SA School Database

1. Relational Data Model Diagram

Departments(department_id, department_name)
Students(student_id, first_name, last_name, birth_date, department_id)
Teachers(teacher_id, first_name, last_name, hire_date, department_id)
Courses(course_id, course_name, department_id, teacher_id)
Enrollments(enrollment_id, student_id, course_id, enrollment_date)
Grades(grade_id, student_id, course_id, grade)

2. SQL Statements

Create Tables

```
CREATE TABLE Departments (
   department_id INT PRIMARY KEY,
   department_name VARCHAR(100)
);

CREATE TABLE Students (
   student_id INT PRIMARY KEY,
   first_name VARCHAR(50),
   last_name VARCHAR(50),
   birth_date DATE,
   department_id INT
);

CREATE TABLE Teachers (
   teacher_id INT PRIMARY KEY,
   first_name VARCHAR(50),
   last_name VARCHAR(50),
   hire_date DATE,
   department_id INT
);

CREATE TABLE Courses (
   course_id INT PRIMARY KEY,
   course_name VARCHAR(100),
   department_id INT,
   teacher_id INT
);

CREATE TABLE Enrollments (
   enrollment_id INT PRIMARY KEY,
   student_id INT,
   course_id INT,
   enrollment_date DATE
);

CREATE TABLE Grades (
   grade_id INT PRIMARY KEY,
   student_id INT,
   course_id INT,
   grade FLOAT  -- Changed to FLOAT from VARCHAR
);
```

Add Foreign Keys Using ALTER TABLE

```
ALTER TABLE Students
ADD CONSTRAINT FK_Students_Department
FOREIGN KEY (department_id) REFERENCES Departments(department_id);

ALTER TABLE Teachers
ADD CONSTRAINT FK_Teachers_Department
FOREIGN KEY (department_id) REFERENCES Departments(department_id);

ALTER TABLE Courses
ADD CONSTRAINT FK_Courses_Department
FOREIGN KEY (department_id) REFERENCES Departments(department_id);

ALTER TABLE Courses
ADD CONSTRAINT FK_Courses_Teacher
FOREIGN KEY (teacher_id) REFERENCES Teachers(teacher_id);

ALTER TABLE Enrollments
ADD CONSTRAINT FK_Enrollments_Student
FOREIGN KEY (student_id) REFERENCES Students(student_id);

ALTER TABLE Enrollments
ADD CONSTRAINT FK_Enrollments_Course
FOREIGN KEY (course_id) REFERENCES Courses(course_id);

ALTER TABLE Grades
ADD CONSTRAINT FK_Grades_Student
FOREIGN KEY (student_id) REFERENCES Students(student_id);

ALTER TABLE Grades
ADD CONSTRAINT FK_Grades_Course
FOREIGN KEY (course_id) REFERENCES Courses(course_id);
```

3. Insert Data into Tables

1. **Insert Data into Departments**

```
INSERT INTO Departments (department_id, department_name)
VALUES
(1, 'Mathematics'),
(2, 'Computer Science'),
(3, 'Physics');
```

2. **Insert Data into Teachers**

```
INSERT INTO Teachers (teacher_id, first_name, last_name, hire_date,
department_id)
VALUES
(1, 'Alice', 'Smith', '2015-08-15', 1),
(2, 'Bob', 'Johnson', '2017-03-12', 2),
(3, 'Carol', 'Davis', '2018-06-20', 3);
```

3. **Insert Data into Students**

```
INSERT INTO Students (student_id, first_name, last_name,
birth_date, department_id)
VALUES
(1, 'John', 'Doe', '2000-09-25', 1),
(2, 'Jane', 'Doe', '1999-12-10', 2),
(3, 'Tom', 'Brown', '2001-07-14', 3),
(4, 'Emily', 'White', '2000-05-03', 2);
```

4. **Insert Data into Courses**

```
INSERT INTO Courses (course_id, course_name, department_id,
teacher_id)
VALUES
(1, 'Calculus', 1, 1),
(2, 'Data Structures', 2, 2),
(3, 'Quantum Physics', 3, 3),
(4, 'Algorithms', 2, 2);
```

5. **Insert Data into Enrollments**

```
INSERT INTO Enrollments (enrollment_id, student_id, course_id,
enrollment_date)
VALUES
(1, 1, 1, '2024-09-01'),
(2, 2, 2, '2024-09-02'),
(3, 3, 3, '2024-09-03'),
(4, 4, 4, '2024-09-04'),
(5, 1, 2, '2024-09-05'),
(6, 3, 1, '2024-09-06');
```

6. **Insert Data into Grades**

```
INSERT INTO Grades (grade_id, student_id, course_id, grade)
VALUES
(1, 1, 1, 95.0),
(2, 1, 2, 88.5),
(3, 2, 2, 92.0),
(4, 3, 3, 76.0),
(5, 4, 4, 84.0),
(6, 2, 1, 90.0),
(7, 1, 3, 82.5);
```

Questions:

1. Fill the tables with the inserted values above (in case you don’t have a computer)

Departments

Teachers

Students

Courses

Enrollments

Grades

2. Write a SQL query that:
   1. Finds the average grade for each course

   2. Finds the total number of students for each department

   3. Creates a CHECK constraint that ensures grades in the Grades table are between 'A' and 'F'.

   4. Finds the names of teachers who belong to the 'Mathematics' department.

   5. finds all students who were born during the 1990s and first name starts with J

   6. Inserts a record in the Teachers table with values 4, Mike, Philip, 2020-02-02,3

   7. Finds all teachers who have not taught any courses

3. What is the output of the following queries

1. `SELECT DISTINCT department_id FROM Students;`

2. SELECT S.first_name, S.last_name,E.course_id
FROM Students S
RIGHT JOIN Enrollments E ON S.student_id = E.student_id
WHERE E.course_id NOT IN (SELECT course_id FROM Courses WHERE department_id = 2);

3. SELECT C.course_name, G.grade FROM Courses C RIGHT JOIN Grades G ON C.course_id = G.course_id WHERE G.student_id NOT IN (SELECT student_id FROM