Basic Documentation for “Modular Stylized Character 2” Package

Models and customization

Method 1

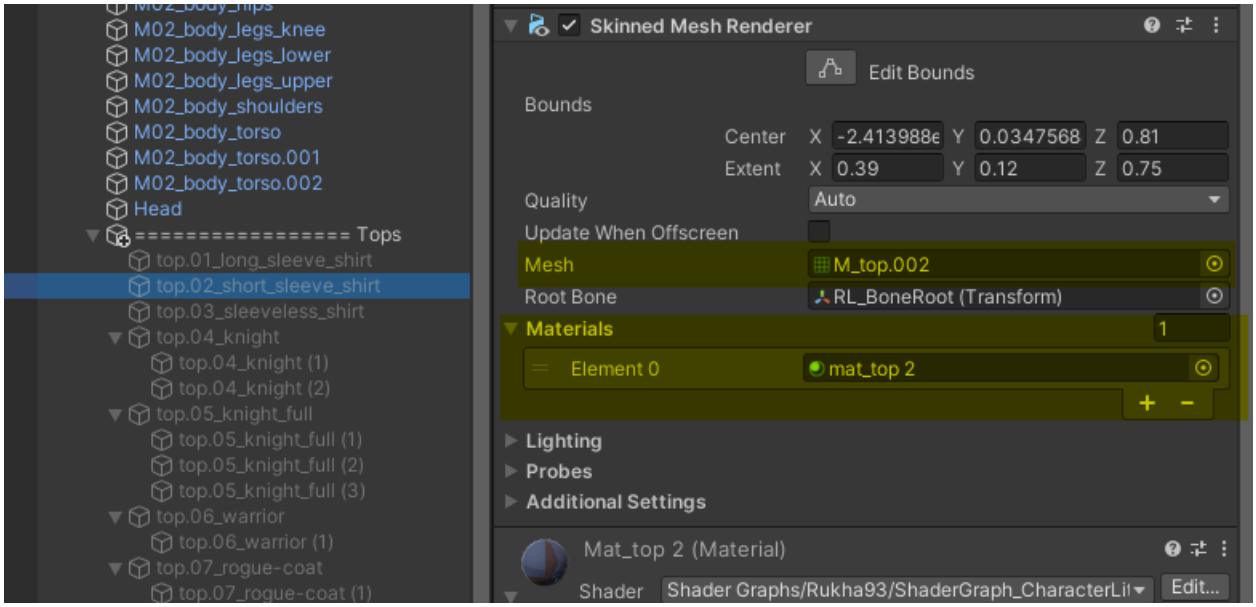
The simplest way to get a custom character ready is to use the “F02_AllOutfits” or “M02_AllOutfits” prefabs. These prefab contains all the customizable pieces inside them, so you can just remove the pieces you dont want from the prefab and save it as a new original prefab.

*If you just disable and don’t remove them, be aware that all the textures and meshes in the prefab will be loaded into the memory when the prefab is loaded to the scene.

Method 2

You can also set the desired mesh and material to the SkinnedMeshRenderer manually. This method is how the “F02_AllOutfits” and “M02_AllOutfits” prefabs were created.
Just duplicate one of the body parts and change the Mesh and Materials in the SkinnedMeshRenderer component.

Duplicating an existing body part is important to make sure both of them are using the armature and bone hierarchy; else the bone hierarchy will need to be initialized through code.

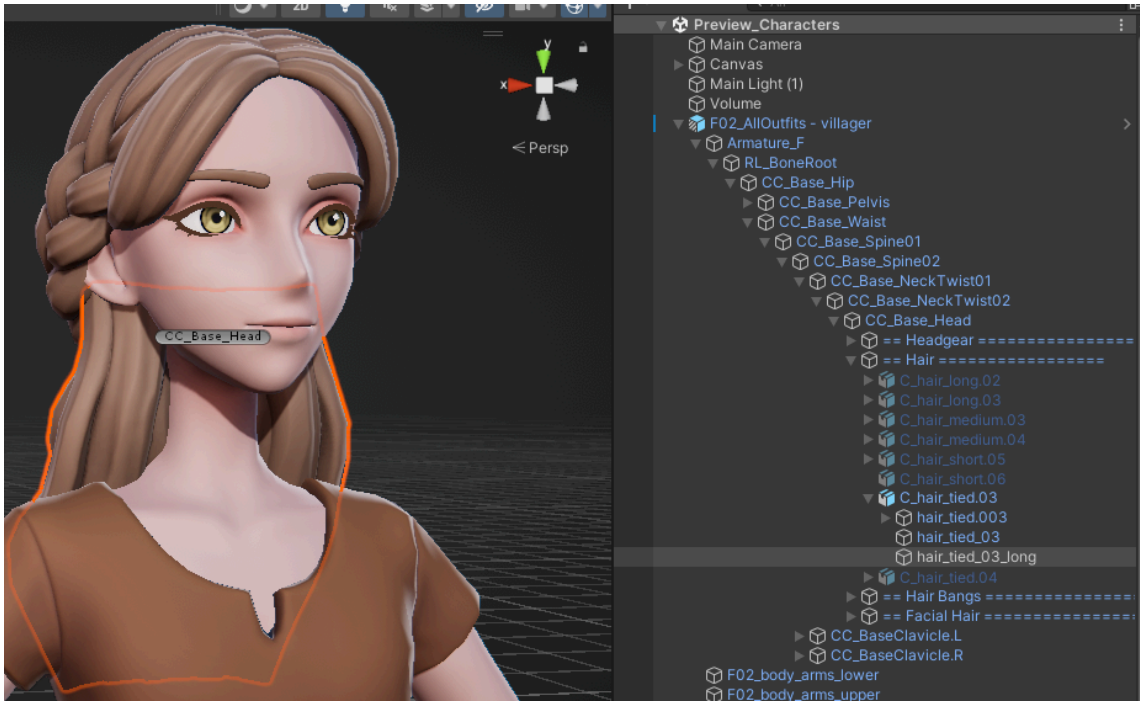


Method 3

The last method is the same as the second but done through script. You instantiate a new GameObject, add a SkinnedMeshRenderer component, set the new mesh and materials and copy the existing SkinnedMeshRenderer propertiers.

Hair and Facial Hair

The hair prefab must be added as a child of the head bone (CC_Base_Head).
Some hair prefabs contains extra meshes than can be toggled for extra customization options.
Some of the hairstyles and facial hair have their own rig so they can be animated with tools like Magica Cloth.



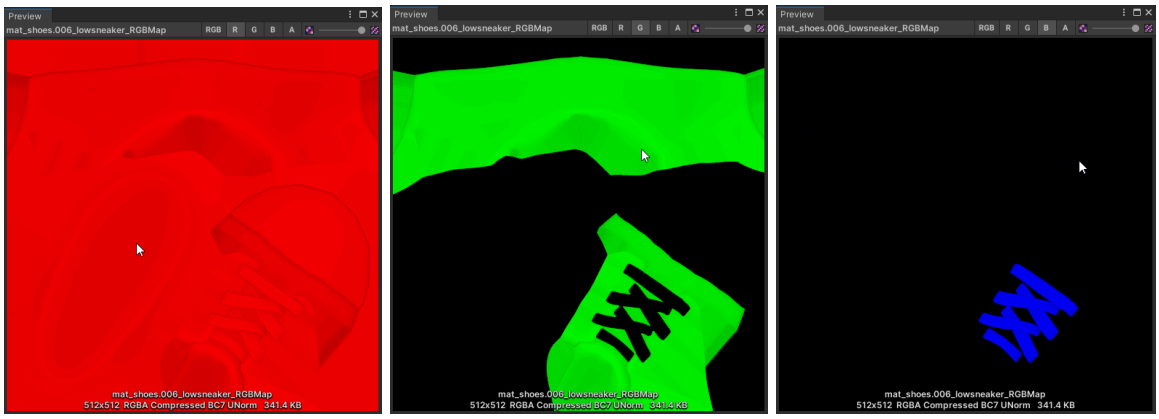
Headgear

The hat and helmet work similarly to the hairstyles, except for some pieces that are must be created like the outfits.
For example, the knight helmet has a piece of cloth that covers the neck and chin, this cloth should be a SkinnedMeshRenderer like the shirt and head is, and it also has blendshapes to match the jaw movement in some facial blendshapes. The other pieces of the helmet are a basic mesh renderer meant to be child of the CC_Base_Head bone transform.

Shader

Overview

The main shader, “ShaderGraph_CharacterLit”, uses a Mask texture where each channel defines an area of the model that can be colorized:



Notice how each channel is not just a flat color, it has some gradients and shading. Thats because the shader will use 2 color for each channel: the first for the darker areas, the second for the lighter areas and blend between them both. And the full black areas work as the mask of which will not be colorized by that channel.

The ShaderGraph implementation for the color customization is in the subgraph file “SubGraph_ColorCustomization”.
It also has a customizable rim light, implemented in the subgraph “SubGraph_Rimlight”.