

SCHOOL OF COMPUTER SCIENCE AND ARTIFICIAL INTELLIGENCE		DEPARTMENT OF COMPUTER SCIENCE ENGINEERING	
Program Name: B. Tech		Assignment Type: Lab	Academic Year:2025-2026
Course Coordinator Name		Venkataramana Veeramsetty	
Instructor(s) Name		Dr. V. Venkataramana (Co-ordinator)	
		Dr. T. Sampath Kumar	
		Dr. Pramoda Patro	
		Dr. Brij Kishor Tiwari	
		Dr.J.Ravichander	
		Dr. Mohammand Ali Shaik	
		Dr. Anirodh Kumar	
		Mr. S.Naresh Kumar	
		Dr. RAJESH VELPULA	
		Mr. Kundhan Kumar	
		Ms. Ch.Rajitha	
		Mr. M Prakash	
		Mr. B.Raju	
		Intern 1 (Dharma teja)	
		Intern 2 (Sai Prasad)	
		Intern 3 (Sowmya)	
NS_2 (Mounika)			
Course Code	24CS002PC215	Course Title	AI Assisted Coding
Year/Sem	II/I	Regulation	R24
Date and Day of Assignment	Week9 - Tuesday	Time(s)	
Duration	2 Hours	Applicable to Batches	
AssignmentNumber:16.2(Present assignment number)/24(Total number of assignments)			
Q.No.	Question		Expected Time to complete
1	1.1. Display all records from the employee’s table. Prompt: Display all records from the employee’s table. <div>--1.1-- SELECT * FROM employees;</div>		Week9 - Monday

Output:

mysql Localhost: SELECT * FROM em... X

emp_id	first_name	last_name	department	salary	hire_date
1	Amit	Sharma	HR	45000	2020-05-20T00:00:00.000Z
2	Priya	Patel	Finance	60000	2021-02-10T00:00:00.000Z
3	Ravi	Kumar	IT	55000	2019-08-14T00:00:00.000Z
4	Neha	Reddy	Marketing	48000	2022-01-05T00:00:00.000Z
5	Arjun	Singh	IT	62000	2020-09-12T00:00:00.000Z

2.2. Display only employee names and their departments.

Prompt: Display only employee names and their departments

```
--2.2--  
SELECT first_name, last_name, department  
FROM employees;
```

Output:

first_name	last_name	department
Amit	Sharma	HR
Priya	Patel	Finance
Ravi	Kumar	IT
Neha	Reddy	Marketing
Arjun	Singh	IT

2. 3. Show unique department names.

Prompt: Show unique department names

```
3.3--  
SELECT DISTINCT department FROM employees;
```

Output:






department
HR
Finance
IT
Marketing

3. 4. Find employees with salary greater than 50000.

Prompt: Find employees with salary greater than 50000




```
4.4--  
SELECT * FROM employees WHERE salary > 50000;
```

Output:

emp_id	first_name	last_name	department	salary
 Filter...	 Filter...	 Filter...	 Filter...	 Filter...
2	Priya	Patel	Finance	60000
3	Ravi	Kumar	IT	55000
5	Arjun	Singh	IT	62000


4. 5. Find employees from the IT department.






Prompt: Find employees from the IT department

emp_id	first_name	last_name	department	salary
 Filter...	 Filter...	 Filter...	 Filter...	 Filter...
3	Ravi	Kumar	IT	55000
5	Arjun	Singh	IT	62000

5. 6. Display employees hired after 2020.

Prompt: Display employees hired after 2020

 MySQL Live: SELECT * FROM em... X

emp_id	first_name	last_name	department	salary
 Filter...	 Filter...	 Filter...	 Filter...	 Filter...
2	Priya	Patel	Finance	60000
4	Neha	Reddy	Marketing	48000

6. 7. Show employees in ascending order of salary.

Prompt: Show employees in ascending order of salary

 MySQL Live: SELECT * FROM em... X

emp_id	first_name	last_name	department	salary
 Filter...	 Filter...	 Filter...	 Filter...	 Filter...
1	Amit	Sharma	HR	45000
4	Neha	Reddy	Marketing	48000
3	Ravi	Kumar	IT	55000
2	Priya	Patel	Finance	60000
5	Arjun	Singh	IT	62000

7. 8. Show top 3 highest-paid employees.

Prompt: Show top 3 highest-paid employees

MySQL Live: SELECT * FROM em... X

emp_id	first_name	last_name	department	salary
abc Filter...	abc Filter...	abc Filter...	abc Filter...	abc F
5	Arjun	Singh	IT	62000
2	Priya	Patel	Finance	60000
3	Ravi	Kumar	IT	55000

8. 9. Count total employees in the company.

Prompt: Count total employees in the company

total_employees
abc Filter...
5

9. 10. Find the average salary of employees.

Prompt: Find the average salary of employees

average_salary
abc Filter...
54000

10. 11. Find the highest and lowest salary.

Prompt: Find the highest and lowest salary.

MySQL Live: SELECT MAX(salar... X

highest_salary	lowest_salary
abc Filter...	abc Filter...
62000	45000

11. 12. Find total salary expenditure per department.

Prompt: Find total salary expenditure per department.

MySQL Live: SELECT departmen... X		
department	total_salary	
abc Filter...	abc Filter...	
HR	45000	
Finance	60000	
IT	117000	
Marketing	48000	

12. 13. Display departments having more than one employee.

Prompt: Display departments having more than one employee.

MySQL Live: SELECT departmen... X		
department	emp_count	
abc Filter...	abc Filter...	
IT	2	

13. 14. Show average salary by department.

Prompt: Show average salary by department.

department	avg_salary	
abc Filter...	abc Filter...	
HR	45000	
Finance	60000	
IT	58500	
Marketing	48000	

14. 15. Count employees hired each year.

Prompt: Count employees hired each year.

hire_year	employees_hired	
abc Filter...	abc Filter...	
2020	2	
2021	1	
2019	1	
2022	1	

15. 16. List employees with their department locations.

Prompt: List employees with their department locations.

MySQL Live: SELECT e.first_n... X			
first_name	last_name	department	location
abc Filter...	abc Filter...	abc Filter...	abc Filter...
No data			

16. 17. Find employees working in Bangalore.

Prompt: Find employees working in Bangalore.

first_name	last_name	
abc Filter...	abc Filter...	
No data		

17. 18. Display all employees even if they don't belong to a department.

Prompt: Display all employees even if they don't belong to a department.

first_name	last_name	dept_name	
abc Filter...	abc Filter...	abc Filter...	
Amit	Sharma	NULL	
Priya	Patel	NULL	
Ravi	Kumar	NULL	
Neha	Reddy	NULL	
Arjun	Singh	NULL	

18. 19. Find departments with no employees.

Prompt: Find departments with no employees.

dept_name	
abc Filter...	
No data	

19. 20. Count employees in each department.

Prompt: Count employees in each department.

department	emp_count	
abc Filter...	abc Filter...	
HR	1	
Finance	1	
IT	2	
Marketing	1	

20. 21. Find employees earning above average salary.

Prompt: Find employees earning above average salary

emp_id	first_name	last_name	department	salary
abc Filter...	abc Filter...	abc Filter...	abc Filter...	abc Filter...
2	Priya	Patel	Finance	60000
3	Ravi	Kumar	IT	55000
5	Arjun	Singh	IT	62000

21. 22. Find the department with the highest average salary.

Prompt: Find the department with the highest average salary.

department	avg_salary	
abc Filter...	abc Filter...	
Finance	60000	

22. 23. Find employees hired most recently.

Prompt: Find employees hired most recently.

emp_id	first_name	last_name	department	salary
abc Filter...	abc Filter...	abc Filter...	abc Filter...	abc Filter...
4	Neha	Reddy	Marketing	48000

23. 24. Find employees earning the second highest salary.

Prompt: Find employees earning the second highest salary.

emp_id	first_name	last_name	department	salary
abc Filter...	abc Filter...	abc Filter...	abc Filter...	abc Filter...
2	Priya	Patel	Finance	60000

24. 25. Find all employees in the same department as 'Amit Sharma'.

Prompt: Find all employees in the same department as 'Amit Sharma'.

emp_id	first_name	last_name	department	salary
abc Filter...	abc Filter...	abc Filter...	abc Filter...	abc Filter...
1	Amit	Sharma	HR	45000

25. 26. Increase salary by 10% for IT employees.
Prompt: Increase salary by 10% for IT employees.
No output
26. 27. Change department of employee 'Ravi' to Marketing.
Prompt: Change department of employee 'Ravi' to Marketing
No output
27. 28. Delete employees with salary below 40000.
Prompt: Delete employees with salary below 40000.
No output
28. 29. Add a new column 'email' to employees.
Prompt: Add a new column 'email' to employees
No output
29. 30. Update email IDs for all employees.
Prompt: Update email IDs for all employees.
No output
30. 31. Find top 2 departments by average salary.
Prompt: Find top 2 departments by average salary.

department	avg_salary	
abc Filter...	abc Filter...	
IT	68200	
Finance	60000	

31. 32. Find how many employees work in each city.
Prompt: Find how many employees work in each city.

location	emp_count	
abc Filter...	abc Filter...	
No data		

32. 33. Show employee count and total salary together.
Prompt: Show employee count and total salary together.

emp_count	total_salary
abc Filter...	abc Filter...
5	281700

33. 34. Display employees with names starting with 'A'.

Prompt: Display employees with names starting with 'A'.

emp_id	first_name	last_name	department	salary
abc Filter...	abc Filter...	abc Filter...	abc Filter...	abc Filter...
1	Amit	Sharma	HR	45000
5	Arjun	Singh	IT	68200

34. 35. Display employees whose last name ends with 'a'.

Prompt: Display employees whose last name ends with 'a'.

emp_id	first_name	last_name	department	salary
abc Filter...	abc Filter...	abc Filter...	abc Filter...	abc Filter...
1	Amit	Sharma	HR	45000

35. 36. Find employees hired in 2020.

Prompt: Find employees hired in 2020.

emp_id	first_name	last_name	department	salary
abc Filter...	abc Filter...	abc Filter...	abc Filter...	abc Filter...
1	Amit	Sharma	HR	45000
5	Arjun	Singh	IT	68200

36. 37. Show number of days since each employee was hired.

Prompt: Show number of days since each employee was hired.

first_name	last_name	days_since_hired
abc Filter...	abc Filter...	abc Filter...
Amit	Sharma	1967
Priya	Patel	1701
Ravi	Kumar	2247
Neha	Reddy	1372
Arjun	Singh	1852

37. 38. Display employee names in uppercase.

Prompt: Display employee names in uppercase.

MySQL Live: SELECT UPPER(fir... X		
first_name	last_name	
abc Filter...	abc Filter...	
AMIT	SHARMA	
PRIYA	PATEL	
RAVI	KUMAR	
NEHA	REDDY	
ARJUN	SINGH	

38. 39. Concatenate first and last names.

Prompt: Concatenate first and last names.

full_name	
abc Filter...	
Amit Sharma	
Priya Patel	
Ravi Kumar	
Neha Reddy	
Arjun Singh	

39. 40. Find employees whose salary is between 45000 and 60000.

Prompt: Find employees whose salary is between 45000 and 60000.

emp_id	first_name	last_name	department	salary
abc Filter...	abc Filter...	abc Filter...	abc Filter...	abc Filter...
1	Amit	Sharma	HR	45000
2	Priya	Patel	Finance	60000
4	Neha	Reddy	Marketing	48000

40. 41. Create a view for high salary employees (>55000).

Prompt: Create a view for high salary employees (>55000).

No output

41. 42. Display all records from the view.

Prompt: Display all records from the view.

emp_id	first_name	last_name	department	salary
2	Priya	Patel	Finance	6000
3	Ravi	Kumar	Marketing	6050
5	Arjun	Singh	IT	6820

42. 43. Add NOT NULL constraint to department name.

Prompt: Add NOT NULL constraint to department name.

No output

43. 44. Drop the view.

Prompt: Drop the view

No output

44. 45. Rename the employees table to staff.

Prompt: Rename the employees table to staff.

No output

45. 46. Create a backup copy of the employees table.

Prompt: Create a backup copy of the employees table.

Error

46. 47. Delete all data but keep the structure.

Prompt: Delete all data but keep the structure.

error

47. 48. Drop the employees backup table.

Prompt: Drop the employees backup table.

error

48. 49. Create an index on employee last name.

Prompt: Create an index on employee last name.

error

49. 50. Drop the index.

Prompt: Drop the index.

Error

CODE:

▶ Run on active connection | ≡ Select block

```
1 CREATE DATABASE company;
2 USE company;
3 CREATE TABLE employees (
4   emp_id INT AUTO_INCREMENT PRIMARY KEY,
5   first_name VARCHAR(50),
6   last_name VARCHAR(50),
7   department VARCHAR(50),
8   salary DECIMAL(10, 2),
9   hire_date DATE
10 );
11 INSERT INTO employees (first_name, last_name, department, salary, hire_date)
12 VALUES
13   ('Amit', 'Sharma', 'HR', 45000, '2020-05-20'),
14   ('Priya', 'Patel', 'Finance', 60000, '2021-02-10'),
15   ('Ravi', 'Kumar', 'IT', 55000, '2019-08-14'),
16   ('Neha', 'Reddy', 'Marketing', 48000, '2022-01-05'),
17   ('Arjun', 'Singh', 'IT', 62000, '2020-09-12');
18 SELECT * FROM employees;
19 SELECT first_name, last_name, department
20 FROM employees;
21 SELECT DISTINCT department FROM employees;
22 SELECT * FROM employees WHERE salary > 50000;
23 SELECT * FROM employees WHERE department = 'IT';
24 SELECT * FROM employees WHERE hire_date > '2020-12-31';
25 SELECT * FROM employees ORDER BY salary ASC;
26 SELECT * FROM employees ORDER BY salary DESC LIMIT 3;
27 SELECT COUNT(*) AS total_employees FROM employees;
28 SELECT AVG(salary) AS average_salary FROM employees;
29 -- 11--
30 SELECT MAX(salary) AS highest_salary, MIN(salary) AS lowest_salary FROM employees;
31 ---12--
32 SELECT department, SUM(salary) AS total_salary
33 FROM employees
34 GROUP BY department;
```

```

35  ---13--
36  SELECT department, COUNT(*) AS emp_count
37  FROM employees
38  GROUP BY department
39  HAVING COUNT(*) > 1;
40  ---14--
41  SELECT department, AVG(salary) AS avg_salary
42  FROM employees
43  GROUP BY department;
44  ---15--
45  SELECT YEAR(hire_date) AS hire_year, COUNT(*) AS employees_hired
46  FROM employees
47  GROUP BY YEAR(hire_date);
48  ---16--
49  -- (You'll need another table like departments for this.)
50  CREATE TABLE departments (
51  |   dept_name VARCHAR(50),
52  |   location VARCHAR(50)
53  );
54  -- Example join:
55  SELECT e.first_name, e.last_name, e.department, d.location
56  FROM employees e
57  JOIN departments d ON e.department = d.dept_name
58  ---17--
59  SELECT e.first_name, e.last_name
60  FROM employees e
61  JOIN departments d ON e.department = d.dept_name
62  WHERE d.location = 'Bangalore';
63  ---18--
64  SELECT e.first_name, e.last_name, d.dept_name
65  FROM employees e
66  LEFT JOIN departments d ON e.department = d.dept_name;
67  ---19--
68  SELECT d.dept_name

```

```
69 FROM departments d
70 LEFT JOIN employees e ON d.dept_name = e.department
71 WHERE e.emp_id IS NULL;
72 ---20--
73 SELECT department, COUNT(*) AS emp_count
74 FROM employees
75 GROUP BY department;
76 ---21--
77 SELECT *
78 FROM employees
79 WHERE salary > (SELECT AVG(salary) FROM employees);
80 --22--
81 SELECT department, AVG(salary) AS avg_salary
82 FROM employees
83 GROUP BY department
84 ORDER BY avg_salary DESC
85 LIMIT 1;
86 ---23--
87 SELECT * FROM employees ORDER BY hire_date DESC LIMIT 1;
88 ---24--
89 SELECT *
90 FROM employees
91 WHERE salary = (
92     SELECT MAX(salary)
93     FROM employees
94     WHERE salary < (SELECT MAX(salary) FROM employees)
95 );
96 ---25--
97 SELECT *
98 FROM employees
99 WHERE department = (
00     SELECT department FROM employees
01     WHERE first_name = 'Amit' AND last_name = 'Sharma'
02 );
```

```
03  ---26--
04  UPDATE employees
05  SET salary = salary * 1.10
06  WHERE department = 'IT';
07  ---27--
08  UPDATE employees
09  SET department = 'Marketing'
10  WHERE first_name = 'Ravi';
11  ---28--
12  DELETE FROM employees
13  WHERE salary < 40000;
14  ---29--
15  ALTER TABLE employees ADD COLUMN email VARCHAR(100);
16  ---30--
17  UPDATE employees
18  SET email = CONCAT(LOWER(first_name), '.', LOWER(last_name), '@company.com')
19  ---31--
20  SELECT department, AVG(salary) AS avg_salary
21  FROM employees
22  GROUP BY department
23  ORDER BY avg_salary DESC
24  LIMIT 2;
25  ---32--
26  -- (Assumes departments has a location column.)
27  SELECT d.location, COUNT(e.emp_id) AS emp_count
28  FROM employees e
29  JOIN departments d ON e.department = d.dept_name
30  GROUP BY d.location;
31  ---33--
32  SELECT COUNT(*) AS emp_count, SUM(salary) AS total_salary FROM employees;
33  ---34--
34  SELECT * FROM employees WHERE first_name LIKE 'A%';
35  ---35--
36  SELECT * FROM employees WHERE last_name LIKE '%a';
```

```

137  ---36--
138  SELECT * FROM employees
139  WHERE YEAR(hire_date) = 2020;
140  ---37--
141  SELECT first_name, last_name, DATEDIFF(CURDATE(), hire_date) AS days_since_hired
142  FROM employees;
143  ---38--
144  SELECT UPPER(first_name) AS first_name, UPPER(last_name) AS last_name FROM employees;
145  ---39--
146  SELECT CONCAT(first_name, ' ', last_name) AS full_name FROM employees;
147  ---40--
148  SELECT * FROM employees
149  WHERE salary BETWEEN 45000 AND 60000;
150  ---41--
151  CREATE VIEW high_salary_employees AS
152  SELECT * FROM employees WHERE salary > 55000;
153  ---42--
154  SELECT * FROM high_salary_employees;
155  ---43--
156  ALTER TABLE employees
157  MODIFY department VARCHAR(50) NOT NULL;
158  ---44--
159  DROP VIEW high_salary_employees;
160  ---45--
161  RENAME TABLE employees TO staff;
162  ---46--
163  CREATE TABLE employees_backup AS SELECT * FROM employees;
164  ---47--
165  TRUNCATE TABLE employees;
166  ---48--
167  DROP TABLE employees_backup;
168  ---49--
169  CREATE INDEX idx_lastname ON employees(last_name);
170  ---50--

```

Employee Table

emp_id	first_name	last_name	department	salary	hire_date
1	Amit	Sharma	HR	45000	2020-05-20
2	Priya	Patel	Finance	60000	2021-02-10
3	Ravi	Kumar	IT	55000	2019-08-14
4	Neha	Reddy	Marketing	48000	2022-01-05
5	Arjun	Singh	IT	62000	2020-09-12

Department Table

dept_id	dept_name	location
1	HR	Hyderabad
2	Finance	Mumbai
3	IT	Bangalore
4	Marketing	Chennai
5	Operations	Delhi