

Group functions can be nested to a depth of two. The slide example displays the maximum average salary.

```
SELECT MAX(AVG(salary)) FROM employees GROUP BY department_id;
```

Summary

In this exercise, students should have learned how to:

- Use the group functions COUNT, MAX, MIN, and AVG
- Write queries that use the GROUP BY clause
- Write queries that use the HAVING clause

```
SELECT column, group_function  
FROM table  
[WHERE condition]  
[GROUP BY group_by_expression]  
[HAVING group_condition]  
[ORDER BY column];
```

Find the Solution for the following:

Determine the validity of the following three statements. Circle either True or False.

1. Group functions work across many rows to produce one result per group.

True/False **TRUE**

2. Group functions include nulls in calculations.

True/False **FALSE**

3. The WHERE clause restricts rows prior to inclusion in a group calculation.

True/False **FALSE**

The HR department needs the following reports:

4. Find the highest, lowest, sum, and average salary of all employees. Label the columns Maximum, Minimum, Sum, and Average, respectively. Round your results to the nearest whole number

```
select Round (MAX (salary)) AS Maximum, Round (MIN (salary))  
AS Minimum, Round (SUM (salary)) AS SUM (AVG (salary))  
AS Average from employees
```

5. Modify the above query to display the minimum, maximum, sum, and average salary for each job type.

```
select Round (MAX (salary)) AS MAXIMUM, Round  
(MIN (salary)) AS MINIMUM, Round (SUM (salary)) AS  
SUM, Round (AVG (salary)) AS AVERAGE FROM EMPLOYEES  
join department on department.dept-id = employees.department-id  
group by dept-name;
```

6. Write a query to display the number of people with the same job. Generalize the query so that the user in the HR department is prompted for a job title.

```
select d.dept_name, COUNT(*) AS NUMBER OF Employees
from Employee e
join department d on e.department_id = d.dept_id
group by d.dept_name;
```

7. Determine the number of managers without listing them. Label the column Number of Managers. Hint: Use the MANAGER_ID column to determine the number of managers.

```
select COUNT(DISTINCT MANAGER_ID) AS "Number of Managers"
from Employees
WHERE MANAGER_ID IS NOT NULL;
```

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

```
select max(salary) - min(salary) AS "DIFFERENCE"
from employees;
```

9. 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) AS "Lowest Salary"
FROM Employees
WHERE MANAGER_ID IS NOT NULL
Group By MANAGER_ID
HAVING MIN(salary) > 6000
Order By "Lowest Salary" DESC;
```

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

```
select extract(year from hire_date) AS "yearly-wise employment",
count(*) from employees
Group by extract(year from hire_date)
Having extract(year from hire_date) IN (1995, 1996, 1997, 1998);
```


11. Create a matrix query to display the job, the salary for that job based on department number, and the total salary for that job, for departments 20, 50, 80, and 90, giving each column an appropriate heading.

```
select d.dept-name, sum(e.salary)
from employees e
join department d on e.department-id = d.dept-id,
where department-id in (20, 50, 80, 90)
group by d.dept-name;
```

12. Write a query to display each department's name, location, number of employees, and the average salary for all the employees in that department. Label the column name-Location, Number of people, and salary respectively. Round the average salary to two decimal places.

```
select d.dept-name AS "Name", d.location-id AS "LOCATION",
COUNT(e.department-id) AS "Number of people",
ROUND(AVG(e.salary), 2) AS "Salary"
from department d
join employees e on d.dept-id = e.department-id
GROUP BY d.dept-name, d.location-id;
```

Evaluation Procedure	Marks awarded
Query(5)	
Execution (5)	
Viva(5)	
Total (15)	
Faculty Signature	