

## EXERCISE-8

### Aggregating Data Using Group Functions

#### Objectives

After the completion of this exercise, the students be will be able to do the following:

- Identify the available group functions
- Describe the use of group functions
- Group data by using the GROUP BY clause
- Include or exclude grouped rows by using the HAVING clause

#### What Are Group Functions?

Group functions operate on sets of rows to give one result per group

#### Types of Group Functions

- AVG
- COUNT
- MAX
- MIN
- STDDEV
- SUM
- VARIANCE

Each of the functions accepts an argument. The following table identifies the options that you can use in the syntax:

Function	Description
AVG ( [DISTINCT ALL] n )	Average value of <i>n</i> , ignoring null values
COUNT ( ( *   [DISTINCT ALL] <i>expr</i> ) )	Number of rows, where <i>expr</i> evaluates to something other than null (count all selected rows using *, including duplicates and rows with nulls)
MAX ( [DISTINCT ALL] <i>expr</i> )	Maximum value of <i>expr</i> , ignoring null values
MIN ( [DISTINCT ALL] <i>expr</i> )	Minimum value of <i>expr</i> , ignoring null values
STDDEV ( [DISTINCT ALL] <i>x</i> )	Standard deviation of <i>n</i> , ignoring null values
SUM ( [DISTINCT ALL] <i>n</i> )	Sum values of <i>n</i> , ignoring null values
VARIANCE ( [DISTINCT ALL] <i>x</i> )	Variance of <i>n</i> , ignoring null values

#### Group Functions: Syntax

```
SELECT [column,] group_function(column), ...  
FROM table  
[WHERE condition]  
[GROUP BY column]  
[ORDER BY column];
```

#### Guidelines for Using Group Functions

- DISTINCT makes the function consider only nonduplicate values; ALL makes it consider every value, including duplicates. The default is ALL and therefore does not need to be specified.



- The data types for the functions with an *expr* argument may be CHAR, VARCHAR2, NUMBER, or DATE.

- All group functions ignore null values.

### Using the AVG and SUM Functions

You can use AVG and SUM for numeric data.

```
SELECT AVG(salary), MAX(salary),  
MIN(salary), SUM(salary)  
FROM employees  
WHERE job_id LIKE '%REP%';
```

### Using the MIN and MAX Functions

You can use MIN and MAX for numeric, character, and date data types.

```
SELECT MIN(hire_date), MAX(hire_date)  
FROM employees;
```

You can use the MAX and MIN functions for numeric, character, and date data types. example displays the most junior and most senior employees.

The following example displays the employee last name that is first and the employee last name that is last in an alphabetized list of all employees:

```
SELECT MIN(last_name), MAX(last_name)  
FROM employees;
```

**Note:** The AVG, SUM, VARIANCE, and STDDEV functions can be used only with numeric data types. MAX and MIN cannot be used with LOB or LONG data types.

### Using the COUNT Function

COUNT(\*) returns the number of rows in a table:

```
SELECT COUNT(*)  
FROM employees
```

WHERE department\_id = 50;

COUNT(*expr*) returns the number of rows with nonnull values for the *expr*:

```
SELECT COUNT(commission_pct)  
FROM employees
```

WHERE department\_id = 80;

### Using the DISTINCT Keyword

- COUNT(DISTINCT *expr*) returns the number of distinct non-null values of the *expr*.

- To display the number of distinct department values in the EMPLOYEES table:

```
SELECT COUNT(DISTINCT department_id) FROM employees;
```



```
SELECT department_id, AVG(salary) FROM employees GROUP BY department_id
HAVING max(salary)>10000;
```

Example displays the job ID and total monthly salary for each job that has a total payroll exceeding \$13,000. The example excludes sales representatives and sorts the list by the total monthly salary.

```
SELECT job_id, SUM(salary) PAYROLL FROM employees WHERE job_id NOT LIKE
'%REP%'
GROUP BY job_id HAVING SUM(salary) > 13000 ORDER BY SUM(salary);
```

### Nesting Group Functions

#### **Display the maximum average salary:**

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

*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

```
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 job-id, Round (min (Salary)) as minimum,  
       Round (max (Salary)) as maximum,  
       Round (Sum (Salary)) as Sum,  
       Round (Avg (Salary)) as Average  
from employees group by job-id;
```

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 job-id, count (*) as number_of_people from  
employees where job-id = '<job-title>' group by job-id;
```

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 min\_salary from employees  
where manager-id is not NULL group by manager-id having  
min (Salary) > 6000 order by min\_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 Count (\*) as total-employees,  
Sum (Case when to\_char (hire-date, 'yy') = '95' Then 1 else 0 End) as hired-1995,  
Sum (Case when to\_char (hire-date, 'yy') = '96' Then 1 else 0 End) as hired-1996,  
Sum (Case when to\_char (hire-date, 'yy') = '97' Then 1 else 0 End) as hired-1997,  
Sum (Case when to\_char (hire-date, 'yy') = '98' Then 1 else 0 End) as hired-1998  
from employees;

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 job-id,  
Sum (Case when department-id = 20 Then Salary End) as dept 20,  
Sum (Case when department-id = 50 Then Salary End) as dept 50,  
Sum (Case when department-id = 80 Then Salary End) as dept 80,  
Sum (Case when department-id = 90 Then Salary End) as dept 90,  
Sum (Salary) as total-salary  
from employees where department-id IN (20, 50, 80, 90)  
group by job-id;

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.department\_name || '-' || l.city as "name-location",  
 count(e.employee\_id) as "Number of people",  
 Round (Avg(e.salary), 2) as Salary

from departments d join locations l on d.location\_id = l.location\_id  
 join employees e on d.department\_id = e.department\_id  
 group by d.department\_name, l.city;

Evaluation Procedure	Marks awarded
Query (3)	5
Execution (3)	9
View (3)	5
Total (15)	15
Faculty Signature	<i>P. A. Chitra</i>



## Practice Questions

### Date Functions

1. For DJs on Demand, display the number of months between the event\_date of the Vigil wedding and today's date. Round to the nearest month.

Select Round (Months - Between (Sysdate, event\_date)) as month  
from events where event\_name = 'vigil wedding';

2. Display the days between the start of last summer's school vacation break and the day school started this year. Assume 30.5 days per month. Name the output "Days."

Select Round ((School\_start\_date - vacation\_start\_date) \* 30.5)  
as days from School\_Calendar;

3. Display the days between January 1 and December 31.

Select (To\_date ('31-DEC-2024', 'DD-MON-YYYY') - To\_date  
( '01-JAN-2024', 'DD-MON-YYYY')) as days from dual;

4. Using one statement, round today's date to the nearest month and nearest year and truncate it to the nearest month and nearest year. Use an alias for each column.

Select Round (Sysdate, 'Month') as Round\_month,  
Round (Sysdate, 'year') as Round\_year,  
Trunc (Sysdate, 'Month') as Trunc\_month,  
Trunc (Sysdate, 'year') as Trunc\_year  
from dual;

5. What is the last day of the month for June 2005? Use an alias for the output.

Select Last\_day (To\_date ('01-JUN-2005', 'DD-MON-YYYY'))  
as "Last\_Day" from dual;



6. Display the number of years between the Global Fast Foods employee Bob Miller's birthday and today. Round to the nearest year.

Select Round (months-between (sysdate, birth-date) / 12)  
as year from employees where last-name = 'Miller' and  
first-name = 'Bob';

7. Your next appointment with the dentist is six months from today. On what day will you go to the dentist? Name the output, "Appointment."

Select Add-months (sysdate, 6) as appointment from dual;

8. The teacher said you have until the last day of this month to turn in your research paper. What day will this be? Name the output, "Deadline."

Select last-day (sysdate) as deadline from dual;

9. How many months between your birthday this year and January 1 next year?

Select months-between-date ('01-jan-2025', 'DD-mon-yy'),  
To-date (To\_char (sysdate, 'yyy') || '-' <your-birthday-MM-DD  
>, 'yyy-MM-DD')) as months from dual;

10. What's the date of the next Friday after your birthday this year? Name the output, "First Friday."

Select next-day (To-date ('<your-birthday-MM-DD-yyy>',  
'MM-DD-yyy'), 'FRIDAY') as "First Friday" from dual;

11. Name a date function that will return a number.

Months-between ();



12. Name a date function that will return a date.

ADD. MONTHS ( )

13. Give one example of why it is important for businesses to be able to manipulate date data?

Businesses use date functions for data payroll processing, employee tenure calculation, sales forecasting, and billing cycles.

#### Conversion Functions

In each of the following exercises, feel free to use labels for the converted column to make the output more readable.

1. List the last names and birthdays of Global Fast Food Employees. Convert the birth dates to character data in the Month DD, YYYY format. Suppress any leading zeros.

Select last-name, To-char (birth-date, "Month 00, yyyy")  
as Birthday from employees;

2. Convert January 3, 04, to the default date format 03-Jan-2004.

Select To-date ('03-Jan-2004', 'DD-MON-YYYY') as  
Converted-date from dual;

3. Format a query from the Global Fast Foods f\_promotional\_menus table to print out the start\_date of promotional code 110 as: The promotion began on the tenth of February 2004.

Select 'The promotion began on the' || To-char (start-date,  
'DDth " of " month YYYY') as promo-date from  
f-promotional-menus where promo-code=110;



4. Convert today's date to a format such as: "Today is the Twentieth of March, Two Thousand Four"

Select 'Today is the ' || ToChar(sysdate, 'both "of" month, YYYY') as "Today" from dual;

5. List the ID, name and salary for all Global Fast Foods employees. Display salary with a \$ sign and two decimal places.

Select employee-id, last.name, ToChar(salary, '\$99,999.99') as salary from employees;

Evaluation Procedure	Marks awarded
Practice Evaluation (5)	5
Viva(5)	5
Total (10)	10
Faculty Signature	<del>P.M.</del> 8/9/25



