**Organising your code.**

When you create a minimalist dotnet Web API, all your code is in the Program.cs

If you intend to add more functionalities or use other REST methods besides GET, it would be a good idea to refactor your code by adding some folders and moving part of the code that is in Program.cs to these new locations.

Create your project

dotnet new webapi -n OrganiseWebApi

**Create your endpoint.**

cd OrganiseWebApi

mkdir EndPoints

copy Program.cs EndPoints/WeatherforecastEndPoint.cs

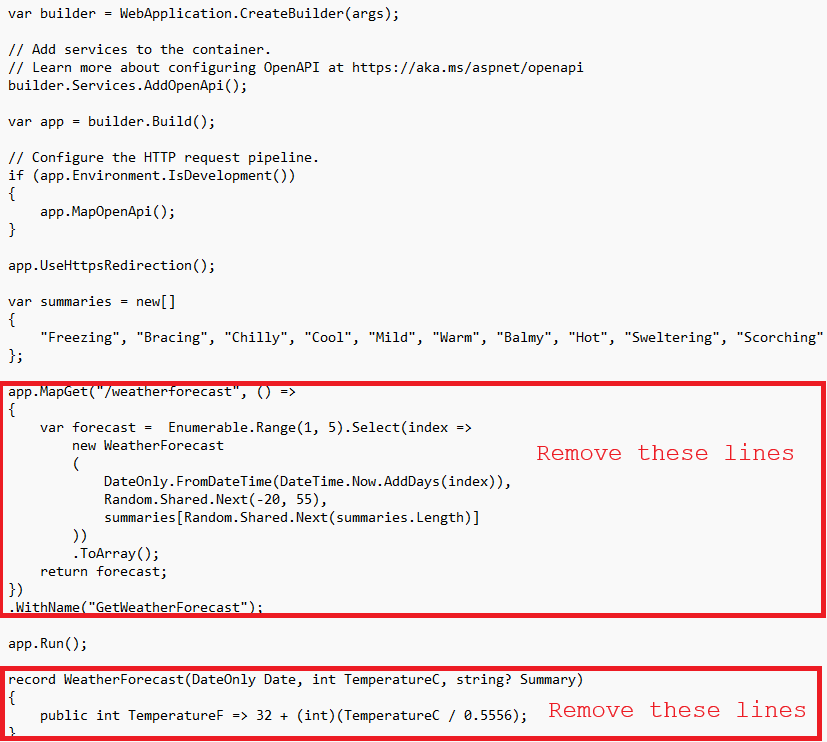
**Create your services**

mkdir Services

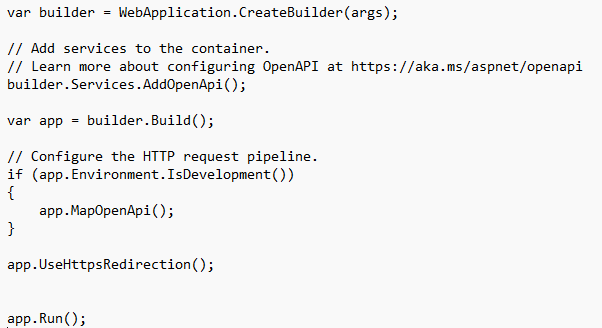
Copy Program.cs Services/WeatherforecastService.cs

**Modify your Program.cs**

Remove the implementations from **Program.cs** for use in the **Services** and the **Endpoints**.



Then your Program.cs will be something like this:



Modify your **endpoint** class.

The idea is to create a class **extension** for WebApplication.

As a **class extension**, we need to define it as **static.**

The **extension** **method** needs to be declared as receiving a parameter preceded by the word ‘**this**’.

Your REST/API methods, (GET, POST, PUT, DELETE, ..), will all be declared here; however, their functionalities will be defined in **services**.

Starting to declare the namespace. The namespace is a path where your class is declared. This will be used for other codes to find where your code is, for example:

D:\dotnet\_tips\**OrganiseWebApi\EndPoints**>

In the example above, the folder **OrganiseWebApi** is where your project was created.

The folder **EndPoints** is where you will create your class **EndPoint**. in this case, the namespace that you will need to create is:

**OrganiseWebApi\EndPoints** in C# dotnet will be **OrganiseWebApi.EndPoints,** then create your

**namespace OrganiseWebApi.EndPoints;**

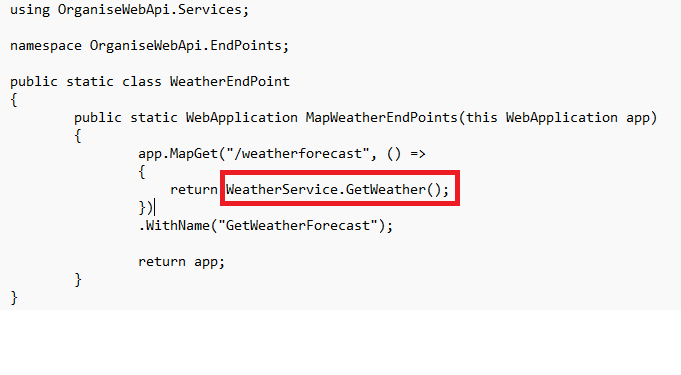
Your **WeatherforecastEndPoint.cs** needs to be something like this:

// This is necessary as we need to call the method **GetWeather()** that is inside a class

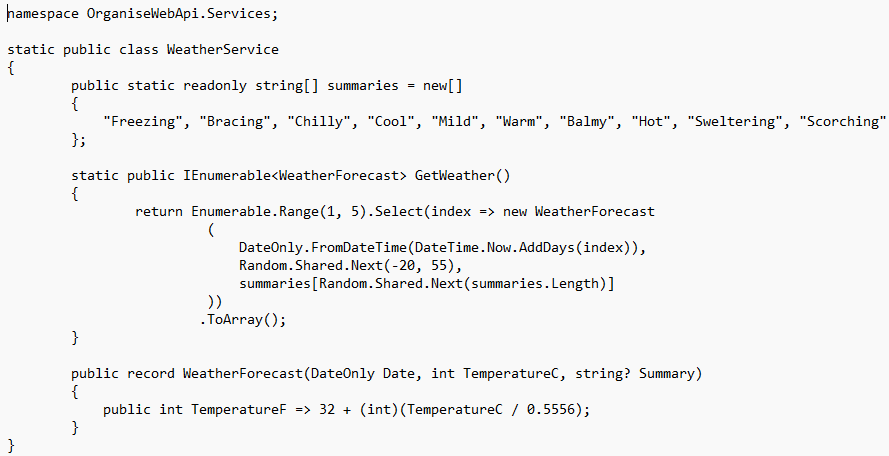
// declared in the code inside this folder:

// **OrganiseWebApi**\**Services**\WeatherforecastService.cs

**using OrganiseWebApi.Services;**

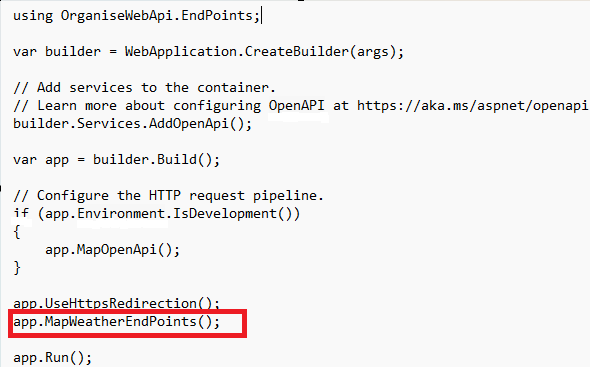


Modify your **service** class.



Modify your **Program.cs**.

Add the call to the new extension method created.



Test your web API:

Use this URL in the web browser:

[localhost:5003/weatherforecast](http://localhost:5003/weatherforecast)

Please note that the port may vary.

