# Lab Practice

**CREATE THE TABLES GIVEN BELOW. INSERT TUPLES SUCH THAT ALL QUERIES SHOULD GIVE PROPER OUTPUT.**

**1. Create the following tables:**

**a. Customer Master Table:** **C**ust

| Attribute Name | Data Type | Constraints |
|---|---|---|
| custid | varchar(3) | Primary Key |
| Lname | Varchar(15) | |
| Fname | Varchar(15) | |
| area | Varchar(2) | |
| phoneno | numeric(8,0) | |

**b. Movies Master Table**: **Movie**

| Attribute Name | Data Type | Constraints |
|---|---|---|
| Mvno | Numeric(2,0) | Primary Key |
| title | Varchar(25) | |
| type | Varchar(10) | |
| star | Varchar(25) | |
| price | Numeric(8,2) | |

**c. Invoice transaction Table : Invoice**

| Attribute Name | Data Type | Constraints |
|---|---|---|
| Invno | Varchar(3) | Primary Key |
| Mvno | Number(2) | Foreign key movie(mvno) |
| custid | Varchar(3) | Foreign key cust(custid) |
| issueDate | Date | |
| retDate | Date | |

**2. Insert the following data into the tables created**

**Table:Cust**

| custid | Lname | Fname | Area | PhoneNo |
|---|---|---|---|---|
| A01 | Bayross | Ivan | sa | 6125467 |
| A02 | Saitwal | Vandana | mu | 5560379 |
| A03 | Jaguste | Pramada | da | 4563891 |
| A04 | Navindgi | Basu | ba | 6125401 |
| A05 | Sreedhar | Ravi | va | NULL |
| A06 | NULL | Rukmini | gh | 5125274 |

**Table:Movie**

| mvno | title | type | star | price |
|---|---|---|---|---|
| 1 | Bloody Vengeance | action | Jackie Chan | 100.00 |
| 2 | The Firm | thriller | Tom cruise | 200.00 |
| 3 | Pretty Woman | romance | Richard Gere | 150.00 |
| 4 | Home Alone | Comedy | Macaulay Culkin | 150.55 |
| 5 | The Fugitive | Thriller | Harrison Ford | 200.00 |
| 6 | Coma | Suspense | Michael Douglas | 100.00 |
| 7 | Dracula | Horror | Gary Oldman | 150.25 |

| | | | | |
|---|---|---|---|---|
| 8 | Quick change | Comedy | Bill Murray | 100.00 |
| 9 | Gone with the wind | Drama | Clarke Gable | 200.00 |
| 10 | Carry on Doctor | Comedy | Leslie Phillips | 100.00 |

**Table:Invoice**

| invno | mvno | custid | issueDate | retDate |
|---|---|---|---|---|
| i01 | 4 | A01 | 23-jul-2012 | 25-jul-2012 |
| i02 | 3 | A02 | 12-aug-2012 | 15-aug-2012 |
| i03 | 1 | A02 | 15-aug-2012 | 18-aug-2012 |
| i04 | 6 | A03 | 10-sep-2012 | 13-sep-2012 |
| i05 | 7 | A04 | 05-aug-2012 | 08-aug-2012 |
| i06 | 2 | A06 | 18-sep-2012 | 20-sep-2012 |
| i07 | 9 | A05 | 07-jul-2012 | 10-jul-2012 |
| i08 | 9 | A01 | 11-aug-2012 | 14-aug-2012 |
| i09 | 5 | A03 | 06-jul-2012 | 09-jul-2012 |
| i10 | 8 | A06 | 03-sep-2012 | 06-sep-2012 |

```
CREATE TABLE Cust (
    custid VARCHAR(3) PRIMARY KEY,
    Lname VARCHAR(15),
    Fname VARCHAR(15),
    area VARCHAR(2),
    phoneno NUMERIC(8,0)
);

INSERT INTO Cust VALUES
('A01', 'Bayross', 'Ivan', 'sa', 6125467),
('A02', 'Saitwal', 'Vandana', 'mu', 5560379),
('A03', 'Jaguste', 'Pramada', 'da', 4563891),
('A04', 'Navindgi', 'Basu', 'ba', 6125401),
('A05', 'Sreedhar', 'Ravi', 'va', NULL),
('A06', NULL, 'Rukmini', 'gh', 5125274);
```

```
CREATE TABLE Movie (
    mvno NUMERIC(2,0) PRIMARY KEY,
    title VARCHAR(25),
    type VARCHAR(10),
    star VARCHAR(25),
    price NUMERIC(8,2)
);

INSERT INTO Movie VALUES
(1, 'Bloody Vengeance', 'action', 'Jackie Chan', 100.00),
(2, 'The Firm', 'thriller', 'Tom cruise', 200.00),
(3, 'Pretty Woman', 'romance', 'Richard Gere', 150.00),
(4, 'Home Alone', 'Comedy', 'Macaulay Culkin', 150.55),
(5, 'The Fugitive', 'Thriller', 'Harrison Ford', 200.00),
(6, 'Coma', 'Suspense', 'Michael Douglas', 100.00),
(7, 'Dracula', 'Horror', 'Gary Oldman', 150.25),
(8, 'Quick change', 'Comedy', 'Bill Murray', 100.00),
(9, 'Gone with the wind', 'Drama', 'Clarke Gable', 200.00),
(10, 'Carry on Doctor', 'Comedy', 'Leslie Phillips', 100.00);
```

```
CREATE TABLE Invoice (
    invno VARCHAR(3) PRIMARY KEY,
    mvno NUMERIC(2,0),
    custid VARCHAR(3),
    issueDate DATE,
    retDate DATE,
    FOREIGN KEY (mvno) REFERENCES Movie(mvno),
    FOREIGN KEY (custid) REFERENCES Cust(custid)
);

INSERT INTO Invoice VALUES
('i01', 4, 'A01', TO_DATE('23-jul-2012','DD-MON-YYYY'), TO_DATE('25-jul-2012','DD-MON-YYYY')),
('i02', 3, 'A02', TO_DATE('12-aug-2012','DD-MON-YYYY'), TO_DATE('15-aug-2012','DD-MON-YYYY')),
('i03', 1, 'A02', TO_DATE('15-aug-2012','DD-MON-YYYY'), TO_DATE('18-aug-2012','DD-MON-YYYY')),
('i04', 6, 'A03', TO_DATE('10-sep-2012','DD-MON-YYYY'), TO_DATE('13-sep-2012','DD-MON-YYYY')),
('i05', 7, 'A04', TO_DATE('05-aug-2012','DD-MON-YYYY'), TO_DATE('08-aug-2012','DD-MON-YYYY')),
('i06', 2, 'A06', TO_DATE('18-sep-2012','DD-MON-YYYY'), TO_DATE('20-sep-2012','DD-MON-YYYY')),
('i07', 9, 'A05', TO_DATE('07-jul-2012','DD-MON-YYYY'), TO_DATE('10-jul-2012','DD-MON-YYYY')),
('i08', 9, 'A01', TO_DATE('11-aug-2012','DD-MON-YYYY'), TO_DATE('14-aug-2012','DD-MON-YYYY')),
('i09', 5, 'A03', TO_DATE('06-jul-2012','DD-MON-YYYY'), TO_DATE('09-jul-2012','DD-MON-YYYY')),
('110', 8, 'A06', TO_DATE('03-sep-2012','DD-MON-YYYY'), TO_DATE('06-sep-2012','DD-MON-YYYY'));
```

**3. Write Query statements for the following**

1. Find out the names of all customers.

```
SELECT Fname, Lname FROM Cust;
```

| | fname character varying (15) | lname character varying (15) |
|---|---|---|
| 1 | Ivan | Bayross |
| 2 | Vandana | Saitwal |
| 3 | Pramada | Jaguste |
| 4 | Basu | Navindgi |
| 5 | Ravi | Sreedhar |
| 6 | Rukmini | [null] |

2. Print the entire customer table.

```
SELECT * FROM Cust;
```

| | custid [PK] character varying (3) | lname character varying (15) | fname character varying (15) | area character varying (2) | phoneno numeric (8) |
|---|---|---|---|---|---|
| 1 | A01 | Bayross | Ivan | sa | 6125467 |
| 2 | A02 | Saitwal | Vandana | mu | 5560379 |
| 3 | A03 | Jaguste | Pramada | da | 4563891 |
| 4 | A04 | Navindgi | Basu | ba | 6125401 |
| 5 | A05 | Sreedhar | Ravi | va | [null] |
| 6 | A06 | [null] | Rukmini | gh | 5125274 |

3. Retrieve the list of fname and the area of all the customers.

```
SELECT Fname, area FROM Cust;
```

| | fname character varying (15) | area character varying (2) |
|---|---|---|
| 1 | Ivan | sa |
| 2 | Vandana | mu |
| 3 | Pramada | da |
| 4 | Basu | ba |
| 5 | Ravi | va |
| 6 | Rukmini | gh |

4. List the various movie types available from the movie title.

```
SELECT DISTINCT type FROM Movie;
```

| | type character varying (10) |
|---|---|
| 1 | Thriller |
| 2 | action |
| 3 | Comedy |
| 4 | thriller |
| 5 | Drama |
| 6 | Horror |
| 7 | Suspense |
| 8 | romance |

5. Find the names of all customers having 'a' in the second letter in their fname

```
SELECT Fname FROM Cust WHERE SUBSTR(Fname,2,1) = 'a';
```

| | fname character varying (15) |
|---|---|
| 1 | Vandana |
| 2 | Basu |
| 3 | Ravi |

6. Find the Lname of all customers that begin with ‘S’ or ‘J’

```
SELECT Lname FROM Cust WHERE Lname LIKE 'S%' OR Lname LIKE 'J%';
```

| | lname character varying (15) |
|---|---|
| 1 | Saitwal |
| 2 | Jaguste |
| 3 | Sreedhar |

7. Find the Lname of all customers that is between ‘B’ and ‘S’

```
SELECT Lname FROM Cust WHERE Lname BETWEEN 'B' AND 'S';
```

| | lname character varying (15) |
|---|---|
| 1 | Bayross |
| 2 | Jaguste |
| 3 | Navindgi |

8. Find out the customers who stay in an area whose second letter is ‘a'.

```
SELECT * FROM Cust WHERE SUBSTR(area,2,1) = 'a';
```

| | custid [PK] character varying (3) | lname character varying (15) | fname character varying (15) | area character varying (2) | phoneno numeric (8) |
|---|---|---|---|---|---|
| 1 | A01 | Bayross | Ivan | sa | 6125467 |
| 2 | A03 | Jaguste | Pramada | da | 4563891 |
| 3 | A04 | Navindgi | Basu | ba | 6125401 |
| 4 | A05 | Sreedhar | Ravi | va | [null] |

9. Find the list of all customers who stay in area 'ba’ or area ‘mu' or area 'gh’.

```
SELECT * FROM Cust WHERE area IN ('ba', 'mu', 'gh');
```

| | custid [PK] character varying (3) | lname character varying (15) | fname character varying (15) | area character varying (2) | phoneno numeric (8) |
|---|---|---|---|---|---|
| 1 | A02 | Saitwal | Vandana | mu | 5560379 |
| 2 | A04 | Navindgi | Basu | ba | 6125401 |
| 3 | A06 | [null] | Rukmini | gh | 5125274 |

10. Print the list of customers whose phone numbers are greater than the value 5550000.

```
SELECT * FROM Cust WHERE phoneno > 5550000;
```

| | custid [PK] character varying (3) | lname character varying (15) | fname character varying (15) | area character varying (2) | phoneno numeric (8) |
|---|---|---|---|---|---|
| 1 | A01 | Bayross | Ivan | sa | 6125467 |
| 2 | A02 | Saitwal | Vandana | mu | 5560379 |
| 3 | A04 | Navindgi | Basu | ba | 6125401 |

11. Print all the information from invoice table of customers who have been issued movies in the month of September.

```
SELECT * FROM Invoice WHERE TO_CHAR(issueDate, 'MON') = 'SEP';
```

| | invno [PK] character varying (3) | mvno numeric (2) | custid character varying (3) | issuedate date | retdate date |
|---|---|---|---|---|---|
| 1 | i04 | 6 | A03 | 2012-09-10 | 2012-09-13 |
| 2 | i06 | 2 | A06 | 2012-09-18 | 2012-09-20 |
| 3 | 110 | 8 | A06 | 2012-09-03 | 2012-09-06 |

12. Display the invoice table information for custid 'A01' and 'A02'.

```
SELECT * FROM Invoice WHERE custid IN ('A01', 'A02');
```

| | invno [PK] character varying (3) | mvno numeric (2) | custid character varying (3) | issuedate date | retdate date |
|---|---|---|---|---|---|
| 1 | i01 | 4 | A01 | 2012-07-23 | 2012-07-25 |
| 2 | i02 | 3 | A02 | 2012-08-12 | 2012-08-15 |
| 3 | i03 | 1 | A02 | 2012-08-15 | 2012-08-18 |
| 4 | i08 | 9 | A01 | 2012-08-11 | 2012-08-14 |

13. Find the movies of type "action" and "Comedy"

```
SELECT * FROM Movie WHERE LOWER(type) IN ('action', 'comedy');
```

| | mvno [PK] numeric (2) | title character varying (25) | type character varying (10) | star character varying (25) | price numeric (8,2) |
|---|---|---|---|---|---|
| 1 | 1 | Bloody Vengeance | action | Jackie Chan | 100.00 |
| 2 | 4 | Home Alone | Comedy | Macaulay Culkin | 150.55 |
| 3 | 8 | Quick change | Comedy | Bill Murray | 100.00 |
| 4 | 10 | Carry on Doctor | Comedy | Leslie Phillips | 100.00 |

14. Find the movies whose price is greater than 150 and less than or equal to 200.

```
SELECT * FROM Movie WHERE price > 150 AND price <= 200;
```

| | mvno [PK] numeric (2) | title character varying (25) | type character varying (10) | star character varying (25) | price numeric (8,2) |
|---|---|---|---|---|---|
| 1 | 2 | The Firm | thriller | Tom cruise | 200.00 |
| 2 | 4 | Home Alone | Comedy | Macaulay Culkin | 150.55 |
| 3 | 5 | The Fugitive | Thriller | Harrison Ford | 200.00 |
| 4 | 7 | Dracula | Horror | Gary Oldman | 150.25 |
| 5 | 9 | Gone with the wind | Drama | Clarke Gable | 200.00 |

15. Find the movies that cost more than 159 and also find the new cost as original cost* 15

```
SELECT *, price*15 AS new_cost FROM Movie WHERE price > 159;
```

| | mvno [PK] numeric (2) | title character varying (25) | type character varying (10) | star character varying (25) | price numeric (8,2) | new_cost numeric |
|---|---|---|---|---|---|---|
| 1 | 2 | The Firm | thriller | Tom cruise | 200.00 | 3000.00 |
| 2 | 5 | The Fugitive | Thriller | Harrison Ford | 200.00 | 3000.00 |
| 3 | 9 | Gone with the wind | Drama | Clarke Gable | 200.00 | 3000.00 |

16. Rename the new column in the above query as new price,

```
SELECT *, price*15 AS "new price" FROM Movie WHERE price > 159;
```

| | mvno [PK] numeric (2) | title character varying (25) | type character varying (10) | star character varying (25) | price numeric (8,2) | new price numeric |
|---|---|---|---|---|---|---|
| 1 | 2 | The Firm | thriller | Tom cruise | 200.00 | 3000.00 |
| 2 | 5 | The Fugitive | Thriller | Harrison Ford | 200.00 | 3000.00 |
| 3 | 9 | Gone with the wind | Drama | Clarke Gable | 200.00 | 3000.00 |

17. List the movies in sorted order of their titles.

```
SELECT * FROM Movie ORDER BY title;
```

| | mvno [PK] numeric (2) | title character varying (25) | type character varying (10) | star character varying (25) | price numeric (8,2) |
|---|---|---|---|---|---|
| 1 | 1 | Bloody Vengeance | action | Jackie Chan | 100.00 |
| 2 | 10 | Carry on Doctor | Comedy | Leslie Phillips | 100.00 |
| 3 | 6 | Coma | Suspense | Michael Douglas | 100.00 |
| 4 | 7 | Dracula | Horror | Gary Oldman | 150.25 |
| 5 | 9 | Gone with the wind | Drama | Clarke Gable | 200.00 |
| 6 | 4 | Home Alone | Comedy | Macaulay Culkin | 150.55 |
| 7 | 3 | Pretty Woman | romance | Richard Gere | 150.00 |
| 8 | 8 | Quick change | Comedy | Bill Murray | 100.00 |
| 9 | 2 | The Firm | thriller | Tom cruise | 200.00 |
| 10 | 5 | The Fugitive | Thriller | Harrison Ford | 200.00 |

18. List the movies in the sorted order of type,title

```
SELECT * FROM Movie ORDER BY type, title;
```

| | mvno [PK] numeric (2) | title character varying (25) | type character varying (10) | star character varying (25) | price numeric (8,2) |
|---|---|---|---|---|---|
| 1 | 1 | Bloody Vengeance | action | Jackie Chan | 100.00 |
| 2 | 10 | Carry on Doctor | Comedy | Leslie Phillips | 100.00 |
| 3 | 4 | Home Alone | Comedy | Macaulay Culkin | 150.55 |
| 4 | 8 | Quick change | Comedy | Bill Murray | 100.00 |
| 5 | 9 | Gone with the wind | Drama | Clarke Gable | 200.00 |
| 6 | 7 | Dracula | Horror | Gary Oldman | 150.25 |
| 7 | 3 | Pretty Woman | romance | Richard Gere | 150.00 |
| 8 | 6 | Coma | Suspense | Michael Douglas | 100.00 |
| 9 | 2 | The Firm | thriller | Tom cruise | 200.00 |
| 10 | 5 | The Fugitive | Thriller | Harrison Ford | 200.00 |

19. Print the names and types of all the movie except horror movies.

```
SELECT title, type FROM Movie WHERE LOWER(type) <> 'horror';
```

| | title character varying (25) | type character varying (10) |
|---|---|---|
| 1 | Bloody Vengeance | action |
| 2 | The Firm | thriller |
| 3 | Pretty Woman | romance |
| 4 | Home Alone | Comedy |
| 5 | The Fugitive | Thriller |
| 6 | Coma | Suspense |
| 7 | Quick change | Comedy |
| 8 | Gone with the wind | Drama |
| 9 | Carry on Doctor | Comedy |

20. Calculate the square root of price of each movie.

```
SELECT title, SQRT(price) AS sqrt_price FROM Movie;
```

| | title character varying (25) | sqrt_price numeric |
|---|---|---|
| 1 | Bloody Vengeance | 10.000000000000000 |
| 2 | The Firm | 14.142135623730950 |
| 3 | Pretty Woman | 12.247448713915890 |
| 4 | Home Alone | 12.269881825021788 |
| 5 | The Fugitive | 14.142135623730950 |
| 6 | Coma | 10.000000000000000 |
| 7 | Dracula | 12.257650672131263 |
| 8 | Quick change | 10.000000000000000 |
| 9 | Gone with the wind | 14.142135623730950 |
| 10 | Carry on Doctor | 10.000000000000000 |

21. Display the type and the average price of each movie type.

```
SELECT type, AVG(price) AS avg_price FROM Movie GROUP BY type;
```

| | type character varying (10) | avg_price numeric |
|---|---|---|
| 1 | Thriller | 200.0000000000000000 |
| 2 | action | 100.0000000000000000 |
| 3 | Comedy | 116.8500000000000000 |
| 4 | thriller | 200.0000000000000000 |
| 5 | Drama | 200.0000000000000000 |
| 6 | Horror | 150.2500000000000000 |
| 7 | Suspense | 100.0000000000000000 |
| 8 | romance | 150.0000000000000000 |

22. Try out: select * from movie group by(type);
23. Try out: select count(mvno) from movie where count(mvno)>2;
24. Find the name and movie of the customer who have issued a movie.

```
SELECT c.Fname, c.Lname, m.title
FROM Cust c
JOIN Invoice i ON c.custid = i.custid
JOIN Movie m ON i.mvno = m.mvno;
```

| | fname character varying (15) | lname character varying (15) | title character varying (25) |
|---|---|---|---|
| 1 | Ivan | Bayross | Home Alone |
| 2 | Vandana | Saitwal | Pretty Woman |
| 3 | Vandana | Saitwal | Bloody Vengeance |
| 4 | Pramada | Jaguste | Coma |
| 5 | Basu | Navindgi | Dracula |
| 6 | Rukmini | [null] | The Firm |
| 7 | Ravi | Sreedhar | Gone with the wind |
| 8 | Ivan | Bayross | Gone with the wind |
| 9 | Pramada | Jaguste | The Fugitive |
| 10 | Rukmini | [null] | Quick change |

25. List the names, areas and cust - id of customers without phone numbers.

```
SELECT Fname, area, custid FROM Cust WHERE phoneno IS NULL;
```

| | fname character varying (15) | area character varying (2) | custid [PK] character varying (3) |
|---|---|---|---|
| 1 | Ravi | va | A05 |

26. Delete all the records having return date before 10th July'93

```
DELETE FROM Invoice WHERE retDate < TO_DATE('10-jul-1993','DD-MON-YYYY');
```

27. Change the area of cust – id 'a05' to 'vs'.

```
UPDATE Cust SET area = 'vs' WHERE custid = 'A05';
```

28. Select the title, custid, mvno for all the movies that are issued.

```
SELECT m.title, i.custid, i.mvno
FROM Invoice i
JOIN Movie m ON i.mvno = m.mvno;
```

| | title character varying (25) | custid character varying (3) | mvno numeric (2) |
|---|---|---|---|
| 1 | Home Alone | A01 | 4 |
| 2 | Pretty Woman | A02 | 3 |
| 3 | Bloody Vengeance | A02 | 1 |
| 4 | Coma | A03 | 6 |
| 5 | Dracula | A04 | 7 |
| 6 | The Firm | A06 | 2 |
| 7 | Gone with the wind | A05 | 9 |
| 8 | Gone with the wind | A01 | 9 |
| 9 | The Fugitive | A03 | 5 |
| 10 | Quick change | A06 | 8 |

29. Display the month (in alphabets) in which customer are supposed to return the movies.

```
SELECT invno, TO_CHAR(retDate, 'Month') AS return_month FROM Invoice;
```

| | invno [PK] character varying (3) | return_month text |
|---|---|---|
| 1 | i01 | July |
| 2 | i02 | August |
| 3 | i03 | August |
| 4 | i04 | September |
| 5 | i05 | August |
| 6 | i06 | September |
| 7 | i07 | July |
| 8 | i08 | August |
| 9 | i09 | July |
| 10 | 110 | September |

30. Find out if the movie starring *Amir* is issued to any customer and list the custid to whom it is issued.

```
SELECT i.custid
FROM Invoice i
JOIN Movie m ON i.mvno = m.mvno
WHERE m.star LIKE '%Amir%';
```

| | custid character varying (3) |
|---|---|

31. Find out the title of the movie that have been issued to the customer whose fname is vandana.

```
SELECT m.title
FROM Invoice i
JOIN Cust c ON i.custid = c.custid
JOIN Movie m ON i.mvno = m.mvno
WHERE LOWER(c.Fname) = 'vandana';
```

| | title character varying (25) |
|---|---|
| 1 | Pretty Woman |
| 2 | Bloody Vengeance |

32. Add a column remark of type varchar and size 25 to the invoice table.

```
ALTER TABLE Invoice ADD remark VARCHAR(25);
```

33. Find out the movie number which has been issued to customer whose first name is ‘pramada’.

```
SELECT i.mvno
FROM Invoice i
JOIN Cust c ON i.custid = c.custid
WHERE LOWER(c.Fname) = 'pramada';
```

| | mvno numeric (2) |
|---|---|
| 1 | 6 |
| 2 | 5 |

34. Change the telephone number of pramada to 466389.
```
UPDATE Cust SET phoneno = 466389 WHERE LOWER(Fname) = 'pramada';
```
35. Change the issuedate of custid 'a01' to 24/07/93.
```
UPDATE Invoice SET issueDate = TO_DATE('24-jul-1993','DD-MON-YYYY') WHERE custid = 'A01';
```
36. Change the price of 'Gone with the wind' to Rs. 250.00.
```
UPDATE Movie SET price = 250.00 WHERE title = 'Gone with the wind';
```
37. Delete the record with invoice number 'i05' from the invoice table.
```
DELETE FROM Invoice WHERE invno = 'i05';
```
38. Delete all the records having return date before 10th July '93
```
DELETE FROM Invoice WHERE retDate < TO_DATE('10-jul-1993','DD-MON-YYYY');
```
39. Change the area of custid a05 to ‘vs’.
```
UPDATE Cust SET area = 'vs' WHERE custid = 'A05';
```
40. Insert fname for customer with lname Rukmini
```
UPDATE Cust SET Fname = 'Devi' WHERE Lname = 'Rukmini';
```
41. Increase the price of all movies by 100.00
```
UPDATE Movie SET price = price + 100.00;
```
42. Change the area of customers name with 2nd letter **r** to LA
```
UPDATE Cust SET area = 'LA' WHERE SUBSTR(Fname,2,1) = 'r';
```
43. Delete all the records of customers who are from ba
```
DELETE FROM Cust WHERE area = 'ba';
```