```
SQL> @D:/Ex05_commands.sql
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> set serveroutput on;
SQL>
SQL> declare
  2  fl products.flavor%type;
  3  fo products.food%type;
  4
  5  begin
  6             fl := '&flavor';
  7             fo := '&food';
  8             update products p set p.price = p.price+0
  9             where p.food = fo and p.flavor = fl;
 10             if SQL%rowcount>0 then
 11                       dbms_output.put_line(sql%rowcount||' products found of given combination');
 12                       return;
 13             end if;
 14             update products p set p.price = p.price+0
 15             where p.flavor = fl;
 16          if SQL%rowcount>0 then
 17                       dbms_output.put_line('only flavor is found');
 18                       return;
 19                       end if;
 20             update products p set p.price = p.price+0
 21             where p.food = fo;
 22             if SQL%rowcount>0   then
 23                       dbms_output.put_line('only food is found');
 24                       return;
 25                       end if;
 26             dbms_output.put_line('neither food nor flavor is found');
 27
 28  end;
 29  /
Enter value for flavor: Chocolate
old   6:  fl := '&flavor';
new   6:             fl := 'Chocolate';
Enter value for food: Icecream
old   7:  fo := '&food';
new   7:             fo := 'Icecream';
only flavor is 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  date_search receipts.r_date%type;
  3  begin
  4
```

```
  5  date_search := '&date_search';
  6  update item_list i
  7  set i.ordinal = i.ordinal+0
  8  where i.rno in (select r.rno from receipts r
  9                        where r.r_date = date_search);
 10  if SQL%found and SQL%rowcount >0 then
 11  dbms_output.put_line('No. of items sold on '||date_search||' is/are:'||SQL%rowcount);
 12  else
 13  dbms_output.put_line('No items sold');
 14  end if;
 15  end;
 16  /
Enter value for date_search: 3-oct-2007
old   5: date_search := '&date_search';
new   5: date_search := '3-oct-2007';
No. of items sold on 03-OCT-07 is/are:23

PL/SQL procedure successfully completed.

SQL>

SQL>
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>
SQL> declare
  2  inputprice products.price%type;
  3  CURSOR c1 is select * from products
  4  where abs(price-inputprice) in (select min(abs(p.price-inputprice))
  5                                  from products p
  6                                                        ) ;
  7  counts integer;
  8  ex_product products%rowtype;
  9  begin
 10  inputprice := &inputprice;
 11  open c1;
 12  select count(*) into counts
 13  from (select * from products
 14  where abs(price - inputprice) in (select min(abs(p.price-inputprice))
 15                                  from products p));
 16  for count in 1..counts loop
 17                  fetch c1 into ex_product.pid,ex_product.flavor,ex_product.food,ex_product.price;
 18                  dbms_output.put_line(ex_product.pid||'  '||ex_product.flavor||'  '||
ex_product.food||' '||ex_product.price );
 19        end loop;
 20  end;
 21  /
```

```
Enter value for inputprice: 0.8
old  10: inputprice := &inputprice;
new  10: inputprice := 0.8;
70-LEM  Lemon               Cookie             .79
70-W  Walnut              Cookie             .79

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 as shown below:
SQL>
SQL>
SQL> declare
  2  cust_name1 customers.lname%type;
  3  cust_name2 customers.fname%type;
  4  qty integer;
  5  rec_sel receipts.rno%type;
  6  counts integer;
  7  food_sel products.food%type;
  8  flavor_sel products.flavor%type;
  9  qtys integer;
 10  cursor c1 is select food, flavor, count(*) as qty
 11  from products p join item_list i on i.item = p.pid
 12  where i.rno = rec_sel
 13  group by (p.food,p.flavor);
 14  cursor c2 is select fname,lname from customers c join receipts r on r.cid = c.cid
 15  where rno = rec_sel ;
 16
 17  begin
 18  rec_sel := &rec_sel;
 19  select count(count(*)) into counts from products p join item_list i on i.item = p.pid
 20  where i.rno = rec_sel
 21  group by (p.food,p.flavor);
 22  select sum(count(*)) into qty from products p join item_list i on i.item = p.pid
 23  where i.rno = rec_sel
 24  group by (p.food,p.flavor);
 25  open c1;
 26  open c2;
 27  fetch c2 into cust_name1,cust_name2;
 28  dbms_output.put_line('Customer name: '||cust_name1||' '||cust_name2);
 29  dbms_output.put_line('FOOD FLAVOR QUANTITY');
 30  dbms_output.put_line('----------------------------------------');
 31  for count in 1..counts loop
 32                  fetch c1 into food_sel,flavor_sel,qtys;
 33                  dbms_output.put_line(flavor_sel||' '||food_sel||' '||qtys);
 34         end loop;
 35
 36  dbms_output.put_line('----------------------------------------');
 37  dbms_output.put_line('Total Quantity='||qty);
 38  end;
```

```
 39  /
Enter value for rec_sel: 46598
old  18: rec_sel := &rec_sel;
new  18: rec_sel := 46598;
Customer name: RAYFORD              SOPKO
FOOD FLAVOR QUANTITY
-----------------------------------------
Raspberry           Cookie              2
Walnut              Cookie              1
-----------------------------------------
Total Quantity=3

PL/SQL procedure successfully completed.

SQL> spool off
```