

## ASSIGNMENT : 04

- ① Create the following tables with the constraints mentioned:  
 Note: the data type and size should be given in relevance with the data to be inserted. Constraints name are not required to be given for this assignment.

### Customer

Movie Cust-id	Fname	Lname	Area	Phone
Primary key A01	Not Null Ivan		Not Null SA	6125467
A02	Vandana	Ray	MU	5560379
A03	Pramada	Jaguste	DA	4560389
A04	Basu	Navindi	BA	6125401
A05	Ravi	Shridhar	NA	null
A06	Rukmini	Aiyer	AN	5125274

- CREATE TABLE Customer (
 Cust-id VARCHAR(3) NOT NULL PRIMARY KEY,  
 Fname VARCHAR(30) NOT NULL,  
 Lname VARCHAR(30),  
 Area VARCHAR(5) NOT NULL  
 Phone VARCHAR(15) NULL  
 );
- INSERT INTO Customer (Cust-id, Fname, Lname, Area, Phone)  
 Values  
 ("A01", "Ivan", "Ross", "SA", 6125467),  
 ("A02", "Vandana", "Ray", "MU", 5560379),  
 ("A03", "Pramada", "Jaguste", "DA", 4560389),  
 ("A04", "Basu", "Navindi", "BA", 6125401),  
 ("A05", "Ravi", "Shridhar", "NA", NULL),

- ( "A06", "Rukmini", "Aiyer", "GH", 5125274 );
- SELECT \* FROM Customer;
- CREATE TABLE Movie (
 MV-no INT NOT NULL PRIMARY KEY,
 Cust-id VARCHAR (3) NOT NULL,
 Title VARCHAR (60) NOT NULL,
 Star VARCHAR (3) NOT NULL,
 Price INT NOT NULL,
 CONSTRAINT chk-price CHECK (Price Between 100 AND 250),
 CONSTRAINT fk-movie-customer FOREIGN KEY (Cust-id) REFERENCES Customer (Cust-id)
 );
- INSERT INTO Movie (MV-no, Cust-id, Title, Star, Price)
 values
 (1, 'A02', 'Bloody', 181),
 (2, 'A04', 'The Firm', 'TC', 200),
 (3, 'A01', 'Pretty Woman', 'RG', 151),
 (4, 'A06', 'Home Alone', 'MC', 150),
 (5, 'A05', 'The Fugitive', 'MF', 200),
 (6, 'A03', 'Coma', 'MD', 100),
 (7, 'A02', 'Dracula', 'GO', 150),
 (8, 'A06', 'Quick Change', 'BM', 100),
 (9, 'A03', 'Gone with the Wind', 'CB', 200),
 (10, 'A05', 'Carry on Doctor', 'LP', 100);

(2) Prove that entity integrity constraint is ensured by both the tables. (2 conditions to be checked).

- `SELECT Cust_id FROM Customer WHERE Cust_id IS NULL;`

Output:

Empty set.

- `SELECT Mv_no FROM Movie WHERE Mv_no IS NULL;`

Output:

Empty set.

(3) Prove that referential integrity constraint is ensured by both the tables.

- `SELECT DISTINCT Cust_id FROM Movie WHERE Cust_id NOT IN (SELECT cust_id FROM Customer);`

Output

Empty set.

(4) Prove that domain integrity constraint is ensured by the Movie table.

- `SELECT * FROM Movie WHERE PRICE Price < 100 OR Price > 250;`

Output

Empty set.

(5) Display the movie titles whose price is greater than 100 but less than 200.

`SELECT Title FROM Movie WHERE Price > 100 AND Price < 200;`

Title

Pretty Women  
Home Alone  
Dracula

⑥ Display the cust\_id who have seen movies having Stars as either JC or TC or MC.

- SELECT cust\_id FROM Movie WHERE Star IN ("JC", "TC", "MC");

Cust_id
A02
A04
A06

⑦ Display the details of those customers who have an A in their area name.

- SELECT \* FROM Customer WHERE Area LIKE "%A%";

Cust_id	Fname	Lname	Area	Phone
A01	Ivan	Ross	SA	6125467
A03	Pramada	Jayuste	DA	4560389
A04	Basu	Navindi	BA	6125401
A05	Ravi	Shridhar	NA	NULL

⑧ Display the movie titles, whose price is within 180 and the movie titles are of exactly 6 letters.

- SELECT Title FROM Movie WHERE Price <= 180 AND LENGTH (Title) = 6;

Title
Coma

⑨ Display the movie name, their original prices and the prices after 10% increment. Give alias name to the increment%. Give alias name to the incremented price column.

- SELECT ~~CONCAT~~ Title, Price AS OriginalPrice, Price \* 1.10 AS IncrementedPrice  
FROM Movie;

Title	Original Price	Incremented Price.
Bloody	181	199.1
The Firm	200	220
Pretty Woman.	151	166.1
Home Alone	150	165
The Fugitive	200	220
Coma	160	110
Dracula	150	165
Quick Change	100	110
Gone with the Wind	200	220
Carry on Doctor	150	110

(10) Display all the customer details in the following way.  
 'Ivan Ross stays in SA and his phone number is  
 6125467.'

- SELECT CONCAT ("Fname", " ", "Lname", "stays in", "Area", " and his phone number is", "Phone", ".") AS DETAILS.

(11) Add not null constraint to the Lname field  
 in Customer.

- ALTER TABLE Customer MODIFY Lname VARCHAR(20) NOT NULL;

Output

Query OK, 0 rows affected.

(12) Display the customer name whose phone number  
 is not recorded.

- SELECT Fname, Lname FROM Customer WHERE Phone IS NULL;

Fname	Lname.
Ravi	Shridhar.

- (13) Add the phone number according to your own wish for the person mentioned in problem no 7.

- UPDATE Customer SET Phone = 99999999 WHERE Cust\_id = "A05";

Cust_id	Phone
A05	99999999

- (14) Display the unique customer id's from movie table.

- SELECT DISTINCT Cust\_id FROM Movie;

Cust_id
A01
A02
A03
A04
A05
A06

- (15) Remove the not null constraint from Star Column in movie table.

- ALTER TABLE Movie MODIFY Star VARCHAR(10)  
NULL;

- (16) Delete any row from the Customer table. If you cannot delete, then note the error message displayed.
- DELETE FROM Customer WHERE Cust\_id = "A01";

### Output

ERROR 1451 (23000): Cannot delete or update a parent row: a foreign key constraint fails.

(17) Delete any row from the Movie table. If you cannot delete, then note the error message displayed.

- `DELETE FROM Movie WHERE Mv-No = 1;`

Output

Query OK, 1 row affected.

(18) Drop the Customer table. If you cannot drop, then note the error message displayed.

- `DROP TABLE Customer;`

Output

ERROR 3730 (HY000): Cannot drop table "Customer" because it is referenced by a foreign key constraint.

(19) Drop the Movie table. If you cannot drop, then note the error message displayed.

- `DROP TABLE Movie;`

Output

Query OK, 0 rows affected.

(20) Drop the Foreign key from Movie table.

- `ALTER TABLE Movie DROP FOREIGN KEY movie_ibfk_1;`

Output

Query OK, 0 rows affected.

A. Alvarz  
22/9/25