T4 code scaffolding in Apincore

Note

This feature can save a lot of time to build required plumbing code for added entity class but still some code customization could be required in some special cases.

About

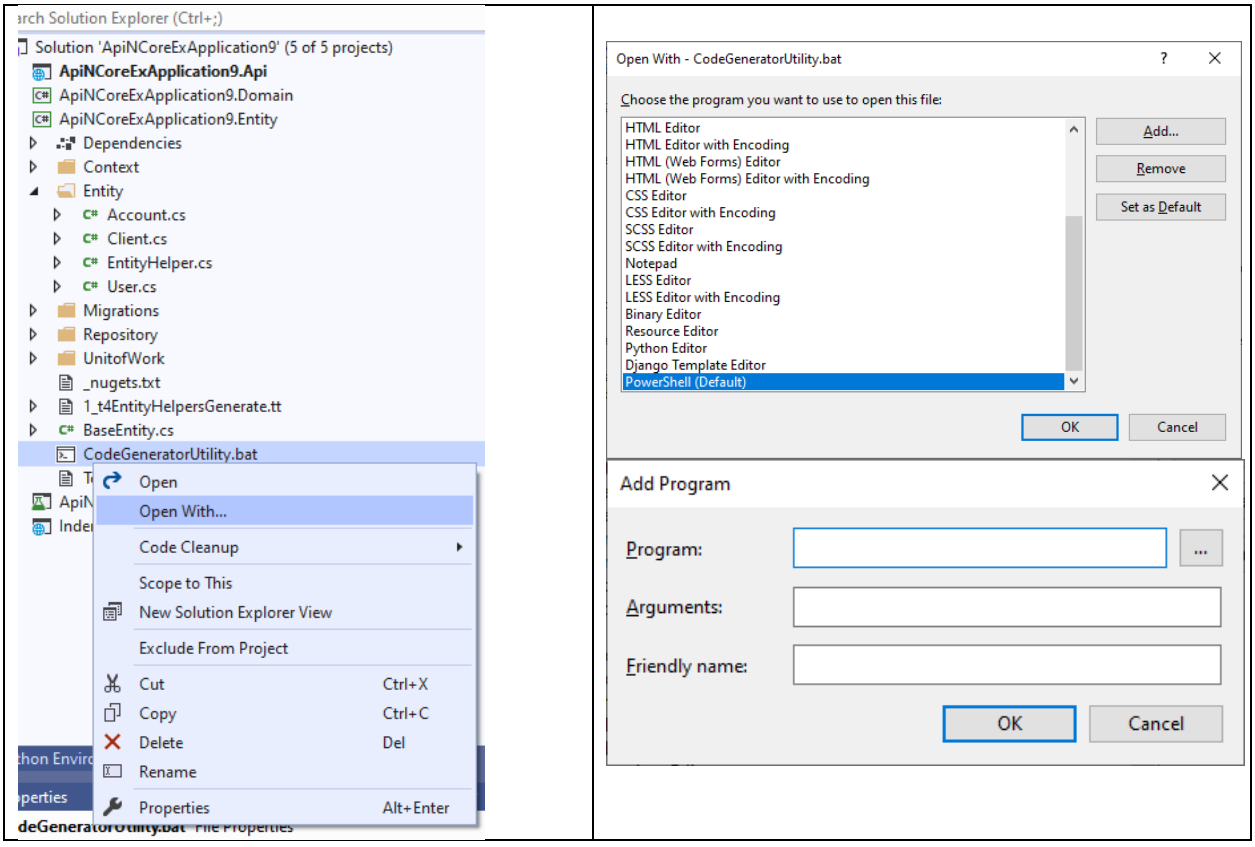
This batch is used to run all T4 scripts (*.tt) in specific order in created Apincore solution. Those scripts can be executed manually as well but following specific order (*.tt scripts prefixed with execution order number).

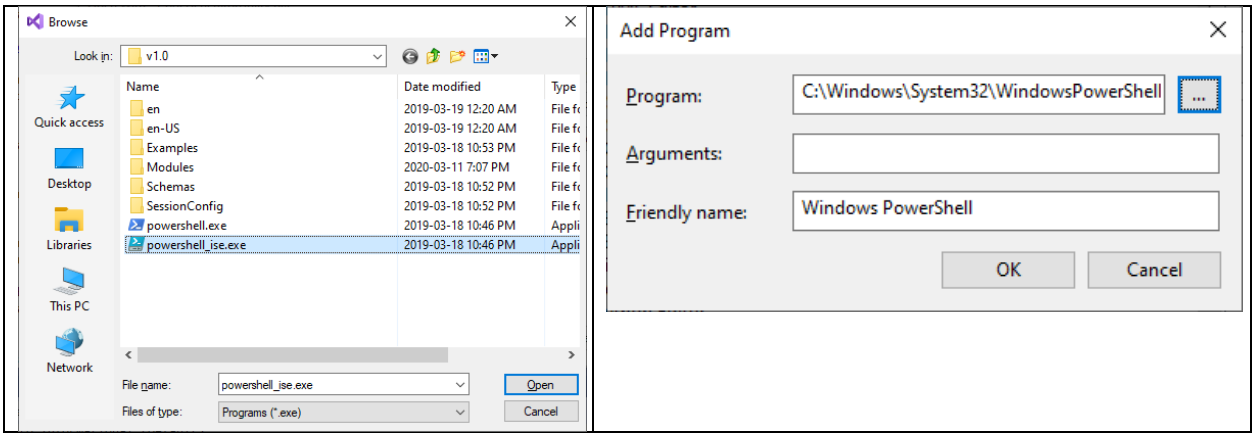
T4 code scaffolding in .NET Core API project is driven by added Entity class. The code scaffolding impacts all projects and builds all required plumbing code for created entity class (es). For added simple entity classes without mandatory table fields after database migration, generated units test should be executed with success right after *CodeGeneratorUtility.bat* executed.

Executing CodeGeneratorUtility.bat

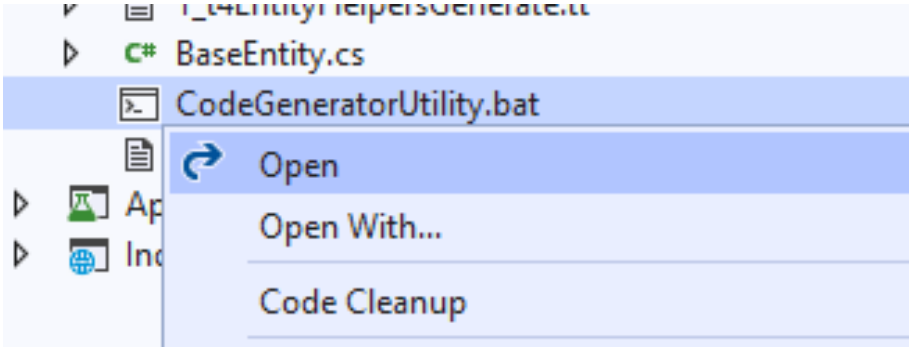
Before this utility is used it has to set an application to use to execute CodeGeneratorUtility.bat file.

Add *PowerShell* executable if it is in list of programs. If not in the list use *Add...* button to add it and click *Set as Default* button.

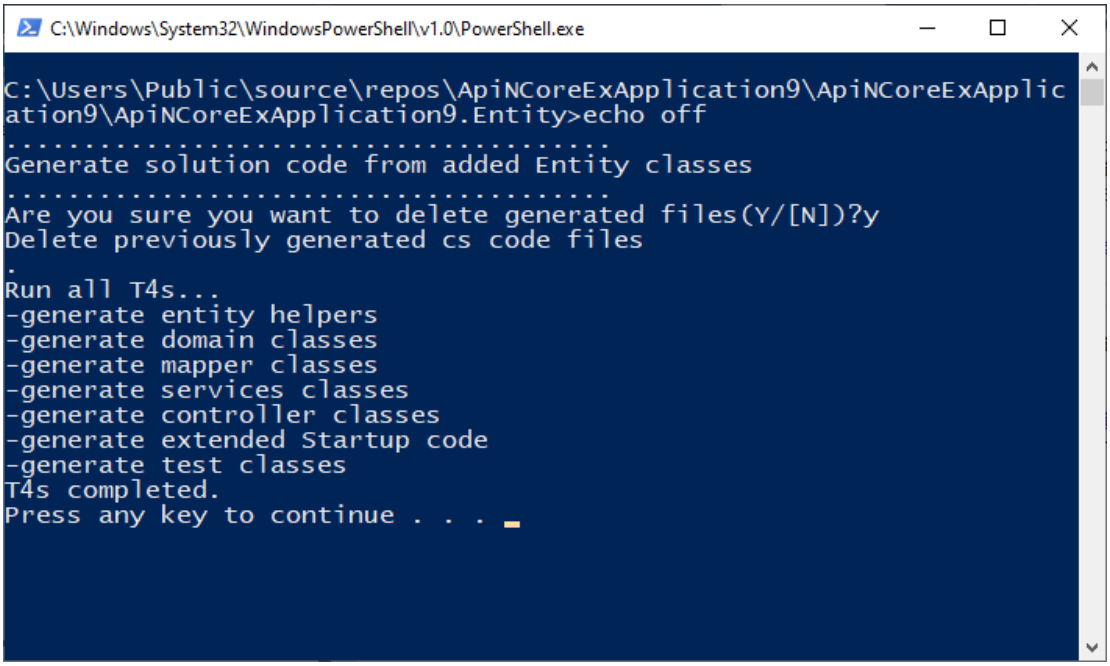




After setting up an application to run CodeGeneratorUtility.bat it can be used only Open menu item to run all T4 scripts.



..and all *.tt scripts are executed in required order after confirmation that previously generated classes will be deleted:



Expand *.tt script modules to find generated classes for added Entity class (es). Note that for existing classes are use partial classes like this example in generated *1_t4EntityHelpersGenerate.cs*:

```
public partial class ApiNCoreExApplication9Context : DbContext
{
            public DbSet<Client> Clients { get; set; }

            /// Add new entities concurrency declarations (Fluent API)
            partial void SetAdditionalConcurency(ModelBuilder modelBuilder)
            {
                    modelBuilder.Entity<Client>().Property(a => a.RowVersion).IsRowVersion();
            }
 }
```

Troubleshooting for known Visual Solution Community issue running T4 script

If there is an issue with running T4 scripts when only Visual Studio 2019 Community installed (most likely this error: "could not find file EnvDTE…") it could be related to registering EnvDTE assemblies and its dependencies.

So to fix that EnvDTE issue use these steps:

1) Delete this line
<#@ assembly name="Microsoft.VisualStudio.Shell" #>
from ApiNCoreExApplication1.Entity -> TemplateCommon.tt file

2) Run these commands from **Visual Studio 2019 Developer Command Prompt as Administrator**:

```
gacutil -i "C:\Program Files (x86)\Microsoft Visual Studio\2019\Community\Common7\IDE\PublicAssemblies\envdte.dll"

gacutil -i "C:\Program Files (x86)\Microsoft Visual Studio\2019\Community\Common7\IDE\PublicAssemblies\envdte80.dll"

gacutil -i "C:\Program Files (x86)\Microsoft Visual Studio\2019\Community\Common7\IDE\PublicAssemblies\Microsoft.VisualStudio.OLE.Interop.dll"

gacutil -i "C:\Program Files (x86)\Microsoft Visual Studio\2019\Community\Common7\IDE\PublicAssemblies\Microsoft.VisualStudio.Shell.Interop.dll"

gacutil -i "C:\Program Files (x86)\Microsoft Visual Studio\2019\Community\Common7\IDE\PublicAssemblies\Microsoft.VisualStudio.Shell.Interop.8.0.dll
```

Apincore team
March 2020