

Editor Terminal Manual

Unity Editor Plugin

BITPURR **DIGITAL**

support@bitpurr.com

bitpurr.com

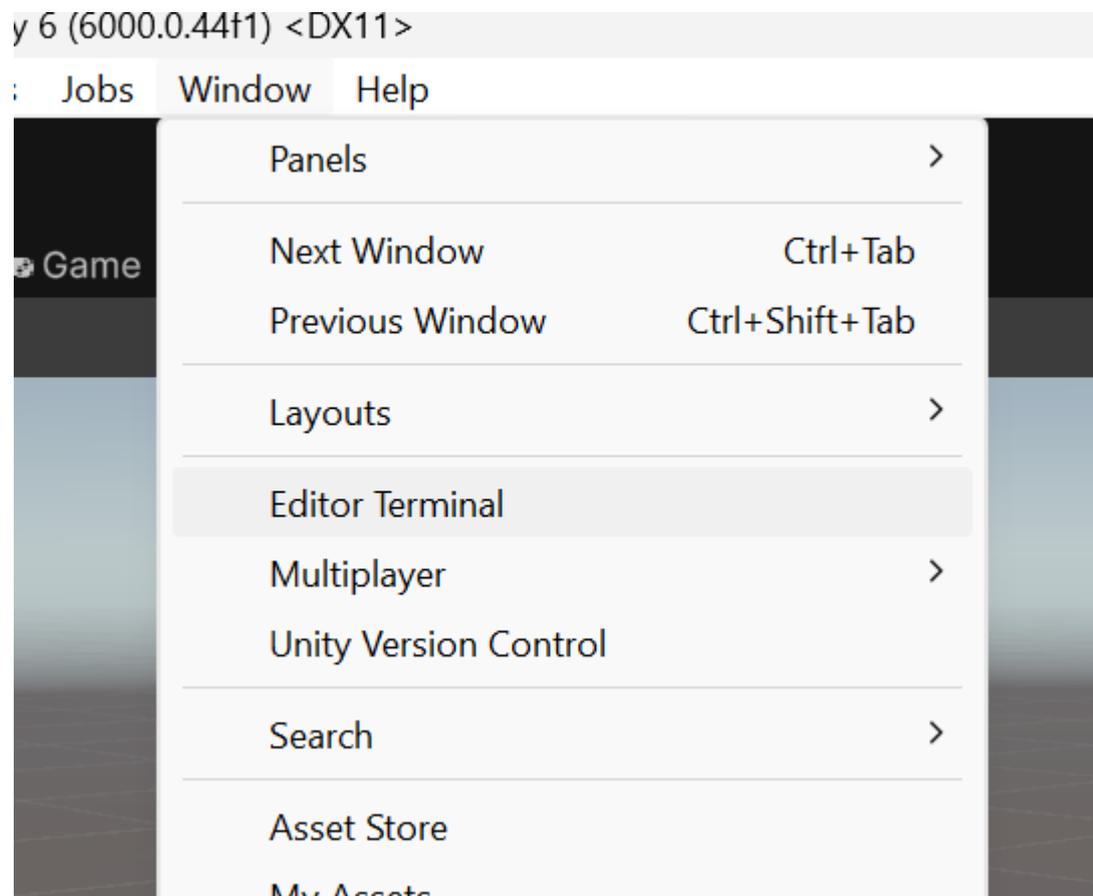
Usage	2
Opening Editor Terminal Window.....	2
Docking the Editor Terminal Window.....	3
Tab Completion.....	3
Clearing The Terminal.....	3
Keyboard Commands.....	4
Commands.....	4
Running commands as root.....	4
Internal Commands.....	4
Configuration	6
Git Optimization.....	6
Shell Selection.....	6
Configuration Storage.....	6
Support	7

Usage

Editor Terminal is designed to emulate a standard terminal closely, so some of the usage should be intuitive. However the integration with Unity means some common terminal features will be missing or work slightly differently.

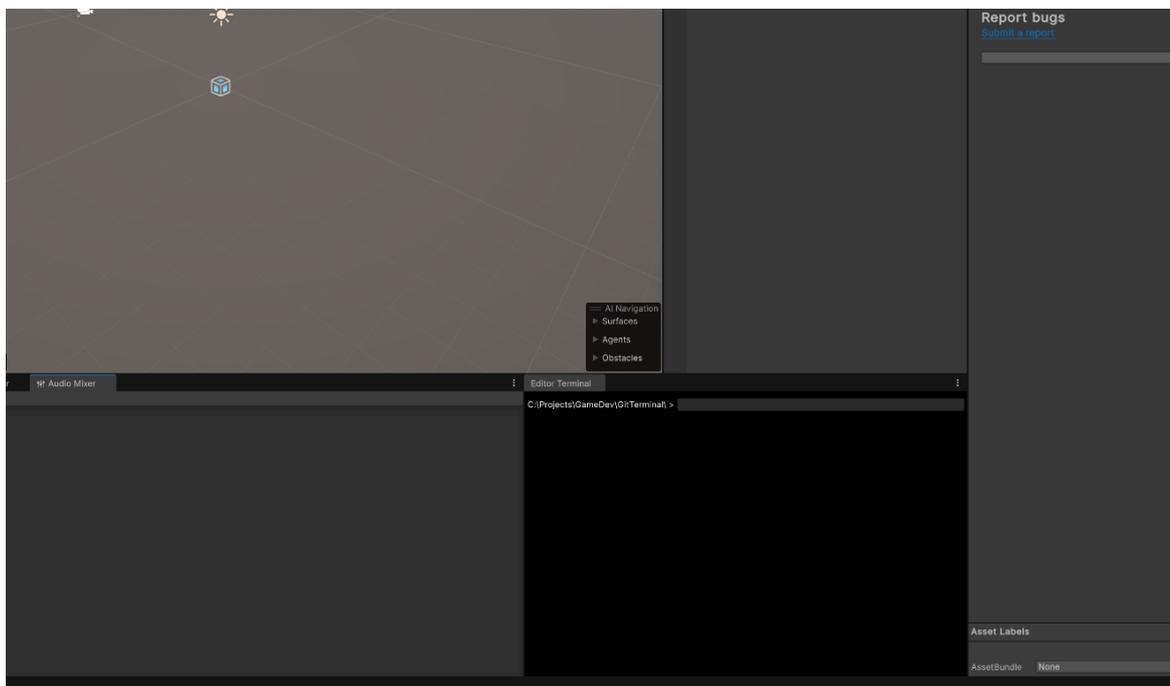
Opening Editor Terminal Window

After the Editor Terminal unitypackage is installed. A new option named 'Editor Terminal' will appear on the 'Window' toolbar option at the top of the Unity Editor. Select 'Editor Terminal' to open the Editor Terminal window. If the window does not appear, ensure that you are using a supported version of Unity (Unity 2022.3 and above).



Docking the Editor Terminal Window

Once opened, the Editor Terminal window can be manipulated and docked like other Unity windows. It can become part of your everyday layout.



Tab Completion

Tab completion in Editor Terminal works by pressing the tab key while the window is focused. The current directory will be searched for the closest matching file or directory and the result will be displayed in the text box. Pressing Tab consecutively will cycle through different options, but pressing any other key will 'Complete' and following tabs will start a new 'cycle' of completions at the updated path.

Clearing The Terminal

The terminal can be cleared by pressing **CTRL + L** on Windows / Linux or **CMD + L** on OSX. It can also be cleared by running the **clear** and **cls** commands.

Keyboard Commands

Editor Terminal supports the following keyboard commands while the window is focused. The commands require holding a modifier key. Which is **CTRL** on Windows / Linux and on OSX is **CMD**.

Key Combination	Description
CTRL / CMD + L	Clears the terminal.
CTRL / CMD + U	Clears the command input field.
CTRL / CMD + K	Removes text after the cursor in the command input field.
CTRL / CMD + Z	Attempts to kill the last run process, if it has not yet exited.
CTRL / CMD + Minus	Decreases the Editor Terminal font size.
CTRL / CMD + Equals	Increases the Editor Terminal font size.

Commands

Editor Terminal should default to a good configuration, ready to run your commands immediately. Running commands should be the same as your regular terminal.

Running commands as root

At this time running commands as root user is not supported by Editor Terminal and is potentially dangerous.

We recommend that you do not attempt it and stick to using your userland permissions when using Editor Terminal.

Internal Commands

Internal commands are used just like regular commands, however they don't execute any background process and are purely for configuration.

They follow a common pattern of "prefix - command - argument"

There are 2 prefixes used to execute an editor command:

`editorterminal`

Or

`et`

Which is the short form.

Likewise, each internal command has a long and short form also:

Command	Short Form	Description
set_git_location	sgl	Manually set your git executable. This is used to optimize git commands.
set_shell	ss	Set the shell to use for background processes. This is either cmd or powershell on windows. This is a path on Mac / Linux. (Do not change this unless you know what you are doing).
set_font_size	sfs	Set the font size. Defaults to 14.
set_font_color	sfc	Set the font color as a hex code.
set_background_color	sbc	Set the background color as a hex code.
set_default_colors	sdc	Sets the defaults Editor Terminal colors. Black background white white font color.
config	c	Displays the current configuration.
help	h	Displays the help text. This is the default if no other valid Editor Terminal command is given.

This means, if we want to set the editor terminal background to be solid red. We can either use the long form or short form commands:

- editorterminal set_font_color ff0000
- et set_font_color ff0000
- editorterminal sfc ff0000
- et sfc ff0000

Configuration

Editor Terminal should be well configured 'out of the box'. However you can configure a few parameters using **internal commands**. You can configure:

- Terminal Colors and font sizes.
- The Shell used (powershell / cmd for windows, your choice for linux / OSX).
- Git Executable Location - This is used for the Editor Terminal's Git Optimization.

Git Optimization

Editor Terminal optimises 'git' commands, by running the 'git' executable directly with the provided arguments, which skips creating a shell process in the background. This reduces some overhead and makes the terminal feel more responsive. When git is installed normally, Editor Terminal should have no issues finding the executable on Windows, Linux and Mac. However if your setup is a different and you still want to use the git optimization feature, you can set the path for the git executable by using the **set_git_location internal command**.

Shell Selection

By default Editor Terminal will use PowerShell on Windows and bash on Linux / OSX. However you can change these if you like using the **set_shell internal command**.

On Windows, you can use either Powershell or CMD. Powershell is the default and recommended.

On Linux / Mac, you can set the path to any shell you choose. Bash is the default and recommended. Best effort is made to support Bash, zsh, sh and fish. If you decide to use a different shell, it is highly recommended that you choose one that will function as a drop in replacement for bash.

Configuration Storage

All configuration settings are stored in the Unity Editor Preferences, this means that you do not need to commit any files to source control or add any extra files to .gitignore. Configuration settings will persist between editor sessions and are on a 'per project' basis. This means you can configure Editor Terminal differently for different projects.

Support

For further support or queries, please reach out to us by emailing support@bitpurr.com.

You can also visit <https://bitpurr.com/support>

Thank you for using Editor Terminal!