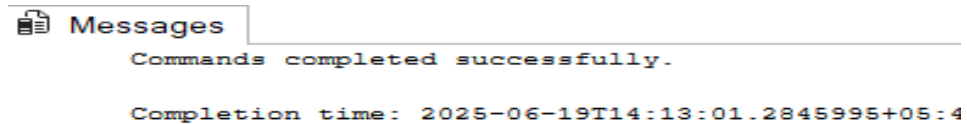


Lab 2: Use of Multiple Tables

Create Database :

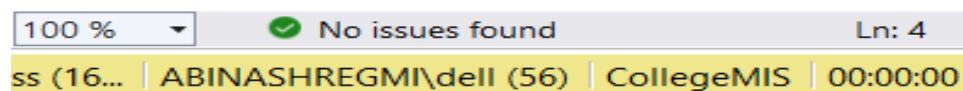
Query: CREATE DATABASE CollegeMIS;
USE CollegeMIS;



Messages

Commands completed successfully.

Completion time: 2025-06-19T14:13:01.2845995+05:4



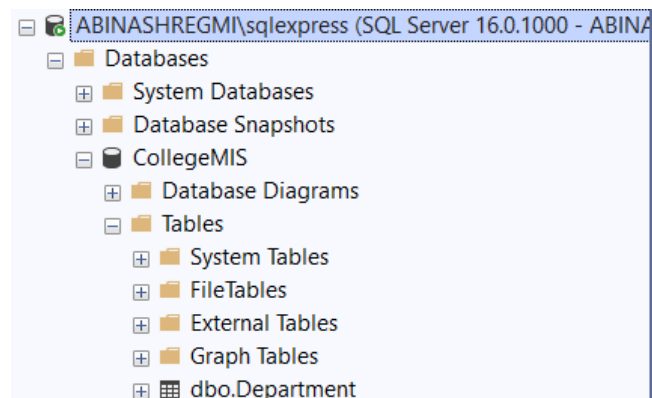
100 % No issues found Ln: 4

ss (16... | ABINASHREGMI\dell (56) | CollegeMIS | 00:00:00

Creating Table:

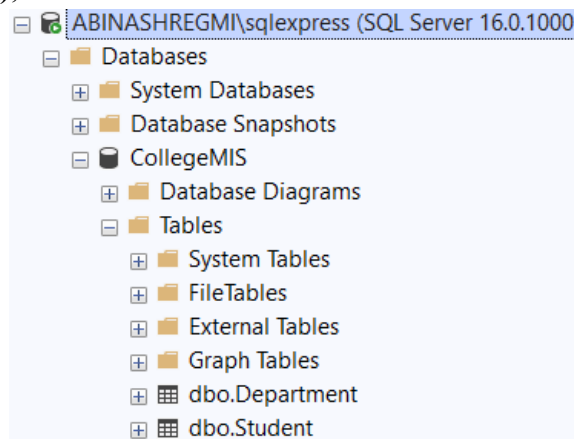
Create Department table:

Query: CREATE TABLE Department (
DepartmentID INT PRIMARY KEY,
DeptName VARCHAR(50)
);



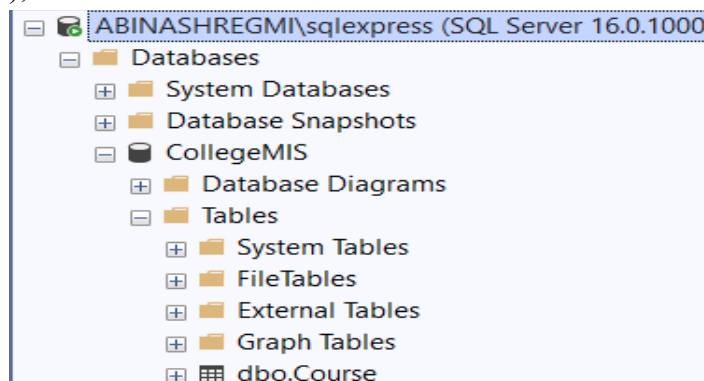
Create Student table:

Query : CREATE TABLE Student (
 StudentID INT PRIMARY KEY,
 Name VARCHAR(50),
 Gender CHAR(1),
 DOB DATE,
 DepartmentID INT,
 Email VARCHAR(100),
 FOREIGN KEY (DepartmentID) REFERENCES Department(DepartmentID)
);



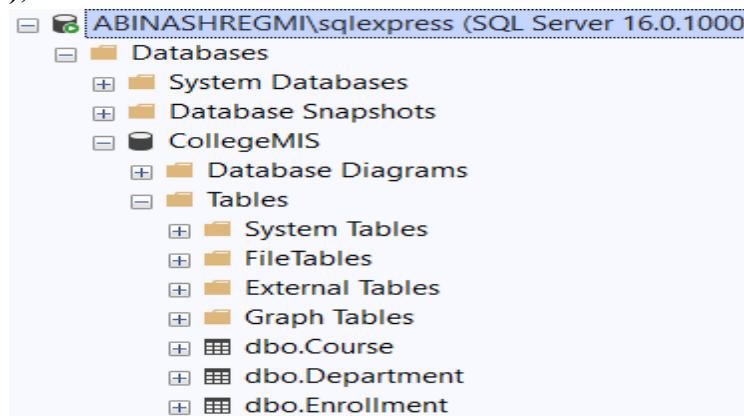
Create Course table:

Query : CREATE TABLE Course (
 CourseID INT PRIMARY KEY,
 CourseName VARCHAR(100),
 DepartmentID INT,
 FOREIGN KEY (DepartmentID) REFERENCES
Department(DepartmentID)
);



Create Enrollment table:

Query : CREATE TABLE Enrollment (
EnrollmentID INT PRIMARY KEY,
StudentID INT,
CourseID INT,
Semester VARCHAR(10),
Marks INT,
FOREIGN KEY (StudentID) REFERENCES Student(StudentID),
FOREIGN KEY (CourseID) REFERENCES Course(CourseID)
);



Insert sample data into all four tables (Student, Department, Course, Enrollment).

Departments:

Query : INSERT INTO Department(DepartmentID,DeptName) VALUES
(1,'Commerce'),
(2, 'Computer Technology'),
(3, 'Business Adminstration'),
(4, 'Electronics');

	DepartmentID	DeptName
1	1	Commerce
2	2	Computer Technology
3	3	Business Adminstration
4	4	Electronics


Students :

Query : INSERT INTO Student (StudentID, Name, Gender, DOB, DepartmentID, Email)

VALUES

(201, 'Aayush Bhattarai', 'M', '2003-05-01', '1', 'notaayush@gmail.com'),
(202, 'Albina Tamang', 'F', '2005-04-03', '2', 'albina@yahoo.com'),
(203, 'Chirag Sharma', 'M', '2004-04-09', '1', 'chirag@outlook.com'),
(204, 'Dikshya Gurung', 'F', '2008-04-29', '3', 'dikshyagurung@gmail.com'),
(205, 'Sujal Bajracharya', 'M', '2003-04-04', '2', 'bajracharyasujal@hotmail.com'),
(206, 'Sofiya Lamichhane', 'F', '2004-04-04', '1', 'lamichhansofiya@gmail.com'),
(207, 'Yalin Baidar', 'M', '2003-04-03', '2', 'Baidar123@gmail.com');

	StudentID	Name	Gender	DOB	DepartmentID	Email
1	201	Aayush Bhattarai	M	2003-05-01	1	notaayush@gmail.com
2	202	Albina Tamang	F	2005-04-03	2	albina@yahoo.com
3	203	Chirag Sharma	M	2004-04-09	1	chirag@outlook.com
4	204	Dikshya Gurung	F	2008-04-29	3	dikshyagurung@gmail.com
5	205	Sujal Bajracharya	M	2003-04-04	2	bajracharyasujal@hotmail.com
6	206	Sofiya Lamichhane	F	2004-04-04	1	lamichhansofiya@gmail.com
7	207	Yalin Baidar	M	2003-04-03	2	Baidar123@gmail.com

 ABINASHREGMI\sqlexpress (16... | ABINASHREGMI\dell (56) | CollegeMIS | 00:00:00 | 7 rows

Courses :

Query : INSERT INTO Course (CourseID, CourseName, DepartmentID) VALUES

(201, 'Data Structures', 1),
(202, 'Operating Systems', 1),
(203, 'Database Systems', 2),
(204, 'Business Management', 3),
(205, 'Computer Networks', 2),
(206, 'DBMS', 2),
(207, 'Electronics I', 4),
(208, 'Microprocessor', 4);

	CourseID	CourseName	DepartmentID
1	201	Data Structures	1
2	202	Operating Systems	1
3	203	Database Systems	2
4	204	Business Management	3
5	205	Computer Networks	2
6	206	DBMS	2
7	207	Electronics I	4
8	208	Microprocessor	4

ABINASHREGMI\sqlexpress (16... | ABINASHREGM

Enrollments:

Query : INSERT INTO Enrollment (EnrollmentID, StudentID, CourseID, Semester,

Marks) VALUES

(301, 204, 201, 'Semester3', 88),

(302, 201, 202, 'Semester3', 91),

(303, 202, 203, 'Semester3', 85),

(304, 203, 201, 'Semester3', 92),

(305, 203, 204, 'Semester3', 78),

(306, 202, 205, 'Semester4', 82),

(307, 204, 206, 'Semester4', 89),

(308, 206, 206, 'Semester4', 95),

(309, 202, 205, 'Semester4', 73),

(310, 207, 205, 'Semester4', 90);

Results Messages

	ENrollmentID	StudentID	CourseID	Semester	Marks
1	301	204	201	Semester3	88
2	302	201	202	Semester3	91
3	303	202	203	Semester3	85
4	304	203	201	Semester3	92
5	305	203	204	Semester3	78
6	306	202	205	Semester4	82
7	307	204	206	Semester4	89
8	308	206	206	Semester4	95
9	309	202	205	Semester4	73
10	310	207	205	Semester4	90

ABINASHREGMI\sqlexpress (16... | ABINASHREGMI\de11 (69)

Display all students with their department name using JOIN.

Query : SELECT s.StudentID, s.Name, d.DeptName FROM Student s
JOIN Department d ON s.DepartmentID = d.DepartmentID;

	StudentId	Name	DeptName
1	201	Aayush Bhattarai	Commerce
2	202	Albina Tamang	Computer Technology
3	203	Chirag Sharma	Commerce
4	204	Dikshya Gurung	Business Adminstration
5	205	Sujal Bajracharya	Computer Technology
6	206	Sofiya Lamichhane	Commerce
7	207	Yalin Baidar	Computer Technology

ABINASHREGMI\sqlexpress (16... | ABINASHREGMI\d

Show all courses offered by the "Computer Technology" department.

Query : SELECT c.CourseName FROM Course c
JOIN Department d ON c.DepartmentID = d.DepartmentID
WHERE d.DeptName = 'Computer Technology';

	CourseName
1	Database Systems
2	Computer Networks
3	DBMS

List all students enrolled in "Data Structures" course.

Query : SELECT s.Name FROM Student s
JOIN Enrollment e ON s.StudentID = e.StudentID
JOIN Course c ON e.CourseID = c.CourseID
WHERE c.CourseName = 'Data Structures';

	Name
1	Dikshya Gurung
2	Chirag Sharma

Count the number of students in each department.

Query : **SELECT d.DeptName, COUNT(s.StudentID) AS StudentCount FROM Department d
LEFT JOIN Student s ON d.DepartmentID = s.DepartmentID
GROUP BY d.DeptName;**

	DeptName	StudentCount
1	Business Administration	1
2	Commerce	3
3	Computer Technology	3
4	Electronics	0

List students who scored more than 80 in any course.

Query : **SELECT DISTINCT s.Name FROM Student s
JOIN Enrollment e ON s.StudentID = e.StudentID
WHERE e.Marks > 80;**

	Name
1	Aayush Bhattarai
2	Albina Tamang
3	Chirag Sharma
4	Dikshya Gurung
5	Sofiya Lamichhane
6	Yalin Baidar

Find the average marks for each course using GROUP BY.

Query : **SELECT c.CourseName, AVG(e.Marks) AS AverageMarks FROM Course c
JOIN Enrollment e ON c.CourseID = e.CourseID
GROUP BY c.CourseName;**

	CourseName	AverageMarks
1	Business Management	78
2	Computer Networks	81
3	Data Structures	90
4	Database Systems	85
5	DBMS	92
6	Operating Systems	91

Display students who have not enrolled in any course.

Query : **SELECT s.Name FROM Student s
LEFT JOIN Enrollment e ON s.StudentID = e.StudentID
WHERE e.EnrollmentID IS NULL;**

	Name
1	Sujal Bajracharya

List all courses with the number of students enrolled in them.

Query : **SELECT c.CourseName, COUNT(e.StudentID) AS EnrolledStudents FROM
Course c
LEFT JOIN Enrollment e ON c.CourseID = e.CourseID
GROUP BY c.CourseName;**

	CourseName	EnrolledStudents
1	Business Management	1
2	Computer Networks	3
3	Data Structures	2
4	Database Systems	1
5	DBMS	2
6	Electronics I	0
7	Microprocessor	0
8	Operating Systems	1

Find the student with the highest marks in "DBMS" course.

Query : **SELECT TOP 1 s.Name, e.Marks FROM Student s
JOIN Enrollment e ON s.StudentID = e.StudentID
JOIN Course c ON e.CourseID = c.CourseID
WHERE c.CourseName = 'DBMS'
ORDER BY e.Marks DESC;**

	Name	Marks
1	Sofiya Lamichhane	95

Update a student's department based on StudentID.

Query : **UPDATE Student
SET DepartmentID = 3
WHERE StudentID = 109;**

After :

	StudentID	Name	Gender	DOB	DepartmentID	Email
1	201	Aayush Bhattarai	M	2003-05-01	1	notaayush@gmail.com
2	202	Albina Tamang	F	2005-04-03	2	albina@yahoo.com
3	203	Chirag Sharma	M	2004-04-09	1	chirag@outlook.com
4	204	Dikshya Gurung	F	2008-04-29	3	dikshyagurung@gmail.com
5	205	Sujal Bajracharya	M	2003-04-04	2	bajracharyasujal@hotmail.com
6	206	Sofiya Lamichhane	F	2004-04-04	1	lamichhansofiya@gmail.com
7	207	Yalin Baidar	M	2003-04-03	2	Baidar123@gmail.com

ABINASHREGMI\sql express (16... | ABINASHREGMI\deII (69) | CollegeMIS | 00:00:00 | 7 rows

Delete a course from the course list and ensure cascade deletion in enrollment.

Query : DELETE FROM Course

WHERE CourseName = 'Electronics I';

Before:

	CourseName
1	Data Structures
2	Operating Systems
3	Database Systems
4	Business Management
5	Computer Networks
6	DBMS
7	Electronics I
8	Microprocessor

After:

	CourseName
1	Data Structures
2	Operating Systems
3	Database Systems
4	Business Management
5	Computer Networks
6	DBMS
7	Microprocessor

-Abinash Regmi

Find the departments that offer more than 3 courses.

**Query : SELECT d.DeptName FROM Department d
JOIN Course c ON d.DepartmentID = c.DepartmentID
GROUP BY d.DeptName
HAVING COUNT(c.CourseID) > 3;**

| DeptName |

ABINASHREGMI\sqlexpress (16

**Show the names of students along with total credits they are enrolled in
(assuming each course = 3 credits).**

**Query : SELECT s.Name, COUNT(e.CourseID) * 3 AS TotalCredits FROM Student s
JOIN Enrollment e ON s.StudentID = e.StudentID
GROUP BY s.Name;**

	Name	TotalCredits
1	Aayush Bhattarai	3
2	Albina Tamang	9
3	Chirag Sharma	6
4	Dikshya Gurung	6
5	Sofiya Lamichhane	3
6	Yalin Baidar	3

Create a view TopPerformers that shows students with average marks > 85.

Query : GO

```
CREATE VIEW TopPerformers AS
SELECT s.StudentID, s.Name, AVG(e.Marks) AS AvgMarks
FROM Student s
JOIN Enrollment e ON s.StudentID = e.StudentID
GROUP BY s.StudentID, s.Name
HAVING AVG(e.Marks) > 85;
GO
```

Data in the view TopPerformers :

	StudentID	Name	AvgMarks
1	201	Aayush Bhattarai	91
2	204	Dikshya Gurung	88
3	206	Sofiya Lamichhane	95
4	207	Yalin Baidar	90

Create a view StudentDetails to show StudentID, Name, Department, and Email.

Query : GO

```
CREATE VIEW StudentDetails AS
SELECT s.StudentID, s.Name, d.DeptName AS Department, s.Email FROM
Student s
JOIN Department d ON s.DepartmentID = d.DepartmentID;
GO
```

Data in the view StudentDetails :

	StudentID	Name	Department	Email
1	201	Aayush Bhattarai	Commerce	notaayush@gmail.com
2	202	Albina Tamang	Computer Technology	albina@yahoo.com
3	203	Chirag Sharma	Commerce	chirag@outlook.com
4	204	Dikshya Gurung	Business Administration	dikshyagurung@gmail.com
5	205	Sujal Bajracharya	Computer Technology	bajracharyasujal@hotmail.com
6	206	Sofiya Lamichhane	Commerce	lamichhansofiya@gmail.com
7	207	Yalin Baidar	Computer Technology	Baidar123@gmail.com

Write a query to show student name, course name, and marks for all enrollments.

Query : **SELECT s.Name AS StudentName, c.CourseName, e.Marks FROM Enrollment e**

JOIN Student s ON e.StudentID = s.StudentID

JOIN Course c ON e.CourseID = c.CourseID;

	StudentName	CourseName	Marks
1	Dikshya Gurung	Data Structures	88
2	Aayush Bhattarai	Operating Systems	91
3	Albina Tamang	Database Systems	85
4	Chirag Sharma	Data Structures	92
5	Chirag Sharma	Business Management	78
6	Albina Tamang	Computer Networks	82
7	Dikshya Gurung	DBMS	89
8	Sofiya Lamichhane	DBMS	95
9	Albina Tamang	Computer Networks	73
10	Yalin Baidar	Computer Networks	90

ABINASHREGMI\sqlexpress (16... | ABINASHREGMI\de11 (69)

Find students who have enrolled in courses from more than one department.

Query : **SELECT s.Name FROM Student s**

JOIN Enrollment e ON s.StudentID = e.StudentID

JOIN Course c ON e.CourseID = c.CourseID

GROUP BY s.StudentID, s.Name

HAVING COUNT(DISTINCT c.DepartmentID) > 1;

	Name
1	Chirag Sharma
2	Dikshya Gurung

Display the list of departments that currently have no students.

Query : **SELECT d.DeptName FROM Department d
LEFT JOIN Student s ON d.DepartmentID = s.DepartmentID
WHERE s.StudentID IS NULL;**

	DeptName
1	Electronics

Rename the column DOB in Student to DateOfBirth.

Query : **EXEC sp_rename 'Student.DOB', 'DateOfBirth', 'COLUMN';**

	StudentID	Name	Gender	DateOfBirth	DepartmentID	Email
1	201	Aayush Bhattarai	M	2003-05-01	1	notaayush@gmail.com
2	202	Albina Tamang	F	2005-04-03	2	albina@yahoo.com
3	203	Chirag Sharma	M	2004-04-09	1	chirag@outlook.com
4	204	Dikshya Gurung	F	2008-04-29	3	dikshyagurung@gmail.com
5	205	Sujal Bajracharya	M	2003-04-04	2	bajracharyasujal@hotmail.com
6	206	Sofiya Lamichhane	F	2004-04-04	1	lamichhansofiya@gmail.com
7	207	Yalin Baidar	M	2003-04-03	2	Baidar123@gmail.com

Find students who are both in the "CT" department and enrolled in "Computer Networks"

Query : **SELECT s.Name FROM Student s
JOIN Department d ON s.DepartmentID = d.DepartmentID
JOIN Enrollment e ON s.StudentID = e.StudentID
JOIN Course c ON e.CourseID = c.CourseID
WHERE d.DeptName = 'Computer Technology' AND c.CourseName =
'Computer Networks';**

	Name
1	Albina Tamang
2	Albina Tamang
3	Yalin Baidar

Find students whose email ends with '@gmail.com'

**Query : SELECT Name FROM Student
WHERE Email LIKE '%@gmail.com';**

	Name
1	Aayush Bhattarai
2	Dikshya Gurung
3	Sofiya Lamichhane
4	Yalin Baidar

Display students who scored more than the average mark of all students.

**Query : SELECT s.Name, e.Marks FROM Student s
JOIN Enrollment e ON s.StudentID = e.StudentID
WHERE e.Marks > (SELECT AVG(Marks) FROM Enrollment);**

	Name	Marks
1	Dikshya Gurung	88
2	Aayush Bhattarai	91
3	Chirag Sharma	92
4	Dikshya Gurung	89
5	Sofiya Lamichhane	95
6	Yalin Baidar	90

Show the average marks of students in each course.

**Query : SELECT c.CourseName, AVG(e.Marks) AS AvgMarks FROM Course c
JOIN Enrollment e ON c.CourseID = e.CourseID
GROUP BY c.CourseName;**

	CourseName	AvgMarks
1	Business Management	78
2	Computer Networks	81
3	Data Structures	90
4	Database Systems	85
5	DBMS	92
6	Operating Systems	91

Show all students with course name and sort by Marks descending.

Query : **SELECT s.Name AS StudentName, c.CourseName, e.Marks FROM Student s
JOIN Enrollment e ON s.StudentID = e.StudentID
JOIN Course c ON e.CourseID = c.CourseID
ORDER BY e.Marks DESC;**

	StudentName	CourseName	Marks
1	Sofiya Lamichhane	DBMS	95
2	Chirag Sharma	Data Structures	92
3	Aayush Bhattarai	Operating Systems	91
4	Yalin Baidar	Computer Networks	90
5	Dikshya Gurung	DBMS	89
6	Dikshya Gurung	Data Structures	88
7	Albina Tamang	Database Systems	85
8	Albina Tamang	Computer Networks	82
9	Chirag Sharma	Business Management	78
10	Albina Tamang	Computer Networks	73

ABINASHREGMI\sqlexpress (16... | ABINASHREGMI\dell (69)