SQL Practice Project -4 submitted by Sujit Sonar:

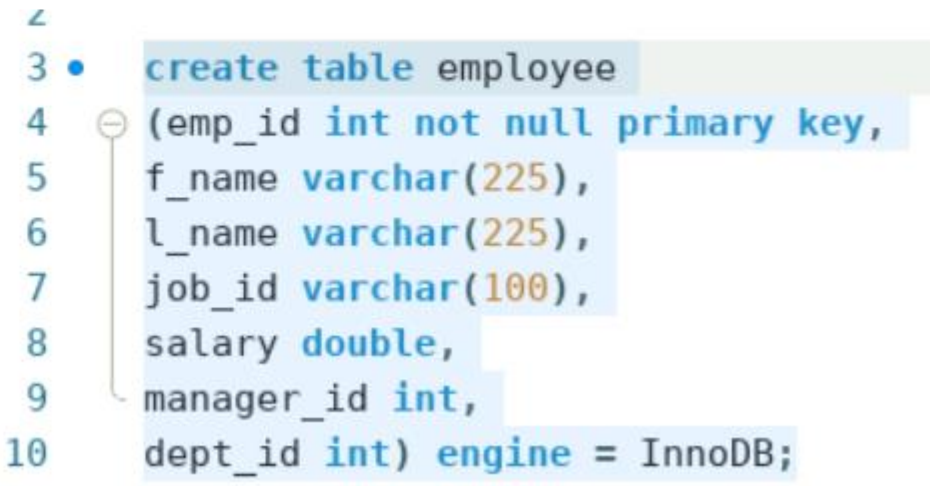
Employee Data Analysis.

DESCRIPTION
An HR of the company wants to extract, update, and delete employee details to maintain their records.

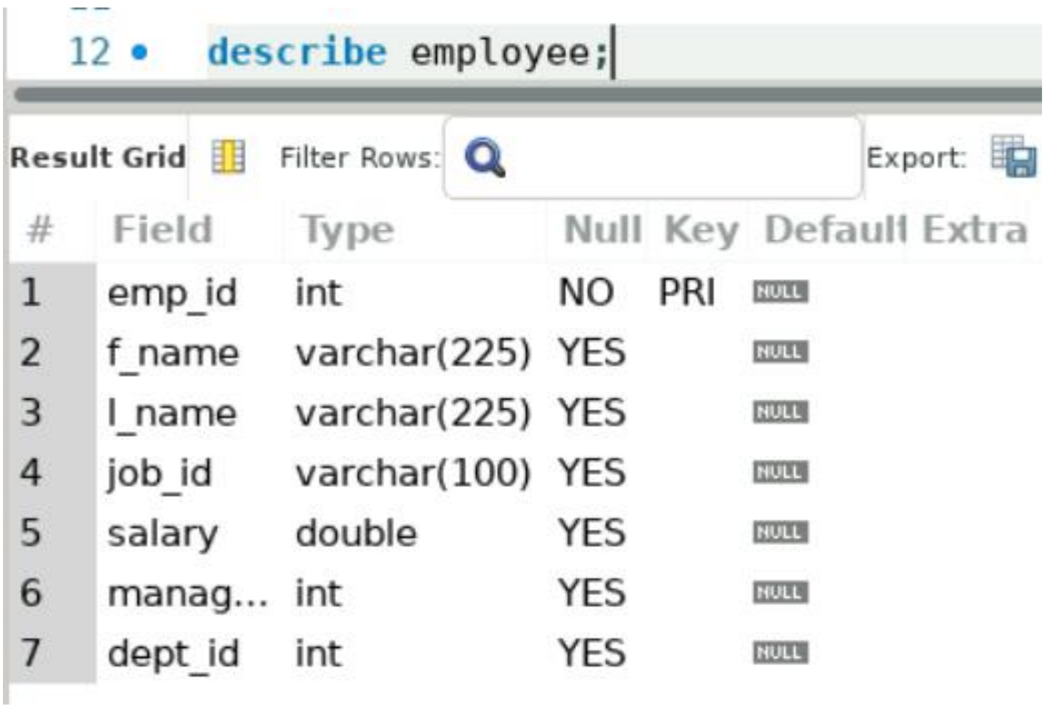
Objective:
The database design helps to calculate the monthly payroll of each employee efficiently.

Task to be performed:

- 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.



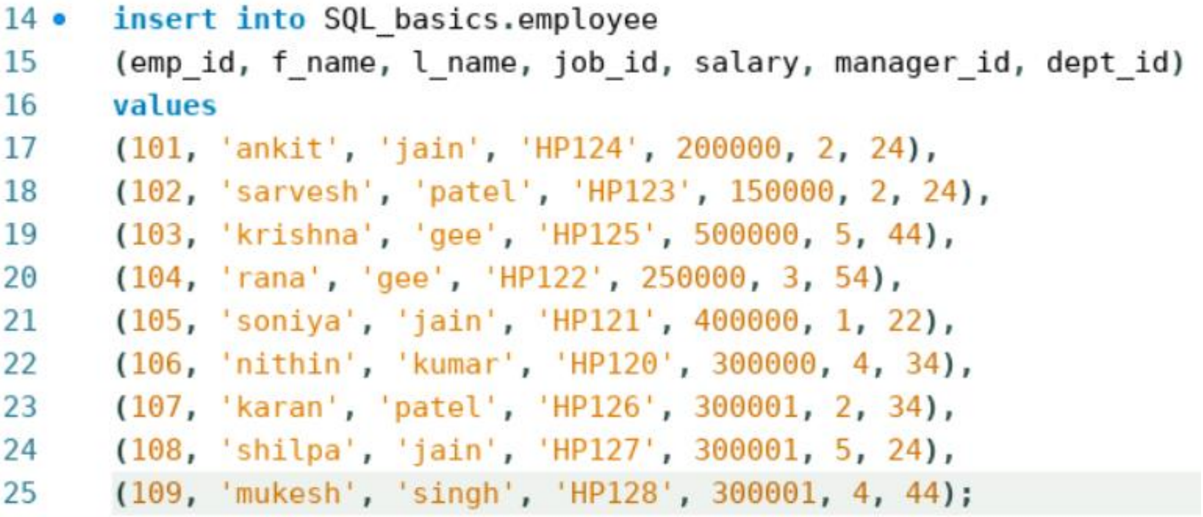
```
create table employee
(emp_id int not null primary key,
f_name varchar(225),
l_name varchar(225),
job_id varchar(100),
salary double,
manager_id int,
dept_id int) engine = InnoDB;
```



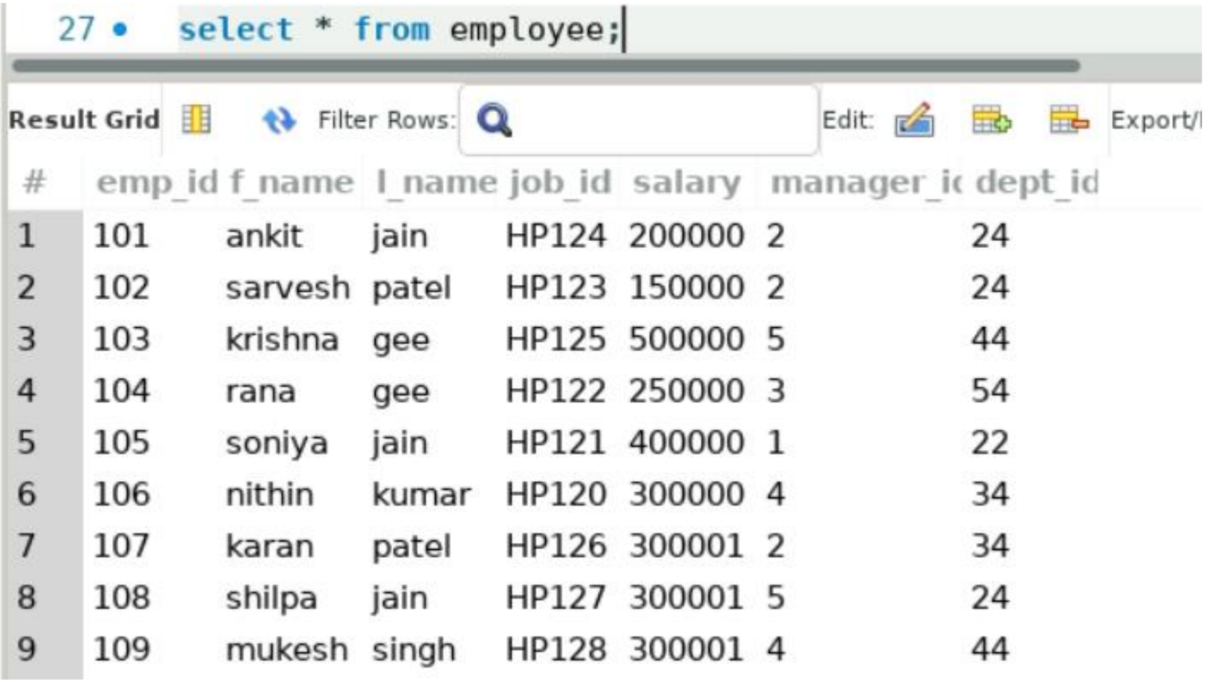
```
describe employee;
```

#	Field	Type	Null	Key	Default	Extra
1	emp_id	int	NO	PRI	NULL	
2	f_name	varchar(225)	YES		NULL	
3	l_name	varchar(225)	YES		NULL	
4	job_id	varchar(100)	YES		NULL	
5	salary	double	YES		NULL	
6	manag...	int	YES		NULL	
7	dept_id	int	YES		NULL	

- Write a query to **insert** values into the employee table.



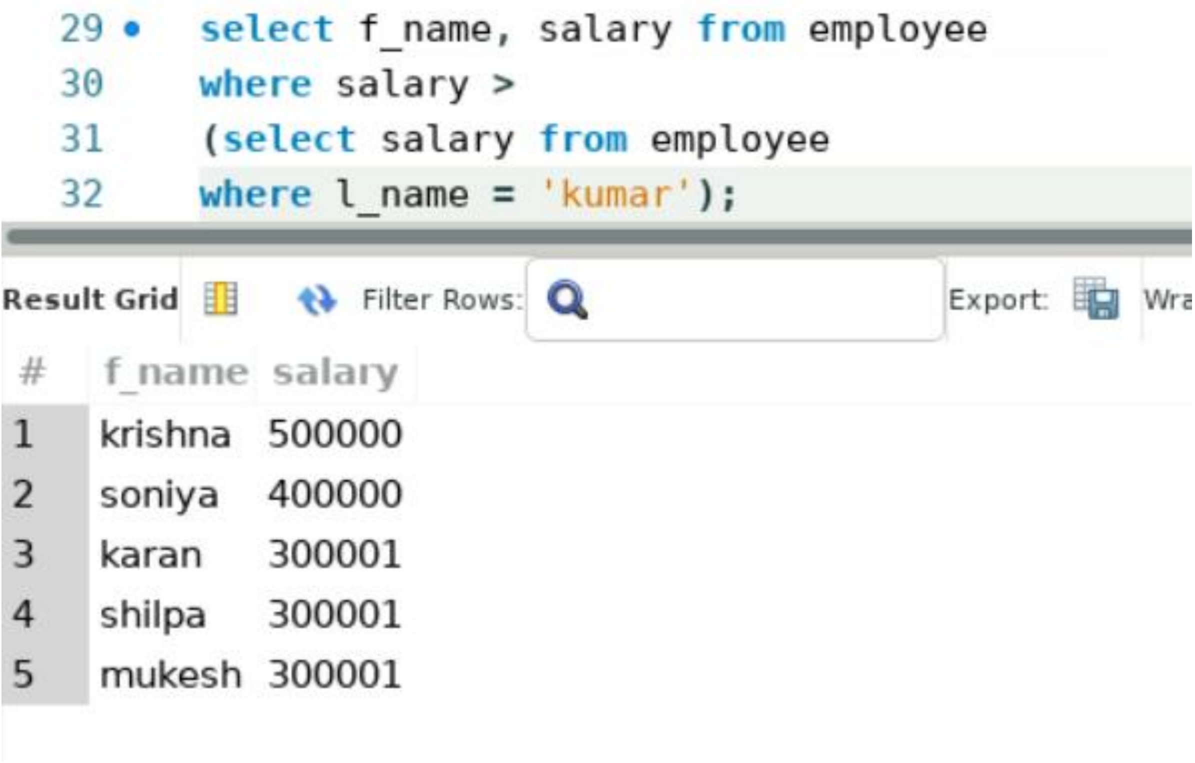
```sql
insert into SQL_basics.employee
(emp_id, f_name, l_name, job_id, salary, manager_id, dept_id)
values
(101, 'ankit', 'jain', 'HP124', 200000, 2, 24),
(102, 'sarvesh', 'patel', 'HP123', 150000, 2, 24),
(103, 'krishna', 'gee', 'HP125', 500000, 5, 44),
(104, 'rana', 'gee', 'HP122', 250000, 3, 54),
(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);
```



```sql
select * from employee;
```

#	emp_id	f_name	l_name	job_id	salary	manager_ic	dept_id
1	101	ankit	jain	HP124	200000	2	24
2	102	sarvesh	patel	HP123	150000	2	24
3	103	krishna	gee	HP125	500000	5	44
4	104	rana	gee	HP122	250000	3	54
5	105	soniya	jain	HP121	400000	1	22
6	106	nithin	kumar	HP120	300000	4	34
7	107	karan	patel	HP126	300001	2	34
8	108	shilpa	jain	HP127	300001	5	24
9	109	mukesh	singh	HP128	300001	4	44

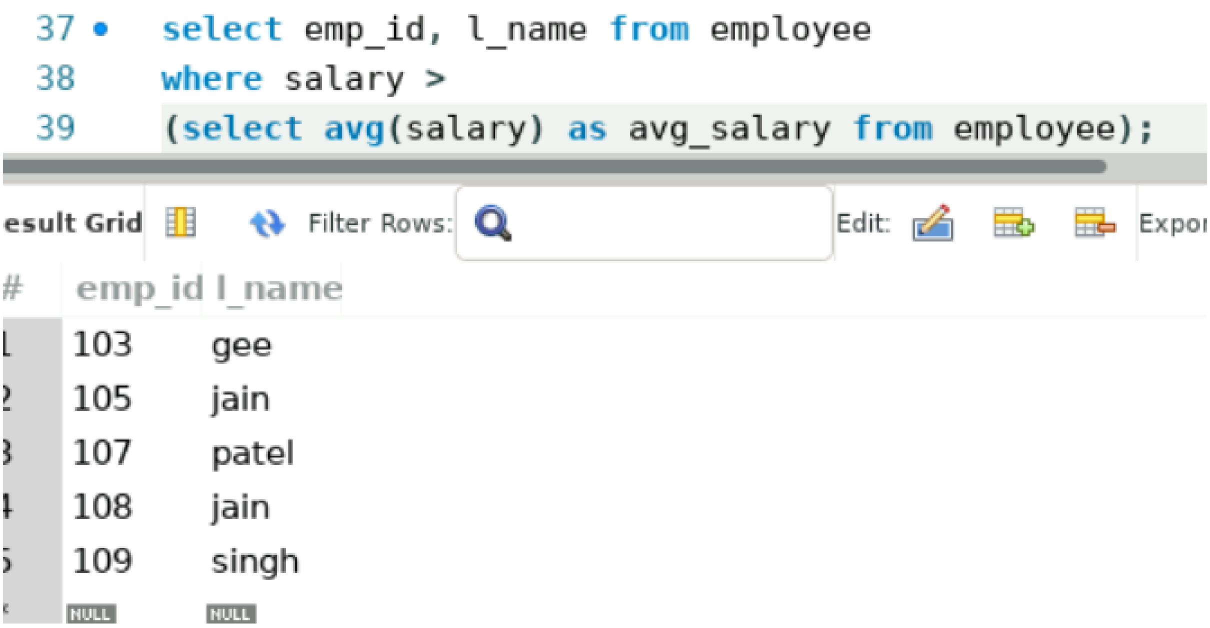
- Write a query to find the **first name** and **salary** of the employee whose **salary is higher than the employee with the last name Kumar** from the employee table.



```sql
select f_name, salary from employee
where salary >
(select salary from employee
where l_name = 'kumar');
```

```
#  f_name   salary
1  krishna  500000
2  soniya   400000
3  karan    300001
4  shilpa   300001
5  mukesh   300001
```

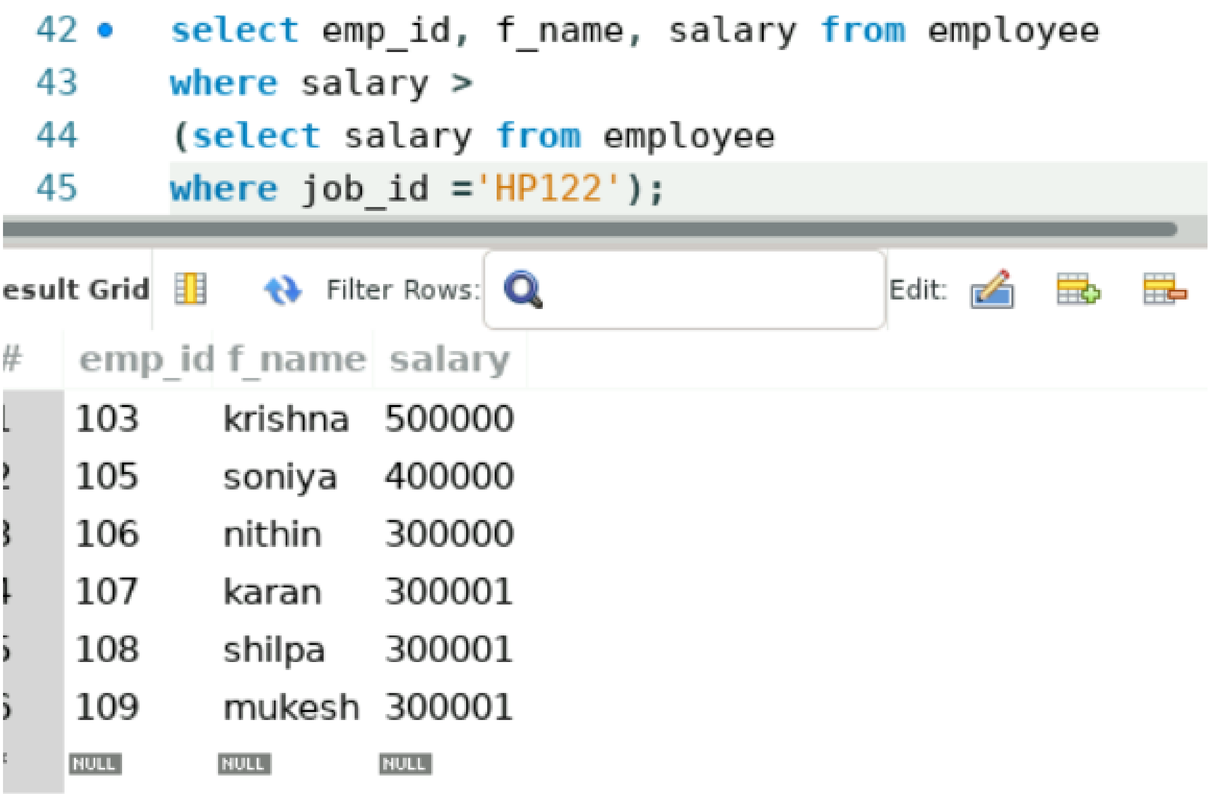
- Write a query to display the **employee id** and **last name** of the employee **whose salary is greater than the average salary** from the employee table.



```sql
select emp_id, l_name from employee
where salary >
(select avg(salary) as avg_salary from employee);
```

```
#  emp_id  l_name
l  103     gee
2  105     jain
3  107     patel
4  108     jain
5  109     singh
   NULL    NULL
```

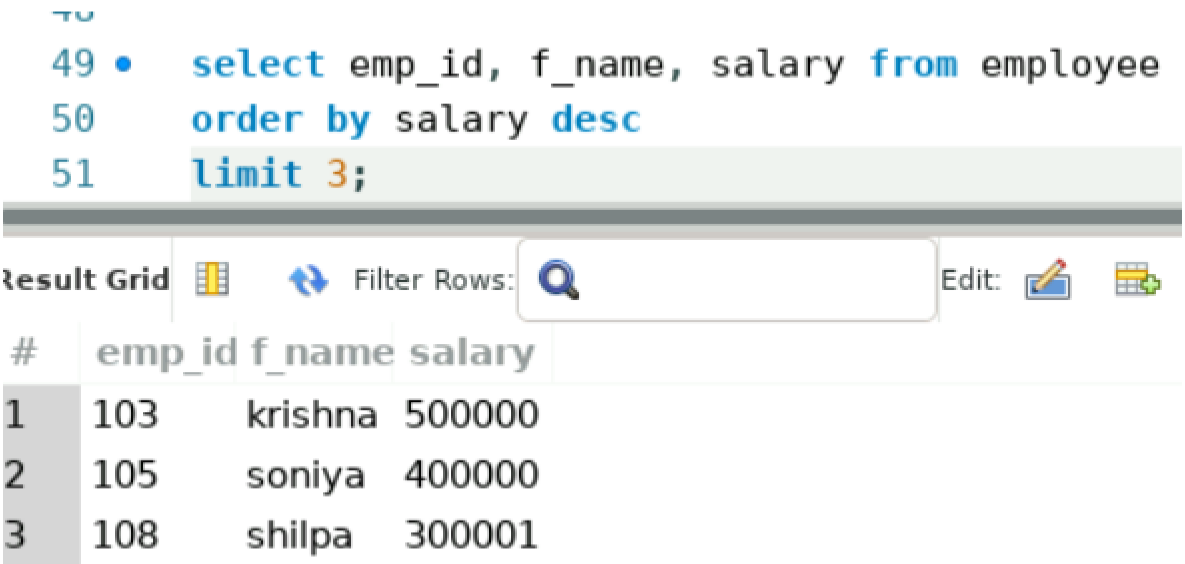
- Write a query to display the **employee id, first name**, and **salary** of the employees who earn **a salary that is higher than the salary** of all the shipping clerks **(JOB_ID = HP122)**. Sort the results of the salary in ascending order.



```
select emp_id, f_name, salary from employee
where salary >
(select salary from employee
where job_id ='HP122');
```

```
#  emp_id  f_name   salary
1  103     krishna  500000
2  105     soniya   400000
3  106     nithin   300000
4  107     karan    300001
5  108     shilpa   300001
6  109     mukesh   300001
*  NULL    NULL     NULL
```

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



```
select emp_id, f_name, salary from employee
order by salary desc
limit 3;
```

```
#  emp_id  f_name   salary
1  103     krishna  500000
2  105     soniya   400000
3  108     shilpa   300001
```

##################################### END#####################################