**Open For Statement**

You may also use the OPEN-FOR statement with a cursor, which adds even more flexibility when processing cursors because you can assign the cursor to a different query.

SET SERVEROUTPUT ON

DECLARE

-- declare a REF CURSOR type named t\_product\_cursor

TYPE t\_product\_cursor IS

REF CURSOR RETURN products%ROWTYPE;

-- declare a t\_product\_cursor object named v\_product\_cursor

v\_product\_cursor t\_product\_cursor;

-- declare an object to store columns from the products table

-- named v\_product (of type products%ROWTYPE)

v\_product products%ROWTYPE;

BEGIN

-- assign a query to v\_product\_cursor and open it using OPEN-FOR

OPEN v\_product\_cursor FOR

SELECT \* FROM products WHERE product\_id < 5;

-- use a loop to fetch the rows from v\_product\_cursor into v\_product

LOOP

FETCH v\_product\_cursor INTO v\_product;

EXIT WHEN v\_product\_cursor%NOTFOUND;

DBMS\_OUTPUT.PUT\_LINE( 'product\_id = ' || v\_product.product\_id || ', name = ' || v\_product.name || ', price = ' || v\_product.price );

END LOOP;

-- close v\_product\_cursor

CLOSE v\_product\_cursor;

END;

A REF CURSOR is a pointer to a cursor, and is similar to a pointer in the C++ programming language.

**Unconstrained Cursors**

The cursors in the previous section all have a specific return type; these cursors are known as constrained cursors. The return type for a constrained cursor must match the columns in the query that is run by the cursor. An unconstrained cursor has no return type, and can therefore run any query.

SET SERVEROUTPUT ON

DECLARE

-- declare a REF CURSOR type named t\_cursor (this has no return

-- type and can therefore run any query)

TYPE t\_cursor IS REF CURSOR;

-- declare a t\_cursor object named v\_cursor

v\_cursor t\_cursor;

-- declare an object to store columns from the products table

-- named v\_product (of type products%ROWTYPE)

v\_product products%ROWTYPE;

-- declare an object to store columns from the customers table

-- named v\_customer (of type customers%ROWTYPE)

v\_customer customers%ROWTYPE;

BEGIN

-- assign a query to v\_cursor and open it using OPEN-FOR

OPEN v\_cursor FOR

SELECT \* FROM products WHERE product\_id < 5;

-- use a loop to fetch the rows from v\_cursor into v\_product

LOOP

FETCH v\_cursor INTO v\_product;

EXIT WHEN v\_cursor%NOTFOUND;

DBMS\_OUTPUT.PUT\_LINE( 'product\_id = ' || v\_product.product\_id || ', name = ' || v\_product.name || ', price = ' || v\_product.price );

END LOOP;

-- assign a new query to v\_cursor and open it using OPEN-FOR

OPEN v\_cursor FOR

SELECT \* FROM customers WHERE customer\_id < 3;

-- use a loop to fetch the rows from v\_cursor into v\_product

LOOP

FETCH v\_cursor INTO v\_customer;

EXIT WHEN v\_cursor%NOTFOUND;

DBMS\_OUTPUT.PUT\_LINE( 'customer\_id = ' || v\_customer.customer\_id || ', first\_name = ' || v\_customer.first\_name || ', last\_name = ' || v\_customer.last\_name );

END LOOP;

-- close v\_cursor

CLOSE v\_cursor;

END;