

## HEXaware SQL Case Study

Name – Thilak Raaj

### Topic - PetPals, The Pet Adoption Platform

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

The screenshot shows the MySQL Workbench interface with the following SQL code in the Query Editor:

```
1 • CREATE DATABASE PetPals;
2 • USE PetPals;
3
4 • CREATE TABLE Pets (
5     PetID INT AUTO_INCREMENT PRIMARY KEY,
6     Name VARCHAR(255) NOT NULL,
7     Age INT,
8     Breed VARCHAR(255) NOT NULL,
9     Type VARCHAR(255) NOT NULL,
10    AvailableForAdoption BIT NOT NULL
11 );
12 • desc table Donations;
13
14
15 • CREATE TABLE Shelters (
16     ShelterID INT AUTO_INCREMENT PRIMARY KEY,
17     Name VARCHAR(255) NOT NULL,
18     Location VARCHAR(255) NOT NULL
19 );
20
21
```

The Output pane shows the results of the executed commands:

#	Time	Action	Message	Duration / Fetch
117	12:15:13	desc table AdoptionEvents	1 row(s) returned	0.000 sec / 0.000 sec
118	12:15:31	desc table Participants	1 row(s) returned	0.000 sec / 0.000 sec
119	12:16:54	Show tables	7 row(s) returned	0.000 sec / 0.000 sec
120	12:17:38	drop database PetPals	7 row(s) affected	0.110 sec
121	12:17:49	CREATE DATABASE PetPals	1 row(s) affected	0.015 sec
122	12:17:52	USE PetPals	0 row(s) affected	0.000 sec

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

The screenshot shows the MySQL Workbench interface with the following SQL code in the Query Editor:

```
49
50 • CREATE TABLE Participants (
51     ParticipantID INT AUTO_INCREMENT PRIMARY KEY,
52     ParticipantName VARCHAR(255) NOT NULL,
53     ParticipantType VARCHAR(255) NOT NULL,
54     EventID INT,
55     FOREIGN KEY (EventID) REFERENCES AdoptionEvents(EventID)
56     ON DELETE SET NULL
57     ON UPDATE SET NULL
58 );
59 • desc table Participants;
60
61
62 • Show tables;
63
64
65 • INSERT INTO Pets (PetID, Name, Age, Breed, Type, AvailableForAdoption) VALUES
66     (1, 'Buddy', 3, 'Labrador Retriever', 'Dog', 1),
67     (2, 'Whiskers', 2, 'Siamese', 'Cat', 1),
68     (3, 'Max', 5, 'Golden Retriever', 'Dog', 0),
69     (4, 'Snowball', 4, 'Persian', 'Cat', 1),
```

The Result Grid pane shows the tables created in the database:

Tables_in_petpals
adopteds
adoptiveevents
donations
owners
participants
pets
shelters

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

```
22 • CREATE TABLE Donations (
23     DonationID INT AUTO_INCREMENT PRIMARY KEY,
24     DonorName VARCHAR(255) NOT NULL,
25     DonationType VARCHAR(255) NOT NULL,
26     DonationAmount DECIMAL(10, 2),
27     DonationItem VARCHAR(255),
28     DonationDate DATETIME NOT NULL
29 );
30
31
32 • CREATE TABLE AdoptionEvents (
33     EventID INT AUTO_INCREMENT PRIMARY KEY,
34     EventName VARCHAR(255) NOT NULL,
35     EventDate DATETIME NOT NULL,
36     Location VARCHAR(255) NOT NULL
37 );
38 • desc table AdoptionEvents;
39
40
41 • CREATE TABLE Participants (
```

Result Grid | Filter Rows | Export | Wrap Cell Contents | Result Grid

id	select_type	table	partitions	type	possible_keys	key	key_len	ref	rows	filtered	Extra
1	SIMPLE	AdoptionEvents	NULL	ALL	NULL	NULL	NULL	NULL	5	100.00	NULL



```
7     Age INT,
8     Breed VARCHAR(255) NOT NULL,
9     Type VARCHAR(255) NOT NULL,
10    AvailableForAdoption BIT NOT NULL
11 );
12 • desc table Donations;
13
14
15 • CREATE TABLE Shelters (
16     ShelterID INT AUTO_INCREMENT PRIMARY KEY,
17     Name VARCHAR(255) NOT NULL,
18     Location VARCHAR(255) NOT NULL
19 );
20
21
22 • CREATE TABLE Donations (
23     DonationID INT AUTO_INCREMENT PRIMARY KEY,
24     DonorName VARCHAR(255) NOT NULL,
25     DonationType VARCHAR(255) NOT NULL,
26     DonationAmount DECIMAL(10, 2),
```

Result Grid | Filter Rows | Export | Wrap Cell Contents | Result Grid

id	select_type	table	partitions	type	possible_keys	key	key_len	ref	rows	filtered	Extra
1	SIMPLE	Donations	NULL	ALL	NULL	NULL	NULL	NULL	10	100.00	NULL



MySQL Workbench

File Edit View Query Database Server Tools Scripting Help

Query 1 ×

```
1 • CREATE DATABASE PetPals;
2 • USE PetPals;
3
4 • CREATE TABLE Pets (
5     PetID INT AUTO_INCREMENT PRIMARY KEY,
6     Name VARCHAR(255) NOT NULL,
7     Age INT,
8     Breed VARCHAR(255) NOT NULL,
9     Type VARCHAR(255) NOT NULL,
10    AvailableForAdoption BIT NOT NULL
11 );
12 • desc table Shelters;
13
14
15 • CREATE TABLE Shelters (
16     ShelterID INT AUTO_INCREMENT PRIMARY KEY,
17     Name VARCHAR(255) NOT NULL,
18     Location VARCHAR(255) NOT NULL
19 );
20
```

Result Grid | Filter Rows | Export: | Wrap Cell Contents: | 15

ID	Select_Type	Table	Partitions	Type	Possible_Keys	Key	Key_Len	Ref	Rows	Filtered	Extra
1	SIMPLE	Shelters	NULL	ALL	NULL	NULL	NULL	NULL	5	100.00	NULL

Result 51 ×

34°C Sunny

MySQL Workbench

File Edit View Query Database Server Tools Scripting Help

Query 1 ×

```
1 • CREATE DATABASE PetPals;
2 • USE PetPals;
3
4 • CREATE TABLE Pets (
5     PetID INT AUTO_INCREMENT PRIMARY KEY,
6     Name VARCHAR(255) NOT NULL,
7     Age INT,
8     Breed VARCHAR(255) NOT NULL,
9     Type VARCHAR(255) NOT NULL,
10    AvailableForAdoption BIT NOT NULL
11 );
12 • desc table Pets;
13
14
15 • CREATE TABLE Shelters (
16     ShelterID INT AUTO_INCREMENT PRIMARY KEY,
17     Name VARCHAR(255) NOT NULL,
18     Location VARCHAR(255) NOT NULL
19 );
20
```

Result Grid | Filter Rows | Export: | Wrap Cell Contents: | 15

ID	Select_Type	Table	Partitions	Type	Possible_Keys	Key	Key_Len	Ref	Rows	Filtered	Extra
1	SIMPLE	Pets	NULL	ALL	NULL	NULL	NULL	NULL	10	100.00	NULL

Result 50 ×

34°C Sunny

MySQL Workbench

File Edit View Query Database Server Tools Scripting Help

Query 1 ×

```
34
35    EventName VARCHAR(255) NOT NULL,
36    EventDate DATETIME NOT NULL,
37    Location VARCHAR(255) NOT NULL
38 );
39
40 • desc table AdoptionEvents;
41
42 • CREATE TABLE Participants (
43     ParticipantID INT AUTO_INCREMENT PRIMARY KEY,
44     ParticipantName VARCHAR(255) NOT NULL,
45     ParticipantType VARCHAR(255) NOT NULL,
46     EventID INT,
47     FOREIGN KEY (EventID) REFERENCES AdoptionEvents(EventID)
48     ON DELETE SET NULL
49     ON UPDATE SET NULL
50 );
51
52
53
```

Result Grid | Filter Rows | Export: | Wrap Cell Contents: | 15

ID	Select_Type	Table	Partitions	Type	Possible_Keys	Key	Key_Len	Ref	Rows	Filtered	Extra
1	SIMPLE	Participants	NULL	ALL	NULL	NULL	NULL	NULL	10	100.00	NULL

Result 54 ×

34°C Sunny

MySQL Workbench

File Edit View Query Database Server Tools Scripting Help

Query 1 ×

```
34
35    EventName VARCHAR(255) NOT NULL,
36    EventDate DATETIME NOT NULL,
37    Location VARCHAR(255) NOT NULL
38 );
39
40 • desc table AdoptionEvents;
41
42 • CREATE TABLE Participants (
43     ParticipantID INT AUTO_INCREMENT PRIMARY KEY,
44     ParticipantName VARCHAR(255) NOT NULL,
45     ParticipantType VARCHAR(255) NOT NULL,
46     EventID INT,
47     FOREIGN KEY (EventID) REFERENCES AdoptionEvents(EventID)
48     ON DELETE SET NULL
49     ON UPDATE SET NULL
50 );
51
52
53
```

Result Grid | Filter Rows | Export: | Wrap Cell Contents: | 15

ID	Select_Type	Table	Partitions	Type	Possible_Keys	Key	Key_Len	Ref	Rows	Filtered	Extra
1	SIMPLE	Participants	NULL	ALL	NULL	NULL	NULL	NULL	10	100.00	NULL

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

The screenshot shows the MySQL Workbench interface. The top window is titled 'Query 1' and contains the following SQL code:

```
108 (5, 'Pet Paradise Adoption Event', '2024-04-30 14:00:00', 'Ooty')  
109  
110  
111 • INSERT INTO Participants (ParticipantID, ParticipantName, ParticipantType, EventID) VALUES  
112 (1, 'Happy', 'Shelter', 1),  
113 (2, 'Glory', 'Shelter', 2),  
114 (3, 'Joy', 'Shelter', 3),  
115 (4, 'Fluffy', 'Shelter', 4),  
116 (5, 'Cute', 'Shelter', 5),  
117 (6, 'Aditya', 'Adopter', 1),  
118 (7, 'Riya', 'Adopter', 2),  
119 (8, 'Ananya', 'Adopter', 3),  
120 (9, 'Vivek', 'Donor', 4),  
121 (10, 'Neha', 'Donor', 5)  
122  
123  
124 • CREATE DATABASE IF NOT EXISTS PetPals;  
125  
126 • SELECT IF( (SELECT COUNT(*) FROM INFORMATION_SCHEMA.SCHEMATA WHERE SCHEMA_NAME = 'PetPals') > 0, 'PetPals database already exists.', 'PetPals database created successfully.' ) AS Message;
```

The bottom window is titled 'Result Grid' and displays the message: 'PetPals database already exists.'

The system tray at the bottom of the screen shows the date and time as 18-04-2024 12:11, and the weather as 34°C Sunny.

#### 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 Workbench

```

111 • INSERT INTO Participants (ParticipantID, ParticipantName, ParticipantType, EventID) VALUES
112     (1, 'Happy', 'Shelter', 1),
113     (2, 'Glory', 'Shelter', 2),
114     (3, 'Joy', 'Shelter', 3),
115     (4, 'Fluffy', 'Shelter', 4),
116     (5, 'Cute', 'Shelter', 5),
117     (6, 'Aditya', 'Adopter', 1),
118     (7, 'Riya', 'Adopter', 2),
119     (8, 'Ananya', 'Adopter', 3),
120     (9, 'Vivek', 'Donor', 4),
121     (10, 'Neha', 'Donor', 5);
122
123
124 • CREATE DATABASE IF NOT EXISTS PetPals;
125
126 • SELECT IF( (SELECT COUNT(*) FROM INFORMATION_SCHEMA.SCHEMATA WHERE SCHEMA_NAME = 'PetPals') > 0, 'PetPals database already exists.', 'PetPals database created successfully.') AS Message;
127
128 • SELECT Name, Age, Breed, Type
129   FROM Pets
130   WHERE AvailableForAdoption = 1;

```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Result Grid | Form Editor | Read Only

Name	Age	Breed	Type
Buddy	3	Labrador Retriever	Dog
Whiskers	2	Siamese	Cat
Snowball	4	Persian	Cat
Rocky	2	German Shepherd	Dog
Luna	1	Maine Coon	Cat
Milo	2	Ragdoll	Cat
Bailey	6	Beagle	Dog
Oliver	3	Russian Blue	Cat

Pets 46 ×

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 Workbench

```

123
124 • CREATE DATABASE IF NOT EXISTS PetPals;
125
126 • SELECT IF( (SELECT COUNT(*) FROM INFORMATION_SCHEMA.SCHEMATA WHERE SCHEMA_NAME = 'PetPals') > 0, 'PetPals database already exists.', 'PetPals database created successfully.') AS Message;
127
128 • SELECT Name, Age, Breed, Type
129   FROM Pets
130   WHERE AvailableForAdoption = 1;
131
132 • SELECT Participants.ParticipantName
133   FROM Participants
134   JOIN AdoptionEvents ON Participants.EventID = AdoptionEvents.EventID
135   WHERE AdoptionEvents.EventID = 3;
136
137
138 DELIMITER //
139
140 • CREATE PROCEDURE UpdateShelterInfo (
141   IN ShelterIDParam INT,
142   IN Newname VARCHAR(255),

```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Result Grid | Form Editor | Read Only

ParticipantName
Joy
Ananya

Result 45 ×

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.

The screenshot shows the MySQL Workbench interface with a query editor window. The code in the editor is:

```
135 WHERE AdoptionEvents.EventID = 3;
136
137
138 DELIMITER //
139
140 • CREATE PROCEDURE UpdateShelterInfo (
141     IN ShelterIDParam INT,
142     IN NewName VARCHAR(255),
143     IN NewLocation VARCHAR(255)
144 )
145 BEGIN
146     UPDATE Shelters
147     SET Name = NewName,
148         Location = NewLocation
149     WHERE ShelterID = ShelterIDParam;
150 END //
151
152 DELIMITER ;
```

The status bar at the bottom indicates the command was successful: `SP/PROC < Name AS SheltersName, IFNULL(SUM(DonationAmount), 0) AS TotalDonationAmount`.

The output pane shows the execution log:

#	Time	Action	Message	Duration / Fetch
103	12:05:41	SELECT DATE_FORMAT(DonationDate, "M YY") AS MonthYear, IFNULL(SUM(DonationAmount), 0) AS TotalDonationAmount	1 row(s) returned	0.000 sec / 0.000 sec
104	12:06:31	SELECT Name, Age, Breed, Type FROM Pets WHERE OwnerID IS NULL LIMIT 0, 1000	7 row(s) returned	0.000 sec / 0.000 sec
105	12:07:28	SELECT s.Name AS ShelterName, IFNULL(SUM(d.DonationAmount), 0) AS TotalDonationAmount FROM Shelters s LEFT JOIN DONATIONS d ON s.ShelterID = d.ShelterID GROUP BY s.Name	5 row(s) returned	0.000 sec / 0.000 sec
106	12:08:52	CREATE PROCEDURE UpdateShelterInfo ( IN ShelterIDParam INT, IN NewName VARCHAR(255), IN NewLocation VARCHAR(255) )	Error Code: 1304. PROCEDURE UpdateShelterInfo already exists	0.000 sec
107	12:08:52	DROP PROCEDURE IF EXISTS UpdateShelterInfo;	0 row(s) affected	0.031 sec
108	12:08:57	CREATE PROCEDURE UpdateShelterInfo ( IN ShelterIDParam INT, IN NewName VARCHAR(255), IN NewLocation VARCHAR(255) )	0 row(s) affected	0.015 sec

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.

The screenshot shows the MySQL Workbench interface with a query editor window. The code in the editor is:

```
144
145 BEGIN
146     UPDATE Shelters
147     SET Name = NewName,
148         Location = NewLocation
149     WHERE ShelterID = ShelterIDParam;
150 END //
151
152 DELIMITER ;
153
154 • SELECT s.Name AS ShelterName, IFNULL(SUM(d.DonationAmount), 0) AS TotalDonationAmount
155     FROM Shelters s
156     LEFT JOIN Donations d ON s.Name = d.DonorName
157     GROUP BY s.Name;
158
159 • CREATE TABLE Owners (
160     OwnerID INT AUTO_INCREMENT PRIMARY KEY,
161     Name VARCHAR(255) NOT NULL,
162     Email VARCHAR(255)
163 );
```

The status bar at the bottom indicates the command was successful: `SP/PROC < Name AS SheltersName, IFNULL(SUM(d.DonationAmount), 0) AS TotalDonationAmount`.

The results grid shows the data:

ShelterName	TotalDonationAmount
Happy Shelter	0.00
Glory Shelter	0.00
Joy Shelter	0.00
Fluffy Shelter	0.00
Cute Shelter	0.00

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.

MySQL Workbench

File Edit View Query Database Server Tools Scripting Help

Query 1

```
177 SET p.OwnerID = o.OwnerID
178 WHERE p.PetID IN (1, 3, 5, 7, 9);
179
180 • SELECT Name, Age, Breed, Type
181   FROM Pets
182 WHERE OwnerID IS NULL;
183
184
185
186
187
188
189
190
191
192
193
194
195
196
```

Result Grid | Filter Rows: Export: Wrap Cell Content: Result Grid Form Editor Read Only

Name	Age	Breed	Type
Whiskers	2	Siamese	Cat
Snowball	4	Persian	Cat
Luna	1	Maine Coon	Cat
Charlie	3	Bulldog	Dog
Milo	2	Ragdoll	Cat
Bailey	6	Beagle	Dog
Oliver	3	Russian Blue	Cat

Pets 43 x

34°C Sunny Search ENG IN 12:06 18-04-2024

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 Workbench

File Edit View Query Database Server Tools Scripting Help

Query 1

```
176 JOIN Owners o ON p.PetID = o.OwnerID
177 SET p.OwnerID = o.OwnerID
178 WHERE p.PetID IN (1, 3, 5, 7, 9);
179
180 • SELECT Name, Age, Breed, Type
181   FROM Pets
182 WHERE OwnerID IS NULL;
183
184 • SELECT DATE_FORMAT(DonationDate, 'MM YY') AS MonthYear,
185   IFNULL(SUM(DonationAmount), 0) AS TotalDonationAmount
186   FROM Donations
187   GROUP BY DATE_FORMAT(DonationDate, 'MM YY')
188   ORDER BY MonthYear;
189
190
191
192
193
194
195
```

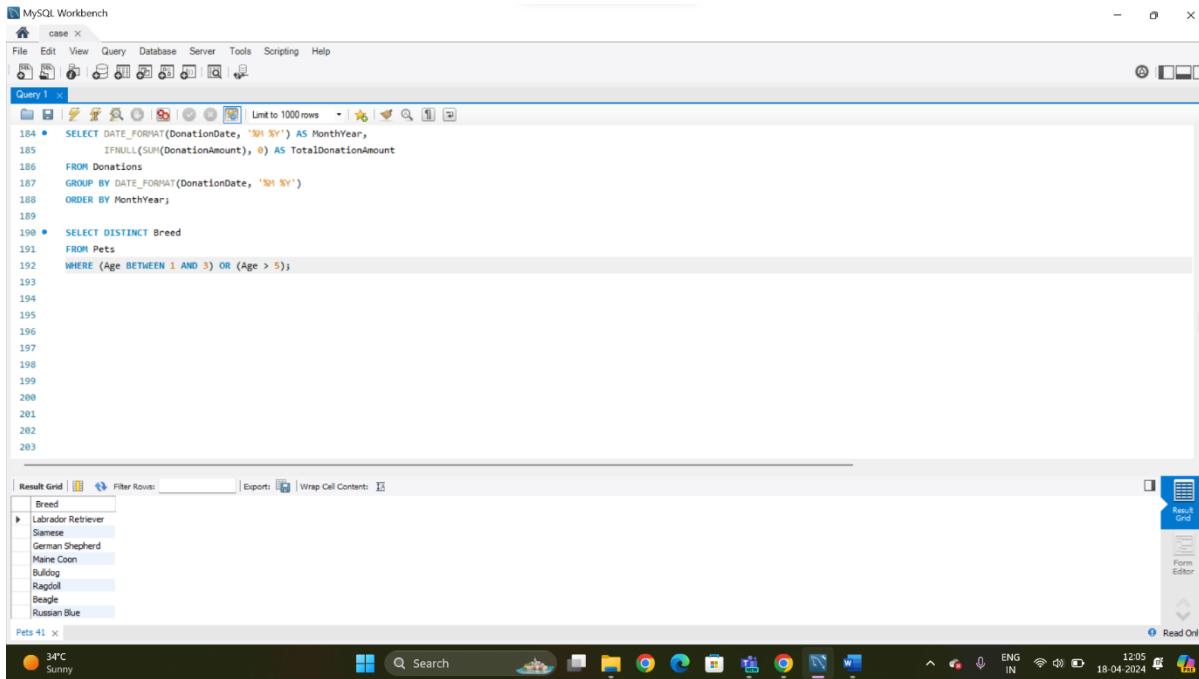
Result Grid | Filter Rows: Export: Wrap Cell Content: Result Grid Form Editor Read Only

MonthYear	TotalDonationAmount
April 2024	725.00

Result 42 x

34°C Sunny Search ENG IN 12:05 18-04-2024

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 Workbench - Query 1.x

File Edit View Query Database Server Tools Scripting Help
Query Grid | Filter Rows: | Export: | Wrap Cell Content: |
Result Grid | Result Grid | Form Editor | Read Only
Breed
Labrador Retriever
Siamese
German Shepherd
Maine Coon
Bulldog
Ragdoll
Beagle
Russian Blue

```

The screenshot shows the MySQL Workbench interface with a query editor containing the following SQL code:

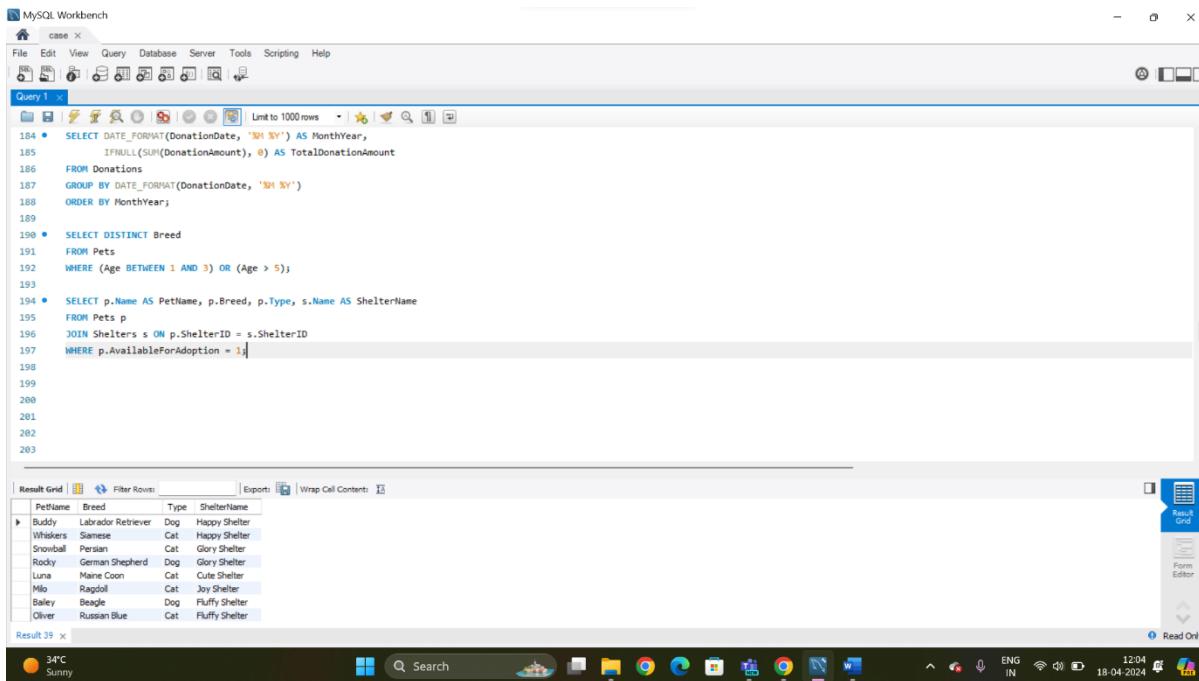
```

184 • SELECT DATE_FORMAT(DonationDate, 'MM YY') AS MonthYear,
185     IFNULL(SUM(DonationAmount), 0) AS TotalDonationAmount
186 FROM Donations
187 GROUP BY DATE_FORMAT(DonationDate, 'MM YY')
188 ORDER BY MonthYear;
189
190 • SELECT DISTINCT Breed
191 FROM Pets
192 WHERE (Age BETWEEN 1 AND 3) OR (Age > 5);
193
194
195
196
197
198
199
200
201
202
203

```

The results grid displays a list of distinct breeds from the Pets table, filtered by age (either between 1 and 3 years or older than 5 years). The breeds listed are Labrador Retriever, Siamese, German Shepherd, Maine Coon, Bulldog, Ragdoll, Beagle, and Russian Blue.

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



```

MySQL Workbench - Query 1.x

File Edit View Query Database Server Tools Scripting Help
Query Grid | Filter Rows: | Export: | Wrap Cell Content: |
Result Grid | Result Grid | Form Editor | Read Only
PetName Breed Type ShelterName
Buddy Labrador Retriever Dog Happy Shelter
Whiskers Cat Happy Shelter
Gumball Persian Cat Glory Shelter
Rocky German Shepherd Dog Glory Shelter
Luna Maine Coon Cat Cute Shelter
Milo Ragdoll Cat Joy Shelter
Bailey Beagle Dog Fluffy Shelter
Oliver Russian Blue Cat Fluffy Shelter

```

The screenshot shows the MySQL Workbench interface with a query editor containing the following SQL code:

```

184 • SELECT DATE_FORMAT(DonationDate, 'MM YY') AS MonthYear,
185     IFNULL(SUM(DonationAmount), 0) AS TotalDonationAmount
186 FROM Donations
187 GROUP BY DATE_FORMAT(DonationDate, 'MM YY')
188 ORDER BY MonthYear;
189
190 • SELECT DISTINCT Breed
191 FROM Pets
192 WHERE (Age BETWEEN 1 AND 3) OR (Age > 5);
193
194 • SELECT p.Name AS PetName, p.Breed, p.Type, s.Name AS ShelterName
195 FROM Pets p
196 JOIN Shelters s ON p.ShelterID = s.ShelterID
197 WHERE p.AvailableForAdoption = 1;
198
199
200
201
202
203

```

The results grid displays a list of pets and their respective shelters, filtered by availability for adoption (AvailableForAdoption = 1). The pets listed are Buddy, Whiskers, Gumball, Rocky, Luna, Milo, Bailey, and Oliver, each associated with a specific breed, type, and shelter name.

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

The screenshot shows the MySQL Workbench interface with a query editor and a results grid. The query retrieves the total number of participants for events organized by shelters located in Ooty. The results show a single row with a value of 2.

```
192 WHERE (Age BETWEEN 1 AND 3) OR (Age > 5);
193
194 • SELECT p.Name AS PetName, p.Breed, p.Type, s.Name AS ShelterName
195 FROM Pets p
196 JOIN Shelters s ON p.ShelterID = s.ShelterID
197 WHERE p.AvailableForAdoption = 1;
198
199 • SELECT COUNT(*) AS TotalParticipants
200 FROM Participants p
201 JOIN AdoptionEvents ae ON p.EventID = ae.EventID
202 JOIN Shelters s ON ae.Location = s.Location
203 WHERE s.Location = 'ooty';
204
205
206
207
208
209
210
211
```

TotalParticipants
2

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

The screenshot shows the MySQL Workbench interface with a query editor and a results grid. The query retrieves a list of unique breeds for pets with ages between 1 and 5 years. The results show a list of breeds: Labrador Retriever, Siamese, Golden Retriever, Persian, German Shepherd, Maine Coon, Bulldog, and Ragdoll.

```
199 • SELECT COUNT(*) AS TotalParticipants
200 FROM Participants p
201 JOIN AdoptionEvents ae ON p.EventID = ae.EventID
202 JOIN Shelters s ON ae.Location = s.Location
203 WHERE s.Location = 'ooty';
204
205 • SELECT DISTINCT Breed
206 FROM Pets
207 WHERE Age BETWEEN 1 AND 5;
208
209
210
211
212
213
214
215
216
217
218
```

Breed
Labrador Retriever
Siamese
Golden Retriever
Persian
German Shepherd
Maine Coon
Bulldog
Ragdoll

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

MySQL Workbench

case x

File Edit View Query Database Server Tools Scripting Help

Query 1.x

Unit to 1000 rows

```
199 • SELECT COUNT(*) AS TotalParticipants
200   FROM Participants p
201  JOIN AdoptionEvents ae ON p.EventID = ae.EventID
202  JOIN Shelters s ON ae.Location = s.Location
203 WHERE s.Location = 'city';
204
205 • SELECT DISTINCT Breed
206   FROM Pets
207 WHERE Age BETWEEN 1 AND 5;
208
209 • SELECT *
210   FROM Pets
211 WHERE OwnerID IS NULL;
212
213
214 • CREATE TABLE adoptedpets (
215   AdoptedPetID INT AUTO_INCREMENT PRIMARY KEY,
216   PetID INT,
217   OwnerID INT,
218   FOREIGN KEY (PetID) REFERENCES Pets(PetID),
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: |

PetID	Name	Age	Breed	Type	AvailableForAdoption	OwnerID	ShelterID
2	Whiskers	2	Siamese	Cat	1	NULL	1
4	Snowball	4	Persian	Cat	1	NULL	2
6	Luna	1	Maine Coon	Cat	1	NULL	5
7	Charlie	3	Bulldog	Dog	0	NULL	3
8	Milo	2	Ragdoll	Cat	1	NULL	3
9	Bailey	6	Beagle	Dog	1	NULL	4
10	Oliver	3	Russian Blue	Cat	1	NULL	4
11	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Pets 36 x

34°C Sunny

Search

12:02 18-04-2024 ENG IN

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

MySQL Workbench

case

File Edit View Query Database Server Tools Scripting Help

Query 1

211 WHERE OwnerID IS NULL;

212

213

214 • CREATE TABLE adoptedpets (

215     AdoptedPetID INT AUTO\_INCREMENT PRIMARY KEY,

216     PetID INT,

217     OwnerID INT,

218     FOREIGN KEY (PetID) REFERENCES Pets(PetID),

219     FOREIGN KEY (OwnerID) REFERENCES Owners(OwnerID)

220 );

221

222 • INSERT INTO adoptedpets (PetID, OwnerID)

223     SELECT PetID, OwnerID

224     FROM Pets

225     WHERE OwnerID IS NOT NULL;

226

227 •     SELECT p.Name AS PetName, o.Name AS OwnerName

228     FROM adoptedpets ap

229     JOIN owners o ON ap.OwnerID = o.OwnerID

230     JOIN pets p ON ap.PetID = p.PetID;

---

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

PetName	OwnerName
Buddy	Rahul Sharma
Max	Aryan Gupta
Rocky	Vikram Verma

Result 35 x

34°C Sunny

Search

12:00 PM 18-04-2024 ENG IN

Read Only

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

MySQL Workbench

```

220    );
221
222 • INSERT INTO adoptedpets (PetID, OwnerID)
223     SELECT PetID, OwnerID
224     FROM Pets
225     WHERE OwnerID IS NOT NULL;
226
227 •     SELECT p.Name AS PetName, o.Name AS OwnerName
228     FROM adoptedpets ap
229     JOIN owners o ON ap.OwnerID = o.OwnerID
230     JOIN pets p ON ap.PetID = p.PetID;
231
232 •     SELECT s.Name AS ShelterName, COUNT(p.PetID) AS AvailablePetsCount
233     FROM Shelters s
234     LEFT JOIN Pets p ON s.ShelterID = p.ShelterID
235     WHERE p.AvailableForAdoption = 1 OR p.AvailableForAdoption IS NULL
236     GROUP BY s.ShelterID;
237
238 •     SELECT p1.Name AS Pet1Name, p1.Breed AS Breed, p2.Name AS Pet2Name, s.Name AS ShelterName
239     FROM Pets p1

```

Result Grid | Filter Rows | Export: | Wrap Cell Content: | Result Grid | Form Editor | Read Only

ShelterName	AvailablePetsCount
Happy Shelter	2
Glory Shelter	2
Joy Shelter	1
Fluffy Shelter	2
Cute Shelter	1

Result 34 x

34°C Sunny

File Edit View Query Database Server Tools Scripting Help

ENG IN 12:00 18-04-2024

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

MySQL Workbench

```

232 •     SELECT s.Name AS ShelterName, COUNT(p.PetID) AS AvailablePetsCount
233     FROM Shelters s
234     LEFT JOIN Pets p ON s.ShelterID = p.ShelterID
235     WHERE p.AvailableForAdoption = 1 OR p.AvailableForAdoption IS NULL
236     GROUP BY s.ShelterID;
237
238 •     SELECT p1.Name AS Pet1Name, p1.Breed AS Breed, p2.Name AS Pet2Name, s.Name AS ShelterName
239     FROM Pets p1
240     LEFT JOIN Pets p2 ON p1.ShelterID = p2.ShelterID AND p1.Breed = p2.Breed AND p1.PetID < p2.PetID
241     JOIN Shelters s ON p1.ShelterID = s.ShelterID
242     ORDER BY p1.ShelterID, p1.Breed, p1.PetID;
243
244 •     SELECT s.Name AS ShelterName, ae.EventName AS EventName
245     FROM Shelters s
246     CROSS JOIN AdoptionEvents ae;
247
248 •     SELECT s.Name AS ShelterName
249     FROM Shelters s
250     LEFT JOIN Pets p ON s.ShelterID = p.ShelterID
251     LEFT JOIN AdoptedPets ap ON p.PetID = ap.PetID

```

Result Grid | Filter Rows | Export: | Wrap Cell Content: | Result Grid | Form Editor | Read Only

Pet1Name	Breed	Pet2Name	ShelterName
Buddy	Labrador Retriever		Happy Shelter
Whiskers	Siamese		Happy Shelter
Rocky	German Shepherd		Glory Shelter
Snowball	Persian		Glory Shelter
Charlie	Bulldog		Joy Shelter
Milo	Ragdoll		Joy Shelter
Baley	Beagle		Fluffy Shelter
Oliver	Russian Blue		Fluffy Shelter

Result 33 x

Finance headline Fed there done t...

File Edit View Query Database Server Tools Scripting Help

ENG IN 11:59 18-04-2024

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

The screenshot shows the MySQL Workbench interface with a query editor and a results grid. The query retrieves shelter names and adoption event names from multiple tables using various joins. The results grid displays the data in two columns: ShelterName and EventName.

```
232 • SELECT s.Name AS ShelterName, COUNT(p.PetID) AS AvailablePetsCount
233   FROM Shelters s
234   LEFT JOIN Pets p ON s.ShelterID = p.ShelterID
235   WHERE p.AvailableForAdoption = 1 OR p.AvailableForAdoption IS NULL
236   GROUP BY s.ShelterID;
237
238 • SELECT p1.Name AS Pet1Name, p1.Breed AS Breed, p2.Name AS Pet2Name, s.Name AS ShelterName
239   FROM Pets p1
240   LEFT JOIN Pets p2 ON p1.ShelterID = p2.ShelterID AND p1.Breed = p2.Breed AND p1.PetID < p2.PetID
241   JOIN Shelters s ON p1.ShelterID = s.ShelterID
242   ORDER BY p1.ShelterID, p1.Breed, p1.PetID;
243
244 • SELECT s.Name AS ShelterName, ae.EventName AS EventName
245   FROM Shelters s
246   CROSS JOIN AdoptionEvents ae;
247
248 • SELECT s.Name AS ShelterName
249   FROM Shelters s
250   LEFT JOIN Pets p ON s.ShelterID = p.ShelterID
251   LEFT JOIN AdoptedPets ap ON p.PetID = ap.PetID
```

ShelterName	EventName
Joy Shelter	Cat Adoption Fair
Glory Shelter	Cat Adoption Fair
Happy Shelter	Cat Adoption Fair
Cute Shelter	Dog Meet-and-Greet
Fluffy Shelter	Dog Meet-and-Greet
Joy Shelter	Dog Meet-and-Greet
Glory Shelter	Dog Meet-and-Greet
Happy Shelter	Dog Meet-and-Greet
Cute Shelter	Pet Show & Meet

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

The screenshot shows the MySQL Workbench interface with a query editor and a results grid. The query retrieves shelter names and the count of adopted pets, then orders the results by the count in descending order and limits the result to one row. The results grid shows that the Happy Shelter has the highest count of adopted pets.

```
247
248 • SELECT s.Name AS ShelterName
249   FROM Shelters s
250   LEFT JOIN Pets p ON s.ShelterID = p.ShelterID
251   LEFT JOIN AdoptedPets ap ON p.PetID = ap.PetID
252   GROUP BY s.ShelterID
253   ORDER BY COUNT(ap.PetID) DESC
254   LIMIT 1;
```

ShelterName
Happy Shelter