

UCS1412 – DBMS Lab
Assignment – 5

Name: Jayannthan P T

Dept: CSE 'A'

Roll No.: 205001049

Title: Control Structures

Spool File Output:

```
SQL>
-- 1. Check whether the given combination of food and flavor is available. If any one or
both are not available, display the relevant message.
SQL> SET SERVEROUTPUT ON;
SQL> DECLARE
  2     PRODUCTROWS PRODUCTS%ROWTYPE;
  3     COUNT_PRODUCTS NUMBER;
  4     SEARCHFOOD PRODUCTS.FOOD%TYPE;
  5     SEARCHFLAVOR PRODUCTS.FLAVOR%TYPE;
  6 BEGIN
  7     SEARCHFOOD:='&FOOD_TO_SEARCH';
  8     SEARCHFLAVOR:='&FLAVOR_TO_SEARCH';
  9     SELECT COUNT(*) INTO COUNT_PRODUCTS FROM PRODUCTS WHERE
LOWER(FOOD)=LOWER(SEARCHFOOD) AND LOWER(FLAVOR)=LOWER(SEARCHFLAVOR);
 10     IF COUNT_PRODUCTS!=0 THEN
 11         DBMS_OUTPUT.PUT_LINE('THE COMBINATION YOU ASKED FOR IS AVAILABLE');
 12         DBMS_OUTPUT.PUT_LINE('THERE IS ' || SEARCHFOOD || ' WITH ' || SEARCHFLAVOR || '
FLAVOR');
 13         SELECT * INTO PRODUCTROWS FROM PRODUCTS WHERE LOWER(FOOD)=LOWER(SEARCHFOOD)
AND LOWER(FLAVOR)=LOWER(SEARCHFLAVOR);
 14         DBMS_OUTPUT.PUT_LINE(PRODUCTROWS.PID || ' ' || PRODUCTROWS.FLAVOR || '
' || PRODUCTROWS.FOOD || ' ' || PRODUCTROWS.PRICE);
 15     END IF;
 16     IF COUNT_PRODUCTS=0 THEN
 17         SELECT COUNT(*) INTO COUNT_PRODUCTS FROM PRODUCTS WHERE
LOWER(FOOD)=LOWER(SEARCHFOOD);
 18     IF COUNT_PRODUCTS!=0 THEN
 19         DBMS_OUTPUT.PUT_LINE('THE COMBINATION YOU ASKED FOR IS NOT AVAILABLE');
 20         DBMS_OUTPUT.PUT_LINE('THE FOOD YOU ASKED FOR IS AVAILABLE IN COMNIATION WITH
OTHER FLAVORS');
 21     END IF;
 22     END IF;
 23     IF COUNT_PRODUCTS=0 THEN
 24         SELECT COUNT(*) INTO COUNT_PRODUCTS FROM PRODUCTS WHERE
LOWER(FLAVOR)=LOWER(SEARCHFLAVOR);
 25     IF COUNT_PRODUCTS!=0 THEN
 26         DBMS_OUTPUT.PUT_LINE('THE COMBINATION YOU ASKED FOR IS NOT AVAILABLE');
```

```

27         DBMS_OUTPUT.PUT_LINE('THE FLAVOR YOU ASKED FOR IS AVAILABLE IN COMNIATION
WITH OTHER FOODS');
28     END IF;
29     END IF;
30     IF COUNT_PRODUCTS=0 THEN
31         SELECT COUNT(*) INTO COUNT_PRODUCTS FROM PRODUCTS WHERE
LOWER(FOOD)<>LOWER(SEARCHFOOD) AND LOWER(FLAVOR)<>LOWER(SEARCHFLAVOR);
32     IF COUNT_PRODUCTS!=0 THEN
33         DBMS_OUTPUT.PUT_LINE('THE COMBINATION YOU ASKED FOR IS NOT AVAILABLE');
34         DBMS_OUTPUT.PUT_LINE('BOTH FOOD AND FLAVOR IS NOT AVAILABLE');
35     END IF;
36     END IF;
37 END;
38 /

```

```

Enter value for food_to_search: cake
old 7: SEARCHFOOD:='&FOOD_TO_SEARCH';
new 7: SEARCHFOOD:='cake';
Enter value for flavor_to_search: lemon
old 8: SEARCHFLAVOR:='&FLAVOR_TO_SEARCH';
new 8: SEARCHFLAVOR:='lemon';
THE COMBINATION YOU ASKED FOR IS AVAILABLE
THERE IS cake WITH lemon FLAVOR
20-BC-L-10 Lemon Cake 8.95

```

PL/SQL procedure successfully completed.

```

SQL> /
Enter value for food_to_search: cake
old 7: SEARCHFOOD:='&FOOD_TO_SEARCH';
new 7: SEARCHFOOD:='cake';
Enter value for flavor_to_search: coco
old 8: SEARCHFLAVOR:='&FLAVOR_TO_SEARCH';
new 8: SEARCHFLAVOR:='coco';
THE COMBINATION YOU ASKED FOR IS NOT AVAILABLE
THE FOOD YOU ASKED FOR IS AVAILABLE IN COMNIATION WITH OTHER FLAVORS

```

PL/SQL procedure successfully completed.

```

SQL> /
Enter value for food_to_search: cake
old 7: SEARCHFOOD:='&FOOD_TO_SEARCH';
new 7: SEARCHFOOD:='cake';
Enter value for flavor_to_search: old 8: SEARCHFLAVOR:='&FLAVOR_TO_SEARCH';
new 8:

```

PL/SQL procedure successfully completed.

```

SQL> /
Enter value for food_to_search: coco
old 7: SEARCHFOOD:='&FOOD_TO_SEARCH';
new 7: SEARCHFOOD:='coco';
Enter value for flavor_to_search: lemon
old 8: SEARCHFLAVOR:='&FLAVOR_TO_SEARCH';

```

```

new 8:      SEARCHFLAVOR:='lemon';
THE COMBINATION YOU ASKED FOR IS NOT AVAILABLE
THE FOOD YOU ASKED FOR IS AVAILABLE IN COMNIATION WITH OTHER FLAVORS
THE COMBINATION YOU ASKED FOR IS NOT AVAILABLE
THE FLAVOR YOU ASKED FOR IS AVAILABLE IN COMNIATION WITH OTHER FOODS

```

PL/SQL procedure successfully completed.

-- 2. On a given date, find the number of items sold (Use Implicit cursor).

```
SQL> SET SERVEROUTPUT ON;
```

```
SQL> DECLARE
```

```
2     COUNTITEMS NUMBER;
```

```
3     GIVENDATE DATE;
```

```
4 BEGIN
```

```
5     GIVENDATE:=TO_DATE('&SEARCHDATE','DD-MM-YY');
```

```
6     SELECT COUNT(ITEM) INTO COUNTITEMS FROM ITEM_LIST JOIN RECEIPTS USING(RNO) GROUP
BY RDATE HAVING RDATE=GIVENDATE;
```

```
7     DBMS_OUTPUT.PUT_LINE('THE NUMBER OF ITEM SOLD ON '||GIVENDATE||' =
'||COUNTITEMS);
```

```
8     EXCEPTION
```

```
9     WHEN NO_DATA_FOUND THEN
```

```
10    DBMS_OUTPUT.PUT_LINE('NO ITEMS SOLD ON '||GIVENDATE);
```

```
11 END;
```

```
12 /
```

Enter value for searchdate: 12-10-07

```
old 5:      GIVENDATE:=TO_DATE('&SEARCHDATE','DD-MM-YY');
```

```
new 5:      GIVENDATE:=TO_DATE('12-10-07','DD-MM-YY');
```

```
THE NUMBER OF ITEM SOLD ON 12-OCT-07 = 37
```

PL/SQL procedure successfully completed.

```
SQL> /
```

Enter value for searchdate: 12-12-07

```
old 5:      GIVENDATE:=TO_DATE('&SEARCHDATE','DD-MM-YY');
```

```
new 5:      GIVENDATE:=TO_DATE('12-12-07','DD-MM-YY');
```

```
NO ITEMS SOLD ON 12-DEC-07
```

PL/SQL procedure successfully completed.

-- 3. An user desired to buy the product with the specific price. Ask the user for a price,

-- find the food item(s) that is equal or closest to the desired price. Print the product

-- number, food type, flavor and price. Also print the number of items that is equal or

-- closest to the desired price.

```
SQL> SET SERVEROUTPUT ON;
```

```
SQL> DECLARE
```

```
2     SPECIFICPRICE PRODUCTS.PRICE%TYPE;
```

```
3     CURSOR SPECIFIC_PRODUCTS IS SELECT * FROM PRODUCTS WHERE PRICE BETWEEN
FLOOR(SPECIFICPRICE-0.5) AND CEIL(SPECIFICPRICE+0.5);
```

```
4     PRODUCTROWS SPECIFIC_PRODUCTS%ROWTYPE;
```

```
5 BEGIN
```

```
6     SPECIFICPRICE:='&PRICE_TO_SEARCH';
```

```

7      OPEN SPECIFIC_PRODUCTS;
8      DBMS_OUTPUT.PUT_LINE('PRODUCT LIST:');
9      LOOP
10     FETCH SPECIFIC_PRODUCTS INTO PRODUCTROWS;
11     IF SPECIFIC_PRODUCTS%NOTFOUND THEN
12         EXIT;
13     ELSE
14         DBMS_OUTPUT.PUT_LINE(PRODUCTROWS.PID||' ' ||PRODUCTROWS.FOOD||'
'||PRODUCTROWS.FLAVOR||' ' ||PRODUCTROWS.PRICE);
15     END IF;
16 END LOOP;
17 DBMS_OUTPUT.PUT_LINE(SPECIFIC_PRODUCTS%ROWCOUNT||' PRODUCTS FOUND');
18 EXCEPTION
19     WHEN NO_DATA_FOUND THEN
20         DBMS_OUTPUT.PUT_LINE('NO PRODUCTS FOUND');
21     CLOSE SPECIFIC_PRODUCTS;
22 END;
23 /

```

Enter value for price_to_search: 2

old 6: SPECIFICPRICE:='&PRICE_TO_SEARCH';

new 6: SPECIFICPRICE:='2';

PRODUCT LIST:

```

70-GA Cookie Ganache 1.15
70-GON Cookie Gongolais 1.15
70-R Cookie Raspberry 1.09
70-M-CH-DZ Meringue Chocolate 1.25
70-M-VA-SM-DZ Meringue Vanilla 1.15
70-MAR Cookie Marzipan 1.25
70-TU Cookie Tuile 1.25
50-ALM Croissant Almond 1.45
50-APP Croissant Apple 1.45
50-APR Croissant Apricot 1.45
50-CHS Croissant Cheese 1.75
50-CH Croissant Chocolate 1.75
51-APR Danish Apricot 1.15
51-APP Danish Apple 1.15
51-ATW Twist Almond 1.15
51-BC Bear Claw Almond 1.95
51-BLU Danish Blueberry 1.15
17 PRODUCTS FOUND

```

PL/SQL procedure successfully completed.

-- 4. Display the customer name along with the details of item and its quantity ordered for

-- the given order number. Also calculate the total quantity ordered as shown below:

SQL>

SQL> DECLARE

```

2      ORDER_NUMBER ITEM_LIST.RNO%TYPE;
3      CFNAME CUSTOMERS.FNAME%TYPE;
4      CLNAME CUSTOMERS.LNAME%TYPE;
5      TOTAL_ITEM NUMBER:=0;

```

```

6      CURSOR DETAILS IS SELECT FOOD,FLAVOR,COUNT(*) AS QUANTITY FROM ITEM_LIST,PRODUCTS
WHERE RNO=ORDER_NUMBER AND PID=ITEM GROUP BY FOOD,FLAVOR;
7      DETAILROW DETAILS%ROWTYPE;
8  BEGIN
9      ORDER_NUMBER:='&ORDER_NUMBER_TO_SEARCH';
10     SELECT FNAME,LNAME INTO CFNAME,CLNAME FROM CUSTOMERS INNER JOIN RECEIPTS
USING(CID) WHERE RNO=ORDER_NUMBER;
11     DBMS_OUTPUT.PUT_LINE('CUSTOMER NAME: ' || CFNAME || ' ' || CLNAME);
12     OPEN DETAILS;
13     DBMS_OUTPUT.PUT_LINE('FOOD          FLAVOR          QUANTITY');
14     LOOP
15         FETCH DETAILS INTO DETAILROW;
16         IF DETAILS%NOTFOUND THEN
17             EXIT;
18         ELSE
19             DBMS_OUTPUT.PUT_LINE(DETAILROW.FOOD || '          ' || DETAILROW.FLAVOR || '
' || DETAILROW.QUANTITY);
20             TOTAL_ITEM:=TOTAL_ITEM+DETAILROW.QUANTITY;
21         END IF;
22     END LOOP;
23     DBMS_OUTPUT.PUT_LINE('TOTAL QUANTITY = ' || TOTAL_ITEM);
24     CLOSE DETAILS;
25 EXCEPTION
26     WHEN NO_DATA_FOUND THEN
27         DBMS_OUTPUT.PUT_LINE('INVALID ORDER NUMBER!');
28 END;
29 /

```

Enter value for order_number_to_search: 51991

old 9: ORDER_NUMBER:='&ORDER_NUMBER_TO_SEARCH';

new 9: ORDER_NUMBER:='51991';

CUSTOMER NAME: SOPKO RAYFORD

FOOD	FLAVOR	QUANTITY
Pie	Apple	1
Cake	Truffle	1
Tart	Apple	1
Tart	Chocolate	1
TOTAL QUANTITY = 4		

PL/SQL procedure successfully completed.

SQL> spool off;

SQL> /

Enter value for order_number_to_search: 1234

old 9: ORDER_NUMBER:='&ORDER_NUMBER_TO_SEARCH';

new 9: ORDER_NUMBER:='1234';

INVALID ORDER NUMBER!

PL/SQL procedure successfully completed.

SQL> spool off;