Database Programming

NULL Functions

ORACLE ACADEMY

Objectives

This lesson covers the following objectives:

- Demonstrate and explain the evaluation of a nested function
- List at least four general functions that work with any data type and relate to handling null values
- Explain the use of the COALESCE and the NVL functions
- Explain the use of general functions to deal with null values in data

Objectives (cont.)

This lesson covers the following objectives:

- Construct and execute a SQL query that correctly applies NVL, NVL2, NULLIF, and COALESCE single-row functions

Purpose

Besides functions that control how data is formatted or converted to another type, SQL uses a set of general functions designed specifically to deal with null values.

You may be wondering how a value that is unavailable, unassigned, unknown, or inapplicable can deserve so much attention. Null may be "nothing," but it can affect how expressions are evaluated, how averages are computed, and where a value appears in a sorted list. This lesson is all about handling null values.

How Functions are Evaluated

Up to now, you have applied single-row functions in simple statements. It is possible, however, to nest functions to any depth. It is important to know how nested functions are evaluated. “Nesting” refers to one thing being contained within another thing (like an egg contained within a nest.) The following example is a nested function. The evaluation process begins from the innermost level to the outermost level.

How Functions are Evaluated (cont.)

```
SELECT TO_CHAR(NEXT_DAY(ADD_MONTHS(hire_date, 6), 'FRIDAY'),
'fmDay, Month DDth, YYYY')  AS "Next Evaluation"
FROM employees
WHERE employee_id=100;
```

The results are:

- Friday, December 18th, 1987

How Functions are Evaluated (cont.)

```
SELECT TO_CHAR(NEXT_DAY(ADD_MONTHS(hire_date, 6), 'FRIDAY'),
'fmDay, Month DDth, YYYY')  AS "Next Evaluation"
FROM employees
WHERE employee_id=100;
```

- Step 1: The hire date is going to have six months added to it.
- Step 2: The first Friday following the future day will be identified.
- Step 3: The default date format will be formatted to read and display the Friday in a format similar to: Friday, December 18th, 1987, and will appear in the output under the column name "Next Evaluation."

Functions Pertaining to Null Values

At the beginning of the course, the term "null" was introduced. Null is the value that is unavailable, unassigned, unknown, or inapplicable.

As a result, we cannot test to see if it is the same as another value, because we do not know what value it has. It isn't equal to anything, not even zero!

But just because it really isn't anything doesn't mean that it is not important.

Functions Pertaining to Null Values (cont.)

Imagine this question: Is it true that X = Y? In order to answer you have to know the values of X and Y. Oracle has four general functions that pertain to the use of null values.

The four functions are:

- NVL
- NVL2
- NULLIF
- COALESCE

NVL Function

The NVL function converts a null value to a known value of a fixed datatype, either date, character, or number. The data types of the null value column and the new value must be the same. The NVL function is:

NVL (value or column that may contain a null, value to substitute for null)

NVL Function (cont.)

The following query uses the NVL function with character data types:

```
SELECT NVL(comments, 'no comment')
FROM D_PLAY_LIST_ITEMS;
```

NVL Function (cont.)

The data types of the null value column and the new value must be the same as shown in the following examples:

```
NVL(auth_expense_amt,0)
NVL(hire_date,'01-JAN-1997')
NVL(speciality,'None Yet')
```

NVL Function (cont.)

You can use the NVL function to convert column values containing nulls to a number before doing calculations. When an arithmetic calculation is performed with null, the result is null. The NVL function can convert the null value to a number before arithmetic calculations are done to avoid a null result.

NVL Function (cont.)

In the example, the auth_expense_amt column in the D_PARTNERS table contains null values. The NVL function is used to change the null to zero before arithmetic calculations.

```
SELECT first_name,
         last_name,NVL(auth_expense_amt, 0) * 1.05 AS Expenses
FROM D_Partners;
```

NVL2 Function

The NVL2 function evaluates an expression with three values. If the first value is not null, then the NVL2 function returns the second expression.

If the first value is null, then the third expression is returned. The values in expression 1 can have any data type.

Expression 2 and expression 3 can have any data type except LONG.

NVL2 Function (cont.)

The data type of the returned value is always the same as the data type of expression 2, unless expression 2 is character data, in which case the returned type is VARCHAR2.

NVL2 Function (cont.)

The NVL2 function is:

```
NVL2 (expression 1 value that may contain a null, expression 2
value to return if expression 1 is not null, expression 3 value to
replace if expression 1 is null)
```

An easy way to remember NVL2 is to think, "if expression 1 has a value, substitute expression 2; if expression 1 is null, substitute expression 3."

NVL2 Function (cont.)

The NVL2 function shown has number data for expression 1 and character data for expressions 2 and 3.

```
SELECT last_name, salary,
      NVL2(commission_pct, salary + (salary * commission_pct),
salary) AS income
FROM employees;
```

NULLIF Function

The NULLIF function compares two functions. If they are equal, the function returns null. If they are not equal, the function returns the first expression.

The NULLIF function is:

```
NULLIF(expression 1, expression 2)
```

NULLIF Function (cont.)

```
SELECT first_name, LENGTH(first_name) "Length FN",
last_name, LENGTH(last_name) "Length LN",
NULLIF(LENGTH(first_name),
LENGTH(last_name)) AS "Compare Them"
FROM D_PARTNERS;
```

FIRST_NAME	Length FN	LAST_NAME	Length LN	Compare Them
Jennifer	8	Cho	3	8
Jason	5	Tsang	5	(null)
Allison	7	Plumb	5	7

COALESCE Function

The COALESCE function is an extension of the NVL function, except COALESCE can take multiple values. The word "coalesce" literally means "to come together" and that is what happens.

If the first expression is null, the function continues down the line until a not null expression is found. Of course, if the first expression has a value, the function returns the first expression and the function stops.

COALESCE Function (cont.)

The COALESCE function is:

```
COALESCE (expression 1, expression 2, ...expression n)
```

COALESCE Function (cont.)

Examine the SELECT statement from the employees table shown at right. Which employees do not receive a commission? How can you tell? Is there anyone who receives neither a commission percentage nor a salary?

```
SELECT    last_name,
COALESCE(commission_pct,
salary, 10) comm
FROM      employees
ORDER BY commission_pct;
```

Grant	.15
Zlotkey	.2
Taylor	.2
Abel	.3
King	24000
Kochhar	17000

Terminology (cont.)

Key terms used in this lesson included:

- NVL
- NVL2
- NULLIF
- COALESCE

Summary

In this lesson, you should have learned how to:

- Demonstrate and explain the evaluation of a nested function
- List at least four general functions that work with any data type and relate to handling null values
- Explain the use of the COALESCE and the NVL functions
- Explain the use of general functions to deal with null values in data

Summary (cont.)

In this lesson, you should have learned how to:

- Construct and execute a SQL query that correctly applies NVL, NVL2, NULLIF, and COALESCE single-row functions