RSCG nr 17 : ApparatusAOT

Info

Nuget : https://www.nuget.org/packages/Apparatus.AOT.Reflection/

You can find more details at : https://github.com/byme8/Apparatus.AOT.Reflection

Author :Stanislav Silin

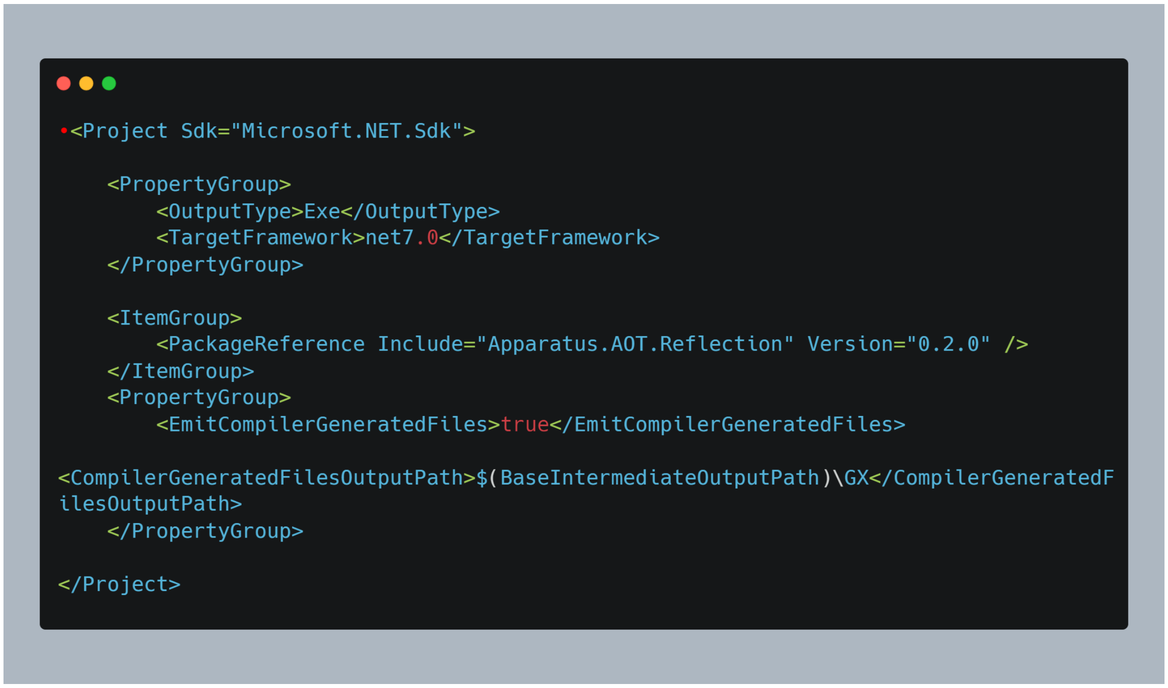
Source : https://github.com/byme8/Apparatus.AOT.Reflection

About

This will generate code for investigating at runtime the properties of an object

How to use

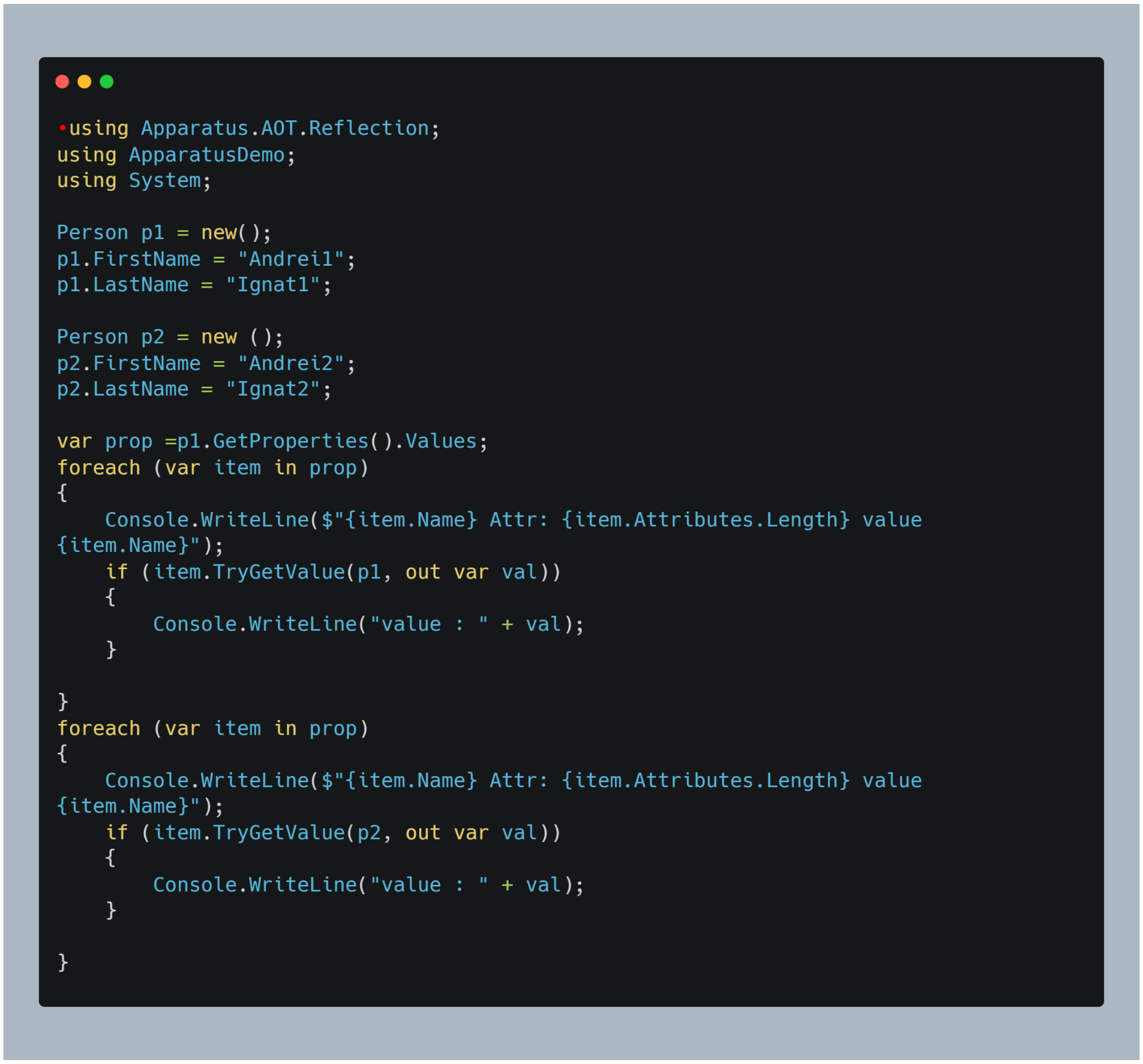
Add reference to the ApparatusAOT in the csproj



```xml
<Project Sdk="Microsoft.NET.Sdk">

    <PropertyGroup>
        <OutputType>Exe</OutputType>
        <TargetFramework>net7.0</TargetFramework>
    </PropertyGroup>

    <ItemGroup>
        <PackageReference Include="Apparatus.AOT.Reflection" Version="0.2.0" />
    </ItemGroup>
    <PropertyGroup>
        <EmitCompilerGeneratedFiles>true</EmitCompilerGeneratedFiles>

<CompilerGeneratedFilesOutputPath>$(BaseIntermediateOutputPath)\GX</CompilerGeneratedFilesOutputPath>
    </PropertyGroup>

</Project>
```

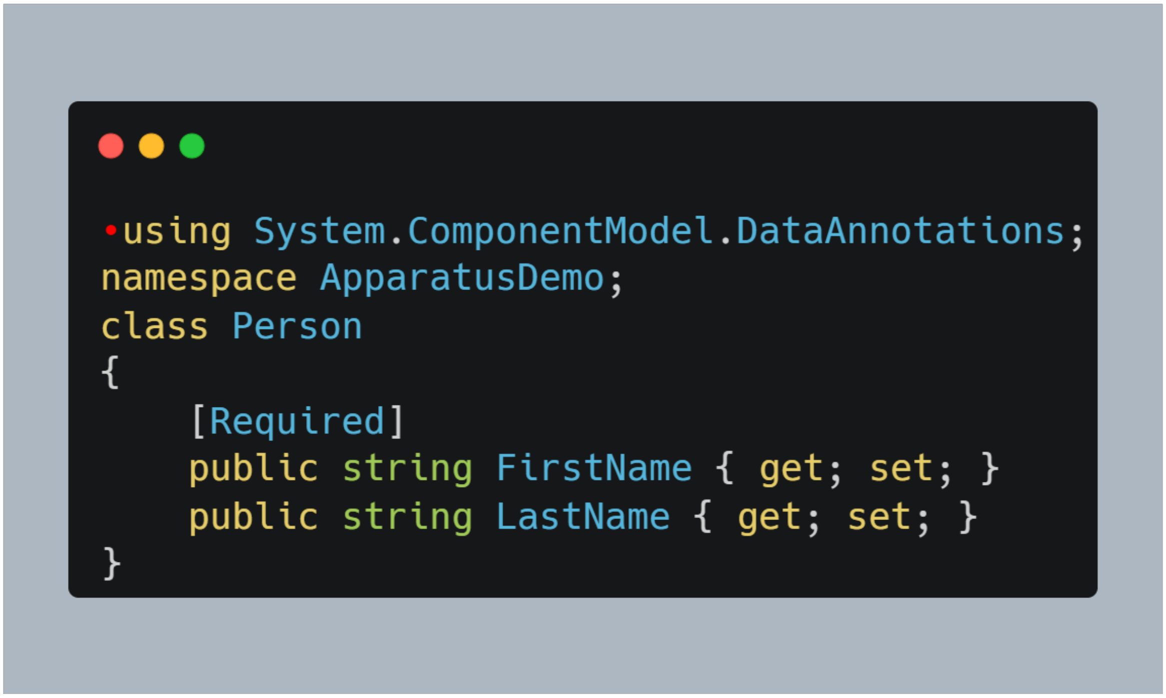
This was for me the starting code

I have **coded** the file Program.cs



```
using Apparatus.AOT.Reflection;
using ApparatusDemo;
using System;

Person p1 = new();
p1.FirstName = "Andrei1";
p1.LastName = "Ignat1";

Person p2 = new ();
p2.FirstName = "Andrei2";
p2.LastName = "Ignat2";

var prop =p1.GetProperties().Values;
foreach (var item in prop)
{
    Console.WriteLine($"{item.Name} Attr: {item.Attributes.Length} value
{item.Name}");
    if (item.TryGetValue(p1, out var val))
    {
        Console.WriteLine("value : " + val);
    }

}
foreach (var item in prop)
{
    Console.WriteLine($"{item.Name} Attr: {item.Attributes.Length} value
{item.Name}");
    if (item.TryGetValue(p2, out var val))
    {
        Console.WriteLine("value : " + val);
    }

}
```

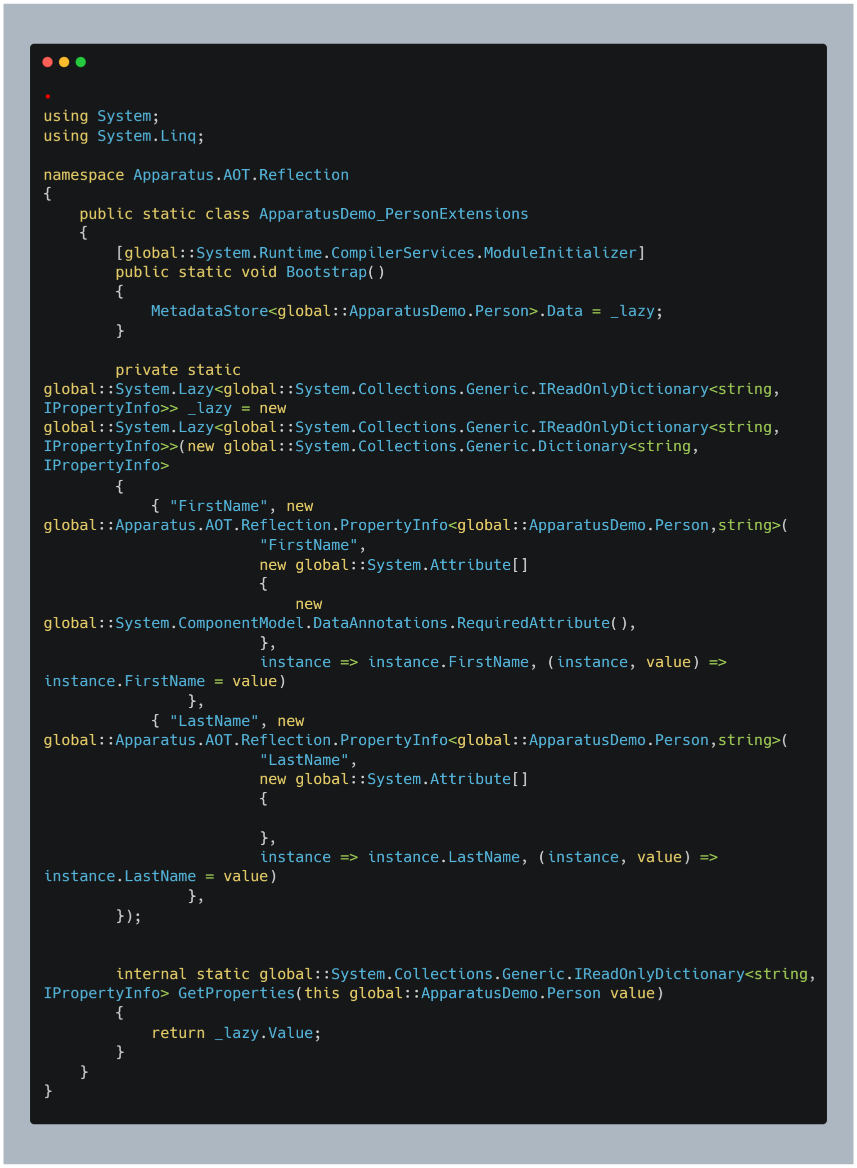
I have **coded** the file Person.cs



```
using System.ComponentModel.DataAnnotations;
namespace ApparatusDemo;
class Person
{
    [Required]
    public string FirstName { get; set; }
    public string LastName { get; set; }
}
```

And here are the *generated* files

The file *generated* is ApparatusDemo_PersonExtensions.cs



```
using System;
using System.Linq;

namespace Apparatus.AOT.Reflection
{
    public static class ApparatusDemo_PersonExtensions
    {
        [global::System.Runtime.CompilerServices.ModuleInitializer]
        public static void Bootstrap()
        {
            MetadataStore<global::ApparatusDemo.Person>.Data = _lazy;
        }

        private static
global::System.Lazy<global::System.Collections.Generic.IReadOnlyDictionary<string,
IPropertyInfo>> _lazy = new
global::System.Lazy<global::System.Collections.Generic.IReadOnlyDictionary<string,
IPropertyInfo>>(new global::System.Collections.Generic.Dictionary<string,
IPropertyInfo>
        {
            { "FirstName", new
global::Apparatus.AOT.Reflection.PropertyInfo<global::ApparatusDemo.Person,string>(
                        "FirstName",
                        new global::System.Attribute[]
                        {
                            new
global::System.ComponentModel.DataAnnotations.RequiredAttribute(),
                        },
                        instance => instance.FirstName, (instance, value) =>
instance.FirstName = value)
                },
            { "LastName", new
global::Apparatus.AOT.Reflection.PropertyInfo<global::ApparatusDemo.Person,string>(
                        "LastName",
                        new global::System.Attribute[]
                        {

                        },
                        instance => instance.LastName, (instance, value) =>
instance.LastName = value)
                },
        });


        internal static global::System.Collections.Generic.IReadOnlyDictionary<string,
IPropertyInfo> GetProperties(this global::ApparatusDemo.Person value)
        {
            return _lazy.Value;
        }
    }
}
```

You can download the code and this page as pdf from https://ignatandrei.github.io/RSCG_Examples/v2/docs/ApparatusAOT

You can see the whole list at https://ignatandrei.github.io/RSCG_Examples/v2/docs/List-of-RSCG