Design at least 10 SQL queries for suitable database application using SQL DML statements: Insert, Select, Update, Delete with operators, functions, and set operator.

CREATE TABLE Student (

student\_id INT PRIMARY KEY,

name VARCHAR(50),

age INT,

dept\_no INT,

address VARCHAR(100)

);

CREATE TABLE Instructor (

instructor\_id INT PRIMARY KEY,

name VARCHAR(50),

dept\_no INT,

salary DECIMAL(10, 2),

phone\_number BIGINT

);

CREATE TABLE Course (

course\_id INT PRIMARY KEY,

title VARCHAR(50),

dept\_no INT,

credits INT

);

CREATE TABLE Enrollment (

enrollment\_id INT PRIMARY KEY,

student\_id INT,

course\_id INT,

enrollment\_date DATE,

FOREIGN KEY (student\_id) REFERENCES Student(student\_id),

FOREIGN KEY (course\_id) REFERENCES Course(course\_id)

);

**10 dml queries**

1. **Insert a New Student Record**

INSERT INTO Student (student\_id, name, age, dept\_no, address)

VALUES (1, 'John Doe', 20, 101, '123 Elm Street');

1. **Insert Another Student Record**

INSERT INTO Student (student\_id, name, age, dept\_no, address)

VALUES (2, 'Jane Smith', 21, 102, '456 Oak Avenue');

1. **Insert a New Instructor Record**

INSERT INTO Instructor (instructor\_id, name, dept\_no, salary, phone\_number)

VALUES (10, 'Dr. Smith', 101, 75000.00, 9876543210);

1. **Insert a New Course Record**

INSERT INTO Course (course\_id, title, dept\_no, credits)

VALUES (201, 'Mathematics', 101, 3);

1. **Enroll a Student in a Course**

INSERT INTO Enrollment (enrollment\_id, student\_id, course\_id, enrollment\_date)

VALUES (1, 1, 201, '2024-11-01'); -- Ensure enrollment\_id is unique

1. **Update a Student’s Address**

UPDATE Student

SET address = '789 Pine Road'

WHERE student\_id = 1;

1. . **Update an Instructor’s Salary**

UPDATE Instructor

SET salary = salary \* 1.10

WHERE instructor\_id = 10;

1. **Delete a Course Enrollment**

DELETE FROM Enrollment

WHERE student\_id = 1 AND course\_id = 201;

1. **Retrieve All Students**

SELECT \* FROM Student;

1. **Count the Number of Courses Offered in Each Department**

SELECT dept\_no, COUNT(course\_id) AS course\_count

FROM Course

GROUP BY dept\_no;

**Theory for SQL DML Queries in a School Database Application**

**Introduction to SQL DML**

Structured Query Language (SQL) is a standard language for managing relational databases. Data Manipulation Language (DML) is a subset of SQL that allows users to perform operations such as inserting, updating, deleting, and retrieving data. In this assignment, we utilize DML commands to interact with a school database consisting of four tables: Student, Instructor, Course, and Enrollment.

**Database Schema Overview**

Student Table: Stores student details including student\_id, name, age, department number (dept\_no), and address.

Instructor Table: Contains instructor information with attributes like instructor\_id, name, dept\_no, salary, and phone number.

Course Table: Holds data about courses, including course\_id, title, dept\_no, and credit hours.

Enrollment Table: Tracks student enrollments in courses, including enrollment\_id, student\_id, course\_id, and enrollment date.

SQL DML Operations

Insert: Used to add new records to the tables.

Example:

sql

Copy code

INSERT INTO Student (student\_id, name, age, dept\_no, address) VALUES (1, 'John Doe', 20, 101, '123 Elm Street');

Update: Allows modification of existing records.

Example:

sql

Copy code

UPDATE Student SET address = '789 Pine Road' WHERE student\_id = 1;

Delete: Removes records from the tables.

Example:

sql

Copy code

DELETE FROM Enrollment WHERE student\_id = 1 AND course\_id = 201;

Select: Retrieves data from the tables for viewing or analysis.

Example:

sql

Copy code

SELECT \* FROM Student;