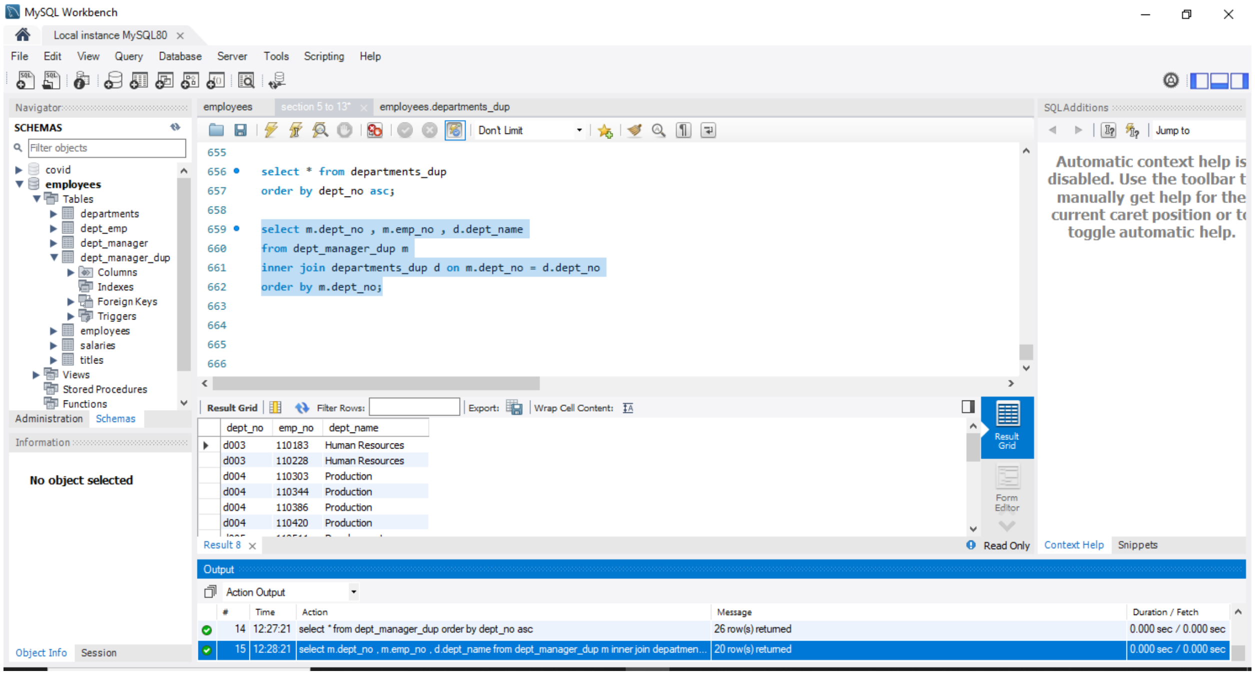
SECTION 14

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:



```sql
select * from departments_dup
order by dept_no asc;

select m.dept_no , m.emp_no , d.dept_name
from dept_manager_dup m
inner join departments_dup d on m.dept_no = d.dept_no
order by m.dept_no;
```

dept_no	emp_no	dept_name
d003	110183	Human Resources
d003	110228	Human Resources
d004	110303	Production
d004	110344	Production
d004	110386	Production
d004	110420	Production

```
#   Time      Action                                                                     Message             Duration / Fetch
14  12:27:21  select * from dept_manager_dup order by dept_no asc                        26 row(s) returned  0.000 sec / 0.000 sec
15  12:28:21  select m.dept_no , m.emp_no , d.dept_name from dept_manager_dup m inner join departmen...  20 row(s) returned  0.000 sec / 0.000 sec
```

Automatic context help is disabled. Use the toolbar t manually get help for the current caret position or t toggle automatic help.

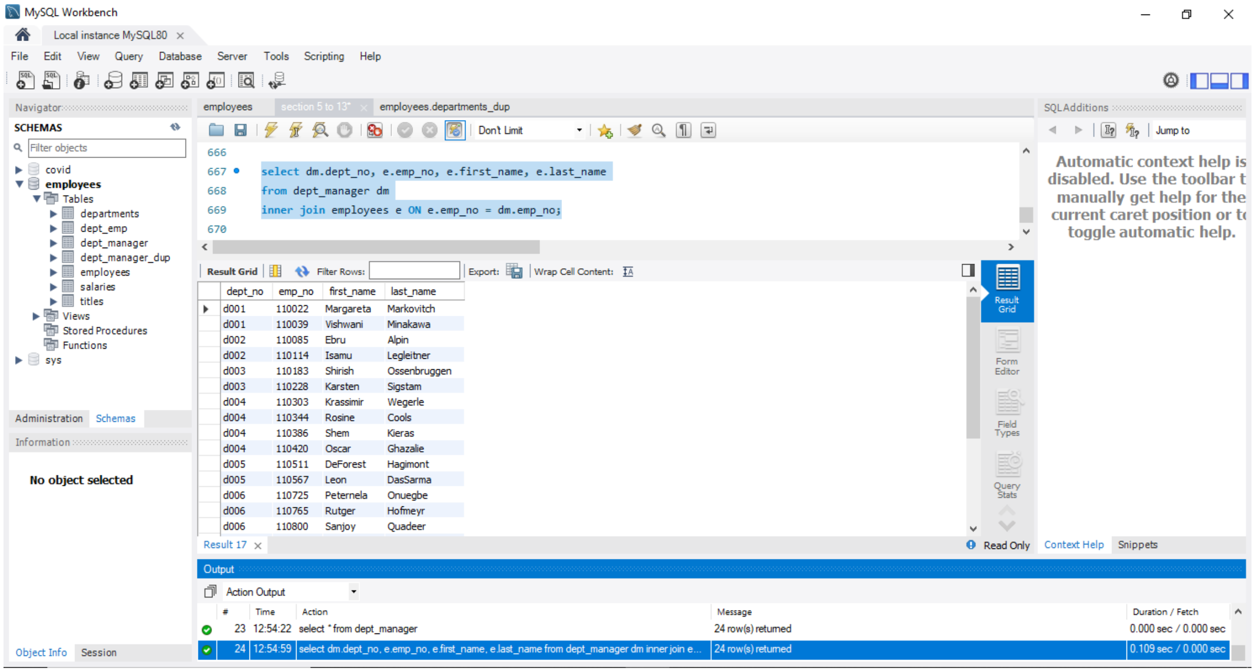
No object selected

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.



```
select dm.dept_no, e.emp_no, e.first_name, e.last_name
from dept_manager dm
inner join employees e ON e.emp_no = dm.emp_no;
```

dept_no	emp_no	first_name	last_name
d001	110022	Margareta	Markovitch
d001	110039	Vishwani	Minakawa
d002	110085	Ebru	Alpin
d002	110114	Isamu	Legleitner
d003	110183	Shirish	Ossenbruggen
d003	110228	Karsten	Sigstam
d004	110303	Krassimir	Wegerle
d004	110344	Rosine	Cools
d004	110386	Shem	Kieras
d004	110420	Oscar	Ghazalie
d005	110511	DeForest	Hagimont
d005	110567	Leon	DasSarma
d006	110725	Peternela	Onuegbe
d006	110765	Rutger	Hofmeyr
d006	110800	Sanjoy	Quadeer

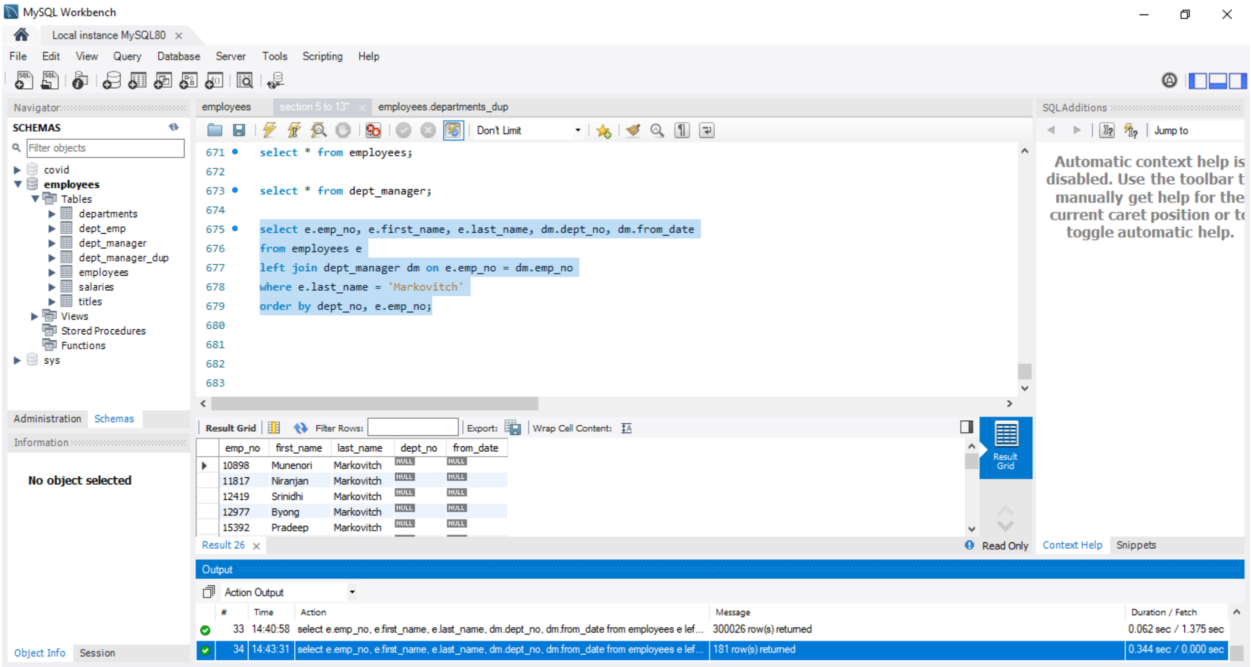
```
#   Time      Action                                                                 Message            Duration / Fetch
23  12:54:22  select * from dept_manager                                             24 row(s) returned 0.000 sec / 0.000 sec
24  12:54:59  select dm.dept_no, e.emp_no, e.first_name, e.last_name from dept_manager dm inner join e...  24 row(s) returned  0.109 sec / 0.000 sec
```

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:



```sql
select * from employees;

select * from dept_manager;

select e.emp_no, e.first_name, e.last_name, dm.dept_no, dm.from_date
from employees e
left join dept_manager dm on e.emp_no = dm.emp_no
where e.last_name = 'Markovitch'
order by dept_no, e.emp_no;
```

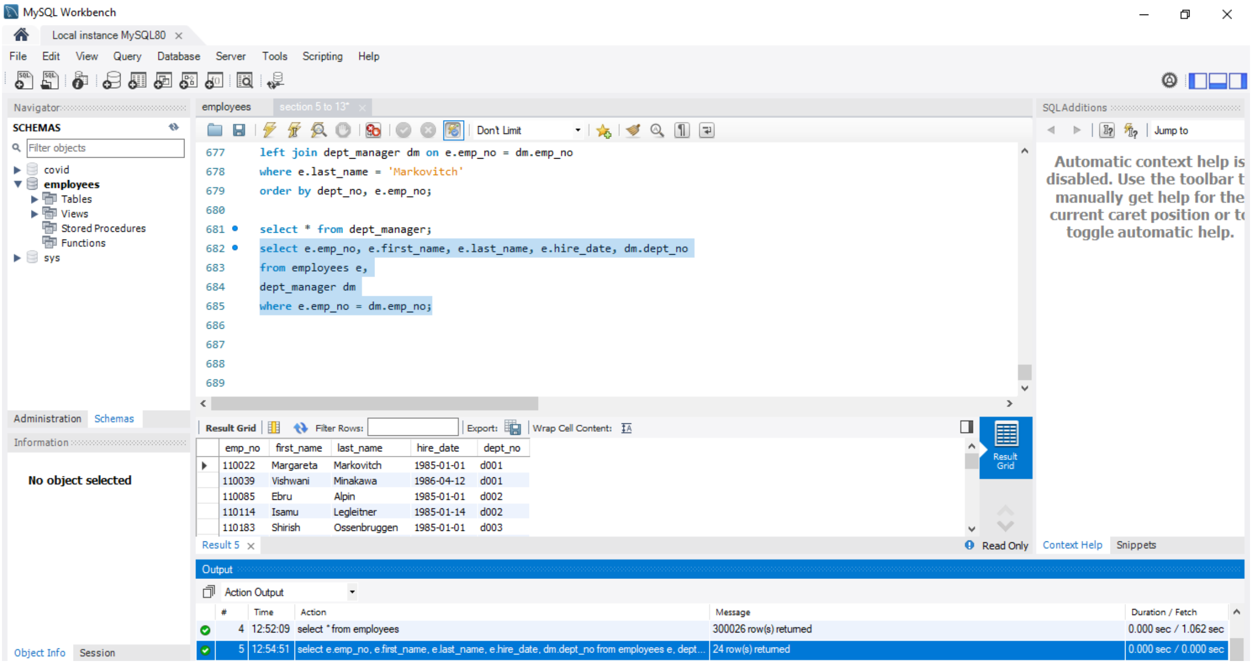
emp_no	first_name	last_name	dept_no	from_date
10898	Munenori	Markovitch	NULL	NULL
11817	Niranjan	Markovitch	NULL	NULL
12419	Srinidhi	Markovitch	NULL	NULL
12977	Byong	Markovitch	NULL	NULL
15392	Pradeep	Markovitch	NULL	NULL

```
#   Time      Action                                                                                  Message                   Duration / Fetch
33  14:40:58  select e.emp_no, e.first_name, e.last_name, dm.dept_no, dm.from_date from employees e lef...  300026 row(s) returned  0.062 sec / 1.375 sec
34  14:43:31  select e.emp_no, e.first_name, e.last_name, dm.dept_no, dm.from_date from employees e lef...  181 row(s) returned     0.344 sec / 0.000 sec
```

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:

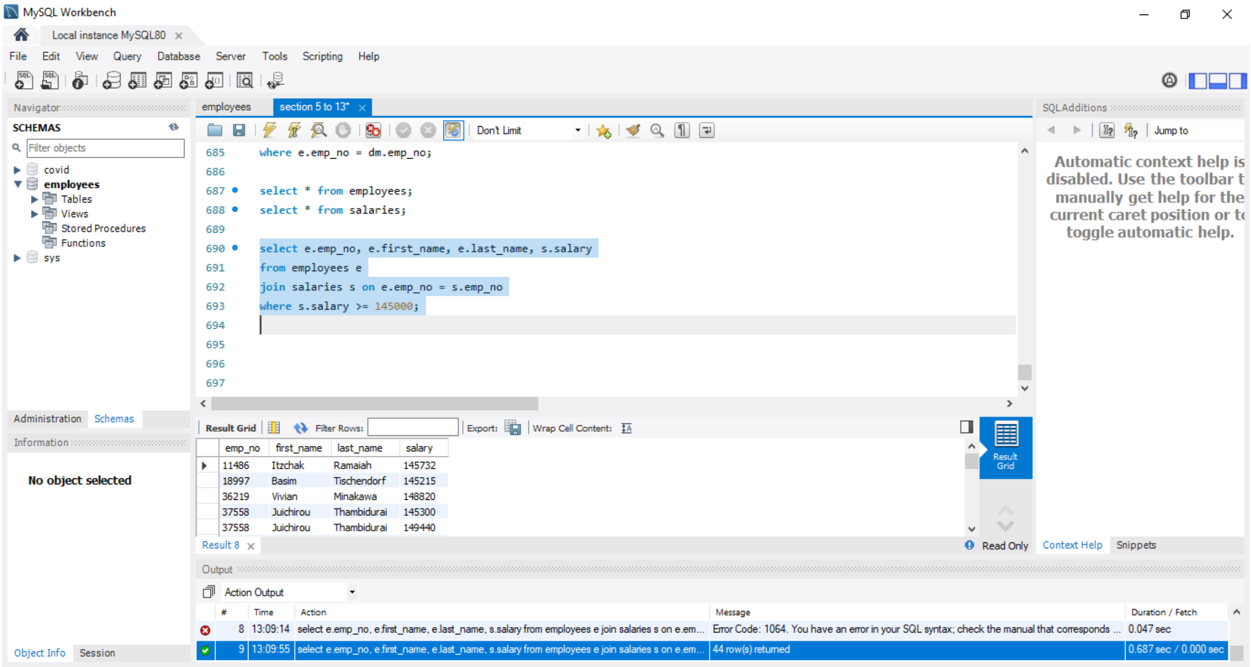


```sql
left join dept_manager dm on e.emp_no = dm.emp_no
where e.last_name = 'Markovitch'
order by dept_no, e.emp_no;

select * from dept_manager;
select e.emp_no, e.first_name, e.last_name, e.hire_date, dm.dept_no
from employees e,
dept_manager dm
where e.emp_no = dm.emp_no;
```

emp_no	first_name	last_name	hire_date	dept_no
110022	Margareta	Markovitch	1985-01-01	d001
110039	Vishwani	Minakawa	1986-04-12	d001
110085	Ebru	Alpin	1985-01-01	d002
110114	Isamu	Legleitner	1985-01-14	d002
110183	Shirish	Ossenbruggen	1985-01-01	d003

```
#  Time      Action                                                                                     Message                   Duration / Fetch
4  12:52:09  select * from employees                                                                    300026 row(s) returned  0.000 sec / 1.062 sec
5  12:54:51  select e.emp_no, e.first_name, e.last_name, e.hire_date, dm.dept_no from employees e, dept...  24 row(s) returned      0.000 sec / 0.000 sec
```

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.



```
where e.emp_no = dm.emp_no;

select * from employees;
select * from salaries;

select e.emp_no, e.first_name, e.last_name, s.salary
from employees e
join salaries s on e.emp_no = s.emp_no
where s.salary >= 145000;
```

emp_no	first_name	last_name	salary
11486	Itzchak	Ramaiah	145732
18997	Basim	Tischendorf	145215
36219	Vivian	Minakawa	148820
37558	Juichirou	Thambidurai	145300
37558	Juichirou	Thambidurai	149440

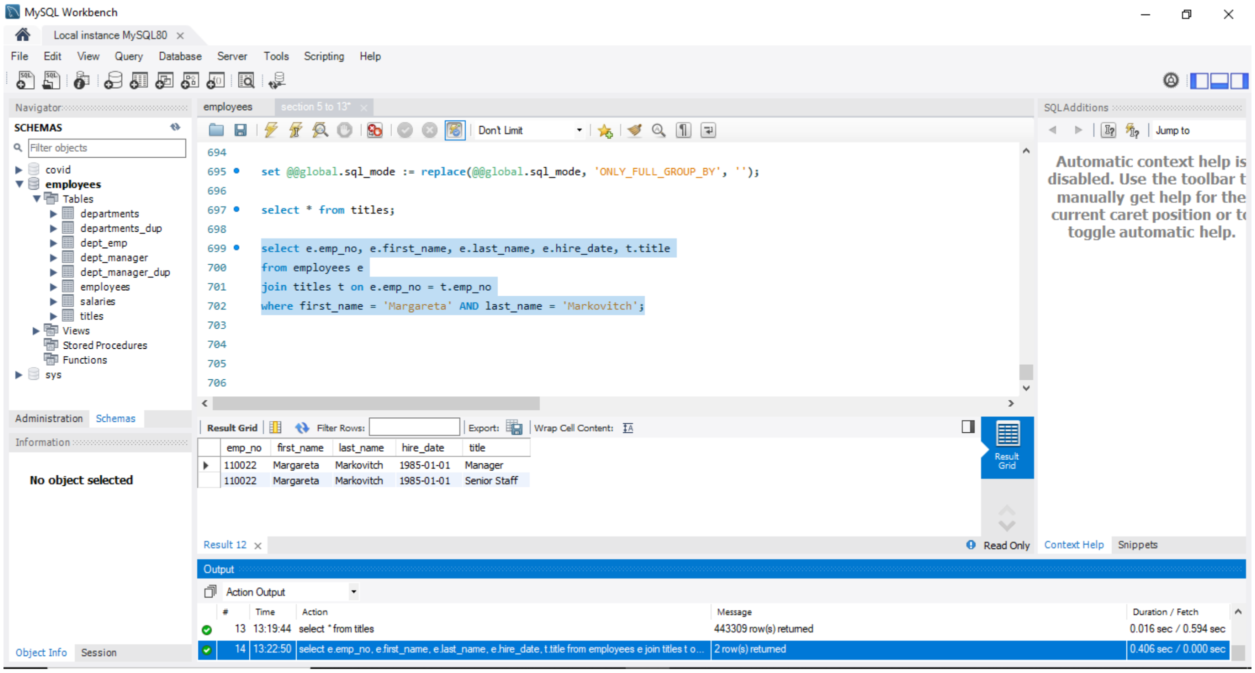
Automatic context help is disabled. Use the toolbar t manually get help for the current caret position or t toggle automatic help.

#	Time	Action	Message	Duration / Fetch
8	13:09:14	select e.emp_no, e.first_name, e.last_name, s.salary from employees e join salaries s on e.em...	Error Code: 1064. You have an error in your SQL syntax; check the manual that corresponds ...	0.047 sec
9	13:09:55	select e.emp_no, e.first_name, e.last_name, s.salary from employees e join salaries s on e.em...	44 row(s) returned	0.687 sec / 0.000 sec

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:



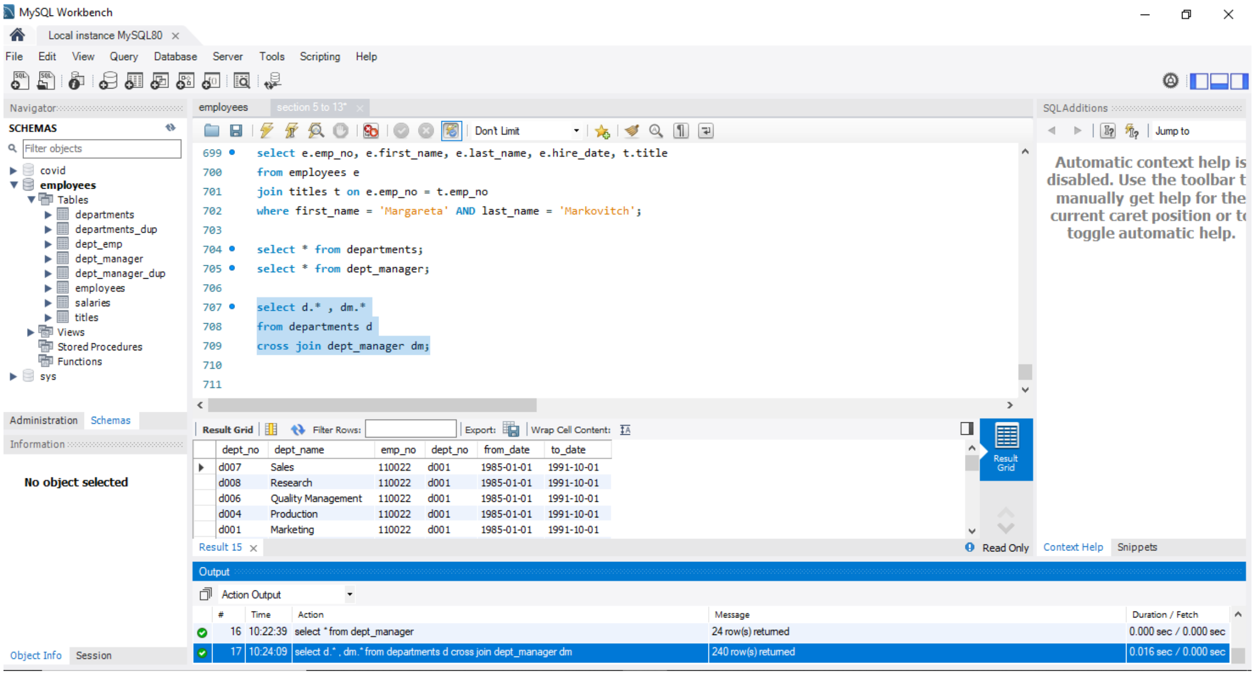
```sql
set @@global.sql_mode := replace(@@global.sql_mode, 'ONLY_FULL_GROUP_BY', '');

select * from titles;

select e.emp_no, e.first_name, e.last_name, e.hire_date, t.title
from employees e
join titles t on e.emp_no = t.emp_no
where first_name = 'Margareta' AND last_name = 'Markovitch';
```

emp_no	first_name	last_name	hire_date	title
110022	Margareta	Markovitch	1985-01-01	Manager
110022	Margareta	Markovitch	1985-01-01	Senior Staff

```
#   Time      Action                                                                        Message                Duration / Fetch
13  13:19:44  select * from titles                                                          443309 row(s) returned 0.016 sec / 0.594 sec
14  13:22:50  select e.emp_no, e.first_name, e.last_name, e.hire_date, t.title from employees e join titles t o...  2 row(s) returned  0.406 sec / 0.000 sec
```

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.



```sql
select e.emp_no, e.first_name, e.last_name, e.hire_date, t.title
from employees e
join titles t on e.emp_no = t.emp_no
where first_name = 'Margareta' AND last_name = 'Markovitch';

select * from departments;
select * from dept_manager;

select d.* , dm.*
from departments d
cross join dept_manager dm;
```

dept_no	dept_name	emp_no	dept_no	from_date	to_date
d007	Sales	110022	d001	1985-01-01	1991-10-01
d008	Research	110022	d001	1985-01-01	1991-10-01
d006	Quality Management	110022	d001	1985-01-01	1991-10-01
d004	Production	110022	d001	1985-01-01	1991-10-01
d001	Marketing	110022	d001	1985-01-01	1991-10-01

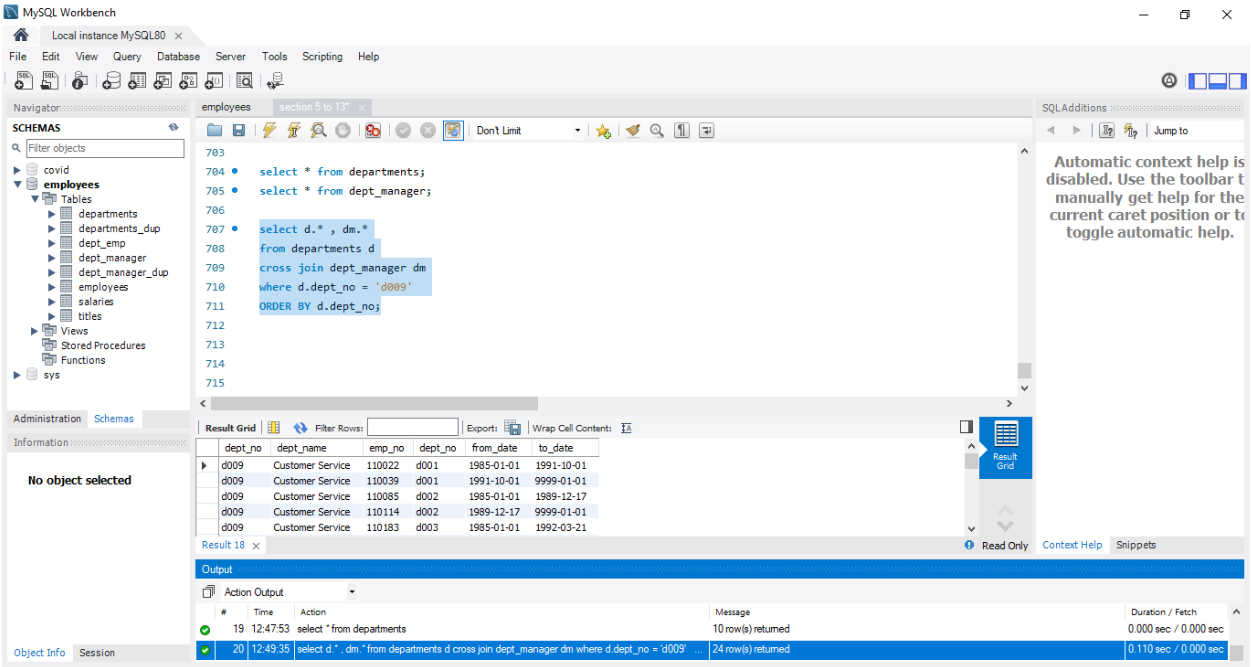
```
#   Time      Action                                                     Message                Duration / Fetch
16  10:22:39  select * from dept_manager                                 24 row(s) returned     0.000 sec / 0.000 sec
17  10:24:09  select d.* , dm.* from departments d cross join dept_manager dm  240 row(s) returned  0.016 sec / 0.000 sec
```

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:



```
select * from departments;
select * from dept_manager;

select d.* , dm.*
from departments d
cross join dept_manager dm
where d.dept_no = 'd009'
ORDER BY d.dept_no;
```

Automatic context help is disabled. Use the toolbar t manually get help for the current caret position or t toggle automatic help.

dept_no	dept_name	emp_no	dept_no	from_date	to_date
d009	Customer Service	110022	d001	1985-01-01	1991-10-01
d009	Customer Service	110039	d001	1991-10-01	9999-01-01
d009	Customer Service	110085	d002	1985-01-01	1989-12-17
d009	Customer Service	110114	d002	1989-12-17	9999-01-01
d009	Customer Service	110183	d003	1985-01-01	1992-03-21

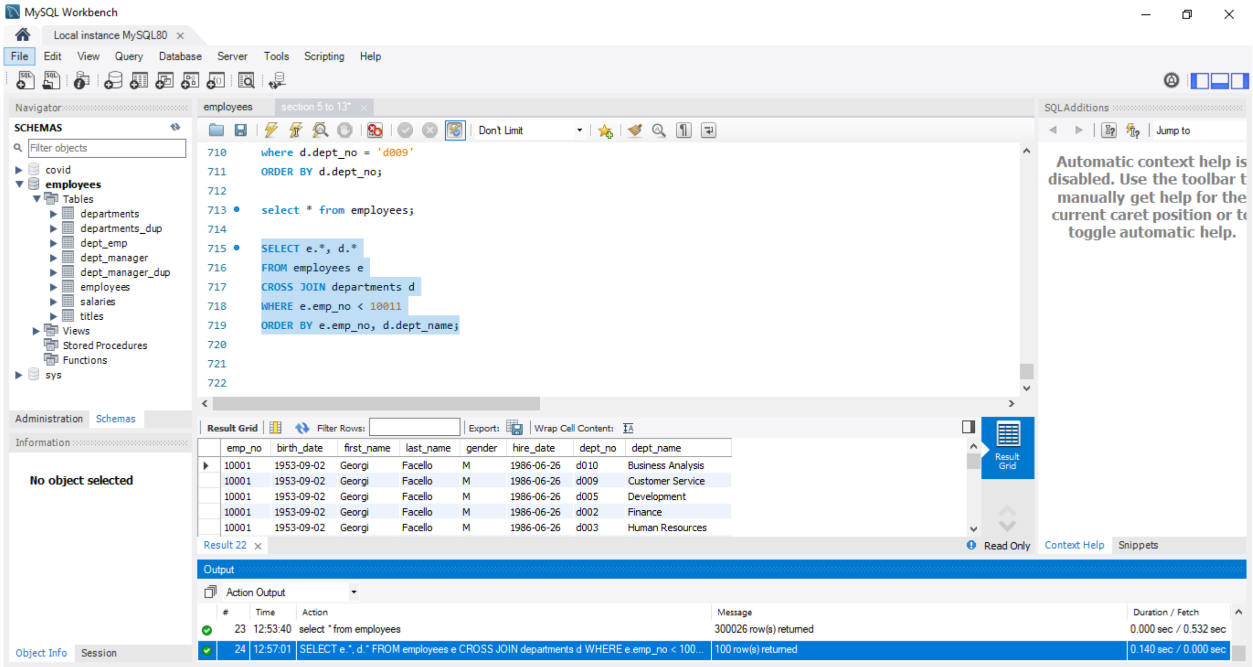
```
#   Time      Action                                                                          Message             Duration / Fetch
19  12:47:53  select * from departments                                                       10 row(s) returned  0.000 sec / 0.000 sec
20  12:49:35  select d.* , dm.* from departments d cross join dept_manager dm where d.dept_no = 'd009'  ...  24 row(s) returned  0.110 sec / 0.000 sec
```

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:

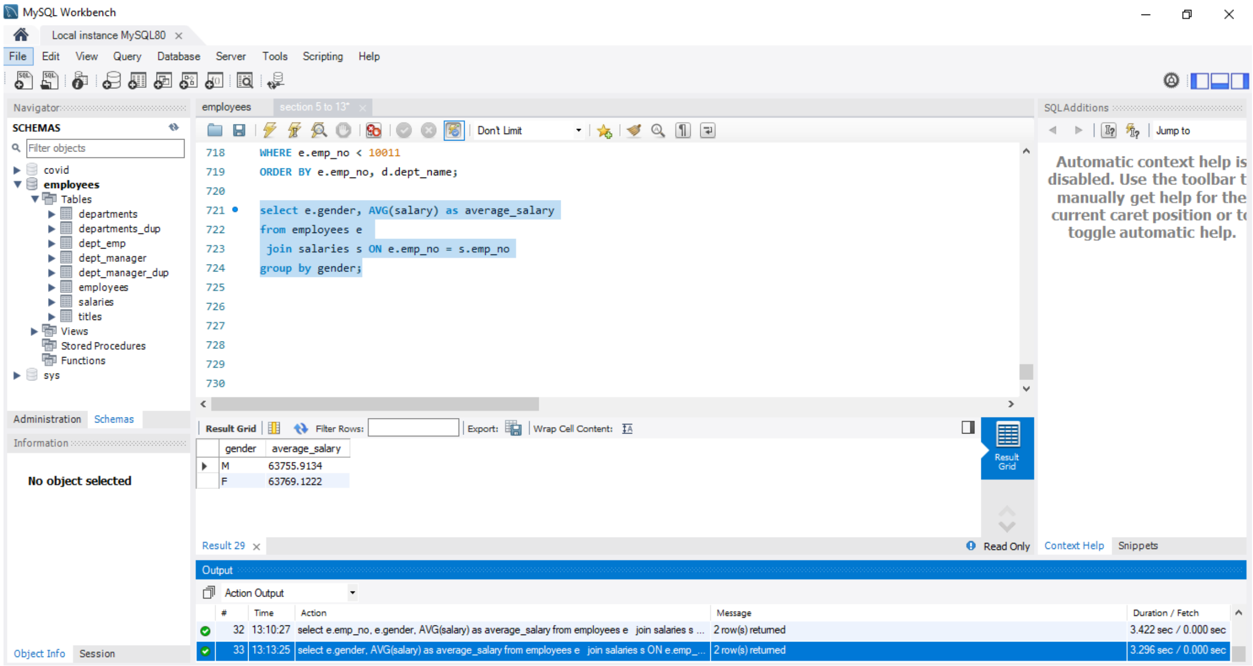


```sql
where d.dept_no = 'd009'
ORDER BY d.dept_no;

select * from employees;

SELECT e.*, d.*
FROM employees e
CROSS JOIN departments d
WHERE e.emp_no < 10011
ORDER BY e.emp_no, d.dept_name;
```

emp_no	birth_date	first_name	last_name	gender	hire_date	dept_no	dept_name
10001	1953-09-02	Georgi	Facello	M	1986-06-26	d010	Business Analysis
10001	1953-09-02	Georgi	Facello	M	1986-06-26	d009	Customer Service
10001	1953-09-02	Georgi	Facello	M	1986-06-26	d005	Development
10001	1953-09-02	Georgi	Facello	M	1986-06-26	d002	Finance
10001	1953-09-02	Georgi	Facello	M	1986-06-26	d003	Human Resources

```
#   Time      Action                                                                        Message                  Duration / Fetch
23  12:53:40  select * from employees                                                       300026 row(s) returned   0.000 sec / 0.532 sec
24  12:57:01  SELECT e.*, d.* FROM employees e CROSS JOIN departments d WHERE e.emp_no < 100...  100 row(s) returned  0.140 sec / 0.000 sec
```

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:



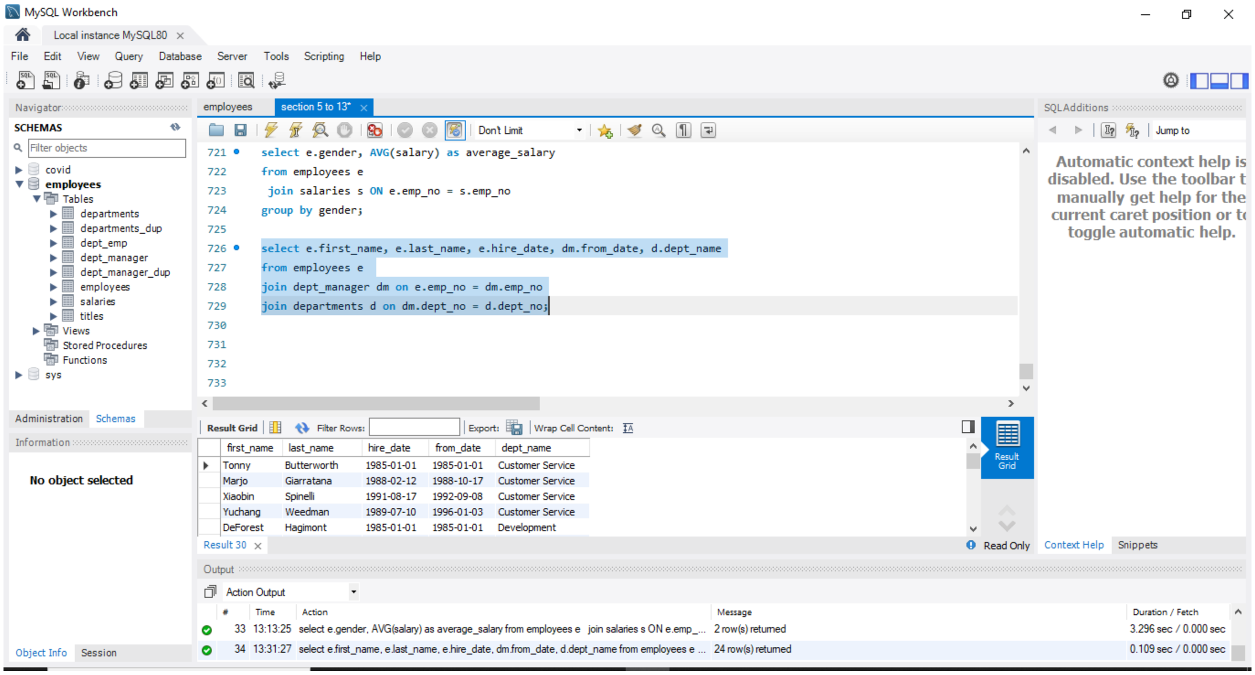
```sql
WHERE e.emp_no < 10011
ORDER BY e.emp_no, d.dept_name;

select e.gender, AVG(salary) as average_salary
from employees e
 join salaries s ON e.emp_no = s.emp_no
group by gender;
```

gender	average_salary
M	63755.9134
F	63769.1222

```
#   Time      Action                                                                            Message                Duration / Fetch
32  13:10:27  select e.emp_no, e.gender, AVG(salary) as average_salary from employees e  join salaries s ...  2 row(s) returned  3.422 sec / 0.000 sec
33  13:13:25  select e.gender, AVG(salary) as average_salary from employees e  join salaries s ON e.emp_...   2 row(s) returned  3.296 sec / 0.000 sec
```

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)



```sql
select e.gender, AVG(salary) as average_salary
from employees e
 join salaries s ON e.emp_no = s.emp_no
group by gender;

select e.first_name, e.last_name, e.hire_date, dm.from_date, d.dept_name
from employees e
join dept_manager dm on e.emp_no = dm.emp_no
join departments d on dm.dept_no = d.dept_no;
```

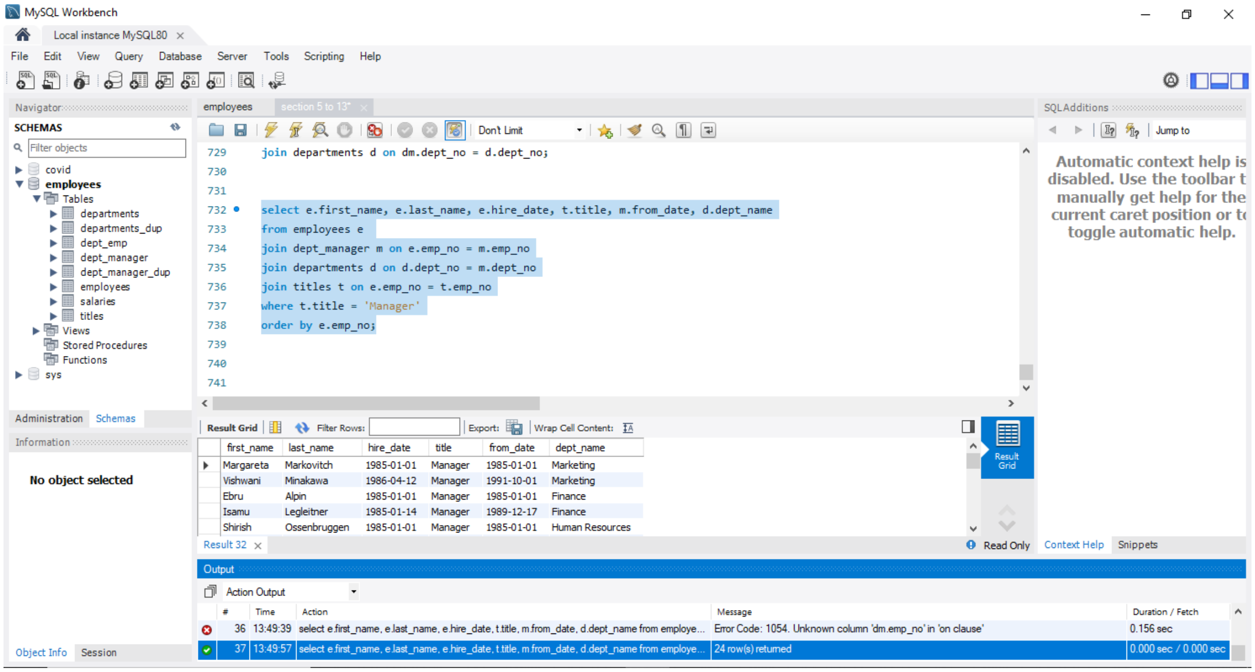
first_name	last_name	hire_date	from_date	dept_name
Tonny	Butterworth	1985-01-01	1985-01-01	Customer Service
Marjo	Giarratana	1988-02-12	1988-10-17	Customer Service
Xiaobin	Spinelli	1991-08-17	1992-09-08	Customer Service
Yuchang	Weedman	1989-07-10	1996-01-03	Customer Service
DeForest	Hagimont	1985-01-01	1985-01-01	Development

```
#   Time      Action                                                                                  Message             Duration / Fetch
33  13:13:25  select e.gender, AVG(salary) as average_salary from employees e   join salaries s ON e.emp_...   2 row(s) returned   3.296 sec / 0.000 sec
34  13:31:27  select e.first_name, e.last_name, e.hire_date, dm.from_date, d.dept_name from employees e ...   24 row(s) returned   0.109 sec / 0.000 sec
```

Exercise:

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

Solution:



```sql
join departments d on dm.dept_no = d.dept_no;


select e.first_name, e.last_name, e.hire_date, t.title, m.from_date, d.dept_name
from employees e
join dept_manager m on e.emp_no = m.emp_no
join departments d on d.dept_no = m.dept_no
join titles t on e.emp_no = t.emp_no
where t.title = 'Manager'
order by e.emp_no;
```

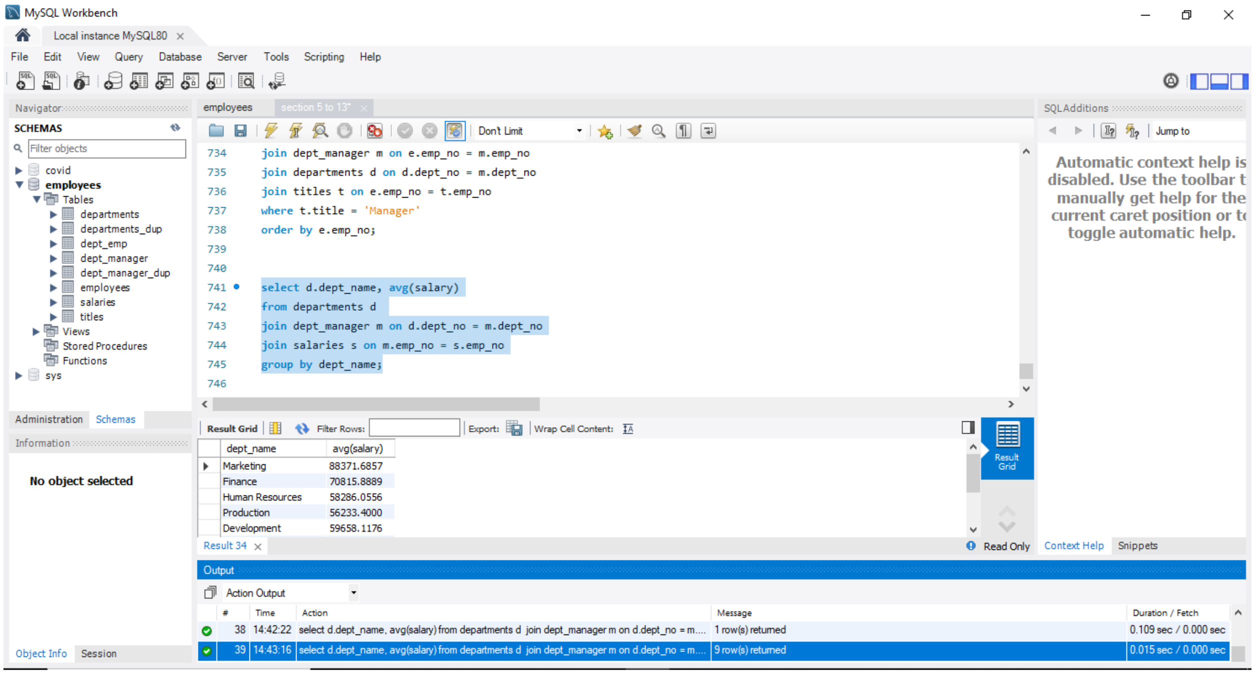
first_name	last_name	hire_date	title	from_date	dept_name
Margareta	Markovitch	1985-01-01	Manager	1985-01-01	Marketing
Vishwani	Minakawa	1986-04-12	Manager	1991-10-01	Marketing
Ebru	Alpin	1985-01-01	Manager	1985-01-01	Finance
Isamu	Legleitner	1985-01-14	Manager	1989-12-17	Finance
Shirish	Ossenbruggen	1985-01-01	Manager	1985-01-01	Human Resources

```
#   Time      Action                                                                                  Message                                                      Duration / Fetch
36  13:49:39  select e.first_name, e.last_name, e.hire_date, t.title, m.from_date, d.dept_name from employe...   Error Code: 1054. Unknown column 'dm.emp_no' in 'on clause'   0.156 sec
37  13:49:57  select e.first_name, e.last_name, e.hire_date, t.title, m.from_date, d.dept_name from employe...   24 row(s) returned                                           0.000 sec / 0.000 sec
```

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).

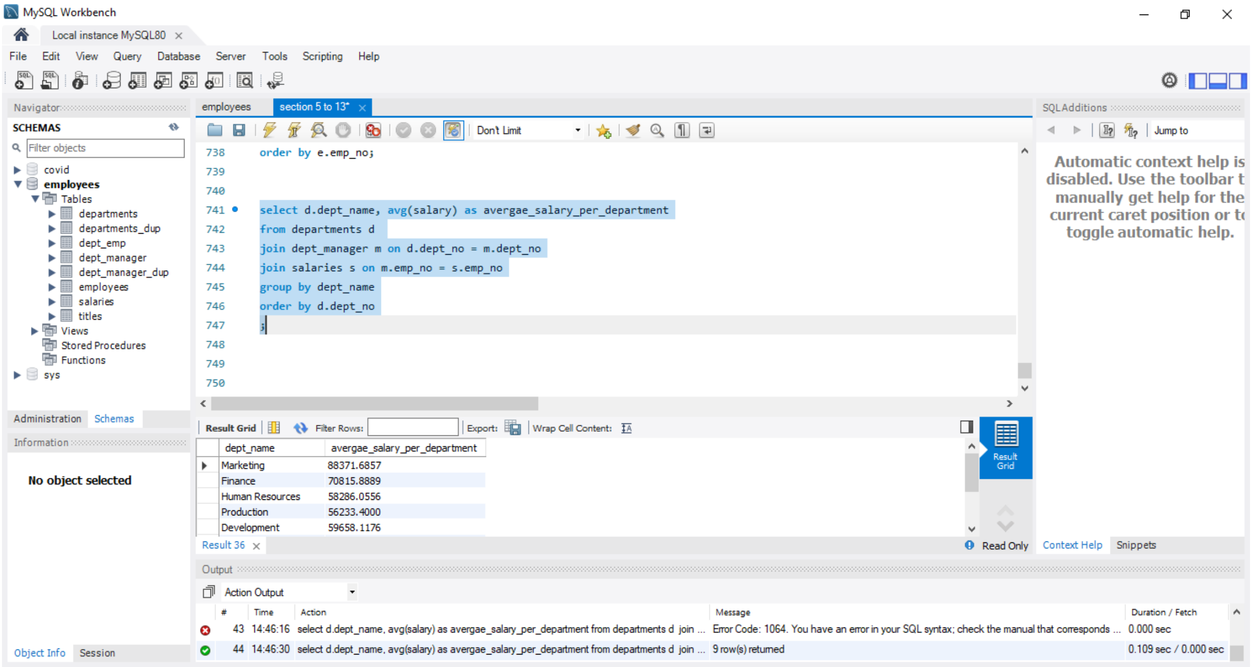
Column: dept_no is the key column that connects dept_manager with departments table and hence, we can connect dept_manager and salaries. The column dept_no here is the linking column.



```
join dept_manager m on e.emp_no = m.emp_no
join departments d on d.dept_no = m.dept_no
join titles t on e.emp_no = t.emp_no
where t.title = 'Manager'
order by e.emp_no;


select d.dept_name, avg(salary)
from departments d
join dept_manager m on d.dept_no = m.dept_no
join salaries s on m.emp_no = s.emp_no
group by dept_name;
```

dept_name	avg(salary)
Marketing	88371.6857
Finance	70815.8889
Human Resources	58286.0556
Production	56233.4000
Development	59658.1176

```
#   Time      Action                                                                            Message             Duration / Fetch
38  14:42:22  select d.dept_name, avg(salary) from departments d  join dept_manager m on d.dept_no = m....  1 row(s) returned   0.109 sec / 0.000 sec
39  14:43:16  select d.dept_name, avg(salary) from departments d  join dept_manager m on d.dept_no = m....  9 row(s) returned   0.015 sec / 0.000 sec
```

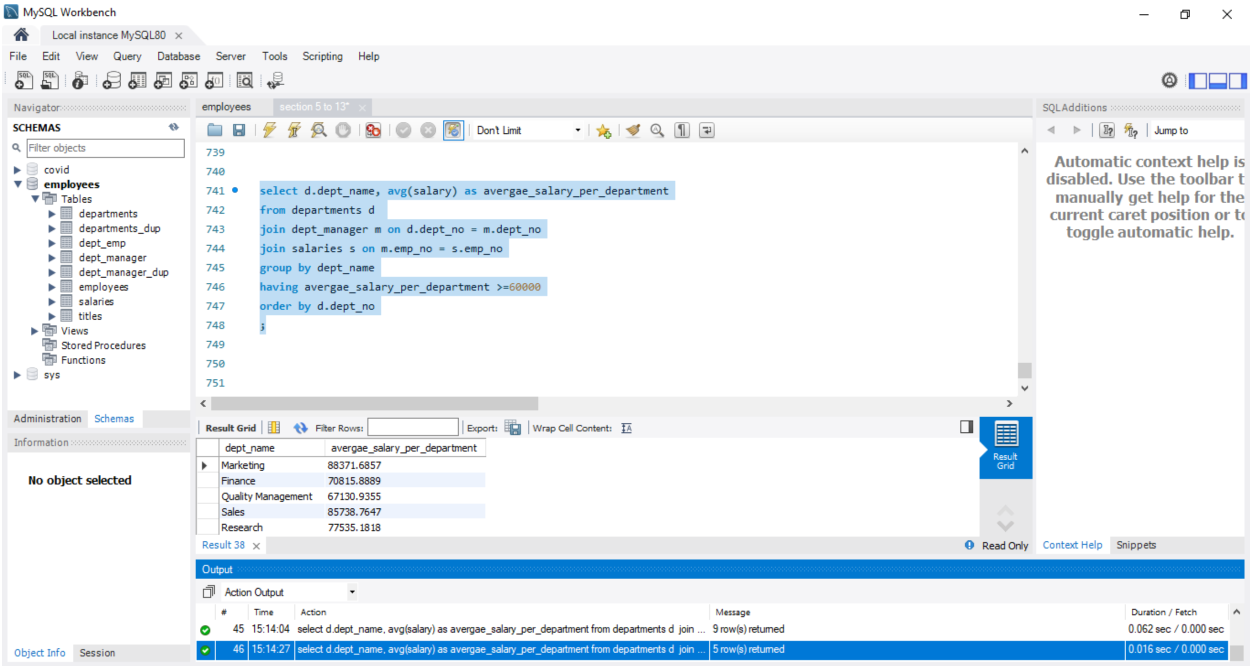


```
order by e.emp_no;


select d.dept_name, avg(salary) as avergae_salary_per_department
from departments d
join dept_manager m on d.dept_no = m.dept_no
join salaries s on m.emp_no = s.emp_no
group by dept_name
order by d.dept_no
;
```

dept_name	avergae_salary_per_department
Marketing	88371.6857
Finance	70815.8889
Human Resources	58286.0556
Production	56233.4000
Development	59658.1176

```
#   Time      Action                                                                              Message                                                                          Duration / Fetch
43  14:46:16  select d.dept_name, avg(salary) as avergae_salary_per_department from departments d  join ...  Error Code: 1064. You have an error in your SQL syntax; check the manual that corresponds ...  0.000 sec
44  14:46:30  select d.dept_name, avg(salary) as avergae_salary_per_department from departments d  join ...  9 row(s) returned                                                                0.109 sec / 0.000 sec
```

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



```sql
select d.dept_name, avg(salary) as avergae_salary_per_department
from departments d
join dept_manager m on d.dept_no = m.dept_no
join salaries s on m.emp_no = s.emp_no
group by dept_name
having avergae_salary_per_department >=60000
order by d.dept_no
;
```

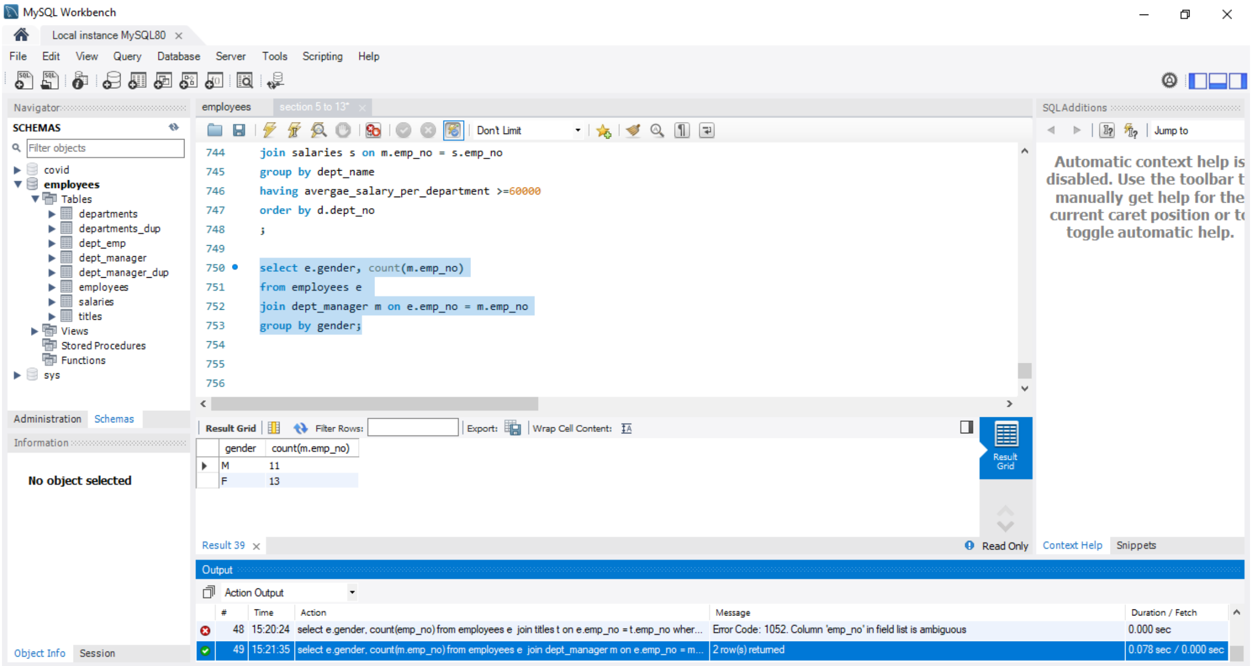
dept_name	avergae_salary_per_department
Marketing	88371.6857
Finance	70815.8889
Quality Management	67130.9355
Sales	85738.7647
Research	77535.1818

```
#   Time      Action                                                                                Message             Duration / Fetch
45  15:14:04  select d.dept_name, avg(salary) as avergae_salary_per_department from departments d join ...  9 row(s) returned  0.062 sec / 0.000 sec
46  15:14:27  select d.dept_name, avg(salary) as avergae_salary_per_department from departments d join ...  5 row(s) returned  0.016 sec / 0.000 sec
```

Exercise:

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

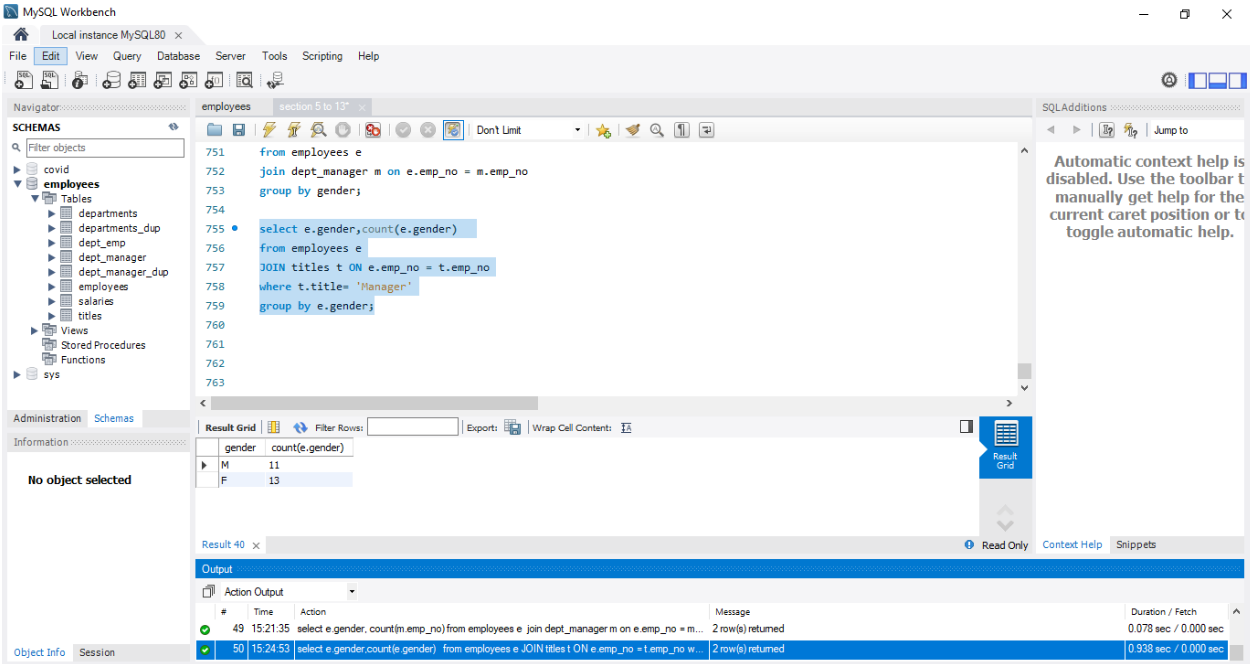
Solution:



```sql
join salaries s on m.emp_no = s.emp_no
group by dept_name
having avergae_salary_per_department >=60000
order by d.dept_no
;

select e.gender, count(m.emp_no)
from employees e
join dept_manager m on e.emp_no = m.emp_no
group by gender;
```

gender	count(m.emp_no)
M	11
F	13

```
#   Time      Action                                                                          Message                                                   Duration / Fetch
48  15:20:24  select e.gender, count(emp_no) from employees e  join titles t on e.emp_no = t.emp_no wher...  Error Code: 1052. Column 'emp_no' in field list is ambiguous  0.000 sec
49  15:21:35  select e.gender, count(m.emp_no) from employees e  join dept_manager m on e.emp_no = m...      2 row(s) returned                                          0.078 sec / 0.000 sec
```



```
from employees e
join dept_manager m on e.emp_no = m.emp_no
group by gender;

select e.gender,count(e.gender)
from employees e
JOIN titles t ON e.emp_no = t.emp_no
where t.title= 'Manager'
group by e.gender;
```

No object selected

Automatic context help is disabled. Use the toolbar t manually get help for the current caret position or t toggle automatic help.

	gender	count(e.gender)
▶	M	11
	F	13

	#	Time	Action	Message	Duration / Fetch
✓	49	15:21:35	select e.gender, count(m.emp_no) from employees e join dept_manager m on e.emp_no = m...	2 row(s) returned	0.078 sec / 0.000 sec
✓	50	15:24:53	select e.gender,count(e.gender) from employees e JOIN titles t ON e.emp_no = t.emp_no w...	2 row(s) returned	0.938 sec / 0.000 sec