

Day 6

PRN : 200243020003

Group Functions

1. First Rewrite /practice all demo queries taught in session.

```
SELECT COUNT(*) FROM EMPLOYEES;
SELECT COUNT(commission_pct) FROM EMPLOYEES;
SELECT COUNT(nvl(commission_pct,0)) FROM EMPLOYEES;
SELECT COUNT(DISTINCT department_id) FROM EMPLOYEES;
SELECT DEPARTMENT_ID, MAX(SALARY) FROM EMPLOYEES GROUP BY DEPARTMENT_ID
ORDER BY DEPARTMENT_ID;
SELECT DEPARTMENT_ID, MAX(HIRE_DATE) FROM EMPLOYEES GROUP BY
DEPARTMENT_ID;
SELECT DEPARTMENT_ID, round(AVG(SALARY)) FROM EMPLOYEES GROUP BY
DEPARTMENT_ID;
SELECT JOB_ID, MAX(SALARY) FROM EMPLOYEES GROUP BY JOB_ID ORDER BY JOB_ID;
SELECT job_id, DEPARTMENT_ID FROM EMPLOYEES ORDER BY DEPARTMENT_ID;
SELECT DEPARTMENT_ID, COUNT(employee_id) FROM EMPLOYEES GROUP BY
DEPARTMENT_ID, JOB_ID ORDER BY DEPARTMENT_ID;
SELECT DEPARTMENT_ID, JOB_ID, COUNT(EMPLOYEE_ID) FROM EMPLOYEES GROUP BY
DEPARTMENT_ID, JOB_ID ORDER BY DEPARTMENT_ID;
SELECT DEPARTMENT_ID, JOB_ID, SUM(SALARY) FROM EMPLOYEES WHERE
DEPARTMENT_ID >40 GROUP BY DEPARTMENT_ID, JOB_ID ORDER BY DEPARTMENT_ID;
SELECT DEPARTMENT_ID, AVG(SALARY) FROM EMPLOYEES WHERE AVG(SALARY) > 8000
GROUP BY DEPARTMENT_ID;
SELECT DEPARTMENT_ID, MAX(SALARY) FROM EMPLOYEES GROUP BY DEPARTMENT_ID;
SELECT DEPARTMENT_ID, MAX(SALARY) sal FROM EMPLOYEES GROUP BY
DEPARTMENT_ID HAVING MAX(SALARY)>10000;
SELECT JOB_ID, SUM(SALARY) FROM EMPLOYEES GROUP BY JOB_ID HAVING
sum(SALARY)>13000 AND JOB_ID NOT LIKE '%REP%';
SELECT EMPLOYEE_ID, LAST_NAME, DEPARTMENT_ID FROM EMPLOYEES JOIN
DEPARTMENTS
```

2. Display maximum salary from department 20

```
SELECT
    DEPARTMENT_ID,
    MAX(SALARY)
FROM
    EMPLOYEES
GROUP BY
    DEPARTMENT_ID
```

```
HAVING  
    DEPARTMENT_ID = 20;
```

3. Display the oldest hire_date from employees

```
SELECT  
    MIN(HIRE_DATE) "Oldest date"  
FROM  
    EMPLOYEES;
```

4. Find out the correct average salary

```
SELECT  
    EMPLOYEE_ID,  
    round(AVG(SALARY)) "AVg Salary"  
FROM  
    EMPLOYEES  
GROUP BY  
    EMPLOYEE_ID;
```

5. Display count of employees earning commission

```
SELECT  
    DEPARTMENT_ID,  
    COUNT(COMMISSION_PCT) "count"  
FROM  
    EMPLOYEES  
GROUP BY  
    DEPARTMENT_ID;
```

6. Display department_id,maximum salary department-wise

```
SELECT  
    DEPARTMENT_ID,  
    MAX(SALARY) "Max salary"  
FROM  
    EMPLOYEES  
GROUP BY  
    DEPARTMENT_ID  
ORDER BY  
    DEPARTMENT_ID;
```

7. Display department_id, minimum salary for those departments whose min sal is less than 7000

```
SELECT
    DEPARTMENT_ID,
    MIN(SALARY) "min salary"
FROM
    EMPLOYEES
GROUP BY
    DEPARTMENT_ID
HAVING
    MIN(SALARY) < 7000;
```

8. Write a query to display the number of people with the same job. (count jobwise)

```
SELECT
    DEPARTMENT_ID,
    COUNT(JOB_ID)
FROM
    EMPLOYEES
GROUP BY
    DEPARTMENT_ID
ORDER BY
    DEPARTMENT_ID;
```

9. Re-write the above query where user is prompted for job_id. (eg. : "REP" if entered they display all emp with job_id having REP)

```
SELECT
    FIRST_NAME
FROM
    EMPLOYEES
WHERE
    JOB_ID = & id;
```

10. Find the difference between the highest and lowest salaries. Label the column DIFFERENCE

```
SELECT
    DEPARTMENT_ID,
    MAX(SALARY) - MIN(SALARY) "difference"
FROM
    EMPLOYEES
GROUP BY
    DEPARTMENT_ID;
```

11. Create a report to display the manager number and the salary of the lowest-paid employee for that manager. Exclude anyone whose manager is not known. Exclude any groups where the minimum salary is \$6,000 or less. Sort the output in descending order of salary.

```
SELECT
    MANAGER_ID,
    MIN(SALARY)
FROM
    EMPLOYEES
WHERE
    MANAGER_ID IS NOT NULL
GROUP BY
    MANAGER_ID
HAVING
    MIN(SALARY) > 6000
ORDER BY
    MIN(SALARY)
    DESC;
```

12. Create a query to display the total number of employees hired in 1995, 1996, 1997, and 1998. Create appropriate column headings.

```
SELECT
    DEPARTMENT_ID,
    COUNT(to_char(HIRE_DATE, 'yyyy'))
FROM
    EMPLOYEES
GROUP BY
    DEPARTMENT_ID
HAVING
    COUNT(to_char(HIRE_DATE, 'yyyy'))
    in(1995, 1996, 1997, 1998);
```