Contents

If there are any issues with this asset or you have a question, then feel free to reach out through email: dreamnoms@gmail.com

Overview

Thank you for downloading Chibi Character (version 2.0.0)!

This package contains an animated character, textures, custom shaders, and two example scenes. Read below to learn how to get started.

Folder Breakdown

Built-in

Contains shaders that are compatible with Unity's Built-in render pipeline. If you are using URP, then this folder and all its contents should be deleted.

Editor

Contains a script needed for setting up the character. Don't delete this folder.

Essential

Contains the model, animations, and textures needed for the asset to work. Assets in this folder are compatible with both the default renderer and URP.

Example

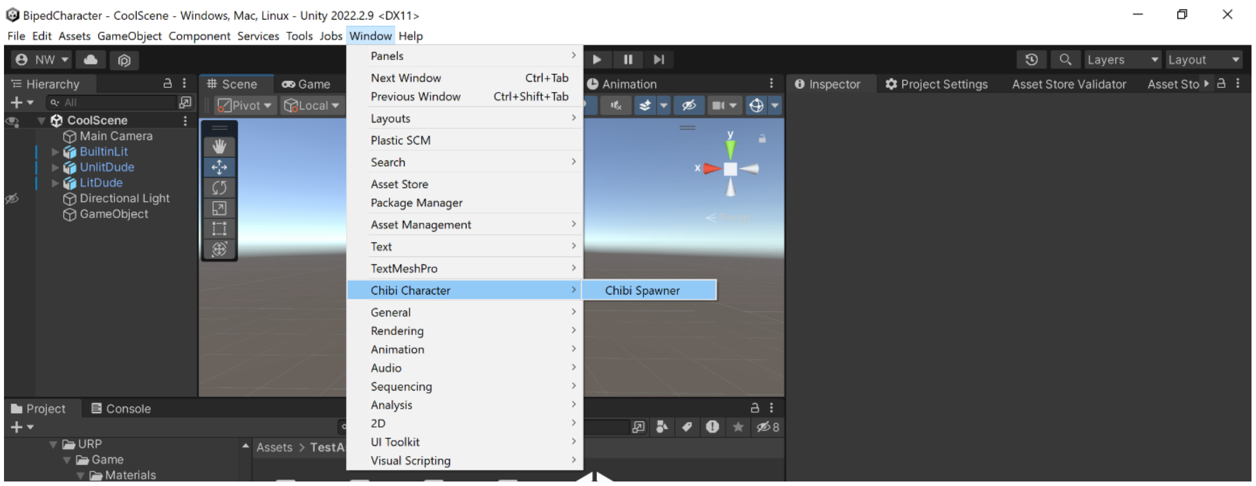
Contains example assets and scenes. The Example folder is optional and can be deleted if you don't need it. The Builtin folder has assets specific to the built-in render pipeline, while the URP folder has assets specific to URP. The shared folder contains assets that are used by both the URP and Built-in example scenes (eg animator controllers and scripts).

URP

Contains shaders that are compatible with Unity's Universal Render Pipeline. If you are using Unity's Built-in render pipeline, then this folder and all its contents should be deleted.

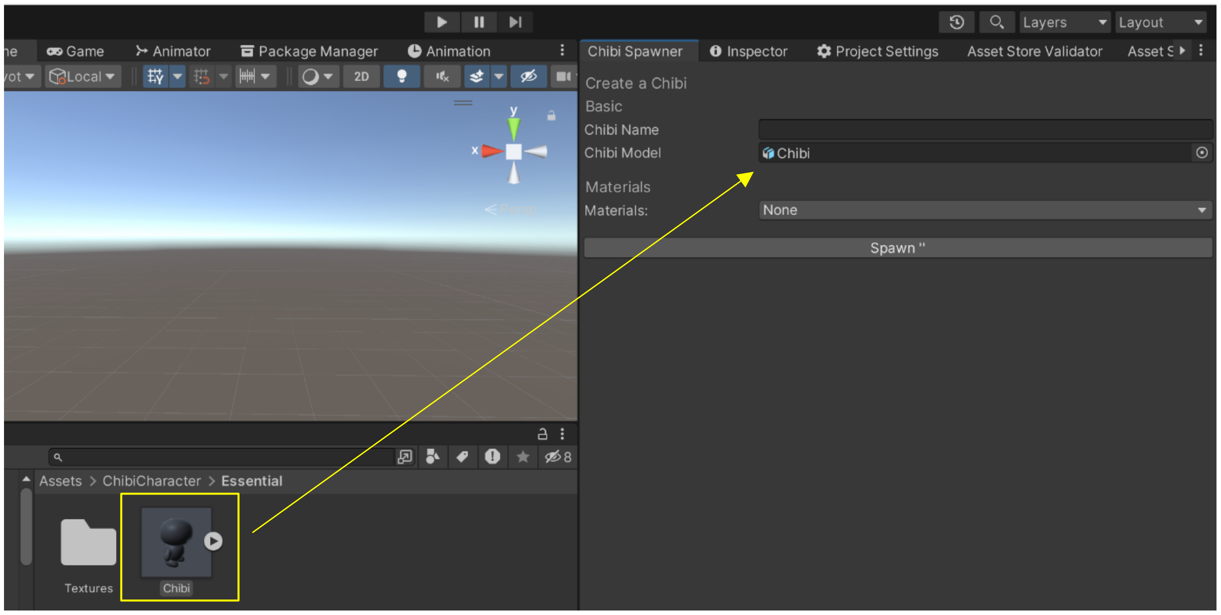
Adding the Character to the Scene

The easiest way to add a chibi character to your scene is through the “Chibi Spawner” window (Window>Chibi Character>Chibi Spawner)



The Chibi Spawner window is where you can change the settings for the chibi character before spawning it in the scene.

The most important value to set is the Chibi Model. This value should be set to the Chibi.fbx in the ChibiCharacter>Essential folder of the project:



Create a Chibi
Basic
Chibi Name
Chibi Model: Chibi
Materials
Materials: None
Spawn "

Once you select the correct Chibi Model, you will have further setting to edit. These settings are explained below.

Chibi Name

Determines what name to give the model when it is spawned into the scene. Additionally, if you are spawning a character with Materials, it determines what to call the folder where those materials will be stored.

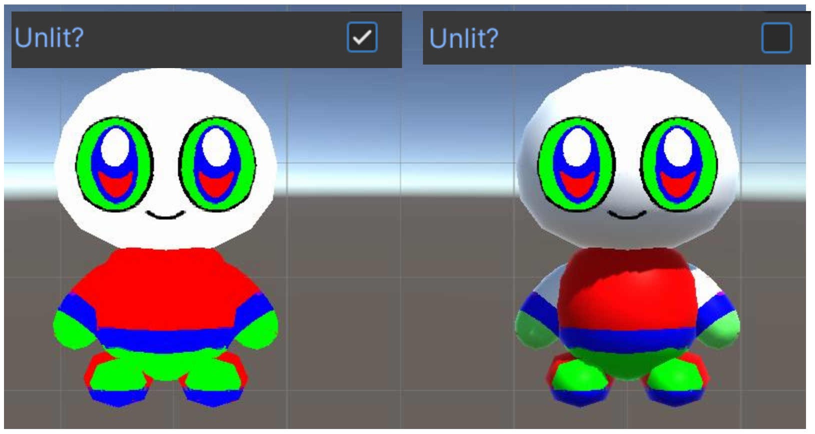
Materials

If set to something other than “None” it will create materials for the character and it will automatically assign those materials to the correct parts of the model.

When spawning a character with materials, new folders will automatically be created in the project to hold the materials. These folders will be created in your current location in the Project directory, so it is a good idea to navigate to a good folder before spawning a character that has materials.

Unlit?

Determines whether materials should be Unlit or versus Lit:



Textures

The Textures section allows you to set up textures for the character. It is fine to leave this section blank and change the textures on the materials afterwards.

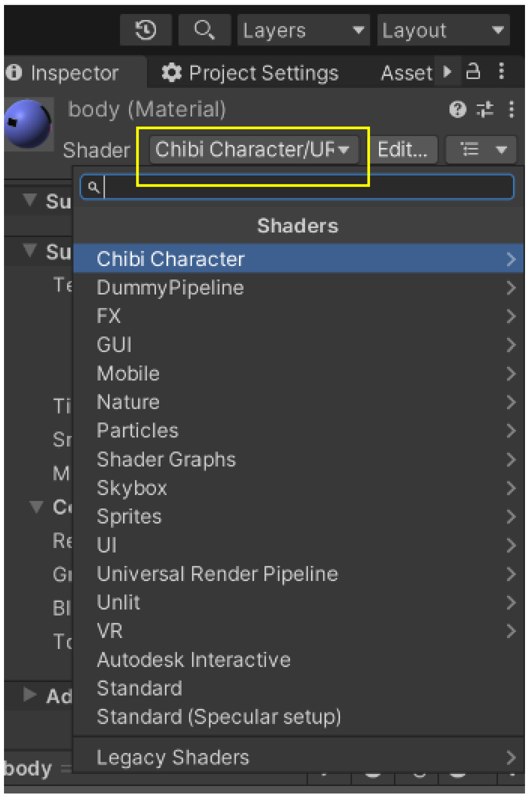
Once you are satisfied with the spawn settings, you can hit the “Spawn” button to place the character into the current scene.

Customizing the Character

This version of Chibi Character comes with its own shaders to offer greater customization over the outfits and colors.

Shader Options

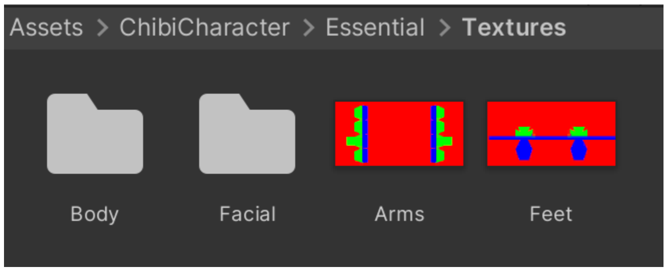
If you want to use one of these special shaders, you can select it from the "Chibi Character" section shown when you select the Shader dropdown:



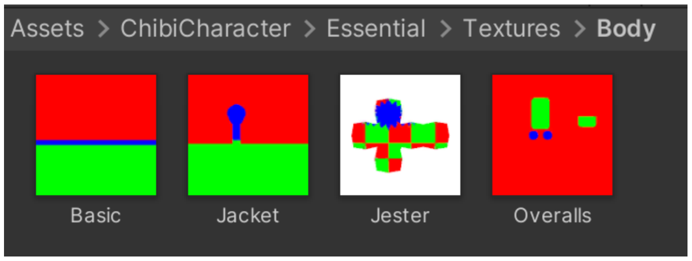
The Eye and Mouth have their own shaders since they have an additional "Expression" parameter, but the rest of the parts can all use either the ColorSwap_Lit or ColorSwap_Unlit shader.

Textures

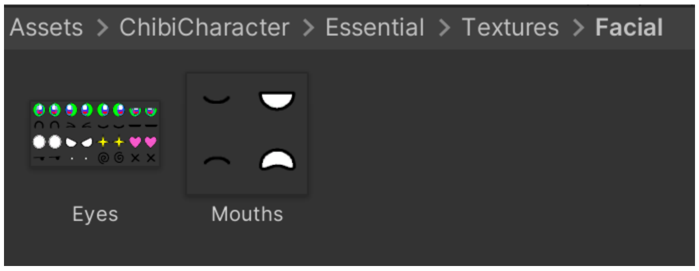
All textures are located in the ChibiCharacter>Essential>Textures folder:



Body textures have their own subfolder, since there are 4 possible body textures:



Additionally, the Facial sub-folder holds the textures for the Eyes and for the Mouth:



You may notice that the textures have strange colors. This is to help with the Color-Swap shader. Red, Green, and Blue are the primary colors of the computer.

The Reds in the texture represent the main color.

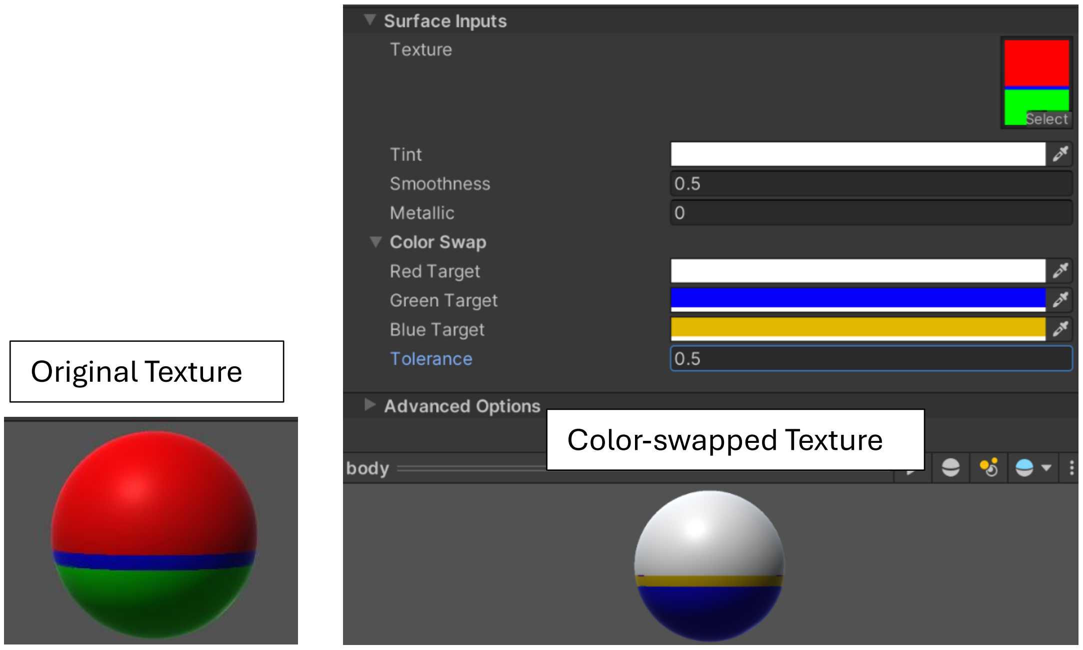
Green represents a secondary important color.

And Blue represents smaller details.

This color representation holds true for most of the textures.

Color Swap

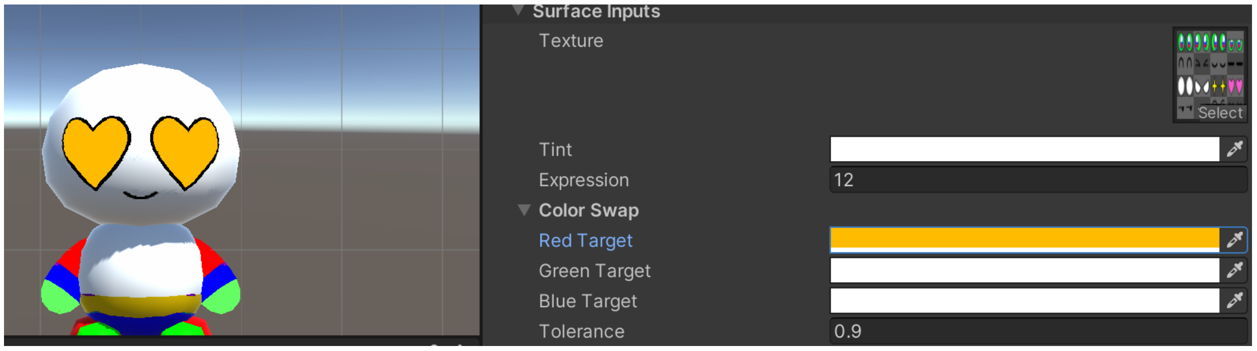
The Color Swap section of each shader allows you to remap the colors of the texture with different ones. The Target colors define the new colors of the image. That is, “Red Target” defines what color to replace the Reds of the image with:



Surface Inputs
Texture
Tint
Smoothness 0.5
Metallic 0
Color Swap
Red Target
Green Target
Blue Target
Tolerance 0.5
Advanced Options

Tolerance defines how effective the swap is. It should be between 0 and 1. Smaller numbers mean the swap is less effective and the original RGB colors of the texture may peak through. But too big could cause colors to overwrite too much.

For example, on the eyes if the tolerance is too high then the Red Target may overwrite the color of the pink Heart expression which is not desirable:



Surface Inputs
Texture
Tint
Expression 12
Color Swap
Red Target
Green Target
Blue Target
Tolerance 0.9

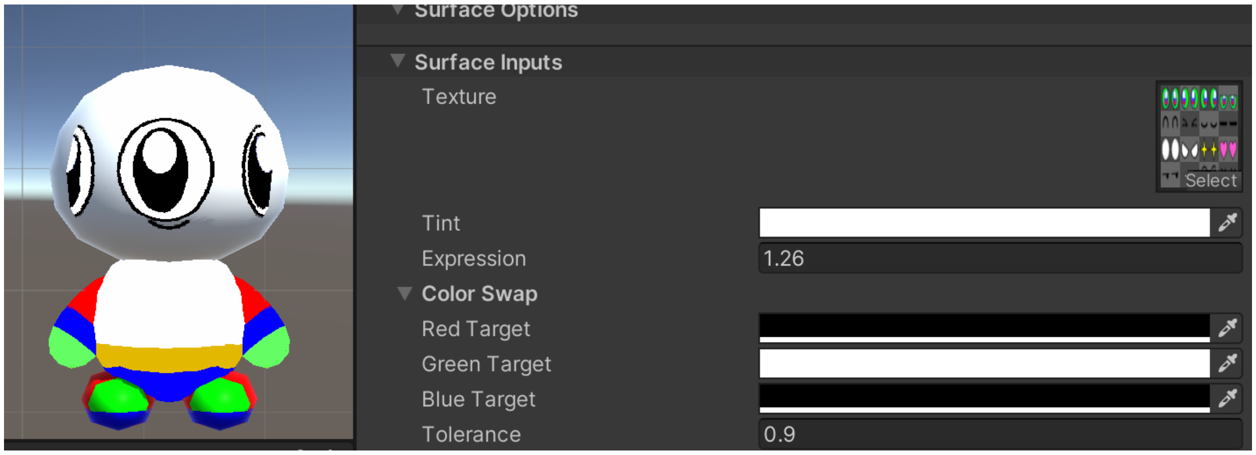
It is important to pick a good balance of tolerance to prevent unwanted artifacts.

Expressions

The mouth and eyes have an additional variable called “Expression” which controls the expression of that part. This expression parameter is an integer that starts on 1.

The 0th expression could be used as a blank expression. That is, if you want the character’s mouth to disappear you could set the expression to 0.

Unfortunately, URP does not have an integer field, so it is possible to set the Expression to a float resulting in strange behaviour:



Surface Options

Surface Inputs

Texture

Tint

Expression 1.26

Color Swap

Red Target

Green Target

Blue Target

Tolerance 0.9

Hopefully this will be patched soon... But for best results when using URP make sure to set the expression to an integer!

Eye

The eye has a total of 16 expressions in a range from 1-16.

1-4: Basic eye expressions using the recolored eye

5-8: blinks

9-12: Open/special emotes

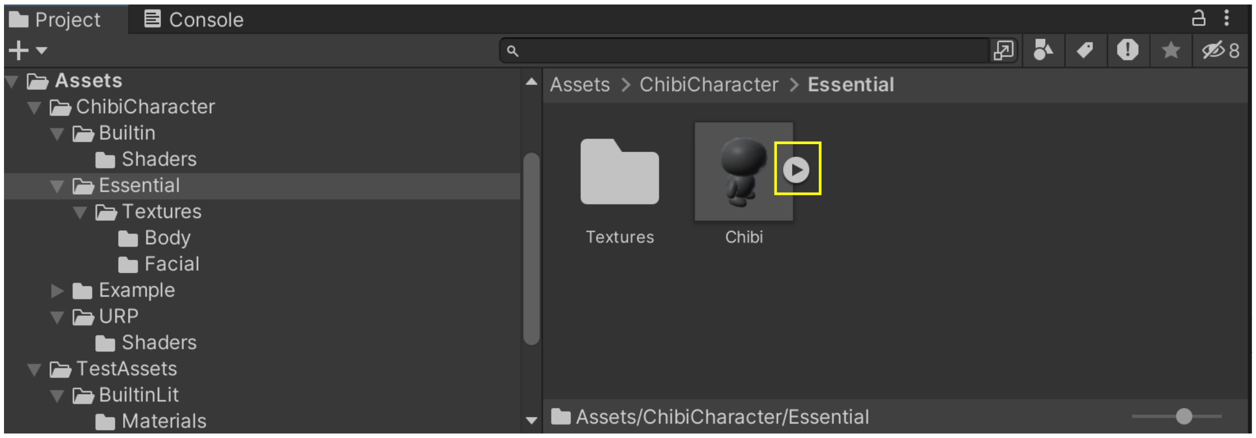
13-16: Other expressions

Mouth

The mouth has a total of 4 expressions in a range from 1-4. Odd expression numbers are closed mouth, while even expression is reserved for an open mouth.

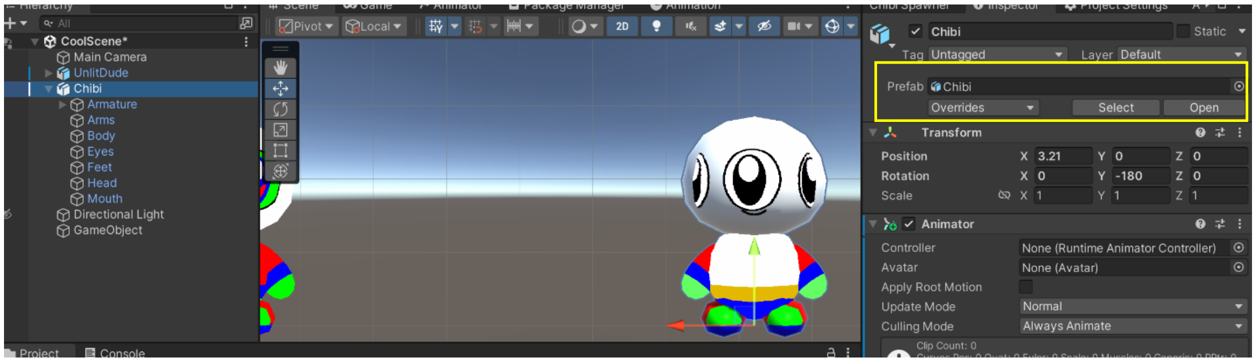
Animations

The animations for the character are all available in the ChibiCharacter>Essential folder. Once you are in that folder, you can press the little arrow next to the Chibi character to expand it and see the possible animations:



Any of the animations can be put on an animator controller.

The animator controller should be placed on the root chibi object and not on its Armature child object. You can be sure that you are placing it on the correct object when the object you have selected shows a Prefab section:



Chibi Static
Tag Untagged Layer Default
Prefab Chibi
Overrides Select Open
Transform
Position X 3.21 Y 0 Z 0
Rotation X 0 Y -180 Z 0
Scale X 1 Y 1 Z 1
Animator
Controller None (Runtime Animator Controller)
Avatar None (Avatar)
Apply Root Motion
Update Mode Normal
Culling Mode Always Animate
Clip Count: 0

As you can see in the picture above, I have added the Animator Controller to this root chibi object.

Examples and Scripting Reference

In the Example Folder there are two scenes: **Game** and **Turntable**.

Game

The Game scene contains a playable platformer level where you can see the character and animations in action. Feel free to look at the CharacterMover.cs script and the Character.controller asset in the Example>Shared folder if you want to see an example of how the character and animations could be set up.

The Character.controller animator and OnExitSetBool.cs script are great assets to keep if you don't want to build an animator controller from scratch.

Note: The Character.controller animator controller does not contain all the animations that the character has, but it does contain a vast majority of them.

Turntable

The Turntable scene is where you can play the various animations and rotate around by dragging on the Rotation Slider.

The RandomizeCharacter.cs script (Example>Shared>Turntable>Scripts), is a great place to look if you want to learn how to change the eye and mouth expressions through script, or change the color and texture of the body parts.

Scripting Reference for Shaders

_RTarget: **Color** that controls the main color

_GTarget: **Color** that controls the secondary color

_BTarget: **Color** that controls the smaller details

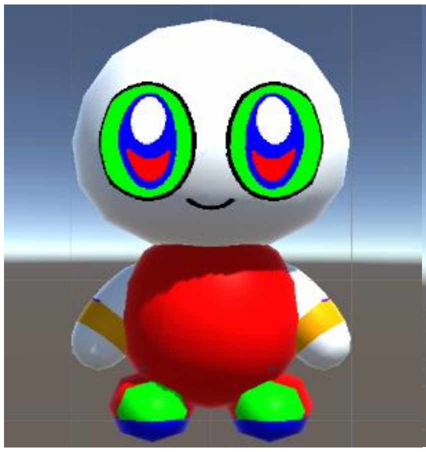
_Color: **Color** that controls overall Tint (same as Material.color)

_MainTex: **Texture2D** that controls the main texture (same as Material.mainTexture)

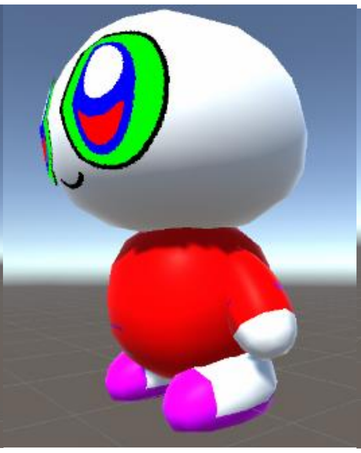
_Expression: **float** (URP) or **integer** (built-in) that controls eye or mouth expression

Character Customization Inspiration

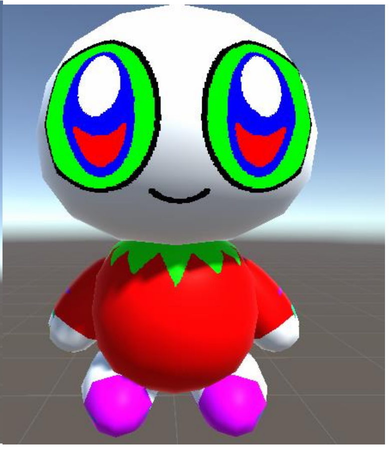
Bracelets



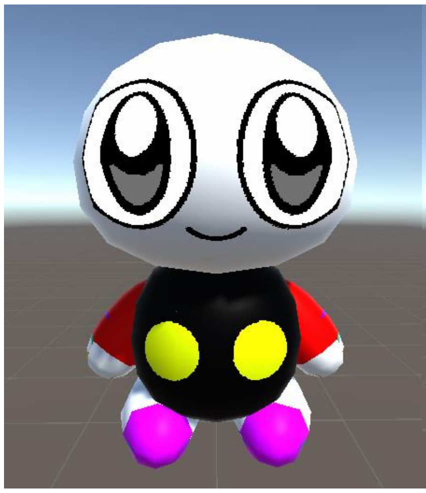
Slippers



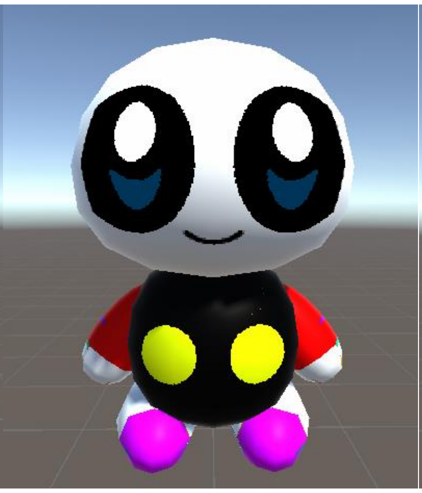
Tomato (using Jester)



Human



Animal



Monster