**SQL Company HR Database**

1. All locations.

SELECT locations.street\_address AS Street, locations.postal\_code AS Zip,

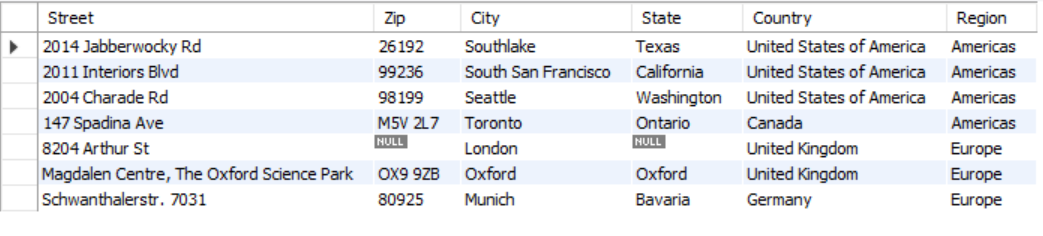
locations.city AS City, locations.state\_province AS State,

countries.country\_name AS Country, regions.region\_name AS Region

FROM countries

JOIN regions ON countries.region\_id = regions.region\_id

JOIN locations ON countries.country\_id = locations.country\_id;



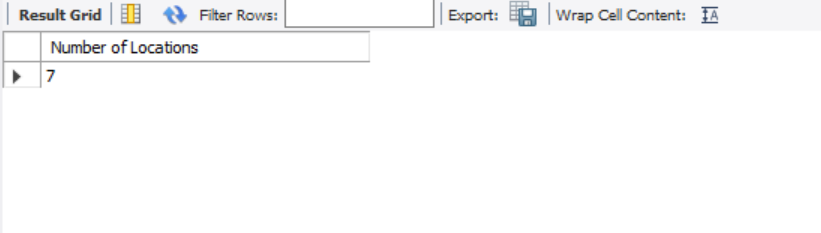
1. Number of locations in each region

SELECT COUNT(\*) AS 'Number of Locations'

FROM countries

JOIN regions ON countries.region\_id = regions.region\_id

JOIN locations ON countries.country\_id = locations.country\_id;

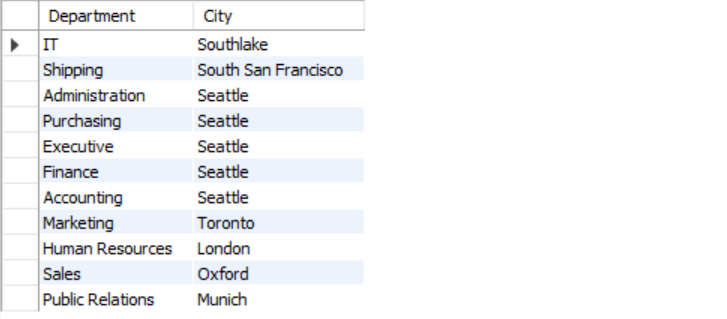


1. Departments and the locations

SELECT d.department\_name AS Department, l.city AS City

FROM departments as d

JOIN locations as l ON d.location\_id = l.location\_id;



1. Employee with the highest salary

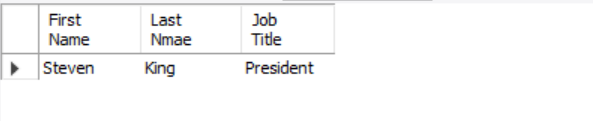
SELECT e.first\_name AS 'First Name', e.last\_name AS 'Last Nmae', j.job\_title AS 'Job Title'

FROM employees AS e

JOIN jobs AS j ON e.job\_id = j.job\_id

ORDER BY e.salary DESC

LIMIT 1;



1. Employees who have children

SELECT CONCAT(e.first\_name, ' ', e.last\_name) AS 'Parent Name',

CONCAT(d.first\_name, ' ', d.last\_name) AS 'Child Name'

FROM employees AS e

INNER JOIN dependents AS d ON e.employee\_id = d.employee\_id;



1. Managers in each department

SELECT CONCAT(e.first\_name, ' ', e.last\_name) AS 'Name', j.job\_title AS 'Title'

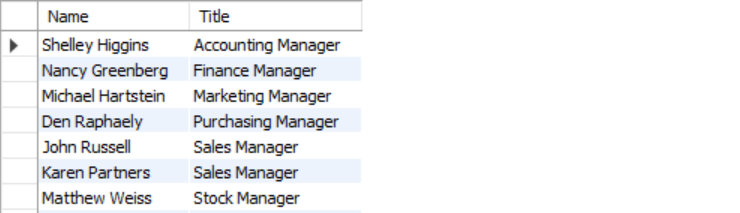
FROM employees AS e

JOIN ( SELECT job\_id, job\_title

FROM jobs

WHERE lower(job\_title) LIKE '%manager') AS j

ON e.job\_id = j.job\_id;



1. Employees who make the most salary in each job title

SELECT CONCAT(e.first\_name, ' ', e.last\_name) AS 'Name',

j.job\_title AS 'Job Title',

e.salary AS 'Salary'

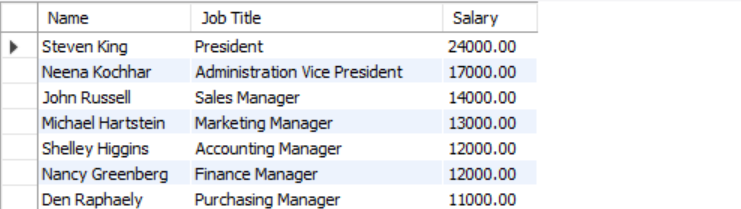
FROM ( SELECT employee\_id, first\_name, last\_name, MAX(salary) AS 'salary', job\_id

FROM employees

GROUP BY job\_id ) AS e

JOIN jobs AS J ON e.job\_id = j.job\_id

ORDER BY e.salary DESC;

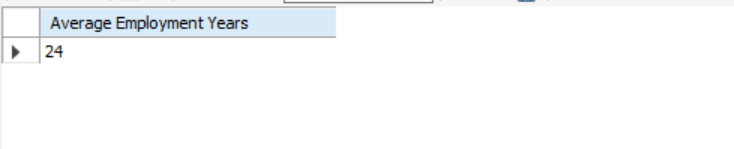


1. Average years of employment

SELECT ROUND(AVG(2020 - e.Year), 0) AS 'Average Employment Years'

FROM ( SELECT employee\_id, YEAR(STR\_TO\_DATE(hire\_date, "%Y-%m-%d")) AS Year

FROM employees ) AS e



1. Average years of employment in each department

SELECT d.department\_name AS 'Department', ROUND(AVG(2020 - e.Year), 0) AS 'Average Employment Years'

FROM ( SELECT employee\_id, department\_id, YEAR(STR\_TO\_DATE(hire\_date, "%Y-%m-%d")) AS Year

FROM employees ) AS e

JOIN departments AS d ON e.department\_id = d.department\_id

GROUP BY d.department\_name

ORDER BY d.department\_name;

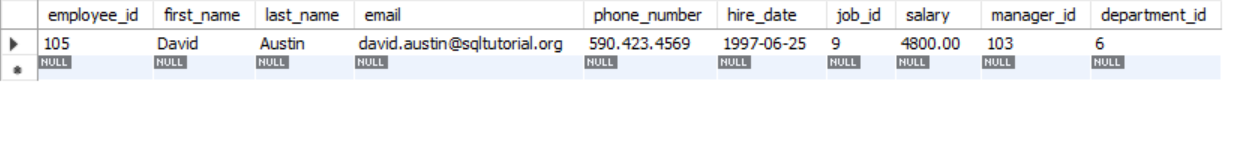


1. Get all employees with the first name starts with ‘a’

SELECT \*

FROM employees

WHERE lower(last\_name) LIKE 'a%';



1. Employees who do not have ‘id’ in their name

SELECT \*

FROM employees

WHERE lower(last\_name) NOT LIKE '%id%'

AND lower(first\_name) NOT LIKE '%id%';



1. Employees whose salaries are between 5000 and 10000

SELECT CONCAT(e.first\_name, ' ', e.last\_name) AS 'Name',

e.salary AS 'Salary',

j.job\_title AS 'Job Title'

FROM employees AS e

JOIN jobs AS j ON e.job\_id = j.job\_id

WHERE e.salary BETWEEN 5000 AND 10000;



1. Employees with their name, job title, salary, and indications that if the salary is high, medium, or low.

SELECT CONCAT(e.first\_name, ' ', e.last\_name) AS 'Name',

j.job\_title AS 'Job Title',

e.salary AS 'Salary',

CASE

WHEN e.salary BETWEEN 0 AND 5000 THEN 'Low'

WHEN e.salary BETWEEN 5001 AND 10000 THEN 'Medium'

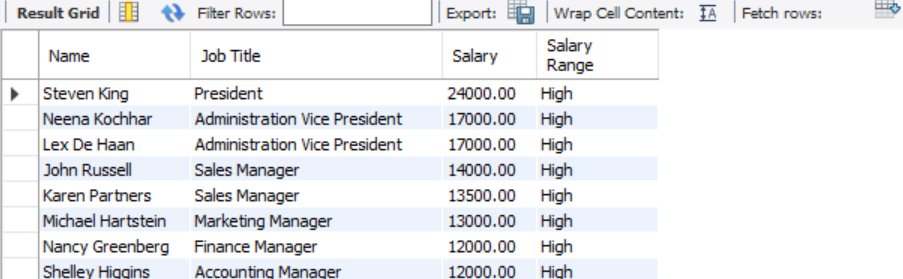
WHEN e.salary > 10001 THEN 'High'

ELSE 'N/A' END AS 'Salary Range'

FROM employees AS e

JOIN jobs AS j ON e.job\_id = j.job\_id

ORDER BY e.salary DESC;



1. Average salary of each job position, number of people in the position, and the range (low, medium, high).

SELECT j.job\_title AS 'Title',

e.number\_of\_employees AS 'Number of Employees',

e.average\_salary AS 'Average Salary',

CASE

WHEN e.average\_salary BETWEEN 0 AND 5000 THEN 'Low'

WHEN e.average\_salary BETWEEN 5001 AND 10000 THEN 'Medium'

WHEN e.average\_salary > 10001 THEN 'High'

ELSE 'N/A' END AS 'Salary Range'

FROM ( SELECT COUNT(\*) AS number\_of\_employees,

ROUND(AVG(salary), 0) AS average\_salary,

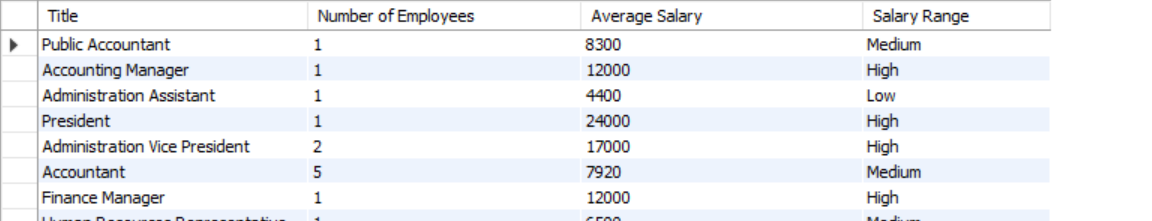
job\_id

FROM employees

GROUP BY job\_id

) AS e

INNER JOIN jobs j ON j.job\_id = e.job\_id



1. Average salary of each department, number of people in the department, and the range (low, medium, high).

SELECT d.department\_name AS 'Department',

e.number\_employees AS 'Number of Employees',

e.average\_salary AS 'Average Salary',

CASE

WHEN e.average\_salary BETWEEN 0 AND 5000 THEN 'Low'

WHEN e.average\_salary BETWEEN 5001 AND 10000 THEN 'Medium'

WHEN e.average\_salary > 10000 THEN 'High'

ELSE 'N/A' END AS 'Salary Range'

FROM ( SELECT COUNT(\*) AS number\_employees,

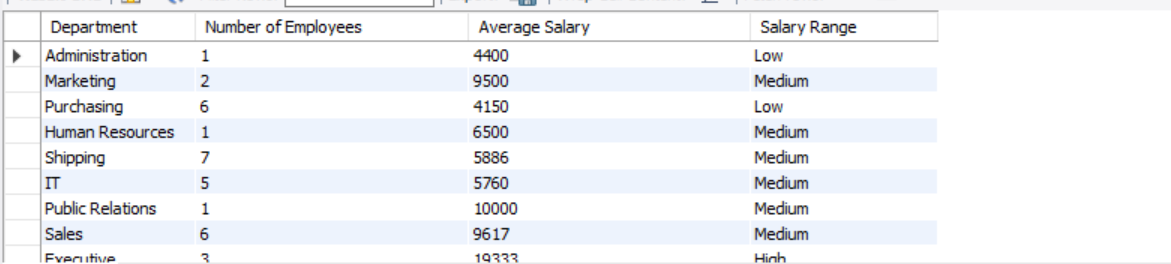
ROUND(AVG(salary), 0) AS average\_salary,

department\_id

FROM employees

GROUP BY department\_id ) AS e

JOIN departments AS d ON e.department\_id = d.department\_id



1. New password for the employees by combining email address ID and the last 4 digits of phone number

SELECT CONCAT(first\_name, ' ', last\_name) AS 'Name',

email AS 'Email',

phone\_number AS 'Phone Number',

CONCAT(SUBSTRING(email, 1, LOCATE('@', email) - 1), SUBSTRING(phone\_number, -4)) AS 'New Password'

FROM employees

