Project 1

Title: Academic Management System (using SQL)

Task 1 Database Creation:

a) Create the StudentInfo table with columns STU_ ID, STU_NAME, DOB, PHONE_NO, EMAIL_ID,ADDRESS.

```
CREATE TABLE StudentInfo (
STU_ID INT PRIMARY KEY,
STU_NAME VARCHAR(100),
DOB DATE,
PHONE_NO VARCHAR(15),
EMAIL_ID VARCHAR(100),
ADDRESS VARCHAR(255)
);
```

b) Create the CoursesInfo table with columns COURSE_ID, COURSE_NAME,COURSE_INSTRUCTOR NAME

```
CREATE TABLE CoursesInfo (
COURSE_ID INT PRIMARY KEY,
COURSE_NAME VARCHAR(100),
COURSE_INSTRUCTOR_NAME VARCHAR(100)
);
```

c) Create the EnrollmentInfo with columns ENROLLMENT_ID, STU_ ID, COURSE_ID, ENROLL_STATUS(Enrolled/Not Enrolled). The FOREIGN KEY constraint in the EnrollmentInfo table references the STU_ID column in the StudentInfo table and the COURSE_ID column in the CoursesInfo table.

```
CREATE TABLE EnrollmentInfo (
ENROLLMENT_ID INT PRIMARY KEY,
STU_ID INT,
COURSE_ID INT,
ENROLL_STATUS VARCHAR(20),
FOREIGN KEY (STU_ID) REFERENCES StudentInfo(STU_ID),
FOREIGN KEY (COURSE_ID) REFERENCES CoursesInfo(COURSE_ID)
);
```

Task 2 Data Creation:

Inserting samples into studentInfo table

```
INSERT INTO StudentInfo (STU_ID, STU_NAME, DOB, PHONE_NO, EMAIL_ID, ADDRESS) VALUES
(1, 'Saranya', '2000-01-01', '1234567890', 'Saranya@gmail.com', '123 Main St'),
(2, 'Abi', '1999-05-15', '9876543210', 'Abi@gmail.com', '456 Elm St'),
(3, 'Priya', '2001-12-12', '5556667777', 'Priya@mail.com', '789 Oak St');
```

Inserting samples into CoursesInfo table

```
INSERT INTO CoursesInfo (COURSE_ID, COURSE_NAME, COURSE_INSTRUCTOR_NAME) VALUES
```

(101, 'Maths', 'Pradeep'),

(102, 'Physics', 'John'),

(103, 'Chemistry', 'Vijay');

Inserting samples into EnrollmentInfo table

INSERT INTO EnrollmentInfo (ENROLLMENT_ID, STU_ID, COURSE_ID, ENROLL_STATUS) VALUES

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

(2, 1, 102, 'Enrolled'),

(3, 2, 101, 'Enrolled'),

(4, 3, 103, 'Not Enrolled');

Task 3 - Retrieve the Student Information:

a) Write a query to retrieve student details, such as student name, contact information, and Enrollment status.

Query:

```
SELECT STU_NAME, PHONE_NO, EMAIL_ID, ENROLL_STATUS
FROM StudentInfo
JOIN EnrollmentInfo
ON StudentInfo.STU_ID = EnrollmentInfo.STU_ID
GROUP BY 1,2,3,4;
```

Output:

STU_NAME	PHONE_NO	EMAIL_ID	ENROLL_STATUS
Saranya	1234567890	Saranya@gmail.com	Enrolled
Abi	9876543210	Abi@gmail.com	Enrolled
Priya	5556667777	Priya@mail.com	Not Enrolled

b) Write a query to retrieve a list of courses in which a specific student is enrolled.

Query:

```
SELECT STU_NAME, COURSE_NAME
FROM CoursesInfo
JOIN EnrollmentInfo
ON CoursesInfo.COURSE_ID = EnrollmentInfo.COURSE_ID
JOIN StudentInfo
ON EnrollmentInfo.STU_ID = StudentInfo.STU_ID
WHERE STU_NAME = 'Priya';
```

Output:

STU_NAME	COURSE_NAME
Priya	Chemistry

c) Write a query to retrieve course information, including course name, instructor information.

Query:

```
SELECT COURSE_NAME, COURSE_INSTRUCTOR_NAME
FROM CoursesInfo;
```

Output:

COURSE_NAME	COURSE_INSTRUCTOR_NAME
Maths	Pradeep
Physics	John
Chemistry	Vijay

d) Write a query to retrieve course information for a specific course.

Query:

```
SELECT COURSE_NAME, COURSE_INSTRUCTOR_NAME
FROM CoursesInfo
WHERE COURSE_ID = 101;
```

Output:

COURSE_NAME	COURSE_INSTRUCTOR_NAME
Maths	Pradeep

e) Write a query to retrieve course information for multiple courses.

Query:

```
SELECT COURSE_NAME, COURSE_INSTRUCTOR_NAME
FROM CoursesInfo
WHERE COURSE_ID IN (101, 102,103);
```

Ouput:

COURSE_NAME	COURSE_INSTRUCTOR_NAME
Maths	Pradeep
Physics	John
Chemistry	Vijay

f) Test the queries to ensure accurate retrieval of student information.(execute the queries and verify the results against the expected output.)

Query:

```
SELECT STU_NAME, PHONE_NO, EMAIL_ID, ENROLL_STATUS, COURSE_NAME
FROM StudentInfo
LEFT JOIN EnrollmentInfo
ON StudentInfo.STU_ID = EnrollmentInfo.STU_ID
LEFT JOIN CoursesInfo
ON EnrollmentInfo.COURSE_ID = CoursesInfo.COURSE_ID
```

ORDER BY 1;

Output:

STU_NAME	PHONE_NO	EMAIL_ID	ENROLL_STATUS	COURSE_NAME
Abi	9876543210	Abi@gmail.com	Enrolled	Maths
Priya	5556667777	Priya@mail.com	Not Enrolled	Chemistry
Saranya	1234567890	Saranya@gmail.com	Enrolled	Maths
Saranya	1234567890	Saranya@gmail.com	Enrolled	Physics

Task 4 - Reporting and Analytics (Using joining queries)

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

Query:

```
SELECT COURSE_NAME, COUNT(EnrollmentInfo.STU_ID) AS Student_Count
FROM CoursesInfo
LEFT JOIN EnrollmentInfo
ON CoursesInfo.COURSE_ID = EnrollmentInfo.COURSE_ID
WHERE ENROLL_STATUS = 'Enrolled'
GROUP BY 1;
```

Output:

COURSE_NAME	Student_Count
Maths	2
Physics	1

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

Query:

```
SELECT STU_NAME, CI.COURSE_ID,COURSE_NAME
FROM StudentInfo SI
```

```
LEFT JOIN EnrollmentInfo EI
ON SI.STU_ID = EI.STU_ID
LEFT JOIN CoursesInfo CI
ON EI.COURSE_ID = CI.COURSE_ID
WHERE CI.COURSE_ID IN (101,102)
AND ENROLL_STATUS = 'Enrolled'
ORDER BY 1;
```

Output:

STU_NAME	COURSE_ID	COURSE_NAME
Abi	101	Maths
Saranya	101	Maths
Saranya	102	Physics

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

Query:

```
SELECT COURSE_INSTRUCTOR_NAME, COUNT(EI.STU_ID) AS Enrolled_Students
FROM CoursesInfo CI
LEFT JOIN EnrollmentInfo EI
ON CI.COURSE_ID = EI.COURSE_ID
WHERE ENROLL_STATUS = 'Enrolled'
GROUP BY 1;
```

Output:

COURSE_INSTRUCTOR_NAME	Enrolled_Students
Pradeep	2
John	1

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

Query:

```
SELECT STU_NAME
FROM StudentInfo SI
JOIN EnrollmentInfo EI
ON SI.STU_ID = EI.STU_ID
WHERE ENROLL_STATUS = 'Enrolled'
GROUP BY SI.STU_ID, STU_NAME
HAVING COUNT(COURSE_ID) > 1;
```

Output:

STU_NAME
Saranya

e. Write a query to retrieve the courses that have the highest number of enrolled students(arranging from highest to lowest).

Query:

```
SELECT COURSE_NAME, COUNT(EI.STU_ID) AS STUDENT_COUNT
FROM CoursesInfo CI
LEFT JOIN EnrollmentInfo EI
ON CI.COURSE_ID = EI.COURSE_ID
WHERE ENROLL_STATUS = 'Enrolled'
GROUP BY COURSE_NAME
ORDER BY Student_Count DESC;
```

Output:

COURSE_NAME	STUDENT_COUNT
Maths	2
Physics	1