使用.net接收POST与GET信息

1. 在Visual Studio中确定自己安装了ASP.net和Web开发，若没有安装进行安装

图形用户界面, 应用程序

描述已自动生成

1. 在Visual Studio中新建项目，模板选择ASP.NET Core Web API

电脑屏幕的截图

描述已自动生成

其余默认即可，创建项目后，会自带一些脚本和一个Controllers文件夹。

文本

描述已自动生成

主要设置在Controllers文件夹中，删除文件夹下项目中自带的脚本，右击文件夹，选择添加->控制器选项

屏幕的截图

描述已自动生成

选择MVC控制器-空

图形用户界面, 文本

描述已自动生成

点击“添加“后命名，即可创建一个控制器

电脑萤幕的截图

描述已自动生成

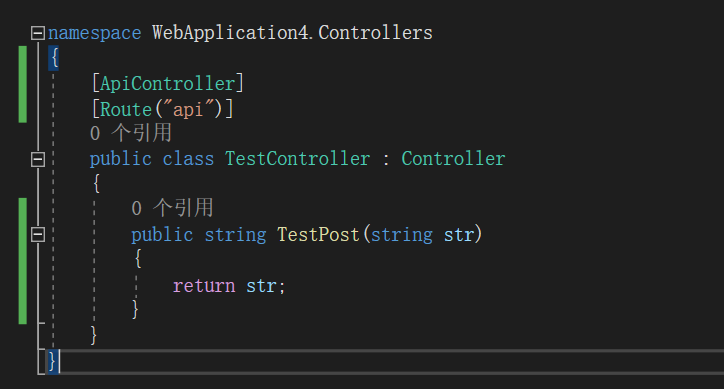
1. 在类名上面输入如图所示的内容

文本

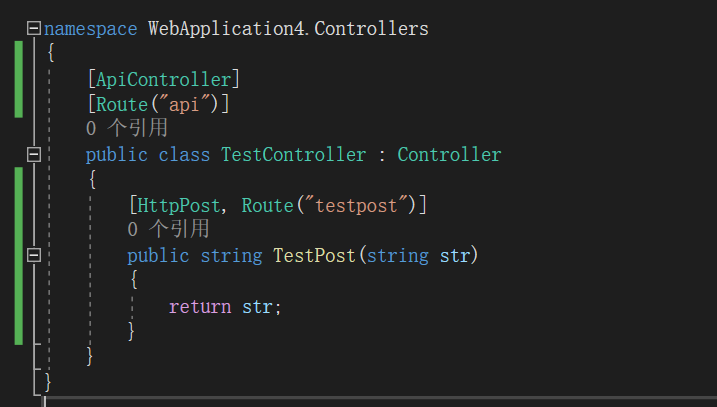
描述已自动生成

其中[Route()]括号中的内容可以理解为地址，比如我设置为api，那么我向该页面中的方法发送post信息时，便需要向：地址/api来发送。

1. 删除类中的内容，新建方法，如图所示



用来测试Post信息，需要在方法之上加上[HttpPost, Route()]



HttpPost表示为接收Post信息，Route（）与上面一样的用处，现在若要向该方法发送需要设置url为：地址/api/testpost。运行后查看控制台

文本

描述已自动生成

<https://localhost:7146>便是地址，若要访问TestPost方法，需要向<https://localhost:7146/api/testpost>发送，使用postman进行测试这种方式需要直接在地址栏中输入https://localhost:7146/api/testpost?str=测试post。

图形用户界面, 文本, 应用程序

描述已自动生成

点击send发送，接收到返回信息。

图片包含 文本

描述已自动生成

1. 由于现在大多使用Json格式发送，使用Json格式需要创建类来接收信息，新建一个类

图形用户界面, 文本

描述已自动生成

在其中设置需要接收的属性，例如：

屏幕的截图

描述已自动生成

回到控制器中，将参数更改为：[FromBody] 类名 对象名

文本

描述已自动生成

即可接收类中设置的格式匹配的Json数据，若要返回Json格式将返回值设置为类名即可

文本

低可信度描述已自动生成

为展现测试，对传入的参数进行更改，运行后使用PostMan进行测试。

电脑萤幕的截图

描述已自动生成

便可以传入与返回Json数据。

1. Get与Post几乎一致，不过需要将方法前的中括号设置为HttpGet

图片包含 文本

描述已自动生成

使用postman测试

图形用户界面, 文本, 应用程序

描述已自动生成