Shaders added

Color correction
Scrolling texture
Toon shader
Specular shader
Rim shader

Color grading

The color grading scripts are taken from the week 5 slides. There are 2 different scripts, one that puts the LUT and a texture into a material and another that assigns the material to the camera. Color grading works by having a lookup table for a bunch of colors and changing the hue and saturation of the color with a tint, this gives everything in the scene an even spread of tint while not changing the otherall scene too much. This can give different feelings or mood depending on the tint it is assigned to, you could easily create a dark or horror theme by giving it a red tint or a brighter one by giving it light colors like blue or yellow. The way it was done here turns down the lighting a bit and gives it a blue color because the lighting had to be a bit stronger so the other shaders would work well. The color grading turns the scene from an oversaturated and too bright scene into a more acceptable and easy on the eyes view.

Scrolling texture

The scrolling texture is a background that I made in krita with gradients on the top and bottom and just a long circle for the tall hills(?). The scroll was added to the background because if you don’t move at all the screen will be completely static and there aren't any moving components you would see in the normal game like enemies or spinning coins but more importantly it adds more moving parts to the existing gameplay because having static backgrounds with no depth isnt that used anymore. Modern 2d games usually have multiple layers of backgrounds so that when you move around it creates a sense of depth in the world and this is supposed to emulate that, the moving parts make the background not just look like an image but instead a part of the world.
The implementation was done by getting a water and foam texture so it would make more movement in the background but I decided to get rid of the foam because I didn’t want it to move that quickly and having it move slowly while also being a not very exciting image makes it look weird after implementation so I decided to just have the water part. Other things that were added were the scrolling speed for x and y but Y ended up not being used because it wouldn't make too much sense, blend strength was also not used as intended because there was no foam to blend into but it did make the background look a bit better as if it were an actually background rather than a solid image. The creation of the shader itself wasn't too complicated as it was just getting the X and Y scrolling offsets and getting the textures into the material, lerp was used for the blend strength and all that ended in the creation of this side scrolling and tiling on all sides scrolling shader.

Toon shader

The toon shader was put onto the character model, more specifically the parts where Yoshi is green, This model was taken off sketchfab https://skfb.ly/6VWoz the many parts of the body allowed easy access to putting different shaders and materials onto the different parts of the body but the only one I think as needed is just the main green parts. I added the toon shader to that part specifically because I thought that the original model had Yoshi not as bright as he was supposed to be when put into the scene so the toon shader made him look brighter and the reason why I choose toon specifically is because it gives the cartoon-ish style that yoshi would look good in. On top of the toon shader I also added specular lighting and shinness to ensure that he would look good in the light and be bright enough for the focus to be on yoshi, the lighting adds to that cartoon feel and makes him look a bit squishy. There is another yoshi in the scene as a comparison for the difference between the outcomes.
Implementation was done first with the lighting model which gets the dot product for the normals for the diffuse then uses unity built in specular calculation and does a similar thing which comes together after when multiplied with the toon ramp makes the lighting work.
The shader part is just using the toon ramp texture and combining it with the lighting from earlier to give the output of what is seen in game

diffuse shader

This one is a general coin that is seen the most, the reason why there are 2 different coins is because I thought this one would fit the scene better so I decided to make the next coin be worth more. This coin is the general coin because it has more of a shine to it and after looking at both I had implemented the other one looks like it has more importance and so I made I want this one to be a coin worth 1 and the other would be something like 10 coins when collected, in other mario games there are things like 3 special star coins to collect and in things like mario maker there is a coin that could be worth 10 or 50 depending on what the creator wants.
This was implemented by getting metallic and shinness because it fits a coin the best, gets the base texture value and calculates the normals and view direction to give the finished output.

Rim shader

The rim lighting was added to a coin to simulate the rounded edge outline effect you would see from a coin usually. The lighting and properties had to be tweaked a bit before it worked the way I wanted them to but overall it creates the illusion of a coin with an outline. There is a bit of specular and shinness added to it as well for the coin shine.
This one is a surface shader with lambert, the rim is made by calculating the normals and view direction to show the outer edge emphasis and specular is applied using view direction and getting a highlight with the dot product.