**Task1**

**Project Title: Academic Management System (using SQL)**

**1. Database Creation:**

CREATE TABLE StudentInfo (

STU\_A INT PRIMARY KEY,

STU\_NAME VARCHAR (100) NOT NULL,

DOB DATE,

PHONE\_NO VARCHAR (15),

EMAIL\_ID VARCHAR (100),

ADDRESS VARCHAR (255)

);

CREATE TABLE CoursesInfo (

COURSE\_B INT PRIMARY KEY,

COURSE\_NAME VARCHAR(100) NOT NULL,

COURSE\_INSTRUCTOR\_NAME VARCHAR(100) NOT NULL

);

CREATE TABLE EnrollmentInfo (

ENROLLMENT\_C INT PRIMARY KEY,

STU\_A INT,

COURSE\_B INT,

ENROLL\_STATUS VARCHAR (15) CHECK (ENROLL\_STATUS IN ('Enrolled', 'Not Enrolled')),

FOREIGN KEY (STU\_A) REFERENCES StudentInfo(STU\_A),

FOREIGN KEY (COURSE\_B) REFERENCES CoursesInfo(COURSE\_B)

);

**2. Inserting sample data**

Inserting Sample Data into StudentInfo Table

INSERT INTO StudentInfo (STU\_A, STU\_NAME, DOB, PHONE\_NO, EMAIL\_ID, ADDRESS) VALUES

(1, 'Arjun Reddy', '2000-01-15', '999999999', 'arjun.reddy@example.com', '17/19MG Road, Bengaluru'),

(2, 'Lakshmi Menon', '1999-05-22', '888888888', 'lakshmi.menon@example.com', ‘18/19 Anna Salai, Chennai'),

(3, 'Rajesh Kumar', '2001-08-30', '777777777', 'rajesh.kumar@example.com', ‘20/19 MG Road, Hyderabad');

-- Inserting Sample Data into CoursesInfo Table

INSERT INTO CoursesInfo (COURSE\_B, COURSE\_NAME, COURSE\_INSTRUCTOR\_NAME) VALUES

(101, 'Mathematics', 'Dr. Paul'),

(102, 'Physics', 'Dr. Marie Curie'),

(103, 'Chemistry', 'Dr.Kalam');

-- Inserting Sample Data into EnrollmentInfo Table

INSERT INTO EnrollmentInfo (ENROLLMENT\_C, STU\_A, COURSE\_B, ENROLL\_STATUS) VALUES

(1001, 1, 101, 'Enrolled'),

(1002, 1, 102, 'Not Enrolled'),

(1003, 2, 102, 'Enrolled'),

(1004, 3, 103, 'Enrolled'),

(1005, 3, 101, 'Not Enrolled');

**3. Data Retrieval**

a) Retrieve student details, such as student name, contact information, and enrollment status

SELECT

stu.STU\_NAME, stu.PHONE\_NO, stu.EMAIL\_ID, ei.ENROLL\_STATUS

FROM

StudentInfo stu

JOIN

EnrollmentInfo ei ON stu.STU\_A = ei.STU\_A;

b) Retrieve a list of courses in which a specific student is enrolled

SELECT

stu.STU\_NAME, crs.COURSE\_NAME

FROM

StudentInfo stu

JOIN

EnrollmentInfo ei ON stu.STU\_A = ei.STU\_A

JOIN

CoursesInfo crs ON ei.COURSE\_B = crs.COURSE\_B

WHERE stu.STU\_NAME = 'Arjun Reddy'; -- Replace with the specific student name

C) Retrieve course information, including course name and instructor information

SELECT

crs.COURSE\_NAME, crs.COURSE\_INSTRUCTOR\_NAME

FROM

CoursesInfo crs;

d) Retrieve course information for a specific course

SELECT

crs.COURSE\_NAME, crs.COURSE\_INSTRUCTOR\_NAME

FROM

CoursesInfo crs

WHERE

crs.COURSE\_NAME = 'Mathematics'; -- Replace with the specific course name

e) Retrieve course information for multiple courses

SELECT

crs.COURSE\_NAME,

crs.COURSE\_INSTRUCTOR\_NAME

FROM

CoursesInfo crs

WHERE

crs.COURSE\_NAME IN ('Mathematics', 'Physics');

**4)**

a) Write a query to retrieve the number of students enrolled in each course

SELECT

crs.COURSE\_NAME,

COUNT(ei.STU\_A) AS Num\_Students\_Enrolled

FROM

CoursesInfo crs

JOIN

EnrollmentInfo ei ON crs.COURSE\_B = ei.COURSE\_B

WHERE

ei.ENROLL\_STATUS = 'Enrolled'

GROUP BY

crs.COURSE\_NAME;

b) Write a query to retrieve the list of students enrolled in a specific course

SELECT

crs.COURSE\_NAME, stu.STU\_NAME

FROM

CoursesInfo crs

JOIN

EnrollmentInfo ei ON crs.COURSE\_B = ei.COURSE\_B

JOIN

StudentInfo stu ON ei.STU\_A = stu.STU\_A

WHERE

crs.COURSE\_NAME = 'Mathematics' -- Replace with the specific course name

AND ei.ENROLL\_STATUS = 'Enrolled';

C) Write a query to retrieve the count of enrolled students for each instructor.

SELECT

crs.COURSE\_INSTRUCTOR\_NAME,

COUNT(ei.STU\_A) AS Num\_Students\_Enrolled

FROM

CoursesInfo crs

JOIN

EnrollmentInfo ei ON crs.COURSE\_B = ei.COURSE\_B

WHERE

ei.ENROLL\_STATUS = 'Enrolled'

GROUP BY

crs.COURSE\_INSTRUCTOR\_NAME;

d) Write a query to retrieve the list of students who are enrolled in multiple courses

SELECT

stu.STU\_NAME

FROM

StudentInfo stu

JOIN

EnrollmentInfo ei ON stu.STU\_A = ei.STU\_A

WHERE

ei.ENROLL\_STATUS = 'Enrolled'

GROUP BY

stu.STU\_NAME

HAVING

COUNT(ei.COURSE\_B) > 1;

e) Query to retrieve the courses that have the highest number of enrolled students (arranged from highest to lowest)

SELECT

crs.COURSE\_NAME,

COUNT(ei.STU\_A) AS Num\_Students\_Enrolled

FROM

CoursesInfo crs

JOIN

EnrollmentInfo ei ON crs.COURSE\_B = ei.COURSE\_B

WHERE

ei.ENROLL\_STATUS = 'Enrolled'

GROUP BY

crs.COURSE\_NAME

ORDER BY

Num\_Students\_Enrolled DESC;