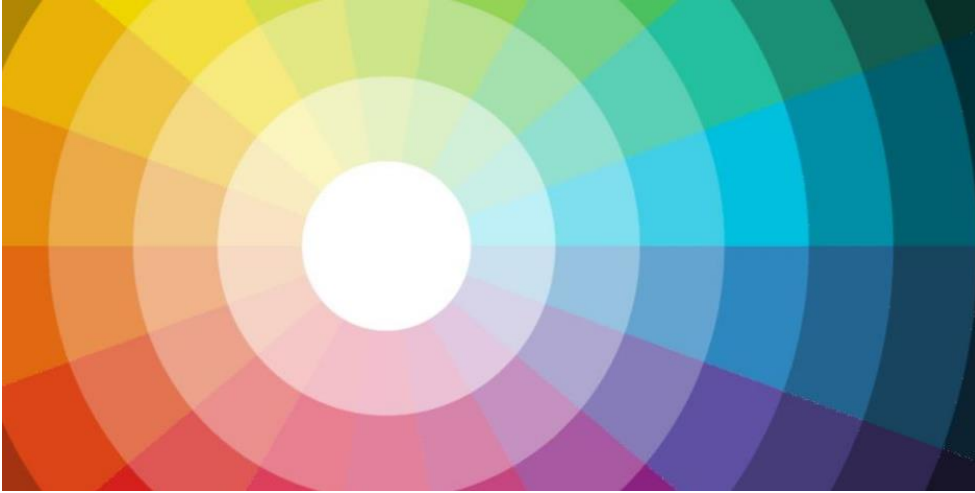


colors



AN INTRODUCTION TO USING COLORS FOR UNITY

v1.1

Q&A https://gamelogic.quandora.com/colors_unity
Knowledgebase
Online <http://gamelogic.co.za/colors/documentation-and-tutorial//>
Documentation
API <http://www.gamelogic.co.za/documentation/colors/>
Documentation
Support support@gamelogic.co.za
Twitter <https://twitter.com/gamelogicza>



gamelogic

CONTENTS

Contents.....	1
Using the node editor	3
Creating a color graph.....	3
Working with nodes.....	3
Computing the output.....	3
Linking generated colors to your scene	3
Node reference	4
Input Nodes	4
Constant Color <None, Color>	4
Random Color <None, Color>	4
Mixing Nodes.....	4
Mix2 <Color, Color>	4
Random Mix 3 <Color, Color>	4
HSL Channel Nodes	4
Set Hue <Color, Color>	4
Set Luminance <Color, Color>	4
Set Saturation <Color, Color>	4
Offset Hue <Color, Color>	5
Offset Luminance <Color, Color>	5
Offset Saturation <Color, Color>	5
Channel Inversion Nodes.....	5

Invert Color <Color, Color>	
Invert Luminance <Color, Color>	
Invert Saturation <Color, Color>	
Gradient Nodes	
Gradient <Color, Gradient>.....	
HSL Gradient <Color, GradientHSL>	
Sample Gradient <Gradient, Color>	
Sample HSL Gradient <GradientHSL, Color>.....	
Working with the Procedural Palette Generator Component	
Selecting an algorithm.....	
Algorithm reference.....	
Uniform randomness	
Random walk	
Gradient.....	
Even	
Jitter [Removed since v1.1].....	
Random.....	
Golden ratio	
Random offset	
Random mix.....	
Random hue.....	
Random saturation	

Random luminance.....	12
Random saturation and luminance	12
Harmony.....	13
Tints.....	13
Shades	13

USING THE NODE EDITOR

Creating a color graph

To create a new color graph, select **Gamelogic | New Color Graph** from the menus. A wizard will appear that allows you to save the color graph in your project.

Once you have saved the file, click on it in the project view, and select **Edit** in the inspector. The Graph editor will open, from where you can add and edit nodes.

Working with nodes

To add a node, click one of the node buttons on the left of the window. Note that the first two buttons are *not* node buttons. A new node will appear, which you can drag to a convenient place on the window.

To remove a node, you can click on the **Remove Node** button on the node.

To remove all nodes from the graph, click the **Clear** button on the left of the window.

To add a link from one node to another, click on the first node's **Add link** button. A link will appear, that you can connect to the second node by clicking on it. Not all nodes can be connected; the link will not form if it is illegal. Use the node reference to see which nodes are legal.

To remove a link, click on the red disk in the middle of the link.

Computing the output

To generate colors, you need to click the **Recompute** button on the left of the window. Colors will only be computed for nodes that have the right inputs. Your graph should have at least one input node (**Constant Color** or **Random Color**). These nodes do not have inputs. All nodes can be used for inputs for other nodes, as long as they are of the right type.

Linking generated colors to your scene

Once you have created a graph with at least one input node, you can link it into your scene through scripting:

- 1) Add an `InspectableGraphNode` public field to the script where you want to write the logic. This will allow you to link in the graph and select the node from which you want to get the colors from. You can add multiple fields of `InspectableGraphNode` if you want to access more than one node from your script.
- 2) In code, you can get the colors of the node if the `InspectableGraphNode` field using its `Colors` property. This is simply a list of colours. You can now write code to apply these colors to materials, sprites, `UIImages`, lights, or anywhere else you can adjust the color.
- 3) In the Unity editor, drag the graph object in the appropriate field, and select the node you want to work from.

The examples that ship with colors show a simple way to follow these three steps to color different scenes.

NODE REFERENCE

In the following, we will use the notation <InputType, OutputType> to show what the input and outputs of a node are. None is used where a node does not have inputs or outputs.

You can only connect top nodes <A, B> and <C, D> if B and C are the same, for example, you can link <None, Color> to <Color Gradient>, but you cannot link <None Color> <Gradient, Color>.

Input Nodes

Constant Color <None, Color>

Takes the colors selected in the nodes properties, and outputs them.

Colors the colors that will become the outputs.

Random Color <None, Color>

Generates random colors, and outputs them.

Color Count the number of colors to generate.

Mixing Nodes

Mix2 <Color, Color>

Takes two colors as input, and mix them together using linear interpolation.

T the blend factor. A value of 0 gives the first color, a value of 1 gives the second, a value of 0.5 gives the color halfway between the two.

Mix Mode The color space in which to do the interpolation.

Random Mix 3 <Color, Color>

Randomly mixes its first three inputs to produce the outputs.

Mix Mode the color space in which to do the mixing.

Grey Control a parameter that determines the effect of one of the three nodes (randomly chosen). When it is 0, one of the inputs have no effect, when it is 1, all three nodes have equal potential effect. When using primary colors as inputs, having low values of this parameter prevents too much greyness in the output colors.

HSL Channel Nodes

Set Hue <Color, Color>

Set Luminance <Color, Color>

Set Saturation <Color, Color>

These nodes work all the same, except that they operate on different channels in the HSL space.

Multi When selected, a set of **Count** random hues (luminances, saturations) is generated, where each set is applied to each color in the inputs, resulting in **Count x Input#** output colors oin total. When not selected, only one hue value is generated for each input color resulting in **Input#** output colors in total.

Count the number of random values to create when multi is true. Count only has an effect if multi is true.

Hue (Luminance Saturation) the random values generated are centered around this value.

Range the range of the random values generated. Use a range of 0 to set values non-randomly.

Example: If Multi is true, Count is 2, Hue is 0.3, Range is 0.2, and the node has 3 inputs, then there will be 6 outputs (two for each input color). The hues will vary from 0.2 to 0.4, and the colors 0, 2, and 4 will have the same hue, and colors 1, 3, and 5 will have the same hue. The other properties will be taken from the input colors, so output colors 0 and 1 will have the same luminance and saturation as input color 0, output colors 2 and 3 will have the same luminance and saturation as input color 1, and so on.

Example: If Multi is false, Count is 2, Hue is 0.3, Range is 0.2, and the node has 3 inputs, there will be 3 outputs. The hues will vary from 0.2 to 0.4, but this time, they may all be different. The other properties will be taken from the three input colors. In this case, count has no effect.

Example: If multi is false, Count is 2, Hue is 0.3, Range is 0, and the node has 3 inputs, there will be 3 outputs, identical to the input colors except that their hues will be set to 0.3.

Offset Hue <Color, Color>

Offset Luminance <Color, Color>

Offset Saturation <Color, Color>

These work exactly the same as the previous three nodes, except that instead of setting channel values, they offset them.

Channel Inversion Nodes

Invert Color <Color, Color>

Inverts all three RGB channels of each of the inputs.

Invert Luminance <Color, Color>

Inverts the luminance channel of each of the inputs.

Invert Saturation <Color, Color>

Inverts the saturation channel of each of the inputs.

Gradient Nodes

Gradient <Color, Gradient>

HSL Gradient <Color, GradientHSL>

These two nodes create gradients from the input colors, placed evenly in order. Gradient uses RGB interpolation, HSL Gradient uses HSL interpolation.

A Gradient node must always be connected to a Sample Gradient node, and a HSL Gradient Node must always be connected to a sample HSL Gradient Node.

Sample Gradient <Gradient, Color>

Sample HSL Gradient <GradientHSL, Color>

These nodes work the same – they sample colors from a gradient – except that they operate on different inputs.

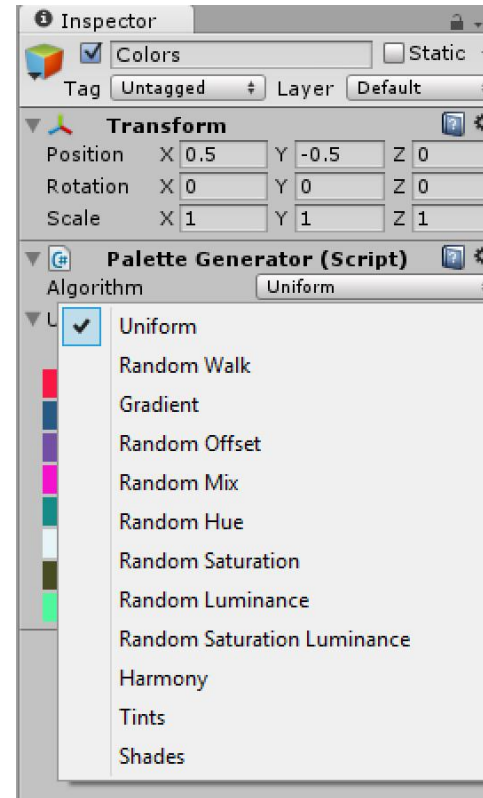
Selection Mode How output colors are selected from the gradient. *Even* selects the colors evenly in order. *Random* selects the colors randomly (uniformly distributed), and *Golden Ratio* gives colors with successive color's ~ 0.618 apart (wrapped).

WORKING WITH THE PROCEDURAL PALETTE GENERATOR COMPONENT (LEGACY)

Selecting an algorithm

Create an empty object in your scene. Add a component, select scripts, and choose Palette Generator.

Inside the Palette Generate component, you can select the algorithm that you want to use.



The generator only generates the colors – it is up to you to link it to your scene. The examples we provide should give you a good starting points for doing this.

ALGORITHM REFERENCE

Uniform randomness

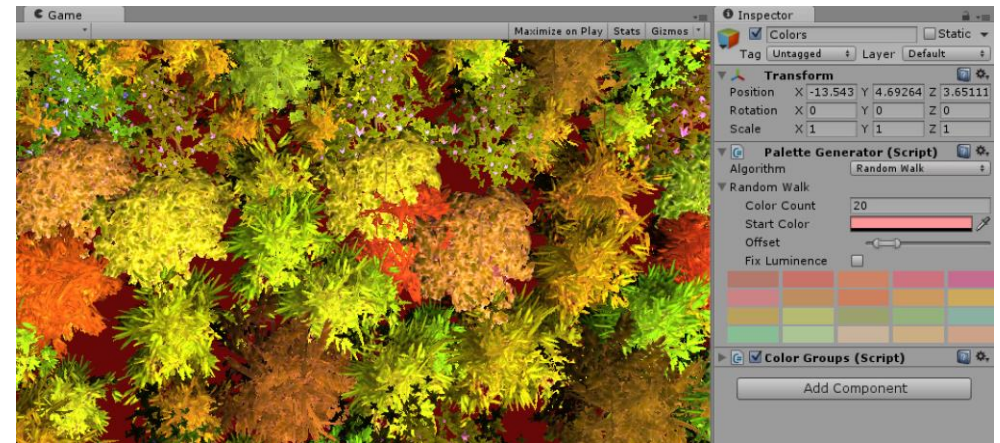
Random values are procedurally chosen from each of the RGB channels. Therefore, there is no structure or relationship between the colors in the palette.

- Select color count (the number of colors that you want in your palette).

Random walk

Each successive color is randomly offset from the previous color, depending on the minimum and maximum offset values that you select.

- Select color count
- Select starting color
- Select minimum offset and maximum offset
- Choose fix luminance or not fix luminance (fix luminance makes all colors in the palette have the same luminance as the starting color).



Gradient

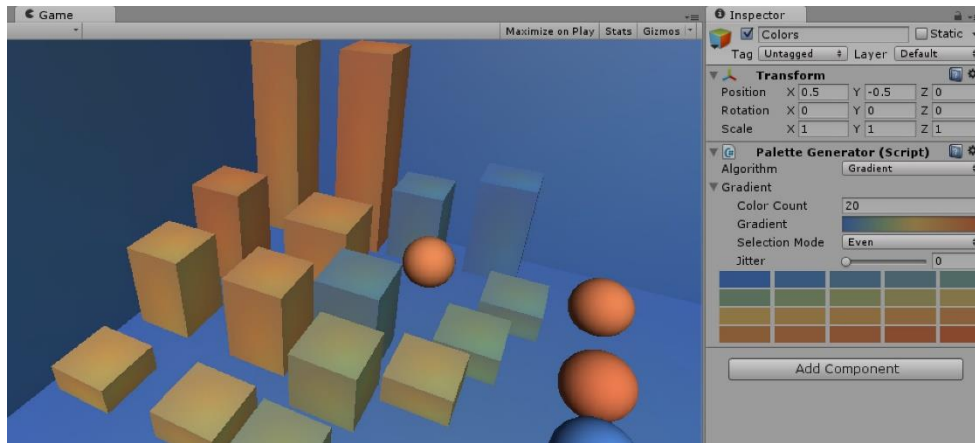
All colors in this algorithm are selected from a color gradient.

- Select color count
- Select gradient (add, remove and position colors on the gradient)
- Select mode

There are four modes.

Even

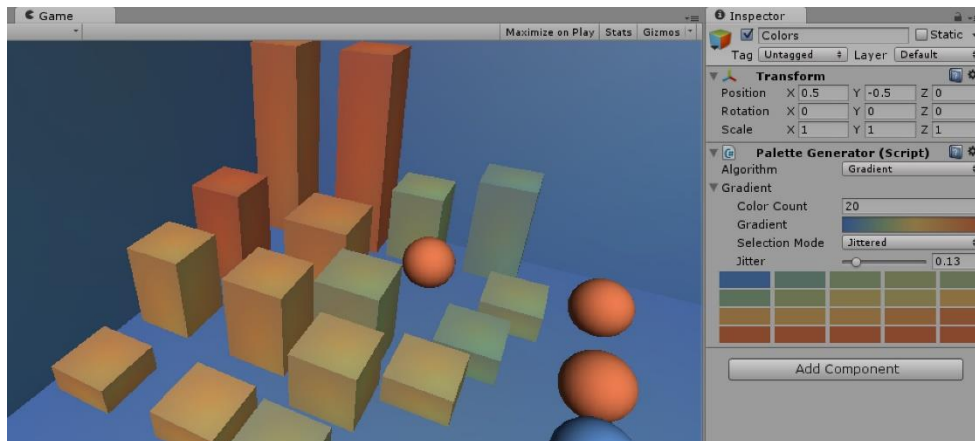
Each color in the palette is evenly spaced across the gradient.



Jitter [Removed since v1.1]

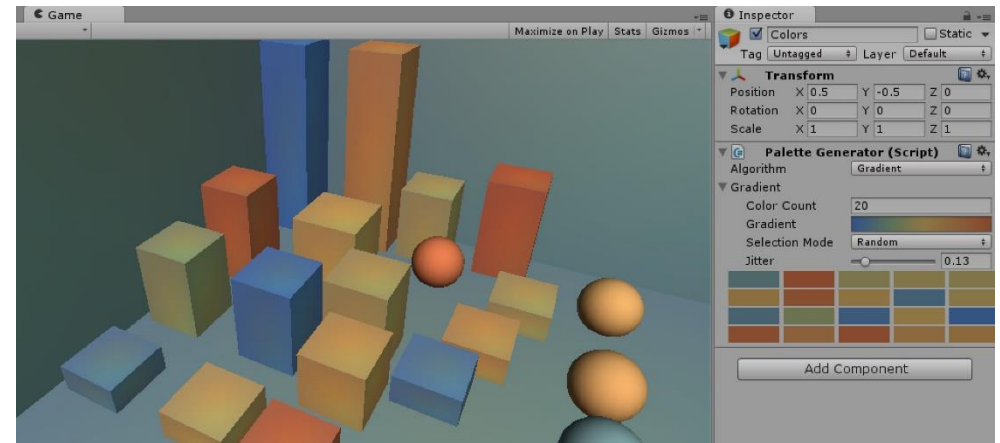
Each color is not evenly spaced across the gradient.

- Select the level of jitter (unevenness)



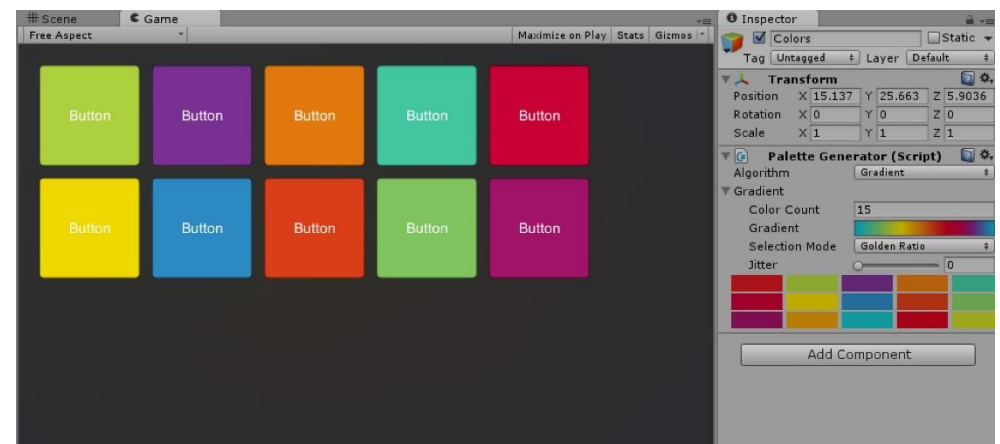
Random

Colors are randomly selected from the color gradient.



Golden ratio

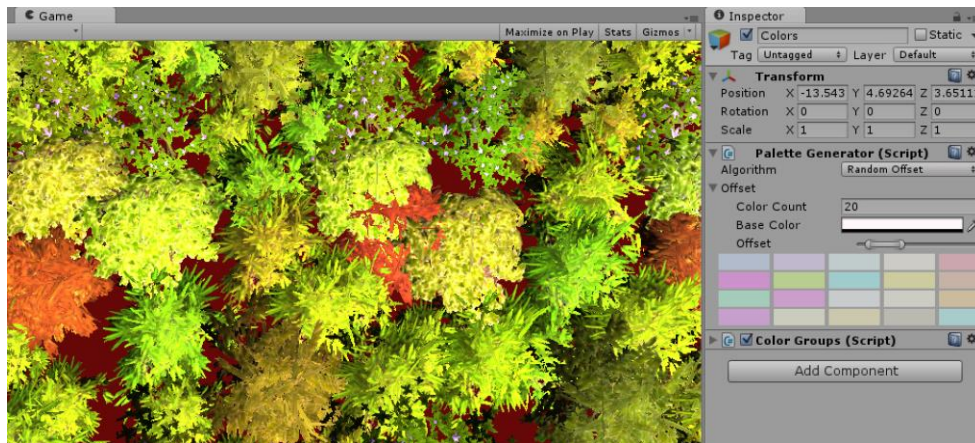
Each successive color has a high contrast to the previous color and no color will ever be repeated. There is suitable for a GUI where a high contrast is required.



Random offset

Each color is randomly offset from the base color, depending on the minimum and maximum offset values that you select.

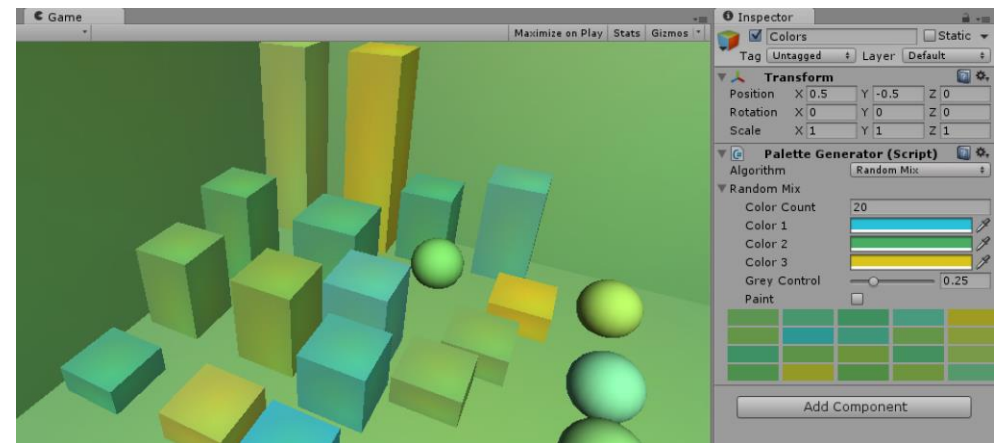
- Select color count
- Select starting color
- Select minimum offset and maximum offset



Random mix

Each color in the palette is generated using a random mix of the three colors selected.

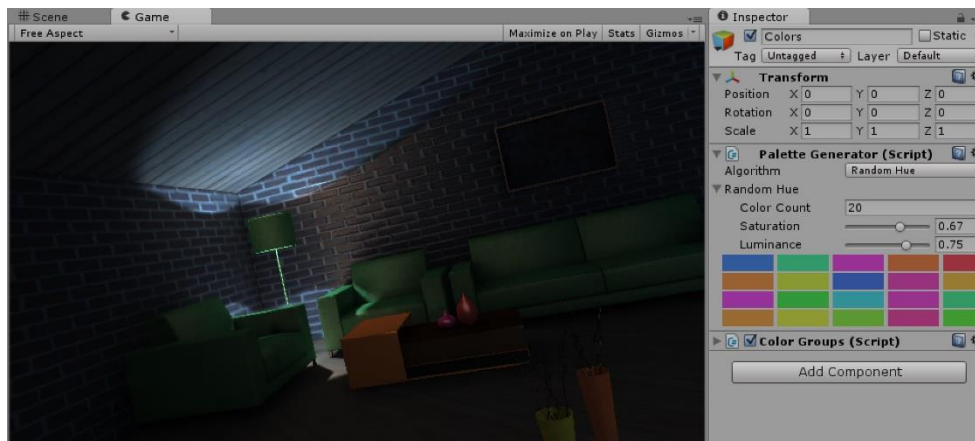
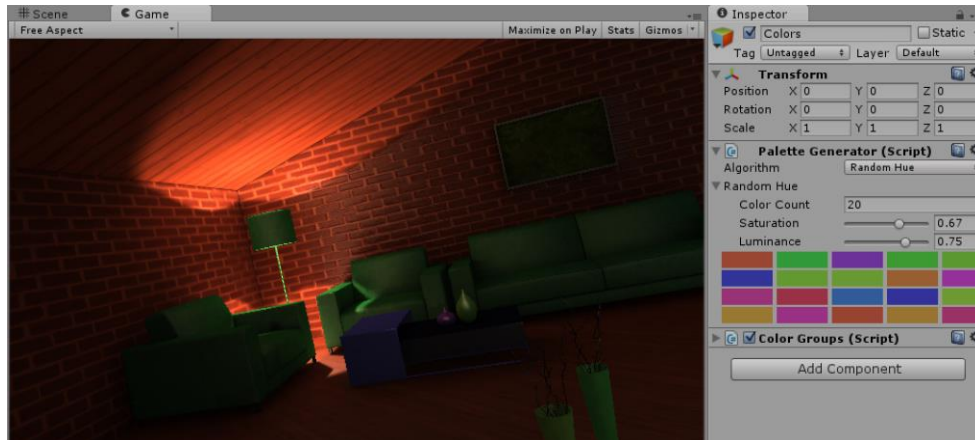
- Select color count
- Select color 1, 2 and 3
- Select level of grey
- Choose between paint mode (a slightly different technique with a different result) and non-paint mode



Random hue

Generates a random level of hue.

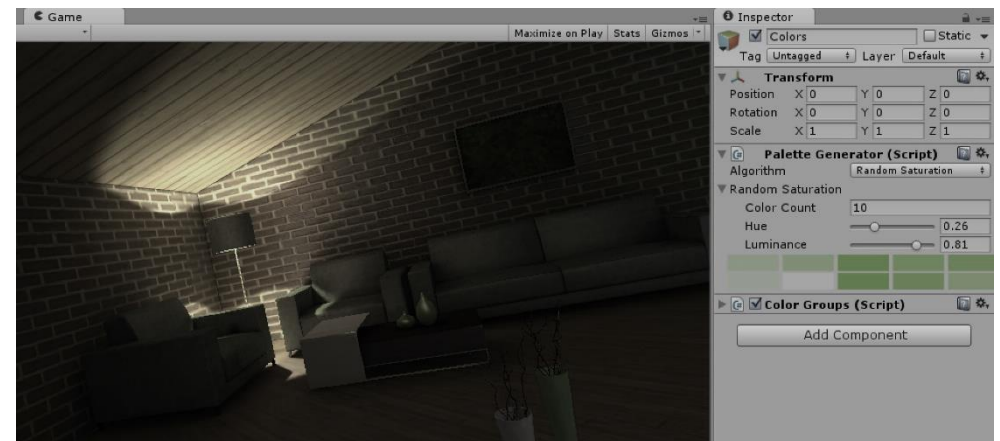
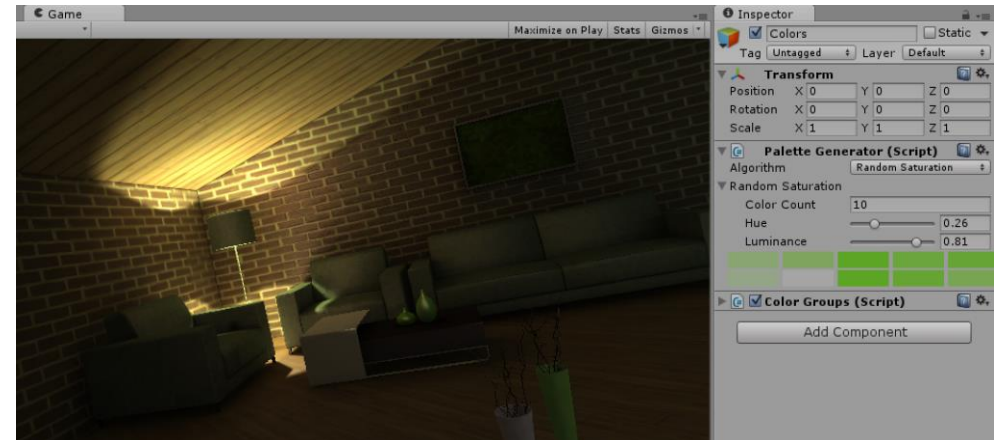
- Select color count
- Select level of saturation
- Select level of luminance



Random saturation

Generates a random level of saturation.

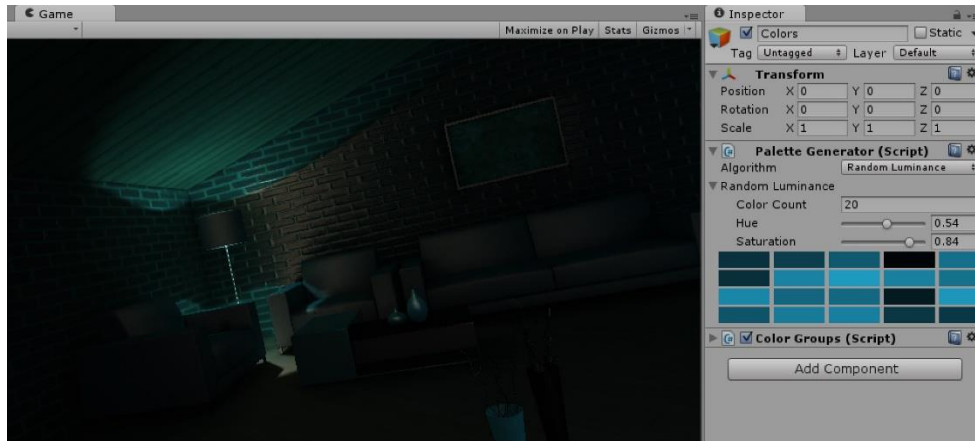
- Select color count
- Select hue
- Select level of luminance



Random luminance

Generates a random level of luminance

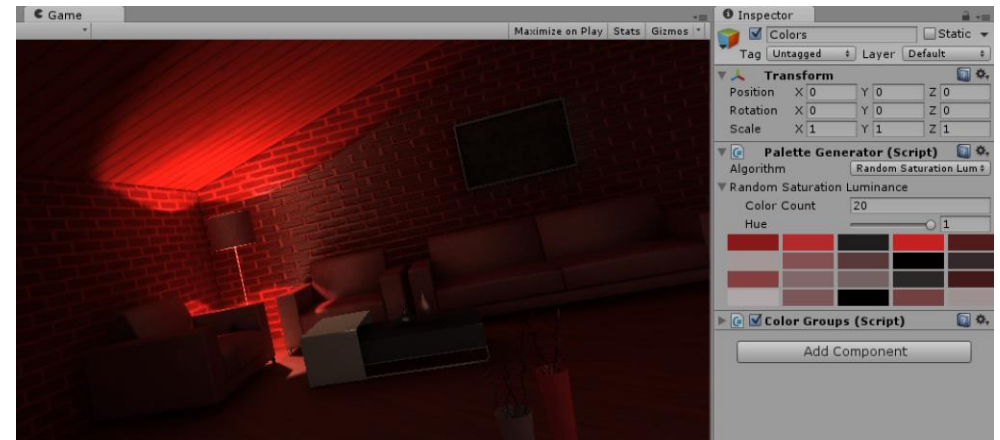
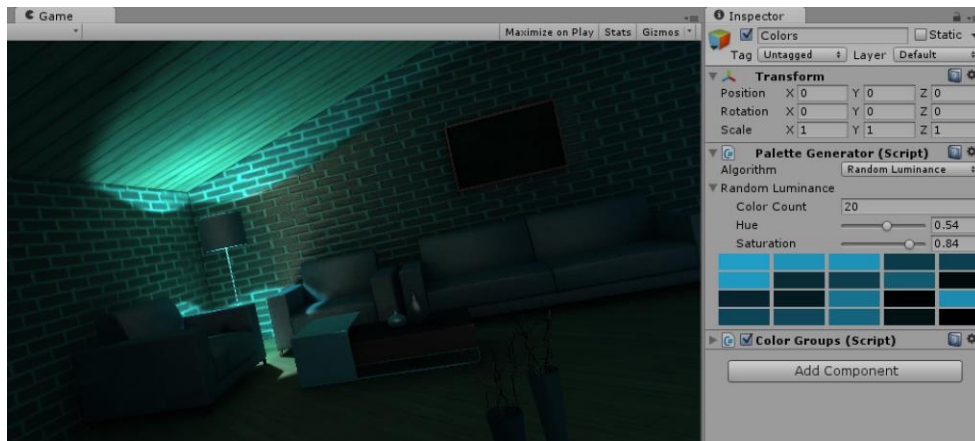
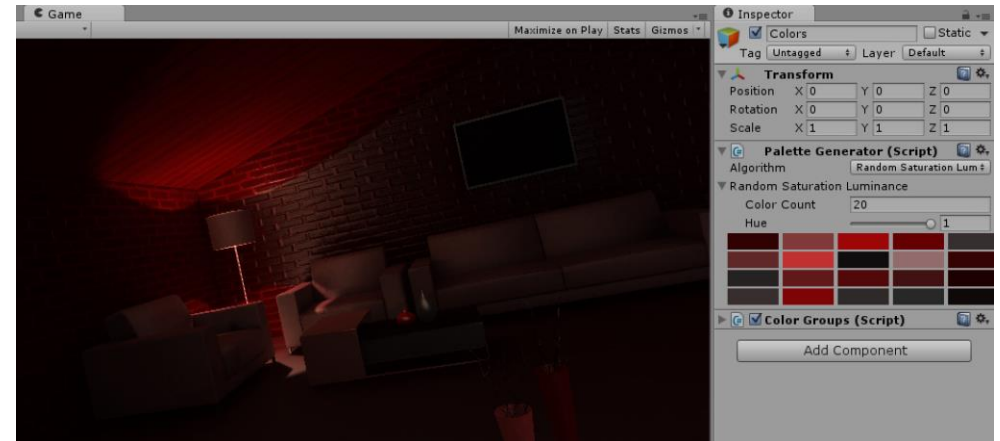
- Select color count
- Select hue and level of saturation



Random saturation and luminance

Generates a random level of saturation and luminance

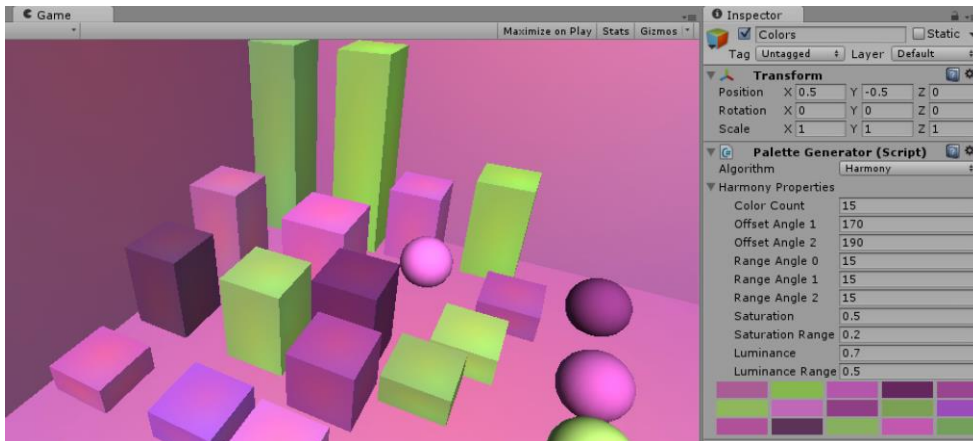
- Select color count
- Select the hue



Harmony

A color is randomly selected off the color wheel. You can select two angles to offset from that color, resulting in two additional colors. After choosing a range for each of the three colors, the palette is randomly selected with in those selected ranges of the three colors.

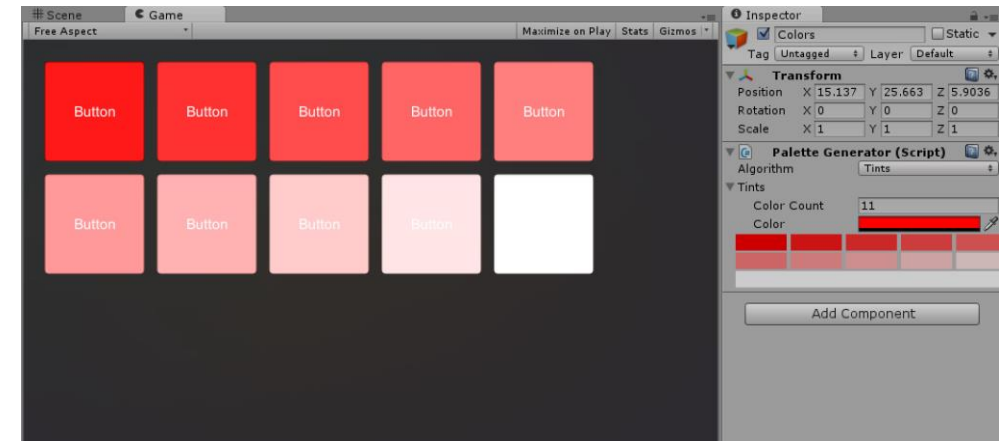
- Select color count
- Select offset angle for color 1 and 2
- Select range of angle for color 1, 2 and 3
- Select level of saturation
- Select range of saturation
- Select level of luminance
- Select range of luminance



Tints

Generates a palette with a varying level of white in a selected color.

- Select color count and color



Shades

Generates a palette with a varying level of black in a selected color.

- Select color count and color

