SQL Exercises (HR Database)

SQL SUBQUERIES

https://www.sqltutorial.org/sql-sample-database/

https://www.w3resource.com/sql/tutorials.php#SQLDBASE

https://www.w3resource.com/sql-exercises/sql-subqueries-exercises.php

1. Write a query to display the name (first name and last name) for those employees who gets more salary than the employee whose ID is 103.

```
SELECT first_name, last_name
FROM employees
WHERE salary >
( SELECT salary
FROM employees
WHERE employee_id=103
);
```

2. Write a query to display the name (first name and 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_id=169
);
```

3. Write a query to display the name (first name and last name), salary, department id for those employees who earn such amount of salary which is the smallest salary of any of the departments.

```
SELECT first_name, last_name, salary, department_id
FROM employees
WHERE salary IN
( SELECT MIN(salary)
FROM employees
GROUP BY department_id
);
```

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 name (first name and 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 the department number, name (first name and last name), job and department name for all employees in the Finance department.

```
SELECT e.department_id, e.first_name, e.job_id , d.department_name
FROM employees e , departments d
WHERE e.department_id = d.department_id
AND  d.department_name = 'Finance';
```

7. Write a query to display all the information of an employee whose salary and reporting person id is 3000 and 121 respectively.

```
SELECT *
FROM employees
WHERE (salary,manager_id)=
(SELECT 3000,121);
```

8. Display all the information of an employee whose id is any of the number 134, 159 and 183.

```
SELECT *
FROM employees
WHERE employee_id IN (134,159,183);
```

9. Write a query to display all the information of the employees whose salary is within the range 1000 and 3000.

```
SELECT * FROM employees
WHERE salary BETWEEN 1000 and 3000;
```

10. Write a query to display all the information of the employees whose salary is within the range of smallest salary and 2500.

```
SELECT *
FROM employees
```

```
WHERE salary BETWEEN
(SELECT MIN(salary)
FROM employees) AND 2500;
```

11. 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);
```

12. Write a query to display all the information for those employees whose id is any id who earn the second highest salary.

```
SELECT *
FROM employees
WHERE employee_id IN
(SELECT employee_id
FROM employees
WHERE salary =
(SELECT MAX(salary)
FROM employees
WHERE salary <
(SELECT MAX(salary)
FROM employees)));
```

13. Write a query to display the employee name(first name and last name) and hiredate for all employees in the same department as Clara. Exclude Clara.

```
SELECT first_name, last_name, hire_date
FROM employees
WHERE department_id =
( SELECT department_id
FROM employees
WHERE first_name = 'Clara')
AND first_name <> 'Clara';
```

14. Write a query to display the employee number and name(first name and last name) for all employees who work in a department with any employee whose name contains a T.

```
SELECT employee_id, first_name, last_name

FROM employees
WHERE department_id IN
( SELECT department_id
FROM employees
WHERE first_name LIKE '%T%' );
```

15. Write a query to display the employee number, name(first name and last name), and salary for all employees who earn more than the average salary and who work in a department with any employee with a J in their name.

```
SELECT employee_id, first_name , salary
FROM employees
WHERE salary >
(SELECT AVG (salary)
FROM employees )
AND  department_id IN
( SELECT department_id
FROM employees
WHERE first_name LIKE '%J%');
```

16. Display the employee name(first name and last name), employee id, and job title for all employees whose department location is Toronto.

```
SELECT  first_name, last_name, employee_id, job_id
FROM employees
WHERE department_id =
(SELECT department_id
FROM departments
WHERE location_id =
(SELECT location_id
FROM locations
WHERE city ='Toronto'));
```

17. Write a query to display the employee number, name(first name and last name) and job title for all employees whose salary is smaller than any salary of those employees whose job title is MK_MAN.

```
SELECT employee_id,first_name,last_name,job_id

FROM employees
WHERE salary < ANY
 ( SELECT salary
 FROM employees
 WHERE job_id = 'MK_MAN' );
```

18. Write a query to display the employee number, name(first name and last name) and job title for all employees whose salary is smaller than any salary of those employees whose job title is MK_MAN. Exclude Job title MK_MAN.

```
SELECT employee_id,first_name,last_name, job_id
FROM employees
WHERE salary < ANY
( SELECT salary
FROM employees
WHERE job_id = 'MK_MAN' )
AND job_id <> 'MK_MAN';
```

19. Write a query to display the employee number, name(first name and last name) and job title for all employees whose salary is more than any salary of those employees whose job title is PU_MAN. Exclude job title PU_MAN.

```
SELECT employee_id, first_name, last_name, job_id
FROM employees
WHERE salary > ALL
( SELECT salary
FROM employees
WHERE job_id = 'PU_MAN' )
AND job_id <> 'PU_MAN';
```

20. Write a query to display the employee number, name(first name and last name) and job title for all employees whose salary is more than any average salary of any department.

```
SELECT employee_id, first_name, last_name, job_id
FROM employees
WHERE salary > ALL
( SELECT AVG(salary)
FROM employees
GROUP BY department_id
);
```

21. Write a query to display the employee name(first name and last name) and department for all employees for any existence of those employees whose salary is more than 3700.

```
SELECT first_name, last_name, department_id
FROM employees
WHERE EXISTS
(SELECT *
FROM employees
WHERE salary >3700 );
```

22. Write a query to display the department id and the total salary for those departments which contains at least one employee.

```
SELECT departments.department_id, result1.total_amt
FROM departments,
( SELECT employees.department_id, SUM(employees.salary) total_amt
FROM employees
GROUP BY department_id) result1
WHERE result1.department_id = departments.department_id;
```

23. Write a query to display the employee id, name (first name and last name) and the job id column with a modified title SALESMAN for those employees whose job title is ST_MAN and DEVELOPER for whose job title is IT_PROG.

```
SELECT  employee_id,  first_name, last_name,
CASE job_id
WHEN 'ST_MAN' THEN 'SALESMAN'
```

```
WHEN 'IT_PROG' THEN 'DEVELOPER'
ELSE job_id
END AS designation,  salary
FROM employees;
```

24. Write a query to display the employee id, name (first name and last name), salary and the SalaryStatus column with a title HIGH and LOW respectively for those employees whose salary is more than and less than the average salary of all employees.

```
SELECT  employee_id,  first_name, last_name, salary,
CASE WHEN salary >= (SELECT AVG(salary)
FROM employees) THEN 'HIGH'
ELSE 'LOW'
END AS SalaryStatus
FROM employees;
```

25. Write a query to display the employee id, name (first name and last name), SalaryDrawn, AvgCompare (salary - the average salary of all employees) and the SalaryStatus column with a title HIGH and LOW respectively for those employees whose salary is more than and less than the average salary of all employees.

```
SELECT  employee_id,  first_name, last_name,  salary AS SalaryDrawn,
ROUND((salary -(SELECT AVG(salary) FROM employees)),2) AS AvgCompare,
CASE  WHEN salary >=
(SELECT AVG(salary)
FROM employees) THEN 'HIGH'
ELSE 'LOW'
END AS SalaryStatus
FROM employees;
```