### **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 ORDER;

CREATE TABLE SALESMAN
(SALESMAN\_ID int,
NAME VARCHAR (20),
CITY VARCHAR (20),
COMMISSION VARCHAR (20),
PRIMARY KEY (SALESMAN\_ID));

Your SQL query has been executed s	uccessfully.				
DESC SALESMAN					
[Edit inline][Edit][Create PHP code	1				
+ Options	,				
Field	Туре	Null	Key	Default	Extra
SALESMAN_ID	int(11)	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 (3),
SALESMAN\_ID int,
PRIMARY KEY (CUSTOMER\_ID),
FOREIGN KEY (SALESMAN\_ID) REFERENCES SALESMAN (SALESMAN\_ID) ON DELETE SET NULL);

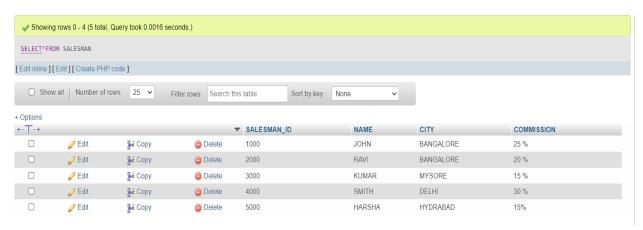
#### Your SQL query has been executed successfully. DESC CUSTOMER [ Edit inline ] [ Edit ] [ Create PHP code ] + Options Field Null Key Default Extra Туре CUSTOMER\_ID int(11) NO PRI NULL CUST\_NAME varchar(20) YES NULL CITY varchar(20) YES NULL GRADE int(3) YES NULL SALESMAN\_ID int(11) YES MUL NULL

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);

Your SQL query has been executed successfully.					
DESC ORDERS					
[ Edit inline ] [ Edit ] [ Create PHP code ]					
[Luit IIIIIIe][Luit][Oleate FFII Code]					
+ Options					
Field	Туре	Null	Key	Default	Extra
ORD_NO	int(11)	NO	PRI	NULL	
PURCHASE_AMT	int(11)	YES		NULL	
				1001	
ORD_DATE	date	YES		NULL	
ORD_DATE CUSTOMER_ID	date int(11)	YES	MUL	NULL NULL	
_			MUL MUL		

INSERT INTO SALESMAN VALUES (1000, 'JOHN', 'BANGALORE', '25 %'), (2000, 'RAVI', 'BANGALORE', '20 %'), (3000, 'KUMAR', 'MYSORE', '15 %'), (4000, 'SMITH', 'DELHI', '30 %'), (5000, 'HARSHA', 'HYDRABAD', '15%');

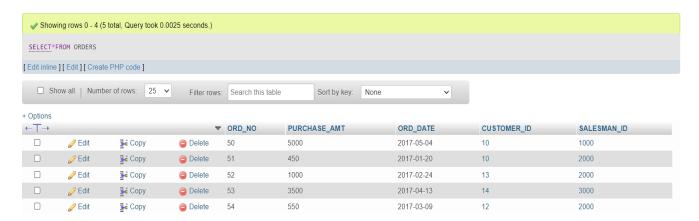


INSERT INTO CUSTOMER VALUES (10, 'PREETHI', 'BANGALORE', 100, 1000),

- (11, 'VIVEK', 'MANGALORE', 300, 1000),
- (12, 'BHASKAR', 'CHENNAI', 400, 2000),
- (13, 'CHETHAN', 'BANGALORE', 200, 2000),
- (14, 'MAMATHA', 'BANGALORE', 400, 3000);



INSERT INTO ORDERS VALUES (50, 5000, '2017-04-04', 10, 1000), (51, 450, '20-JAN-17', 10, 2000), (52,1000,'24-FEB-17',13,2000), (53,3500,'13-APR-17',14,3000), (54, 550, '09-MAR-17', 12, 2000);



#### Write SQL queries to

### 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 S

WHERE (SELECT COUNT (\*)

FROM CUSTOMER C

WHERE C. SALESMAN\_ID=S.SALESMAN\_ID) > 1;



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

SELECT S.SALESMAN ID, S.NAME, C.CUST NAME, S.COMMISSION

FROM SALESMAN S, CUSTOMER C

WHERE S.CITY=C.CITY

UNION

SELECT S.SALESMAN\_ID,S.NAME,'NO MATCH',S.COMMISSION

FROM SALESMAN S

WHERE CITY NOT IN

(SELECT CITY

FROM CUSTOMER)

ORDER BY 1 ASC;



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

CREATE VIEW V\_SALESMAN AS
SELECT O.ORDER\_DATE, S.SALESMAN\_ID, S.NAME
FROM SALESMAN S,ORDERS O
WHERE S.SALESMAN\_ID = O.SALESMAN\_ID

AND O.PURCHASE\_AMOUNT= (SELECT MAX(PURCHASE\_AMOUNT)

FROM ORDERS C

WHERE C.ORDER\_DATE=O.ORDER\_DATE);



## 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;

	✓ 1 row affected. (Query took 0.0207 seconds.)
I	DELETE FROM SALESMAN WHERE SALESMAN_ID=1000
ı	[Edit inline] [Edit] [Create PHP code]