

- ▼ [Omni Shader Tools for Unity](#)
 - [Shader Build Settings](#)
 - [Shader Stripper](#)
 - [Shader Programming Support](#)

Omni Shader Tools for Unity

Welcome to Omni Shader!

Omni Shader Tools For Unity is a powerful toolset for Unity developers, designed to streamline shader development and enhance your workflow. Explore the features through the navigation menu on the left and take your Unity projects to the next level!

Below are features of Omni Shader Tools For Unity.

Shader Build Settings

Configure the rules that define what shaders or shader variants should be skipped when building to improve the build time and performance.

Below are descriptions of each settings or buttons:

- **Add Rule Area** is on top of the page which will add rule that defines what build target this rule applies to and how to filter out shaders or shader variants with the value input
- **Shader Skip Rules Table** show all the rules added for this project. Filter the table by platform dropdown menu and remove selected rules by clicking the remove button

Shader Stripper

Shader Stripper will strip code from shaders or shader graphs by removing keywords, passes and comments, following the below settings. And save the result to a .shader file. Very useful for reducing shader variants and improving performance from code level.

Below are descriptions of each settings or buttons:

- **Target Shader** define what shader will be stripped
- Click **Refresh** Button will get keywords and passes from target shader automatically
- Click **Generate** Button will generate new shader
- **Exclude Keywords** define what keywords will be removed from shaders
- **Exclude Passes** define what passes will be removed from shaders
- **Save Shader Path** will be the new shader name

- **Save File Name** will be the new file name (end with .shader) which will be saved at same directory of the original shader

Shader Programming Support

Omni Shader Tools For Unity has extensions of popular IDEs that support shaderlab programming. It provides syntax highlighting, code completion, and other features to make shader development easier. We can preview core language features here: <https://omnishader.amlovey.com/docs/?doc=core-features>

Currently supported IDEs:

- Visual Studio Code
- Visual Studio