**LAB 6: Order Database**

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;



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;



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;



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;



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;



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;



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');



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);



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;



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;



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;



