SQL Practice exercises

Goal: Put into practice what you have learned so far in the SQL lectures (basic queries and more advanced queries)

There are 15 questions to solve with queries that are roughly of increasing difficulty:

* Level 1: Stretching
* Level 2: Pouring some spices
* Level 3: Why so serious?

**The document to submit should be either a doc file or a pdf file with the answers to each question:**

1. **The query you ran**
2. **The result you got**

# Setup instructions

1. [Download the repository](https://github.com/datacharmer/test_db/archive/refs/heads/master.zip). Unzip the file.
2. Launch your terminal/command shell and go to the repository where you downloaded the file. Enter the folder (that you have unzipped).
3. Run the following command: *mysql -u root -p < employees.sql*

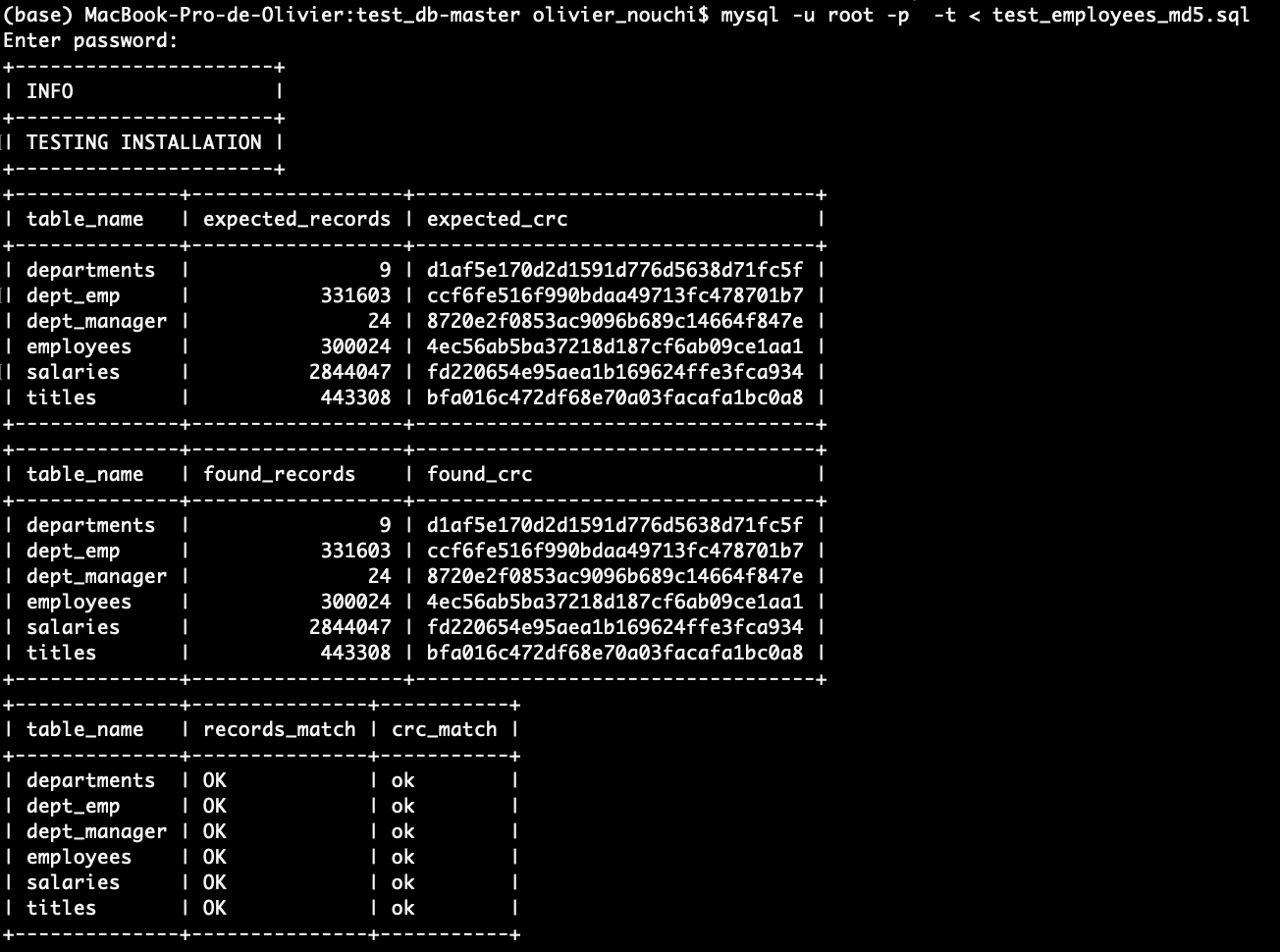
You should see the tables loading:



You have created the employees’ table with the tables cited above.

1. Then run the following command line: *mysql -u root -p -t < test\_employees\_md5.sql*

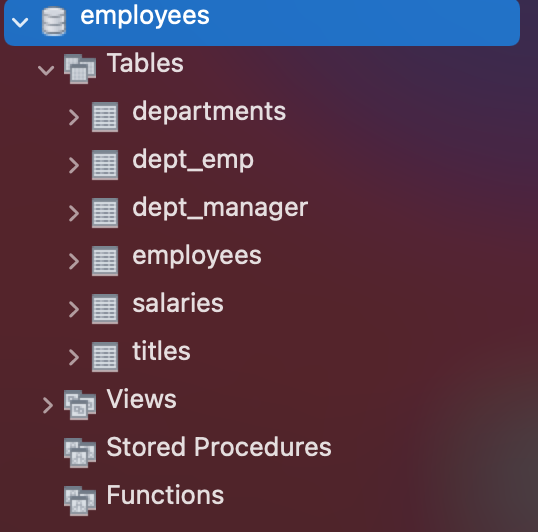
It will make sure that you have loaded everything correctly. You should be able to see the following (records matched).



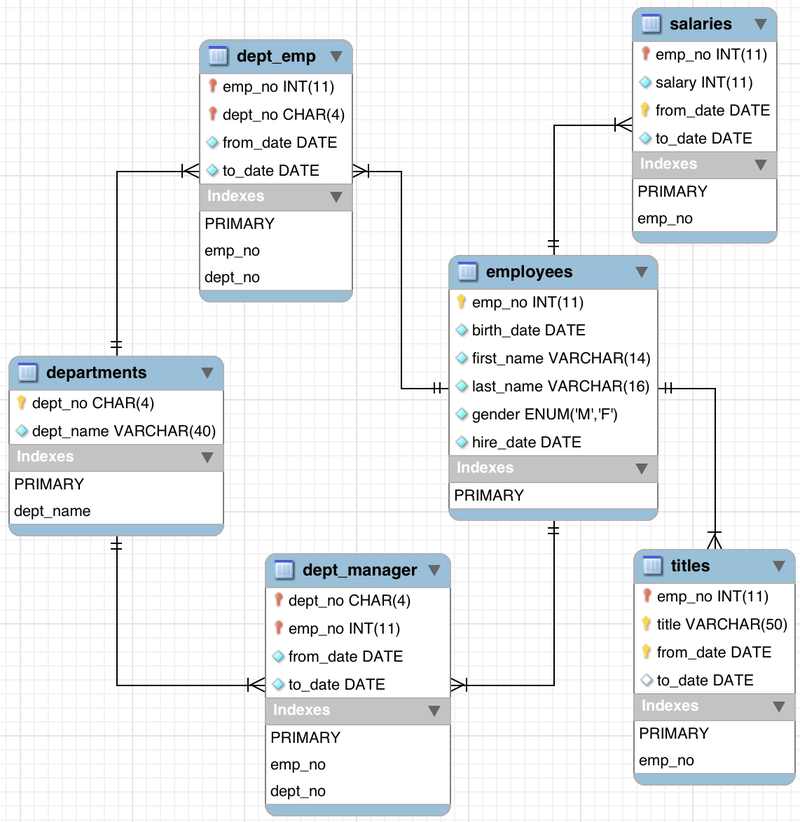
You are all set!

You can now work in MySQL Workbench.

Don’t forget to “Refresh All” if you don’t see it appear in MySQL Workbench.



# Database schema



# 

# 

# Stretching

Q1 - How many unique employees are there?

SELECT DISTINCT count(emp\_no)

FROM employees;

**Result:**



Q2 - How many males and females employees are there? Order the counts in descending order.

SELECT gender, count(gender)

FROM employees

GROUP BY gender

ORDER BY gender DESC;

**Result:**



Q3 - Display the year and total hires for the year with the most hires

SELECT YEAR(hire\_date), count(hire\_date)

FROM employees

GROUP BY YEAR(hire\_date)

ORDER BY count(hire\_date) DESC

LIMIT 1;

**Result:**



Q4 - What is the name of the department with the most employees

SELECT departments.dept\_name, count(dept\_emp.emp\_no)

FROM departments

INNER JOIN dept\_emp

ON departments.dept\_no = dept\_emp.dept\_no

GROUP BY departments.dept\_name

ORDER BY count(dept\_emp.emp\_no) DESC LIMIT 1;

**Result:**



Q5 - How many employees were born on November 12? What's the percentage out of all the employees?

SELECT count(birth\_date), count(birth\_date)\*100 / (SELECT count(\*) FROM employees) AS percentage

FROM employees

WHERE MONTH(birth\_date) = 11 AND DAY(birth\_date) = 12;

**Result:**



Q6 - What are the 3 most common employee titles (display the employee titles and the number of times they occur)

SELECT count(emp\_no), title

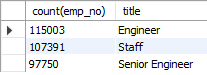
FROM titles

GROUP BY title

ORDER BY count(emp\_no) DESC

LIMIT 3;

**Result:**



Q7 - Find the avg salary for each department (department name). Round to the nearest integer and order by avg salary from the highest to the lowest.

SELECT departments.dept\_name, ROUND(AVG(salary),0) as salary\_average

FROM salaries

INNER JOIN dept\_emp

ON salaries.emp\_no = dept\_emp.emp\_no

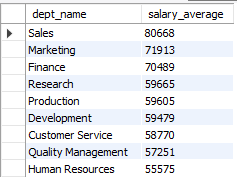
INNER JOIN departments

ON departments.dept\_no = dept\_emp.dept\_no

GROUP BY dept\_name

ORDER BY salary\_average DESC;

**Result:**



# Pouring some spices

Q8 - Find the average salary by employee title. Round to 2 decimals and order by descending order

SELECT titles.title, ROUND(AVG(salaries.salary),2) as salary\_average

FROM salaries

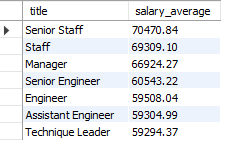
INNER JOIN titles

ON salaries.emp\_no = titles.emp\_no

GROUP BY title

ORDER BY salary\_average DESC;

**Result:**



Q9 - Find the number of employees who have worked in at least 2 departments

SELECT COUNT(\*)

FROM (SELECT emp\_no, COUNT(\*) AS CNT FROM dept\_emp GROUP BY emp\_no) AS T

WHERE CNT > 1;

**Result:**



Q10 - Get the distribution of the year of the hire dates. (hint: you should end up with a number of employees per year of hiring date)

Do you notice any pattern? Assuming there is no missing data, is the company hiring more or less as time goes by?

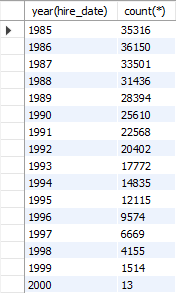
SELECT year(hire\_date), count(\*)

FROM employees

GROUP BY year(hire\_date)

ORDER BY year(hire\_date) ASC;

**Result:**



The company is hiring less employees as time goes by.

Q11 - Display the first name, last name, and salary of the highest paid employee

SELECT employees.first\_name, employees.last\_name, salaries.salary

FROM salaries

INNER JOIN employees

ON salaries.emp\_no = employees.emp\_no

ORDER BY salaries.salary DESC LIMIT 1;

**Result:**

****

Q12 - Display the first name, last name, and salary of the THIRD highest paid employee

SELECT employees.first\_name, employees.last\_name, salaries.salary

FROM salaries

INNER JOIN employees

ON salaries.emp\_no = employees.emp\_no

ORDER BY salaries.salary DESC LIMIT 1 OFFSET 2;

**Result:**

****

# Why so serious?

Q13 - Display each department name and the age of the youngest employee at hire date

SELECT dept.dept\_name, MIN(TIMESTAMPDIFF(YEAR, emp.birth\_date, emp.hire\_date)) AS age\_hire\_date

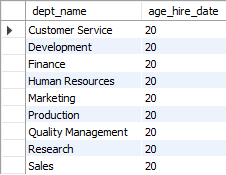
FROM employees AS emp

INNER JOIN dept\_emp d\_emp ON emp.emp\_no = d\_emp.emp\_no

INNER JOIN departments dept ON d\_emp.dept\_no = dept.dept\_no

GROUP BY dept.dept\_name;

**Result:**



Q14 - What's the range of age the employees would be today (calculate their age whole years)

SELECT min(age), max(age)

FROM (SELECT (YEAR(current\_date()) - YEAR(birth\_date)) AS age FROM employees)

AS emp;

**Result:**



Q15 - How many employees were born on the same date (day-month-year) in 1955.

NB: Change the according to preferences parameters if your query fails because of a timeout

SELECT sum(quantity) as 'Same date in 1955' FROM

(SELECT count(birth\_date) AS quantity

FROM employees

WHERE YEAR(birth\_date) = '1955'

GROUP BY birth\_date) AS birth;

**Result:**

****