

```
1 #include "CalculatorForm.h"
2
3 using namespace System;
4 using namespace System::ComponentModel::Composition;
5 using namespace System::ComponentModel::Composition::Hosting;
6
7 namespace SimpleCalculator {
8
9     CalculatorForm::CalculatorForm() {
10         InitializeComponent();
11
12         // Import/Export カタログをつくる
13         auto catalog = gcnew AggregateCatalog();
14         // まずは自分自身のアセンブリから
15         catalog->Catalogs->Add(gcnew AssemblyCatalog(CalculatorForm::typeid->Assembly));
16
17         // そして自分自身の置かれたディレクトリから見つけてくる
18         String^ myLocation = System::IO::Path::GetDirectoryName(
19             System::Reflection::MethodInfo::GetCurrentMethod()
20             ->DeclaringType->Assembly->Location);
21         catalog->Catalogs->Add(gcnew DirectoryCatalog(myLocation));
22
23         // カタログから作られたコンテナを基にImport/Exportを結びつける
24         AttributedModelServices::ComposeParts(gcnew CompositionContainer(catalog), this);
25
26         // 得られた演算子(Symbol)をComboBoxに追加する
27         for each (String^ symbol in calculator->Symbols()) {
28             cbxOpr->Items->Add(symbol);
29         }
30         cbxOpr->SelectedIndex = 0;
31     }
32
33     CalculatorForm::~CalculatorForm() {
34         if (components) {
35             delete components;
36         }
37     }
38
39     System::Void CalculatorForm::btnExec_Click(System::Object^ sender, System::EventArgs^ e) {
40         Int32 left;
41         Int32 right;
42         // フォームから 左辺/右辺/演算子を取り出し、計算して結果を表示する
43         if (Int32::TryParse(tbxLeft->Text, left) && Int32::TryParse(tbxRight->Text, right)) {
44             try {
45                 int result = calculator->Calculate(left, cbxOpr->SelectedItem->ToString(), right);
46                 lblResult->Text = result.ToString();
47             } catch (Exception^ ) {
48                 lblResult->Text = L"error";
49             }
50         } else {
51             lblResult->Text = L"?";
52         }
53     }
54 }
55
56 }
```