1.Create the database named "SISDB"

CREATE DATABASE student_information_system;

2. Define the schema for the Students, Courses, Enrollments, Teacher, and Payments tables based on the provided schema. Write SQL scripts to create the mentioned tables with appropriate data types, constraints, and relationships. a. Students b. Courses c. Enrollments d. Teacher e. Payments

```
CREATE TABLE Students (
    student_id INT PRIMARY KEY AUTO_INCREMENT,
    first_name VARCHAR(50) NOT NULL,
    last_name VARCHAR(50) NOT NULL,
    date_of_birth DATE NOT NULL,
    email VARCHAR(100) UNIQUE NOT NULL,
    phone_number VARCHAR(15) NOT NULL
);

CREATE TABLE Teachers (
    teacher_id INT PRIMARY KEY AUTO_INCREMENT,
    first_name VARCHAR(50) NOT NULL,
    last_name VARCHAR(50) NOT NULL,
    email VARCHAR(100) UNIQUE NOT NULL
);

CREATE TABLE Courses (
    course_id INT PRIMARY KEY AUTO_INCREMENT,
    course_name VARCHAR(100) NOT NULL,
    credits INT NOT NULL,
    teacher_id INT,
    FOREIGN KEY (teacher_id) REFERENCES Teachers(teacher_id) ON DELETE SET NULL
);

CREATE TABLE Enrollments (
    enrollment_id INT PRIMARY KEY AUTO_INCREMENT,
```

```
    student_id INT,
    course_id INT,
    enrollment_date DATE,
    FOREIGN KEY (student_id) REFERENCES Students(student_id) ON DELETE CASCADE ON UPDATE CASCADE,
    FOREIGN KEY (course_id) REFERENCES Courses(course_id) ON DELETE CASCADE ON UPDATE CASCADE
);

CREATE TABLE Payments (
    payment_id INT PRIMARY KEY AUTO_INCREMENT,
    student_id INT,
    amount INT NOT NULL,
    payment_date DATE,
    FOREIGN KEY (student_id) REFERENCES Students(student_id) ON DELETE CASCADE
);
```

5. Create appropriate Primary Key and Foreign Key constraints for referential integrity

```
FOREIGN KEY (student_id) REFERENCES Students(student_id) ON DELETE CASCADE ON UPDATE CASCADE,
    FOREIGN KEY (course_id) REFERENCES Courses(course_id) ON DELETE CASCADE ON UPDATE CASCADE
FOREIGN KEY (teacher_id) REFERENCES Teachers(teacher_id) ON DELETE SET NULL
FOREIGN KEY (teacher_id) REFERENCES Teachers(teacher_id) ON DELETE SET NULL
    payment_id INT PRIMARY KEY AUTO_INCREMENT,
    teacher_id INT PRIMARY KEY AUTO_INCREMENT,
    student_id INT PRIMARY KEY AUTO_INCREMENT,
```

6.Insert at least 10 sample records into each of the following tables. i. Students ii. Courses iii. Enrollments iv. Teacher v. Payments

```
INSERT INTO Students (first_name, last_name, date_of_birth, email, phone_number) VALUES
('John', 'Doe', '2002-05-15', 'john.doe@example.com', '9876543210'),
('Emma', 'Smith', '2001-09-20', 'emma.smith@example.com', '9876543211'),
```

('Michael', 'Brown', '2003-02-10', 'michael.brown@example.com', '9876543212'),

('Sophia', 'Johnson', '2000-08-05', 'sophia.johnson@example.com', '9876543213'),

('Liam', 'Williams', '2002-11-12', 'liam.williams@example.com', '9876543214'),

('Olivia', 'Jones', '2001-04-18', 'olivia.jones@example.com', '9876543215'),

('Noah', 'Davis', '2003-07-30', 'noah.davis@example.com', '9876543216'),

('Ava', 'Miller', '2002-06-25', 'ava.miller@example.com', '9876543217'),

('James', 'Wilson', '2000-09-09', 'james.wilson@example.com', '9876543218'),

('Isabella', 'Moore', '2001-10-03', 'isabella.moore@example.com', '9876543219');

INSERT INTO Teachers (first_name, last_name, email) VALUES

('Alice', 'Johnson', 'alice.johnson@example.com'),

('Bob', 'Williams', 'bob.williams@example.com'),

('Charlie', 'Brown', 'charlie.brown@example.com'),

('David', 'Smith', 'david.smith@example.com'),

('Emma', 'Miller', 'emma.miller@example.com'),

('Frank', 'Davis', 'frank.davis@example.com'),

('Grace', 'Moore', 'grace.moore@example.com'),

('Henry', 'Taylor', 'henry.taylor@example.com'),

('Isabella', 'Anderson', 'isabella.anderson@example.com'),

('Jack', 'Thomas', 'jack.thomas@example.com');

```
ysql> SELECT * FROM Teachers;
RROR 1146 (42S02): Table 'student_information_system.teachers' doesn't exist
ysql> SELECT * FROM Courses;
+-----------+------------------------+---------+------------+
| course_id | course_name            | credits | teacher_id |
+-----------+------------------------+---------+------------+
|         1 | Database Systems       |       4 |          1 |
|         2 | Computer Networks      |       3 |          2 |
|         3 | Software Engineering   |       3 |          3 |
|         4 | Artificial Intelligence |       4 |          4 |
|         5 | Web Development        |       3 |          5 |
|         6 | Mobile App Development |       4 |          6 |
|         7 | Cloud Computing        |       3 |          7 |
|         8 | Cyber Security         |       4 |          8 |
|         9 | Data Science           |       4 |          9 |
|        10 | Operating Systems      |       3 |         10 |
+-----------+------------------------+---------+------------+
0 rows in set (0.00 sec)
```

```
INSERT INTO Courses (course_name, credits, teacher_id) VALUES
('Database Systems', 4, 1),
('Computer Networks', 3, 2),
('Software Engineering', 3, 3),
('Artificial Intelligence', 4, 4),
('Web Development', 3, 5),
('Mobile App Development', 4, 6),
('Cloud Computing', 3, 7),
('Cyber Security', 4, 8),
('Data Science', 4, 9),
('Operating Systems', 3, 10);

INSERT INTO Enrollments (student_id, course_id, enrollment_date) VALUES
(1, 1, '2025-01-10'),
(2, 2, '2025-01-12'),
(3, 3, '2025-01-14'),
(4, 4, '2025-01-16'),
(5, 5, '2025-01-18'),
(6, 6, '2025-01-20'),
(7, 7, '2025-01-22'),
(8, 8, '2025-01-24'),
(9, 9, '2025-01-26'),
(10, 10, '2025-01-28');
```

```
mysql> SELECT * FROM Enrollments;
+---------------+------------+-----------+-----------------+
| enrollment_id | student_id | course_id | enrollment_date |
+---------------+------------+-----------+-----------------+
|             1 |          1 |         1 | 2025-01-10      |
|             2 |          2 |         2 | 2025-01-12      |
|             3 |          3 |         3 | 2025-01-14      |
|             4 |          4 |         4 | 2025-01-16      |
|             5 |          5 |         5 | 2025-01-18      |
|             6 |          6 |         6 | 2025-01-20      |
|             7 |          7 |         7 | 2025-01-22      |
|             8 |          8 |         8 | 2025-01-24      |
|             9 |          9 |         9 | 2025-01-26      |
|            10 |         10 |        10 | 2025-01-28      |
+---------------+------------+-----------+-----------------+
```

INSERT INTO Payments (student_id, amount, payment_date) VALUES

(1, 1000, '2025-02-01'),

(2, 1500, '2025-02-03'),

(3, 1200, '2025-02-05'),

(4, 1100, '2025-02-07'),

(5, 1300, '2025-02-09'),

(6, 1400, '2025-02-11'),

(7, 1250, '2025-02-13'),

(8, 1350, '2025-02-15'),

(9, 1450, '2025-02-17'),

(10, 1550, '2025-02-19');

```
mysql> SELECT * FROM Payments;
+------------+------------+--------+--------------+
| payment_id | student_id | amount | payment_date |
+------------+------------+--------+--------------+
|          1 |          1 |   1000 | 2025-02-01   |
|          2 |          2 |   1500 | 2025-02-03   |
|          3 |          3 |   1200 | 2025-02-05   |
|          4 |          4 |   1100 | 2025-02-07   |
|          5 |          5 |   1300 | 2025-02-09   |
|          6 |          6 |   1400 | 2025-02-11   |
|          7 |          7 |   1250 | 2025-02-13   |
|          8 |          8 |   1350 | 2025-02-15   |
|          9 |          9 |   1450 | 2025-02-17   |
|         10 |         10 |   1550 | 2025-02-19   |
+------------+------------+--------+--------------+
10 rows in set (0.00 sec)
```

1. Write an SQL query to insert a new student into the "Students" table with the following details: a. First Name: John

   b. Last Name: Doe c. Date of Birth: 1995-08-15 d. Email: john.doe@example.com e. Phone Number: 1234567890

INSERT INTO Students VALUES (NULL, 'John', 'Dum', '1995-08-15', 'john.doe@example.com', '1234567890');

2. Write an SQL query to enroll a student in a course. Choose an existing student and course and insert a record into the "Enrollments" table with the enrollment date

INSERT INTO Enrollments (student_id, course_id, enrollment_date)

-> VALUES (1, 3, '2025-03-19');

3.Write an SQL query to delete a specific enrollment record from the "Enrollments" table. Select an enrollment record based on the student and course

UPDATE Teacher

-> SET email = 'alice.michel@example.com'

-> WHERE teacher_id = 1;

4. Write an SQL query to delete a specific enrollment record from the "Enrollments" table. Select an enrollment record based on the student and course.

delete from enrollments where student_id=6 and course_id=6;

5. Update the "Courses" table to assign a specific teacher to a course. Choose any course and teacher from the respective tables

> update courses

-> set teacher_id=2

-> where course_id=4;

6. Delete a specific student from the "Students" table and remove all their enrollment records from the "Enrollments" table. Be sure to maintain referential integrity

DELETE FROM Students

-> WHERE student_id = 3;

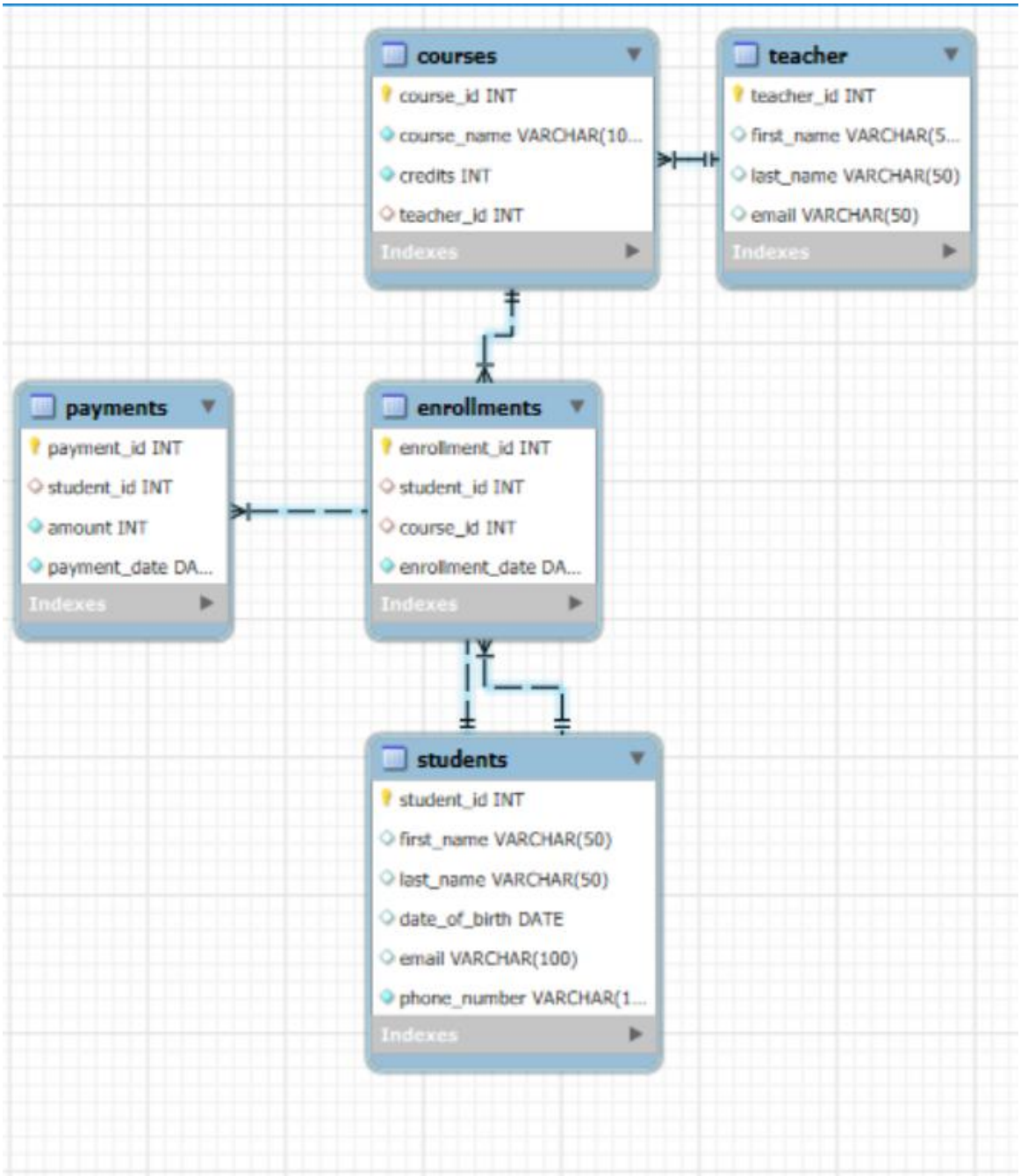
7. Update the payment amount for a specific payment record in the "Payments" table. Choose any payment record and modify the payment amount

update payments set amount=1500 where payment_id=1;

```
mysql> select * from payments;
+------------+------------+--------+--------------+
| payment_id | student_id | amount | payment_date |
+------------+------------+--------+--------------+
|          1 |          1 |   1000 | 2025-02-01   |
|          2 |          2 |   1500 | 2025-02-03   |
|          4 |          4 |   1100 | 2025-02-07   |
|          5 |          5 |   1300 | 2025-02-09   |
|          6 |          6 |   1400 | 2025-02-11   |
|          7 |          7 |   1250 | 2025-02-13   |
|          8 |          8 |   1350 | 2025-02-15   |
|          9 |          9 |   1450 | 2025-02-17   |
|         10 |         10 |   1550 | 2025-02-19   |
+------------+------------+--------+--------------+
```

```
select * from students;
+-------+------------+-----------+---------------+---------------------------+--------------+
ent_id | first_name | last_name | date_of_birth | email                     | phone_number |
+-------+------------+-----------+---------------+---------------------------+--------------+
     1 | John       | Doe       | 2002-05-15    | john.doe@example.com      | 9876543210   |
     2 | Emma       | Smith     | 2001-09-20    | emma.smith@example.com    | 9876543211   |
     4 | Sophia     | Johnson   | 2000-08-05    | sophia.johnson@example.com | 9876543213   |
     5 | Liam       | Williams  | 2002-11-12    | liam.williams@example.com | 9876543214   |
     6 | Olivia     | Jones     | 2001-04-18    | olivia.jones@example.com  | 9876543215   |
     7 | Noah       | Davis     | 2003-07-30    | noah.davis@example.com    | 9876543216   |
     8 | Ava        | Miller    | 2002-06-25    | ava.miller@example.com    | 9876543217   |
     9 | James      | Wilson    | 2000-09-09    | james.wilson@example.com  | 9876543218   |
    10 | Isabella   | Moore     | 2001-10-03    | isabella.moore@example.com | 9876543219   |
    11 | John       | Dum       | 1995-08-15    | john.doe@example.com      | 1234567890   |
+-------+------------+-----------+---------------+---------------------------+--------------+
```

SCHEMA:


courses
course_id INT
course_name VARCHAR(10...
credits INT
teacher_id INT
Indexes
teacher
teacher_id INT
first_name VARCHAR(5...
last_name VARCHAR(50)
email VARCHAR(50)
Indexes
payments
payment_id INT
student_id INT
amount INT
payment_date DA...
Indexes
enrollments
enrollment_id INT
student_id INT
course_id INT
enrollment_date DA...
Indexes
students
student_id INT
first_name VARCHAR(50)
last_name VARCHAR(50)
date_of_birth DATE
email VARCHAR(100)
phone_number VARCHAR(1...
Indexes