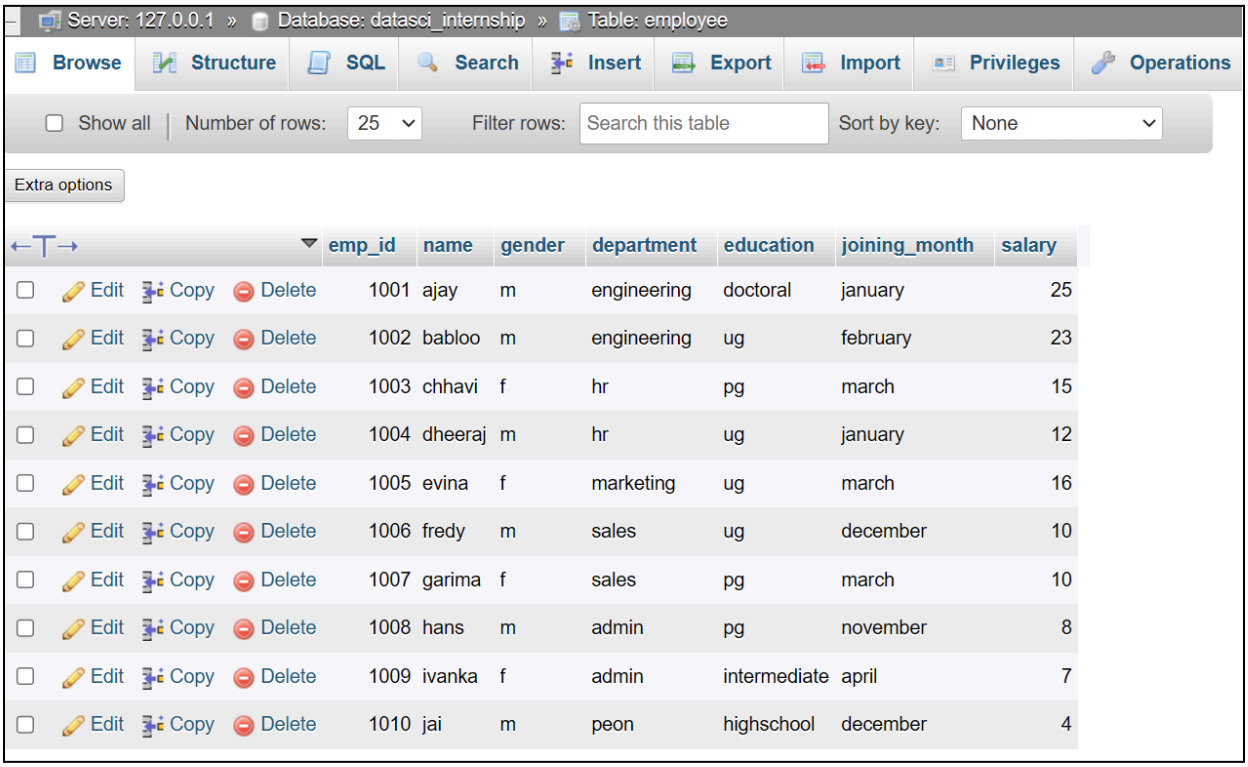
```
create table employee (emp_id int(5), name varchar(20), gender char(1), department
varchar(10), education varchar(10), joining_month varchar(10), salary int(10));
INSERT INTO employee(emp_id, name, gender, department, education, joining_month, salary)
VALUES
    (1001, 'ajay', 'm', 'engineering', 'doctoral', 'january', 25),
    (1002, 'babloo', 'm', 'engineering', 'ug', 'february', 23),
    (1003, 'chhavi', 'f', 'hr', 'pg', 'march', 15),
    (1004, 'dheeraj', 'm', 'hr', 'ug', 'january', 12),
    (1005, 'evina', 'f', 'marketing', 'ug', 'march', 16),
    (1006, 'fredy', 'm', 'sales', 'ug', 'december', 10),
    (1007, 'garima', 'f', 'sales', 'pg', 'march', 10),
    (1008, 'hans', 'm', 'admin', 'pg', 'november', 8),
    (1009, 'ivanka', 'f', 'admin', 'intermediate', 'april', 7),
    (1010, 'jai', 'm', 'peon', 'highschool', 'december', 4);
```



	emp_id	name	gender	department	education	joining_month	salary
Edit Copy Delete	1001	ajay	m	engineering	doctoral	january	25
Edit Copy Delete	1002	babloo	m	engineering	ug	february	23
Edit Copy Delete	1003	chhavi	f	hr	pg	march	15
Edit Copy Delete	1004	dheeraj	m	hr	ug	january	12
Edit Copy Delete	1005	evina	f	marketing	ug	march	16
Edit Copy Delete	1006	fredy	m	sales	ug	december	10
Edit Copy Delete	1007	garima	f	sales	pg	march	10
Edit Copy Delete	1008	hans	m	admin	pg	november	8
Edit Copy Delete	1009	ivanka	f	admin	intermediate	april	7
Edit Copy Delete	1010	jai	m	peon	highschool	december	4

QUERIES:

1. Find the department in which SUM salary is greater than or equal to 20 lacs, but the education of employees is not UG.

```
-> SELECT department FROM employee WHERE education!= 'UG' GROUP BY department
HAVING SUM(salary) >= 20;
```

2. Find the departments in which the SUM of the salaries is greater than or equal to 15 lacs and arrange the Salary in descending Order.

-> SELECT department, SUM(salary) AS total_salary FROM employee GROUP BY department HAVING SUM(salary) >= 15 ORDER BY total_salary DESC;

department	total_salary 1
engineering	48
hr	27
sales	20
marketing	16
admin	15

3. Write the query to select all departments whose average is greater than 35 from the 'salary' column in the employee table.

-> SELECT department FROM employee GROUP BY department HAVING AVG(salary) > 35;

MySQL returned an empty result set (i.e. zero rows). (Query took 0.0016 seconds.)

```
SELECT department FROM employee GROUP BY department HAVING AVG(salary) > 35;
```

4. Write the query to find the employee's name whose salary is between 14 and 24 in the 'employee' table.

-> SELECT name FROM employee WHERE salary BETWEEN 14 AND 24;

name
babloo
chhavi
evina

5. Write an SQL query to display the total salary of each employee adding the Salary with Variable value. Consider variable value is 5
-> SELECT emp_id, name, salary + 5 AS total_salary FROM employee;

emp_id	name	total_salary
1001	ajay	30
1002	babloo	28
1003	chhavi	20
1004	dheeraj	17
1005	evina	21
1006	fredy	15
1007	garima	15
1008	hans	13
1009	ivanka	12
1010	jai	9

---

emp_id	first_name	last_name	dept	age	city	salary
1	Ajay	Kumar	Engineering	30	Mumbai	25000.00
2	Babloo	Singh	Engineering	28	Delhi	23000.00
3	Chhavi	Sharma	HR	32	Bangalore	15000.00
4	Dheeraj	Verma	HR	29	Hyderabad	12000.00
5	Evina	Thomas	Marketing	26	Chennai	16000.00
6	Fredy	Fernandes	Sales	27	Pune	10000.00

QUERIES:

1. Write SQL query where the city contains the value “ai”.
   SELECT * FROM VCET_employee WHERE city LIKE '%ai%';

emp_id	first_name	last_name	dept	age	city	salary
1	Ajay	Kumar	Engineering	30	Mumbai	25000.00
5	Evina	Thomas	Marketing	26	Chennai	16000.00

2. Write a SQL query where the first name starts from “F”.
   SELECT * FROM VCET_employee WHERE first_name LIKE 'F%';

emp_id	first_name	last_name	dept	age	city	salary
6	Fredy	Fernandes	Sales	27	Pune	10000.00

3. Retrieve emp_id, first_name, last_name and city whose name contains “ee” in second substring
   SELECT emp_id, first_name, last_name, city FROM VCET_employee WHERE first_name LIKE '%ee%';

emp_id	first_name	last_name	city
4	Dheeraj	Verma	Hyderabad

4. SQL Query to get First_name and Last_name of customers who have cities in Mumbai and Delhi.
   SELECT first_name, last_name FROM VCET_employee WHERE city IN ('Mumbai', 'Delhi');

first_name	last_name
Ajay	Kumar
Babloo	Singh

5. Write SQL query for members whose age is greater than 25 yrs and location is Mumbai.
   SELECT * FROM VCET_employee WHERE age > 25 AND city = 'Mumbai';

emp_id	first_name	last_name	dept	age	city	salary
1	Ajay	Kumar	Engineering	30	Mumbai	25000.00

6. Write SQL query for members where location is Delhi or last name is Sharma.
   SELECT * FROM VCET_employee WHERE city = 'Delhi' OR last_name = 'Sharma';

emp_id	first_name	last_name	dept	age	city	salary
2	Babloo	Singh	Engineering	28	Delhi	23000.00
3	Chhavi	Sharma	HR	32	Bangalore	15000.00

7. Write SQL query to find the members that were outside the Pune
   SELECT * FROM VCET_employee WHERE city != 'Pune';

emp_id	first_name	last_name	dept	age	city	salary
1	Ajay	Kumar	Engineering	30	Mumbai	25000.00
2	Babloo	Singh	Engineering	28	Delhi	23000.00
3	Chhavi	Sharma	HR	32	Bangalore	15000.00
4	Dheeraj	Verma	HR	29	Hyderabad	12000.00
5	Evina	Thomas	Marketing	26	Chennai	16000.00

8. Query to groups the employee table based on age and counts the number of records in each group
   SELECT age, COUNT(*) AS record_count FROM VCET_employee GROUP BY age;

age	record_count
26	1
27	1
28	1
29	1
30	1
32	1

9. Find departments in which the sum of salary is greater than or equal to 30000
   SELECT dept, SUM(salary) AS total_salary FROM VCET_employee GROUP BY dept HAVING SUM(salary) >= 30000;

dept	total_salary
Engineering	48000.00