

# Template Variable Extender

## Documentation

Template Variable Extender is to be used in conjunction and to extend the functionality of Unity template files. This tool expands the available template variables in order to match the functionality offered by other editors.

### Installation

#### Asset Store

No further installation steps are required when installed via the Unity Asset Store.

#### Manual

Clone the git repository into your `Assets` folder. It is preferred that you clone this within the editor directory within your project, however it can be placed anywhere without issue.

Run this command to clone:

```
git clone https://github.com/PolymerCo/UnityKeywordProcessor.git
```

### Creating Templates

The Unity template engine is practically untouched by this extension, as this extension simply interacts with newly imported files and runs a custom template pass. As such, my ability to resolve issues with the template engine itself is limited.

Once you have created your custom template, check out the **Available Variables** section to discover what template variables are available for you to use.

### Project Level Templates

**This feature is only available in 2021+ Unity Editors**

Follow the below steps to create templates on a per-project level:

1. Create a new directory in your `Assets` directory named `ScriptTemplates` (beware: must be called exactly this)
2. Create a new template file using the following naming convention:  
`{TemplateName}.{TemplateExtensions}.txt`
  - a. To override existing templates, such as the "Create > C# Script" template, navigate to the `%EDITOR_PATH%/Data/Resources/ScriptTemplates` directory and examine the templates within there. Find the filename for the template you want to overwrite, then create a new file with the same name in the project's `ScriptTemplates` directory you just made.
3. Restart the Unity Editor, the new template should now be available.

### Editor Level Templates

1. Navigate to the `%EDITOR_PATH%\Data\Resources\ScriptTemplates` directory. Note that this is usually a protected directory, so you will need admin permissions to modify any files in here. To retrieve your editor path, you can right click on the editor within Unity Hub and click "Show in Explorer" (substitute for your OS specific nomenclature)
2. Create a new template file with the following naming convention  
`{TemplateName}.{TemplateExtension}.txt`
  - a. To override existing templates, simply modify that template file. I would recommend making a backup before changing the file to ensure you have a reference point.
3. Restart the Unity Editor, the new template should now be available.

### Template Help

The only semi-official help from Unity is this article from an ex-Unity employee:

<https://support.unity.com/hc/en-us/articles/210223733-How-to-customize-Unity-script-templates>

As you can probably tell, support for custom templates is very limited. Fortunately, not much can go wrong.

# Available Variables

Variable Name	Example	Description
COPY	©	Copyright symbol.
NAMESPACE	script.path	The scripts' namespace based on it's path (1).
DATE	3/06/2022	The current date.
DAY	3	The current day.
MONTH	6	The current month.
MONTH_NAME_SHORT	Jun	The current month's short name.
MONTH_NAME_FULL	June	The current month's full name.
YEAR	2022	The current year.
TIME	1:45 PM	The current time.
HOURL	13	The current hour (24h format).
MINUTE	45	The current minute.
SECOND	32	The current second.
DIR_PATH	Assets/Scripts/MyFile.cs	The path of the file from the project root.
FILE_NAME	MyFile	The name of the file without the extension.
FILE_NAME_FULL	MyFile.cs	The name of the file with the extensions.
EXTENSION	.cs	The file extension.
PRODUCT_NAME	Unity	The name of the editor (will likely always be Unity!).
PRODUCT_VERSION	2022.1.21f1	The version of the Unity editor.
PLATFORM	WindowEditor	The platform of the editor.
PROJECT_NAME	MyProject	The name of your project.
CLOUD_PROJECT_ID	1234	The cloud project ID.
COMPANY	MyCompany	Your company name.
USER	john	The user of the account running the Unity process.

(1) the namespace will include all directories after the root Assets directory, excluding a number of keywords including: scripts, src (2). For example, the file Project/Assets/Scripts/World/Viewer/WorldViewer.cs will produce the namespace World.Viewer.

(2) the excluded directories can be altered by editing the Processors.NamespaceKeywordProcessor.NamespaceExclusions array.

## Using Variables in a Template

When using variables within a template, simply wrap the variable name in # characters. For example, if you wished to use your company name in a template, you would use: #COMPANY#

# General Support + Contribution

Feel free to submit issue reports on the repository GitHub page:

<https://github.com/PolymerCo/TemplateVariableExtender/issues>

Similarly, if you wish to contribute, please submit a PR