Color Script

Introduction

Color Script is a Unity Editor enhancement that displays C# scripts within Unity in a much clearer and more usable way than the default Unity MonoScript Inspector. It can completely replace Unity's existing Inspector*, and it can also be opened in a new Editor Window.

Features

- Syntax highlighting
- Color assignments are replaced with color swatches (modifiable In Color Scipt Pro).
- Hyperlinks in comments are clickable
- Text can be selected and copied
- Errors displayed. Click on one to open the External Script Editor
- A wide range of customization options, including a choice of 10 visual themes.

Color Script is available as both a paid (*Color Script Pro*) and free version. The free version, while still very capable, does not feature any toolbar functionality, or ability to modify assigned colors.

Usage

Once imported into a Unity project, *Color Script* will automatically embed itself into the monoScript inspector and replace the default inspector view. The inspector appearance can be toggled between the old default Unity inspector and Color Script using the toolbar buttons at the top of the script view.

The **Window>Color Script** menu item can be used to open an additional *Color Script instance* in a separate Editor Window.
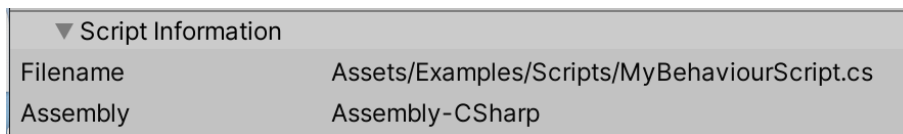
Color Script Inspector
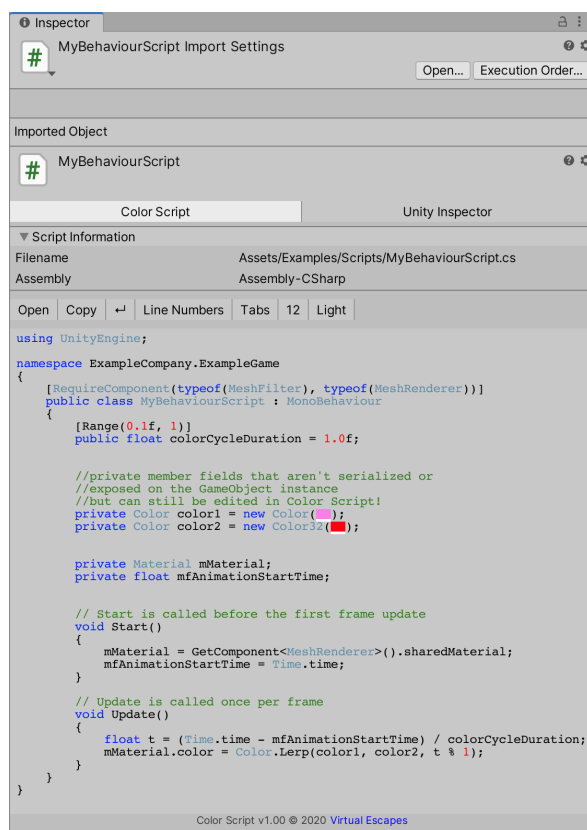
Unity Default Inspector

* Unity2019.x and later

Color Script v1.0 User Guide

User Interface

The Script Information foldout

| ▼ Script Information	
Filename	Assets/Examples/Scripts/MyBehaviourScript.cs
Assembly	Assembly-CSharp

The script information foldout can be opened and closed by tapping the **Script Information** heading.
Filename displays the location of the currently selected MonoScript within your Unity project.
Assembly displays the C# assembly within which the current script is located.

Hover your mouse over each entry for more verbose information.

The Toolbar

Open	Copy	↵	Line Numbers	Tabs	13	Light	🔒
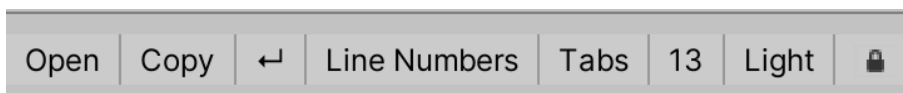
Open
Opens the currently selected script in your external script editor, eg Visual Studio.

Copy
Copies the entire script into your computers clipboard, allowing you to easily paste it elsewhere.

Line Endings ↵ (Default Off)
Toggles the visibility of line endings. Any line endings that do not match your current platform will always be displayed in red.

Line Numbers (Default Off)
Toggles the visibility of line numbers.

Tabs (Default Off)
Toggles the visibility of tab characters. If your code uses spaces to indent, then no change will be visible.

Font Size (Default 13)
Specify the size of the text. Allowed values are between 8 and 20 points, inclusive.

Theme (Default *Light*)
Pick between ten different color themes, to match your mood or taste.

Lock (Default Off, Windowed mode only)
Toggle the lock to prevent the script contents from changing with your selection. May also be referred to as 'Pinning'. This can be useful if you want to compare one script with another one selected in the Editor Inspector.
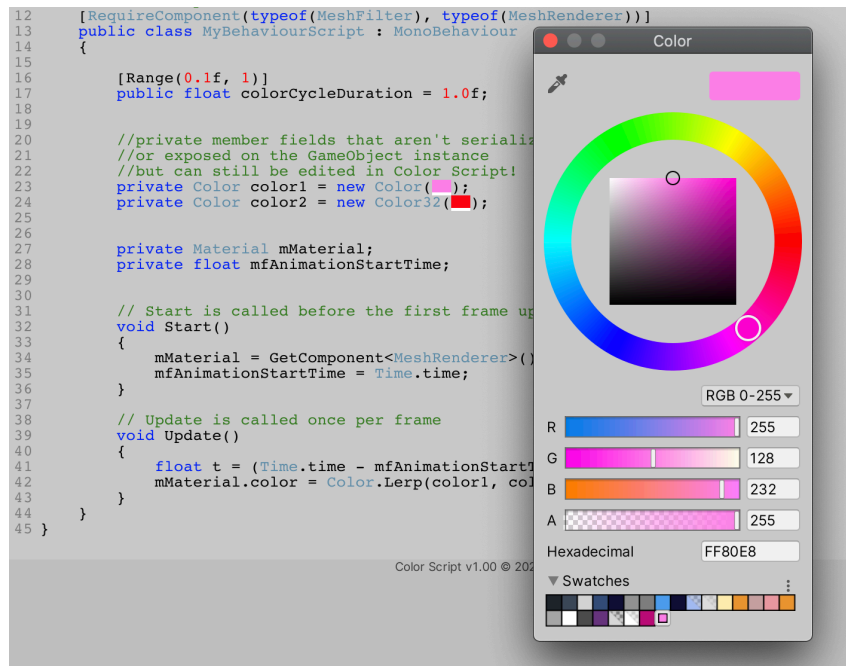
Additional Functionality

Color Selection

Any color assignments that Color Script identifies in your code will display as color swatches. Tapping on the color swatch will open a color wheel, enabling you to easily assign colors.

Once any colors have been modified the toolbar will change appearance:
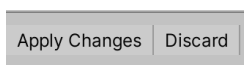
Once you have finished modifying your colors tap Apply changes to save the changes back into the C# file*, or Discard to leave the file as it is.

```
12    [RequireComponent(typeof(MeshFilter), typeof(MeshRenderer))]
13    public class MyBehaviourScript : MonoBehaviour
14    {
15
16        [Range(0.1f, 1)]
17        public float colorCycleDuration = 1.0f;
18
19
20        //private member fields that aren't serializ
21        //or exposed on the GameObject instance
22        //but can still be edited in Color Script!
23        private Color color1 = new Color();
24        private Color color2 = new Color32();
25
26
27        private Material mMaterial;
28        private float mfAnimationStartTime;
29
30
31        // Start is called before the first frame up
32        void Start()
33        {
34            mMaterial = GetComponent<MeshRenderer>()
35            mfAnimationStartTime = Time.time;
36        }
37
38        // Update is called once per frame
39        void Update()
40        {
41            float t = (Time.time - mfAnimationStartT
42            mMaterial.color = Color.Lerp(color1, col
43        }
44    }
45 }
```

Color Script v1.00 © 202

Mismatched Line Endings

Color Script will display a warning if it encounters any mismatched line endings within the selected script. It will also display them as as ↵ characters. These are line endings that do not match your current platform - eg Unix endings while running on a Windows PC. Unity also displays a warning of mismatched line endings in the console but provides no simple way to fix them. *Color Script* does.

```
⚠ 16 mismatched line endings found                    Fix

using System.Collections;↵
using System.Collections.Generic;↵
using UnityEngine;↵

public class NewBehaviourScript : MonoBehaviour↵
{↵
    // Start is called before the first frame update↵
    void Start()↵
    {↵
        ↵
    }↵
```

Click on the **Fix** button and the mismatched line endings will be replaced* for line endings that match your development platform.
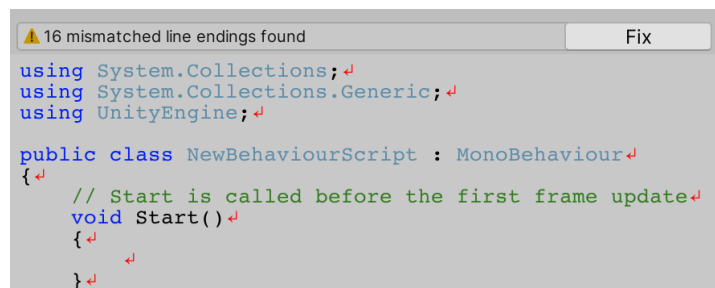
Important Disclaimer: Color Script and Virtual Escapes AS can not be held responsible in the very unlikely event that work is lost while modifying scripts. Always back up your work regularly and be aware that saving files can result in data loss, particularly (but not limited to) during electrical storms and/or when important deadlines are approaching.

Selection

Code in the *Color Script* window can be selected, either by double-clicking or dragging. Once selected Ctrl / ⌘ C will copy the text to the clipboard. Copy is also accessible from the context menu, accessible by right-clicking on the selected text.
Selection areas that are larger than the *Color Script* window can be made by holding the mouse close to the borders of the window while drag-selecting. This will scroll the display.
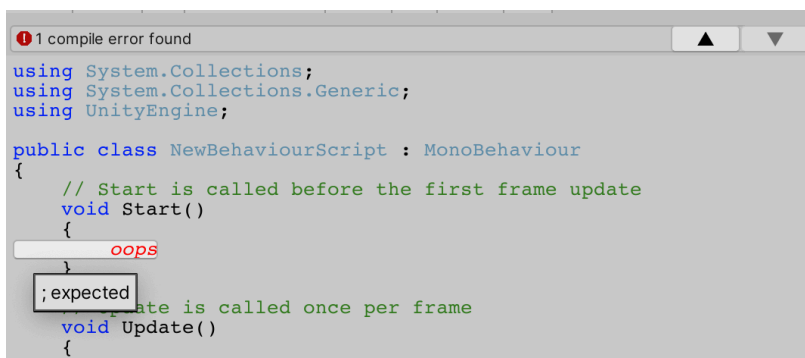
```
using System.Collections.Generic;
using UnityEngine;

public class NewBehaviourScript : MonoBehaviour
{
    // Start is called before the first frame update
    void Start()
    {

    }

    // Update is called once per frame
    void Update()
    {
```

Compilation Errors and Runtime Exceptions

Color Script will identify any problems with your script by highlighting the offending line of code and displaying it in red. Hovering over it with your mouse will give you details on what the error is and clicking on it will open your external script editor at the location of the error.

```
🛑 1 compile error found                    ▲    ▼
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class NewBehaviourScript : MonoBehaviour
{
    // Start is called before the first frame update
    void Start()
    {
            oops
    }
    ; expected
    ...ate is called once per frame
    void Update()
    {
```

Limitations and Known Issues
These are some of the things we hope to get working in future versions.

Unity pre 2019.x
In Unity versions older than 2019.1, rather than replacing the MonoScript Inspector, *Color Script* can only be opened in a separate Editor Window. In addition to the menu item **Window>Color Script**, a button to open the *Color Script* window is displayed within the MonoScript Inspector. This window can still be docked anywhere within the Unity Editor, for example below the Inspector.

Color Tags. *Color Script* currently can not correctly display code with rich text color tags (eg: <b><color=red> etc). If you open code which contains these tags, the output will appear corrupted.

Multiple Scripts with Compile Errors
Color Script is currently unable to update a scripts error information if there are errors spread across multiple files until all the errors are fixed.

Feedback

We at Virtual Escapes want you to be happy with the products and software we build.

If you have any feature requests, bug reports or any feedback at all, please get in touch:
info@virtualescapes.no