ConfigureServices方法用于设置应用程序所需要的服务（例如， ASP.NET Core 应用将为实体框架（Entity Framework）、标识（Identity）和 MVC 注册服务）。  
●该方法可选择定义或不定义。  
●在Configure方法配置应用程序服务之前被主机（WebHost）调用。

Application StartUP

ASP.NET Core为你的应用程序提供了处理每个请求的完整控制。StartUp类是应用程序的入口，这个类可设置配置(configuration)，并且将应用程序使用的服务连接起来。开发人员可以在Startup类中配置请求管道，该管道将用于处理应用程序的所有请求。

1，Startup类

在ASP.NETCore中，Startup类提供了应用程序的入口，而且在所有应用程序中都有Startup类。可能会存在特定环境的启动类和方法，但无论如何，Startup类都将被充当为应用程序的启动点。

Startup类能够选择性地在构造函数中接受通过依赖注入提供的依赖项。一般而言，将要被配置的应用程序的方法定义于Startup类的构造函数中，如Configuration。Startup类必须定义Configure方法，可以选择定义一个ConfigureServices方法，这些方法将在应用程序启动时被调用。

 2，Configure方法

Configure方法用于指定ASP.NET应用程序将如何响应每一个HTTP请求。简单来说，你可以配置每个请求都接收相同的响应。然而，大多数现实世界应用程序都需要比这更多的功能。更复杂的管道配置可以封装于中间件（middleware）中，并通过扩展方法添加到IApplicationBuilder上。

Configure方法必须接收一个IApplicationBuilder参数。一些额外服务，比如IHostingEnviroment或ILoggerFactory也可以被指定，并且在它们可用的情况下，这些服务将会被服务器注入进来。在下面的例子（来源于默认的Web站点模板）中，多个扩展方法被用于配置管道，以支持BrowserLink、错误页、静态文件、ASP.NET MVC以及Identity.

3，ConfigureServices方法

正如Configure,建议在IServiceColletion上使用扩展方法来包装需要大量配置细节的ConfigureServices.

4，在启动时服务可用

ASP.NET Core在应用程序启动期间提供了一些应用服务和对象。使用这些服务的方法非常简单，只要在Startup类的构造函数中包含合适的接口，或是在它的Configure方法、ConfigureServices方法中包含合适的接口即可。下面定义了在Startup类中对每个方法可用的服务。框架服务和对象包括如下内容。

IApplicationBuilder

被用于构建应用程序的请求管道。只能在Startup的Configure方法里使用。

IApplicationEnvironment

提供了访问应用程序属性，类似于ApplicationName、ApplicationVersion以及ApplicationBasePath。可以在Startup的构造函数和Configure方法中使用。