

SQL Assignment 2

Code:

Create database if not Exists assignment 2;
use assignment2;

Task 1:

List the names of all pet owners along with the names of their pets.

Inner Join

```
select pets.Name as pet_name,petowners.Name as Pet_Owner  
from petowners join pets on petowners.OwnerID=pets.OwnerID;
```

Task 2:

List all pets and their owner names, including pets that don't have recorded owners.

Left Join

```
select pets.Name as pet_name,petowners.Name as Pet_Owner  
from pets left join petowners on petowners.OwnerID=pets.OwnerID;
```

Task 3:

Combine the information of pets and their owners, including those pets without owners and owners without pets.

Full Join

```
select * from pets left join petowners on petowners.OwnerID=pets.OwnerID  
  
Union  
  
select * from pets right join petowners on petowners.OwnerID=pets.OwnerID;
```

Task 4:

List all pet owners and the number of dogs they own

Inner Join

```
select petowners.OwnerID,PetOwners.Name,Count(pets.Kind)
from petowners
join pets on petowners.OwnerID=pets.OwnerID AND pets.kind = 'Dog'
group by petowners.OwnerID,PetOwners.Name;
```

Task 5:

Identify pets that have not had any procedures

Left Join

```
select pets.Name,procedureshistory.proceduretype
from pets
left join procedureshistory on pets.PetID=procedureshistory.PetID
where proceduretype is null;
```

Task 6:

Find the name of the oldest pet.

Order By/Where

```
select Name,age from pets
where age='15'; /*Order By limit 3*/
```

Task 7:

Find the details of procedures performed on 'Cuddles'.

Inner Join/Where

```
select pets.name,procedureshistory.procedureType
from pets
join procedureshistory on pets.PetID=procedureshistory.PetID
where pets.name="Cuddles";
```

Task 8:

List the pets who have undergone a procedure called 'VACCINATIONS'.

Inner Join/Where

```
select pets.name,procedureshistory.procedureType
from pets
join procedureshistory on pets.PetID=procedureshistory.PetID
where procedureshistory.procedureType="VACCINATIONS";
```

Task 9:

Count the number of pets of each kind.

Group By

```
select Kind,Count(Kind)
from pets
group by Kind;
```

Task 10:

Group pets by their kind and gender and count the number of pets in each group.

Group By

```
select * from pets;
select gender,Kind,Count(*)as total_Pets
from pets
group by gender,Kind;
```

Task 11:

Show the average age of pets for each kind, but only for kinds that have more than 5 pets.

Group By/Having

```
select Kind,avg(age)
from pets
group by Kind
```

Having count(Name)>5;

Task 12:

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

Group By /Having/Aggragates

```
select * from proceduresdetails;
```

```
select procedureType,avg(Price)
```

```
from proceduresdetails
```

```
group by procedureType
```

```
Having avg(Price)>50;
```

Task 13:

Classify pets as 'Young', 'Adult', or 'Senior' based on their age. Age less then 3 Young, Age between 3 and 8 Adult, else Senior.

CASE

```
select * from pets;
```

```
Select *,
```

```
CASE
```

```
    When Age<3 Then 'Young'
```

```
    When Age between 3 and 8 Then 'Adult'
```

```
    When Age> 8 Then 'Senior'
```

```
    End as classification
```

```
from pets;
```

Task 14:

Show the gender of pets with a custom label ('Boy' for male, 'Girl' for female).

UPDATE

```
SET SQL_SAFE_UPDATES=0;
```

```
update  pets set Gender="Boy" where Gender='male';
```

```
update pets set Gender="Girl" where Gender='female';
```

Task 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.

CASE/LEFT JOIN/GROUP BY

```
select pets.Name,count(procedureshistory.procedureType) as Procedures_Count,  
  
CASE  
  
when count(procedureshistory.procedureType) between 1 and 3 then 'Regular'  
  
when count(procedureshistory.procedureType) between 4 and 7 then 'Frequent'  
  
when count(procedureshistory.procedureType) >7 then 'Super User'  
  
end as status_label  
  
from pets  
  
left join procedureshistory on pets.petID=procedureshistory.petID  
  
group by procedureType,pets.Name  
  
;
```