

# IsClass, Namespace, FullName, IsAbstract, IsPublic, IsInterface, IsEnum

```
using System;
using System.Reflection;

class MainClass {
public static void ShowTypes(string name, Assembly assembly) {
Type[] typeArray = assembly.GetTypes();
Console.WriteLine(name);
foreach (Type type in typeArray) {
if (type.IsClass) {
Console.WriteLine("
Namespace : {0}", type.Namespace);
Console.WriteLine("Class : {0}", type.FullName);
if (type.BaseType != null)
Console.WriteLine("Base Class : {0}",
type.BaseType.FullName);
else
Console.WriteLine("Class not derived from anything");

if (type.IsAbstract)
Console.WriteLine("Abstract base class");
else
Console.WriteLine("Instantiable class");
if (type.IsPublic)
Console.WriteLine("Scope: Public");
else
Console.WriteLine("Scope: Private");
} else if (type.IsInterface) {
Console.WriteLine("
Namespace : {0}", type.Namespace);
Console.WriteLine("Interface : {0}", type.FullName);
```

```
if (type.IsPublic)
Console.WriteLine("Scope: Public");
else
Console.WriteLine("Scope: Private");
} else
if (type.IsEnum) {
Console.WriteLine("
Enumeration: {0}", type.FullName);
} else
Console.WriteLine("
Type: {0}", type.FullName);
}
}
public static void Main(string[] args) {
for (int i = 0; i < args.Length; ++i) { Assembly assembly =
Assembly.LoadFrom(args[0]); ShowTypes(args[0], assembly); } }
[/csharp][[/csharp][[/csharp][[/csharp][[/csharp][[/csharp][[/csharp]
]
```