Smart Console

Unity Package

User Manual

Version 1.0
November 11, 2022

SMART CONSOLE
FREE

Table of content

1. Introduction

Thank you for your interest in this package! I'm sure it will serve you well and help you realize your game ideas in Unity!

In case you run into any problems or if you need any help with something related to this package, feel free to contact me at edgarcovarel@yahoo.fr or using the discord server.

The same goes for any feedback or thoughts you might have on this package, of course! I'd love to hear your ideas!

2. About this Package

1. What is this?

Smart Console is a **runtime console commands**. Add attribute *[Command]* to your own methods to be able to use it in the console!

It's both easy to understand and easy to use for your game's specific needs.

I specifically design the console like the default Unity Editor console, but you are open to change it as your wishes.

My main goal is to provide a stable and robust solution at a **free cost**.

2. What kind of projects can I use this for?

For any project that need to implement **cheat codes**, or need to have **debugging logs** or even for projects that need to implement **advanced QA systems**.

3. Create a Command

1. Inherit your class from CommandBehaviour

Instead of inheriting from *Monobehaviour*, Inherit your class from *CommandBehaviour*.

It allows the system to know that it needs to look there to find command methods.

CommandBehaviour is inheriting from *Monobehaviour* so it does not change anything for you.

Your class is now ready to host command methods!

2. Use the [Command] attribute

To identify a method as a command, simply write *[Command]* above it.

It support :

- Public/private method
- Static/non-static method
- Method with int/float/bool/string parameter(s)
- Method with optional parameter(s)

It **does not** support method overloading

Your method is now ready to use in the console!

4. Use the Console

1. Import the Console

Import the *Console* prefab (SmartConsole/Prefabs/Console.prefab) in the desired scene. Make sure that you have one *Event System* in the scene.

2. Open/Close the Console

By default, you can open and close the console pressing the quote key (^{2}). You can also use the close button to close the console.

You can change the key in the *Console* prefab in its *Console System* component.

3. Autocompletion

By default, you can autocomplete your field pressing the tab key. It will look for commands that start with the current input field's value, keep pressing the key to navigate through the found's commands.

You can change the key in the *Console* prefab in its *Console System* component.

4. History copy

By default, you can copy log message history pressing the up arrow key or down arrow key. It will copy your previous log messages into the input field, navigate up or down your historic pressing the up arrow or down arrow key.

You can change the key in the *Console* prefab in its *Console System* component.

5. Use a command

To use a command, make sure that the console's input field should be filled by the name of the command, by writing or using *autocompletion* or *history copy* and press enter key or the submit button.

To use a command containing parameters, fill the console's input field like a normal command and add your parameters separating them with a space.

6. With the Legacy Input Manager

To use the console with the legacy input manager, make sure to have an adapted *Event System* in the scene and make sure that key parameters has been set in the *Console* prefab in its *Console System* component.

7. Default commands

Default commands are included in this package, I recommend you test them and use it as reference for your own commands. The script is located at (SmartConsole/Demo/Scripts/DefaultCommands.cs) and just need to be attached to a GameObject in a scene to run.

It contains :

- Hello world command → print Hello World!
- Reload current scene command → reload current scene
- Load scene command → load scene using parameter int index
- Help command → list every usable command