

KẾT NỐI CSDL SỬ DỤNG ENTITY FRAMEWORK

Biên soạn: **ThS. Nguyễn Thị Anh Thư**

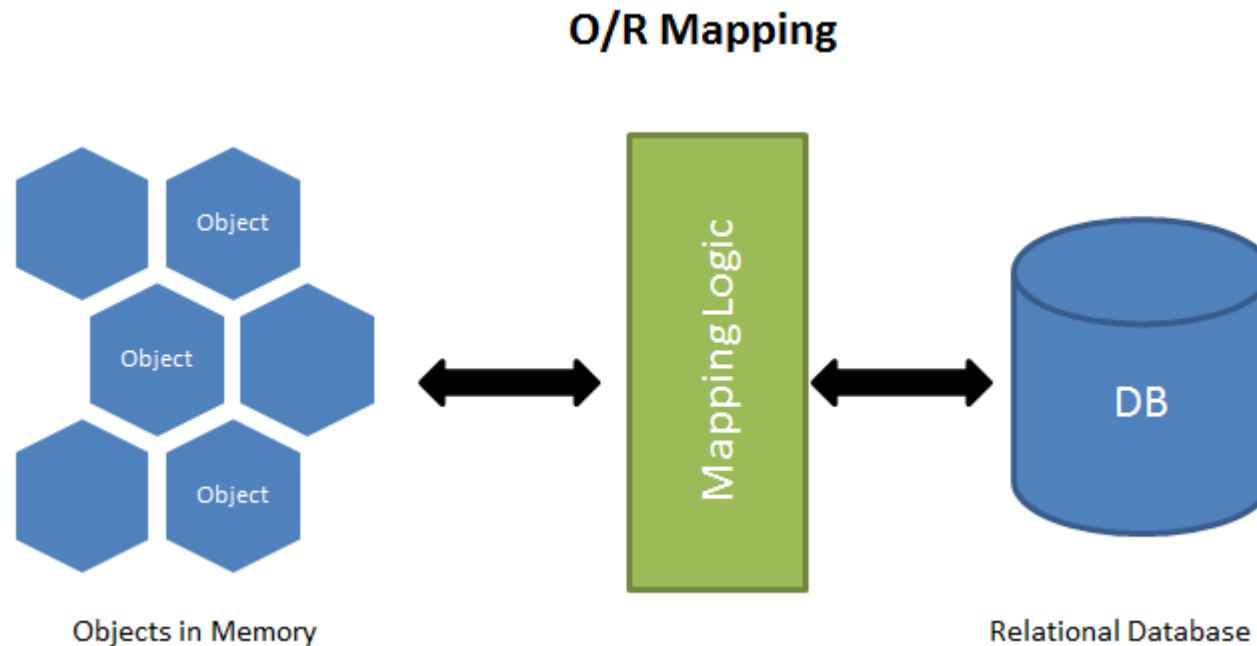
1

NỘI DUNG

1. Entity Framework là gì?
2. ADO.NET và Entity Framework
3. Tại sao nên sử dụng Entity Framework?
4. Tính năng của Entity Framework
5. Kiến trúc Entity Framework
6. Sử dụng Entity Framework

1. ENTITY FRAMEWORK LÀ GÌ?

- **Entity Framework (EF)** được phát hành lần đầu tiên vào năm 2008, là phương tiện tương tác chính giữa các ứng dụng .NET và cơ sở dữ liệu quan hệ.
- **EF là một ánh xạ quan hệ đối tượng (ORM)**, là một loại công cụ giúp đơn giản hóa ánh xạ giữa các đối tượng trong phần mềm với các bảng và cột của CSDL quan hệ.



1. ENTITY FRAMEWORK LÀ GÌ?

Đặc điểm của EF:

- EF là một framework ORM mã nguồn mở độc lập với .NET Framework.
- EF đảm nhiệm việc tạo các kết nối CSDL và thực thi các lệnh, cũng như lấy kết quả truy vấn và tự động ánh xạ các kết quả đó thành các đối tượng trong ứng dụng.
- EF giúp theo dõi các thay đổi của đối tượng và cập nhật các thay đổi đó trở lại CSDL.

2. ADO.NET VÀ ENTITY FRAMEWORK

- **ADO.NET và Entity Framework là các công nghệ ứng dụng .NET giao tiếp với CSDL.**
- Các loại ứng dụng phát triển trên nền tảng .NET của Microsoft sử dụng ADO.NET để làm việc với CSDL. EF là một ORM xây dựng bên trên ADO.NET để tiếp tục đơn giản hóa lập trình với CSDL.
- ADO.NET và Entity Framework không phải là 2 công nghệ độc lập. **EF được xây dựng bên trên ADO.NET, sử dụng một bộ phận (Data Provider) của ADO.NET.**

2. ADO.NET VÀ ENTITY FRAMEWORK

- ADO.NET (*ActiveX Data Object*) là tập hợp các thư viện lớp qua đó cho phép ứng dụng tương tác (*lấy về, cập nhật, xóa*) với các nguồn dữ liệu (*SQLServer, XML, MySQL, Oracle Database, ...*).
 - **Phần thứ nhất của ADO.NET gọi là Data Provider:** là các thư viện lớp cung cấp chức năng tạo kết nối đến nguồn dữ liệu, thi hành các lệnh trên nguồn dữ liệu đó insert, update, delete, read.
 - **SQL Server:** Loại Data Provider mặc định trong .NET Core là **SqlClient** ở namespace **System.Data.SqlClient** cung cấp khả năng kết nối đến SQL Server.
 - **Hướng dẫn mô hình 3 layer ADO.NET và LINQ:** [Link](#).
- Entity Framework được sử dụng đặc biệt phổ biến trong phát triển ứng dụng .NET thay cho lập trình trực tiếp với ADO.NET.
 - EF giúp quá trình làm việc với CSDL trở nên dễ dàng hơn.
 - EF làm cho chương trình chạy chậm hơn so với việc lập trình trực tiếp với ADO.NET.

3. TẠI SAO NÊN SỬ DỤNG ENTITY FRAMEWORK?

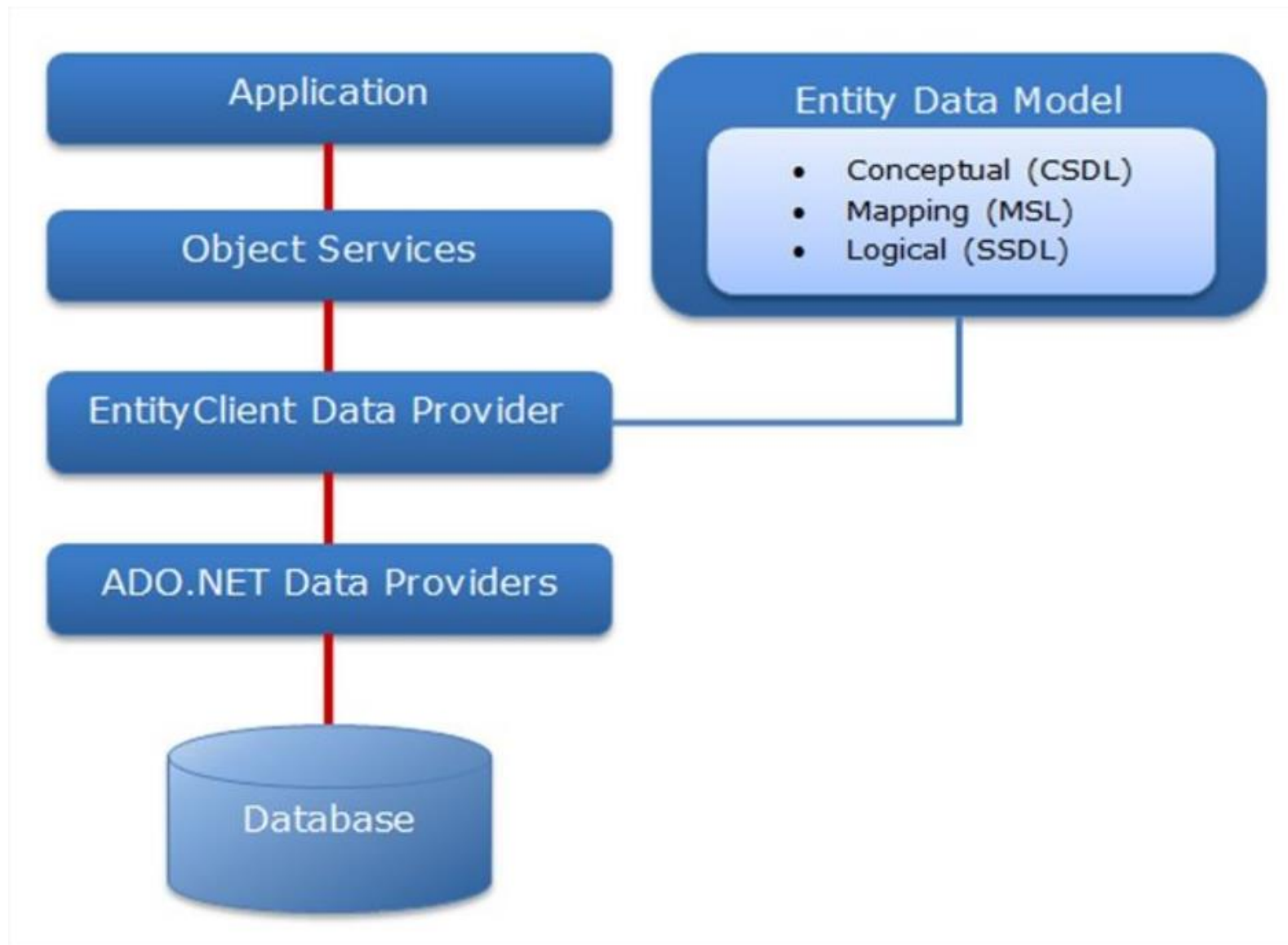
EF được thiết kế nhằm mục đích tăng năng suất của nhà phát triển bằng cách giảm các công việc dư thừa như là duy trì dữ liệu được sử dụng trong các ứng dụng.

- EF có thể tạo các câu lệnh truy vấn cần thiết để đọc hoặc ghi dữ liệu trong CSDL và thực thi chúng.
- **Có thể sử dụng LINQ to Entities để truy vấn dữ liệu.**
- EF sẽ thực hiện các truy vấn có liên quan trong CSDL và sau đó ánh xạ kết quả trả về thành các đối tượng trong ứng dụng.

4. TÍNH NĂNG CỦA ENTITY FRAMEWORK

- EF là một công cụ của Microsoft.
- EF được phát triển như một sản phẩm mã nguồn mở.
- Làm việc với bất kỳ CSDL nào có EF provider hợp lệ.
- Tạo truy vấn SQL từ LINQ to Entities.
- EF sẽ tạo các truy vấn có tham số.
- Theo dõi các thay đổi của các đối tượng trong bộ nhớ.
- Cho phép tạo các câu lệnh thêm, cập nhật và xóa.
- EF hỗ trợ stored procedure.

5. KIẾN TRÚC ENTITY FRAMEWORK



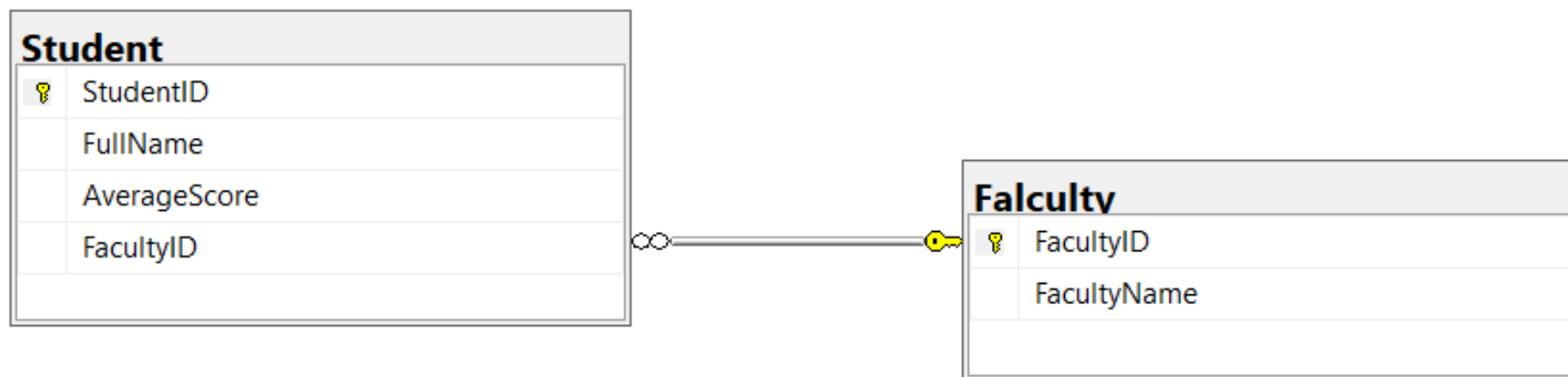
6. SỬ DỤNG ENTITY FRAMEWORK

- **Bước 1:** Tạo CSDL.
- **Bước 2:** Tạo chương trình ứng dụng quản lý cho phép người dùng xem, thêm, xóa, sửa.
- **Bước 3:** Add ADO.NET Entity Data Model vào project.
- **Bước 4:** Chọn loại Models muốn sử dụng.
 - **Data First:** Tạo một mô hình trong EF Designer dựa trên cơ sở dữ liệu hiện có. Bạn có thể chọn kết nối cơ sở dữ liệu, cài đặt cho mô hình và các đối tượng cơ sở dữ liệu để đưa vào mô hình. Các lớp mà ứng dụng của bạn sẽ tương tác được tạo ra từ mô hình.
 - **Code First:** Tạo mô hình Code First dựa trên cơ sở dữ liệu hiện có. Bạn có thể chọn kết nối cơ sở dữ liệu, cài đặt cho mô hình và các đối tượng cơ sở dữ liệu để đưa vào mô hình.
 - **Empty:** Tạo mô hình trống.

6. SỬ DỤNG ENTITY FRAMEWORK

Ví dụ:

- **Bước 1:** Tạo CSDL.



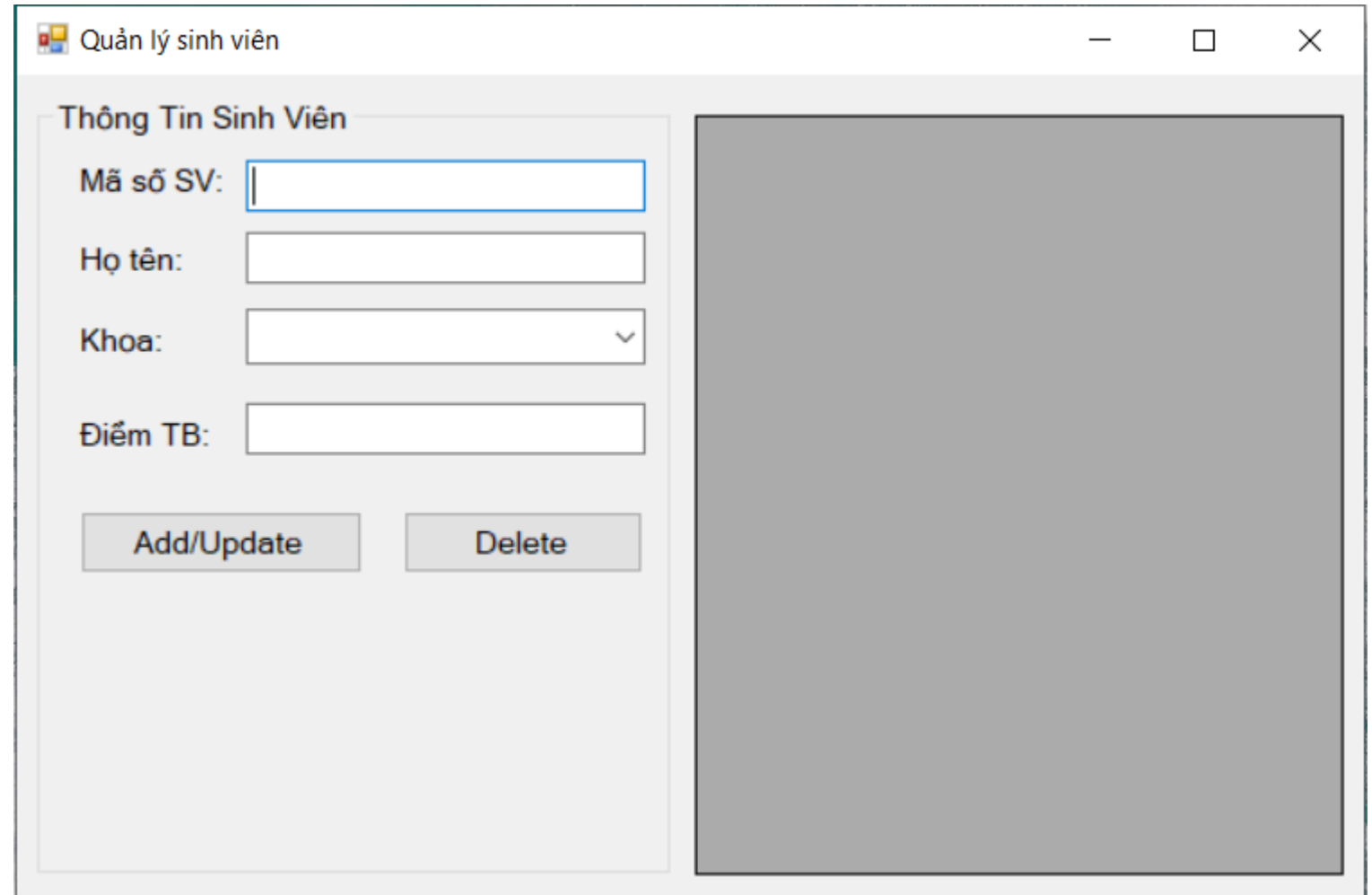
	StudentID	FullName	AverageScore	FacultyID
1	1	Nguyễn Thị Anh Thư	9	1
2	2	Nguyễn Quang Phúc	9.2	1
3	3	Võ Tấn Khoa	9.5	1
4	4	Tạ Thu Thủy	8	2

	FacultyID	FacultyName
1	1	Công nghệ thông tin
2	2	Khoa học máy tính

6. SỬ DỤNG ENTITY FRAMEWORK

Ví dụ:

- **Bước 2:** Tạo chương trình ứng dụng quản lý cho phép người dùng xem, thêm, xóa, sửa.

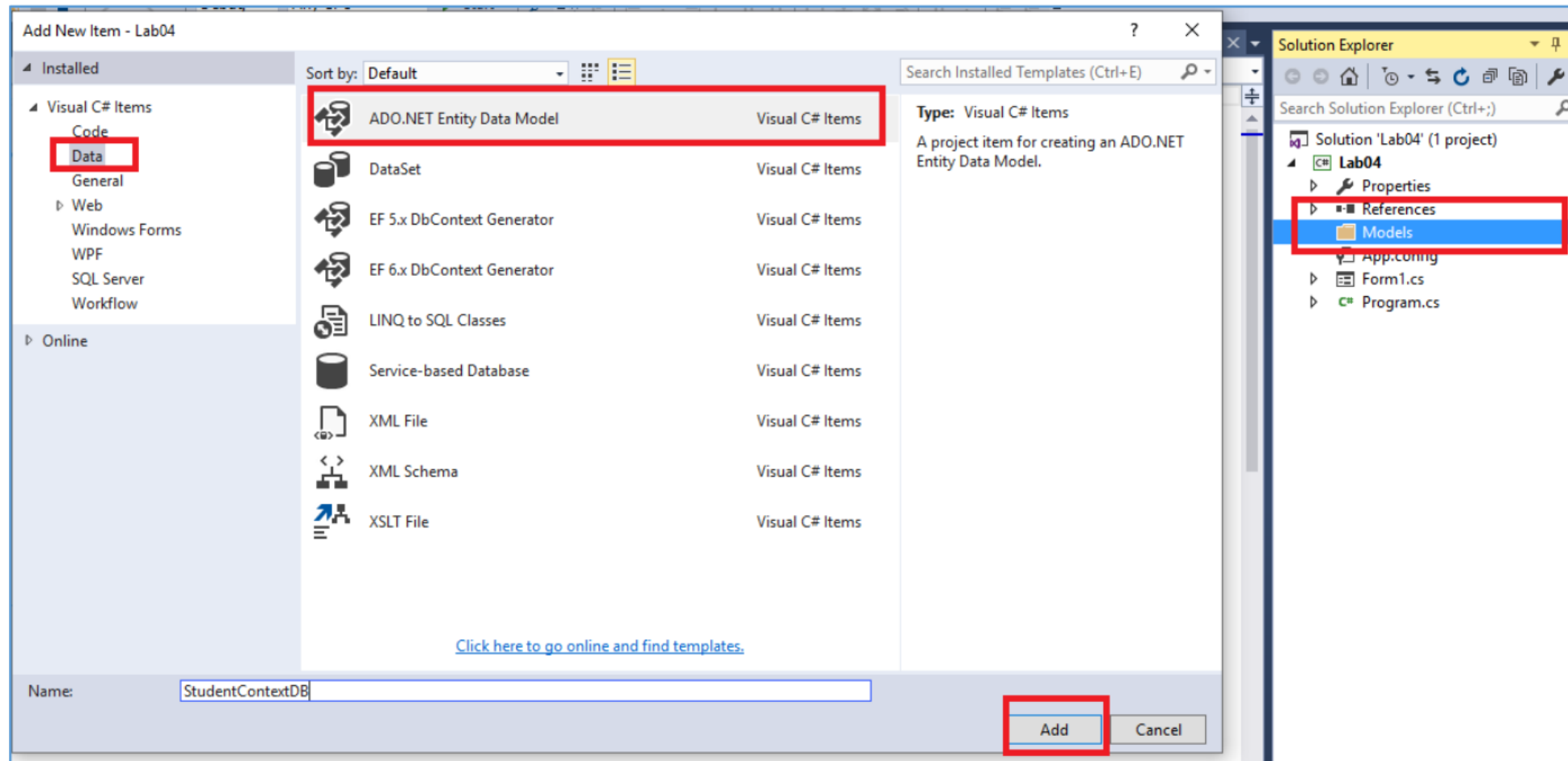


The screenshot shows a Windows application window titled "Quản lý sinh viên" (Student Management). The window is divided into two main sections. The left section, titled "Thông Tin Sinh Viên" (Student Information), contains four input fields: "Mã số SV:" (Student ID), "Họ tên:" (Full Name), "Khoa:" (Faculty), and "Điểm TB:" (Average Score). Below these fields are two buttons: "Add/Update" and "Delete". The right section is a large, empty gray rectangular area, likely intended for displaying a list of students or a detailed view.

6. SỬ DỤNG ENTITY FRAMEWORK

Ví dụ:

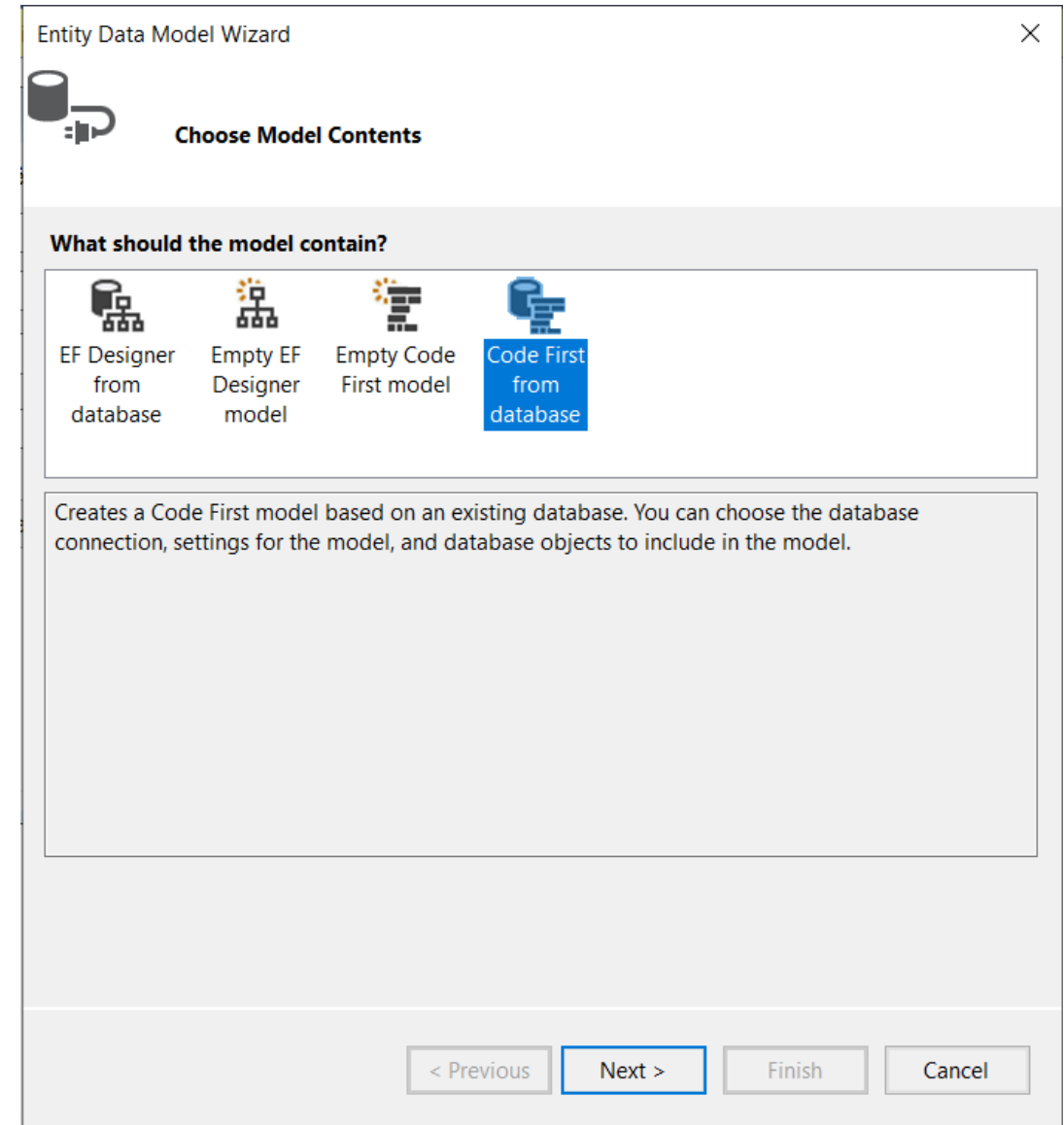
- **Bước 3:** Add ADO.NET Entity Data Model vào project.



6. SỬ DỤNG ENTITY FRAMEWORK

Ví dụ:

- **Bước 4:** Chọn loại Models muốn sử dụng.
 - **Code First:** Tạo mô hình Code First dựa trên cơ sở dữ liệu hiện có. Bạn có thể chọn kết nối cơ sở dữ liệu, cài đặt cho mô hình và các đối tượng cơ sở dữ liệu để đưa vào mô hình.



6. SỬ DỤNG ENTITY FRAMEWORK

Ví dụ:

- Copy **Server name** từ SQL Server vào.
- Chọn **Database** muốn kết nối.

Connection Properties

Enter information to connect to the selected data source or click "Change" to choose a different data source and/or provider.

Data source:
Microsoft SQL Server (SqlClient) Change...

Server name:
(local) Refresh

Log on to the server

Authentication: Windows Authentication

User name:

Password:

☐ Save my password

Connect to a database

☒ Select or enter a database name:
QuanLySinhVien

☐ Attach a database file:
 Browse...

Logical name:

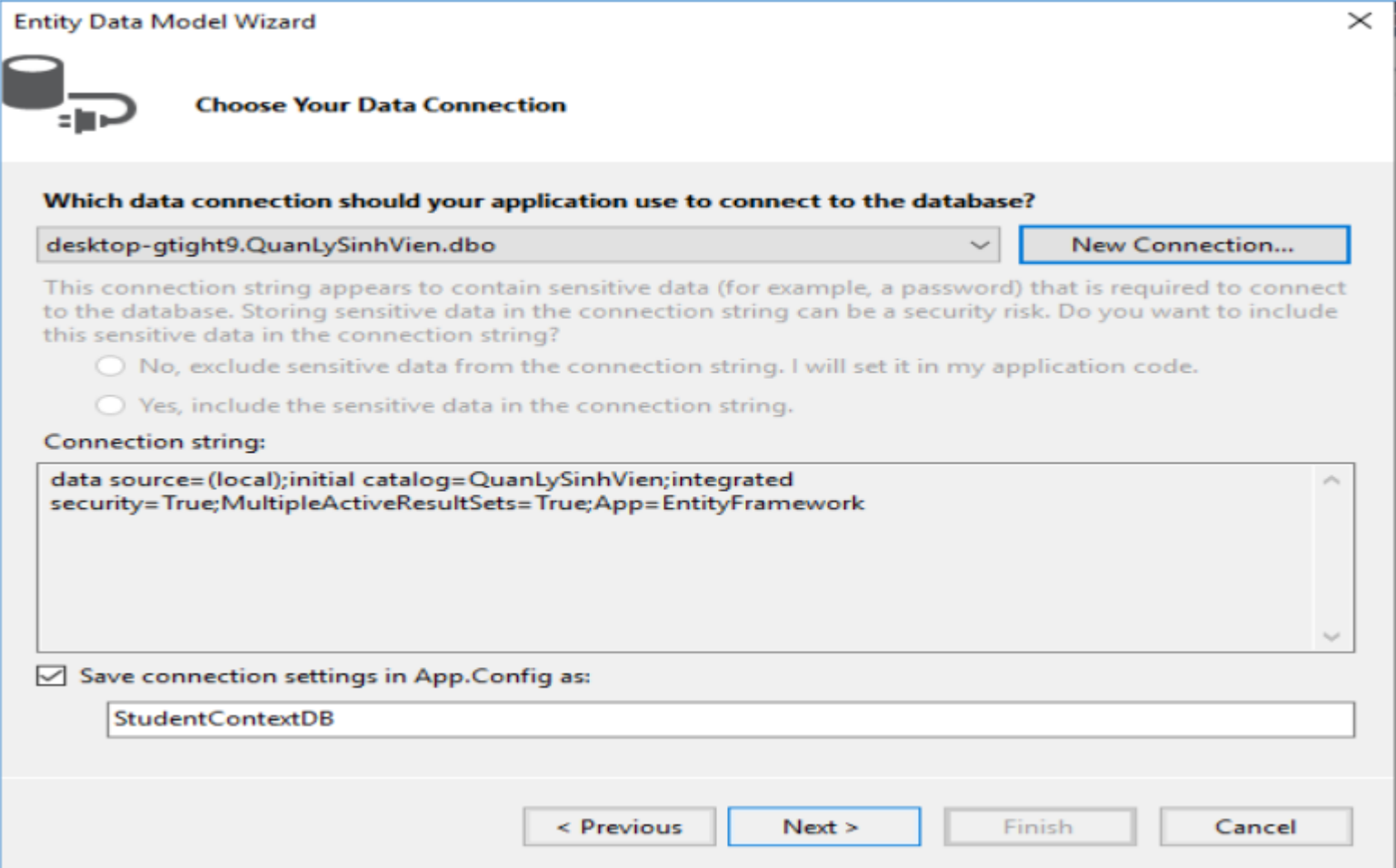
Advanced...

Test Connection OK Cancel

6. SỬ DỤNG ENTITY FRAMEWORK

Ví dụ:

- Copy **Server name** từ SQL Server vào.
- Chọn **Database** muốn kết nối.



The image shows the 'Entity Data Model Wizard' dialog box, specifically the 'Choose Your Data Connection' step. The title bar reads 'Entity Data Model Wizard'. Below the title bar is a sub-header 'Choose Your Data Connection' with a database icon. The main question is 'Which data connection should your application use to connect to the database?'. A dropdown menu shows 'desktop-gtight9.QuanLySinhVien.dbo'. To the right of the dropdown is a 'New Connection...' button. Below this, a warning message states: 'This connection string appears to contain sensitive data (for example, a password) that is required to connect to the database. Storing sensitive data in the connection string can be a security risk. Do you want to include this sensitive data in the connection string?'. There are two radio buttons: 'No, exclude sensitive data from the connection string. I will set it in my application code.' (selected) and 'Yes, include the sensitive data in the connection string.'. Below the radio buttons is a section labeled 'Connection string:' with a text box containing the string: 'data source=(local);initial catalog=QuanLySinhVien;integrated security=True;MultipleActiveResultSets=True;App=EntityFramework'. At the bottom, there is a checkbox 'Save connection settings in App.Config as:' which is checked. Below the checkbox is a text box containing 'StudentContextDB'. At the very bottom are four buttons: '< Previous', 'Next >', 'Finish', and 'Cancel'.

Entity Data Model Wizard

Choose Your Data Connection

Which data connection should your application use to connect to the database?

desktop-gtight9.QuanLySinhVien.dbo

New Connection...

This connection string appears to contain sensitive data (for example, a password) that is required to connect to the database. Storing sensitive data in the connection string can be a security risk. Do you want to include this sensitive data in the connection string?

☐ No, exclude sensitive data from the connection string. I will set it in my application code.

☐ Yes, include the sensitive data in the connection string.

Connection string:

data source=(local);initial catalog=QuanLySinhVien;integrated security=True;MultipleActiveResultSets=True;App=EntityFramework

☒ Save connection settings in App.Config as:

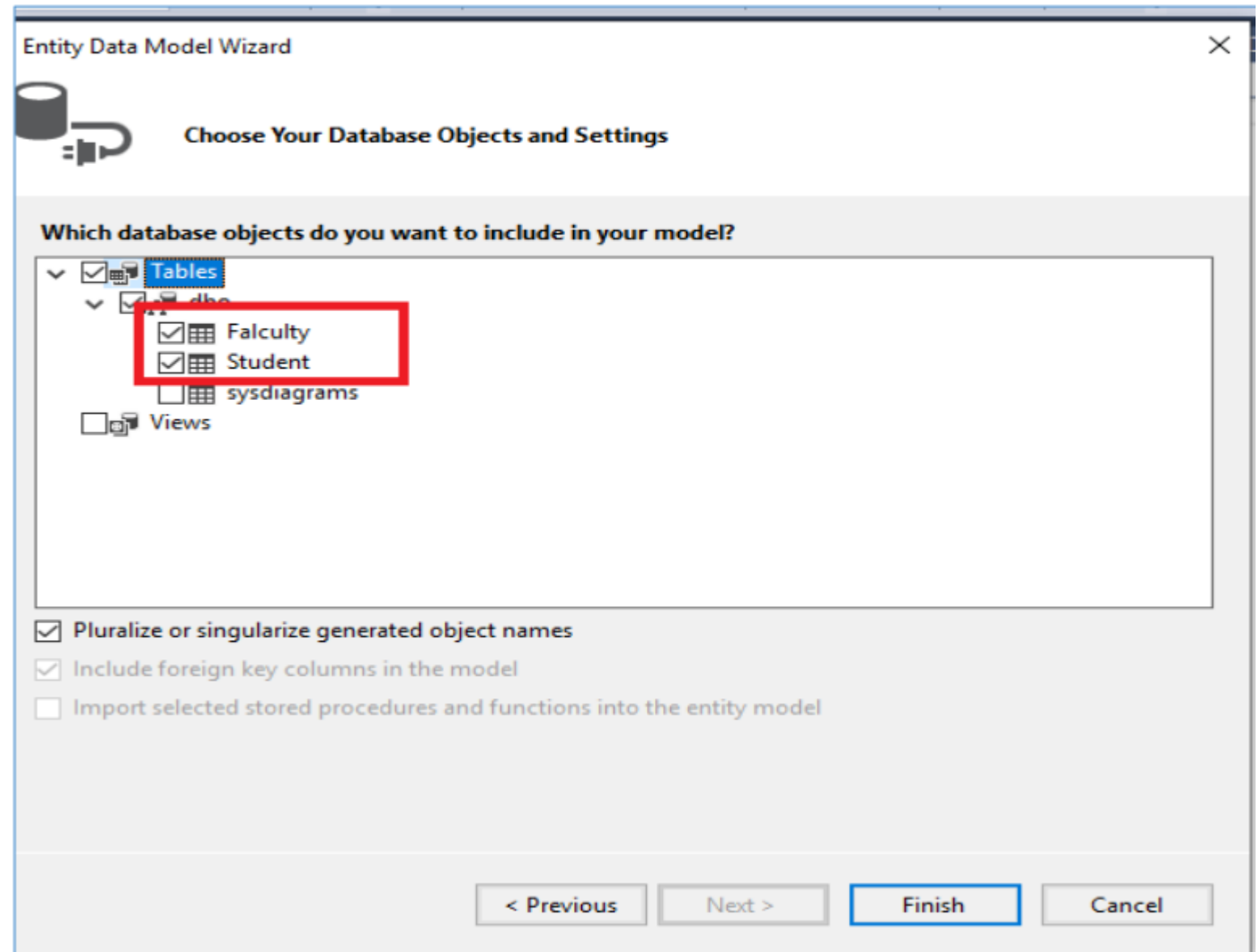
StudentContextDB

< Previous Next > Finish Cancel

6. SỬ DỤNG ENTITY FRAMEWORK

Ví dụ:

- Chọn bảng trong CSDL muốn tạo object trong project.



6. SỬ DỤNG ENTITY FRAMEWORK

Ví dụ:

- Sau khi tạo xong EF, chúng ta sẽ có:
 - **App.config**: Chứa chuỗi kết nối **connectionStrings**.
 - **StudentContextDB.cs**: Chứa tập hợp DataSet các table.
 - Các class tương ứng với các bảng trong CSDL.

6. SỬ DỤNG ENTITY FRAMEWORK

Ví dụ:

- Sử dụng các chức năng của công cụ EF:
 - Tạo đối tượng của class **StudentContextDB**.
 - Tương tác với CSDL thông qua đối tượng của class **StudentContextDB**.
- Code trong sự kiện của các controls trên giao diện.

6. SỬ DỤNG ENTITY FRAMEWORK

Ví dụ:

- Tạo sự kiện form load.
- Tạo đối tượng của class **StudentContextDB**.
- Tương tác với CSDL thông qua đối tượng của class **StudentContextDB**.

```
private void frmQLSinhVien_Load(object sender, EventArgs e)
{
    //Tạo đối tượng của class StudentContextDB
    StudentContextDB context = new StudentContextDB();
    //Lấy danh sách sinh viên từ CSDL thông qua đối tượng context
    List<Student> listStudents = context.Students.ToList();
    //Gán danh sách sinh viên vào controls cần tương tác
    dgvSinhVien.DataSource = listStudents;
}
```

6. SỬ DỤNG ENTITY FRAMEWORK

Ví dụ:

Quản lý sinh viên

Thông Tin Sinh Viên

Mã số SV:

Họ tên:

Khoa:

Điểm TB:

Add/Update Delete

	StudentID	FullName	AverageScore	Fa
▶	1	Nguyễn Thị Anh ...	9	1
	2	Nguyễn Quang P...	9.2	1
	3	Võ Tấn Khoa	9.5	1
	4	Tạ Thu Thủy	8	2

6. SỬ DỤNG ENTITY FRAMEWORK

Ví dụ:

```
private void frmQLSinhVien_Load(object sender, EventArgs e)
{
    //Tạo đối tượng của class StudentContextDB
    StudentContextDB context = new StudentContextDB();
    //Lấy danh sách sinh viên từ CSDL thông qua đối tượng context
    List<Student> listStudents = context.Students.ToList();
    //Gán danh sách sinh viên vào controls cần tương tác
    dgvSinhVien.DataSource = listStudents;

    List<Faculty> listFal = context.Faculties.ToList();
    cmbKhoa.DataSource = listFal;
    cmbKhoa.ValueMember = "FacultyID";
    cmbKhoa.DisplayMember = "FacultyName";
}

1 reference
private void dgvSinhVien_CellClick(object sender, DataGridViewCellEventArgs e)
{
    if (e.RowIndex != -1)
    {
        DataGridViewRow dgvRow = dgvSinhVien.Rows[e.RowIndex];
        txtMaSo.Text = dgvRow.Cells[0].Value.ToString();
        txtHoTen.Text = dgvRow.Cells[1].Value.ToString();
        txtDiem.Text = dgvRow.Cells[2].Value.ToString();
        cmbKhoa.SelectedValue = dgvRow.Cells[3].Value;
    }
}
```

6. SỬ DỤNG ENTITY FRAMEWORK

Ví dụ:

Quản lý sinh viên

Thông Tin Sinh Viên

Mã số SV:

Họ tên:

Khoa:

Điểm TB:

	StudentID	FullName	AverageScore	Fa
	1	Nguyễn Thị Anh ...	9	1
	2	Nguyễn Quang P...	9.2	1
	3	Võ Tấn Khoa	9.5	1
▶	4	Tạ Thu Thủy	8	2



Q & A