ASSIGNMENT-3

SQL:

- Structured Query Language, is designed for managing and manipulating relational database systems.
- SQL is used for tasks such as querying data, updating data, inserting data, and managing database structures.

Types of SQL commands:

1) DDL (Data Definition Language)
2) DML (Data Manipulation Language)
3) DCL (Data Control Language)
4) TCL (Transcation Control Language)

1) Data Definition Language:

- DDL changes the structure of the table like creating a table, deleting a table, altering a table, etc.,
- All the command of DDL are auto-committed that means it permanently save all the changes in the database.
- DDL commands are CREATE, ALTER, DROP, TRUNCATE

2) Data Manipulation Language:

- DML commands are used to modify the database. It is responsible for all form of changes in the database.
- The command of DML is not auto-committed that means it can't permanently save all the changes in the database. They can be rollback.
- DML commands are INSERT, UPDATE, DELETE

3) Data Control Language:

- DCL commands are used to grant and take back authority from any database user.
- DCL commands are GRANT, REVOKE
- GRANT- It is used to give user access privileges to a database.
- REVOKE- It is used to take back permissions from the user.

4) Transaction Control Language:

- TCL commands can only use with DML commands like INSERT, DELETE and UPDATE only.
- These operations are automatically committed in the database that's why they cannot be used while creating tables or dropping them.
- TCL commands are COMMIT, ROLLBACK , SAVEPOINT

SQL Queries:

1) Create database:

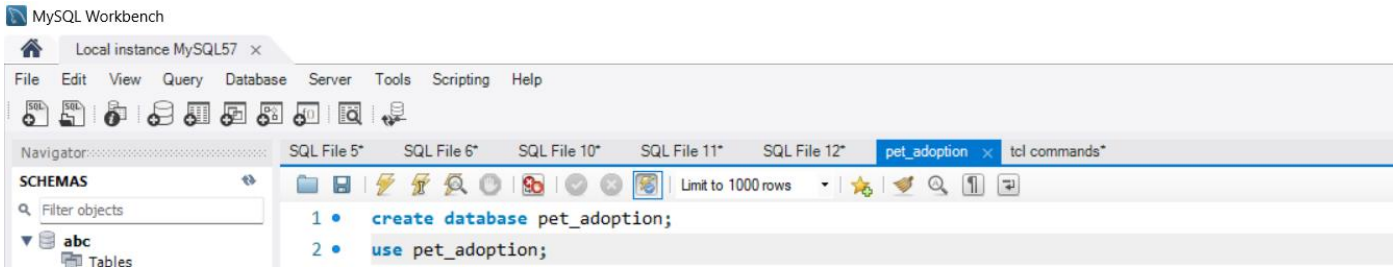
```
create database pet_adoption;
```

It creates the database named pet_adoption

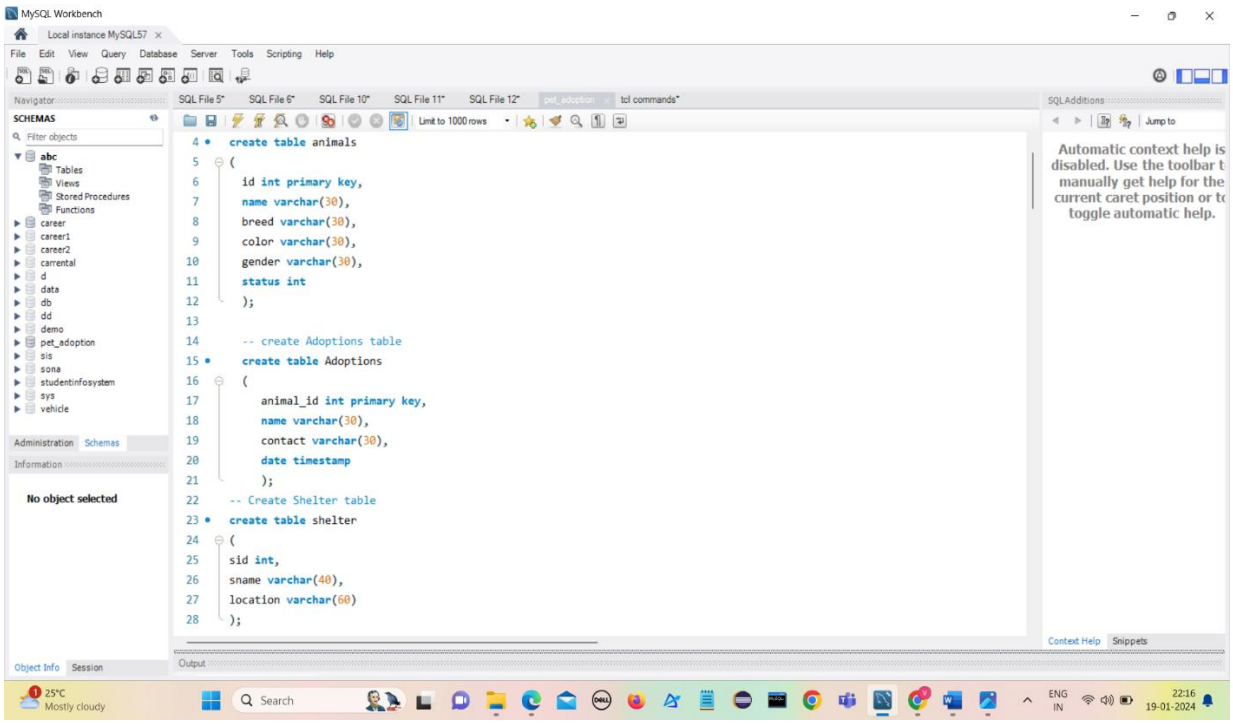
2) Use database:

```
use pet_adoption;
```

It makes pet_adoption database as the current working database on which operations are performed.



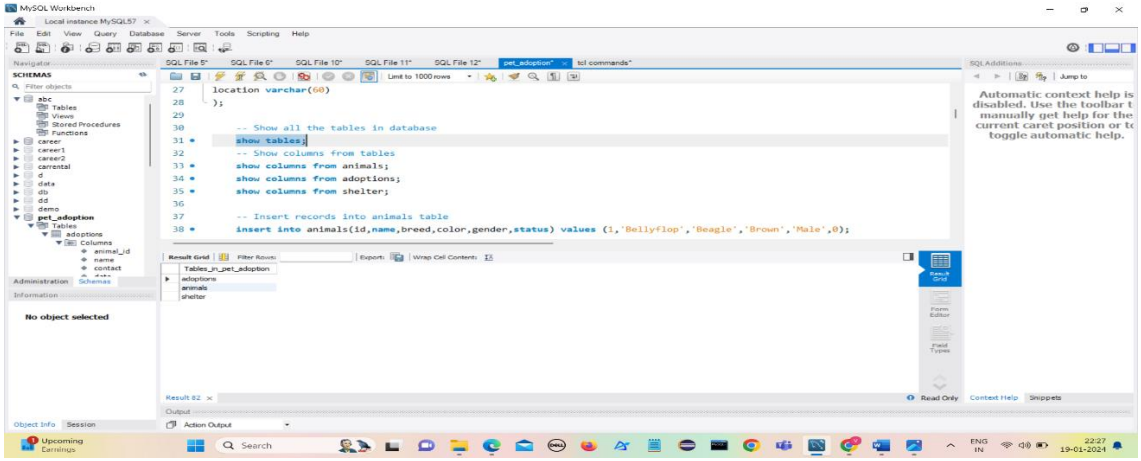
3) Create tables:



create command creates animals table, adoptions table and shelter table with different attributes.

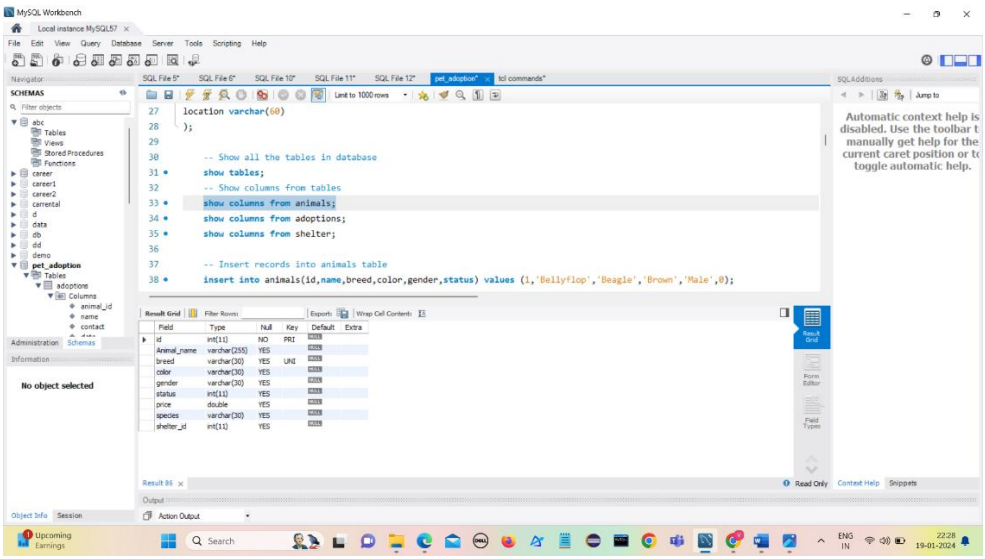
4) Show tables in the database

It shows all the tables in the database



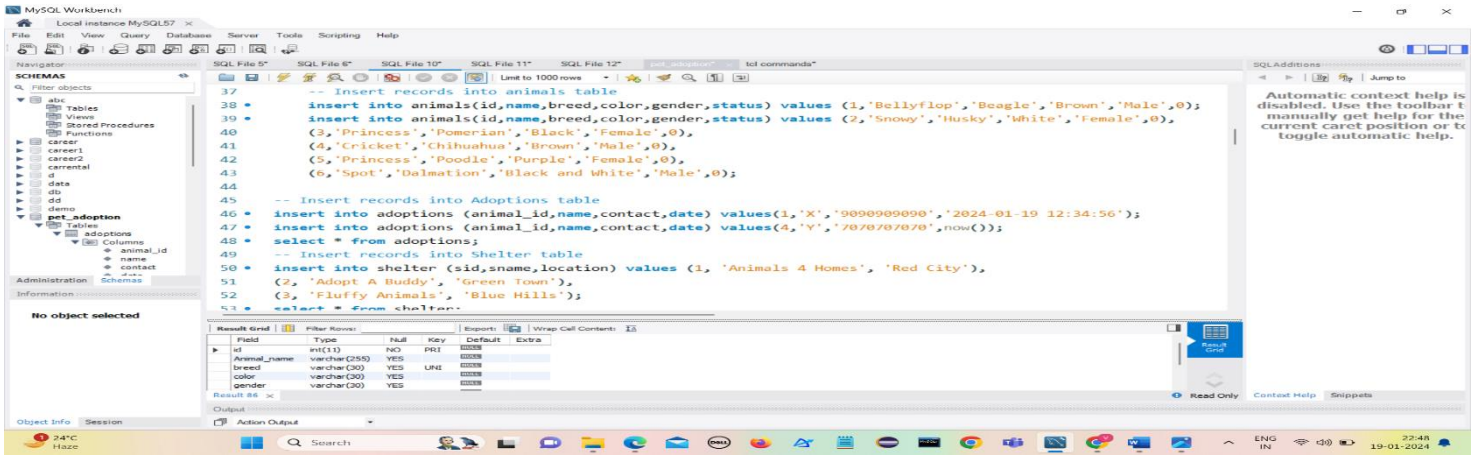
5) Show columns from tables

It displays all the columns in the particular table.

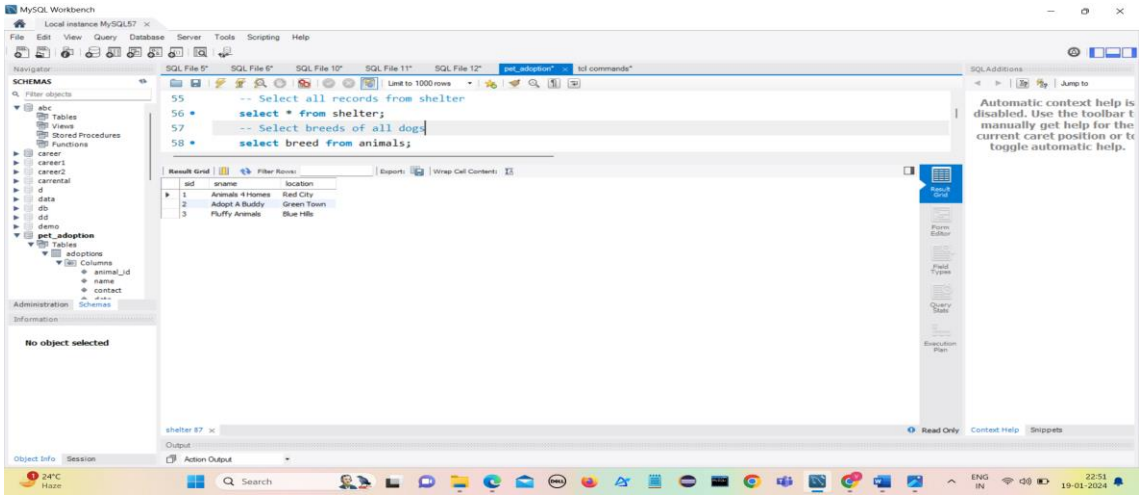


6) INSERT STATEMENT

Insert records into animals, adoptions and shelter table.

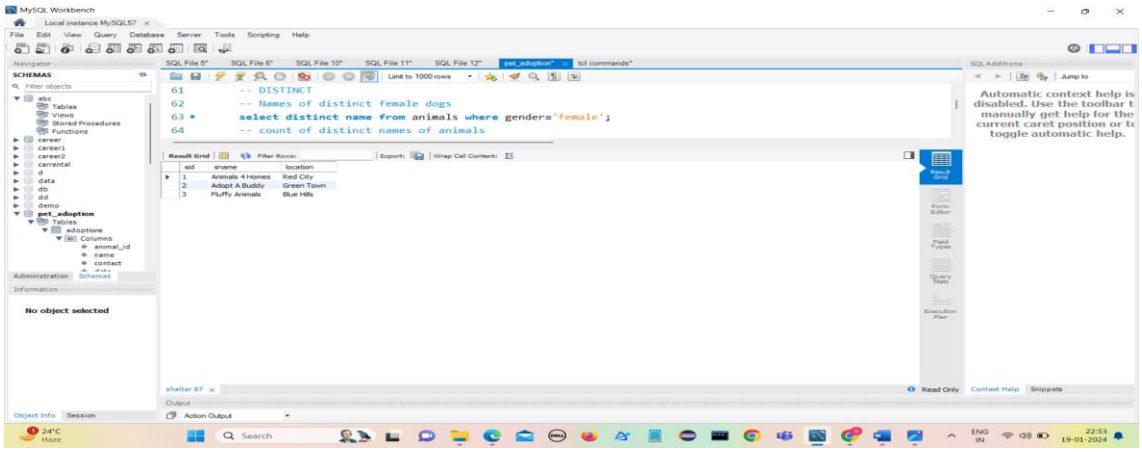


7) Select statement



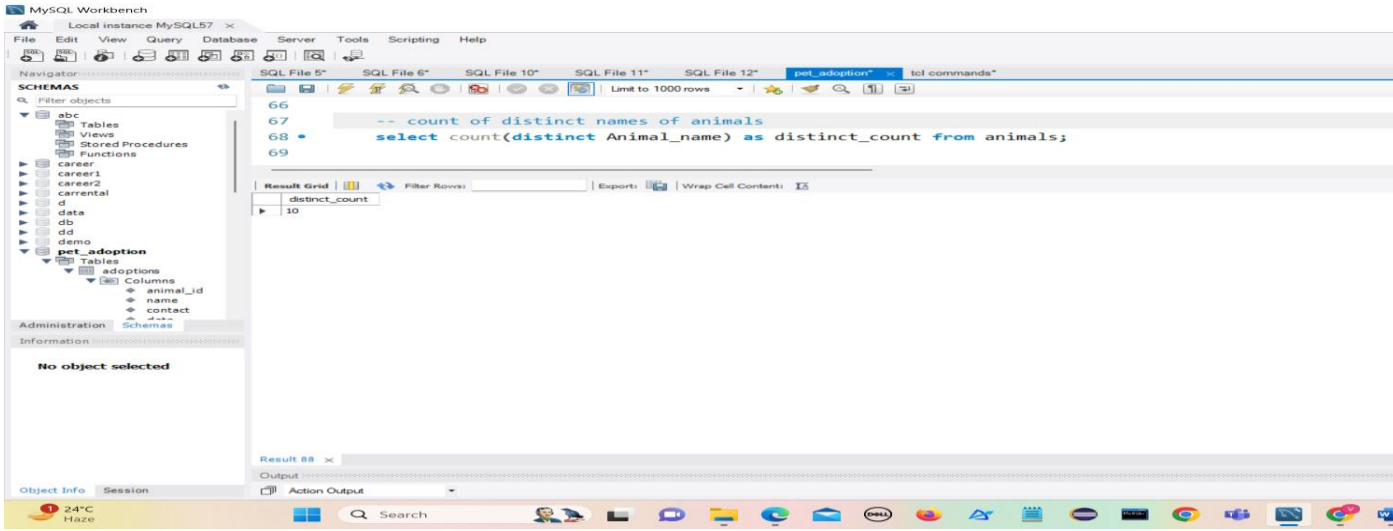
This statement selects all the records from shelter table

8) SELECT DISTINCT



It selects all the distinct values i.e., distinct names of all female dogs

9) Count of distinct names of animals



This query gives the count of distinct animal names from animals table.

10) ALTER COMMANDS

-- ALTER command ADD column

```
alter table animals add price float;
```

It adds a column price in animals table

-- ALTER command change column datatype

```
alter table animals modify column price double;
```

It changes the data type of price from float to double

-- ALTER command add constraints on column

```
alter table animals add constraint  unique_constraint unique(breed);
```

It adds a unique constraint to breed column which doesnot allow repeated values

-- ALTER command DROP column

```
alter table animals drop column price;
```

It drops the price column from animals table

-- ALTER command RENAME column

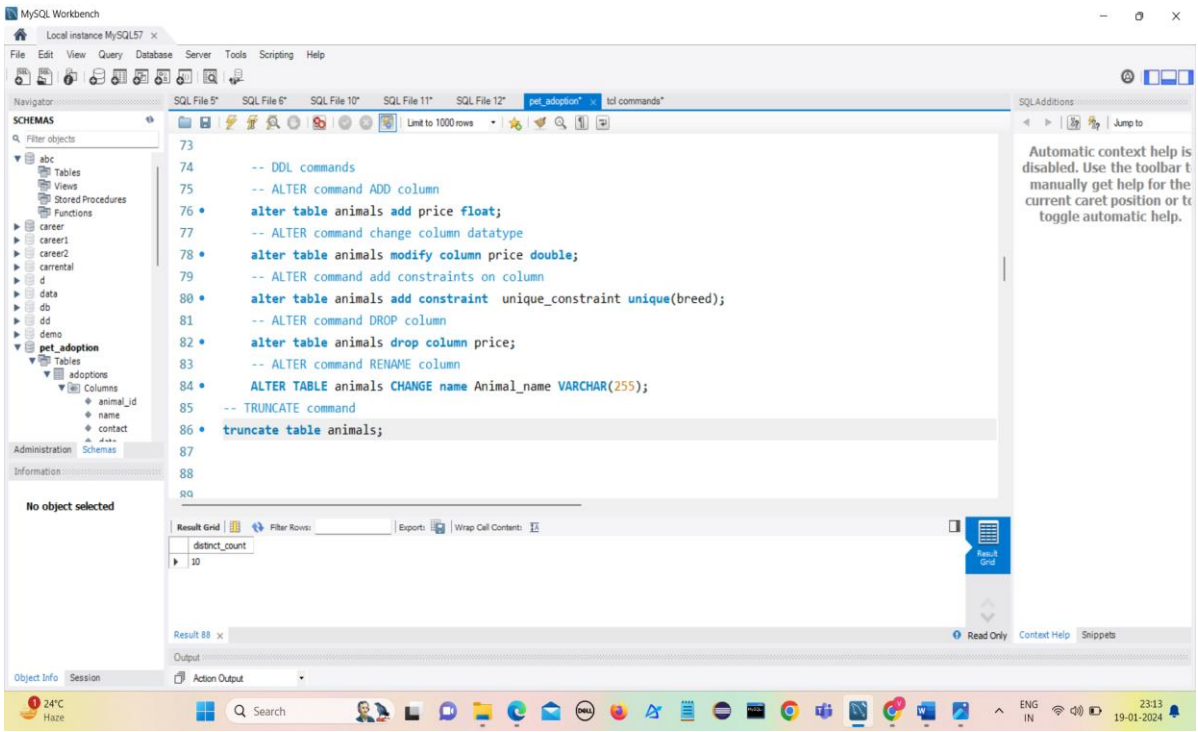
```
alter table animals CHANGE name Animal_name VARCHAR(255);
```

It renames the name in animals table to Animal_name

-- TRUNCATE command

```
truncate table animals;
```

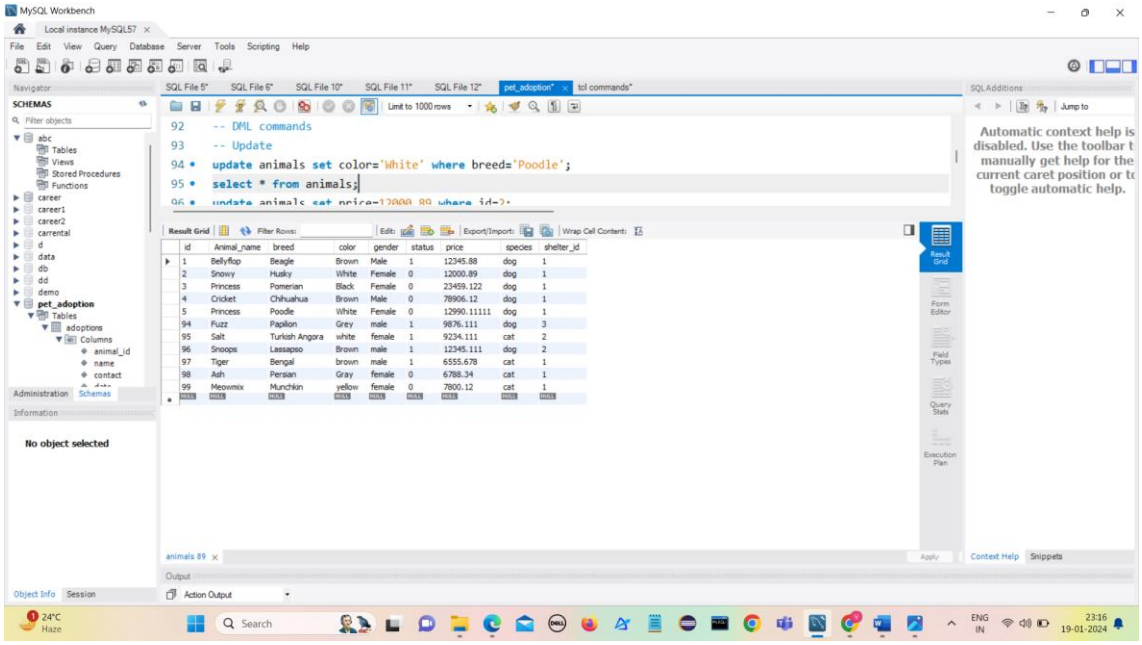
It truncates animals table



11) DML Commands

-- Update statement

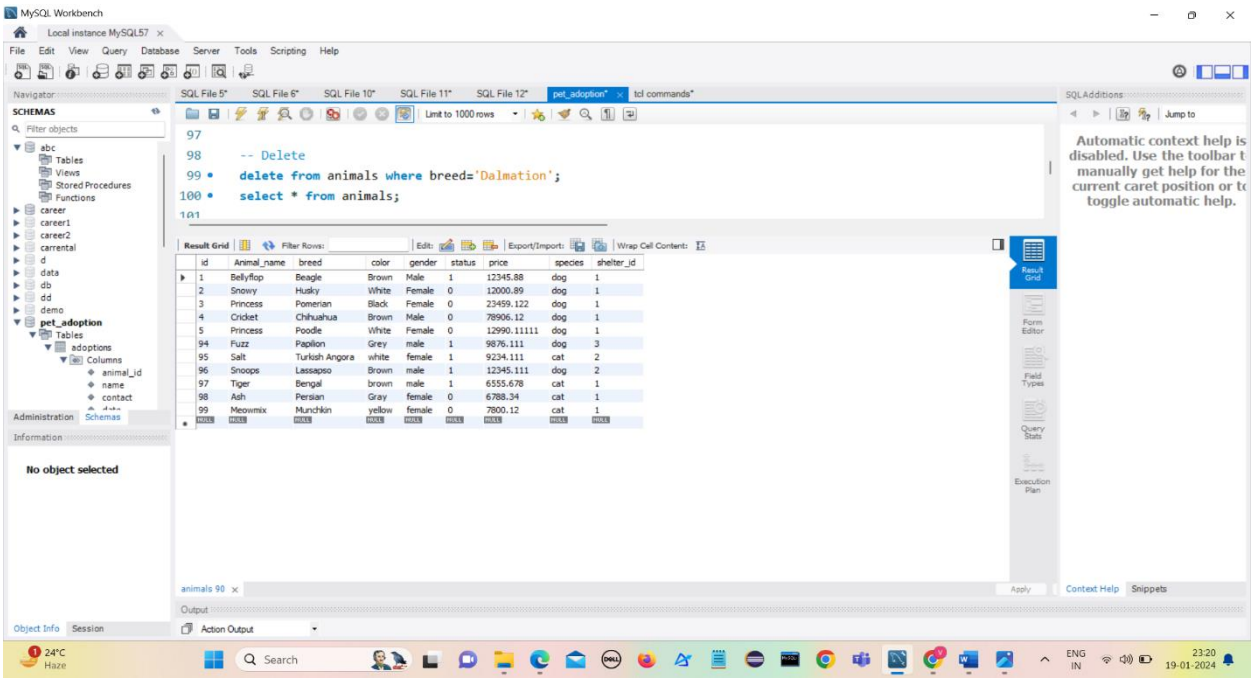
update animals set color='White' where breed='Poodle';



It updates the color of animal to White whose breed is Poodle.

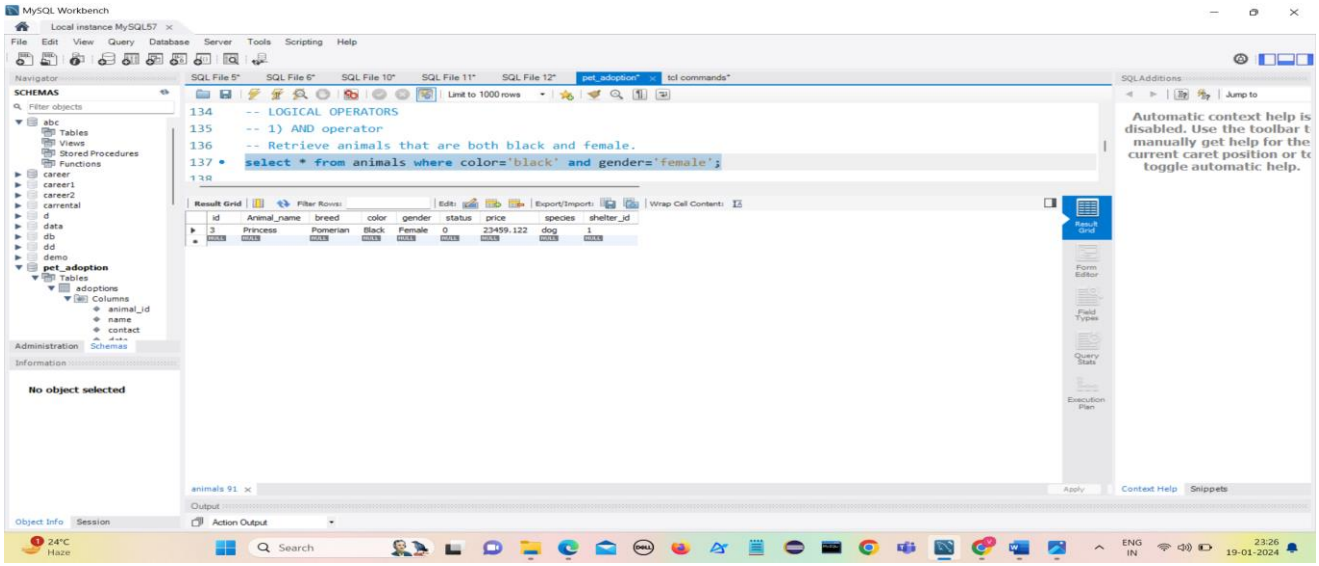
-- Delete statement

delete from animals where breed='Dalmation';It deletes animal whose breed is Dalmation from animals table



12) LOGICAL OPERATORS

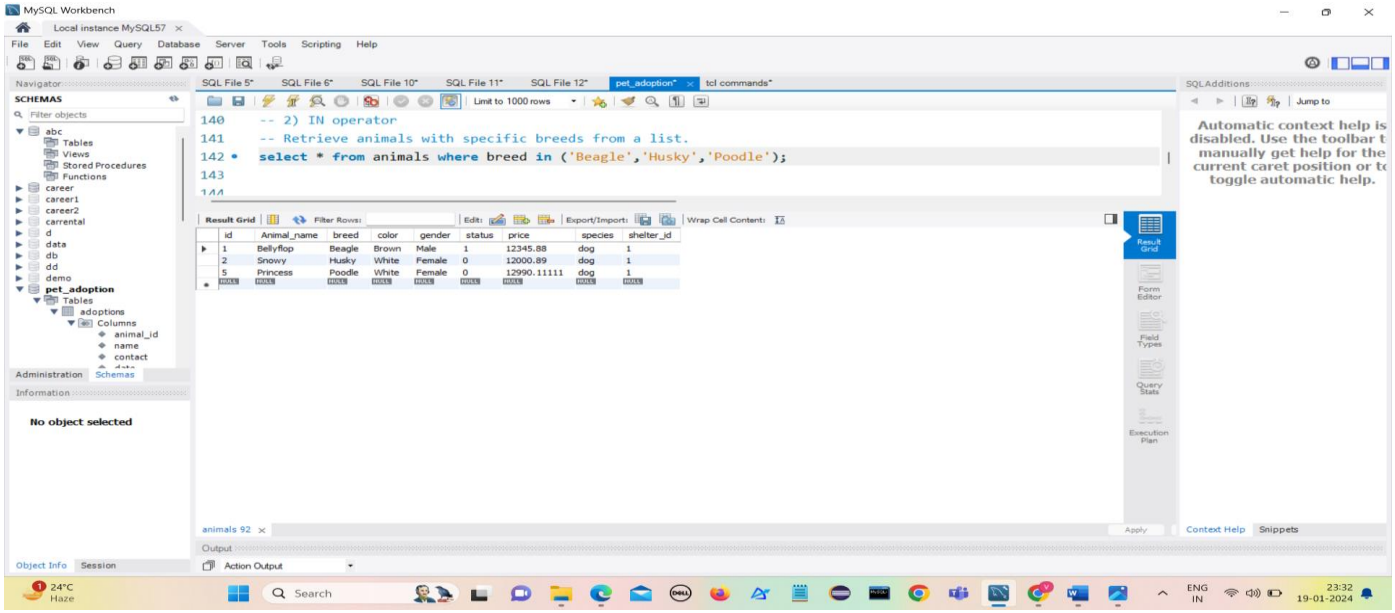
1) AND operator



This select statement retrieves animals that are both black and female. AND operator returns records if both conditions are true. It returns TRUE if both Boolean expressions are TRUE.

2) IN operator

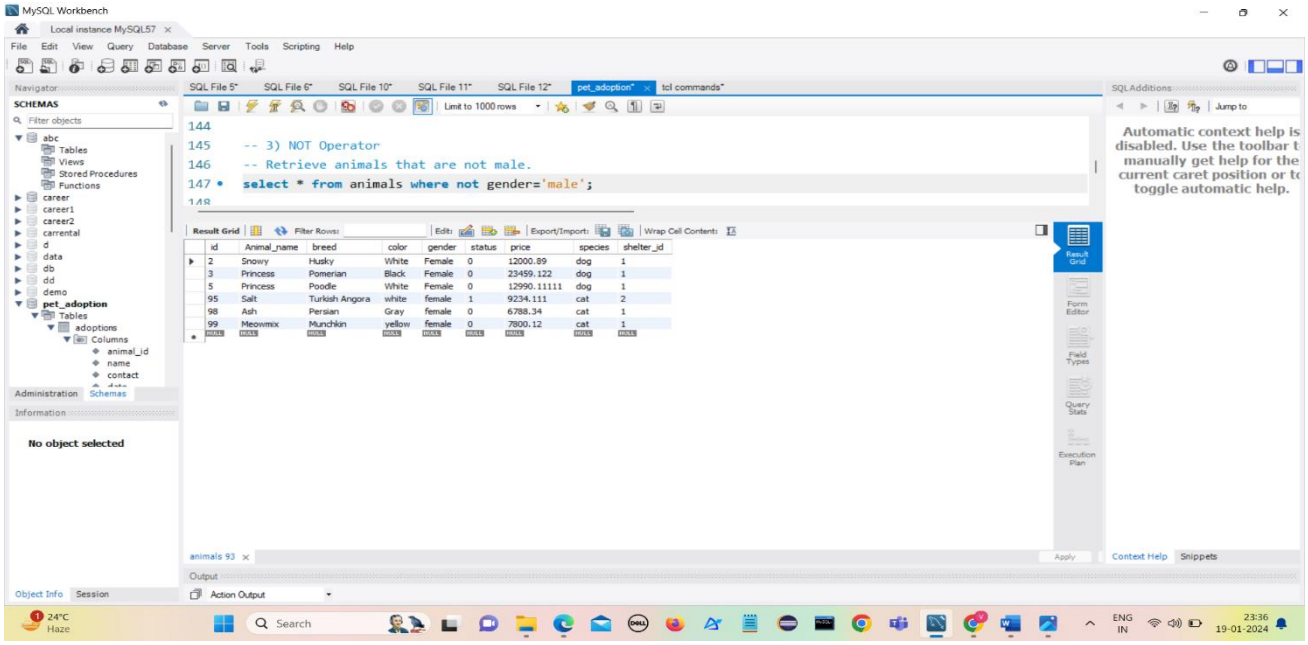
Retrieve animals with specific breeds from a list.



This select statement returns all the animals whose breed is in Beagle or Husky or Poodle. It returns TRUE if the operand is equal to one of a list of expressions.

3) **NOT operator**

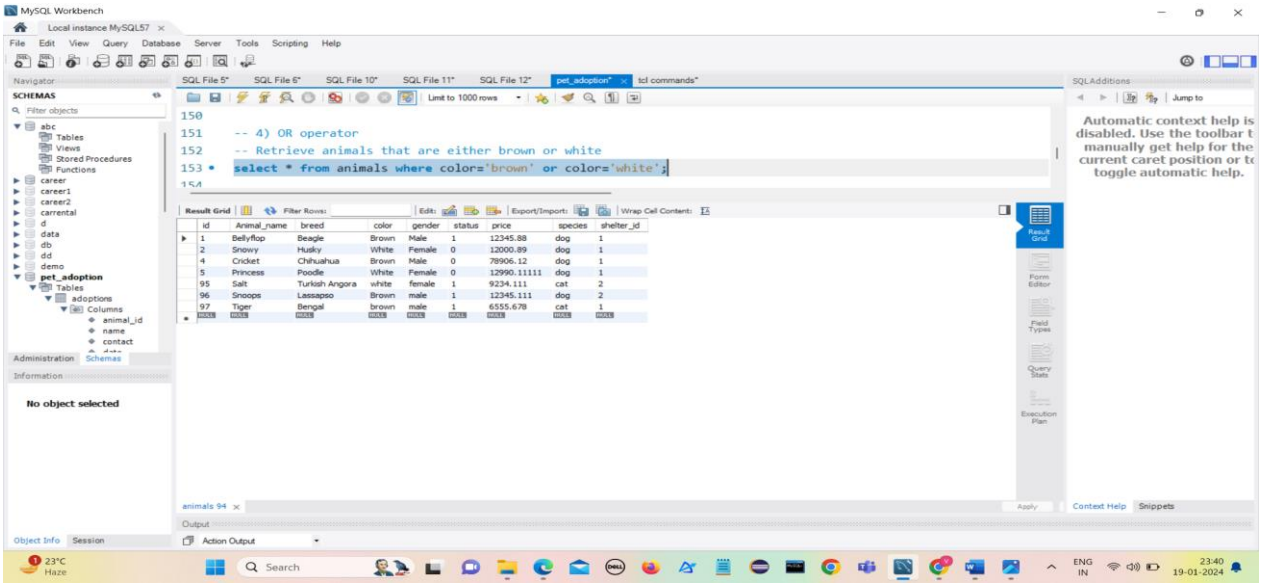
Retrieve animals that are not male.



It returns all the animals that are not male i.e., female. It Reverses the value of any other Boolean operator.

4) **OR operator**

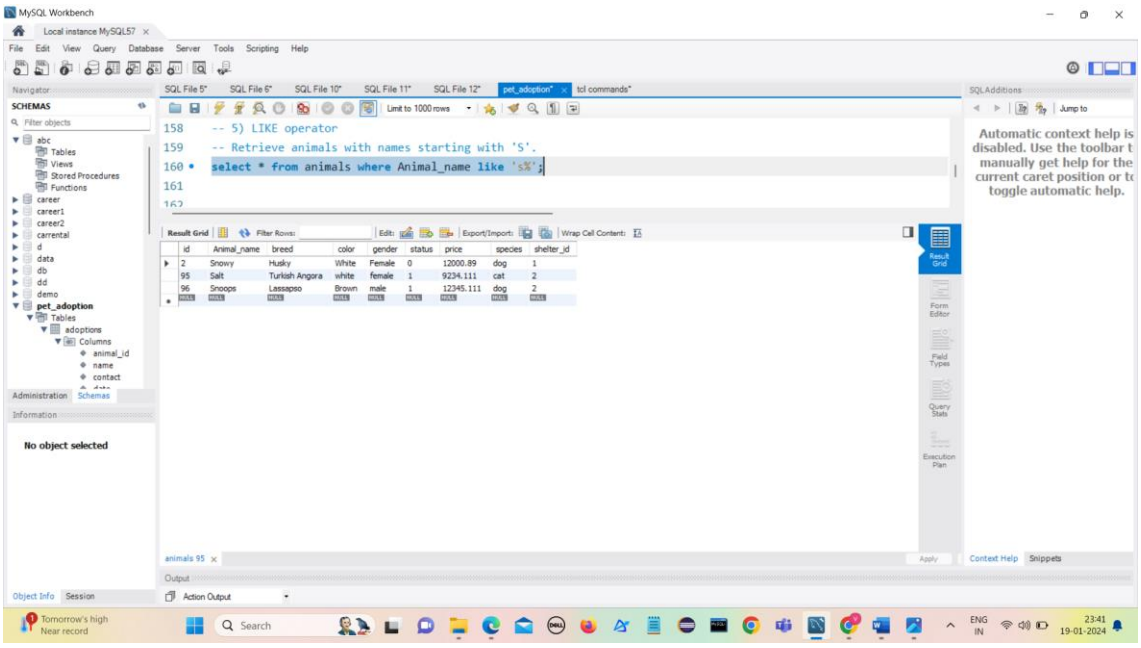
Retrieve animals that are either brown or white



This query returns all animals whose color is brown or white. It returns TRUE if either Boolean expression is TRUE.

5) LIKE operator

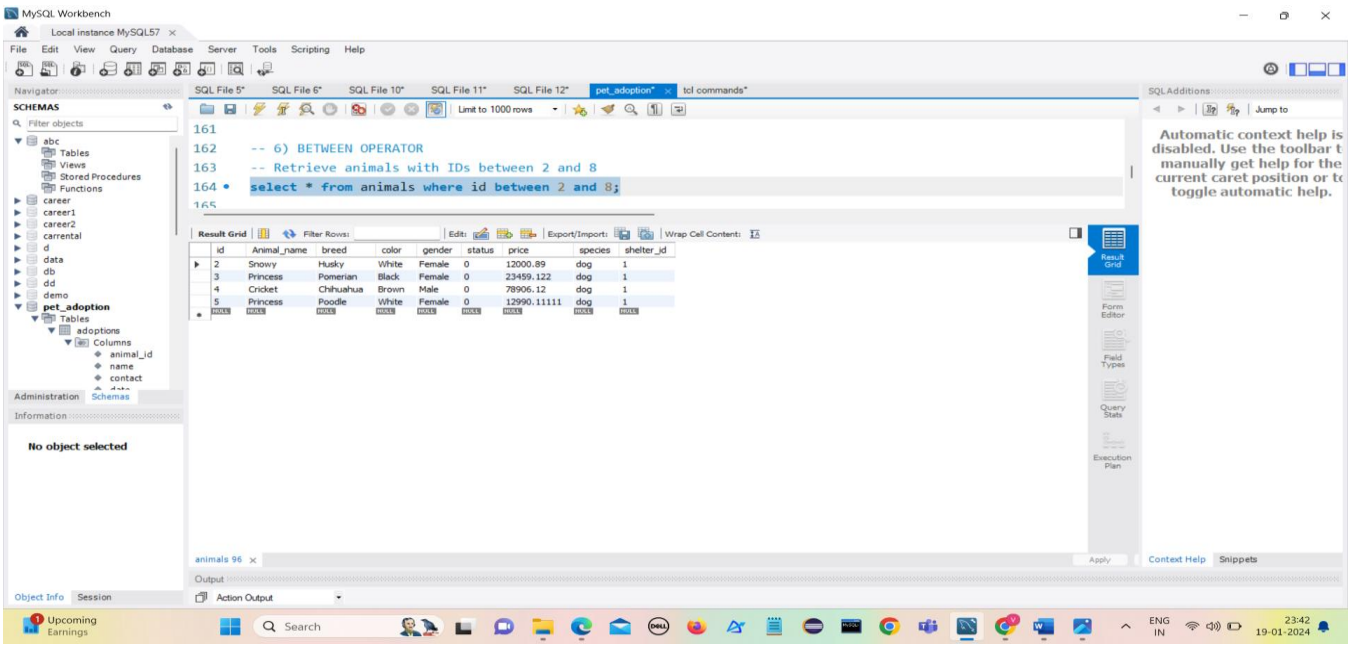
Retrieve animals with names starting with 'S'



This query returns animals whose name starting with 'S'. It returns TRUE if the operand matches a pattern.

6) BETWEEN OPERATOR

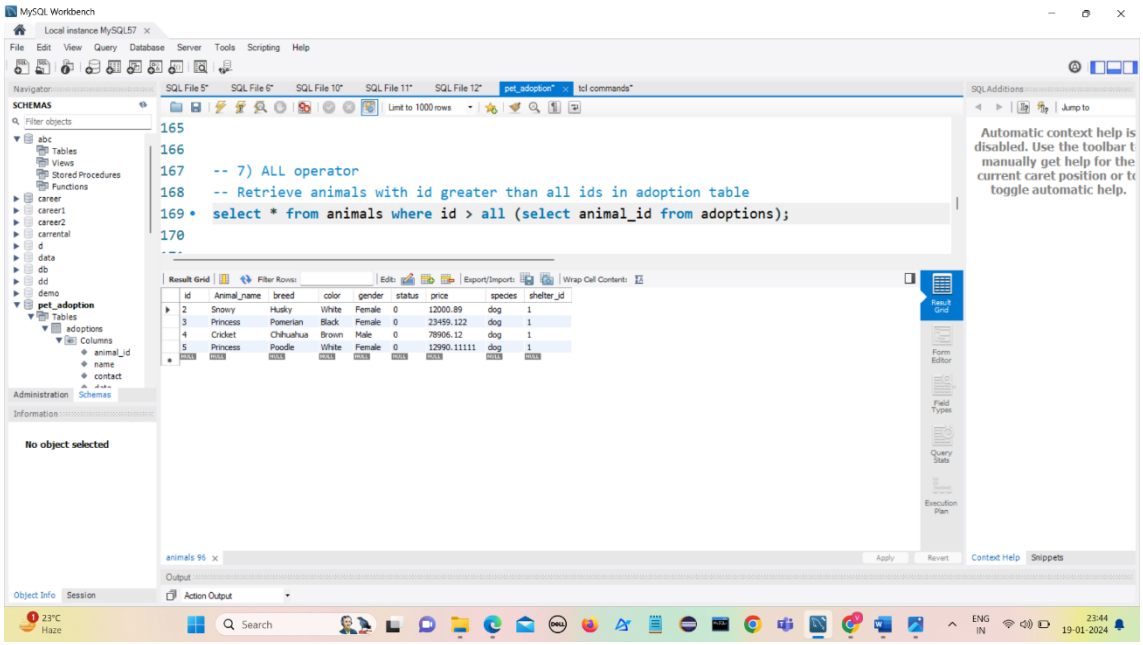
Retrieve animals with ids between 2 and 8



This query gives animals whose ids lie between 2 and 8. It returns TRUE if the operand is within a range.

7) ALL Operator

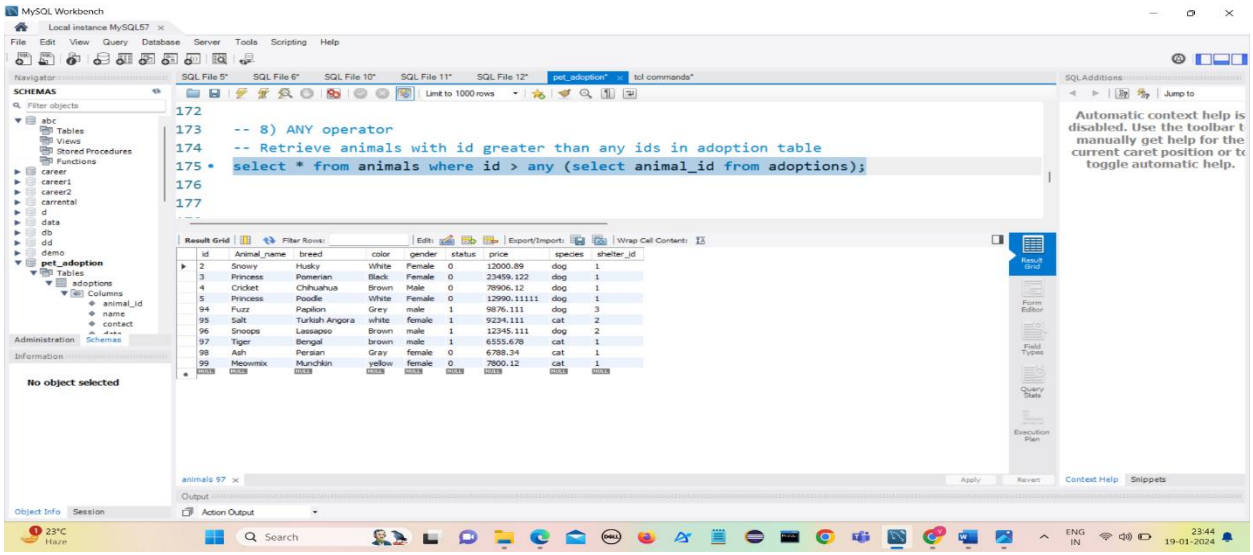
Retrieve animals with id greater than all ids in adoption table



This query returns all animals whose id greater than all ids in adoption table. It returns TRUE if all of a set of comparisons are TRUE.

8) ANY Operator

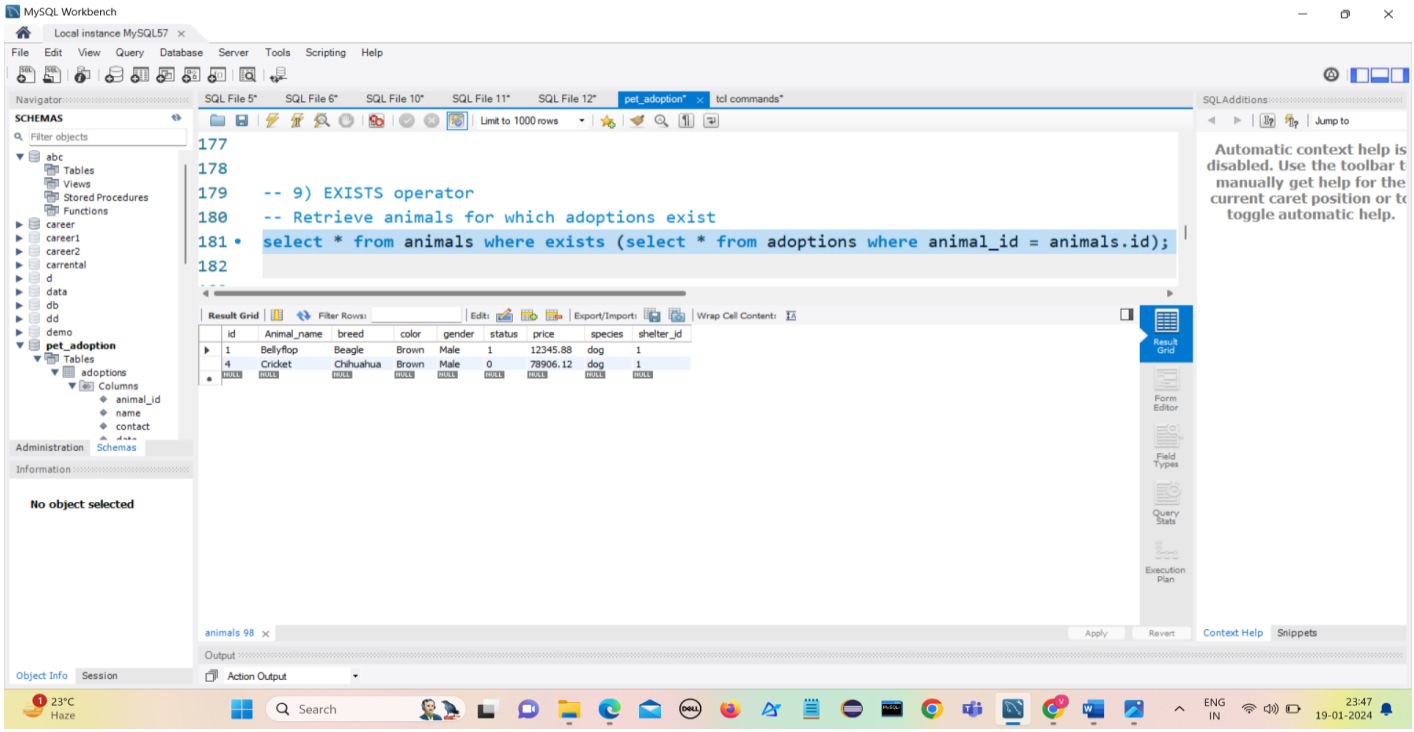
Retrieve animals with id greater than any ids in adoption table



This query returns all animals with id greater than any ids in adoption table. It returns TRUE if any one of a set of comparisons is TRUE.

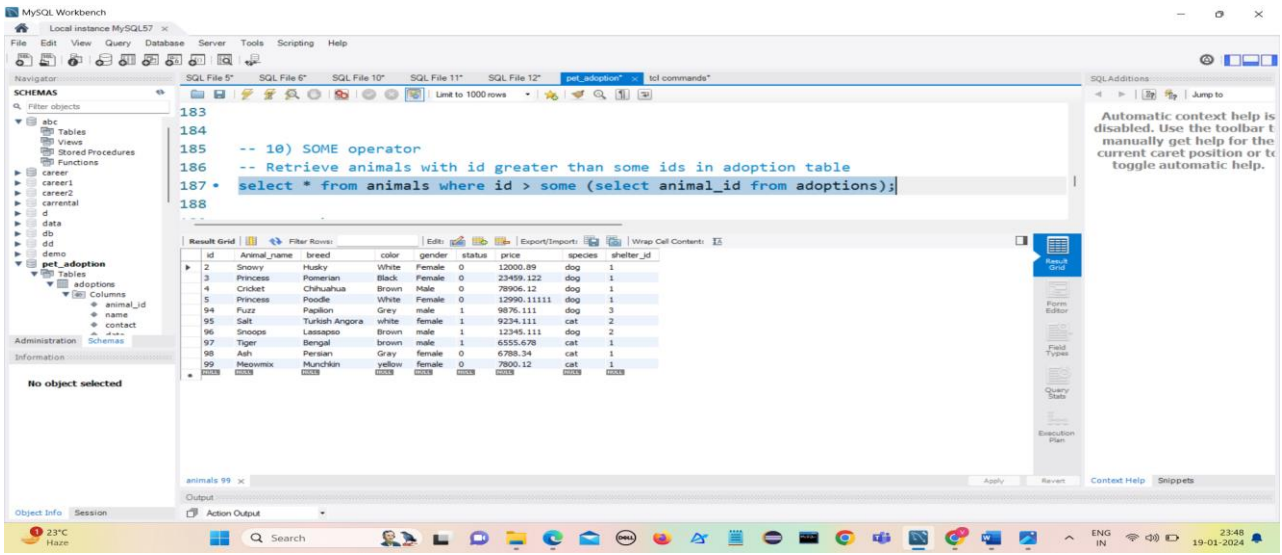
9) EXISTS Operator

Retrieve animals for which adoptions exist



It returns all the records of animals that are available for adoption. It returns TRUE if a subquery contains any rows.

10) SOME Operator

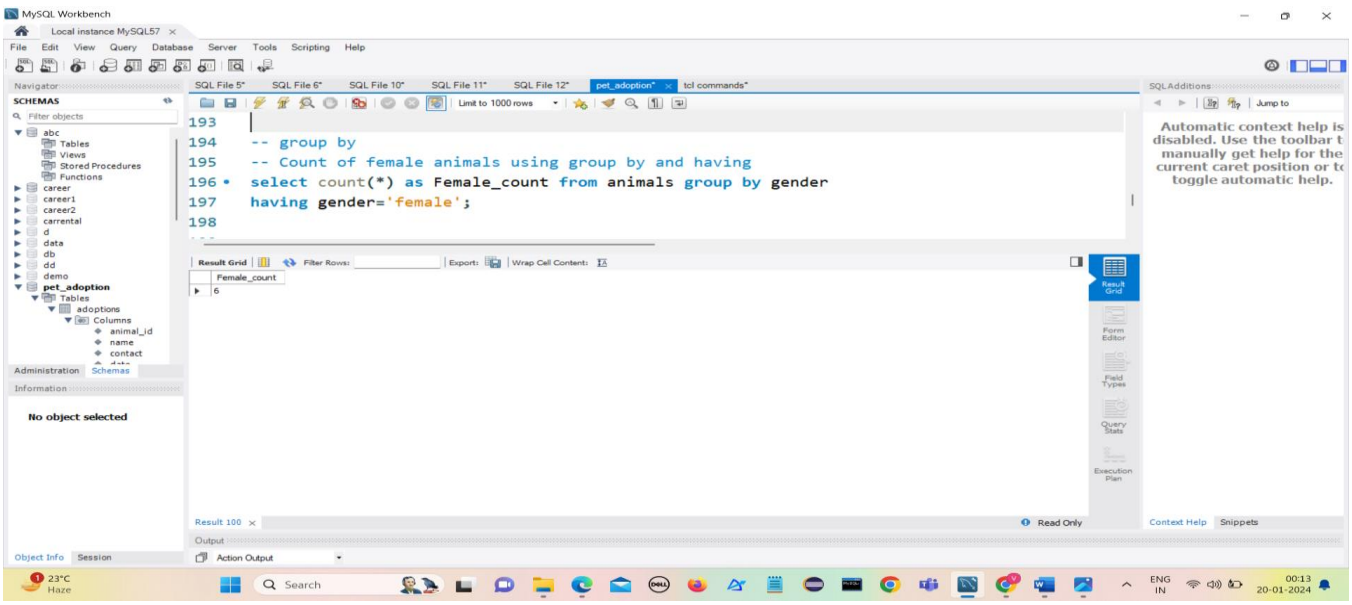


This query returns animals with id greater than some ids in adoption table. It returns TRUE if some of a set of comparisons are TRUE.

13) GROUP BY AND HAVING CLAUSE

The HAVING clause was added to SQL because the WHERE keyword cannot be used with aggregate functions

Count of female animals using group by and having clause

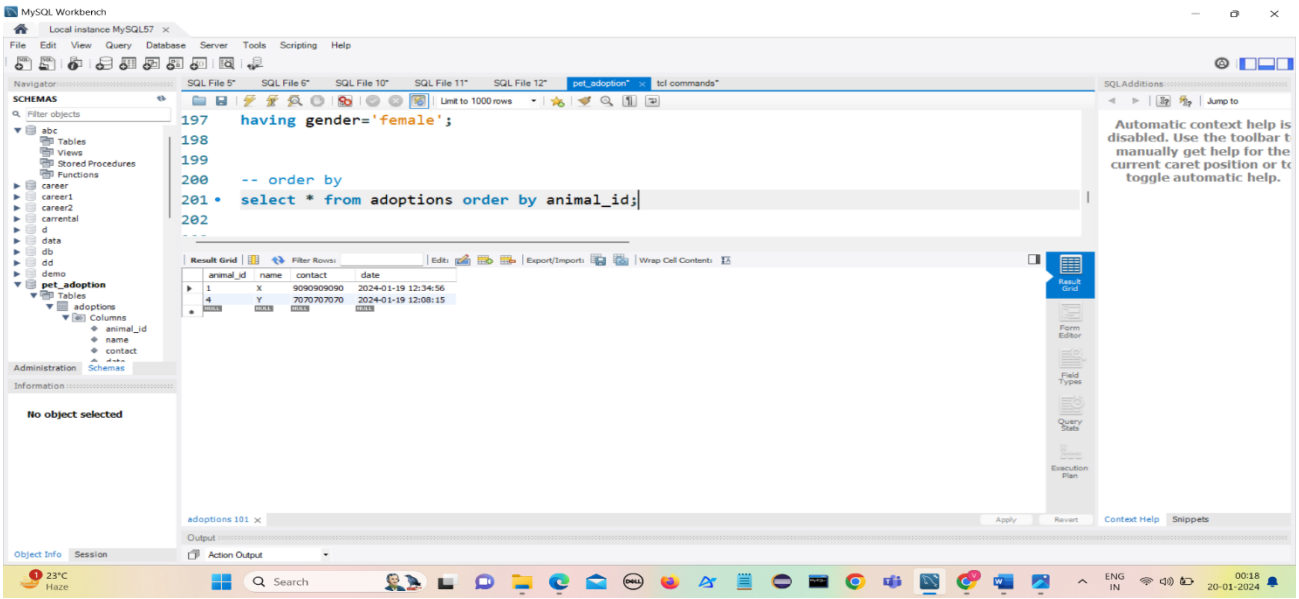


This query returns the count of all the female animals using group by and having clause.

14) ORDER BY CLAUSE

The ORDER BY keyword is used to sort the result-set in ascending or descending order.

Retrieve all the adoptions in the increasing order of animal_id



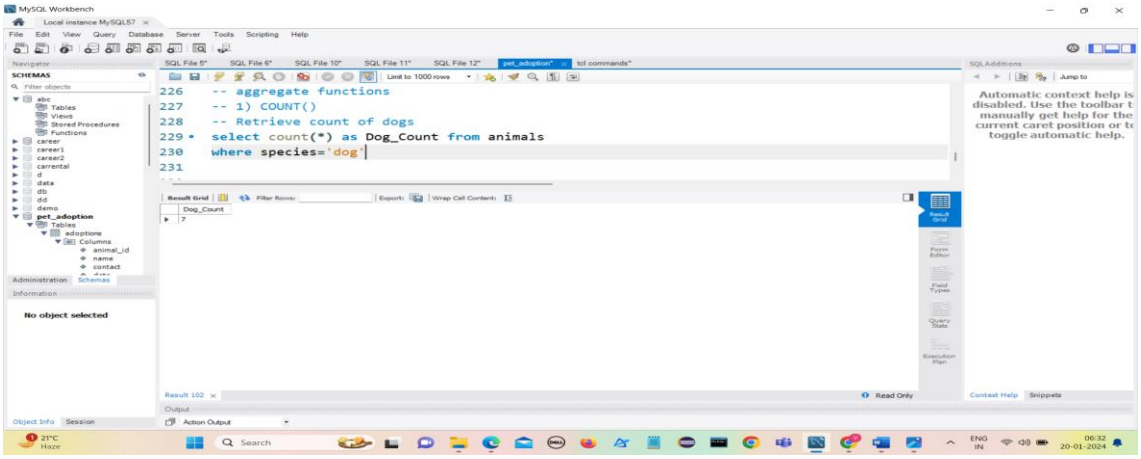
This query returns all the adoptions in the increasing order of animal_id.

15) AGGREGATE FUNCTIONS

1) COUNT()

The COUNT() function returns the number of rows that matches a specified criterion.

-- Retrieve count of dogs

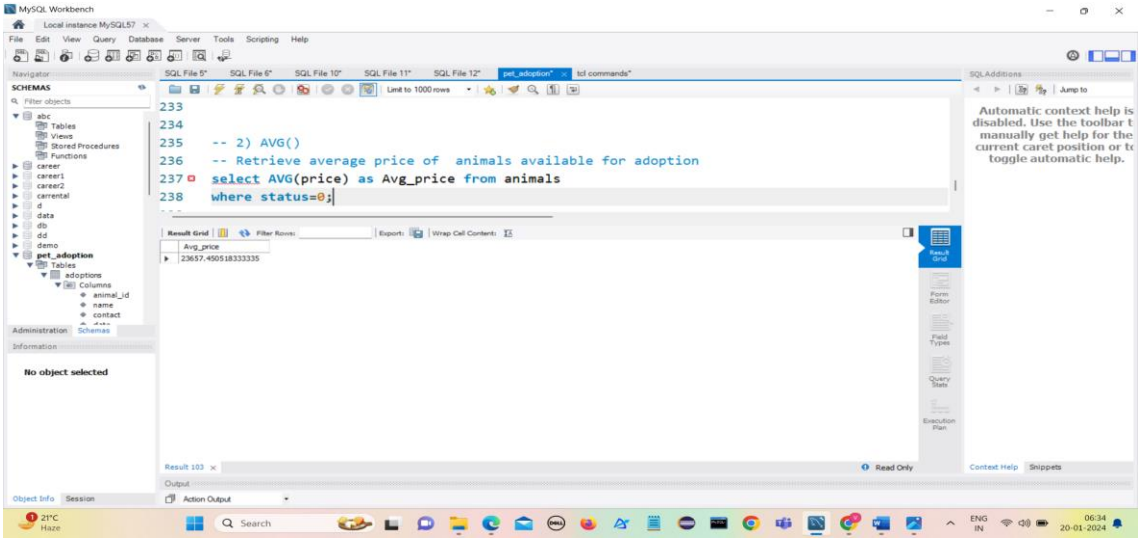


This query retrieves the number of dogs in animal table.

2) AVG()

The AVG() function returns the average value of a numeric column.

Retrieve average price of animals available for adoption

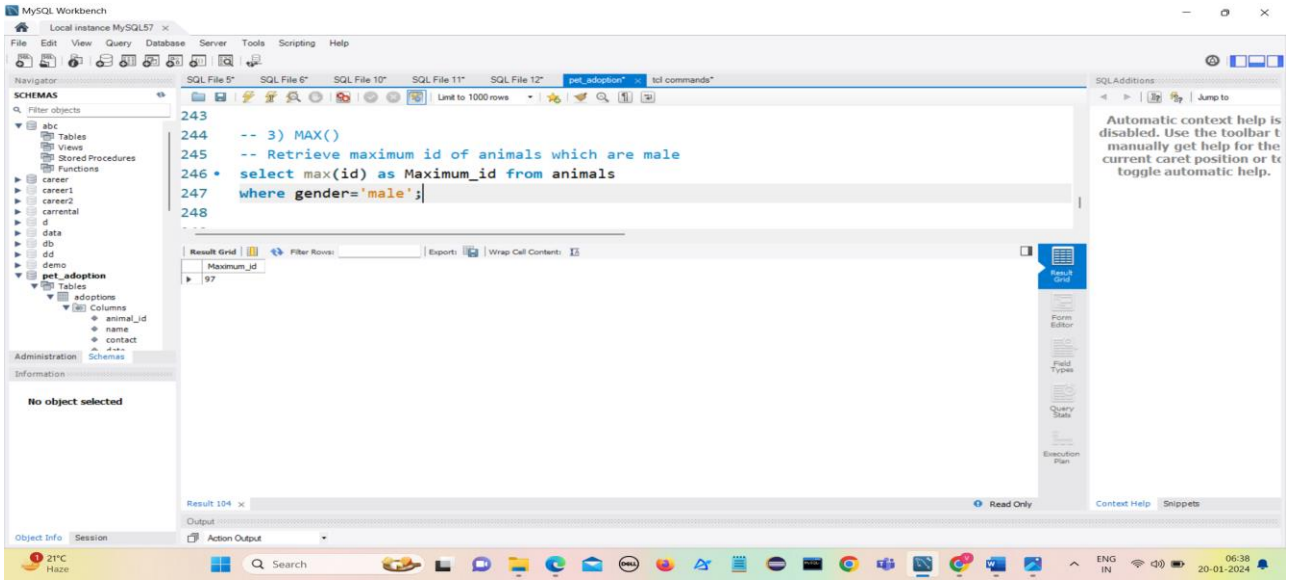


This query returns the average price of all animals that are available for adoption using AVG() function.

3) **MAX()**

The MAX() function returns the largest value of the selected column.

Retrieve maximum id of animals which are male.

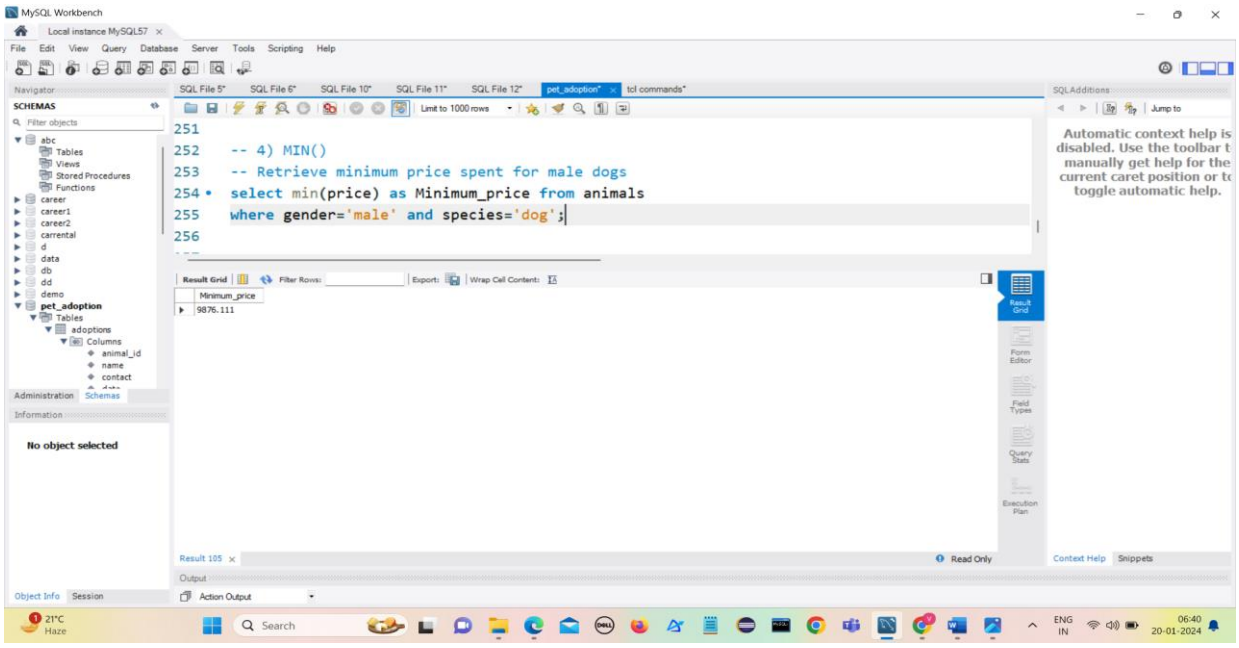


This query returns maximum id of animals which are male using MAX() function.

4) **MIN()**

The MIN() function returns the smallest value of the selected column.

Retrieve minimum price spent for male dogs

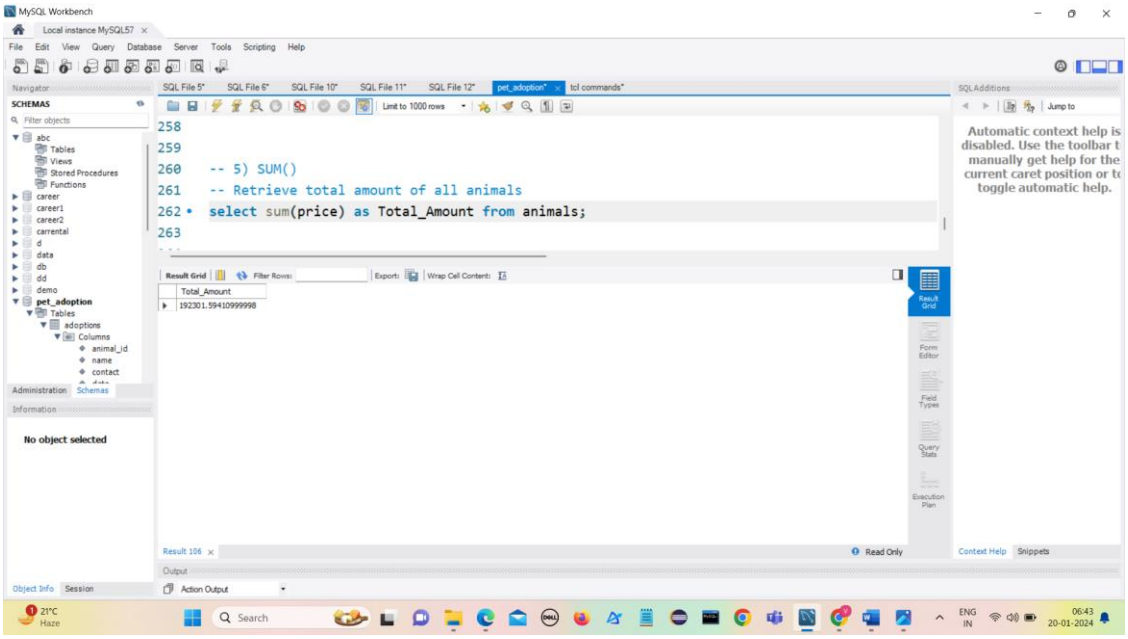


This query returns the minimum price that is spent on dogs which are dogs using MIN() function.

5) SUM()

The SUM() function returns the total sum of a numeric column.

Retrieve total amount of all animals



This query returns the total amount on animals that can be cats or dogs.

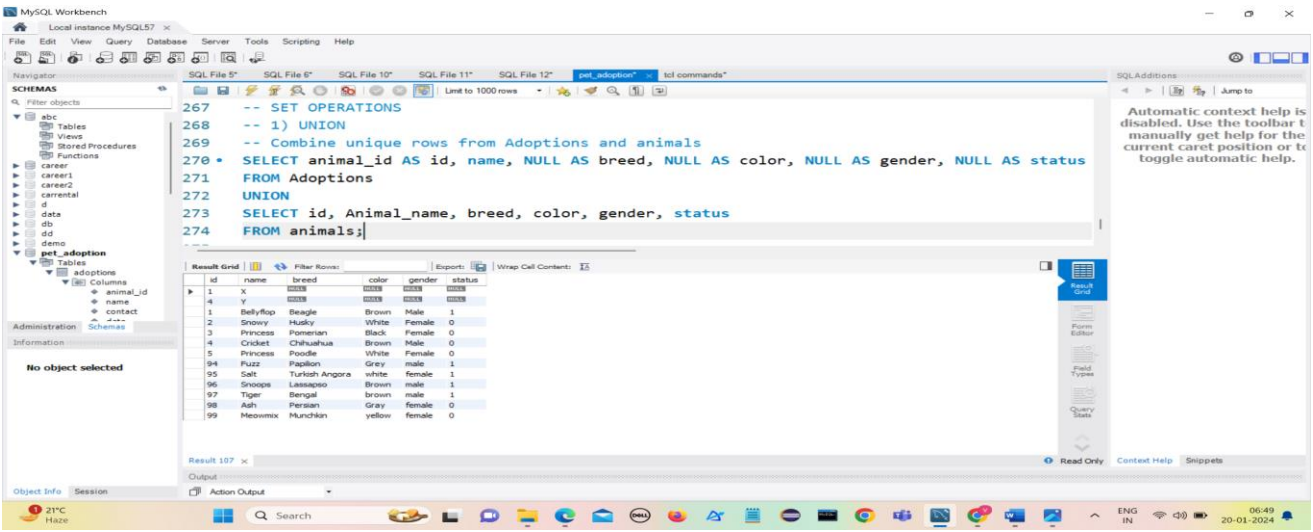
16) SET Operations in SQL

SET operators are special type of operators which are used to combine the result of two queries.

1) UNION

UNION will be used to combine the result of two select statements.

Duplicate rows will be eliminated from the results obtained after performing the UNION operation.

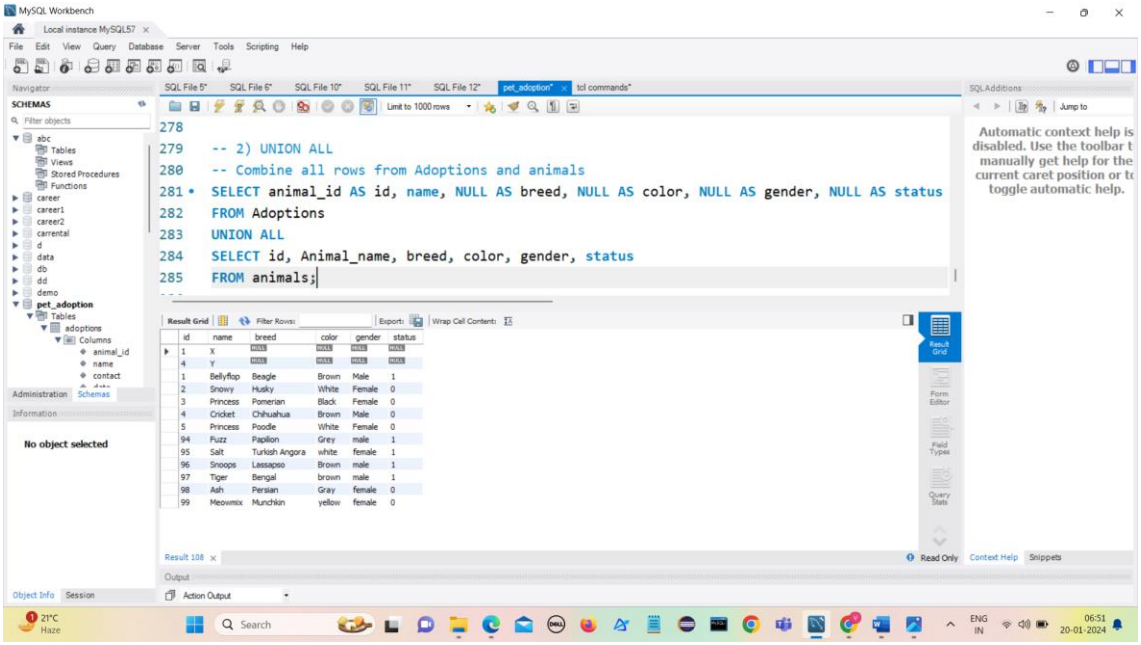


This query returns unique rows from both animals and adoptions table.

2) UNION ALL

This operator combines all the records from both the queries.

Duplicate rows will be not be eliminated from the results obtained after performing the UNION ALL operation.



This query returns all the rows from Animals and Adoptions table without removing duplicate elements.

17) Transaction Control Commands

Transactional control commands are only used with the **DML Commands** such as - INSERT, UPDATE and DELETE. They cannot be used while creating tables or dropping them because these operations are automatically committed in the database.

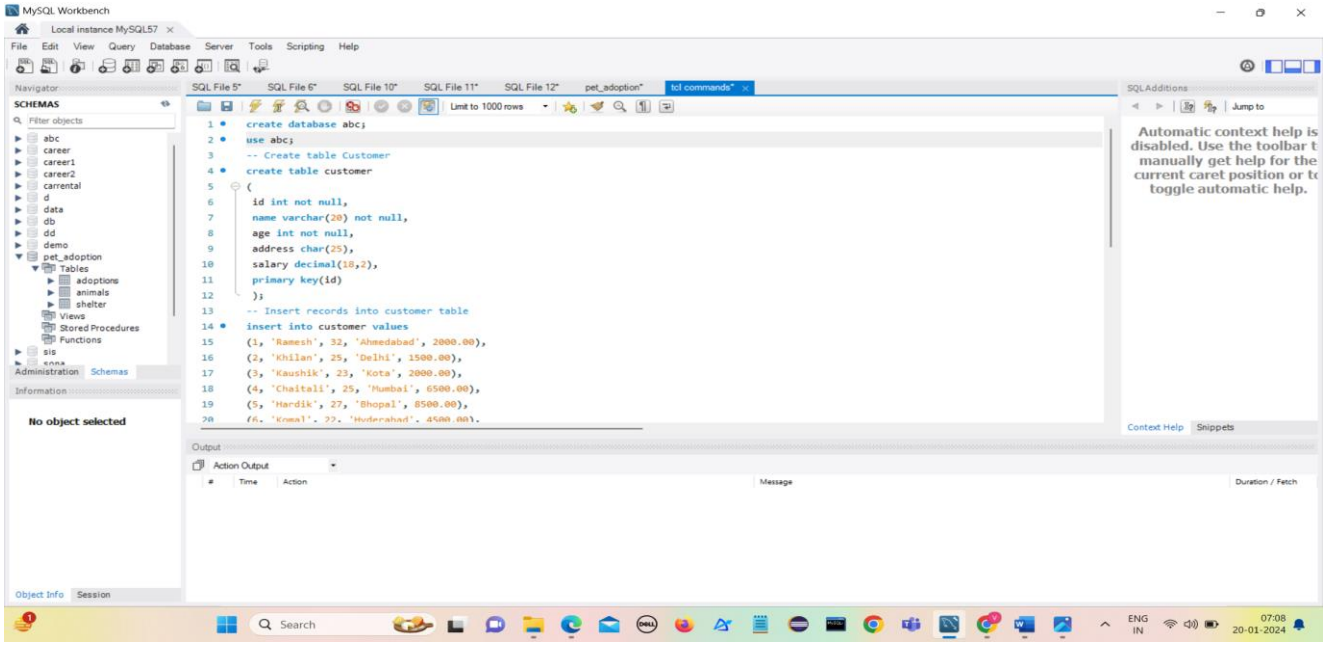
1) COMMIT

The COMMIT command is the transactional command used to save changes invoked by a transaction.

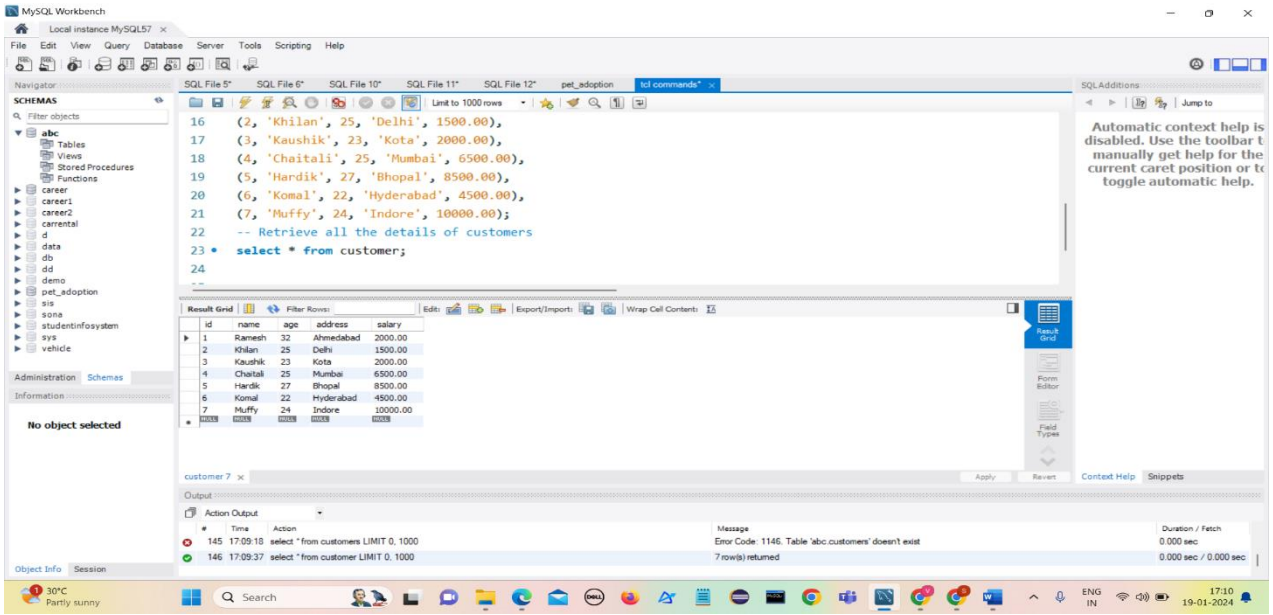
```
create database abc;
use abc;
```

Create the database abc and make abc database as the current working database on which operations can be performed.

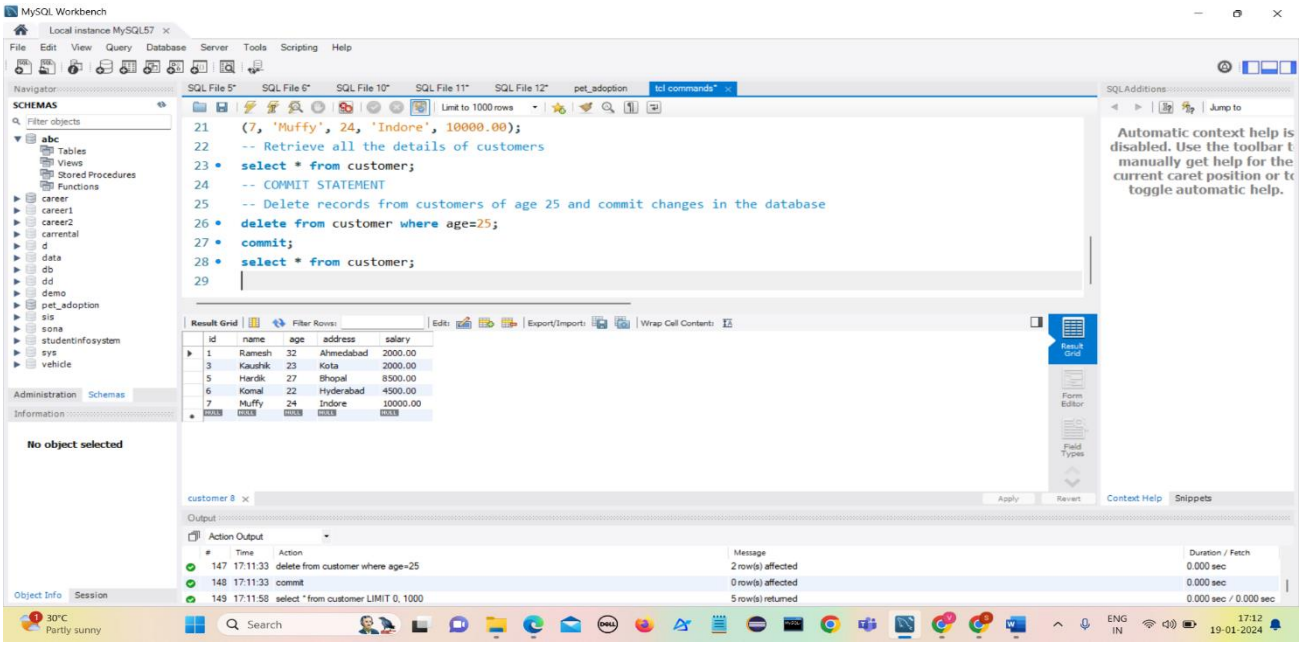
Now create a table named customer using create statement and insert values into the customer table.



Retrieve all the details from customer table using select statement

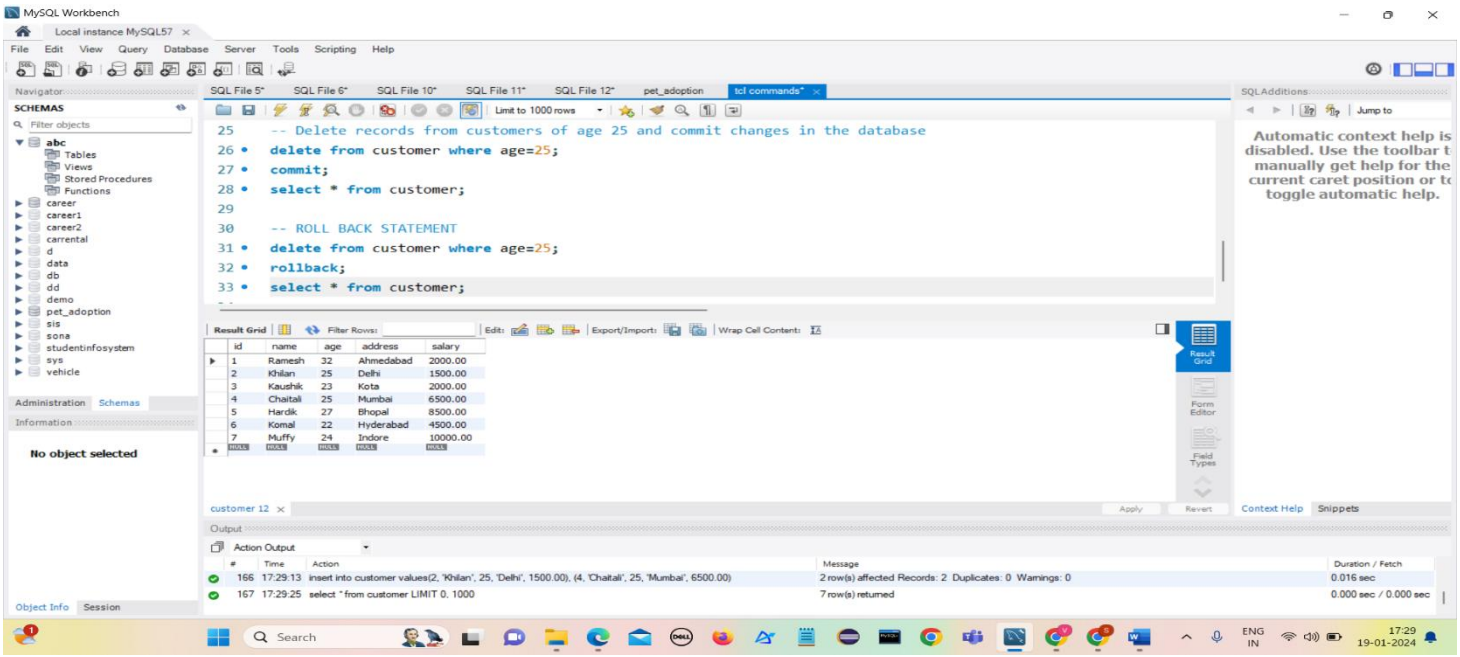


Use commit statement



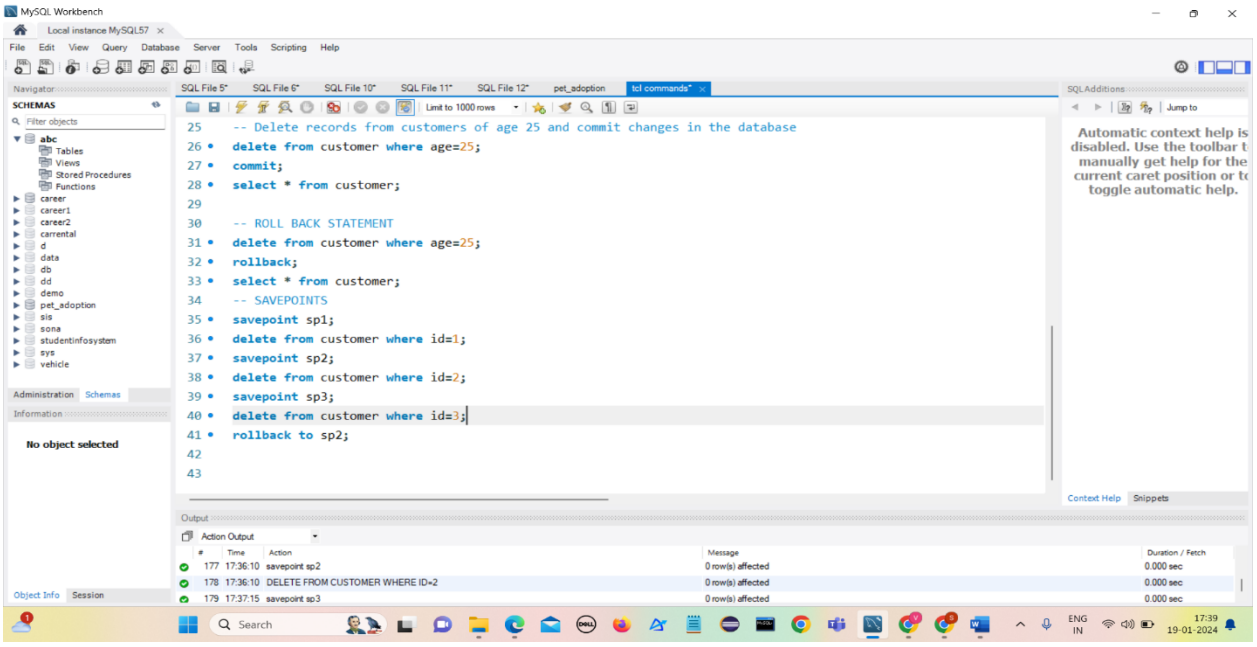
2) ROLLBACK

The ROLLBACK command is the transactional command used to undo transactions that have not already been saved to the database.



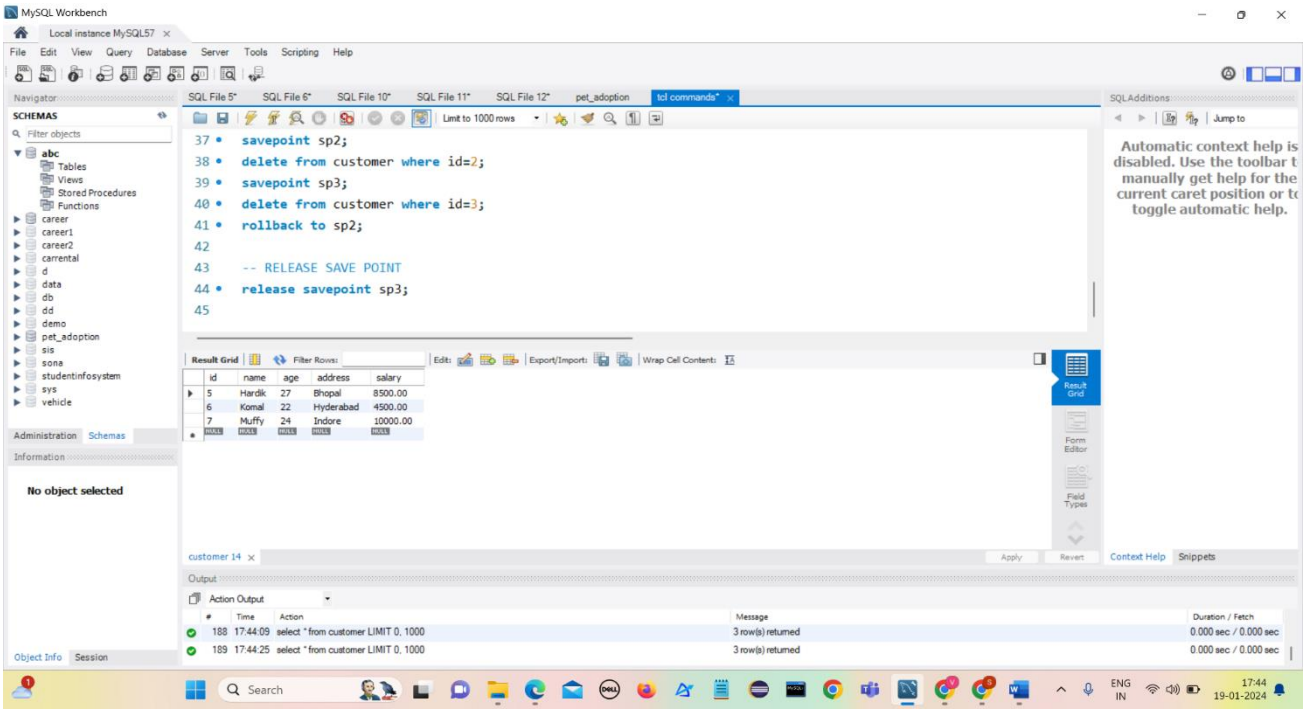
3) SAVEPOINT

A SAVEPOINT is a logical rollback point in a transaction.



4) RELEASE SAVEPOINT

The RELEASE SAVEPOINT command is used to remove an existing SAVEPOINT.



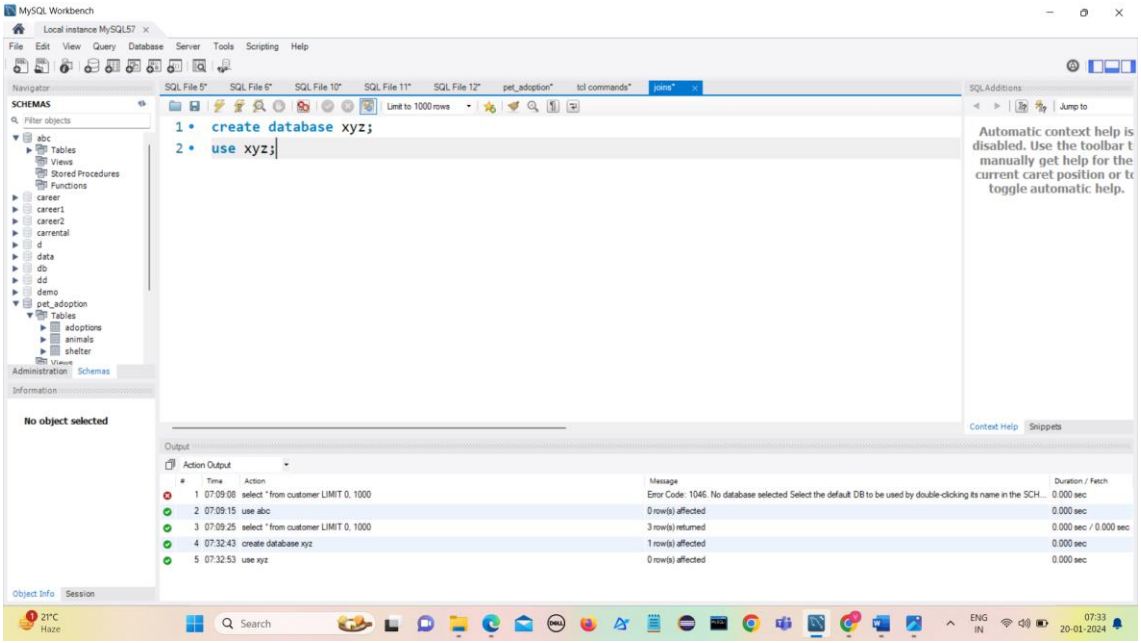
18) JOINS

Join statement is used to combine data or rows from two or more tables based on a common field between them.

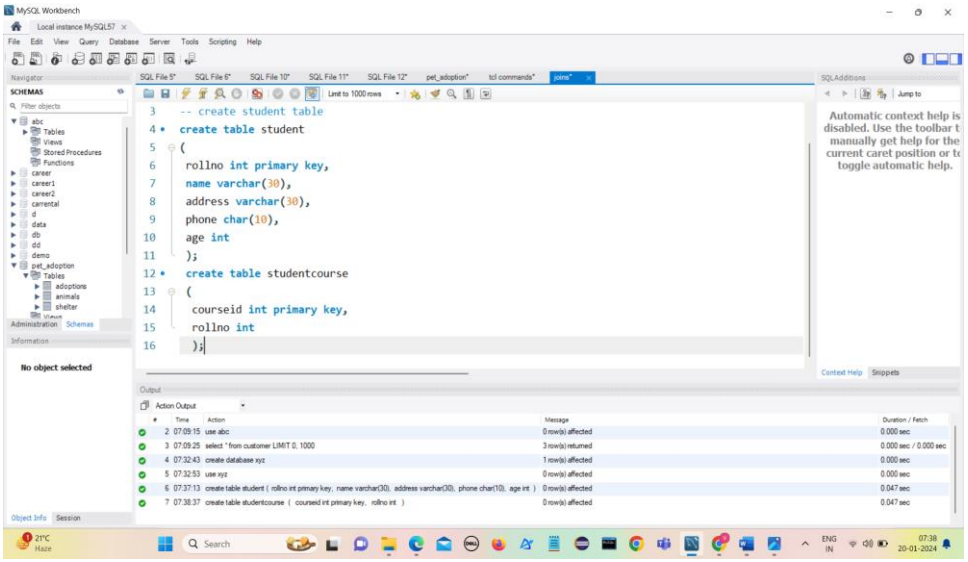
1) INNER JOIN

The INNER JOIN keyword selects all rows from both the tables as long as the condition is satisfied. This keyword will create the result-set by combining all rows from both the tables where the condition satisfies i.e value of the common field will be the same.

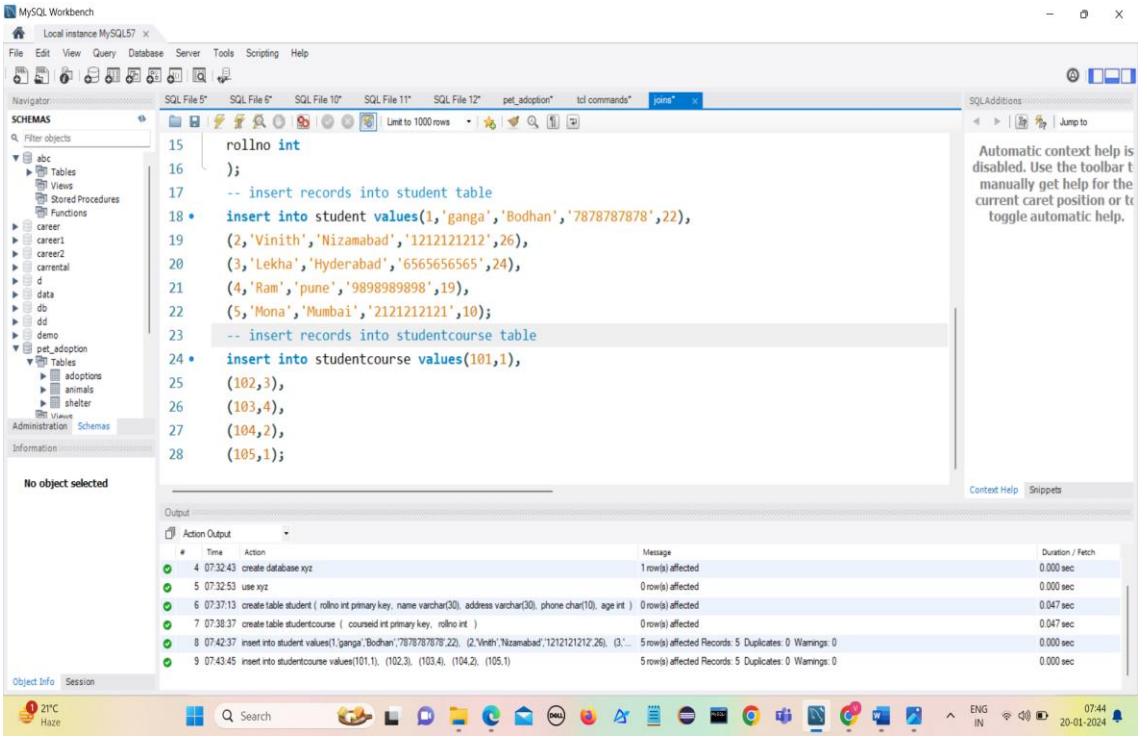
Create a table xyz and use the database to make it as the current working database.



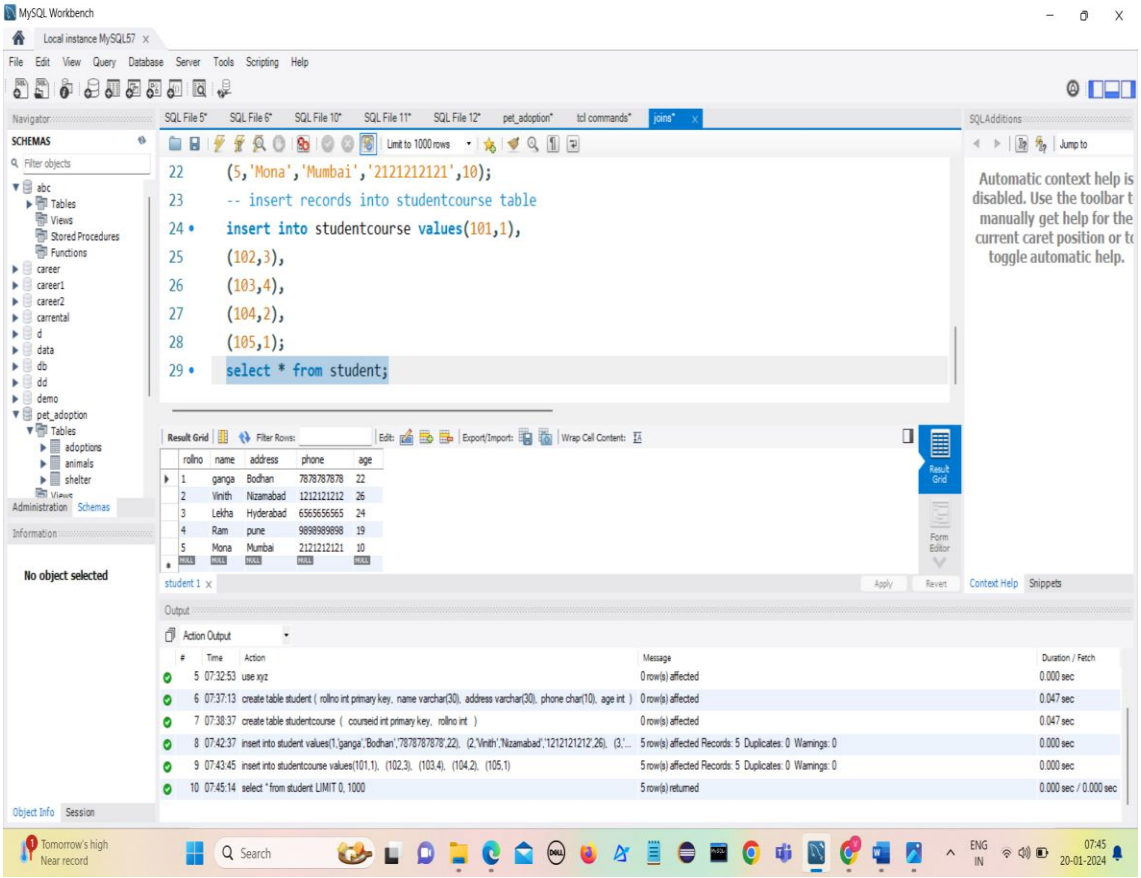
Create two tables Student and Studentcourse using create table statement.

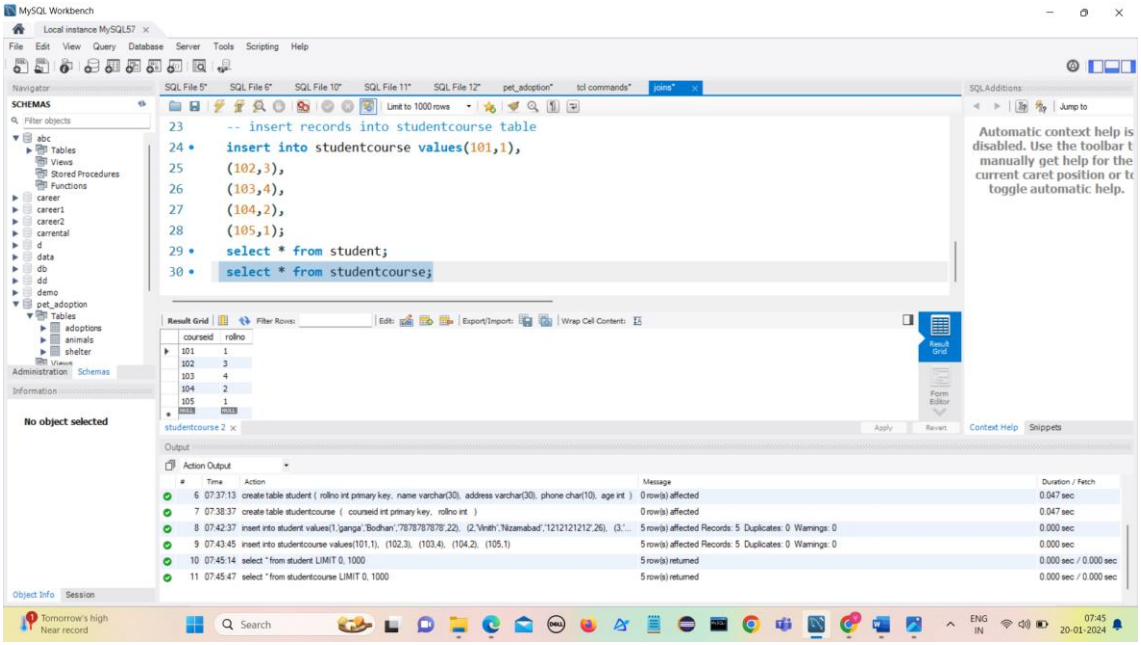


Insert records into Student and Studentcourse table using insert statement.

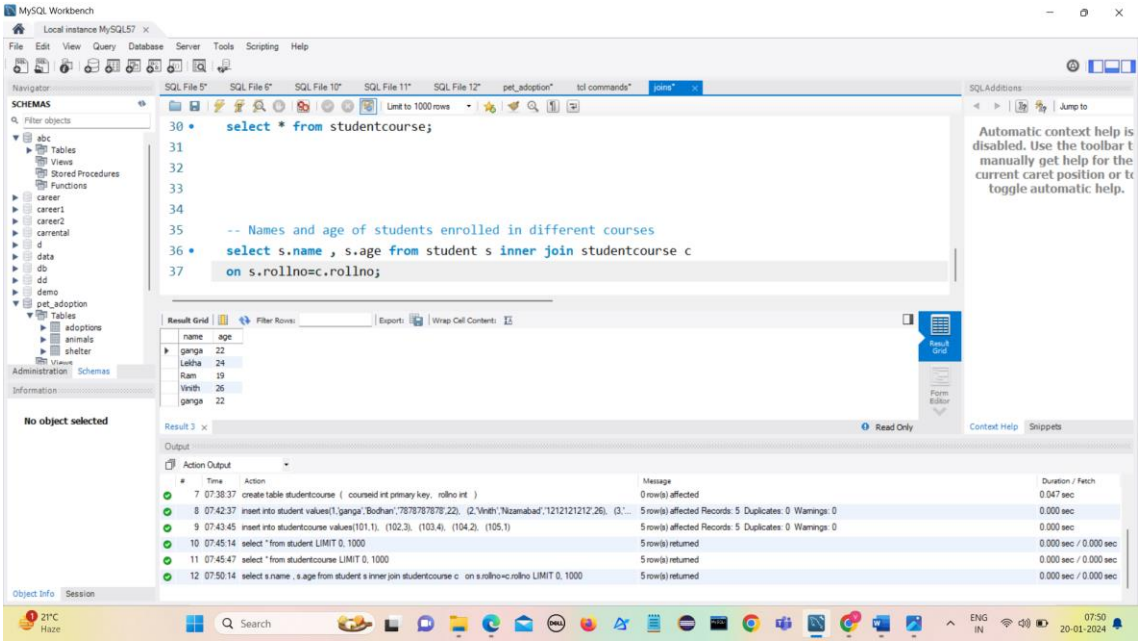


Retrieve records from student and studentcourse table





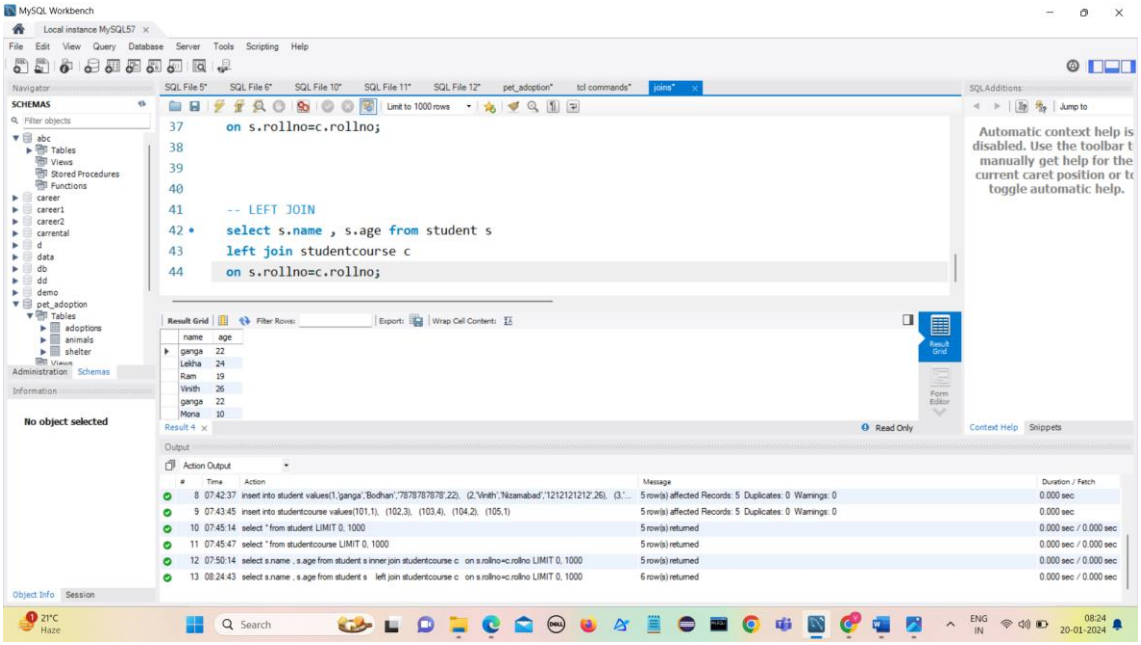
Inner join



This query gives the names and age of students enrolled in different courses.

2) **LEFT JOIN**

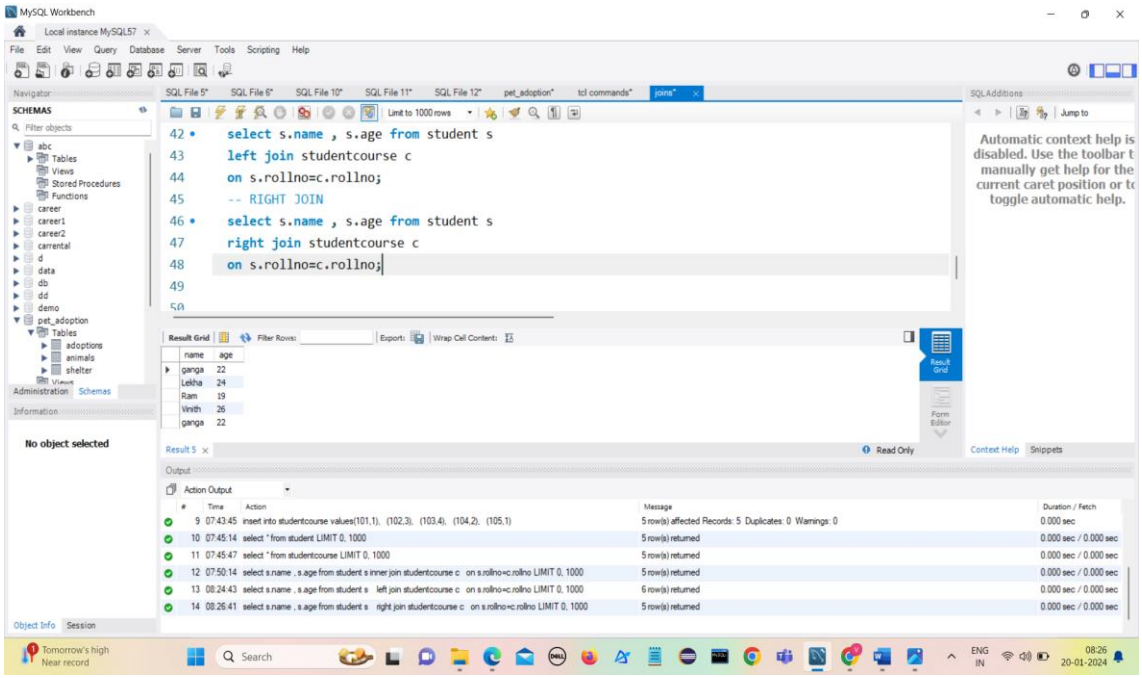
This join returns all the rows of the table on the left side of the join and matches rows for the table on the right side of the join. For the rows for which there is no matching row on the right side, the result-set will contain *null*.



3) RIGHT JOIN

RIGHT JOIN is similar to LEFT JOIN. This join returns all the rows of the table on the right side of the join and matching rows for the table on the left side of the join.

For the rows for which there is no matching row on the left side, the result-set will contain *null*



Vuthur Sriganga
Data Engineering Batch-1