*Edson Sanchez* ITI-3400 Database Programming

**6-1**

CREATE OR REPLACE FUNCTION dollar\_fmt\_sf (

p\_num NUMBER

) RETURN VARCHAR2 IS

lv\_amt\_txt VARCHAR2(20);

BEGIN

lv\_amt\_txt := TO\_CHAR(p\_num,'$99,999.99');

RETURN lv\_amt\_txt;

END;

Function DOLLAR\_FMT\_SF compiled

DECLARE

lv\_amt\_num NUMBER(8,2) := 9999.55;

BEGIN

dbms\_output.put\_line(dollar\_fmt\_sf(lv\_amt\_num) );

END;

$9,999.55

PL/SQL procedure successfully completed.

**6-2**

CREATE OR REPLACE FUNCTION tot\_purch\_sf (

f\_shopid NUMBER

) RETURN NUMBER IS

lv\_total NUMBER(7,2);

BEGIN

SELECT

SUM(total)

INTO lv\_total

FROM

bb\_basket

WHERE

idshopper = f\_shopid;

RETURN lv\_total;

END;

Function TOT\_PURCH\_SF compiled

SELECT

idshopper,

tot\_purch\_sf(idshopper) total

FROM

bb\_shopper;

IDSHOPPER TOTAL

21 66.76

22 210.21

23 81.75

24 61.1

25 83.66

26 0

27 43.38

7 rows selected.

**6-5**

CREATE OR REPLACE FUNCTION ord\_ship\_sf (

f\_baskid NUMBER

) RETURN VARCHAR2 IS

lv\_ship\_txt VARCHAR2(20);

lv\_orderdate DATE;

lv\_shipdate DATE;

BEGIN

SELECT

b.dtordered,

bs.dtstage

INTO

lv\_orderdate,

lv\_shipdate

FROM

bb\_basket b,

bb\_basketstatus bs

WHERE

b.idbasket = bs.idbasket

AND bs.idbasket = f\_baskid

AND bs.idstage = 5;

IF

lv\_shipdate - lv\_orderdate > 1

THEN

lv\_ship\_txt := 'CHECK';

ELSE

lv\_ship\_txt := 'OK';

END IF;

RETURN lv\_ship\_txt;

EXCEPTION

WHEN no\_data\_found THEN

lv\_ship\_txt := 'Not shipped';

RETURN lv\_ship\_txt;

END;

Function ORD\_SHIP\_SF compiled

BEGIN

dbms\_output.put\_line(ord\_ship\_sf(3) );

END;

CHECK

PL/SQL procedure successfully completed.

BEGIN

dbms\_output.put\_line(ord\_ship\_sf(4) );

END;

OK

PL/SQL procedure successfully completed.

BEGIN

dbms\_output.put\_line(ord\_ship\_sf(12) );

END;

Not Shipped

PL/SQL procedure successfully completed.

**6-6**

CREATE OR REPLACE FUNCTION status\_desc\_sf (

f\_stageid NUMBER

) RETURN VARCHAR2 IS

lv\_stat\_desc VARCHAR2(20);

BEGIN

CASE

WHEN f\_stageid = 1 THEN

lv\_stat\_desc := 'Order submitted';

WHEN f\_stageid = 2 THEN

lv\_stat\_desc := 'Accepted, sent to shipping';

WHEN f\_stageid = 3 THEN

lv\_stat\_desc := 'Back-Ordered';

WHEN f\_stageid = 4 THEN

lv\_stat\_desc := 'Cancelled';

WHEN f\_stageid = 5 THEN

lv\_stat\_desc := 'Shipped';

END CASE;

RETURN lv\_stat\_desc;

END;

Function STATUS\_DESC\_SF compiled

SELECT

idstage,

status\_desc\_sf(idstage) description

FROM

bb\_basketstatus

WHERE

idbasket = 4;

IDSTAGE DESCRIPTION

1 Order submitted

5 Shipped

**TAX\_CALC\_SF Function**

CREATE OR REPLACE FUNCTION tax\_calc\_sf (

f\_basketid NUMBER

) RETURN VARCHAR2 IS

lv\_tax\_total VARCHAR2(20);

BEGIN

SELECT

dollar\_fmt\_sf(b.subtotal \* t.taxrate)

INTO lv\_tax\_total

FROM

bb\_basket b,

bb\_tax t

WHERE

b.shipstate = t.state

AND b.idbasket = f\_basketid;

RETURN lv\_tax\_total;

EXCEPTION

WHEN no\_data\_found THEN

lv\_tax\_total := '$0';

RETURN lv\_tax\_total;

END;

Function TAX\_CALC\_SF compiled

SELECT

idbasket,

tax\_calc\_sf(idbasket) "Total Tax Amount"

FROM

bb\_basket

WHERE

idbasket = 4;

IDBASKET Total Tax Amount

4 $.86

SELECT

idbasket,

tax\_calc\_sf(idbasket) "Total Tax Amount"

FROM

bb\_basket

WHERE

idbasket = 3;

IDBASKET Total Tax Amount

3 $0