**Classe Directory**

Espone i metodi statici per la creazione, lo spostamento e l'enumerazione attraverso directory e sottodirectory.

**Spazio dei nomi System.IO:** using System.IO;

**Metodi:**

|  |  |
| --- | --- |
| **Nome** | **Descrizione** |
| [CreateDirectory(String)](http://msdn.microsoft.com/it-it/library/54a0at6s%28v=vs.110%29.aspx) | Crea tutte le directory e sottodirectory nel percorso specificato. |
| [Delete(String)](http://msdn.microsoft.com/it-it/library/62t64db3%28v=vs.110%29.aspx) | Consente di eliminare una directory vuota dal percorso specificato. |
| [Delete(String, Boolean)](http://msdn.microsoft.com/it-it/library/fxeahc5f%28v=vs.110%29.aspx) | Elimina la directory specificata e, se indicate, le sottodirectory e i file in essa contenuti. |
| [Exists](http://msdn.microsoft.com/it-it/library/system.io.directory.exists%28v=vs.110%29.aspx) | Consente di stabilire se il percorso specificato fa riferimento a una directory esistente sul disco. |
| [GetCurrentDirectory](http://msdn.microsoft.com/it-it/library/system.io.directory.getcurrentdirectory%28v=vs.110%29.aspx) | Recupera la directory di lavoro corrente dell'applicazione. |
| [GetDirectories(String)](http://msdn.microsoft.com/it-it/library/c1sez4sc%28v=vs.110%29.aspx) | Ottiene i nomi delle sottodirectory, inclusi i relativi percorsi, nella directory specificata. |
| [GetDirectories(String, String)](http://msdn.microsoft.com/it-it/library/6ff71z1w%28v=vs.110%29.aspx) | Ottiene i nomi delle sottodirectory (incluso il relativo percorso) che soddisfano il modello di ricerca specificato nella directory corrente. |
| [GetDirectories(String, String, SearchOption)](http://msdn.microsoft.com/it-it/library/ms143314%28v=vs.110%29.aspx) | Ottiene i nomi delle sottodirectory (incluso il relativo percorso) che corrispondono a un modello di ricerca specificato nella directory corrente e con ricerca facoltativa nelle sottodirectory. |
| [GetDirectoryRoot](http://msdn.microsoft.com/it-it/library/system.io.directory.getdirectoryroot%28v=vs.110%29.aspx) | Restituisce le informazioni sul volume, sulla directory radice, o su entrambi, per il percorso specificato. |
| [GetFiles(String)](http://msdn.microsoft.com/it-it/library/07wt70x2%28v=vs.110%29.aspx) | Restituisce i nomi dei file, inclusi i percorsi, nella directory specificata. |
| [GetFiles(String, String)](http://msdn.microsoft.com/it-it/library/wz42302f%28v=vs.110%29.aspx) | Restituisce i nomi dei file (inclusi i percorsi) corrispondenti al criterio di ricerca specificato nella directory specificata. |
| [GetFiles(String, String, SearchOption)](http://msdn.microsoft.com/it-it/library/ms143316%28v=vs.110%29.aspx) | Restituisce i nomi dei file (inclusi i percorsi) che corrispondono ai criteri di ricerca specificati nella directory specificata, utilizzando un valore per determinare se eseguire la ricerca nelle sottodirectory. |
| [GetFileSystemEntries(String)](http://msdn.microsoft.com/it-it/library/416y61e6%28v=vs.110%29.aspx) | Restituisce i nomi di tutti i file e di tutte le sottodirectory nella directory specificata. |
| [GetFileSystemEntries(String, String)](http://msdn.microsoft.com/it-it/library/d3eayw32%28v=vs.110%29.aspx) | Restituisce una matrice di voci di file system corrispondenti ai criteri di ricerca specificati. |
| [GetFileSystemEntries(String, String, SearchOption)](http://msdn.microsoft.com/it-it/library/dd383460%28v=vs.110%29.aspx) | Restituisce una matrice di tutti i nomi di file e di directory che corrispondono a un criterio di ricerca in un percorso specificato e con ricerca facoltativa nelle sottodirectory. |
| [GetLogicalDrives](http://msdn.microsoft.com/it-it/library/system.io.directory.getlogicaldrives%28v=vs.110%29.aspx) | Recupera i nomi delle unità logiche sul computer nel formato "<lettera unità>:\". |
| [GetParent](http://msdn.microsoft.com/it-it/library/system.io.directory.getparent%28v=vs.110%29.aspx) | Recupera la directory padre del percorso specificato, inclusi il percorso assoluto e relativo. |
| [Move](http://msdn.microsoft.com/it-it/library/system.io.directory.move%28v=vs.110%29.aspx) | Sposta un file o una directory e il suo contenuto in un nuovo percorso. |
| [SetCurrentDirectory](http://msdn.microsoft.com/it-it/library/system.io.directory.setcurrentdirectory%28v=vs.110%29.aspx) | Imposta la directory di lavoro corrente dell'applicazione sulla directory specificata. |

**Note**

Utilizzare la classe Directory per le operazioni tipiche, quali copia, spostamento, ridenominazione, creazione ed eliminazione di directory. È possibile utilizzare la classe Directory anche per ottenere e impostare le informazioni di DateTime relative alla creazione, all'accesso e alla scrittura di una directory.

Poiché tutti i metodi di Directory sono statici, potrebbe essere più efficace utilizzare un metodo Directory anziché un metodo dell'istanza [DirectoryInfo](http://msdn.microsoft.com/it-it/library/system.io.directoryinfo%28v=vs.110%29.aspx) corrispondente, se si desidera eseguire una sola operazione. La maggior parte dei metodi di Directory richiede il percorso alla directory modificata.

I metodi statici della classe Directory eseguono controlli di sicurezza su tutti i metodi. Se si intende riutilizzare più volte un oggetto, si consideri l'utilizzo del corrispondente metodo dell'istanza di [DirectoryInfo](http://msdn.microsoft.com/it-it/library/system.io.directoryinfo%28v=vs.110%29.aspx), dal momento che il controllo di sicurezza non sarà sempre necessario.

**Esempio:**

using System;

using System.IO;

using System.Collections.Generic;

public class Program

{

public static void Main()

{

string dirPath = Environment.GetFolderPath(Environment.SpecialFolder.DesktopDirectory);

//Directory corrente

Console.WriteLine("Directory corrente: {0}", Directory.GetCurrentDirectory());

Directory.SetCurrentDirectory(dirPath);

Console.WriteLine("Directory corrente: {0}", Directory.GetCurrentDirectory());

//Directory Root

Console.WriteLine("\nDirectory root: {0}", Directory.GetDirectoryRoot(dirPath));

//Directory Parent

Console.WriteLine("\nDirectory parent: {0}", Directory.GetParent(dirPath));

//Logical Drives.

Console.WriteLine("\n--- Logical Drives ---\n");

string[] logDrive = Directory.GetLogicalDrives();

foreach (string Drive in logDrive)

Console.WriteLine("Drive '{0}'.", Drive);

//Files nella Directory.

Console.WriteLine("\n--- FILES ---\n");

string[] fileEntries = Directory.GetFiles(dirPath);

foreach (string fileName in fileEntries)

Console.WriteLine("Processed file '{0}'.", fileName);

//SubDirectories della Directory.

Console.WriteLine("\n--- SUBDIRECTORIES ---\n");

string[] subdirectoryEntries = Directory.GetDirectories(dirPath);

foreach (string subdirectory in subdirectoryEntries)

Console.WriteLine("Processed subDirectory '{0}'.", subdirectory);

//Files c\* nella directory.

Console.WriteLine("\n--- FILES c\* ---\n");

fileEntries = Directory.GetFiles(dirPath, "c\*", SearchOption.TopDirectoryOnly);

foreach (string fileName in fileEntries)

Console.WriteLine("Processed file '{0}'.", fileName);

Console.ReadKey();

}

}

**Output:**

Directory corrente: C:\Users\roberto.veneziani\Documents\Visual Studio 2012\Proj

ects\ConsoleApplication1\ConsoleApplication1\bin\Debug

Directory corrente: C:\Users\roberto.veneziani\Desktop

Directory root: C:\

Directory parent: C:\Users\roberto.veneziani

--- Logical Drives ---

Drive 'C:\'.

Drive 'D:\'.

Drive 'E:\'.

Drive 'F:\'.

Drive 'L:\'.

Drive 'M:\'.

Drive 'N:\'.

Drive 'Z:\'.

--- FILES ---

Processed file 'C:\Users\roberto.veneziani\Desktop\dotNetFx45\_Full\_setup.exe'.

Processed file 'C:\Users\roberto.veneziani\Desktop\eserc9.pdf'.

Processed file 'C:\Users\roberto.veneziani\Desktop\Esercizi a casa.txt'.

Processed file 'C:\Users\roberto.veneziani\Desktop\Orario Classi.pdf'.

Processed file 'C:\Users\roberto.veneziani\Desktop\Orario Docenti.pdf'.

Processed file 'C:\Users\roberto.veneziani\Desktop\PhotoFiltre-en-653.zip'.

Processed file 'C:\Users\roberto.veneziani\Desktop\Prova.rar'.

Processed file 'C:\Users\roberto.veneziani\Desktop\RarToZip.txt'.

--- SUBDIRECTORIES ---

Processed subDirectory 'C:\Users\roberto.veneziani\Desktop\PhotoFiltre'.

Processed subDirectory 'C:\Users\roberto.veneziani\Desktop\Temp'.

Processed subDirectory 'C:\Users\roberto.veneziani\Desktop\Zucchini'.

--- FILES c\* ---

Processed file 'C:\Users\roberto.veneziani\Desktop\C# - Classe File.docx'.

Processed file 'C:\Users\roberto.veneziani\Desktop\C# - StreamReader & StreamWri

ter.docx'.

Processed file 'C:\Users\roberto.veneziani\Desktop\CfakepathElenco\_alunni\_Proget

to\_Scuola\_Sostenibile.pdf'.

Processed file 'C:\Users\roberto.veneziani\Desktop\Circ. 354 Convocazione Scruti

ni 1 Quad.pdf'.

**Classe File**

Fornisce i metodi statici per creare, copiare, eliminare, spostare e aprire i file, nonché supporto per la creazione di oggetti FileStream.

**Spazio dei nomi System.IO:** using System.IO;

**Metodi:**

|  |  |
| --- | --- |
| **Nome** | **Descrizione** |
| [Copy(String, String)](http://msdn.microsoft.com/it-it/library/c6cfw35a(v=vs.110).aspx) | Copia un file esistente in un nuovo file. Non è consentito sovrascrivere un file con lo stesso nome. |
| [Copy(String, String, Boolean)](http://msdn.microsoft.com/it-it/library/9706cfs5(v=vs.110).aspx) | Copia un file esistente in un nuovo file. È consentito sovrascrivere un file con lo stesso nome. |
| [Create(String)](http://msdn.microsoft.com/it-it/library/d62kzs03(v=vs.110).aspx) | Crea o sovrascrive un file nel percorso specificato. |
| [Delete](http://msdn.microsoft.com/it-it/library/system.io.file.delete(v=vs.110).aspx) | Elimina il file specificato. |
| [Exists](http://msdn.microsoft.com/it-it/library/system.io.file.exists(v=vs.110).aspx) | Determina se il file specificato esiste. |
| [Move](http://msdn.microsoft.com/it-it/library/system.io.file.move(v=vs.110).aspx) | Sposta il file specificato in un nuovo percorso, consentendo di specificare per esso un nuovo nome. |
| [Replace(String, String, String)](http://msdn.microsoft.com/it-it/library/9d9h163f(v=vs.110).aspx) | Sostituisce il contenuto di un file specificato con il contenuto di un altro file, eliminando il file originale e creando una copia di backup del file sostituito. |

**Note**

Utilizzare la classe File per le operazioni tipiche, quali copia, spostamento, ridenominazione, creazione, apertura, eliminazione e aggiunta a file. È possibile utilizzare la classe File anche per ottenere e impostare gli attributi dei file o le informazioni di DateTime relative alla creazione, all'accesso e alla scrittura di un file.

Molti metodi File restituiscono altri tipi di I/O quando si creano o aprono file. È possibile utilizzare gli altri tipi per modificare ulteriormente un file. Per ulteriori informazioni, vedere i membri specifici di File, quale OpenText, CreateText o Create.

Poiché tutti i metodi File sono statici, potrebbe essere efficace utilizzare un metodo File anziché un metodo dell'istanza FileInfo corrispondente, se si desidera eseguire una sola operazione. Tutti i metodi di File richiedono il percorso al file modificato.

I metodi statici della classe File eseguono controlli di sicurezza su tutti i metodi. Se si intende riutilizzare più volte un oggetto, si consideri l'utilizzo del corrispondente metodo dell'istanza FileInfo, poiché il controllo di sicurezza non sarà sempre necessario.

Per impostazione predefinita, l'accesso completo in lettura/scrittura ai nuovi file è consentito a tutti gli utenti.

**Esempio:**

using System;

using System.IO;

using System.IO.Compression;

namespace ConsoleApplication

{

class Program

{

static void Main(string[] args)

{

string myDesktop = Environment.GetFolderPath(Environment.SpecialFolder.DesktopDirectory);

string MyText = myDesktop + "\\MyTest.txt";

string MyTextBkp = myDesktop + "\\MyTestBkp.txt";

string MyTextRename = myDesktop + "\\MyTestRename.txt";

File.WriteAllText(MyText, "Esempio con classe File");

// Delete the file if it exists.

if (File.Exists(MyTextBkp))

{

File.Delete(MyTextBkp);

}

// Will not overwrite if the destination file already exists.

File.Copy(MyText, MyTextBkp);

// Will overwrite if the destination file already exists.

File.Copy(MyText, MyTextBkp, true);

Console.WriteLine("\n{0} è stato copiato in {1}", MyText, MyTextBkp);

string dirTemp = myDesktop + "\\Temp";

if (Directory.Exists(dirTemp))

Directory.Delete(dirTemp, true);

Directory.CreateDirectory(dirTemp);

string MyText2 = dirTemp + "\\MyTest.txt";

// Ensure that the target does not exist.

if (File.Exists(MyText2))

File.Delete(MyText2);

// Move the file.

File.Move(MyText, MyText2);

Console.WriteLine("\n{0} è stato spostato in {1}", MyText, MyText2);

// Rename the file.

if (File.Exists(MyTextRename))

File.Delete(MyTextRename);

File.Move(MyTextBkp, MyTextRename);

Console.WriteLine("\n{0} è stato rinominato in {1}", MyTextBkp, MyTextRename);

//Create the File

File.WriteAllText(MyText, "Contenuto da sostituire...");

// Replace the file. Whithout backup copy

File.Replace(MyText2, MyText, null);

File.Copy(MyText, MyText2);

// Replace the file. With backup copy

File.Replace(MyText2, MyText, MyTextBkp);

Console.WriteLine("\nIl contenuto del file {0} è stato sostituito", MyText);

Console.ReadKey();

}

}

}

**Output:**

C:\Users\ERREVI\Desktop\MyTest.txt è stato copiato in C:\Users\ERREVI\Desktop\My

TestBkp.txt

C:\Users\ERREVI\Desktop\MyTest.txt è stato spostato in C:\Users\ERREVI\Desktop\T

emp\MyTest.txt

C:\Users\ERREVI\Desktop\MyTestBkp.txt è stato rinominato in C:\Users\ERREVI\Desk

top\MyTestRename.txt

Il contenuto del file C:\Users\ERREVI\Desktop\MyTest.txt è stato sostituito