RSCG Examples

Andrei Ignat

Table of Contents

## About this book

### Content of the book

You will find in this book code examples about >10 Roslyn Source Code Generator (RSCG ) that can be usefull for you. That means, you will write more elegant and concise code - even if the generators code is not always nice to look.

### Are those examples ready for production ?

I have done due diligence to test the RSCG that I have show to you here. However, I cannot guarantee that will fit your code . That means that you can test it for your case and, because all are open source on Github.com , you can contribute to improve them ;-)

### How to read this book

For each chapter, you will find

1. Name of the RSCG and link to the NuGet package / GitHub repository
2. What the RSCG can do
3. What will be the initial code
4. How to use the Code generated by RSCG
5. Code Generated by RSCG
6. Link to the downloadable code to practice

### I have a suggestion for a new RSCG that is worth mentioning in this book . What can I do ?

Please send me an email to [ignatandrei@yahoo.com](mailto:ignatandrei@yahoo.com)

### I want to make a RSCG that will be useful. How can I do ?

In the introduction I have put the links to get you started with RSCG .

And, if you bought this book from Amazon , you are entitled to have 1 hour free of consultancy with me . I can help you make one.

## Introduction

### What is a Roslyn Source Code Generator ?

A Roslyn Source Code Generator (RSCG ) is a program that generates code in the compile time, based on the previous source code and/or another data . This new source code is added to the compilation and compile with the previous source code.

### How can I make a Roslyn Source Code Generator ?

For creating the RSCG you will simply create a .NET Standard 2.0 project,add those 2 references

<PackageReference Include="Microsoft.CodeAnalysis.Analyzers" Version="3.3.1" PrivateAssets="all" />  
 <PackageReference Include="Microsoft.CodeAnalysis.CSharp" Version="3.8.0" />

and start implementing

public interface ISourceGenerator  
{  
 void Initialize(GeneratorInitializationContext context);  
 void Execute(GeneratorExecutionContext context);  
}

Start from examples at <https://github.com/dotnet/roslyn-sdk/tree/main/samples/CSharp/SourceGenerators> Also, you can read the source code for the RSCG presented in this book.

### Show me some code for RSCG

Start read

<https://github.com/dotnet/roslyn/blob/main/docs/features/source-generators.md>

and

<https://github.com/dotnet/roslyn/blob/main/docs/features/source-generators.cookbook.md> .

After that , you can play with the examples from <https://github.com/dotnet/roslyn-sdk/tree/main/samples/CSharp/SourceGenerators> or from <https://sourcegen.dev/> ( see AutoNotify in the dropdown )

### How the RSCG can help me to write faster / better the code ?

Glad that you asked. You can see in action a RSCG for automatically generating code for automating testing ( see DynamicMocking ) , parsing enum ( see Enum ) , generating controllers actions from a interface ( SkinnyControllers ), currying functions and many more. In this book you will find more than 10 examples of some RSCG that can help you. Also, you can find the source code of the examples at <https://github.com/ignatandrei/RSCG_Examples>.

## RSCG number 1 : ThisAssembly

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

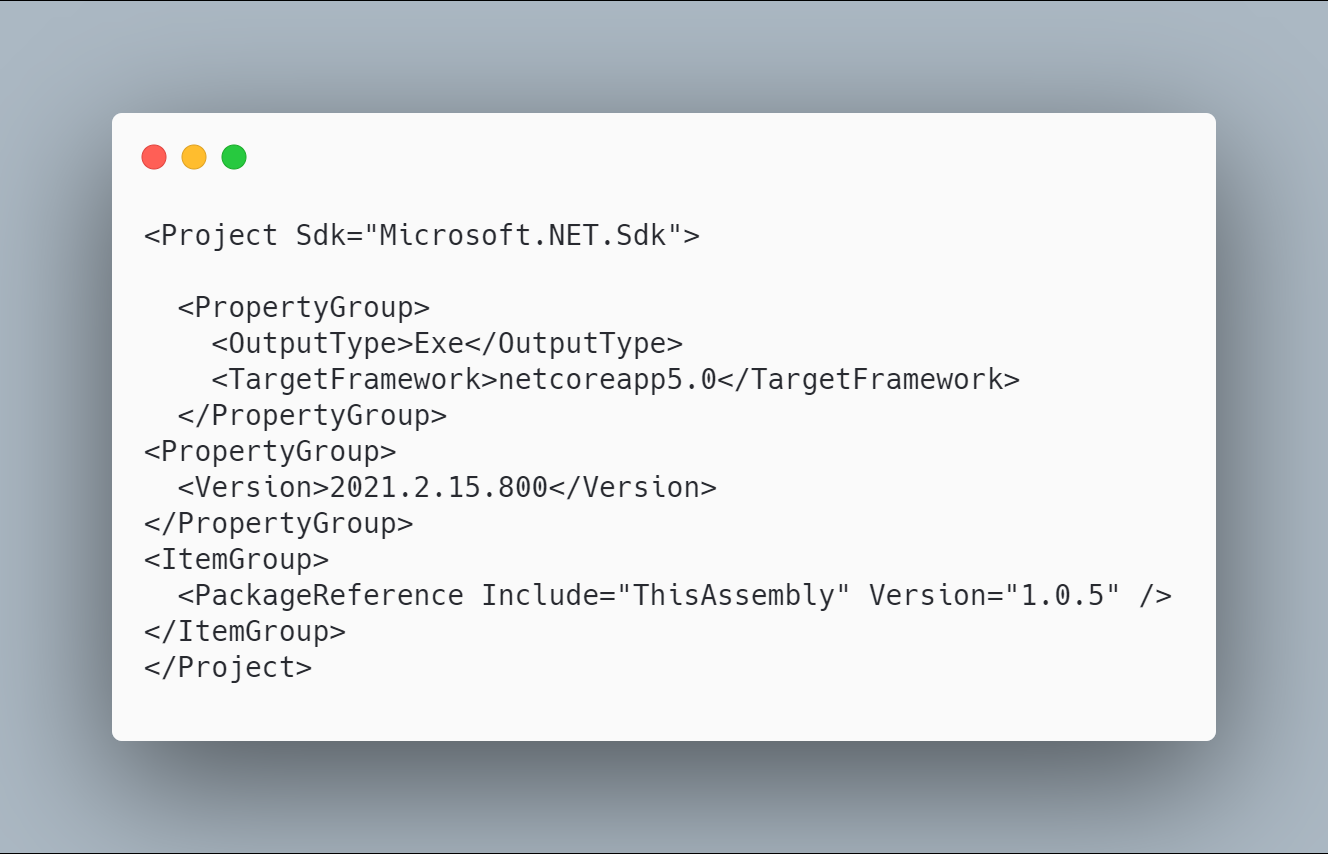
link : <https://www.clarius.org/ThisAssembly/>

author :Daniel Cazzulino

### What RSCG ThisAssembly can do

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.

### Here is the csproj with the references for RSCG ThisAssembly

 [code](http://ignatandrei.github.io/RSCG_Examples/images/ThisAssembly/The.csproj)

### The code that you start with is

 [code](http://ignatandrei.github.io/RSCG_Examples/images/ThisAssembly/ExistingCode.cs)

### The code that you will use is

 [code](http://ignatandrei.github.io/RSCG_Examples/images/ThisAssembly/Usage.cs)

### The code that is generated by RSCG ThisAssembly

 [code](http://ignatandrei.github.io/RSCG_Examples/images/ThisAssembly/GeneratedCode.cs)

### Link to Example Code:

[<https://github.com/ignatandrei/RSCG_Examples/tree/main/ApplicationVersion>](https://github.com/ignatandrei/RSCG_Examples/tree/main/ApplicationVersion)

## RSCG number 2 : Enum

Nuget : <https://www.nuget.org/packages/AOPMethodsCommon/> <https://www.nuget.org/packages/AOPMethodsGenerator/>

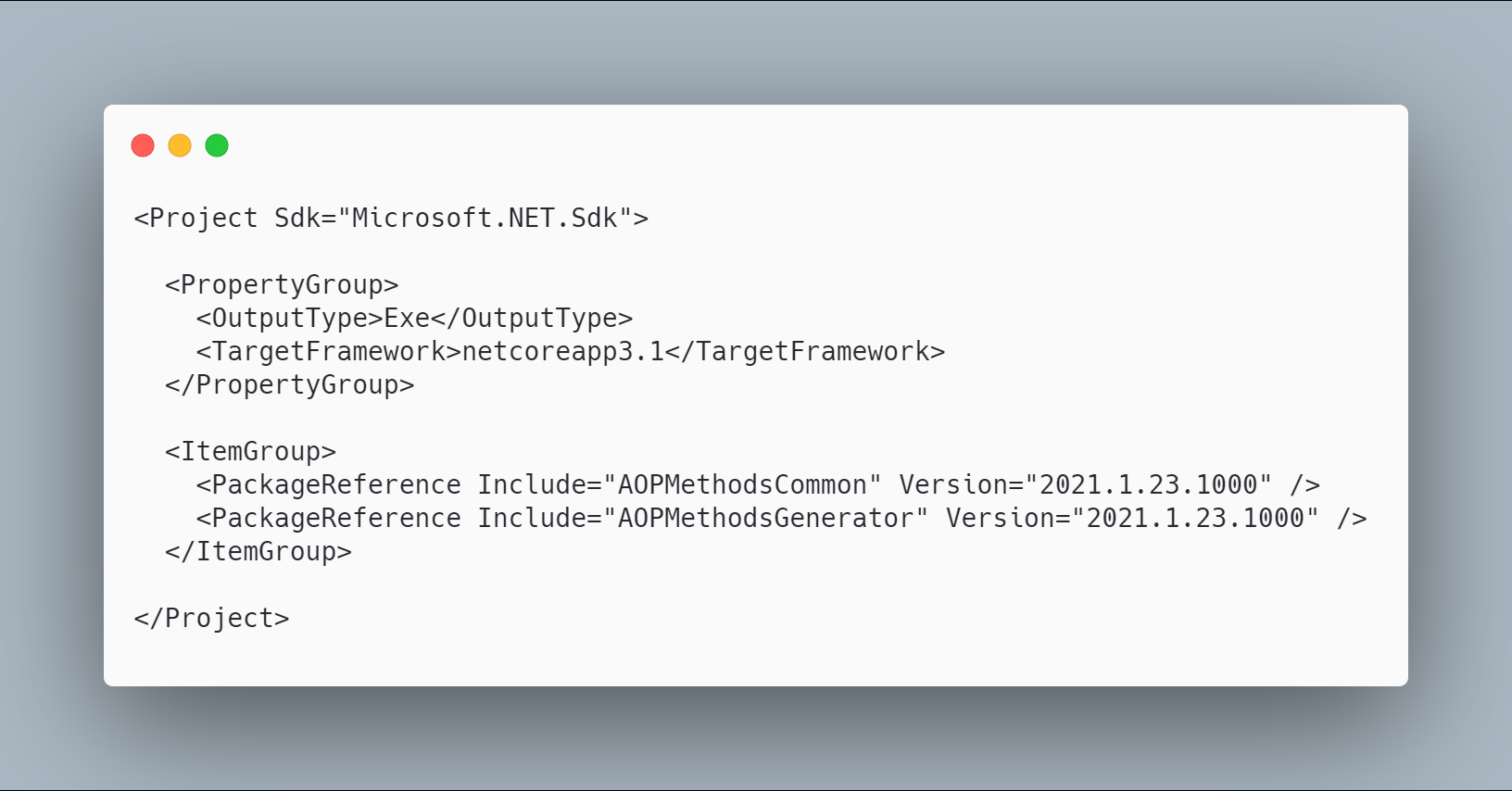
link : <http://msprogrammer.serviciipeweb.ro/category/roslyn/>

author :Andrei Ignat

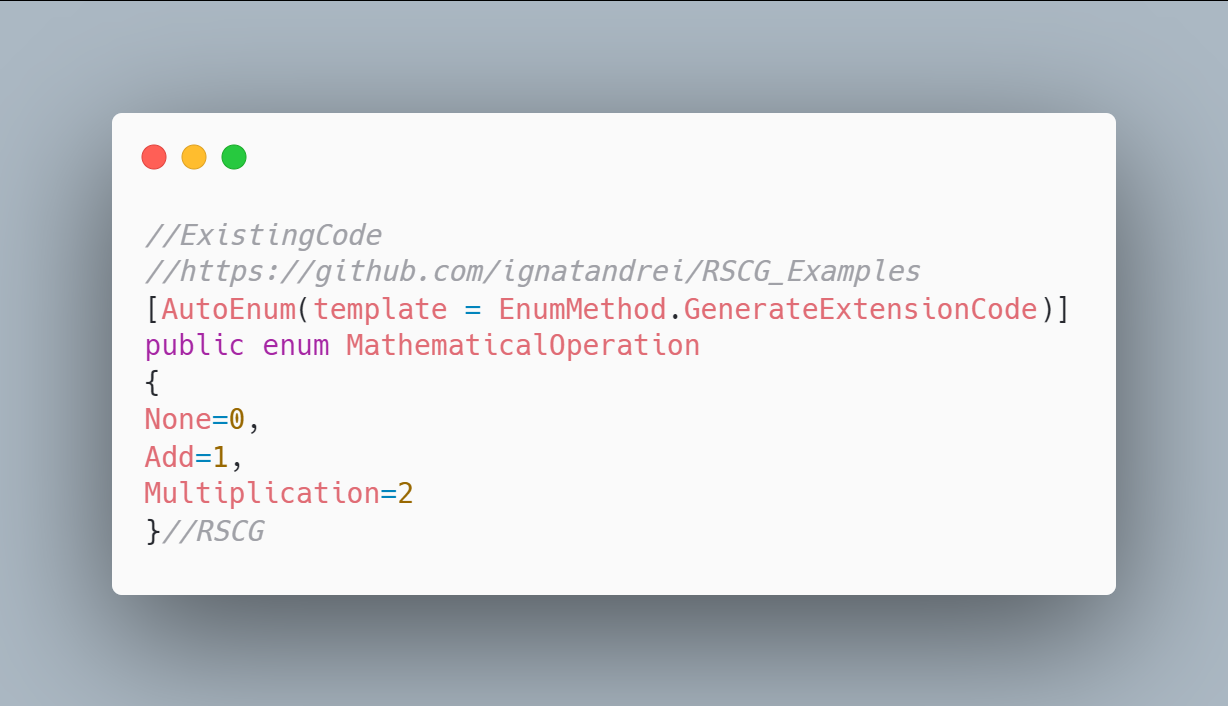
### What RSCG Enum can do

This will generate code to fast parsing a int or a string to an enum

### Here is the csproj with the references for RSCG Enum

 [code](http://ignatandrei.github.io/RSCG_Examples/images/Enum/The.csproj)

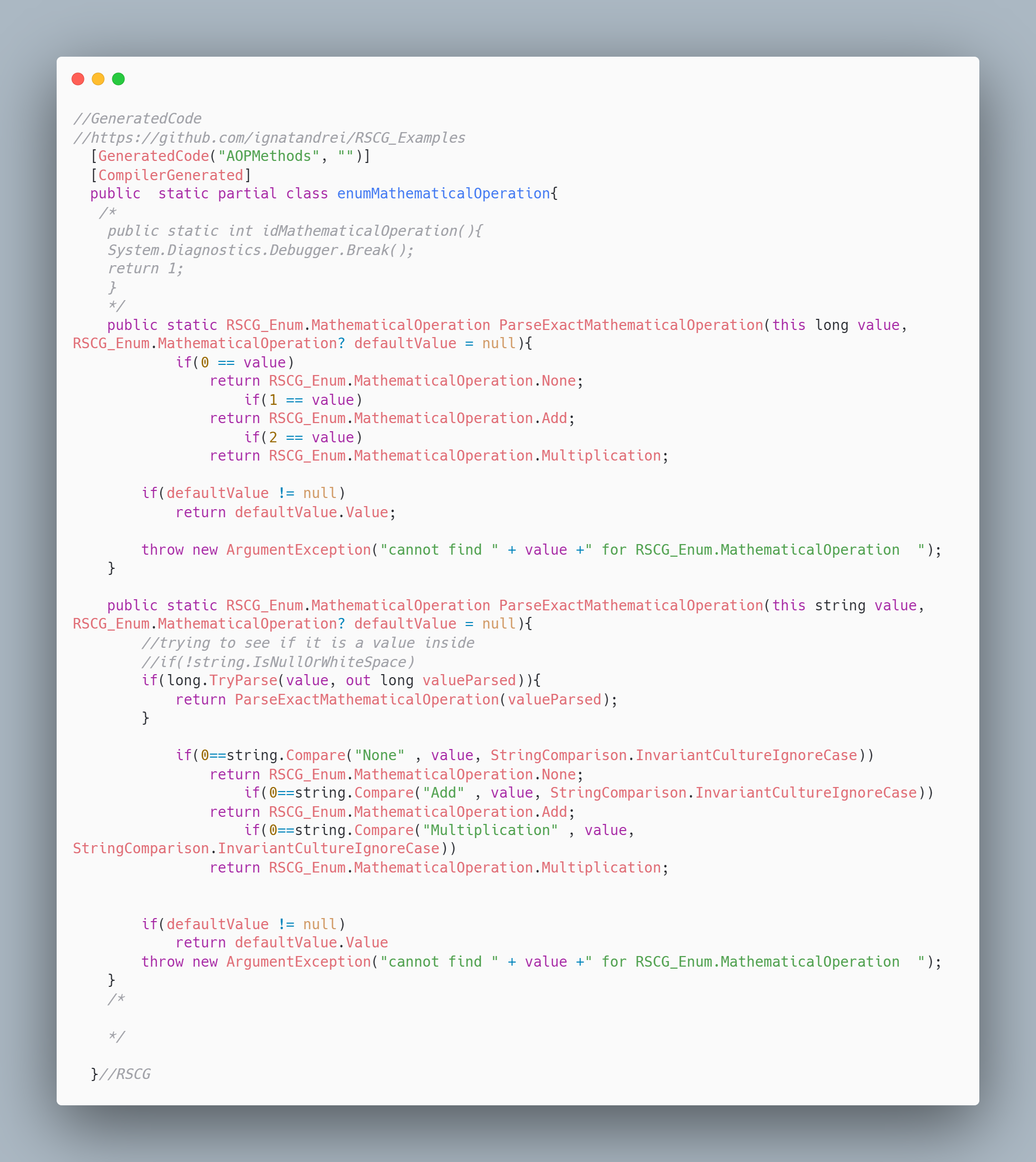
### The code that you start with is

 [code](http://ignatandrei.github.io/RSCG_Examples/images/Enum/ExistingCode.cs)

### The code that you will use is

 [code](http://ignatandrei.github.io/RSCG_Examples/images/Enum/Usage.cs)

### The code that is generated by RSCG Enum

 [code](http://ignatandrei.github.io/RSCG_Examples/images/Enum/GeneratedCode.cs)

### Link to Example Code:

[<https://github.com/ignatandrei/RSCG_Examples/tree/main/Enum>](https://github.com/ignatandrei/RSCG_Examples/tree/main/Enum)

## RSCG number 3 : JsonByExampleGenerator

Nuget : <https://www.nuget.org/packages/JsonByExampleGenerator/>

link : <https://github.com/hermanussen/JsonByExampleGenerator/>

author :Robin Hermanussen

### What RSCG JsonByExampleGenerator can do

This will generate C# classes from json files.

### Here is the csproj with the references for RSCG JsonByExampleGenerator

 [code](http://ignatandrei.github.io/RSCG_Examples/images/JsonByExampleGenerator/The.csproj)

### The code that you start with is

 [code](http://ignatandrei.github.io/RSCG_Examples/images/JsonByExampleGenerator/ExistingCode.cs)

### The code that you will use is

 [code](http://ignatandrei.github.io/RSCG_Examples/images/JsonByExampleGenerator/Usage.cs)

### The code that is generated by RSCG JsonByExampleGenerator

 [code](http://ignatandrei.github.io/RSCG_Examples/images/JsonByExampleGenerator/GeneratedCode.cs)

### Link to Example Code:

[<https://github.com/ignatandrei/RSCG_Examples/tree/main/JsonToClass>](https://github.com/ignatandrei/RSCG_Examples/tree/main/JsonToClass)

## RSCG number 4 : CopyConstructor + Deconstructor

Nuget : <https://www.nuget.org/packages/AOPMethodsCommon/> <https://www.nuget.org/packages/AOPMethodsGenerator/>

link : <http://msprogrammer.serviciipeweb.ro/category/roslyn/>

author :Andrei Ignat

### What RSCG CopyConstructor + Deconstructor can do

This will generate code for a POCO to generate copy constructor and deconstructor

### Here is the csproj with the references for RSCG CopyConstructor + Deconstructor

![CopyConstructor + Deconstructor](<http://ignatandrei.github.io/RSCG_Examples/images/CopyConstructor> + Deconstructor/The.csproj.png) [code](<http://ignatandrei.github.io/RSCG_Examples/images/CopyConstructor> + Deconstructor/The.csproj)

### The code that you start with is

![CopyConstructor + Deconstructor](<http://ignatandrei.github.io/RSCG_Examples/images/CopyConstructor> + Deconstructor/ExistingCode.cs.png) [code](<http://ignatandrei.github.io/RSCG_Examples/images/CopyConstructor> + Deconstructor/ExistingCode.cs)

### The code that you will use is

![CopyConstructor + Deconstructor](<http://ignatandrei.github.io/RSCG_Examples/images/CopyConstructor> + Deconstructor/Usage.cs.png) [code](<http://ignatandrei.github.io/RSCG_Examples/images/CopyConstructor> + Deconstructor/Usage.cs)

### The code that is generated by RSCG CopyConstructor + Deconstructor

![CopyConstructor + Deconstructor](<http://ignatandrei.github.io/RSCG_Examples/images/CopyConstructor> + Deconstructor/GeneratedCode.cs.png) [code](<http://ignatandrei.github.io/RSCG_Examples/images/CopyConstructor> + Deconstructor/GeneratedCode.cs)

### Link to Example Code:

[<https://github.com/ignatandrei/RSCG_Examples/tree/main/CopyConstructor>](https://github.com/ignatandrei/RSCG_Examples/tree/main/CopyConstructor)

## RSCG number 5 : GeneratedMapper

Nuget : <https://www.nuget.org/packages/GeneratedMapper/>

link : <https://github.com/ThomasBleijendaal/GeneratedMapper>

author :Thomas Bleijendaal

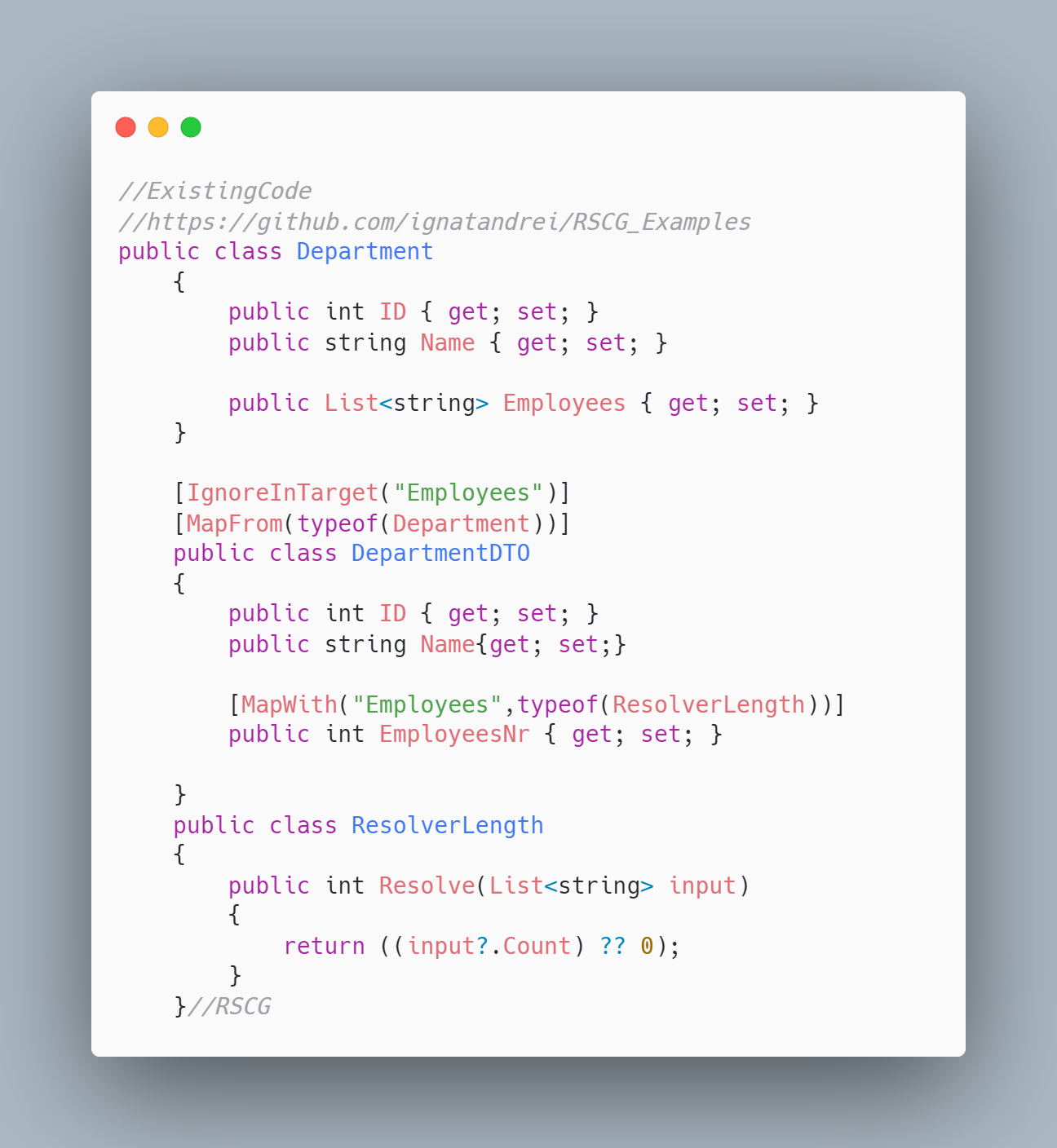
### What RSCG GeneratedMapper can do

AutoMapping from a POCO to a DTO. Lots of customizations

### Here is the csproj with the references for RSCG GeneratedMapper

 [code](http://ignatandrei.github.io/RSCG_Examples/images/GeneratedMapper/The.csproj)

### The code that you start with is

 [code](http://ignatandrei.github.io/RSCG_Examples/images/GeneratedMapper/ExistingCode.cs)

### The code that you will use is

 [code](http://ignatandrei.github.io/RSCG_Examples/images/GeneratedMapper/Usage.cs)

### The code that is generated by RSCG GeneratedMapper

 [code](http://ignatandrei.github.io/RSCG_Examples/images/GeneratedMapper/GeneratedCode.cs)

### Link to Example Code:

[<https://github.com/ignatandrei/RSCG_Examples/tree/main/DTOMapper>](https://github.com/ignatandrei/RSCG_Examples/tree/main/DTOMapper)

## RSCG number 6 : Skinny Controllers

Nuget : <https://www.nuget.org/packages/SkinnyControllersCommon/> <https://www.nuget.org/packages/SkinnyControllersGenerator/>

link : <http://msprogrammer.serviciipeweb.ro/category/roslyn/>

author :Andrei Ignat

### What RSCG Skinny Controllers can do

This will generate code for WebAPI for each method of a field in the controller

### Here is the csproj with the references for RSCG Skinny Controllers

![Skinny Controllers](<http://ignatandrei.github.io/RSCG_Examples/images/Skinny> Controllers/The.csproj.png) [code](<http://ignatandrei.github.io/RSCG_Examples/images/Skinny> Controllers/The.csproj)

### The code that you start with is

![Skinny Controllers](<http://ignatandrei.github.io/RSCG_Examples/images/Skinny> Controllers/ExistingCode.cs.png) [code](<http://ignatandrei.github.io/RSCG_Examples/images/Skinny> Controllers/ExistingCode.cs)

### The code that you will use is

![Skinny Controllers](<http://ignatandrei.github.io/RSCG_Examples/images/Skinny> Controllers/Usage.cs.png) [code](<http://ignatandrei.github.io/RSCG_Examples/images/Skinny> Controllers/Usage.cs)

### The code that is generated by RSCG Skinny Controllers

![Skinny Controllers](<http://ignatandrei.github.io/RSCG_Examples/images/Skinny> Controllers/GeneratedCode.cs.png) [code](<http://ignatandrei.github.io/RSCG_Examples/images/Skinny> Controllers/GeneratedCode.cs)

### Link to Example Code:

[<https://github.com/ignatandrei/RSCG_Examples/tree/main/SkinnyControllers>](https://github.com/ignatandrei/RSCG_Examples/tree/main/SkinnyControllers)

## RSCG number 7 : data-builder-generator

Nuget : <https://www.nuget.org/packages/DasMulli.DataBuilderGenerator/>

link : <https://github.com/dasMulli/data-builder-generator>

author :Martin Andreas Ulrich

### What RSCG data-builder-generator can do

Implements the Builder Design pattern for any class. Useful , at least, for test projects

### Here is the csproj with the references for RSCG data-builder-generator

 [code](http://ignatandrei.github.io/RSCG_Examples/images/data-builder-generator/The.csproj)

### The code that you start with is

 [code](http://ignatandrei.github.io/RSCG_Examples/images/data-builder-generator/ExistingCode.cs)

### The code that you will use is

 [code](http://ignatandrei.github.io/RSCG_Examples/images/data-builder-generator/Usage.cs)

### The code that is generated by RSCG data-builder-generator

 [code](http://ignatandrei.github.io/RSCG_Examples/images/data-builder-generator/GeneratedCode.cs)

### Link to Example Code:

[<https://github.com/ignatandrei/RSCG_Examples/tree/main/DP_Builder>](https://github.com/ignatandrei/RSCG_Examples/tree/main/DP_Builder)

## RSCG number 8 : Metadata from object

Nuget : <https://www.nuget.org/packages/AOPMethodsCommon/> <https://www.nuget.org/packages/AOPMethodsGenerator/>

link : <http://msprogrammer.serviciipeweb.ro/category/roslyn/>

author :Andrei Ignat

### What RSCG Metadata from object can do

This will generate code to retrieve the values of properties directly, not by reflection

### Here is the csproj with the references for RSCG Metadata from object

![Metadata from object](<http://ignatandrei.github.io/RSCG_Examples/images/Metadata> from object/The.csproj.png) [code](<http://ignatandrei.github.io/RSCG_Examples/images/Metadata> from object/The.csproj)

### The code that you start with is

![Metadata from object](<http://ignatandrei.github.io/RSCG_Examples/images/Metadata> from object/ExistingCode.cs.png) [code](<http://ignatandrei.github.io/RSCG_Examples/images/Metadata> from object/ExistingCode.cs)

### The code that you will use is

![Metadata from object](<http://ignatandrei.github.io/RSCG_Examples/images/Metadata> from object/Usage.cs.png) [code](<http://ignatandrei.github.io/RSCG_Examples/images/Metadata> from object/Usage.cs)

### The code that is generated by RSCG Metadata from object

![Metadata from object](<http://ignatandrei.github.io/RSCG_Examples/images/Metadata> from object/GeneratedCode.cs.png) [code](<http://ignatandrei.github.io/RSCG_Examples/images/Metadata> from object/GeneratedCode.cs)

### Link to Example Code:

[<https://github.com/ignatandrei/RSCG_Examples/tree/main/MetadataFromObject>](https://github.com/ignatandrei/RSCG_Examples/tree/main/MetadataFromObject)

## RSCG number 9 : MockSourceGenerator

Nuget : <https://www.nuget.org/packages/MockSourceGenerator/>

link : <https://github.com/hermanussen/MockSourceGenerator/>

author :Robin Hermanussen

### What RSCG MockSourceGenerator can do

This will generate Mock classes directly for any interface - with your implementation.

### Here is the csproj with the references for RSCG MockSourceGenerator

 [code](http://ignatandrei.github.io/RSCG_Examples/images/MockSourceGenerator/The.csproj)

### The code that you start with is

 [code](http://ignatandrei.github.io/RSCG_Examples/images/MockSourceGenerator/ExistingCode.cs)

### The code that you will use is

 [code](http://ignatandrei.github.io/RSCG_Examples/images/MockSourceGenerator/Usage.cs)

### The code that is generated by RSCG MockSourceGenerator

 [code](http://ignatandrei.github.io/RSCG_Examples/images/MockSourceGenerator/GeneratedCode.cs)

### Link to Example Code:

[<https://github.com/ignatandrei/RSCG_Examples/tree/main/DynamicMocking>](https://github.com/ignatandrei/RSCG_Examples/tree/main/DynamicMocking)

## RSCG number 10 : Method decorator

Nuget : <https://www.nuget.org/packages/AOPMethodsCommon/> <https://www.nuget.org/packages/AOPMethodsGenerator/>

link : <http://msprogrammer.serviciipeweb.ro/category/roslyn/>

author :Andrei Ignat

### What RSCG Method decorator can do

This will generate code to decorate methods with anything you want ( stopwatch, logging , authorization...)

### Here is the csproj with the references for RSCG Method decorator

![Method decorator](<http://ignatandrei.github.io/RSCG_Examples/images/Method> decorator/The.csproj.png) [code](<http://ignatandrei.github.io/RSCG_Examples/images/Method> decorator/The.csproj)

### The code that you start with is

![Method decorator](<http://ignatandrei.github.io/RSCG_Examples/images/Method> decorator/ExistingCode.cs.png) [code](<http://ignatandrei.github.io/RSCG_Examples/images/Method> decorator/ExistingCode.cs)

### The code that you will use is

![Method decorator](<http://ignatandrei.github.io/RSCG_Examples/images/Method> decorator/Usage.cs.png) [code](<http://ignatandrei.github.io/RSCG_Examples/images/Method> decorator/Usage.cs)

### The code that is generated by RSCG Method decorator

![Method decorator](<http://ignatandrei.github.io/RSCG_Examples/images/Method> decorator/GeneratedCode.cs.png) [code](<http://ignatandrei.github.io/RSCG_Examples/images/Method> decorator/GeneratedCode.cs)

### Link to Example Code:

[<https://github.com/ignatandrei/RSCG_Examples/tree/main/MethodDecorator>](https://github.com/ignatandrei/RSCG_Examples/tree/main/MethodDecorator)

## RSCG number 11 : PartiallyApplied

Nuget : <https://www.nuget.org/packages/PartiallyApplied/>

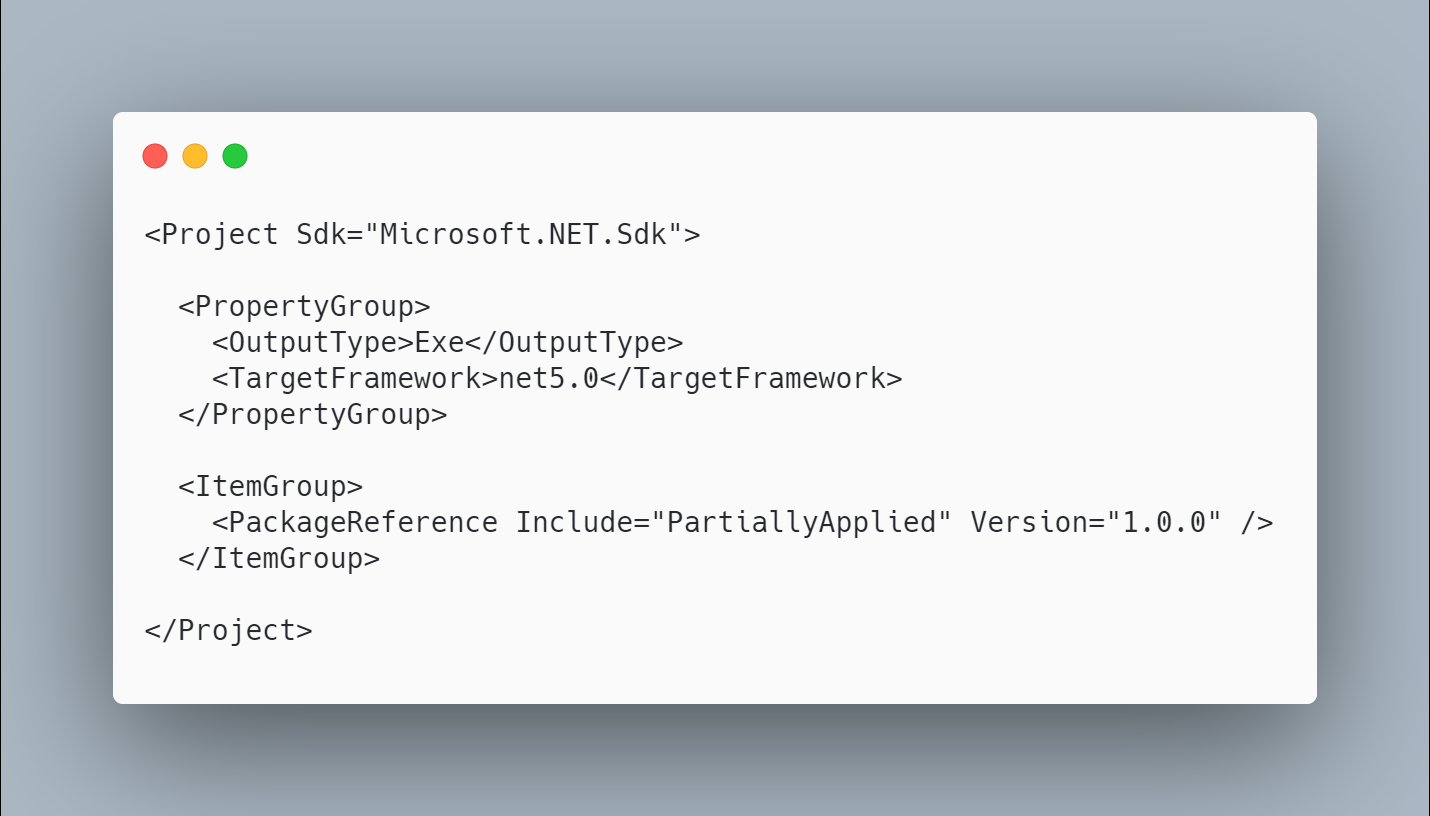
link : <https://github.com/JasonBock/PartiallyApplied>

author :Andrei Ignat

### What RSCG PartiallyApplied can do

This will generate curry for your functions

### Here is the csproj with the references for RSCG PartiallyApplied

 [code](http://ignatandrei.github.io/RSCG_Examples/images/PartiallyApplied/The.csproj)

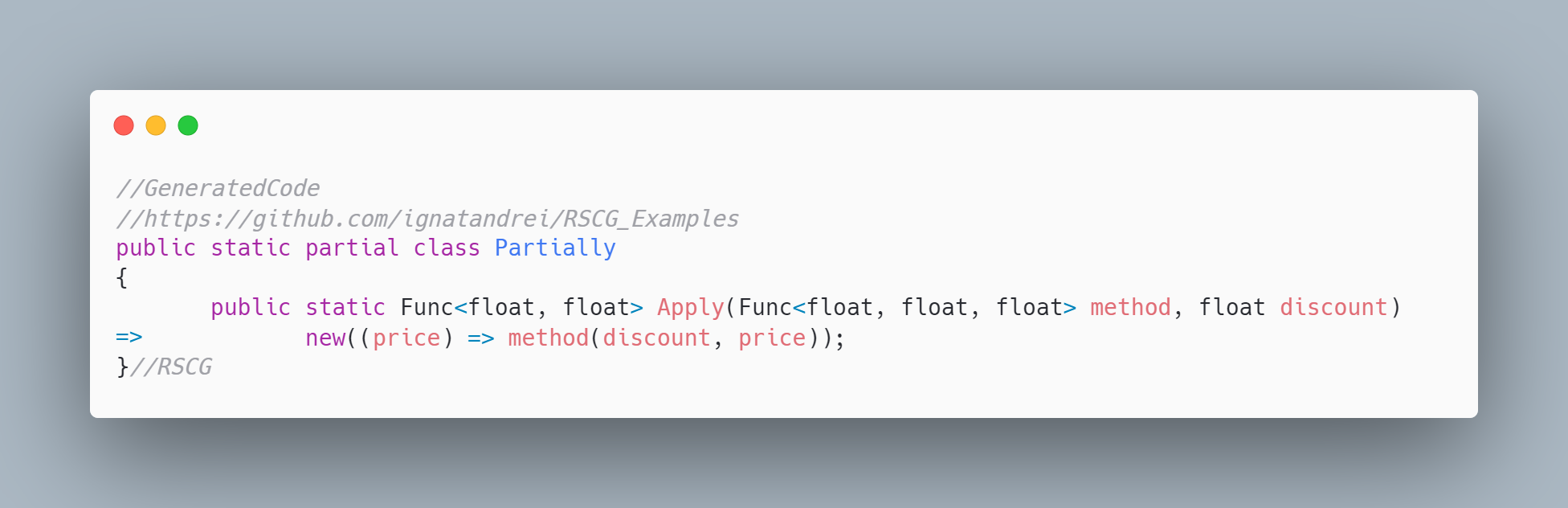
### The code that you start with is

 [code](http://ignatandrei.github.io/RSCG_Examples/images/PartiallyApplied/ExistingCode.cs)

### The code that you will use is

 [code](http://ignatandrei.github.io/RSCG_Examples/images/PartiallyApplied/Usage.cs)

### The code that is generated by RSCG PartiallyApplied

 [code](http://ignatandrei.github.io/RSCG_Examples/images/PartiallyApplied/GeneratedCode.cs)

### Link to Example Code:

[<https://github.com/ignatandrei/RSCG_Examples/tree/main/PartiallyFunction>](https://github.com/ignatandrei/RSCG_Examples/tree/main/PartiallyFunction)

## RSCG number 12 : IFormattable

Nuget : <https://www.nuget.org/packages/AOPMethodsCommon/> <https://www.nuget.org/packages/AOPMethodsGenerator/>

link : <http://msprogrammer.serviciipeweb.ro/category/roslyn/>

author :Andrei Ignat

### What RSCG IFormattable can do

This will generate code to add IFormattable to any class, based on the properties of the class

### Here is the csproj with the references for RSCG IFormattable

 [code](http://ignatandrei.github.io/RSCG_Examples/images/IFormattable/The.csproj)

### The code that you start with is

 [code](http://ignatandrei.github.io/RSCG_Examples/images/IFormattable/ExistingCode.cs)

### The code that you will use is

 [code](http://ignatandrei.github.io/RSCG_Examples/images/IFormattable/Usage.cs)

### The code that is generated by RSCG IFormattable

 [code](http://ignatandrei.github.io/RSCG_Examples/images/IFormattable/GeneratedCode.cs)

### Link to Example Code:

[<https://github.com/ignatandrei/RSCG_Examples/tree/main/IFormattable>](https://github.com/ignatandrei/RSCG_Examples/tree/main/IFormattable)

## RSCG number 13 : AutoInterface

Nuget : <https://www.nuget.org/packages/BeaKona.AutoInterfaceGenerator>

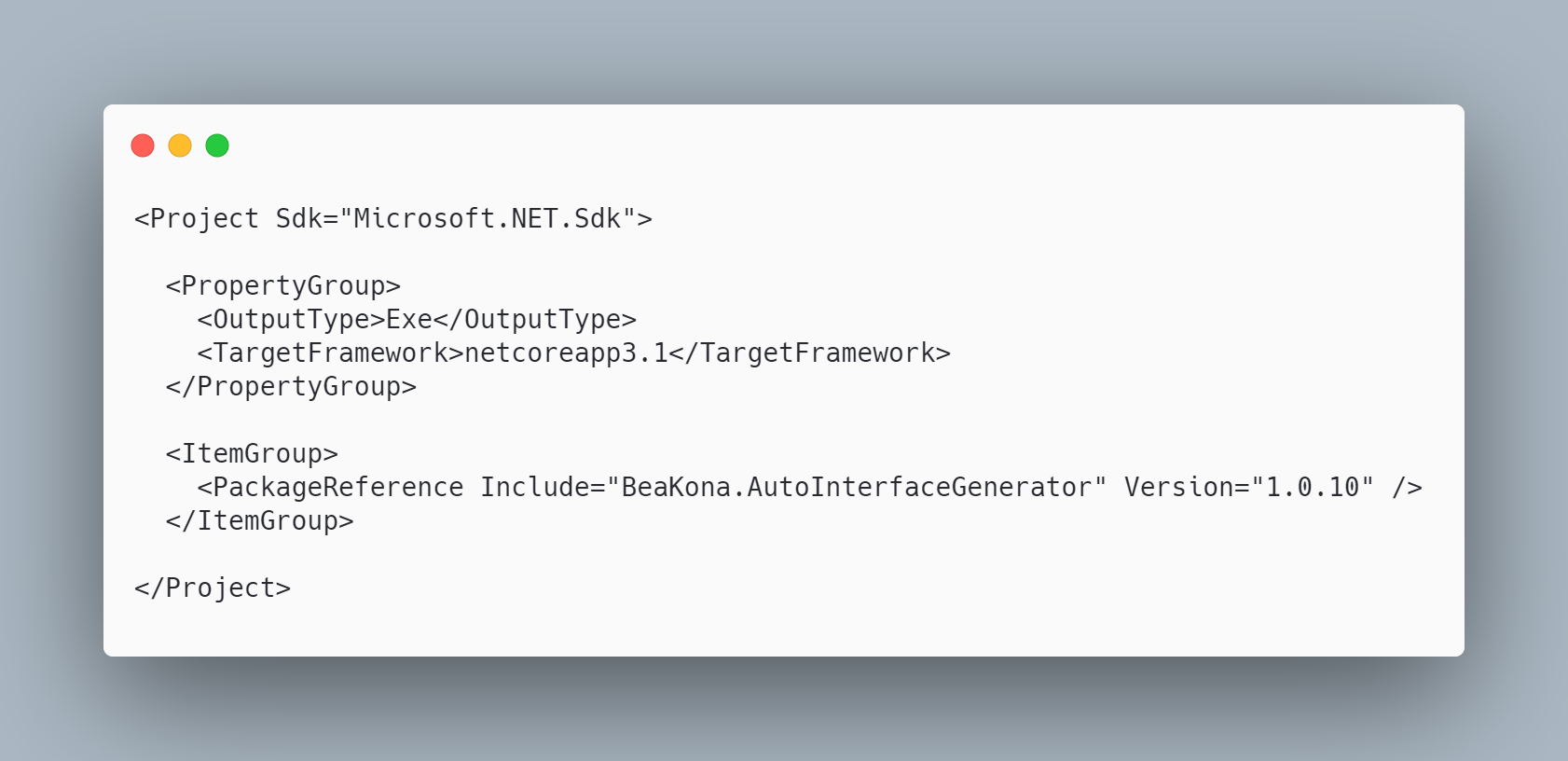
link : <https://github.com/beakona/AutoInterface>

author :beakona

### What RSCG AutoInterface can do

Implement the Design Pattern Decorator. Based on template - you can modify the source code generated

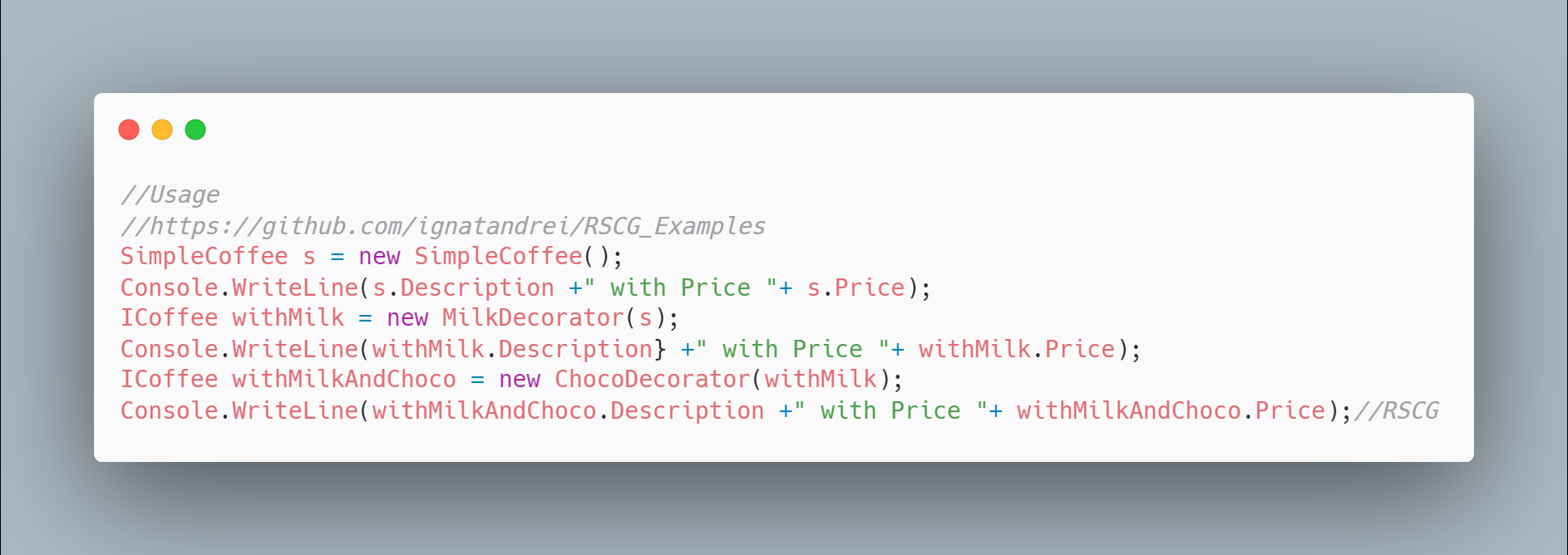
### Here is the csproj with the references for RSCG AutoInterface

 [code](http://ignatandrei.github.io/RSCG_Examples/images/AutoInterface/The.csproj)

### The code that you start with is

 [code](http://ignatandrei.github.io/RSCG_Examples/images/AutoInterface/ExistingCode.cs)

### The code that you will use is

 [code](http://ignatandrei.github.io/RSCG_Examples/images/AutoInterface/Usage.cs)

### The code that is generated by RSCG AutoInterface

 [code](http://ignatandrei.github.io/RSCG_Examples/images/AutoInterface/GeneratedCode.cs)

### Link to Example Code:

[<https://github.com/ignatandrei/RSCG_Examples/tree/main/DP_Decorator>](https://github.com/ignatandrei/RSCG_Examples/tree/main/DP_Decorator)

## RSCG number 14 : Property Expression Generator

Nuget : <https://www.nuget.org/packages/AOPMethodsCommon/> <https://www.nuget.org/packages/AOPMethodsGenerator/>

link : <http://msprogrammer.serviciipeweb.ro/category/roslyn/>

author :Andrei Ignat

### What RSCG Property Expression Generator can do

This will generate code to add function to be used with Entity Framework to search for any property of a class

### Here is the csproj with the references for RSCG Property Expression Generator

![Property Expression Generator](<http://ignatandrei.github.io/RSCG_Examples/images/Property> Expression Generator/The.csproj.png) [code](<http://ignatandrei.github.io/RSCG_Examples/images/Property> Expression Generator/The.csproj)

### The code that you start with is

![Property Expression Generator](<http://ignatandrei.github.io/RSCG_Examples/images/Property> Expression Generator/ExistingCode.cs.png) [code](<http://ignatandrei.github.io/RSCG_Examples/images/Property> Expression Generator/ExistingCode.cs)

### The code that you will use is

![Property Expression Generator](<http://ignatandrei.github.io/RSCG_Examples/images/Property> Expression Generator/Usage.cs.png) [code](<http://ignatandrei.github.io/RSCG_Examples/images/Property> Expression Generator/Usage.cs)

### The code that is generated by RSCG Property Expression Generator

![Property Expression Generator](<http://ignatandrei.github.io/RSCG_Examples/images/Property> Expression Generator/GeneratedCode.cs.png) [code](<http://ignatandrei.github.io/RSCG_Examples/images/Property> Expression Generator/GeneratedCode.cs)

### Link to Example Code:

[<https://github.com/ignatandrei/RSCG_Examples/tree/main/PropertyExpressionGenerator>](https://github.com/ignatandrei/RSCG_Examples/tree/main/PropertyExpressionGenerator)

## RSCG - worth mention

There are more RSCG that you could see - here is a list that you may want to look at:

1. AutoEmbed <https://github.com/chsienki/AutoEmbed>
2. Cloneable <https://github.com/mostmand/Cloneable>
3. fonderie <https://github.com/jeromelaban/fonderie>
4. Generators.Blazor <https://github.com/excubo-ag/Generators.Blazor>
5. Generators.Grouping <https://github.com/excubo-ag/Generators.Grouping>
6. JsonMergePatch <https://github.com/ladeak/JsonMergePatch>
7. MemoizeSourceGenerator <https://github.com/Zoxive/MemoizeSourceGenerator>
8. MiniRazor <https://github.com/Tyrrrz/MiniRazor/>
9. MockGen <https://github.com/thomas-girotto/MockGen>
10. ProxyGen <https://github.com/Sholtee/ProxyGen>
11. Rocks <https://github.com/JasonBock/Rocks>
12. RoslynWeave <https://github.com/Jishun/RoslynWeave>
13. SmallSharp <https://github.com/devlooped/SmallSharp>
14. StaticProxyGenerator <https://github.com/robertturner/StaticProxyGenerator>
15. ValueChangedGenerator <https://github.com/ufcpp/ValueChangedGenerator>
16. Web-Anchor <https://github.com/mattiasnordqvist/Web-Anchor>
17. WrapperValueObject <https://github.com/martinothamar/WrapperValueObject>