*Edson Sanchez* ITI-3400 Database Programming

**Five-day sale problem**

DECLARE

price bb\_product.price%TYPE;

productid bb\_product.idproduct%TYPE := 1;

sale NUMBER(2,2) :=.05;

BEGIN

SELECT

price

INTO price

FROM

bb\_product

WHERE

idproduct = productid;

dbms\_output.put\_line('The original price is $' || price);

FOR i IN 1..5 LOOP

price := price - ( price \* sale );

dbms\_output.put\_line('The sale price on day #'

|| i

|| ' is $'

|| price);

END LOOP;

EXCEPTION

WHEN no\_data\_found THEN

dbms\_output.put\_line('Product ID does not exist!');

END;

The original price is $99.99

The sale price on day #1 is $94.99

The sale price on day #2 is $90.24

The sale price on day #3 is $85.73

The sale price on day #4 is $81.44

The sale price on day #5 is $77.37

PL/SQL procedure successfully completed.

**4-1**

DECLARE

CURSOR cur\_basket IS SELECT

bi.idbasket,

bi.quantity,

p.stock

FROM

bb\_basketitem bi

INNER JOIN bb\_product p USING ( idproduct )

WHERE

bi.idbasket = 6;

TYPE type\_basket IS RECORD ( basket bb\_basketitem.idbasket%TYPE,

qty bb\_basketitem.quantity%TYPE,

stock bb\_product.stock%TYPE );

rec\_basket type\_basket;

lv\_flag\_txt CHAR(1) := 'Y';

BEGIN

OPEN cur\_basket;

LOOP

FETCH cur\_basket INTO rec\_basket;

EXIT WHEN cur\_basket%notfound;

IF

rec\_basket.stock < rec\_basket.qty

THEN

lv\_flag\_txt := 'N';

END IF;

END LOOP;

CLOSE cur\_basket;

IF

lv\_flag\_txt = 'Y'

THEN

dbms\_output.put\_line('All items in stock!');

END IF;

IF

lv\_flag\_txt = 'N'

THEN

dbms\_output.put\_line('All items NOT in stock!');

END IF;

END;

All items in stock!

PL/SQL procedure successfully completed.

**4-2**

DECLARE

CURSOR cur\_shopper IS SELECT

a.idshopper,

a.promo,

b.total

FROM

bb\_shopper a,

(

SELECT

b.idshopper,

SUM(bi.quantity \* bi.price) total

FROM

bb\_basketitem bi,

bb\_basket b

WHERE

bi.idbasket = b.idbasket

GROUP BY

idshopper

) b

WHERE

a.idshopper = b.idshopper

FOR UPDATE OF a.idshopper NOWAIT;

lv\_promo\_txt CHAR(1);

BEGIN

FOR rec\_shopper IN cur\_shopper LOOP

lv\_promo\_txt := 'X';

IF

rec\_shopper.total > 100

THEN

lv\_promo\_txt := 'A';

END IF;

IF

rec\_shopper.total BETWEEN 50 AND 99

THEN

lv\_promo\_txt := 'B';

END IF;

IF

lv\_promo\_txt <> 'X'

THEN

UPDATE bb\_shopper

SET

promo = lv\_promo\_txt

WHERE

CURRENT OF cur\_shopper;

END IF;

END LOOP;

COMMIT;

END;

PL/SQL procedure successfully completed.

**4-3**

UPDATE bb\_shopper

SET

promo = NULL;

UPDATE bb\_shopper

SET

promo = 'B'

WHERE

idshopper IN (

21,

23,

25

);

UPDATE bb\_shopper

SET

promo = 'A'

WHERE

idshopper = 22;

COMMIT;

BEGIN

UPDATE bb\_shopper

SET

promo = NULL

WHERE

promo IS NOT NULL;

dbms\_output.put\_line(SQL%rowcount);

END;

4

PL/SQL procedure successfully completed.