

AGGREGATING DATA USING GROUP FUNCTIONS

EXP : 10

DATE :

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.

Ans : False

2. Group functions include nulls in calculations.

Ans : False

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

Ans : True

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 .

Ans : SELECT

ROUND(MAX(salary)) AS "MAXIMUM",

ROUND(MIN(salary)) AS "MINIMUM",

ROUND(SUM(salary)) AS "SUM",

ROUND(AVG(salary)) AS "AVERAGE"

FROM MY_EMPLOYEE;

| MAXIMUM | MINIMUM | SUM | AVERAGE |
|---------|---------|------|---------|
| 1550 | 1000 | 3650 | 1217 |

1 rows returned in 0.03 seconds [Download](#) [Activate](#) [Go to Settings](#)

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

Ans : select round(max(e.salary)) as maximum,round(min(e.salary)) as minimum,round(sum(e.salary)) as sum,round(avg(e.salary)) as average
from EMPLOYEETABLE e join DEPARTMENTTABLE d on d.dept_id = e.DEPARTMENT_ID group by e.DEPARTMENT_ID;

| MAXIMUM | MINIMUM | SUM | AVERAGE |
|---------|---------|-------|---------|
| 80000 | 80000 | 80000 | 80000 |
| 90000 | 90000 | 90000 | 90000 |
| 60000 | 60000 | 60000 | 60000 |
| 90000 | 90000 | 90000 | 90000 |

4 rows returned in 0.04 seconds [Download](#)

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.

Ans : select d.JOB_CUR,count(*) as NUMBEROFEMPLOYEES from employeetable e
join department d on e.department_id=d.dept_id group by d.job_cur

no data found

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.

Ans : select count(distinct manager_id) as "Number of Managers" from employeetable where manager_id is NOT NULL

| Number of Managers |
|--|
| 5 |
| 1 rows returned in 0.01 seconds Download |

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

Ans : select max(salary)-min(salary) as difference from employeetable

| DIFFERENCE |
|--|
| 30000 |
| 1 rows returned in 0.01 seconds Download |

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.

Ans : select manager_id,min(salary) as "Lowest Salary" from employeeetable

where manager_id is not null group by manager_id having min(salary)>6000 order by "Lowest Salary" desc

| MANAGER_ID | Lowest Salary |
|------------|---------------|
| 883 | 90000 |
| 881 | 90000 |
| 882 | 80000 |
| 884 | 70000 |
| 885 | 60000 |

5 rows returned in 0.01 seconds [Download](#)

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.

Ans : select extract(year from HIRE_DATE) as "yearly wise employment",count(*) from employeeetable group by extract(year from HIRE_DATE) having extract(year from HIRE_DATE) in (1995,1996,1997,1998)

no data found

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.

Ans : `SELECT d.job_cur, SUM(e.salary) FROM employeetable e JOIN DEPARTMENT d ON e.DEPARTMENT_ID = d.dept_id WHERE e.DEPARTMENT_ID IN (20, 50, 80, 90) GROUP BY d.job_cur;`

no data found

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.

Ans : `select d.job_cur as "Name", d.country as "Location", count(e.department_id) as "Number Of People", round(avg(e.salary), 2) "Salary" from department d join employeetable e on e.department_id = d.dept_id group by job_cur, d.country;`

| Name | Location | Number of People | Salary |
|-----------|-----------|------------------|--------|
| FULSTACK | HELL | 1 | 5500 |
| BACKEND | ATLANTICA | 2 | 80000 |
| executive | PAKISTAN | 1 | 4000 |

3 rows returned in 0.03 seconds [Download](#)

