

## Using Explicit Cursor Attributes

### Terminology

Directions: Identify the vocabulary word for each definition below:

1. \_\_\_\_\_ Declares a record with the same fields as the cursor on which it is based.
2. \_\_\_\_\_ A composite data type in PL/SQL, consisting of a number of fields each with their own name and data type.
3. \_\_\_\_\_ Returns the status of the cursor.
4. \_\_\_\_\_ An attribute that processes an exact number of rows or counts the number of rows fetched in a loop.
5. \_\_\_\_\_ An attribute used to determine whether the most recent FETCH statement successfully returned a row.

### Try It/Solve It

1. In your own words, explain the advantage of using %ROWTYPE to declare a record structure based on a cursor declaration.
2. Write a PL/SQL block to read through rows in the wf\_countries table for all countries in region 5 (South America region). For each selected country, display the country\_name, national\_holiday\_date, and national\_holiday\_name. Use a record structure to hold all the columns selected from the wf\_countries table.

Hint: This exercise is very similar to question 4G in the previous lesson. Use your solution as a starting point for this exercise.

3. For this exercise, you use the employees table. Create a PL/SQL block that fetches and displays the six employees with the highest salary. For each of these employees, display the first name, last name, job id and salary. Order your output so that the employee with the highest salary is displayed first. Use %ROWTYPE and the explicit cursor attribute %ROWCOUNT.
4. Look again at the block you created in question 3. What if you wanted to display 21 employees instead of 6? There are only 20 rows in the employees table. What do you think would happen?
5. In real life we would not know how many rows the table contained. Modify your block from question 3 so that it will exit from the loop when either 21 rows have been fetched and displayed, or when there are no more rows to fetch. Test the block again.