**PROJECT NAME: STUDENT ATTENDANCE AND GRADES SYSTEM**

**COURSE: DBMS LAB**

**FACULTY:MISS HIBA**

**GROUP MEMBERS**

**SHAIKH ABDULLAH SIDDIQUE 16979**

**SHAIKH ABDULLAH SIDDIQUI 16530**

**MUHAMMAD AVAIS SIDDIQUI 15848**

**MUHAMMAD YAWAR 16289**

**Introduction**

The Student-Course Management System is a simple database designed to manage students, courses, attendance, and grades. It helps schools and colleges keep records of students, track their course enrollments, check attendance, and store grades. This system makes it easy to manage information and create useful reports.

1. **Table Creation:**

* **Students Table:**

**Attributes:**

StudentID, FirstName, LastName, EnrollmentDate

**Primary Key:**

StudentID

* **Courses Table:**

**Attributes:**

CourseID, CourseName, InstructorName

**Primary Key:**

CourseID

* **Enrollments Table:**

**Attributes:**

EnrollmentID, StudentID, CourseID, EnrollmentDate

**Primary Key:**

EnrollmentID

**Foreign Keys:**

StudentID references Students(StudentID)

CourseID references Courses(CourseID)

* **Attendance Table:**

**Attributes:**

AttendanceID, StudentID, CourseID, AttendanceDate, StatusPrimary Key:

AttendanceID

**Foreign Keys:**

StudentID references Students(StudentID)

CourseID references Courses(CourseID)

* **Grades Table:**

**Attributes:**

GradeID, StudentID, CourseID, Score

**Primary Key:**

GradeIDForeign Keys:

StudentID references Students(StudentID)

CourseID references Courses(CourseID)

**2. Sample Data Insertion:**

* **Students:** Adds three students with enrollment dates.
* **Courses:** Adds two courses with instructor names.
* **Enrollments:** Enrolls all students in both courses.
* **Attendance**: Adds attendance records for all students across multiple dates.
* **Grades:** Assigns scores to students in each course.

**3. Queries:**

* **Attendance Report:** Lists each student’s attendance status for each course on specific dates.
* **Average Course Scores:** Calculates the average score for each course.
* **Top Performers:** Identifies the top scorer for each course.
* **Student Average Scores:** Computes the average score of each student across all their courses.

**PROJECT CODE:**

-- Create Tables

CREATE TABLE Students (

StudentID INT AUTO\_INCREMENT PRIMARY KEY,

FirstName VARCHAR(50) NOT NULL,

LastName VARCHAR(50) NOT NULL,

EnrollmentDate DATE NOT NULL

);

CREATE TABLE Courses (

CourseID INT AUTO\_INCREMENT PRIMARY KEY,

CourseName VARCHAR(100) NOT NULL,

InstructorName VARCHAR(100) NOT NULL

);

CREATE TABLE Enrollments (

EnrollmentID INT AUTO\_INCREMENT PRIMARY KEY,

StudentID INT NOT NULL,

CourseID INT NOT NULL,

EnrollmentDate DATE NOT NULL,

FOREIGN KEY (StudentID) REFERENCES Students(StudentID),

FOREIGN KEY (CourseID) REFERENCES Courses(CourseID)

);

CREATE TABLE Attendance (

AttendanceID INT AUTO\_INCREMENT PRIMARY KEY,

StudentID INT NOT NULL,

CourseID INT NOT NULL,

AttendanceDate DATE NOT NULL,

Status ENUM('Present', 'Absent') NOT NULL,

FOREIGN KEY (StudentID) REFERENCES Students(StudentID),

FOREIGN KEY (CourseID) REFERENCES Courses(CourseID)

);

CREATE TABLE Grades (

GradeID INT AUTO\_INCREMENT PRIMARY KEY,

StudentID INT NOT NULL,

CourseID INT NOT NULL,

Score INT NOT NULL,

FOREIGN KEY (StudentID) REFERENCES Students(StudentID),

FOREIGN KEY (CourseID) REFERENCES Courses(CourseID)

);

-- Insert Students

INSERT INTO Students (FirstName, LastName, EnrollmentDate) VALUES

('Muhammad', 'Yawar', '2023-09-01'),

('Sheikh', 'Abdullah', '2023-09-01'),

('Muhammad', 'Awais', '2023-09-01');

-- Insert Courses

INSERT INTO Courses (CourseName, InstructorName) VALUES

('Mathematics', 'Dr. Bilal'),

('Physics', 'Dr. Haider');

-- Insert Enrollments for all students in all courses

INSERT INTO Enrollments (StudentID, CourseID, EnrollmentDate) VALUES

(1, 1, '2023-09-02'), -- Muhammad Yawar's enrollment in Mathematics

(1, 2, '2023-09-02'), -- Muhammad Yawar's enrollment in Physics

(2, 1, '2023-09-02'), -- Sheikh Abdullah's enrollment in Mathematics

(2, 2, '2023-09-02'), -- Sheikh Abdullah's enrollment in Physics

(3, 1, '2023-09-02'), -- Muhammad Awais's enrollment in Mathematics

(3, 2, '2023-09-02'); -- Muhammad Awais's enrollment in Physics

-- Insert Attendance Records for Multiple Dates

INSERT INTO Attendance (StudentID, CourseID, AttendanceDate, Status) VALUES

-- Mathematics Course (CourseID 1)

(1, 1, '2023-09-03', 'Absent'),

(2, 1, '2023-09-03', 'Present'),

(3, 1, '2023-09-03', 'Present'),

(1, 1, '2023-09-04', 'Present'),

(2, 1, '2023-09-04', 'Present'),

(3, 1, '2023-09-04', 'Absent'),

(1, 1, '2023-09-05', 'Present'),

(2, 1, '2023-09-05', 'Present'),

(3, 1, '2023-09-05', 'Present'),

-- Physics Course (CourseID 2)

(1, 2, '2023-09-03', 'Present'),

(2, 2, '2023-09-03', 'Absent'),

(3, 2, '2023-09-03', 'Present'),

(1, 2, '2023-09-04', 'Present'),

(2, 2, '2023-09-04', 'Present'),

(3, 2, '2023-09-04', 'Present'),

(1, 2, '2023-09-05', 'Absent'),

(2, 2, '2023-09-05', 'Present'),

(3, 2, '2023-09-05', 'Absent');

-- Insert Grades for all students in all courses

INSERT INTO Grades (StudentID, CourseID, Score) VALUES

(1, 1, 85), -- Muhammad Yawar's grade in Mathematics

(2, 1, 78), -- Sheikh Abdullah's grade in Mathematics

(3, 1, 82), -- Muhammad Awais's grade in Mathematics

(1, 2, 92), -- Muhammad Yawar's grade in Physics

(2, 2, 75), -- Sheikh Abdullah's grade in Physics

(3, 2, 88); -- Muhammad Awais's grade in Physics

-- Query to show attendance report for all students

SELECT

s.FirstName,

s.LastName,

c.CourseName,

a.AttendanceDate,

a.Status

FROM Attendance a, Students s, Courses c

WHERE a.StudentID = s.StudentID

AND a.CourseID = c.CourseID;

-- Query to calculate average scores per course

SELECT

c.CourseName,

AVG(g.Score) AS AverageScore

FROM Grades g, Courses c

WHERE g.CourseID = c.CourseID

GROUP BY c.CourseID;

-- Query to show top performers in each course

SELECT

c.CourseName,

s.FirstName,

s.LastName,

g.Score

FROM Grades g, Students s, Courses c

WHERE g.StudentID = s.StudentID

AND g.CourseID = c.CourseID

AND g.Score = (

SELECT MAX(Score)

FROM Grades

WHERE CourseID = g.CourseID

);

-- Query to calculate the average score of each student across all their courses

SELECT

s.FirstName,

s.LastName,

AVG(g.Score) AS StdPercentage

FROM Grades g, Students s

WHERE g.StudentID = s.StudentID

GROUP BY s.StudentID;

-- Query to calculate the attendance percentage for each student in each course

SELECT

s.FirstName,

s.LastName,

c.CourseName,

AVG(CASE WHEN a.Status = 'Present' THEN 1 ELSE 0 END) \* 100 AS AttendancePercentage

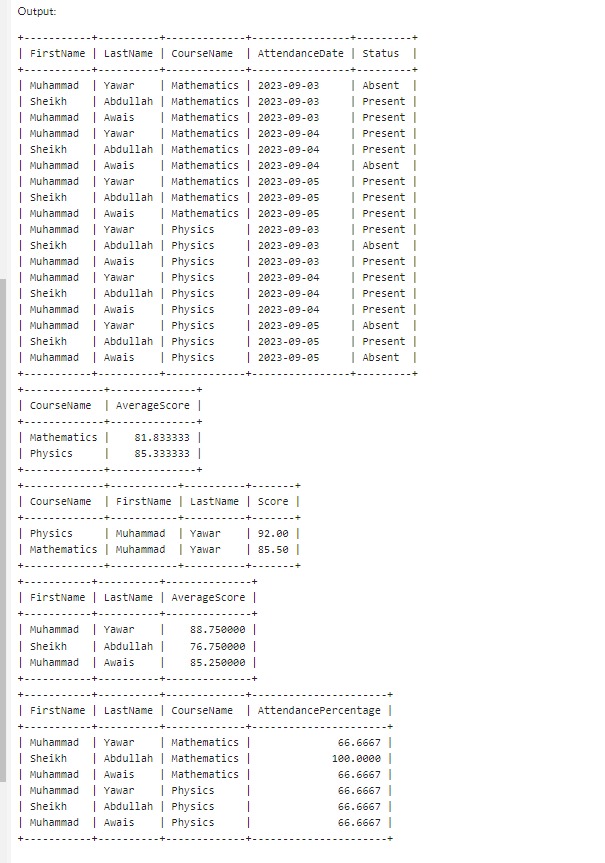
FROM Attendance a, Students s, Courses c

WHERE a.StudentID = s.StudentID

AND a.CourseID = c.CourseID

GROUP BY s.StudentID, c.CourseID;

**OUTPUT:**



**ENTITY RELATION DIAGRAM (ERD):**

