

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 IHostingEnvironment 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 IServiceCollection 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 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。