

BOARD INFINITY

DATABASE MANAGEMENT SYSTEM & SQL

PROJECT

**THESE SQL QUERIES ARE WRITTEN FOR HR DATABASE WHICH IS
ORIGINALLY CREATED BY MICROSOFT**

DATABASE LINK: <https://www.kaggle.com/datasets/sirajahmad/hr-schema-mysql>

*** TASKS***

1. Write a query to find the addresses (location_id, street_address, city, state_province, country_name) of all the departments.

```
SELECT
    l.location_id,
    l.street_address,
    l.city,
    l.state_province,
    c.country_name
FROM
    departments d
JOIN
    locations l ON d.location_id = l.location_id
JOIN
    countries c ON l.country_id = c.country_id;
```

2. Write a query to find the name (first_name, last_name), department ID and name of all the employees.

```
SELECT
    e.first_name,
    e.last_name,
    d.department_id,
    d.department_name
FROM
    employees e
JOIN
    departments d ON e.department_id = d.department_id;
```

3. Write a query to find the name (first_name, last_name), job, department ID and name of the employees who works in London

```
SELECT
    e.first_name,
    e.last_name,
    j.job_title,
    d.department_id,
    d.department_name
FROM
    employees e
JOIN
    departments d ON e.department_id = d.department_id
```

JOIN

locations l ON d.location_id = l.location_id

JOIN

jobs j ON e.job_id = j.job_id

WHERE

l.city = 'London';

4. Write a query to find the employee id, name (last_name) along with their manager_id and name (last_name)

SELECT

e.employee_id,

e.last_name AS employee_last_name,

m.employee_id AS manager_id,

m.last_name AS manager_last_name

FROM

employees e

LEFT JOIN

employees m ON e.manager_id = m.employee_id;

5. Write a query to find the name (first_name, last_name) and hire date of the employees who was hired after 'Jones'

```
SELECT
    e.first_name,
    e.last_name,
    e.hire_date
FROM
    employees e
WHERE
    e.hire_date > (SELECT hire_date FROM employees WHERE last_name = 'Jones');
```

6. Write a query to get the department name and number of employees in the department

```
SELECT
    d.department_name,
    COUNT(e.employee_id) AS num_employees
FROM
    departments d
LEFT JOIN
    employees e ON d.department_id = e.department_id
GROUP BY
    d.department_name;
```

7. Write a query to display department name, name (first_name, last_name), hire date, salary of the manager for all managers whose experience is more than 15 years

SELECT

d.department_name,

e.first_name,

e.last_name,

e.hire_date,

e.salary

FROM

employees e

JOIN

departments d ON e.department_id = d.department_id

WHERE

e.employee_id IN (SELECT DISTINCT manager_id FROM employees)

AND DATEDIFF(CURDATE(), e.hire_date) > 15 * 365;

8. Write a query to find the name (first_name, last_name) and the salary of the employees who have a higher salary than the employee whose last_name='Bull'

SELECT

e.first_name,

e.last_name,

```
e.salary  
FROM  
employees e  
WHERE  
e.salary > (SELECT salary FROM employees WHERE last_name = 'Bull');
```

9. Write a query to find the name (first_name, last_name) of all employees who works in the IT department

```
SELECT  
e.first_name,  
e.last_name  
FROM  
employees e  
JOIN  
departments d ON e.department_id = d.department_id  
WHERE  
d.department_name = 'IT';
```

10. Write a query to find the name (first_name, last_name) of the employees who have a manager and worked in a USA based department

```
SELECT  
e.first_name,
```

```
e.last_name
FROM
employees e
JOIN
departments d ON e.department_id = d.department_id
JOIN
locations l ON d.location_id = l.location_id
JOIN
countries c ON l.country_id = c.country_id
WHERE
e.manager_id IS NOT NULL
AND c.country_name = 'USA';
```

11. Write a query to find the name (first_name, last_name), and salary of the employees whose salary is greater than the average salary

```
SELECT
e.first_name,
e.last_name,
e.salary
FROM
employees e
WHERE
e.salary > (SELECT AVG(salary) FROM employees);
```

12. Write a query to find the name (first_name, last_name), and salary of the employees whose salary is equal to the minimum salary for their job grade

```
SELECT
    e.first_name,
    e.last_name,
    e.salary
FROM
    employees e
JOIN
    jobs j ON e.job_id = j.job_id
WHERE
    e.salary = j.min_salary;
```

13. Write a query to find the name (first_name, last_name), and salary of the employees who earns more than the average salary and works in any of the IT departments

```
SELECT
    e.first_name,
    e.last_name,
    e.salary
FROM
    employees e
```


JOIN

departments d ON e.department_id = d.department_id

WHERE

e.salary > (SELECT AVG(salary) FROM employees)

AND d.department_name = 'IT';

14. Write a query to find the name (first_name, last_name), and salary of the employees who earn the same salary as the minimum salary for all departments.

SELECT

e.first_name,

e.last_name,

e.salary

FROM

employees e

JOIN

(SELECT department_id, MIN(salary) AS min_salary FROM employees GROUP BY department_id) m

ON e.department_id = m.department_id AND e.salary = m.min_salary;

15. Write a query to find the name (first_name, last_name) and salary of the employees who earn a salary that is higher than the salary of all the Shipping Clerk (JOB_ID = 'SH_CLERK'). Sort the results of the salary of the lowest to highest

```
SELECT
    e.first_name,
    e.last_name,
    e.salary
FROM
    employees e
WHERE
    e.salary > (SELECT MAX(salary) FROM employees WHERE job_id = 'SH_CLERK')
ORDER BY
    e.salary ASC;
```

THANK YOU