

Returns an array of abstract pathnames representing the files and directories of the specified path.

#region License and Copyright

/* -----

* Dotnet Commons IO

*

*

* This library is free software; you can redistribute it and/or modify it

* under the terms of the GNU Lesser General Public License as published by

* the Free Software Foundation; either version 2.1 of the License, or

* (at your option) any later version.

*

* This library is distributed in the hope that it will be useful, but

* WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY

* or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License

* for more details.

*

* You should have received a copy of the GNU Lesser General Public License

* along with this library; if not, write to the

*

* Free Software Foundation, Inc.,

* 59 Temple Place,

* Suite 330,

```
* Boston,  
* MA 02111-1307  
* USA  
*  
* _____  
*/  
#endregion
```

```
using System;  
using System.Collections;  
using System.Globalization;  
using System.IO;
```

```
namespace Dotnet.Commons.IO  
{
```

```
///
```

```
///
```

```
/// This class provides basic facilities for manipulating  
files and file paths.
```

```
///
```

```
///
```

File-related methods

```
/// There are methods to
```

```
/// /// copy a file to another file,
```

```
/// compare the content of 2 files,
```

```
/// delete files using the wildcard character,
```

```
/// etc
```

```
/// ///
```

```
///
```

```
public sealed class FileUtils
```

```
{
```

```
/// _____
```

```
///
```

```
/// Returns an array of abstract pathnames representing the
```

files and directories of the specified path.

```
///
```

```
/// The abstract pathname to list its children. /// An array of  
abstract pathnames children of the path specified or
```

```
/// null if the path is not a directory
```

```
/// _____
```

```
public static FileInfo[] GetFiles(FileInfo path)
```

```
{
```

```
if ((path.Attributes & FileAttributes.Directory) > 0)
```

```
{
```

```
String[] fullpathnames =
```

```
Directory.GetFileSystemEntries(path.FullName);
```

```
FileInfo[] result = new FileInfo[fullpathnames.Length];
```

```
for (int i = 0; i < result.Length; i++) result[i] = new  
FileInfo(fullpathnames[i]); return result; } return null; } }
```

```
} [/csharp]
```