AIM: To describe the use of single row functions

EXERCISE-6

Date: 3/9/25

Evaluation Procedure	Marks awarded
Practice Evaluation (5)	5
Viva(5)	4
Total (10)	9
Faculty Signature	

Single Row Functions

Objective

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

- Describe various types of functions available in SQL.
- Use character, number and date functions in SELECT statement.
- Describe the use of conversion functions.

Single row functions:

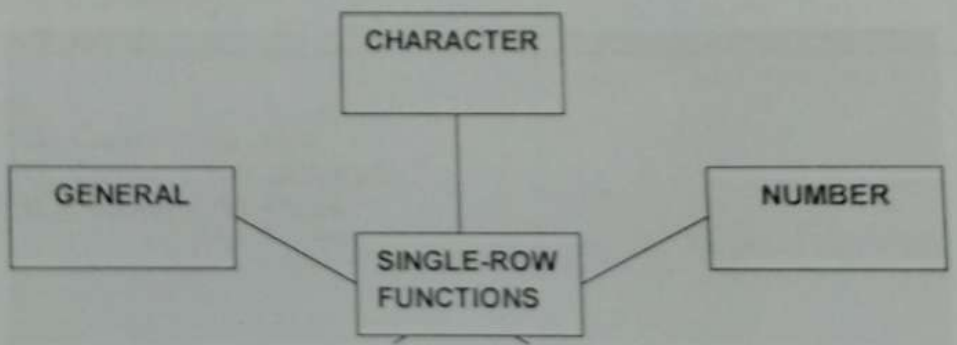
Manipulate data items.
Accept arguments and return one value.
Act on each row returned.
Return one result per row.
May modify the data type.
Can be nested.
Accept arguments which can be a column or an expression

Syntax

Function_name(arg1,...argn)

An argument can be one of the following

- ✓ User-supplied constant
- ✓ Variable value
- ✓ Column name
- ✓ Expression



Find the Solution for the following:

1. Write a query to display the current date. Label the column Date.

```
Select current_date() AS Date;
```

2. The HR department needs a report to display the employee number, last name, salary, and increased by 15.5% (expressed as a whole number) for each employee. Label the column New Salary.

```
Select employee_no, l_name, salary, ROUND(salary * 1.155)
AS New_Salary FROM employees;
```

3. Modify your query lab_03_02.sql to add a column that subtracts the old salary from the new salary. Label the column Increase.

```
Select emp_id, last_name, salary, ROUND(salary *1.155) AS
New_Salary, (ROUND(salary *1.155) - salary) AS Increase FROM
employees;
```

4. Write a query that displays the last name (with the first letter uppercase and all other letters lowercase) and the length of the last name for all employees whose name starts with the letters J, A, or M. Give each column an appropriate label. Sort the results by the employees' last names.

```
Select concat(Upper(Substr(last_name, 1,1)), Lower(Substr
(last_name,2))) AS last_name_formatted, LENGTH(l_name) AS
length_of_name FROM employees WHERE last_name like 'J%' OR
'A%' OR 'M%' ORDER BY last_name;
```

5. Rewrite the query so that the user is prompted to enter a letter that starts the last name. For example, if the user enters H when prompted for a letter, then the output should show all employees whose last name starts with the letter H.

```
Select emp_id, l_name FROM employees WHERE
l_name like CONCAT('&letter', '%');
```

6. The HR department wants to find the length of employment for each employee. For each employee, display the last name and calculate the number of months between today and the date on which the employee was hired. Label the column MONTHS_WORKED. Order your results by the number of months employed. Round the number of months up to the closest whole number.

```
Select l_name, CEIL(Datediff(curdate(), hire_date)/
30.4375) AS MONTHS_WORKED FROM employees
ORDER BY MONTHS_WORKED;
```

Note: Your results will differ.

7) Date

9/8/2025

2)

ID	l_name	Salary	New Salary
4	—	—	—
1	patel	1000	1155
3	Drexler	11000	1271

3)

ID	l_name	Salary	New salary	Increase
4	—	—	—	—
1	patel	1000	1155	155
3	Drexler	1100	1271	171

4)

~~id~~	l_name	length
	Messi	5

5)

First_name	l_name
Lionel	Messi

6)

l_name	Months_worked
Messi	42
Ray	254
patel	329
Drexler	376

7. Create a report that produces the following for each employee:
<employee last name> earns <salary> monthly but wants <3 times salary>. Label the column Dream Salaries.

```
Select CONCAT (last_name,' earns', salary, 'monthly
but wants', salary *3) AS Dream_Salaries FROM
employees;
```

8. Create a query to display the last name and salary for all employees. Format the salary to be 15 characters long, left-padded with the $ symbol. Label the column SALARY.

```
Select l_name, LPAD (salary, 15, '$') AS SALARY
FROM employees;
```

9. Display each employee's last name, hire date, and salary review date, which is the first Monday after six months of service. Label the column REVIEW. Format the dates to appear in the format similar to "Monday, the Thirty-First of July, 2000."

```
Select l_name, hire_date, DATE_FORMAT (DATE_ADDC
hire_date, INTERVAL 6 MONTH), '%.W, the %.D of %.M, %.Y'
) AS REVIEW FROM employees;
```

10. Display the last name, hire date, and day of the week on which the employee started. Label the column DAY. Order the results by the day of the week, starting with Monday.

```
Select l_name, hire_date, DAYNAME (hire_date) AS DAY
FROM employees ORDER BY FIELD(DAYNAME(hire_date),
'Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday', 'saturday',
'Sunday');
```

Evaluation Procedure	Marks awarded
Query(5)	5
Execution (5)	5
Viva(5)	4
Total (15)	14
Faculty Signature	

RESULT:
Thus the use of single rows are described.

7) · Dream Salaries

Patel earns 1000 monthly but ~~wants~~ wants 3000

Raj earns 1100 monthly but wants 3300

8) l_name Salary

Patel $ $ $ $ $ 10000

Dresler $ $ $ $ $ 1100

9) l_name Hire date Leviticus

Patel 4/5/1993 Monday the 12th of October 1993

Dresler 5/5/1994 Monday the 11th of October 1994

10) L_name Hire_date

Patel 4/5/1993

Dresler 5/5/1994

Practice Questions

Introduction to Functions

1. For each task, choose whether a single-row or multiple row function would be most appropriate:

a. Showing all of the email addresses in upper case letters Single - row function like UPPER()
b. Determining the average salary for the employees in the sales department Multiple - row functio
c. Showing hire dates with the month spelled out *(September 1, 2004)* Single - row funct Multiple - row function
d. Finding out the employees in each department that had the most seniority (the earliest hire date) Multiple - row functio
e. Displaying the employees' salaries rounded to the hundreds place
f. Substituting zeros for null values when displaying employee commissions. Single - row functio

2. The most common multiple-row functions are: AVG, COUNT, MAX, MIN, and SUM. Give your own definition for each of these functions.

- AVG() - Calculates the average of a set of numbers. It ignores NULL values.
- COUNT() - Counts the number of rows in a table or the number of non-NULL values in a specific column.
- MAX() - Finds the maximum of a given set of numbers.
- MIN() - Finds the minimum of a given set of numbers.

3. Test your definitions by substituting each of the multiple-row functions in this query. SUM() - Finds the total sum of a given set of numbers.

```
SELECT FUNCTION(salary)
FROM employees
```

Write out each query and its results.

1. Select AVG(salary) FROM employees;
   Result: This query would return a single value representing the average salary of all employees.
2. Select COUNT(salary) FROM employees;
   Result: This query would return a single value representing the number of employees who have a salary.
3. Select MAX(salary) FROM employees;
   Result: This query would return a single value representing the highest salary among all employees.

4. Select MIN(salary) FROM employees;

Result: This query would return a single value representing the lowest salary among all employees.

5. Select SUM(salary) FROM employees;

Result: This query would return a single value representing the total sum of all salaries paid to employees.

Case and Character Manipulation

1. Using the three separate words "Oracle," "Internet," and "Academy," use one command to produce the following output:
The Best Class Oracle Internet Academy

```
Select 'The Best Class ' || 'Oracle' || ' ' || 'Internet' || ' ' ||
'Academy' FROM DUAL;
```

2. Use the string "Oracle Internet Academy" to produce the following output:
The Net net

```
Select 'The' || substr ('Oracle Internet Academy', 8, 3) ||
' ' || substr ('Oracle Internet Academy', 8, 3) FROM DUAL
```

3. What is the length of the string "Oracle Internet Academy"?

```
select length ('Oracle Internet Academy') FROM DUAL;
```

The result is 23.

4. What's the position of "I" in "Oracle Internet Academy"?

```
Select INSTR('Oracle Internet Academy', 'I') FROM DUAL;
```

The result is 8

5. Starting with the string "Oracle Internet Academy", pad the string to create ****Oracle****Internet****Academy****

```
Select LPAD('Oracle', 8, '*') H RPAD ('Internet', 12, '*')
|| LPAD ('Academy', 12, '*') FROM DUAL;
```

Number Functions

1. Display Oracle database employee last_name and salary for employee_ids between 100 and 102. Include a third column that divides each salary by 1.55 and rounds the result to two decimal places.

```
Select l_name, salary, ROUND(salary/1.55,2) AS
calculated_salary FROM employees WHERE emp_id
Between 100 and 102;
```

2. Display employee last_name and salary for those employees who work in department 80. Give each of them a raise of 5.333% and truncate the result to two decimal places.

```
Select l_name, salary, TRUNC(salary * 1.05333,2) AS
new_salary FROM employees WHERE dept_id = 80;
```

3. Use a MOD number function to determine whether 38873 is an even number or an odd number.

```
Select MOD(38873,2) FROM DUAL;
```

38873 is an odd number

4. Use the DUAL table to process the following numbers:

845.553 - round to one decimal place
30695.348 - round to two decimal places
30695.348 - round to -2 decimal Places
2.3454 - truncate the 454 from the decimal place

```
Select ROUND(845.553,1) FROM DUAL;
Select ROUND(30695.348,2) FROM DUAL;
Select ROUND(30695.348,-2) FROM DUAL;
Select TRUNC(2.3454) FROM DUAL;
```

5. Divide each employee's salary by 3. Display only those employees' last names and salaries who earn a salary that is a multiple of 3.

```
select l_name, salary from employees where MOD(salary,3)
= 0;
```

6. Divide 34 by 8. Show only the remainder of the division. Name the output as EXAMPLE.

```
Select MOD(34,8) AS EXAMPLE from DUAL;
```

7. How would you like your paycheck – rounded or truncated? What if your paycheck was calculated to be $565.784 for the week, but you noticed that it was issued for $565.78. The loss of .004 cent would probably make very little difference to you. However, what if this was done to a thousand people, a 100,000 people, or a million people! Would it make a difference then? How much difference?

The difference in a single pay check is $.004. The difference for

1000 people is 1000 * 0.004 = $4. For 100,000 people the difference is 100000 * 0.004 = $400. For a million people the difference would be 1000000 * 0.004 = $4000

Evaluation Procedure	Marks awarded
Practice Evaluation (5)	5
Viva(5)	4
Total (10)	9
Faculty Signature	