

A PRACTICAL FILE ON

Database Management Systems Practical File

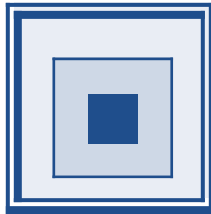
BCA · SQL practical file

Submitted in partial fulfilment of the practical examination requirements

BACHELOR OF COMPUTER APPLICATIONS

in

BCA IV Sem



SUBMITTED BY

Simran Kaur

Roll No. BCA24-062

PROGRAMME

BCA IV Sem

Department of Computer Applications

YOUR COLLEGE OF MANAGEMENT & IT, CHANDIGARH

Session 2025–26

CERTIFICATE

This is to certify that Simran Kaur, a student of BCA IV Sem, has satisfactorily completed the practical work in Database Management Systems prescribed by the Department of Computer Applications during the academic session 2025–26.

The experiments recorded in this file were performed by the student in the laboratory under my supervision, and the observations and results are the student's own.

Mr. Harpreet Singh
Subject Teacher

External Examiner
Date: _____

Dr. Neelam Arora
Head of Department

| Acknowledgement

I would like to express my sincere gratitude to my subject teacher, Mr. Harpreet Singh, for guidance and encouragement throughout the practical sessions, and for patiently clearing my doubts in the laboratory.

I am also thankful to Dr. Neelam Arora, Head of Department, for providing a well-equipped laboratory, and to the laboratory staff for their help in arranging the apparatus for every experiment.

Finally, I thank my parents and classmates for their constant support while I completed this practical file.

Simran Kaur

Index

S. No.	Name of the Experiment	Date	Remarks	Teacher's Sign
1	Creating tables and inserting records	14 Jan 2026		
2	Queries with WHERE, ORDER BY and LIKE	21 Jan 2026		
3	Aggregate functions and GROUP BY	28 Jan 2026		
4	Joins between two tables	04 Feb 2026		
5	Views, updates and deletes	11 Feb 2026		

Write the page number of each experiment in the remarks column once the file is complete.

Experiment 1: Creating tables and inserting records

Date: 14 Jan 2026

Page No.: ____

Teacher's sign: _____

Aim

To create the tables EMPLOYEE and DEPARTMENT with suitable constraints and insert records.

Software / Requirements

- MySQL 8 / MariaDB
- MySQL Workbench or command-line client

Algorithm

1. Create the database and select it.
2. Create DEPARTMENT with a primary key.
3. Create EMPLOYEE with a foreign key to DEPARTMENT.
4. Insert sample records and display them.

Program

SQL (MySQL 8)

```
CREATE TABLE department (
    dept_id INT PRIMARY KEY,
    dept_name VARCHAR(30) NOT NULL
);
CREATE TABLE employee (
    emp_id INT PRIMARY KEY,
    name VARCHAR(40) NOT NULL,
    salary DECIMAL(10,2) CHECK (salary > 0),
    dept_id INT,
    FOREIGN KEY (dept_id) REFERENCES department(dept_id)
);
INSERT INTO department VALUES (10, 'Sales'), (20, 'Accounts'), (30, 'IT');
INSERT INTO employee VALUES
(1, 'Ritu Sen', 42000, 10), (2, 'Amit Rao', 55000, 30), (3, 'Neha Jain', 38000, 20);
SELECT * FROM employee;
```

Output

Sample output

emp_id	name	salary	dept_id
1	Ritu Sen	42000.00	10
2	Amit Rao	55000.00	30
3	Neha Jain	38000.00	20

Result

Result

Both tables were created with constraints and the records were inserted successfully.

Experiment 2: Queries with WHERE, ORDER BY and LIKE

Date: 21 Jan 2026

Page No.: ____

Teacher's sign: _____

Aim

To retrieve records using conditions, pattern matching and sorting.

Software / Requirements

- MySQL 8

Algorithm

- Select employees earning more than a given salary.
- Find names starting with a given letter using LIKE.
- Sort the output by salary in descending order.

Program

SQL (MySQL 8)

```
SELECT name, salary FROM employee
WHERE salary > 40000
ORDER BY salary DESC;
```

```
SELECT name FROM employee WHERE name LIKE 'N%';
```

Output

Sample output

```
+-----+-----+
| name   | salary |
+-----+-----+
| Amit Rao | 55000.00 |
| Ritu Sen | 42000.00 |
+-----+-----+
+-----+
| name   |
+-----+
| Neha Jain |
+-----+
```

Result

Result

The queries return the expected filtered and sorted results.

Experiment 3: Aggregate functions and GROUP BY

Date: 28 Jan 2026

Page No.: ____

Teacher's sign: _____

Aim

To use aggregate functions with GROUP BY and HAVING clauses.

Software / Requirements

- MySQL 8

Algorithm

- Count employees in each department.
- Find the average salary per department.
- Use HAVING to show departments with average salary above a limit.

Program

SQL (MySQL 8)

```
SELECT dept_id, COUNT(*) AS staff, AVG(salary) AS avg_salary
FROM employee
GROUP BY dept_id
HAVING AVG(salary) > 40000;
```

Output

Sample output

dept_id	staff	avg_salary
10	1	42000.00
30	1	55000.00

Result

Result

Aggregate values are computed correctly per group.

Experiment 4: Joins between two tables

Date: 04 Feb 2026

Page No.: ____

Teacher's sign: _____

Aim

To display employee names with their department names using an inner join.

Software / Requirements

- MySQL 8

Algorithm

- Write an INNER JOIN on dept_id.
- Write a LEFT JOIN to include departments without employees.
- Compare the outputs.

Program

SQL (MySQL 8)

```
SELECT e.name, d.dept_name
FROM employee e
JOIN department d ON e.dept_id = d.dept_id;
```

Output

Sample output

```
+-----+-----+
| name   | dept_name |
+-----+-----+
| Ritu Sen | Sales     |
| Neha Jain | Accounts  |
| Amit Rao | IT        |
+-----+-----+
```

Result

Result

The join combines rows from both tables based on the common column.

Experiment 5: Views, updates and deletes

Date: 11 Feb 2026

Page No.: ____

Teacher's sign: _____

Aim

To create a view, update records and delete records with conditions.

Software / Requirements

- MySQL 8

Algorithm

- Create a view of high earners.
- Increase salaries of one department by 10%.
- Delete an employee by id and display the view again.

Program

SQL (MySQL 8)

```
CREATE VIEW high_earners AS
  SELECT name, salary FROM employee WHERE salary >= 40000;
UPDATE employee SET salary = salary * 1.10 WHERE dept_id = 20;
DELETE FROM employee WHERE emp_id = 1;
SELECT * FROM high_earners;
```

Output

Sample output

```
+-----+-----+
| name   | salary |
+-----+-----+
| Amit Rao | 55000.00 |
| Neha Jain | 41800.00 |
+-----+-----+
```

Result

Result

The view reflects updates and deletions in the base table.

| Bibliography

- Laboratory manual issued by the department / school.
- Prescribed textbook for the subject (latest edition).
- Class notes and practical demonstrations by the subject teacher.