**5.Execute DDL/DML statements which demonstrate the use of views. Update the base table using its corresponding view. Also consider restrictions on updatable views and perform view creation from multiple tables.**

**Ans :**

**-- Step 1: Create the database**

CREATE DATABASE IndianCollegeDB;

**-- Step 2: Use the database**

USE IndianCollegeDB;

**-- Step 3: Create the Instructor table with Primary Key**

CREATE TABLE Instructor (

InstructorID INT PRIMARY KEY,

Name VARCHAR(100),

Department VARCHAR(100)

);

**-- Step 4: Create the Student table with Primary Key**

CREATE TABLE Student (

StudentID INT PRIMARY KEY,

Name VARCHAR(100),

Email VARCHAR(100)

);

**-- Step 5: Create the Course table with Foreign Key reference to Instructor table**

CREATE TABLE Course (

CourseID INT PRIMARY KEY,

Title VARCHAR(100),

InstructorID INT,

FOREIGN KEY (InstructorID) REFERENCES Instructor(InstructorID)

);

**-- Step 6: Create the Enrollment table (junction table) with Foreign Keys**

CREATE TABLE Enrollment (

EnrollmentID INT PRIMARY KEY,

StudentID INT,

CourseID INT,

EnrollmentDate DATE,

FOREIGN KEY (StudentID) REFERENCES Student(StudentID),

FOREIGN KEY (CourseID) REFERENCES Course(CourseID)

);

**-- Step 7: Insert sample data into Instructor table**

INSERT INTO Instructor (InstructorID, Name, Department) VALUES

(1, 'Dr. Rajesh Kumar', 'Computer Science'),

(2, 'Prof. Meena Agarwal', 'Mathematics'),

(3, 'Dr. Arvind Sharma', 'Physics');

**-- Step 8: Insert sample data into Student table**

INSERT INTO Student (StudentID, Name, Email) VALUES

(1, 'Amit Patel', 'amit.patel@example.com'),

(2, 'Priya Sharma', 'priya.sharma@example.com'),

(3, 'Ravi Kumar', 'ravi.kumar@example.com'),

(4, 'Neha Singh', 'neha.singh@example.com'),

(5, 'Vikram Joshi', 'vikram.joshi@example.com');

**-- Step 9: Insert sample data into Course table**

INSERT INTO Course (CourseID, Title, InstructorID) VALUES

(1, 'Introduction to Programming', 1),

(2, 'Data Structures', 1),

(3, 'Calculus I', 2),

(4, 'Quantum Mechanics', 3),

(5, 'Linear Algebra', 2);

**-- Step 10: Insert sample data into Enrollment table**

INSERT INTO Enrollment (EnrollmentID, StudentID, CourseID, EnrollmentDate) VALUES

(1, 1, 1, '2025-04-01'),

(2, 2, 2, '2025-04-02'),

(3, 3, 3, '2025-04-03'),

(4, 4, 4, '2025-04-04'),

(5, 5, 5, '2025-04-05');

**-- Step 11: Create a simple view that selects student names and emails**

CREATE VIEW StudentView AS

SELECT Name, Email

FROM Student;

**-- Query the StudentView to show all students and their emails**

SELECT \* FROM StudentView;

**-- Step 12: Create a view that joins Student, Enrollment, and Course tables**

CREATE VIEW StudentCourseView AS

SELECT S.Name AS StudentName, C.Title AS CourseTitle, E.EnrollmentDate

FROM Student S

JOIN Enrollment E ON S.StudentID = E.StudentID

JOIN Course C ON E.CourseID = C.CourseID;

**-- Query the StudentCourseView to show student names, course titles, and enrollment** dates

SELECT \* FROM StudentCourseView;

**-- Step 13: Update through the view on the Student table (StudentView)**

UPDATE StudentView

SET Email = 'amit.newemail@example.com'

WHERE Name = 'Amit Patel';

**-- Verify the update in the base Student table**

SELECT \* FROM Student WHERE Name = 'Amit Patel';

**-- Step 14: Create a view with aggregate function (non-updatable view)**

CREATE VIEW CourseEnrollmentCount AS

SELECT C.Title AS CourseTitle, COUNT(E.StudentID) AS StudentCount

FROM Course C

LEFT JOIN Enrollment E ON C.CourseID = E.CourseID

GROUP BY C.Title;

**-- Query the CourseEnrollmentCount view (non-updatable view)**

SELECT \* FROM CourseEnrollmentCount;

**-- Try updating through a non-updatable view (this will fail)**

**-- Uncomment the following line to see the error**

UPDATE CourseEnrollmentCount

SET StudentCount = 10

WHERE CourseTitle = 'Introduction to Programming';

**-- Step 15: Drop the StudentCourseView if no longer needed**

DROP VIEW StudentCourseView;

**-- Step 16: Drop the database if no longer needed (optional)**

DROP DATABASE IndianCollegeDB;

**-- ------------- Commands to Show Desired Output ------------**

**-- Show the data in the StudentView (Names and Emails)**

SELECT \* FROM StudentView;

**-- Show the data in the StudentCourseView (Student Name, Course Title, Enrollment Date)**

SELECT \* FROM StudentCourseView;

**-- Show the data in the CourseEnrollmentCount (Course Title, Student Count)**

SELECT \* FROM CourseEnrollmentCount;

**-- After updating through the view, show the updated email for Amit Patel**

SELECT \* FROM Student WHERE Name = 'Amit Patel';

**-- Show the error when trying to update a non-updatable view**

**-- Uncomment the following lines to see the error**

UPDATE CourseEnrollmentCount

SET StudentCount = 10

WHERE CourseTitle = 'Introduction to Programming';