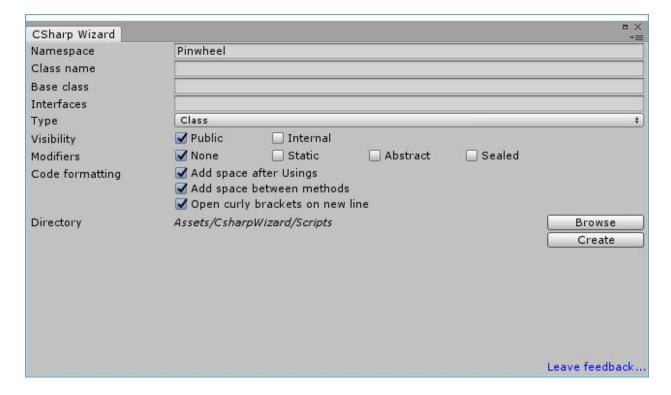
CSHARP WIZARD

User guide

TABLE OF CONTENT

Table	of Content	2
Overview		3
Creati	Creating a script file	
	Open the editor in you desired folder	
	Fill in some basic info	
	Click Create and see the result	
	e code modification	

OVERVIEW



Creating new script file is one of the most common tasks of any programmer working with Unity. For simple projects, there is nothing to worry about. But for larger projects, where inheritance and interfaces are everywhere, and you need to put your class into separate namespaces, it would be a trouble.

Until now, Unity only provides you a solution to deal with this: edit the template files, located in *InstallationPath/Editor/Data/Resources/ScriptTemplates/*, but is it enough? No, it's not! Because it simply find-and-replace, and would affect multiple projects.

CsharpWizard was made to solve that. The only thing you have to do is to tell what you want to create, and everything will be done automatically. Some highlights on what it can do:

- Generate the skeleton code with no compile error (of course!)

- Namespace, base class and interface suggestion based on what you type, sorting and giving the most relevant result
- Error handling, warning you before generating the code
- Auto override abstract methods from base class
- Auto implement properties and methods of the inherited interfaces
- Code formatting
- Multiple type of code to generate, from class, struct, to interface and enum

More feature will be added in the next update:

- Save and load custom settings template
- Template gallery to create CustomEditor, CustomInspector, etc.
- Adding more type of code like shader, etc.

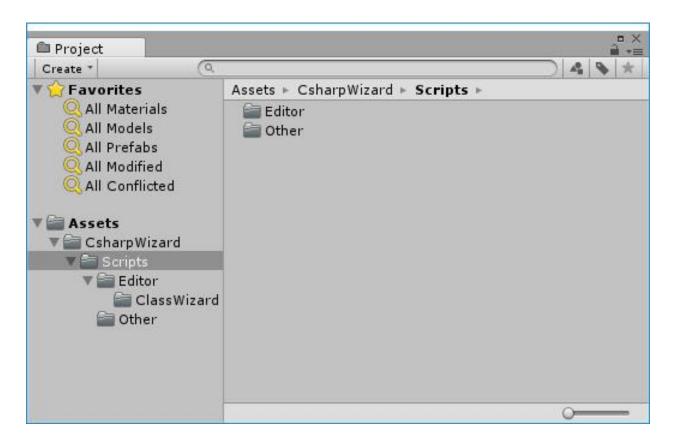
To request a feature, leave a feedback, or report a bug, please contact us via email: pinwheel.customer@gmail.com

CREATING A SCRIPT FILE

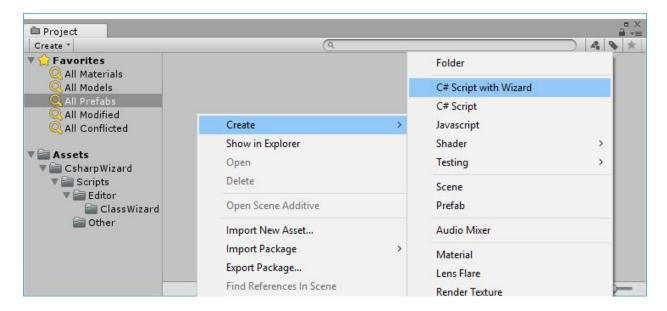
The creation of class, struct, interface and enum is nearly the same. The editor only shows you the appropriate options based on what you want to create. Below is a short guide to create a class.

1. Open the editor in you desired folder

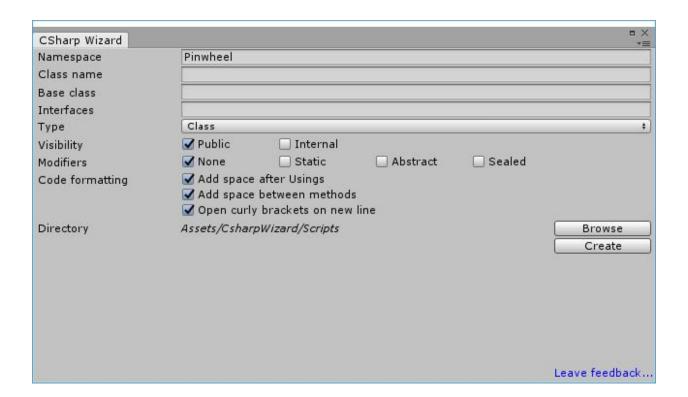
In the Project window, select the folder you want to save the new file to, for example: Assets/CsharpWizard/Scripts/



Right-click on the empty space of the window, then select Create/C# Script with Wizard



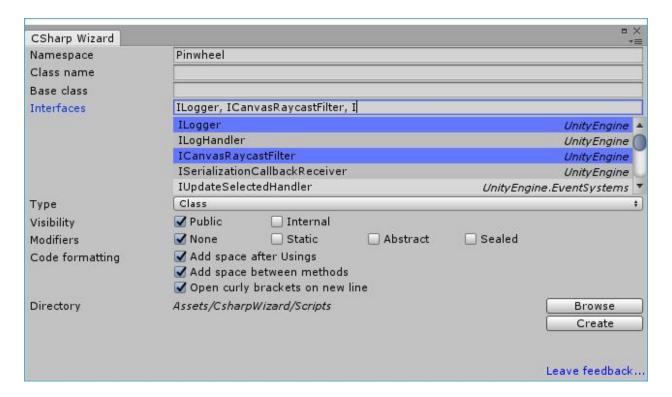
It will open up the Wizard, you will notice that Directory is set to Assets/CsharpWizard/Scripts



2. Fill in some basic info

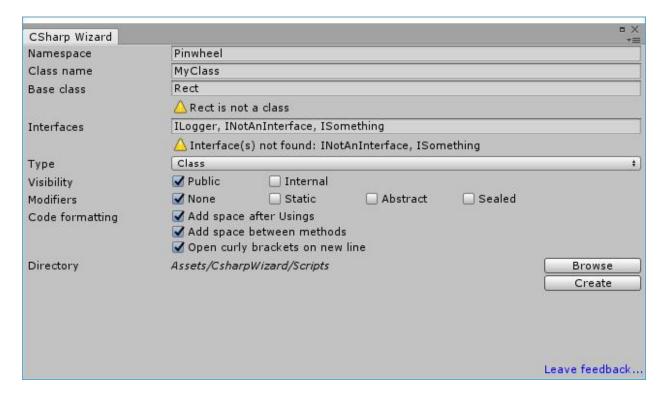
Type your namespace, class name, base class name and interfaces into the appropriate textbox. If you don't want it to inherit from any base class or interface, leave them blank. If you want to implement multiple interfaces, separate their name by commas.

After typing some characters, you would see a suggestion box shows up, like this:



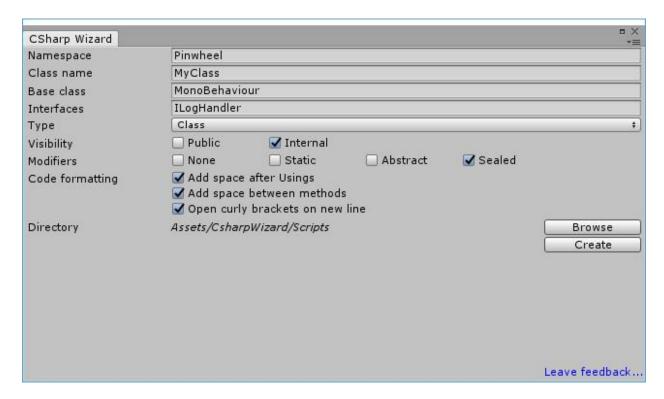
You can click on an item to select it.

If there is any error with these info, the warnings will show up:



Set Type to Class.

Select its visibility, modifiers and formatting options.



3. Click Create and see the result

Here is what you will get:

```
CsharpWizard

| CsharpWizard | Pinwheel.MyClass | DegFormat(LogType logType, Object context, string, or logs | DegFormat(LogType logType, object | DegFormat(L
```

SOURCE CODE MODIFICATION

You can modify the source code to fit your need. Everything is fully commented. Please see the .cs file for detail.

```
CsharpWizard - ClassWizard.cs
遭 Csharp Wizard. Editor
                                          🗸 🔩 Pinwheel Class Wizard
                                                                                        🕶 🐾 namespaceName
      1 Pusing ...
           ⊏namespace Pinwheel
                 public partial class ClassWizard : EditorWindow
                          (summary> Indicate type of the script to be generated
                     private enum ScriptType
                         Class, Struct, Interface, Enum
                     private static string namespaceName;
                     private static string className; //not only the name of the new class, but also the name of the new stru
                     private static string baseClassName; //name of the class which new class will inherite from
                     private static string baseClassNameError; //error when attempt to inherite from the base class, ex: base
                     private static string interfaceList; //list of interface names to implement, separated by comma
                     private static string interfaceListError; //error when attempt to implement these interfaces
                     private static ScriptType scriptType; //you want to generate a class, a struct, an interface or an enum?
                     private static string directory; //where to save the new file
                     private static bool isStatic; //is the new class static?
                     private static bool isAbstract; //is the new class abstract?
                     private static bool isSealed; //is the new class sealed?
private static bool isInternal: //is the class marked as internal?
107 %
```