



SEED

خوشحال خیرپختونخوا

atomcamp

SQL: Assignment No. 02

Course Instructor

Miss Mahnoor Fatima

Teaching Assistant

Miss Syeda Faryal

COHORT-1A DS

Submitted By: Tila Muhammad

Data Science & AI Boot Camp

Question: 01 List the names of all pet owners along with the names of their pets?

Solution:

The screenshot shows the SQL Server Enterprise Manager interface. On the left, the 'Schemas' pane is open, showing the 'atom' schema with tables 'employees', 'feedback', 'petowners', 'pets', and 'service_usage'. The 'petowners' table is selected, and its columns (OwnerID, Name, Surname, StreetAddress, City, State) are listed. The main pane displays a SQL query:

```
1 #SELECT * FROM atom.pets;
2 • SELECT petowners.name AS OwnerName, pets.name AS PetName
3 FROM petowners
4 JOIN pets ON petowners.OwnerID = pets.OwnerID
```

Below the query, the 'Result Grid' shows the output:

OwnerName	PetName
Robert	Blackie
Charles	Roomba
Ed	Simba
Paula	Keller
George	Cuddles
Florence	Vuitton
Bobbie	Priya
Wm	Simba
Edna	Cookie
Joe	Heisenberg
Rosa	Stowe
Elvis	Scout

The 'Output' pane at the bottom shows the execution details:

#	Time	Action	Message
1	22:44:28	SELECT petowners.name AS OwnerName, pets.name AS PetName from petowners join pets ON ...	100 row(s) returned

Question: 02 List all pets and their owner names, including pets that don't have recorded owners?

Solution:

The screenshot shows the SQL Server Enterprise Manager interface. The 'petowners' table is selected in the 'Schemas' pane. The main pane displays a SQL query:

```
1 #SELECT * FROM atom.petowners;
2 • SELECT pets.name AS PetName, petowners.name AS OwnerName
3 FROM pets
4 LEFT JOIN petowners ON pets.ownerid = petowners.ownerid;
```

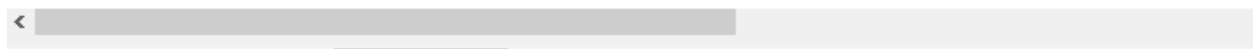
Below the query, the 'Result Grid' shows the output:





PetName	OwnerName
Blackie	Robert
Roomba	Charles
Simba	Ed
Keller	Paula
Cuddles	George
Vuitton	Florence
Priya	Bobbie
Simba	Wm
Cookie	Edna
Heisenberg	Joe

Question: 03 Combine the information of pets and their owners, including those pets without owners and owners without pets. Hint: We cannot use Full join in MySQL. We will use Left Join + union + Right Join?


Solution:


```
1  #SELECT * FROM atom.petowners;
2  • SELECT pets.name AS PetName, petowners.name AS OwnerName
3  FROM pets
4  LEFT JOIN petowners ON pets.ownerid = petowners.ownerid
5
6  UNION
7
8  SELECT pets.name AS PetName, petowners.name AS OwnerName
9  FROM petowners
10 LEFT JOIN pets ON petowners.ownerid = pets.ownerid
11 WHERE pets.ownerid IS NULL;
```



< 

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

	PetName	OwnerName
▶	Blackie	Robert
	Roomba	Charles
	Simba	Ed
	Keller	Paula
	Cuddles	George
	Vuitton	Florence

Result 3 x 

Output 

 Action Output 




#	Time	Action	Message
✓ 1	22:57:45	SELECT pets.name AS PetName, petowners.name AS OwnerName FROM pets LEFT JOIN peto...	99 row(s) returned

Question: 04 List all pet owners and the number of dogs they own?

Solution:

```
1 #SELECT * FROM atom.petowners;
2 • SELECT petowners.name, COUNT(pets.PetId) AS NumberOfDogs
3 FROM petowners
4 LEFT JOIN pets ON petowners.ownerid = pets.ownerid
5 WHERE pets.kind = 'Dog'
6 GROUP BY petowners.name;
```


<

Result Grid  Filter Rows: Export:  Wrap Cell Content: 

	name	NumberOfDogs
▶	Debbie	1
	John	1
	Connie	1
	Lena	1
	Jessica	1
	Bessie	1
	Luisa	1
	Karen	1
	Mary	1
	Susan	2

Result 2 x

Output



Action Output 

#	Time	Action	Message
✓ 1	23:01:01	SELECT petowners.name, COUNT(pets.PetId) AS NumberOfDogs FROM petowners LEFT JOIN ...	47 row(s) returned

Question: 05 Identify pets that have not had any procedures?

Solution:




```
1  #SELECT * FROM atom.petowners;
2  • SELECT p.PetId, p.name
3  FROM pets p
4  LEFT JOIN procedureshistory ph ON p.PetId = ph.PetId
5  WHERE ph.PetId IS NULL;
```

<		
Result Grid		
Filter Rows: <input type="text"/>		
Export:  Wrap Cell Content: 		
PetId	name	
Q0-2001	Roomba	
R3-7551	Keller	
Z4-5652	Priya	
Z4-4045	Simba	
J2-3320	Heisenberg	
T2-2142	Stowe	
H8-1429	Lily	

Question: 06 Find the name of the oldest pet?

Solution:

```
1  #SELECT * FROM atom.petowners;
2  • SELECT name
3  FROM pets
4  ORDER BY age DESC
5  LIMIT 1;
```

<		
Result Grid		
Filter Rows: <input type="text"/>		
Export:  Wrap Cell Content:  Fetch rows: 		
name		
▶	Stowe	

Question: 07 Find the details of procedures performed on 'Cuddles'?

Solution:

```
1 #SELECT * FROM atom.proceduresdetails;
2 • SELECT p.name AS PetName, ph.date, pd.ProcedureType, pd.ProcedureSubCode, pd.Description, pd.Price
3 FROM pets p
4 JOIN procedureshistory ph ON p.PetId = ph.PetId
5 JOIN proceduresdetails pd ON ph.proceduretype = pd.ProcedureType AND ph.proceduresubcode = pd.ProcedureSubCode
6 WHERE p.name = 'Cuddles';
```

	PetName	date	ProcedureType	ProcedureSubCode	Description	Price
▶	Cuddles	2016-01-30	ORTHOPEDIC	07	Pinning-I.M.	325
	Cuddles	2016-02-04	VACCINATIONS	05	Rabies	10
	Cuddles	2016-10-05	VACCINATIONS	05	Rabies	10

Result 2 x

Output

Action Output

#	Time	Action	Message
✓ 1	23:14:00	SELECT p.name AS PetName, ph.date, pd.ProcedureType, pd.ProcedureSubCode, pd.Descr...	3 row(s) returned

Question: 08 List the pets who have undergone a procedure called 'VACCINATIONS'?

Solution:



```
1 #SELECT * FROM atom.procedureshistory;
2 • SELECT p.name AS PetName
3 FROM pets p
4 JOIN procedureshistory ph ON p.PetId = ph.PetId
5 JOIN proceduresdetails pd ON ph.proceduretype = pd.ProcedureType AND ph.proceduresubcode = pd.ProcedureSubCode
6 WHERE pd.ProcedureType = 'VACCINATIONS';
```

	PetName
▶	Simba
	Cuddles
	Vuitton
	Cookie
	Scout
	Biscuit
	Cookie
	Bright
	Briat

Question: 09 Count the number of pets of each kind?

Solution:



```
1 #SELECT * FROM atom.procedureshistory
2 • SELECT Kind, COUNT(*) AS NumberOfPets
3 FROM pets
4 GROUP BY kind;
```

<		
Result Grid		
Filter Rows: <input type="text"/>		
Export:  Wrap Cell Content: 		
	Kind	NumberOfPets
▶	Dog	57
	Cat	31
	Parrot	12

Question: 10 Group pets by their kind and gender and count the number of pets in each group?

Solution:

```
1 #SELECT * FROM atom.procedureshistory
2 • SELECT kind, gender, COUNT(*) AS NumberOfPets
3 FROM pets
4 GROUP BY kind, gender;
```

<			
Result Grid			
Filter Rows: <input type="text"/>			
Export:  Wrap Cell Content: 			
	kind	gender	NumberOfPets
▶	Dog	male	35
	Cat	male	19
	Parrot	female	7
	Cat	female	12
	Dog	female	22
	Parrot	male	5

Question: 11 Show the average age of pets for each kind, but only for kinds that have more than 5 pets?

Solution:

```
1      #SELECT * FROM atom.procedureshistory
2 •    SELECT kind, AVG(age) AS AverageAge
3      FROM pets
4      GROUP BY kind
5      HAVING COUNT(PetId) > 5;
```

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
kind	AverageAge			
Dog	6.7895			
Cat	7.3226			
Parrot	6.5833			

Question: 12 Find the types of procedures that have an average cost greater than \$50.

Solution:

```
1      #SELECT * FROM atom.procedureshistory
2 •    SELECT ProcedureType, AVG(Price) AS AverageCost
3      FROM proceduresdetails
4      GROUP BY ProcedureType
5      HAVING AVG(Price) > 50;
```

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
ProcedureType	AverageCost			
OFFICE FEES	52.0000			
ORTHOPEDIC	196.3333			
GENERAL SURGERIES	312.5263			

Question: 13 Classify pets as 'Young', 'Adult', or 'Senior' based on their age. Age less than 3 Young, Age between 3 and 8 Adult, else senior?

Solution:

```
1  #SELECT * FROM atom.procedureshistory
2  • SELECT name,
3      CASE
4          WHEN age < 3 THEN 'Young'
5          WHEN age BETWEEN 3 AND 8 THEN 'Adult'
6          ELSE 'Senior'
7      END AS AgeGroup
8  FROM pets;
```

<

Result Grid

	name	AgeGroup
▶	Blackie	Senior
	Roomba	Senior
	Simba	Young
	Keller	Young
	Cuddles	Senior
	Vuitton	Senior
	Priya	Adult
	Simba	Young

Result 6 ×

Output

Action Output

#	Time	Action	Message
✓ 1	23:28:38	SELECT name, CASE WHEN age < 3 THEN 'Young' WHEN age BETWEEN...	100 row(s) returned

Question: 14 Show the gender of pets with a custom label ('Boy' for male, 'Girl' for female)?

Solution:

```
1  #SELECT * FROM atom.procedureshistory
2  • SELECT name,
3      CASE
4          WHEN gender = 'male' THEN 'Boy'
5          WHEN gender = 'female' THEN 'Girl'
6          ELSE 'Unknown'
7      END AS GenderLabel
8  FROM pets;
```

Result Grid	
Filter Rows:	Export: Wrap Cell Content:
name	GenderLabel
Blackie	Boy
Roomba	Boy
Simba	Boy
Keller	Girl
Cuddles	Boy
Vuitton	Girl
Priya	Girl
Simba	Boy

Question: 15 For each pet, display the pet's name, the number of procedures they've had, and a status label: 'Regular' for pets with 1 to 3 procedures, 'Frequent' for 4 to 7 procedures, and 'Super User' for more than 7 procedures?

Solution:

```
1      #SELECT * FROM atom.procedureshistory
2 •    SELECT p.name AS PetName,
3          COUNT(ph.PetId) AS NumberOfProcedures,
4          CASE
5              WHEN COUNT(ph.PetId) BETWEEN 1 AND 3 THEN 'Regular'
6              WHEN COUNT(ph.PetId) BETWEEN 4 AND 7 THEN 'Frequent'
7              WHEN COUNT(ph.PetId) > 7 THEN 'Super User'
8              ELSE 'None'
9          END AS StatusLabel
10     FROM pets p
11    LEFT JOIN procedureshistory ph ON p.PetId = ph.PetId
12    GROUP BY p.name;
```

Result Grid			
		Filter Rows:	
		Export:	Wrap Cell Content:
	PetName	NumberOfProcedures	StatusLabel
▶	Blackie	2	Regular
	Roomba	0	None
	Simba	1	Regular
	Keller	0	None
	Cuddles	3	Regular
	Vuitton	2	Regular