# Path

## 一、提示和注释

路径是提供文件或目录位置的字符串。 路径不必指向磁盘上的位置；例如，路径可以映射到内存中或设备上的位置。 路径的准确格式是由当前平台确定的。 例如，在某些系统上，路径可以驱动器号或卷号开始，而此元素在其他系统中是不存在的。 在某些系统上，文件路径可以包含扩展名，扩展名指示在文件中存储的信息的类型。 文件扩展名的格式是与平台相关的；例如，某些系统将扩展名的长度限制为 3 个字符，而其他系统则没有这样的限制。 当前平台还确定用于分隔路径中各元素的字符集，以及确定在指定路径时不能使用的字符集。 因为这些差异，所以 Path 类的字段以及 Path 类的某些成员的准确行为是与平台相关的。

路径可以包含绝对或相对位置信息。 绝对路径完整指定一个位置：文件或目录可被唯一标识，而与当前位置无关。 相对路径指定部分位置：当定位用相对路径指定的文件时，当前位置用作起始点。 若要确定当前目录，请调用 Directory.GetCurrentDirectory。

Path 类的大多数成员不与文件系统进行交互，且不验证路径字符串所指定的文件是否存在。 修改路径字符串（如 ChangeExtension）的 Path 类对文件系统中的文件名没有任何影响。 但是，Path 成员验证指定路径字符串的内容，如果字符串在路径字符串中包含无效字符（如 InvalidPathChars 中定义），则引发 ArgumentException。 例如，在基于 Windows 的桌面平台上，无效路径字符可能包括引号 (")、小于号 (<)、大于号 (>)、管道符号 (|)、退格 (\b)、null (\0) 以及从 16 到 18 和从 20 到 25 的 Unicode 字符。

Path 类的成员使您可以快速方便地执行常见操作，例如确定文件扩展名是否是路径的一部分，以及将两个字符串组合成一个路径名。

Path 类的所有成员都是静态的，因此无需具有路径的实例即可被调用。

注意

在接受路径作为输入字符串的成员中，路径必须是格式良好的，否则将引发异常。 例如，如果路径是完全限定的但以空格开头，则路径在类的方法中不会被修剪。 因此，路径的格式不正确，并将引发异常。 同样，路径或路径的组合不能被完全限定两次。 例如，“c:\temp c:\windows”在大多数情况下也将引发异常。 在使用接受路径字符串的方法时，请确保路径是格式良好的。

在接受路径的成员中，路径可以是指文件或仅是目录。 指定路径也可以是相对路径或者服务器和共享名称的统一命名约定 (UNC) 路径。 例如，以下都是可接受的路径：

C# 中的“c:\\MyDir\\MyFile.txt”或 Visual Basic 中的“c:\MyDir\MyFile.txt”。

C# 中的“c:\\MyDir”或 Visual Basic 中的“c:\MyDir”。

C# 中的“MyDir\\MySubdir”或 Visual Basic 中的“MyDir\MySubDir”。

C# 中的“\\\\MyServer\\MyShare”或 Visual Basic 中的“\\MyServer\MyShare”。

因为所有这些操作都是对字符串执行的，所以不可能验证结果是否在所有方案中都有效。 例如，GetExtension 方法分析您传递给它的字符串，并且从该字符串返回扩展名。 但是，这并不意味着在磁盘上存在具有该扩展名的文件。

|  |
| --- |
| 初始化 FileInfo 类的新实例，它作为文件路径的包装。 |

## 二、成员/方法

|  |  |
| --- | --- |
| **方法名称** | **方法描述** |
|  |  |
| [*AppendText*](http://www.csref.cn/vs100/method/System-IO-FileInfo-AppendText.html) | 创建一个 StreamWriter，它向 FileInfo 的此实例表示的文件追加文本。 |
| [*CopyTo(String)*](http://www.csref.cn/vs100/method/System-IO-FileInfo-CopyTo.html) | 将现有文件复制到新文件，不允许覆盖现有文件。 |
| [*CopyTo(String, Boolean)*](http://www.csref.cn/vs100/method/System-IO-FileInfo-CopyTo-1.html) | 将现有文件复制到新文件，允许覆盖现有文件。 |
| [*Create*](http://www.csref.cn/vs100/method/System-IO-FileInfo-Create.html) | 创建文件。 |
| [*CreateObjRef*](http://www.csref.cn/vs100/method/System-IO-FileInfo-CreateObjRef.html) | 创建一个对象，该对象包含生成用于与远程对象进行通信的代理所需的全部相关信息。 （继承自 MarshalByRefObject。） |
| [*CreateText*](http://www.csref.cn/vs100/method/System-IO-FileInfo-CreateText.html) | 创建写入新文本文件的 StreamWriter。 |
| [*Decrypt*](http://www.csref.cn/vs100/method/System-IO-FileInfo-Decrypt.html) | 使用 Encrypt 方法解密由当前帐户加密的文件。 |
| [*Delete*](http://www.csref.cn/vs100/method/System-IO-FileInfo-Delete.html) | 永久删除文件。 （重写 FileSystemInfo.Delete()。） |
| [*Encrypt*](http://www.csref.cn/vs100/method/System-IO-FileInfo-Encrypt.html) | 将某个文件加密，使得只有加密该文件的帐户才能将其解密。 |
| [*Equals(Object)*](http://www.csref.cn/vs100/method/System-IO-FileInfo-Equals.html) | 确定指定的 Object 是否等于当前的 Object。 （继承自 Object。） |
| [*Finalize*](http://www.csref.cn/vs100/method/System-IO-FileInfo-Finalize.html) | 允许对象在“垃圾回收”回收之前尝试释放资源并执行其他清理操作。 （继承自 Object。） |
| [*GetAccessControl()*](http://www.csref.cn/vs100/method/System-IO-FileInfo-GetAccessControl.html) | 获取 FileSecurity 对象，该对象封装当前 FileInfo 对象所描述的文件的访问控制列表 (ACL) 项。 |
| [*GetAccessControl(AccessControlSections)*](http://www.csref.cn/vs100/method/System-IO-FileInfo-GetAccessControl-1.html) | 获取 FileSecurity 对象，该对象封装当前 FileInfo 对象所描述的文件的指定类型的访问控制列表 (ACL) 项。 |
| [*GetHashCode*](http://www.csref.cn/vs100/method/System-IO-FileInfo-GetHashCode.html) | 用作特定类型的哈希函数。 （继承自 Object。） |
| [*GetLifetimeService*](http://www.csref.cn/vs100/method/System-IO-FileInfo-GetLifetimeService.html) | 检索控制此实例的生存期策略的当前生存期服务对象。 （继承自 MarshalByRefObject。） |
| [*GetObjectData*](http://www.csref.cn/vs100/method/System-IO-FileInfo-GetObjectData.html) | 设置带有文件名和附加异常信息的 SerializationInfo 对象。 （继承自 FileSystemInfo。） |
| [*GetType*](http://www.csref.cn/vs100/method/System-IO-FileInfo-GetType.html) | 获取当前实例的 Type。 （继承自 Object。） |
| [*InitializeLifetimeService*](http://www.csref.cn/vs100/method/System-IO-FileInfo-InitializeLifetimeService.html) | 获取控制此实例的生存期策略的生存期服务对象。 （继承自 MarshalByRefObject。） |
| [*MemberwiseClone()*](http://www.csref.cn/vs100/method/System-IO-FileInfo-MemberwiseClone.html) | 创建当前 Object 的浅表副本。 （继承自 Object。） |
| [*MemberwiseClone(Boolean)*](http://www.csref.cn/vs100/method/System-IO-FileInfo-MemberwiseClone-1.html) | 创建当前 MarshalByRefObject 对象的浅表副本。 （继承自 MarshalByRefObject。） |
| [*MoveTo*](http://www.csref.cn/vs100/method/System-IO-FileInfo-MoveTo.html) | 将指定文件移到新位置，并提供指定新文件名的选项。 |
| [*Open(FileMode)*](http://www.csref.cn/vs100/method/System-IO-FileInfo-Open.html) | 在指定的模式中打开文件。 |
| [*Open(FileMode, FileAccess)*](http://www.csref.cn/vs100/method/System-IO-FileInfo-Open-1.html) | 用读、写或读/写访问权限在指定模式下打开文件。 |
| [*Open(FileMode, FileAccess, FileShare)*](http://www.csref.cn/vs100/method/System-IO-FileInfo-Open-2.html) | 用读、写或读/写访问权限和指定的共享选项在指定的模式中打开文件。 |
| [*OpenRead*](http://www.csref.cn/vs100/method/System-IO-FileInfo-OpenRead.html) | 创建只读 FileStream。 |
| [*OpenText*](http://www.csref.cn/vs100/method/System-IO-FileInfo-OpenText.html) | 创建使用 UTF8 编码、从现有文本文件中进行读取的 StreamReader。 |
| [*OpenWrite*](http://www.csref.cn/vs100/method/System-IO-FileInfo-OpenWrite.html) | 创建只写 FileStream。 |
| [*Refresh*](http://www.csref.cn/vs100/method/System-IO-FileInfo-Refresh.html) | 刷新对象的状态。 （继承自 FileSystemInfo。） |
| [*Replace(String, String)*](http://www.csref.cn/vs100/method/System-IO-FileInfo-Replace.html) | 使用当前 FileInfo 对象所描述的文件替换指定文件的内容，这一过程将删除原始文件，并创建被替换文件的备份。 |
| [*Replace(String, String, Boolean)*](http://www.csref.cn/vs100/method/System-IO-FileInfo-Replace-1.html) | 使用当前 FileInfo 对象所描述的文件替换指定文件的内容，这一过程将删除原始文件，并创建被替换文件的备份。 还指定是否忽略合并错误。 |
| [*SetAccessControl*](http://www.csref.cn/vs100/method/System-IO-FileInfo-SetAccessControl.html) | 将 FileSecurity 对象所描述的访问控制列表 (ACL) 项应用于当前 FileInfo 对象所描述的文件。 |
| [*ToString*](http://www.csref.cn/vs100/method/System-IO-FileInfo-ToString.html) | 以字符串形式返回路径。 （重写 Object.ToString()。） |

## 三、示例程序

屏幕上有字

描述已自动生成