

**DIGITAL ASSIGNMENT 2**

**NAME: AYUSH SHARMA**

**REGISTER NUMBER: 21BDS0058**

**COURSE CODE: BCSE302P**

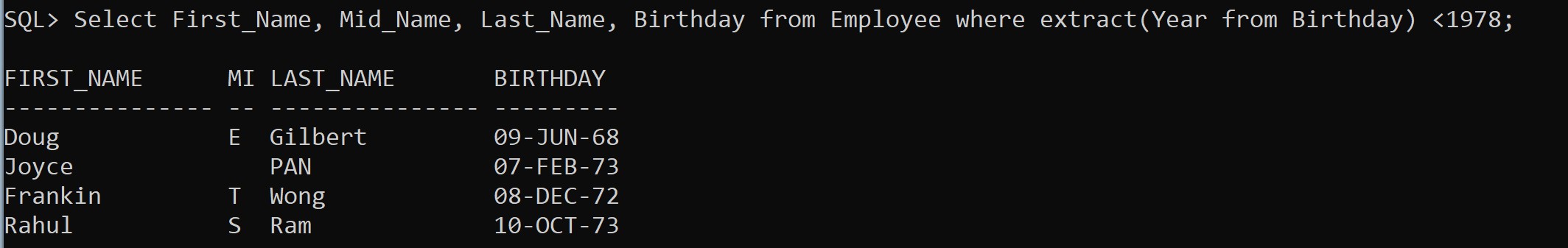
**COURSE NAME: DATABASE SYSTEM LAB**

# Aim: To understand different types of function in SQL Consider the following schema (used in EX2):

1. **Display the employee names whose bdate is on or before 1978. SQL Command:**

Select First\_Name, Mid\_Name, Last\_Name, Birthday from Employee where extract(Year from Birthday) <1978;

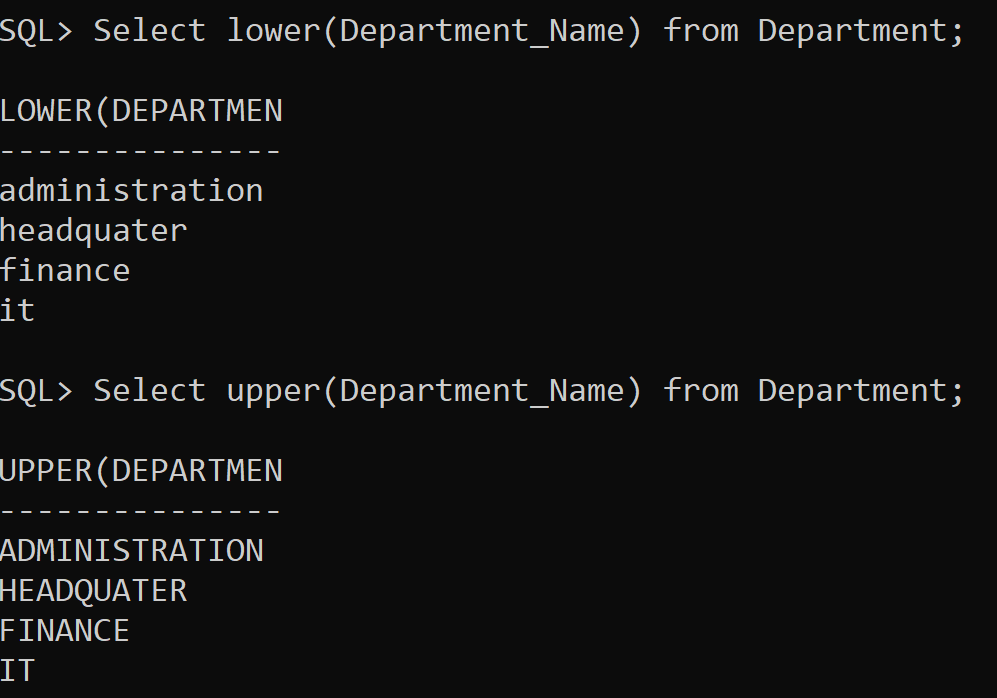
# Screenshot:



1. **Display all the department names in upper case and lower case. SQL Command:**

Select lower(Department\_Name) from Department; Select upper(Department\_Name) from Department;

# Screenshot:



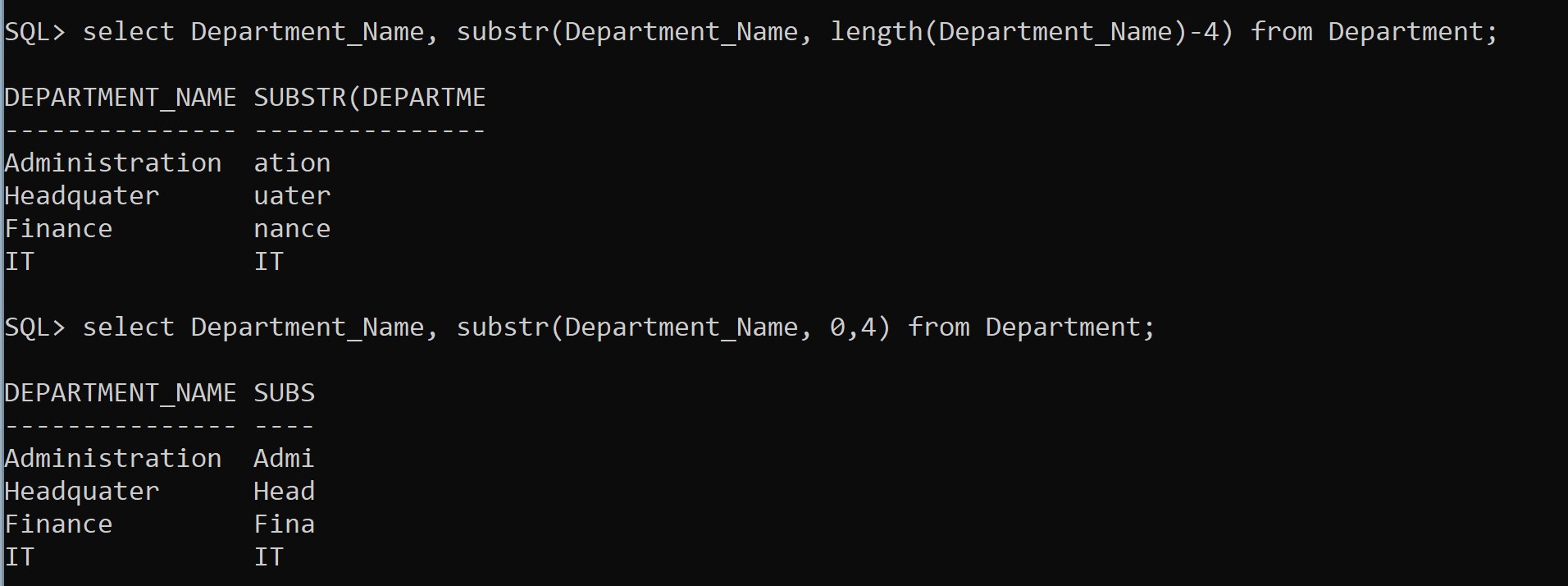
1. **Display the first four characters and last four of the department names using substring function.**

# SQL Command:

select Department\_Name, substr(Department\_Name, 0,4) from Department;

select Department\_Name, substr(Department\_Name, length(Department\_Name)-4) from Department;

# Screenshot:

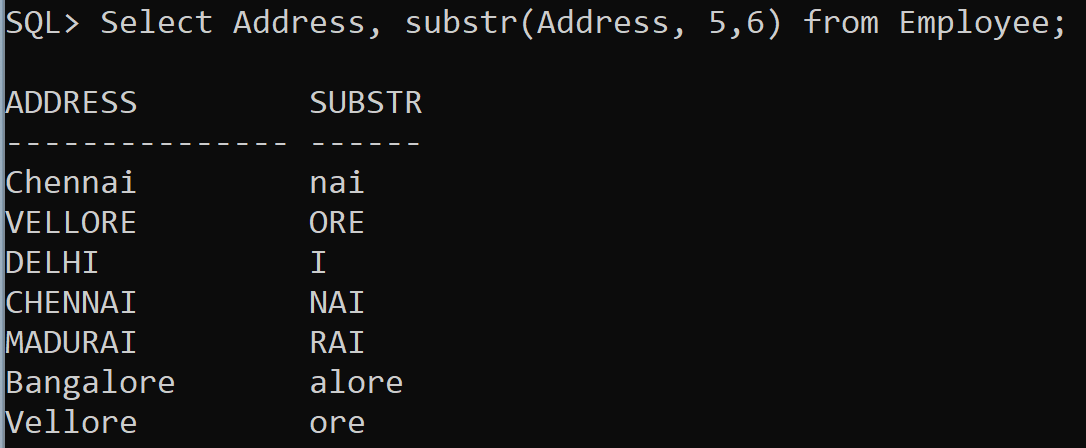


1. **Display the substring of the Address (starting from 5th position to 11 th position) of all employees.**

# SQL Command:

Select Address, substr(Address, 5,6) from Employee;

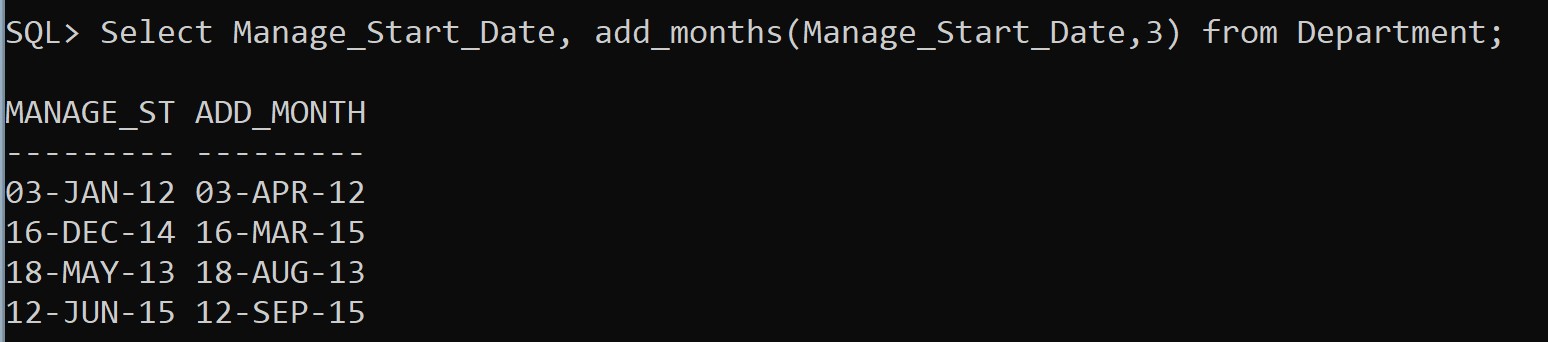
# Screenshot:



1. **Display the Mgrstartdate on adding three months to it. SQL Command:**

Select Manage\_Start\_Date, add\_months(Manage\_Start\_Date,3) from Department;

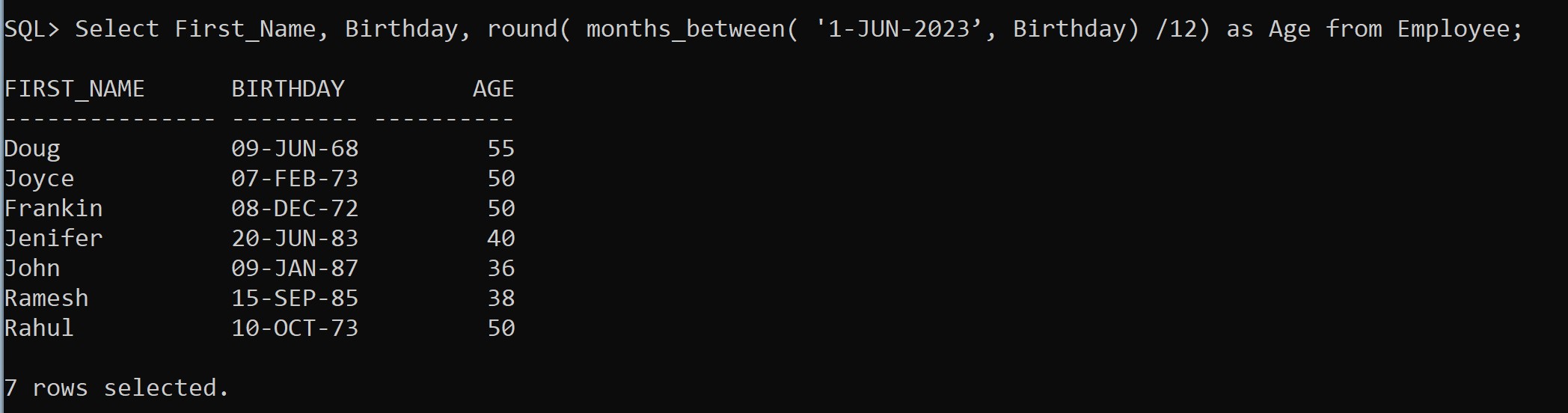
# Screenshot:



1. **Display the age of all the employees rounded to two digits. SQL Command:**

Select First\_Name, Birthday, round( months\_between( '1-JUN- 2023’, Birthday) /12) as Age from Employee;

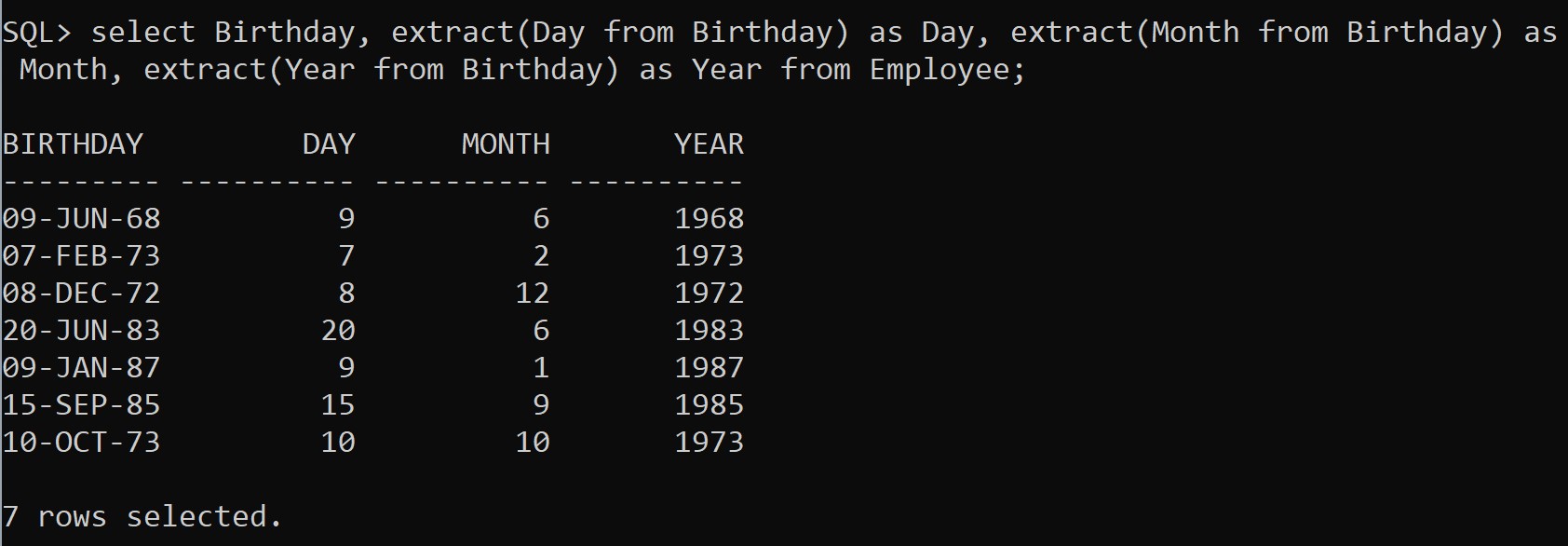
# Screenshot:



1. **Display the bdate of all employees in the format ‘DDthMonthYYYY’. SQL Command:**

select Birthday, extract(Day from Birthday) as Day, extract(Month from Birthday) as Month, extract(Year from Birthday) as Year from Employee;

# Screenshot:

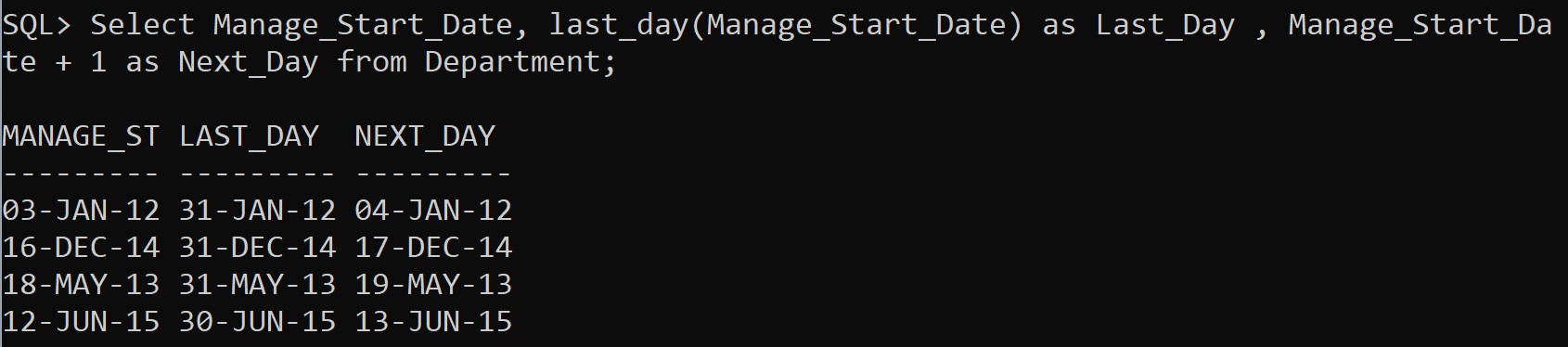


1. **Find the last day and next day of the month in which each manager has joined.**

# SQL Command:

Select Manage\_Start\_Date, last\_day(Manage\_Start\_Date) as Last\_Day , Manage\_Start\_Date + 1 as Next\_Day from Department;

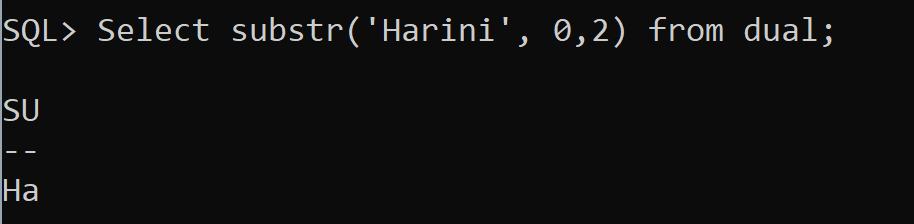
# Screenshot:



1. **Print a substring from the string ‘Harini’. SQL Command:**

Select substr('Harini', 0,2) from dual;

# Screenshot:

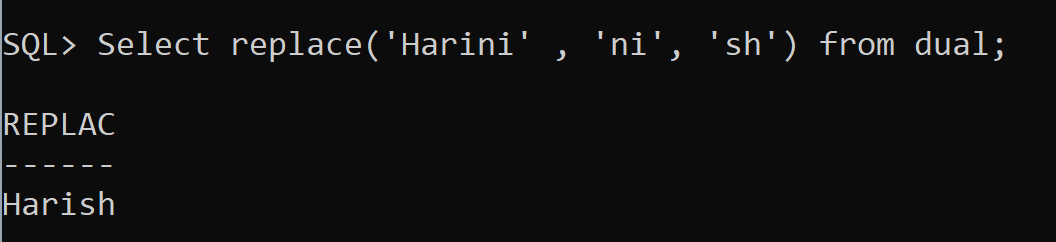


1. **Replace the string ‘ni’ from ‘Harini’ by ‘sh’.**

# SQL Command:

Select replace('Harini' , 'ni', 'sh') from dual;

# Screenshot:

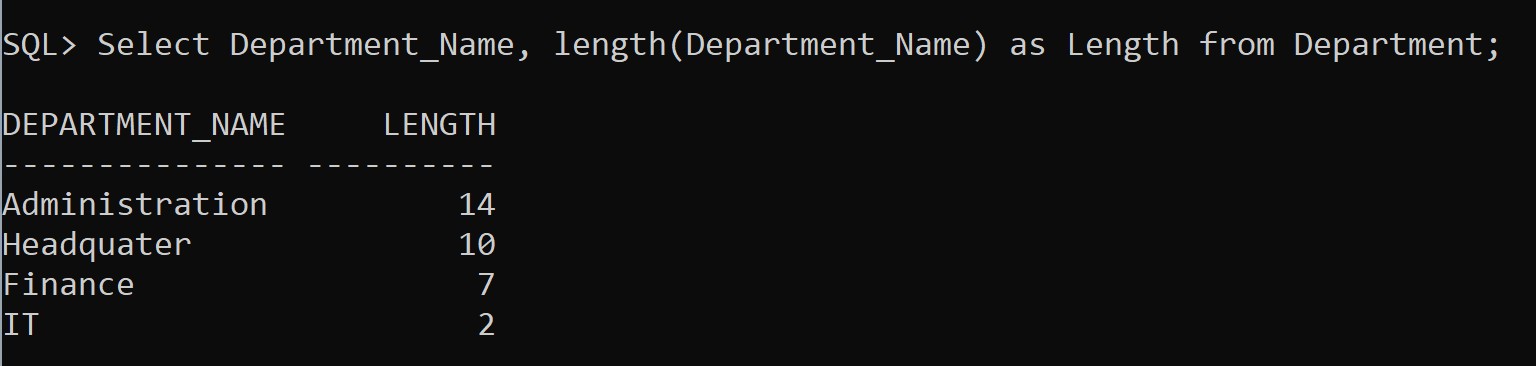


1. **Print the length of all the department names.**

# SQL Command:

Select Department\_Name, length(Department\_Name) as Length from Department;

# Screenshot:

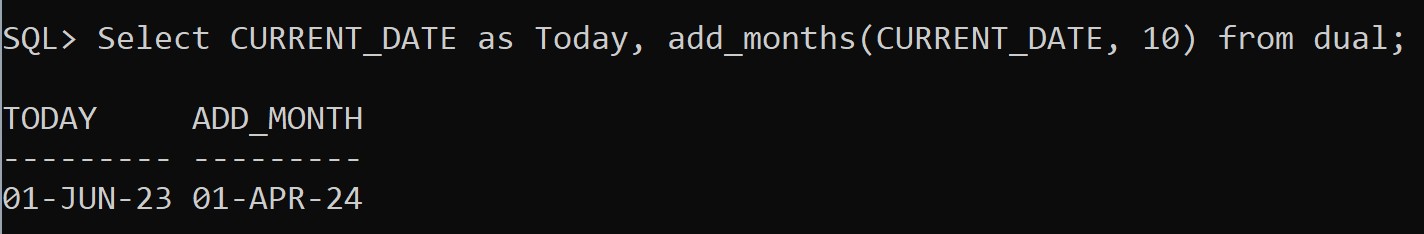


1. **Display the date after 10 months from current date.**

# SQL Command:

Select CURRENT\_DATE as Today, add\_months(CURRENT\_DATE, 10) from dual;

# Screenshot:

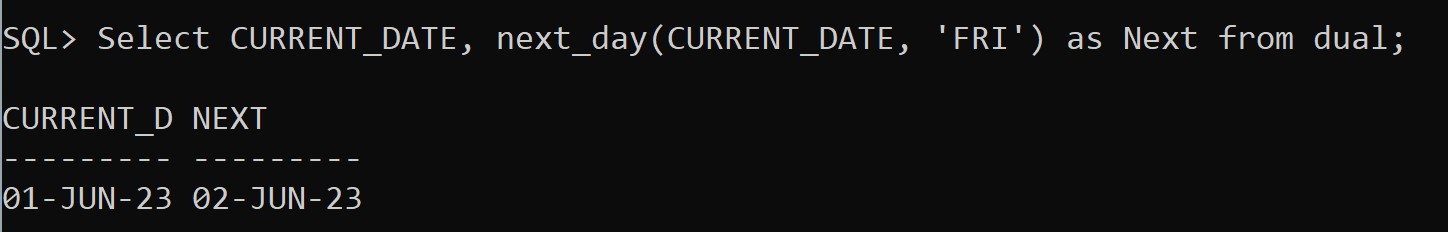


1. **Display the next occurrence of Friday in this month.**

# SQL Command:

Select CURRENT\_DATE, next\_day(CURRENT\_DATE, 'FRI') as Next from dual;

# Screenshot:

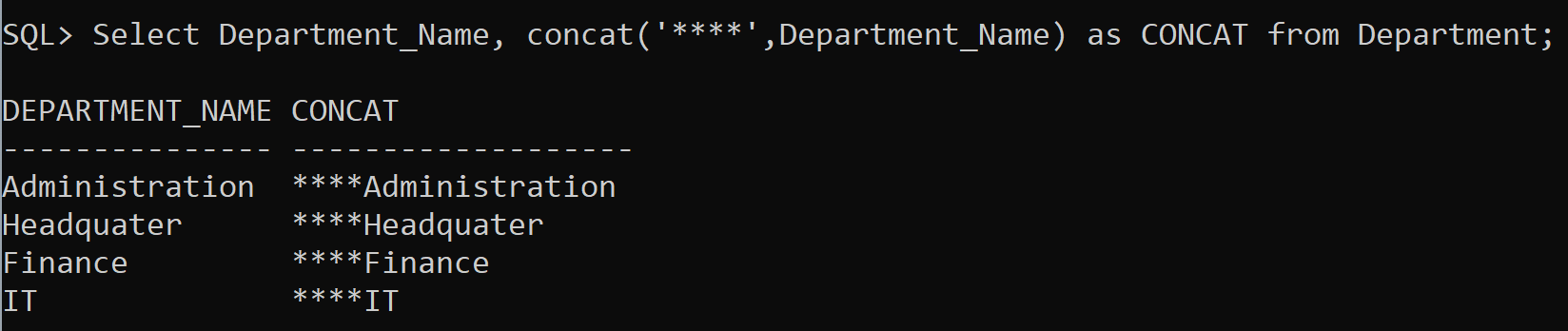


1. **Display the project location padded with \*\*\*\* on left side.**

# SQL Command:

Select Department\_Name, concat('\*\*\*\*',Department\_Name) as CONCAT from Department;

# Screenshot:

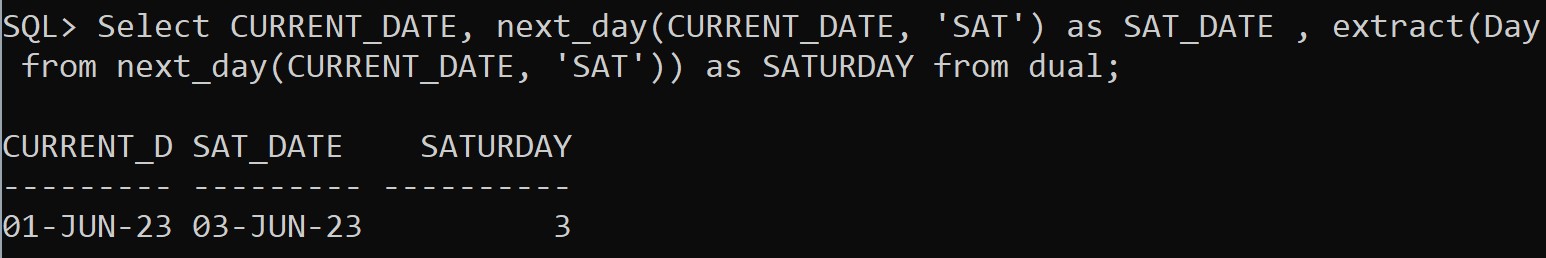


1. **Display the day of next Saturday.**

# SQL Command:

Select CURRENT\_DATE, next\_day(CURRENT\_DATE, 'SAT') as SAT\_DATE , extract(Day from next\_day(CURRENT\_DATE, 'SAT')) as SATURDAY from dual;

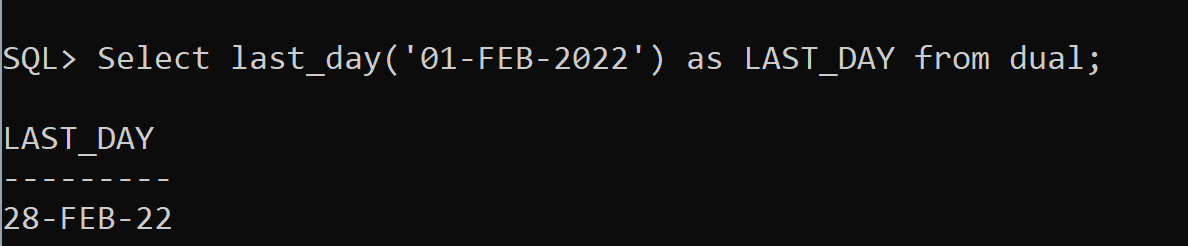
# Screenshot:



1. **Display the last date of Feb 2022 SQL Command:**

Select last\_day('01-FEB-2022') as LAST\_DAY from dual;

# Screenshot:



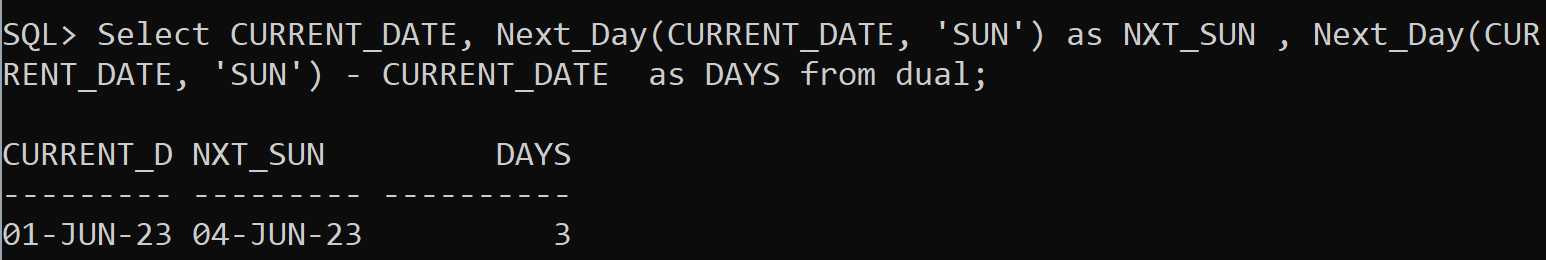
1. **Count the number of days present between today and Sunday.**

# SQL Command:

Select CURRENT\_DATE, Next\_Day(CURRENT\_DATE, 'SUN') as NXT\_SUN , Next\_Day(CURRENT\_DATE, 'SUN') - CURRENT\_DATE as

DAYS from dual;

# Screenshot:

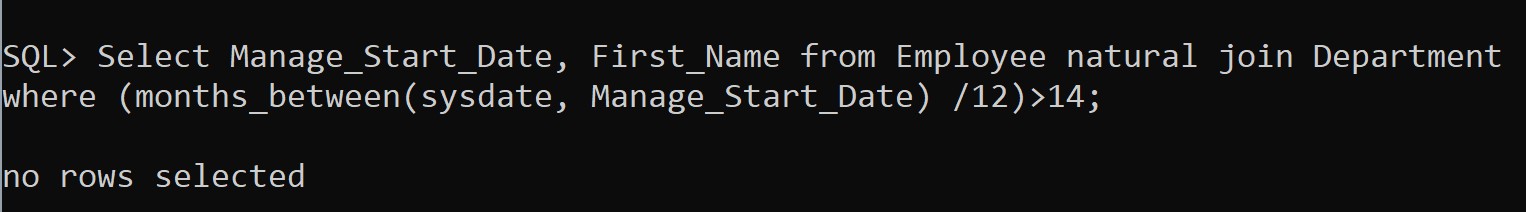


1. **Display all the employees whose experience is greater than 14 yrs.**

# SQL Command:

Select Manage\_Start\_Date, First\_Name from Employee natural join Department where (months\_between(sysdate, Manage\_Start\_Date) /12)>14;

# Screenshot:

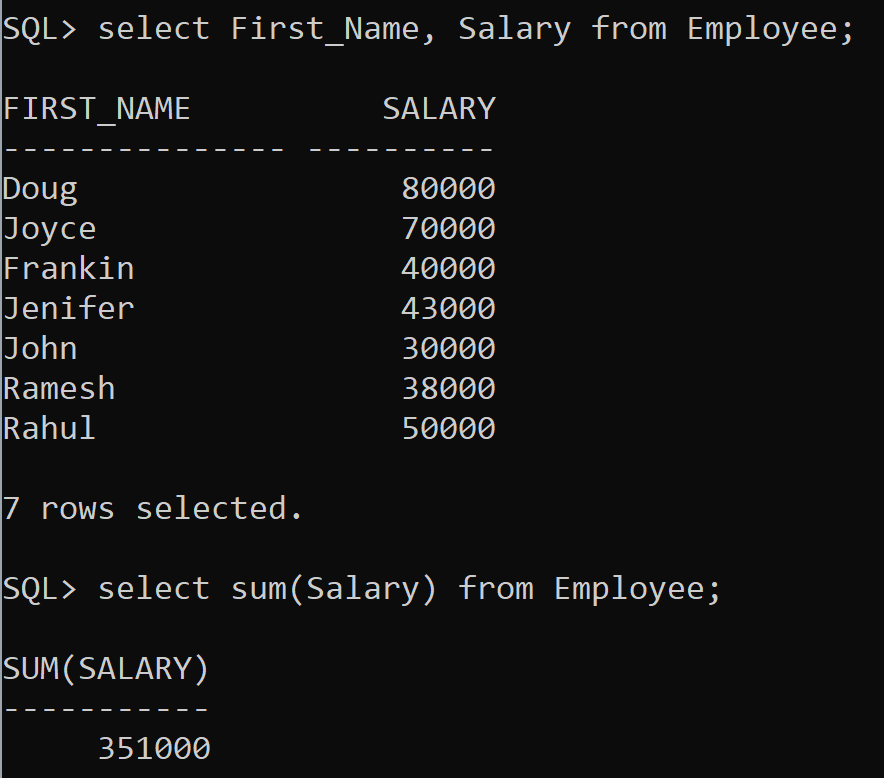


**Group Functions**

# Find the total salary of all employees SQL Command:

select sum(Salary) from Employee;

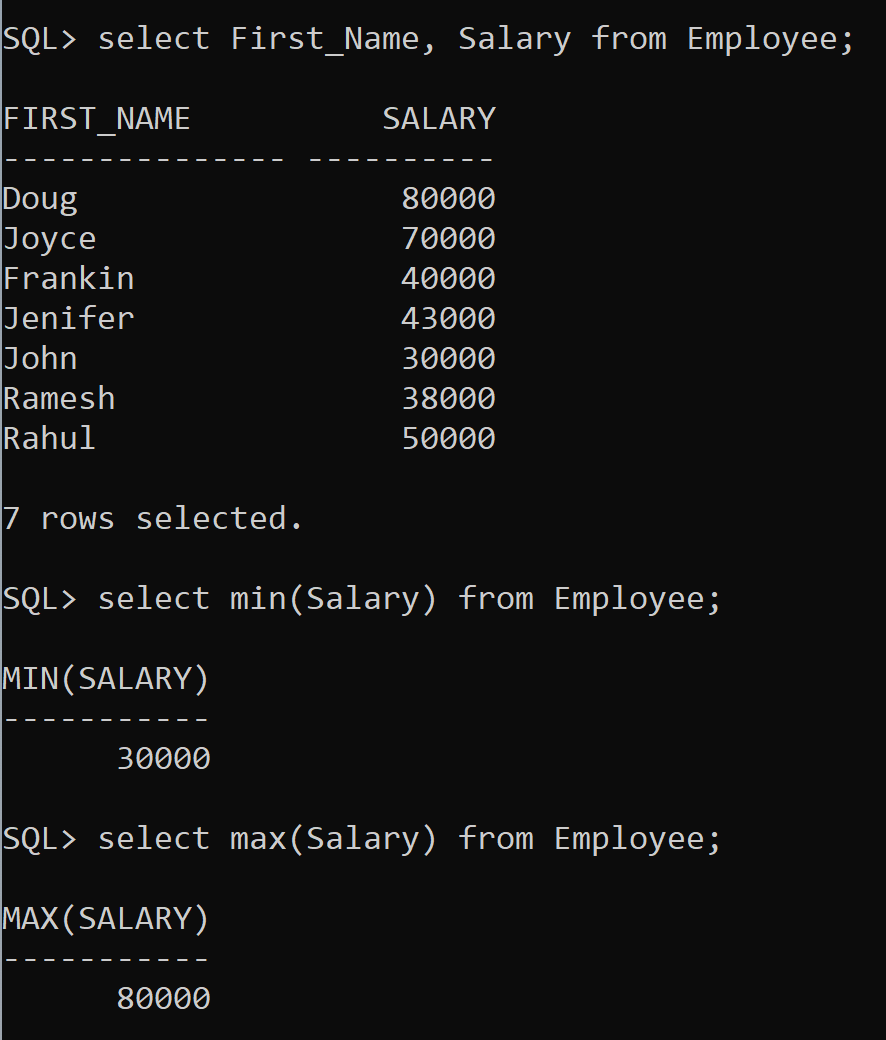
# Screenshot:



1. **Display the min, max salary given to employees SQL Command:**

select min(Salary) from Employee; select max(Salary) from Employee;

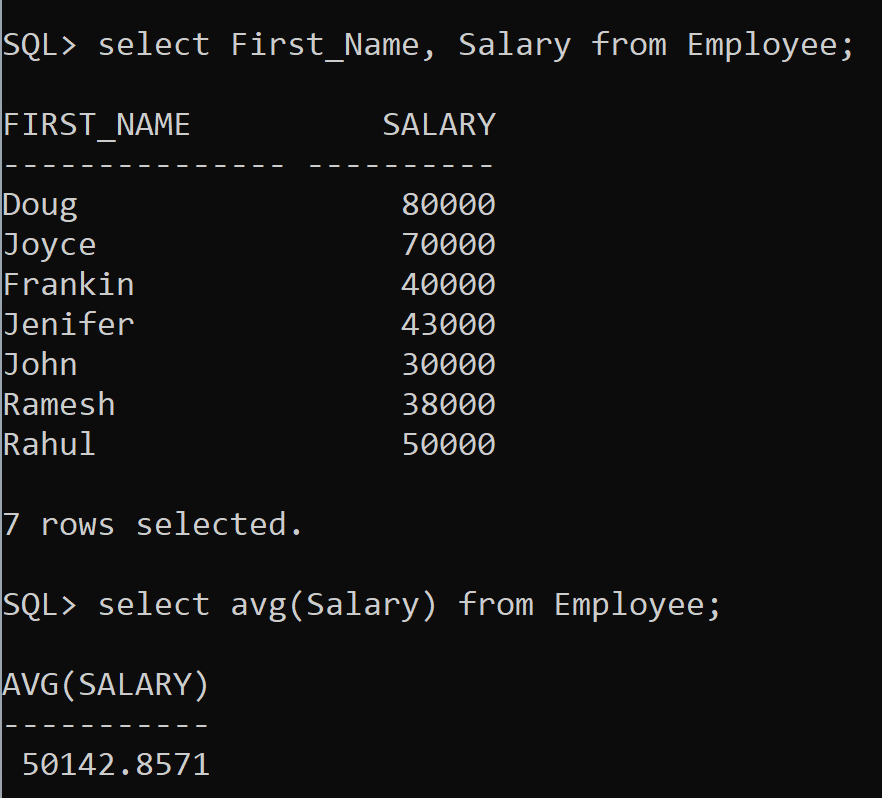
# Screenshot:



1. **Display the avg salary of all employees SQL Command:**

select avg(Salary) from Employee;

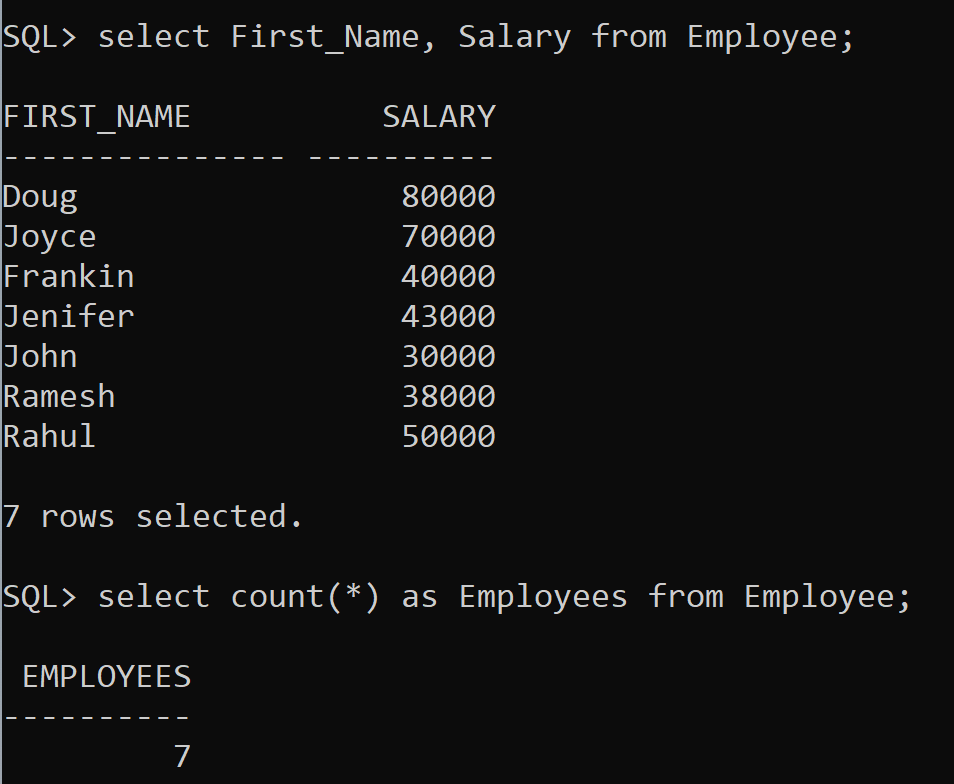
# Screenshot:



1. **Display the total number of employees SQL Command:**

select count(\*) as Employees from Employee;

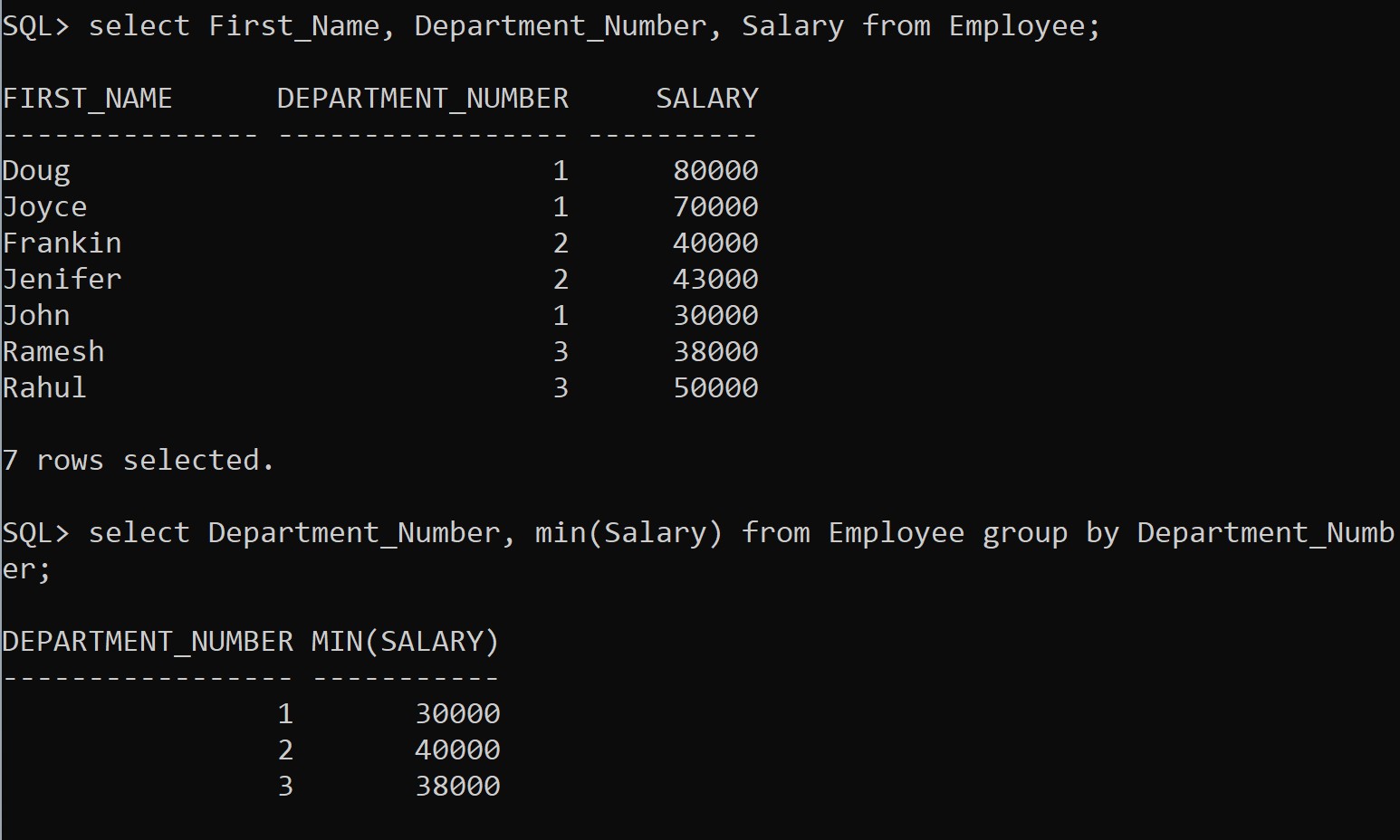
# Screenshot:



1. **How many different departments are there in the ‘employee’ table SQL Command:**

select count(distinct Department\_Number) as No\_Dep from Employee;

# Screenshot:

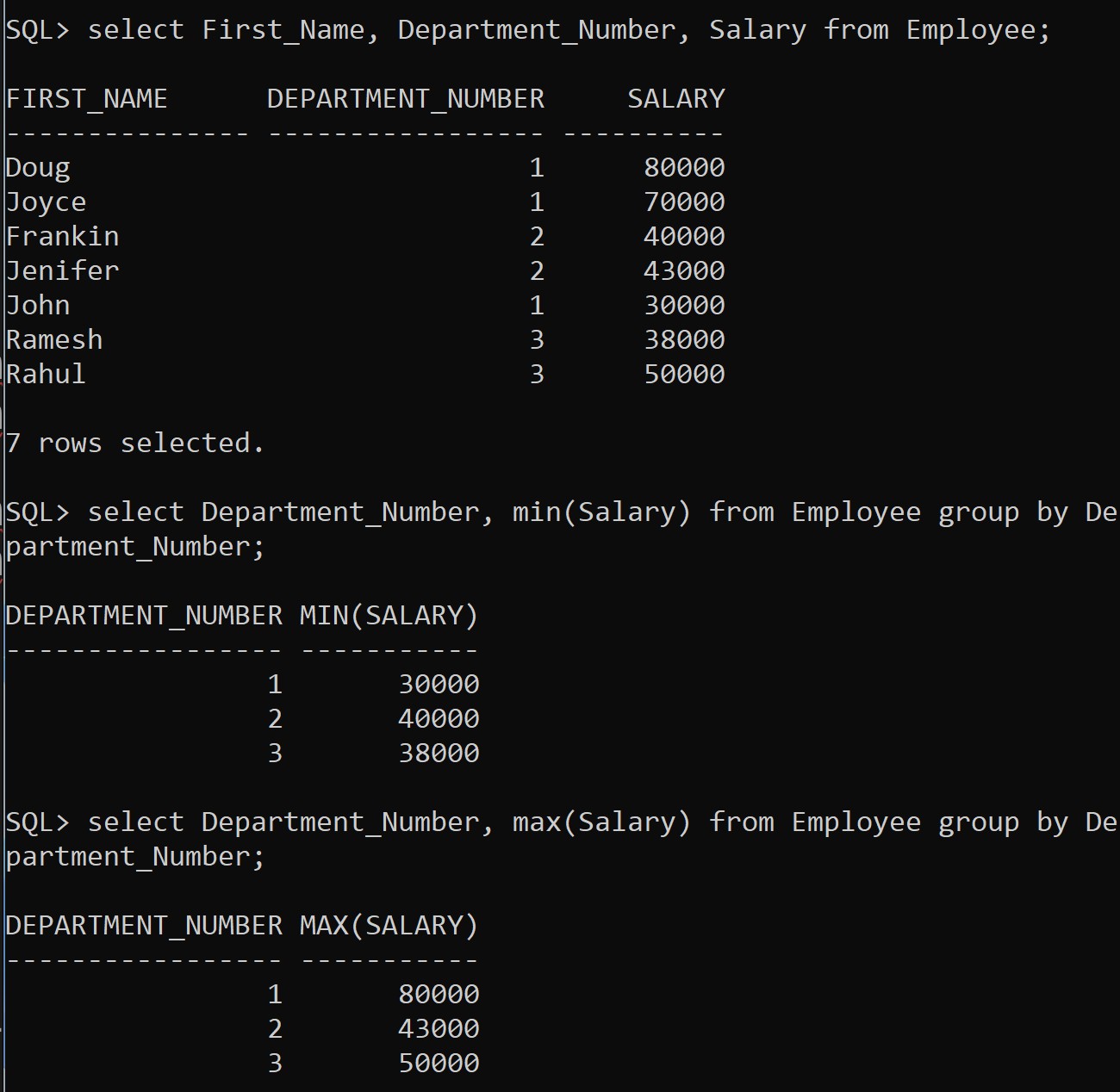


1. **For each department display the minimum and maximum employee salaries SQL Command:**

select Department\_Number, min(Salary) from Employee group by Department\_Number;

select Department\_Number, max(Salary) from Employee group by Department\_Number;

# Screenshot:

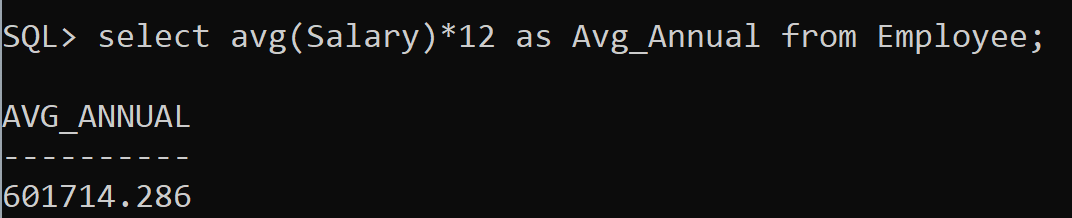


1. **Print the average annual salary.**

# SQL Command:

select avg(Salary)\*12 as Avg\_Annual from Employee;

# Screenshot:

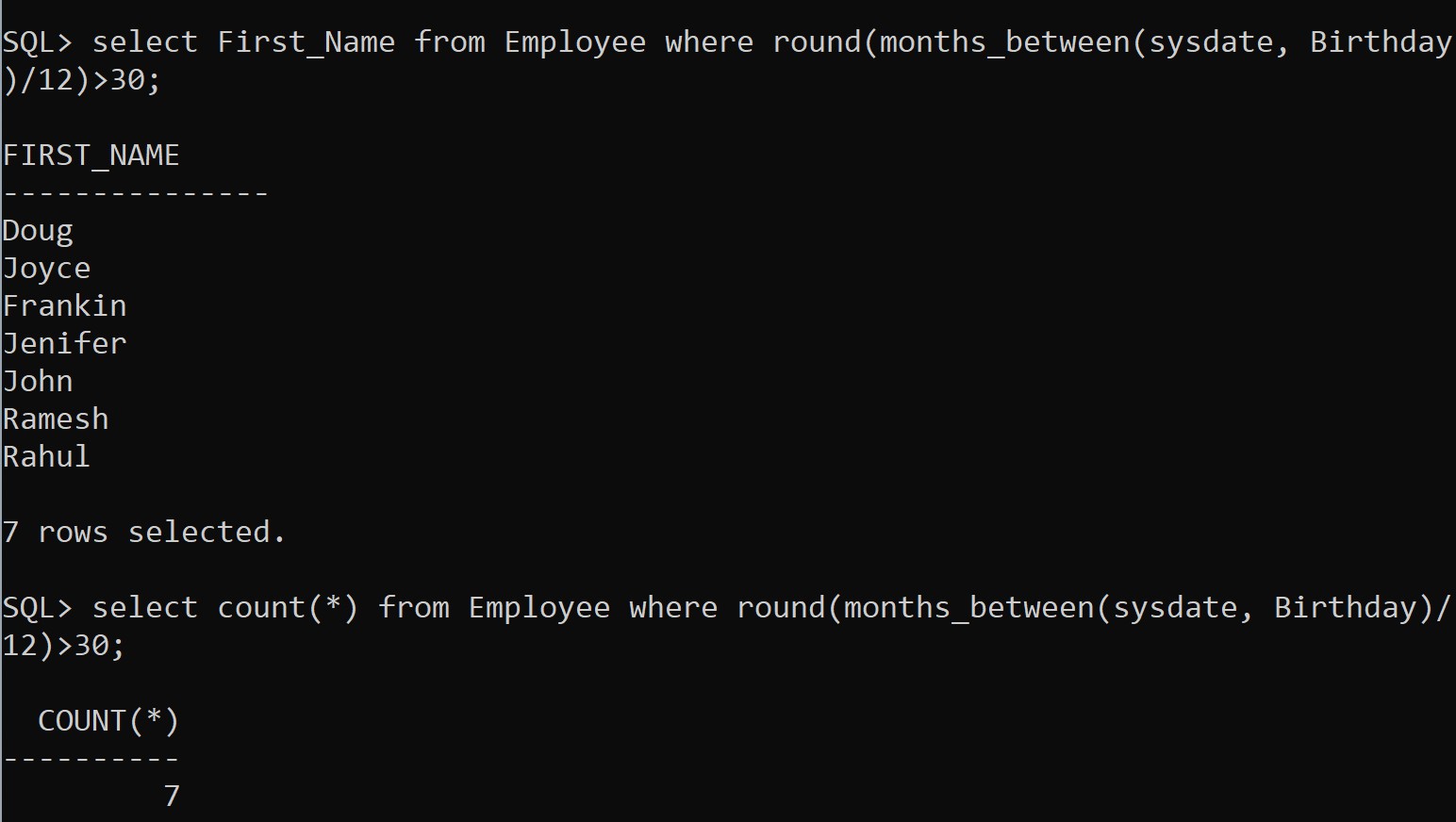


1. **Count the number of employees over 30 age.**

# SQL Command:

select count(\*) from Employee where round(months\_between(sysdate, Birthday)/12)>30;

# Screenshot:

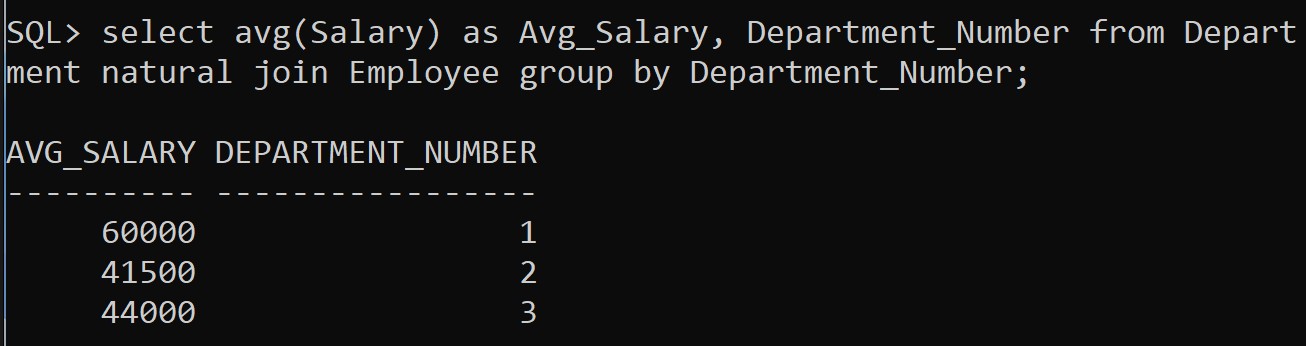


1. **Print the Department name and average salary of each department.**

# SQL Command:

select avg(Salary) as Avg\_Salary, Department\_Number from Department natural join Employee group by Department\_Number;

# Screenshot:

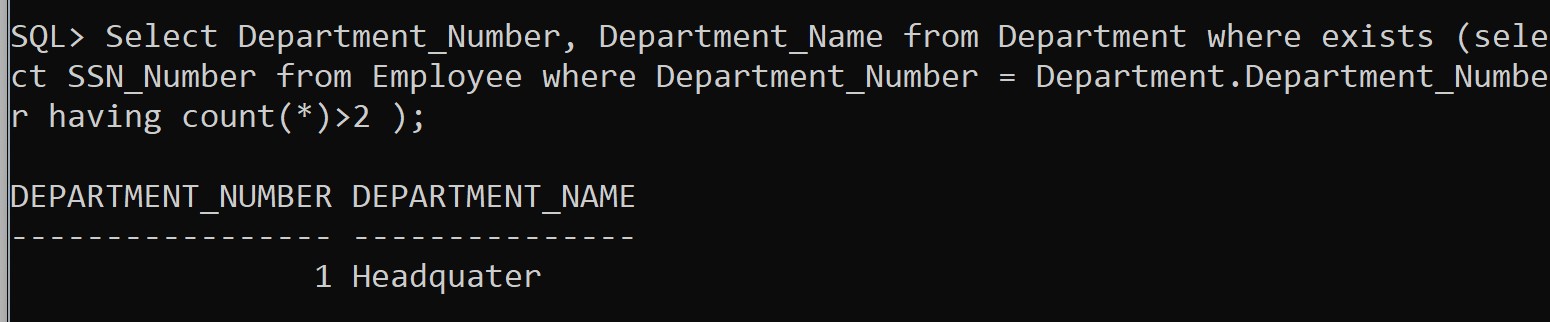


1. **Display the department name which contains more than 2 employees.**

# SQL Command:

Select Department\_Number, Department\_Name,count(\*) from Department where exists (select SSN\_Number from Employee where Department\_Number = Department.Department\_Number having count(\*)>2 );

# Screenshot:

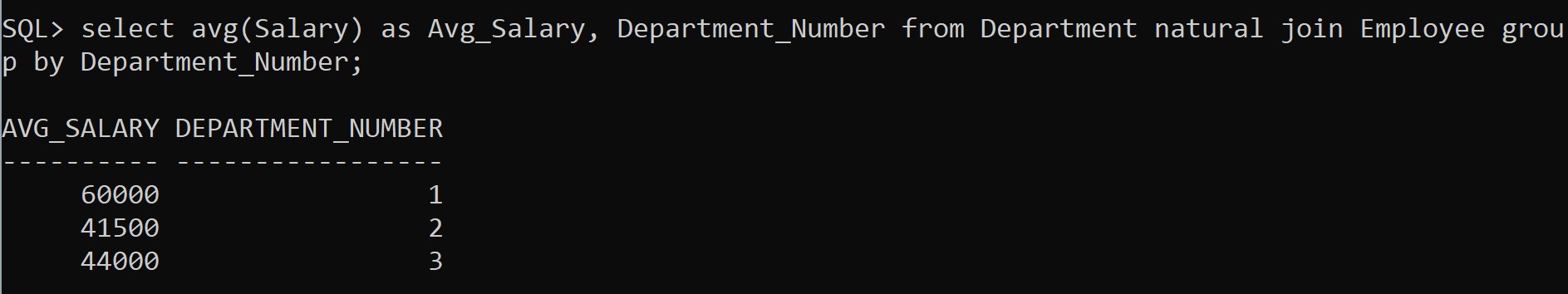


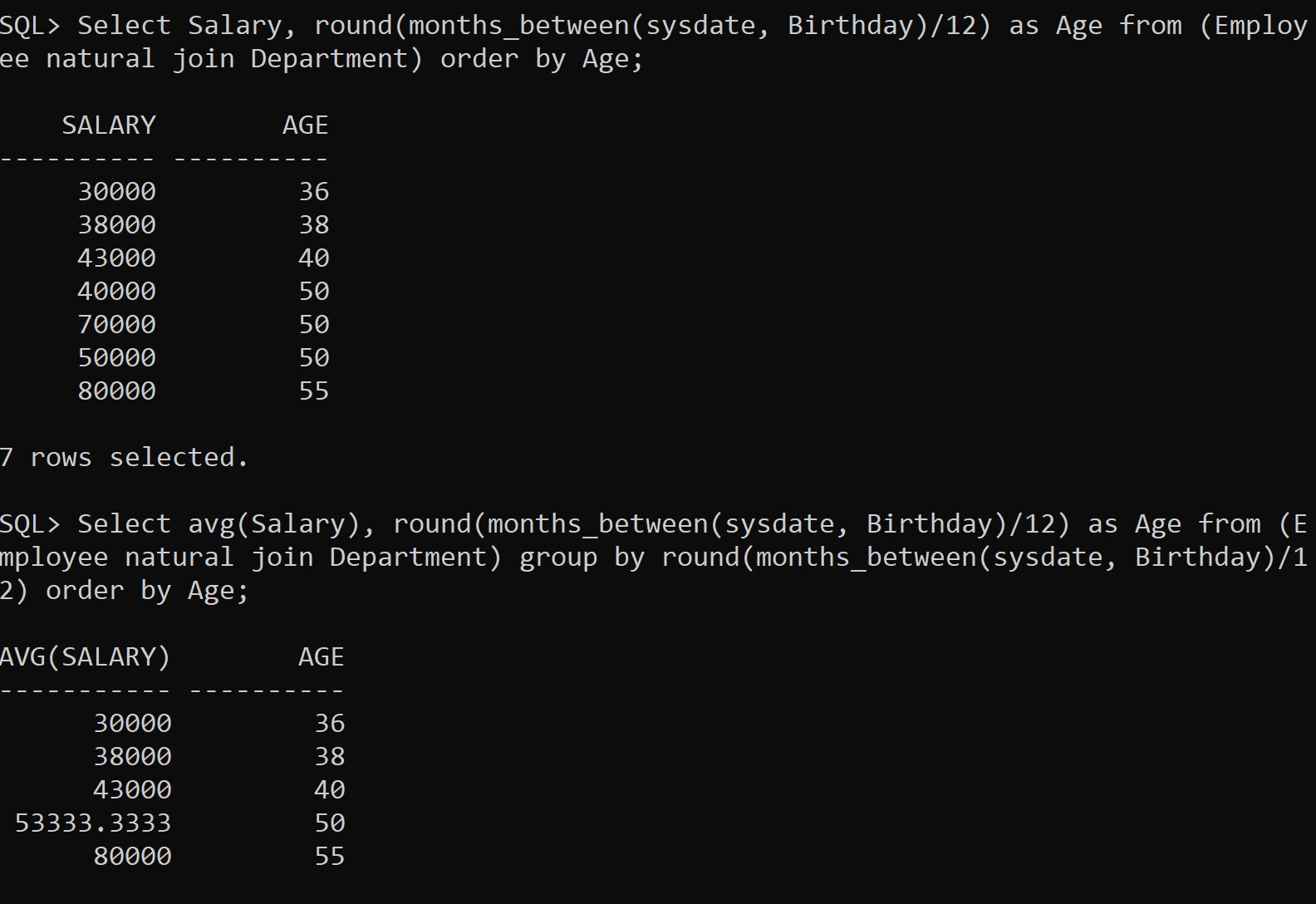
1. **Calculate the average salary of employees by department and age SQL Command:**

Select Department\_Number, avg(Salary) from (Employee natural join Department) group by Department\_Number;

Select avg(Salary), round(months\_between(sysdate, Birthday)/12) as Age from (Employee natural join Department) group by round(months\_between(sysdate, Birthday)/12) order by Age;

# Screenshot:





1. **Count separately the number of employees in the finance and administration department.**

# SQL Command:

Select First\_Name from (Employee natural join Department) where Department\_Name = 'Finance';

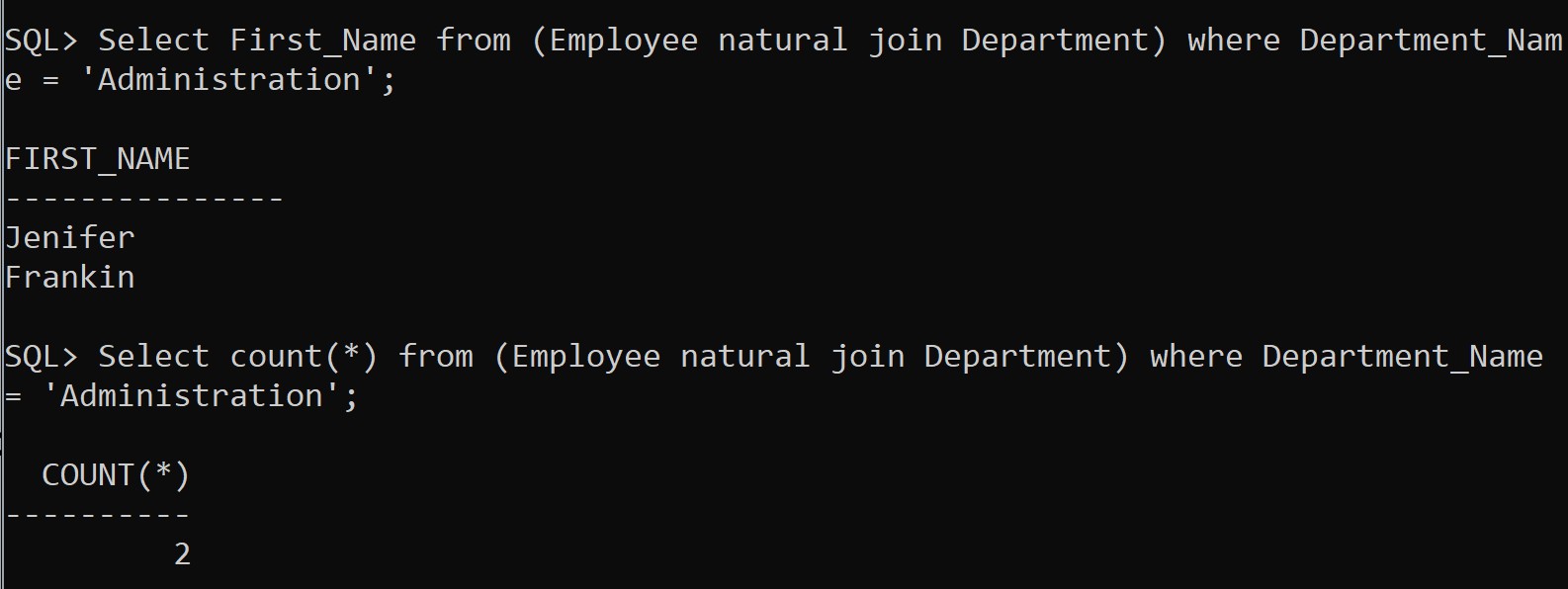
Select count(\*) from (Employee natural join Department) where Department\_Name = 'Finance';

Select First\_Name from (Employee natural join Department) where Department\_Name = 'Administration';

Select count(\*) from (Employee natural join Department) where Department\_Name = 'Administration';

# Screenshot:



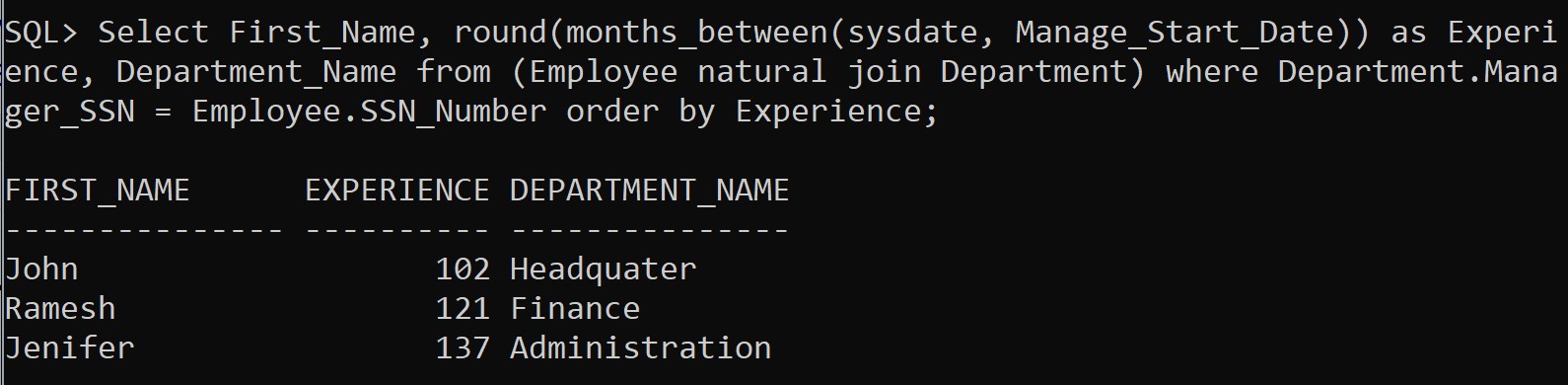


1. **List out the employees based on their seniority.**

# SQL Command:

Select First\_Name, round(months\_between(sysdate, Manage\_Start\_Date)) as Experience, Department\_Name from (Employee natural join Department) where Department.Manager\_SSN = Employee.SSN\_Number order by Experience;

# Screenshot:



1. **Display the total number of employees who are assigned to a department.**

# SQL Command:

Select Department\_Number, count(\*) from (Department natural join Employee) group by Department\_Number;

# Screenshot:

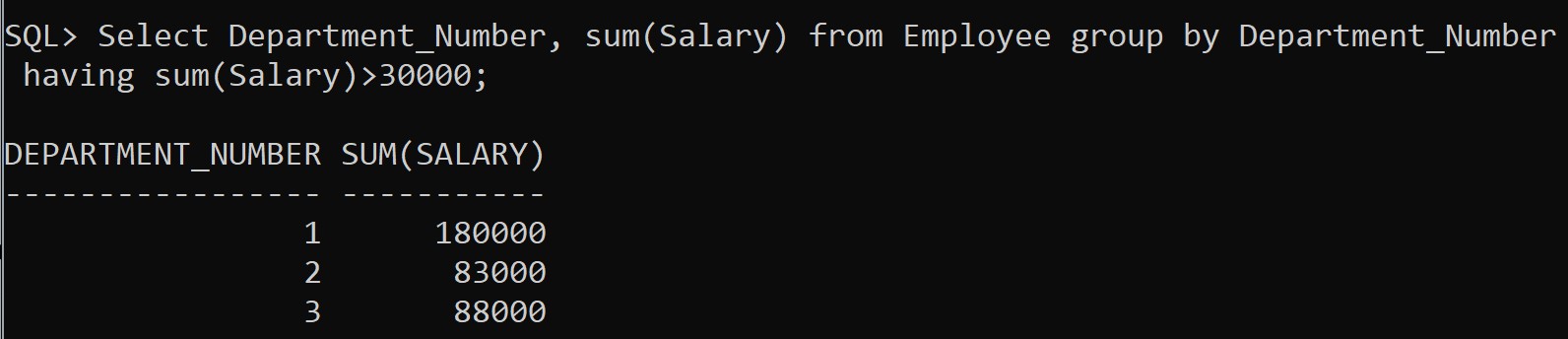


1. **Display only department numbers that are paying a total salary of more than 30000 to its employees**

# SQL Command:

Select Department\_Number, sum(Salary) from Employee group by Department\_Number having sum(Salary)>30000;

# Screenshot:

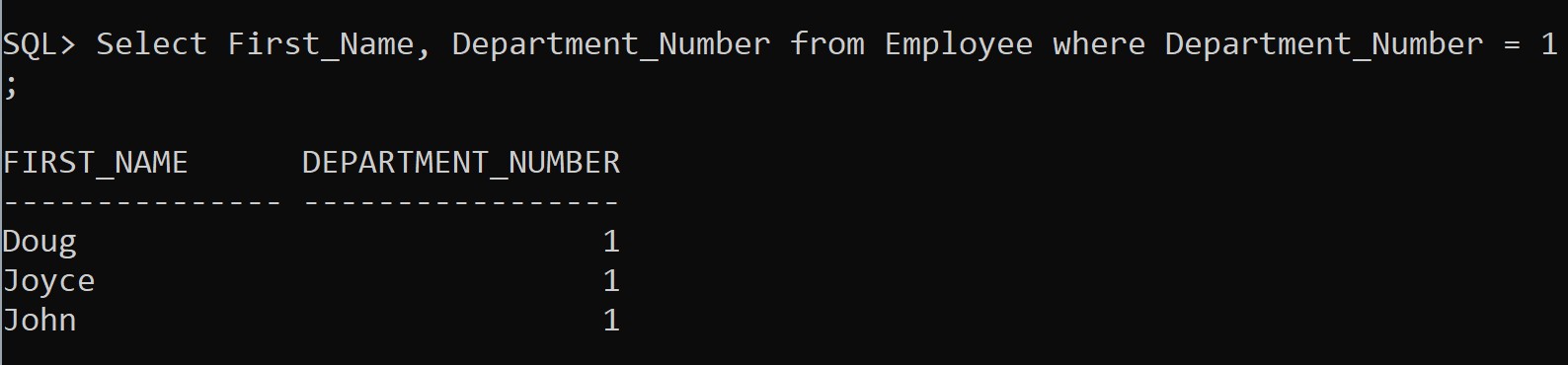


1. **Display all the employees working under the department 1.**

# SQL Command:

Select First\_Name, Department\_Number from Employee where Department\_Number = 1;1

# Screenshot:



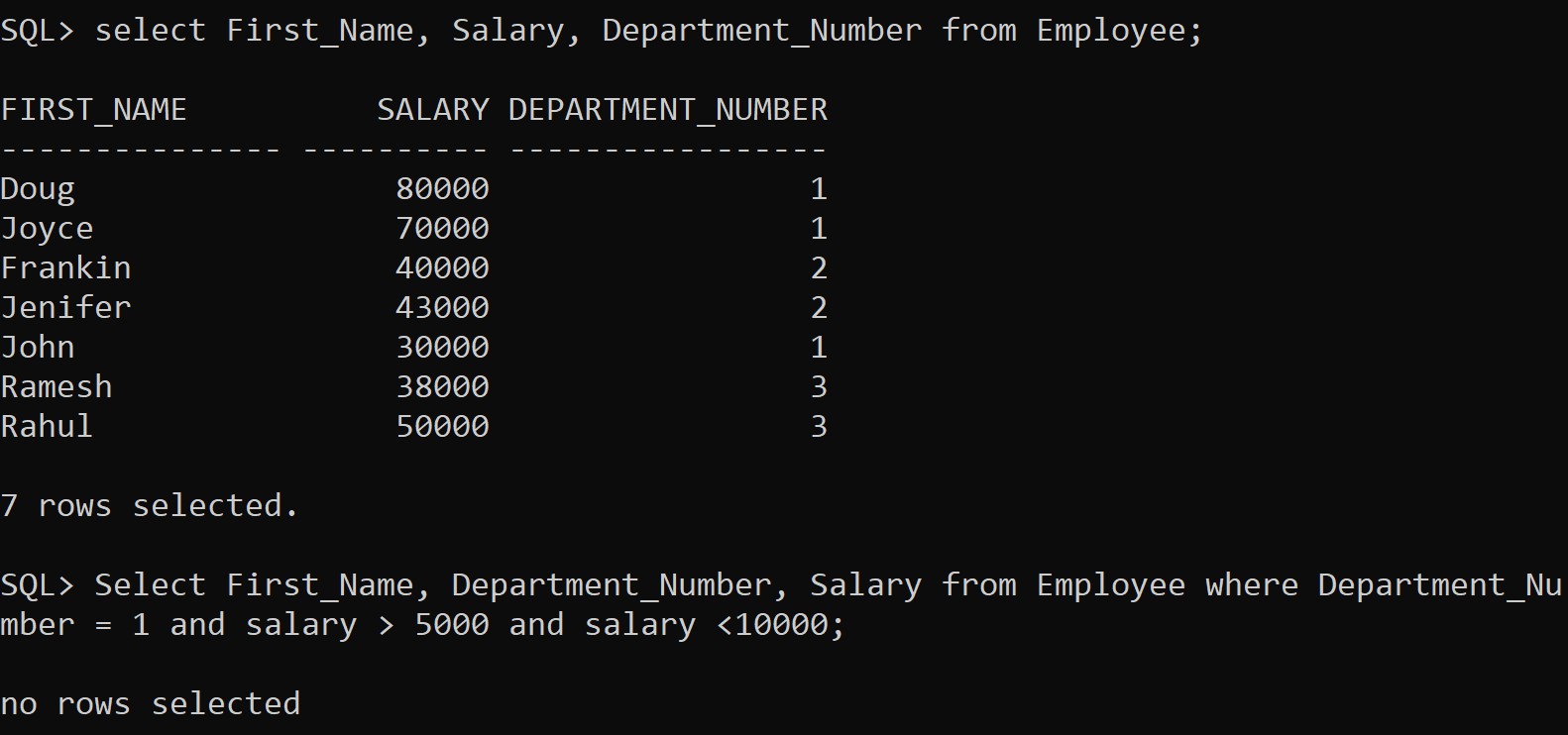
1. **Display employees whose salary is between 5000 and 10000 in department no 1.**

# SQL Command:

Select First\_Name, Department\_Number, Salary from Employee where Department\_Number = 1 and salary > 5000 and salary

<10000;

# Screenshot:



**SUB QUERY AND VIEW IN SQL**

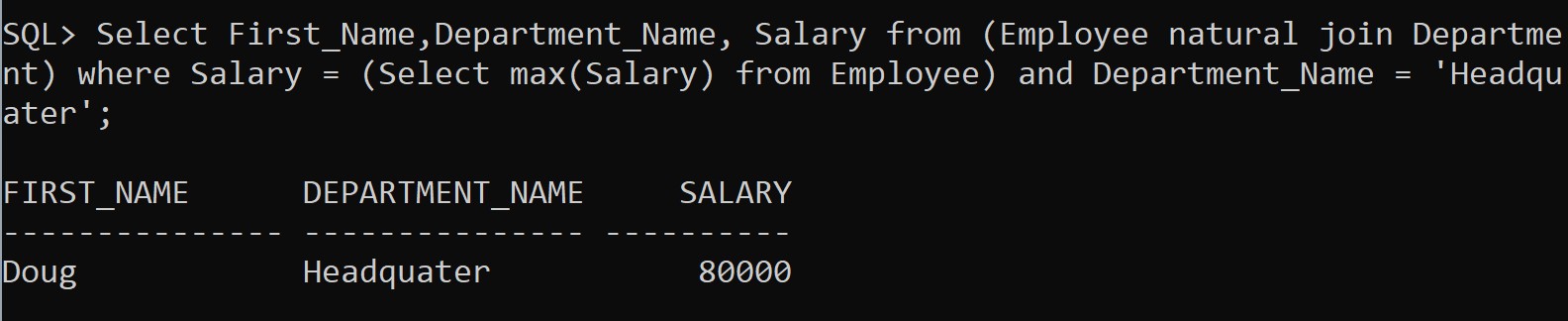
# Aim: To understand different types of function in SQL Consider the schema (used in EX2):

1. **Find the employee who is getting highest salary in the department headquarters.**

# SQL Command:

Select First\_Name,Department\_Name, Salary from (Employee natural join Department) where Salary = (Select max(Salary) from Employee) and Department\_Name = 'Headquater';

# Screenshot:

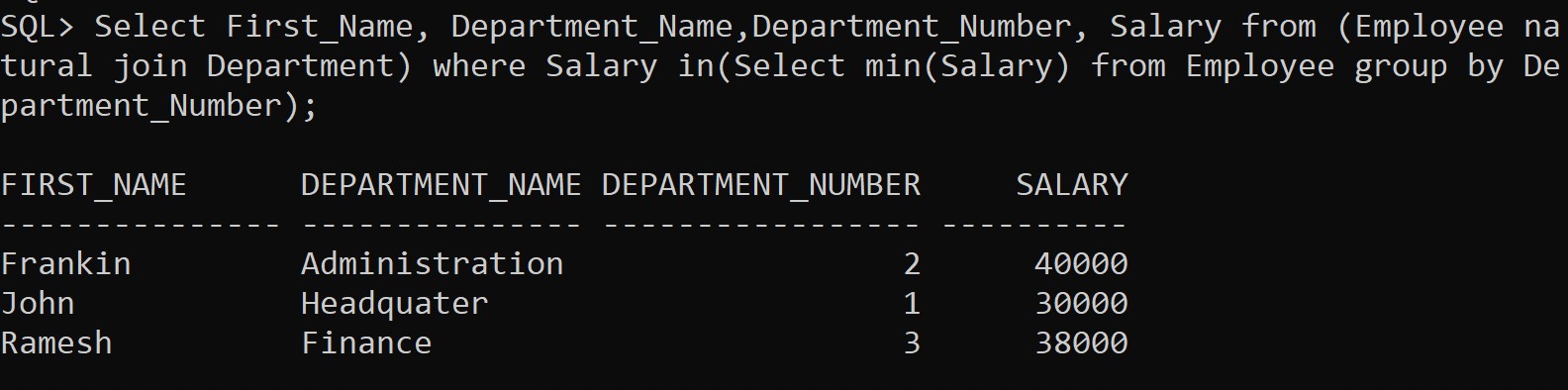


1. **Find the employees who earn the same salary as the minimum salary for each Department.**

# SQL Command:

Select First\_Name, Department\_Name,Department\_Number, Salary from (Employee natural join Department) where Salary in(Select min(Salary) from Employee group by Department\_Number);

# Screenshot:



1. **Find the employee whose salary is greater than average salary of department 2.**

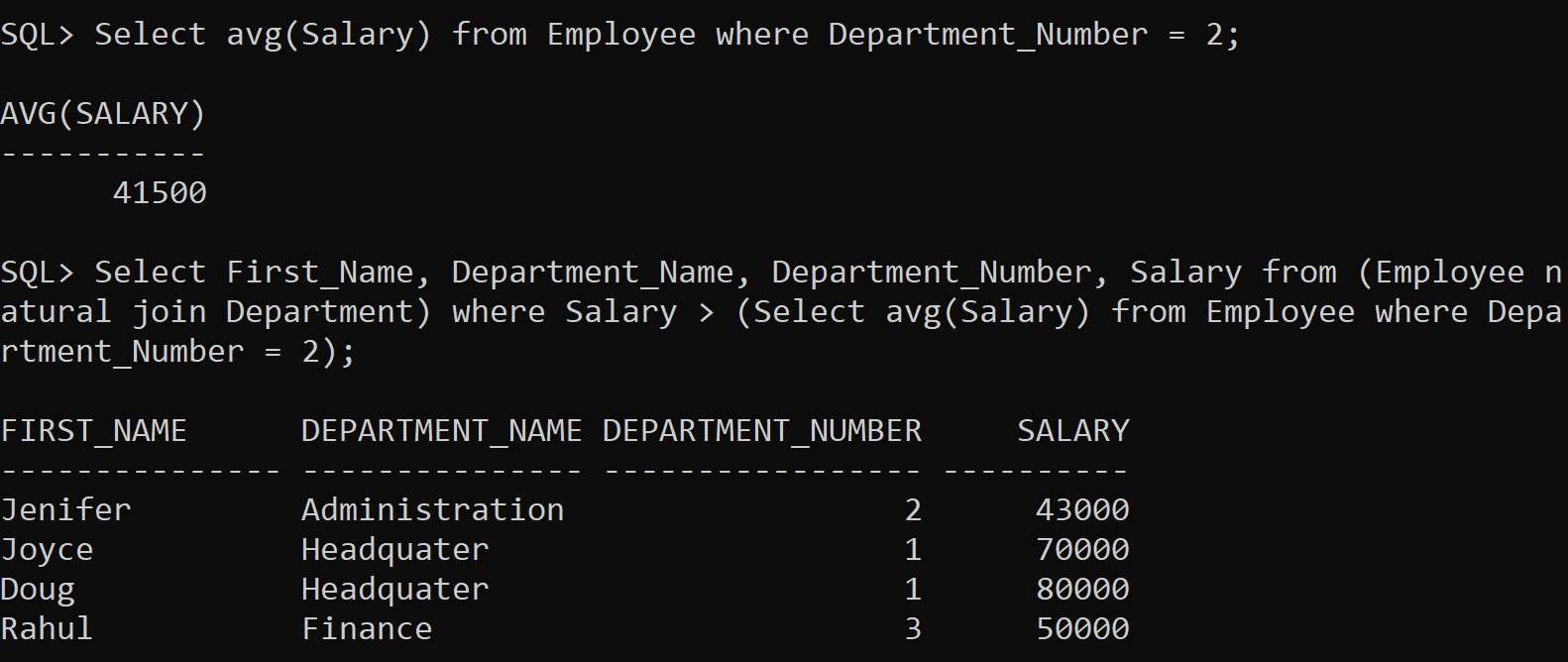
# SQL Command:

Select First\_Name, Department\_Name, Department\_Number, Salary from (Employee natural join Department) where Salary

> (Select avg(Salary) from Employee where Department\_Number

= 2);

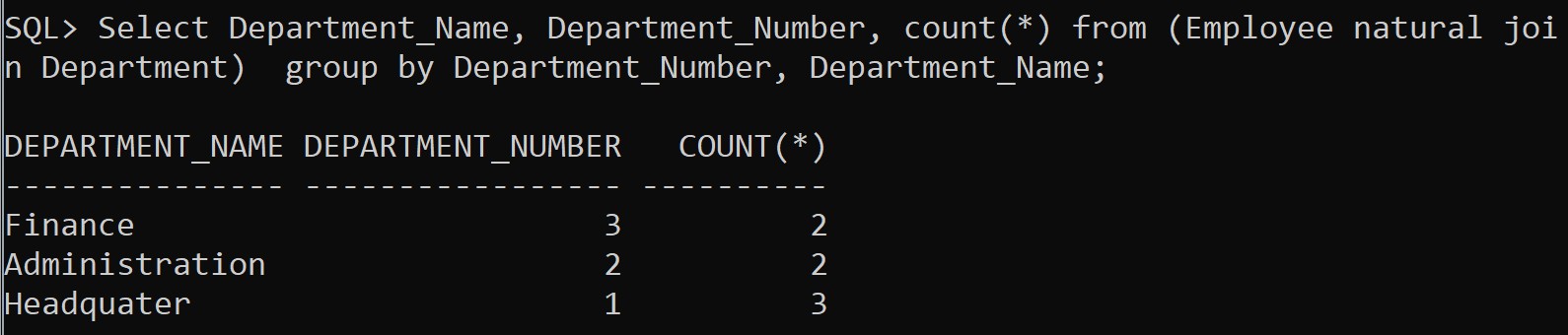
# Screenshot:



1. **List out all the department names with their individual employee’s strength. SQL Command:**

Select Department\_Name, Department\_Number, count(\*) from (Employee natural join Department) group by Department\_Number, Department\_Name;

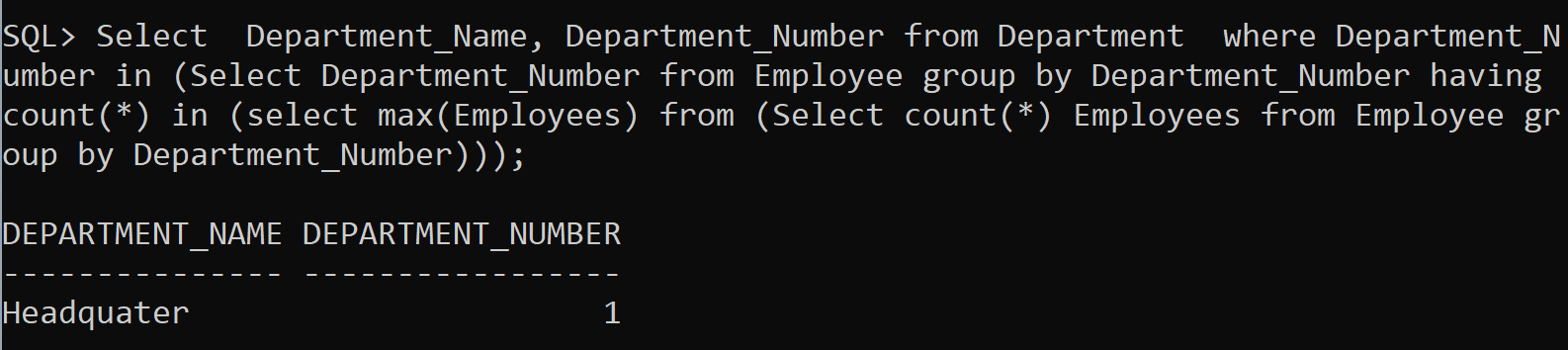
# Screenshot:



1. **Find out the department name having highest employee strength. SQL Command:**

Select Department\_Name, Department\_Number from Department where Department\_Number in (Select Department\_Number from Employee group by Department\_Number having count(\*) in (select max(Employees) from (Select count(\*) Employees from Employee group by Department\_Number)));

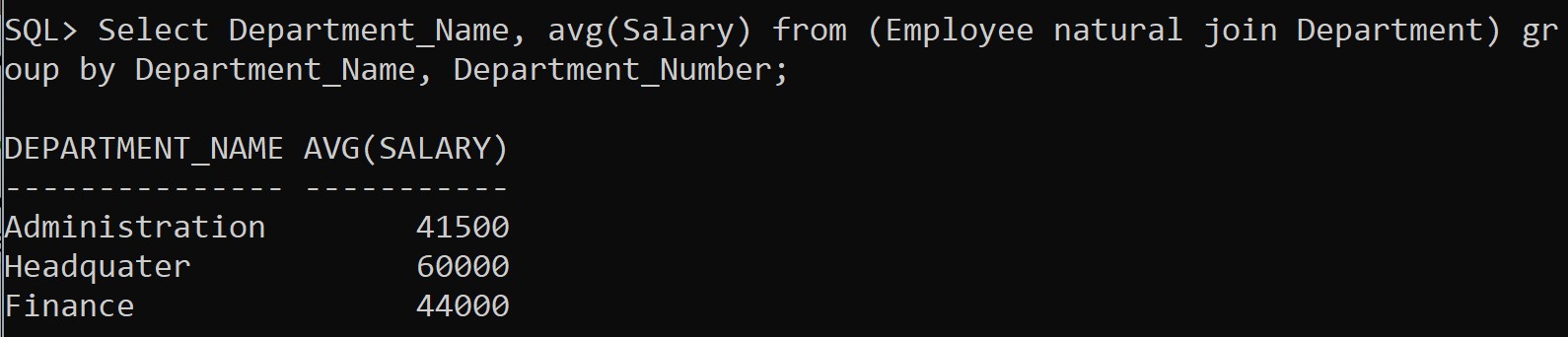
# Screenshot:



1. **List out all the departments and average salary drawn by their employees. SQL Command:**

Select Department\_Name, avg(Salary) from (Employee natural join Department) group by Department\_Name, Department\_Number;

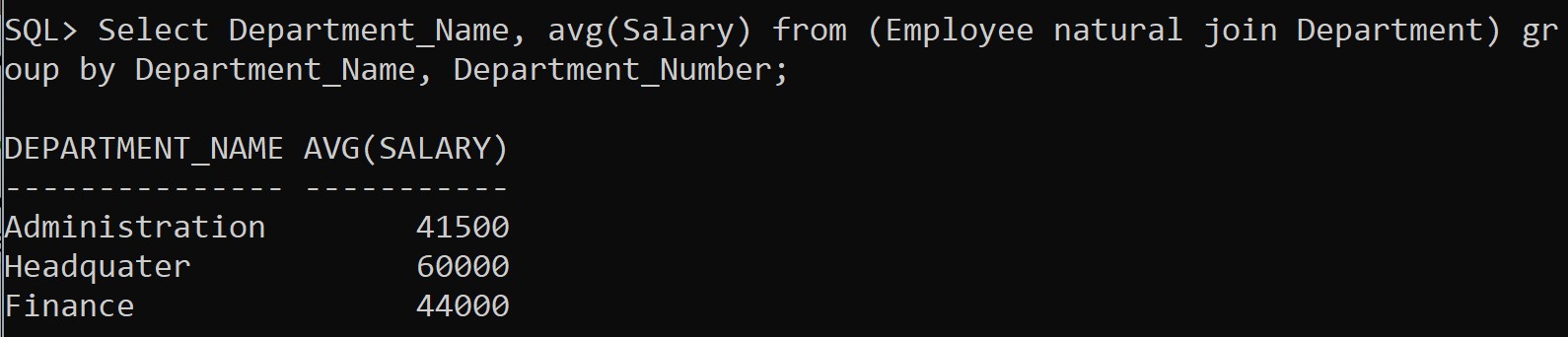
# Screenshot:



1. **Find average salary for each department. SQL Command:**

Select Department\_Name, avg(Salary) from (Employee natural join Department) group by Department\_Name, Department\_Number;

# Screenshot:



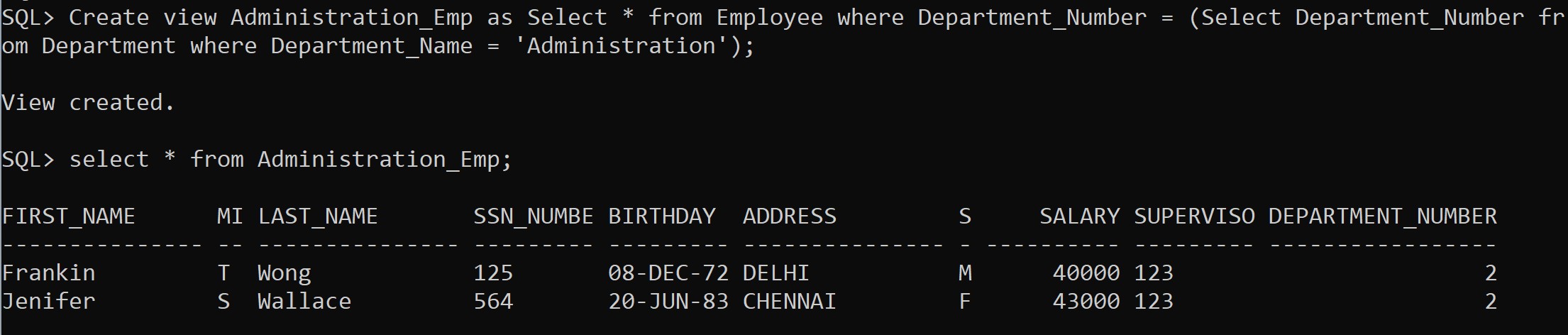
1. **Create a view to display the employee details who is working in Administration department.**

# SQL Command:

Create view Administration\_Emp as Select \* from Employee where Department\_Number = (Select Department\_Number from Department where Department\_Name = 'Administration');

Select \* from Administration\_Emp;

# Screenshot:



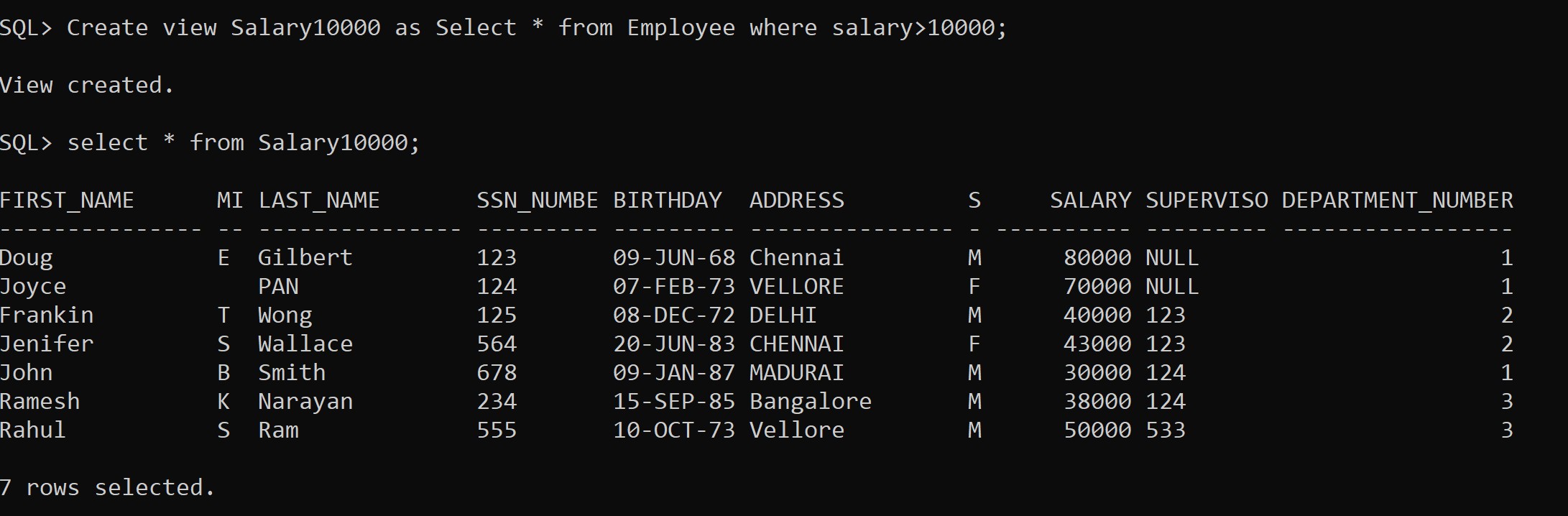
1. **Create a logical table to store employee details who is getting salary more than 10000.**

# SQL Command:

Create view Salary10000 as Select \* from Employee where salary>10000;

select \* from Salary10000;

# Screenshot:



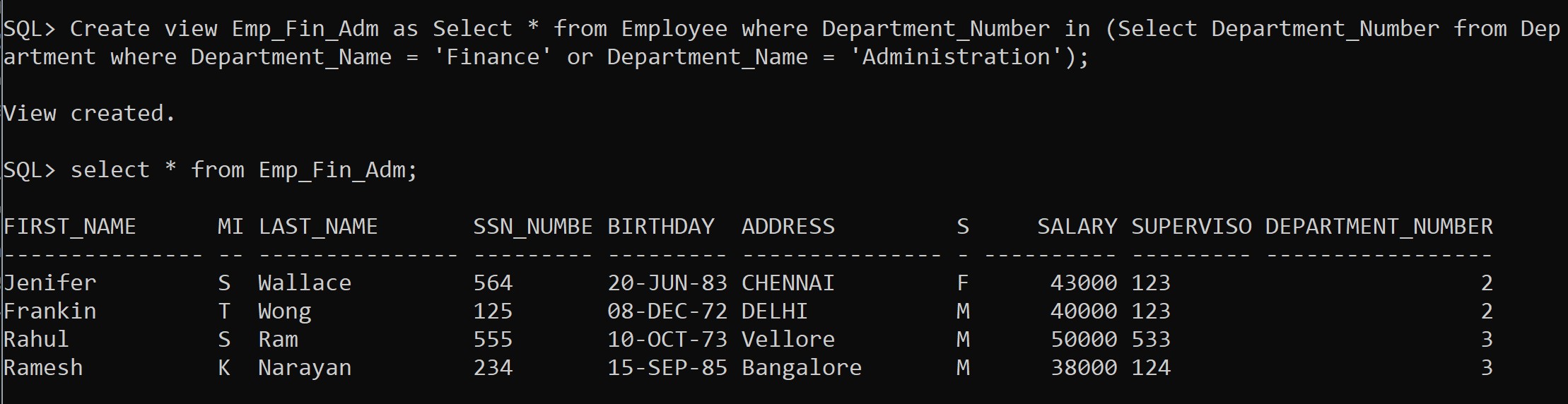
1. **Create a view to display the employee details who is working in finance or administration department.**

# SQL Command:

Create view Emp\_Fin\_Adm as Select \* from Employee where Department\_Number in (Select Department\_Number from Department where Department\_Name = 'Finance' or Department\_Name = 'Administration');

select \* from Emp\_Fin\_Adm;

# Screenshot:



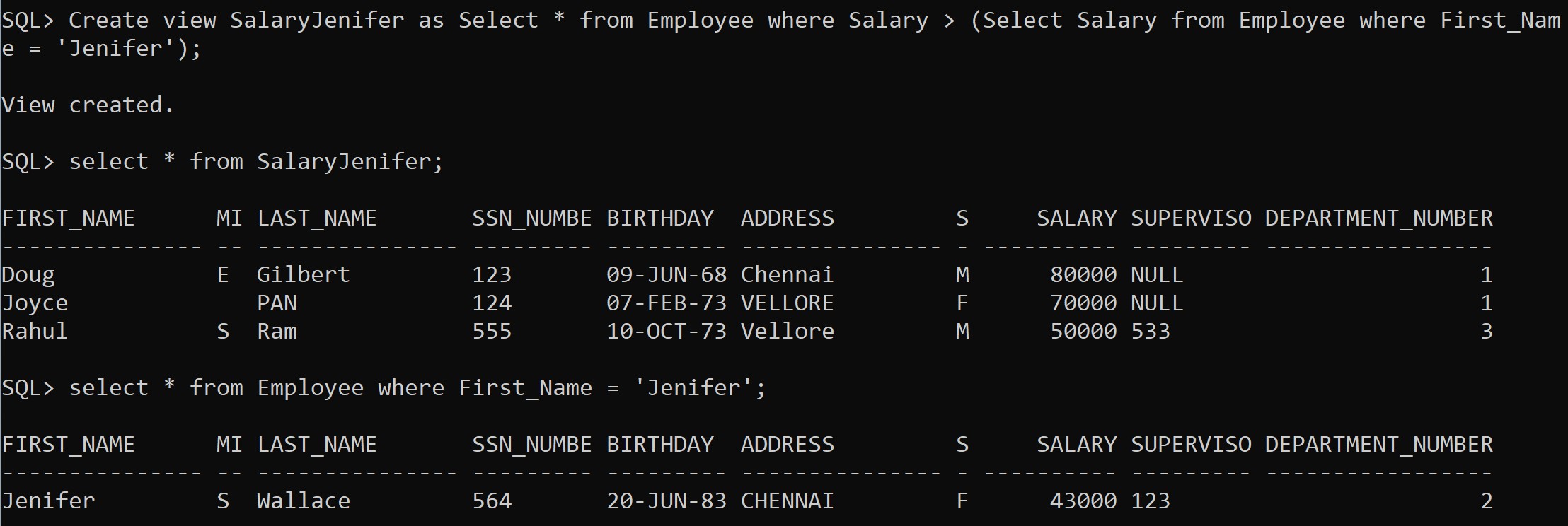
1. **Create a view to display all employees whose salary is greater than “Jennifer”.**

# SQL Command:

Create view SalaryJenifer as Select \* from Employee where Salary > (Select Salary from Employee where First\_Name = 'Jenifer');

select \* from SalaryJenifer;

# Screenshot:



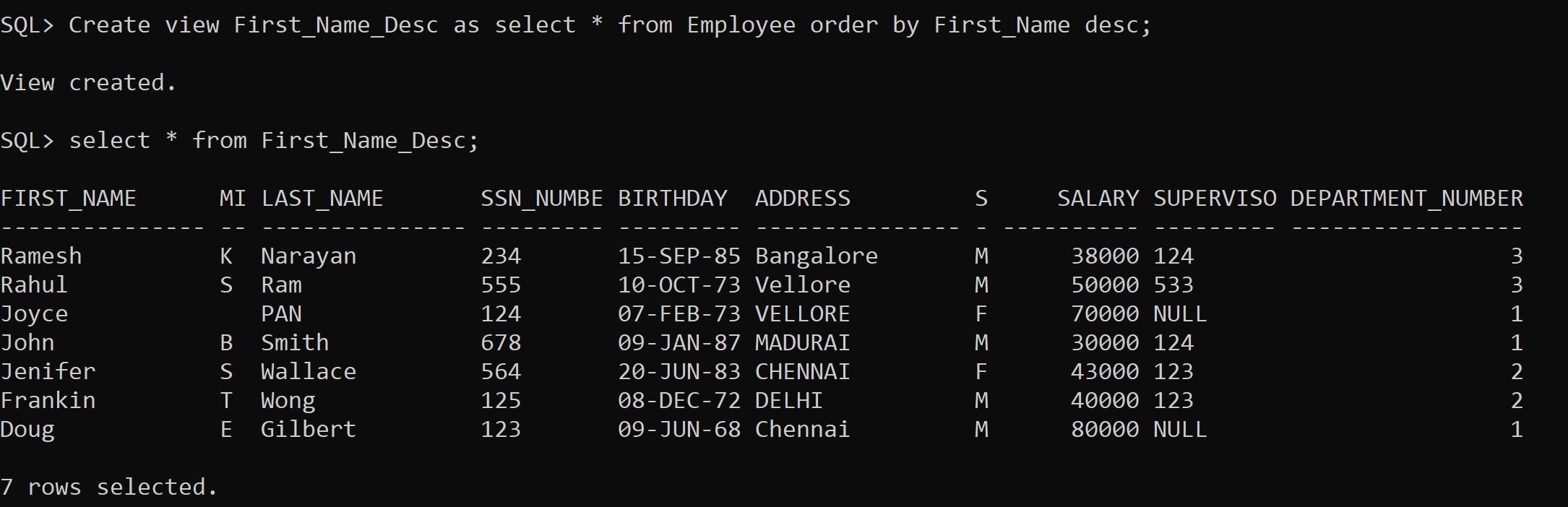
1. **Create a view to display all employees in descending order of employee first name**

# SQL Command:

Create view First\_Name\_Desc as select \* from Employee order by First\_Name desc;

select \* from First\_Name\_Desc;

# Screenshot:



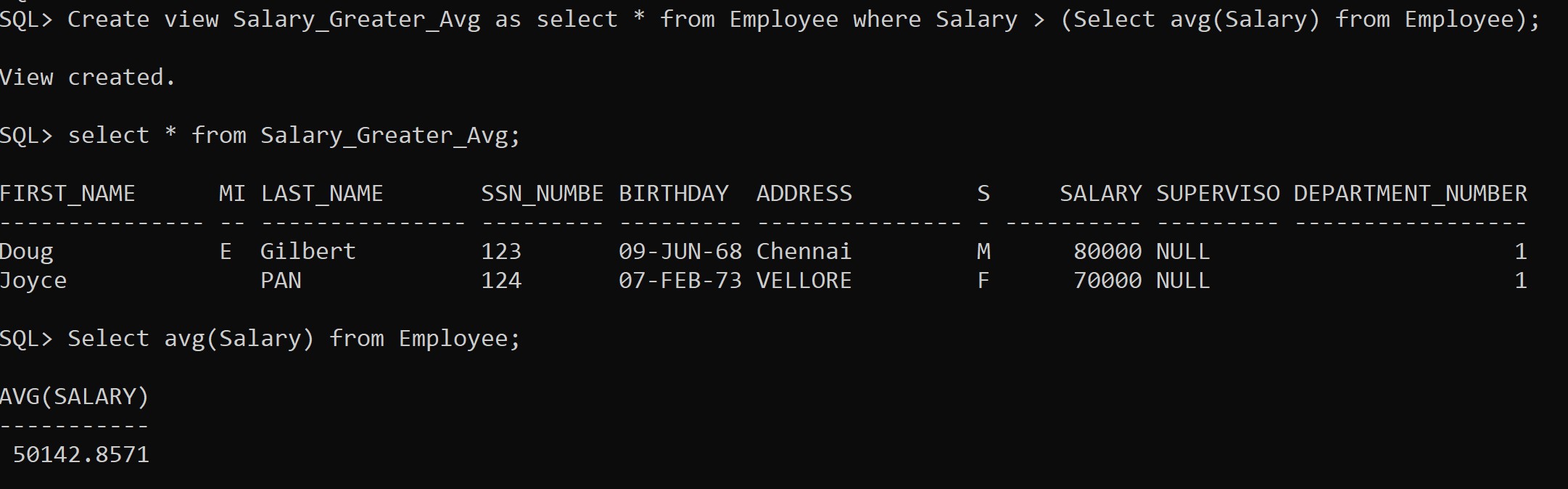
1. **Create a view to display all employees whose salary is greater than the average salary of the employees.**

# SQL Command:

Create view Salary\_Greater\_Avg as select \* from Employee where Salary > (Select avg(Salary) from Employee);

select \* from Salary\_Greater\_Avg;

# Screenshot:



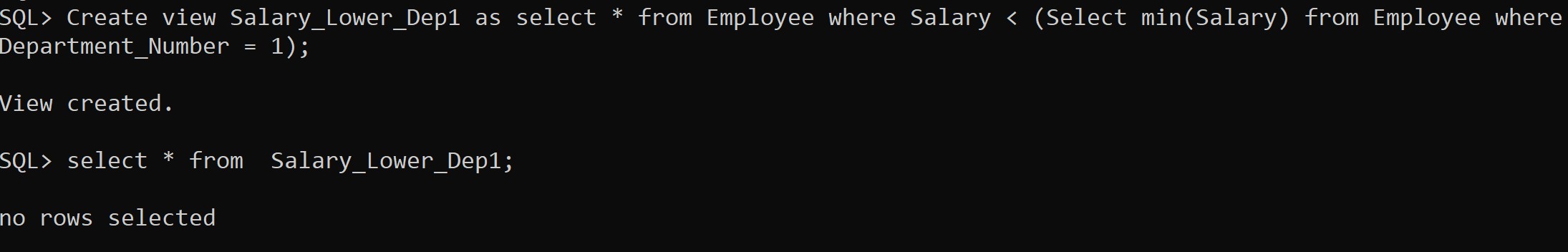
1. **Create a view to display all employees whose salary is lower than all the employees of the department 1.**

# SQL Command:

Create view Salary\_Lower\_Dep1 as select \* from Employee where Salary < (Select min(Salary) from Employee where Department\_Number = 1);

Select \* from Salary\_Lower\_Dep1;

# Screenshot:

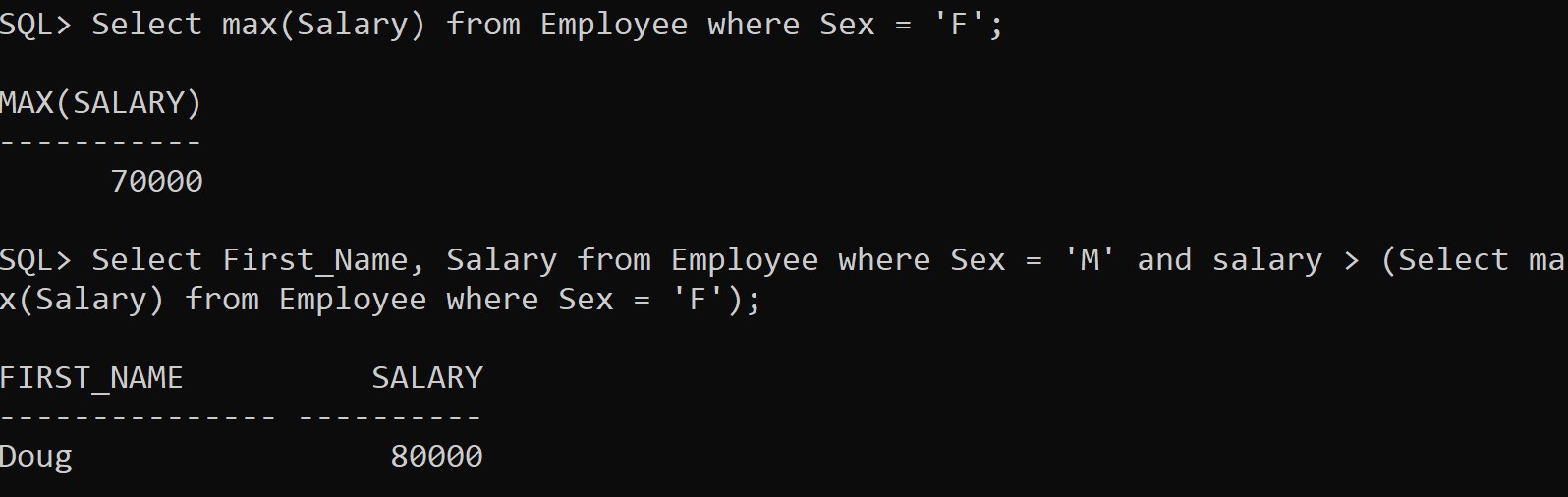


1. **Display all male employees whose salary is greater than the maximum salary of the female employees**

# SQL Command:

Select First\_Name, Salary from Employee where Sex = 'M' and salary > (Select max(Salary) from Employee where Sex = 'F');

# Screenshot:



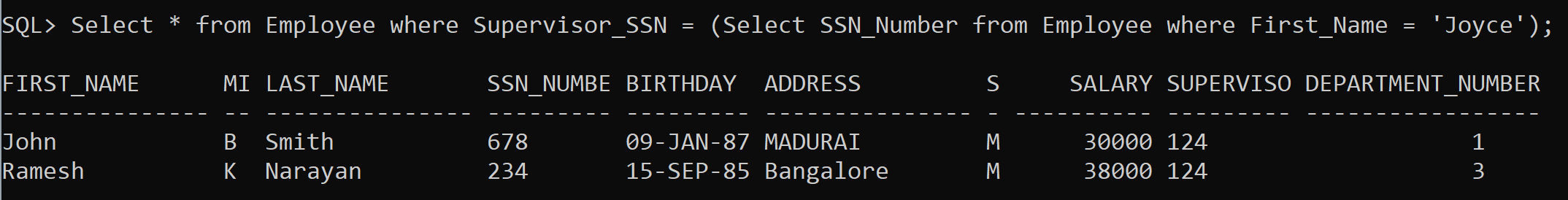
**Aim: To understand how to relate and access data from multiple tables.**

# Consider the schema given in exercise 2, and execute the following queries

1. **Find the names of all the employees who are directly supervised by ‘Joyce’. SQL Command:**

Select \* from Employee where Supervisor\_SSN = (Select SSN\_Number from Employee where First\_Name = 'Joyce');

# Screenshot:

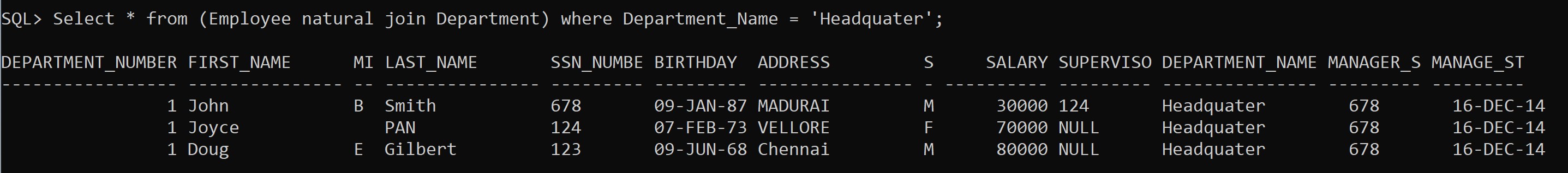


1. **Find the names of all the employees who are working in department ‘Headquarter’.**

# SQL Command:

Select \* from (Employee natural join Department) where Department\_Name = 'Headquater';

# Screenshot:



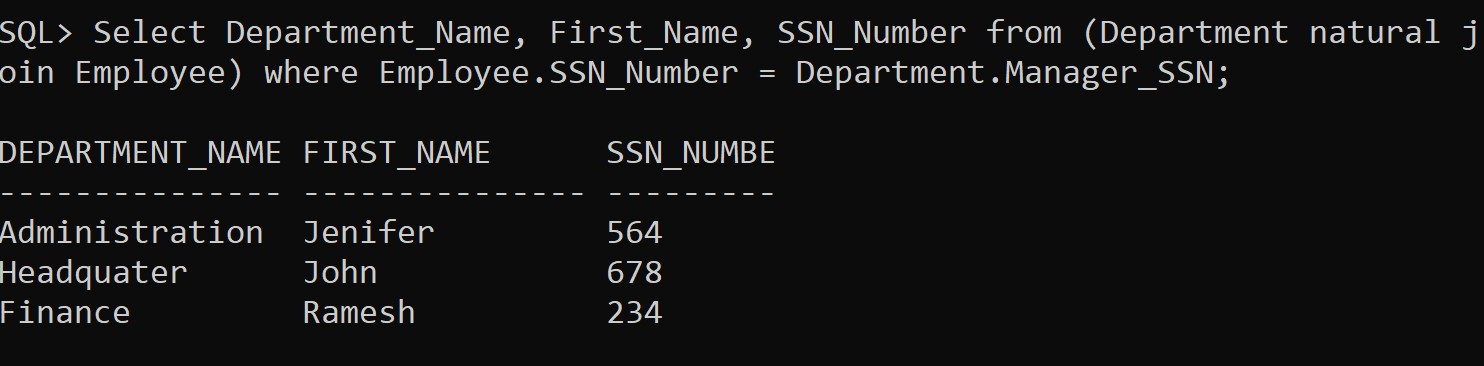
1. **List the department names and if has a manager then display the manager name too.**

# SQL Command:

Select Department\_Name, First\_Name, SSN\_Number from (Department natural join Employee) where Employee.SSN\_Number

= Department.Manager\_SSN;

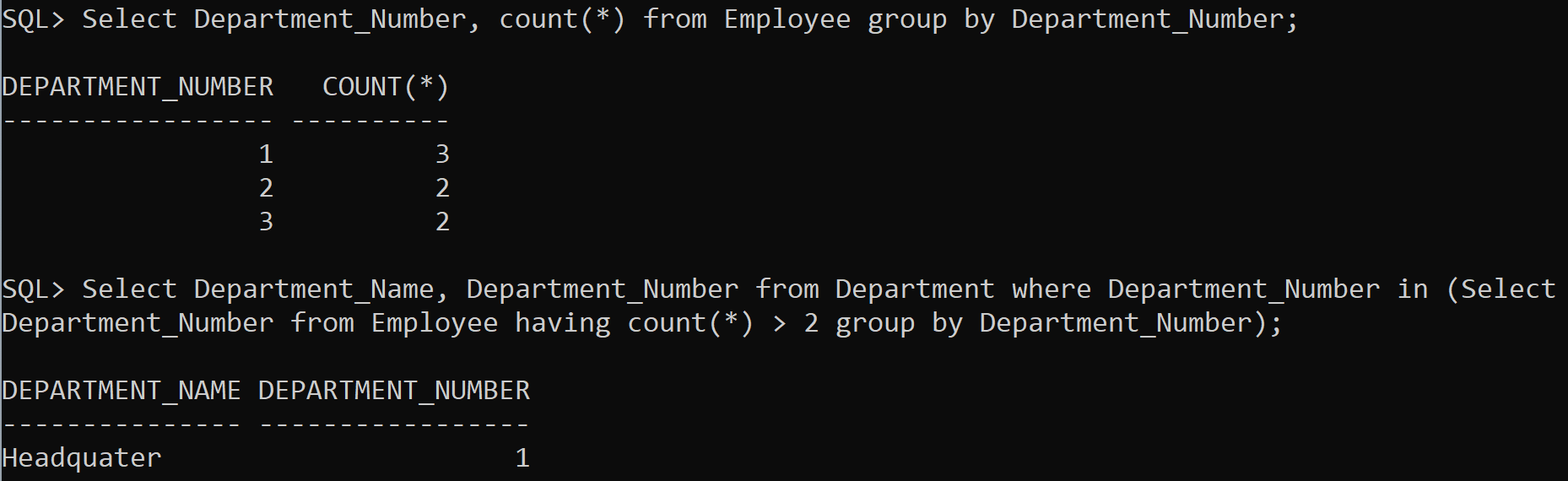
# Screenshot:



1. **Retrieve the names of the departments which have more than 2 employees. SQL Command:**

Select Department\_Name, Department\_Number from Department where Department\_Number in (Select Department\_Number from Employee having count(\*) > 2 group by Department\_Number);

# Screenshot:

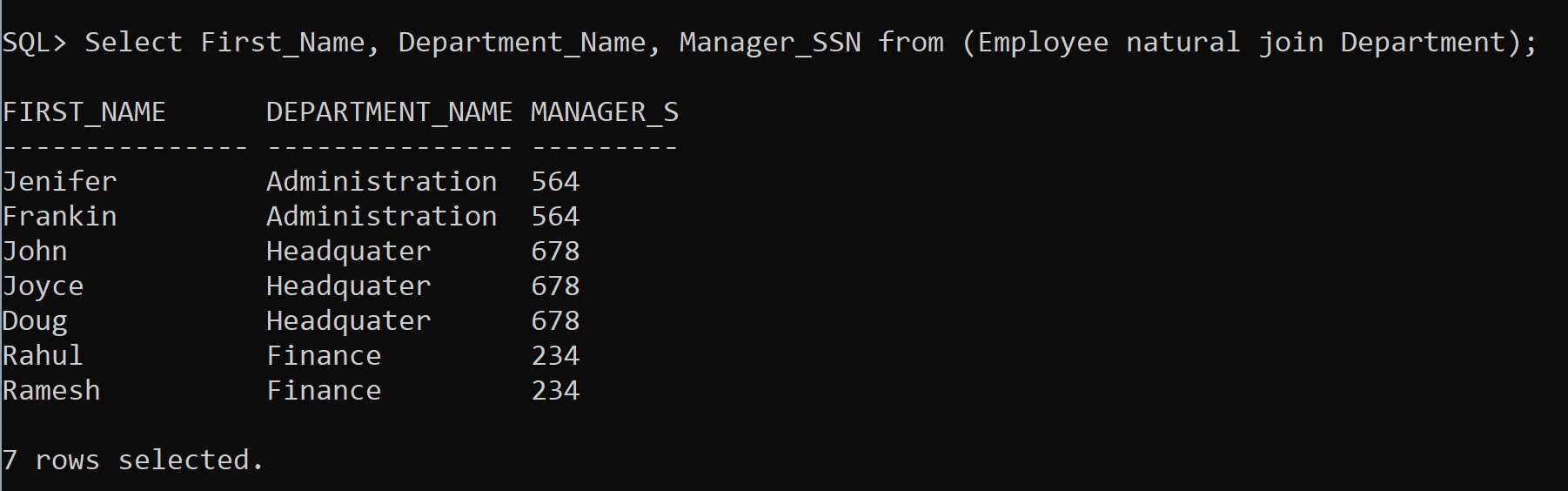


1. **Display the employee name along with the deptname and MgrSSN of that department.**

# SQL Command:

Select First\_Name, Department\_Name, Manager\_SSN from (Employee natural join Department);

# Screenshot:



1. **Find out the ename, deptname, deptno, eno of those who are working under some department number (1 or 2 or 3).**

# SQL Command:

Select First\_Name, SSN\_Number, Department\_Name, Department\_Number from (Employee natural join Department) where Department\_Number = 1;

Select First\_Name, SSN\_Number, Department\_Name, Department\_Number from (Employee natural join Department) where Department\_Number = 2;

Select First\_Name, SSN\_Number, Department\_Name, Department\_Number from (Employee natural join Department) where Department\_Number = 3;

# Screenshot:

