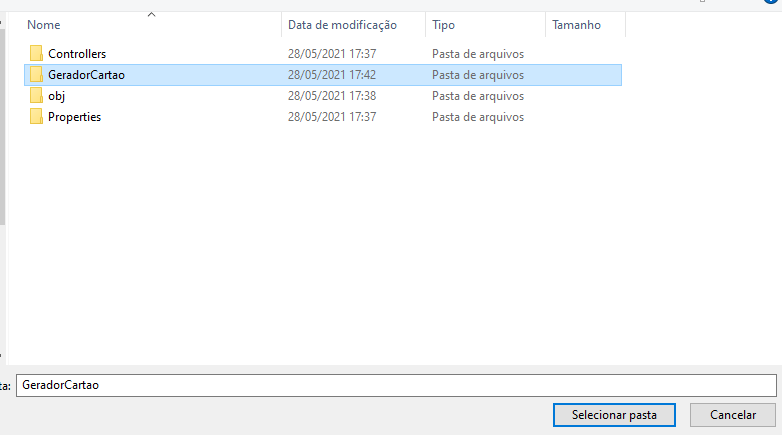
***Passa-a-passo da construção de uma API***

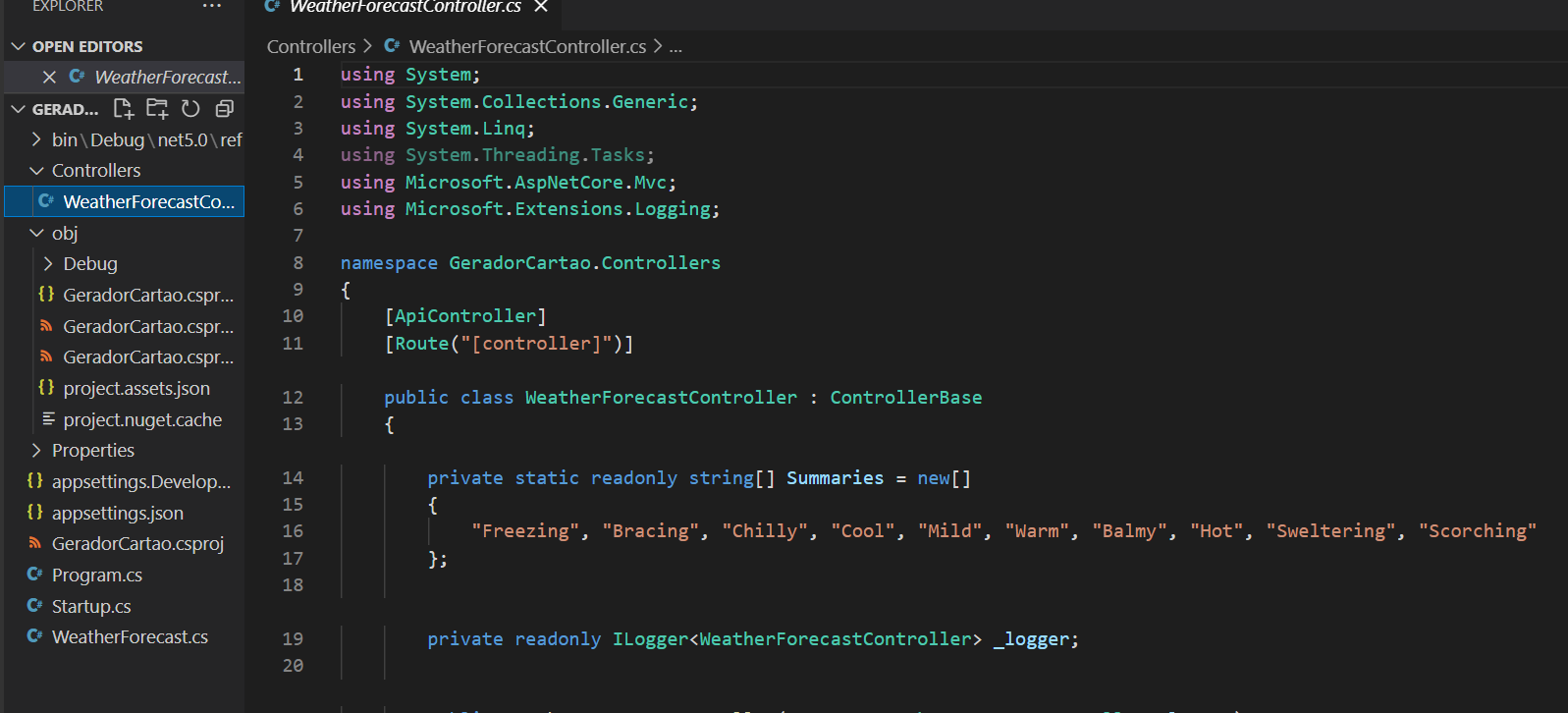
Abrir o VS CODE e abrir o TERMINAL (na parte superior da aba)

E criar o projeto com o seguinte comando: **dotnet new webapi -o GeradorCartao** (GeradorCartao é o nome do projeto, pode ser qualquer outro).

Em seguida executar o comando: **dotnet add package Microsoft.EntityFrameworkCore.InMemory** (Esse comando serve para não precisar utilizar banco de dados, faz com que fique tudo salvo na memória)

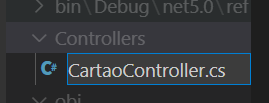
Agora em em File > Open Folder e selecione a pasta do projeto criado .

Após selecionar a pasta abrira um projeto padrão que se chama WeatherForeCast, é algo relacionado a medição de temperatura.



Você pode excluir as duas classes do WeatherForeCast.

Em seguida crie a classe CartaoController.cs dentro da pasta Controllers.



Crie também uma pasta com o nome de Models (que lá criaremos nossa classe com as propriedades que desejamos.



Agora voltando a classe CartaoController adicionaremos o seguinte código:

using System;

using System.Collections.Generic;

using System.Threading.Tasks;

using GeradorCartao.Data;

using GeradorCartao.Models;

using Microsoft.AspNetCore.Mvc;

using Microsoft.EntityFrameworkCore;

namespace GeradorCartao.Controllers

{

    [ApiController]

    [Route("")]

    public class CartaoController : ControllerBase

    {

        [HttpGet]

        [Route("")]

        public async Task<ActionResult<List<Cartao>>> Get([FromServices] DataContext context)

        {

            var cartoes = await context.Cartoes.ToListAsync();

            return cartoes;

        }

        [HttpPost]

        [Route("email}")]

        public async Task<ActionResult<Cartao>> Post([FromServices] DataContext context, string email)

        {

            if (ModelState.IsValid)

            {

                var cartaoNovo = new Models.Cartao

                {

                    Email = email,

                    CodigoCartao = GerarCodigo(),

                };

                context.Cartoes.Add(cartaoNovo);

                var data = await context.SaveChangesAsync();

                return Ok(cartaoNovo);

            }

            else

            {

                return BadRequest(ModelState);

            }

        }

        private static string GerarCodigo()

        {

            Random rnd = new Random();

            string codigoCartao = "";

            for (int i = 0; i < 19; i++)

            {

                if (i == 4 || i == 9 || i == 14)

                {

                    codigoCartao += " ";

                }

                else

                {

                    codigoCartao += rnd.Next(10);

                }

            }

            return codigoCartao;

        }

    }

}

Tentem não se preocupar em gravar os using, pois as IDEs tem a opção de implementar o que o código pede facilitando toda a incrementação dos códigos.

Agora indo para a classe Cartao.cs

using System.ComponentModel.DataAnnotations;

namespace GeradorCartao.Models

{

    public class Cartao

    {

        [Key]

        public int Id { get; set; }

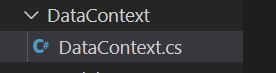
        public string Email { get; set; }

        public string CodigoCartao { get; set; }

    }

}

Agora iremos a pasta DataContext e a classe com o mesmo nome, ele serve para fazer o mapeamento dos dados.



Após a criação, insira o seguinte código:

using System.ComponentModel.DataAnnotations;

namespace GeradorCartao.Models

{

    public class Cartao

    {

        [Key]

        public int Id { get; set; }

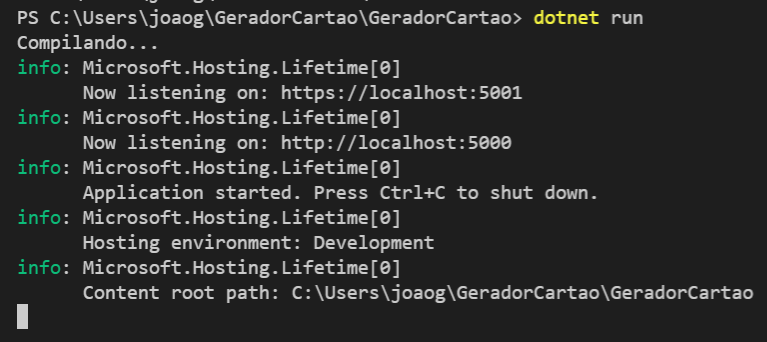
        public string Email { get; set; }

        public string CodigoCartao { get; set; }

    }

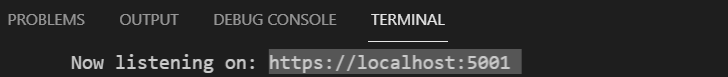
}

Agora para executar, abra terminal e digite “dotnet run”

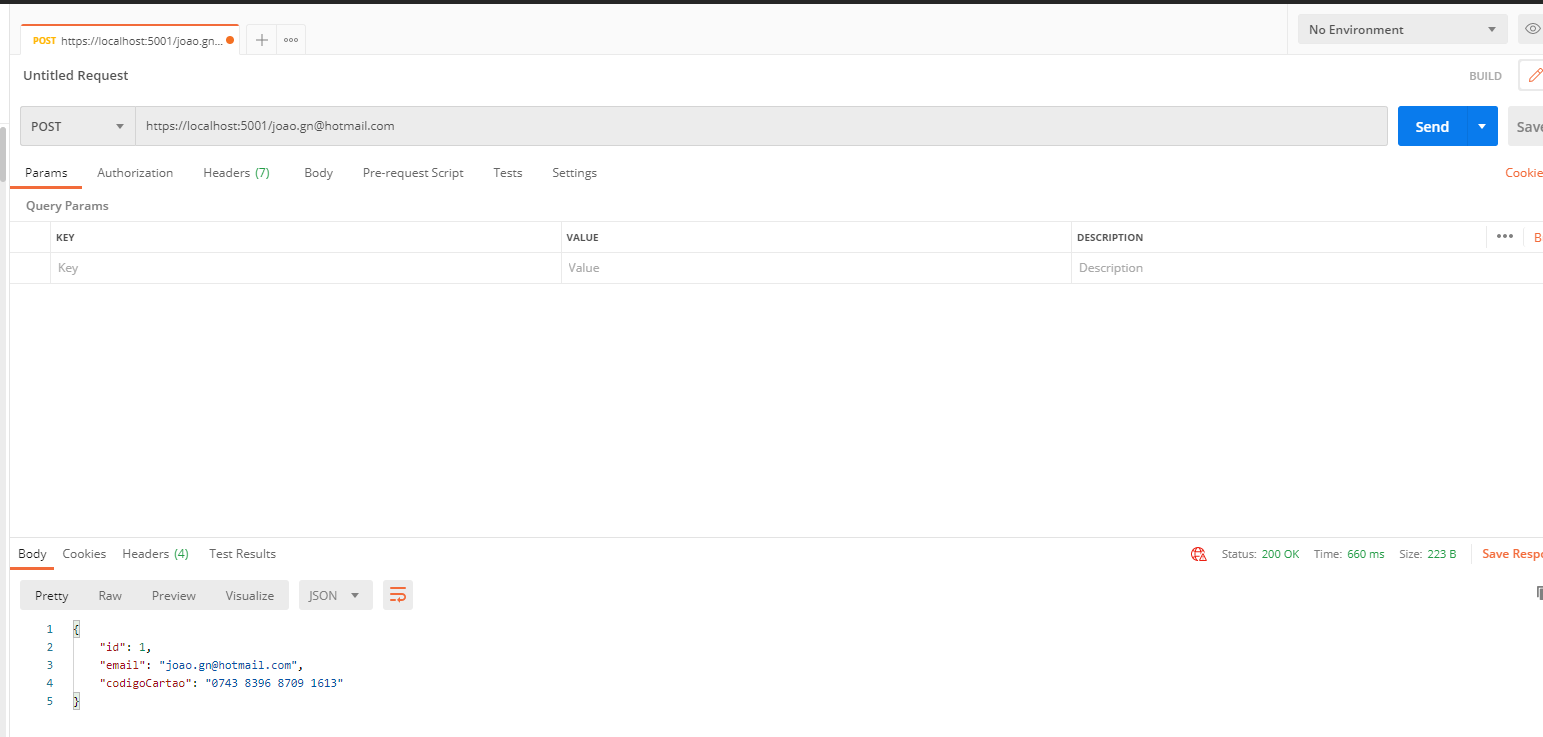


Agora baixe e instale o Postman: <https://www.postman.com>

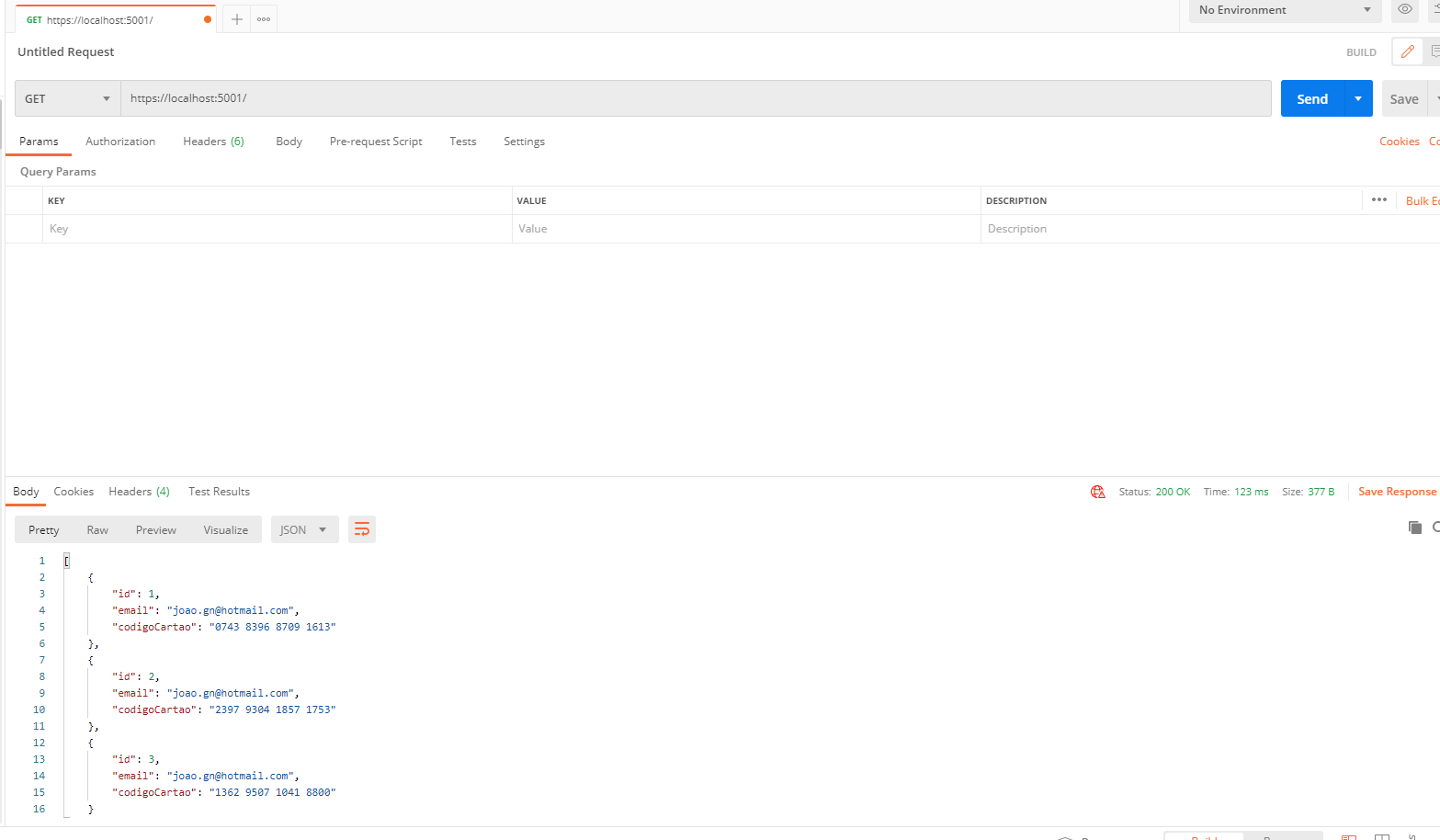
Depois abra e clique no ícone de “+” que é para abrir uma nova aba, sem seguida copie o https que é gerado no VS CODE



E coloque no postman seguido de ‘/algum email desejado’ e pode salvar, e pronto, ira gerar os dados do cartão.



Em seguida mude a opção POST (que é para enviar algum dado) por GET (que é para retornar), apague o e-mail e clique em send que ele irá trazer todas as informações salvas



Pronto, seu gerador de cartão com número randômico e com ordem de criação está feito !