

REPORT

SUBSYSTEM STUDENT GRADING MANAGEMENT

THÁNG 7/ 2022

Hệ thống con quản lý điểm của học sinh, sinh viên – Assignment of DBI202

Tác giả: Nguyễn Nhật Nam – IA1604



Lời nói đầu

Hệ thống con quản lí điểm của học sinh, sinh viên

Hiện nay, xã hội ngày càng phát triển, nhận thức của con người về thế giới xung quanh không ngừng nâng cao. Chúng ta không chỉ đơn thuần chỉ là quan sát một cách khách quan nữa, mà đã khám phá, tìm hiểu một cách triệt để. Điều này đem lại những kiến thức mới mà ta chưa từng biết trước đó.

Tuy vậy, việc đem lại kiến thức mới cũng mang tới ý nghĩa chúng ta cần tới việc lưu trữ và khai thác điều đó một cách triệt để. Vấn đề là việc lưu trữ thông thường đã không thể đáp ứng được việc chúng ta lưu trữ dữ liệu thu thập được một cách đầy đủ và hiệu quả. Vậy con người đã làm gì để cải thiện vấn đề này?

Sau hàng trăm năm phát triển, chúng ta đã tạo ra công nghệ mới mang tên là Máy tính điện tử - nó đã giúp chúng ta lưu trữ và khai thác dữ liệu trở nên dễ dàng hơn. Và để máy tính hoạt động trơn tru cũng như tối đa hóa trong quá trình khai phá và xử lí dữ liệu, Cơ sở dữ liệu và Hệ quản trị cơ sở dữ liệu đã ra đời.

Cơ sở dữ liệu là nền tảng được áp dụng trong hầu hết các ngành trong đất nước ta nói riêng và thế giới nói chung. Chúng ta không thể không kể đến một ngành vô cùng quan trọng đối với con người – Ngành Giáo dục. Việc quản lí hàng triệu, hàng chục triệu học sinh, sinh viên trên khắp cả nước là một điều rất khó khăn nếu áp dụng bằng phương pháp truyền thống. Nhưng với sử dụng Cơ sở dữ liệu, việc này trở nên đơn giản hơn rất nhiều.

Và hôm nay, em xin được đề cập cũng như là có những nhận định chi tiết qua một trong những phần mềm quản lí học sinh, sinh viên: Hệ thống con quản lí điểm học sinh, sinh viên.

MỤC LỤC

- I. Giới thiệu chung về Hệ thống con quản lí điểm của học sinh, sinh viên
- II. Phân tích các bảng đã được thiết kế sẵn
 - 1. Bảng đánh giá điểm từng môn
 - 2. Bảng kết quả học tập
 - 3. Bảng điểm theo môn học
- III. Những giả thiết về thực thể cần có
 - 1. Giả thiết về các thực thể
 - 2. Các attributes xuất hiện trong các thử thể giả định
- IV. Chuẩn hóa dữ liệu dạng 3NF
 - 1. Xác định và chuyển đổi các thực thể
 - 1.1. Xác định mối quan hệ giữa các thực thể
 - 1.2. Chuyển đổi các mối quan hệ
 - 1.3. Bảng ERD tổng quan
 - 2. Xác định các attributes cho các thực thể chuyển đổi
- V. Tạo bảng dữ liệu
- VI. Thêm dữ liệu
- VII. Các câu lệnh Query
- VIII. Tổng kết

I. Giới thiệu chung về Hệ thống con quản lý điểm của học sinh, sinh viên

Hệ thống con gồm có 3 bảng được cung cấp sẵn dữ liệu:

- Đối với mỗi môn học mà sinh viên đã tham dự, giảng viên sẽ cho điểm đánh giá cho mỗi bài đánh giá của họ. Đây là ví dụ về môn DBI202:

Category	Type	Part	Weight	Completion Criteria	Duration	L.O.	Question Type	No. Question	Knowledge and Skill	Grading Guide	Note
Progress Tests	quiz	2	10.0%	>0	20'		Multiple choices Marked by Computer or a suitable format	20	up to 04 covered chapters	by instructor using computer	Instruction and schedules for Progress tests must be presented in the Course Implementation Plan approved by director of the campus. Progress test must be taken right after the last lectures of required material. Instructor has responsibility to review the test for students after graded.
Assignment	on-going	1	20.0%	>0	at home		Design; Implementation; Presentation		Simple RDBS design and implementation using a DBMS	guided by instructor, prepare at home present in class	40% Design, 20% Implementation, 40% Presentation of the whole Project
Labs	on-going	5	15.0%	>0	in lab session		practical exercises		related to studied modules	Guided by instructor	may be continued at home.
Practical Exam	practical exam	1	25.0%	>0	85'		Preferable to be marked by Scripts		DB programming skills	by exam board and department	Practical Exam database is up load in CMS in advanced.
Final Exam	final exam	1	30.0%	5	60'		Multiple choices Marked by Computer	60	Knowledge and skills in the course, but with much focus on the items in Chapters 2 to 6, >= 70% new questions (for the current semester).	by exam board	

- Sinh viên có thể kiểm tra kết quả của mình vào cuối học kỳ như sau:

NO.	SUBJECT CODE	SUBJECT NAME	SEMESTER	GROUP	STARTDATE	ENDDATE	AVERAGE MARK	STATUS
1	SSL101c	Academic Skills for University Success	Spring2021					Not Passed
2	SSG103	Communication and In-Group Working Skills	Summer2021					Passed
3	NWC203c	Computer Networking	Summer2021					Passed
4	CEA201	Computer Organization and Architecture	Spring2021					Passed
5	MAD101	Discrete mathematics	Summer2021					Passed
6	JPD113	Elementary Japanese 1-A1.1	Fall2021					Passed
7	CSI104	Introduction to Computer Science	Spring2021					Passed
8	DBI202	Introduction to Databases	Fall2021					Not Passed
9	LUK1	Level 1	Fall2019					Passed
10	LUK2	Level 2	Spring2020					Passed
11	LUK3	Level 3	Spring2020					Passed
12	LUK4	Level 4	Summer2020					Pass (with conditions)
13	LUK5	Level 5	Summer2020					Passed
14	LUK6	Level 6	Fall2020					Passed
15	MAE101	Mathematics for Engineering	Spring2021					Passed

- Mỗi mã môn học, sinh viên có thể kiểm tra kết quả chi tiết của mình như ví dụ dưới đây:

GRADE CATEGORY	GRADE ITEM	WEIGHT	VALUE	COMMENT
Quiz 2	Quiz 2	7.0 %	7.8	
	Total	7.0 %	7.8	
Quiz 1	Quiz 1	8.0 %	7.6	
	Total	8.0 %	7.6	
Activity	Activity	10.0 %	8.5	
	Total	10.0 %	8.5	
Group Assignment	Group Assignment	15.0 %	9	
	Total	15.0 %	9	
Group Project	Group Project	30.0 %	8.3	
	Total	30.0 %	8.3	
Final Exam	Final Exam	30.0 %	8.6	
	Total	30.0 %	8.6	
Final Exam Resit	Final Exam Resit	30.0 %		
	Total	30.0 %		
COURSE TOTAL		AVERAGE	8.4	
		STATUS	PASSED	

II. Phân tích các bảng đã được thiết kế sẵn

1. BẢNG VỀ ĐÁNH GIÁ ĐIỂM CỦA CÁC MÔN HỌC

Assessment(s)											
Category	Type	Part	Weight	Completion Criteria	Duration	LO	Question Type	No. Question	Knowledge and Skill	Grading Guide	Note
Progress Tests	quiz	2	10.0%	>0	20'		Multiple choices Marked by Computer or a suitable format	20	up to 04 covered chapters	by instructor using computer	Instruction and schedules for Progress tests must be presented in the Course Implementation Plan approved by director of the campus.
											Progress test must be taken right after the last lectures of required material.
											Instructor has responsibility to review the test for students after graded.
Assignment	on-going	1	20.0%	>0	at home		Design; Implementation; Presentation		Simple RDBS design and implementation using a DBMS	guided by instructor; prepare at home present in class	40% Design, 20% Implementation, 40% Presentation of the whole Project
Labs	on-going	5	15.0%	>0	in lab session		practical exercises		related to studied modules	Guided by instructor	may be continued at home.
Practical Exam	practical exam	1	25.0%	>0	85'		Preferable to be marked by Scripts		DB programming skills	by exam board and department	Practical Exam database is up load in CMS in advanced.
Final Exam	final exam	1	30.0%	5	60'		Multiple choices Marked by Computer	60	Knowledge and skills in the course, but with much focus on the items in Chapters 2 to 6, >= 70% new questions (for the current semester).	by exam board	

- Đánh giá sơ bộ: Các dữ liệu vô cùng đầy đủ, các yêu cầu và ghi chú về các thành phần điểm được xác định một cách chi tiết
- Phân tích thành phần bảng:
 - + Category: Các điểm thành phần
 - + Type: Kiểu kiểm tra
 - + Part: Số bài cần thiết
 - + Weight: Trọng số của từng điểm thành phần
 - + Completion Criteria: Yêu cầu để hoàn thành môn
 - + Duration: Thời gian hoàn thành các điểm thành phần
 - + LO:
 - + Question Type: Câu hỏi kiểm tra
 - + No Question: Số lượng câu hỏi
 - + Knowledge and Skill: Kiến thức và kỹ năng sẽ đạt được
 - + Grade Guide: Hướng dẫn lấy điểm
 - + Note: Ghi chú cần thiết
- Phân tích sơ bộ:
 - + Cấu thành của bảng gồm nhiều Category(Thành phần điểm) khác nhau cấu thành.

- + Đa phần Weight sẽ phân bố trong khoảng dưới 50%(tùy theo từng môn).
- + Các điểm thành phần đều có yêu cầu trên 0, riêng điểm Final Exam phải đạt mốc trên 4.

➔ Kết luận: Category nắm vai trò quan trọng quyết định đến bảng đánh giá điểm.

2. BẢNG KẾT QUẢ MÔN HỌC.

NO.	SUBJECT CODE	SUBJECT NAME	SEMESTER	GROUP	STARTDATE	ENDDATE	AVERAGE MARK	STATUS
1	SSL101c	Academic Skills for University Success	Spring2021					Not Passed
2	SSG103	Communication and In-Group Working Skills	Summer2021					Passed
3	NWC203c	Computer Networking	Summer2021					Passed
4	CEA201	Computer Organization and Architecture	Spring2021					Passed
5	MAD101	Discrete mathematics	Summer2021					Passed
6	JPD113	Elementary Japanese 1-A1.1	Fall2021					Passed
7	CSI104	Introduction to Computer Science	Spring2021					Passed
8	DBI202	Introduction to Databases	Fall2021					Not Passed
9	LUK1	Level 1	Fall2019					Passed
10	LUK2	Level 2	Spring2020					Passed
11	LUK3	Level 3	Spring2020					Passed
12	LUK4	Level 4	Summer2020					Pass (with conditions)
13	LUK5	Level 5	Summer2020					Passed
14	LUK6	Level 6	Fall2020					Passed
15	MAE101	Mathematics for Engineering	Spring2021					Passed

- Đánh giá sơ bộ: Bảng điểm gồm chi tiết điểm của môn trong từng kì và tình trạng môn học.
- Phân tích thành phần bảng:
 - + Subject code: Mã môn học
 - + Subject name: Tên môn học
 - + Semester: Kì học

- + Group: Nhóm học tham gia
- + StartDate: Thời gian bắt đầu môn học
- + EndDate: Thời gian kết thúc môn học
- + Average Mark: Điểm trung bình môn học
- + Status: Tình trạng môn học

- Phân tích sơ bộ:

- + Có nhiều môn học sẽ tham gia vào một kì, thời gian môn học tương đương với một kì.
- + Một Group có thể tham gia vào nhiều môn học trong cùng 1 kì, hoặc tham gia nhiều kì, nhiều môn học.
- + Status phụ thuộc vào Average Mark.

➔ Kết luận: Sơ bộ có nhiều hơn 2 thực thể được xác định. Điều này gây ra sự rối loạn dữ liệu nếu như chúng cùng chung 1 bảng -> cần sử dụng các chuẩn NF để tách ra giúp dễ dàng khai thác.

3. BẢNG ĐIỂM TỪNG MÔN

GRADE CATEGORY	GRADE ITEM	WEIGHT	VALUE	COMMENT
Quiz 2	Quiz 2	7.0 %	7.8	
	Total	7.0 %	7.8	
Quiz 1	Quiz 1	8.0 %	7.6	
	Total	8.0 %	7.6	
Activity	Activity	10.0 %	8.5	
	Total	10.0 %	8.5	
Group Assignment	Group Assignment	15.0 %	9	
	Total	15.0 %	9	
Group Project	Group Project	30.0 %	8.3	
	Total	30.0 %	8.3	
Final Exam	Final Exam	30.0 %	8.6	
	Total	30.0 %	8.6	
Final Exam Resit	Final Exam Resit	30.0 %		
	Total	30.0 %		
COURSE TOTAL	AVERAGE	8.4		
	STATUS	PASSED		

-
- Đánh giá sơ bộ: Bảng thể hiện mối quan hệ giữa các điểm thành phần với điểm trung bình và tình trạng của môn học
 - Phân tích thành phần bảng:
 - + Grade Category: Các điểm thành phần
 - + Grade Item: Hạng mục
 - + Weight: Trọng số
 - + Value: Điểm thành phần
 - + Comment: Bình luận/ Lưu ý
 - + Average: Điểm trung bình
 - + Status: Trạng thái môn học
 - Phân tích sơ bộ:
 - + Cách tính điểm trung bình dựa trên tổng của Value * Weight.
 - + Status phụ thuộc vào Average
- ➔ Kết luận: có sự tham gia của Category trong cả hai bảng Đánh giá điểm và Bảng điểm từng môn -> thực thể Category có thể xác định là 1 trong các thực thể cần thiết sẽ có.

III. Giả thiết về những thực thể cần có.

1. GIẢ THIẾT VỀ CÁC THỰC THỂ.

- **Student:** Đây là thực thể đầu tiên có thể xác định ngay khi nhìn vào tên đầu đề hệ thống. Điều này có thể xác định thông qua các bảng đã được cung cấp. Các thành phần trong bảng đã đề cập trực tiếp hoặc gián tiếp tới **Student**.
- **Lecture:** Tất cả các bảng điểm của học sinh, sinh viên đều dựa trên sự đánh giá của giáo viên. Hơn nữa, giáo viên phụ trách việc giảng dạy bộ môn. Bộ môn giáo viên giảng dạy sẽ tùy theo từng kì. Vì vậy, **Lecture** chắc chắn xác định có.
- **Semester:** Các môn học đều có khả năng xuất hiện trong một hay nhiều kì học, mà các môn học đều chỉ bắt đầu và kết thúc trong 1 kì, không có môn nào diễn ra quá 1 kì. **Semester** sẽ là 1 trong những thực thể có thể xuất hiện.
- **Group:** Trong 1 kì học, 1 **Group** sẽ có nhiều học sinh, sinh viên tham gia. Tuy vậy, không phải học sinh, sinh viên nào cũng học 1 môn. Và 1 học sinh có thể tham gia nhiều Group khác nhau để học các môn khác nhau.

-
- **Course(Subject)**: Yếu tố chính, quyết định tới từng sinh viên và giáo viên đánh giá. Một môn học có thể học bởi nhiều **Group** và có thể xuất hiện trong nhiều kì.
 - **Category**: Như đã đề cập trong các kết luận khi phân tích các bảng. Category là thực thể đánh giá cũng như quyết định tới điểm của học sinh, sinh viên và việc có hoàn thành hay không **Course**.

2. GIẢI ĐỊNH VỀ MỐI QUAN HỆ GIỮA CÁC THỰC THỂ

- Thực thể **Student** gồm các attributes:
 - + StudentID: Mã số sinh viên
 - + FirstName: Họ
 - + LastName: Tên
 - + Gender: Giới tính
 - + DateOfBirth: Ngày tháng năm sinh
 - + Address: Địa chỉ
 - + Mail: mail của học sinh, sinh viên

- Thực thể **Lecture** gồm các attributes:

- + LectureID: Mã số giáo viên
- + FullName: Họ và tên giáo viên
- + Mail: mail của giáo viên

- Thực thể **Semester** gồm các attributes:

- + SemesterID: Mã kì học
- + CourseID: Mã môn học
- + StartTime: Thời gian bắt đầu kì/môn học
- + EndTime: Thời gian kết thúc kì/môn học

- Thực thể **Course** gồm các attributes:

- + CourseID: Mã môn học
- + CourseName: Tên môn học
- + Total: Tổng số slot

- Thực thể **Group** gồm các attributes:

- + GroupID: mã số group
- + Major: chuyên ngành

- Thực thể **Category** gồm các attributes:

-
- + CategoryID: Mã điểm thành phần
 - + Type: Kiểu kiểm tra
 - + Weight: Trọng số
 - + [Completion Criteria]: điều kiện hoàn thành điểm

IV. Chuẩn hóa dữ liệu 3NF

1. XÁC ĐỊNH VÀ CHUYỂN ĐỔI CÁC THỰC THỂ

1.1. Xác định các mối quan hệ giữa các thực thể.

+ **Student** và **Group**: 1 học sinh có thể tham gia nhiều group và 1 group có thể có nhiều học sinh
-> Đây là mối quan hệ N-N

+ **Lecture** và **Semester**: Giảng viên có thể dạy nhiều kì khác nhau và 1 kì cũng có thể có nhiều giáo viên giảng dạy.
-> Đây là mối quan hệ N-N

+ **Semester** và **Group**: 1 kì có thể chứa nhiều group và 1 group có thể học trong nhiều kì.

-> Đây là mối quan hệ N-N

+ **Category** và **Course**: có nhiều điểm thành phần cho 1 môn, một vài điểm thành phần có thể có nhiều môn trùng nhau.

-> Mối quan hệ N-N

+ **Student** và **Semester**: 1 học sinh có thể học nhiều kì, có thể tạm bảo lưu 1 vài kì. 1 kì sẽ có nhiều sinh viên học.

-> Mối quan hệ N-N

1.2. Chuyển đổi các mối quan hệ của các thực thể

+ Đối với **Student** và **Group**: sẽ có thêm 1 cầu nối là **Join**

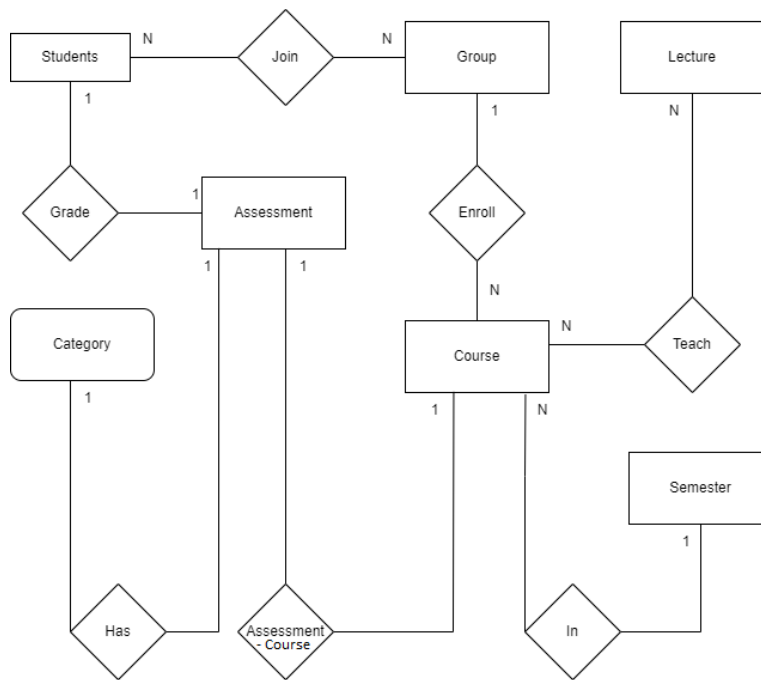
+ Đối với **Lecture** và **Semester**: sẽ có thêm 1 cầu nối là **Teach**

+ Đối với **Semester** và **Group**: sẽ có thêm cầu nối là **Enroll**

+ Đối với **Category** và **Course**: sẽ có thêm **Assessment**(Bảng đánh giá) và cầu nối giữa **Course** và **Assessment** là **Course_Assessment**

1.3. Bảng ERD tổng quan

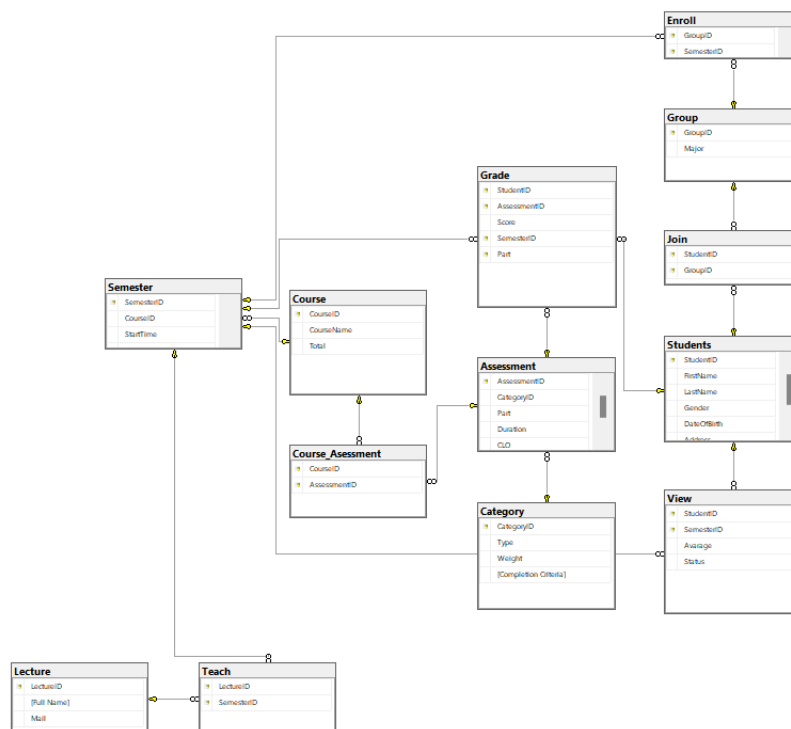
+ Bảng ERD trước khi thực hiện chia tách:



Subsystem Student Grading Management

➔ Dữ liệu sơ sài, chưa có tính đúng đắn. Các bảng chưa thể hiện rõ tính chất của nó.

+ Bảng ERD sau khi phân tách:



➔ Dữ liệu đã được tổ hợp 1 các gọn gàng hơn. Các dữ liệu có tính nhất quán.

2. XÁC ĐỊNH CÁC ATTRIBUTES CHO CÁC THỰC THỂ CHUYỂN ĐỔI

2.1. **Students:**

- + StudentID: nvarchar(8)
- + FirstName: nvarchar(50)
- + LastName: nvarchar(50)
- + Gender: bit
- + DateOfBirth: date
- + Address: nvarchar(50)
- + Mail: nvarchar(50)
- > StudentID sẽ là khóa chính và khóa ngoại

2.2. **Lecture**

- + LectureID: nvarchar(20)
- + FullName: nvarchar(150)
- + Mail: nvarchar(150)
- > LectureID sẽ là khóa chính và khóa ngoại

2.3. **Course**

- + CourseID: nvarchar(6)
- + CourseName: nvarchar(150)
- + Total: int
- > CourseID sẽ là khóa chính và khóa ngoại

2.4. **Group**

GroupID: nvarchar(6)

Major: nvachar(50)

-> GroupID sẽ là khóa chính

2.5. Join(Student_Group)

+ StudentID: nvarchar(8)

+ GroupID: nvarchar(6)

-> StudentID, GroupID sẽ là khóa chính và khóa ngoại

2.6. Enroll(Group_Course)

+ GroupID: nvarchar(6)

+ SemesterID: nvarchar(50)

-> GroupID, SemesterID sẽ là khóa chính và khóa ngoại

2.7. Teach

+ LectureID: nvarchar(20)

+ SemesterID: nvarchar(50)

-> LectureID, Semester sẽ là khóa chính và khóa ngoại

2.8. Assessments

- + AssessmentID: nvarchar(10)
- + CategoryID: nvarchar(20)
- + Part: int (null)
- + Duration: nvarchar(20)
- + LO: nvarchar(150)(null)
- + QuestionType: nvarchar(150)(null)

-> AssessmentID sẽ là khóa chính và khóa ngoại
CategoryID sẽ là khóa ngoại

2.9. Category

- + CategoryID: nvarchar(20)
- + Type: nvarchar(50)
- + Weight: float
- + Completion Criteria: nvarchar(2)

-> CategoryID sẽ là khóa chính và khóa ngoại

2.10. Grade

- + StudentID: nvarchar(8)
- + AssessmentID: nvarchar(150)
- + Score: float
- + SemesterID: nvarchar(50)
- + Part: int

-> StudentID, AssessmentID, SemesterID, Part sẽ
là khóa chính.

SemesterID và AssementID là khóa ngoại

2.11. Semester

SemesterID: nvarchar(50)

CourseID: nvarchar(50)

StartTime: date

EndTime: date

-> SemesterID sẽ là khóa chính và khóa ngoại

2.12. Assessment_Course

+ AssessmentID: nvarchar(10)

+ CourseID: nvarchar(6)

-> AssessmentID, CourseID sẽ là khóa chính và khóa ngoại

2.13. View

+ StudentID: nvarchar(8)

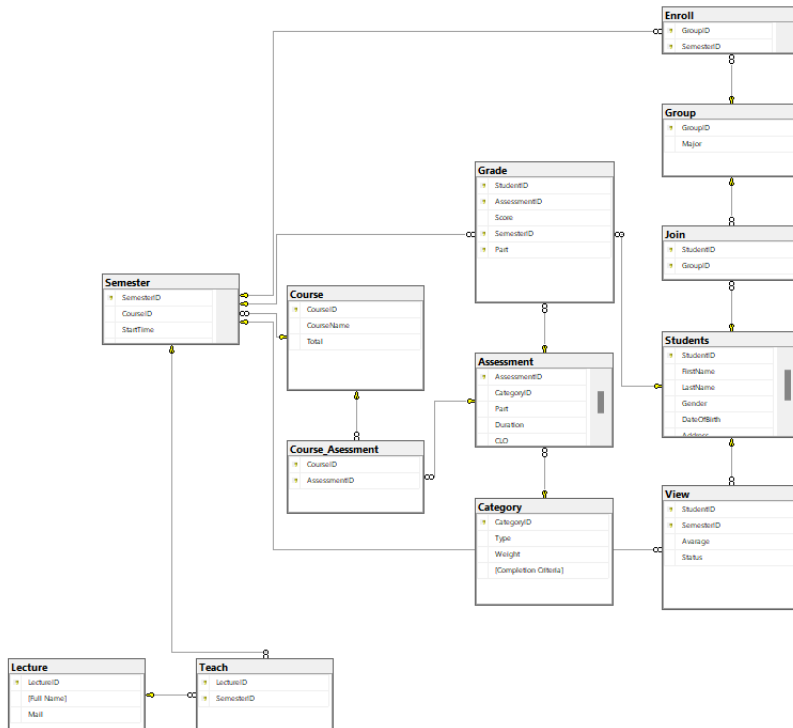
+ SemesterID: nvarchar(50)

+ Avarage: float

+ Status: nvarchar(20)

-> StudentID, Semester sẽ là khóa chính và khóa ngoại

Database Diagram:



V. Tạo bảng.

1. BẢNG ASSESSMENT

```

CREATE TABLE [dbo].[Assessment](
    [AssessmentID] [nvarchar](10) NOT NULL,
    [CategoryID] [nvarchar](20) NOT NULL,
    [Part] [int] NULL,
    [Duration] [nvarchar](20) NOT NULL,
    [CLO] [nvarchar](150) NULL,
    [Question Type] [nvarchar](150) NULL,
    CONSTRAINT [PK_Assessment] PRIMARY KEY CLUSTERED
(
    [AssessmentID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,
ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY =
OFF) ON [PRIMARY]
) ON [PRIMARY]
ALTER TABLE [dbo].[Assessment] WITH CHECK ADD
CONSTRAINT [FK_Assessment_Category] FOREIGN
KEY([CategoryID])
REFERENCES [dbo].[Category] ([CategoryID])
ALTER TABLE [dbo].[Assessment] CHECK CONSTRAINT
[FK_Assessment_Category]

```

2. BẢNG CATEGORY

```

REATE TABLE [dbo].[Category](
    [CategoryID] [nvarchar](20) NOT NULL,
    [Type] [nvarchar](50) NOT NULL,
    [Weight] [float] NOT NULL,
    [Completion Criteria] [nvarchar](2) NOT NULL,
    CONSTRAINT [PK_Category] PRIMARY KEY CLUSTERED
(
    [CategoryID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,
ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY =
OFF) ON [PRIMARY]
) ON [PRIMARY]
GO

```

3. BẢNG COURSE

```
CREATE TABLE [dbo].[Course](
    [CourseID] [nvarchar](6) NOT NULL,
    [CourseName] [nvarchar](150) NOT NULL,
    [Total] [int] NOT NULL,
    CONSTRAINT [PK_Course] PRIMARY KEY CLUSTERED
(
    [CourseID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,
ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY =
OFF) ON [PRIMARY]
) ON [PRIMARY]
```

4. BẢNG COURSE_ASSESSMENT

```
CREATE TABLE [dbo].[Course_Assessment](
    [CourseID] [nvarchar](6) NOT NULL,
    [AssessmentID] [nvarchar](10) NOT NULL,
    CONSTRAINT [PK_Course_Assessment] PRIMARY KEY
CLUSTERED
(
    [CourseID] ASC,
    [AssessmentID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,
ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY =
OFF) ON [PRIMARY]
) ON [PRIMARY]
ALTER TABLE [dbo].[Course_Assessment] WITH CHECK ADD
CONSTRAINT [FK_Course_Assessment_Assessment] FOREIGN
KEY([AssessmentID])
REFERENCES [dbo].[Assessment] ([AssessmentID])
GO
```

```

ALTER TABLE [dbo].[Course_Asessment] CHECK CONSTRAINT
[FK_Course_Asessment_Assessment]
GO
ALTER TABLE [dbo].[Course_Asessment] WITH CHECK ADD
CONSTRAINT [FK_Course_Asessment_Course] FOREIGN
KEY([CourseID])
REFERENCES [dbo].[Course] ([CourseID])
GO
ALTER TABLE [DBO].[COURSE_ASESSMENT] CHECK
CONSTRAINT [FK_COURSE_ASESSMENT_COURSE]

```

5. BẢNG ENROLL

```

CREATE TABLE [dbo].[Enroll](
    [GroupID] [nvarchar](6) NOT NULL,
    [SemesterID] [nvarchar](50) NOT NULL,
    CONSTRAINT [PK_Enroll] PRIMARY KEY CLUSTERED
(
    [GroupID] ASC,
    [SemesterID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,
ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY =
OFF) ON [PRIMARY]
) ON [PRIMARY]
ALTER TABLE [dbo].[Enroll] WITH CHECK ADD CONSTRAINT
[FK_Enroll_Group] FOREIGN KEY([GroupID])
REFERENCES [dbo].[Group] ([GroupID])
GO
ALTER TABLE [dbo].[Enroll] CHECK CONSTRAINT
[FK_Enroll_Group]

```

```

GO
ALTER TABLE [dbo].[Enroll] WITH CHECK ADD CONSTRAINT
[FK_Enroll_Semester] FOREIGN KEY([SemesterID])
REFERENCES [dbo].[Semester] ([SemesterID])
GO
ALTER TABLE [DBO].[ENROLL] CHECK CONSTRAINT
[FK_ENROLL_SEMESTER]

```

6. BẢNG GRADE

```

REATE TABLE [dbo].[Grade](
    [StudentID] [nvarchar](8) NOT NULL,
    [AssessmentID] [nvarchar](10) NOT NULL,
    [Score] [float] NOT NULL,
    [SemesterID] [nvarchar](50) NOT NULL,
    [Part] [int] NOT NULL,
    CONSTRAINT [PK_Grade] PRIMARY KEY CLUSTERED
(
    [StudentID] ASC,
    [AssessmentID] ASC,
    [SemesterID] ASC,
    [Part] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,
ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY =
OFF) ON [PRIMARY]
) ON [PRIMARY]

```



```

ALTER TABLE [dbo].[Grade] WITH CHECK ADD CONSTRAINT
[FK_Grade_Assessment] FOREIGN KEY([AssessmentID])
REFERENCES [dbo].[Assessment] ([AssessmentID])
GO
ALTER TABLE [dbo].[Grade] CHECK CONSTRAINT
[FK_Grade_Assessment]
GO
ALTER TABLE [dbo].[Grade] WITH CHECK ADD CONSTRAINT
[FK_Grade_Semester] FOREIGN KEY([SemesterID])
REFERENCES [dbo].[Semester] ([SemesterID])
GO
ALTER TABLE [dbo].[Grade] CHECK CONSTRAINT
[FK_Grade_Semester]
GO
ALTER TABLE [dbo].[Grade] WITH CHECK ADD CONSTRAINT
[FK_Grade_Students] FOREIGN KEY([StudentID])
REFERENCES [dbo].[Students] ([StudentID])
GO
ALTER TABLE [DBO].[GRADE] CHECK CONSTRAINT
[FK_GRADE_STUDENTS]

```

7. BẢNG GROUP

```

CREATE TABLE [dbo].[Group](
    [GroupID] [nvarchar](6) NOT NULL,
    [Major] [nvarchar](50) NOT NULL,
    CONSTRAINT [PK_Group] PRIMARY KEY CLUSTERED
(
    [GroupID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,
ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY =
OFF) ON [PRIMARY]
) ON [PRIMARY]

```

8. BẢNG JOIN

```

CREATE TABLE [dbo].[Join](

```

```

        [StudentID] [nvarchar](8) NOT NULL,
        [GroupID] [nvarchar](6) NOT NULL,
        CONSTRAINT [PK_Join] PRIMARY KEY CLUSTERED
    (
        [StudentID] ASC,
        [GroupID] ASC
    )WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
    IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,
    ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY =
    OFF) ON [PRIMARY]
    ) ON [PRIMARY]
ALTER TABLE [dbo].[Join] WITH CHECK ADD CONSTRAINT
[FK_Join_Group] FOREIGN KEY([GroupID])
REFERENCES [dbo].[Group] ([GroupID])
GO
ALTER TABLE [dbo].[Join] CHECK CONSTRAINT [FK_Join_Group]
GO
ALTER TABLE [dbo].[Join] WITH CHECK ADD CONSTRAINT
[FK_Join_Students] FOREIGN KEY([StudentID])
REFERENCES [dbo].[Students] ([StudentID])
GO
ALTER TABLE [DBO].[JOIN] CHECK CONSTRAINT
[FK_JOIN_STUDENTS]

```

9. BẢNG LECTURE

```

CREATE TABLE [dbo].[Lecture](
    [LectureID] [nvarchar](20) NOT NULL,
    [Full Name] [nvarchar](150) NOT NULL,
    [Mail] [nvarchar](150) NOT NULL,
    CONSTRAINT [PK_Lecture] PRIMARY KEY CLUSTERED
    (
        [LectureID] ASC
    )WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
    IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,
    ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY =
    OFF) ON [PRIMARY]
    ) ON [PRIMARY]

```

10. BẢNG SEMESTER

```

CREATE TABLE [dbo].[Semester](
    [SemesterID] [nvarchar](50) NOT NULL,
    [CourseID] [nvarchar](6) NOT NULL,
    [StartTime] [date] NOT NULL,
    [EndTime] [date] NOT NULL,
    CONSTRAINT [PK_Semester] PRIMARY KEY CLUSTERED
(
    [SemesterID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,
ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY =
OFF) ON [PRIMARY]
) ON [PRIMARY]
ALTER TABLE [dbo].[Semester] WITH CHECK ADD CONSTRAINT
[FK_Semester_Course] FOREIGN KEY([CourseID])
REFERENCES [dbo].[Course] ([CourseID])
GO

ALTER TABLE [DBO].[SEMESTER] CHECK CONSTRAINT
[FK_SEMESTER_COURSE]

```

11. BẢNG STUDENTS

```

CREATE TABLE [dbo].[Students](
    [StudentID] [nvarchar](8) NOT NULL,
    [FirstName] [nvarchar](50) NOT NULL,
    [LastName] [nvarchar](50) NOT NULL,
    [Gender] [bit] NOT NULL,
    [DateOfBirth] [date] NOT NULL,
    [Address] [nvarchar](50) NOT NULL,
    [Mail] [nvarchar](50) NOT NULL,
    CONSTRAINT [PK_Students] PRIMARY KEY CLUSTERED
(
    [StudentID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE =
OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,
ALLOW_PAGE_LOCKS = ON,
OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF) ON [PRIMARY]
) ON [PRIMARY]

```

12. BẢNG TEACH

```
CREATE TABLE [dbo].[Teach](
    [LectureID] [nvarchar](20) NOT NULL,
    [SemesterID] [nvarchar](50) NOT NULL,
    CONSTRAINT [PK_Teach] PRIMARY KEY CLUSTERED
(
    [LectureID] ASC,
    [SemesterID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,
ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY =
OFF) ON [PRIMARY]
) ON [PRIMARY]
ALTER TABLE [dbo].[Teach] WITH CHECK ADD CONSTRAINT
[FK_Teach_Lecture] FOREIGN KEY([LectureID])
REFERENCES [dbo].[Lecture] ([LectureID])
GO
ALTER TABLE [dbo].[Teach] CHECK CONSTRAINT
[FK_Teach_Lecture]
GO
ALTER TABLE [dbo].[Teach] WITH CHECK ADD CONSTRAINT
[FK_Teach_Semester1] FOREIGN KEY([SemesterID])
REFERENCES [dbo].[Semester] ([SemesterID])
GO
ALTER TABLE [DBO].[TEACH] CHECK CONSTRAINT
[FK_TEACH_SEMESTER1]
```

13. BẢNG VIEW

```
CREATE TABLE [dbo].[View](
    [StudentID] [nvarchar](8) NOT NULL,
    [SemesterID] [nvarchar](50) NOT NULL,
    [Avarage] [float] NOT NULL,
    [Status] [nvarchar](20) NOT NULL,
    CONSTRAINT [PK_View_1] PRIMARY KEY CLUSTERED
(
    [StudentID] ASC,
```

```

        [SemesterID] ASC
    )WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
    IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,
    ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY =
    OFF) ON [PRIMARY]
    ) ON [PRIMARY]
ALTER TABLE [dbo].[Teach] CHECK CONSTRAINT
[FK_Teach_Semester1]
GO
ALTER TABLE [dbo].[View] WITH CHECK ADD CONSTRAINT
[FK_View_Semester] FOREIGN KEY([SemesterID])
REFERENCES [dbo].[Semester] ([SemesterID])
GO
ALTER TABLE [dbo].[View] CHECK CONSTRAINT
[FK_View_Semester]
GO
ALTER TABLE [dbo].[View] WITH CHECK ADD CONSTRAINT
[FK_View_Students] FOREIGN KEY([StudentID])
REFERENCES [dbo].[Students] ([StudentID])
GO
ALTER TABLE [DBO].[VIEW] CHECK CONSTRAINT
[FK_VIEW_STUDENTS]

```

VI. Thêm dữ liệu

1. Bảng Students

```

INSERT INTO [dbo].[Students](StudentID, FirstName, LastName, Gender,
DateOfBirth, [Address], Mail) VALUES ('HE163780', 'Vương
Đình', 'Ngân', '1', CAST(N'2002-12-02' AS DATE), 'Hà
Nội', 'NganVDHE163780@fpt.edu.vn');
INSERT INTO [dbo].[Students](StudentID, FirstName, LastName, Gender,
DateOfBirth, [Address], Mail) VALUES ('SE143042', 'Tăng
Chí', 'Khiêm', '1', CAST(N'2000-09-08' AS DATE), 'Cao
Bằng', 'KhiemTCSE143042@fpt.edu.vn');
INSERT INTO [dbo].[Students](StudentID, FirstName, LastName, Gender,
DateOfBirth, [Address], Mail) VALUES ('IS149076', 'Nguyễn
Quang', 'Thạch', '1', CAST(N'2000-12-19' AS DATE), 'Bắc
Kạn', 'ThachNQIS149076@fpt.edu.vn');
INSERT INTO [dbo].[Students](StudentID, FirstName, LastName, Gender,
DateOfBirth, [Address], Mail) VALUES ('HE159082', 'Đoàn

```

```

Công', 'Lộc', '1', CAST(N'2001-01-15' AS DATE), 'Cần
Thơ', 'LocDCHE159082@fpt.edu.vn');
INSERT INTO [dbo].[Students](StudentID, FirstName, LastName, Gender,
DateOfBirth, [Address], Mail) VALUES ('SE162847', 'Phạm
Hương', 'Tiền', '1', CAST(N'2002-11-19' AS DATE), 'Phú
Thọ', 'TienPHSE162847@fpt.edu.vn');
INSERT INTO [dbo].[Students](StudentID, FirstName, LastName, Gender,
DateOfBirth, [Address], Mail) VALUES ('HE132932', 'Nguyễn
Khuyến', 'Học', '1', CAST(N'1999-07-19' AS DATE), 'Tuyên
Quang', 'HocNKHE132932@fpt.edu.vn');
INSERT INTO [dbo].[Students](StudentID, FirstName, LastName, Gender,
DateOfBirth, [Address], Mail) VALUES ('IS176801', 'Đỗ
Trường', 'Sinh', '1', CAST(N'2003-07-19' AS DATE), 'Hải
Phòng', 'SinhDTIS175801@fpt.edu.vn');
INSERT INTO [dbo].[Students](StudentID, FirstName, LastName, Gender,
DateOfBirth, [Address], Mail) VALUES ('IS168901', 'Đặng
Bội', 'Linh', '0', CAST(N'2002-08-04' AS DATE), 'Đà
Nẵng', 'LinhDBIS168901@fpt.edu.vn');
INSERT INTO [dbo].[Students](StudentID, FirstName, LastName, Gender,
DateOfBirth, [Address], Mail) VALUES ('HE173690', 'Ngũ
Băng', 'Yên', '0', CAST(N'2003-06-07' AS DATE), 'Hà
Nội', 'YenNBHE173690@fpt.edu.vn');
INSERT INTO [dbo].[Students](StudentID, FirstName, LastName, Gender,
DateOfBirth, [Address], Mail) VALUES ('HE156721', 'Lý
Đình', 'Hương', '0', CAST(N'2001-03-17' AS DATE), 'Tuyên
Quang', 'HuongLDHE156721@fpt.edu.vn');
INSERT INTO [dbo].[Students](StudentID, FirstName, LastName, Gender,
DateOfBirth, [Address], Mail) VALUES ('HE179012', 'Vương
Hồng', 'Châu', '0', CAST(N'2003-01-02' AS DATE), 'Bắc
Ninh', 'ChauVHHE179012@fpt.edu.vn');
INSERT INTO [dbo].[Students](StudentID, FirstName, LastName, Gender,
DateOfBirth, [Address], Mail) VALUES ('HE178013', 'Trần
Minh', 'Hiếu', '1', CAST(N'2003-08-23' AS DATE), 'Phú
Thọ', 'HieuTMHE179013@fpt.edu.vn');
INSERT INTO [dbo].[Students](StudentID, FirstName, LastName, Gender,
DateOfBirth, [Address], Mail) VALUES ('IS159011', 'Đỗ
Thế', 'Cường', '1', CAST(N'2001-12-09' AS DATE), 'Bắc
Ninh', 'CuongDTIS159200@fpt.edu.vn');
INSERT INTO [dbo].[Students](StudentID, FirstName, LastName, Gender,
DateOfBirth, [Address], Mail) VALUES ('IS159200', 'Tần
Linh', 'Đam', '0', CAST(N'2001-10-10' AS DATE), 'Vũng
Tàu', 'DanTLIS159200@fpt.edu.vn');
INSERT INTO [dbo].[Students](StudentID, FirstName, LastName, Gender,
DateOfBirth, [Address], Mail) VALUES ('SE149001', 'Tổ
Đào', 'Vân', '0', CAST(N'09-08-2000' AS DATE), 'Hòa
Bình', 'VanTDSE149001@fpt.edu.vn');

```

2. Bảng Lecture

```
INSERT INTO [dbo].[Lecture](LectureID, [Full Name], Mail) VALUES
('SonNT5', 'Ngô Tùng Sơn', 'sonnt69@fe.edu.vn')
INSERT INTO [dbo].[Lecture](LectureID, [Full Name], Mail) VALUES
('VanDT', 'Đỗ Thị Vân', 'vandt4@fe.edu.vn')
INSERT INTO [dbo].[Lecture](LectureID, [Full Name], Mail) VALUES
('DuchM29', 'Hoàng Mạnh Đức', 'DuchM29@fe.edu.vn')
INSERT INTO [dbo].[Lecture](LectureID, [Full Name], Mail) VALUES
('NangNTH', 'Nguyễn Thị Hải Năng', 'NangNTH@fe.edu.vn')
INSERT INTO [dbo].[Lecture](LectureID, [Full Name], Mail) VALUES
('AnNT79', 'Nguyễn Tân Ân', 'annt79@fpt.edu.vn')
```

3. Bảng Course

```
INSERT INTO [dbo].[Course](CourseID, CourseName, [Total]) VALUES
('CEA201', 'Computer Organization and Architecture', '30');
INSERT INTO [dbo].[Course](CourseID, CourseName, [Total]) VALUES
('CSI104', 'Introduction to Computer Science', '30');
INSERT INTO [dbo].[Course](CourseID, CourseName, [Total]) VALUES
('PRF192', 'Programming Fundamentals', '29');
INSERT INTO [dbo].[Course](CourseID, CourseName, [Total]) VALUES
('NWC204', 'Computer Networking', '30');
INSERT INTO [dbo].[Course](CourseID, CourseName, [Total]) VALUES
('PRO192', 'Object-Oriented Programming', '30');
INSERT INTO [dbo].[Course](CourseID, CourseName, [Total]) VALUES
('IA0202', 'Introduction to Information Assurance', '30');
INSERT INTO [dbo].[Course](CourseID, CourseName, [Total]) VALUES
('JPD113', 'Elementary Japanese 1-A1.1', '30');
INSERT INTO [dbo].[Course](CourseID, CourseName, [Total]) VALUES
('CSD201', 'Data Structures and Algorithms', '30');
INSERT INTO [dbo].[Course](CourseID, CourseName, [Total]) VALUES
('DBI202', 'Introduction to Databases', '30');
```

4. Bảng Group

```
INSERT INTO [dbo].[Group](GroupID, Major) VALUES ('IA1604', 'Information Assurance')
INSERT INTO [dbo].[Group](GroupID, Major) VALUES ('IS1704', 'Information System')
INSERT INTO [dbo].[Group](GroupID, Major) VALUES ('CS1604', 'Computer Science')
```

5. Bảng Category

```
INSERT INTO [dbo].[Category] ([CategoryID], [Type], [Weight],
[Completion Criteria]) VALUES (N'Assignment', N'on-going', 0.2,
N'>0');
INSERT INTO [dbo].[Category] ([CategoryID], [Type], [Weight],
[Completion Criteria]) VALUES (N'Assignment1', N'on-going', 0.15,
N'>0');
INSERT INTO [dbo].[Category] ([CategoryID], [Type], [Weight],
[Completion Criteria]) VALUES (N'Final Exam', N'final exam', 0.3,
N'>4');
INSERT INTO [dbo].[Category] ([CategoryID], [Type], [Weight],
[Completion Criteria]) VALUES (N'Final Exam1', N'final exam', 0.4,
N'>4');
INSERT INTO [dbo].[Category] ([CategoryID], [Type], [Weight],
[Completion Criteria]) VALUES (N'Labs', N'on-going', 0.017, N'>0');
INSERT INTO [dbo].[Category] ([CategoryID], [Type], [Weight],
[Completion Criteria]) VALUES (N'Labs1', N'on-going', 0.03, N'>0');
INSERT INTO [dbo].[Category] ([CategoryID], [Type], [Weight],
[Completion Criteria]) VALUES (N'Labs2', N'on-going', 0.02, N'>0');
INSERT INTO [dbo].[Category] ([CategoryID], [Type], [Weight],
[Completion Criteria]) VALUES (N'Practical Exam', N'practical exam',
0.3, N'>0');
INSERT INTO [dbo].[Category] ([CategoryID], [Type], [Weight],
[Completion Criteria]) VALUES (N'Progress Test', N'Quiz', 0.1,
N'>0');
INSERT INTO [dbo].[Category] ([CategoryID], [Type], [Weight],
[Completion Criteria]) VALUES (N'Progress Test1', N'Quiz', 0.05,
N'>0');
INSERT INTO [dbo].[Category] ([CategoryID], [Type], [Weight],
[Completion Criteria]) VALUES (N'Mid-Term', N' on-going', 0.3,
N'>0');
INSERT INTO [dbo].[Category] ([CategoryID], [Type], [Weight],
[Completion Criteria]) VALUES (N'Participation', N'on-going', 0.1,
N'>0');
INSERT INTO [dbo].[Category] ([CategoryID], [Type], [Weight],
[Completion Criteria]) VALUES (N'FE_Listening', N'final listening',
0.1, N'>4');
INSERT INTO [dbo].[Category] ([CategoryID], [Type], [Weight],
[Completion Criteria]) VALUES (N'FE_GVR', N'final exam gvr', 0.3,
N'>4');
INSERT INTO [dbo].[Category] ([CategoryID], [Type], [Weight],
[Completion Criteria]) VALUES (N'Progress Test2', N'Quiz', 0.2,
N'>0');
INSERT INTO [dbo].[Category] ([CategoryID], [Type], [Weight],
[Completion Criteria]) VALUES (N'Exercies', N'Quiz', 0.2, N'>0');
```

6. Bảng Semester

```
INSERT INTO [dbo].[Semester](SemesterID,CourseID, StartTime,EndTime) VALUES
('Fall12021a', 'CEA201',CAST(N'2021-09-06' AS DATE), CAST(N'2021-11-12' AS
DATE));
```



```

INSERT INTO [dbo].[Semester](SemesterID,CourseID, StartTime,EndTime) VALUES
('Fall2021b', 'CSI104',CAST(N'2021-09-07' AS DATE), CAST(N'2021-11-11' AS
DATE));
INSERT INTO [dbo].[Semester](SemesterID,CourseID, StartTime,EndTime) VALUES
('Spring2022a', 'NWC204',CAST(N'2022-01-05' AS DATE), CAST(N'2022-03-25' AS
DATE));
INSERT INTO [dbo].[Semester](SemesterID,CourseID, StartTime,EndTime) VALUES
('Spring2022b', 'PRO192', CAST(N'2022-01-05' AS DATE), CAST(N'2022-03-23' AS
DATE));
INSERT INTO [dbo].[Semester](SemesterID,CourseID, StartTime,EndTime) VALUES
('Summer2022a', 'JPD113', CAST(N'2022-05-10' AS DATE), CAST(N'2022-07-21' AS
DATE));
INSERT INTO [dbo].[Semester](SemesterID,CourseID, StartTime,EndTime) VALUES
('Summer2022b', 'DBI202', CAST(N'2022-05-10' AS DATE), CAST(N'2022-07-20' AS
DATE));
INSERT INTO [dbo].[Semester](SemesterID,CourseID, StartTime,EndTime) VALUES
('Summer2022c', 'CSD201', CAST(N'2022-05-09' AS DATE), CAST(N'2022-07-20' AS
DATE));

```

7. Bảng Teach

```

INSERT INTO [dbo].[Teach]([LectureID],[SemesterID]) VALUES ('SonNT5',
'Fall2021');
INSERT INTO [dbo].[Teach]([LectureID],[SemesterID]) VALUES ('SonNT5',
'Summer2022');
INSERT INTO [dbo].[Teach]([LectureID],[SemesterID]) VALUES ('NangNTH',
'Spring2022');
INSERT INTO [dbo].[Teach]([LectureID],[SemesterID]) VALUES ('NangNTH',
'Summer2022');
INSERT INTO [dbo].[Teach]([LectureID],[SemesterID]) VALUES ('DucHM29',
'Fall2021');
INSERT INTO [dbo].[Teach]([LectureID],[SemesterID]) VALUES ('DucHM29',
'Spring2022');
INSERT INTO [dbo].[Teach]([LectureID],[SemesterID]) VALUES ('DucHM29',
'Summer2022');
INSERT INTO [dbo].[Teach]([LectureID],[SemesterID]) VALUES ('VanDT',
'Summer2022');
INSERT INTO [dbo].[Teach]([LectureID],[SemesterID]) VALUES ('AnNT79',
'Spring2022');
INSERT INTO [dbo].[Teach]([LectureID],[SemesterID]) VALUES ('AnNT79',
'Summer2022');

```

8. Bảng Assessment

```

INSERT [dbo].[Assessment] ([AssessmentID], [CategoryID], [Part],
[Duration], [CLO], [Question Type]) VALUES (N'ASM', N'Assignment',
1, N'30 slots', NULL, N'by instructor')

```

```

INSERT [dbo].[Assessment] ([AssessmentID], [CategoryID], [Part],
[Duration], [CLO], [Question Type]) VALUES (N'ASM1', N'Assignment1',
1, N'30 slots', NULL, N'by instructor')
INSERT [dbo].[Assessment] ([AssessmentID], [CategoryID], [Part],
[Duration], [CLO], [Question Type]) VALUES (N'FE', N'Final Exam', 1,
N'60'', NULL, NULL)
INSERT [dbo].[Assessment] ([AssessmentID], [CategoryID], [Part],
[Duration], [CLO], [Question Type]) VALUES (N'FE_GVR', N'FE_GVR', 1,
N'60'', NULL, NULL)
INSERT [dbo].[Assessment] ([AssessmentID], [CategoryID], [Part],
[Duration], [CLO], [Question Type]) VALUES (N'FE_LSN',
N'FE_Listening', 1, N'60'', NULL, NULL)
INSERT [dbo].[Assessment] ([AssessmentID], [CategoryID], [Part],
[Duration], [CLO], [Question Type]) VALUES (N'FE1', N'Final Exam1',
1, N'60'', NULL, NULL)
INSERT [dbo].[Assessment] ([AssessmentID], [CategoryID], [Part],
[Duration], [CLO], [Question Type]) VALUES (N'FE2', N'Final Exam 2',
1, N'60'', NULL, NULL)
INSERT [dbo].[Assessment] ([AssessmentID], [CategoryID], [Part],
[Duration], [CLO], [Question Type]) VALUES (N'Labs', N'Labs', 6,
N'90'', NULL, NULL)
INSERT [dbo].[Assessment] ([AssessmentID], [CategoryID], [Part],
[Duration], [CLO], [Question Type]) VALUES (N'Labs1', N'Labs1', 10,
N'30 slots', NULL, NULL)
INSERT [dbo].[Assessment] ([AssessmentID], [CategoryID], [Part],
[Duration], [CLO], [Question Type]) VALUES (N'Labs2', N'Labs2', 5,
N'30 slots', NULL, NULL)
INSERT [dbo].[Assessment] ([AssessmentID], [CategoryID], [Part],
[Duration], [CLO], [Question Type]) VALUES (N'MD', N'Mid-Term', 1,
N'60'', NULL, NULL)
INSERT [dbo].[Assessment] ([AssessmentID], [CategoryID], [Part],
[Duration], [CLO], [Question Type]) VALUES (N'PE', N'Practical
Exam', 1, N'90'', NULL, NULL)
INSERT [dbo].[Assessment] ([AssessmentID], [CategoryID], [Part],
[Duration], [CLO], [Question Type]) VALUES (N'PT', N'Progress Test',
2, N'30'', NULL, N'Multiple Choice on EOS, Google form')
INSERT [dbo].[Assessment] ([AssessmentID], [CategoryID], [Part],
[Duration], [CLO], [Question Type]) VALUES (N'PT1', N'Progress
Test1', 2, N'25'', NULL, N'Multiple Choice on EOS, Google form')
INSERT [dbo].[Assessment] ([AssessmentID], [CategoryID], [Part],
[Duration], [CLO], [Question Type]) VALUES (N'PT2', N'Progress
Test2', 2, N'30'', NULL, N'Multiple Choice on EOS, Google form')
INSERT [dbo].[Assessment] ([AssessmentID], [CategoryID], [Part],
[Duration], [CLO], [Question Type]) VALUES (N'PTT',
N'Participation', 1, N'30 slots', NULL, NULL)
INSERT [dbo].[Assessment] ([AssessmentID], [CategoryID], [Part],
[Duration], [CLO], [Question Type]) VALUES (N'EXX', N'Exercises', 4,
N'30'', NULL, NULL)

```

9. Bảng Join

```
INSERT [dbo].[Join] ([StudentID], [GroupID]) VALUES (N'HE132932',  
N'CS1604')  
INSERT [dbo].[Join] ([StudentID], [GroupID]) VALUES (N'IS149076',  
N'IA1604')  
INSERT [dbo].[Join] ([StudentID], [GroupID]) VALUES (N'HE156721',  
N'IA1604')  
INSERT [dbo].[Join] ([StudentID], [GroupID]) VALUES (N'HE163780',  
N'IS1704')  
INSERT [dbo].[Join] ([StudentID], [GroupID]) VALUES (N'IS159200',  
N'IA1604')  
INSERT [dbo].[Join] ([StudentID], [GroupID]) VALUES (N'IS168901',  
N'IS1704')  
INSERT [dbo].[Join] ([StudentID], [GroupID]) VALUES (N'SE143042',  
N'CS1604')  
INSERT [dbo].[Join] ([StudentID], [GroupID]) VALUES (N'SE149001',  
N'CS1604')  
INSERT [dbo].[Join] ([StudentID], [GroupID]) VALUES (N'HE159082',  
N'IS1704')
```

10. Bảng Assessment_Course

```
INSERT [dbo].[Course_Assessment] ([CourseID], [AssessmentID]) VALUES  
(N'CEA201', N'ASM1')  
INSERT [dbo].[Course_Assessment] ([CourseID], [AssessmentID]) VALUES  
(N'CEA201', N'EXX')  
INSERT [dbo].[Course_Assessment] ([CourseID], [AssessmentID]) VALUES  
(N'CEA201', N'FE1')  
INSERT [dbo].[Course_Assessment] ([CourseID], [AssessmentID]) VALUES  
(N'DBI202', N'ASM')  
INSERT [dbo].[Course_Assessment] ([CourseID], [AssessmentID]) VALUES  
(N'DBI202', N'FE')  
INSERT [dbo].[Course_Assessment] ([CourseID], [AssessmentID]) VALUES  
(N'DBI202', N'Labs2')  
INSERT [dbo].[Course_Assessment] ([CourseID], [AssessmentID]) VALUES  
(N'DBI202', N'PE')  
INSERT [dbo].[Course_Assessment] ([CourseID], [AssessmentID]) VALUES  
(N'DBI202', N'PT1')  
INSERT [dbo].[Course_Assessment] ([CourseID], [AssessmentID]) VALUES  
(N'JPD113', N'FE_GVR')  
INSERT [dbo].[Course_Assessment] ([CourseID], [AssessmentID]) VALUES  
(N'JPD113', N'FE_LSN')  
INSERT [dbo].[Course_Assessment] ([CourseID], [AssessmentID]) VALUES  
(N'JPD113', N'MD')  
INSERT [dbo].[Course_Assessment] ([CourseID], [AssessmentID]) VALUES  
(N'JPD113', N'PT')  
INSERT [dbo].[Course_Assessment] ([CourseID], [AssessmentID]) VALUES  
(N'JPD113', N'PTT')  
INSERT [dbo].[Course_Assessment] ([CourseID], [AssessmentID]) VALUES  
(N'NWC204', N'FE')
```

```

INSERT [dbo].[Course_Assessment] ([CourseID], [AssessmentID]) VALUES
(N'NWC204', N'Labs1')
INSERT [dbo].[Course_Assessment] ([CourseID], [AssessmentID]) VALUES
(N'NWC204', N'PTT')
INSERT [dbo].[Course_Assessment] ([CourseID], [AssessmentID]) VALUES
(N'PRO192', N'ASM')
INSERT [dbo].[Course_Assessment] ([CourseID], [AssessmentID]) VALUES
(N'PRO192', N'FE')
INSERT [dbo].[Course_Assessment] ([CourseID], [AssessmentID]) VALUES
(N'PRO192', N'Labs')
INSERT [dbo].[Course_Assessment] ([CourseID], [AssessmentID]) VALUES
(N'PRO192', N'PE')
INSERT [dbo].[Course_Assessment] ([CourseID], [AssessmentID]) VALUES
(N'PRO192', N'PT1')

```

11. Bảng Enroll

```

INSERT [dbo].[Enroll] ([GroupID], [SemesterID]) VALUES (N'CS1604',
N'Summer2022a')
INSERT [dbo].[Enroll] ([GroupID], [SemesterID]) VALUES (N'IA1604',
N'Fall2021a')
INSERT [dbo].[Enroll] ([GroupID], [SemesterID]) VALUES (N'IA1604',
N'Spring2022a')
INSERT [dbo].[Enroll] ([GroupID], [SemesterID]) VALUES (N'IA1604',
N'Summer2022a')
INSERT [dbo].[Enroll] ([GroupID], [SemesterID]) VALUES (N'IS1704',
N'Spring2022b')
INSERT [dbo].[Enroll] ([GroupID], [SemesterID]) VALUES (N'IS1704',
N'Summer2022b')

```

12. Bảng Grade

```

INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'HE132932', N'FE_GVR', 7, N'Summer2022a', 1)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'HE132932', N'FE_LSN', 8.2, N'Summer2022a', 1)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'HE132932', N'MD', 7.5, N'Summer2022a', 1)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'HE132932', N'PT', 9, N'Summer2022a', 1)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'HE132932', N'PT', 8, N'Summer2022a', 2)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'HE132932', N'PTT', 10, N'Summer2022a', 1)

INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'SE143042', N'FE_GVR', 3, N'Summer2022a', 1)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'SE143042', N'FE_LSN', 3, N'Summer2022a', 1)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'SE143042', N'MD', 4.5, N'Summer2022a', 1)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'SE143042', N'PT', 7, N'Summer2022a', 1)

```

```

INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'SE143042', N'PT', 5, N'Summer2022a', 2)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'SE143042', N'PTT', 6, N'Summer2022a', 1)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'SE143042', N'FE_GVR', 3, N'Summer2022a', 2)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'SE143042', N'FE_LSN', 2, N'Summer2022a', 2)

INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID], [Part])
VALUES (N'SE149001', N'FE_GVR', 7.6, N'Summer2022a', 1)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID], [Part])
VALUES (N'SE149001', N'FE_LSN', 8.1, N'Summer2022a', 1)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID], [Part])
VALUES (N'SE149001', N'MD', 7.5, N'Summer2022a', 1)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID], [Part])
VALUES (N'SE149001', N'PT', 9.2, N'Summer2022a', 1)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID], [Part])
VALUES (N'SE149001', N'PT', 8.4, N'Summer2022a', 2)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID], [Part])
VALUES (N'SE149001', N'PTT', 10, N'Summer2022a', 1)

INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID], [Part])
VALUES (N'HE156721', N'ASM1', 10, N'Fall2021a', 1)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID], [Part])
VALUES (N'HE156721', N'ASM1', 10, N'Fall2021a', 1)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID], [Part])
VALUES (N'HE156721', N'EXX', 9, N'Fall2021a', 1)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID], [Part])
VALUES (N'HE156721', N'EXX', 8.9, N'Fall2021a', 2)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID], [Part])
VALUES (N'HE156721', N'EXX', 10, N'Fall2021a', 3)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID], [Part])
VALUES (N'HE156721', N'EXX', 9.8, N'Fall2021a', 4)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID], [Part])
VALUES (N'HE156721', N'FE1', 9, N'Fall2021a', 1)

INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID], [Part])
VALUES (N'IS149076', N'ASM1', 10, N'Fall2021a', 1)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID], [Part])
VALUES (N'IS149076', N'ASM1', 10, N'Fall2021a', 1)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID], [Part])
VALUES (N'IS149076', N'EXX', 9, N'Fall2021a', 1)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID], [Part])
VALUES (N'IS149076', N'EXX', 8.9, N'Fall2021a', 2)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID], [Part])
VALUES (N'IS149076', N'EXX', 10, N'Fall2021a', 3)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID], [Part])
VALUES (N'IS149076', N'EXX', 9.8, N'Fall2021a', 4)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID], [Part])
VALUES (N'IS149076', N'FE1', 9, N'Fall2021a', 1)

INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID], [Part])
VALUES (N'IS159200', N'ASM1', 5, N'Fall2021a', 1)

```



```

INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'IS149076', N'Labs1', 10, N'Spring2022a', 5)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'IS149076', N'Labs1', 10, N'Spring2022a', 6)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'IS149076', N'Labs1', 10, N'Spring2022a', 7)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID], [Part])
VALUES (N'IS149076', N'Labs1', 9.8, N'Spring2022a', 8)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'IS149076', N'Labs1', 10, N'Spring2022a', 9)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'IS149076', N'Labs1', 9, N'Spring2022a', 10)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'IS149076', N'PT', 10, N'Spring2022a', 1)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'IS149076', N'PT', 8.5, N'Spring2022a', 2)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'IS149076', N'FE', 9, N'Spring2022a', 1)

INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'IS159200', N'PTT', 8, N'Spring2022a', 1)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'IS159200', N'Labs1', 5, N'Spring2022a', 1)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'IS159200', N'Labs1', 5, N'Spring2022a', 2)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'IS159200', N'Labs1', 6, N'Spring2022a', 3)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'IS159200', N'Labs1', 7, N'Spring2022a', 4)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'IS159200', N'Labs1', 1, N'Spring2022a', 5)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'IS159200', N'Labs1', 1, N'Spring2022a', 6)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'IS159200', N'Labs1', 1, N'Spring2022a', 7)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'IS159200', N'Labs1', 4.8, N'Spring2022a', 8)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'IS159200', N'Labs1', 6, N'Spring2022a', 9)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'IS159200', N'Labs1', 1, N'Spring2022a', 10)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'IS159200', N'PT', 7, N'Spring2022a', 1)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'IS159200', N'PT', 8, N'Spring2022a', 2)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'IS159200', N'FE', 3, N'Spring2022a', 1)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'IS159200', N'FE', 3.5, N'Spring2022a', 2)

INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'HE156721', N'FE_GVR', 7.6, N'Summer2022a', 1)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'HE156721', N'FE_LSN', 8.1, N'Summer2022a', 1)

```

```

INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'HE156721', N'MD', 7.5, N'Summer2022a', 1)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'HE156721', N'PT', 9.2, N'Summer2022a', 1)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'HE156721', N'PT', 8.4, N'Summer2022a', 2)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'HE156721', N'PTT', 10, N'Summer2022a', 1)

INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'IS149076', N'FE_GVR', 7, N'Summer2022a', 1)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'IS149076', N'FE_LSN', 8, N'Summer2022a', 1)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'IS149076', N'MD', 5, N'Summer2022a', 1)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'IS149076', N'PT', 9, N'Summer2022a', 1)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'IS149076', N'PT', 8, N'Summer2022a', 2)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'IS149076', N'PTT', 10, N'Summer2022a', 1)

INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'IS159200', N'FE_GVR', 7.8, N'Summer2022a', 1)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'IS159200', N'FE_LSN', 8, N'Summer2022a', 1)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'IS159200', N'MD', 9.5, N'Summer2022a', 1)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'IS159200', N'PT', 9, N'Summer2022a', 1)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'IS159200', N'PT', 8.6, N'Summer2022a', 2)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'IS159200', N'PTT', 10, N'Summer2022a', 1)

INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'IS168901', N'Labs', 6.5, N'Spring2022b', 1)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'IS168901', N'Labs', 7, N'Spring2022b', 2)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'IS168901', N'Labs', 6.8, N'Spring2022b', 3)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'IS168901', N'Labs', 7.7, N'Spring2022b', 4)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'IS168901', N'Labs', 7, N'Spring2022b', 5)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID], [Part])
VALUES (N'IS168901', N'Labs', 7.5, N'Spring2022b', 6)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'IS168901', N'PT1', 7.2, N'Spring2022b', 1)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'IS168901', N'PT1', 8.3, N'Spring2022b', 2)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'IS168901', N'ASM', 7, N'Spring2022b', 1)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'IS168901', N'PE', 6, N'Spring2022b', 1)

```


[illegible]

```

INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID],
[Part]) VALUES (N'HE159082', N'PE', 5.5, N'Summer2022b', 1)
INSERT [dbo].[Grade] ([StudentID], [AssessmentID], [Score], [SemesterID], [Part])
VALUES (N'HE159082', N'FE', 7.5, N'Summer2022b', 1)

```

13. Bảng View

```

INSERT [dbo].[View] ([StudentID], [SemesterID], [Avarage], [Status]) VALUES
(N'HE132932', N'Summer2022a', 7.87, N'PASSED')
INSERT [dbo].[View] ([StudentID], [SemesterID], [Avarage], [Status]) VALUES
(N'HE156721', N'Fall2021a', 9.43, N'PASSED')
INSERT [dbo].[View] ([StudentID], [SemesterID], [Avarage], [Status]) VALUES
(N'HE156721', N'Spring2022a', 10, N'PASSED')
INSERT [dbo].[View] ([StudentID], [SemesterID], [Avarage], [Status]) VALUES
(N'HE156721', N'Summer2022a', 7.2, N'PASSED')
INSERT [dbo].[View] ([StudentID], [SemesterID], [Avarage], [Status]) VALUES
(N'HE159082', N'Spring2022b', 4, N'NOT PASSED')
INSERT [dbo].[View] ([StudentID], [SemesterID], [Avarage], [Status]) VALUES
(N'HE159082', N'Summer2022b', 6.87, N'PASSED')
INSERT [dbo].[View] ([StudentID], [SemesterID], [Avarage], [Status]) VALUES
(N'HE163780', N'Spring2022b', 7.03, N'PASSED')
INSERT [dbo].[View] ([StudentID], [SemesterID], [Avarage], [Status]) VALUES
(N'HE163780', N'Summer2022b', 7.25, N'PASSED')
INSERT [dbo].[View] ([StudentID], [SemesterID], [Avarage], [Status]) VALUES
(N'IS149076', N'Fall2021a', 9.43, N'PASSED')
INSERT [dbo].[View] ([StudentID], [SemesterID], [Avarage], [Status]) VALUES
(N'IS149076', N'Spring2022a', 8.41, N'PASSED')
INSERT [dbo].[View] ([StudentID], [SemesterID], [Avarage], [Status]) VALUES
(N'IS149076', N'Summer2022a', 7.1, N'PASSED')
INSERT [dbo].[View] ([StudentID], [SemesterID], [Avarage], [Status]) VALUES
(N'IS159200', N'Fall2021a', 4.75, N'NOT PASSED')
INSERT [dbo].[View] ([StudentID], [SemesterID], [Avarage], [Status]) VALUES
(N'IS159200', N'Spring2022a', 4.5, N'NOT PASSED')
INSERT [dbo].[View] ([StudentID], [SemesterID], [Avarage], [Status]) VALUES
(N'IS159200', N'Summer2022a', 8.75, N'PASSED')
INSERT [dbo].[View] ([StudentID], [SemesterID], [Avarage], [Status]) VALUES
(N'IS168901', N'Spring2022b', 6.23, N'PASSED')
INSERT [dbo].[View] ([StudentID], [SemesterID], [Avarage], [Status]) VALUES
(N'IS168901', N'Summer2022b', 5, N'PASSED')
INSERT [dbo].[View] ([StudentID], [SemesterID], [Avarage], [Status]) VALUES
(N'SE143042', N'Summer2022a', 4.25, N'NOT PASSED')
INSERT [dbo].[View] ([StudentID], [SemesterID], [Avarage], [Status]) VALUES
(N'SE149001', N'Summer2022a', 8.1, N'PASSED')

```

VII. Các câu lệnh Query

1. INDEX

```
-- INDEX  
CREATE INDEX [FullName] ON Students([FirstName],  
[LastName])
```

2.VIEW

3.STORE-PROCEDURE

4.TRIGGER

5.10 câu query

5.1 A. query that uses ORDER BY

- Sắp xếp Student theo StudentID

```
Select * from Students  
order by StudentID
```

- Kết quả:

	StudentID	FirstName	LastName	Gender	DateOfBirth	Address	Mail
1	HE132932	Nguyễn Khuyến	Học	1	1999-07-19	Tuyên Quang	HocNKHE132932@fpt.edu.vn
2	HE156721	Lý Đình	Huong	0	2001-03-17	Tuyên Quang	HuongLDHE156721@fpt.edu.vn
3	HE159082	Đoàn Công	Lộc	1	2001-01-15	Cần Thơ	LocDCHE159082@fpt.edu.vn
4	HE163780	Vương Đình	Ngân	1	2002-12-02	Hà Nội	NganVDHE163780@fpt.edu.vn
5	HE173690	Ngữ Băng	Yên	0	2003-06-07	Hà Nội	YenNBHE173690@fpt.edu.vn
6	HE178013	Trần Minh	Hiếu	1	2003-08-23	Phủ Thọ	HieuTMHE179013@fpt.edu.vn
7	HE179012	Vương Hồng	Châu	0	2003-01-02	Bắc Ninh	ChauVHHE179012@fpt.edu.vn
8	HE179013	Trần Minh	Hiếu	1	2003-08-23	Phủ Thọ	HieuTMHE179013@fpt.edu.vn
9	IS149076	Nguyễn Quang	Thạch	1	2000-12-19	Bắc Kạn	ThachNQIS149076@fpt.edu.vn
10	IS159011	Đỗ Thế	Cường	1	2001-12-09	Bắc Ninh	CuongDTIS159200@fpt.edu.vn
11	IS159200	Tân Linh	Đan	0	2001-10-10	Vũng Tàu	DanTLIS159200@fpt.edu.vn
12	IS168901	Đặng Bội	Linh	0	2002-08-04	Đà Nẵng	LinhDBIS168901@fpt.edu.vn
13	IS176801	Đỗ Trường	Sinh	1	2003-07-19	Hải Phòng	SinhDTIS175801@fpt.edu.vn
14	SE143042	Tăng Chí	Khiêm	1	2000-09-08	Cao Bằng	KhiemTCSE143042@fpt.edu.vn
15	SE149001	Tổ Đào	Vân	0	2000-09-08	Hòa Bình	VanTDSE149001@fpt.edu.vn
16	SE162847	Phạm Hưởng	Tiền	1	2002-11-19	Phủ Thọ	TienPHSE162847@fpt.edu.vn

5.2. A query that uses INNER JOINS

- In ra tất cả FullName của học sinh có điểm trung bình trong bảng View

```
Select distinct CONCAT_WS(
',s.FirstName, s.LastName) from
Students s
Inner join [View] v on v.StudentID
= s.StudentID
```

- Kết quả:

	FullName
1	Đặng Bội Linh
2	Đoàn Công Lộc
3	Lý Đình Hương
4	Nguyễn Khuyến Học
5	Nguyễn Quang Thạch
6	Tần Linh Đan
7	Tăng Chí Khiêm
8	Tổ Đào Văn
9	Vương Đình Ngân

- Mục đích: Kiểm tra xem có bao nhiêu học sinh có điểm kiểm tra

5.3. A query that uses aggregate functions

- In ra tổng số Category có trong Category

```
Select COUNT(CategoryID) as Total
from Category
```

- Kết quả:

	Total
1	17

- Mục đích: Kiểm tra tổng Category có trong Database

5.4. A query that uses the GROUP BY and HAVING clauses

- In ra tổng số học sinh nam và nữ

```
Select COUNT(s.StudentID), gender  
from Students s  
Group by Gender  
Having COUNT(s.StudentID) > 2
```

- Kết quả:

	Total	gender
1	6	0
2	10	1

- Mục đích: Phân loại, tổng hợp số học sinh theo từng giới tính nhanh hơn

5.5. A query that uses a sub-query as a relation

5.6. A query that uses a sub-query in the WHERE clause

- In ra tất cả các bạn có giới tính là nữ trong bảng Students

```
select * from Students
where Gender in (Select Gender
from Students where Gender = 0)
```

- Kết quả:

	StudentID	FirstName	LastName	Gender	DateOfBirth	Address	Mail
1	HE156721	Lý Đình	Huong	0	2001-03-17	Tuyên Quang	HuongLDHE156721@fpt.edu.vn
2	HE173690	Ngữ Băng	Yên	0	2003-06-07	Hà Nội	YenNBHE173690@fpt.edu.vn
3	HE179012	Vương Hồng	Châu	0	2003-01-02	Bắc Ninh	ChauVHHE179012@fpt.edu.vn
4	IS159200	Tân Linh	Đan	0	2001-10-10	Vũng Tàu	DanTLIS159200@fpt.edu.vn
5	IS168901	Đặng Bội	Linh	0	2002-08-04	Đà Nẵng	LinhDBIS168901@fpt.edu.vn
6	SE149001	Tổ Đào	Vân	0	2000-09-08	Hòa Bình	VanTDSE149001@fpt.edu.vn

- Mục đích: Truy vấn thông tin các bạn nữ nhanh chóng

5.7. A query that uses partial matching in the WHERE clause

- In ra tất cả thông tin học sinh có chữ “H”

```
Select * from Students
```



```
where LastName like 'H%'
```

- Kết quả:

	StudentID	FirstName	LastName	Gender	DateOfBirth	Address	Mail
1	HE132932	Nguyễn Khuyến	Học	1	1999-07-19	Tuyên Quang	HocNKHE132932@fpt.edu.vn
2	HE156721	Lý Đình	Huong	0	2001-03-17	Tuyên Quang	HuongLDHE156721@fpt.edu.vn
3	HE178013	Trần Minh	Hiếu	1	2003-08-23	Phú Thọ	HieuTMHE179013@fpt.edu.vn
4	HE179013	Trần Minh	Hiếu	1	2003-08-23	Phú Thọ	HieuTMHE179013@fpt.edu.vn

- Mục tiêu: Sàng lọc nhanh học sinh có bảng chữ cái

5.8. A query that uses a self-JOIN

- Tìm tên sinh viên có tuổi nhỏ nhất

- Kết quả:

VIII. TỔNG KẾT

Như vậy, SQL (viết tắt của Structured Query Language) là hệ thống ngôn ngữ được sử dụng cho các hệ quản trị cơ sở dữ liệu quan hệ. Thông qua SQL có thể thực hiện được các thao tác trên cơ sở dữ liệu như định nghĩa dữ liệu, thao tác dữ liệu, điều khiển truy cập, quản lý toàn vẹn dữ liệu...

SQL là một thành phần quan trọng và không thể thiếu trong hệ quản trị cơ sở dữ liệu quan hệ. SQL ra đời nhằm sử dụng cho các cơ sở dữ liệu theo mô hình quan hệ.

Trong một cơ sở dữ liệu quan hệ, dữ liệu được tổ chức và lưu trữ trong các bảng. Mỗi một bảng là một tập hợp bao gồm các dòng và các cột; mỗi một dòng là một bản ghi và mỗi một cột tương ứng với một trường, tập các tên cột cùng với kiểu dữ liệu và các tính chất khác tạo nên cấu trúc của bảng, tập các dòng trong bảng chính là dữ liệu của bảng

Các bảng trong một cơ sở dữ liệu có mối quan hệ với nhau. Các mối quan hệ được biểu diễn thông qua khoá chính và khoá ngoài của các bảng. Khoá chính của bảng là tập một hoặc nhiều cột có giá trị duy nhất trong bảng và do đó giá trị của nó xác định duy nhất một dòng dữ liệu trong bảng. Một khoá ngoài là một tập một hoặc nhiều cột có giá trị được xác định từ khoá chính của các bảng khác.