

Bài thực hành 02

Truy vấn dữ liệu với Entity Framework Core

1. Mục tiêu

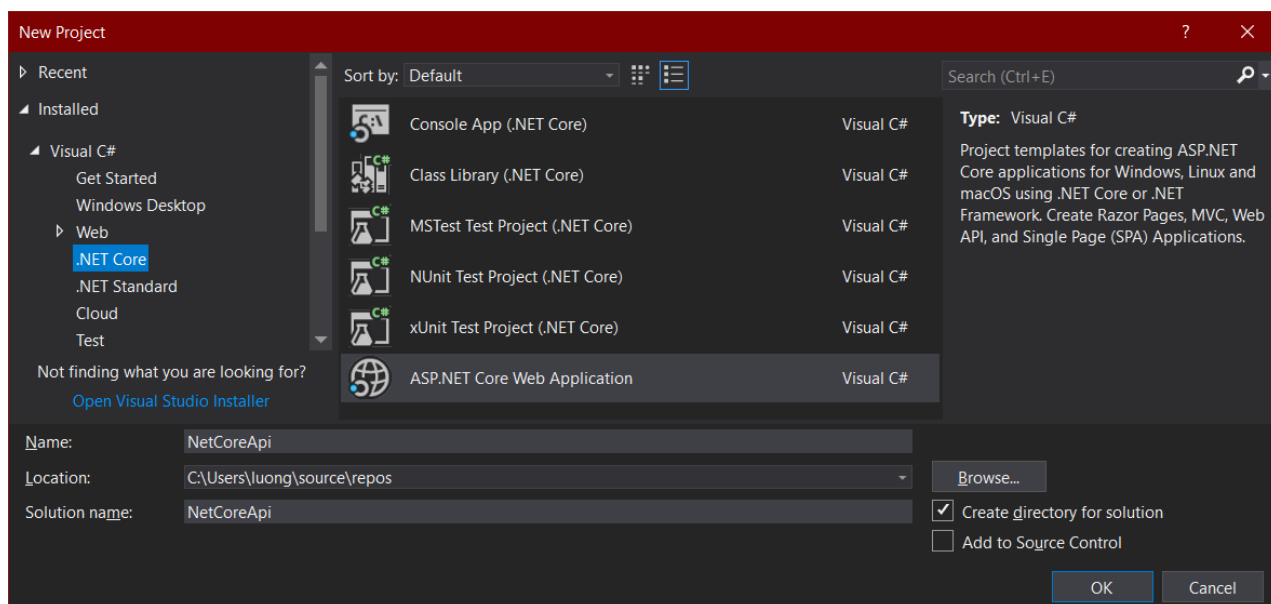
- Tạo project ASP.NETCore Api
- Kết nối CSDL Sql server truy vấn dữ liệu sử dụng EntityFrameworkCore
- Sử dụng Advanced Client Rest để test Api

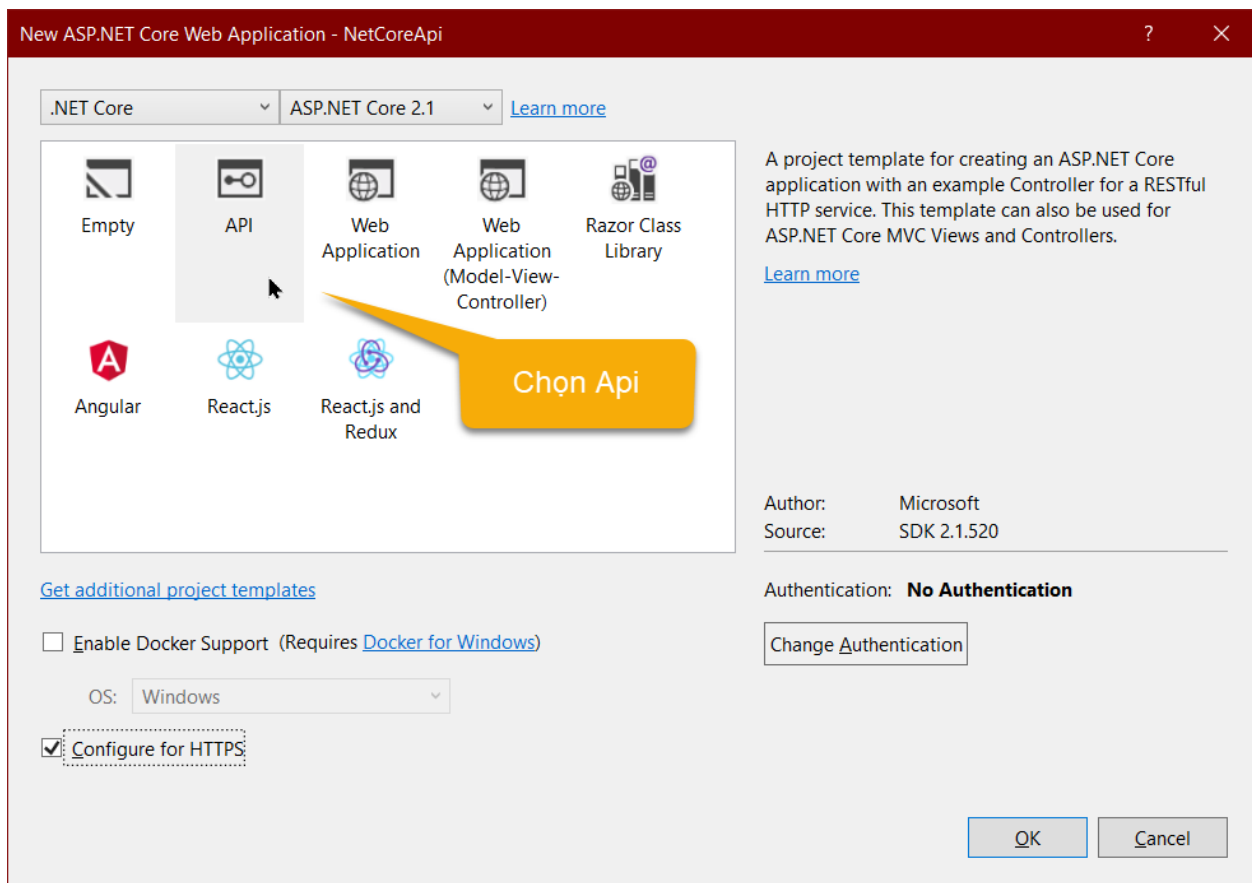
2. Bài thực hành Step by Step (kế thừa từ lab 07)

Bài 1: Tạo project ASP.NET Core Api

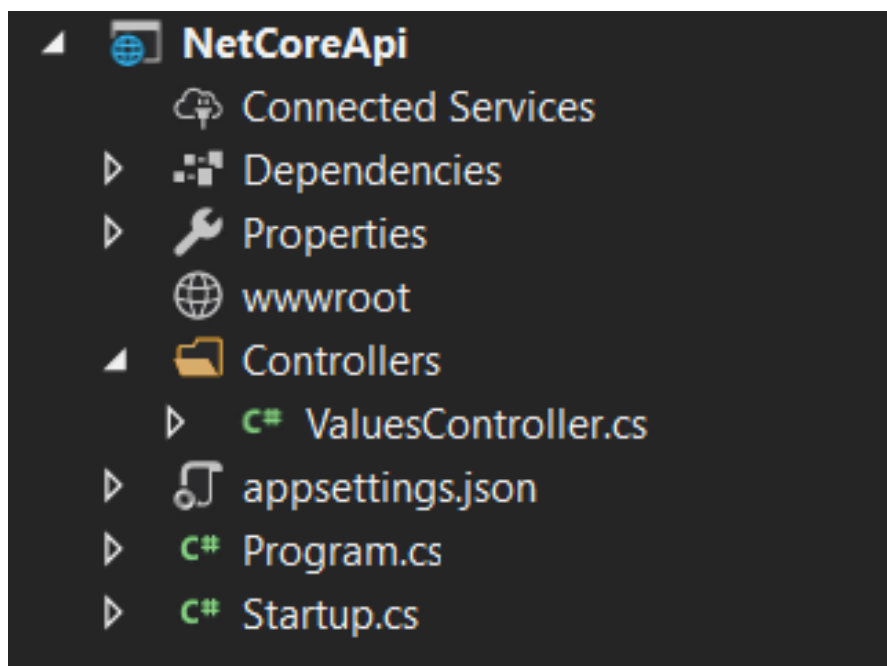
Bước 1: Tạo một project mới với template Api

Mở Visual Studio vào tạo project với tên là **NetCoreApi** chọn template là Api, xem hình



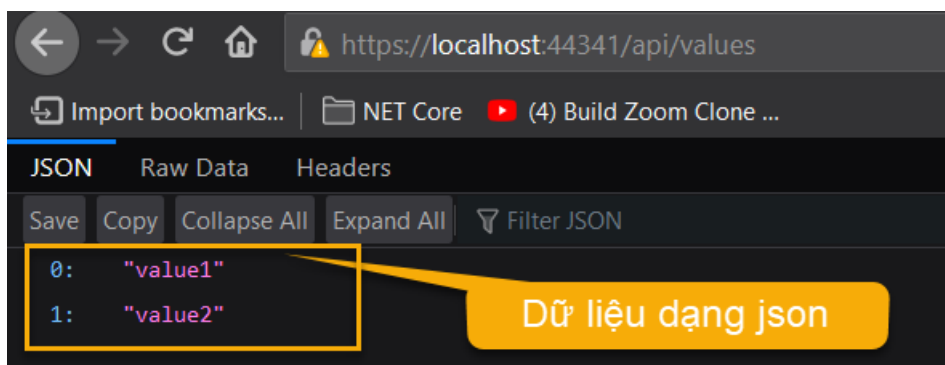


Thư mục dự án có dạng



Bước 2: Build dự án và truy cập thử dữ liệu có dạng

Truy cập link: <https://localhost:44341/api/values>



Dữ liệu này được trả về từ ValuesController

Bài 2: Tạo Api có đầy đủ CRUD theo tác với CSDL

Sử dụng phương pháp Code First để tạo CSDL (Xem lại các lab trước)

Bước 1: Tạo thêm thư mục Models và tạo model Category gồm các thuộc tính sau

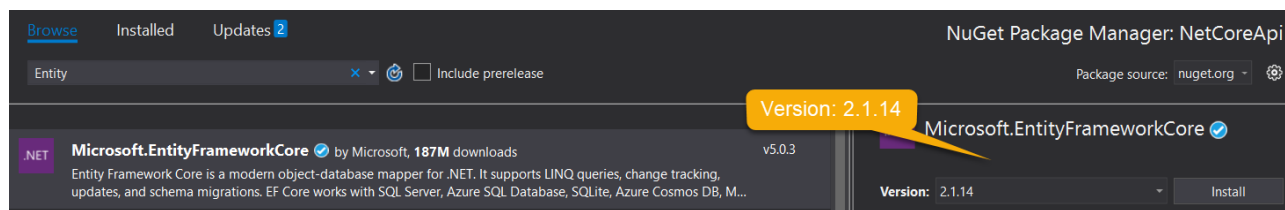
```
using System.ComponentModel.DataAnnotations;

namespace NetCoreApi.Models
{
    [Table("Category")]
    public class Category
    {
        [Key]
        public int Id { get; set; }
        [Required(ErrorMessage = "Tên danh mục không để trống")]
        public string Name { get; set; }
        public int Status { get; set; }
    }
}
```

Bước 2: Cài đặt Entity Framework

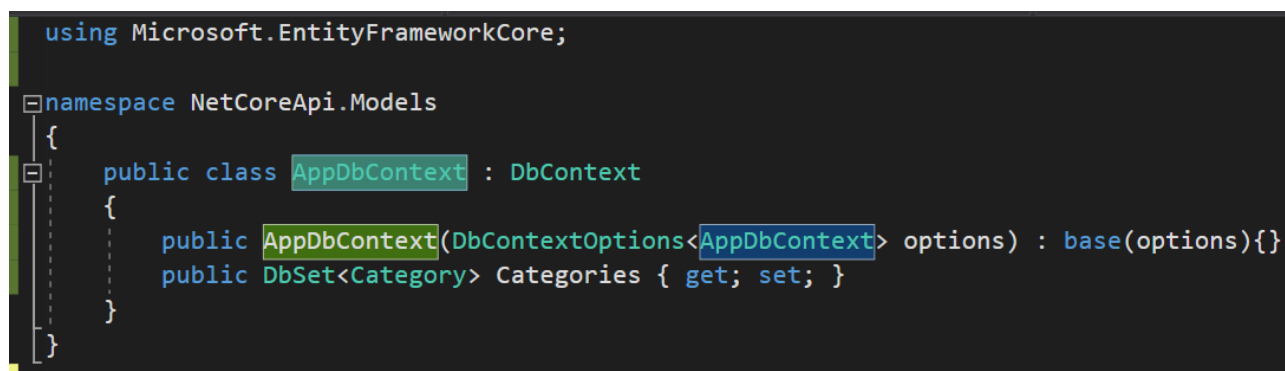


Mở NuGet Packages lên tìm Entity



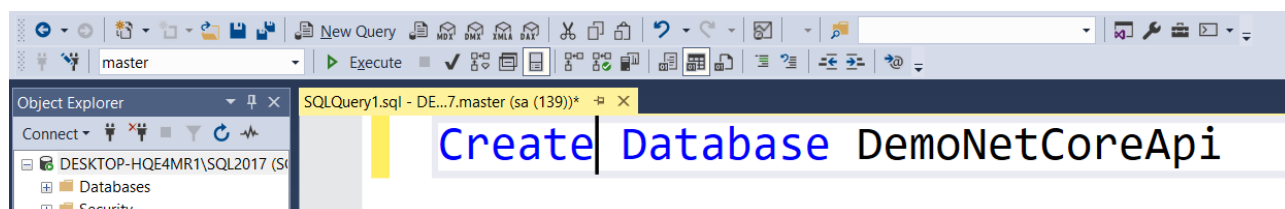
Bước 3: Tạo ApplicationDbContext

Trong thư mục Models tạo class mới tên là **AppDbContext**



Bước 4: Tạo mới một CSDL

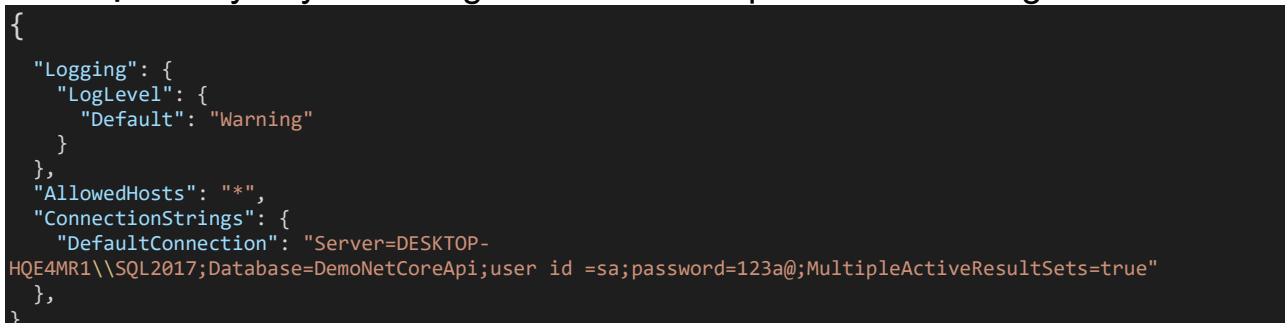
Mở SQL server lên và tạo CSDL **DemoNetCoreApi**



Bước 5: Cấu hình chuỗi kết nối CSDL DemoNetCoreApi

Mở file **appsettings.json** và cấu hình kết nối tương tự như sau

Các bạn chú ý hay đổi thông tin kết nối tới sql server cho đúng

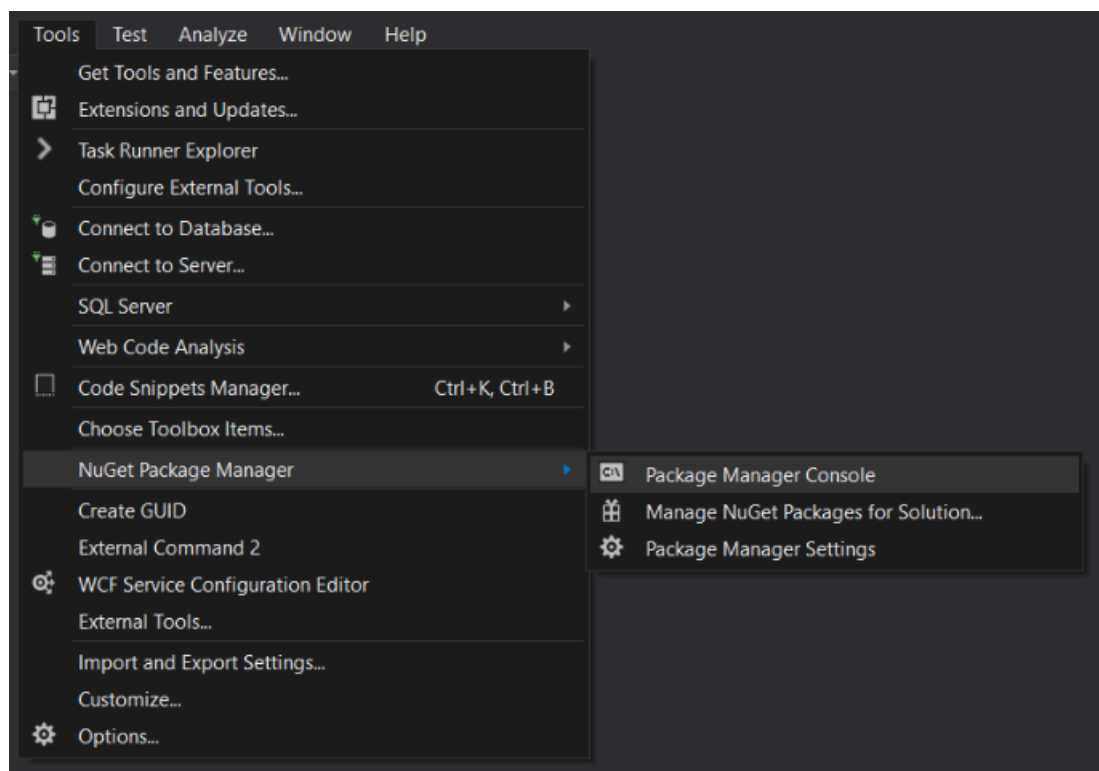


Bước 6: Cấu hình kết nối trong file Startup

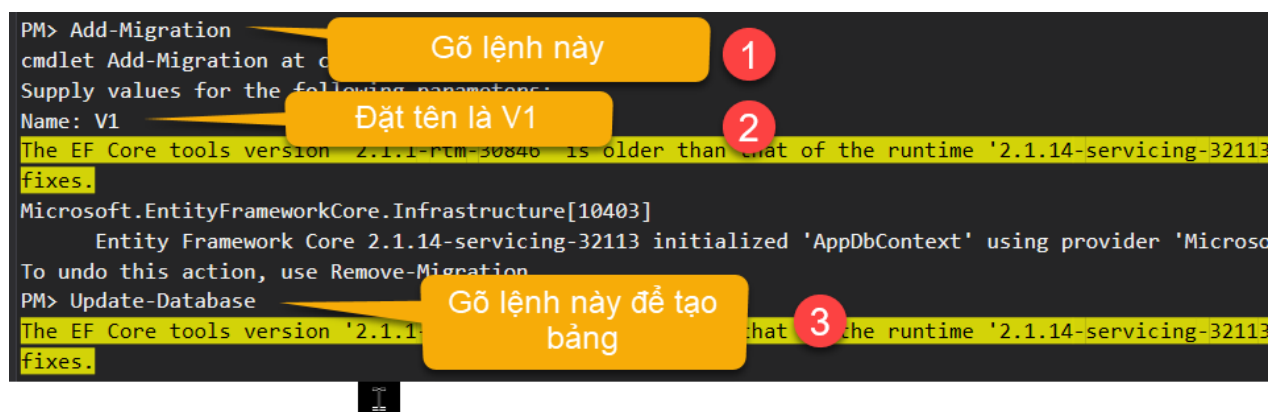
Mở file Startup và tìm đến phương thức ConfigureServices và thêm kết nối

```
// This method gets called by the runtime. Use this method to add services to the container.  
public void ConfigureServices(IServiceCollection services)  
{  
    services.AddMvc().SetCompatibilityVersion(CompatibilityVersion.Version_2_1);  
    // Cấu hình kết nối Database  
    services.AddDbContext<AppDbContext>(op => op.UseSqlServer(Configuration.GetConnectionString("DefaultConnection")));  
}
```

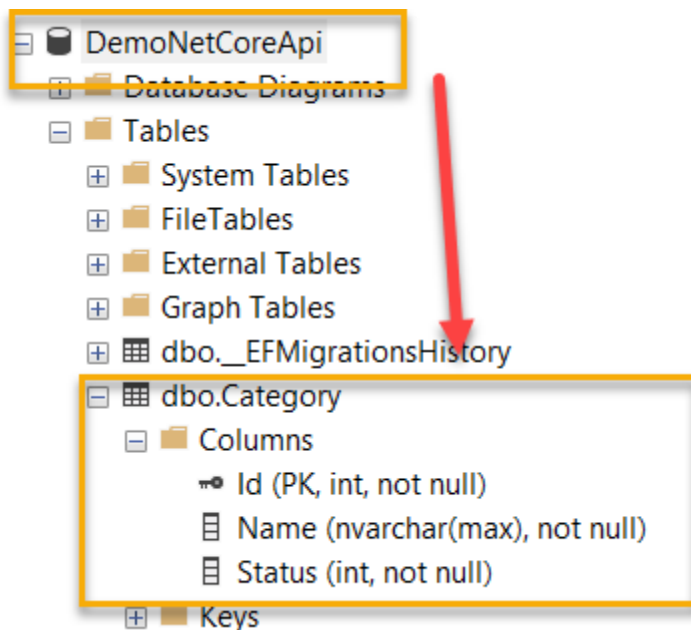
Bước 7: Vào Tools để mở NuGet Console lên và Add Migration



Sau đó gõ lần lượt các lệnh sau để tạo bảng Category

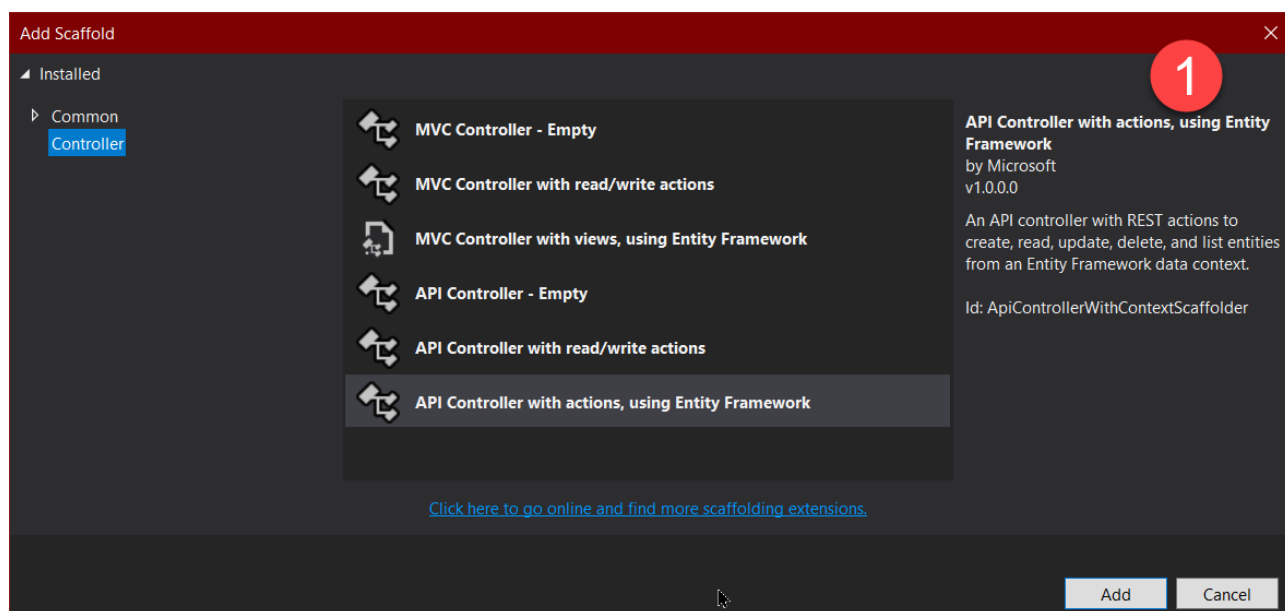


Sau khi áp dụng thành công, kiểm tra lại bảng Category trong Sql Server

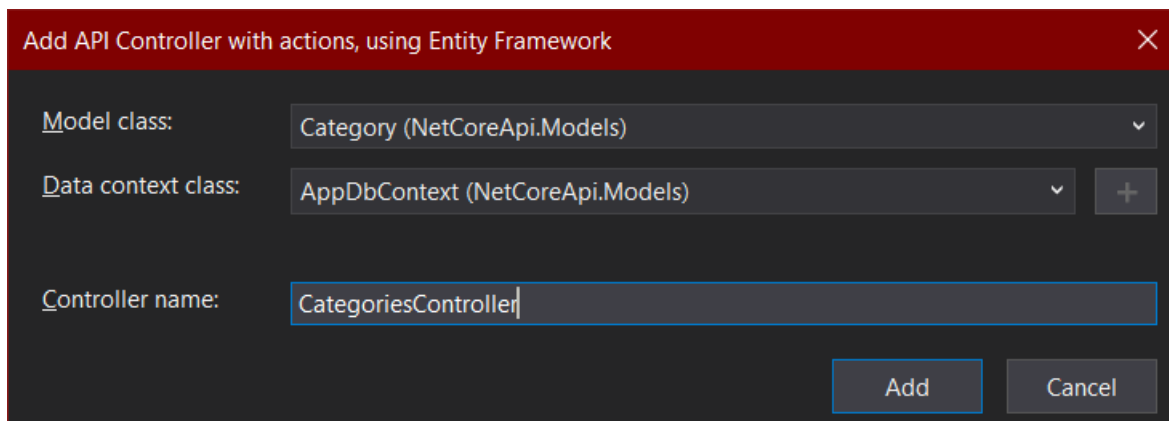


Bước 8: Tạo CategoryController sử dụng template Api Controller with action, using Entity Framework

Xem các bước



Màn hình tiếp theo chọn Model, và DbContext như hình



Bước 9: Build lại dự án chạy thử và test các api

Danh sách các Api sẽ có

Link	Phương thức	Trả về
https://localhost:44341/api/categories	GET	Danh sách
https://localhost:44341/api/categories/{id}	GET	Chi tiết
https://localhost:44341/api/categories	POST	Thêm mới
https://localhost:44341/api/categories/{id}	PUT	Cập nhật
https://localhost:44341/api/categories/{id}	DELETE	Xóa bỏ

Bài 3: Cài đặt Advanced Client Rest để test api

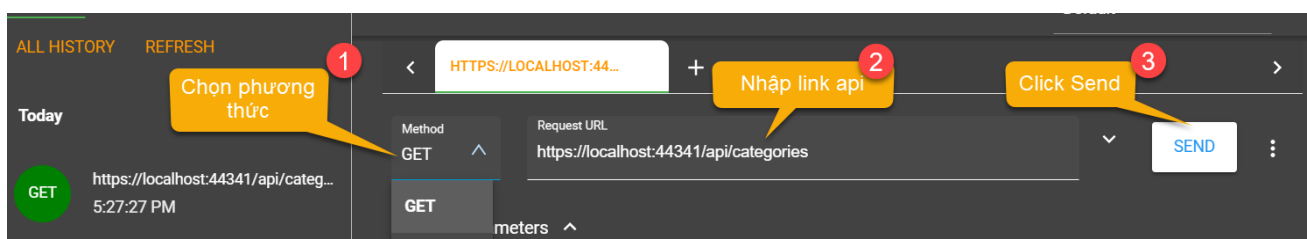
Bước 1: Tải phần mềm => Link tải phần mềm

<https://install.advancedrestclient.com/install>

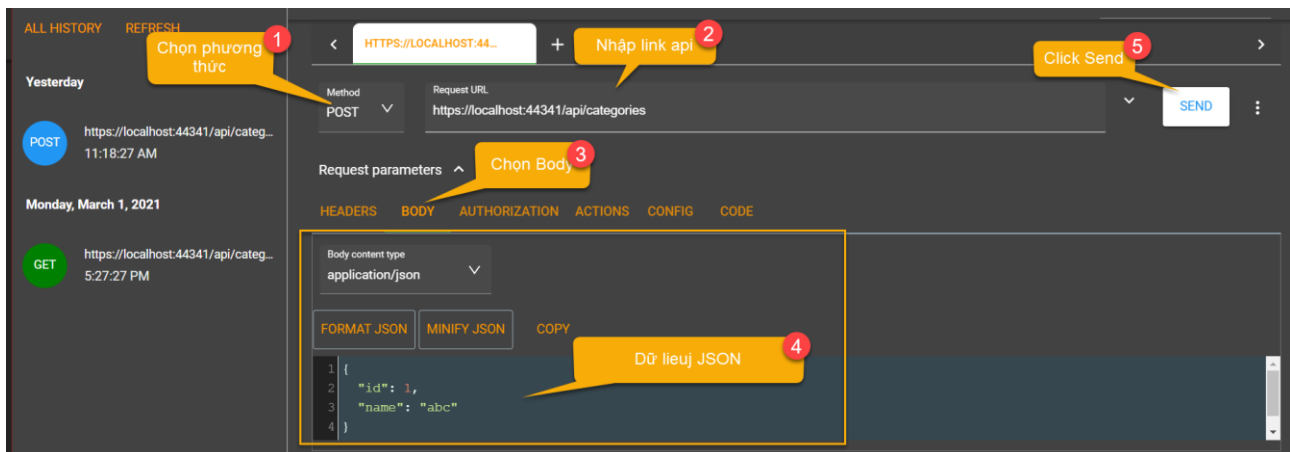
Tải xong cài đặt bình thường

Cách sử dụng, mở phần mềm lên và làm theo hướng dẫn

Phương thức GET, POST, PUT, DELETE



Phương thức POST, PUT



Bài tập tự làm

Dựa vào bài lab hướng dẫn trên

Sinh viên tạo thêm model Product gồm các thuộc tính sau

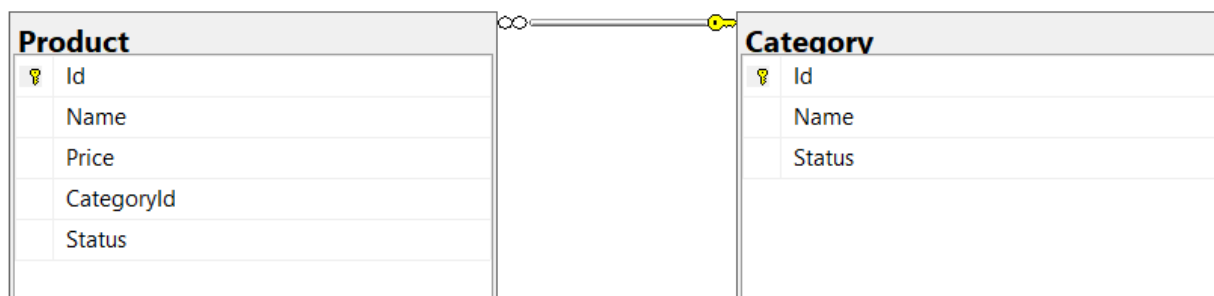
```
using System.ComponentModel.DataAnnotations;
using System.ComponentModel.DataAnnotations.Schema;

namespace NetCoreApi.Models
{
    [Table("Product")]
    public class Product
    {
        [Key]
        public int Id { get; set; }
        [Required(ErrorMessage = "Tên sản phẩm không để trống")]
        public string Name { get; set; }
        public float Price { get; set; }
        public int CategoryId { get; set; }
        public int Status { get; set; }
        // Khóa ngoại tới bảng Category
        public Category Category { get; set; }
    }
}
```

Add DbSet trong ApplicationDbContext

Add Migration

Kết qu CSDL có các bảng



Tạo controller và test như trên