SQL> set serveroutput on

SQL> REM 1. Check whether the given combination of food and flavor is available. If any one or

SQL> REM both are not available, display the relevant message.

SQL>

SQL> DECLARE

2

3 id products.pid%type;

4 cfood products.food%type;

5 cflavor products.flavor%type;

6

7 cursor c1 is select food,flavor from products where food=cfood and flavor=cflavor ;

8 cursor c2 is select food from products where food=cfood;

9 cursor c3 is select flavor from products where flavor=cflavor ;

10

11 BEGIN

12 cfood:='&food';

13 cflavor:='&flavor';

14 open c1;

15 open c2;

16 open c3;

17 fetch c1 into cfood,cflavor;

18 fetch c2 into cfood;

19 fetch c3 into cflavor;

20 if c1%found then

21 dbms\_output.put\_line('Given food and flavor is found');

22 elsif c2%found then

23 dbms\_output.put\_line('Given food is found');

24 elsif c3%found then

25 dbms\_output.put\_line('Given flavor is found');

26 else

27 dbms\_output.put\_line('Given food and flavor not found');

28 end if;

29 END;

30 /

Enter value for food: Cookie

old 12: cfood:='&food';

new 12: cfood:='Cookie';

Enter value for flavor: Chocolate

old 13: cflavor:='&flavor';

new 13: cflavor:='Chocolate';

Given food is found

PL/SQL procedure successfully completed.

SQL> DECLARE

2

3 id products.pid%type;

4 cfood products.food%type;

5 cflavor products.flavor%type;

6

7 cursor c1 is select food,flavor from products where food=cfood and flavor=cflavor ;

8 cursor c2 is select food from products where food=cfood;

9 cursor c3 is select flavor from products where flavor=cflavor ;

10

11 BEGIN

12 cfood:='&food';

13 cflavor:='&flavor';

14 open c1;

15 open c2;

16 open c3;

17 fetch c1 into cfood,cflavor;

18 fetch c2 into cfood;

19 fetch c3 into cflavor;

20 if c1%found then

21 dbms\_output.put\_line('Given food and flavor is found');

22 elsif c2%found then

23 dbms\_output.put\_line('Given food is found');

24 elsif c3%found then

25 dbms\_output.put\_line('Given flavor is found');

26 else

27 dbms\_output.put\_line('Given food and flavor not found');

28 end if;

29 END;

30 /

Enter value for food: Twist

old 12: cfood:='&food';

new 12: cfood:='Twist';

Enter value for flavor: Lemon

old 13: cflavor:='&flavor';

new 13: cflavor:='Lemon';

Given food is found

PL/SQL procedure successfully completed.

SQL> DECLARE

2

3 id products.pid%type;

4 cfood products.food%type;

5 cflavor products.flavor%type;

6

7 cursor c1 is select food,flavor from products where food=cfood and flavor=cflavor ;

8 cursor c2 is select food from products where food=cfood;

9 cursor c3 is select flavor from products where flavor=cflavor ;

10

11 BEGIN

12 cfood:='&food';

13 cflavor:='&flavor';

14 open c1;

15 open c2;

16 open c3;

17 fetch c1 into cfood,cflavor;

18 fetch c2 into cfood;

19 fetch c3 into cflavor;

20 if c1%found then

21 dbms\_output.put\_line('Given food and flavor is found');

22 elsif c2%found then

23 dbms\_output.put\_line('Given food is found');

24 elsif c3%found then

25 dbms\_output.put\_line('Given flavor is found');

26 else

27 dbms\_output.put\_line('Given food and flavor not found');

28 end if;

29 END;

30 /

Enter value for food: Pizza

old 12: cfood:='&food';

new 12: cfood:='Pizza';

Enter value for flavor: Almond

old 13: cflavor:='&flavor';

new 13: cflavor:='Almond';

Given flavor is found

PL/SQL procedure successfully completed.

SQL> DECLARE

2

3 id products.pid%type;

4 cfood products.food%type;

5 cflavor products.flavor%type;

6

7 cursor c1 is select food,flavor from products where food=cfood and flavor=cflavor ;

8 cursor c2 is select food from products where food=cfood;

9 cursor c3 is select flavor from products where flavor=cflavor ;

10

11 BEGIN

12 cfood:='&food';

13 cflavor:='&flavor';

14 open c1;

15 open c2;

16 open c3;

17 fetch c1 into cfood,cflavor;

18 fetch c2 into cfood;

19 fetch c3 into cflavor;

20 if c1%found then

21 dbms\_output.put\_line('Given food and flavor is found');

22 elsif c2%found then

23 dbms\_output.put\_line('Given food is found');

24 elsif c3%found then

25 dbms\_output.put\_line('Given flavor is found');

26 else

27 dbms\_output.put\_line('Given food and flavor not found');

28 end if;

29 END;

30 /

Enter value for food: Burger

old 12: cfood:='&food';

new 12: cfood:='Burger';

Enter value for flavor: Mexican

old 13: cflavor:='&flavor';

new 13: cflavor:='Mexican';

Given food and flavor not found

PL/SQL procedure successfully completed.

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

SQL>

SQL> DECLARE

2

3 gn\_date date;

4 count\_val number(3);

5

6 BEGIN

7

8 gn\_date:='&date';

9 SELECT COUNT(\*) into count\_val FROM RECEIPTS NATURAL JOIN ITEM\_LIST

10 GROUP BY RDATE

11 HAVING RDATE=gn\_date;

12 dbms\_output.put\_line('No of items sold: ' || count\_val);

13 END;

14 /

Enter value for date: 23-Oct-2007

old 8: gn\_date:='&date';

new 8: gn\_date:='23-Oct-2007';

No of items sold: 18

PL/SQL procedure successfully completed.

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

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

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

SQL> REM closest to the desired price.

SQL>

SQL> DECLARE

2

3 cprice products.price%type;

4 prod\_row products%rowtype;

5 cursor c is select \* from products where

6 abs(cprice-price) = (select min(abs(cprice-price)) from products);

7

8 BEGIN

9

10 cprice:='&dprice';

11 dbms\_output.put\_line('ProductID Food Flavor Price');

12 dbms\_output.put\_line('---------------------------------');

13 open c;

14 fetch c into prod\_row;

15 while c%found loop

16 dbms\_output.put\_line(prod\_row.pid||' '||prod\_row.food||' '||prod\_row.flavor||' '||prod\_row.price);

17 fetch c into prod\_row;

18 end loop;

19 dbms\_output.put\_line('---------------------------------');

20 dbms\_output.put\_line(c%rowcount || 'product(s) found EQUAL/CLOSEST to given price');

21 close c;

22 END;

23 /

Enter value for dprice: 0.8

old 10: cprice:='&dprice';

new 10: cprice:='0.8';

ProductID Food Flavor Price

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

70-LEM Cookie Lemon .79

70-W Cookie Walnut .79

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

2product(s) found EQUAL/CLOSEST to given price

PL/SQL procedure successfully completed.

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

SQL> REM the given order number. Also calculate the total quantity ordered.

SQL>

SQL> DECLARE

2

3 cfname customers.fname%type;

4 clname customers.lname%type;

5 cfood products.food%type;

6 cflavor products.flavor%type;

7 qty number(3);

8 ono receipts.rno%type;

9

10 cursor c2 is select food,flavor,count(\*)

11 from item\_list join products on item=pid

12 where rno=ono

13 group by food,flavor;

14

15 BEGIN

16

17 ono:='&rid';

18 select fname,lname into cfname,clname

19 from customers natural join receipts

20 where rno=ono;

21 dbms\_output.put\_line('Customer name: '||cfname||' '||clname);

22

23 dbms\_output.put\_line('Ordered Following Items:');

24 dbms\_output.put\_line('Flavor Food Qty');

25 dbms\_output.put\_line('--------------------------');

26 open c2;

27 fetch c2 into cfood,cflavor,qty;

28 while c2%found loop

29 dbms\_output.put\_line(cflavor||' '||cfood||' '||qty);

30 fetch c2 into cfood,cflavor,qty;

31 end loop;

32 dbms\_output.put\_line('--------------------------');

33

34 dbms\_output.put\_line('Total Qty: '||c2%rowcount);

35 close c2;

36 END;

37 /

Enter value for rid: 51991

old 17: ono:='&rid';

new 17: ono:='51991';

Customer name: SOPKO RAYFORD

Ordered Following Items:

Flavor Food Qty

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

Apple Pie 1

Apple Tart 1

Truffle Cake 1

Chocolate Tart 1

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

Total Qty: 4

PL/SQL procedure successfully completed.

SQL> spool off