

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”.

```
D2_92814_Krushna>select cname , city , 'high rating' as rating from customers where rating >=200 union select cname , city , 'low rating' as rating from customers where rating <200 ;
```

cname	city	rating
Giovanni	Rome	high rating
Liu	San Jose	high rating
Grass	Berlin	high rating
Hoffman	London	low rating
Clemens	London	low rating
Pereira	Rome	low rating

6 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.

```
D2_92814_Krushna>SELECT sname AS name, snum AS number  
-> FROM Salespeople NATURAL JOIN Orders  
-> GROUP BY sname, snum  
-> HAVING COUNT(*) > 1  
-> UNION  
-> SELECT cname AS name, cnum AS number  
-> FROM Customers NATURAL JOIN Orders  
-> GROUP BY cname, cnum  
-> HAVING COUNT(*) > 1  
-> ORDER BY name;
```

name	number
Axelrod	1003
Giovanni	2002
Rifkin	1007
Serres	1002

4 rows in set (0.10 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.

```
D2_92814_Krushna>SELECT snum
-> FROM Salespeople
-> WHERE city = 'San Jose'
->
-> UNION
->
-> SELECT cnum
-> FROM Customers
-> WHERE city = 'San Jose'
->
-> UNION ALL
->
-> SELECT onum
-> FROM Orders
-> WHERE odate = '1990-10-03';
```

snum
1002
2003
3001
3003
3003
3005
3006

```
7 rows in set (0.00 sec)
```