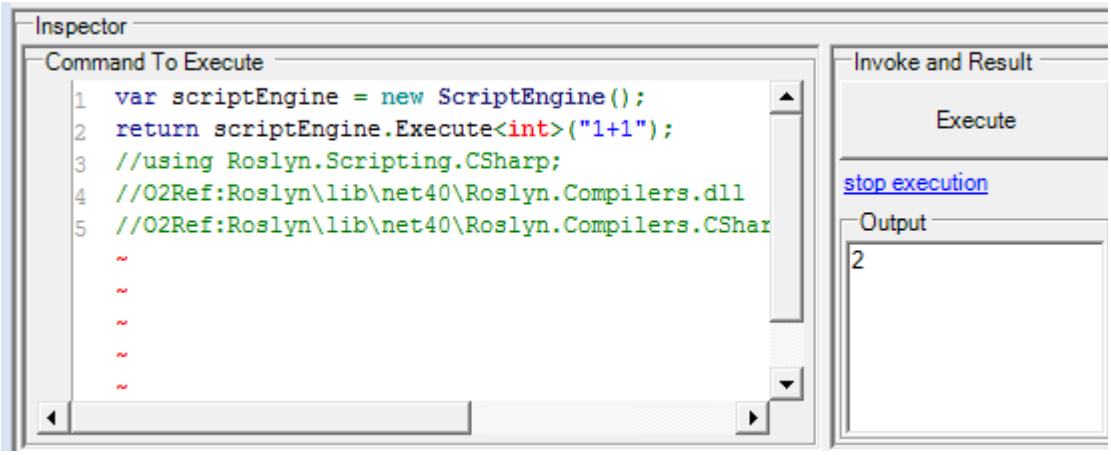
Using Roslyn (first tests)

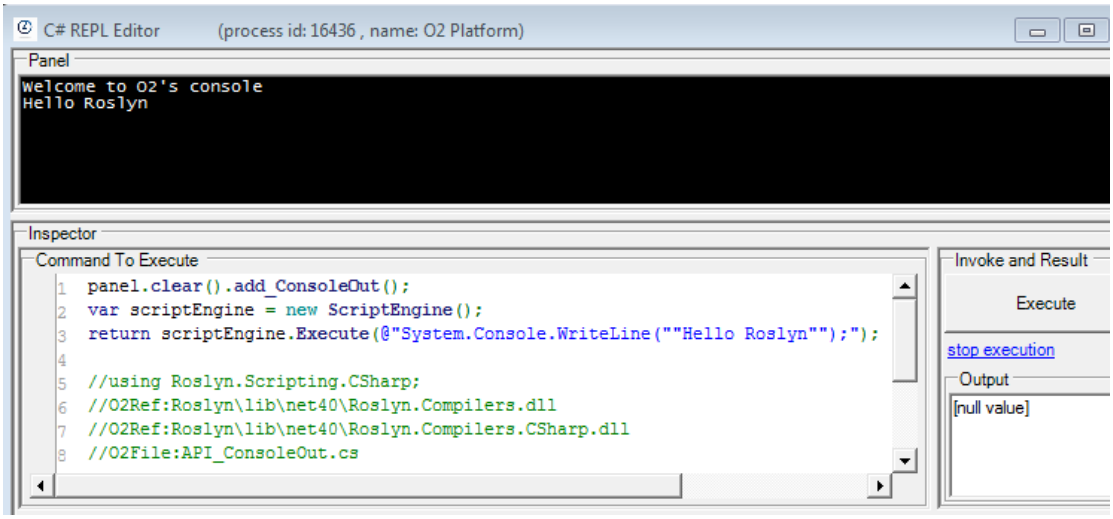
Executing C# snippet

```
var scriptEngine = new ScriptEngine();
return scriptEngine.Execute<int>("1+1");
//using Roslyn.Scripting.CSharp;
//O2Ref:Roslyn\lib\net40\Roslyn.Compilers.dll
//O2Ref:Roslyn\lib\net40\Roslyn.Compilers.CSharp.dll
```



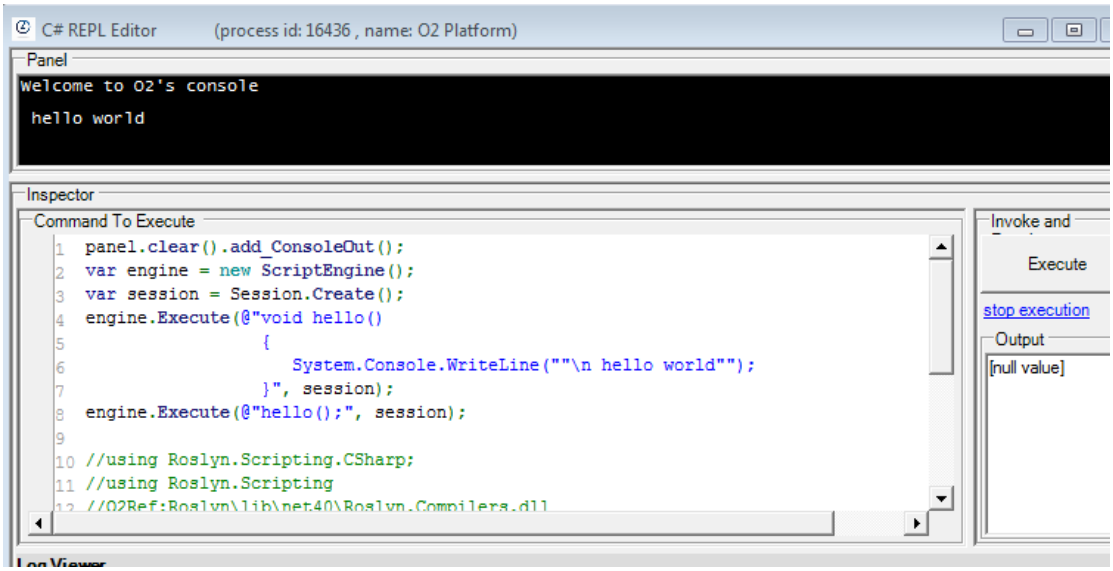
Invoking Console.WriteLine

```
panel.clear().add_ConsoleOut();
var scriptEngine = new ScriptEngine();
return scriptEngine.Execute(@"System.Console.WriteLine(""Hello Roslyn"");");

//using Roslyn.Scripting.CSharp;
//O2Ref:Roslyn\lib\net40\Roslyn.Compilers.dll
//O2Ref:Roslyn\lib\net40\Roslyn.Compilers.CSharp.dll
//O2File:API_ConsoleOut.cs
```



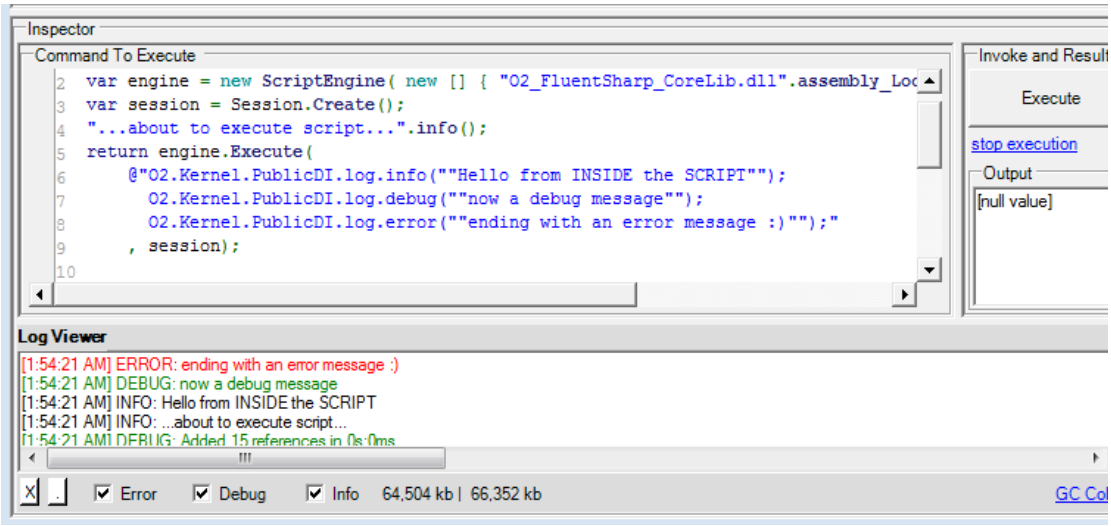
Creating methods inline

```
panel.clear().add_ConsoleOut();
var engine = new ScriptEngine();
var session = Session.Create();
engine.Execute(@"void hello()
                            {
                                  System.Console.WriteLine(""\n hello
world"");
                            }", session);
engine.Execute(@"hello();", session);

//using Roslyn.Scripting.CSharp;
//using Roslyn.Scripting
//O2Ref:Roslyn\lib\net40\Roslyn.Compilers.dll
//O2Ref:Roslyn\lib\net40\Roslyn.Compilers.CSharp.dll
//O2File:API_ConsoleOut.cs
```



Consuming O2 methods and interacting with the current process

```
panel.clear().add_ConsoleOut();
var engine = new ScriptEngine( new [] { "O2
_FluentSharp_CoreLib.dll".assembly_Location()});
var session = Session.Create();
"...about to execute script...".info();
return engine.Execute(
       @"O2.Kernel.PublicDI.log.info(""Hello from INSIDE the SCRIPT"");
         O2.Kernel.PublicDI.log.debug(""now a debug message"");
         O2.Kernel.PublicDI.log.error(""ending with an error message :)"");"
       , session);

//using Roslyn.Scripting.CSharp;
//using Roslyn.Scripting
//O2Ref:Roslyn\lib\net40\Roslyn.Compilers.dll
//O2Ref:Roslyn\lib\net40\Roslyn.Compilers.CSharp.dll
//O2File:API_ConsoleOut.cs
```

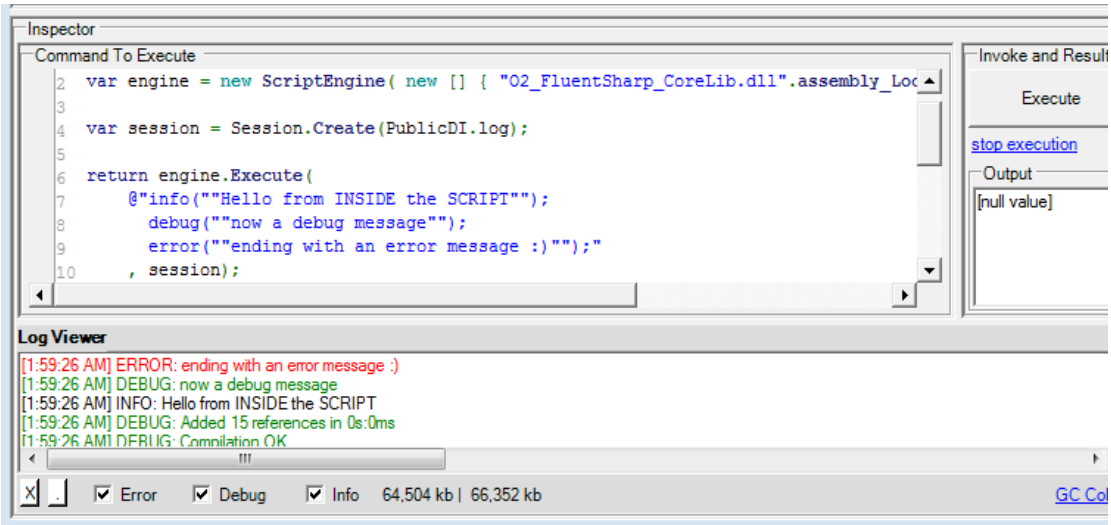


Using 'using' declarations to define namespaces

```
return engine.Execute(
       @"using O2.Kernel;
         PublicDI.log.info(""Hello from INSIDE the SCRIPT"");
         PublicDI.log.debug(""now a debug message"");
         PublicDI.log.error(""ending with an error message :)"");"
       , session);
```

Passing an object to Session.Create allows it to be used directly

```
var engine = new ScriptEngine( new [] { "O2
_FluentSharp_CoreLib.dll".assembly_Location()});

var session = Session.Create(PublicDI.log);

return engine.Execute(
       @"info(""Hello from INSIDE the SCRIPT"");
         debug(""now a debug message"");
         error(""ending with an error message :)"");"
       , session);
```



Creating a class and invoking it

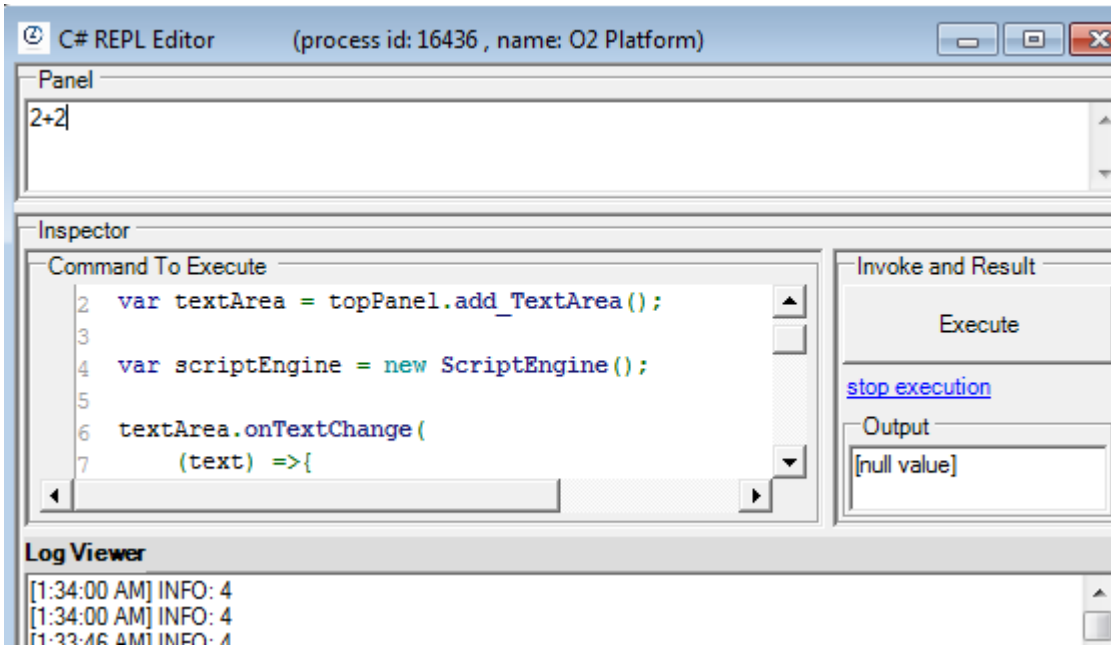
```
var engine = new ScriptEngine();
var session = Session.Create();

return engine.Execute(
       @"     public class DynamicClass
              {
                     public string Test()
                     {
                            return ""Test Method !!! "";
                     }
              }

              new DynamicClass().Test();"
       , session);

//using Roslyn.Scripting.CSharp;
//using Roslyn.Scripting
//O2Ref:Roslyn\lib\net40\Roslyn.Compilers.dll
//O2Ref:Roslyn\lib\net40\Roslyn.Compilers.CSharp.dll
```

.

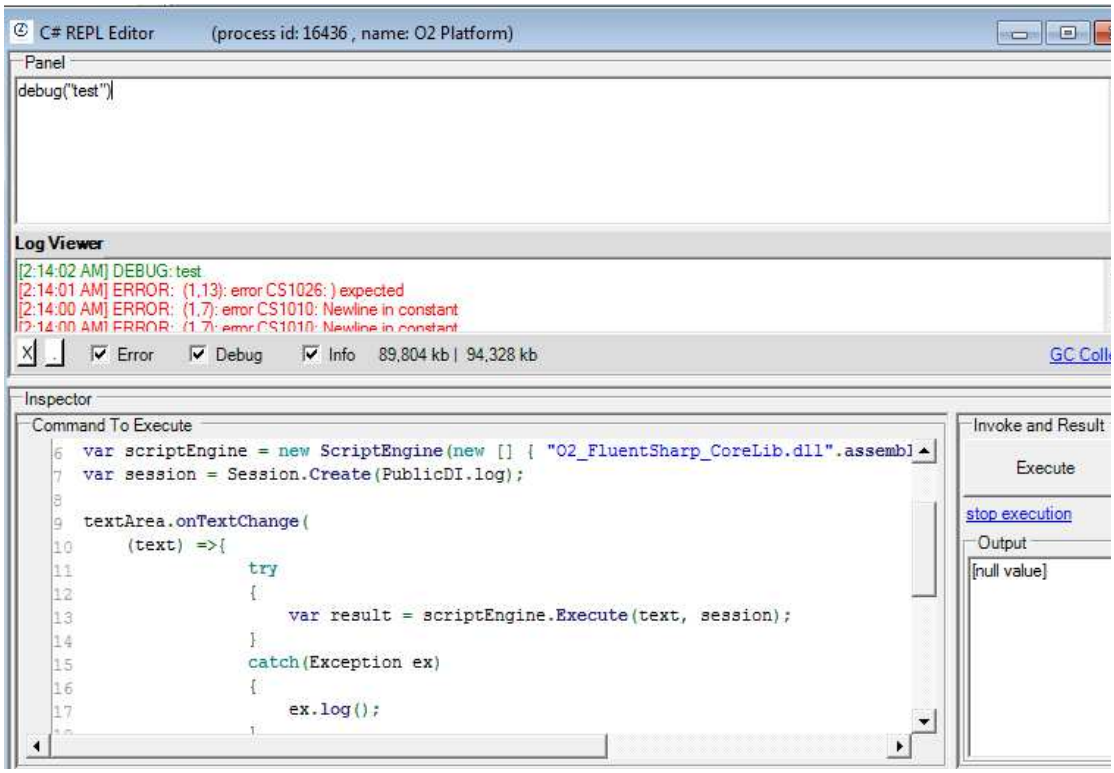
Creating a GUI to execute these scripts

Executing the code via a TextBox

```
var topPanel = panel.clear().add_Panel();
var textArea = topPanel.add_TextArea();

var scriptEngine = new ScriptEngine();

textArea.onTextChange(
       (text) =>{
                            try
                            {
                                   var result = scriptEngine.Execute(text);
                                   result.str().info();
                            }
                            catch(Exception ex)
                            {
                                   ex.log();
                            }
                     });

//using Roslyn.Scripting.CSharp;
//O2Ref:Roslyn\lib\net40\Roslyn.Compilers.dll
//O2Ref:Roslyn\lib\net40\Roslyn.Compilers.CSharp.dll
```



Accessing O2 classes from script:



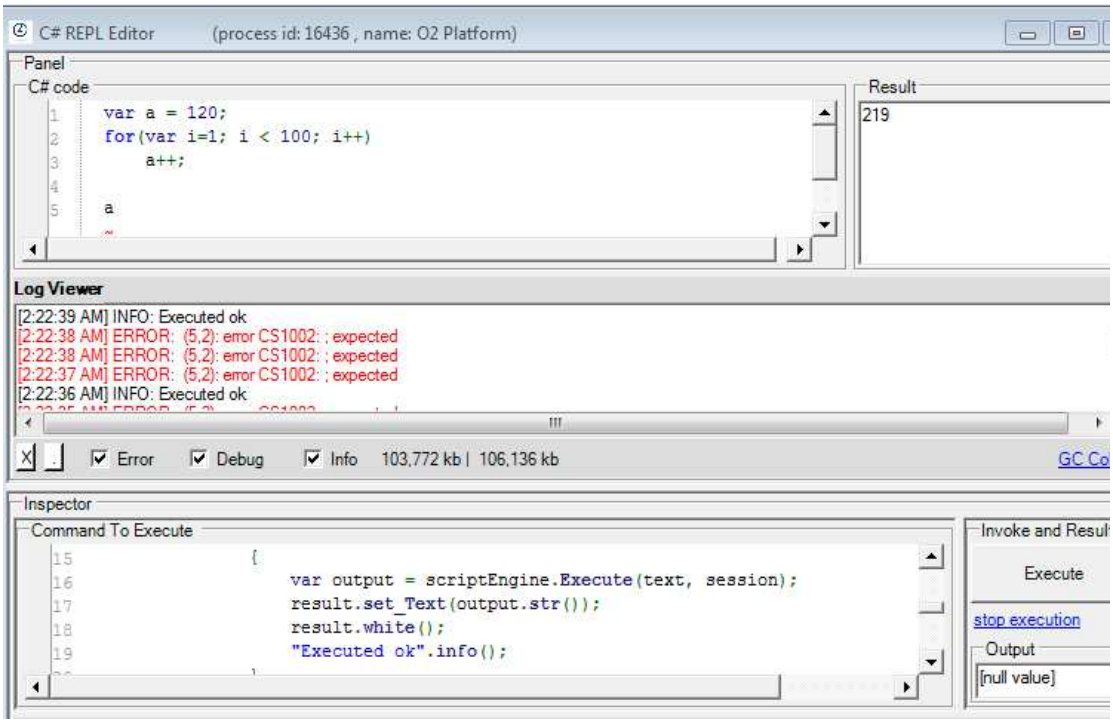
Using a Source Code Editor (instead of a TextBox)

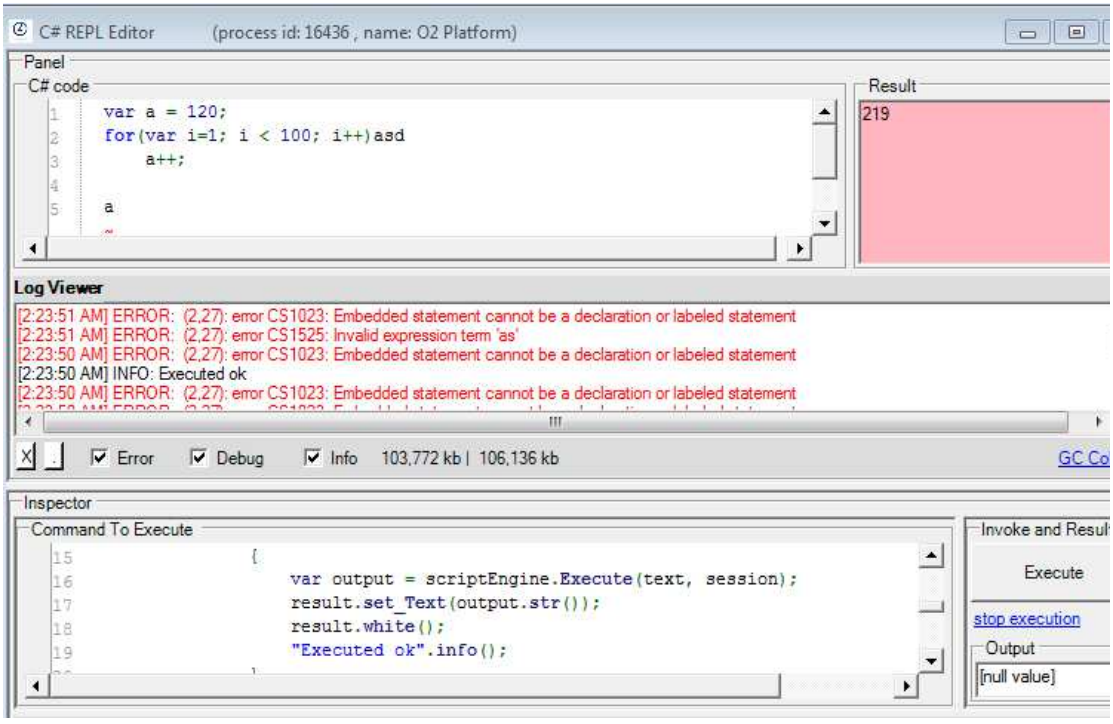
```
var topPanel = panel.clear().add_Panel();
topPanel.insert_LogViewer();

var codeEditor = topPanel.title("C# code").add_SourceCodeViewer();
codeEditor.set_ColorsForCSharp();
```

```
var scriptEngine = new ScriptEngine(new [] { "O2
_FluentSharp_CoreLib.dll".assembly_Location()});
var session = Session.Create(PublicDI.log);

codeEditor.onTextChange(
	(text) =>{
				try
				{
					var result = scriptEngine.Execute(text,
session);
					"Executed ok".info();
				}
				catch(Exception ex)
				{
					ex.log();
				}
			});

//using Roslyn.Scripting.CSharp;
//using Roslyn.Scripting;
//O2Ref:Roslyn\lib\net40\Roslyn.Compilers.dll
//O2Ref:Roslyn\lib\net40\Roslyn.Compilers.CSharp.dll
```

Adding an output window (with color coding depending on compilation status)

```
var topPanel = panel.clear().add_Panel();
topPanel.insert_LogViewer();

var codeEditor = topPanel.title("C# code")
                                        .add_SourceCodeViewer()
                                        .set_ColorsForCSharp();
var result = topPanel.insert_Right(200,"Result")
                                     .add_TextArea();

var scriptEngine = new ScriptEngine(new [] { "O2
_FluentSharp_CoreLib.dll".assembly_Location()});
var session = Session.Create(PublicDI.log);

codeEditor.onTextChange(
	(text) =>{
				try
				{
					var output = scriptEngine.Execute(text,
session);
					result.set_Text(output.str());
					result.white();
					"Executed ok".info();
				}
				catch(Exception ex)
				{
					result.pink();
					ex.log();
				}
			});

//using Roslyn.Scripting.CSharp;
//using Roslyn.Scripting;
//O2Ref:Roslyn\lib\net40\Roslyn.Compilers.dll
//O2Ref:Roslyn\lib\net40\Roslyn.Compilers.CSharp.dll
```

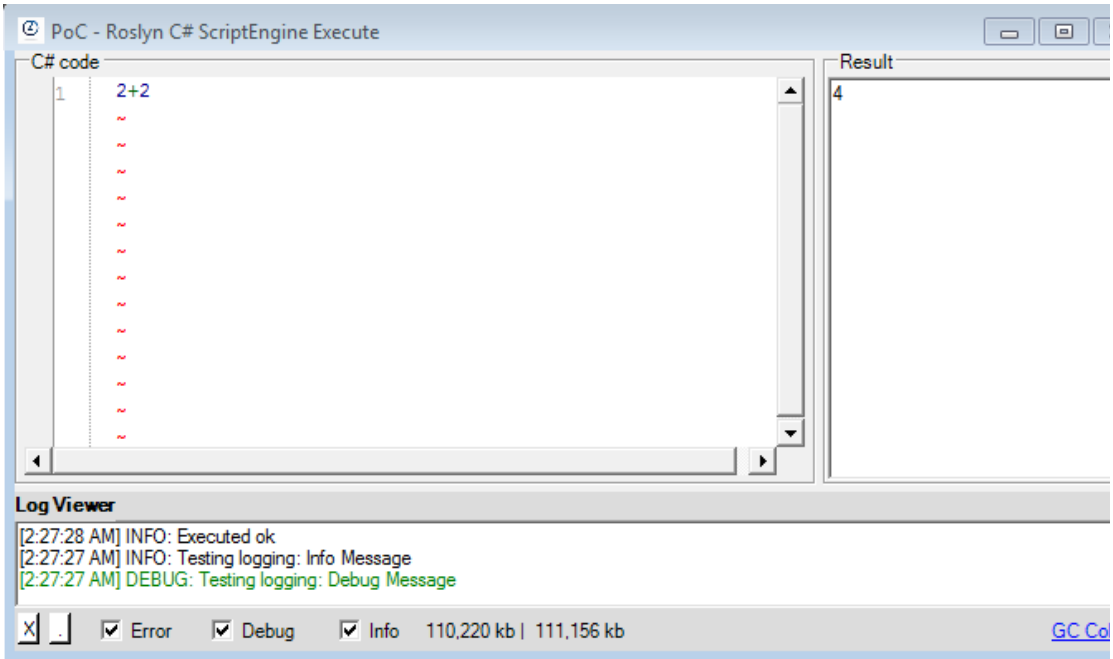


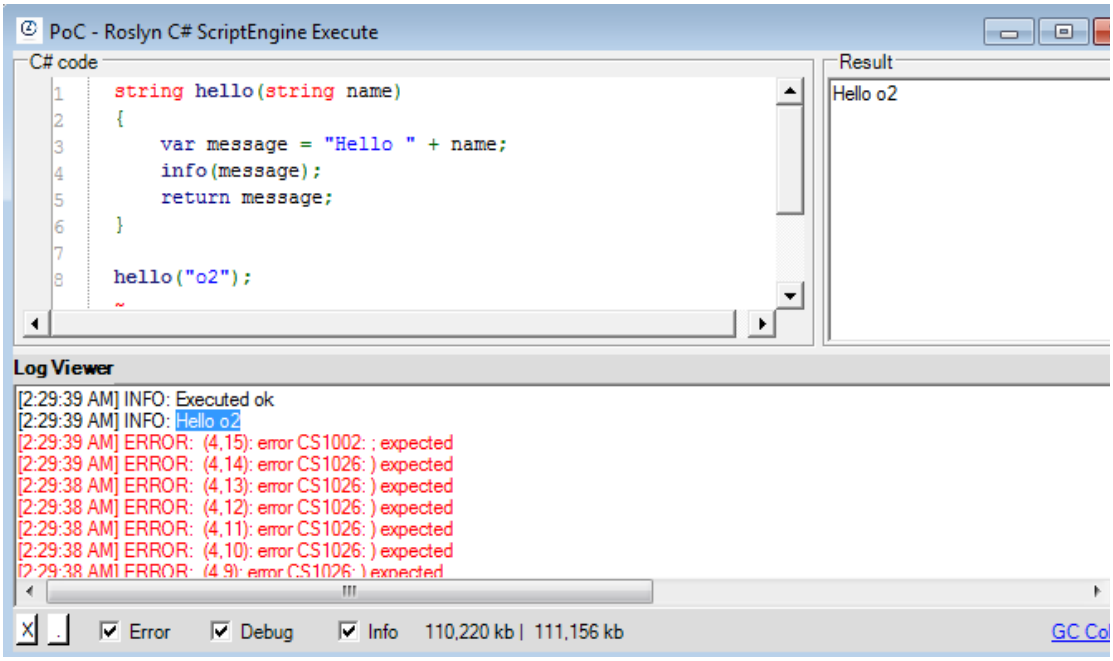


Making it a stand alone tool/script (by modifing the first line)

```
//var topPanel = panel.clear().add_Panel();
var topPanel = "PoC - Roslyn C# ScriptEngine Execute".popupWindow(700,400);
topPanel.insert_LogViewer();

var codeEditor = topPanel.title("C# code")
```

```
                                        .add_SourceCodeViewer()
                                        .set_ColorsForCSharp();
var result = topPanel.insert_Right(200,"Result")
                                    .add_TextArea();

var scriptEngine = new ScriptEngine(new [] { "O2
_FluentSharp_CoreLib.dll".assembly_Location()});
var session = Session.Create(PublicDI.log);

codeEditor.onTextChange(
       (text) =>{
                            try
                            {
                                    var output = scriptEngine.Execute(text,
session);
                                    result.set_Text(output.str());
                                    result.white();
                                    "Executed ok".info();
                            }
                            catch(Exception ex)
                            {
                                    result.pink();
                                    ex.log();
                            }
                    });

codeEditor.set_Text("2+2");
//using Roslyn.Scripting.CSharp;
//using Roslyn.Scripting;
//O2Ref:Roslyn\lib\net40\Roslyn.Compilers.dll
//O2Ref:Roslyn\lib\net40\Roslyn.Compilers.CSharp.dll
```




PoC - Roslyn C# ScriptEngine Execute
C# code
1 string hello(string name)
2 {
3 var message = "Hello " + name;
4 info(message);
5 return message;
6 }
7
8 hello("o2");
Result
Hello o2
Log Viewer
[2:29:39 AM] INFO: Executed ok
[2:29:39 AM] INFO: Hello o2
[2:29:39 AM] ERROR: (4,15): error CS1002: ; expected
[2:29:39 AM] ERROR: (4,14): error CS1026:) expected
[2:29:38 AM] ERROR: (4,13): error CS1026:) expected
[2:29:38 AM] ERROR: (4,12): error CS1026:) expected
[2:29:38 AM] ERROR: (4,11): error CS1026:) expected
[2:29:38 AM] ERROR: (4,10): error CS1026:) expected
Error
Debug
Info
110,220 kb | 111,156 kb

-
-
-
-
-