



LẬP TRÌNH C# 3

BÀI 5: ENTITY FRAMEWORK VÀ EF 6 DATABASE-FIRST

- Entity Framework
- EF 6 Database-First

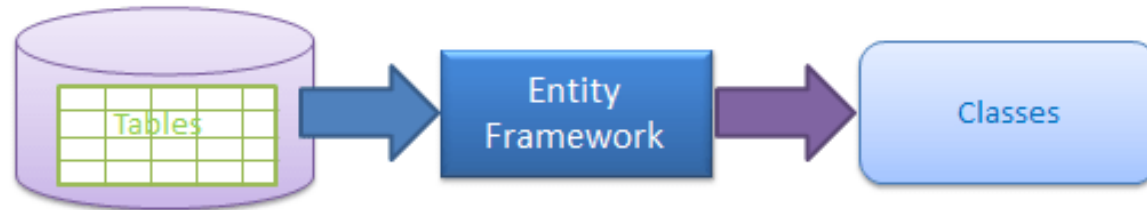


- ❑ Entity Framework là một nền tảng được sử dụng để làm việc với database thông qua cơ chế ánh xạ Object/Relational Mapping (ORM).
- ❑ Là framework open source phát triển bởi Microsoft
- ❑ Nó là một sự cải tiến tới ADO.NET và đưa cho các lập trình viên một cơ chế tự động cho việc truy xuất và sắp xếp dữ liệu trong CSDL
- ❑ Entity Framework đặt trong .NET Framework và tích hợp trong Visual Studio

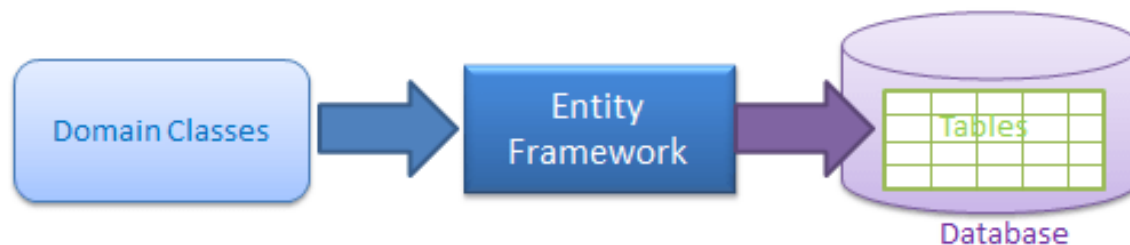
❑ Entity framework là rất hữu ích trong ba tình huống:

- ❖ Nếu bạn đã có một CSDL hoặc muốn thiết kế CSDL trước khi làm các phần khác của ứng dụng
- ❖ Muốn tập trung vào các domain class rồi mới tạo CSDL từ các domain class đó.
- ❖ Muốn thiết kế schema của CSDL trên visual designer rồi mới tạo CSDL và các class.

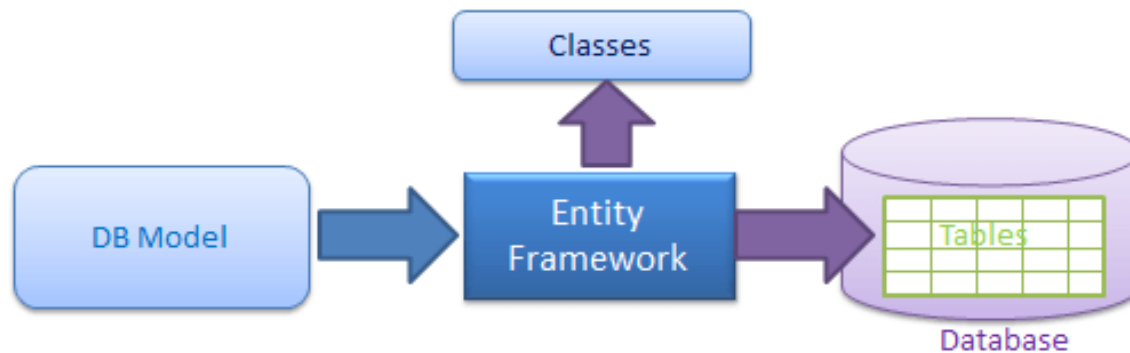
GIỚI THIỆU ENTITY FRAMEWORK



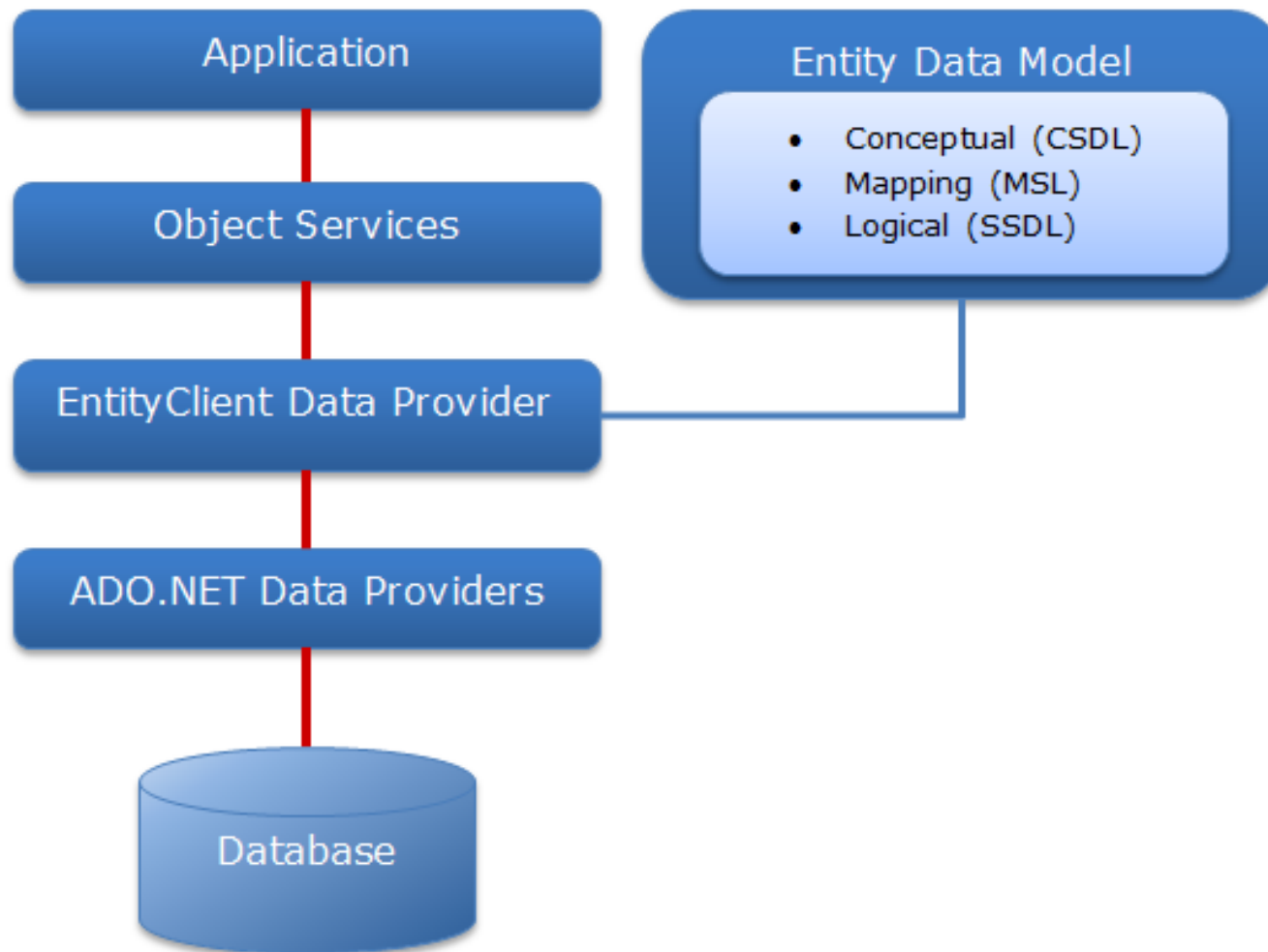
Generate Data Access Classes for Existing Database



Create Database from the Domain Classes

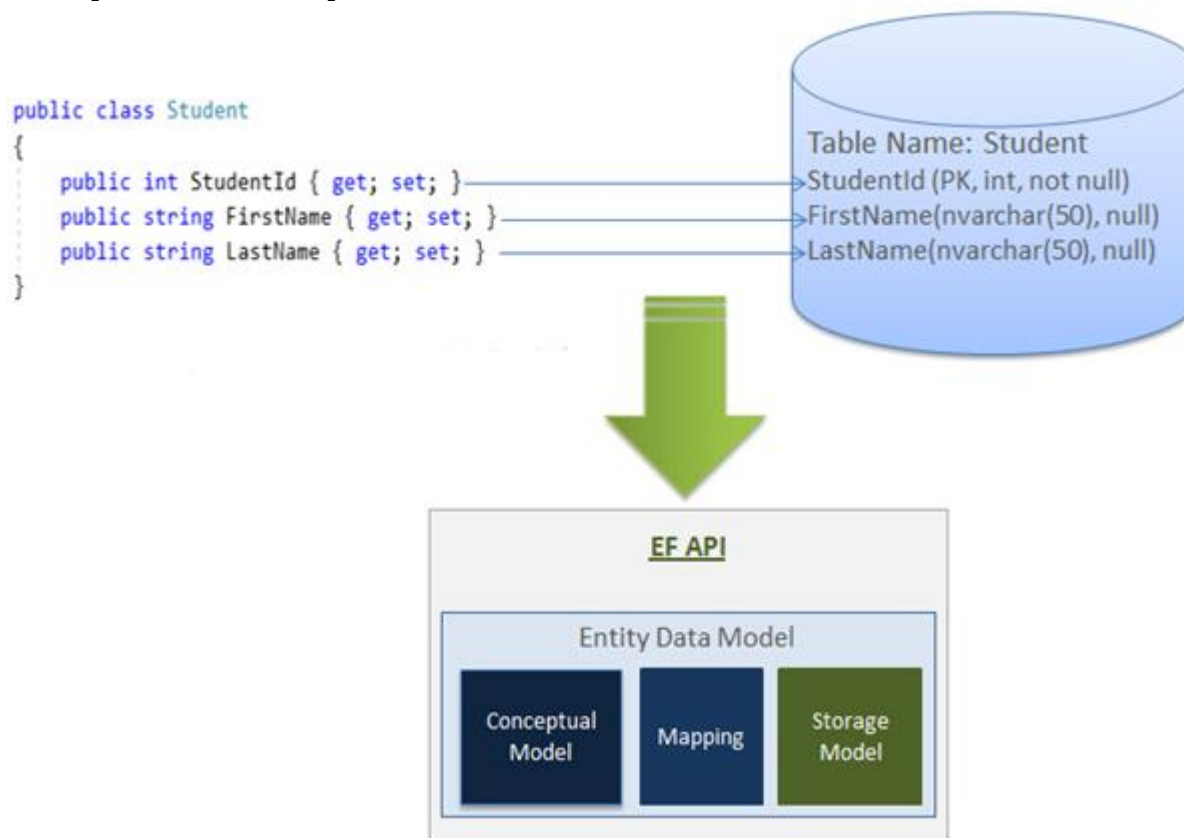


Create Database and Classes from the DB Model design



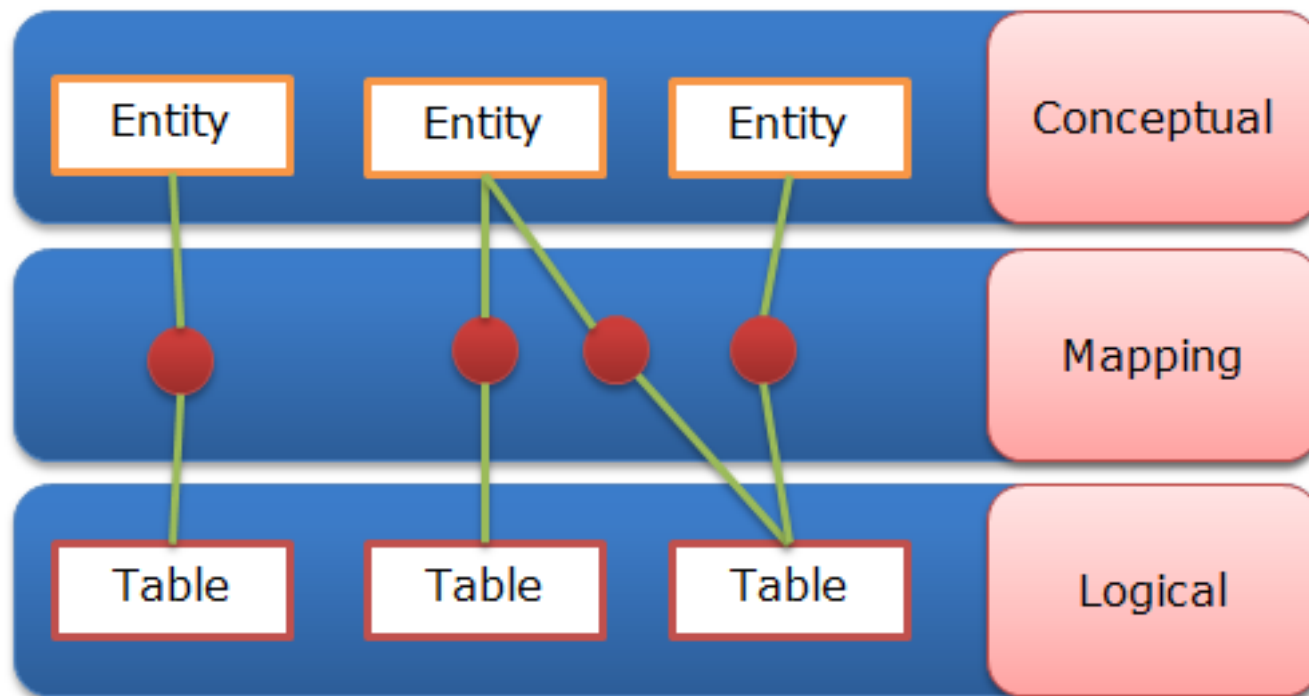
- ❑ Object Services: là các class tự động sinh ra tương ứng với mô hình dữ liệu. Các class này bao gồm:
 - ❖ObjectContext đại diện cho một database
 - ❖ObjectSet<TEntity> là một tập hợp các entity. Mỗi đối tượng này tương ứng với một table.
 - ❖EntityObject, ComplexObject là các lớp tương ứng cho một dòng dữ liệu của table trong database
 - ❖EntityCollection<TEntity> và EntityReference<TEntity>: là các đối tượng thể hiện mối quan hệ (relationship) giữa hai entity class. Mỗi đối tượng này có thể được truy xuất thông qua các property của entity class.

- Entity Data Model (EDM): là mô hình dữ liệu được mô tả thông qua các ngôn ngữ theo chuẩn XML. EDM được chia làm 3 lớp là: Conceptual, Mapping và Logical

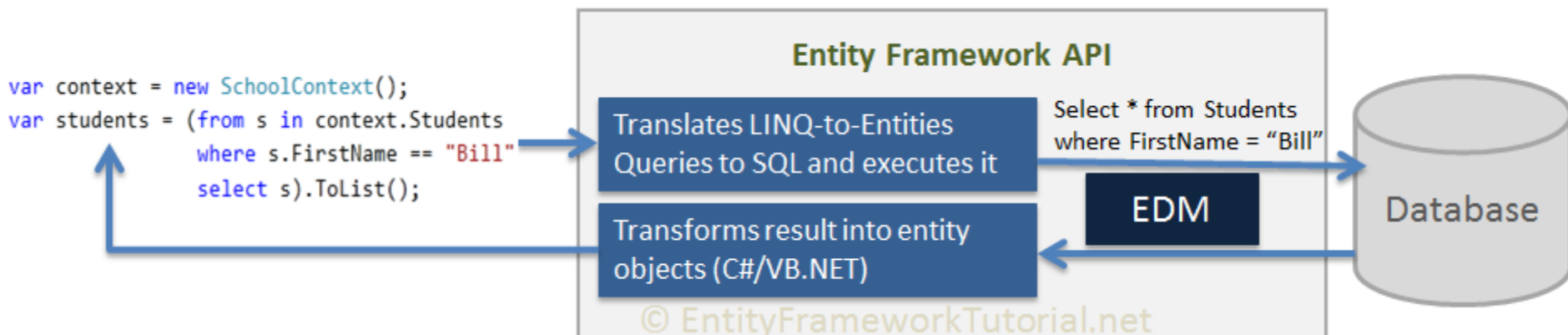


- ❑ Conceptual – Conceptual Schema Definition Language (CSDL): là ngôn ngữ định nghĩa các entity, relationship, hàm trong tập tin với phần mở rộng .csdl. Có thể tạo được các entity class (object layer).
- ❑ Mapping – Mapping specification language (MSL): định nghĩa các ánh xạ giữa lớp conceptual và logical, nội dung này được lưu trong tập tin .msl.
- ❑ Storage Model (SSDL): Lưu thông tin về csdl, bao gồm các table, view, proc, khóa, quan hệ...

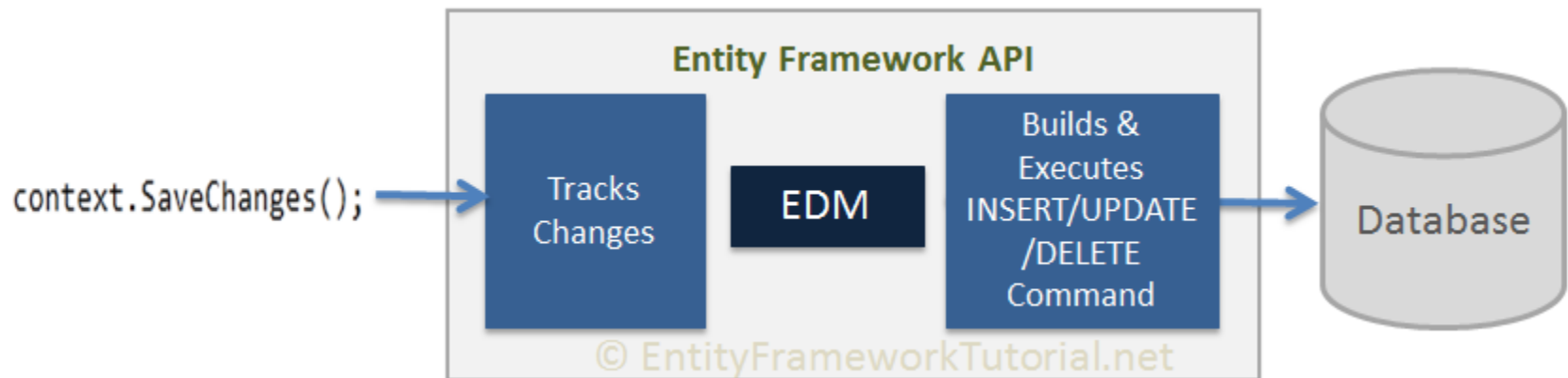
Entity Data Model



- ❑ Truy vấn dữ liệu (Querying): EF API chuyển đổi truy vấn LINQ-to-Entities sang truy vấn Sql bằng EMD và cũng chuyển đổi ngược lại dạng các thực thể



- ❑ Saving: sử dụng `SaveChanges()` tương tác các hành động INSERT, UPDATE, và DELETE

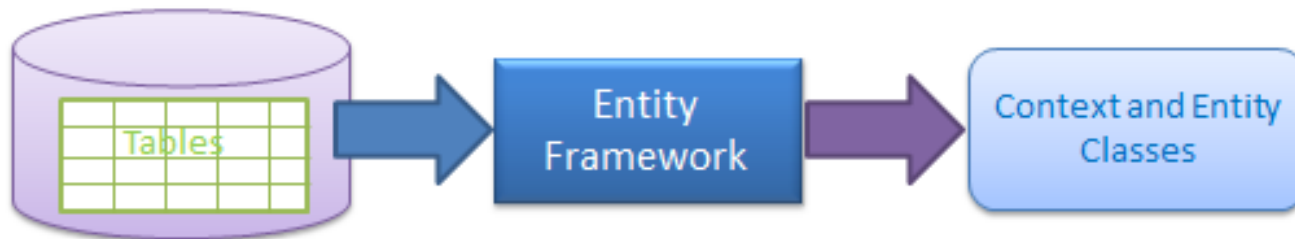


❑ Entity Client Data Provider:

- ❖ EntityClient là một data provider mới của ADO.NET dùng để truy xuất đến database
- ❖ Thực hiện chuyển đổi truy vấn Linq To Entities về truy vấn Sql
- ❖ EntityClient không truy xuất trực tiếp dữ liệu mà thông qua các data provider khác dựa vào các thông tin dữ liệu từ Entity Data Model.

❑ Database-First

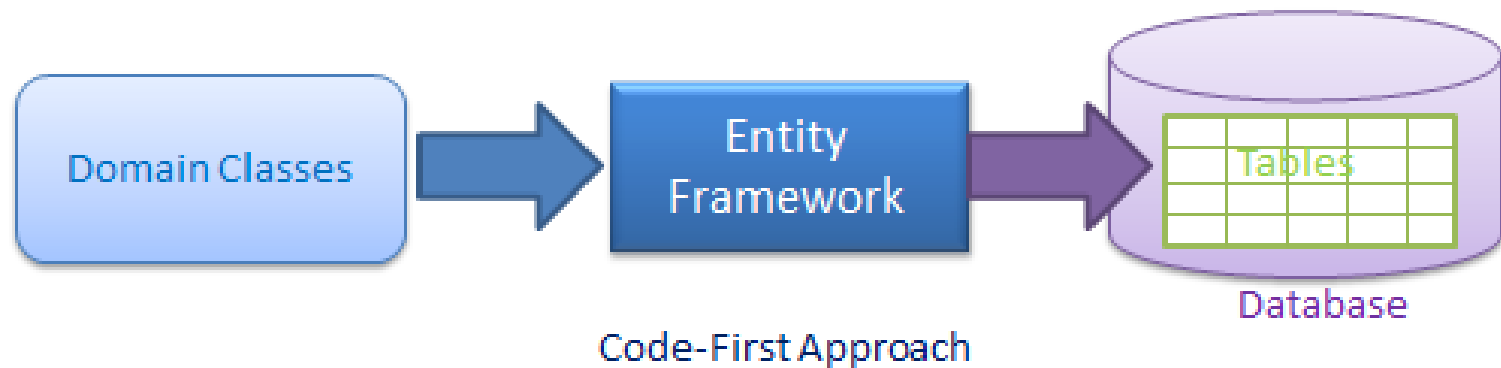
- ❖ Sinh ra model từ csdl có sẵn, từ stored procedures hoặc views
- ❖ Xuất phát từ Storage Model có sẵn, EF sinh ra Conceptual model và Mapping



Database-First Approach

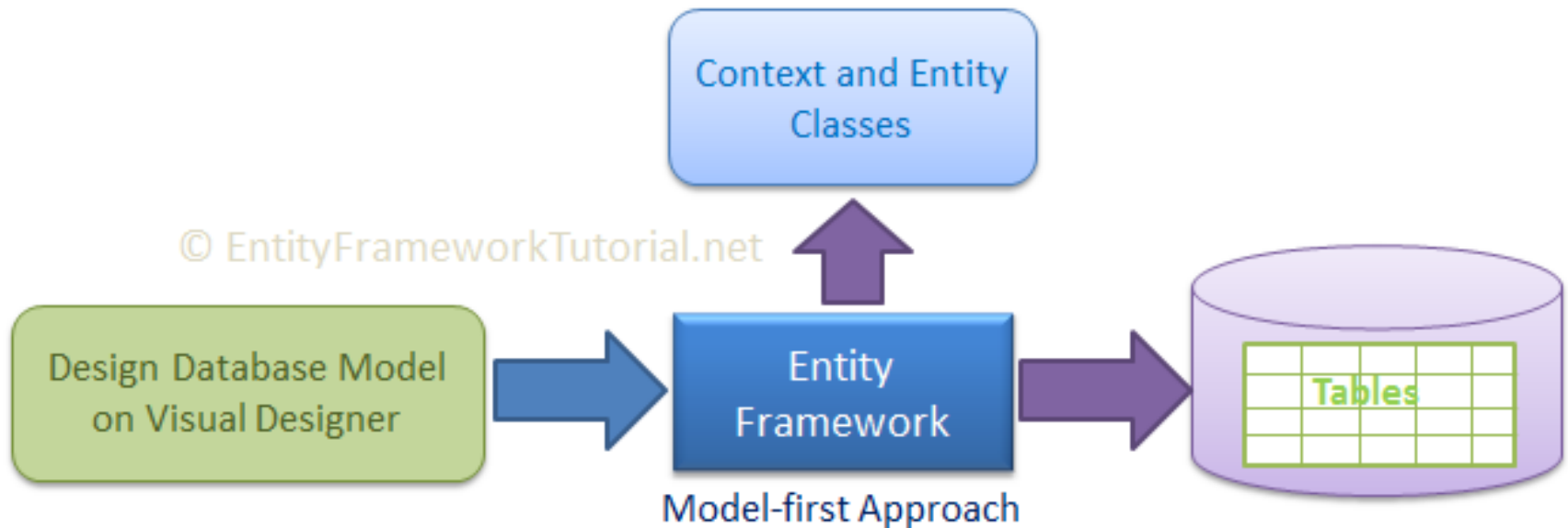
❑ Code-First

- ❖ Xây dựng các entities (domain classes) và context class trước, từ đó phát sinh database
- ❖ Tạo Conceptual model, EF sinh ra Mapping và Storage Model

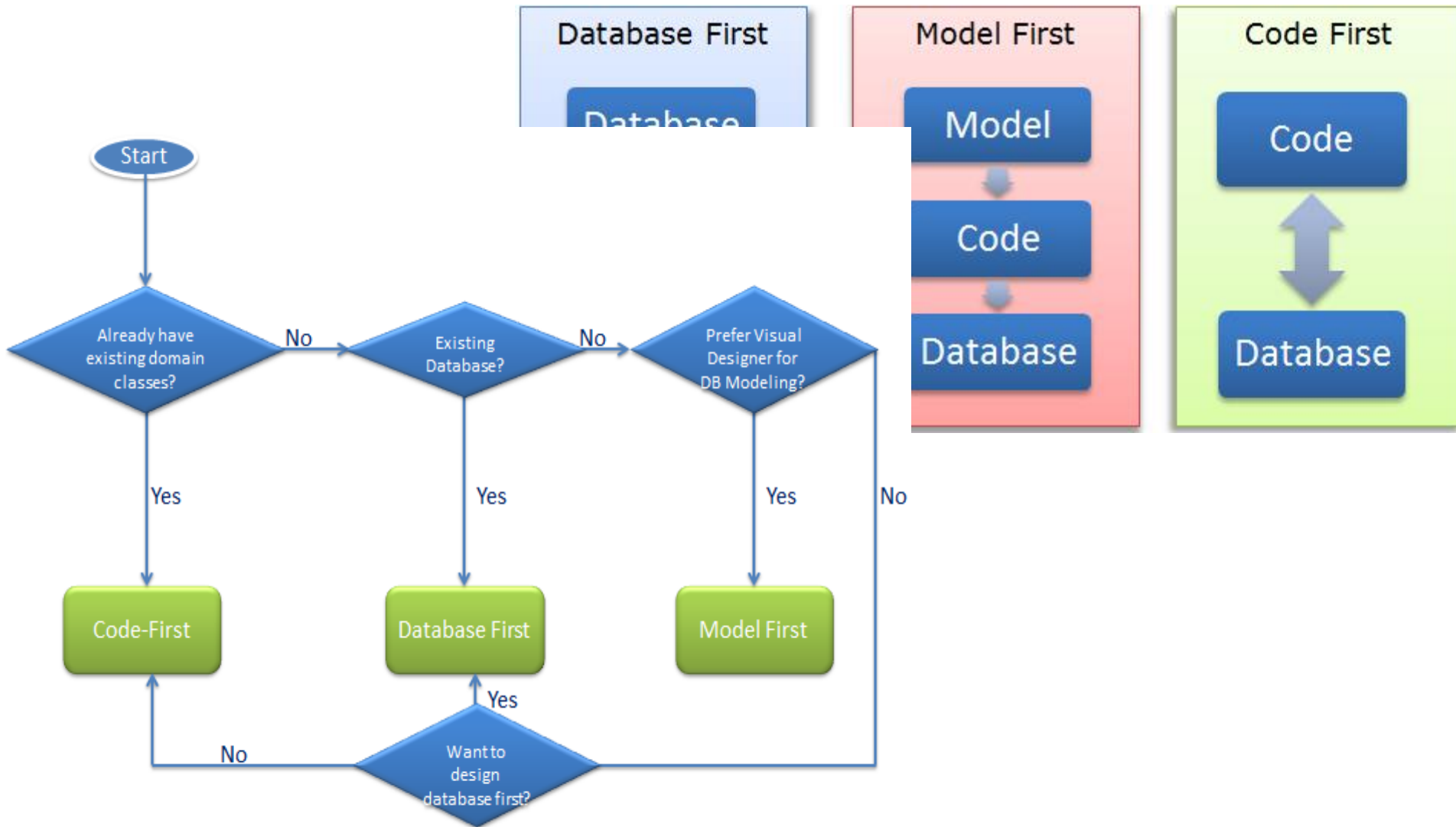


□ Model-First:

- ❖ Dùng giao diện đồ họa thiết kế các enties, relationships...sau đó phát sinh ra context class, database script

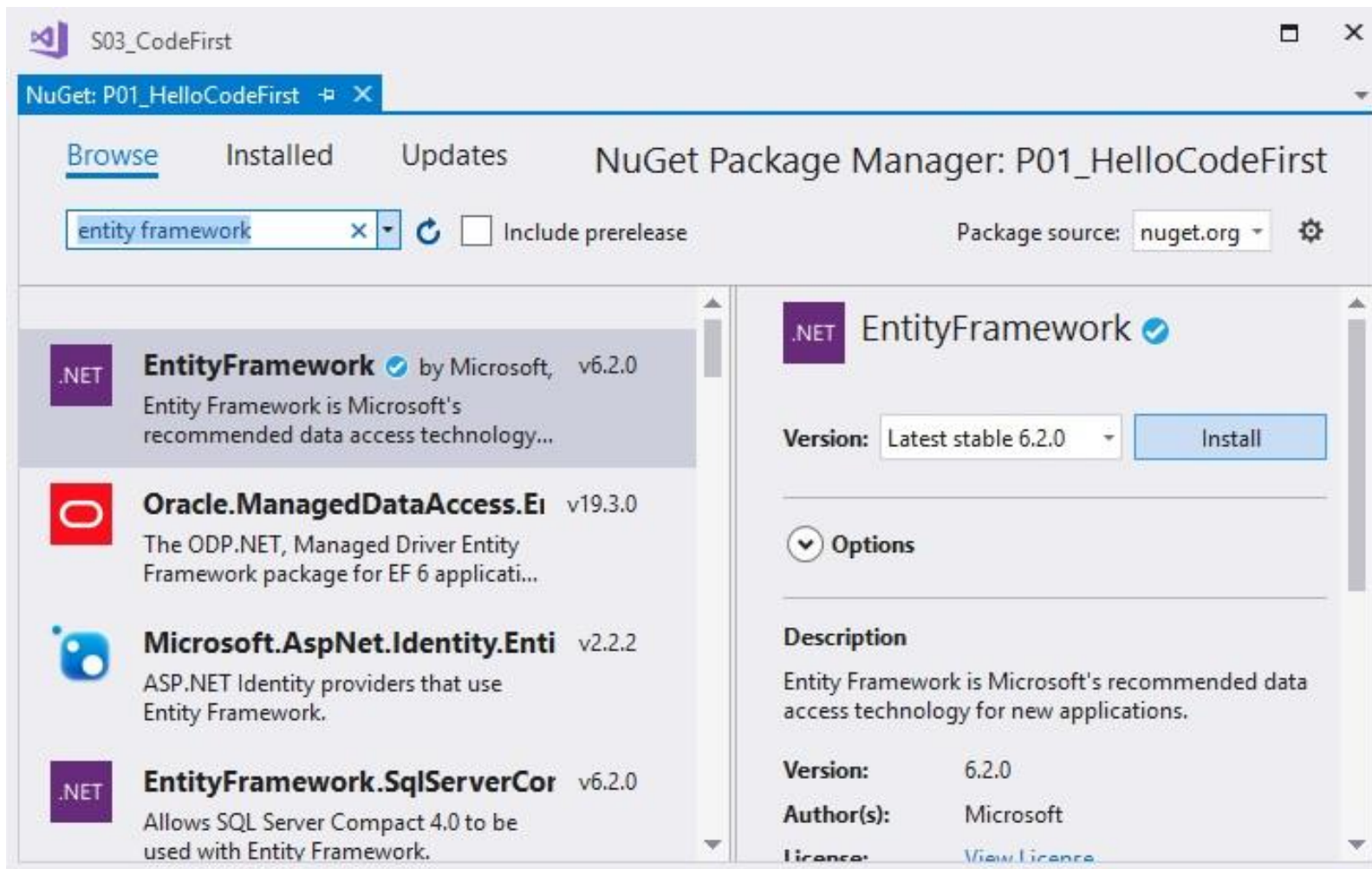


❑ Lựa chọn hướng tiếp cận



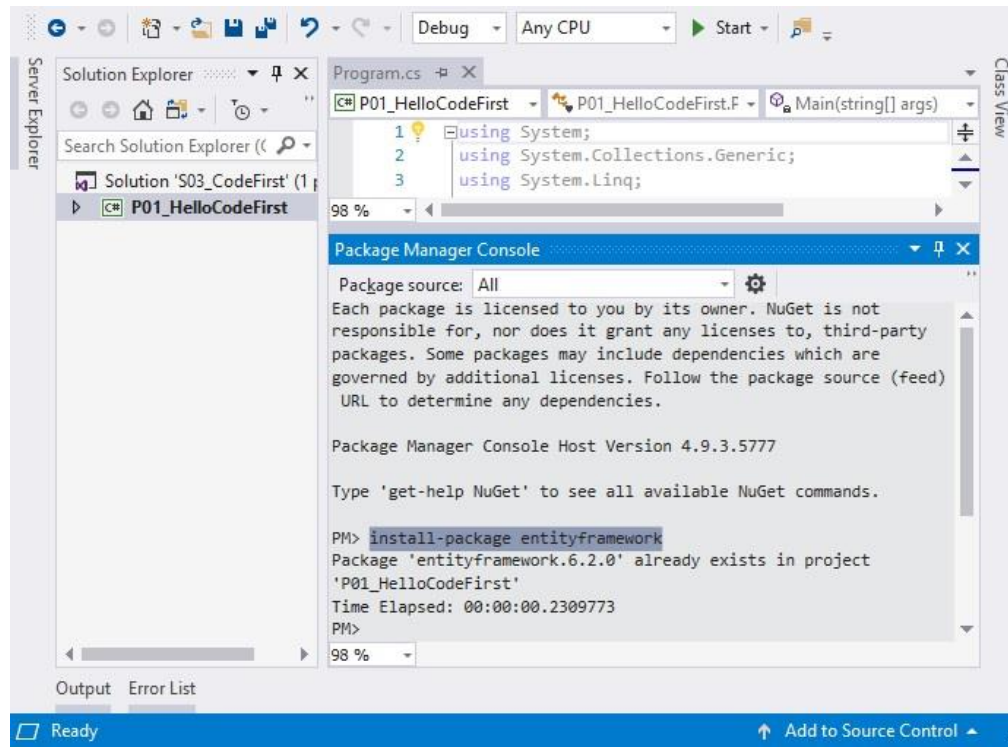
- ❑ Sử dụng EF Tool: sử dụng Visual Studio và tải công cụ ở địa chỉ <https://www.microsoft.com/en-us/download/details.aspx?id=40762> sau đó cài đặt.
- ❑ Sử dụng NuGet Package Manager
 - ❖ Bước 1. Click phải chuột vào tên project và chọn "Manage NuGet Packages" để mở ra cửa sổ (tab) NuGet Package Manager
 - ❖ Bước 2. Chọn tab Browse, gõ tìm kiếm entity framework. Lựa chọn gói EntityFramework ở danh sách và ấn Install để bắt đầu cài đặt

❑ Sử dụng NuGet Package Manager



❑ Sử dụng Package Manager Console

- ❖ Bước 1. Mở cửa sổ này từ View => Other Windows
=> Package Manager Console
- ❖ Bước 2. Từ dấu nhắc lệnh gõ install-package entityframework (không phân biệt hoa-thường) và ấn Enter.

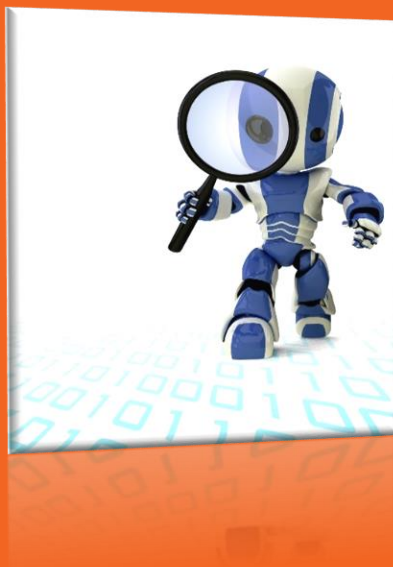




DEMO

-Hiện thực các ví dụ

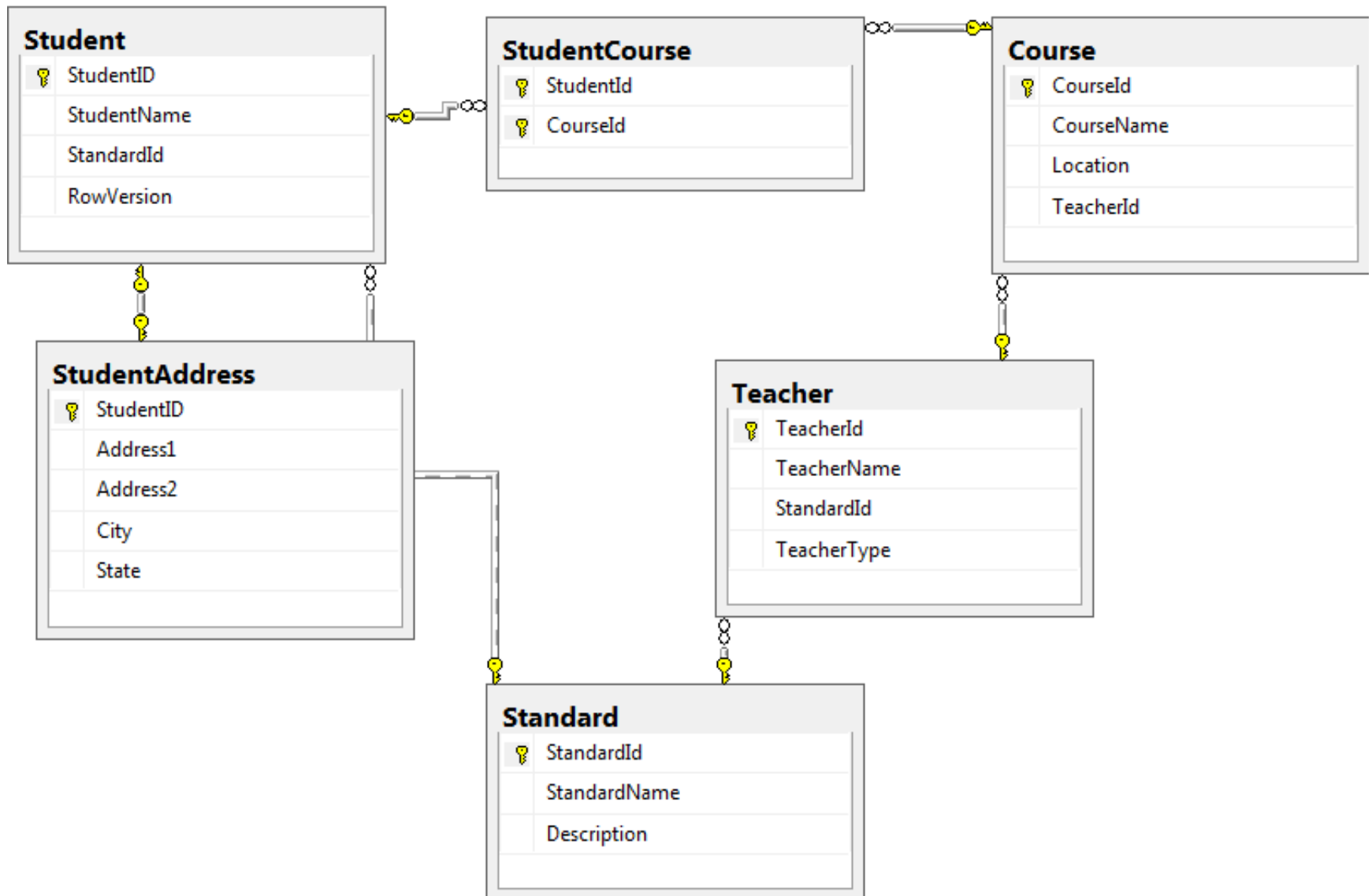




LẬP TRÌNH C# 3

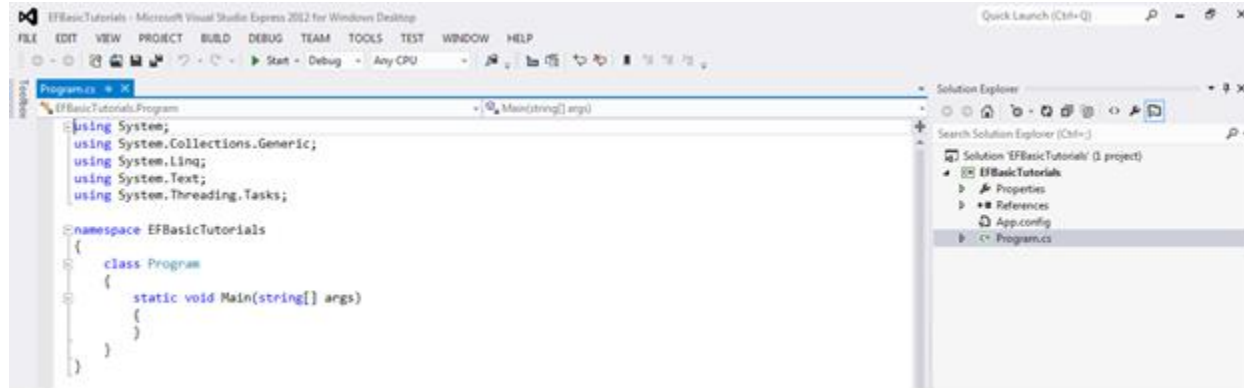
BÀI 5: ENTITY FRAMEWORK VÀ EF 6 DATABASE-FIRST (P2)

- ❑ EF sẽ tự động tạo ra data model và các class (object layer) cho bạn từ database có sẵn thông qua công cụ Entity Data Model Wizard.
- ❑ Bạn có thể thay đổi data model và cập nhật lại vào database
- ❑ Mô hình này được lưu trong tập tin tên là EDMX (.edmx)
- ❑ Các ví dụ sẽ sử dụng csdl SchoolDB.mdf (<https://github.com/entityframeworktutorial/EF6-DBFirst-Demo/blob/master/EF6DBFirstDemo/SchoolDB.mdf>)

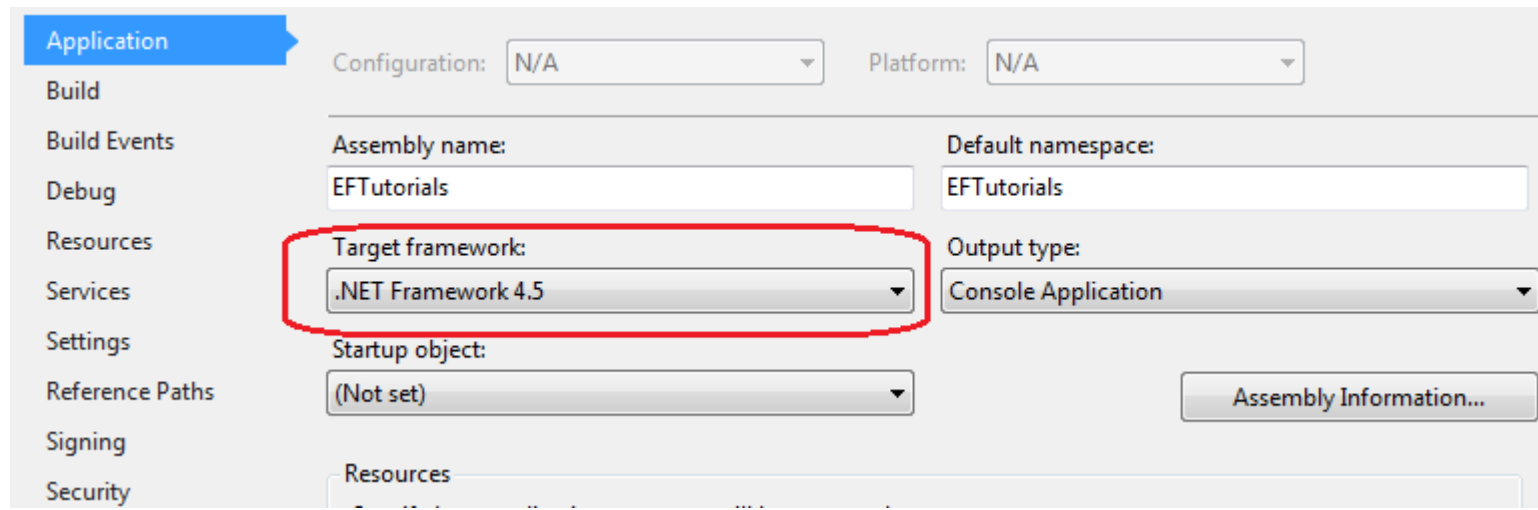


- ❑ **One-to-One:** Student và StudentAddress có quan hệ one-to-one. Student có một hoặc không có StudentAddress
- ❑ **One-to-Many:** Standard và Teacher có quan hệ one-to-many. nhiều Teachers có thể kết hợp với một Standard
- ❑ **Many-to-Many:** Student và Course có quan hệ many-to-many sử dụng bảng StudentCourse và bảng StudentCourse bao gồm StudentId và CourseId. Do đó một sinh viên có thể tham gia nhiều khóa học và một khóa học cũng có thể có nhiều sinh viên

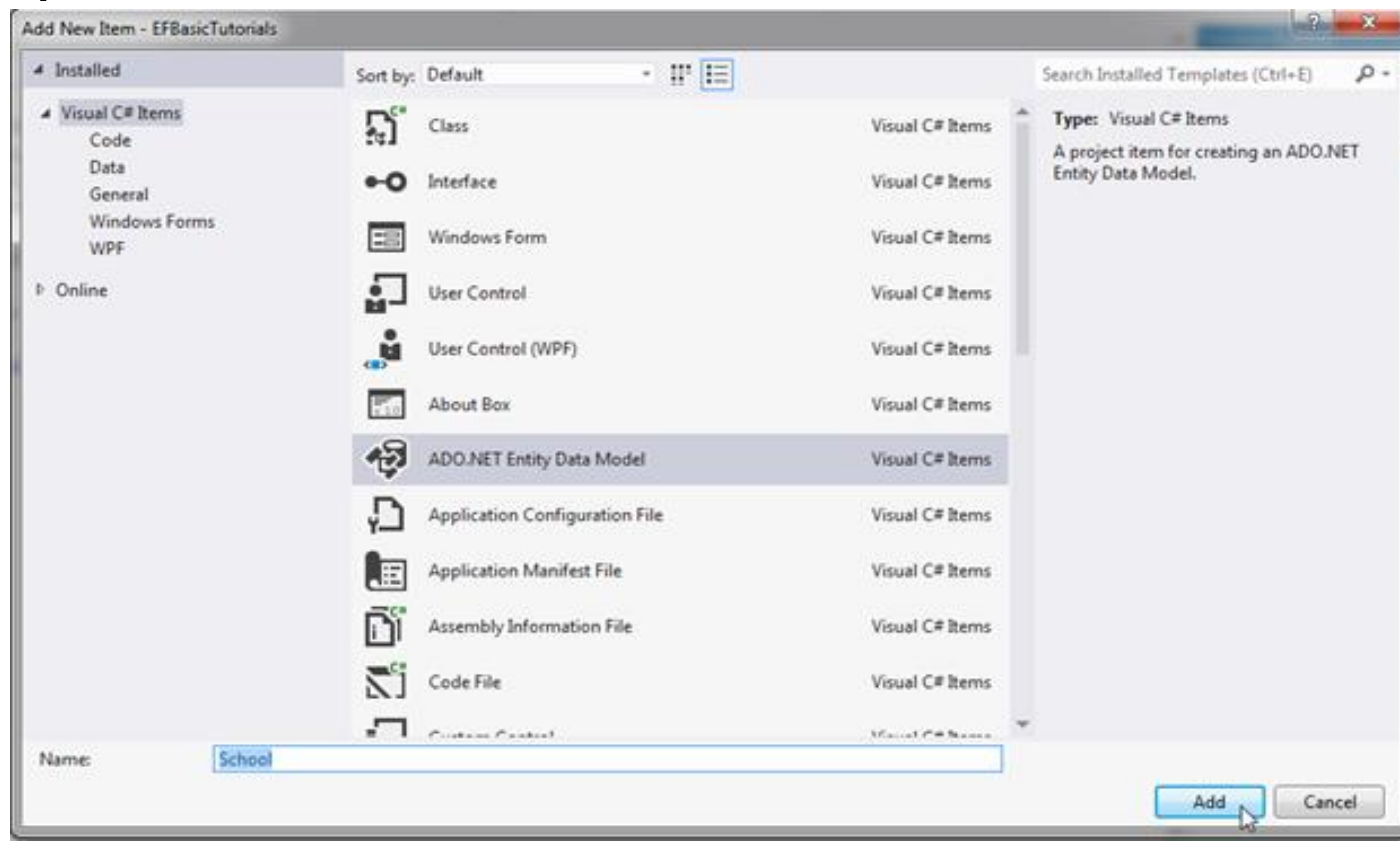
❖ Mở Visual Studio và tạo một console project



❖ Chọn target framework của project là .NET Framework 4.5

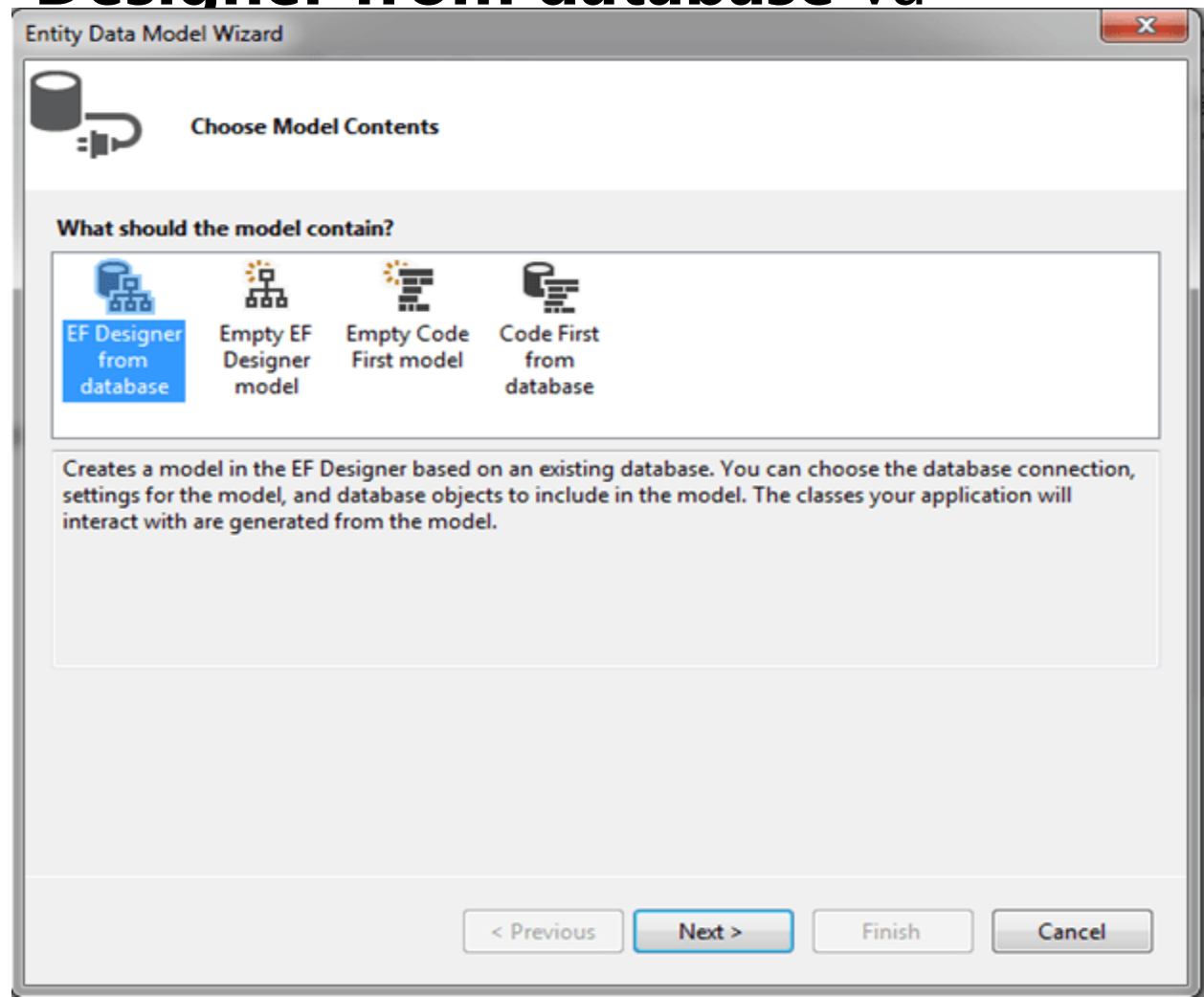


- ❑ Thêm EDM bằng cách click chuột phải vào project trong solution explorer -> Add -> click New Item và chọn ADO.NET Entity Data Model từ hộp thoại, đặt tên là 'School' và click nút Add.

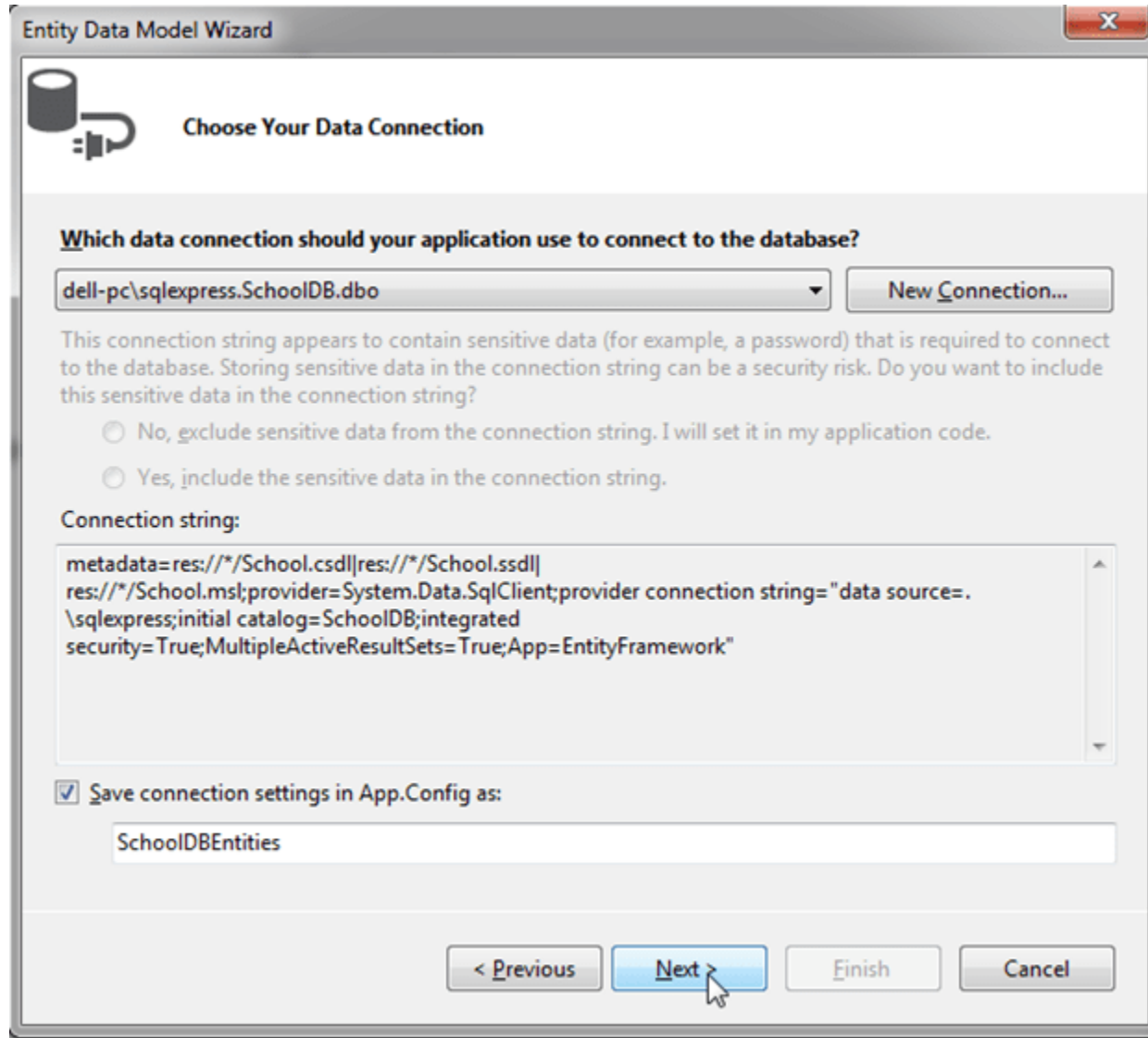


- ❑ Entity Data Model Wizard trong VS mở ra với 4 lựa chọn:
 - ❖ **EF Designer from database** cho phương pháp tiếp cận Database First
 - ❖ **Empty EF Designer model** cho phương pháp tiếp cận Model First
 - ❖ **Empty Code First model** và **Code First from database** cho phương pháp tiếp cận Code First

- ❑ Chọn phương pháp tiếp cận Database-First vì vậy chọn **EF Designer from database** và click **Next**



□ Tạo kết nối tới csdl SchoolDB



The image shows the 'Entity Data Model Wizard' dialog box. It has a title bar with a close button. Below the title bar is a section with a database icon and the text 'Choose Your Data Connection'. The main area contains the question 'Which data connection should your application use to connect to the database?'. Below this is a dropdown menu showing 'dell-pc\sqlexpress.SchoolDB.dbo' and a 'New Connection...' button. A warning message follows: '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.' and 'Yes, include the sensitive data in the connection string.'. Below this is a 'Connection string:' label and a text area containing the connection string: 'metadata=res://*/School.csdl|res://*/School.ssdl|res://*/School.msl;provider=System.Data.SqlClient;provider connection string="data source=. \sqlexpress;initial catalog=SchoolDB;integrated security=True;MultipleActiveResultSets=True;App=EntityFramework"'. At the bottom, there is a checkbox 'Save connection settings in App.Config as:' which is checked, and a text box containing 'SchoolDBEntities'. The bottom of the dialog has four buttons: '< Previous', 'Next >', 'Finish', and 'Cancel'. A mouse cursor is pointing at the 'Next >' button.

Entity Data Model Wizard

Choose Your Data Connection

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

dell-pc\sqlexpress.SchoolDB.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:

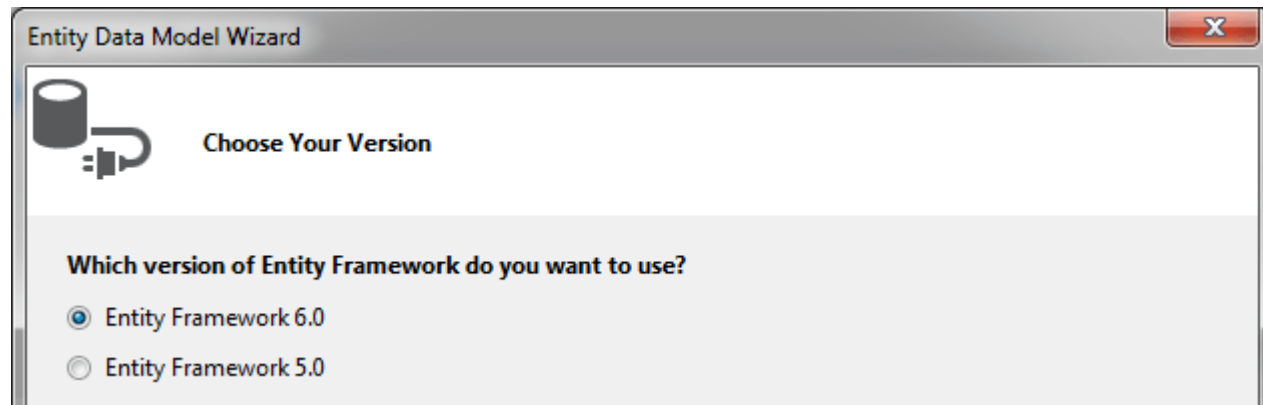
metadata=res://*/School.csdl|res://*/School.ssdl|res://*/School.msl;provider=System.Data.SqlClient;provider connection string="data source=. \sqlexpress;initial catalog=SchoolDB;integrated security=True;MultipleActiveResultSets=True;App=EntityFramework"

☒ Save connection settings in App.Config as:

SchoolDBEntities

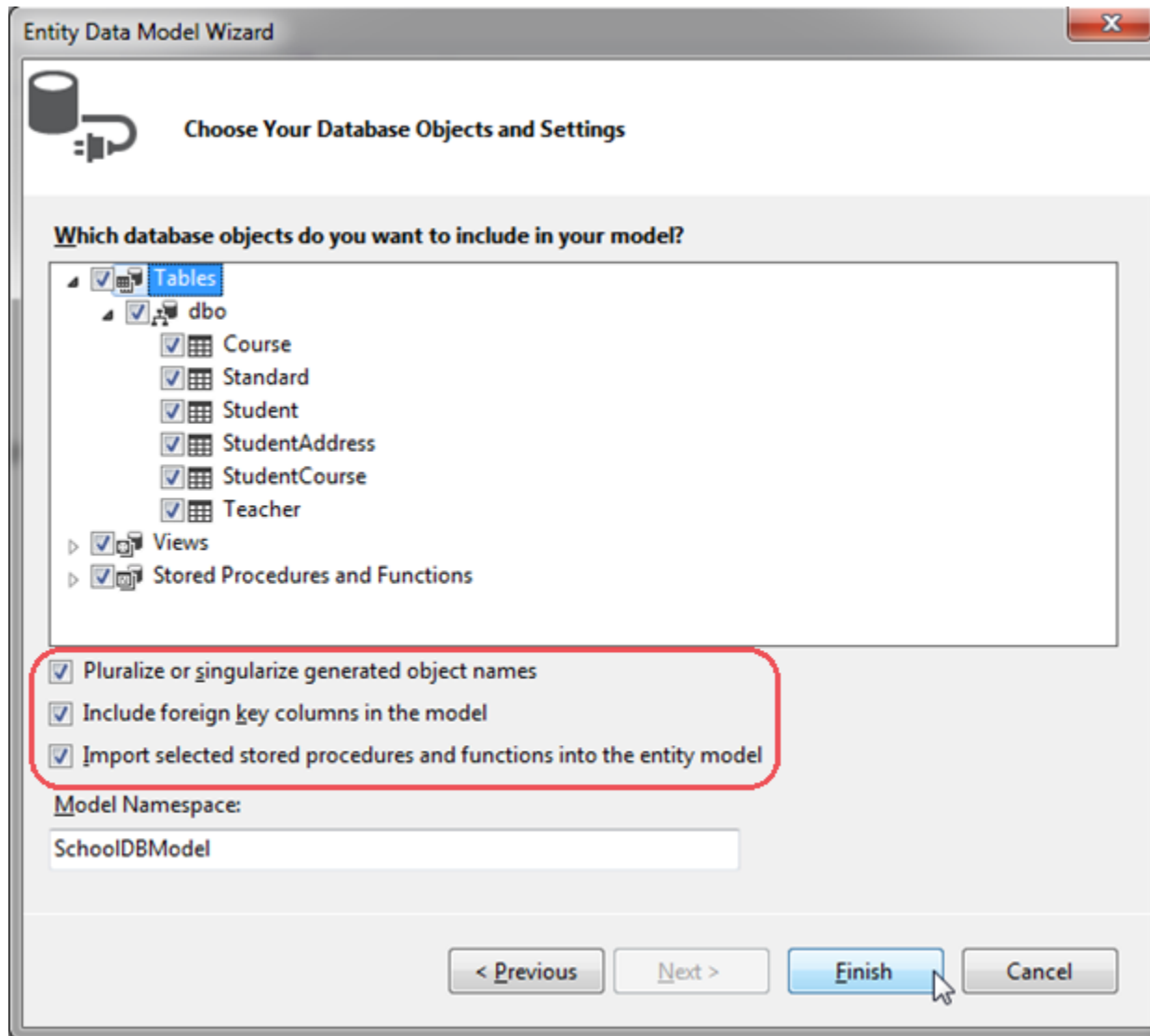
< Previous Next > Finish Cancel

❑ Chọn phiên bản của Entity Framework



❑ Bước tiếp là lựa chọn các table, view và stored procedure bạn muốn sử dụng thông qua các checkbox

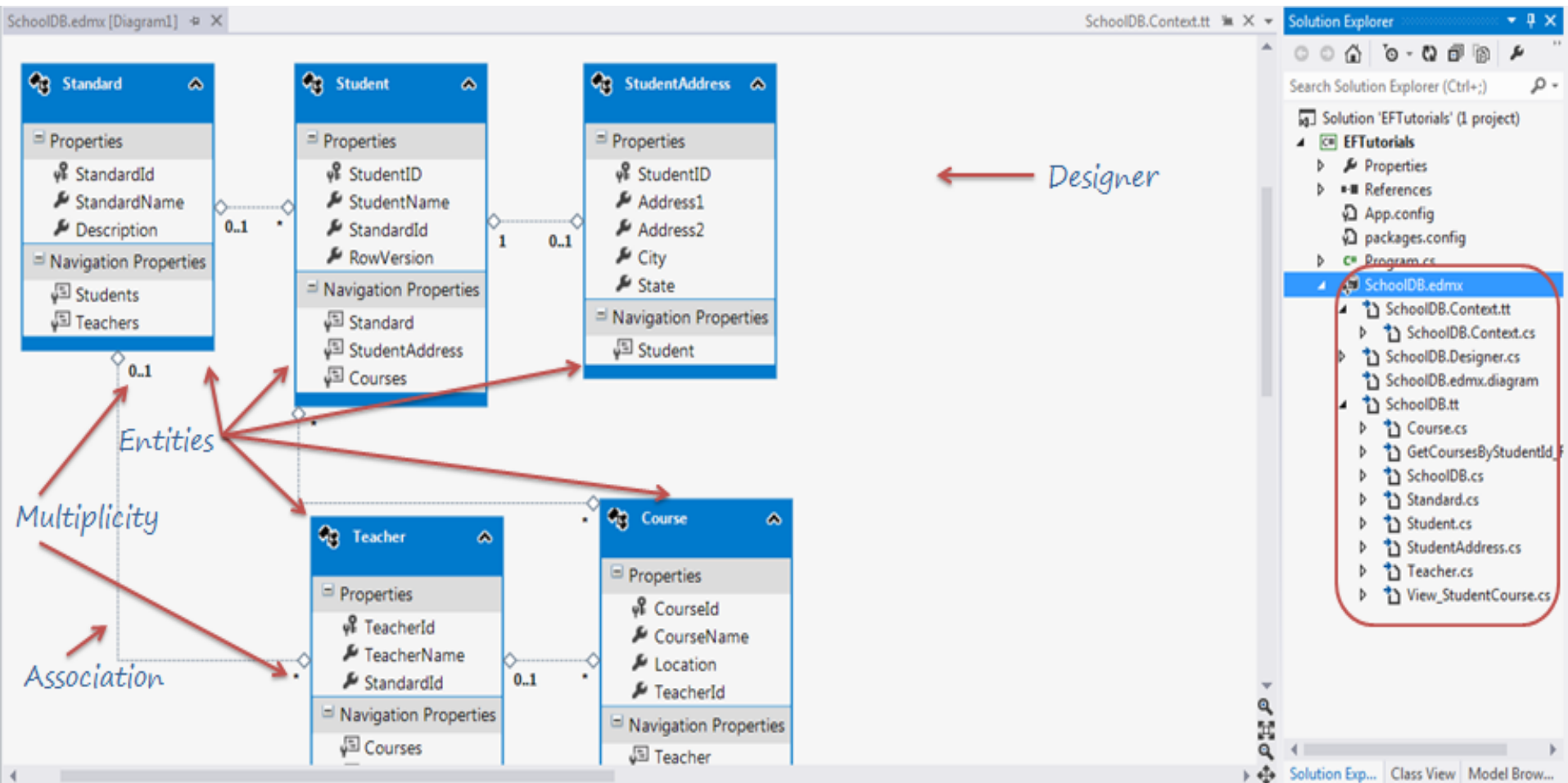
TẠO ENTITY DATA MODEL (EDM)



- ☐ **Pluralize or singularize generated object names** checkbox chuyển tên một tập thực thể thành số ít, nếu tên bảng trong CSDL là số nhiều. VD: nếu SchoolDB có tên bảng Students rồi tập thực thể sẽ chuyển thành số ít Student
- ☐ **Include foreign key columns in the model**, gồm thuộc tính khóa ngoại rõ ràng để trình bày cho khóa ngoại trong entity
- ☐ **Import selected stored procedures and functions into entity model**, tự động tạo Function Imports cho các stored procedures và functions. Bạn không cần import thủ công

TẠO ENTITY DATA MODEL (EDM)

- ❑ Sau khi click 'Finish', một file School.edmx sẽ được thêm vào project



□ EDM cũng thêm một chuỗi kết nối vào file config

```
<?xml version="1.0"?>
<configuration>
  <configSections>
    <!-- For more information on Entity Framework configuration, visit http://go.m
    <section name="entityFramework" type="System.Data.Entity.Internal.ConfigFile.En
  </configSections>
  <startup>
    <supportedRuntime version="v4.0" sku=".NETFramework,Version=v4.5"/>
  </startup>
  <entityFramework>
    <defaultConnectionFactory type="System.Data.Entity.Infrastructure.SqlConnection
    <providers>
      <provider invariantName="System.Data.SqlClient" type="System.Data.Entity.SqlS
    </providers>
  </entityFramework>
  <connectionStrings>
    <add name="SchoolDBEntities" connectionString="metadata=res://*/SchoolDB.csdl|r
  </connectionStrings>
</configuration>
```

TẠO ENTITY DATA MODEL (EDM)

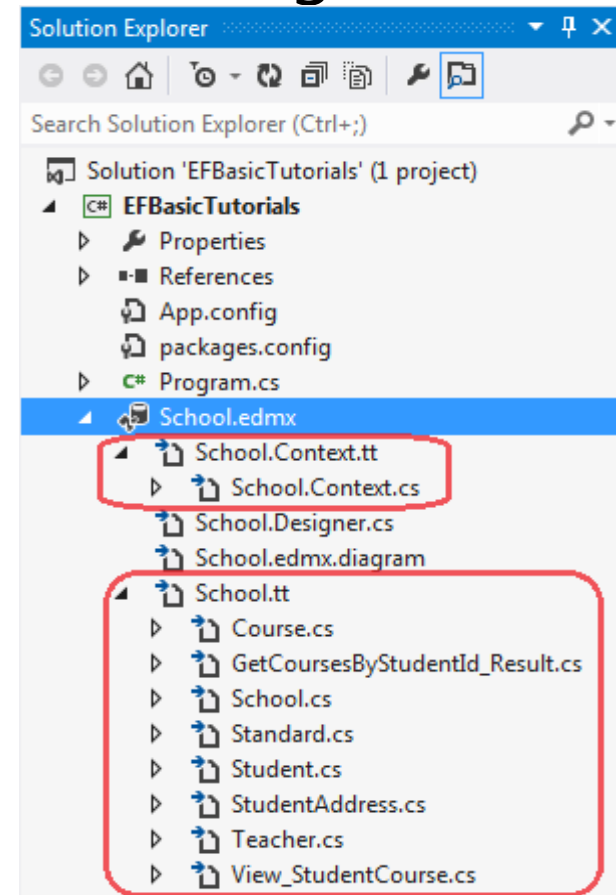
- ❑ Mỗi Entity Data Model khởi tạo một lớp context và lớp thực thể cho mỗi bảng CSDL trong EDM
- ❑ Lớp context (SchoolDBEntities) hỗ trợ tương tác với CSDL các thao tác truy vấn, thêm, xóa, sửa...

```
EFBasicTutorials.SchoolDBEntities
namespace EFBasicTutorials
{
    using System;
    using System.Data.Entity;
    using System.Data.Entity.Infrastructure;
    using System.Data.Entity.Core.Objects;
    using System.Linq;

    public partial class SchoolDBEntities : DbContext
    {
        public SchoolDBEntities()
            : base("name=SchoolDBEntities")
        {
        }

        protected override void OnModelCreating(DbModelBuilder modelBuilder)
        {
            throw new UnintentionalCodeFirstException();
        }

        public virtual DbSet<Course> Courses { get; set; }
        public virtual DbSet<Standard> Standards { get; set; }
        public virtual DbSet<Student> Students { get; set; }
        public virtual DbSet<StudentAddress> StudentAddresses { get; set; }
        public virtual DbSet<Teacher> Teachers { get; set; }
        public virtual DbSet<View_StudentCourse> View_StudentCourse { get; set; }
    }
}
```

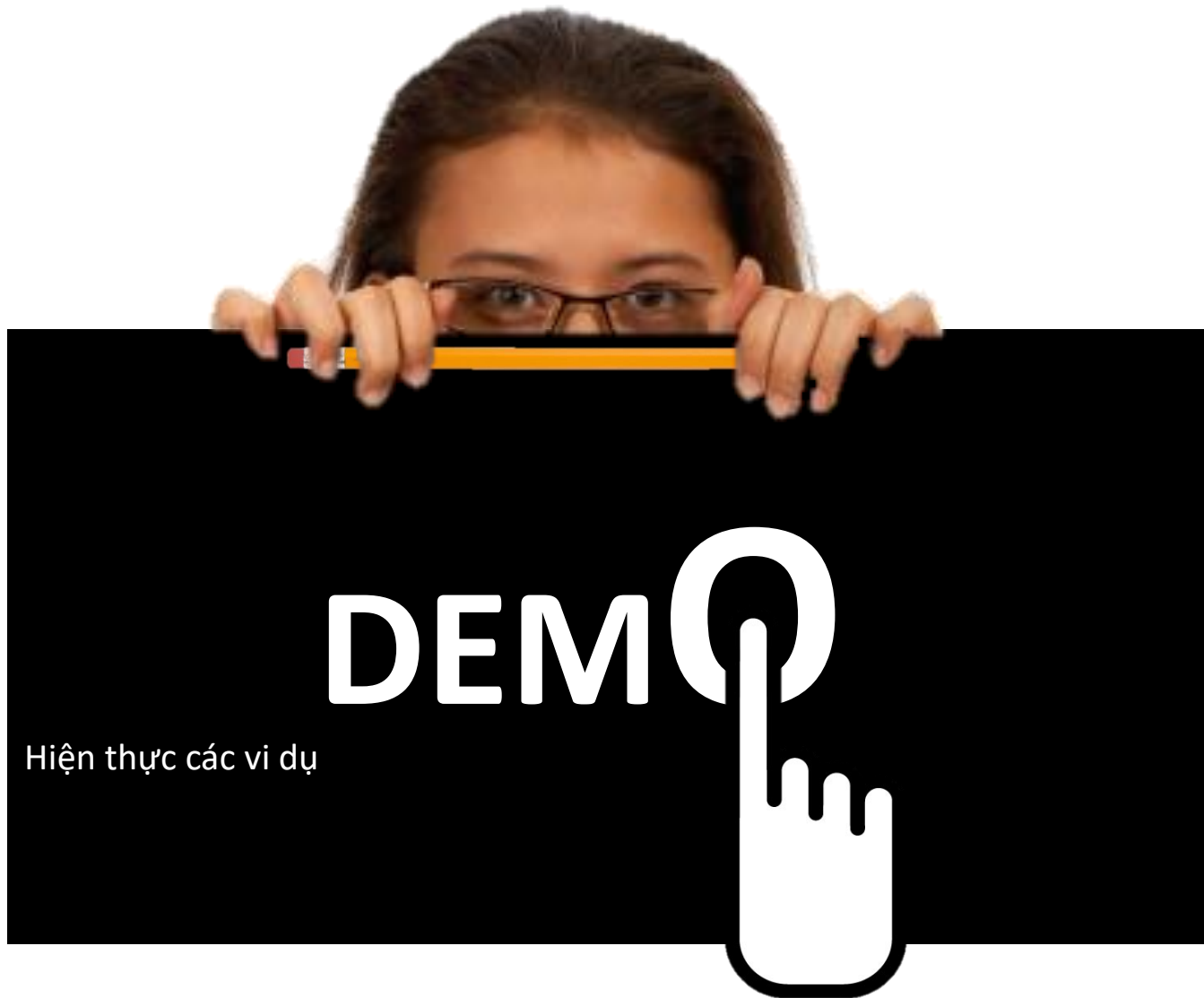


- ❑ Ví dụ tìm tên của student có StudentID=, cần tạo đối tượng SchoolDBEntities và dùng cú pháp truy vấn LINQ Method

```
static void Main(string[] args)
{
    using (var ctx = new SchoolDBEntities())
    {
        var student = ctx.Students
                        .Where(s => s.StudentID ==1)
                        .FirstOrDefault<Student>();
        Console.WriteLine(student.StudentName);
    }
}
```

- ❑ Hoặc cú pháp LINQ Query

```
using (var ctx = new SchoolDBEntities())
{
    var student = (from s in ctx.Students
                   where s.StudentID ==1
                   select s).FirstOrDefault<Student>();
    Console.WriteLine(student.StudentName);
}
```



Tổng kết bài học

- ☉ Entity Framework

- ☉ EF 6 Database-First





KẾT THÚC