*Edson Sanchez* ITI-3400 Database Programming

**5-1**

CREATE OR REPLACE PROCEDURE prod\_name\_sp (

p\_prodid IN bb\_product.idproduct%TYPE,

p\_prodname IN bb\_product.productname%TYPE

)

IS

BEGIN

UPDATE bb\_product

SET

productname = p\_prodname

WHERE

idproduct = p\_prodid;

COMMIT;

END;

Procedure PROD\_NAME\_SP compiled

select idproduct, productname from bb\_product;

IDPRODUCT PRODUCTNAME

1 CapressoBar Model #351

2 Capresso Ultima

3 Eileen 4-cup French Press

4 Coffee Grinder

5 Sumatra

6 Guatamala

7 Columbia

8 Brazil

9 Ethiopia

10 Espresso

10 rows selected.

BEGIN

prod\_name\_sp(1,'CapressorBar Model #388');

END;

PL/SQL procedure successfully completed.

select idproduct, productname from bb\_product;

IDPRODUCT PRODUCTNAME

1 CapressorBar Model #388

2 Capresso Ultima

3 Eileen 4-cup French Press

4 Coffee Grinder

5 Sumatra

6 Guatamala

7 Columbia

8 Brazil

9 Ethiopia

10 Espresso

10 rows selected.

**5-2**

CREATE OR REPLACE PROCEDURE prod\_add\_sp (

prod\_id IN bb\_product.idproduct%TYPE,

prod\_name IN bb\_product.productname%TYPE,

description IN bb\_product.description%TYPE,

image\_file IN bb\_product.productimage%TYPE,

price IN bb\_product.price%TYPE,

status IN bb\_product.active%TYPE

)

IS

BEGIN

INSERT INTO bb\_product (

idproduct,

productname,

description,

productimage,

price,

active

) VALUES (

prod\_id,

prod\_name,

description,

image\_file,

price,

status

);

COMMIT;

END;

Procedure PROD\_ADD\_SP compiled

**The other parameter added was the product ID number as it needs one to be put into the table because it is a variable that cannot be NULL.**

BEGIN

prod\_add\_sp(11,'Roasted Blend','Well-balanced mix of roasted beans, a medium body.','roasted.jpg',9.50,1);

END;

PL/SQL procedure successfully completed.

SELECT

idproduct,

productname,

description,

productimage,

price,

active

FROM

bb\_product;

IDPRODUCT PRODUCTNAME DESCRIPTION PRODUCTIMAGE PRICE ACTIVE

---------- ------------------------- ---------------------------------------------------------------------------------------------------- ------------------------- ---------- ----------

1 CapressorBar Model #388 A fully programmable pump espresso machine and 10-cup coffee maker complete with GoldTone filter capresso.gif 99.99 1

2 Capresso Ultima Coffee and Espresso and Cappuccino Machine. Brews from one espresso to two six ounce cups of coffee capresso2.gif 129.99 1

3 Eileen 4-cup French Press A unique coffeemaker from those proud craftsmen in windy Normandy. frepress.gif 32.5 1

4 Coffee Grinder Avoid blade grinders! This mill grinder allows you to choose a fine grind to a coarse grind. grind.gif 28.5 1

5 Sumatra Spicy and intense with herbal aroma. sumatra.jpg 10.5 1

6 Guatamala heavy body, spicy twist, aromatic and smokey flavor. Guatamala.jpg 10 1

7 Columbia dry, nutty flavor and smoothness columbia.jpg 10.8 1

8 Brazil well-balanced mellow flavor, a medium body with hints of cocoa and a mild, nut-like aftertaste brazil.jpg 10.8 1

9 Ethiopia distinctive berry-like flavor and aroma, reminds many of a fruity, memorable wine. ethiopia.jpg 10 1

10 Espresso dense, caramel-like sweetness with a soft acidity. Roasted somewhat darker than traditional Italian. espresso.jpg 10 1

11 Roasted Blend Well-balanced mix of roasted beans, a medium body. roasted.jpg 9.5 1

11 rows selected.

**5-3**

CREATE OR REPLACE PROCEDURE tax\_cost\_sp (

p\_state IN bb\_tax.state%TYPE,

p\_amount IN NUMBER,

p\_taxamount OUT NUMBER

)

IS

BEGIN

SELECT

p\_amount \* taxrate

INTO p\_taxamount

FROM

bb\_tax

WHERE

state = p\_state;

END;

Procedure TAX\_COST\_SP compiled

DECLARE

tax NUMBER(5,2);

BEGIN

tax\_cost\_sp('VA',100,tax);

dbms\_output.put\_line(tax);

END;

4.5

PL/SQL procedure successfully completed.

**5-6**

CREATE OR REPLACE PROCEDURE status\_sp (

p\_id IN bb\_basketstatus.idbasket%TYPE,

p\_status OUT VARCHAR2,

p\_date OUT DATE

)

IS

BEGIN

SELECT

dtstage,

idstage

INTO

p\_date,

p\_status

FROM

bb\_basketstatus

WHERE

dtstage = (

SELECT

MAX(dtstage)

FROM

bb\_basketstatus

WHERE

idbasket = p\_id

)

GROUP BY

dtstage,

idstage;

CASE

WHEN p\_status = 1 THEN

p\_status := 'Submitted and received.';

WHEN p\_status = 2 THEN

p\_status := 'Confirmed, processed, sent to shipping.';

WHEN p\_status = 3 THEN

p\_status := 'Shipped.';

WHEN p\_status = 4 THEN

p\_status := 'Cancelled.';

WHEN p\_status = 5 THEN

p\_status := 'Back-ordered.';

END CASE;

EXCEPTION

WHEN no\_data\_found THEN

p\_status := 'No status available.';

END;

Procedure STATUS\_SP compiled

**Not sure if I did this how it was intended.**

DECLARE

status VARCHAR2(25);

recent\_dt DATE;

BEGIN

status\_sp(4,status,recent\_dt);

dbms\_output.put\_line('DATE:'

|| recent\_dt

|| ' STATUS:'

|| status);

END;

DATE:14-FEB-12 STATUS:Back-ordered.

PL/SQL procedure successfully completed.

DECLARE

status VARCHAR2(25);

recent\_dt DATE;

BEGIN

status\_sp(6,status,recent\_dt);

dbms\_output.put\_line('DATE:'

|| recent\_dt

|| ' STATUS:'

|| status);

END;

DATE: STATUS:No status available.

PL/SQL procedure successfully completed.