

GAIN PRACTICE ON SQL USING THE TABLES AND QUESTIONS GIVEN BELOW.

Salespeople

SNUM	SNAME	CITY	COMM
1001	Peel	London	.12
1002	Serres	San Jose	.13
1004	Motika	London	.11
1007	Rifkin	Barcelona	.15
1003	AxelRod	New York	.10
1005	Fran	London	.26

Customers

CNUM	CNAME	CITY	RATING	SNUM
2001	Hoffman	London	100	1001
2002	Giovanni	Rome	200	1003
2003	Liu	San Jose	200	1002
2004	Grass	Berlin	300	1002
2006	Clemens	London	100	1001
2008	Cisneros	San Jose	300	1007
2007	Pereira	Rome	100	1004

Orders

ONUM	AMT	ODATE	CNUM	SNUM
3001	18.69	10/03/96	2008	1007
3003	767.19	10/03/96	2001	1001
3002	1900.10	10/03/96	2007	1004
3005	5160.45	10/03/96	2003	1002
3006	1098.16	10/03/96	2008	1007
3009	1713.23	10/04/96	2002	1003
3007	75.75	10/04/96	2002	1003
3008	4723.00	10/05/96	2006	1001
3010	1309.95	10/06/96	2004	1002
3010	9891.88	10/06/96	2006	1001

Queries

1. List information of all the salespeople.
Select * from salespeople;
2. List all Customers with a rating of 100.
Select * from customers where rating=100;

3. Find all customers with city is NULL, if city is null then display it as “not mentioned”.

```
Select cnum,cname,city,ifnull(city,' not mentioned')
From customers;
```

4. Find which salespeople with atleast 2 orders.

```
Select snum,sname
From salespeople s
Where snum in (select snum
                From Orders
                Group by snum
                Having count(*)>=2);
```

5. List names of all customers and salespeople serving them.

```
Select cnum,cname,snum,sname
From customer c inner join salespeople s
on c.snum=s.snum;
```

6. Find the names and numbers of all salespeople who had more than 2 customer.

```
Select snum,sname
From salespeople s
Where snum in (select snum
                From customers
                Group by snum
                Having count(*)>=2);
```

7. List all customer who live in same city of salespeople

```
Select cnum,cname,city
From customers c
Where city (select city from salespeople s where s.snum=c.snum);
```

8. Find the largest order taken by each salesperson.

```
Select snum,max(amt)
From orders
Group by snum;
```

9. Find all customers with orders on October 3.

Select cnum,cname

From customers

Where cnum in (select cnum

From orders

Where month(odate)=10 and day(odate)=3;

10. Find all pairs of customers having the same rating.

Select c.cnum,c.cname,c1.cnum,c1.cname

From customers c, customers c1

Where c.rating=c1.rating and c.cname!=c1.cname and c.cnum < c1.cnum;

11. Find all customers located in cities where Serres(SNUM 1002) has customers.

Select cnum,cname

From customers

Where city in (select city

From customers

Where snum =(select snum

From salespeople

Where sname='Serres'));

12. Find all orders with amounts smaller than any amount for a customer in San Jose.

Select *

From orders

Where amt < any (select amt

From orders

Where cnum in (

Select cnum

From customers

Where city='san jose')) ;

13. Write a query that calculates the amount of the salespeople's commission on each order by a customer with a rating above 100.00.

```
Select snum,sname,(ifnull(s.comm,0)*o.amt) comm
From salespeople s inner join orders o on s.snum=o.snum
Where cnum in (select cnum
                From customers c
                Where rating>100)
```

14. select all salespeople(by name and number) who have customers in their cities who they do not service

```
select *
from salesperson
where cnum in (select cnum
                from customers c
                where c.city=s.city and s.snum!=c.snum);
```

15. Write a query that will find all salespeople who have no customers located in their city.

```
Select snum,sname
From salespeople s
Where not exists (select * from customers c where c.city !=s.city);
```

16. Write a query that selects all customers whose names begin with 'C' or 'G' or 'L'. (note write by using both like and REGEP)

```
Select *
From customers
Where cname like 'C%' or cname like 'G%' or cname like 'L%';
```

Or

```
Select *
From customers
Where cname REGEXP '^[CGL];
```

17. Write a query on the Customers table will find the highest rating in each city. Put the output in this form: for the city(city) the highest rating is : (rating).

```
Select *
From (
Select cnum,cname, rating, dense_rank() over (partition by city
order by rating desc) drk) e
Where drk=1;
```

18. Which is the date, order number, amt and city for each salespeople(by name) for the maximum order he has obtained?

```
Select odate,onum, city,sname,snum
From salespeople s inner join Customers c on s.snum=c.snum inner
join orders o on s.snum=o.snum
where o.amt =(
        select max(amt)
        from orders p
        where p.snum=o.snum);
```

Q2. Write a procedure to display all customer names, who has brought product from Serres, and rating > 200

```
Delimiter //
Create procedure displanames()
Begin
    Select * from customers
    Where snum=(select snum
                From saleman
                Where sname='serres') and rating >200;
End//
```

Q3. Write a procedure (use Cursors) which will return designation for salesman based on commission percent

If it is in range 0.9 to 0.13 then junior

If it is 0.13 to 0.17 then assistant otherwise Senior

```
delimiter //
```

```
create procedure displaysalespeople()
```

```
begin
```

```

declare vsnum,vstop int default 0;

declare vsname,vcity,vdesg varchar(20);

declare vcomm double(10,2);

declare scurr cursor for select snum,sname,city,comm from salespeople;

declare continue handler for NOT FOUND set vstop=1;

open scurr;

label1: loop

            fetch scurr into vsnum,vsname,vcity,vcomm;

            if vstop=1 then

                leave label1;

            end if;

if vcomm >=0.09 and vcomm < 0.13 then

    set vdesg='junior';

elseif vcomm <0.17 then

    set vdesg='assistant';

else

    set vdesg='senior';

end if

select vsnum,vsname,vcity,vcomm,vdesg;

end loop;

close scurr;

end//

delimiter ;

```

Q4. Write a insert and update before trigger for table customer

Insert data in customor_monitor

To store data (custid, oldcustname, new custname, oldrating , new rating, user() ,date, action)

Create table customor_monitor(

 Custid int,

 Oldcname varchar(20),

 Newname varchar(20),

 Oldrating int,

 Newrating int,

 Uname varchar(20),

 Chk_dt date,

 Action varchar(20)

)

Create trigger insert_cust before insert

On customer

For each row

Begin

 Insert into customor_monitor

values(new.cnum,null,new.cname,null,new.rating,user(),now(),'insert
');

End//

Create trigger update_cust before update

On customer

For each row

Begin

 Insert into customor_monitor

values(new.cnum,old.cname,new.cname,old.rating,new.rating,user(),n
ow(),'insert');

End//

Q5. Write a function to generatecode, to generate code for every customer

Which is combination of 3 letters/digit of (cname and city, cid)

Create function generatecode(cnm varchar(20), city varchar(20),cid varchar(20))

begin

 return concat(substr(cname,1,3),substr(city,1,3),substr(cid,1,3));
end//