Name: Aarya Sanjay Dange

Assignment – 14

Entering, Deleting, and Changing Field Values.

1) Write a command that puts the following values, in their given order, into the salespeople table: city – San Jose, name – Blanco, comm – NULL, snum – 1100.

```
mysql> insert into salespeople (city,sname,comm,snum) values('San Jose','Blanco',null,1100);
```

2) Write a command that removes all orders from customer Clemens from the Orders table.

```
mysql> delete from orders where cnum=(select cnum from customers where cname='Clemens');
```

3) Write a command that increases rating of all customers in Rome by 100.

```
mysql> update customers set rating=rating+100 where city='Rome';
```

4) Salesperson Serres has left the company. Assign her customers to Motika.

```
mysql> UPDATE customers
    -> SET snum = 1004
    -> WHERE snum = 1002;
```

Assignment – 15

Using Subqueries with DML Commands.

1) Assume there is a table called Multicust, with all of the same column definitions as Salespeople. Write a command that inserts all salespeople with more than one customer into this table.

```
mysql> insert into Multicust (snum, sname, city, comm)
    -> select s.snum, s.sname, s.city, s.comm
    -> from salespeople s join customers c on
    -> s.snum=c.snum group by c.cnum
    -> having count(c.cnum)>1;
```

2) Write a command that deletes all customers with no current orders.

```
mysql> delete from customers where cnum not in(select cnum from
orders);
```

3) Write a command that increases by twenty percent the commissions of all salespeople with total orders above Rs. 3,000.

```
mysql> update salespeople set comm=comm+0.20 where snum=(select
snum from orders where amt>3000);
```

Assignment – 16

Creating Tables and Indexes

1) Write a command that will enable a user to pull orders grouped by date out of the Orders table quickly.

```
mysql> create index idx1 on orders(odate);
mysql> select odate,count(*) from orders group by odate;
```

2) If the Orders table has already been created, how can you force the onum field to be unique (assume all current values are unique)?

```
mysql> create unique index idx2 on orders(onum);
```

3) Create an index that would permit each salesperson to retrieve his or her orders grouped by date quickly.

```
mysql> select odate,group_concat(onum),snum from orders group by snum,odate;
```

4) Let us assume that each salesperson is to have only one customer of a given rating, and that this is currently the case. Enter a command that enforces it.

```
mysql> ALTER TABLE customers
    -> ADD CONSTRAINT unique_snum_rating
    -> UNIQUE (Snum, Rating);
```