| Experiment No.6 |
| --- |
| Pattern Matching Queries |
| Date of Performance: 27/3/24 |
| Date of Submission: 19/3/24 |

**Experiment No. 6: Implement Pattern Matching queries**

**Course Outcome [CSL503.3]:** Apply SQL queries ,triggers and procedures for specific module/task

**Aim**: **Implement pattern matching queries using SQL**

**Theory:**

Pattern matching queries in SQL involve searching for specific patterns or substrings within textual data stored in a database. These queries enable users to find records that match a particular pattern or criteria, facilitating various data analysis and retrieval tasks.

In SQL, pattern matching is typically performed using the LIKE operator, which allows for the comparison of a column value against a pattern using wildcard characters such as % (for matching any sequence of characters) and \_ (for matching any single character).

For example, to find all records in a table where a certain column starts with a specific prefix, you could use a pattern matching query like:

sql

SELECT \* FROM table\_name WHERE column\_name LIKE 'prefix%';

Pattern matching queries are useful in scenarios such as searching for specific words or phrases in textual data, filtering records based on partial matches, and extracting information from unstructured text.

By implementing pattern matching queries, users can enhance the flexibility and effectiveness of their SQL queries, enabling more precise data retrieval and analysis.

**Code:**

CREATE TABLE Emp (

id INT(20),

Name VARCHAR(50),

city CHAR(20),

salary INT(50),

age INT(50)

);

INSERT INTO Emp (id, Name, city, salary, age) VALUES

(1, "Priyank S", 'Nashik', 26000, 20),

(2, "Riya S", "Mumbai", 72000, 28),

(3, "Neha V", "Varanasi", 37000, 19),

(4, "Neeta Desai", "Nashik", 395000, 21),

(5, "Priya Wagh", "Udaipur", 60000, 22);

SELECT \* FROM Emp;

-- Pattern matching queries

SELECT \* FROM Emp WHERE Name LIKE 'P%';

SELECT \* FROM Emp WHERE Name LIKE '%V';

SELECT \* FROM Emp WHERE Name LIKE '%ta%';

SELECT \* FROM Emp WHERE city LIKE ''; -- 6 under score

SELECT \* FROM Emp WHERE city LIKE '\_an%'; -- 3 under score

SELECT \* FROM Emp WHERE Name LIKE '%de%';

SELECT \* FROM Emp WHERE salary LIKE '%50%';

SELECT \* FROM Emp WHERE city LIKE 'Na\_\_ik';

SELECT \* FROM Emp WHERE salary LIKE '3\_\_00';

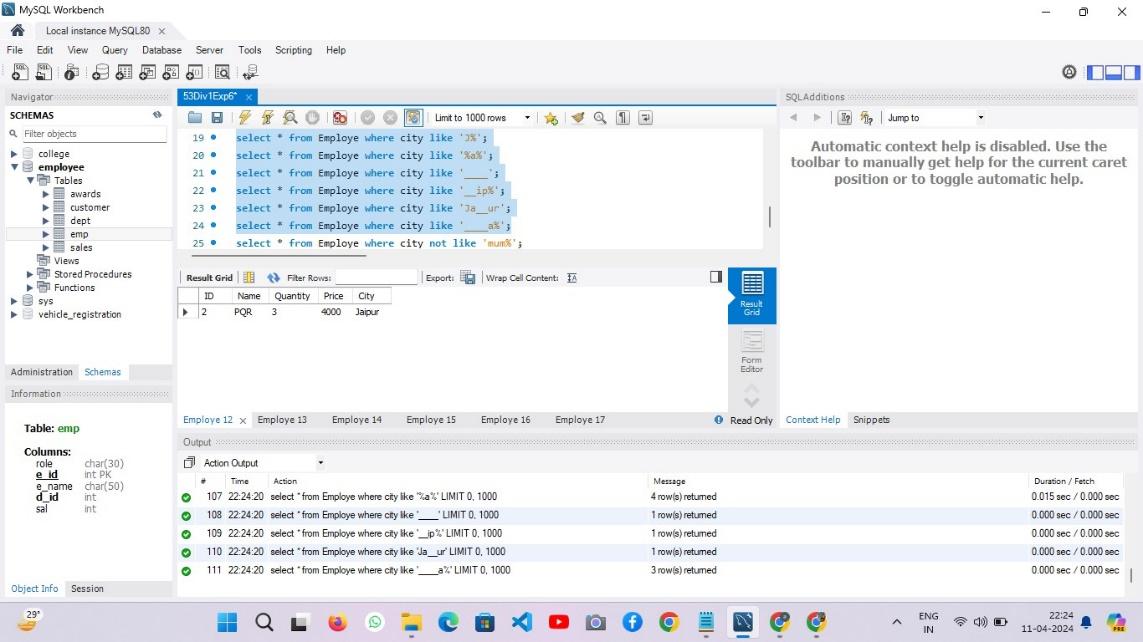
SELECT \* FROM Emp WHERE Name LIKE '\_\_a%';

SELECT \* FROM Emp WHERE salary LIKE '\_\_00%';

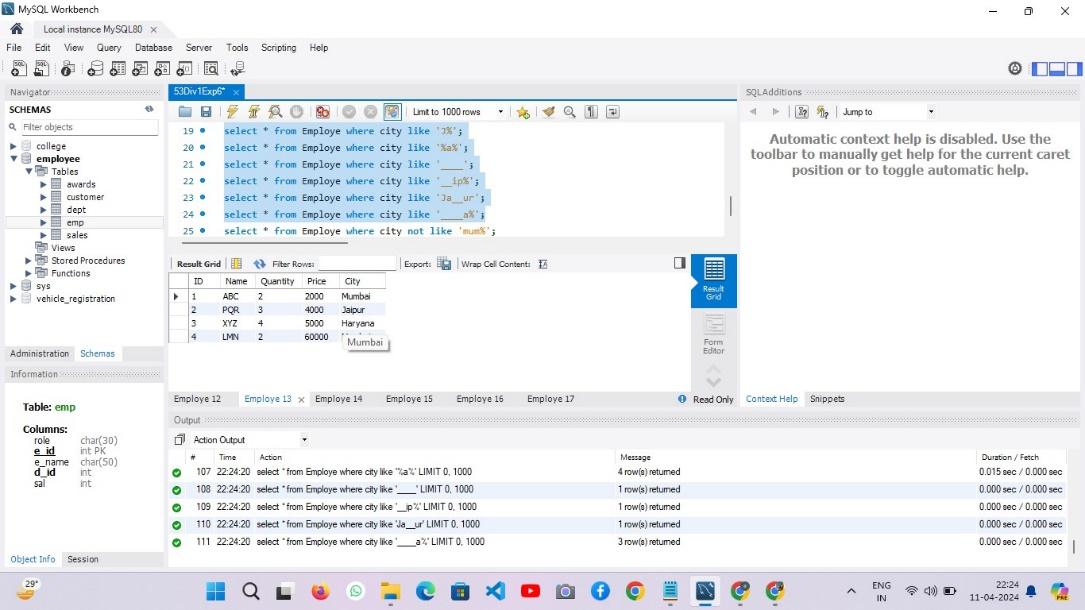
SELECT \* FROM Emp WHERE Name NOT LIKE 'Priya%';

**Output:**

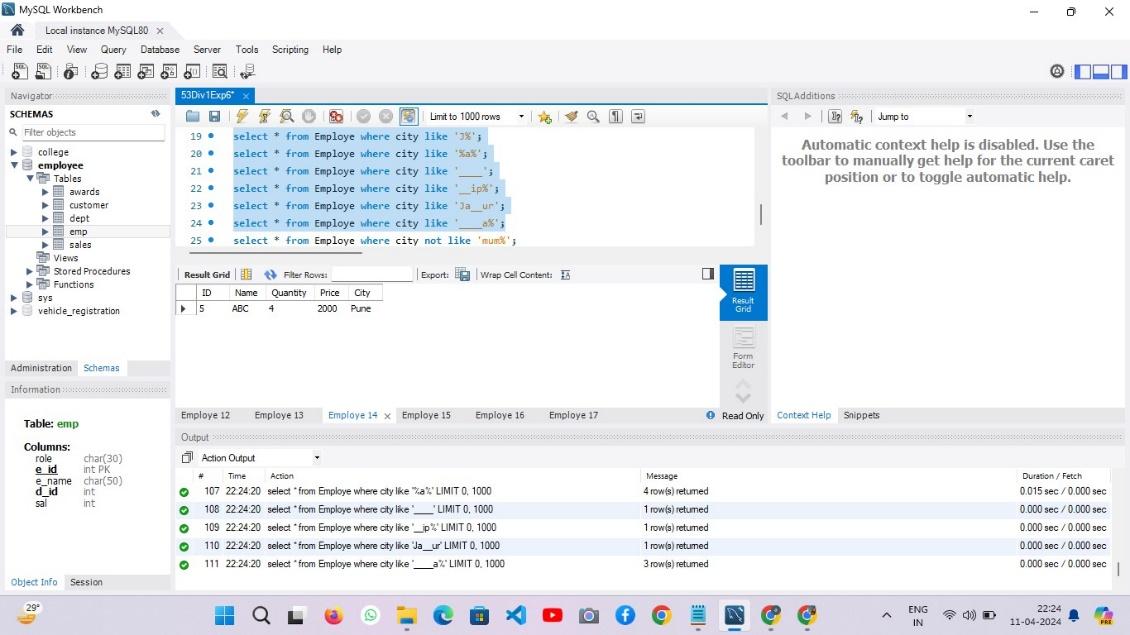
**1.All city name that start with J:**



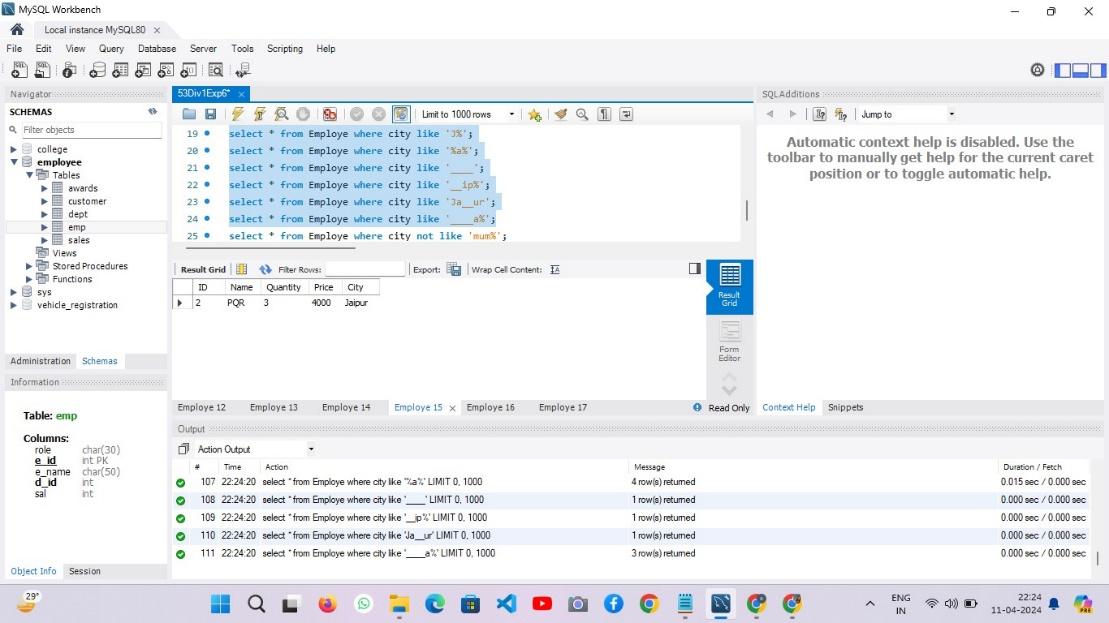
**2] Retrieve all city whose names contain the substring "a" anywhere.**



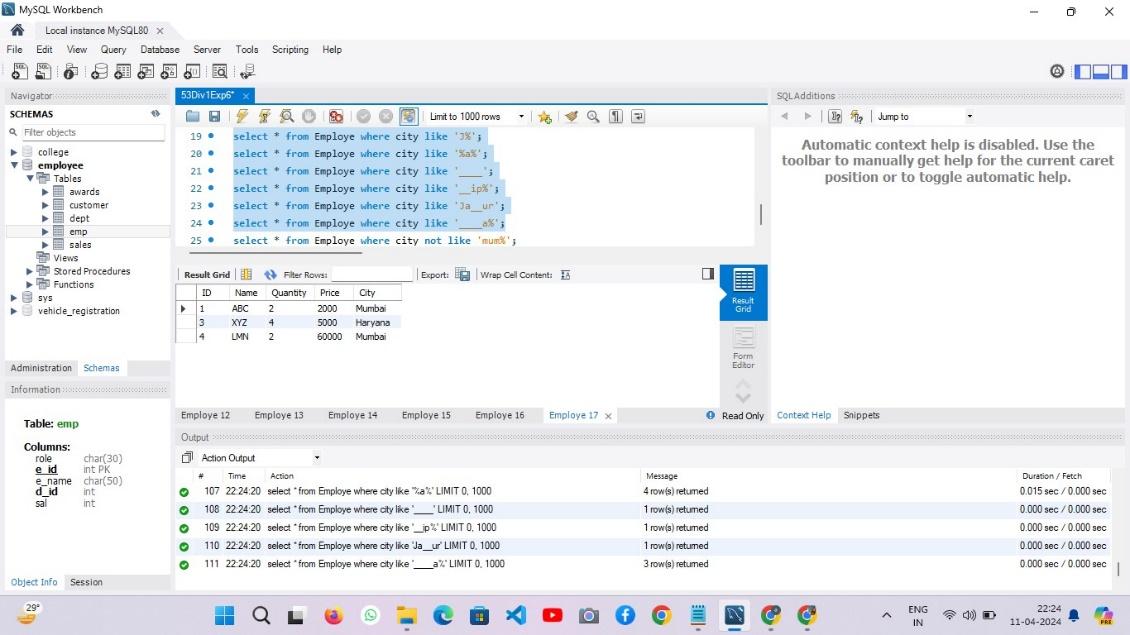
**3] Retrieve all city whose name has exactly four characters.**



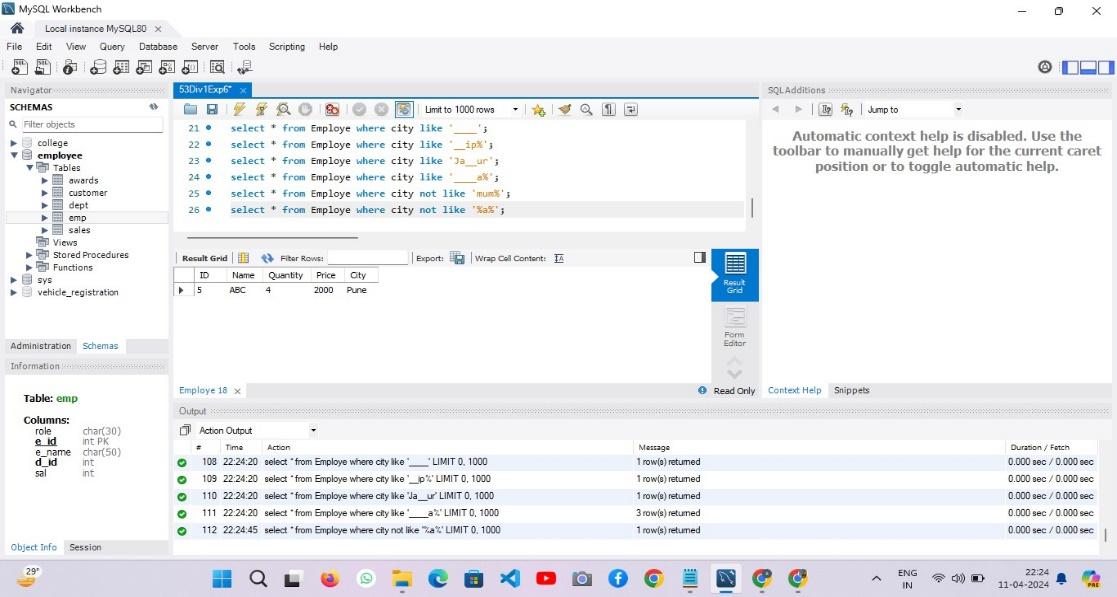
**4] Retrieve all city whose city name starts with any single character followed by"ja".**



**5] Retrieve all city name who’s name ends at a:**



**6.write the all city name in which the in the entire name this character is not present:**



**Conclusion:**In this experiment, pattern matching queries in SQL were implemented to retrieve specific records based on predefined patterns or substrings within textual data. By utilizing the LIKE operator and wildcard characters such as % and \_, various search criteria were applied to the dataset.

The experiment successfully demonstrated the capability of SQL pattern matching for filtering and extracting relevant information from a database. Through these queries, users can efficiently retrieve desired data subsets, enhancing data analysis and retrieval tasks within SQL databases.