Ex.No.: 10		
Date:	12/09/2024	AGGREGATING DATA USING GROUP FUNCTIONS

Find the Solution for the following:

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

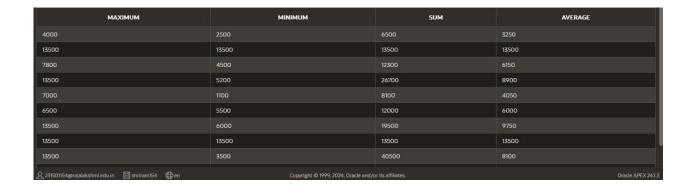
- 1.Group functions work across many rows to produce one result per group. True/False TRUE
- 2. Group functions include nulls in calculations. True/False FALSE
- 3.The WHERE clause restricts rows prior to inclusion in a group calculation. True/False FALSE
- 4) Find the highest, lowest, sum, and average 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, ROUND(AVG(salary)) AS Average FROM employees;



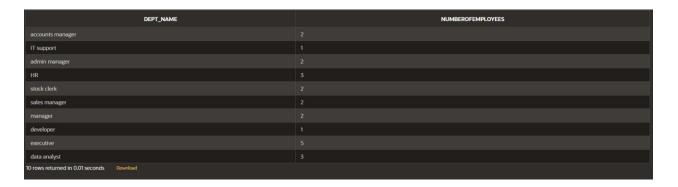
5) Modify the above query to display the minimum, maximum, sum, and average salary for each job type.

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_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 NumberOfEmployees FROM Employees 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

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;

Name	Location	Number of People	Salary			
sales manager			6000			
data analyst	1700		9753.33			
stock clerk			4050			
HR			8900			
admin manager						
manager	10		9750			
accounts manager						
executive			6333.33			
developer			13500			
executive			10750			
More than 10 rows available. Increase rows selector to view more rows.						
10 rows returned in 0.03 seconds Developed						