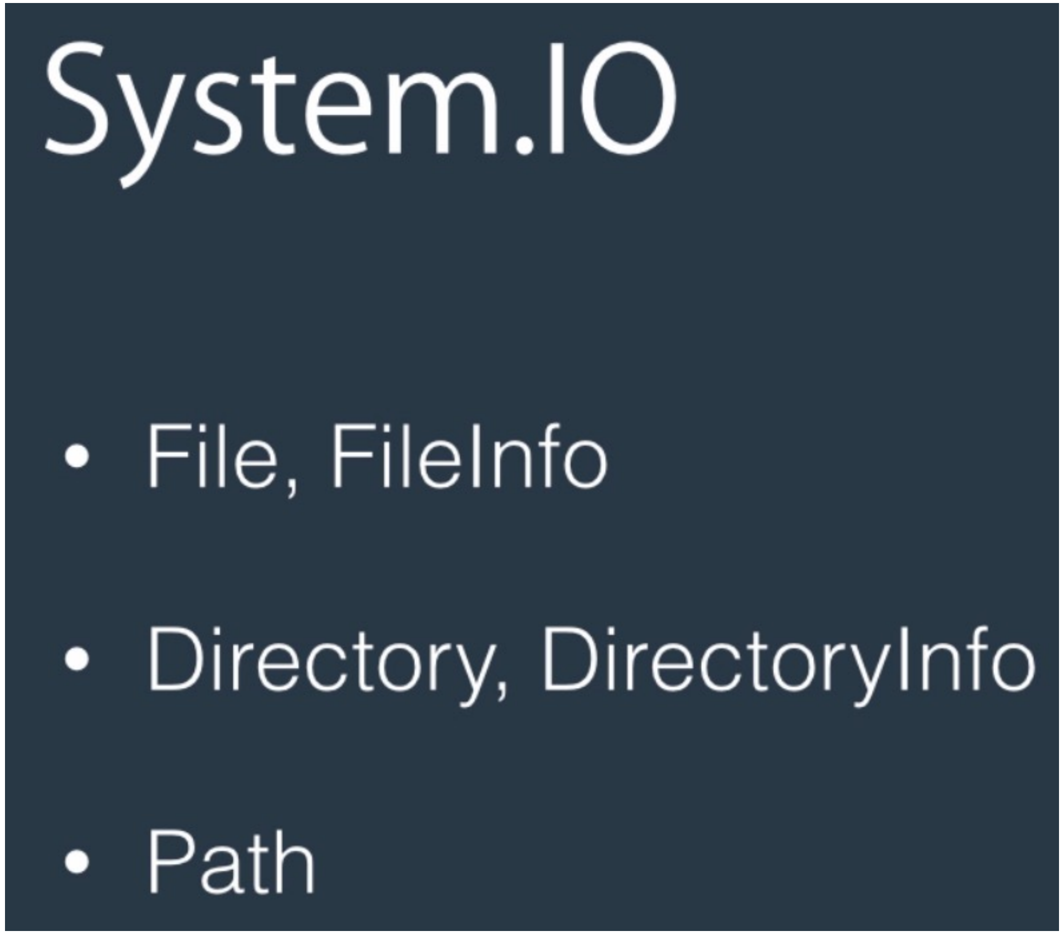
Working with Files

System.IO Namespace

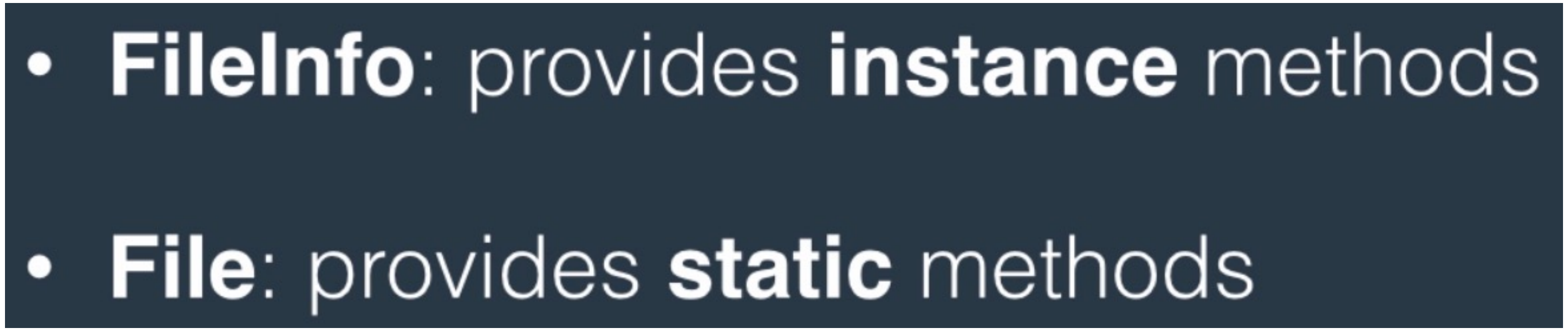


System.IO

- File, FileInfo
- Directory, DirectoryInfo
- Path

File, FileInfo

- Provide methods for creating, copying, deleting, moving, and opening of files.



- **FileInfo**: provides **instance** methods
- **File**: provides **static** methods

Differecnes

The difference between them so file and file info classes. Both provide methods for creating copying deleting moving and opening files. They have very similar interfaces.

The only difference is FileInfo provides instance methods where whereas file provides static methods So why do we need different classes for static or instance methods.

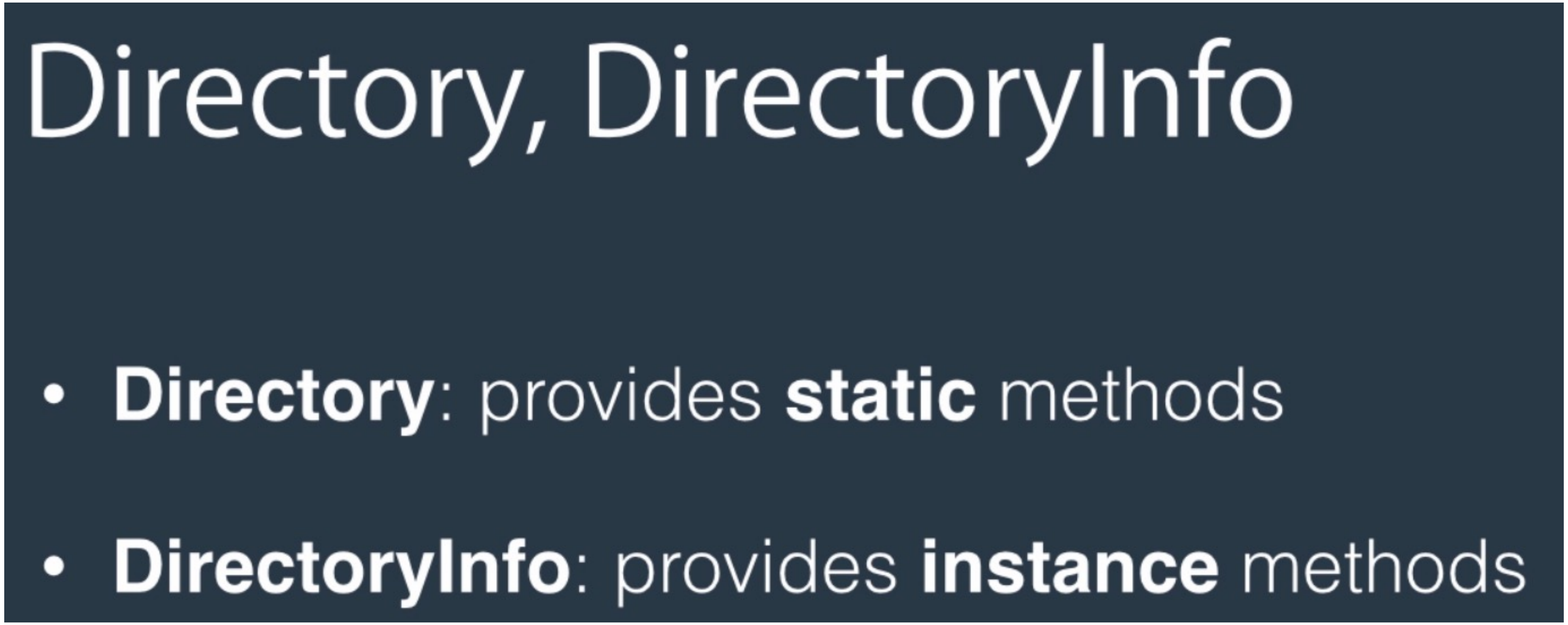
The difference is if you're going to have a small number of operations that say good attributes about the file it's more convenient to access the static methods of the file class.

But the problem with this static methods is every time you call it a static methods some security checking is done by the operating system to make sure that the current user has access to the file.

So if you're going to have a large number of operations this is going to affect the performance of your application. So in that case it's more efficient to create a filing for class and access all its instance methods This way security check in is done only once during creation of a file object.

File, FileInfo

- Create()
- Copy()
- Delete()
- Exists()
- GetAttributes()
- Move()
- ReadAllText()

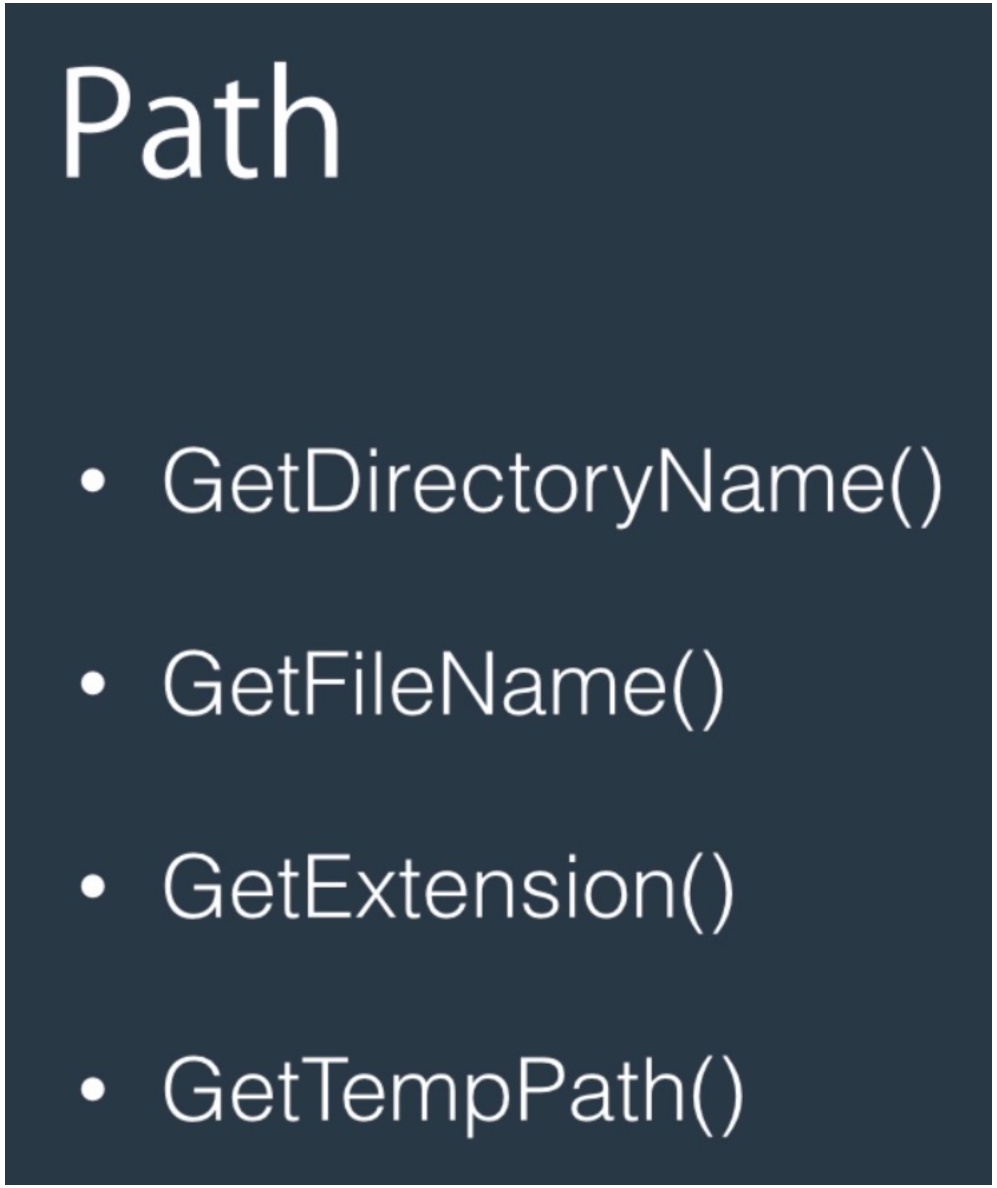


Directory, DirectoryInfo

- **Directory**: provides **static** methods
- **DirectoryInfo**: provides **instance** methods

Directory, DirectoryInfo

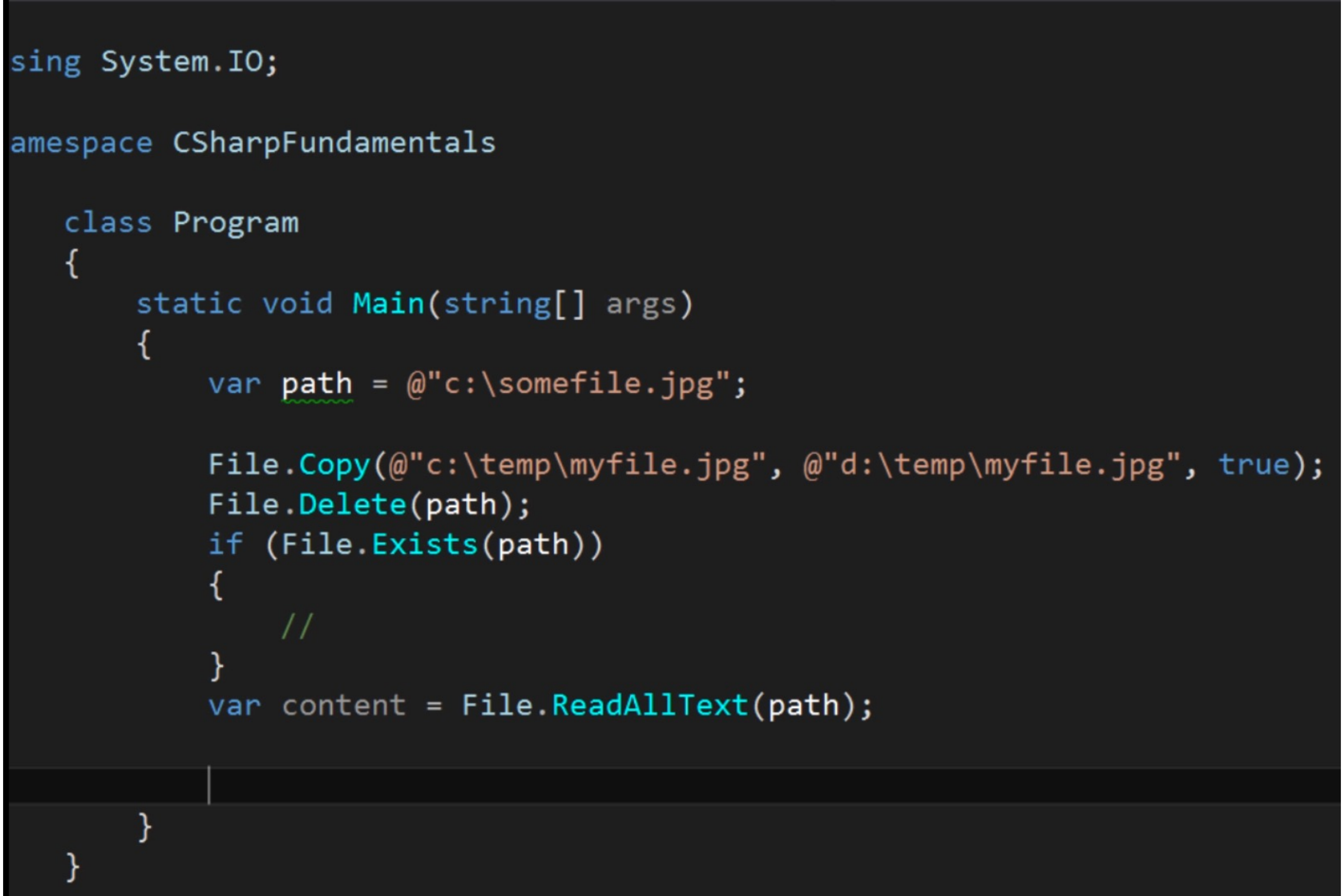
- CreateDirectory()
- Delete()
- Exists()
- GetCurrentDirectory()
- GetFiles()
- Move()
- GetLogicalDrives()



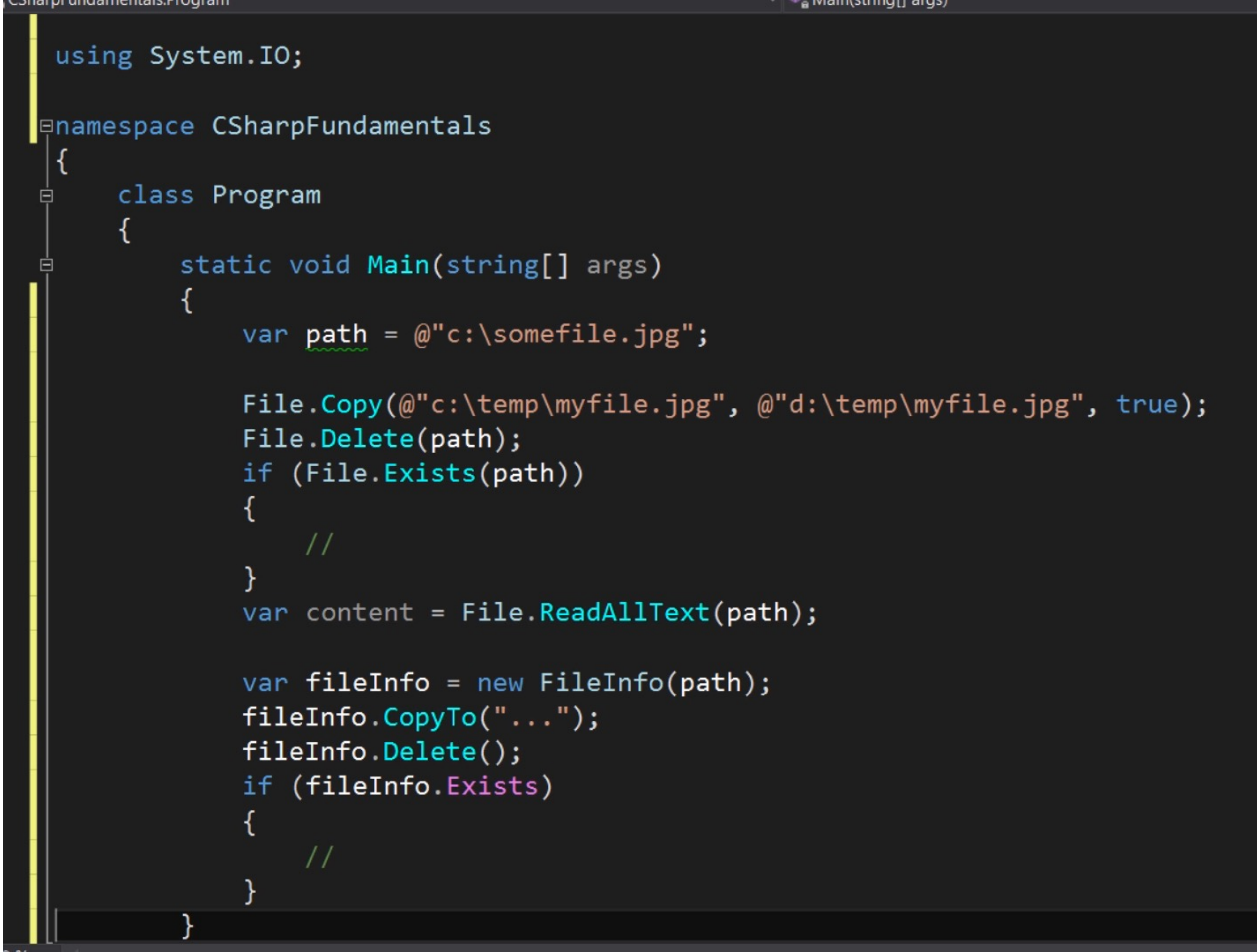
Path

- GetDirectoryName()
- GetFileName()
- GetExtension()
- GetTempPath()

We also have the path class which provides methods to work with a string that contains a file or directory path information.



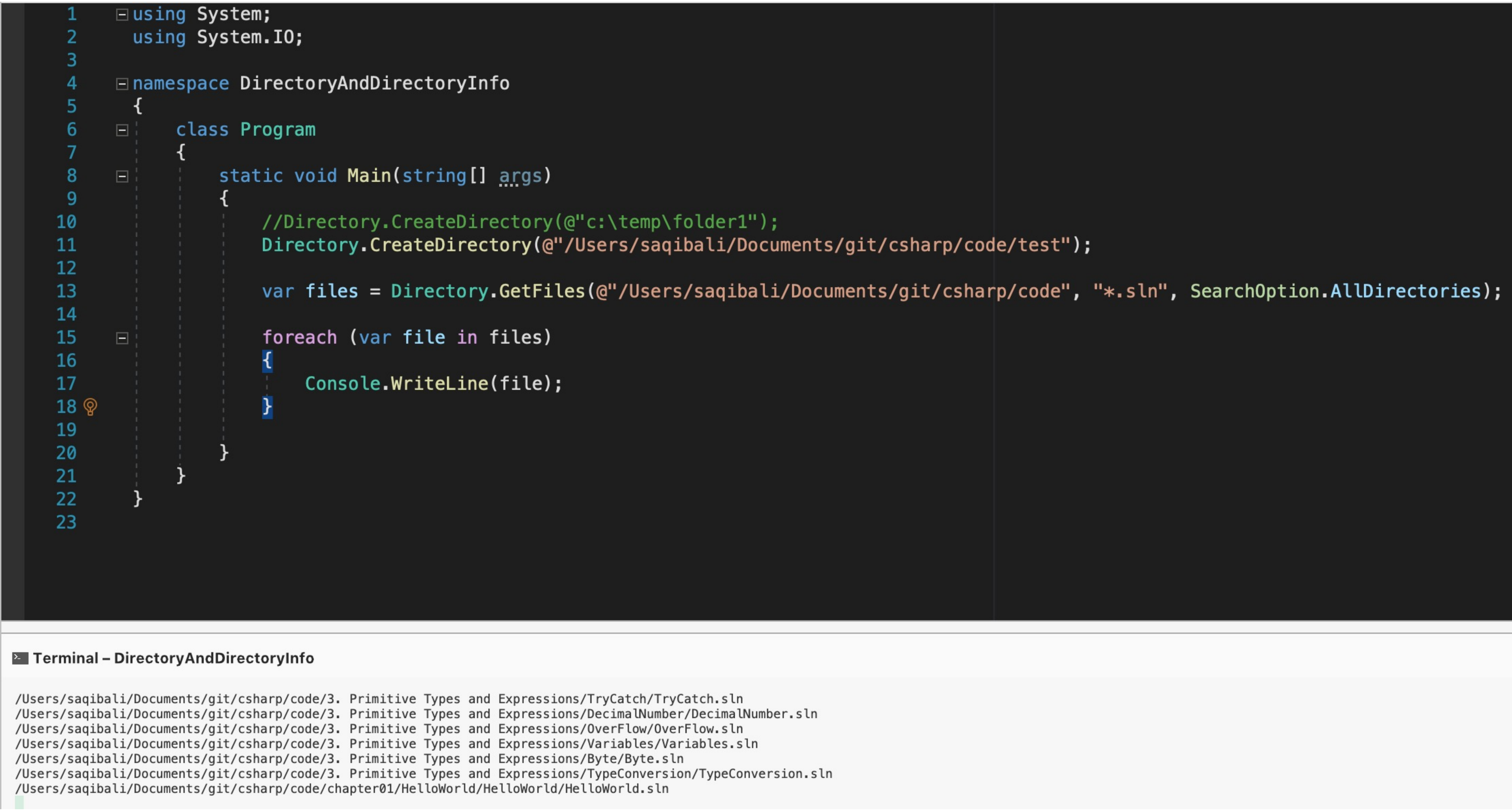
```
sing System.IO;

amespace CSharpFundamentals

   class Program
   {
       static void Main(string[] args)
       {
           var path = @"c:\somefile.jpg";

           File.Copy(@"c:\temp\myfile.jpg", @"d:\temp\myfile.jpg", true);
           File.Delete(path);
           if (File.Exists(path))
           {
               //
           }
           var content = File.ReadAllText(path);


       }
   }
```



```
using System.IO;

namespace CSharpFundamentals
{
    class Program
    {
        static void Main(string[] args)
        {
            var path = @"c:\somefile.jpg";

            File.Copy(@"c:\temp\myfile.jpg", @"d:\temp\myfile.jpg", true);
            File.Delete(path);
            if (File.Exists(path))
            {
                //
            }
            var content = File.ReadAllText(path);

            var fileInfo = new FileInfo(path);
            fileInfo.CopyTo("...");
            fileInfo.Delete();
            if (fileInfo.Exists)
            {
                //
            }
        }
```

Note that FileInfo for does not have a read all text method. It's only available as a static method in the file class. With the FileInfo we need to call the open read method which returns a file stream but the stream is a different way of reading a file. It's just a little bit complex. Later we will see a very simple way to read the file.

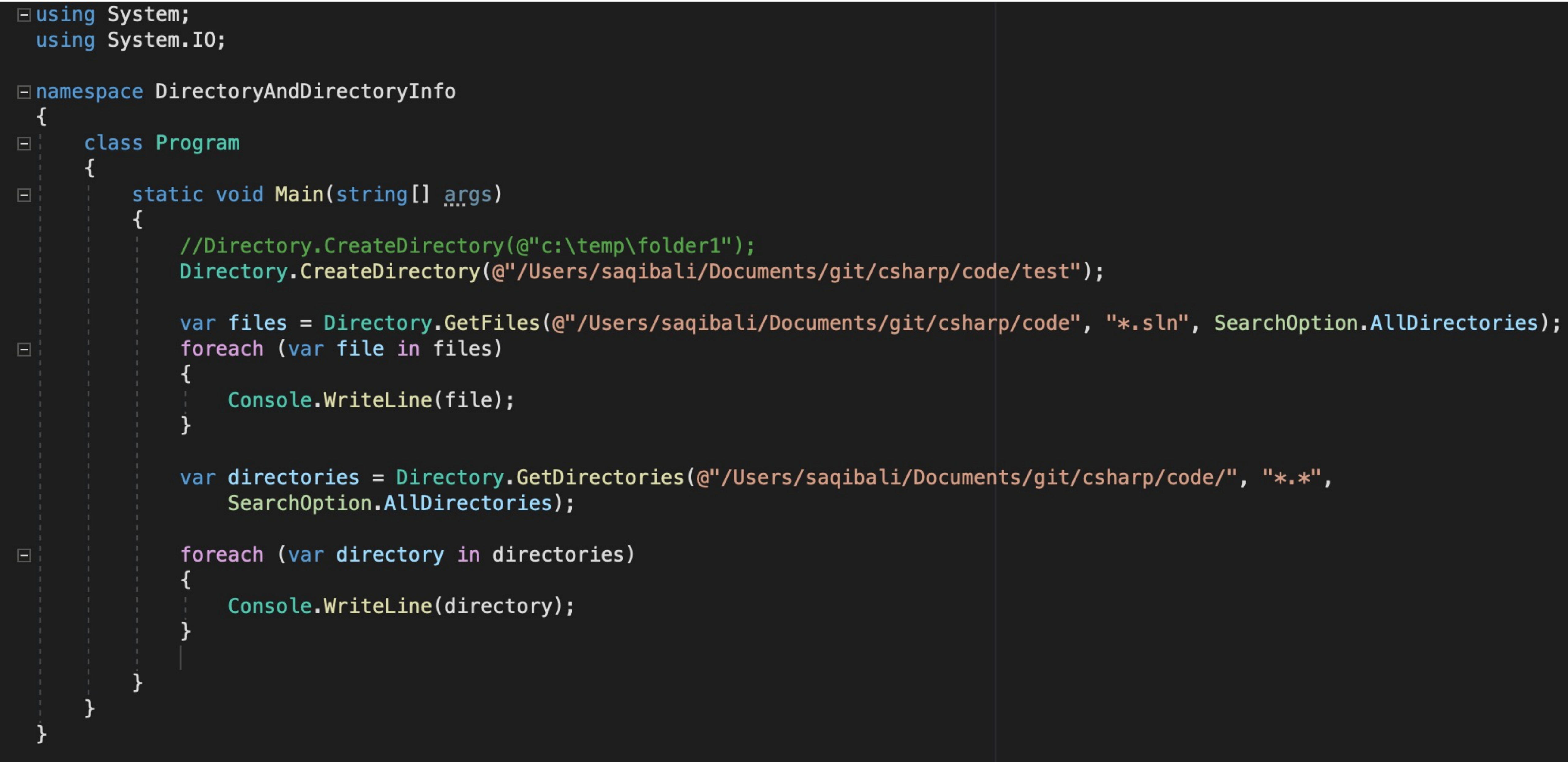
Demo Directory and DirectoryInfo



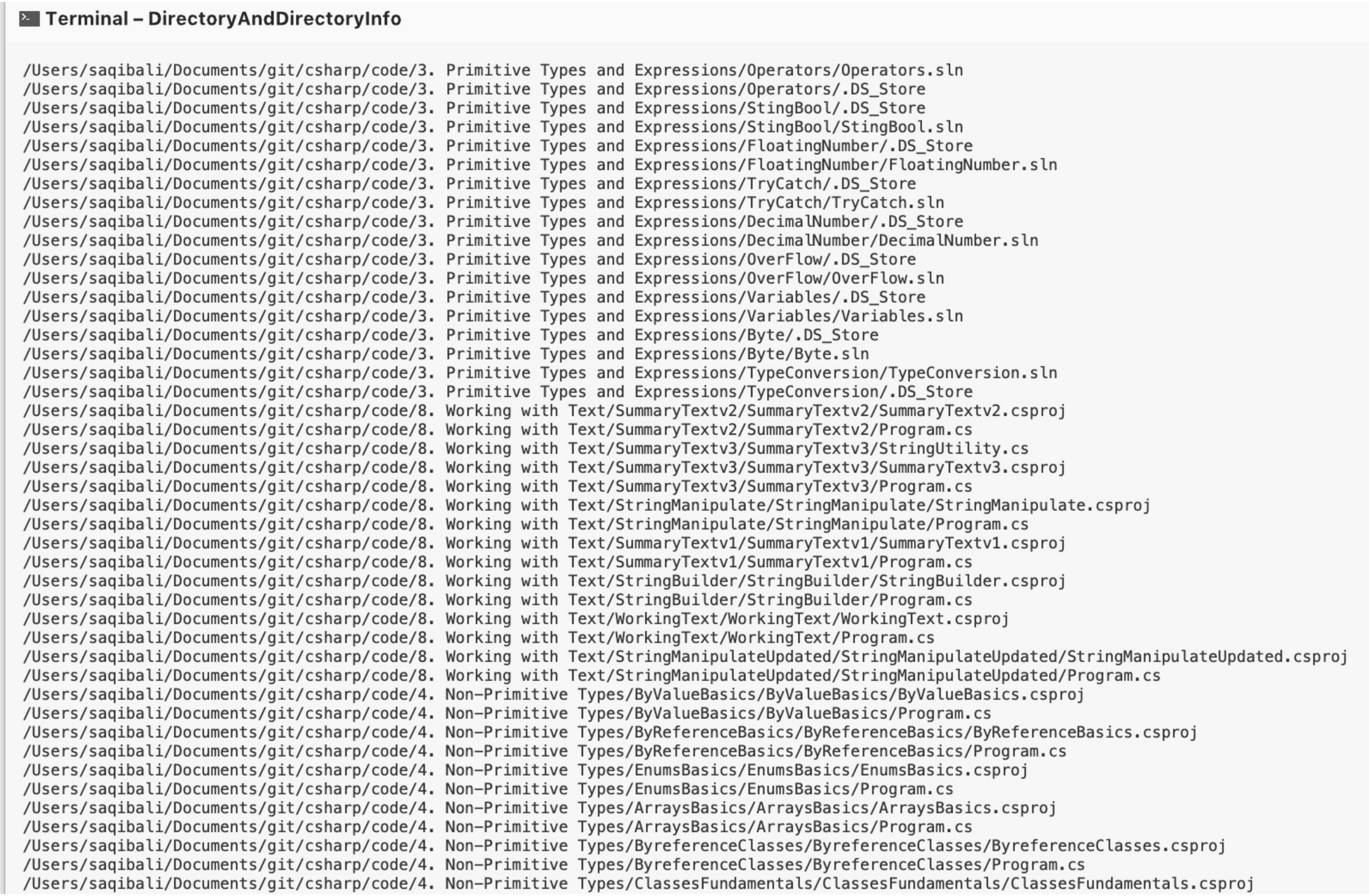
```csharp
using System;
using System.IO;

namespace DirectoryAndDirectoryInfo
{
    class Program
    {
        static void Main(string[] args)
        {
            //Directory.CreateDirectory(@"c:\temp\folder1");
            Directory.CreateDirectory(@"/Users/saqibali/Documents/git/csharp/code/test");

            var files = Directory.GetFiles(@"/Users/saqibali/Documents/git/csharp/code", "*.sln", SearchOption.AllDirectories);

            foreach (var file in files)
            {
                Console.WriteLine(file);
            }

        }
    }
}

```

Terminal – DirectoryAndDirectoryInfo

```
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/TryCatch/TryCatch.sln
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/DecimalNumber/DecimalNumber.sln
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/OverFlow/OverFlow.sln
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/Variables/Variables.sln
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/Byte/Byte.sln
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/TypeConversion/TypeConversion.sln
/Users/saqibali/Documents/git/csharp/code/chapter01/HelloWorld/HelloWorld/HelloWorld.sln
```

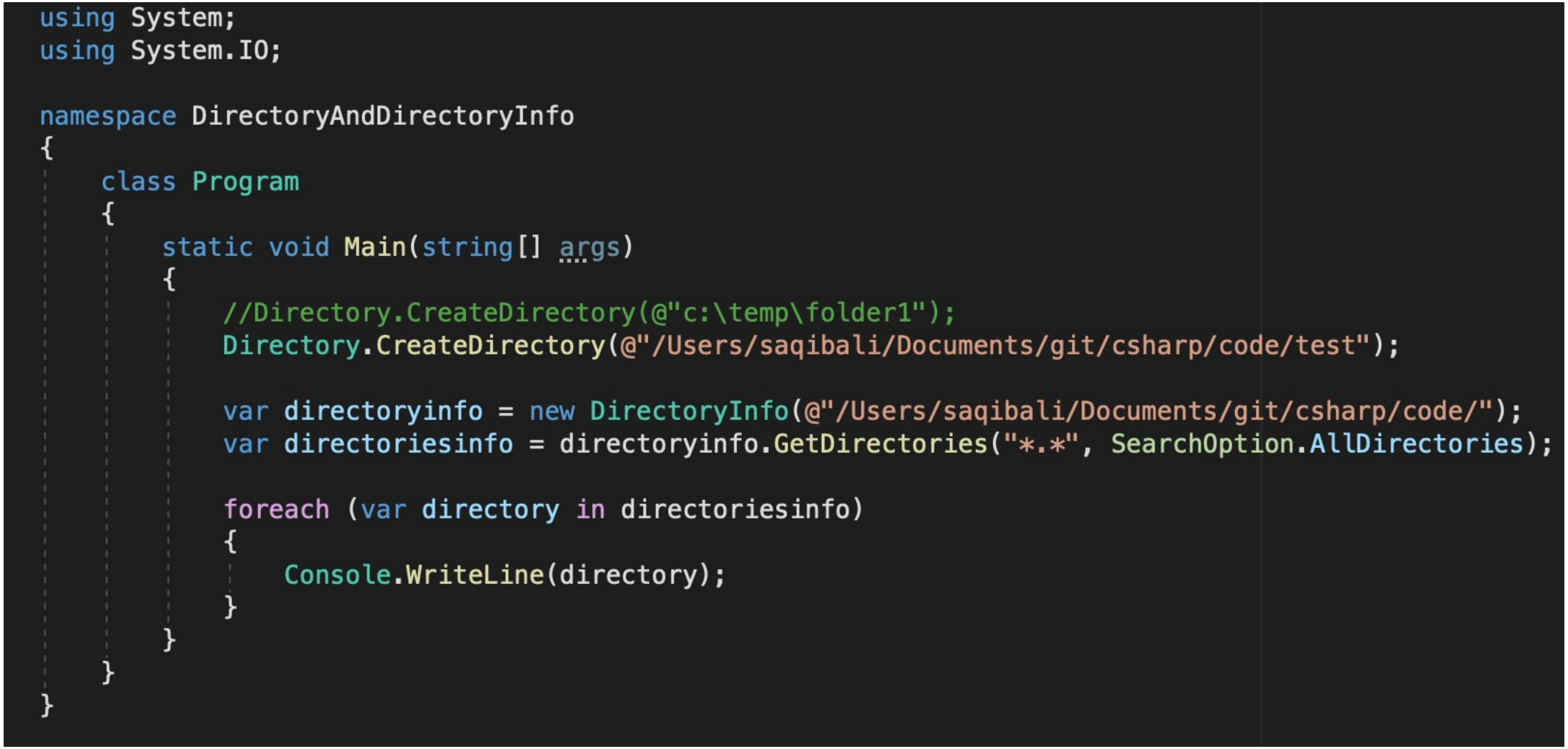


```
using System;
using System.IO;

namespace DirectoryAndDirectoryInfo
{
    class Program
    {
        static void Main(string[] args)
        {
            //Directory.CreateDirectory(@"c:\temp\folder1");
            Directory.CreateDirectory(@"/Users/saqibali/Documents/git/csharp/code/test");

            var files = Directory.GetFiles(@"/Users/saqibali/Documents/git/csharp/code", "*.sln", SearchOption.AllDirectories);
            foreach (var file in files)
            {
                Console.WriteLine(file);
            }

            var directories = Directory.GetDirectories(@"/Users/saqibali/Documents/git/csharp/code/", "*.*",
                SearchOption.AllDirectories);

            foreach (var directory in directories)
            {
                Console.WriteLine(directory);
            }

        }
    }
}
```

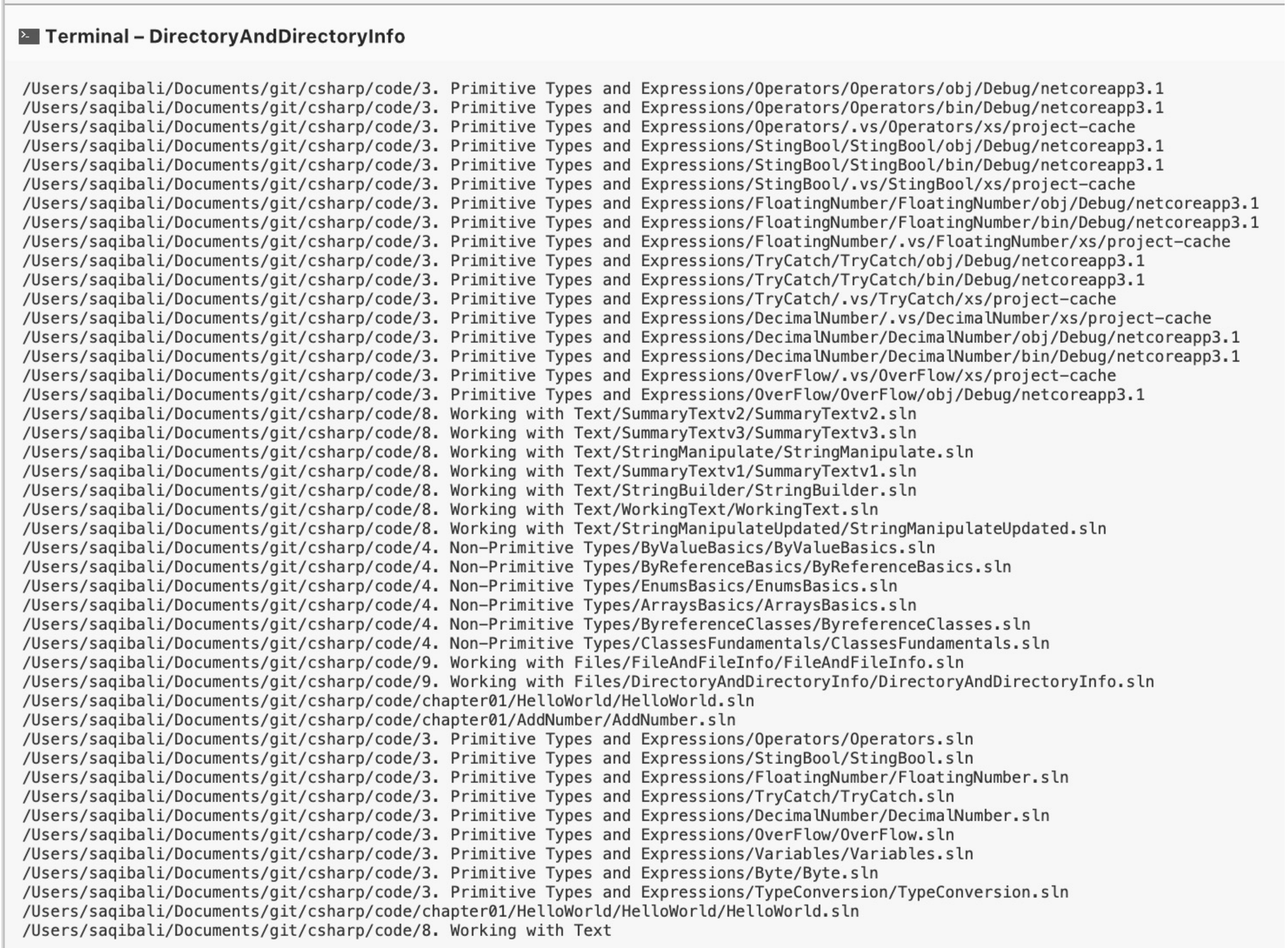


Terminal – DirectoryAndDirectoryInfo

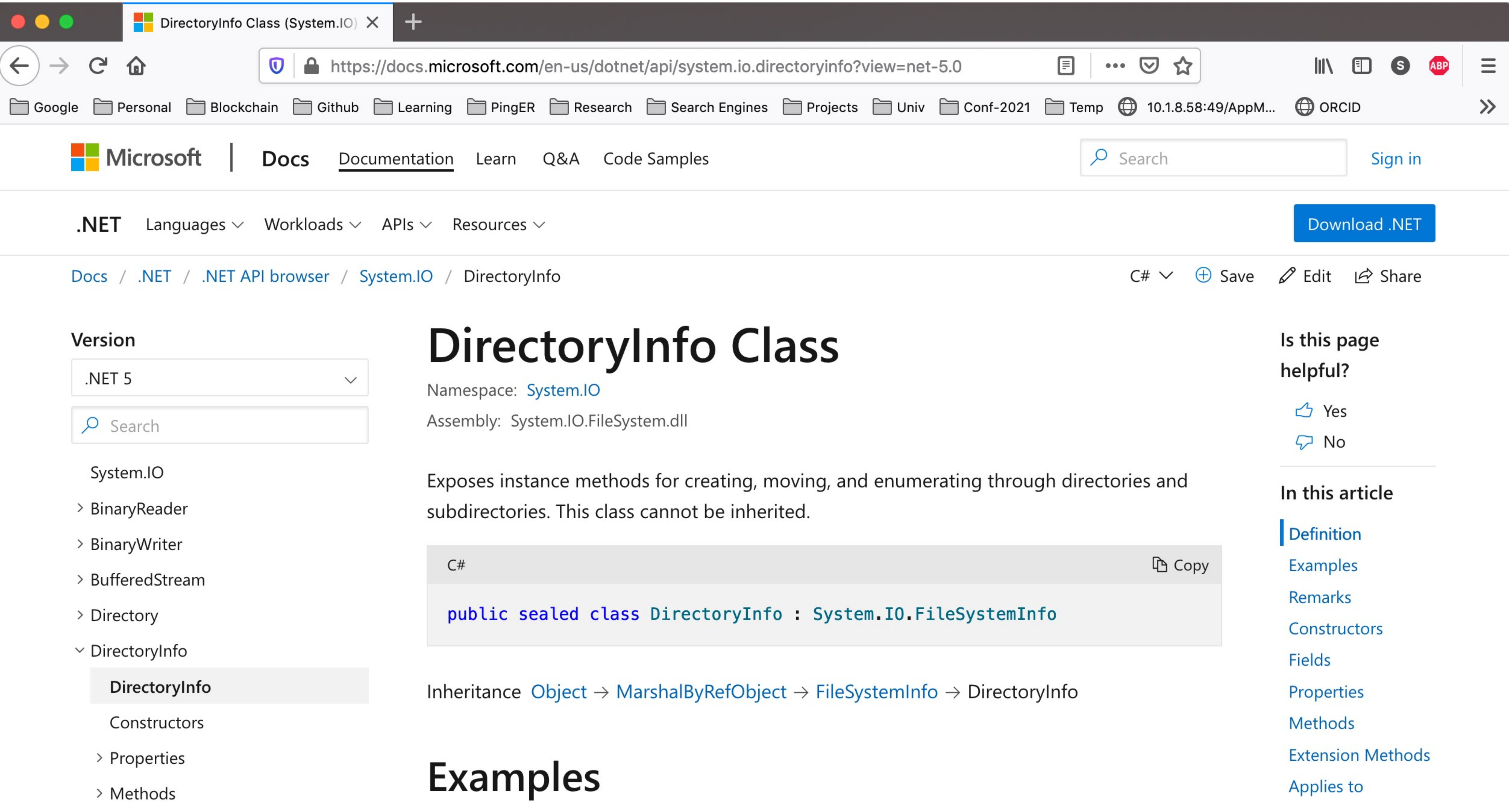
```
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/Operators/Operators.sln
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/Operators/.DS_Store
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/StingBool/.DS_Store
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/StingBool/StingBool.sln
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/FloatingNumber/.DS_Store
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/FloatingNumber/FloatingNumber.sln
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/TryCatch/.DS_Store
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/TryCatch/TryCatch.sln
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/DecimalNumber/.DS_Store
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/DecimalNumber/DecimalNumber.sln
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/OverFlow/.DS_Store
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/OverFlow/OverFlow.sln
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/Variables/.DS_Store
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/Variables/Variables.sln
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/Byte/.DS_Store
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/Byte/Byte.sln
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/TypeConversion/TypeConversion.sln
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/TypeConversion/.DS_Store
/Users/saqibali/Documents/git/csharp/code/8. Working with Text/SummaryTextv2/SummaryTextv2/SummaryTextv2.csproj
/Users/saqibali/Documents/git/csharp/code/8. Working with Text/SummaryTextv2/SummaryTextv2/Program.cs
/Users/saqibali/Documents/git/csharp/code/8. Working with Text/SummaryTextv3/SummaryTextv3/StringUtility.cs
/Users/saqibali/Documents/git/csharp/code/8. Working with Text/SummaryTextv3/SummaryTextv3/SummaryTextv3.csproj
/Users/saqibali/Documents/git/csharp/code/8. Working with Text/SummaryTextv3/SummaryTextv3/Program.cs
/Users/saqibali/Documents/git/csharp/code/8. Working with Text/StringManipulate/StringManipulate/StringManipulate.csproj
/Users/saqibali/Documents/git/csharp/code/8. Working with Text/StringManipulate/StringManipulate/Program.cs
/Users/saqibali/Documents/git/csharp/code/8. Working with Text/SummaryTextv1/SummaryTextv1/SummaryTextv1.csproj
/Users/saqibali/Documents/git/csharp/code/8. Working with Text/SummaryTextv1/SummaryTextv1/Program.cs
/Users/saqibali/Documents/git/csharp/code/8. Working with Text/StringBuilder/StringBuilder/StringBuilder.csproj
/Users/saqibali/Documents/git/csharp/code/8. Working with Text/StringBuilder/StringBuilder/Program.cs
/Users/saqibali/Documents/git/csharp/code/8. Working with Text/WorkingText/WorkingText/WorkingText.csproj
/Users/saqibali/Documents/git/csharp/code/8. Working with Text/WorkingText/WorkingText/Program.cs
/Users/saqibali/Documents/git/csharp/code/8. Working with Text/StringManipulateUpdated/StringManipulateUpdated/StringManipulateUpdated.csproj
/Users/saqibali/Documents/git/csharp/code/8. Working with Text/StringManipulateUpdated/StringManipulateUpdated/Program.cs
/Users/saqibali/Documents/git/csharp/code/4. Non-Primitive Types/ByValueBasics/ByValueBasics/ByValueBasics.csproj
/Users/saqibali/Documents/git/csharp/code/4. Non-Primitive Types/ByValueBasics/ByValueBasics/Program.cs
/Users/saqibali/Documents/git/csharp/code/4. Non-Primitive Types/ByReferenceBasics/ByReferenceBasics/ByReferenceBasics.csproj
/Users/saqibali/Documents/git/csharp/code/4. Non-Primitive Types/ByReferenceBasics/ByReferenceBasics/Program.cs
/Users/saqibali/Documents/git/csharp/code/4. Non-Primitive Types/EnumsBasics/EnumsBasics/EnumsBasics.csproj
/Users/saqibali/Documents/git/csharp/code/4. Non-Primitive Types/EnumsBasics/EnumsBasics/Program.cs
/Users/saqibali/Documents/git/csharp/code/4. Non-Primitive Types/ArraysBasics/ArraysBasics/ArraysBasics.csproj
/Users/saqibali/Documents/git/csharp/code/4. Non-Primitive Types/ArraysBasics/ArraysBasics/Program.cs
/Users/saqibali/Documents/git/csharp/code/4. Non-Primitive Types/ByreferenceClasses/ByreferenceClasses/ByreferenceClasses.csproj
/Users/saqibali/Documents/git/csharp/code/4. Non-Primitive Types/ByreferenceClasses/ByreferenceClasses/Program.cs
/Users/saqibali/Documents/git/csharp/code/4. Non-Primitive Types/ClassesFundamentals/ClassesFundamentals/ClassesFundamentals.csproj
```



```csharp
using System;
using System.IO;

namespace DirectoryAndDirectoryInfo
{
    class Program
    {
        static void Main(string[] args)
        {
            //Directory.CreateDirectory(@"c:\temp\folder1");
            Directory.CreateDirectory(@"/Users/saqibali/Documents/git/csharp/code/test");

            var directoryinfo = new DirectoryInfo(@"/Users/saqibali/Documents/git/csharp/code/");
            var directoriesinfo = directoryinfo.GetDirectories("*.*", SearchOption.AllDirectories);

            foreach (var directory in directoriesinfo)
            {
                Console.WriteLine(directory);
            }
        }
    }
}
```



```
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/Operators/Operators/obj/Debug/netcoreapp3.1
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/Operators/Operators/bin/Debug/netcoreapp3.1
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/Operators/.vs/Operators/xs/project-cache
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/StingBool/StingBool/obj/Debug/netcoreapp3.1
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/StingBool/StingBool/bin/Debug/netcoreapp3.1
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/StingBool/.vs/StingBool/xs/project-cache
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/FloatingNumber/FloatingNumber/obj/Debug/netcoreapp3.1
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/FloatingNumber/FloatingNumber/bin/Debug/netcoreapp3.1
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/FloatingNumber/.vs/FloatingNumber/xs/project-cache
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/TryCatch/TryCatch/obj/Debug/netcoreapp3.1
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/TryCatch/TryCatch/bin/Debug/netcoreapp3.1
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/TryCatch/.vs/TryCatch/xs/project-cache
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/DecimalNumber/.vs/DecimalNumber/xs/project-cache
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/DecimalNumber/DecimalNumber/obj/Debug/netcoreapp3.1
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/DecimalNumber/DecimalNumber/bin/Debug/netcoreapp3.1
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/OverFlow/.vs/OverFlow/xs/project-cache
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/OverFlow/OverFlow/obj/Debug/netcoreapp3.1
/Users/saqibali/Documents/git/csharp/code/8. Working with Text/SummaryTextv2/SummaryTextv2.sln
/Users/saqibali/Documents/git/csharp/code/8. Working with Text/SummaryTextv3/SummaryTextv3.sln
/Users/saqibali/Documents/git/csharp/code/8. Working with Text/StringManipulate/StringManipulate.sln
/Users/saqibali/Documents/git/csharp/code/8. Working with Text/SummaryTextv1/SummaryTextv1.sln
/Users/saqibali/Documents/git/csharp/code/8. Working with Text/StringBuilder/StringBuilder.sln
/Users/saqibali/Documents/git/csharp/code/8. Working with Text/WorkingText/WorkingText.sln
/Users/saqibali/Documents/git/csharp/code/8. Working with Text/StringManipulateUpdated/StringManipulateUpdated.sln
/Users/saqibali/Documents/git/csharp/code/4. Non-Primitive Types/ByValueBasics/ByValueBasics.sln
/Users/saqibali/Documents/git/csharp/code/4. Non-Primitive Types/ByReferenceBasics/ByReferenceBasics.sln
/Users/saqibali/Documents/git/csharp/code/4. Non-Primitive Types/EnumsBasics/EnumsBasics.sln
/Users/saqibali/Documents/git/csharp/code/4. Non-Primitive Types/ArraysBasics/ArraysBasics.sln
/Users/saqibali/Documents/git/csharp/code/4. Non-Primitive Types/ByreferenceClasses/ByreferenceClasses.sln
/Users/saqibali/Documents/git/csharp/code/4. Non-Primitive Types/ClassesFundamentals/ClassesFundamentals.sln
/Users/saqibali/Documents/git/csharp/code/9. Working with Files/FileAndFileInfo/FileAndFileInfo.sln
/Users/saqibali/Documents/git/csharp/code/9. Working with Files/DirectoryAndDirectoryInfo/DirectoryAndDirectoryInfo.sln
/Users/saqibali/Documents/git/csharp/code/chapter01/HelloWorld/HelloWorld.sln
/Users/saqibali/Documents/git/csharp/code/chapter01/AddNumber/AddNumber.sln
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/Operators/Operators.sln
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/StingBool/StingBool.sln
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/FloatingNumber/FloatingNumber.sln
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/TryCatch/TryCatch.sln
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/DecimalNumber/DecimalNumber.sln
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/OverFlow/OverFlow.sln
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/Variables/Variables.sln
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/Byte/Byte.sln
/Users/saqibali/Documents/git/csharp/code/3. Primitive Types and Expressions/TypeConversion/TypeConversion.sln
/Users/saqibali/Documents/git/csharp/code/chapter01/HelloWorld/HelloWorld/HelloWorld.sln
/Users/saqibali/Documents/git/csharp/code/8. Working with Text
```



DirectoryInfo Class

Namespace: System.IO

Assembly: System.IO.FileSystem.dll

Exposes instance methods for creating, moving, and enumerating through directories and subdirectories. This class cannot be inherited.

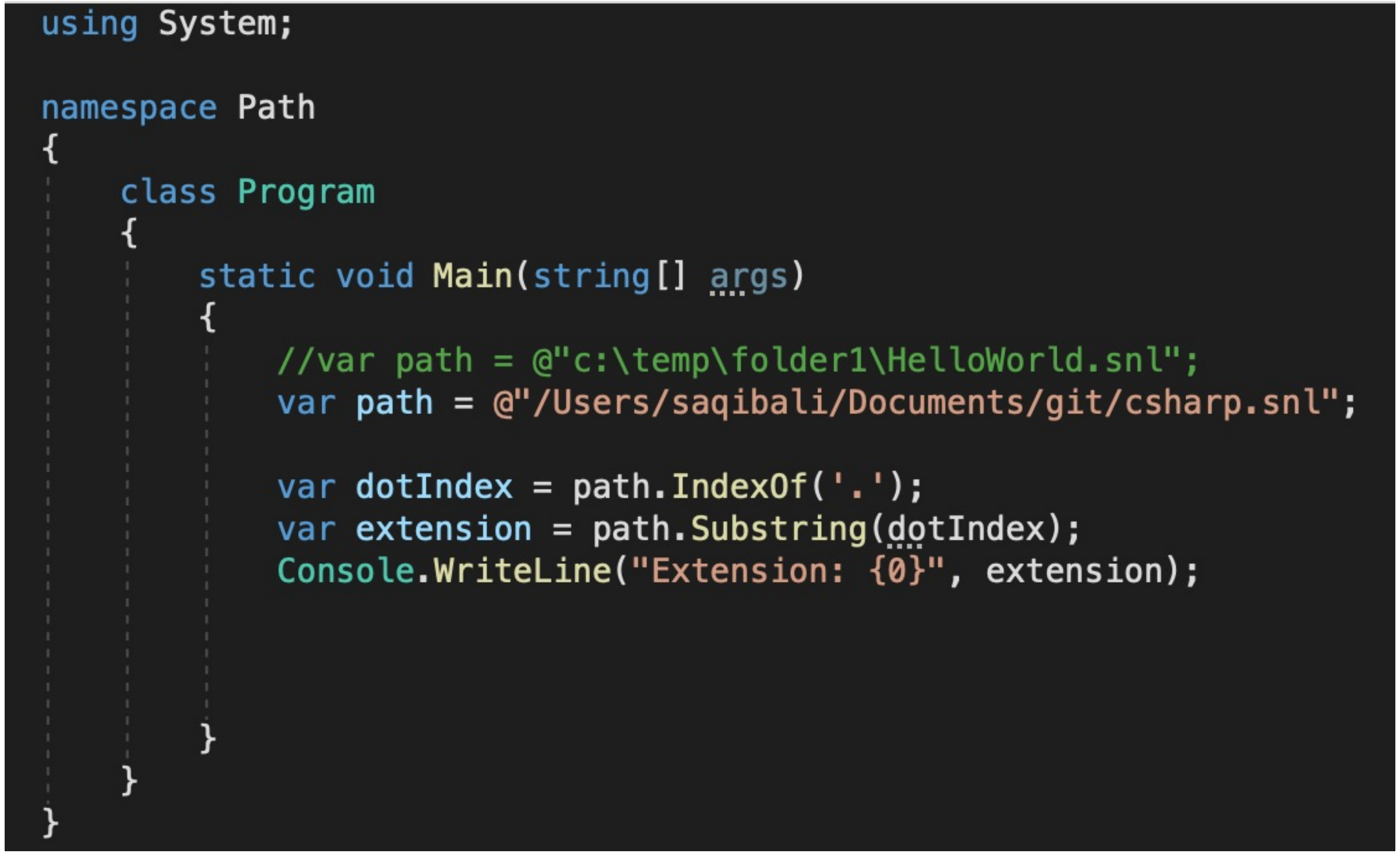
C# Copy

```
public sealed class DirectoryInfo : System.IO.FileSystemInfo
```

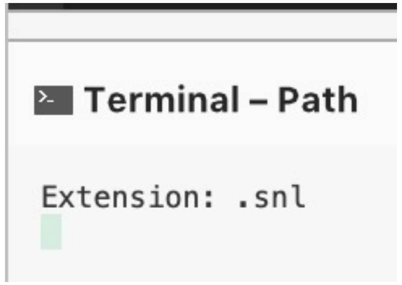
Inheritance Object → MarshalByRefObject → FileSystemInfo → DirectoryInfo

Examples

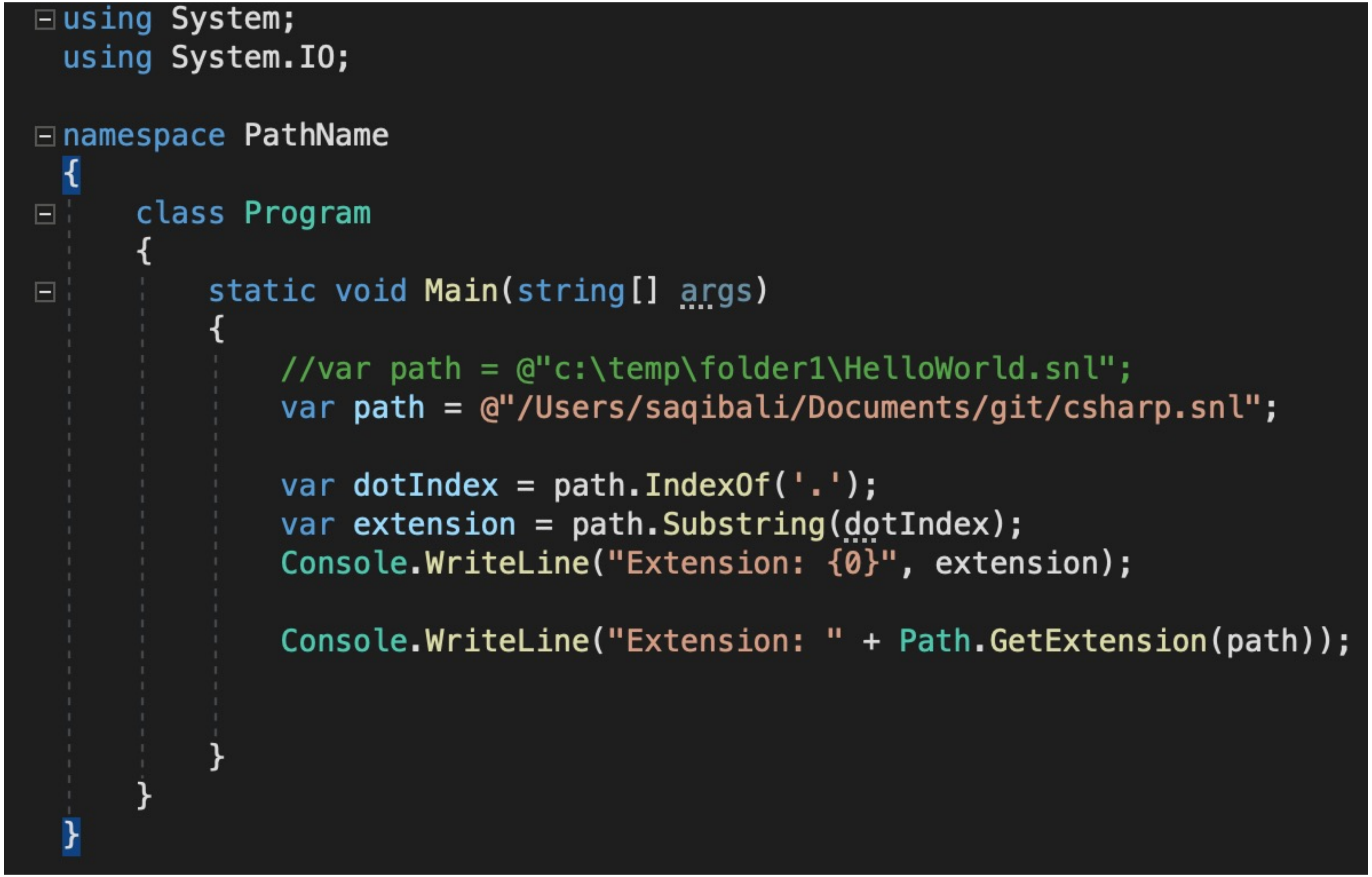
Path



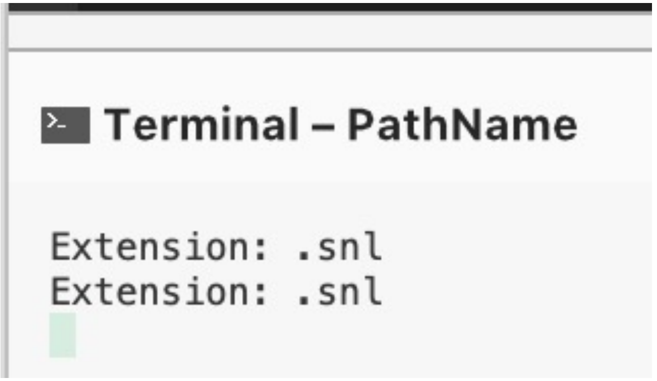
```
using System;

namespace Path
{
    class Program
    {
        static void Main(string[] args)
        {
            //var path = @"c:\temp\folder1\HelloWorld.snl";
            var path = @"/Users/saqibali/Documents/git/csharp.snl";

            var dotIndex = path.IndexOf('.');
            var extension = path.Substring(dotIndex);
            Console.WriteLine("Extension: {0}", extension);



        }
    }
}
```



```
Extension: .snl
```

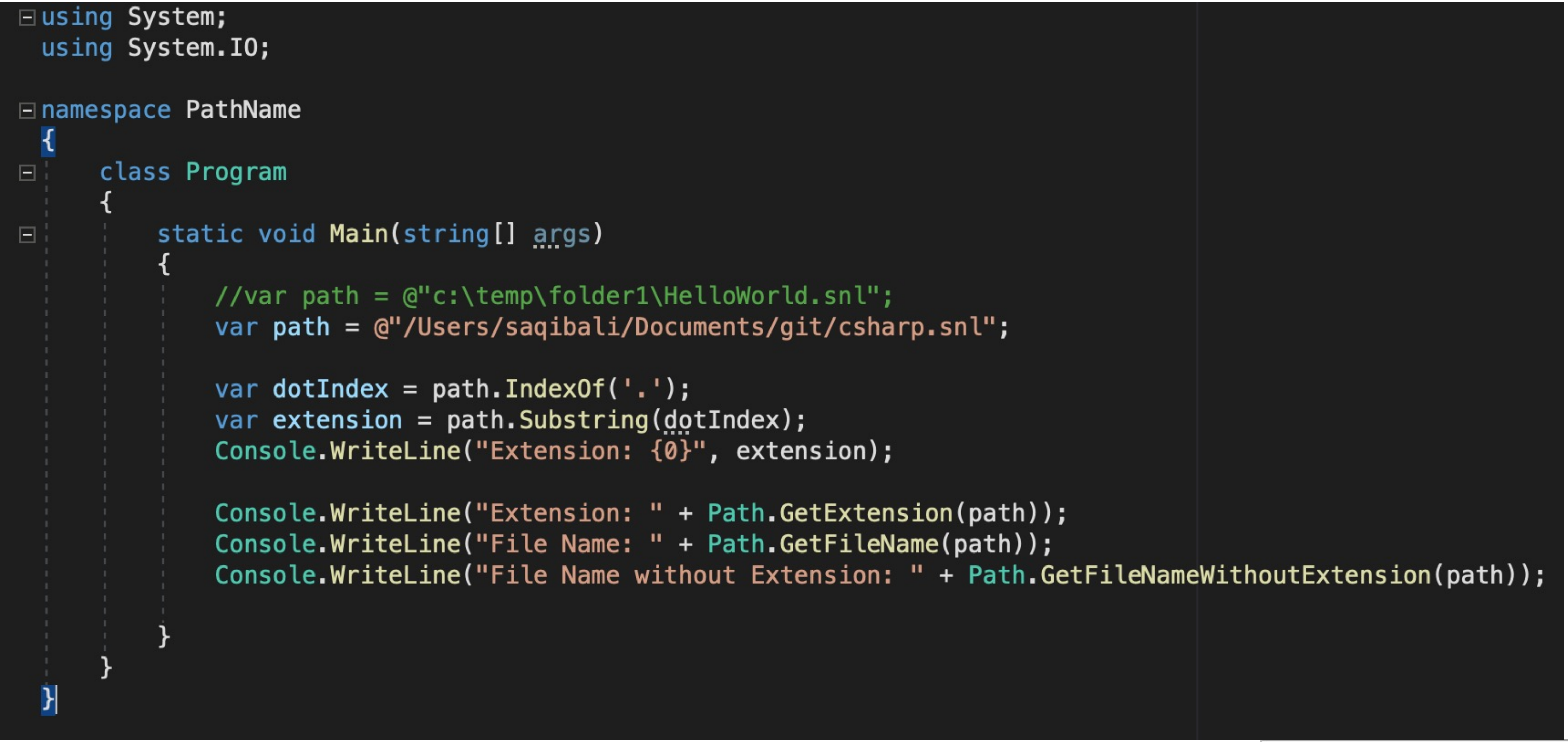


```
using System;
using System.IO;

namespace PathName
{
    class Program
    {
        static void Main(string[] args)
        {
            //var path = @"c:\temp\folder1\HelloWorld.snl";
            var path = @"/Users/saqibali/Documents/git/csharp.snl";

            var dotIndex = path.IndexOf('.');
            var extension = path.Substring(dotIndex);
            Console.WriteLine("Extension: {0}", extension);

            Console.WriteLine("Extension: " + Path.GetExtension(path));


        }
    }
}
```

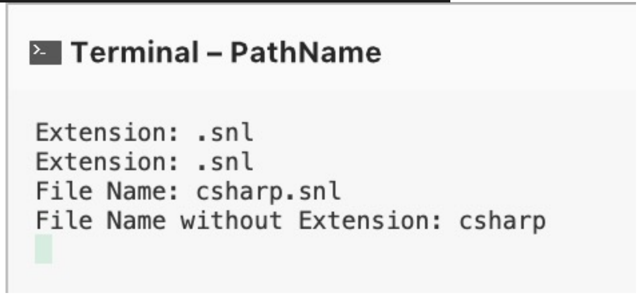


Terminal – PathName

```
Extension: .snl
Extension: .snl
```

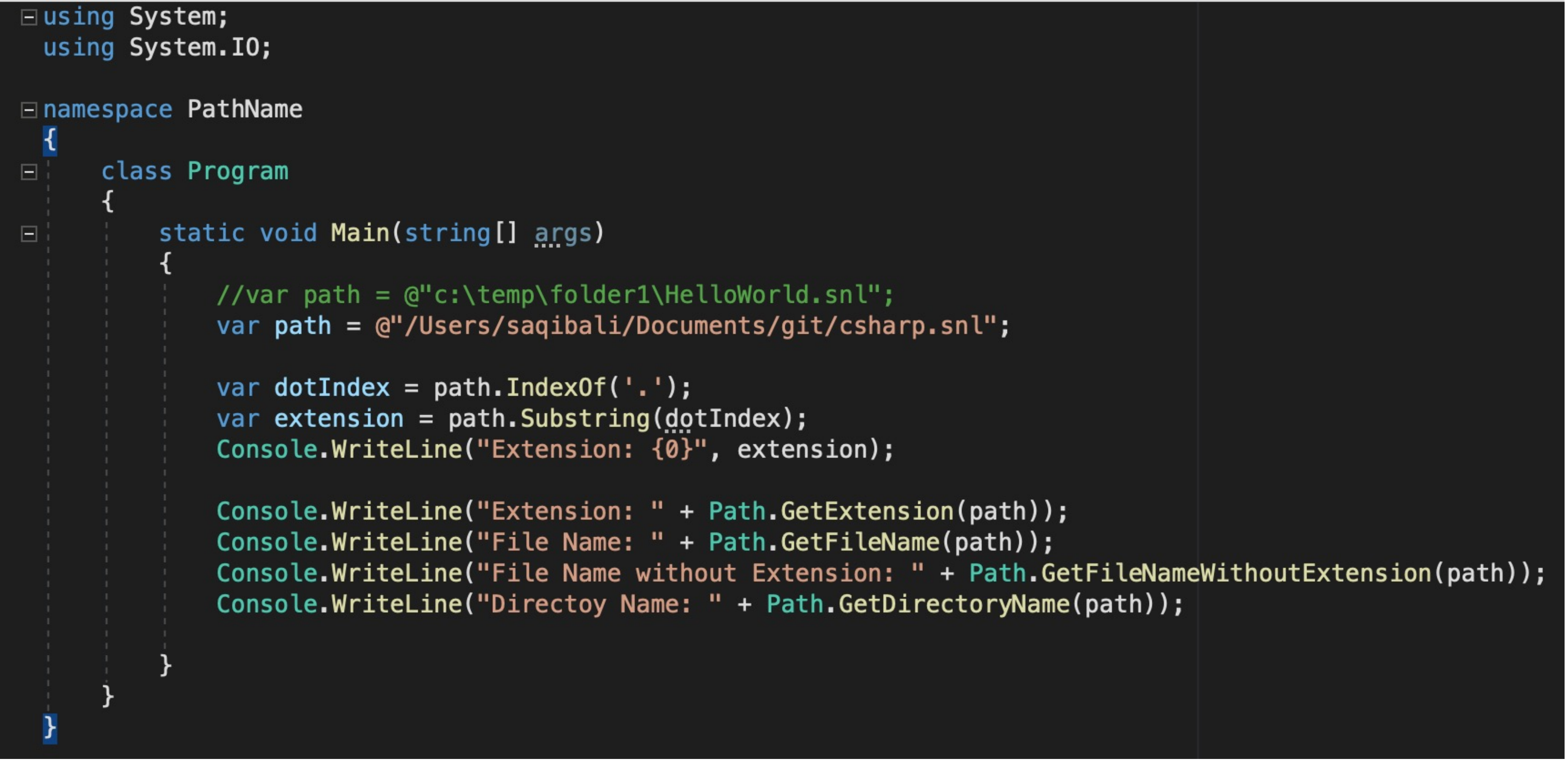


```
using System;
using System.IO;

namespace PathName
{
    class Program
    {
        static void Main(string[] args)
        {
            //var path = @"c:\temp\folder1\HelloWorld.snl";
            var path = @"/Users/saqibali/Documents/git/csharp.snl";

            var dotIndex = path.IndexOf('.');
            var extension = path.Substring(dotIndex);
            Console.WriteLine("Extension: {0}", extension);

            Console.WriteLine("Extension: " + Path.GetExtension(path));
            Console.WriteLine("File Name: " + Path.GetFileName(path));
            Console.WriteLine("File Name without Extension: " + Path.GetFileNameWithoutExtension(path));

        }
    }
}
```

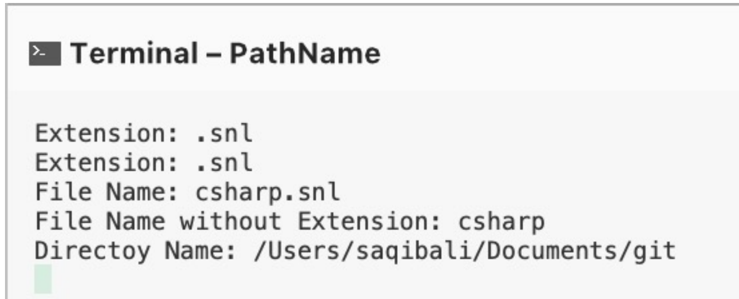


Terminal – PathName

```
Extension: .snl
Extension: .snl
File Name: csharp.snl
File Name without Extension: csharp
```

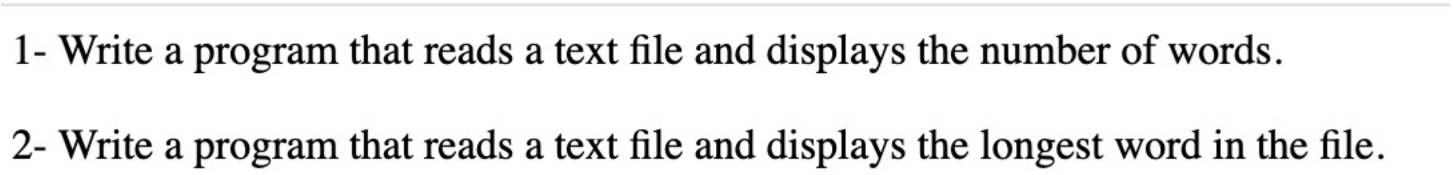


```
using System;
using System.IO;

namespace PathName
{
    class Program
    {
        static void Main(string[] args)
        {
            //var path = @"c:\temp\folder1\HelloWorld.snl";
            var path = @"/Users/saqibali/Documents/git/csharp.snl";

            var dotIndex = path.IndexOf('.');
            var extension = path.Substring(dotIndex);
            Console.WriteLine("Extension: {0}", extension);

            Console.WriteLine("Extension: " + Path.GetExtension(path));
            Console.WriteLine("File Name: " + Path.GetFileName(path));
            Console.WriteLine("File Name without Extension: " + Path.GetFileNameWithoutExtension(path));
            Console.WriteLine("Directoy Name: " + Path.GetDirectoryName(path));

        }
    }
}
```



```
Extension: .snl
Extension: .snl
File Name: csharp.snl
File Name without Extension: csharp
Directoy Name: /Users/saqibali/Documents/git
```

Exercises



1- Write a program that reads a text file and displays the number of words.

2- Write a program that reads a text file and displays the longest word in the file.