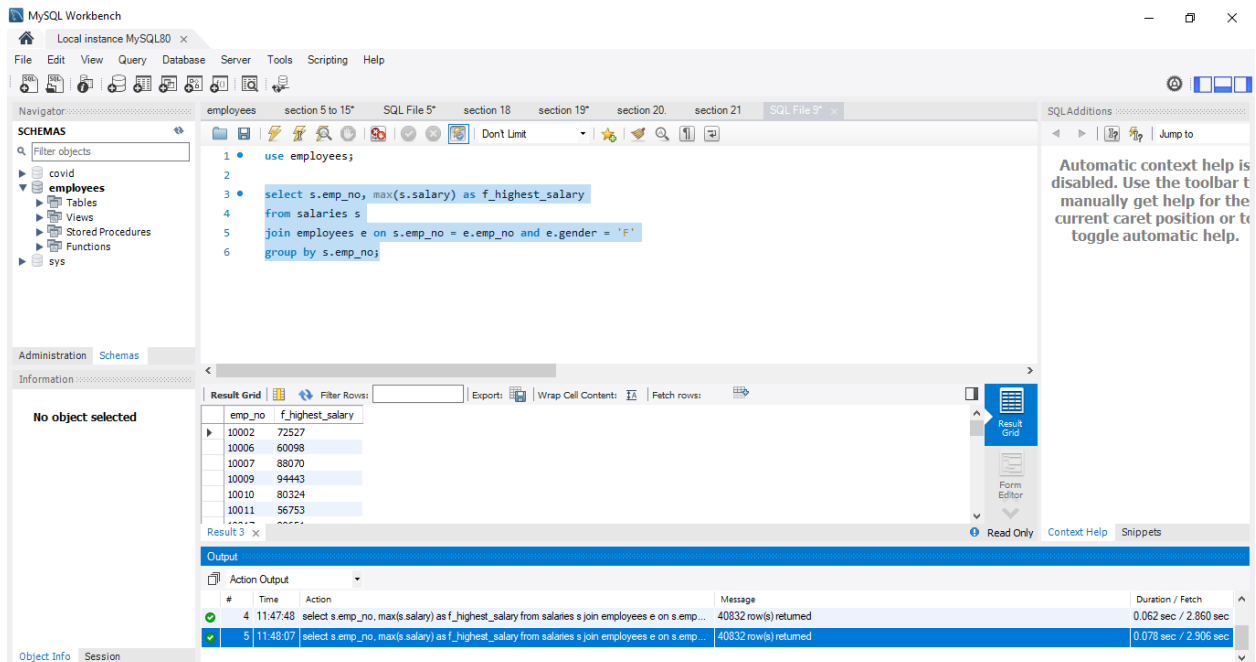


## SECTION 22

In this lesson, we covered MySQL temporary tables.

Our task is: Obtain a list containing highest salary contract values signed by all female employees who have worked in a company.

To do this:



The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 use employees;
2
3 select s.emp_no, max(s.salary) as f_highest_salary
4 from salaries s
5 join employees e on s.emp_no = e.emp_no and e.gender = 'F'
6 group by s.emp_no;
```

The Results pane shows the output of the query:

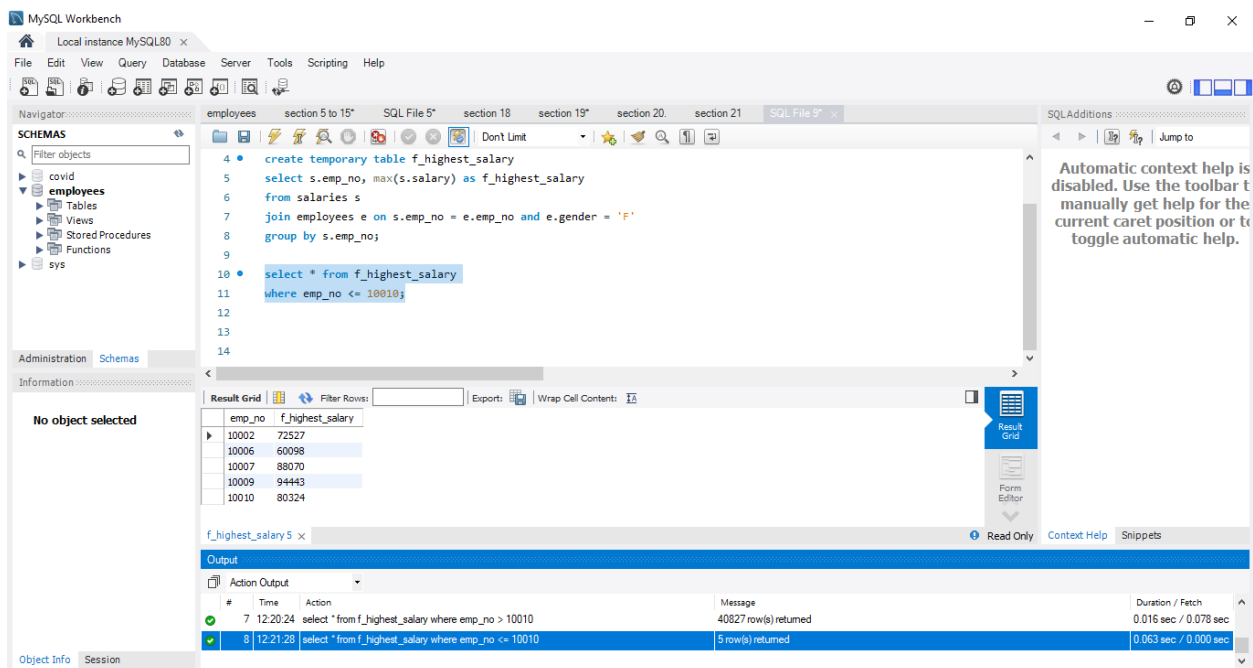
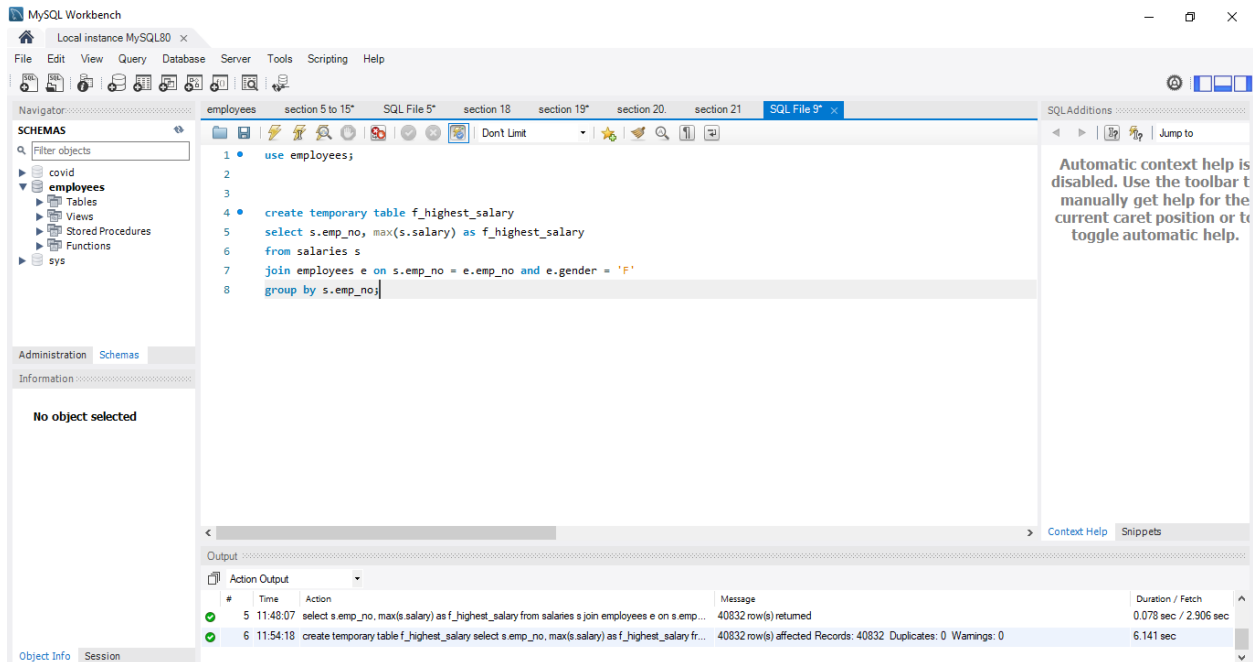
emp_no	f_highest_salary
10002	72527
10006	60098
10007	88070
10009	94443
10010	80324
10011	56753

The Output pane shows the execution details:

#	Time	Action	Message	Duration / Fetch
4	11:47:48	select s.emp_no, max(s.salary) as f_highest_salary from salaries s join employees e on s.emp...	40832 row(s) returned	0.062 sec / 2.860 sec
5	11:48:07	select s.emp_no, max(s.salary) as f_highest_salary from salaries s join employees e on s.emp...	40832 row(s) returned	0.078 sec / 2.906 sec

We got 40,832 rows.

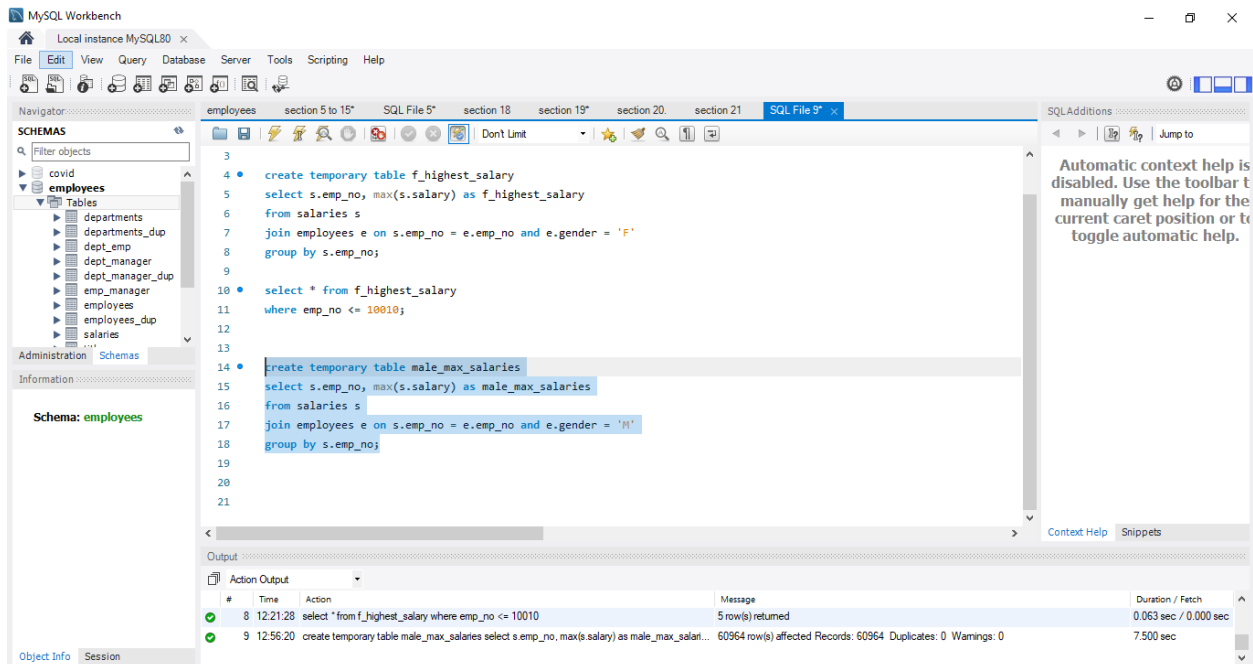
Imagine you want to refer to this list later in your session or to do any other work. In that case, we can avoid permanent salaries and employees tables. Instead, we can store the output we just obtained in our temporary table. We can create our temporary table by following the syntax:



## Exercise #1:

Store the highest contract salary values of all male employees in a temporary table called *male\_max\_salaries*.

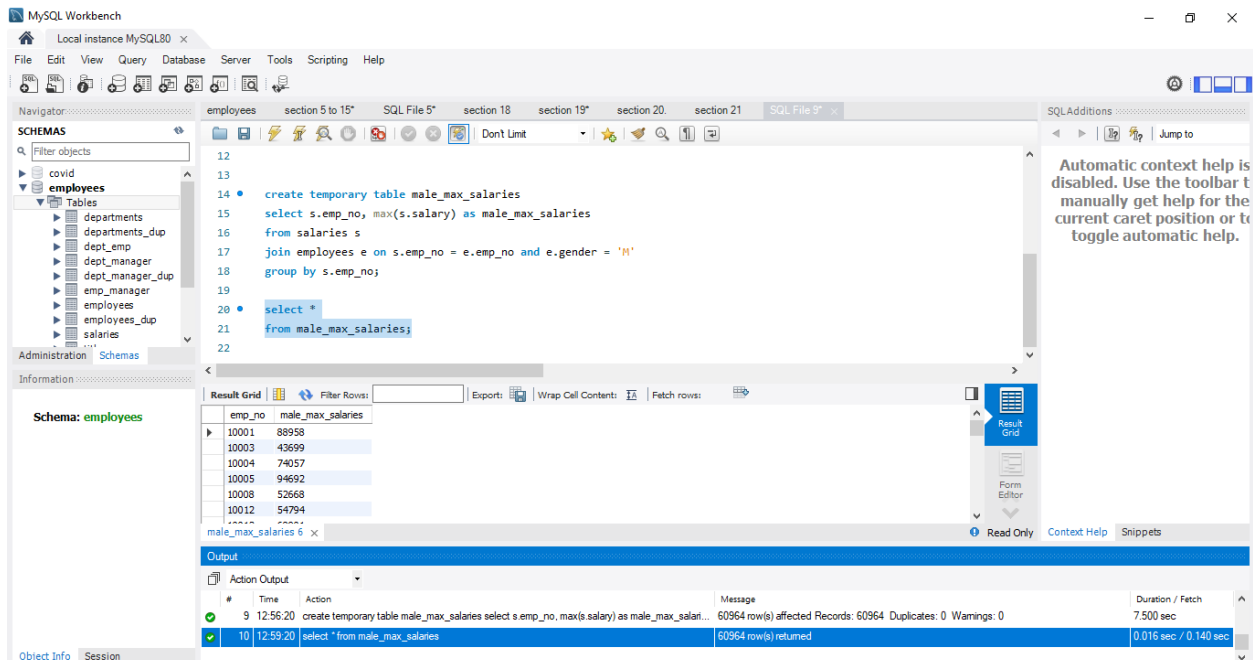
## Solution:



## Exercise #2:

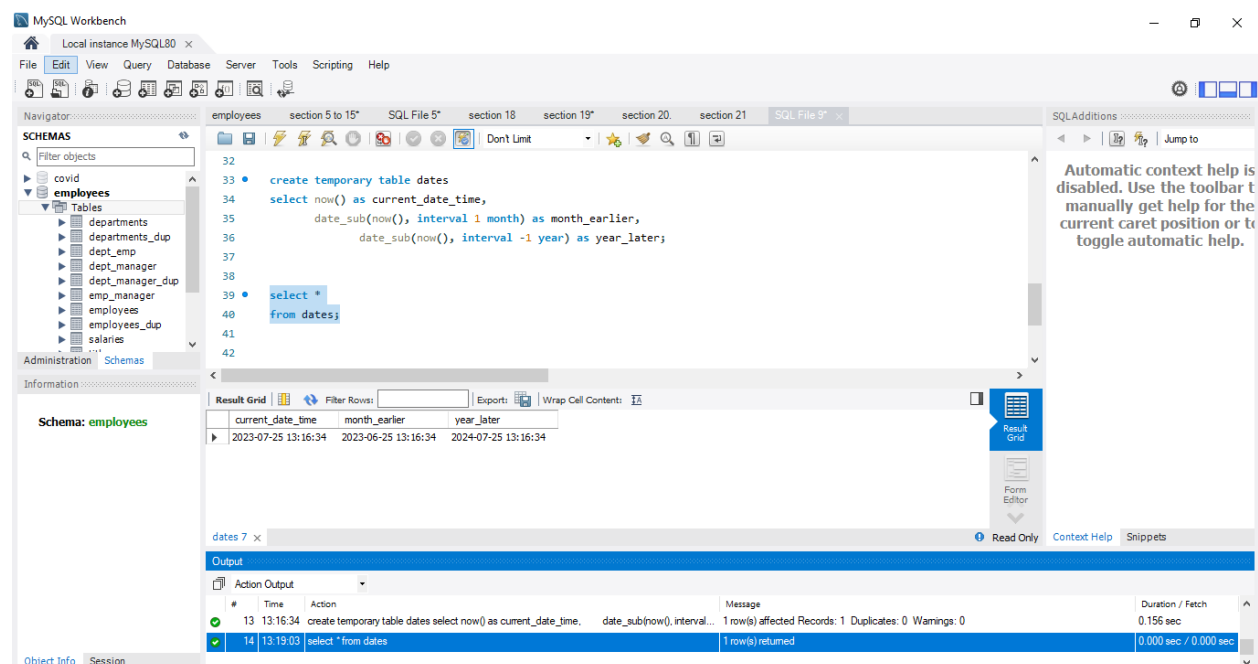
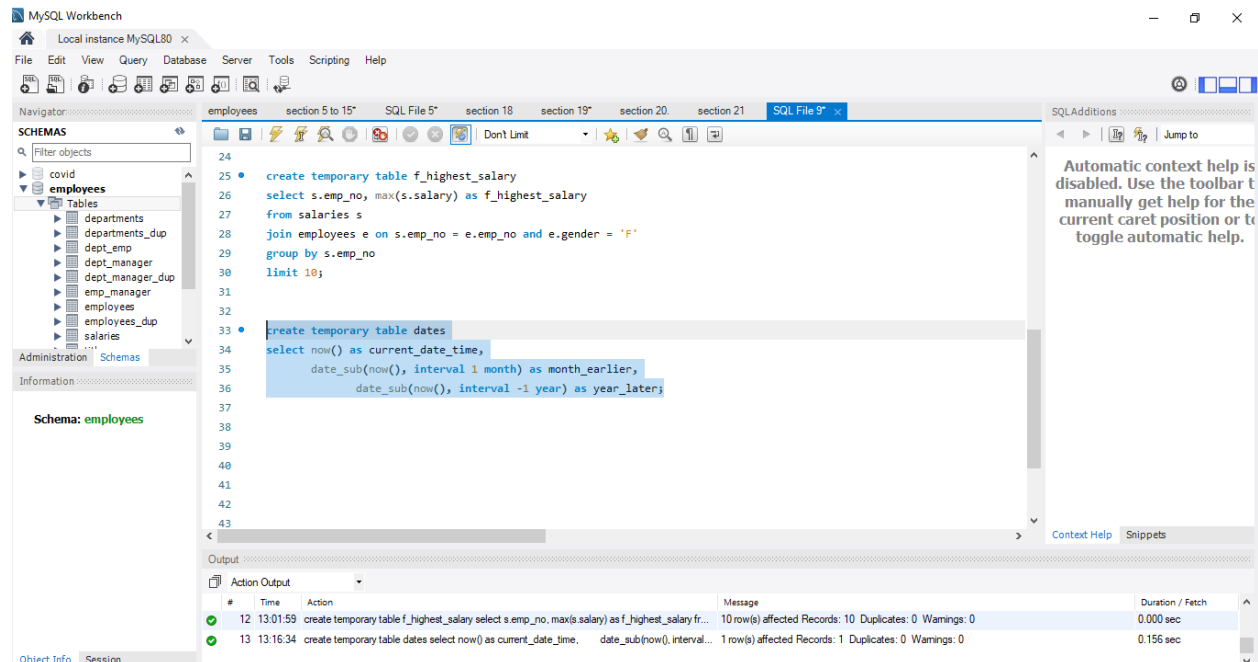
Write a query that, upon execution, allows you to check the result set contained in the *male\_max\_salaries* temporary table you created in the previous exercise.

## Solution:

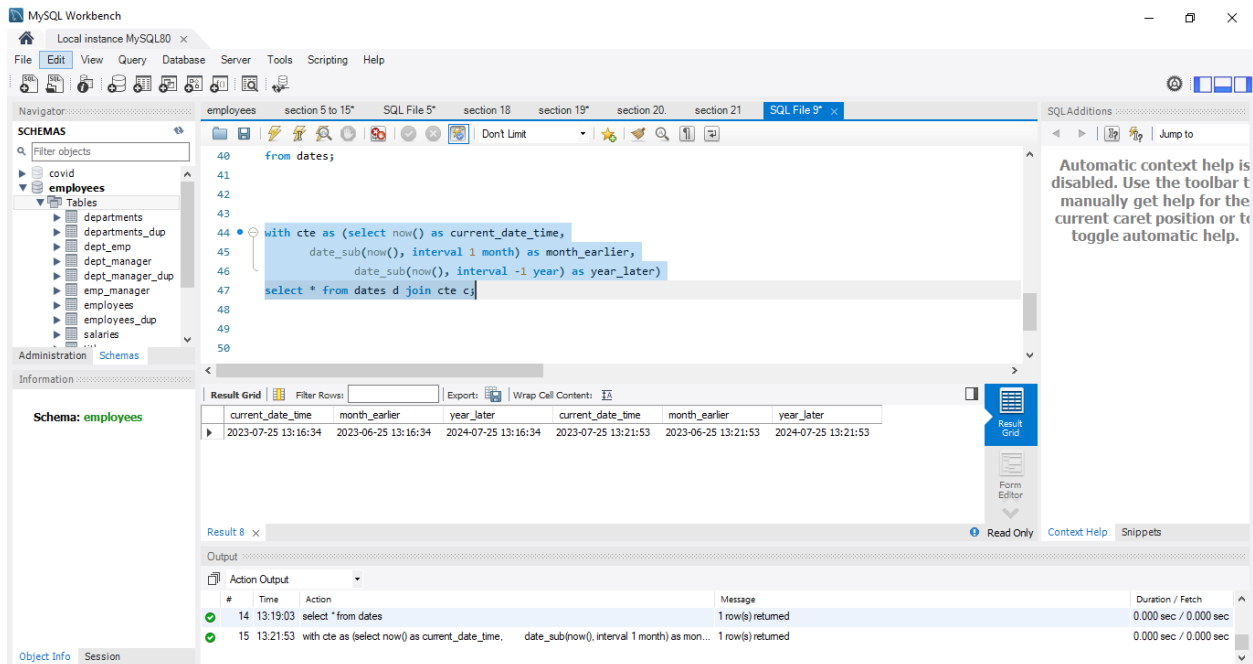


Now, let's create a temporary table called "dates", which contains the following the three datetime values:

1. The current date and time
2. A month earlier than the current date and time
3. A year later than the current date and time



We can use CTEs to invoke our temporary table more than once. That's how:

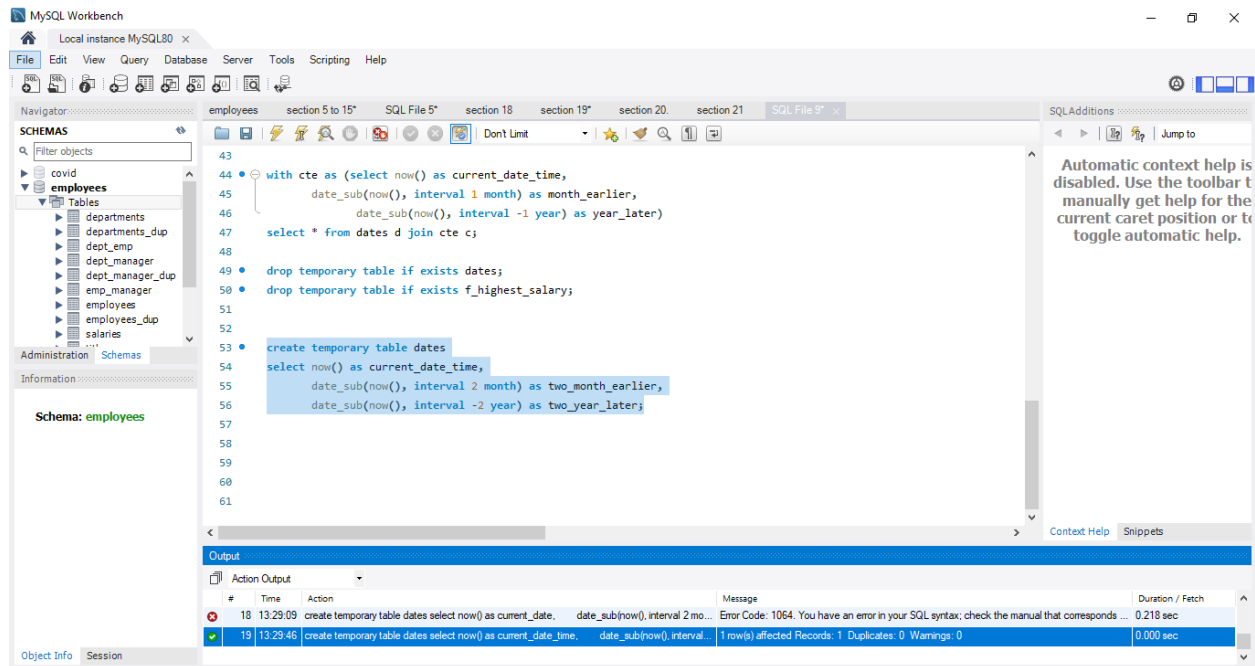


## Exercise #1:

Create a temporary table called *dates* containing the following three columns:

- one displaying the current date and time,
- another one displaying two months earlier than the current date and time, and a
- third column displaying two years later than the current date and time.

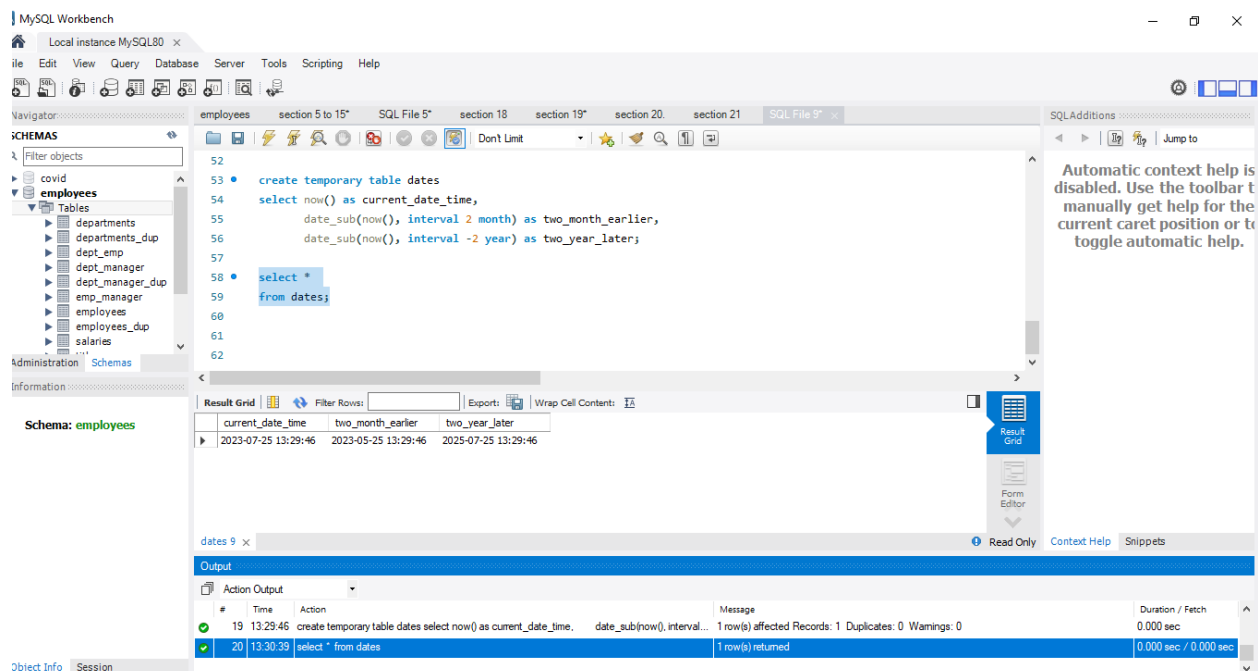
## Solution:



## Exercise #2:

Write a query that, upon execution, allows you to check the result set contained in the *dates* temporary table you created in the previous exercise.

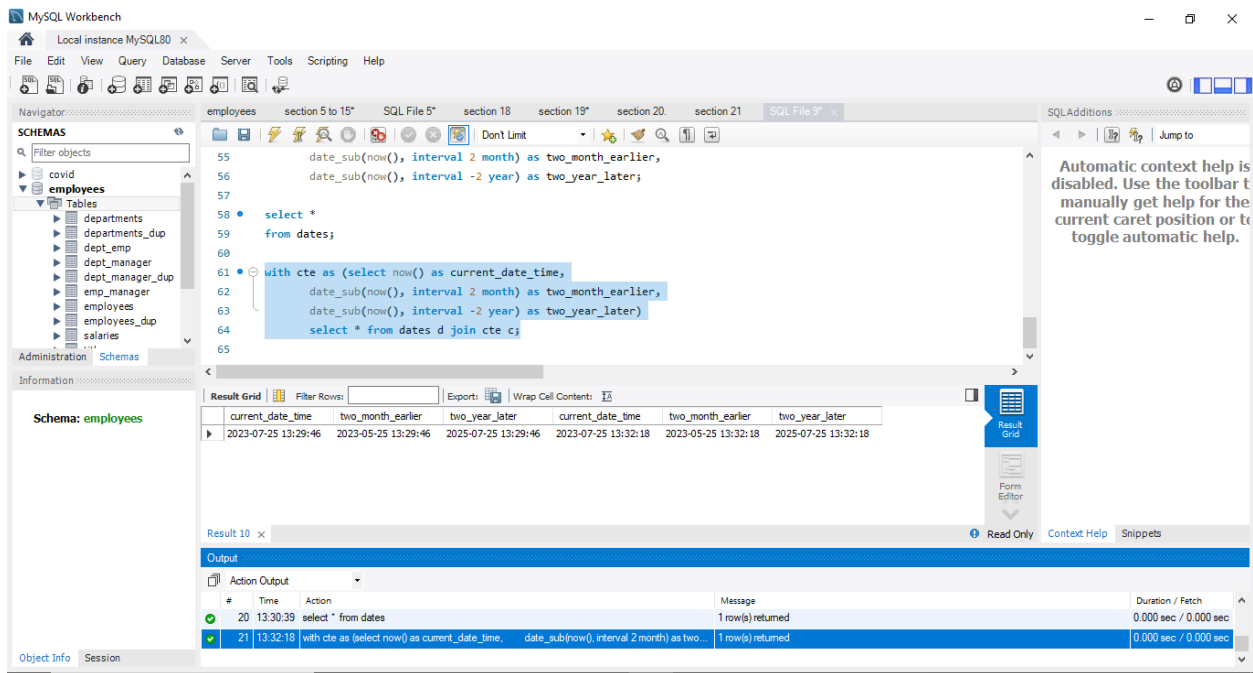
## Solution:



### Exercise #3:

Create a query joining the result sets from the *dates* temporary table you created during the previous lecture with a new Common Table Expression (CTE) containing the same columns. Let all columns in the result set appear on the same row.

### Solution:



The screenshot shows the MySQL Workbench interface. The SQL Editor contains the following query:

```
55 date_sub(now(), interval 2 month) as two_month_earlier,  
56 date_sub(now(), interval -2 year) as two_year_later;  
57  
58 select *  
59 from dates;  
60  
61 with cte as (select now() as current_date_time,  
62 date_sub(now(), interval 2 month) as two_month_earlier,  
63 date_sub(now(), interval -2 year) as two_year_later)  
64 select * from dates d join cte c;  
65
```

The Results window shows the output of the query, which is a table with 6 columns: *current\_date\_time*, *two\_month\_earlier*, *two\_year\_later*, *current\_date\_time*, *two\_month\_earlier*, and *two\_year\_later*. The first row of data shows the following values:

current_date_time	two_month_earlier	two_year_later	current_date_time	two_month_earlier	two_year_later
2023-07-25 13:29:46	2023-05-25 13:29:46	2025-07-25 13:29:46	2023-07-25 13:32:18	2023-05-25 13:32:18	2025-07-25 13:32:18

The Action Output window shows the execution of the query, with the following message:

#	Time	Action	Message	Duration / Fetch
20	13:30:39	select * from dates	1 row(s) returned	0.000 sec / 0.000 sec
21	13:32:18	with cte as (select now() as current_date_time, date_sub(now(), interval 2 month) as two...	1 row(s) returned	0.000 sec / 0.000 sec

### Exercise #4:

Again, create a query joining the result sets from the *dates* temporary table you created during the previous lecture with a new Common Table Expression (CTE) containing the same columns. This time, combine the two sets vertically.

### Solution:

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

```

61 with cte as (select now() as current_date_time,
62              date_sub(now(), interval 2 month) as two_month_earlier,
63              date_sub(now(), interval -2 year) as two_year_later)
64 select * from dates d join cte c;
65
66
67 with cte as (select now() as current_date_time,
68              date_sub(now(), interval 2 month) as two_month_earlier,
69              date_sub(now(), interval -2 year) as two_year_later)
70 select * from dates d union select * from cte c;
71

```

current_date_time	two_month_earlier	two_year_later
2023-07-25 13:29:46	2023-05-25 13:29:46	2025-07-25 13:29:46
2023-07-25 13:34:05	2023-05-25 13:34:05	2025-07-25 13:34:05

#	Time	Action	Message	Duration / Fetch
22	13:33:46	with cte as (select now() as current_date_time, date_sub(now(), interval 2 month) as two...	Error Code: 1054. Unknown column 'cte' in 'field list'	0.000 sec
23	13:34:05	with cte as (select now() as current_date_time, date_sub(now(), interval 2 month) as two...	2 row(s) returned	0.094 sec / 0.000 sec

## Exercise #5:

Drop the *male\_max\_salaries* and *dates* temporary tables you recently created.

## Solution:

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

```

58 select *
59 from dates;
60
61 with cte as (select now() as current_date_time,
62              date_sub(now(), interval 2 month) as two_month_earlier,
63              date_sub(now(), interval -2 year) as two_year_later)
64 select * from dates d join cte c;
65
66
67 with cte as (select now() as current_date_time,
68              date_sub(now(), interval 2 month) as two_month_earlier,
69              date_sub(now(), interval -2 year) as two_year_later)
70 select * from dates d union select * from cte c;
71
72
73 drop temporary table if exists dates;
74 drop temporary table if exists male_max_salaries;
75
76

```

#	Time	Action	Message	Duration / Fetch
25	13:35:16	drop temporary table if exists dates	0 row(s) affected	0.000 sec
26	13:35:39	drop temporary table if exists male_max_salaries	0 row(s) affected	0.015 sec