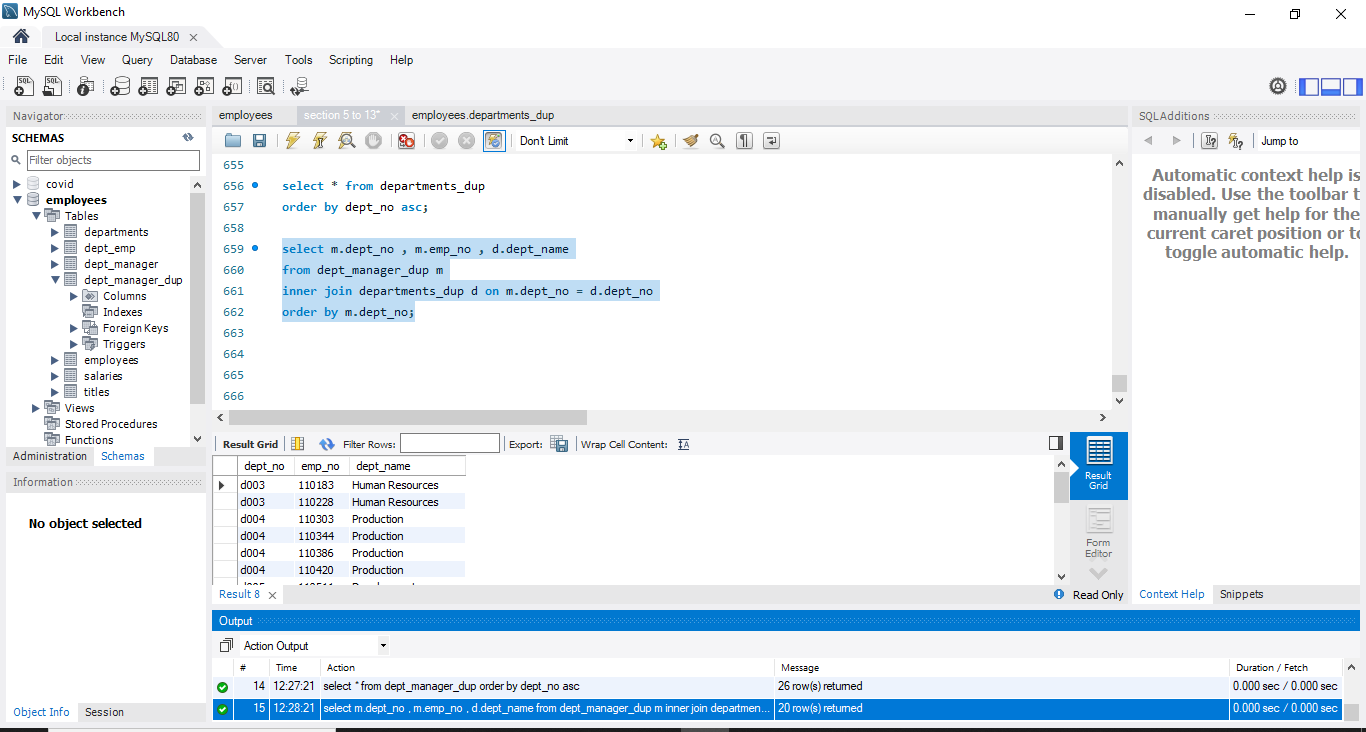
**SECTION 14**

In this section, we will explore our next SQL tool – JOINS – that helps us build a relationship between objects.

Our first subtopic is Inner JOIN.

We will use department duplicate and departments manager duplicate tables to JOIN. To simplify, we used D & M respectively as aliases for above-mentioned tables. Here is the syntax:

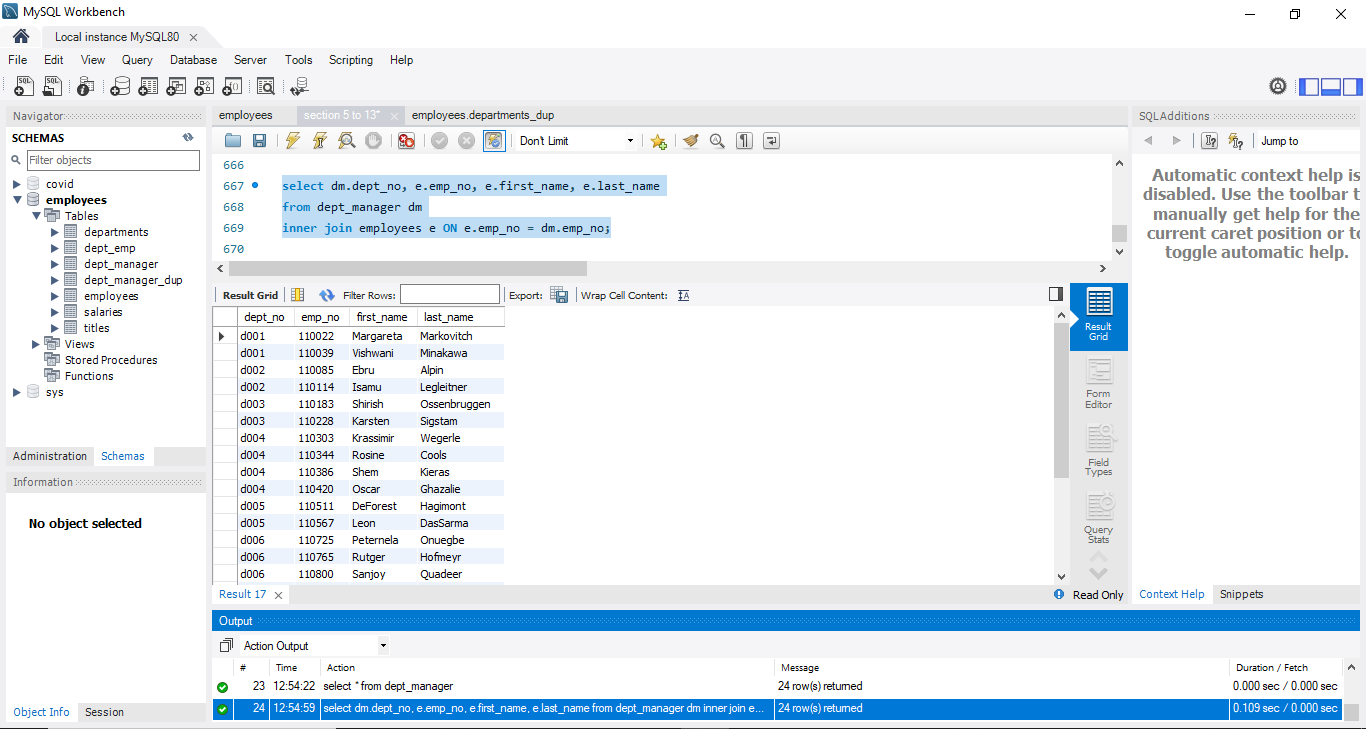


**Exercise:**

Extract a list containing information about all managers’ employee number, first and last name, department number, and hire date.

**Solution:**

We can get employees details (employee number, first name, last name, and hire date) from employees table. As for department number, we can get that information from department manager table. Below is the query and output to get our final result.

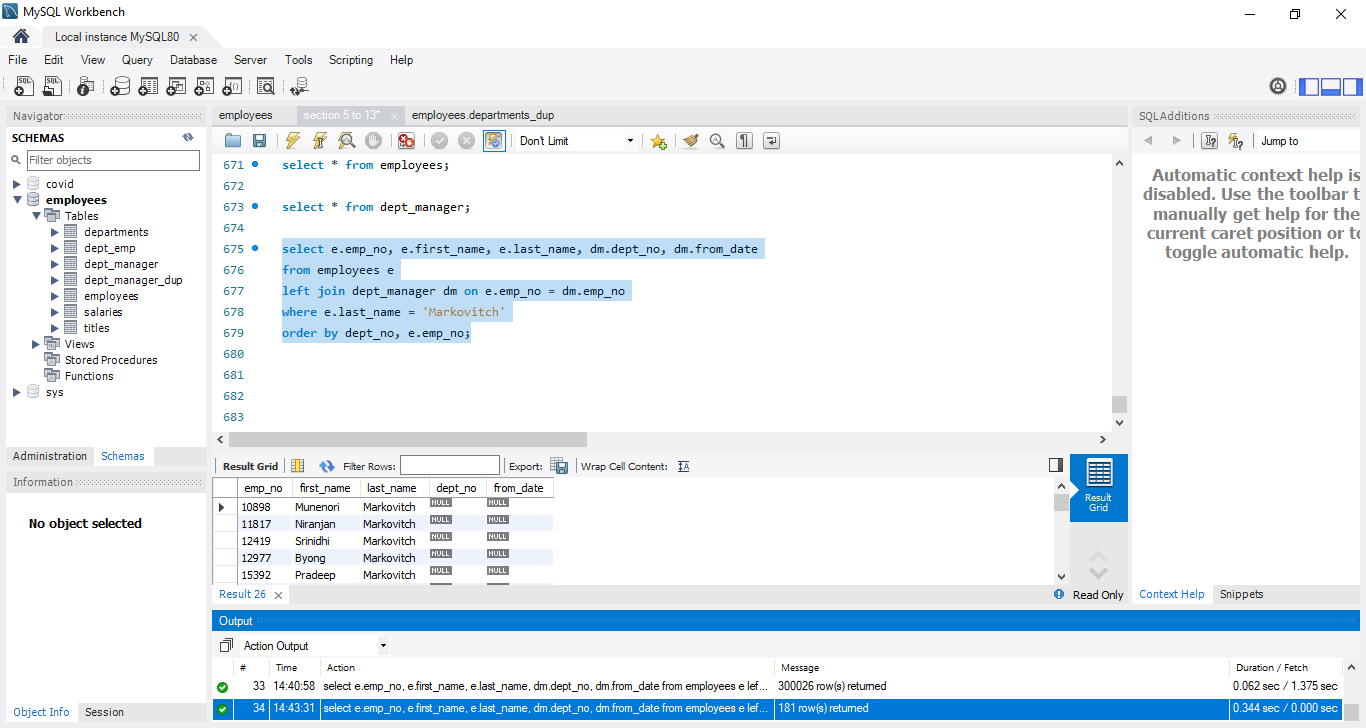


**Exercise:**

Join the 'employees' and the 'dept\_manager' tables to return a subset of all the employees whose last name is Markovitch. See if the output contains a manager with that name.

*Hint: Create an output containing information corresponding to the following fields: ‘emp\_no’, ‘first\_name’, ‘last\_name’, ‘dept\_no’, ‘from\_date’. Order by 'dept\_no' descending, and then by 'emp\_no'.*

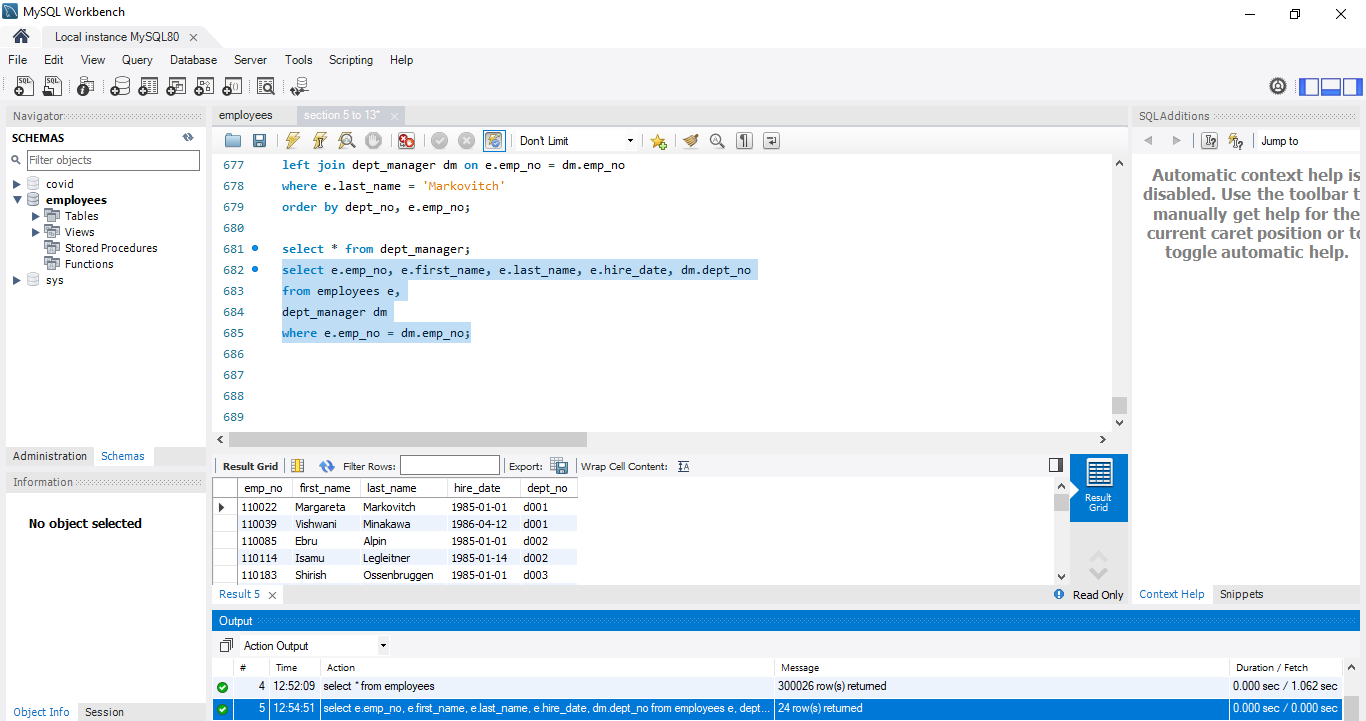
**Solution:**



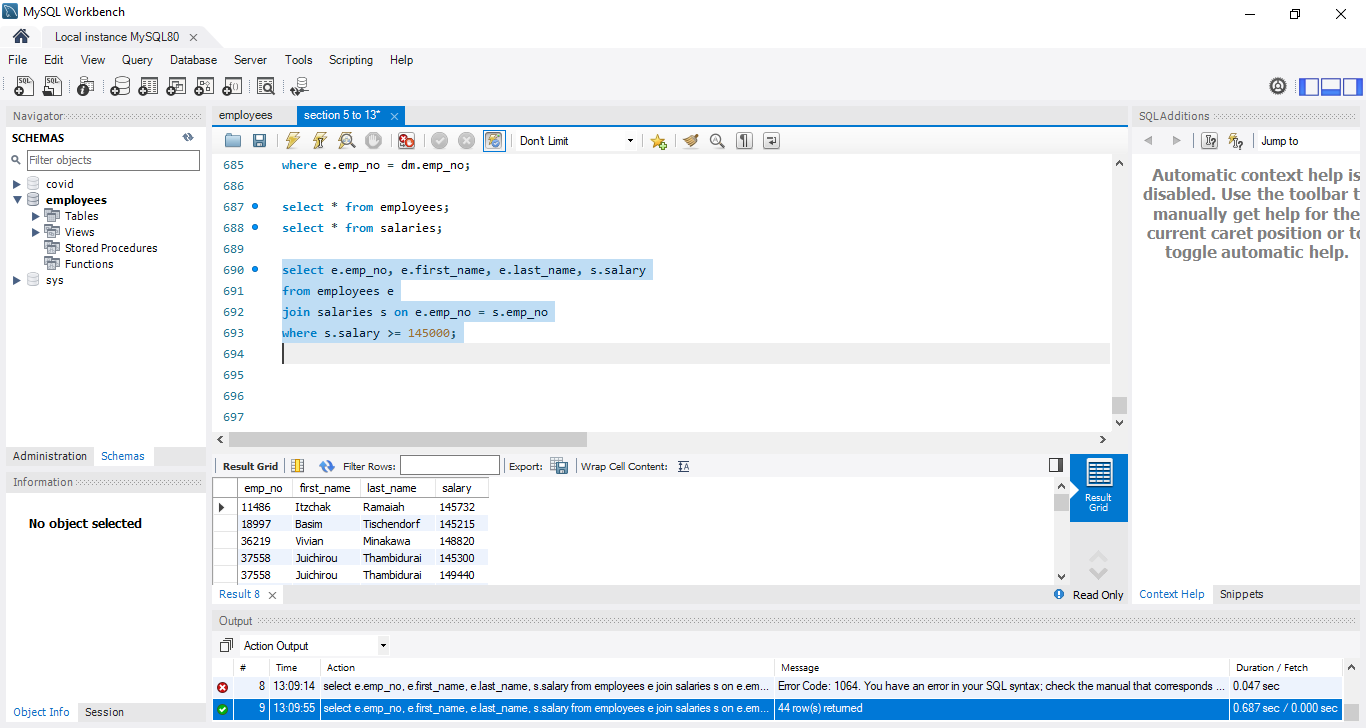
**Exercise:**

Extract a list containing information about all managers’ employee number, first and last name, department number, and hire date. Use the old type of join syntax to obtain the result.

**Solution:**



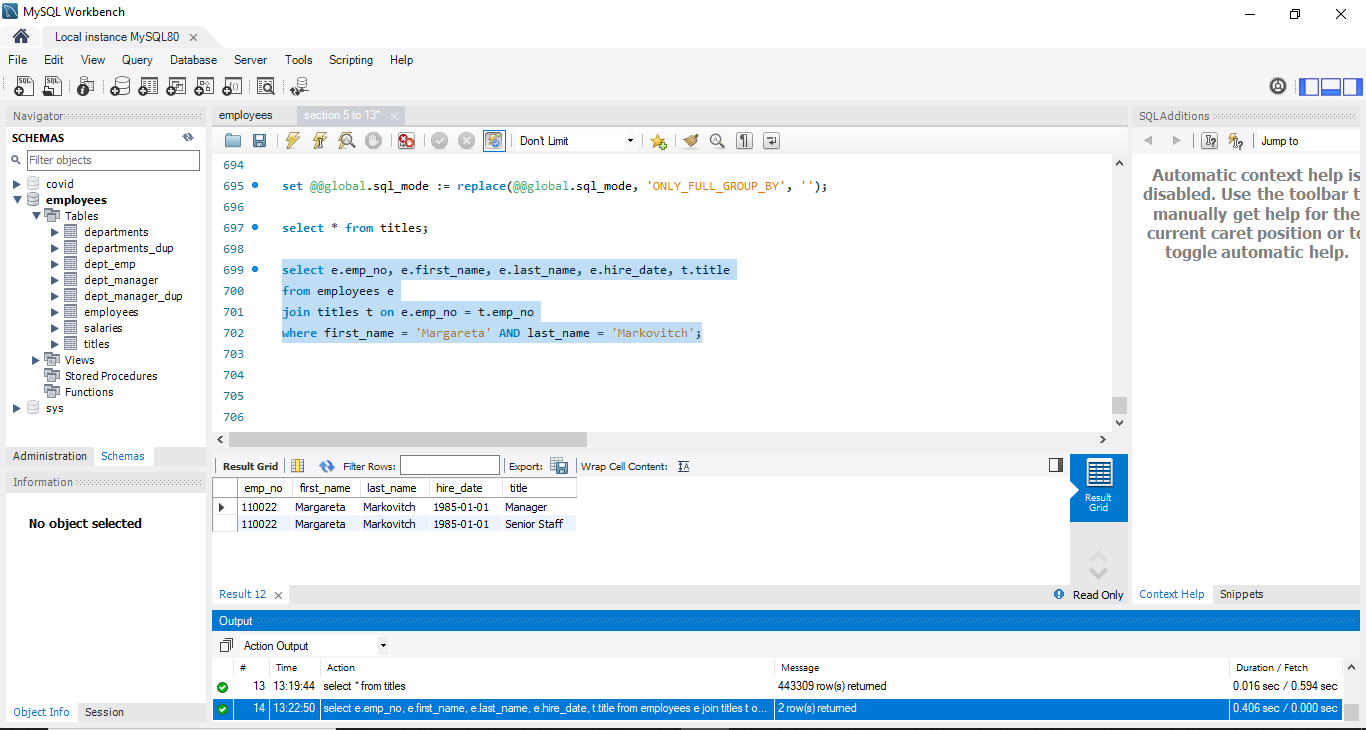
Now, we will use JOIN and WHERE together. Our question here is we want to join two tables: employees and salary and see the first name and last name of the employees whose annual salary is greater than $145,000.



**Exercise:**

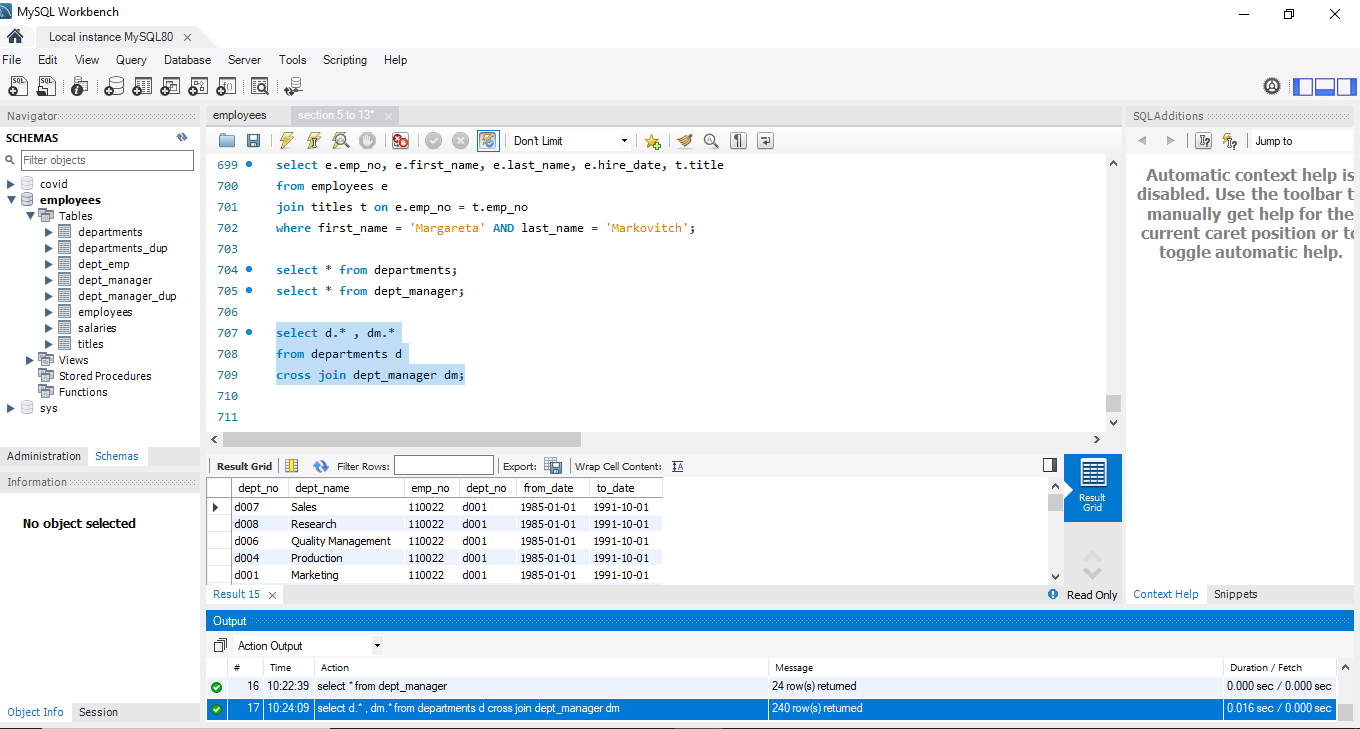
Select the first and last name, the hire date, and the job title of all employees whose first name is “Margareta” and have the last name “Markovitch”.

**Solution:**



Our next topic is Cross Join. While our employees database is well-connected, we can still see some examples of cross join. Here is our first example.

Suppose we wish to obtain a result set with data containing all the department managers and the departments they can be assigned to. This means we will need department manager table and departments table.

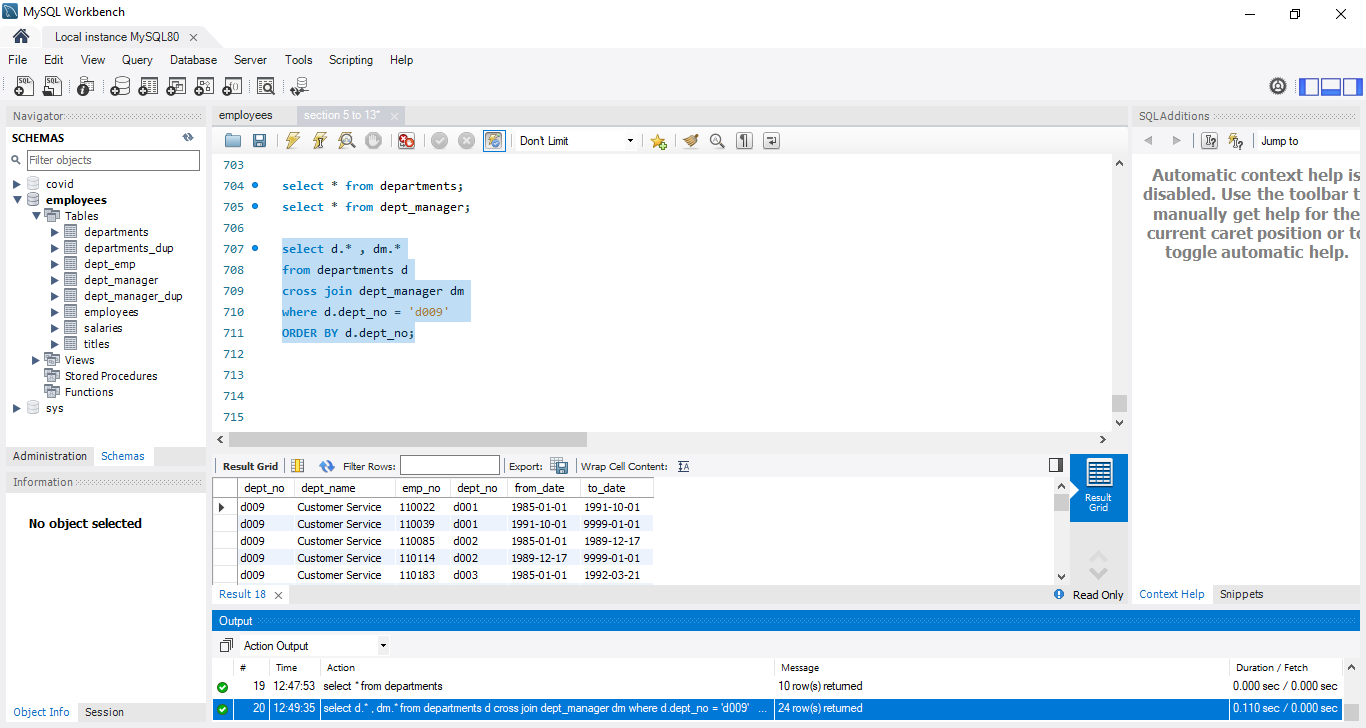


There are nine different department numbers that correspond to the employee number of each manager.

**Exercise:**

Use a CROSS JOIN to return a list with all possible combinations between managers from the dept\_manager table and department number 9.

**Solution:**

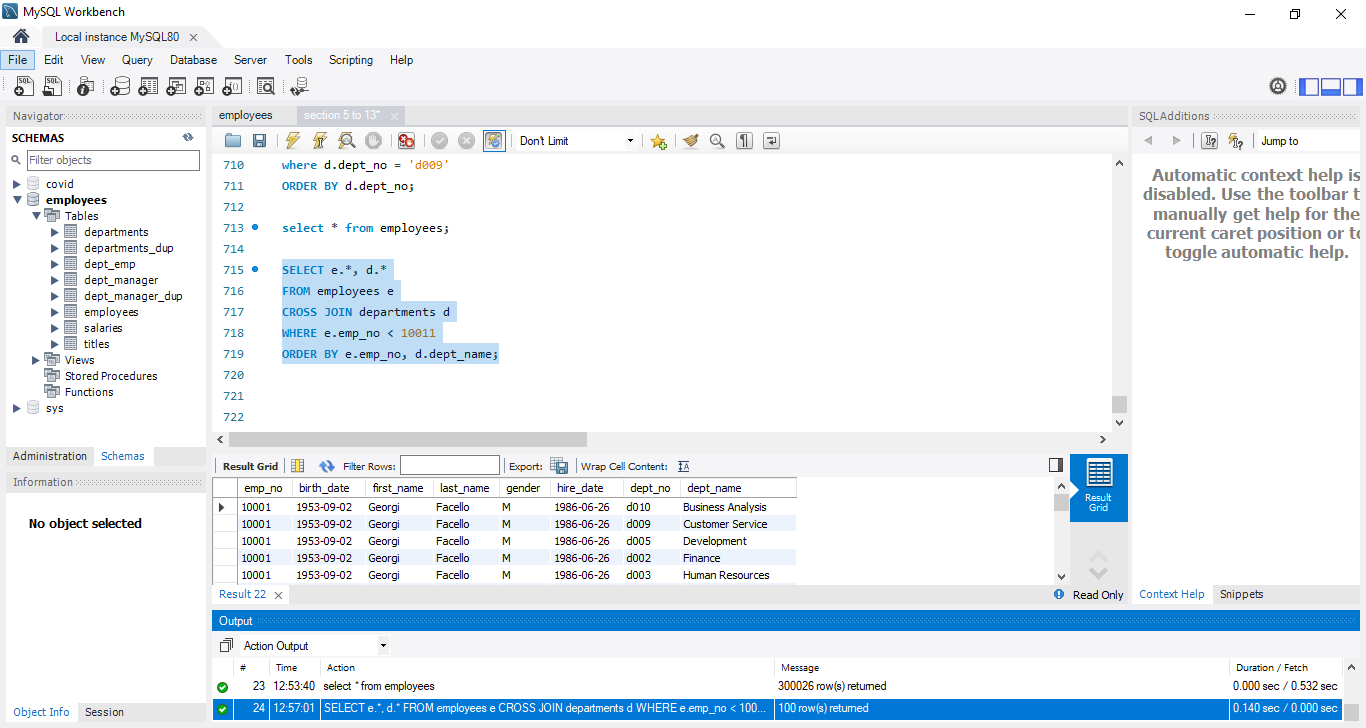


**Exercise:**

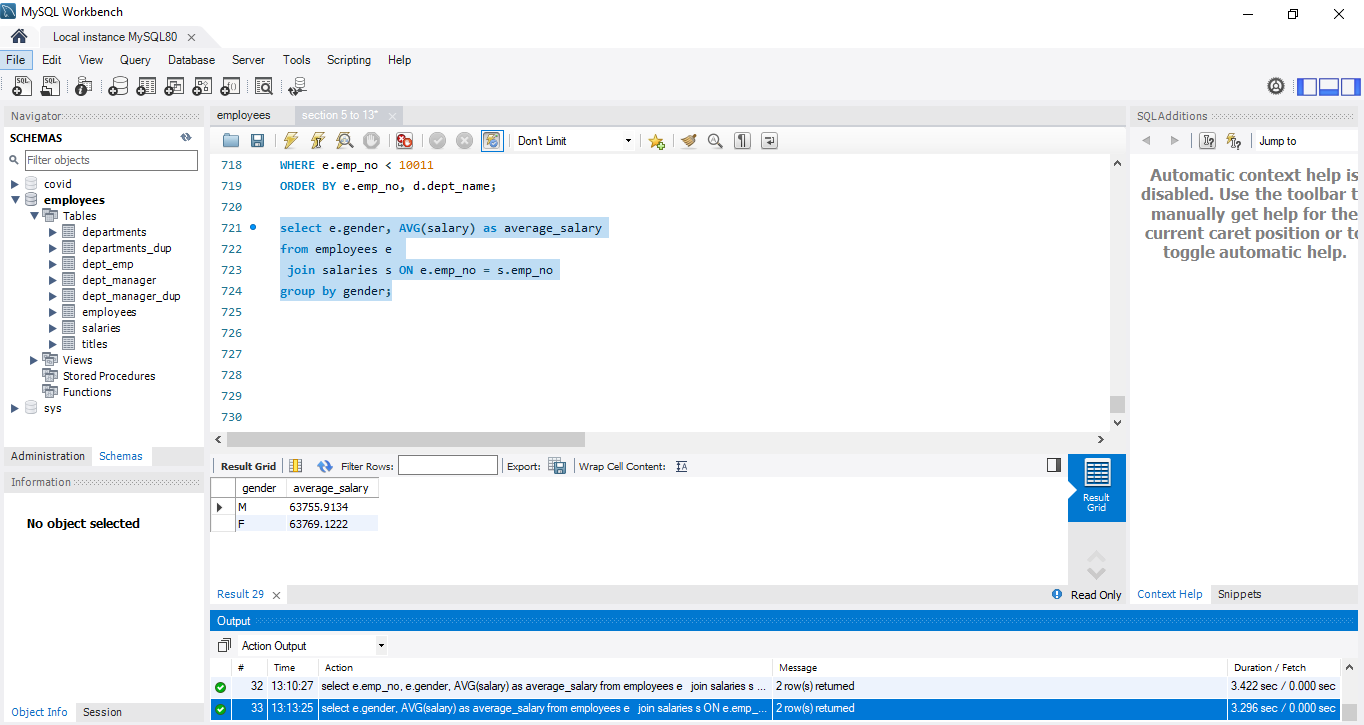
Return a list with the first 10 employees with all the departments they can be assigned to.

*Hint: Don’t use LIMIT; use a WHERE clause.*

**Solution:**

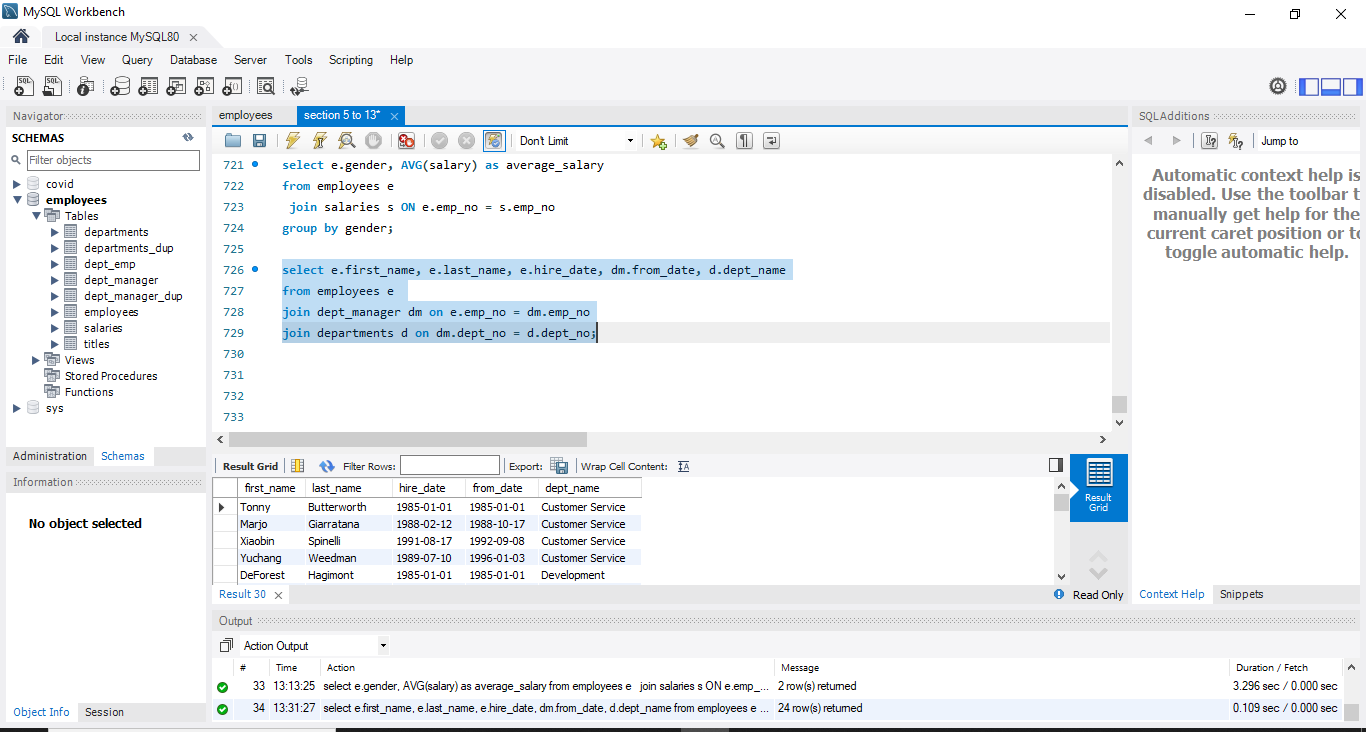


Now, our next question is we want to know the average salary of men and women. We can employ JOIN followed by group by to find our answer. See the result below:



Let’s suppose we want to join more than two tables. We want first name, last name, hire date, from date, and department name all in one place. By looking at relational schema, we can say we will be using three tables: employees, department manager, and departments for our query to get the desired result.

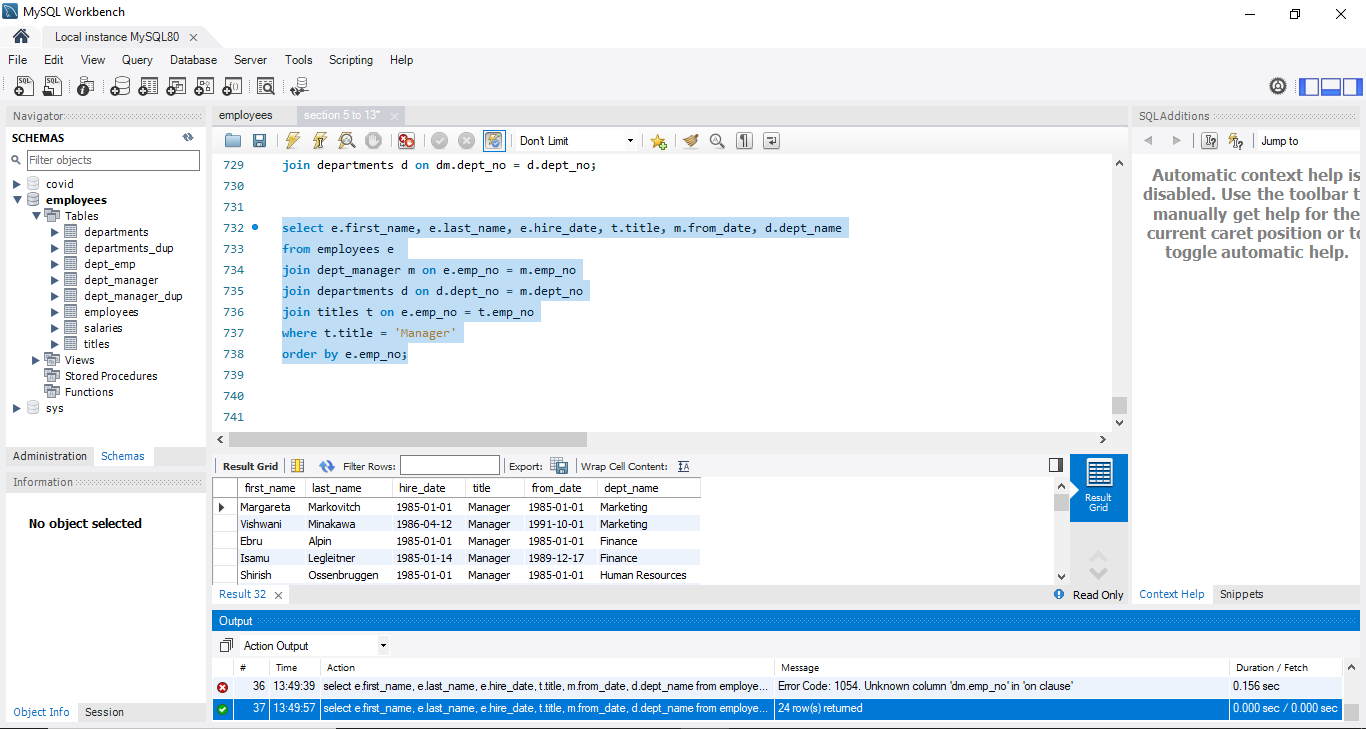
(Column emp\_no will connect employees and dept\_manager tables and dept\_no will join departments and dept\_manager table)



**Exercise:**

Select all managers’ first and last name, hire date, job title, start date, and department name.

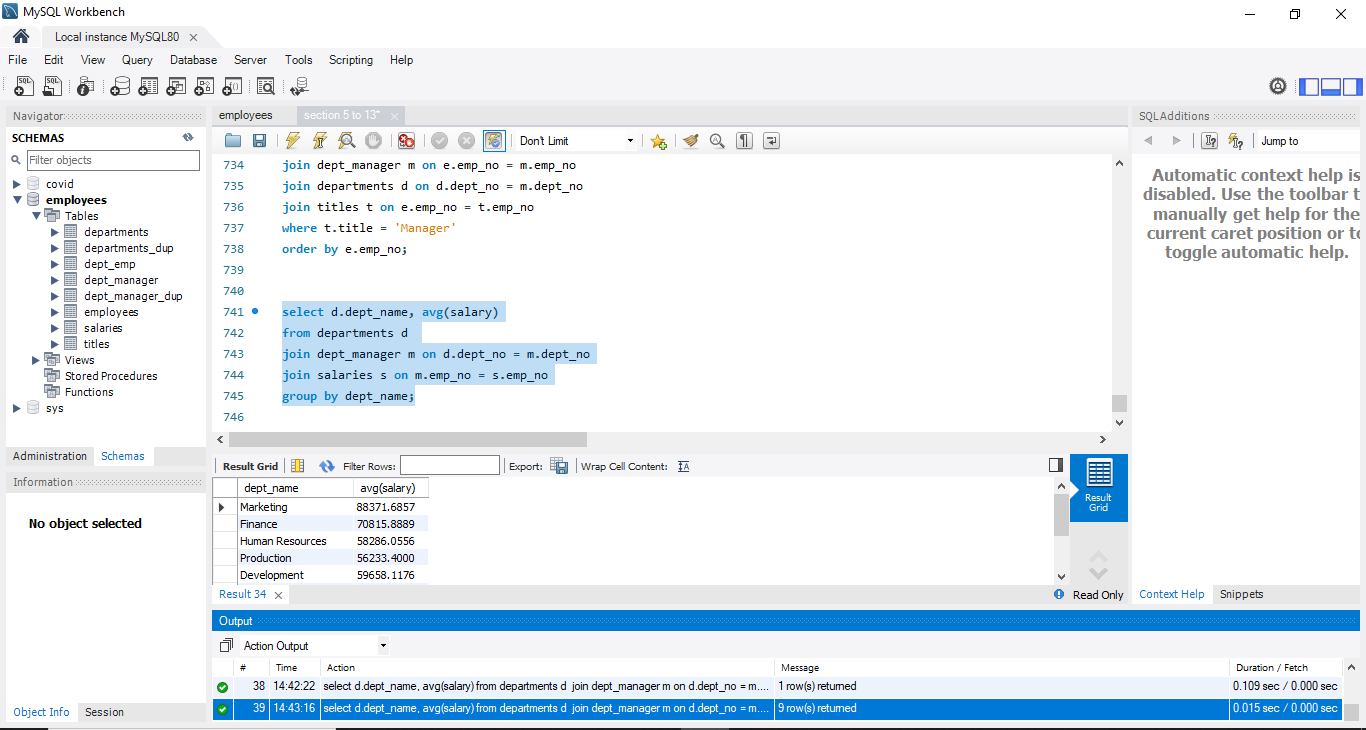
**Solution:**

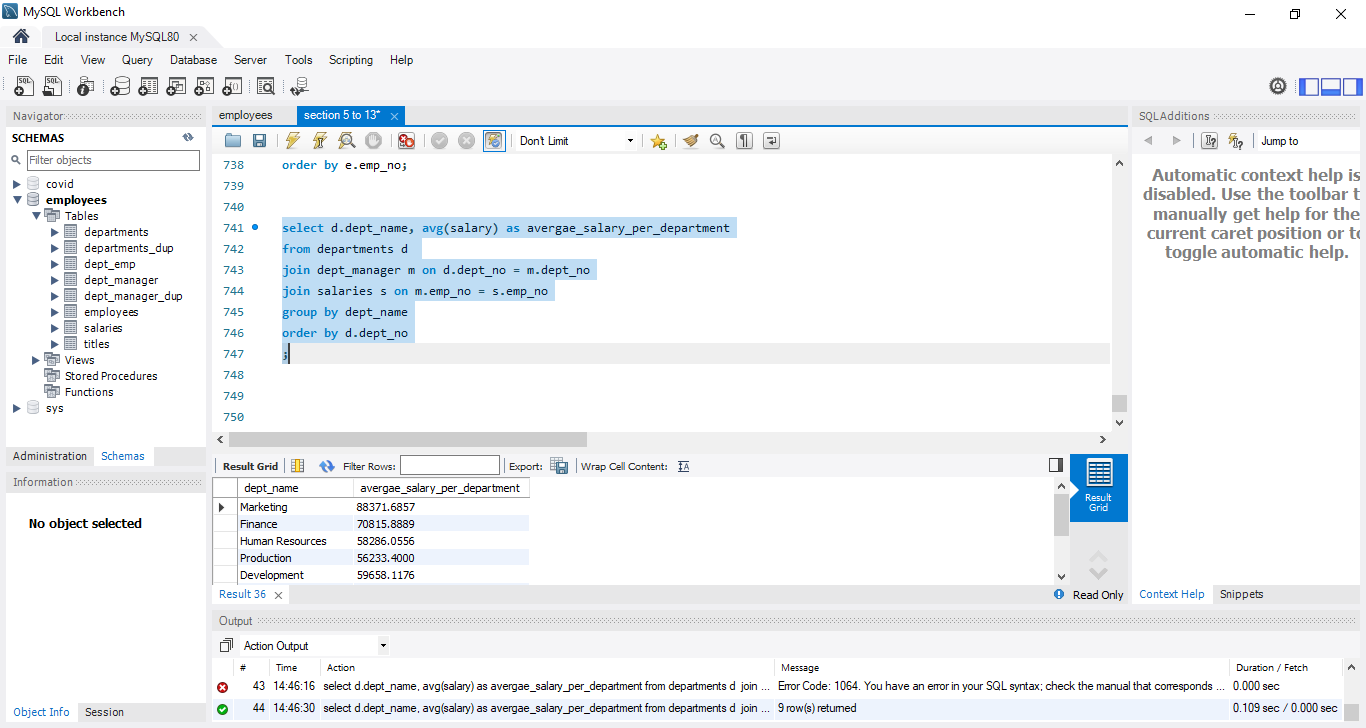


Now, to sum up let’s obtain the names of all departments and calculate the average salary paid to the managers in each department.

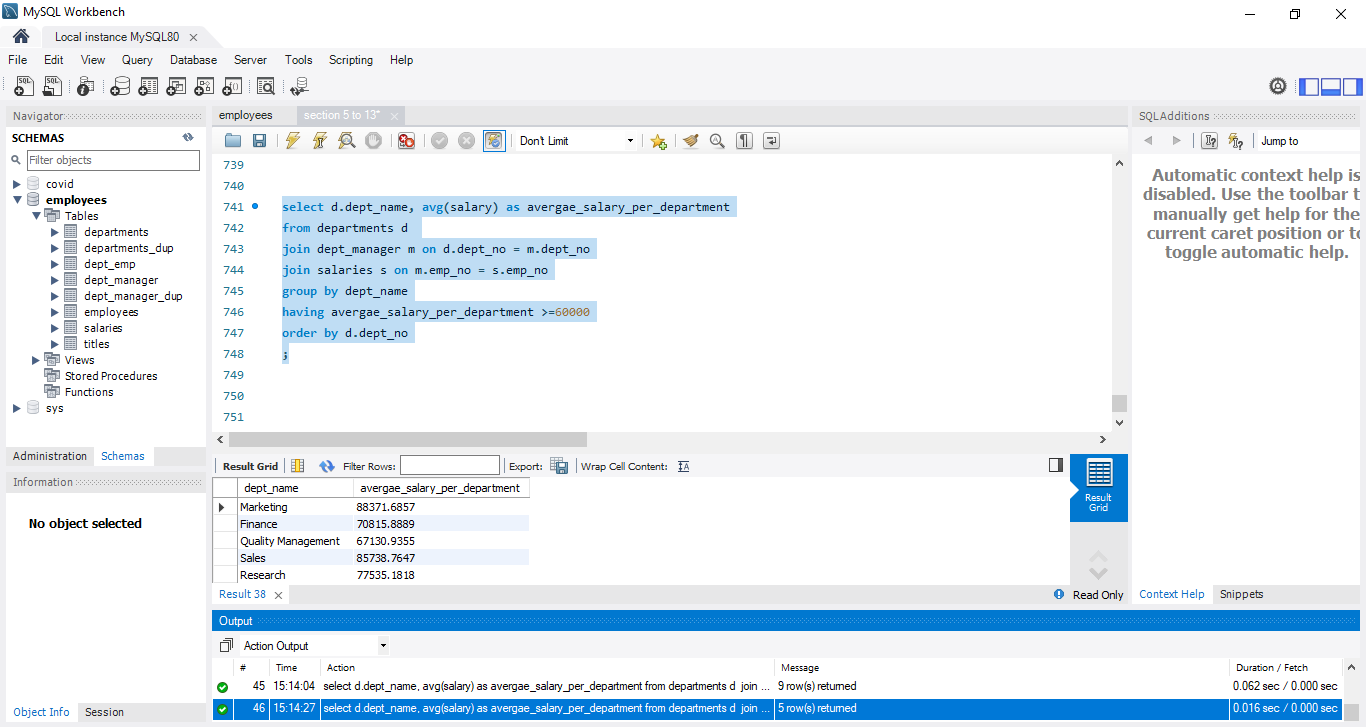
To solve this, we know we can get department names from departments table and salary from salaries. But they don’t connect directly in relational schema. So we will look for key columns (column doesn’t need to be private or foreign key).

Column: dept\_no is the key column that connects dept\_manager with departments table and hence, we can connect dept\_manager and salaries. The column dept\_no here is the linking column.





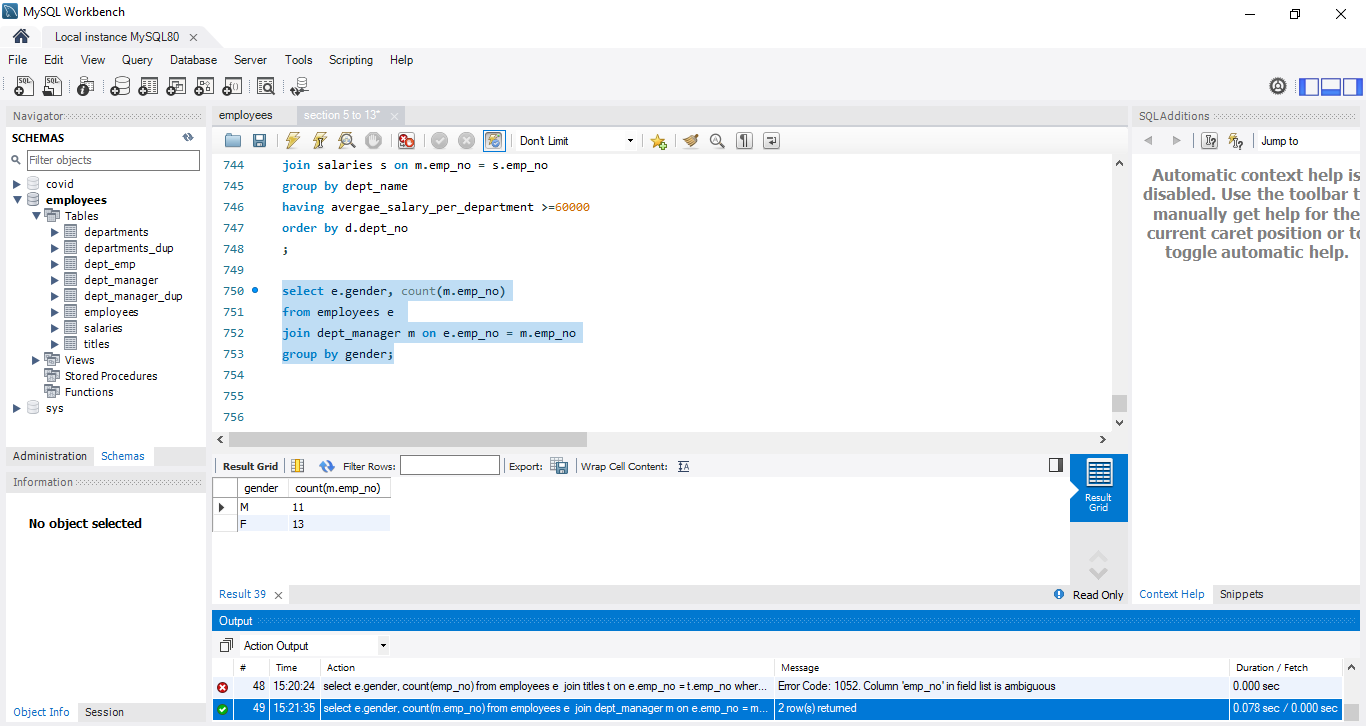
Now I wanted to know the departments whose average salaries is greater than $60,000. I will write the following query:

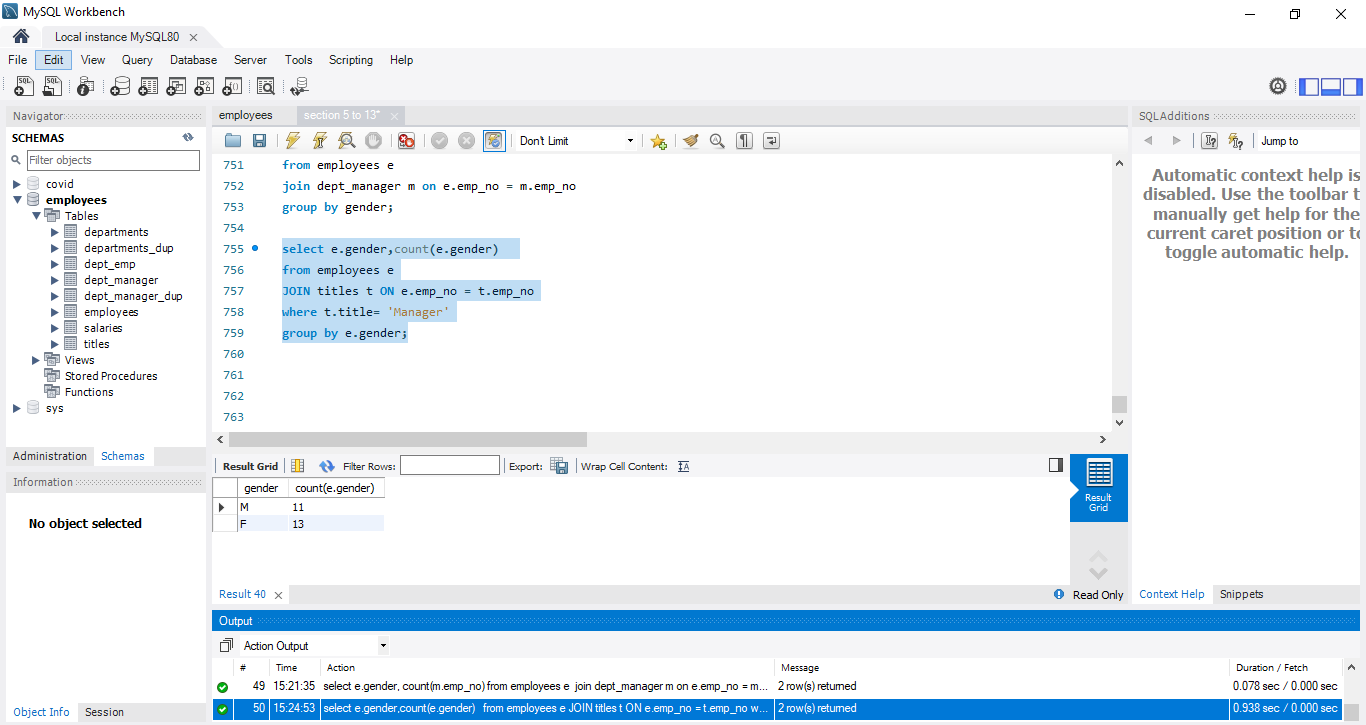


**Exercise:**

How many male and how many female managers do we have in the ‘employees’ database?

**Solution:**





Lastly, we want to unify two tables: employees\_dup and dept\_managers.