Assignment – 13
Using the UNION clause.

1) Create a union of two queries that shows the names, cities, and ratings of all customers. Those with rating of 200 or greater will also have the words “High Rating”, while the others will have the words “Low Rating”.

```
mysql> SELECT Cname, City , 'High Rating'AS rating_category
    -> FROM customers
    -> WHERE rating>=200
    -> UNION
    -> SELECT Cname, City , 'Low Rating'AS rating_category
    -> FROM customers
    -> WHERE rating<200;
+----------+----------+-----------------+
| Cname    | City     | rating_category |
+----------+----------+-----------------+
| Giovanni | Rome     | High Rating     |
| Liu      | San Jose | High Rating     |
| Grass    | Berlin   | High Rating     |
| Cisneros | San Jose | High Rating     |
| Hoffman  | London   | Low Rating      |
| Clemens  | London   | Low Rating      |
| Pereira  | Rome     | Low Rating      |
+----------+----------+-----------------+
7 rows in set (0.00 sec)
```

2) Write a command that produces the name and number of each salesperson and each customer with more than one current order. Put the results in alphabetical order.

```
mysql> SELECT c.cname AS name, c.cnum AS number
    -> FROM orders o
    -> JOIN customers c ON o.cnum=c.cnum
    -> GROUP BY c.cname, o.cnum
    -> HAVING COUNT(o.onum)>1
    -> UNION
    -> SELECT s.sname AS name, s.snum AS number
    -> FROM orders o
    -> JOIN salespeople s ON o.snum=s.snum
    -> GROUP BY s.sname, s.snum
    -> HAVING COUNT(o.onum)>1
    -> ORDER BY name;
+----------+--------+
| name     | number |
+----------+--------+
| Cisneros |   2008 |
| Clemens  |   2006 |
| Grass    |   2004 |
| Peel     |   1001 |
| Rifkin   |   1007 |
| Serres   |   1002 |
+----------+--------+
6 rows in set (0.01 sec)
```

3) Form a union of three queries. Have the first select the snums of all salespeople in San Jose; the second, the cnums of all customers in San Jose; and the third the onums of all orders on October 3. Retain duplicates between the last two queries but eliminate any redundancies between either of them and the first.
(Note: in the sample tables as given, there would be no such redundancy. This is besides the point.)

```
mysql> (SELECT snum AS Number
    -> FROM salespeople
    -> WHERE city= 'San Jose')
    -> UNION
    -> (SELECT cnum AS Number
    -> FROM customers
    -> WHERE city = 'San Jose'
    -> AND cnum NOT IN (SELECT snum FROM salespeople WHERE city= 'San Jose')
    -> )
    -> UNION ALL
    -> (SELECT onum AS Number
    -> FROM orders
    -> WHERE odate= '1990-10-03'
    -> AND onum NOT IN (SELECT snum FROM salespeople WHERE city='San Jose')
    -> );
+--------+
| Number |
+--------+
|   1002 |
|   2003 |
|   2008 |
|   3001 |
|   3003 |
|   3002 |
|   3005 |
|   3006 |
+--------+
8 rows in set (0.00 sec)
```