**Single Row functions in SQL**

Single row functions are the one who work on single row and return one output per row. For example, length and case conversion functions are single row functions.

**Types of Single row function In SQL**

**Character Function**

Accepts character input and returns number or character value. Functions under the category are CONCAT, LENGTH, SUBSTR, INSTR, LPAD, RPAD, TRIM and REPLACE.

* CONCAT function concatenates two string values.
* LENGTH function returns the length of the input string.
* SUBSTR function returns a portion of a string from a given start point to an end point.
* INSTR function returns numeric position of a character or a string in a given string.
* LPAD and RPAD functions pad the given string up to a specific length with a given character.
* TRIM function trims the string input from the start or end.
* REPLACE function replaces characters from the input string with a given character.

Syntax:

SELECT CONCAT (first\_name, last\_name)

FROM employees

WHERE rownum < 5;

CONCAT(FIRST\_NAME,LAST\_NAME)

Example:

**General Function**

Usually contains NULL handling functions. The functions under the category are NVL, NVL2, NULLIF, COALESCE, CASE, DECODE.

Syntax:

SELECT first\_name, last\_name, salary, NVL (commission\_pct,0)

FROM employees

WHERE rownum < 5;

Using the NVL Function

Syntax:

SELECT last\_name, salary, NVL(commission\_pct, 0),

(salary\*12) + (salary\*12\*NVL(commission\_pct, 0)) AN\_SAL

FROM employees;

Using the NVL2 Function

Syntax:

SELECT last\_name, salary, commission\_pct, NVL2(commission\_pct, 'SAL+COMM', 'SAL') income

FROM employees

WHERE department\_id IN (50, 80);

Using the NULLIF Function

Syntax:

SELECT first\_name, LENGTH(first\_name) "expr1", last\_name, LENGTH(last\_name) "expr2",

NULLIF(LENGTH(first\_name), LENGTH(last\_name)) result

FROM employees;

Using the COALESCE Function

The advantage of the COALESCE function over the NVL function is that the COALESCE function can take multiple alternate values.

If the first expression is not null, it returns that expression; otherwise, it does a COALESCE of the remaining expressions.

Syntax:

SELECT last\_name,COALESCE(commission\_pct, salary, 10) comm

FROM employees

ORDER BY commission\_pct;

**Case conversion Function**

Accepts character input and returns a character value. Functions under the category are UPPER, LOWER and INITCAP.

* UPPER function converts a string to upper case.
* LOWER function converts a string to lower case.
* INITCAP function converts only the initial alphabets of a string to upper case.

Syntax:

SELECT UPPER (first\_name), INITCAP (last\_name), LOWER (job\_id)

FROM employees

WHERE rownum < 5;

**Date Function**

Date arithmetic operations return date or numeric values. Functions under the category are MONTHS\_BETWEEN, ADD\_MONTHS, NEXT\_DAY, LAST\_DAY, ROUND and TRUNC.

* MONTHS\_BETWEEN function returns the count of months between the two dates.
* ADD\_MONTHS function add 'n' number of months to an input date.
* NEXT\_DAY function returns the next day of the date specified.
* LAST\_DAY function returns last day of the month of the input date.
* ROUND and TRUNC functions are used to round and truncates the date value.

Syntax:

SELECT employee\_id, MONTHS\_BETWEEN (sysdate, hire\_date) Employment\_months

FROM employees

WHERE rownum < 5;

**Number Function**

Accepts numeric input and returns numeric values. Functions under the category are ROUND, TRUNC, and MOD.

* ROUND and TRUNC functions are used to round and truncate the number value.
* MOD is used to return the remainder of the division operation between two numbers.

Syntax:

SELECT ROUND (1372.472,1)

FROM dual;

SELECT TRUNC(45.923,2), TRUNC(45.923),

TRUNC(45.923,-2)

FROM DUAL;

SELECT last\_name, salary, MOD(salary, 5000)

FROM employees

WHERE job\_id = 'SA\_REP';