RSCG nr 1 : ThisAssembly

Info

Nuget : https://www.nuget.org/packages/ThisAssembly

You can find more details at : https://www.clarius.org/ThisAssembly/

Author :Daniel Cazzulino

Source : https://github.com/devlooped/ThisAssembly

About

The ThisAssembly.Info allows you access to the Assembly Information as constants, instead of going to reflection each time.I found useful to see the assembly version right away in any project that I have.

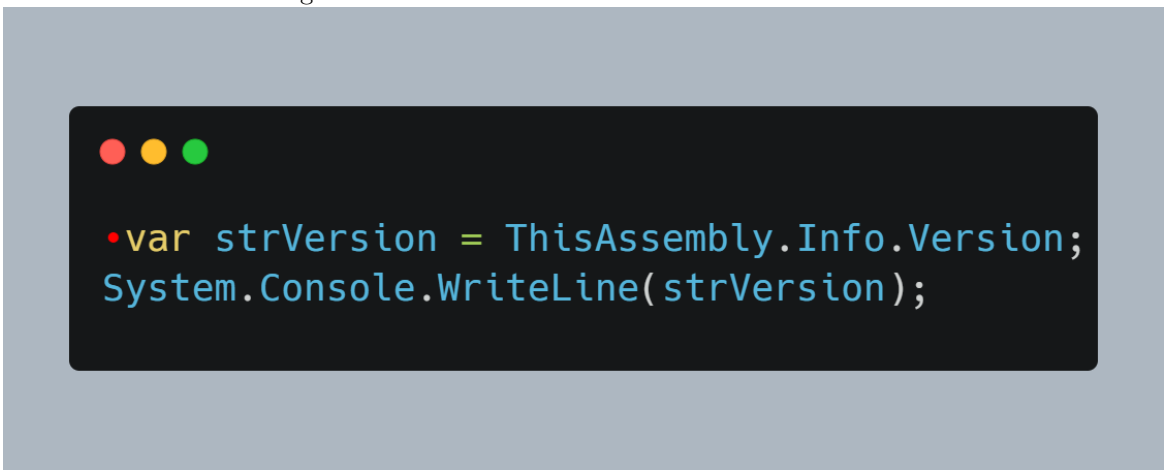
How to use

Add reference to the ThisAssembly in the csproj

```
<Project Sdk="Microsoft.NET.Sdk">
    <PropertyGroup>
        <OutputType>Exe</OutputType>
        <TargetFramework>netcoreapp7.0</TargetFramework>
    </PropertyGroup>
    <PropertyGroup>
        <EmitCompilerGeneratedFiles>true</EmitCompilerGeneratedFiles>

<CompilerGeneratedFilesOutputPath>$(BaseIntermediateOutputPath)\GX</CompilerGeneratedF
ilesOutputPath>
    </PropertyGroup>
    <PropertyGroup>
        <Version>2023.5.7.800</Version>
    </PropertyGroup>
    <ItemGroup>
        <PackageReference Include="ThisAssembly" Version="1.2.14"
OutputItemType="Analyzer" ReferenceOutputAssembly="false">
            <PrivateAssets>all</PrivateAssets>
            <IncludeAssets>runtime; build; native; contentfiles; analyzers;
buildtransitive</IncludeAssets>
        </PackageReference>
    </ItemGroup>
</Project>
```

This was for me the starting code

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



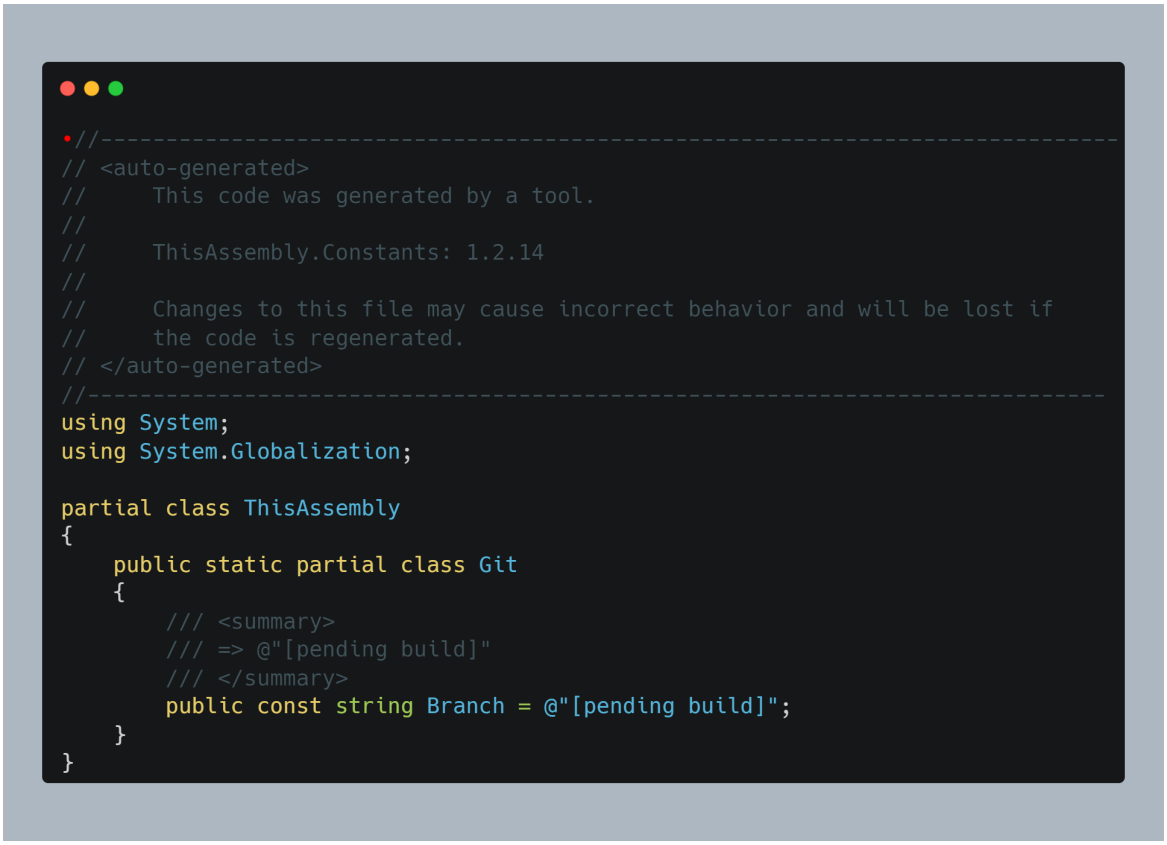
```
var strVersion = ThisAssembly.Info.Version;
System.Console.WriteLine(strVersion);
```

And here are the *generated* files

The file *generated* is ThisAssembly.AssemblyInfo.g.cs

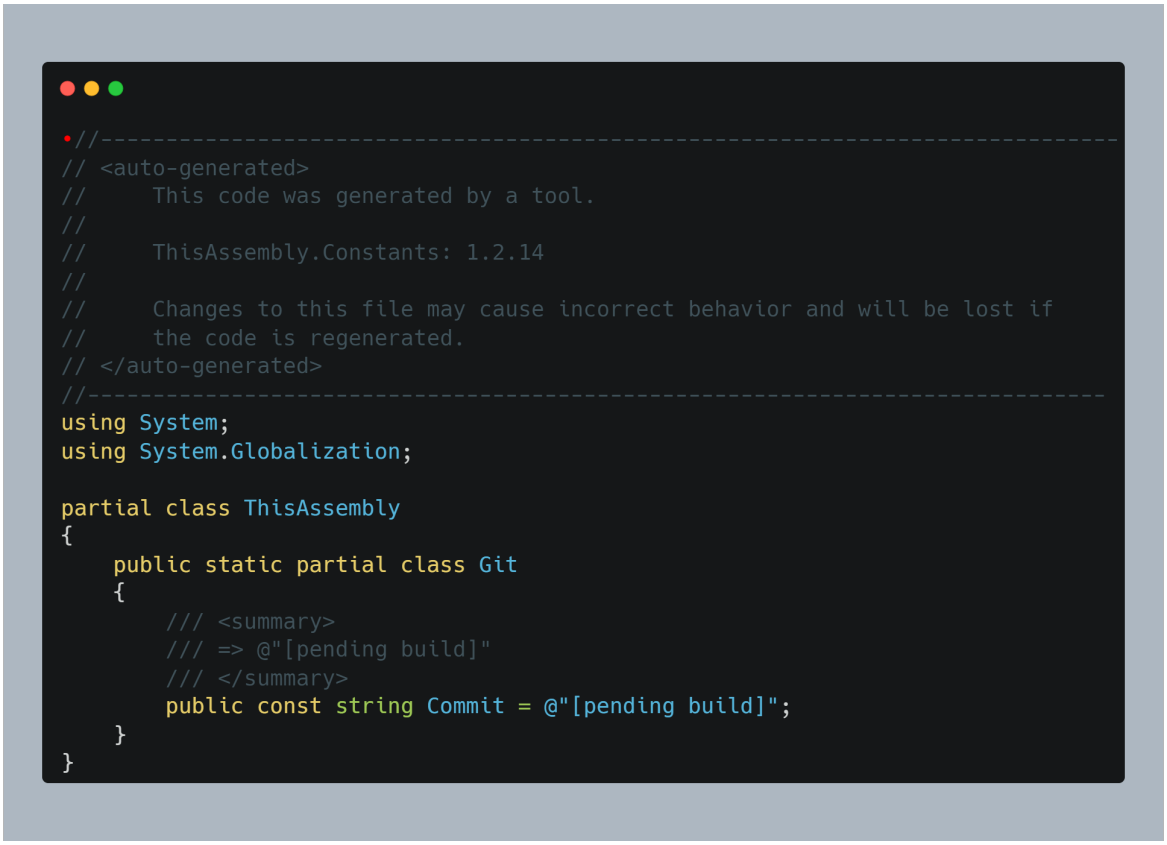
```
//------------------------------------------------------------------------------
// <auto-generated>
//     This code was generated by a tool.
//
//     Changes to this file may cause incorrect behavior and will be lost if
//     the code is regenerated.
// </auto-generated>
//------------------------------------------------------------------------------

using System.CodeDom.Compiler;
using System.Runtime.CompilerServices;

/// <summary>
/// Provides access to the current assembly information as pure constants,
///  without requiring reflection.
/// </summary>
partial class ThisAssembly
{
    /// <summary>
    /// Gets the AssemblyInfo attributes.
    /// </summary>
    [GeneratedCode("ThisAssembly.AssemblyInfo", "1.2.14")]
    [CompilerGenerated]
    public static partial class Info
    {
        public const string Company = @"RSCG_Version";

        public const string Configuration = @"Debug";

        public const string FileVersion = @"2023.5.7.800";

        public const string InformationalVersion = @"2023.5.7.800";

        public const string Product = @"RSCG_Version";

        public const string Title = @"RSCG_Version";

        public const string Version = @"2023.5.7.800";

    }
}
```

The file *generated* is Branch.g.cs



```
//------------------------------------------------------------------------------
// <auto-generated>
//     This code was generated by a tool.
//
//     ThisAssembly.Constants: 1.2.14
//
//     Changes to this file may cause incorrect behavior and will be lost if
//     the code is regenerated.
// </auto-generated>
//------------------------------------------------------------------------------
using System;
using System.Globalization;

partial class ThisAssembly
{
    public static partial class Git
    {
        /// <summary>
        /// => @"[pending build]"
        /// </summary>
        public const string Branch = @"[pending build]";
    }
}
```

The file *generated* is Commit.g.cs

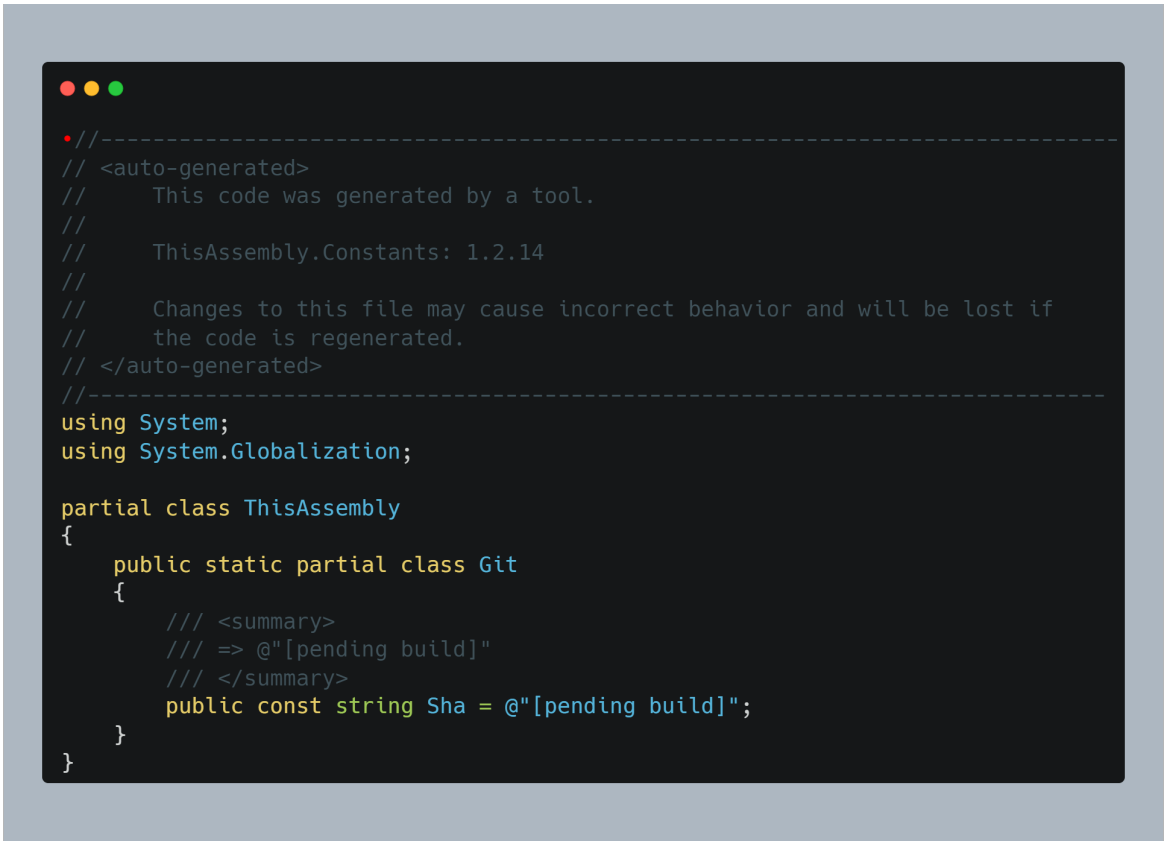


```
//------------------------------------------------------------------------------
// <auto-generated>
//     This code was generated by a tool.
//
//     ThisAssembly.Constants: 1.2.14
//
//     Changes to this file may cause incorrect behavior and will be lost if
//     the code is regenerated.
// </auto-generated>
//------------------------------------------------------------------------------
using System;
using System.Globalization;

partial class ThisAssembly
{
    public static partial class Git
    {
        /// <summary>
        /// => @"[pending build]"
        /// </summary>
        public const string Commit = @"[pending build]";
    }
}
```

The file *generated* is Root.g.cs

```
//------------------------------------------------------------------------------
// <auto-generated>
//     This code was generated by a tool.
//
//     ThisAssembly.Constants: 1.2.14
//
//     Changes to this file may cause incorrect behavior and will be lost if
//     the code is regenerated.
// </auto-generated>
//------------------------------------------------------------------------------
using System;
using System.Globalization;

partial class ThisAssembly
{
    public static partial class Git
    {
        /// <summary>
        /// => @"[pending build]"
        /// </summary>
        public const string Root = @"[pending build]";
    }
}
```

The file *generated* is Sha.g.cs



```
//------------------------------------------------------------------------------
// <auto-generated>
//     This code was generated by a tool.
//
//     ThisAssembly.Constants: 1.2.14
//
//     Changes to this file may cause incorrect behavior and will be lost if
//     the code is regenerated.
// </auto-generated>
//------------------------------------------------------------------------------
using System;
using System.Globalization;

partial class ThisAssembly
{
    public static partial class Git
    {
        /// <summary>
        /// => @"[pending build]"
        /// </summary>
        public const string Sha = @"[pending build]";
    }
}
```

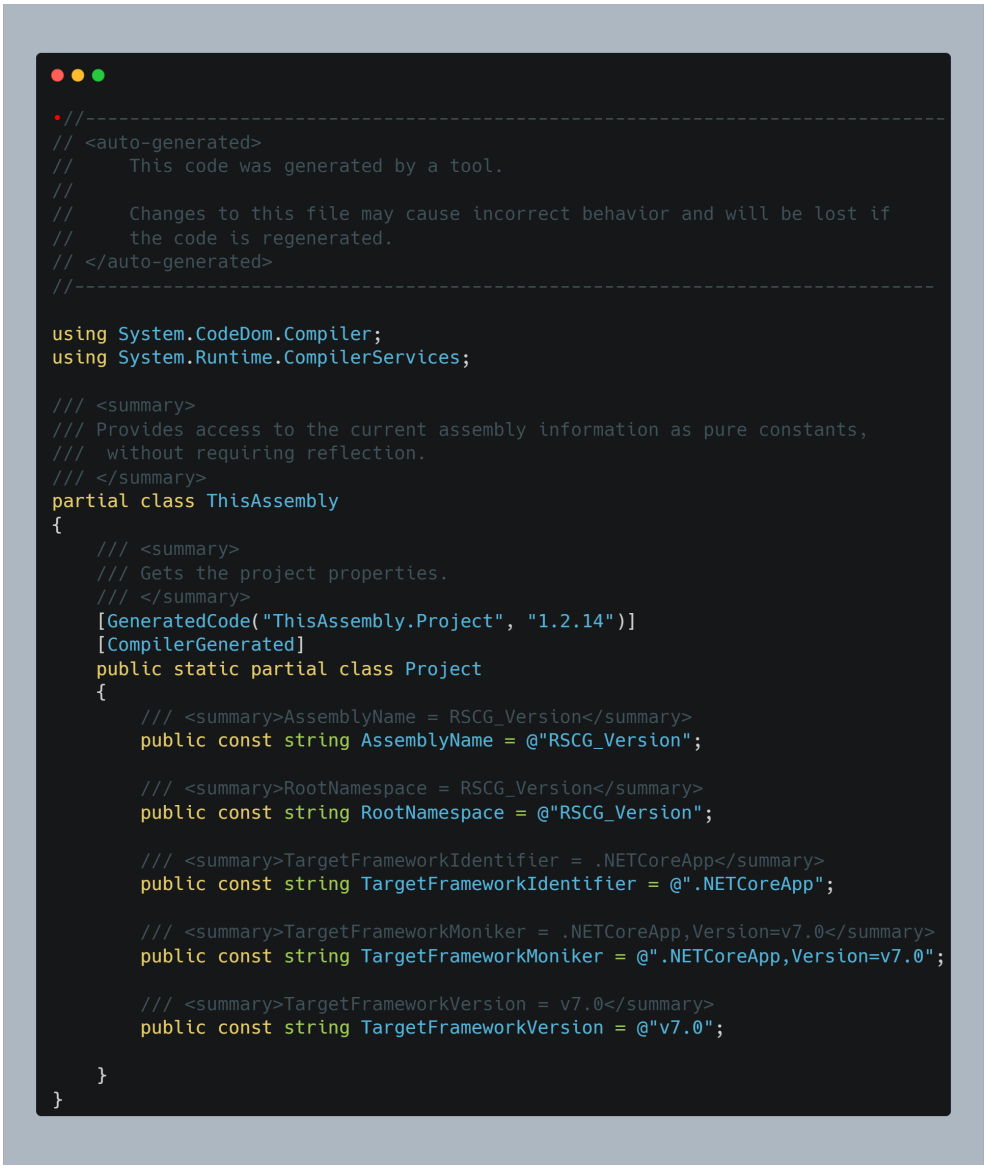
The file *generated* is Url.g.cs

```
//------------------------------------------------------------------------------
// <auto-generated>
//     This code was generated by a tool.
//
//     ThisAssembly.Constants: 1.2.14
//
//     Changes to this file may cause incorrect behavior and will be lost if
//     the code is regenerated.
// </auto-generated>
//------------------------------------------------------------------------------
using System;
using System.Globalization;

partial class ThisAssembly
{
    public static partial class Git
    {
        /// <summary>
        /// => @"[pending build]"
        /// </summary>
        public const string Url = @"[pending build]";
    }
}
```

The file *generated* is ThisAssembly.Metadata.g.cs

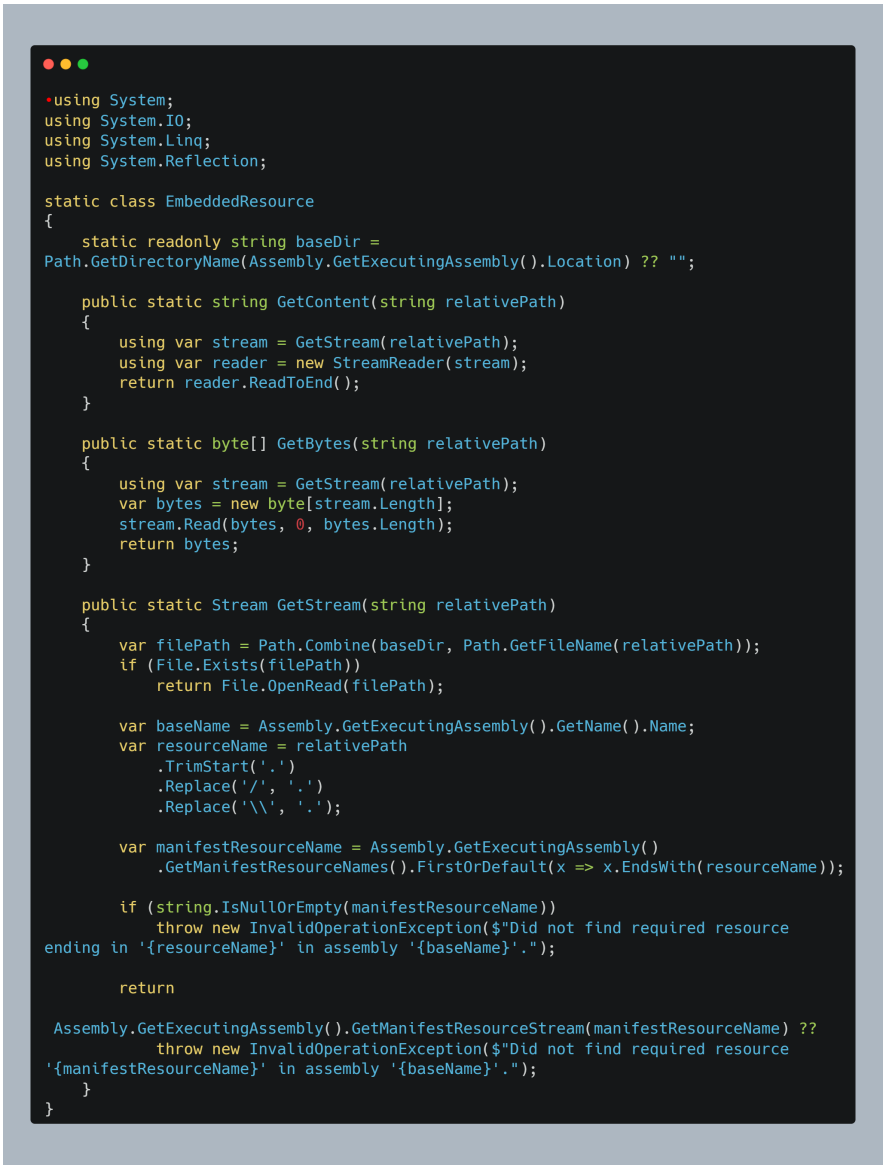
```cs
//------------------------------------------------------------------------------
// <auto-generated>
//     This code was generated by a tool.
//
//     Changes to this file may cause incorrect behavior and will be lost if
//     the code is regenerated.
// </auto-generated>
//------------------------------------------------------------------------------

using System.CodeDom.Compiler;
using System.Runtime.CompilerServices;

/// <summary>
/// Provides access to the current assembly information as pure constants,
///  without requiring reflection.
/// </summary>
partial class ThisAssembly
{
    /// <summary>
    /// Gets the assembly metadata.
    /// </summary>
    [GeneratedCode("ThisAssembly.Metadata", "1.2.14")]
    [CompilerGenerated]
    public static partial class Metadata
    {
    }
}
```

The file *generated* is ThisAssembly.Property.g.cs



```
//------------------------------------------------------------------------------
// <auto-generated>
//     This code was generated by a tool.
//
//     Changes to this file may cause incorrect behavior and will be lost if
//     the code is regenerated.
// </auto-generated>
//------------------------------------------------------------------------------

using System.CodeDom.Compiler;
using System.Runtime.CompilerServices;

/// <summary>
/// Provides access to the current assembly information as pure constants,
///  without requiring reflection.
/// </summary>
partial class ThisAssembly
{
    /// <summary>
    /// Gets the project properties.
    /// </summary>
    [GeneratedCode("ThisAssembly.Project", "1.2.14")]
    [CompilerGenerated]
    public static partial class Project
    {
        /// <summary>AssemblyName = RSCG_Version</summary>
        public const string AssemblyName = @"RSCG_Version";

        /// <summary>RootNamespace = RSCG_Version</summary>
        public const string RootNamespace = @"RSCG_Version";

        /// <summary>TargetFrameworkIdentifier = .NETCoreApp</summary>
        public const string TargetFrameworkIdentifier = @".NETCoreApp";

        /// <summary>TargetFrameworkMoniker = .NETCoreApp,Version=v7.0</summary>
        public const string TargetFrameworkMoniker = @".NETCoreApp,Version=v7.0";

        /// <summary>TargetFrameworkVersion = v7.0</summary>
        public const string TargetFrameworkVersion = @"v7.0";

    }
}
```

The file *generated* is ThisAssembly.Resources.EmbeddedResource.cs



```
using System;
using System.IO;
using System.Linq;
using System.Reflection;

static class EmbeddedResource
{
    static readonly string baseDir =
Path.GetDirectoryName(Assembly.GetExecutingAssembly().Location) ?? "";

    public static string GetContent(string relativePath)
    {
        using var stream = GetStream(relativePath);
        using var reader = new StreamReader(stream);
        return reader.ReadToEnd();
    }

    public static byte[] GetBytes(string relativePath)
    {
        using var stream = GetStream(relativePath);
        var bytes = new byte[stream.Length];
        stream.Read(bytes, 0, bytes.Length);
        return bytes;
    }

    public static Stream GetStream(string relativePath)
    {
        var filePath = Path.Combine(baseDir, Path.GetFileName(relativePath));
        if (File.Exists(filePath))
            return File.OpenRead(filePath);

        var baseName = Assembly.GetExecutingAssembly().GetName().Name;
        var resourceName = relativePath
            .TrimStart('.')
            .Replace('/', '.')
            .Replace('\\', '.');

        var manifestResourceName = Assembly.GetExecutingAssembly()
            .GetManifestResourceNames().FirstOrDefault(x => x.EndsWith(resourceName));

        if (string.IsNullOrEmpty(manifestResourceName))
            throw new InvalidOperationException($"Did not find required resource
ending in '{resourceName}' in assembly '{baseName}'.");

        return

 Assembly.GetExecutingAssembly().GetManifestResourceStream(manifestResourceName) ??
            throw new InvalidOperationException($"Did not find required resource
'{manifestResourceName}' in assembly '{baseName}'.");
    }
}
```

The file *generated* is ThisAssembly.Strings.g.cs

```
using System.Collections.Concurrent;
using System.Resources;
using System.Runtime.CompilerServices;

/// <summary>
/// Provides access to the current assembly information as pure constants,
///  without requiring reflection.
/// </summary>
partial class ThisAssembly
{
    /// <summary>
    /// Access the strings provided by resource files in the project.
    /// </summary>
    [CompilerGenerated]
    public static partial class Strings
    {
        static ConcurrentDictionary<string, ResourceManager> resourceManagers = new
ConcurrentDictionary<string, ResourceManager>();

        static ResourceManager GetResourceManager(string resourceName)
            => resourceManagers.GetOrAdd(resourceName, name => new
ResourceManager(name, typeof(Strings).Assembly));
    }
}
```

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

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