

Assignment No:-54

Name:-Suryawanshi Sangramsingh Sambhaji

Batch: - Delta - DCA (Java) 2024 Date:-29/7/2024

MYSQL PLACEMENT CENTRIC QUESTIONS:

```
CREATE TABLE employees (  
  employee_id INT PRIMARY KEY,  
  first_name VARCHAR(50),  
  last_name VARCHAR(50),  
  department VARCHAR(50),  
  salary DECIMAL(10,2)  
);
```

```
INSERT INTO employees (employee_id, first_name, last_name, department, salary)  
VALUES
```

```
(1, 'John', 'Doe', 'IT', 60000.00),  
(2, 'Jane', 'Smith', 'HR', 55000.00),  
(3, 'Bob', 'Johnson', 'Finance', 70000.00),  
(4, 'Eva', 'Brown', 'Marketing', 62000.00),  
(5, 'Michael', 'Clark', 'IT', 75000.00),  
(6, 'Emily', 'Jones', 'Finance', 68000.00),  
(7, 'Alex', 'Miller', 'HR', 58000.00),  
(8, 'Sophia', 'Wilson', 'Marketing', 67000.00),  
(9, 'Daniel', 'Lee', 'IT', 72000.00),  
(10, 'Olivia', 'Davis', 'Finance', 71000.00),  
(11, 'William', 'Garcia', 'Marketing', 63000.00),
```

(12, 'Ava', 'Martinez', 'HR', 59000.00),
 (13, 'James', 'Taylor', 'IT', 68000.00),
 (14, 'Mia', 'Anderson', 'Finance', 70000.00),
 (15, 'Benjamin', 'Hill', 'Marketing', 64000.00),
 (16, 'Emma', 'White', 'HR', 60000.00),
 (17, 'Liam', 'Harris', 'IT', 71000.00),
 (18, 'Chloe', 'Moore', 'Finance', 69000.00),
 (19, 'Noah', 'Clarkson', 'Marketing', 66000.00),
 (20, 'Grace', 'Cooper', 'IT', 73000.00);

Question on Subquery:

1. Retrieve employees who have a salary greater than the average salary in the IT department.

select * from employees where salary > (select avg(salary) from employees where department = 'IT');

```
mysql> select avg(salary) as Average_Salary from employees where department = 'IT';
+-----+
| Average_Salary |
+-----+
| 69833.333333 |
+-----+
1 row in set (0.00 sec)
```

```
mysql> select * from employees where salary > (select avg(salary) from employees where department = 'IT');
+-----+-----+-----+-----+-----+
| employee_id | first_name | last_name | department | salary |
+-----+-----+-----+-----+-----+
| 3 | Bob | Johnson | Finance | 70000.00 |
| 5 | Michael | Clark | IT | 75000.00 |
| 9 | Daniel | Lee | IT | 72000.00 |
| 10 | Olivia | Davis | Finance | 71000.00 |
| 14 | Mia | Anderson | Finance | 70000.00 |
| 17 | Liam | Harris | IT | 71000.00 |
| 20 | Grace | Cooper | IT | 73000.00 |
+-----+-----+-----+-----+-----+
7 rows in set (0.00 sec)
```

2. Find employees whose last name is the same as the manager's last name in the Finance department.

select * from employees where department='Finance' and first_name like '%e' OR (select department from employees where department = 'HR' AND last_name like '%e');

```
mysql> select * from employees where department = 'HR' AND last_name like '%e';
+-----+-----+-----+-----+-----+
| employee_id | first_name | last_name | department | salary |
+-----+-----+-----+-----+-----+
| 16 | Emma | White | HR | 60000.00 |
+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql> select * from employees where department='Finance' and first_name like '%e' OR (select department from employees where department = 'HR' AND last_name like '%e');
+-----+-----+-----+-----+-----+
| employee_id | first_name | last_name | department | salary |
+-----+-----+-----+-----+-----+
| 18 | Chloe | Moore | Finance | 69000.00 |
+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

3. Identify employees who work in departments with more than five employees.

select count(department)As Department_Count from employees group by department having count(employee_id)>5;

```
mysql> select count(department)As Department_Count from employees group by department having count(employee_id)>5;
+-----+
| Department_Count |
+-----+
| 6 |
+-----+
1 row in set (0.00 sec)
```

4. List employees who have a salary greater than the highest salary in the Marketing department.

select * from employees where salary>(select max(salary) from employees where department ='marketing');

```
mysql> select max(salary) from employees where department = 'marketing';
+-----+
| max(salary) |
+-----+
| 67000.00 |
+-----+
1 row in set (0.00 sec)

mysql> select * from employees where salary>(select max(salary) from employees where department ='marketing');
+-----+-----+-----+-----+-----+
| employee_id | first_name | last_name | department | salary |
+-----+-----+-----+-----+-----+
| 3 | Bob | Johnson | Finance | 70000.00 |
| 5 | Michael | Clark | IT | 75000.00 |
| 6 | Emily | Jones | Finance | 68000.00 |
| 9 | Daniel | Lee | IT | 72000.00 |
| 10 | Olivia | Davis | Finance | 71000.00 |
| 13 | James | Taylor | IT | 68000.00 |
| 14 | Mia | Anderson | Finance | 70000.00 |
| 17 | Liam | Harris | IT | 71000.00 |
| 18 | Chloe | Moore | Finance | 69000.00 |
| 20 | Grace | Cooper | IT | 73000.00 |
+-----+-----+-----+-----+-----+
10 rows in set (0.00 sec)
```

5. Select employees whose salary is within 10% of the highest salary in the company.

select * from employees where salary>=(select max(salary)*0.9 from employees);

```
mysql> select max(salary)*0.9 from employees;
+-----+
| max(salary)*0.9 |
+-----+
|      67500.000 |
+-----+
1 row in set (0.00 sec)
```

```
mysql> select * from employees where salary>=(select max(salary)*0.9 from employees);
+-----+-----+-----+-----+-----+
| employee_id | first_name | last_name | department | salary |
+-----+-----+-----+-----+-----+
|          3 | Bob       | Johnson   | Finance    | 70000.00 |
|          5 | Michael   | Clark     | IT          | 75000.00 |
|          6 | Emily     | Jones     | Finance    | 68000.00 |
|          9 | Daniel    | Lee       | IT          | 72000.00 |
|         10 | Olivia    | Davis     | Finance    | 71000.00 |
|         13 | James     | Taylor    | IT          | 68000.00 |
|         14 | Mia       | Anderson  | Finance    | 70000.00 |
|         17 | Liam      | Harris    | IT          | 71000.00 |
|         18 | Chloe     | Moore     | Finance    | 69000.00 |
|         20 | Grace     | Cooper    | IT          | 73000.00 |
+-----+-----+-----+-----+-----+
10 rows in set (0.00 sec)
```

6. Find employees who do not have a manager.

select * from employees where department != 'HR';

```
mysql> select * from employees where department != 'HR';
```

employee_id	first_name	last_name	department	salary
1	John	Doe	IT	60000.00
3	Bob	Johnson	Finance	70000.00
4	Eva	Brown	Marketing	62000.00
5	Michael	Clark	IT	75000.00
6	Emily	Jones	Finance	68000.00
8	Sophia	Wilson	Marketing	67000.00
9	Daniel	Lee	IT	72000.00
10	Olivia	Davis	Finance	71000.00
11	William	Garcia	Marketing	63000.00
13	James	Taylor	IT	68000.00
14	Mia	Anderson	Finance	70000.00
15	Benjamin	Hill	Marketing	64000.00
17	Liam	Harris	IT	71000.00
18	Chloe	Moore	Finance	69000.00
19	Noah	Clarkson	Marketing	66000.00
20	Grace	Cooper	IT	73000.00

```
16 rows in set (0.00 sec)
```

7. List departments where the average salary is greater than the average salary in the IT department.

select avg(salary) as average_salary from employees where salary > (select avg(salary) from employees where department = 'IT');

```
mysql> select avg(salary) from employees where department = 'IT';
```

avg(salary)
69833.333333

```
1 row in set (0.00 sec)
```

```
mysql> select avg(salary) as average_salary from employees where salary > (select avg(salary) from employees where department = 'IT');
```

average_salary
71714.285714

```
1 row in set (0.00 sec)
```

8. Retrieve employees who have the same first name as their manager.

select * from employees (select first_name from employees where first_name like '%a');

```
mysql> select * from employees where first_name like 'J%' and department != 'HR' or (select first_name from employees where department = 'HR' and first_name like 'J%');
```

employee_id	first_name	last_name	department	salary
1	John	Doe	IT	60000.00
13	James	Taylor	IT	68000.00

```
2 rows in set (0.00 sec)
```

9. Identify employees who have a salary higher than the average salary of their department.

select * from employees where salary > (select avg(salary) from employees) order by salary desc;

```
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version
mysql> select * from employees where salary > (select avg(salary) from employees) order by salary desc;
```

employee_id	first_name	last_name	department	salary
5	Michael	Clark	IT	75000.00
20	Grace	Cooper	IT	73000.00
9	Daniel	Lee	IT	72000.00
10	Olivia	Davis	Finance	71000.00
17	Liam	Harris	IT	71000.00
3	Bob	Johnson	Finance	70000.00
14	Mia	Anderson	Finance	70000.00
18	Chloe	Moore	Finance	69000.00
6	Emily	Jones	Finance	68000.00
13	James	Taylor	IT	68000.00
8	Sophia	Wilson	Marketing	67000.00

11 rows in set (0.00 sec)

10. Find employees who have a higher salary than anyone in the HR department.

Select * from employees where department != 'HR' and salary > (select avg(salary) from employees where department = 'HR');

```
mysql> select avg(salary) from employees where department = 'HR';
```

avg(salary)
58000.000000

1 row in set (0.00 sec)

```
mysql> Select * from employees where department != 'HR' and salary > (select avg(salary) from employees where department = 'HR');
```

employee_id	first_name	last_name	department	salary
1	John	Doe	IT	60000.00
3	Bob	Johnson	Finance	70000.00
4	Eva	Brown	Marketing	62000.00
5	Michael	Clark	IT	75000.00
6	Emily	Jones	Finance	68000.00
8	Sophia	Wilson	Marketing	67000.00
9	Daniel	Lee	IT	72000.00
10	Olivia	Davis	Finance	71000.00
11	William	Garcia	Marketing	63000.00
13	James	Taylor	IT	68000.00
14	Mia	Anderson	Finance	70000.00
15	Benjamin	Hill	Marketing	64000.00
17	Liam	Harris	IT	71000.00
18	Chloe	Moore	Finance	69000.00
19	Noah	Clarkson	Marketing	66000.00
20	Grace	Cooper	IT	73000.00

16 rows in set (0.00 sec)

11. List departments where the total salary is greater than the total salary in the Finance department.

Select * from employees where department != 'Finance' and salary > (select avg(salary) from employees where department = 'Finance');

```
mysql> Select * from employees where department != 'Finance' and salary > (select avg(salary) from employees where department = 'Finance');
```

employee_id	first_name	last_name	department	salary
5	Michael	Clark	IT	75000.00
9	Daniel	Lee	IT	72000.00
17	Liam	Harris	IT	71000.00
20	Grace	Cooper	IT	73000.00

4 rows in set (0.00 sec)

12. Retrieve employees who do not belong to any department.

Select * from employees where department is null;

```
mysql> select * from employees where department is null;
Empty set (0.00 sec)

mysql>
```

13. Find employees who joined after the employee with the highest salary.

select * from employees order by salary desc limit 1 offset 1;

```
mysql> select * from employees order by salary desc limit 1 offset 1;
+-----+-----+-----+-----+-----+
| employee_id | first_name | last_name | department | salary |
+-----+-----+-----+-----+-----+
|          20 | Grace     | Cooper   | IT         | 73000.00 |
+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql> select max(salary) from employees;
+-----+
| max(salary) |
+-----+
|    75000.00 |
+-----+
1 row in set (0.00 sec)
```

14. List employees whose salary is greater than the average salary of employees in the same department.

select * from employees where department = 'IT' and salary > (select avg(salary) from employees where department = 'IT');

```
mysql> select * from employees where department = 'IT' and salary > (select avg(salary) from employees where department = 'IT');
+-----+-----+-----+-----+-----+
| employee_id | first_name | last_name | department | salary |
+-----+-----+-----+-----+-----+
|          5 | Michael   | Clark    | IT         | 75000.00 |
|          9 | Daniel    | Lee      | IT         | 72000.00 |
|         17 | Liam     | Harris   | IT         | 71000.00 |
|         20 | Grace    | Cooper   | IT         | 73000.00 |
+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)

mysql> select avg(salary) from employees where department = 'IT';
+-----+
| avg(salary) |
+-----+
| 69833.333333 |
+-----+
1 row in set (0.00 sec)
```

15. Select departments with more than three employees earning a salary greater than \$60,000.

select department FROM employees WHERE salary > 60000 group by department having count(employee_id)>3;

```
ERROR 1054 (42S22): Unknown column 'department' in 'group statement'
mysql> select department FROM employees WHERE salary > 60000 group by department having count(employee_id)>3;
+-----+
| department |
+-----+
| Finance    |
| Marketing  |
| IT         |
+-----+
3 rows in set (0.00 sec)
```

16. Retrieve employees who have a salary higher than the average salary of employees with the same job title.

select * from employees d WHERE salary >(select avg(salary) from employees where department=d.department);

```
mysql> select * from employees d WHERE salary >(select avg(salary) from employees where department=d.department);
+-----+-----+-----+-----+-----+
| employee_id | first_name | last_name | department | salary |
+-----+-----+-----+-----+-----+
| 3 | Bob | Johnson | Finance | 70000.00 |
| 5 | Michael | Clark | IT | 75000.00 |
| 8 | Sophia | Wilson | Marketing | 67000.00 |
| 9 | Daniel | Lee | IT | 72000.00 |
| 10 | Olivia | Davis | Finance | 71000.00 |
| 12 | Ava | Martinez | HR | 59000.00 |
| 14 | Mia | Anderson | Finance | 70000.00 |
| 16 | Emma | White | HR | 60000.00 |
| 17 | Liam | Harris | IT | 71000.00 |
| 19 | Noah | Clarkson | Marketing | 66000.00 |
| 20 | Grace | Cooper | IT | 73000.00 |
+-----+-----+-----+-----+-----+
11 rows in set (0.00 sec)

mysql> select avg(salary) from employees d where department=d.department;
+-----+
| avg(salary) |
+-----+
| 66050.000000 |
+-----+
1 row in set (0.00 sec)
```

17. Find employees who have a salary higher than the average salary of their department and joined after 2020.

18. List employees who have the same salary as at least one employee in the IT department.

`select * from employees where department != 'IT' and salary in (select salary from employees where department = 'IT');`

```
mysql> select * from employees where department != 'IT' and salary in (select salary from employees where department = 'IT');
+-----+-----+-----+-----+-----+
| employee_id | first_name | last_name | department | salary |
+-----+-----+-----+-----+-----+
|          6 | Emily     | Jones    | Finance    | 68000.00 |
|         10 | Olivia    | Davis    | Finance    | 71000.00 |
|         16 | Emma     | White    | HR         | 60000.00 |
+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)
```

19. Identify employees who have a salary higher than the average salary of employees with the same manager.

`select * from employees where department != 'HR' and salary > (select avg(salary) from employees where department = 'HR');`

```
department= HR ) at line 1
mysql> select * from employees where department != 'HR' and salary > (select avg(salary) from employees where department = 'HR');
+-----+-----+-----+-----+-----+
| employee_id | first_name | last_name | department | salary |
+-----+-----+-----+-----+-----+
|          1 | John      | Doe       | IT         | 60000.00 |
|          3 | Bob       | Johnson   | Finance    | 70000.00 |
|          4 | Eva       | Brown     | Marketing  | 62000.00 |
|          5 | Michael   | Clark     | IT         | 75000.00 |
|          6 | Emily     | Jones     | Finance    | 68000.00 |
|          8 | Sophia    | Wilson    | Marketing  | 67000.00 |
|          9 | Daniel    | Lee       | IT         | 72000.00 |
|         10 | Olivia    | Davis     | Finance    | 71000.00 |
|         11 | William   | Garcia    | Marketing  | 63000.00 |
|         13 | James     | Taylor    | IT         | 68000.00 |
|         14 | Mia       | Anderson  | Finance    | 70000.00 |
|         15 | Benjamin  | Hill      | Marketing  | 64000.00 |
|         17 | Liam     | Harris    | IT         | 71000.00 |
|         18 | Chloe     | Moore     | Finance    | 69000.00 |
|         19 | Noah     | Clarkson  | Marketing  | 66000.00 |
|         20 | Grace     | Cooper    | IT         | 73000.00 |
+-----+-----+-----+-----+-----+
16 rows in set (0.00 sec)
```

20. Retrieve employees who have a salary higher than the average salary in their department and belong to the IT department.

`select * from employees d where department = 'IT' and salary > (select avg(salary) from employees where department = d.department);`

```
mysql> select * from employees d where department = 'IT' and salary > (select avg(salary) from employees where department = d.department);
+-----+-----+-----+-----+-----+
| employee_id | first_name | last_name | department | salary |
+-----+-----+-----+-----+-----+
|          5 | Michael   | Clark     | IT         | 75000.00 |
|          9 | Daniel    | Lee       | IT         | 72000.00 |
|         17 | Liam     | Harris    | IT         | 71000.00 |
|         20 | Grace     | Cooper    | IT         | 73000.00 |
+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

Question on Limit:

1. Retrieve the first 5 employees in the table.

`select * from employees limit 5;`

```
mysql> select * from employees limit 5;
+-----+-----+-----+-----+-----+
| employee_id | first_name | last_name | department | salary |
+-----+-----+-----+-----+-----+
| 1 | John | Doe | IT | 60000.00 |
| 2 | Jane | Smith | HR | 55000.00 |
| 3 | Bob | Johnson | Finance | 70000.00 |
| 4 | Eva | Brown | Marketing | 62000.00 |
| 5 | Michael | Clark | IT | 75000.00 |
+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)
```

2. List the top 10 highest-paid employees.

`select * from employees order by salary desc limit 10;`

```
mysql> select * from employees order by salary desc limit 10;
+-----+-----+-----+-----+-----+
| employee_id | first_name | last_name | department | salary |
+-----+-----+-----+-----+-----+
| 5 | Michael | Clark | IT | 75000.00 |
| 20 | Grace | Cooper | IT | 73000.00 |
| 9 | Daniel | Lee | IT | 72000.00 |
| 10 | Olivia | Davis | Finance | 71000.00 |
| 17 | Liam | Harris | IT | 71000.00 |
| 3 | Bob | Johnson | Finance | 70000.00 |
| 14 | Mia | Anderson | Finance | 70000.00 |
| 18 | Chloe | Moore | Finance | 69000.00 |
| 6 | Emily | Jones | Finance | 68000.00 |
| 13 | James | Taylor | IT | 68000.00 |
+-----+-----+-----+-----+-----+
10 rows in set (0.00 sec)
```

3. Select the first 3 departments in alphabetical order.

`select * from employees order by department desc limit 3;`

```
mysql> select * from employees order by department desc limit 3;
+-----+-----+-----+-----+-----+
| employee_id | first_name | last_name | department | salary |
+-----+-----+-----+-----+-----+
| 11 | William | Garcia | Marketing | 63000.00 |
| 8 | Sophia | Wilson | Marketing | 67000.00 |
| 4 | Eva | Brown | Marketing | 62000.00 |
+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)
```

4. Retrieve the oldest 8 employees based on their hire date.

5. List the 5 departments with the most employees.

select department from employees group by department having count(department)>5;

```
mysql> select department from employees group by department having count(department)>5;
+-----+
| department |
+-----+
| IT         |
+-----+
1 row in set (0.00 sec)

mysql>
```

6. Select the first 5 employees in the HR department.

select * from employees where department='HR' limit 5;

```
mysql> select * from employees where department='HR' limit 5
-> ;
+-----+-----+-----+-----+-----+
| employee_id | first_name | last_name | department | salary |
+-----+-----+-----+-----+-----+
|          2 | Jane      | Smith    | HR         | 55000.00 |
|          7 | Alex      | Miller   | HR         | 58000.00 |
|         12 | Ava       | Martinez | HR         | 59000.00 |
|         16 | Emma      | White    | HR         | 60000.00 |
+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

7. Retrieve the 3 employees with the lowest salaries.

select * from employees order by salary limit 3;

```
mysql> select * from employees order by salary limit 3;
+-----+-----+-----+-----+-----+
| employee_id | first_name | last_name | department | salary |
+-----+-----+-----+-----+-----+
|          2 | Jane      | Smith    | HR         | 55000.00 |
|          7 | Alex      | Miller   | HR         | 58000.00 |
|         12 | Ava       | Martinez | HR         | 59000.00 |
+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)
```

8. List the top 7 departments with the highest average salary.

select department,avg(salary) from employees group by department limit 7;

```
mysql> select department,avg(salary) from employees group by department limit 7;
+-----+-----+
| department | avg(salary) |
+-----+-----+
| IT         | 69833.333333 |
| HR         | 58000.000000 |
| Finance    | 69600.000000 |
| Marketing  | 64400.000000 |
+-----+-----+
4 rows in set (0.00 sec)
```

9. Select the first 4 employees who joined the company in the year 2021.

10. Retrieve the 6 employees with the highest salaries in the Finance department.

select * from employees where department='Finance' order by salary desc limit 6;

```
mysql> select * from employees where department='Finance' order by salary desc limit 6;
+-----+-----+-----+-----+-----+
| employee_id | first_name | last_name | department | salary |
+-----+-----+-----+-----+-----+
| 10          | Olivia    | Davis    | Finance    | 71000.00 |
| 3           | Bob       | Johnson  | Finance    | 70000.00 |
| 14          | Mia       | Anderson | Finance    | 70000.00 |
| 18          | Chloe     | Moore    | Finance    | 69000.00 |
| 6           | Emily     | Jones    | Finance    | 68000.00 |
+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)
```

Questions on Conditions:

1. Select employees who work in the IT department.

```
select * from employees where department=IT;
```

```
mysql> select * from employees where department='IT';
```

employee_id	first_name	last_name	department	salary
1	John	Doe	IT	60000.00
5	Michael	Clark	IT	75000.00
9	Daniel	Lee	IT	72000.00
13	James	Taylor	IT	68000.00
17	Liam	Harris	IT	71000.00
20	Grace	Cooper	IT	73000.00

6 rows in set (0.00 sec)

2. Retrieve employees with a salary greater than \$60,000.

```
select * from employees where salary >60000;
```

```
mysql> select * from employees where salary >60000;
```

employee_id	first_name	last_name	department	salary
3	Bob	Johnson	Finance	70000.00
4	Eva	Brown	Marketing	62000.00
5	Michael	Clark	IT	75000.00
6	Emily	Jones	Finance	68000.00
8	Sophia	Wilson	Marketing	67000.00
9	Daniel	Lee	IT	72000.00
10	Olivia	Davis	Finance	71000.00
11	William	Garcia	Marketing	63000.00
13	James	Taylor	IT	68000.00
14	Mia	Anderson	Finance	70000.00
15	Benjamin	Hill	Marketing	64000.00
17	Liam	Harris	IT	71000.00
18	Chloe	Moore	Finance	69000.00
19	Noah	Clarkson	Marketing	66000.00
20	Grace	Cooper	IT	73000.00

15 rows in set (0.00 sec)

3. List employees hired after January 1, 2022.

4. Find employees whose last name starts with 'S'.

select * from employees where last_name like 's%';

```
mysql> select * from employees where last_name like 's%';
+-----+-----+-----+-----+-----+
| employee_id | first_name | last_name | department | salary |
+-----+-----+-----+-----+-----+
|          2 | Jane      | Smith    | HR         | 55000.00 |
+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

5. Select employees who do not belong to any department.

select * from employees e where department!=e.department;

```
mysql> select * from employees e where department!=e.department;
Empty set (0.00 sec)
```

6. Retrieve employees whose first name is either 'John' or 'Jane'.

select * from employees where first_name in('John','Jane');

```
mysql> select * from employees where first_name in('John','Jane');
+-----+-----+-----+-----+-----+
| employee_id | first_name | last_name | department | salary |
+-----+-----+-----+-----+-----+
|          1 | John      | Doe       | IT         | 60000.00 |
|          2 | Jane      | Smith    | HR         | 55000.00 |
+-----+-----+-----+-----+-----+
2 rows in set (0.00 sec)
```

7. List employees who have 'Manager' in their job title.

select * from employees e where department='HR';

```
mysql> select * from employees e where department='HR';
+-----+-----+-----+-----+-----+
| employee_id | first_name | last_name | department | salary |
+-----+-----+-----+-----+-----+
|          2 | Jane      | Smith    | HR         | 55000.00 |
|          7 | Alex      | Miller   | HR         | 58000.00 |
|         12 | Ava      | Martinez | HR         | 59000.00 |
|         16 | Emma     | White    | HR         | 60000.00 |
+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

8. Find employees who joined before the year 2021 and have a salary greater than \$50,000.

9. Select employees who work in the Finance department and have a salary greater than \$70,000.

select * from employees where department='Finance' and salary >70000;

```
mysql> select * from employees where department='Finance' and salary >70000;
+-----+-----+-----+-----+-----+
| employee_id | first_name | last_name | department | salary |
+-----+-----+-----+-----+-----+
|          10 | Olivia    | Davis    | Finance    | 71000.00 |
+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

10. Retrieve employees who do not have a manager.

select * from employees e where department !='HR';

```
mysql> select * from employees e where department !='HR';
+-----+-----+-----+-----+-----+
| employee_id | first_name | last_name | department | salary |
+-----+-----+-----+-----+-----+
|          1 | John       | Doe       | IT          | 60000.00 |
|          3 | Bob        | Johnson   | Finance     | 70000.00 |
|          4 | Eva        | Brown     | Marketing   | 62000.00 |
|          5 | Michael    | Clark     | IT          | 75000.00 |
|          6 | Emily      | Jones     | Finance     | 68000.00 |
|          8 | Sophia     | Wilson    | Marketing   | 67000.00 |
|          9 | Daniel     | Lee       | IT          | 72000.00 |
|         10 | Olivia     | Davis     | Finance     | 71000.00 |
|         11 | William    | Garcia    | Marketing   | 63000.00 |
|         13 | James      | Taylor    | IT          | 68000.00 |
|         14 | Mia        | Anderson  | Finance     | 70000.00 |
|         15 | Benjamin   | Hill      | Marketing   | 64000.00 |
|         17 | Liam       | Harris    | IT          | 71000.00 |
|         18 | Chloe      | Moore     | Finance     | 69000.00 |
|         19 | Noah       | Clarkson  | Marketing   | 66000.00 |
|         20 | Grace      | Cooper    | IT          | 73000.00 |
+-----+-----+-----+-----+-----+
16 rows in set (0.00 sec)
```

Questions on Alter statement:

1. Add a new column named "birth_date" to the employees table with the data type DATE.

Alter table employees add birth_date date;

```
ERROR 1146 (42S02): Table 'assignmentno_1.employees' doesn't exist
mysql> Alter table employees add birth_date date;
Query OK, 0 rows affected (0.06 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> desc birth_date;
ERROR 1146 (42S02): Table 'assignmentno_1.birth_date' doesn't exist
mysql> desc birth_date date;
ERROR 1146 (42S02): Table 'assignmentno_1.birth_date' doesn't exist
mysql> DESCRIBE employees;
+-----+-----+-----+-----+-----+-----+
| Field      | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| employee_id | int           | NO   | PRI | NULL    |       |
| first_name  | varchar(50)   | YES  |     | NULL    |       |
| last_name   | varchar(50)   | YES  |     | NULL    |       |
| department  | varchar(50)   | YES  |     | NULL    |       |
| salary      | decimal(10,2) | YES  |     | NULL    |       |
| birth_date  | date          | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
6 rows in set (0.01 sec)
```

2. Modify the data type of the "salary" column to accommodate larger values.

Alter table employees modify salary bigint;

```
mysql> Alter table employees modify salary bigint;
Query OK, 20 rows affected (0.06 sec)
Records: 20 Duplicates: 0 Warnings: 0

mysql> DESCRIBE employees;
+-----+-----+-----+-----+-----+-----+
| Field      | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| employee_id | int           | NO   | PRI | NULL    |       |
| first_name  | varchar(50)   | YES  |     | NULL    |       |
| last_name   | varchar(50)   | YES  |     | NULL    |       |
| department  | varchar(50)   | YES  |     | NULL    |       |
| salary      | bigint        | YES  |     | NULL    |       |
| birth_date  | date          | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
6 rows in set (0.00 sec)
```


3. Rename the "department" column to "dept" in the employees table.

Alter table employees rename column department to dept;

```
mysql> Alter table employees rename column department to dept;
Query OK, 0 rows affected (0.02 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> DESCRIBE employees;
+-----+-----+-----+-----+-----+-----+
| Field          | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| employee_id    | int           | NO   | PRI | NULL    |       |
| first_name     | varchar(50)   | YES  |     | NULL    |       |
| last_name      | varchar(50)   | YES  |     | NULL    |       |
| dept           | varchar(50)   | YES  |     | NULL    |       |
| salary         | bigint        | YES  |     | NULL    |       |
| birth_date     | date          | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
6 rows in set (0.00 sec)
```

4. Add a NOT NULL constraint to the "email" column.

Alter table employees add email varchar(30) NOT NULL;

```
mysql> Alter table employees add email varchar(30) NOT NULL;
Query OK, 0 rows affected (0.02 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> DESCRIBE employees;
+-----+-----+-----+-----+-----+-----+
| Field          | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| employee_id    | int           | NO   | PRI | NULL    |       |
| first_name     | varchar(50)   | YES  |     | NULL    |       |
| last_name      | varchar(50)   | YES  |     | NULL    |       |
| dept           | varchar(50)   | YES  |     | NULL    |       |
| salary         | bigint        | YES  |     | NULL    |       |
| birth_date     | date          | YES  |     | NULL    |       |
| email          | varchar(30)   | NO   |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
7 rows in set (0.00 sec)
```

5. Drop the "birth_date" column from the employees table.

Alter table employees drop column birth_date;

```
mysql> Alter table employees drop column birth_date;
Query OK, 0 rows affected (0.02 sec)
Records: 0 Duplicates: 0 Warnings: 0
```

```
mysql> DESCRIBE employees;
```

Field	Type	Null	Key	Default	Extra
employee_id	int	NO	PRI	NULL	
first_name	varchar(50)	YES		NULL	
last_name	varchar(50)	YES		NULL	
dept	varchar(50)	YES		NULL	
salary	bigint	YES		NULL	
email	varchar(30)	NO		NULL	

6 rows in set (0.00 sec)