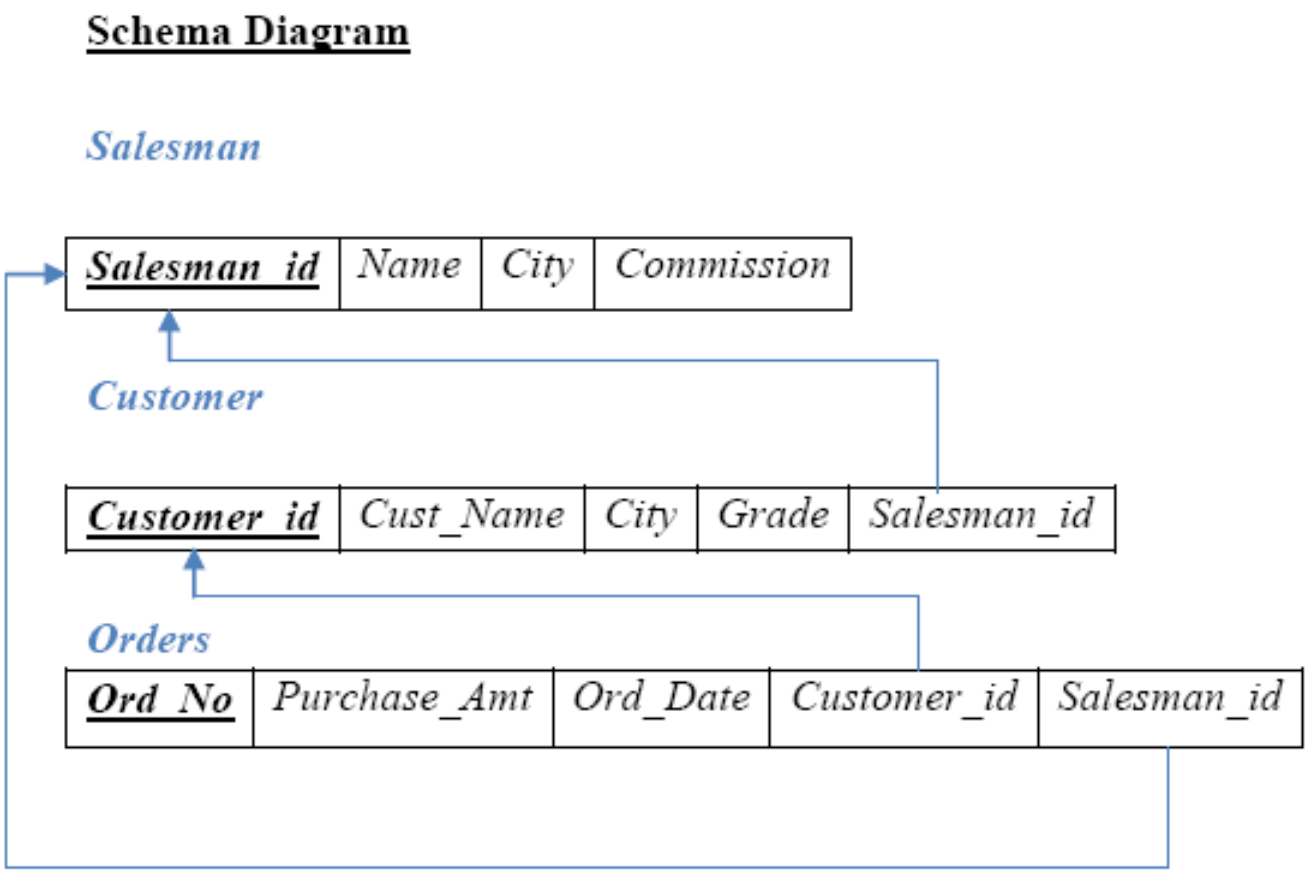
Program 6 : Order Database

Consider the following schema for Order Database:

SALESMAN (*Salesman_id, Name, City, Commission*)
CUSTOMER (*Customer_id, Cust_Name, City, Grade, Salesman_id*)
ORDERS (*Ord_No, Purchase_Amt, Ord_Date, Customer_id, Salesman_id*)




```
create database Lab6;
use Lab6;

create table salesman (
        salesman_id int,
        name varchar (20),
        city varchar (20),
        commission varchar (20),
```

```
        primary key (salesman_id));
desc salesman;
```

	Field	Type	Null	Key	Default	Extra
▶	salesman_id	int	NO	PRI	NULL	
	name	varchar(20)	YES		NULL	
	city	varchar(20)	YES		NULL	
	commission	varchar(20)	YES		NULL	

```
create table customer (
        customer_id int,
        cust_name varchar (20),
        city varchar (20),
        grade int ,
   salesman_id int,
        primary key (customer_id),
        foreign key (salesman_id) references salesman(salesman_id) on delete set
null);
desc customer;
```

	Field	Type	Null	Key	Default	Extra
▶	customer_id	int	NO	PRI	NULL	
	cust_name	varchar(20)	YES		NULL	
	city	varchar(20)	YES		NULL	
	grade	int	YES		NULL	
	salesman_id	int	YES	MUL	NULL	

```
create table orders (
        ord_no int ,
        purchase_amt int,
        ord_date date,
   customer_id int,
        salesman_id int,
        primary key (ord_no),
        foreign key (customer_id) references customer (customer_id) on delete
cascade,
        foreign key (salesman_id) references salesman (salesman_id) on delete
cascade);
desc orders;
```

Field	Type	Null	Key	Default	Extra
ord_no	int	NO	PRI	NULL	
purchase_amt	int	YES		NULL	
ord_date	date	YES		NULL	
customer_id	int	YES	MUL	NULL	
salesman_id	int	YES	MUL	NULL	

```
insert into salesman values (1000, 'john','bangalore','25 %');
insert into salesman values (2000, 'ravi','bangalore','20 %');
insert into salesman values (3000, 'kumar','mysore','15 %');
insert into salesman values (4000, 'smith','delhi','30 %');
insert into salesman values (5000, 'harsha','hydrabad','15 %');
select * from salesman;
```

salesman_id	name	city	commission
1000	john	bangalore	25 %
2000	ravi	bangalore	20 %
3000	kumar	mysore	15 %
4000	smith	delhi	30 %
5000	harsha	hydrabad	15 %
NULL	NULL	NULL	NULL

```
insert into customer values (10, 'preethi','bangalore', 100, 1000);
insert into customer values (11,'vivek','mangalore', 300, 1000);
insert into customer values (12, 'bhaskar','chennai', 400, 2000);
insert into customer values (13, 'chethan','bangalore', 200, 2000);
insert into customer values (14, 'mamatha','bangalore', 400, 3000);
select * from customer;
```

customer_id	cust_name	city	grade	salesman_id
10	preethi	bangalore	100	1000
11	vivek	mangalore	300	1000
12	bhaskar	chennai	400	2000
13	chethan	bangalore	200	2000
14	mamatha	bangalore	400	3000
NULL	NULL	NULL	NULL	NULL

```
insert into orders values (50, 5000, '04-06-17', 10, 1000);
insert into orders values (51, 450, '20-01-17', 10, 2000);
insert into orders values (52, 1000, '24-02-17', 13, 2000);
insert into orders values (53, 3500, '13-04-17', 14, 3000);
insert into orders values (54, 550, '09-03-17', 12, 2000);
```

select * from orders;

ord_no	purchase_amt	ord_date	customer_id	salesman_id
50	5000	2004-06-17	10	1000
51	450	2020-01-17	10	2000
52	1000	2024-02-17	13	2000
53	3500	2013-04-17	14	3000
54	550	2009-03-17	12	2000
NULL	NULL	NULL	NULL	NULL

1. Count the customers with grades above Bangalore's average.

```
SELECT grade, count(DISTINCT customer_id)
FROM customer
GROUP BY grade
HAVING grade > (SELECT AVG(grade)
FROM customer
WHERE city='bangalore');
```

grade	count(DISTINCT customer_id)
300	1
400	2

2. Find the name and numbers of all salesmen who had more than one customer.

```
SELECT salesman_id, NAME
FROM salesman a
WHERE 1 < (SELECT count(*)
FROM customer
WHERE salesman_id=a.salesman_id);
```

salesman_id	NAME
1000	john
2000	ravi
NULL	NULL

3. List all salesmen and indicate those who have and don't have customers in their cities (Use UNION operation.)

```
SELECT salesman.salesman_id, NAME, cust_name, commission
FROM salesman, customer
WHERE salesman.city = customer.city
UNION
SELECT salesman_id, name, 'no customer', commission
FROM salesman
```

```
WHERE NOT city = ANY
(SELECT city
FROM customer)
ORDER BY 2 DESC;
```

salesman_id	NAME	cust_name	commission
4000	smith	no customer	30 %
2000	ravi	preethi	20 %
2000	ravi	chethan	20 %
2000	ravi	mamatha	20 %
3000	kumar	no customer	15 %
1000	john	preethi	25 %
1000	john	chethan	25 %
1000	john	mamatha	25 %
5000	harsha	no customer	15 %

4. Create a view that finds the salesman who has the customer with the highest order of a day.

```
CREATE VIEW highsalesman AS
SELECT b.ord_date, a.salesman_id, a.name
FROM salesman a, orders b
WHERE a.salesman_id = b.salesman_id
AND b.purchase_amt=(SELECT max(purchase_amt)
FROM orders c
WHERE c.ord_date = b.ord_date);
SELECT * FROM highsalesman;
```

ord_date	salesman_id	name
2004-06-17	1000	john
2020-01-17	2000	ravi
2024-02-17	2000	ravi
2013-04-17	3000	kumar
2009-03-17	2000	ravi

5. Demonstrate the DELETE operation by removing salesman with id 1000. All his orders must also be deleted.

```
DELETE FROM salesman
WHERE salesman_id=1000;
SELECT * FROM salesman;
SELECT * FROM orders;
```

salesman_id	name	city	commission
2000	ravi	bangalore	20 %
3000	kumar	mysore	15 %
4000	smith	delhi	30 %
5000	harsha	hydrabad	15 %
NULL	NULL	NULL	NULL

	ord_no	purchase_amt	ord_date	customer_id	salesman_id
▶	51	450	2020-01-17	10	2000
	52	1000	2024-02-17	13	2000
	53	3500	2013-04-17	14	3000
	54	550	2009-03-17	12	2000
*	NULL	NULL	NULL	NULL	NULL