

# SQL 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”.

**Ans-:** select Cname, City, Rating, concat('High',' ',Rating) Comment from CUSTOMERS  
where Rating >= 200  
union  
select Cname, City, Rating, concat('Low',' ',Rating) Comment from CUSTOMERS  
where Rating < 200;

**OR**

select Cname, City, 'High Rating' Rating from CUSTOMERS  
where Rating >= 200  
union  
select Cname, City, 'Low Rating' Rating from CUSTOMERS  
where Rating < 200;

```
amit@amit-Lenovo-ideapad-510-15IKB: ~  
D5_Amit_84097>select Cname, City, 'High Rating' Rating from CUSTOMERS  
-> where Rating >= 200  
-> union  
-> select Cname, City, 'Low Rating' Rating from CUSTOMERS  
-> where Rating < 200;  
+-----+-----+-----+  
| Cname   | City    | Rating    |  
+-----+-----+-----+  
| 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)  
  
D5_Amit_84097>
```

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.

Ans-: select ORDERS.Snum, Sname from SALESPEOPLE, ORDERS  
where SALESPEOPLE.Snum = ORDERS.Snum  
group by ORDERS.Snum, Sname  
having count(ORDERS.Snum) > 1  
union  
select ORDERS.Cnum, Cname from CUSTOMERS, ORDERS  
where CUSTOMERS.Cnum = ORDERS.Cnum  
group by ORDERS.Cnum, Cname  
having count(ORDERS.Cnum) > 1  
order by Sname;

```
amit@amit-Lenovo-ideapad-510-15IKB: ~  
D5_Amit_84097>select ORDERS.Snum, Sname from SALESPEOPLE, ORDERS  
-> where SALESPEOPLE.Snum = ORDERS.Snum  
-> group by ORDERS.Snum, Sname  
-> having count(ORDERS.Snum) > 1  
-> union  
-> select ORDERS.Cnum, Cname from CUSTOMERS, ORDERS  
-> where CUSTOMERS.Cnum = ORDERS.Cnum  
-> group by ORDERS.Cnum, Cname  
-> having count(ORDERS.Cnum) > 1  
-> order by Sname;  
+-----+-----+  
| Snum | Sname |  
+-----+-----+  
| 2008 | Cisneros |  
| 2006 | Clemens |  
| 2004 | Grass |  
| 1001 | Peel |  
| 1007 | Rifkin |  
| 1002 | Serres |  
+-----+-----+  
6 rows in set (0.00 sec)  
  
D5_Amit_84097>
```

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.)

**Ans-:** 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';

```
D5_Amit_84097>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 |
| 2008 |
| 3003 |
| 3002 |
| 3005 |
| 3001 |
| 3006 |
+-----+
```

8 rows in set (0.00 sec)

```
D5_Amit_84097>
```