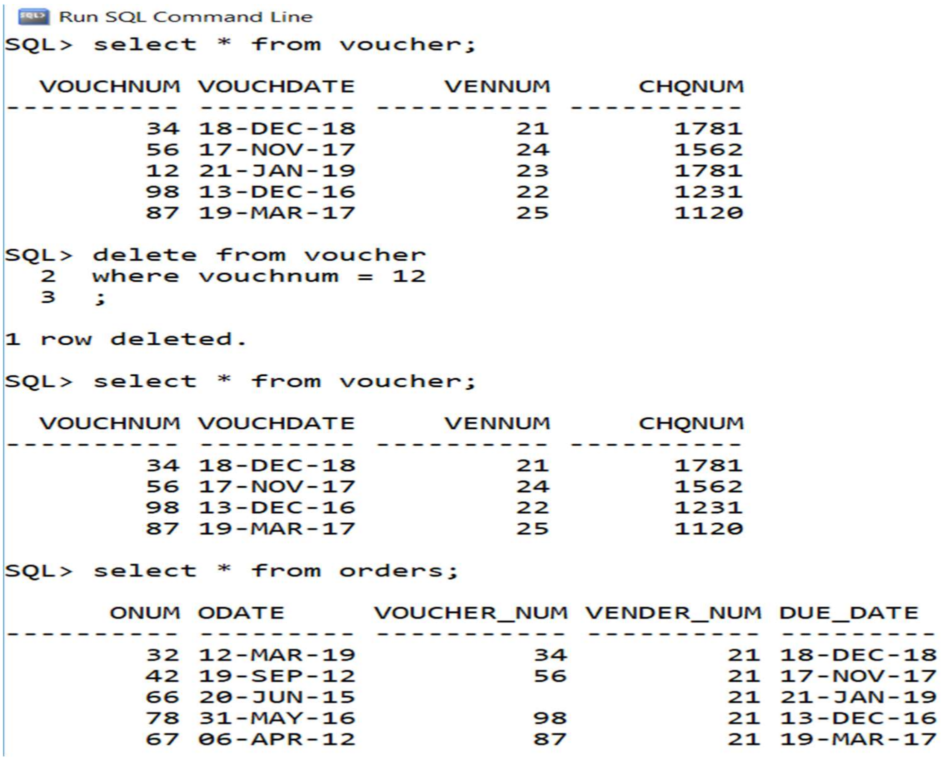
1. **Delete a voucher details from voucher table given by voucher no and make sure that, this operation automatically inserts null to all related tuples in a system.**

   Q. Delete voucher details of the voucher with voucher number 12.

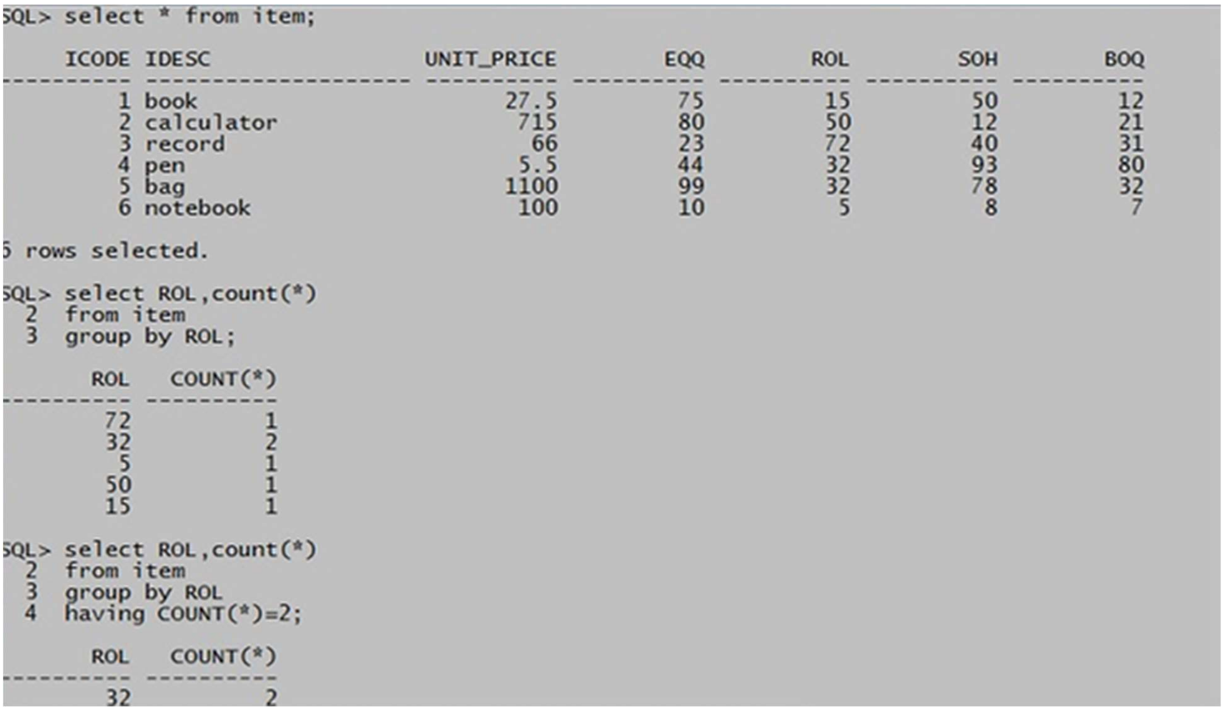


Run SQL Command Line

```
SQL> select * from voucher;

  VOUCHNUM VOUCHDATE      VENNUM     CHQNUM
---------- --------- ---------- ----------
        34 18-DEC-18         21       1781
        56 17-NOV-17         24       1562
        12 21-JAN-19         23       1781
        98 13-DEC-16         22       1231
        87 19-MAR-17         25       1120

SQL> delete from voucher
  2  where vouchnum = 12
  3  ;

1 row deleted.

SQL> select * from voucher;

  VOUCHNUM VOUCHDATE      VENNUM     CHQNUM
---------- --------- ---------- ----------
        34 18-DEC-18         21       1781
        56 17-NOV-17         24       1562
        98 13-DEC-16         22       1231
        87 19-MAR-17         25       1120

SQL> select * from orders;

      ONUM ODATE     VOUCHER_NUM VENDER_NUM DUE_DATE
---------- --------- ----------- ---------- ---------
        32 12-MAR-19          34         21 18-DEC-18
        42 19-SEP-12          56         21 17-NOV-17
        66 20-JUN-15                     21 21-JAN-19
        78 31-MAY-16          98         21 13-DEC-16
        67 06-APR-12          87         21 19-MAR-17
```

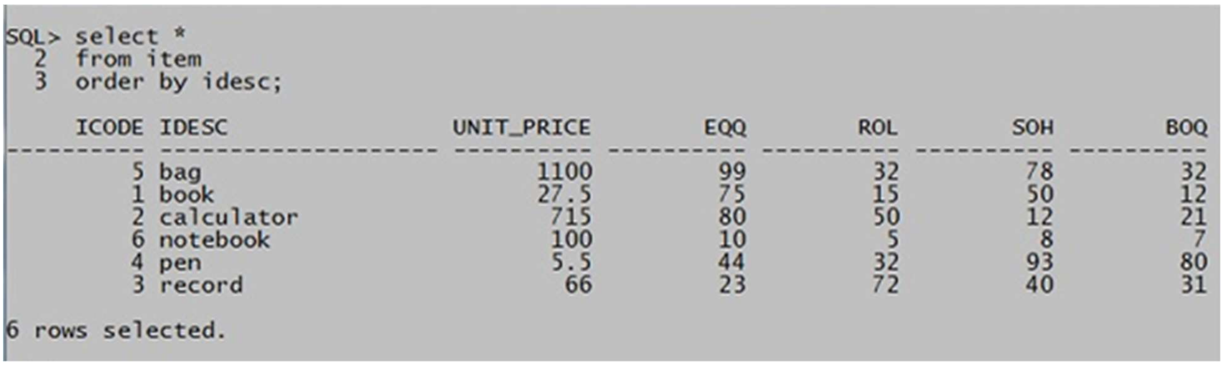
2. **Demonstrate with suitable example, group by, having, order by clauses.**

   Q1. For each item display the Reorder level and number of items for each reorder level.



```
SQL> select * from item;

     ICODE IDESC                UNIT_PRICE        EQQ        ROL        SOH        BOQ
---------- -------------------- ---------- ---------- ---------- ---------- ----------
         1 book                       27.5         75         15         50         12
         2 calculator                  715         80         50         12         21
         3 record                       66         23         72         40         31
         4 pen                         5.5         44         32         93         80
         5 bag                        1100         99         32         78         32
         6 notebook                    100         10          5          8          7

6 rows selected.

SQL> select ROL,count(*)
  2  from item
  3  group by ROL;

       ROL   COUNT(*)
---------- ----------
        72          1
        32          2
         5          1
        50          1
        15          1

SQL> select ROL,count(*)
  2  from item
  3  group by ROL
  4  having COUNT(*)=2;

       ROL   COUNT(*)
---------- ----------
        32          2
```

Q2. Retrieve the details of all items, ordered by their description.



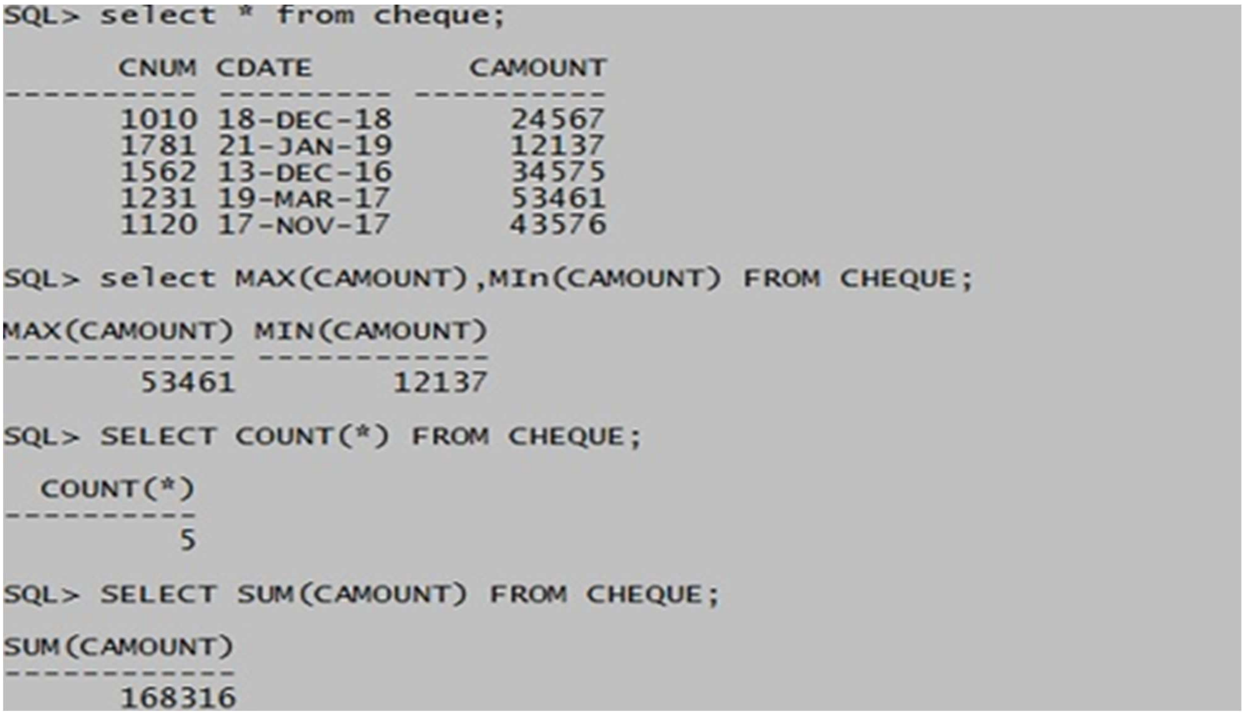
```
SQL> select *
  2  from item
  3  order by idesc;

     ICODE IDESC                UNIT_PRICE        EQQ        ROL        SOH        BOQ
---------- -------------------- ---------- ---------- ---------- ---------- ----------
         5 bag                        1100         99         32         78         32
         1 book                       27.5         75         15         50         12
         2 calculator                  715         80         50         12         21
         6 notebook                    100         10          5          8          7
         4 pen                         5.5         44         32         93         80
         3 record                       66         23         72         40         31

6 rows selected.
```

3. **Demonstrate all aggregation operations in SQL, with suitable examples.**

Q1. Retrieve Min and Max cheque amount.

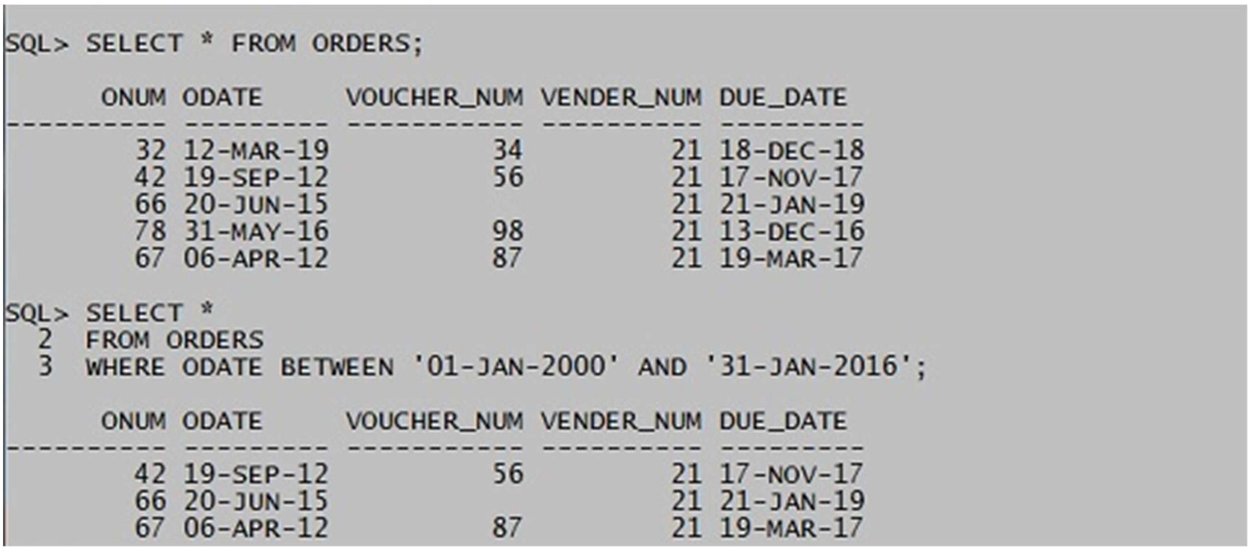
Q2. Retrieve number of payments done by cheque.

Q3. Find the sum of money paid through cheques.



```
SQL> select * from cheque;

      CNUM CDATE        CAMOUNT
---------- --------- ----------
      1010 18-DEC-18      24567
      1781 21-JAN-19      12137
      1562 13-DEC-16      34575
      1231 19-MAR-17      53461
      1120 17-NOV-17      43576

SQL> select MAX(CAMOUNT),MIn(CAMOUNT) FROM CHEQUE;

MAX(CAMOUNT) MIN(CAMOUNT)
------------ ------------
       53461        12137

SQL> SELECT COUNT(*) FROM CHEQUE;

  COUNT(*)
----------
         5

SQL> SELECT SUM(CAMOUNT) FROM CHEQUE;

SUM(CAMOUNT)
------------
      168316
```

4. **Produce the list of orders between Jan 2000 to Jan 2006.**

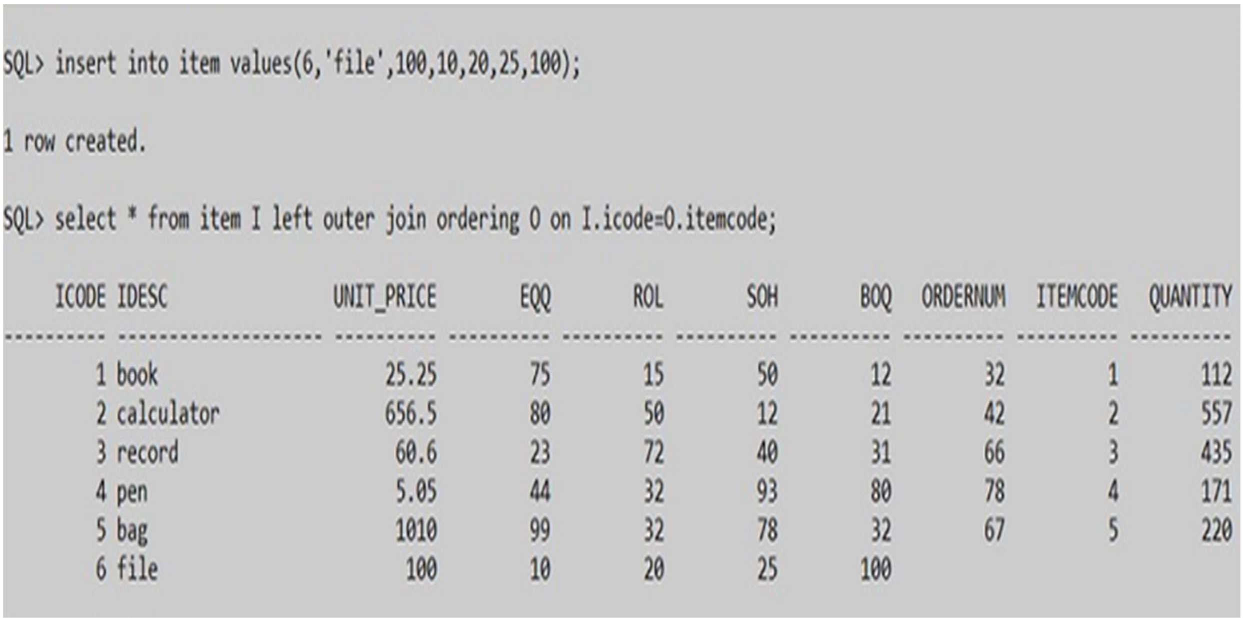


```
SQL> SELECT * FROM ORDERS;

      ONUM ODATE     VOUCHER_NUM VENDER_NUM DUE_DATE
---------- --------- ----------- ---------- ---------
        32 12-MAR-19          34         21 18-DEC-18
        42 19-SEP-12          56         21 17-NOV-17
        66 20-JUN-15                     21 21-JAN-19
        78 31-MAY-16          98         21 13-DEC-16
        67 06-APR-12          87         21 19-MAR-17

SQL> SELECT *
  2  FROM ORDERS
  3  WHERE ODATE BETWEEN '01-JAN-2000' AND '31-JAN-2016';

      ONUM ODATE     VOUCHER_NUM VENDER_NUM DUE_DATE
---------- --------- ----------- ---------- ---------
        42 19-SEP-12          56         21 17-NOV-17
        66 20-JUN-15                     21 21-JAN-19
        67 06-APR-12          87         21 19-MAR-17
```

5. **Demonstrate with suitable example, Left outer join, Right outer join and Full outer join.**

Left outer join

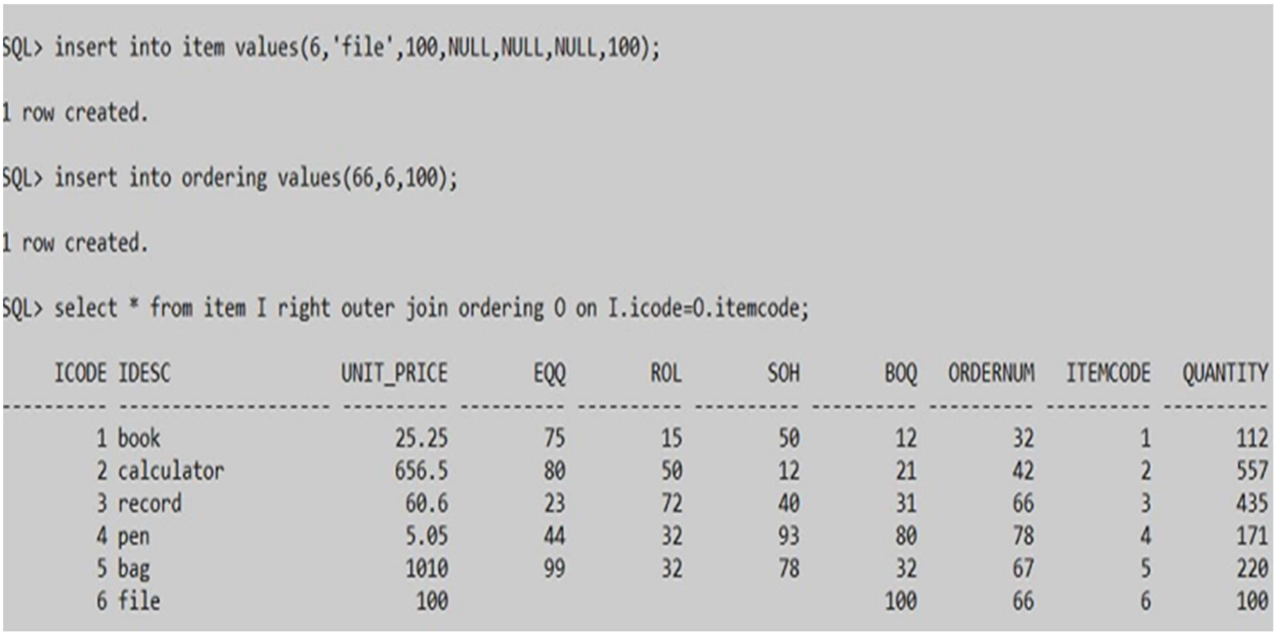
Q1. For all the items in the item table, show their corresponding order number and quantity.



```
SQL> insert into item values(6,'file',100,10,20,25,100);

1 row created.

SQL> select * from item I left outer join ordering O on I.icode=O.itemcode;

     ICODE IDESC              UNIT_PRICE        EQQ        ROL        SOH        BOQ   ORDERNUM   ITEMCODE   QUANTITY
---------- ------------------ ---------- ---------- ---------- ---------- ---------- ---------- ---------- ----------
         1 book                    25.25         75         15         50         12         32          1        112
         2 calculator              656.5         80         50         12         21         42          2        557
         3 record                   60.6         23         72         40         31         66          3        435
         4 pen                      5.05         44         32         93         80         78          4        171
         5 bag                      1010         99         32         78         32         67          5        220
         6 file                      100         10         20         25        100
```

Right outer join

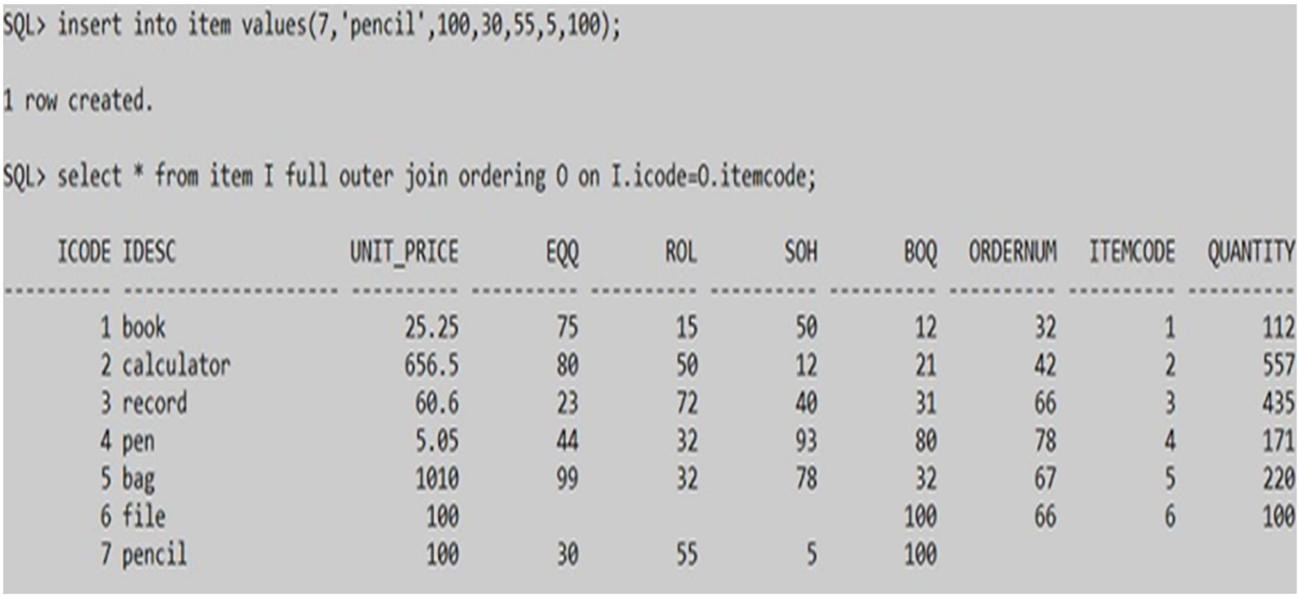
Q2. For all the tuples in ordering table, write their corresponding item details.



```
SQL> insert into item values(6,'file',100,NULL,NULL,NULL,100);

1 row created.

SQL> insert into ordering values(66,6,100);

1 row created.

SQL> select * from item I right outer join ordering O on I.icode=O.itemcode;

     ICODE IDESC              UNIT_PRICE        EQQ        ROL        SOH        BOQ   ORDERNUM   ITEMCODE   QUANTITY
---------- ------------------ ---------- ---------- ---------- ---------- ---------- ---------- ---------- ----------
         1 book                    25.25         75         15         50         12         32          1        112
         2 calculator              656.5         80         50         12         21         42          2        557
         3 record                   60.6         23         72         40         31         66          3        435
         4 pen                      5.05         44         32         93         80         78          4        171
         5 bag                      1010         99         32         78         32         67          5        220
         6 file                      100                                         100         66          6        100
```

Full outer join

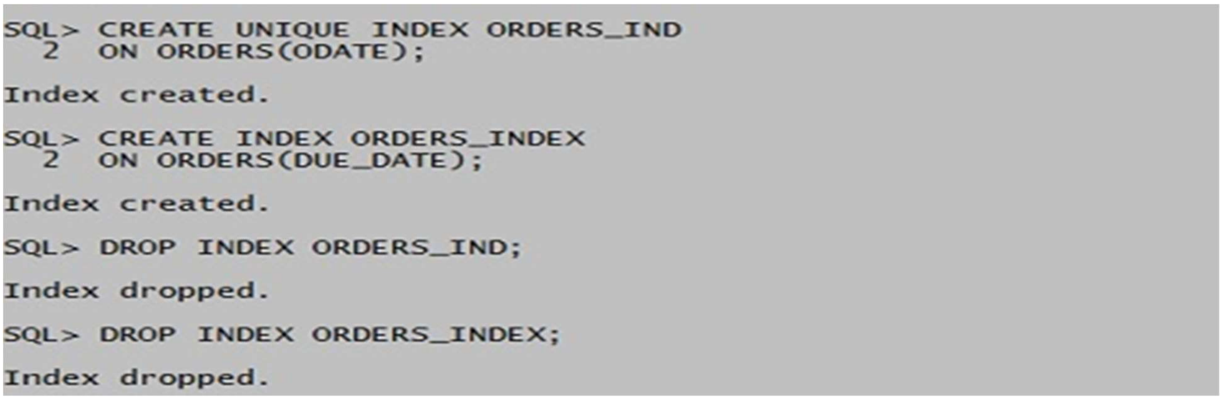
Q3. For all the items in item table and all the orders in ordering table write their corresponding item and order detail respectively.



```
SQL> insert into item values(7,'pencil',100,30,55,5,100);

1 row created.

SQL> select * from item I full outer join ordering O on I.icode=O.itemcode;

     ICODE IDESC              UNIT_PRICE        EQQ        ROL        SOH        BOQ   ORDERNUM   ITEMCODE   QUANTITY
---------- ------------------ ---------- ---------- ---------- ---------- ---------- ---------- ---------- ----------
         1 book                    25.25         75         15         50         12         32          1        112
         2 calculator              656.5         80         50         12         21         42          2        557
         3 record                   60.6         23         72         40         31         66          3        435
         4 pen                      5.05         44         32         93         80         78          4        171
         5 bag                      1010         99         32         78         32         67          5        220
         6 file                      100                                         100         66          6        100
         7 pencil                    100         30         55          5        100
```

6. **Demonstrate Create Index and Drop index on any table.**

Q1. Create unique index ORDERS_IND on order date in orders table.
Q2. Create index ORDERS_INDEX on due date in orders table.
Q3. Drop index ORDERS_IND.
Q4. Drop index ORDERS_INDEX.

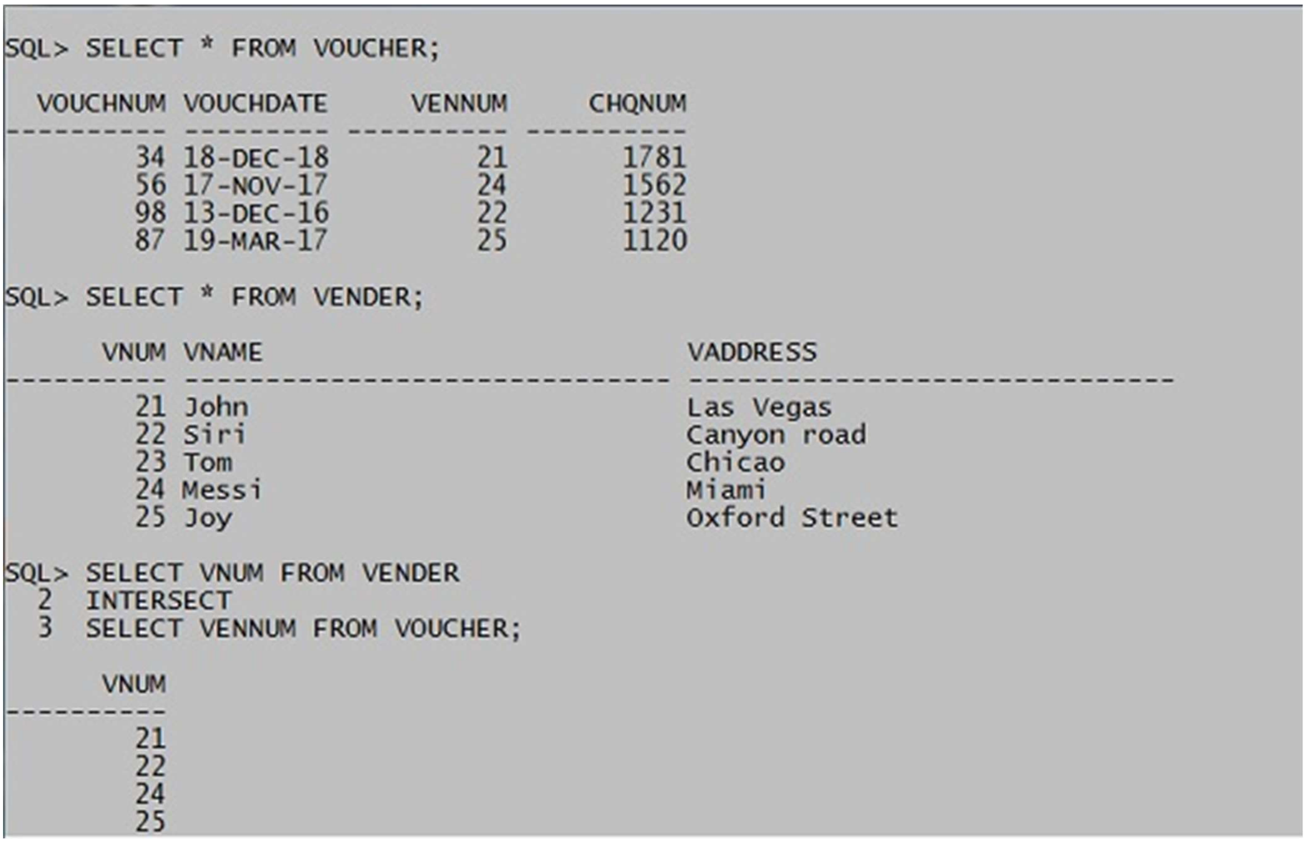


```
SQL> CREATE UNIQUE INDEX ORDERS_IND
  2  ON ORDERS(ODATE);

Index created.

SQL> CREATE INDEX ORDERS_INDEX
  2  ON ORDERS(DUE_DATE);

Index created.

SQL> DROP INDEX ORDERS_IND;

Index dropped.

SQL> DROP INDEX ORDERS_INDEX;

Index dropped.
```

7. **Demonstrate with suitable example, Union, Intersection, and Except operations.**

Intersection

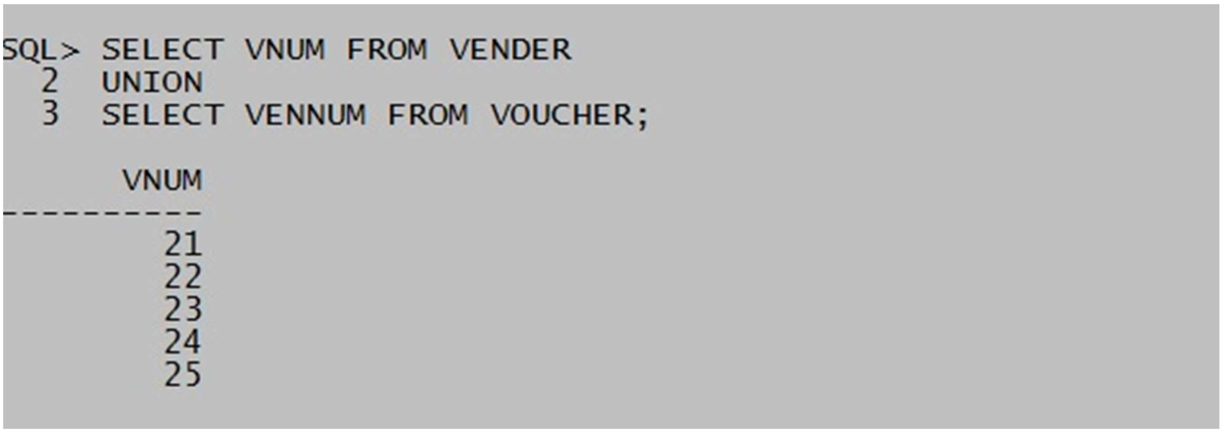
Q1. Retrieve the vender numbers which are in vendor table and also in voucher table.



```
SQL> SELECT * FROM VOUCHER;

  VOUCHNUM VOUCHDATE     VENNUM     CHQNUM
---------- --------- ---------- ----------
        34 18-DEC-18         21       1781
        56 17-NOV-17         24       1562
        98 13-DEC-16         22       1231
        87 19-MAR-17         25       1120

SQL> SELECT * FROM VENDER;

      VNUM VNAME                          VADDRESS
---------- ------------------------------ ------------------------------
        21 John                           Las Vegas
        22 Siri                           Canyon road
        23 Tom                            Chicao
        24 Messi                          Miami
        25 Joy                            Oxford Street

SQL> SELECT VNUM FROM VENDER
  2  INTERSECT
  3  SELECT VENNUM FROM VOUCHER;

      VNUM
----------
        21
        22
        24
        25
```

Union

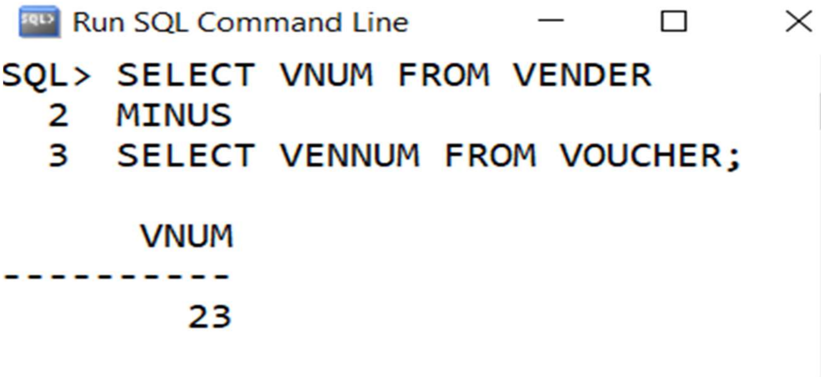
Q2. Retrieve the vender numbers which are either in vendor table or in voucher table.



```
SQL> SELECT VNUM FROM VENDER
  2  UNION
  3  SELECT VENNUM FROM VOUCHER;

      VNUM
----------
        21
        22
        23
        24
        25
```

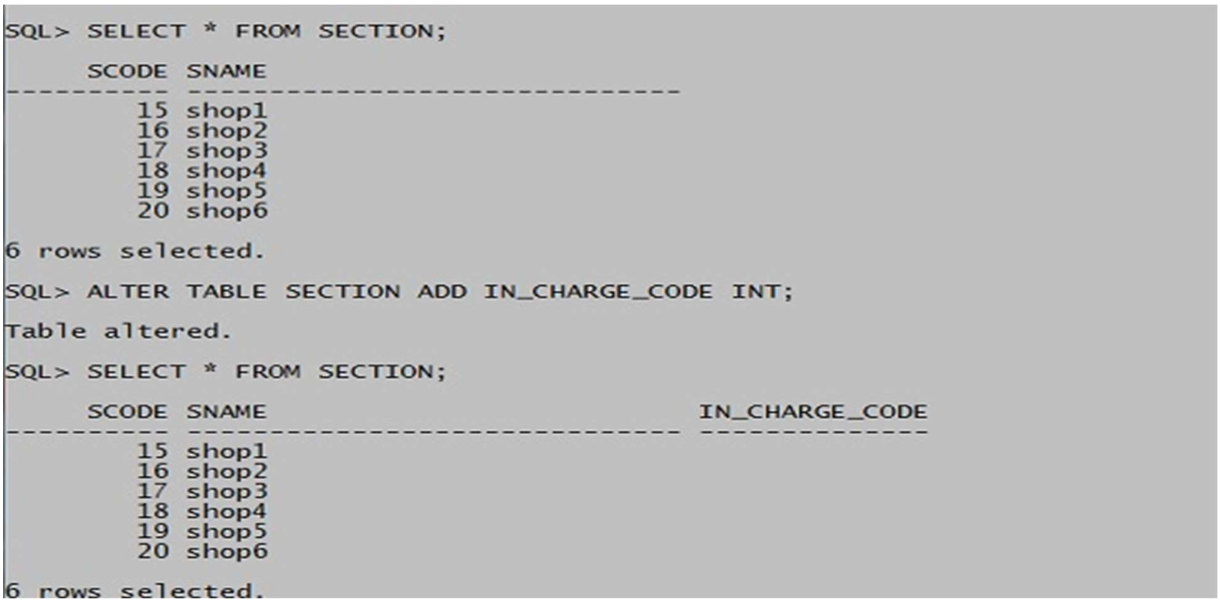
Except

Q3. Retrieve the vender numbers which are in vendor table but not in voucher table.



```
SQL> SELECT VNUM FROM VENDER
  2  MINUS
  3  SELECT VENNUM FROM VOUCHER;

      VNUM
----------
        23
```

8. **Alter the table SECTION by adding section In-charge-Code.**



```
SQL> SELECT * FROM SECTION;

     SCODE SNAME
---------- ------------------------------
        15 shop1
        16 shop2
        17 shop3
        18 shop4
        19 shop5
        20 shop6

6 rows selected.

SQL> ALTER TABLE SECTION ADD IN_CHARGE_CODE INT;

Table altered.

SQL> SELECT * FROM SECTION;

     SCODE SNAME                          IN_CHARGE_CODE
---------- ------------------------------ --------------
        15 shop1
        16 shop2
        17 shop3
        18 shop4
        19 shop5
        20 shop6

6 rows selected.
```