Week 3 LAB 2: Practice 2 Question

1. Create a table named STUDENT, COURSE, ENROLLMENT with the following attributes:

Student -

student_id – INTEGER, PRIMARY KEY

name – VARCHAR(50), NOT NULL

department – VARCHAR(30)

year – INTEGER

Course -

course_id – INTEGER, PRIMARY KEY

course_name – VARCHAR(50), NOT NULL

department – VARCHAR(30)

credits – INTEGER

Enrollment -

student_id – INTEGER

course_id – INTEGER

semester – INTEGER

In which Set:

- student_id as a FOREIGN KEY referencing STUDENT(student_id)
- course_id as a FOREIGN KEY referencing COURSE(course_id)

After creating a Table in a database, insert at least 10 records in the STUDENT and COURSE tables. Insert records into the ENROLLMENT table using valid student and course IDs.

2. Write an SQL query to display the names of all students enrolled in any course or display all student, course and enrolment tables. Include outcome in the assignment file also.
3. Write an SQL query using INNER JOIN to display:
   - Student name
   - Course name
4. Use the EXPLAIN command on the join query discussed in Question 3 to identify:
   - The join algorithm used
   - The order of table access
   - Note - DBMS uses a Nested Loop / Hash Join based on optimization.

5. Write an SQL query to display the names of students enrolled in courses offered by the CSE department. Use EXPLAIN to observe pipelining of operations.
6. Write an SQL query to find students who are enrolled in more than one course. Use EXPLAIN to analyze the execution plan and note:
   - Estimated cost
   - Number of rows processed
7. Write two equivalent SQL queries to display student names: Using JOIN and Using a subquery. Use EXPLAIN to compare the execution plans of both queries. Explain outcome in brief of both in the assignment work.
8. Create an index on student_id in the ENROLLMENT table. Execute a join query again and use EXPLAIN. Compare the execution cost before and after the index is created.
9. Consider the ENROLLMENT table. Perform the following operations using transaction control commands:
   - Start a transaction.
   - Insert a new enrollment record for a student.
   - Display the ENROLLMENT table.
   - Rollback the transaction and display the table again.
   - Commit the transaction and verify the final state.
   - Use EXPLAIN to observe the query processing of the SELECT statement.