Guided Practice

**-- Create the database**

CREATE DATABASE IF NOT EXISTS exam\_db;

USE exam\_db;

**-- Create the students table**

CREATE TABLE IF NOT EXISTS students (

student\_id INT PRIMARY KEY,

name VARCHAR(50),

age INT,

major VARCHAR(50)

);

**-- Create the courses table**

CREATE TABLE IF NOT EXISTS courses (

course\_id INT PRIMARY KEY,

course\_name VARCHAR(50),

credits INT

);

**-- Create the enrollments table**

CREATE TABLE IF NOT EXISTS enrollments (

enrollment\_id INT PRIMARY KEY,

student\_id INT,

course\_id INT,

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

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

);

**-- Create the grades table**

CREATE TABLE IF NOT EXISTS grades (

grade\_id INT PRIMARY KEY,

enrollment\_id INT,

grade INT,

FOREIGN KEY (enrollment\_id) REFERENCES enrollments(enrollment\_id)

);

**-- Insert sample data into the tables**

INSERT INTO students VALUES (1, 'John', 20, 'Computer Science');

INSERT INTO students VALUES (2, 'Mary', 22, 'Mathematics');

INSERT INTO courses VALUES (1, 'Programming', 4);

INSERT INTO courses VALUES (2, 'Algebra', 3);

INSERT INTO enrollments VALUES (1, 1, 1);

INSERT INTO enrollments VALUES (2, 2, 2);

INSERT INTO grades VALUES (1, 1, 90)

INSERT INTO grades VALUES (2, 2, 85);

Produce the relational Graph:

Diagrama

Descripción generada automáticamente

**-- Example queries for the exam**

**-- 1. Show all students majoring in Computer Science.**

SELECT \* FROM students WHERE major LIKE “Computer%”;

**-- 2. Get the course name and grade of John.**

SELECT courses.course\_name, grades.grade FROM courses INNER JOIN enrollments ON courses.course\_id=enrollments.course\_id INNER JOIN students ON students.student\_id=enrollments.student\_id INNER JOIN grades ON enrollments.enrollment\_id=grades.enrollment\_id WHERE students.name=“John”;

**-- 3. Show courses with more than 3 credits.**

SELECT **\*** FROM courses WHERE credits>=3;

**--4. Show the name and age of students who are older than 21.**

SELECT name,age FROM students WHERE age>21;

**--5. Get the names of courses with no enrollments.**

**--6. Show students who have scored above 80.**

**--7 Get the average grades of all students.**

**--8 Get the names and credits of courses John is enrolled in.**

**--9 Show students enrolled in more than one course.**

**--10. Show courses in which Mary is enrolled.**

**--11. Get the total number of credits for all courses.**

**--12. Show courses with no grades assigned**

**--13. Get the names and ages of students enrolled in the Programming course.**

**--14 Show the average grade for each course.**

**--15 Get the names and majors of students enrolled in any course.**