**Project Introduction**

In this project, we will learn about a company's details record and how its related data tables collaborate with the main table. The schema for this project includes six tables: countries, departments, dependents, jobs, locations, regions & employees.

The company's entity relationship diagram is shown in the illustration below. As you can see, the employees table is the parent table, which is linked to countries, departments, dependents, jobs, locations, and regions.

Please take note that, as per the requirement document, below table have 33 data entries or more.

Diagram

Description automatically generated

**Project Goal**

The project is aiming to cover all functions learned throughout the timeline of six week classes or otherwise gathered knowledge from personal experiences and self-assessment. It helps appointment fundamental details around creation of table, manipulating data, creating views, Sums, JOINS, aggregate functions, etc.

**Relational Schema**

Graphical user interface, application

Description automatically generated

**Data Table Creation (with SQL) and its Relevant Data Entries:**

Create Queries:

**Graphical user interface, text

Description automatically generated**

**Text

Description automatically generated**

Data Creation for Regions :

**Graphical user interface, text, application, email

Description automatically generated**

Data Creation for Countries : Graphical user interface, text

Description automatically generated

Data Creation for Countries :

Graphical user interface, text, application, email

Description automatically generated

Data Creation for Departments :

Graphical user interface, text, application

Description automatically generated

Data Creation for Jobs :

Text

Description automatically generated

Data Creation for Employees :

A screenshot of a computer

Description automatically generated with medium confidence

Data Creation for Dependents :

Graphical user interface, application

Description automatically generated

**Queries:**

**--**Display employees which has unique department id.

SELECT DISTINCT department\_id

FROM employees;

Graphical user interface

Description automatically generated

-- Display details of all employees from the employee table in ascending order by their first name.

SELECT \*

FROM employees

ORDER BY first\_name ASC;

Table

Description automatically generated with medium confidence

--What are the total salaries payable to employees.

SELECT SUM(salary)

FROM employees;

Graphical user interface, text, email

Description automatically generated

-- Display the maximum and minimum salary paid to the employees.

SELECT MAX(salary), MIN(salary)

FROM employees;

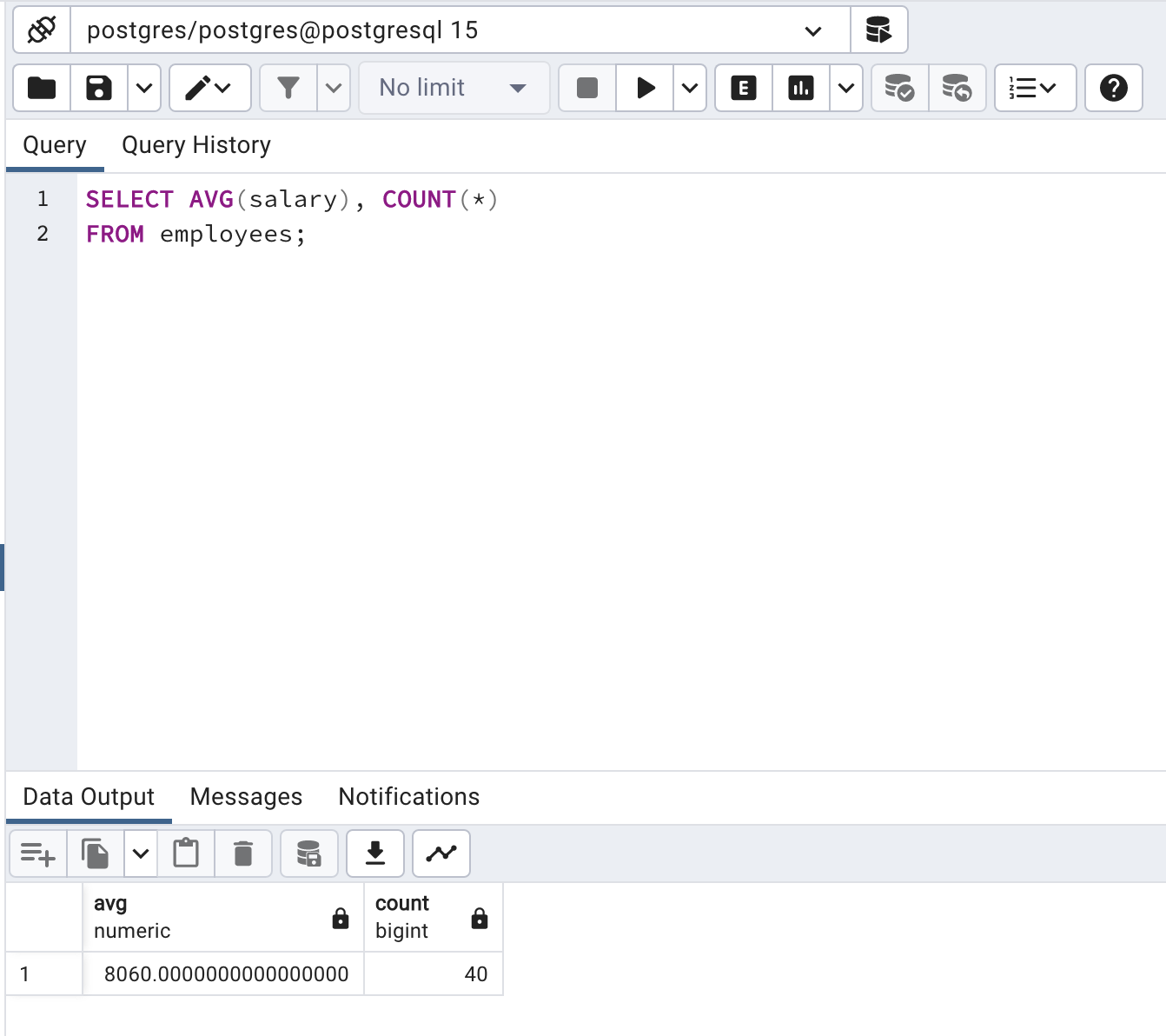
Graphical user interface, text, email

Description automatically generated

--What’s the average salary and number of employees are working.

SELECT AVG(salary), COUNT(\*)

FROM employees;



6. Display the first names of employees after removing all the leading and trailing blanks of all the employees from employees table also order by ascending name.

SELECT TRIM(first\_name)

FROM employees

ORDER BY first\_name ASC;

Graphical user interface, text, email

Description automatically generated

7. Which employees belongs to which departments including their department ids.

SELECT first\_name, last\_name, department\_id, department\_name

FROM employees

JOIN departments USING (department\_id);

Graphical user interface, text, email

Description automatically generated

8. What is the highest, lowest, total, and average salary of all employees of the company.

SELECT ROUND(MAX(salary),0) "Maximum",

ROUND(MIN(salary),0) "Minimum",

ROUND(SUM(salary),0) "Sum",

ROUND(AVG(salary),0) "Average"

FROM employees;

Graphical user interface, text, email

Description automatically generated

9. What is the number of employees who were hired in year 1987

SELECT first\_name, last\_name, hire\_date

FROM employees

WHERE TO\_CHAR(hire\_date, 'YYYY') LIKE '%87';

Text

Description automatically generated

10. What is the total salary payable in each department along with department ID.

SELECT department\_id, SUM(salary)

FROM employees

GROUP BY department\_id;

Graphical user interface, text, email

Description automatically generated

References

The sample data for this SQL project was obtained from the website listed below.

*PostgreSQL Exercises, Practice, Solution*. (2022, August 19). w3resource. Retrieved February 25, 2023, from <https://www.w3resource.com/postgresql-exercises/>