--UPDATE MEMBER\_STAR

DECLARE

STAR NUMBER(20);

CUS\_ID\_COUNT NUMBER(20);

return\_value number;

return\_value2 number;

MEMBER\_NAME VARCHAR2(20);

ID\_NUMBER NUMBER(20);

BEGIN

IF :TOTAL\_S.MEMBER\_ID IS NOT NULL THEN

SELECT COUNT(\*) INTO CUS\_ID\_COUNT FROM MEMBER

WHERE MEM\_ID =:TOTAL\_S.MEMBER\_ID;

IF CUS\_ID\_COUNT = 0 THEN

-- MESSAGE('This ID does not exist. Please type again.');

-- RAISE FORM\_TRIGGER\_FAILURE;

SET\_ALERT\_PROPERTY('ALERT\_CHECKOUT', alert\_message\_text,'This ID does not exist. Please type again.');

return\_value:=show\_alert('ALERT\_CHECKOUT');

IF return\_value= alert\_BUTTON1 THEN

RAISE FORM\_TRIGGER\_FAILURE;

END IF;

ELSIF CUS\_ID\_COUNT =1 THEN

ID\_NUMBER:=:TOTAL\_S.MEMBER\_ID;

SELECT MEM\_NAME INTO MEMBER\_NAME FROM MEMBER

WHERE MEM\_ID =:TOTAL\_S.MEMBER\_ID;

SET\_ALERT\_PROPERTY('ALERT\_MEM\_ID\_FOR\_STAR', alert\_message\_text,'Are you proceed to add the star of ' ||MEMBER\_NAME||' with the ID ' ||ID\_NUMBER);

return\_value:=show\_alert('ALERT\_MEM\_ID\_FOR\_STAR');

IF return\_value = alert\_BUTTON1 THEN

SELECT MEM\_STAR INTO STAR FROM MEMBER WHERE MEM\_ID=:TOTAL\_S.MEMBER\_ID;

UPDATE MEMBER

SET MEM\_STAR=STAR+:TOTAL\_S.TOTAL

WHERE MEM\_ID =:TOTAL\_S.MEMBER\_ID;

forms\_ddl ('Commit');

ELSE RAISE FORM\_TRIGGER\_FAILURE;

END IF;

END IF;

END IF;

END;

DECLARE

CURRENT\_ORDER\_ID NUMBER(20);

NEW\_ORDER\_ID NUMBER(20);

BEGIN

SELECT MAX(ORDERS\_ID) INTO CURRENT\_ORDER\_ID FROM ORDERS;

NEW\_ORDER\_ID:=CURRENT\_ORDER\_ID+1;

INSERT INTO ORDERS (ORDERS\_ID,ORDERS\_STATUS,ORDERS\_AMOUNT,MEM\_ID) VALUES (NEW\_ORDER\_ID,'PENDING',:TOTAL\_S.TOTAL,:TOTAL\_S.MEMBER\_ID);

forms\_ddl ('Commit');

END;

DECLARE

Current\_DETAIL\_ID NUMBER (20);

New\_DETAIL\_ID NUMBER(20);

PROD\_ID NUMBER(20);

ORD\_ID NUMBER(20);

DETAIL\_Q NUMBER(20);

USED\_QUANTITY NUMBER(10);

INV\_NUMBER NUMBER(10);

ING\_NUMBER NUMBER(10);

i NUMBER(10);

BEGIN

FIRST\_RECORD;

LOOP

GO\_BLOCK('SC\_S');

--Generate DETAIL\_ID

SELECT MAX (DETAIL\_ID) INTO Current\_DETAIL\_ID FROM DETAIL;

New\_DETAIL\_ID:=Current\_DETAIL\_ID+1;

SELECT PRODUCT\_ID INTO PROD\_ID FROM PRODUCT

WHERE PRODUCT\_NAME = Lower(:SC\_S.PRODUCT\_NAME);

SELECT MAX(ORDERS\_ID) INTO ORD\_ID FROM ORDERS;

INSERT INTO DETAIL (DETAIL\_ID,DETAIL\_TEMPERATURE,DETAIL\_COLDNESS,DETAIL\_SWEETNESS,

DETAIL\_ADDPARTS,DETAIL\_QUANTITY,DETAIL\_STATUS,PRODUCT\_ID,ORDERS\_ID)

VALUES (New\_DETAIL\_ID,:SC\_S.DETAIL\_TEMPERATURE,:SC\_S.DETAIL\_COLDNESS,

:SC\_S.DETAIL\_SWEETNESS,:SC\_S.DETAIL\_ADDPARTS,

:SC\_S.DETAIL\_QUANTITY,'PENDING',PROD\_ID,ORD\_ID);

SELECT DETAIL\_QUANTITY

INTO DETAIL\_Q

FROM DETAIL

WHERE PRODUCT\_ID= PROD\_ID;

--UPDATE THE LATEST QUANTITY OF THE INVENTORY

SELECT COUNT(ING\_ID) INTO ING\_NUMBER

FROM(

SELECT INGREDIENT.ING\_ID

FROM DETAIL, INGREDIENT

WHERE DETAIL.PRODUCT\_ID = INGREDIENT.PRODUCT\_ID AND DETAIL\_ID = New\_DETAIL\_ID

GROUP BY INGREDIENT.ING\_ID);

FOR i IN 1..ING\_NUMBER

LOOP

SELECT Y\*Q , INVENTORY\_ID

INTO USED\_QUANTITY, INV\_NUMBER

FROM(

SELECT INVENTORY.INVENTORY\_ID, DETAIL\_QUANTITY AS Y, ING\_QUANTITY AS Q,ROWNUM RN

FROM DETAIL, INGREDIENT,INVENTORY

WHERE DETAIL.PRODUCT\_ID = INGREDIENT.PRODUCT\_ID

AND INVENTORY.INVENTORY\_ID = INGREDIENT.INVENTORY\_ID

AND DETAIL\_ID = New\_DETAIL\_ID

GROUP BY INVENTORY.INVENTORY\_ID, DETAIL\_QUANTITY, ING\_QUANTITY,ROWNUM)

WHERE RN = i;

UPDATE INVENTORY

SET INVENTORY\_QUANTITY = INVENTORY\_QUANTITY - USED\_QUANTITY

WHERE INVENTORY\_ID = INV\_NUMBER;

END LOOP;

forms\_ddl('Commit');

EXIT WHEN :SYSTEM.LAST\_RECORD = 'TRUE';

NEXT\_RECORD;

END LOOP;

GO\_BLOCK('QUANTITY\_S');

CLEAR\_BLOCK(NO\_VALIDATE);

GO\_BLOCK('TRANS\_COM\_S');

END;