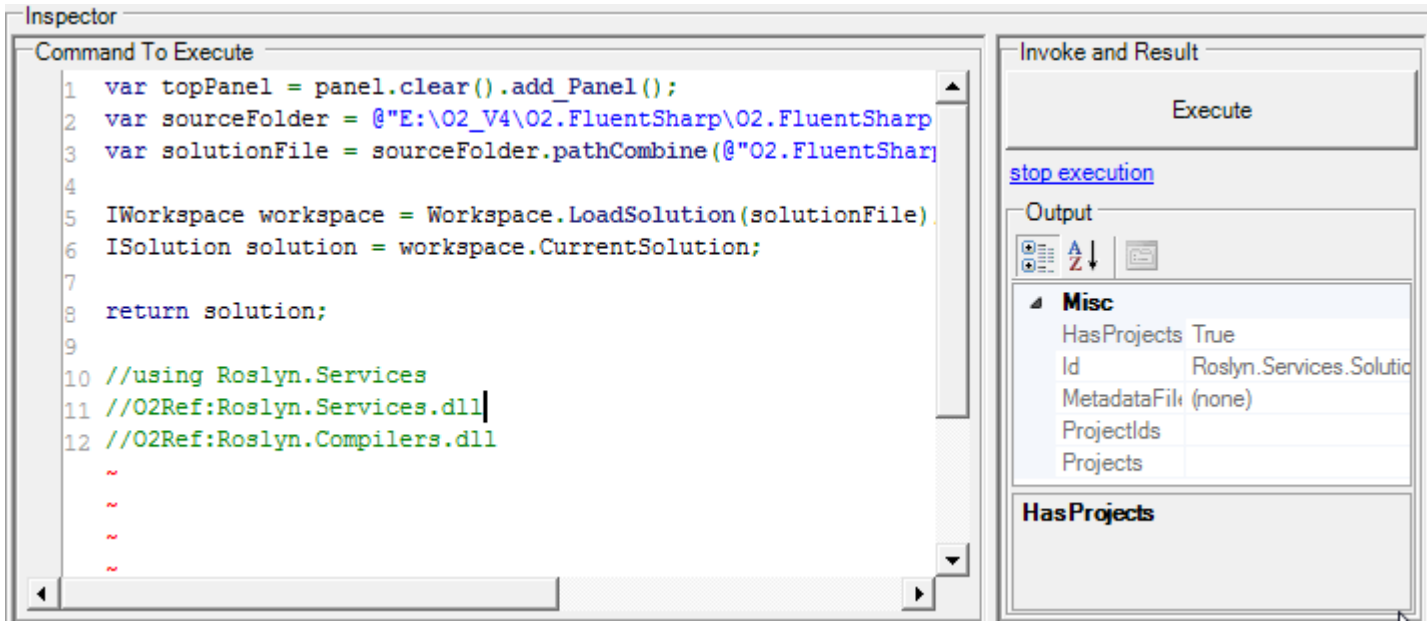# Using Roslyn - Consuming C# Solutions and Projects

## Loading a Solution File

```
var topPanel = panel.clear().add_Panel();
var sourceFolder = @"E:\O2_V4\O2.FluentSharp\O2.FluentSharp.CoreLib";
var solutionFile = sourceFolder.pathCombine(@"O2.FluentSharp.CoreLib.sln");

IWorkspace workspace = Workspace.LoadSolution(solutionFile);
ISolution solution = workspace.CurrentSolution;

return solution;

//using Roslyn.Services
//O2Ref:Roslyn.Services.dll
//O2Ref:Roslyn.Compilers.dll
```
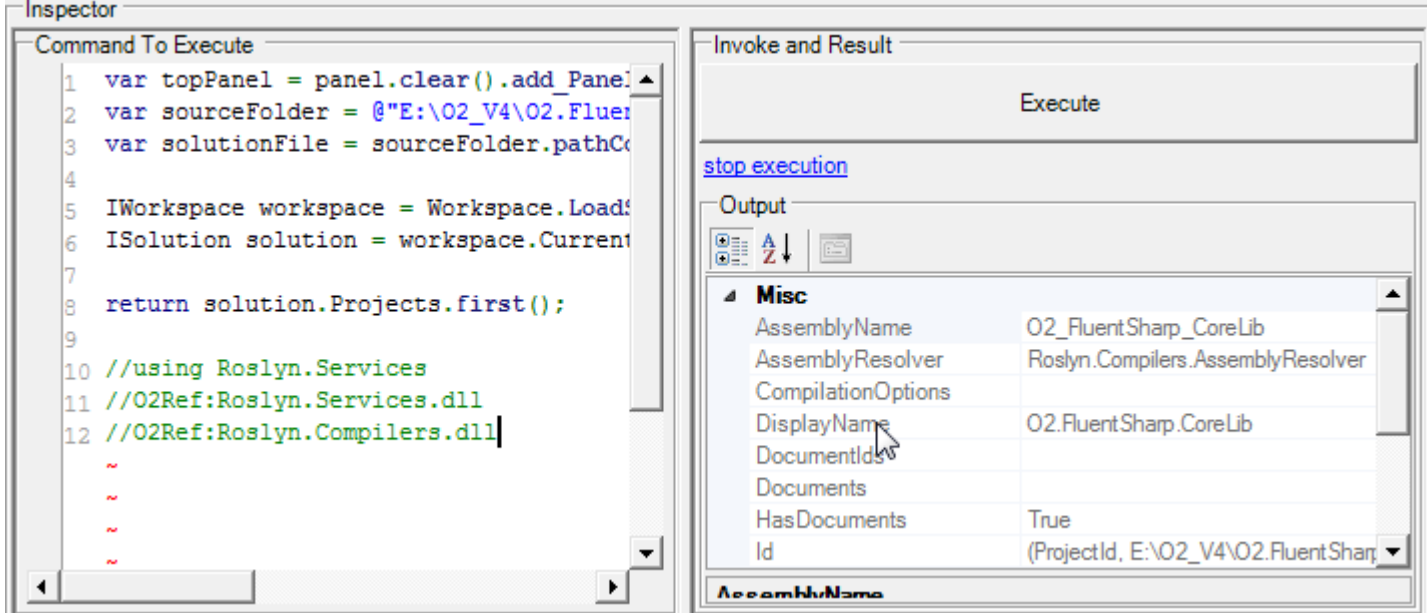


## Getting first Project

```
var topPanel = panel.clear().add_Panel();
var sourceFolder = @"E:\O2_V4\O2.FluentSharp\O2.FluentSharp.CoreLib";
var solutionFile = sourceFolder.pathCombine(@"O2.FluentSharp.CoreLib.sln");

IWorkspace workspace = Workspace.LoadSolution(solutionFile);
ISolution solution = workspace.CurrentSolution;

return solution.Projects.first();

//using Roslyn.Services
//O2Ref:Roslyn.Services.dll
//O2Ref:Roslyn.Compilers.dll
```
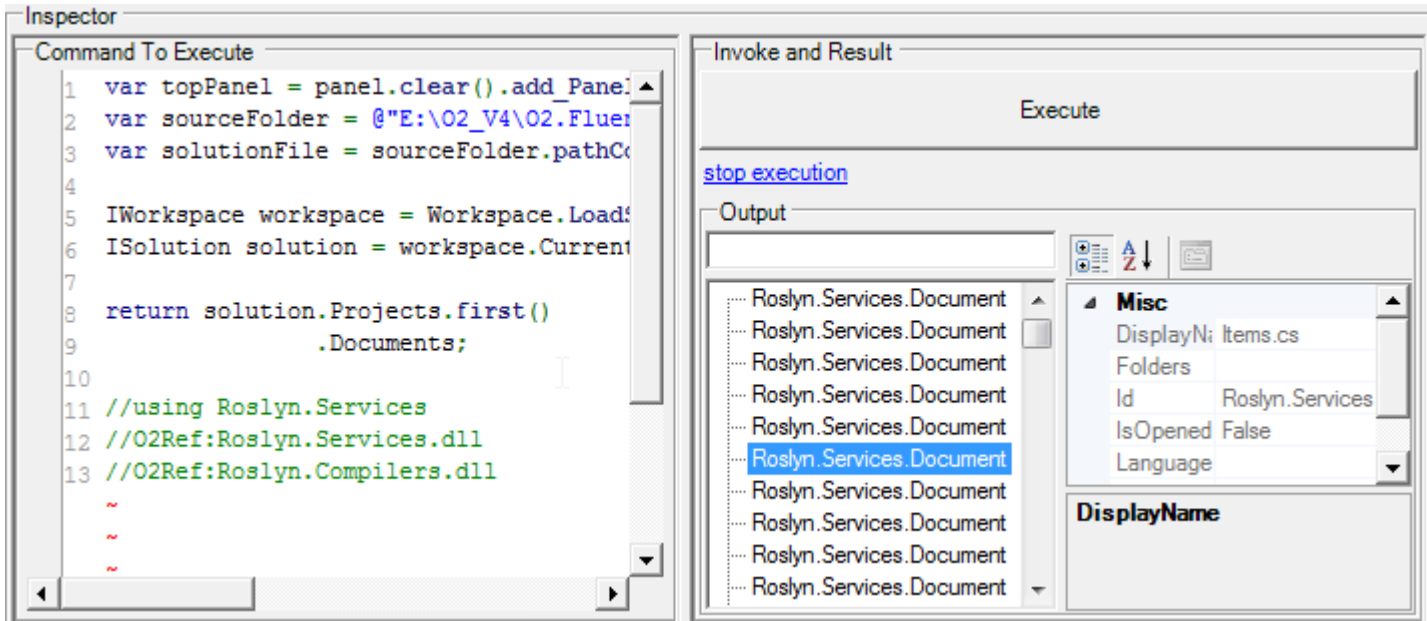
## Getting Files (from first project)

```
IWorkspace workspace = Workspace.LoadSolution(solutionFile);
ISolution solution = workspace.CurrentSolution;

return solution.Projects.first()
                        .Documents;
```



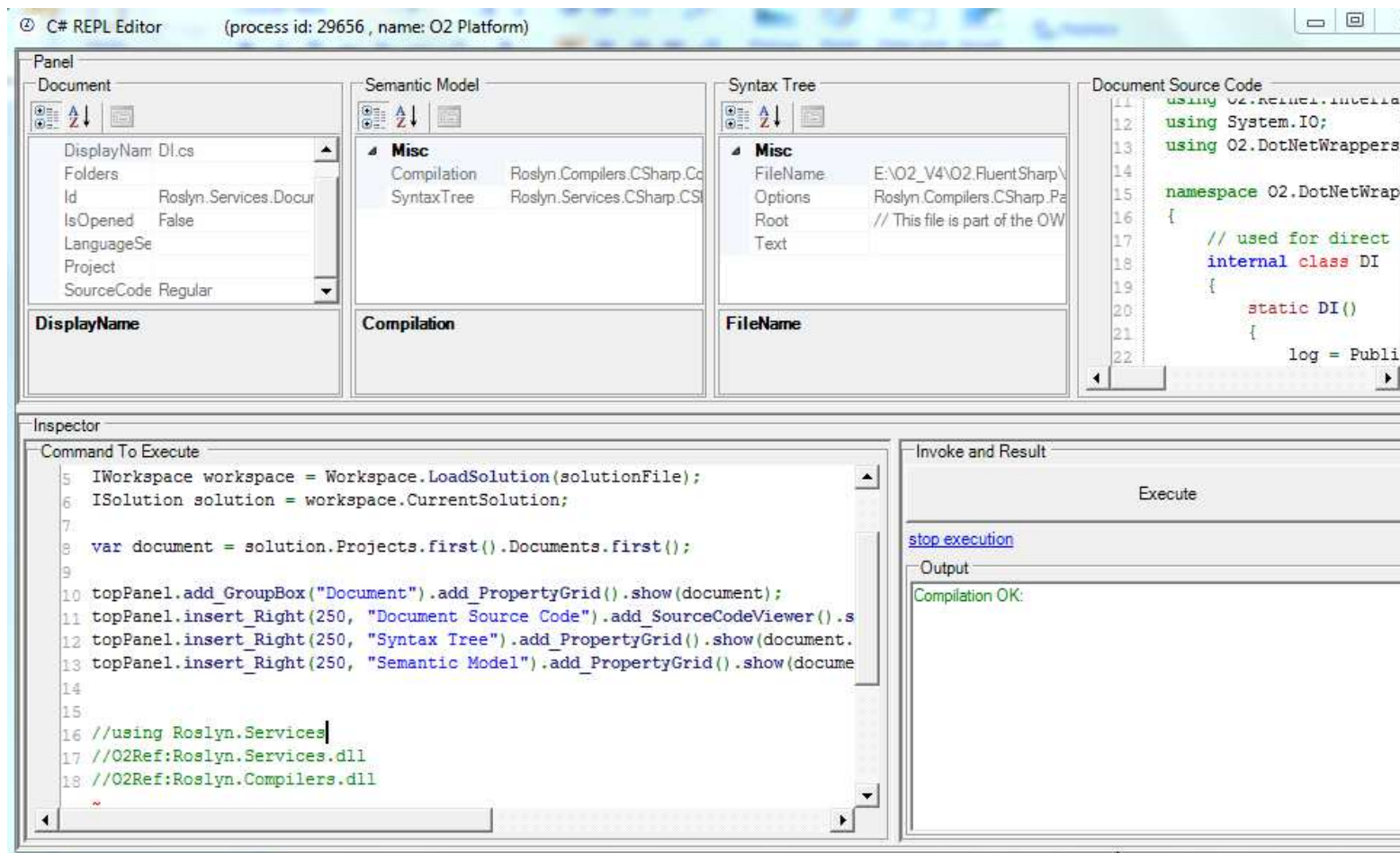## For first Document see: its properties, source code, semantic model and syntax tree

```
var topPanel = panel.clear().add_Panel();
var sourceFolder = @"E:\O2_V4\O2.FluentSharp\O2.FluentSharp.CoreLib";
var solutionFile = sourceFolder.pathCombine(@"O2.FluentSharp.CoreLib.sln");

IWorkspace workspace = Workspace.LoadSolution(solutionFile);
ISolution solution = workspace.CurrentSolution;

var document = solution.Projects.first().Documents.first();

topPanel.add_GroupBox("Document").add_PropertyGrid().show(document);
topPanel.insert_Right(250, "Document Source Code").add_SourceCodeViewer().set_Text(document.GetText().str
());
topPanel.insert_Right(250, "Syntax Tree").add_PropertyGrid().show(document.GetSyntaxTree());
topPanel.insert_Right(250, "Semantic Model").add_PropertyGrid().show(document.GetSemanticModel());
```

```
//using Roslyn.Services
//O2Ref:Roslyn.Services.dll
//O2Ref:Roslyn.Compilers.dll
```



**Opening up a Solution, Compiling it and seeing the errors created:**

```
var topPanel = panel.clear().add_Panel();
var actionsPanel = topPanel.insert_Above(40,"Actions Panel");
var codeViewer = topPanel.add_SourceCodeViewer();

var sourceFolder = @"E:\O2_V4\O2.FluentSharp\O2.FluentSharp.CoreLib";
var solutionFile = sourceFolder.pathCombine(@"O2.FluentSharp.CoreLib.sln");
var errorsList = topPanel.insert_Right().add_TreeView();

Action compile =
        ()=>{
                    IWorkspace workspace = Workspace.LoadSolution(solutionFile);
                    "Compling Project".info();
                    var o2Timer = new O2Timer("Solution Compilation").start();
                    ISolution solution = workspace.CurrentSolution;
                    var project = solution.Projects.first();
                    var cancellationToken = new CancellationToken(false);
                    var compilation = project.GetCompilation(cancellationToken);
                    o2Timer.stop();
                    errorsList.clear()
                               .add_Nodes(compilation.errors())
                               .selectFirst();
            };

errorsList.afterSelect<Diagnostic>(
        (diagnostic)=>{
                            diagnostic.details();
                    });

actionsPanel.add_Link("Compile", ()=> compile());
compile();

//using Roslyn.Compilers.CSharp
//O2File:API_Roslyn.cs
//using System.Threading
```
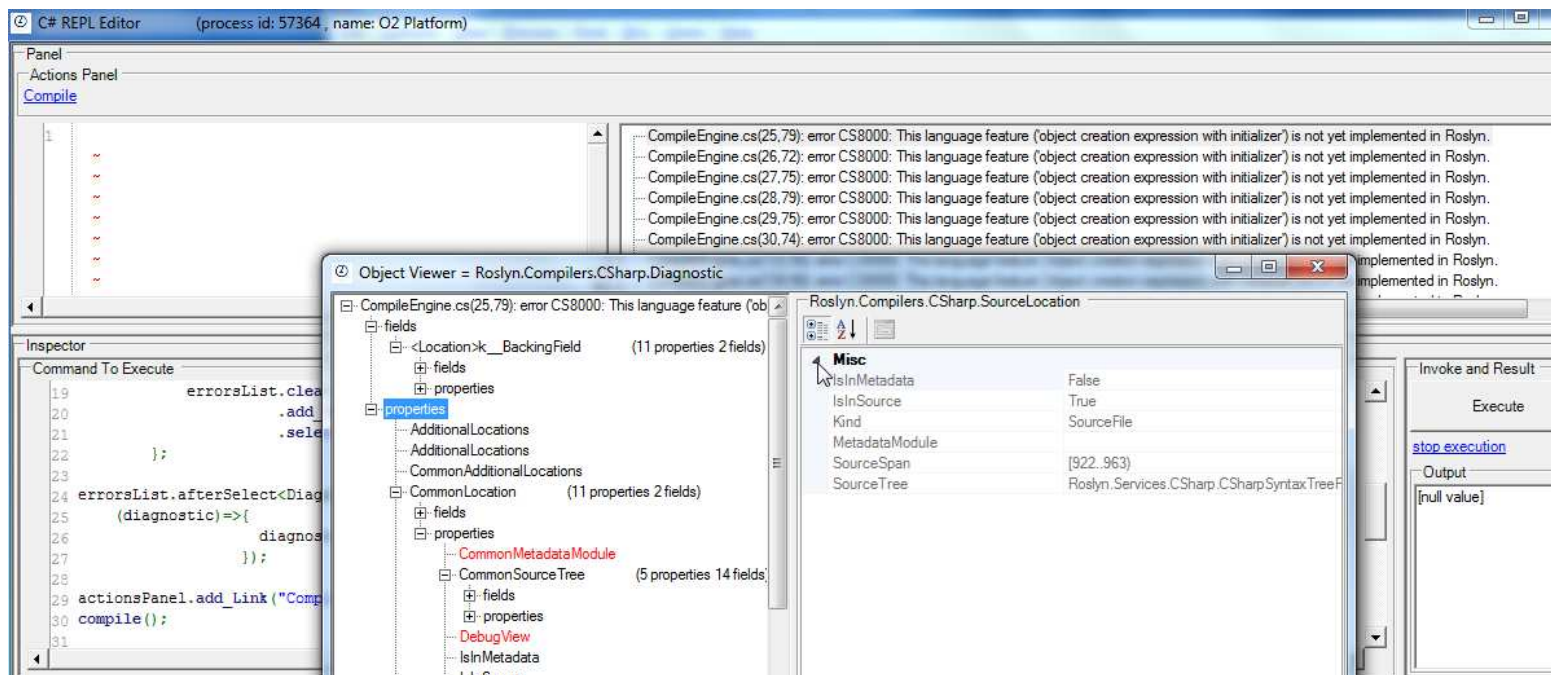
```
//using Roslyn.Services
//O2Ref:Roslyn\lib\net40\Roslyn.Services.dll
//O2Ref:Roslyn\lib\net40\Roslyn.Compilers.dll
//O2Ref:Roslyn\lib\net40\Roslyn.Compilers.CSharp.dll
```



**Show LineSpan on property grid (good to see what is inside that object)**

```
var topPanel = panel.clear().add_Panel();
var actionsPanel = topPanel.insert_Above(40,"Actions Panel");
//var codeViewer = topPanel.add_SourceCodeViewer();
var propertyGrid = topPanel.add_PropertyGrid();

var sourceFolder = @"E:\O2_V4\O2.FluentSharp\O2.FluentSharp.CoreLib";
var solutionFile = sourceFolder.pathCombine(@"O2.FluentSharp.CoreLib.sln");
var errorsList = topPanel.insert_Right().add_TreeView();

Action compile =
        ()=>{
                    IWorkspace workspace = Workspace.LoadSolution(solutionFile);
                    "Compling Project".info();
                    var o2Timer = new O2Timer("Solution Compilation").start();
                    ISolution solution = workspace.CurrentSolution;
                    var project = solution.Projects.first();
                    var cancellationToken = new CancellationToken(false);
                    var compilation = project.GetCompilation(cancellationToken);
                    o2Timer.stop();
                    errorsList.clear()
                            .add_Nodes(compilation.errors())
                            .selectFirst();
            };

errorsList.afterSelect<Diagnostic>(
        (diagnostic)=>{
                            var lineSpan = diagnostic.Location.GetLineSpan(true);
                            propertyGrid.show(lineSpan);

                    });

actionsPanel.add_Link("Compile", ()=> compile());
compile();

//using Roslyn.Compilers.CSharp
//O2File:API_Roslyn.cs
//using System.Threading
//using Roslyn.Services
//O2Ref:Roslyn\lib\net40\Roslyn.Services.dll
//O2Ref:Roslyn\lib\net40\Roslyn.Compilers.dll
//O2Ref:Roslyn\lib\net40\Roslyn.Compilers.CSharp.dll
```

4
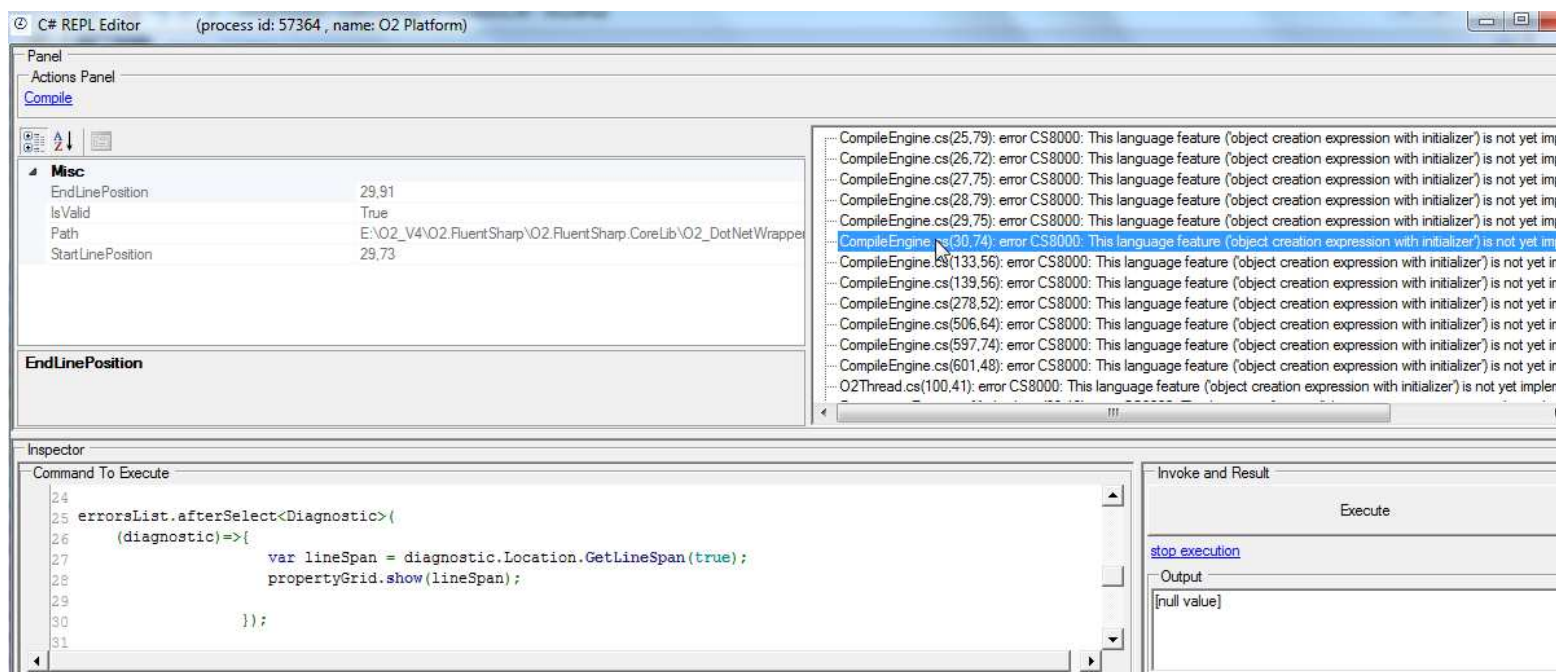
**Opening up source code file on node click:**

```csharp
var topPanel = panel.clear().add_Panel();
var actionsPanel = topPanel.insert_Above(40,"Actions Panel");
var codeViewer = topPanel.add_SourceCodeViewer();

var sourceFolder = @"E:\O2_V4\O2.FluentSharp\O2.FluentSharp.CoreLib";
var solutionFile = sourceFolder.pathCombine(@"O2.FluentSharp.CoreLib.sln");
var errorsList = topPanel.insert_Right().add_TreeView();


Action compile =
        ()=>{
                    IWorkspace workspace = Workspace.LoadSolution(solutionFile);

                    "Compling Project".info();
                    var o2Timer = new O2Timer("Solution Compilation").start();
                    ISolution solution = workspace.CurrentSolution;
                    var project = solution.Projects.first();
                    var cancellationToken = new CancellationToken(false);
                    var compilation = project.GetCompilation(cancellationToken);
                    o2Timer.stop();
                    var errors = compilation.errors();
                    "There are {0} errors".info(errors.size());
                    errorsList.clear()
                                .add_Nodes(errors).
                              selectFirst();
            };

errorsList.afterSelect<Diagnostic>(
        (diagnostic)=>{
                            var lineSpan = diagnostic.Location.GetLineSpan(true);
                            codeViewer.open(lineSpan.Path);
                            codeViewer.gotoLine(lineSpan.StartLinePosition.Line + 1);
                            errorsList.focus();
                        });

actionsPanel.add_Link("Compile", ()=> compile());
compile();

//using Roslyn.Compilers.CSharp
//O2File:API_Roslyn.cs
//using System.Threading
//using Roslyn.Services
//O2Ref:Roslyn\lib\net40\Roslyn.Services.dll
//O2Ref:Roslyn\lib\net40\Roslyn.Compilers.dll
//O2Ref:Roslyn\lib\net40\Roslyn.Compilers.CSharp.dll
```
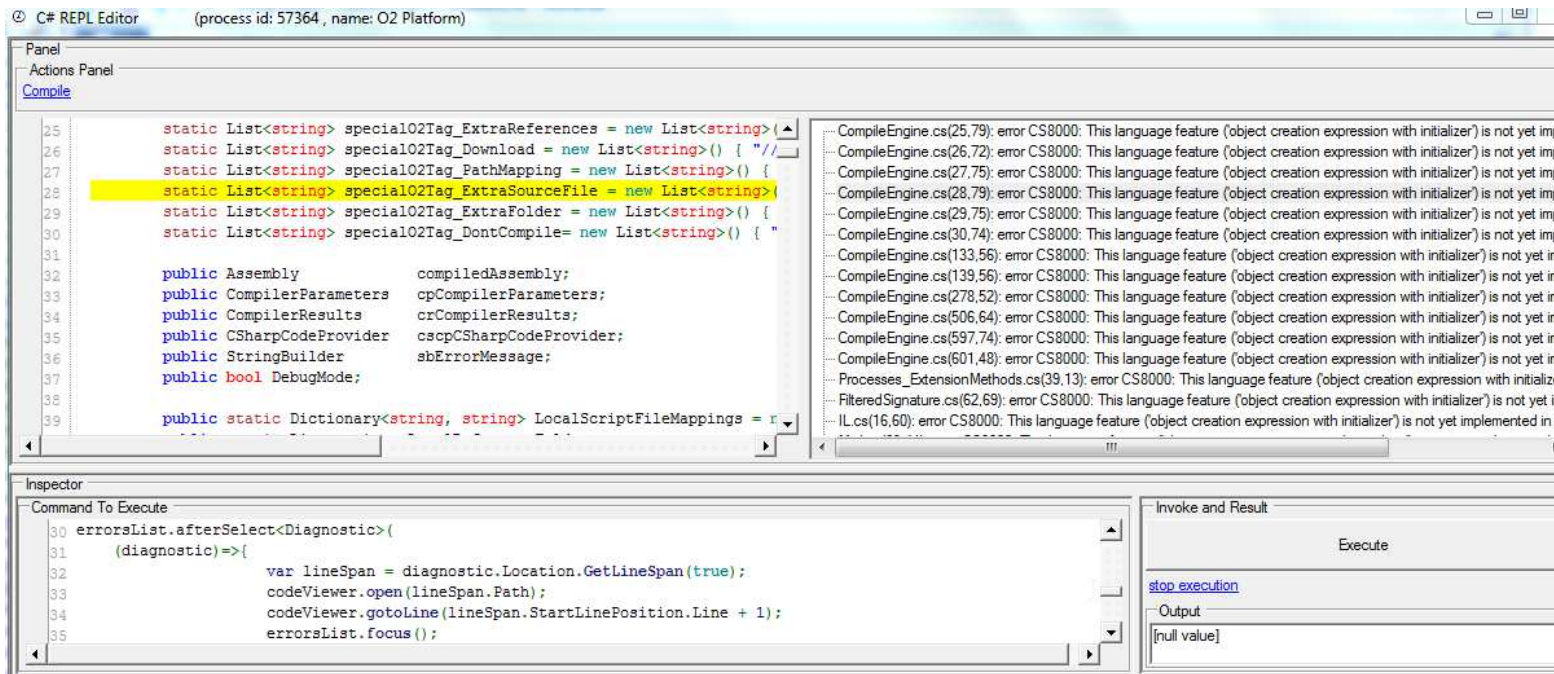
5

**Adding feature to load solution on Drag'n'Drop (into to the top Panel)**

```
actionsPanel.onDrop(
        (file)=>{
                        "File droppped: {0}".info();
                        if (file.extension(".sln"))
                        {
                                solutionFile = file;
                                compile();
                        }
                });
```

... here is the GUI after dropping the SuperSecureBank.sln file:
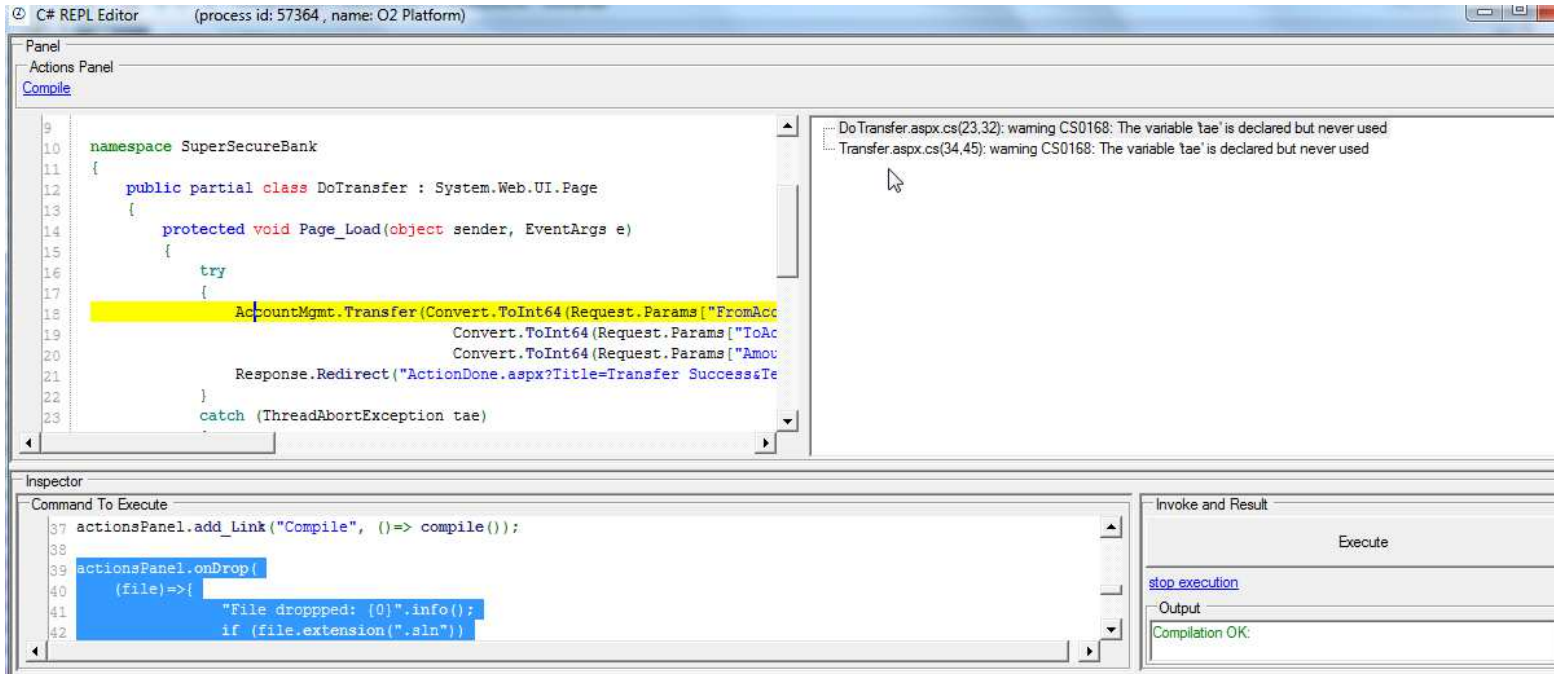


# O2 tool: "Util - Roslyn - Compile Solution and view errors"

Here is a full tool (written as a script) which provides a nice GUI to view solutions and try the Roslyn Compiler on Solution files

**Tool with the the O2.FluentSharp.CoreLib.sln loaded** (which uses a couple C# capabilities not supported by the Roslyn compiler)
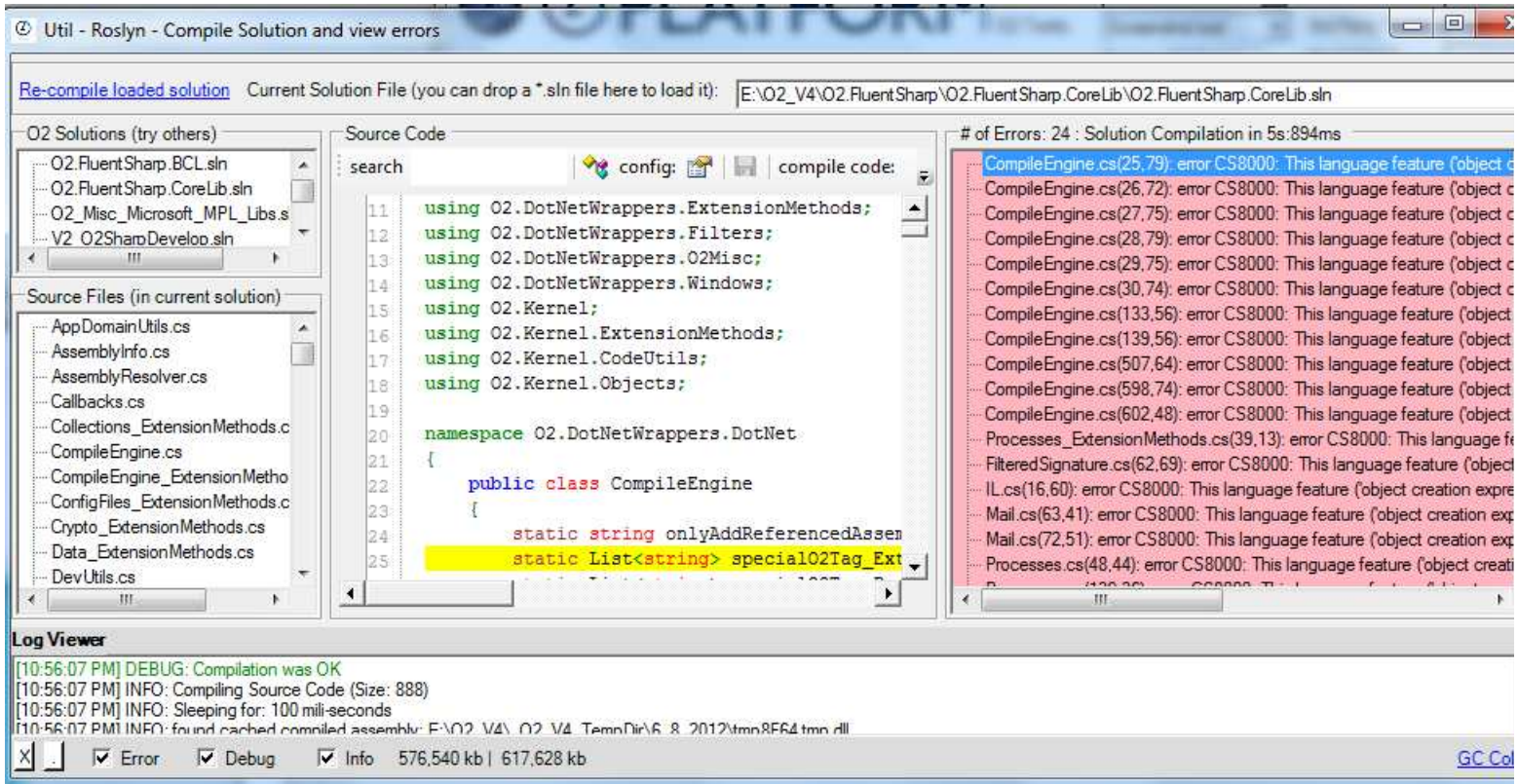


**Tool with the the SuperSecureBank.sln solution loaded** (which is compiled ok using the Roslyn compiler)
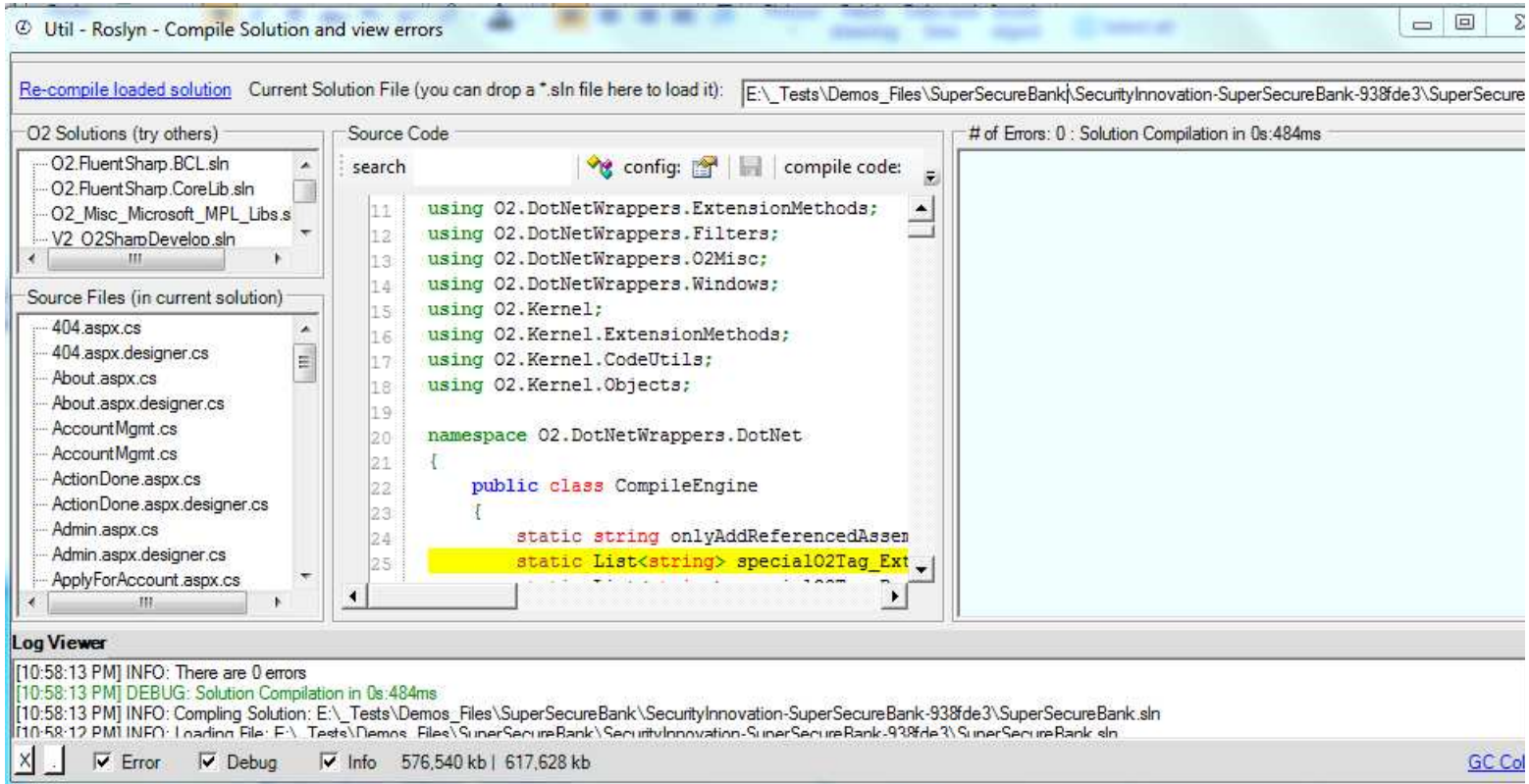


**Source Code**

This is the code that creates the Gui (shown above):

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

```
var topPanel = "Util - Roslyn - Compile Solution and view errors".popupWindow(1000,500)
                           .insert_LogViewer();
var actionsPanel = topPanel.insert_Above(40,"");
var codeViewer = topPanel.title("Source Code").add_SourceCodeEditor();
var sourceCodeFiles = topPanel.insert_Left(200, "Source Files (in current solution)").add_TreeView().sort
();
var errorsList = topPanel.insert_Right("Errors").add_TreeView();
var o2Solutions = sourceCodeFiles.parent().parent().insert_Above(100,"O2 Solutions (try
others)").add_TreeView();

var solutionFile = "";
sourceCodeFiles.afterSelect<IDocument>(
        (document)=> codeViewer.open(document.FilePath));

Action compile =
        ()=>{
                O2Thread.mtaThread(
                        ()=>{
                                var o2Timer = new O2Timer("Solution Compilation").start();
                                IWorkspace workspace = Workspace.LoadSolution(solutionFile);

                                "Compling Solution: {0}".info(solutionFile);
                                ISolution solution = workspace.CurrentSolution;
                                var project = solution.Projects.first();
                                var cancellationToken = new CancellationToken(false);
                                var compilation = project.GetCompilation(cancellationToken);

                                var errors = compilation.errors();
                                var time = o2Timer.stop();
                                "There are {0} errors".info(errors.size());
                                errorsList.clear()
                                        .add_Nodes(errors)
                                        .backColor(errors.size() > 0 ? Color.LightPink :
Color.Azure)

                                        .selectFirst()
                                        .focus()
                                        .parent<GroupBox>().set_Text("# of Errors: {0} :
{1}".format(errors.size(), time));
                                        //var assembly = compilation.create_Assembly();
                        });
        };

Action<string> loadSolution =
        (_solutionFile)=>{
                                "Loading File: {0}".info(_solutionFile);
                                if (_solutionFile.extension(".sln").isFalse())
                                        return;
                                        solutionFile = _solutionFile;
                                sourceCodeFiles.clear();
                                errorsList.clear().add_Node("Compiling solution....");
                                codeViewer.open("");
                                IWorkspace workspace = Workspace.LoadSolution(solutionFile);
                                var files = (from project in workspace.CurrentSolution.Projects
                                                from document in project.Documents
                                                select document).toList();
                                sourceCodeFiles.add_Nodes(files, (file)=>file.Name);
                                actionsPanel.control<TextBox>(true).set_Text(solutionFile);
                                compile();
                        };

errorsList.afterSelect<Diagnostic>(
        (diagnostic)=>{
                                var lineSpan = diagnostic.Location.GetLineSpan(true);
                                codeViewer.open(lineSpan.Path);
                                codeViewer.gotoLine(lineSpan.StartLinePosition.Line + 1);

                });

actionsPanel.add_Link("Re-compile loaded solution", ()=> compile())
        .append_Label("Current Solution File (you can drop a *.sln file here to load it):").autoSize()
        .append_TextBox("").align_Right();

actionsPanel.onDrop(loadSolution);
sourceCodeFiles.onDrop(loadSolution);
errorsList.onDrop(loadSolution);

o2Solutions.afterSelect<string>(loadSolution);

var sourceFolder = PublicDI.config.CurrentExecutableDirectory.pathCombine(@"..\..\O2.FluentSharp
\O2.FluentSharp.CoreLib");
```

```
loadSolution(sourceFolder.pathCombine(@"O2.FluentSharp.CoreLib.sln"));

o2Solutions.add_Files(PublicDI.config.CurrentExecutableDirectory.pathCombine(@"..\..\"), "*.sln");

//using Roslyn.Compilers.CSharp
//O2File:API_Roslyn.cs
//using System.Threading
//using Roslyn.Services
//O2Ref:Roslyn\lib\net40\Roslyn.Services.dll
//O2Ref:Roslyn\lib\net40\Roslyn.Compilers.dll
//O2Ref:Roslyn\lib\net40\Roslyn.Compilers.CSharp.dll
```

.

```
loadSolution(sourceFolder.pathCombine(@"O2.FluentSharp.CoreLib.sln"));

o2Solutions.add_Files(PublicDI.config.CurrentExecutableDirectory.pathCombine(@"..\..\"), "*.sln");

//using Roslyn.Compilers.CSharp
//O2File:API_Roslyn.cs
//using System.Threading
//using Roslyn.Services
//O2Ref:Roslyn\lib\net40\Roslyn.Services.dll
//O2Ref:Roslyn\lib\net40\Roslyn.Compilers.dll
//O2Ref:Roslyn\lib\net40\Roslyn.Compilers.CSharp.dll
```