby
Boxer
Ragdoll
Poodle
Sphynx
Bengal
Husky

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

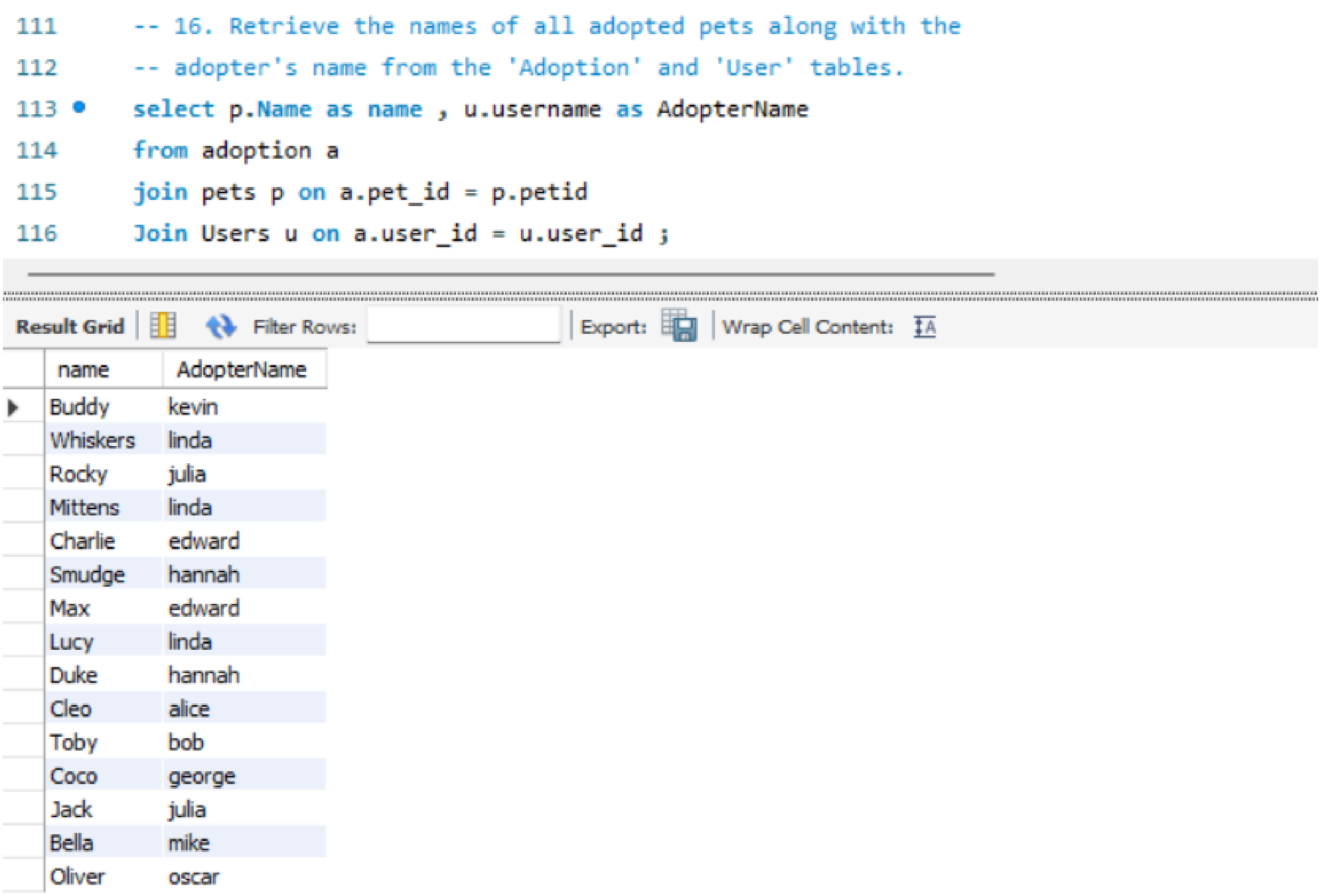


```sql
-- 15. Find the pets that have not been adopted by selecting
-- their information from the 'Pet' table.

select * from pets where AvailableForAdoption = 1 ;
```

PetID	Name	Age	Breed	Type	AvailableForAdoption	OwnerID
1	Buddy	3	Golden Retriever	Dog	1	NULL
2	Whiskers	5	Siamese	Cat	1	502
4	Mittens	1	Persian	Cat	1	NULL
5	Charlie	4	Beagle	Dog	1	505
7	Max	7	Labrador	Dog	1	NULL
8	Lucy	2	Tabby	Cat	1	508
10	Cleo	4	Ragdoll	Cat	1	510
11	Toby	5	Poodle	Dog	1	NULL
13	Jack	6	Dachshund	Dog	1	513
14	Bella	2	Bengal	Cat	1	NULL
NULL	NULL	NULL	NULL	NULL	NULL	NULL

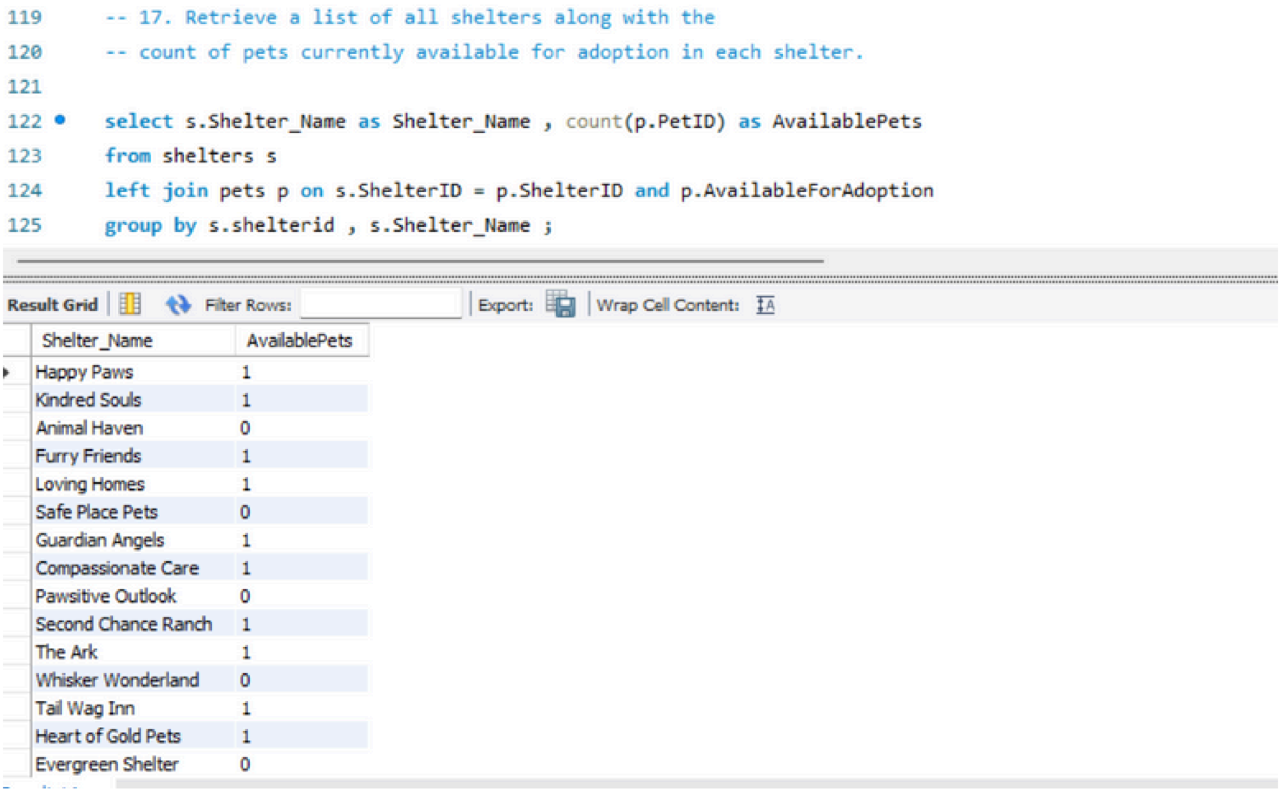
16. Retrieve the names of all adopted pets along with the adopter's name from the 'Adoption' and 'User' tables.



```
-- 16. Retrieve the names of all adopted pets along with the
-- adopter's name from the 'Adoption' and 'User' tables.
select p.Name as name , u.username as AdopterName
from adoption a
join pets p on a.pet_id = p.petid
Join Users u on a.user_id = u.user_id ;
```

name	AdopterName
Buddy	kevin
Whiskers	linda
Rocky	julia
Mittens	linda
Charlie	edward
Smudge	hannah
Max	edward
Lucy	linda
Duke	hannah
Cleo	alice
Toby	bob
Coco	george
Jack	julia
Bella	mike
Oliver	oscar

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

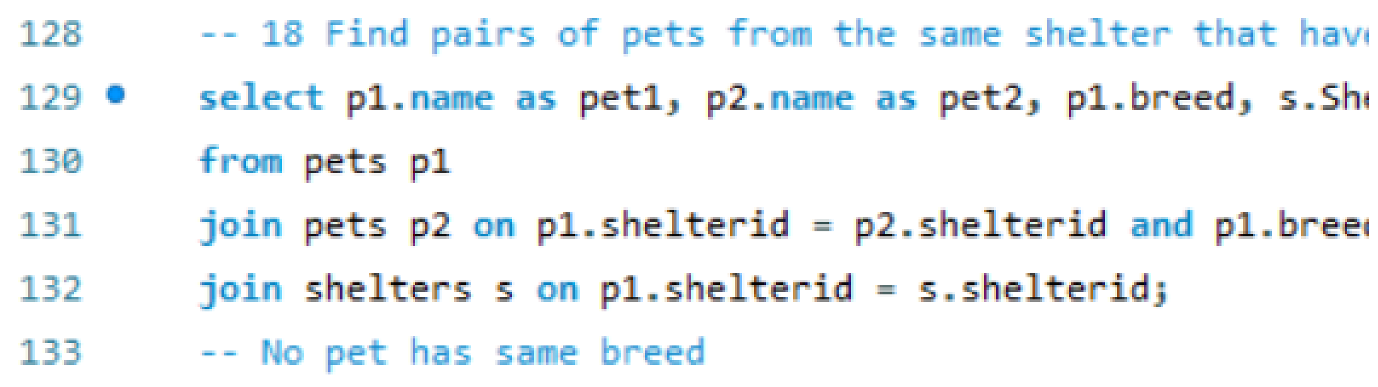


```
-- 17. Retrieve a list of all shelters along with the
-- count of pets currently available for adoption in each shelter.

select s.Shelter_Name as Shelter_Name , count(p.PetID) as AvailablePets
from shelters s
left join pets p on s.ShelterID = p.ShelterID and p.AvailableForAdoption
group by s.shelterid , s.Shelter_Name ;
```

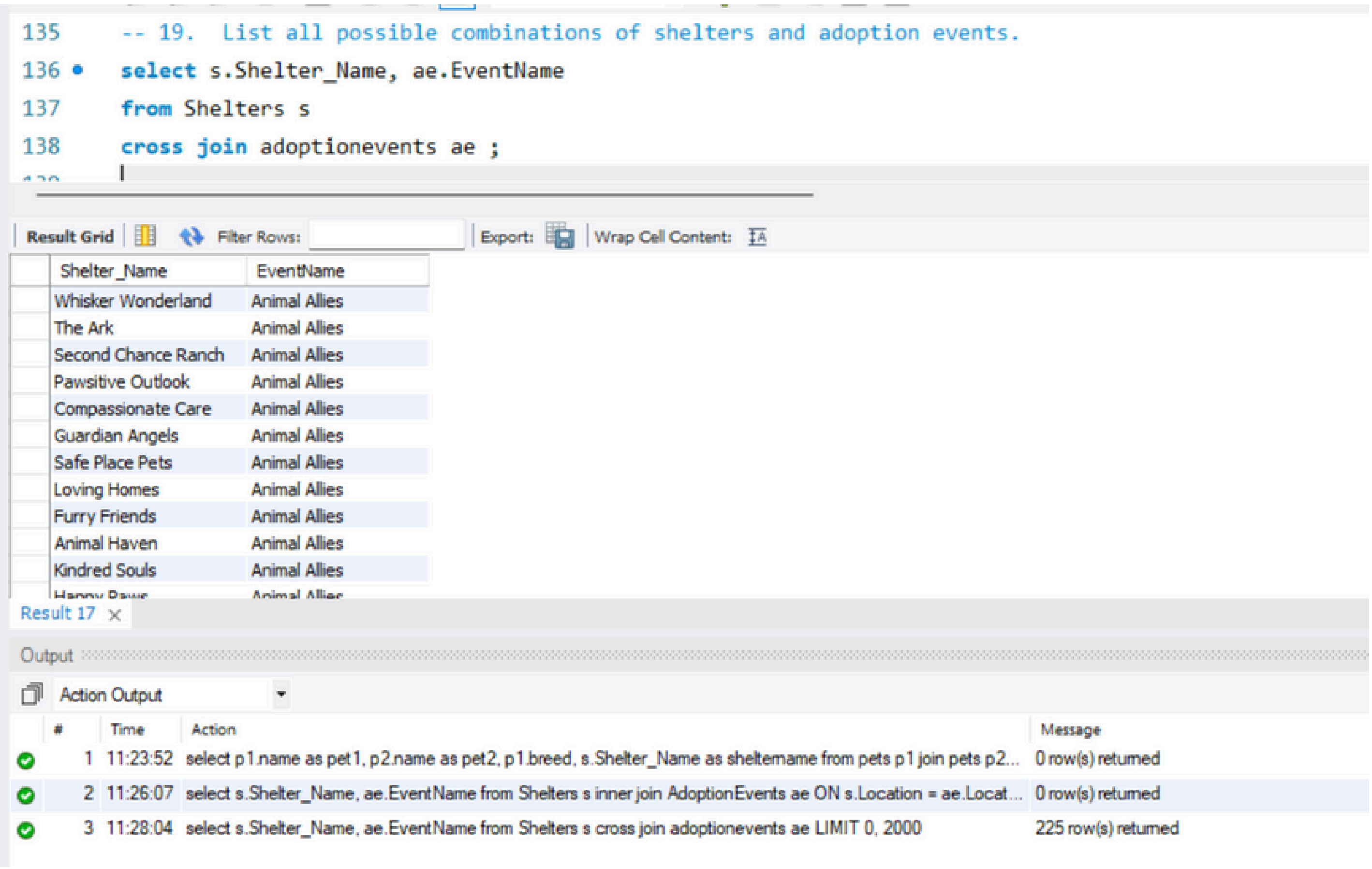
Shelter_Name	AvailablePets
Happy Paws	1
Kindred Souls	1
Animal Haven	0
Furry Friends	1
Loving Homes	1
Safe Place Pets	0
Guardian Angels	1
Compassionate Care	1
Pawsitive Outlook	0
Second Chance Ranch	1
The Ark	1
Whisker Wonderland	0
Tail Wag Inn	1
Heart of Gold Pets	1
Evergreen Shelter	0

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



```sql
-- 18 Find pairs of pets from the same shelter that hav
select p1.name as pet1, p2.name as pet2, p1.breed, s.Sh
from pets p1
join pets p2 on p1.shelterid = p2.shelterid and p1.bree
join shelters s on p1.shelterid = s.shelterid;
-- No pet has same breed
```

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

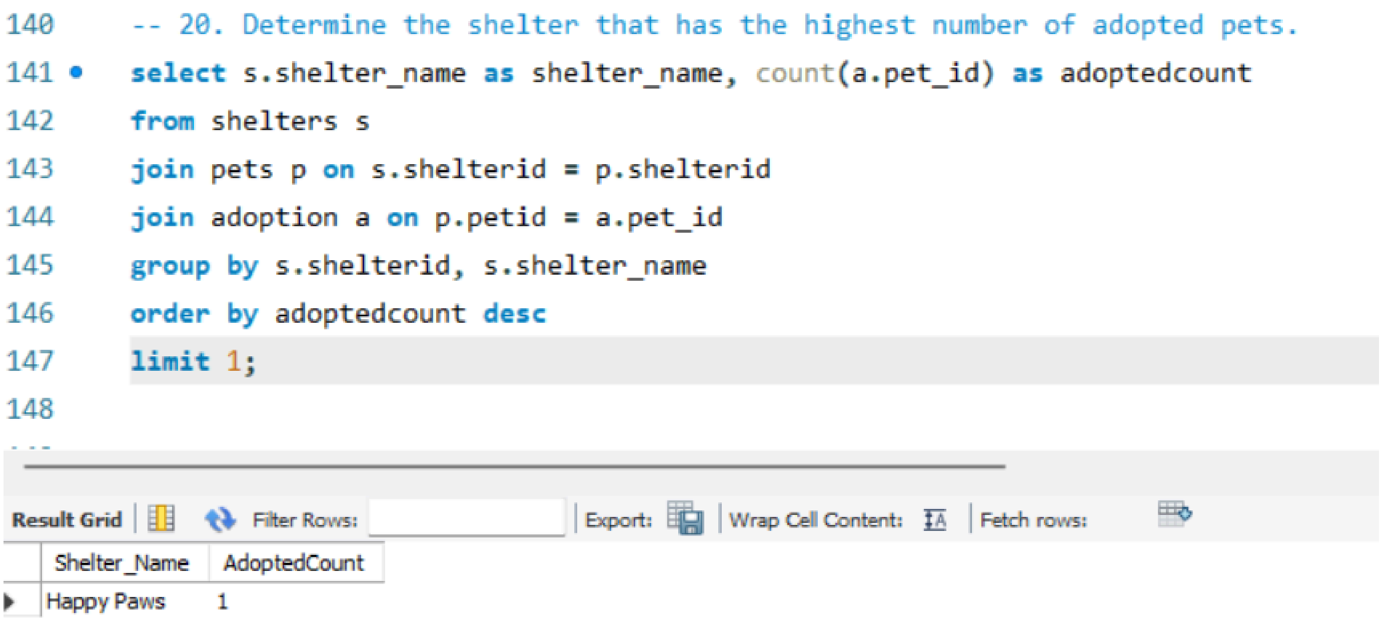


```sql
-- 19.  List all possible combinations of shelters and adoption events.
select s.Shelter_Name, ae.EventName
from Shelters s
cross join adoptionevents ae ;
```

Shelter_Name	EventName
Whisker Wonderland	Animal Allies
The Ark	Animal Allies
Second Chance Ranch	Animal Allies
Pawsitive Outlook	Animal Allies
Compassionate Care	Animal Allies
Guardian Angels	Animal Allies
Safe Place Pets	Animal Allies
Loving Homes	Animal Allies
Furry Friends	Animal Allies
Animal Haven	Animal Allies
Kindred Souls	Animal Allies
Happy Paws	Animal Allies

#	Time	Action	Message
1	11:23:52	select p1.name as pet1, p2.name as pet2, p1.breed, s.Shelter_Name as sheltername from pets p1 join pets p2...	0 row(s) returned
2	11:26:07	select s.Shelter_Name, ae.EventName from Shelters s inner join AdoptionEvents ae ON s.Location = ae.Locat...	0 row(s) returned
3	11:28:04	select s.Shelter_Name, ae.EventName from Shelters s cross join adoptionevents ae LIMIT 0, 2000	225 row(s) returned

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



```sql
-- 20. Determine the shelter that has the highest number of adopted pets.
select s.shelter_name as shelter_name, count(a.pet_id) as adoptedcount
from shelters s
join pets p on s.shelterid = p.shelterid
join adoption a on p.petid = a.pet_id
group by s.shelterid, s.shelter_name
order by adoptedcount desc
limit 1;
```

Shelter_Name	AdoptedCount
Happy Paws	1