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)

Write SQL queries to

1. Count the customers with grades above Bangalore's average.
2. Find the name and numbers of all salesmen who had more than one customer.
3. List all salesmen and indicate those who have and don't have customers in their cities (Use UNION operation.)
4. Create a view that finds the salesman who has the customer with the highest order of a day.
5. Demonstrate the DELETE operation by removing salesman with id 1000. All his orders must also be deleted.

Create the above tables by properly specifying the primary keys and the foreign keys using alter table option.

```
CREATE TABLE SALESMAN
(SALESMAN_ID INTEGER (4),
NAME VARCHAR (20),
CITY VARCHAR (20),
COMMISSION VARCHAR (20),
PRIMARY KEY (SALESMAN_ID));

CREATE TABLE CUSTOMER1
(CUSTOMER_ID INTEGER (4),
CUST_NAME VARCHAR (20),
CITY VARCHAR (20),
GRADE INTEGER (3),
PRIMARY KEY (CUSTOMER_ID),
SALESMAN_ID INTEGER (4) REFERENCES SALESMAN (SALESMAN_ID) ON DELETE SET NULL);

CREATE TABLE ORDERS
(ORD_NO INTEGER (5),
PURCHASE_AMT INTEGER (10, 2),
ORD_DATE DATE,
PRIMARY KEY (ORD_NO),
CUSTOMER_ID INTEGER(4) REFERENCES CUSTOMER1(CUSTOMER_ID), SALESMAN_ID INTEGER (4) REFERENCES SALESMAN (SALESMAN_ID) ON DELETE CASCADE);
```

Insertion of Values to Tables

```
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 %');

INSERT INTO CUSTOMER1 VALUES (10, 'PREETHI','BANGALORE', 100, 1000);
INSERT INTO CUSTOMER1 VALUES (11, 'VIVEK','MANGALORE', 300, 1000);
INSERT INTO CUSTOMER1 VALUES (12, 'BHASKAR','CHENNAI', 400, 2000);
INSERT INTO CUSTOMER1 VALUES (13, 'CHETHAN','BANGALORE', 200, 2000);
INSERT INTO CUSTOMER1 VALUES (14, 'MAMATHA','BANGALORE', 400, 3000);

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

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

```
SELECT GRADE, COUNT (DISTINCT CUSTOMER_ID)
FROM CUSTOMER1
GROUP BY GRADE
HAVING GRADE > (SELECT AVG(GRADE)
FROM CUSTOMER1
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 CUSTOMER1
 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, CUSTOMER1
WHERE SALESMAN.CITY = CUSTOMER1.CITY
UNION
SELECT SALESMAN_ID, NAME, 'NO MATCH', COMMISSION
FROM SALESMAN
WHERE NOT CITY = ANY
 (SELECT CITY
  FROM CUSTOMER1)
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 ELITSALESMAN 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);
```

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

Use ON DELETE CASCADE at the end of foreign key definitions while creating child table orders and then execute the following:

Use ON DELETE SET NULL at the end of foreign key definitions while creating child table

```
DELETE FROM SALESMAN
WHERE SALESMAN_ID=1000;
```

6. Update the purchase amount to 9000 for given order.

```
UPDATE ORDERS
SET PURCHASE_AMT=9000
WHERE ORDER_NO='&ORDER_NO';
```