

T4 code scaffolding

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 generated REST API solution. Those scripts can be executed manually as well but following specific order (*.tt scripts prefixed with execution order number).

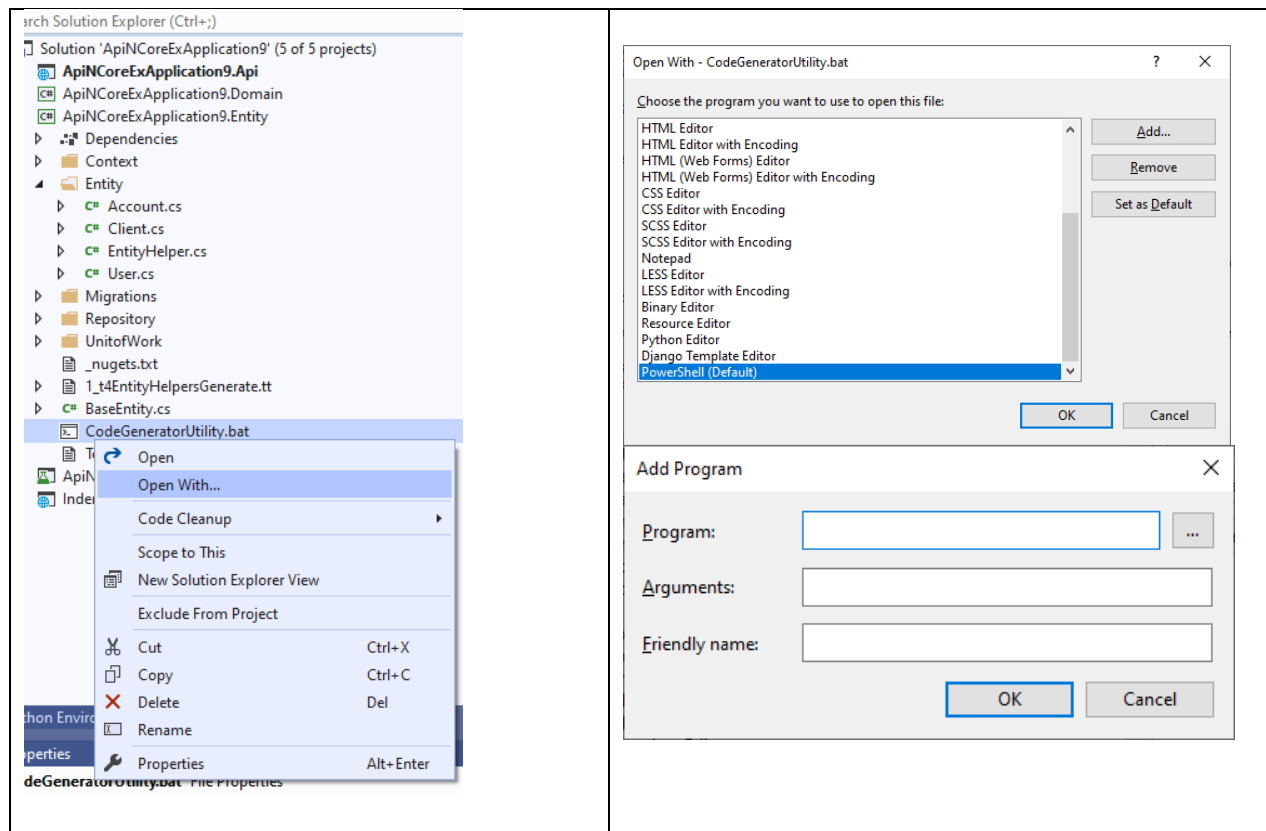
Note that API must be run (Run solution) at least once before T4 to initiate default db deployment.

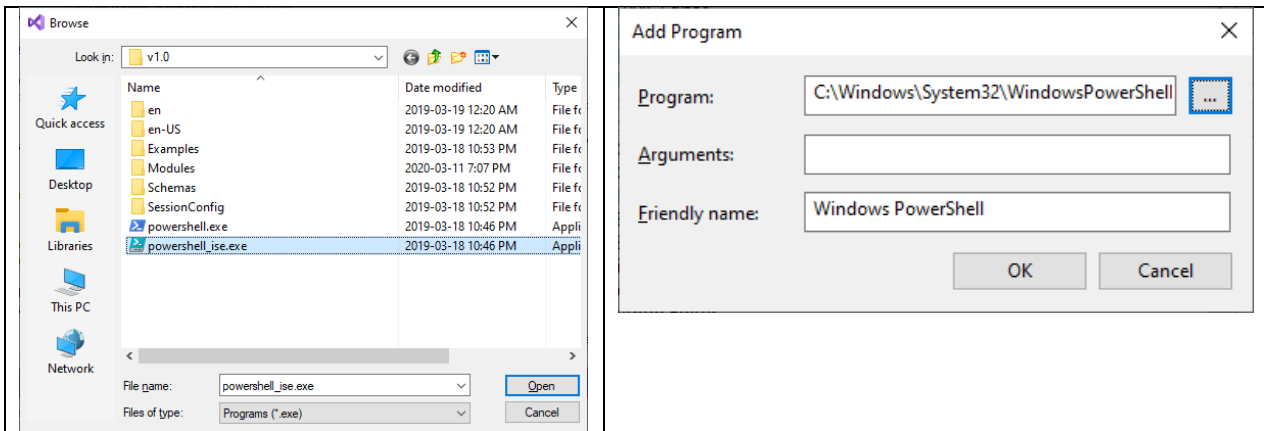
T4 code scaffolding in .NET API project is **driven by added Entity class**. The code scaffolding impacts all projects and builds all required plumbing code for added entity class(es). Generated units tests will be executed with success right after *CodeGeneratorUtility.bat* executed for added entity classes without mandatory table fields after database migration. Otherwise generated tests has to be updated to add mandatory columns data in order to run them successfully.

Executing CodeGeneratorUtility.bat

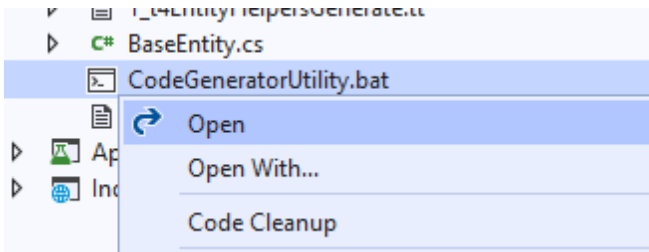
Before this utility is used it has to be set *PowerShell* executable to execute *CodeGeneratorUtility.bat* file.

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.

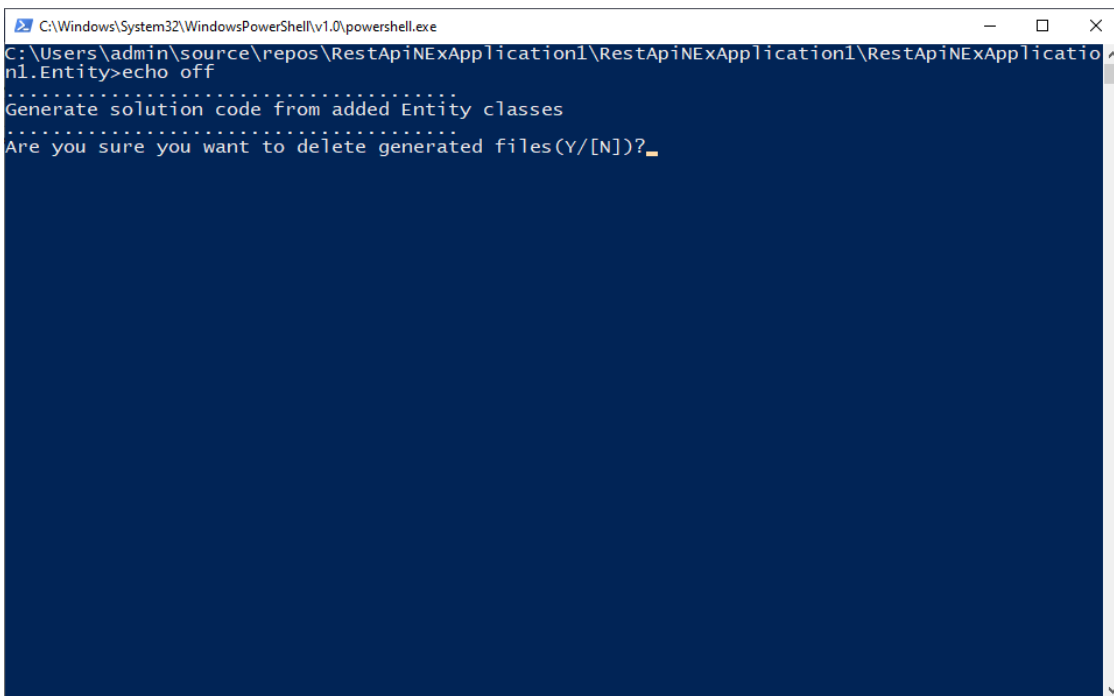


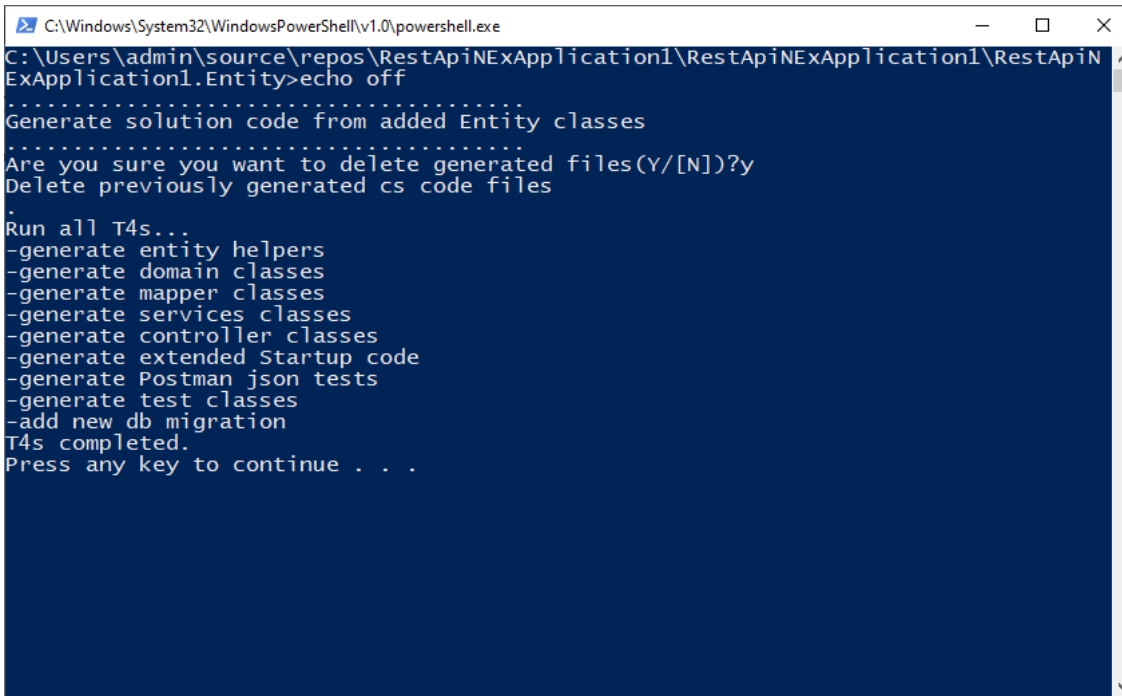


After setting up an application to run **CodeGeneratorUtility.bat** it can be used only **Open** menu item to run all T4 scripts.



..and all *.tt scripts are executed in required order after confirmation **that previously generated t4_*.cs classes will be deleted (you can move code or rename generated cs file to avoid the deletion):**





```
C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe
C:\Users\admin\source\repos\RestApiNExApplication1\RestApiNExApplication1\RestApiNExApplication1.Entity>echo off
.....
Generate solution code from added Entity classes
.....
Are you sure you want to delete generated files(Y/[N])?y
Delete previously generated cs code files
.
Run all T4s...
-generate entity helpers
-generate domain classes
-generate mapper classes
-generate services classes
-generate controller classes
-generate extended Startup code
-generate Postman json tests
-generate test classes
-add new db migration
T4s completed.
Press any key to continue . . .
```

In a case that error is displayed for DB migration just run **CodeGeneratorUtility.bat** again.

Run REST API (Run solution) to deploy generated migration code to the database.

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

```
public partial class RestApiNExApplication1Context : DbContext
{
    public DbSet<Client> Clients { get; set; }

    /// Add new entities concurrency declarations (Fluent API)
    partial void SetAdditionalConcurrency(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 is 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"

More instructions:

<https://www.youtube.com/watch?v=dSZu7ISwtQI>

https://www.youtube.com/watch?v=9k4_BWHXO84&t=3s

<https://www.youtube.com/channel/UC5XyWfG0nGYp7Q9buusealA>

Anasoft VSIX team

December 2020