Ganpat University
Faculty of Engineering & Technology
Computer Science & Engineering

Practical_5

***Name:-* Dwij Vatsal Desai**
***Sem:-* 3**
***Sub: -* DBMS**
***Enrollment No.:-* 23162121027**

Practical: 5 Perform Queries using Group by and Having clause.

The SQL GROUP BY clause is used in collaboration with the SELECT statement to arrange identical data into groups. This GROUP BY clause follows the WHERE clause in a SELECT statement and precedes the ORDER BY clause.

Syntax

The basic syntax of a GROUP BY clause is shown in the following code block. The GROUP BY clause must follow the conditions in the WHERE clause and must precede the ORDER BY clause if one is used.

```
SELECT column1, column2
FROM table_name
WHERE [ conditions ]
GROUP BY column1, column2
ORDER BY column1, column2
SELECT column1, column2
FROM table_name
GROUP BY column1, column2
Having [CONDITION]
```

Example

Consider the CUSTOMERS table is having the following records −

```
+----+----------+-----+-----------+----------+
| ID | NAME | AGE | ADDRESS | SALARY |
+----+----------+-----+-----------+----------+
| 1 | Ramesh | 32 | Ahmedabad | 2000.00 |
| 2 | Khilan | 25 | Delhi | 1500.00 |
| 3 | kaushik | 23 | Kota | 2000.00 |
| 4 | Chaitali | 25 | Mumbai | 6500.00 |
| 5 | Hardik | 27 | Bhopal | 8500.00 |
| 6 | Komal | 22 | MP | 4500.00 |
| 7 | Muffy | 24 | Indore | 10000.00 |
+----+----------+-----+-----------+----------+
```

If you want to know the total amount of the salary on each customer, then the GROUP BY query would be as follows.

```
SQL> SELECT NAME, SUM(SALARY) FROM CUSTOMERS
GROUP BY NAME;
```

This would produce the following result −

```
+----------+-------------+
| NAME | SUM(SALARY) |
+----------+-------------+
| Chaitali | 6500.00 |
| Hardik | 8500.00 |
| kaushik | 2000.00 |
| Khilan | 1500.00 |
| Komal | 4500.00 |
| Muffy | 10000.00 |
| Ramesh | 2000.00 |
+----------+-------------+
```

Now, let us look at a table where the CUSTOMERS table has the following records with duplicate names −

```
+----+----------+-----+-----------+----------+
| ID | NAME | AGE | ADDRESS | SALARY |
+----+----------+-----+-----------+----------+
| 1 | Ramesh | 32 | Ahmedabad | 2000.00 |
| 2 | Ramesh | 25 | Delhi | 1500.00 |
| 3 | kaushik | 23 | Kota | 2000.00 |
| 4 | kaushik | 25 | Mumbai | 6500.00 |
| 5 | Hardik | 27 | Bhopal | 8500.00 |
| 6 | Komal | 22 | MP | 4500.00 |
| 7 | Muffy | 24 | Indore | 10000.00 |
+----+----------+-----+-----------+----------+
```

Now again, if you want to know the total amount of salary on each customer, then the GROUP BY query would be as follows −

```
SQL> SELECT NAME, SUM(SALARY) FROM CUSTOMERS
GROUP BY NAME;
```

This would produce the following result −

```
+---------+-------------+
| NAME | SUM(SALARY) |
+---------+-------------+
| Hardik | 8500.00 |
| kaushik | 8500.00 |
| Komal | 4500.00 |
| Muffy | 10000.00 |
| Ramesh | 3500.00 |
+---------+-------------+
```

The SQL HAVING Clause

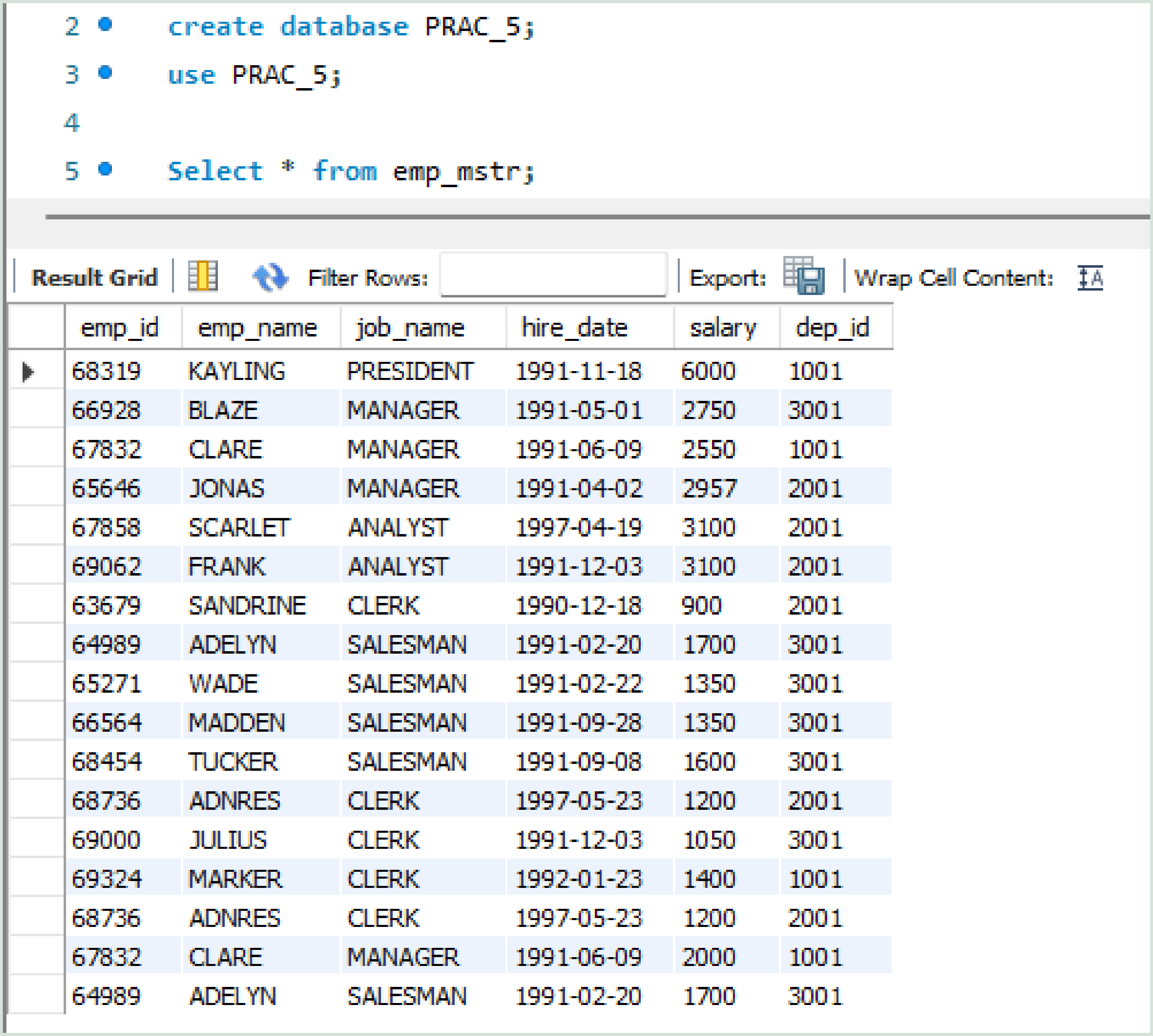
The HAVING clause was added to SQL because the WHERE keyword cannot be used with aggregate functions.

Solution Query:

Transfered table:

```
create database PRAC_5;
use PRAC_5;

Select * from emp_mstr;
```

Image:



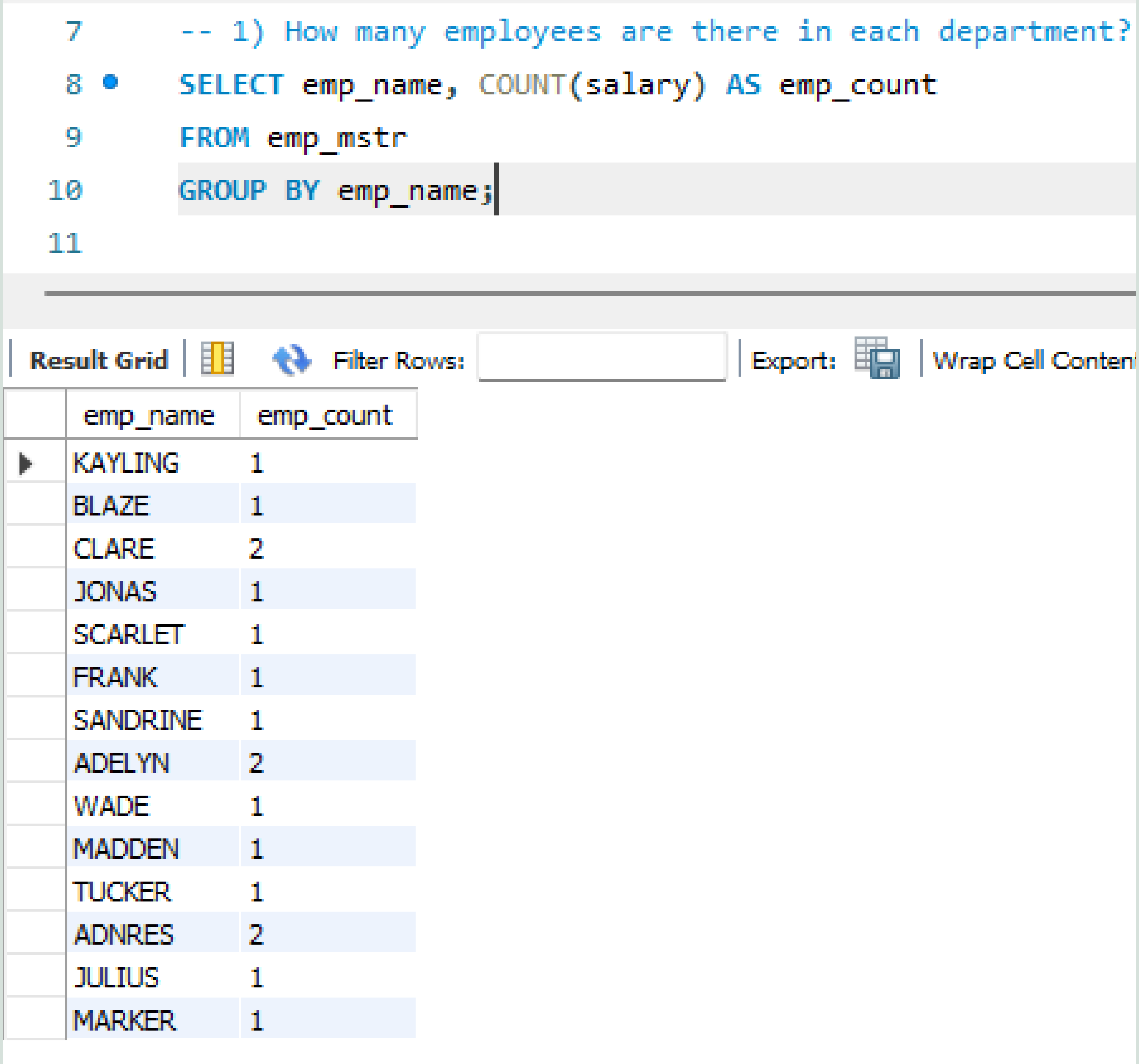
```
create database PRAC_5;
use PRAC_5;

Select * from emp_mstr;
```

emp_id	emp_name	job_name	hire_date	salary	dep_id
68319	KAYLING	PRESIDENT	1991-11-18	6000	1001
66928	BLAZE	MANAGER	1991-05-01	2750	3001
67832	CLARE	MANAGER	1991-06-09	2550	1001
65646	JONAS	MANAGER	1991-04-02	2957	2001
67858	SCARLET	ANALYST	1997-04-19	3100	2001
69062	FRANK	ANALYST	1991-12-03	3100	2001
63679	SANDRINE	CLERK	1990-12-18	900	2001
64989	ADELYN	SALESMAN	1991-02-20	1700	3001
65271	WADE	SALESMAN	1991-02-22	1350	3001
66564	MADDEN	SALESMAN	1991-09-28	1350	3001
68454	TUCKER	SALESMAN	1991-09-08	1600	3001
68736	ADNRES	CLERK	1997-05-23	1200	2001
69000	JULIUS	CLERK	1991-12-03	1050	3001
69324	MARKER	CLERK	1992-01-23	1400	1001
68736	ADNRES	CLERK	1997-05-23	1200	2001
67832	CLARE	MANAGER	1991-06-09	2000	1001
64989	ADELYN	SALESMAN	1991-02-20	1700	3001

-- 1) How many employees are there in each department?

```
SELECT emp_name, COUNT(salary) AS emp_count
FROM emp_mstr
GROUP BY emp_name;
```

Image:



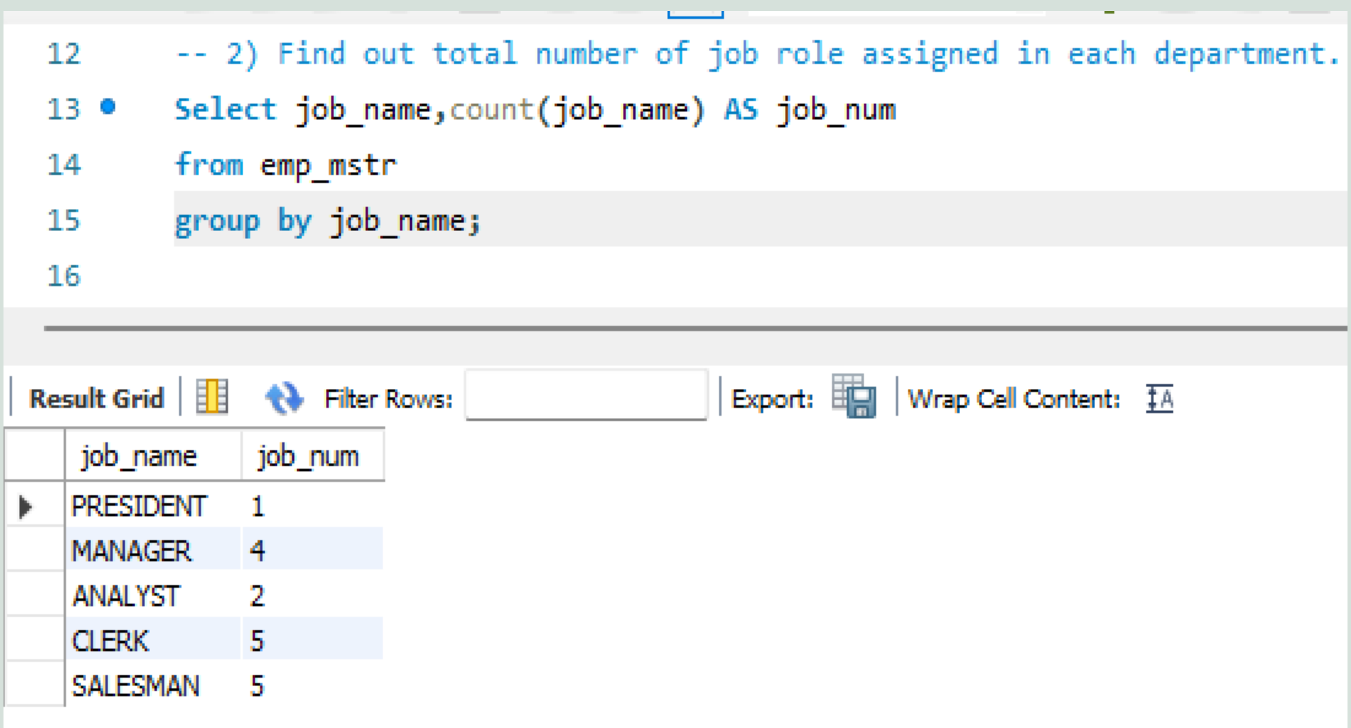
```
-- 1) How many employees are there in each department?
SELECT emp_name, COUNT(salary) AS emp_count
FROM emp_mstr
GROUP BY emp_name;
```

emp_name	emp_count
KAYLING	1
BLAZE	1
CLARE	2
JONAS	1
SCARLET	1
FRANK	1
SANDRINE	1
ADELYN	2
WADE	1
MADDEN	1
TUCKER	1
ADNRES	2
JULIUS	1
MARKER	1

-- 2) Find out total number of job role assigned in each department.

```
Select job_name,count(job_name) AS job_num
from emp_mstr
group by job_name;
```

Image:



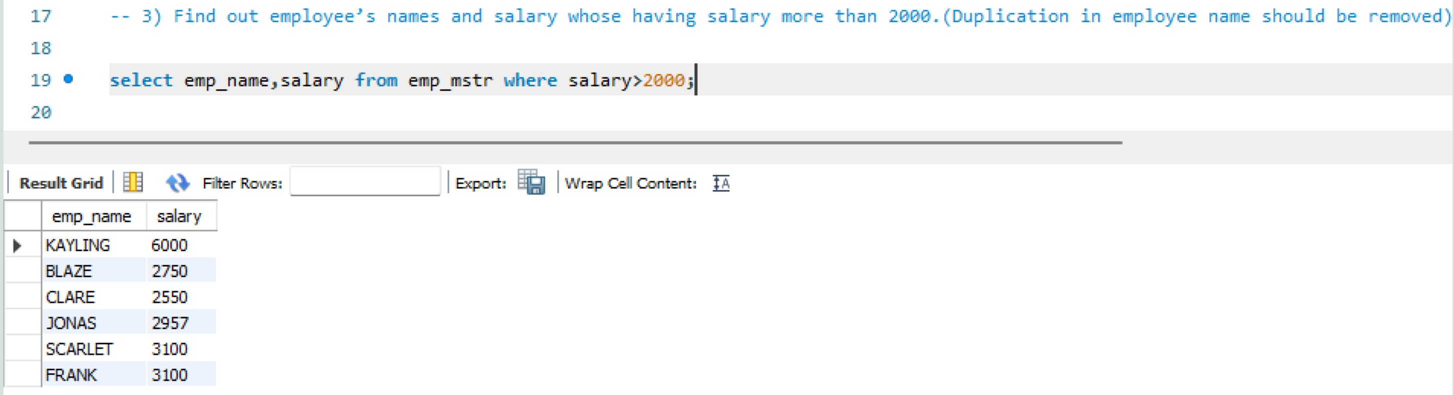
```
-- 2) Find out total number of job role assigned in each department.
Select job_name,count(job_name) AS job_num
from emp_mstr
group by job_name;
```

job_name	job_num
PRESIDENT	1
MANAGER	4
ANALYST	2
CLERK	5
SALESMAN	5

-- 3) Find out employee’s names and salary whose having salary more than 2000.(Duplication in employee name should be removed)

select emp_name,salary from emp_mstr where salary>2000;

Image:



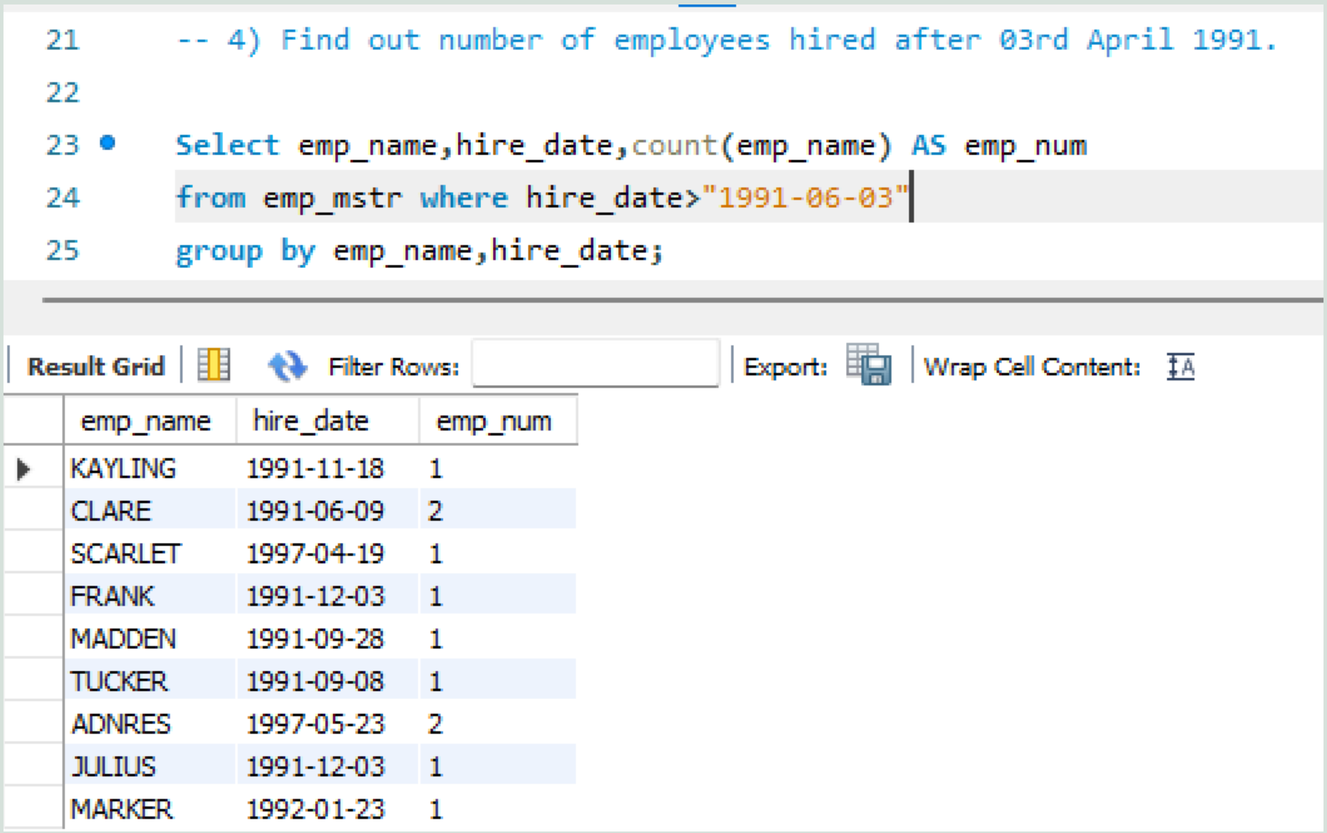
```
-- 3) Find out employee’s names and salary whose having salary more than 2000.(Duplication in employee name should be removed)

select emp_name,salary from emp_mstr where salary>2000;
```

emp_name	salary
KAYLING	6000
BLAZE	2750
CLARE	2550
JONAS	2957
SCARLET	3100
FRANK	3100

-- 4) Find out number of employees hired after 03rd April 1991.

Select emp_name,hire_date,count(emp_name) AS emp_num
from emp_mstr where hire_date>"1991-06-03"
group by emp_name,hire_date;

Image:



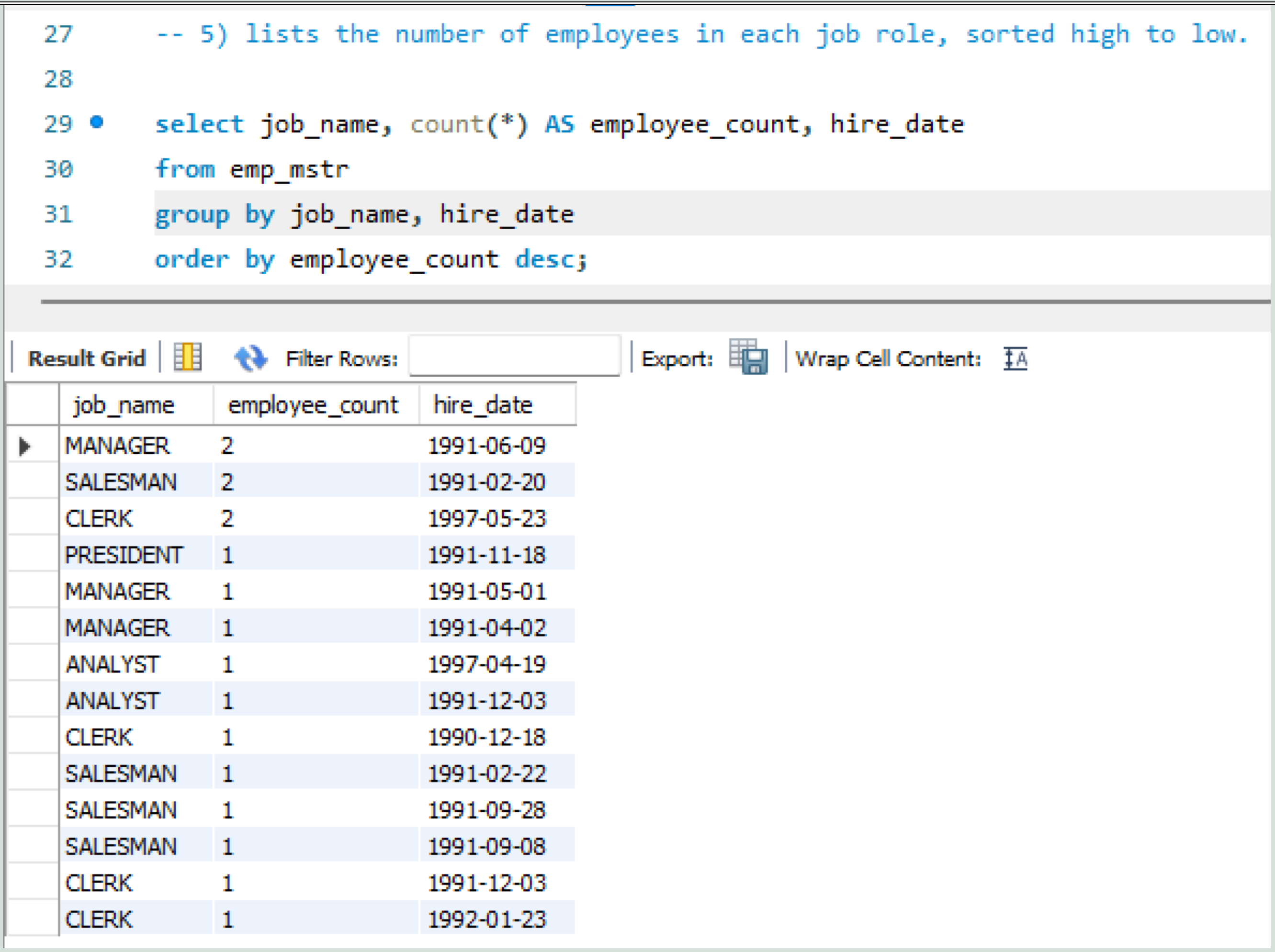
```
-- 4) Find out number of employees hired after 03rd April 1991.

Select emp_name,hire_date,count(emp_name) AS emp_num
from emp_mstr where hire_date>"1991-06-03"
group by emp_name,hire_date;
```

emp_name	hire_date	emp_num
KAYLING	1991-11-18	1
CLARE	1991-06-09	2
SCARLET	1997-04-19	1
FRANK	1991-12-03	1
MADDEN	1991-09-28	1
TUCKER	1991-09-08	1
ADNRES	1997-05-23	2
JULIUS	1991-12-03	1
MARKER	1992-01-23	1

-- 5) lists the number of employees in each job role, sorted high to low.

```
select job_name, count(*) AS employee_count, hire_date
from emp_mstr
group by job_name, hire_date
order by employee_count desc;
```

Image:



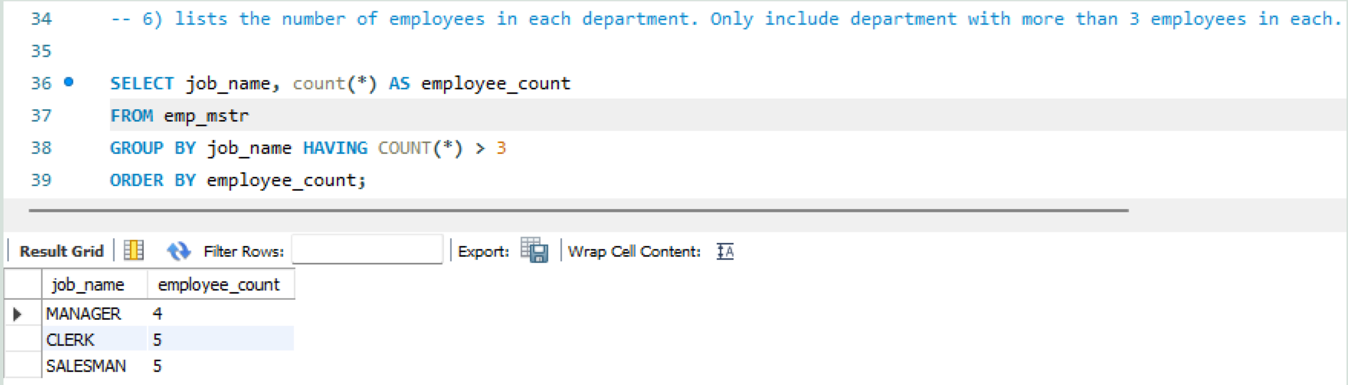
```
-- 5) lists the number of employees in each job role, sorted high to low.

select job_name, count(*) AS employee_count, hire_date
from emp_mstr
group by job_name, hire_date
order by employee_count desc;
```

job_name	employee_count	hire_date
MANAGER	2	1991-06-09
SALESMAN	2	1991-02-20
CLERK	2	1997-05-23
PRESIDENT	1	1991-11-18
MANAGER	1	1991-05-01
MANAGER	1	1991-04-02
ANALYST	1	1997-04-19
ANALYST	1	1991-12-03
CLERK	1	1990-12-18
SALESMAN	1	1991-02-22
SALESMAN	1	1991-09-28
SALESMAN	1	1991-09-08
CLERK	1	1991-12-03
CLERK	1	1992-01-23

-- 6) lists the number of employees in each department. Only include department with more than 3 employees in each.

```
SELECT job_name, count(*) AS employee_count
FROM emp_mstr
GROUP BY job_name HAVING COUNT(*) > 3
ORDER BY employee_count;
```

Image:



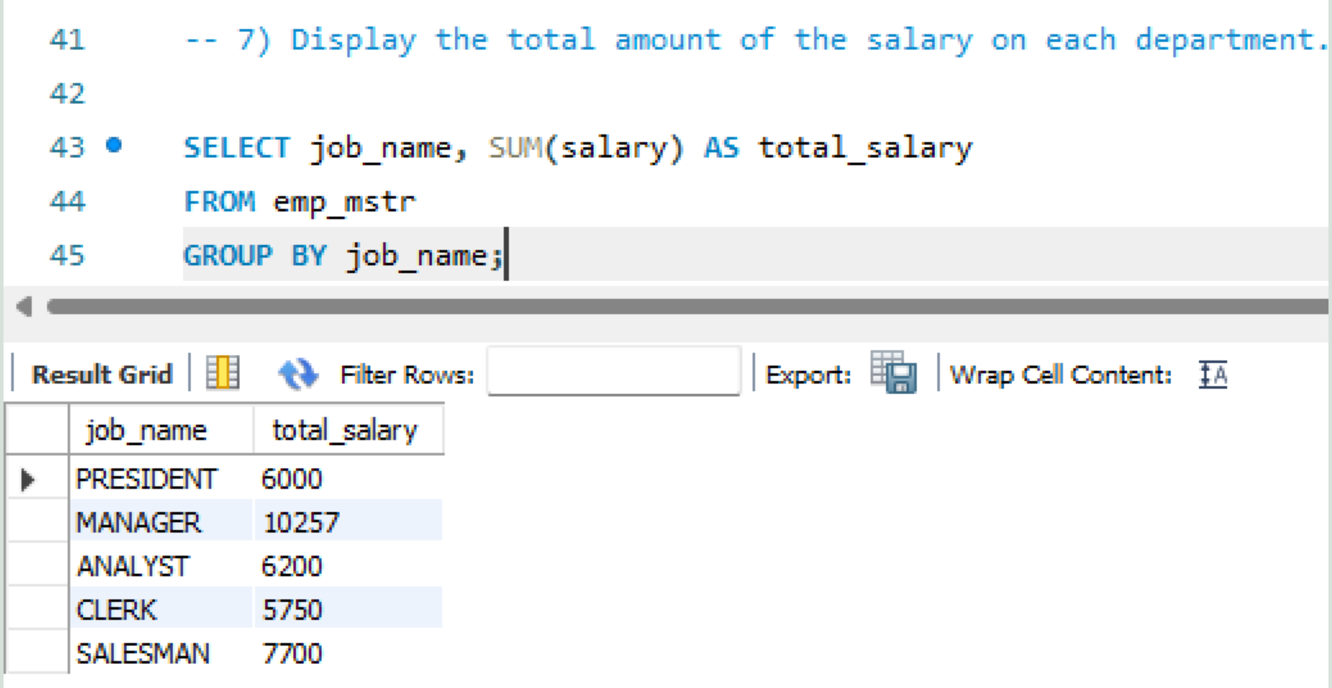
```
-- 6) lists the number of employees in each department. Only include department with more than 3 employees in each.

SELECT job_name, count(*) AS employee_count
FROM emp_mstr
GROUP BY job_name HAVING COUNT(*) > 3
ORDER BY employee_count;
```

job_name	employee_count
MANAGER	4
CLERK	5
SALESMAN	5

-- 7) Display the total amount of the salary on each department.

```
SELECT job_name, SUM(salary) AS total_salary
FROM emp_mstr
GROUP BY job_name;
```

<u>Image</u>:



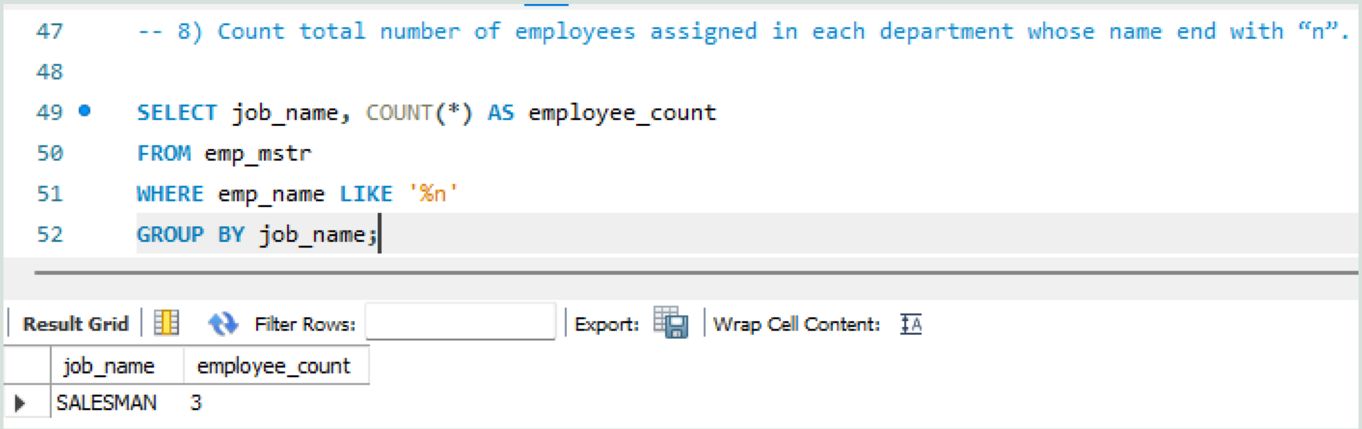
```
-- 7) Display the total amount of the salary on each department.

SELECT job_name, SUM(salary) AS total_salary
FROM emp_mstr
GROUP BY job_name;
```

job_name	total_salary
PRESIDENT	6000
MANAGER	10257
ANALYST	6200
CLERK	5750
SALESMAN	7700

-- 8) Count total number of employees assigned in each department whose name end with “n”.

```
SELECT job_name, COUNT(*) AS employee_count
FROM emp_mstr
WHERE emp_name LIKE '%n'
GROUP BY job_name;
```

Image:



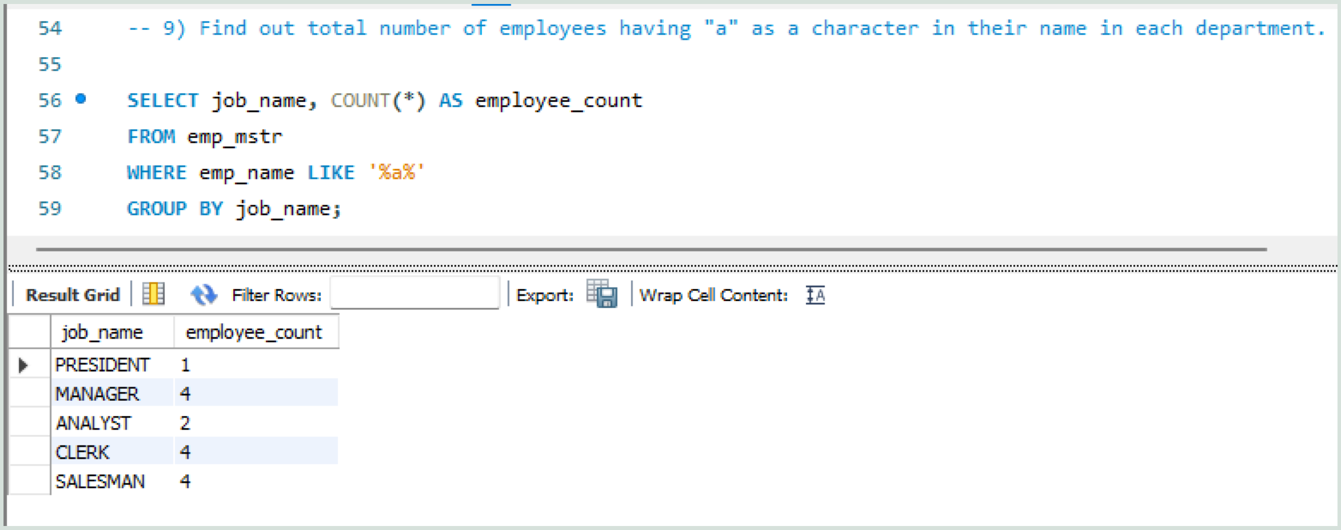
```sql
-- 8) Count total number of employees assigned in each department whose name end with “n”.

SELECT job_name, COUNT(*) AS employee_count
FROM emp_mstr
WHERE emp_name LIKE '%n'
GROUP BY job_name;
```

job_name	employee_count
SALESMAN	3

-- 9) Find out total number of employees having "a" as a character in their name in each department.

SELECT job_name, COUNT(*) AS employee_count
FROM emp_mstr
WHERE emp_name LIKE '%a%'
GROUP BY job_name;

Image:



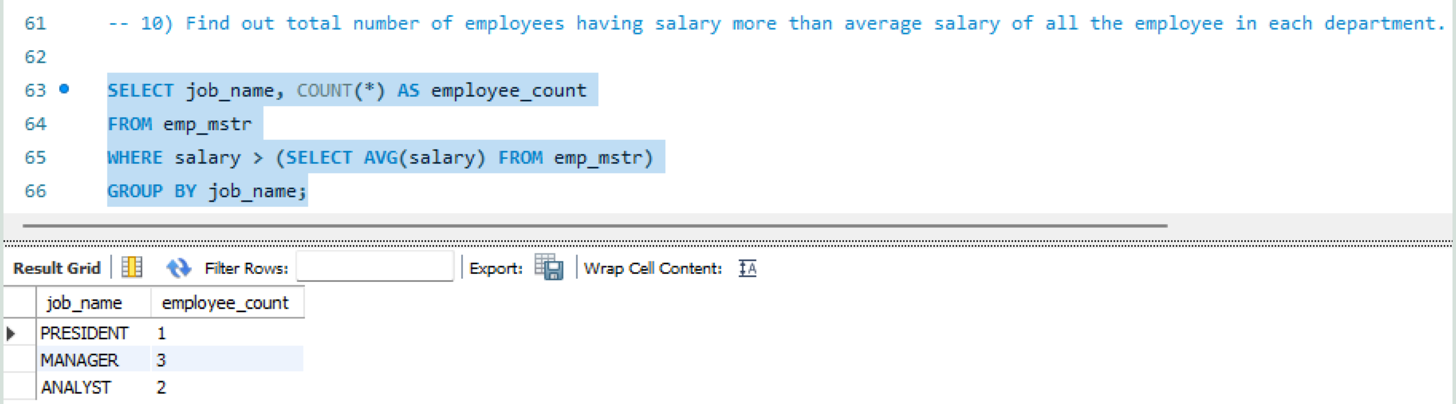
```sql
-- 9) Find out total number of employees having "a" as a character in their name in each department.

SELECT job_name, COUNT(*) AS employee_count
FROM emp_mstr
WHERE emp_name LIKE '%a%'
GROUP BY job_name;
```

job_name	employee_count
PRESIDENT	1
MANAGER	4
ANALYST	2
CLERK	4
SALESMAN	4

-- 10) Find out total number of employees having salary more than average salary of all the employee in each department.

SELECT job_name, COUNT(*) AS employee_count
FROM emp_mstr
WHERE salary > (SELECT AVG(salary) FROM emp_mstr)
GROUP BY job_name;

Image:



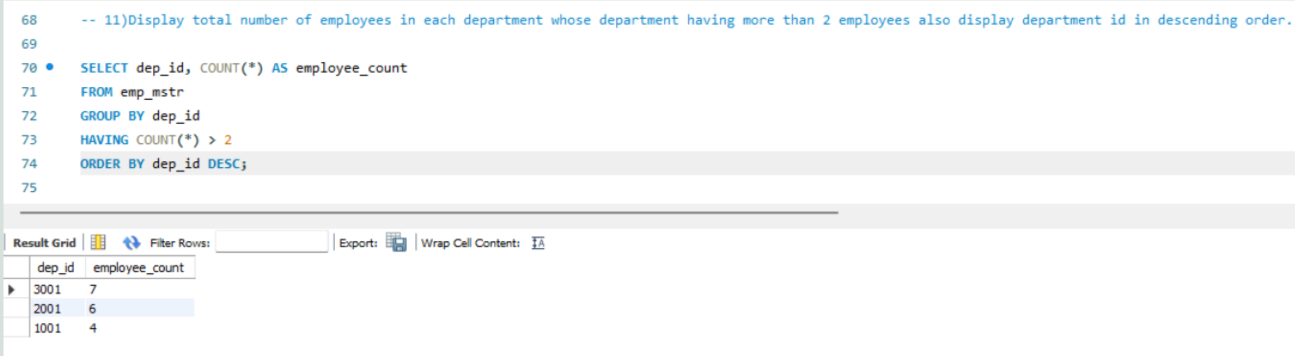
```
-- 10) Find out total number of employees having salary more than average salary of all the employee in each department.

SELECT job_name, COUNT(*) AS employee_count
FROM emp_mstr
WHERE salary > (SELECT AVG(salary) FROM emp_mstr)
GROUP BY job_name;
```

job_name	employee_count
PRESIDENT	1
MANAGER	3
ANALYST	2

-- 11)Display total number of employees in each department whose department having more than 2 employees also display department id in descending order.

SELECT dep_id, COUNT(*) AS employee_count
FROM emp_mstr
GROUP BY dep_id
HAVING COUNT(*) > 2
ORDER BY dep_id DESC;

Image:



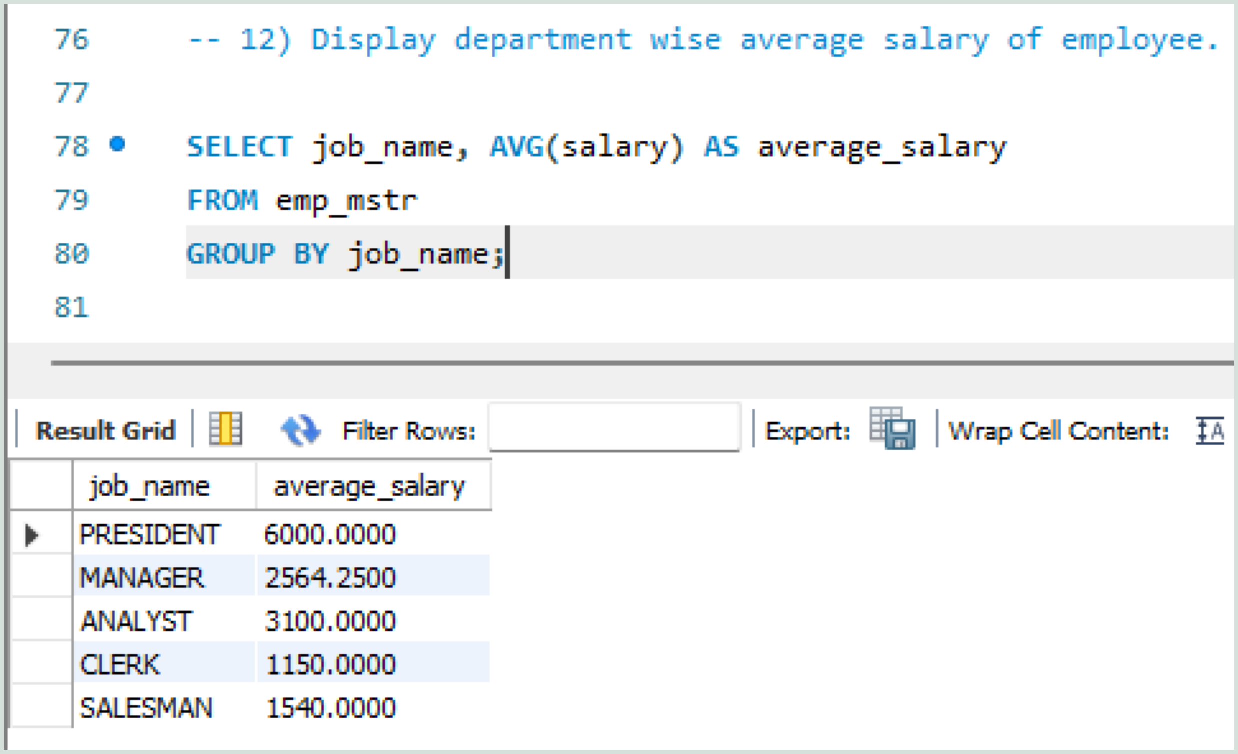
```
-- 11)Display total number of employees in each department whose department having more than 2 employees also display department id in descending order.

SELECT dep_id, COUNT(*) AS employee_count
FROM emp_mstr
GROUP BY dep_id
HAVING COUNT(*) > 2
ORDER BY dep_id DESC;
```

dep_id	employee_count
3001	7
2001	6
1001	4

-- 12) Display department wise average salary of employee.

```
SELECT job_name, AVG(salary) AS average_salary
FROM emp_mstr
GROUP BY job_name;
```

Image:



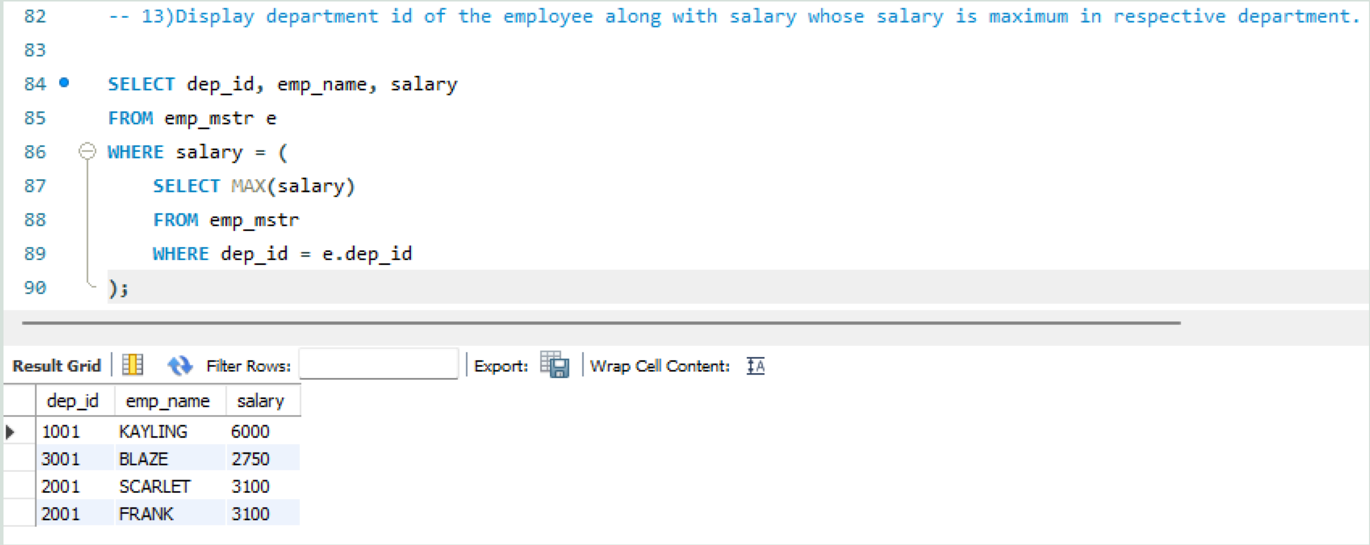
```
-- 12) Display department wise average salary of employee.

SELECT job_name, AVG(salary) AS average_salary
FROM emp_mstr
GROUP BY job_name;
```

job_name	average_salary
PRESIDENT	6000.0000
MANAGER	2564.2500
ANALYST	3100.0000
CLERK	1150.0000
SALESMAN	1540.0000

-- 13)Display department id of the employee along with salary whose salary is maximum in respective department.

```
SELECT dep_id, emp_name, salary
FROM emp_mstr e
WHERE salary = (
   SELECT MAX(salary)
   FROM emp_mstr
   WHERE dep_id = e.dep_id
);
```

Image:



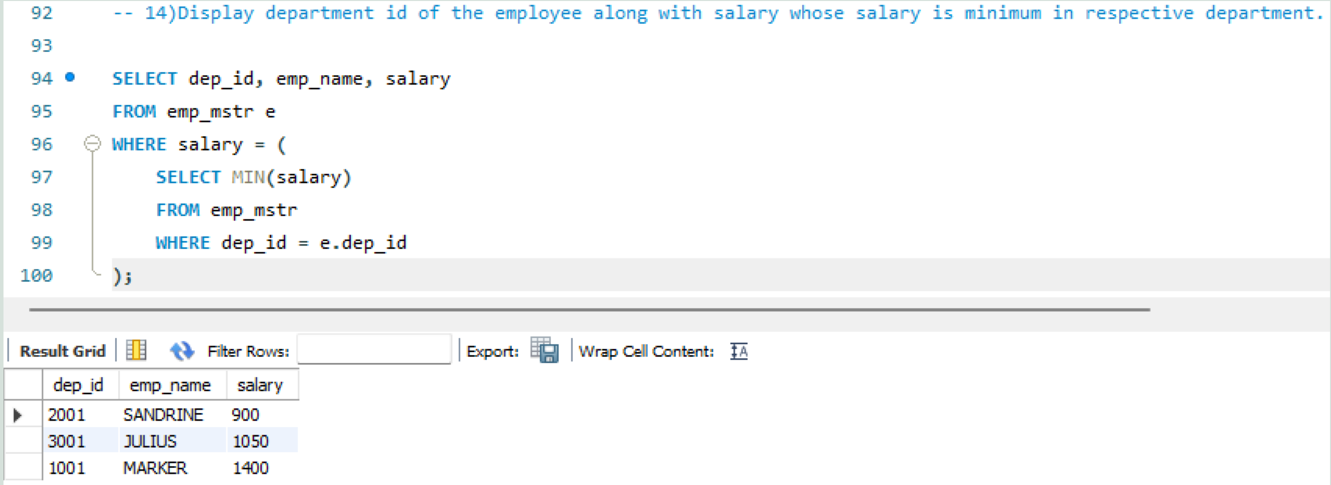
```
-- 13)Display department id of the employee along with salary whose salary is maximum in respective department.

SELECT dep_id, emp_name, salary
FROM emp_mstr e
WHERE salary = (
    SELECT MAX(salary)
    FROM emp_mstr
    WHERE dep_id = e.dep_id
);
```

dep_id	emp_name	salary
1001	KAYLING	6000
3001	BLAZE	2750
2001	SCARLET	3100
2001	FRANK	3100

-- 14)Display department id of the employee along with salary whose salary is minimum in respective department.

```
SELECT dep_id, emp_name, salary
FROM emp_mstr e
WHERE salary = (
    SELECT MAX(salary)
    FROM emp_mstr
    WHERE dep_id = e.dep_id
);
```

Image:



```
-- 14)Display department id of the employee along with salary whose salary is minimum in respective department.

SELECT dep_id, emp_name, salary
FROM emp_mstr e
WHERE salary = (
    SELECT MIN(salary)
    FROM emp_mstr
    WHERE dep_id = e.dep_id
);
```

dep_id	emp_name	salary
2001	SANDRINE	900
3001	JULIUS	1050
1001	MARKER	1400