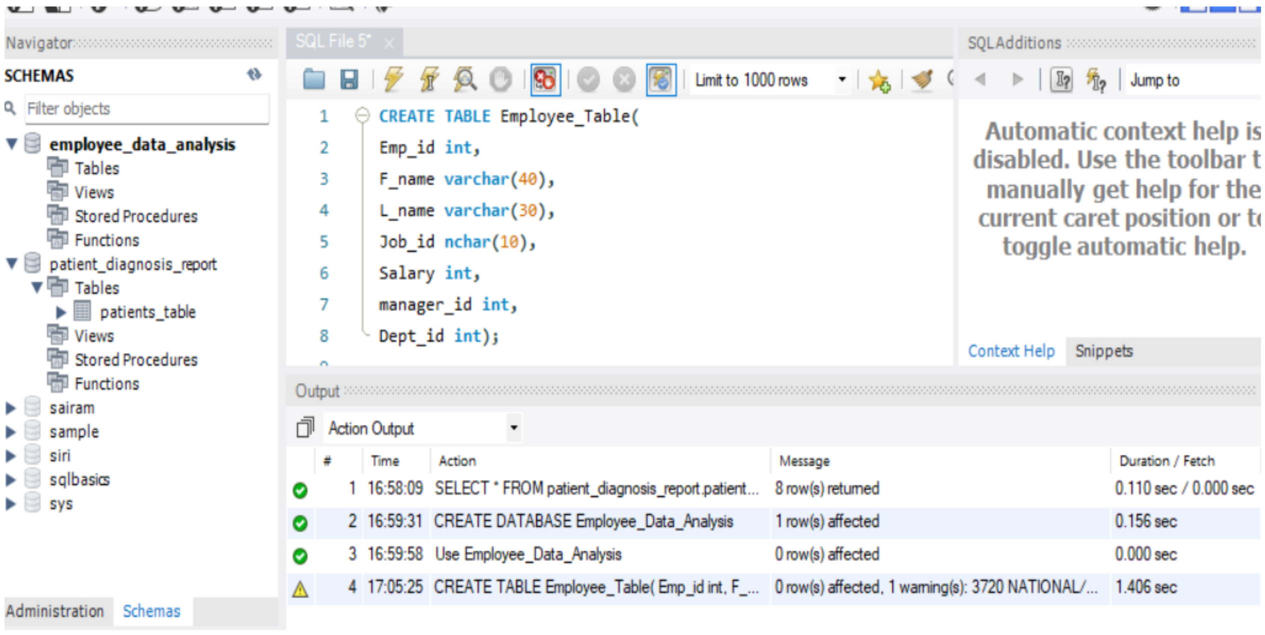
Practice project 8.15 Employee Data Analysis.

Course 4 SQL Training

Task

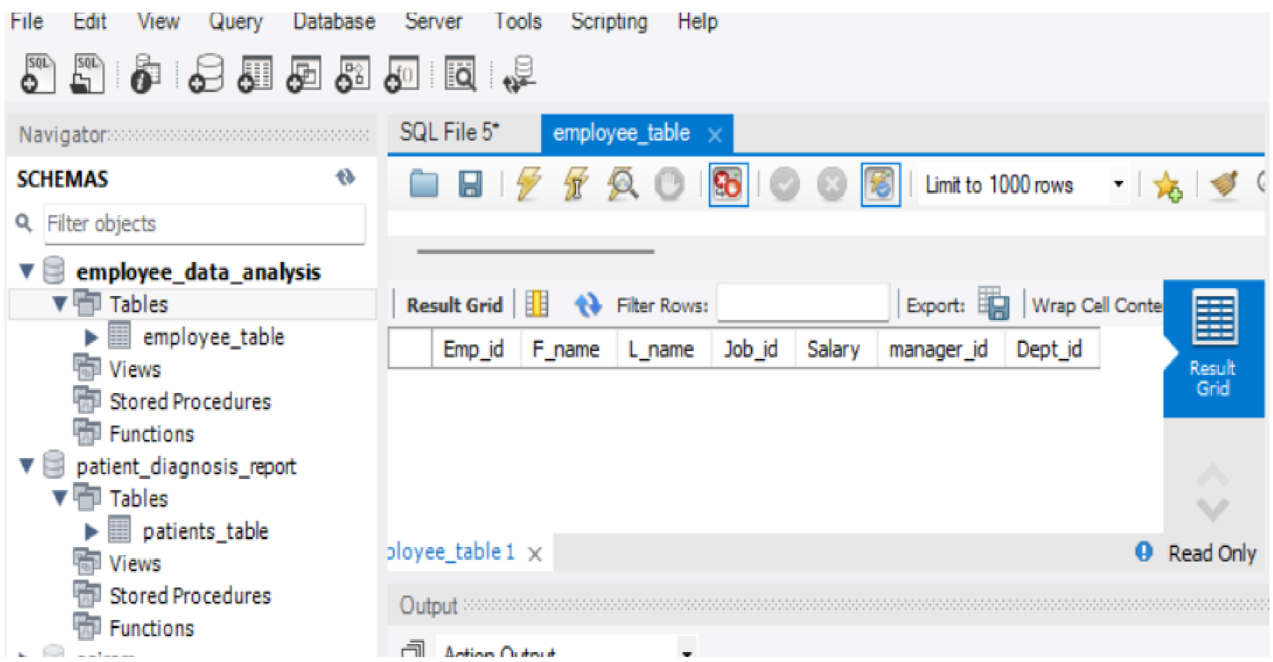
1. Write a query to **create** an **employee table** with the fields employee id, first name, last name, job id, salary, manager id, and department id.



```
CREATE TABLE Employee_Table(
Emp_id int,
F_name varchar(40),
L_name varchar(30),
Job_id nchar(10),
Salary int,
manager_id int,
Dept_id int);
```

```
#  Time      Action                                                   Message                                              Duration / Fetch
1  16:58:09  SELECT * FROM patient_diagnosis_report.patient...        8 row(s) returned                                    0.110 sec / 0.000 sec
2  16:59:31  CREATE DATABASE Employee_Data_Analysis                   1 row(s) affected                                    0.156 sec
3  16:59:58  Use Employee_Data_Analysis                               0 row(s) affected                                    0.000 sec
4  17:05:25  CREATE TABLE Employee_Table( Emp_id int, F_...           0 row(s) affected, 1 warning(s): 3720 NATIONAL/...   1.406 sec
```

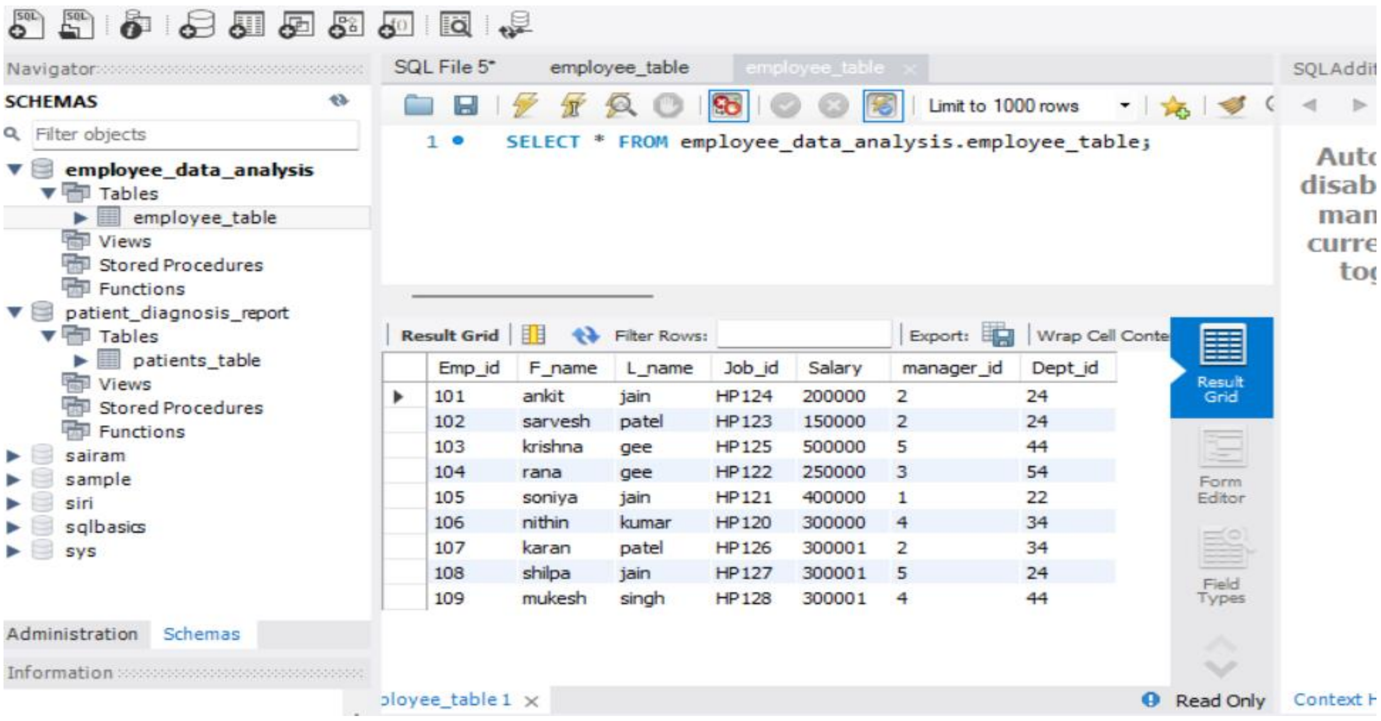
Automatic context help is disabled. Use the toolbar t manually get help for the current caret position or t toggle automatic help.



```
Emp_id  F_name  L_name  Job_id  Salary  manager_id  Dept_id
```

employee_table
Result Grid
Read Only

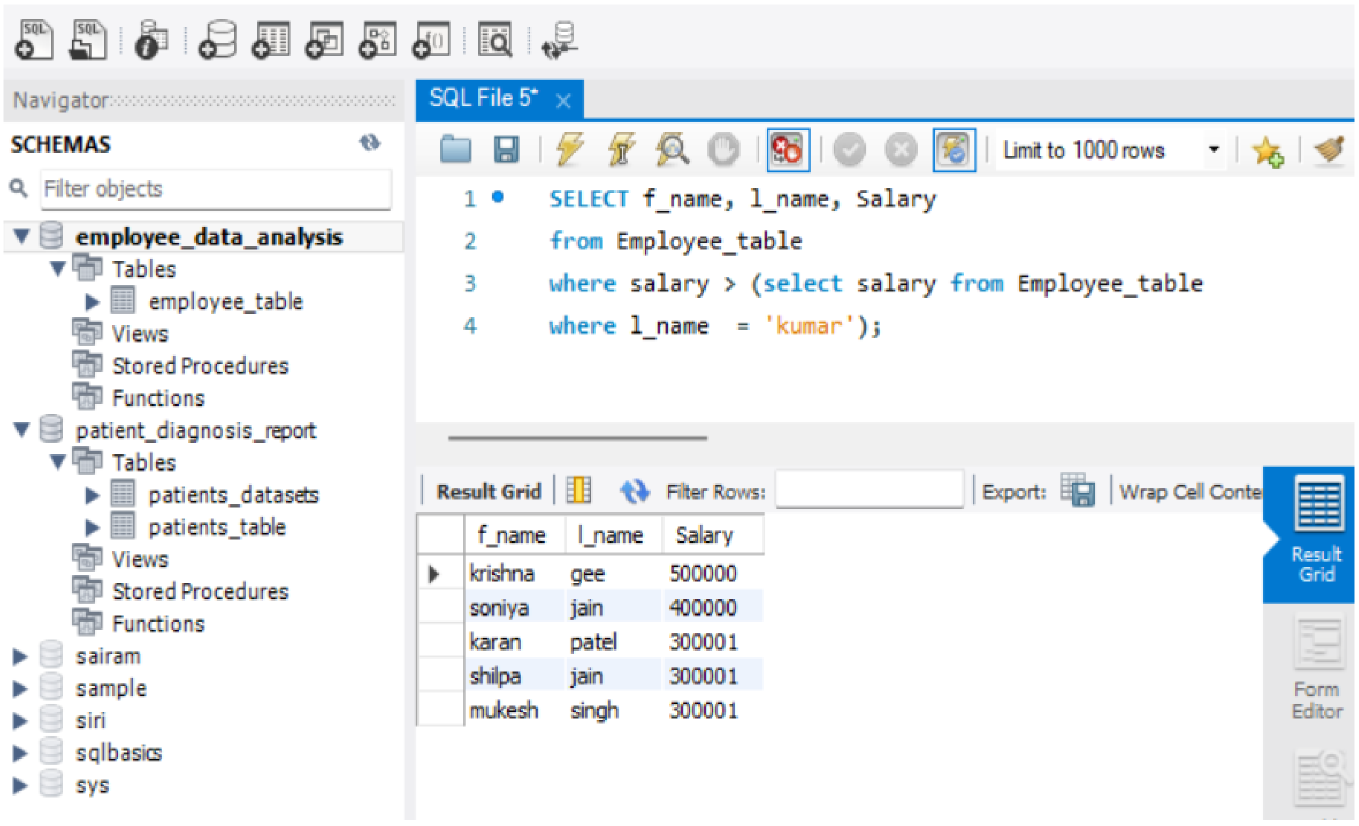
2. Write a query to **insert** values into the employee table. (*Imported the dataset into Employee_table…*)



```
SELECT * FROM employee_data_analysis.employee_table;
```

Emp_id	F_name	L_name	Job_id	Salary	manager_id	Dept_id
101	ankit	jain	HP124	200000	2	24
102	sarvesh	patel	HP123	150000	2	24
103	krishna	gee	HP125	500000	5	44
104	rana	gee	HP122	250000	3	54
105	soniya	jain	HP121	400000	1	22
106	nithin	kumar	HP120	300000	4	34
107	karan	patel	HP126	300001	2	34
108	shilpa	jain	HP127	300001	5	24
109	mukesh	singh	HP128	300001	4	44

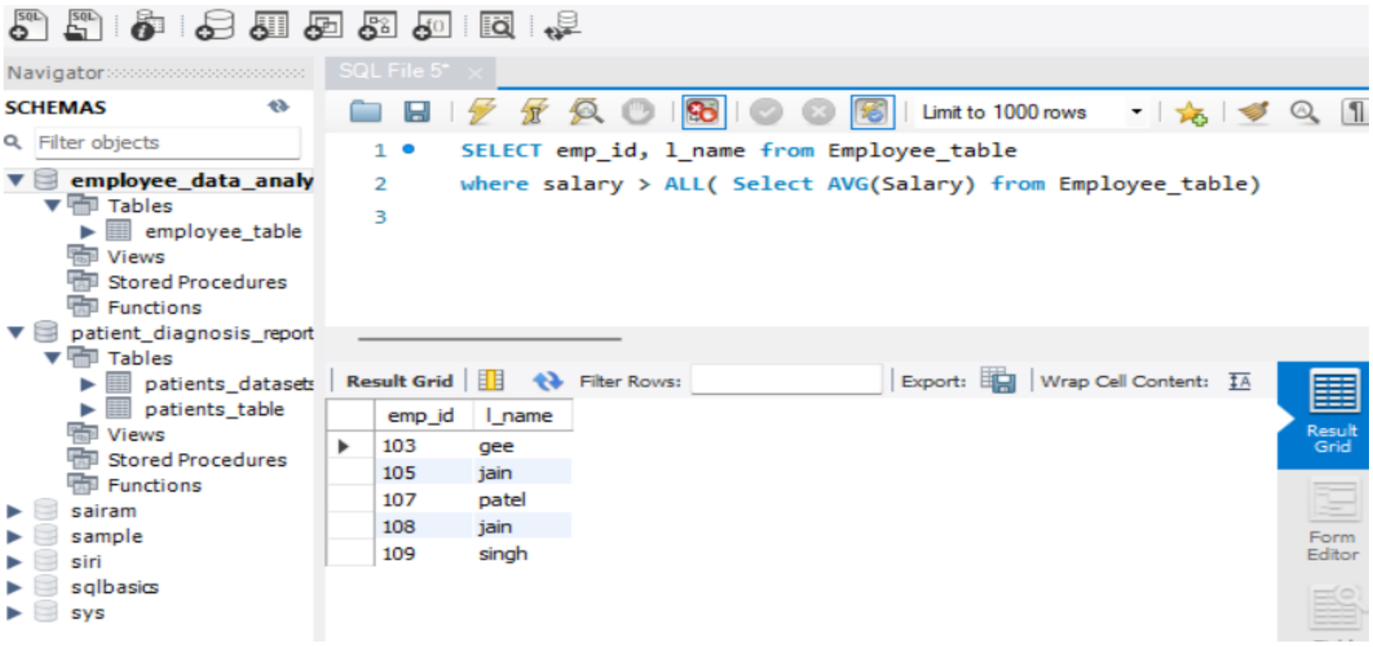
3. Write a query to find the **first name** and **salary** of the employee whose **salary is higher than the employee with the last name Kumar** from the employee table.



```
SELECT f_name, l_name, Salary
from Employee_table
where salary > (select salary from Employee_table
where l_name  = 'kumar');
```

f_name	l_name	Salary
krishna	gee	500000
soniya	jain	400000
karan	patel	300001
shilpa	jain	300001
mukesh	singh	300001

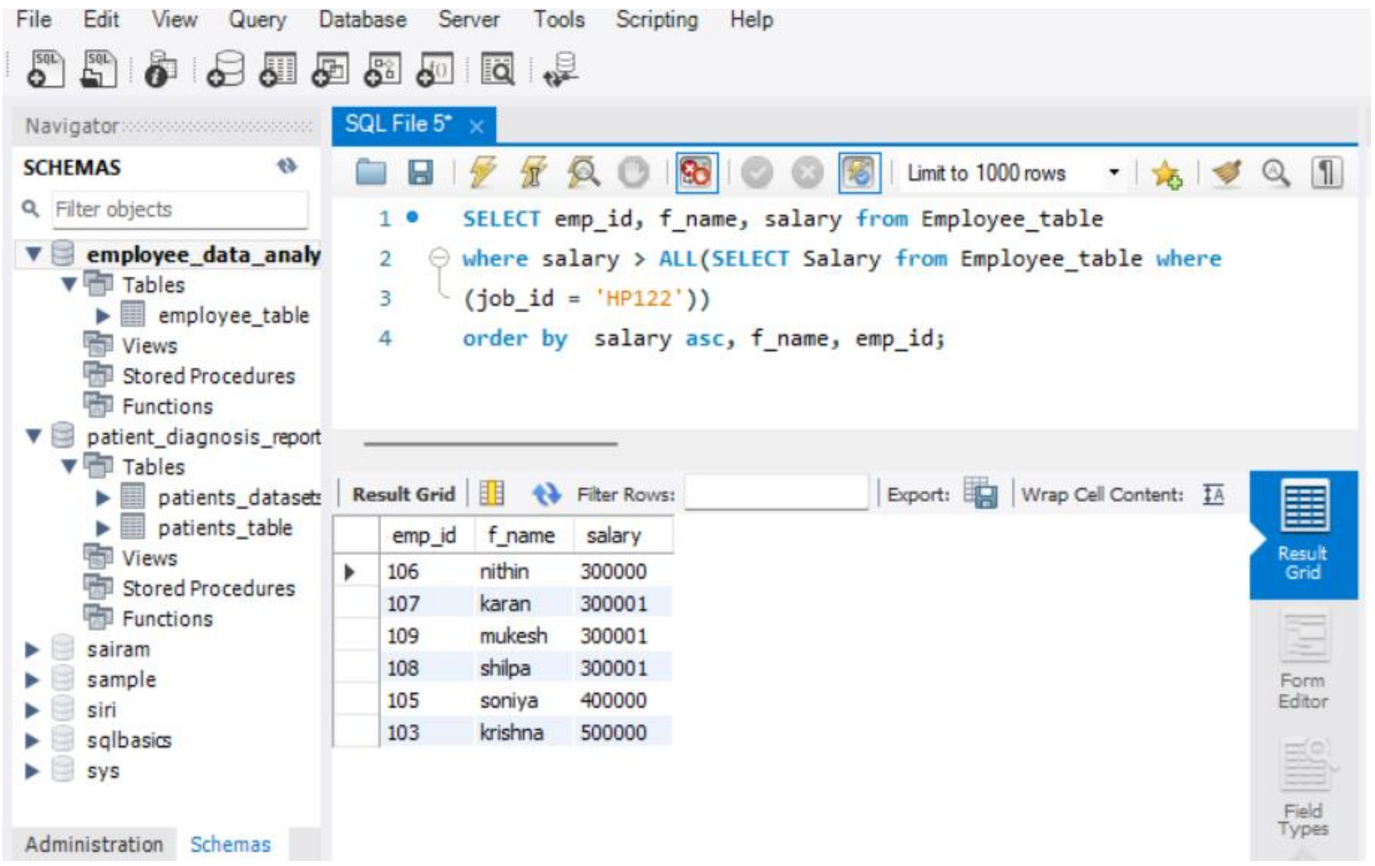
4. Write a query to display the **employee id** and **last name** of the employee **whose salary is greater than the average salary** from the employee table.



```
SELECT emp_id, l_name from Employee_table
where salary > ALL( Select AVG(Salary) from Employee_table)
```

emp_id	l_name
103	gee
105	jain
107	patel
108	jain
109	singh

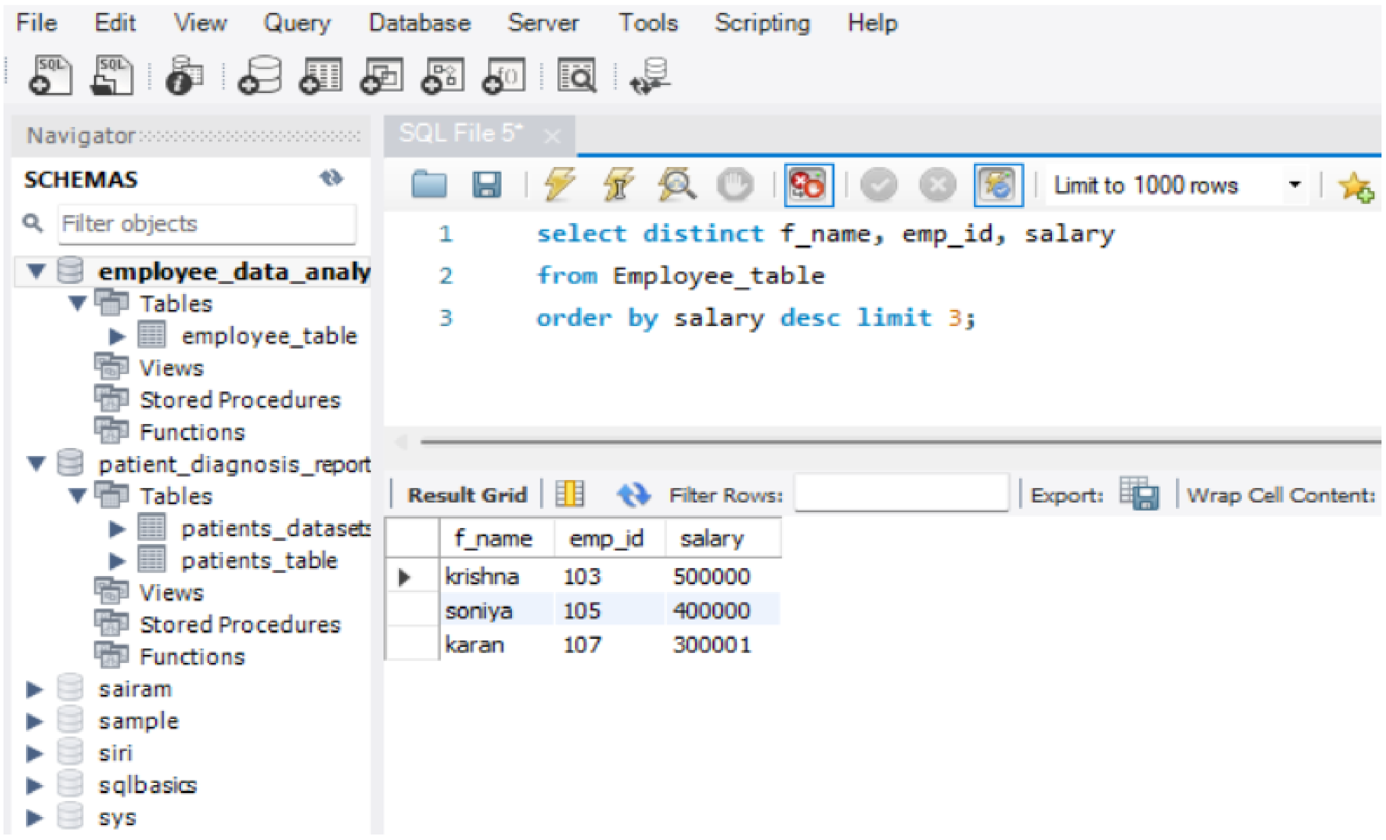
5. Write a query to display the **employee id, first name**, and **salary** of the employees who earn **a salary that is higher than the salary** of all the shipping clerks **(JOB_ID = HP122**). Sort the results of the salary in ascending order.



```
SELECT emp_id, f_name, salary from Employee_table
where salary > ALL(SELECT Salary from Employee_table where
(job_id = 'HP122'))
order by  salary asc, f_name, emp_id;
```

emp_id	f_name	salary
106	nithin	300000
107	karan	300001
109	mukesh	300001
108	shilpa	300001
105	soniya	400000
103	krishna	500000

6. Write a query to display the **first name, employee id,** and **salary** of the first three employees with **highest salaries**.



```sql
select distinct f_name, emp_id, salary
from Employee_table
order by salary desc limit 3;
```

f_name	emp_id	salary
krishna	103	500000
soniya	105	400000
karan	107	300001