NESTED QUERIES

COMPARISON OPERATOR IN

- Compares value *v* with a set (or multiset) of values *V*
- Evaluates to TRUE if *v* is one of the elements in *V*

1. **Write a query to retrieve the first name, last name, salary and department id of emplotees whose salary is greater than all the employees working in department 5 ordered by department id.**

SELECT FIRST_NAME, LAST_NAME, SALARY, DEPARTMENT_ID FROM EMPLOYEES

WHERE SALARY > ALL(SELECT SALARY FROM EMPLOYEES WHERE DEPARTMENT_ID=5)

ORDER BY DEPARTMENT_ID;

2. **Write a query to retrieve the first name, last name, department name and location id of each employee who are working in the departments located in location id 1700**

SELECT FIRST_NAME, LAST_NAME, DEPARTMENT_NAME, DEPARTMENTS.LOCATION_ID FROM EMPLOYEES, DEPARTMENTS

WHERE EMPLOYEES.DEPARTMENT_ID IN (SELECT DEPARTMENT_ID FROM DEPARTMENTS WHERE LOCATION_ID=1700)

AND EMPLOYEES.DEPARTMENT_ID=DEPARTMENTS.DEPARTMENT_ID ;

3. **Write a query to display the first name, last name, salary, department id, job id for those employees who works in the same designation as the employee works whose id is 169.**

SELECT FIRST_NAME, LAST_NAME, SALARY, DEPARTMENT_ID, JOB_ID

FROM EMPLOYEES

WHERE JOB_ID = (SELECT JOB_ID FROM EMPLOYEES WHERE EMPLOYEE=169);

4. **Write a query to display the employee id, employee name (first name and last name) for all employees who earn more than the average salary.**

```
SELECT employee_id, first_name,last_name

FROM employees

WHERE salary >  ( SELECT AVG(salary)  FROM employees );
```

5. **Write a query to display the employee first name, last name, employee id and salary of all employees who report to Payam.**

```
SELECT first_name, last_name, employee_id, salary

FROM employees

WHERE manager_id = (SELECT employee_id  FROM employees  WHERE first_name = 'Payam' );
```

6. **Write a query to display all the information of the employees who does not work in those departments where some employees works whose manager id within the range 100 and 200.**

```
SELECT *

FROM employees

WHERE department_id NOT IN (SELECT department_id FROM departments

WHERE manager_id BETWEEN 100 AND 200);
```

7. **Write a query that will identify all employees who work in departments located in the United Kingdom.**

```
SELECT first_name

FROM employees

WHERE department_id IN

(SELECT department_id

FROM departments

WHERE location_id IN

(SELECT location_id
FROM locations
WHERE country_id =
(SELECT country_id
FROM countries
WHERE country_name='United Kingdom')));
```

8. **Write a query which is looking for the names of all employees whose salary is greater than 50% of their department’s total salary bill.**

```
SELECT e1.first_name, e1.last_name
FROM employees e1
WHERE salary >
( SELECT (SUM(salary))*.5
FROM employees e2
WHERE e1.department_id=e2.department_id);
```

9. **Write a query to display the department name and Id for all departments where they located, that Id is equal to the Id for the location where department number 30 is located.**

```
SELECT department_name, department_id
FROM departments
WHERE location_id = (SELECT location_id FROM  departments WHERE
department_id = 30);
```

10. **Write a query to display the full name (first and last name) of manager who is supervising 4 or more employees.**

```
SELECT first_name || ' ' || last_name AS Manager_name,department_id FROM employees
WHERE employee_id IN (SELECT manager_id FROM employees GROUP BY
manager_id HAVING COUNT(*)>=4);
```

11. **Write a query in SQL to display the details of the current job for those employees who worked as a Sales Representative in the past.**

```
SELECT * FROM jobs
WHERE job_id IN (SELECT job_id FROM employees WHERE employee_id IN
 (SELECT employee_id FROM job_history WHERE job_id='SA_REP'));
```