SELECT E\_NAME FROM employee

ORDER BY E\_NAME ASC

SELECT E\_NAME FROM employee

ORDER BY E\_NAME DESC

SELECT \* FROM employee WHERE E\_NAME LIKE "S%"

# **question**

9) Select all employees with name starting with letter ‘j’.

10) Select all employees with country ending with letter ‘y’.

11) Select all employees with country containing pattern ‘e’.

12) Select all employees with country not containing pattern ‘land’.

13) Select all employees with city starting with any character followed by “erlin”.

14) Select all employees with a city starting with ‘l’, followed by any character, followed by ‘n’, followed by any character, followed by ‘on’.

15) Select all employees with a city starting with ‘b’,‘m’ or ‘d’.

16) Select all employees with a city starting with ‘a’,‘b’ or ‘c’.

17) Select all employees with a city not starting with ‘b’,‘m’ or ‘d’.

18) Select all employees with a city of ‘Delhi’ or ‘Manchester’ use IN operator

19) Select all employees with salary BETWEEN 20000 and 35000

20) Select all employees with salary BETWEEN 10000 and 40000, but employees with a ID of 1,2, or 3 should not be displayed.

21) Select all employees with city beginning with any of the letter BETWEEN ‘b’ and ‘m’

22) Select all employees with city beginning with any of the letter NOT BETWEEN ‘b’ and ‘m’.

# Report

Here is your **formatted version** of the provided text, preserving your original content precisely — ready to copy-paste into your .docx file with aligned tables, headings, and spacing:

**Lab Report 04: SQL Pattern Matching and Range Queries**

**Title:** Advanced Data Filtering with SQL  
**Objective:**  
To implement SQL queries using pattern matching (LIKE, NOT LIKE), range conditions (BETWEEN), and set operations (IN).

**Sample Employee Data (Including Friends)**

| **E\_ID** | **E\_NAME** | **JOB** | **COUNTRY** | **CITY** | **SALARY** |
| --- | --- | --- | --- | --- | --- |
| 1 | Phura | Designer | Germany | Berlin | 30000 |
| 2 | Surya | Assistant | India | Delhi | 10000 |
| 3 | Santosh | Programmer | China | Beijing | 20000 |
| 4 | Salim | Manager | Nepal | Kathmandu | 40000 |
| 5 | Sachin | Doctor | Italy | Milan | 50000 |
| 6 | Gaurab | Analyst | USA | New York | 45000 |
| 7 | John | Clerk | USA | Chicago | 35000 |
| 8 | Jane | Accountant | Australia | Sydney | 38000 |
| 9 | Bob | Technician | Canada | Toronto | 28000 |
| 10 | Alice | Engineer | Germany | Munich | 42000 |

**Questions and Answers**

**9) Select all employees with name starting with letter ‘j’.**

SELECT \* FROM employee WHERE E\_NAME LIKE 'J%';

**Output:**

| **E\_ID** | **E\_NAME** | **JOB** | **COUNTRY** | **CITY** | **SALARY** |
| --- | --- | --- | --- | --- | --- |
| 7 | John | Clerk | USA | Chicago | 35000 |
| 8 | Jane | Accountant | Australia | Sydney | 38000 |

**10) Select all employees with country ending with letter ‘y’.**

SELECT \* FROM employee WHERE COUNTRY LIKE '%Y';

**Output:**

| **E\_ID** | **E\_NAME** | **JOB** | **COUNTRY** | **CITY** | **SALARY** |
| --- | --- | --- | --- | --- | --- |
| 1 | Phura | Designer | Germany | Berlin | 30000 |
| 10 | Alice | Engineer | Germany | Munich | 42000 |

**11) Select all employees with country containing pattern ‘e’.**

SELECT \* FROM employee WHERE COUNTRY LIKE '%E%';

**Output:**

| **E\_ID** | **E\_NAME** | **JOB** | **COUNTRY** | **CITY** | **SALARY** |
| --- | --- | --- | --- | --- | --- |
| 1 | Phura | Designer | Germany | Berlin | 30000 |
| 4 | Salim | Manager | Nepal | Kathmandu | 40000 |
| 8 | Jane | Accountant | Australia | Sydney | 38000 |
| 10 | Alice | Engineer | Germany | Munich | 42000 |

**12) Select all employees with country not containing pattern ‘land’.**

SELECT \* FROM employee WHERE COUNTRY NOT LIKE '%LAND%';

**Output:**

| ***E\_ID*** | ***E\_NAME*** | ***JOB*** | ***COUNTRY*** | ***CITY*** | ***SALARY*** |
| --- | --- | --- | --- | --- | --- |
| *1* | *Phura* | *Designer* | *Germany* | *Berlin* | *30000* |
| *2* | *Surya* | *Assistant* | *India* | *Delhi* | *10000* |
| *3* | *Santosh* | *Programmer* | *China* | *Beijing* | *20000* |
| *4* | *Salim* | *Manager* | *Nepal* | *Kathmandu* | *40000* |
| *5* | *Sachin* | *Doctor* | *Italy* | *Milan* | *50000* |
| *6* | *Gaurab* | *Analyst* | *USA* | *New York* | *45000* |
| *7* | *John* | *Clerk* | *USA* | *Chicago* | *35000* |
| *8* | *Jane* | *Accountant* | *Australia* | *Sydney* | *38000* |
| *9* | *Bob* | *Technician* | *Canada* | *Toronto* | *28000* |
| *10* | *Alice* | *Engineer* | *Germany* | *Munich* | *42000* |

**13) Select all employees with city starting with any character followed by “erlin”.**

SELECT \* FROM employee WHERE CITY LIKE '\_ERLIN%';

**Output:**

| **E\_ID** | **E\_NAME** | **JOB** | **COUNTRY** | **CITY** | **SALARY** |
| --- | --- | --- | --- | --- | --- |
| 1 | Phura | Designer | Germany | Berlin | 30000 |

**14) Select all employees with a city starting with ‘l’, followed by any character, followed by ‘n’, followed by ‘any’ character, followed by ‘on’.**

SELECT \* FROM employee WHERE CITY LIKE 'L\_N\_ON%';

**Output:**  
*(No matching records in the sample data.)*

**15) Select all employees with a city starting with ‘b’,‘m’ or ‘d’.**

SELECT \* FROM employee WHERE CITY LIKE 'B%' OR CITY LIKE 'M%' OR CITY LIKE 'D%';

**Output:**

| **E\_ID** | **E\_NAME** | **JOB** | **COUNTRY** | **CITY** | **SALARY** |
| --- | --- | --- | --- | --- | --- |
| 1 | Phura | Designer | Germany | Berlin | 30000 |
| 2 | Surya | Assistant | India | Delhi | 10000 |
| 5 | Sachin | Doctor | Italy | Milan | 50000 |
| 10 | Alice | Engineer | Germany | Munich | 42000 |

**16) Select all employees with a city starting with ‘a’,‘b’ or ‘c’.**

SELECT \* FROM employee WHERE CITY LIKE 'A%' OR CITY LIKE 'B%' OR CITY LIKE 'C%';

**Output:**

| **E\_ID** | **E\_NAME** | **JOB** | **COUNTRY** | **CITY** | **SALARY** |
| --- | --- | --- | --- | --- | --- |
| 3 | Santosh | Programmer | China | Beijing | 20000 |
| 7 | John | Clerk | USA | Chicago | 35000 |

**17) Select all employees with a city not starting with ‘b’,‘m’ or ‘d’.**

SELECT \* FROM employee

WHERE CITY NOT LIKE 'B%' AND CITY NOT LIKE 'M%' AND CITY NOT LIKE 'D%';

**Output:**

| ***E\_ID*** | ***E\_NAME*** | ***JOB*** | ***COUNTRY*** | ***CITY*** | ***SALARY*** |
| --- | --- | --- | --- | --- | --- |
| *4* | *Salim* | *Manager* | *Nepal* | *Kathmandu* | *40000* |
| *6* | *Gaurab* | *Analyst* | *USA* | *New York* | *45000* |
| *7* | *John* | *Clerk* | *USA* | *Chicago* | *35000* |
| *8* | *Jane* | *Accountant* | *Australia* | *Sydney* | *38000* |
| *9* | *Bob* | *Technician* | *Canada* | *Toronto* | *28000* |

**18) Select all employees with a city of ‘Delhi’ or ‘Manchester’ use IN operator.**

SELECT \* FROM employee WHERE CITY IN ('Delhi', 'Manchester');

**Output:**

| **E\_ID** | **E\_NAME** | **JOB** | **COUNTRY** | **CITY** | **SALARY** |
| --- | --- | --- | --- | --- | --- |
| 2 | Surya | Assistant | India | Delhi | 10000 |

**19) Select all employees with salary BETWEEN 20000 and 35000.**

SELECT \* FROM employee WHERE SALARY BETWEEN 20000 AND 35000;

**Output:**

| **E\_ID** | **E\_NAME** | **JOB** | **COUNTRY** | **CITY** | **SALARY** |
| --- | --- | --- | --- | --- | --- |
| 1 | Phura | Designer | Germany | Berlin | 30000 |
| 3 | Santosh | Programmer | China | Beijing | 20000 |
| 7 | John | Clerk | USA | Chicago | 35000 |
| 9 | Bob | Technician | Canada | Toronto | 28000 |

**20) Select all employees with salary BETWEEN 10000 and 40000, but employees with a ID of 1,2, or 3 should not be displayed.**

SELECT \* FROM employee

WHERE SALARY BETWEEN 10000 AND 40000

AND E\_ID NOT IN (1, 2, 3);

**Output:**

| **E\_ID** | **E\_NAME** | **JOB** | **COUNTRY** | **CITY** | **SALARY** |
| --- | --- | --- | --- | --- | --- |
| 4 | Salim | Manager | Nepal | Kathmandu | 40000 |
| 7 | John | Clerk | USA | Chicago | 35000 |
| 9 | Bob | Technician | Canada | Toronto | 28000 |

**21) Select all employees with city beginning with any of the letter BETWEEN ‘b’ and ‘m’.**

SELECT \* FROM employee

WHERE SUBSTRING(CITY, 1, 1) BETWEEN 'B' AND 'M';

**Output:**

| **E\_ID** | **E\_NAME** | **JOB** | **COUNTRY** | **CITY** | **SALARY** |
| --- | --- | --- | --- | --- | --- |
| 1 | Phura | Designer | Germany | Berlin | 30000 |
| 3 | Santosh | Programmer | China | Beijing | 20000 |
| 8 | Jane | Accountant | Australia | Sydney | 38000 |

**22) Select all employees with city beginning with any of the letter NOT BETWEEN ‘b’ and ‘m’.**

SELECT \* FROM employee

WHERE SUBSTRING(CITY, 1, 1) NOT BETWEEN 'B' AND 'M';

**Output:**

| **E\_ID** | **E\_NAME** | **JOB** | **COUNTRY** | **CITY** | **SALARY** |
| --- | --- | --- | --- | --- | --- |
| 1 | Phura | Designer | Germany | Berlin | 30000 |
| 2 | Surya | Assistant | India | Delhi | 10000 |
| 3 | Santosh | Programmer | China | Beijing | 20000 |
| 5 | Sachin | Doctor | Italy | Milan | 50000 |
| 10 | Alice | Engineer | Germany | Munich | 42000 |

**Conclusion**

This lab demonstrated advanced SQL filtering techniques for precise data retrieval. Key operations included:

* **Pattern matching** (LIKE, NOT LIKE) for flexible text searches.
* **Range conditions** (BETWEEN) for numeric and alphabetic filtering.
* **Set operations** (IN) for matching specific values.

These skills are essential for efficient database querying in real-world applications.

**Report Generated on 2025-08-03**  
*(Note: Outputs are based on the provided sample data.)*

Let me know if you want it saved as a .docx file — I can generate and send the download link.