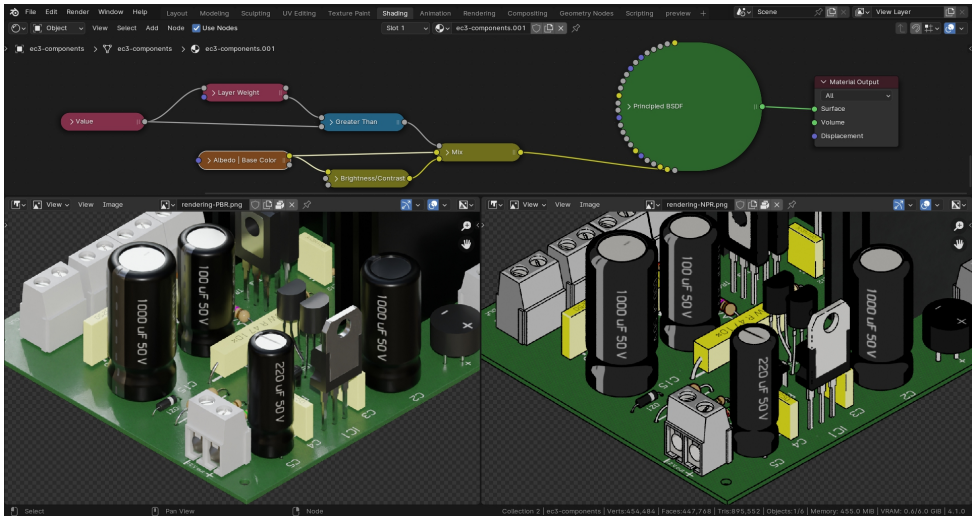


This PDF is available for free; feel free to share it!

[Subscribe to my YT channel](#) & turn on notifications to get updates on new videos!



Hello everyone! In this tutorial, I will describe the node setup I created to achieve an **NPR (Non-Photorealistic Rendering)** from a 3D model with PBR materials and textures. Specifically, we will use a 3D model of an electronic circuit to obtain a rendering similar to the **illustrations in electronics magazines** that I used to read many years ago.

I'll start by showing you the difference between the rendering obtained by importing the 3D model, which already has its PBR Textures connected to a Principled node, and the non-photorealistic rendering, which uses only the Base Color Texture and a suitably configured Principled node.

This tutorial was created in **Blender 4.1**, and the rendering engine used is **Cycles**.

The tutorial will follow this outline:

- we will **disable the existing lighting** in the scene;
- we will make an initial editing of the default inputs of the Principled node;

- we will disable some **Ray Visibility options** to eliminate **Color Bleeding** and other unwanted shading effects;
- we will enable **Freestyle** to provide an outline to the geometries in post-production;
- finally, we will modify the inputs of the Principled node again to create simple **shading based on the orientation of the objects' faces relative to the camera**, making the geometries' appearance less monotonous.

DISABLING THE LIGHTING OF THE SCENE

The first thing to do is to **disable the scene's lighting**.

The initial PBR rendering was created with an HDR Texture on the World Background and a Sun light source, but in this case, the object itself will act as a light emitter to provide uniform shading to all parts of the object.

Therefore, **delete the existing light sources** in the scene and set the **World lighting intensity to 0**.

In a Rendered preview of the 3D view, **we won't see anything**, precisely because **there is no light source in the scene**.

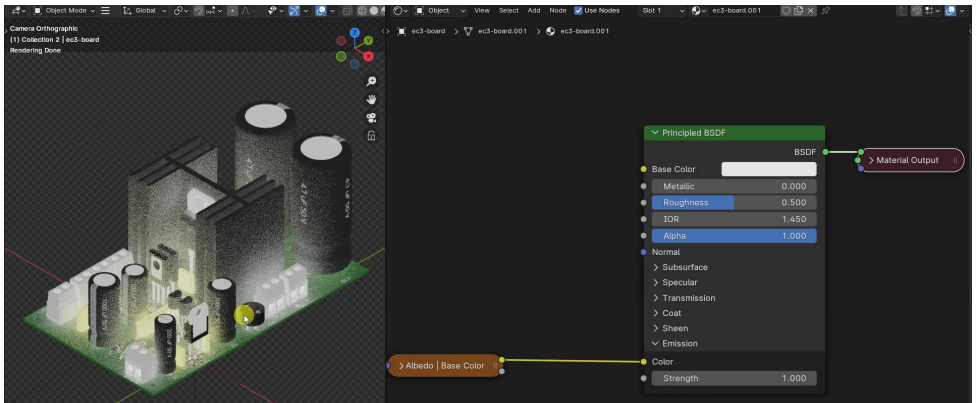
MODIFYING THE PRINCIPLED NODE

Next, we modify the connections of the Principled node. We want to achieve uniform illumination of the objects while preserving the colors and text available in the Albedo Texture.

Disconnect the Albedo or Base Color Texture from the Color input of the Principled node and connect it to the **Emission** input, where we can keep the emission intensity at 1.

The **Metallic** and **Normal Textures** are not of interest to us, so we can **remove** them.

We also **remove the Roughness Texture** and set a uniform **value of 1** for this parameter in the Principled node.

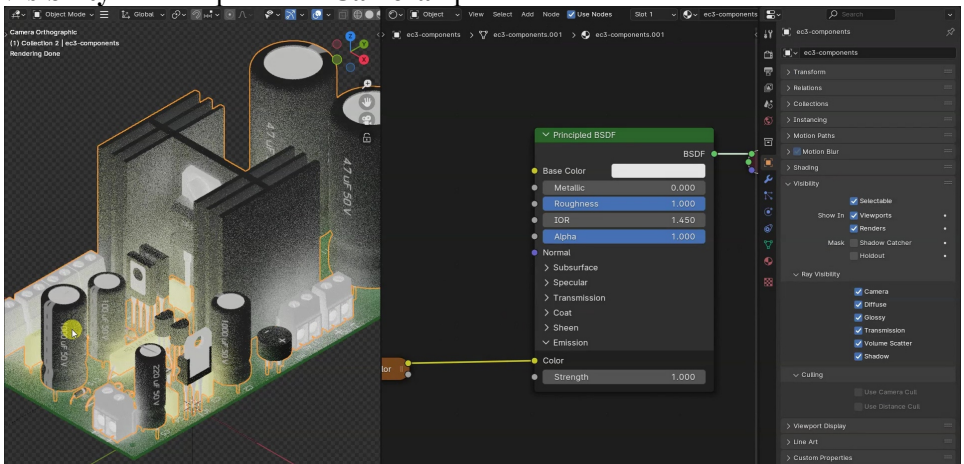


These operations need to be performed **for all the Materials available in the scene**, which in this case are two: one for the printed circuit board and one for the electronic components.

In a Rendered preview of the 3D view, we will now see all elements clearly with **flat lighting**. However, at this moment, the lighting is too uniform, there are colored reflections on nearby objects (that is: **Color Bleeding**), and there are no Freestyle outlines. Let's solve these problems one by one.

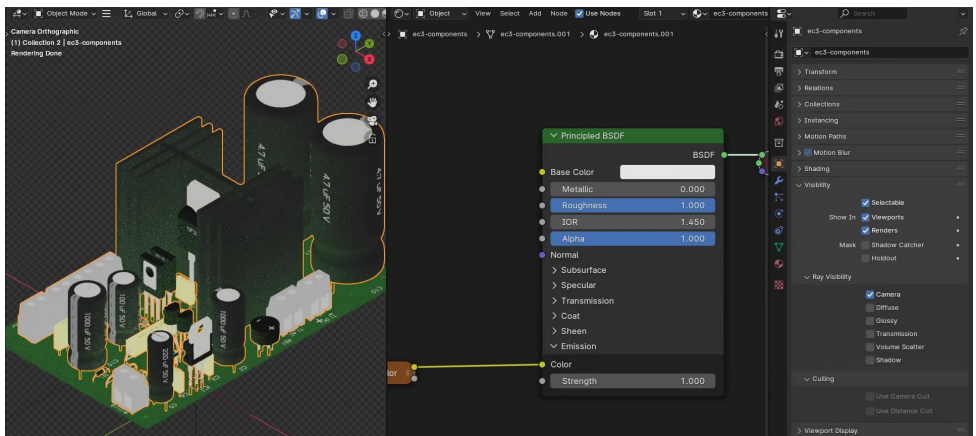
COLOR BLEEDING: DISABLING RAY VISIBILITY

Open an **Object** tab in the **Properties** editor for one of the objects in the scene and access the **Visibility** section. In this section, **disable all options in the Ray Visibility** tab except for the **Camera** option.



This operation also needs to be performed **for all the Materials** present in the project.

This way, the only light rays present in the scene are those emitted from the surfaces and directed towards the camera, so we will **no longer have the annoying Color Bleeding effects** that were visible until a few moments ago.

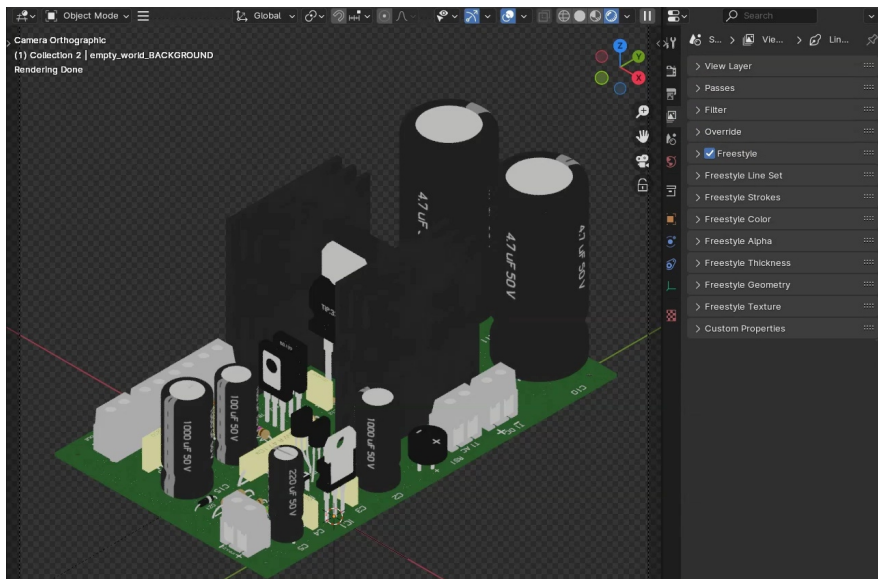
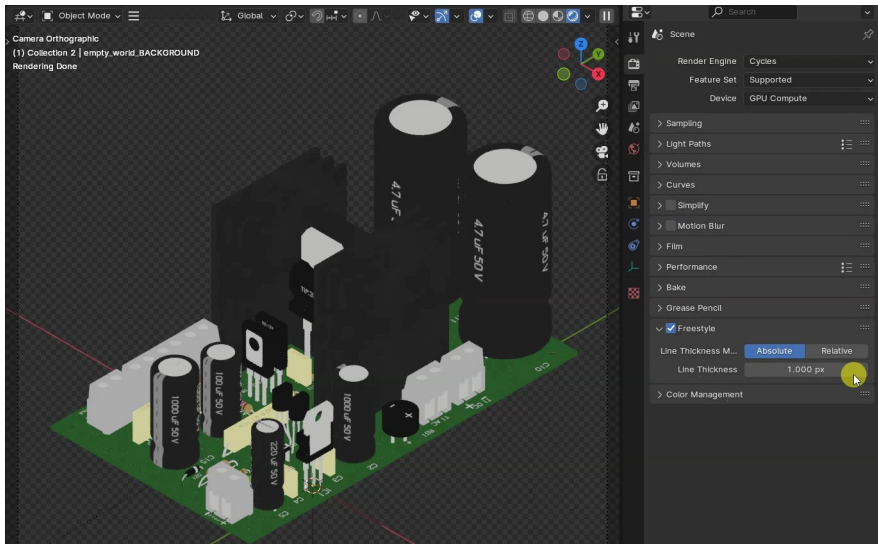


ACTIVATING FREESTYLE

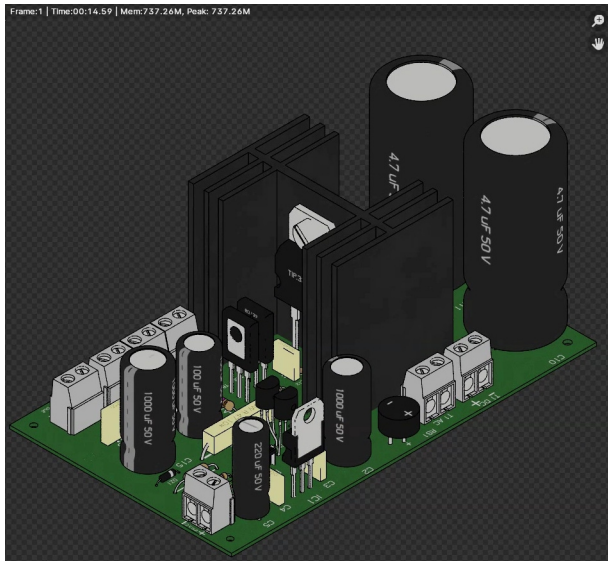
In the Rendered preview of the 3D Viewport, we have a result that is already interesting by itself, but it can be improved by adding **some lines to highlight the contours of the objects**.

The **Freestyle** module is perfect for this. We can enable it in the specific section of the **Render** tab within the Properties editor. In this tab, we can also set the **line thickness** implementation mode and, in the case of absolute mode, the line thickness in pixels.

In the **View Layers** tab, there will now be a **Freestyle** section where we can set many other parameters, but I will not go through them now. The rendering I showed you at the beginning of the tutorial was done simply by activating Freestyle and leaving the default settings.



However, Freestyle is **applied to the finished rendering**, so its effects are not visible in the Rendered preview of a 3D Viewport. Therefore, **let's start an initial rendering** to observe the result achieved up to this point.



ACHIEVING SHADING BASED ON FACE ORIENTATION

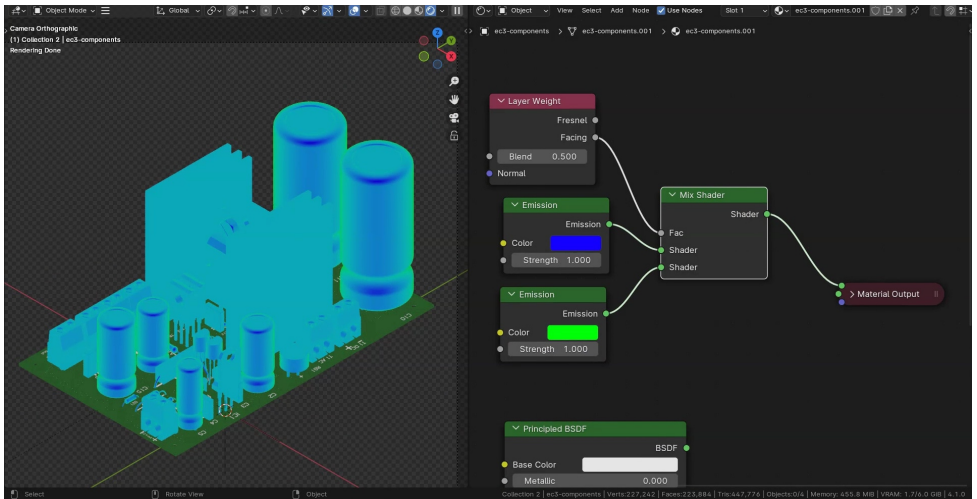
To achieve shading based on the orientation of the object faces **relative to the camera**, I will use the **Layer Weight** node in **Facing** mode as the **mixing factor** between two different versions of the Base Color Texture.

Before implementing the final node setup, I'll show you the effect of the Layer Weight Facing node by using **a couple of Emission nodes** with very different colors, connected to a **Mix Shader node**, and using **Layer Weight Facing** as the mixing factor.

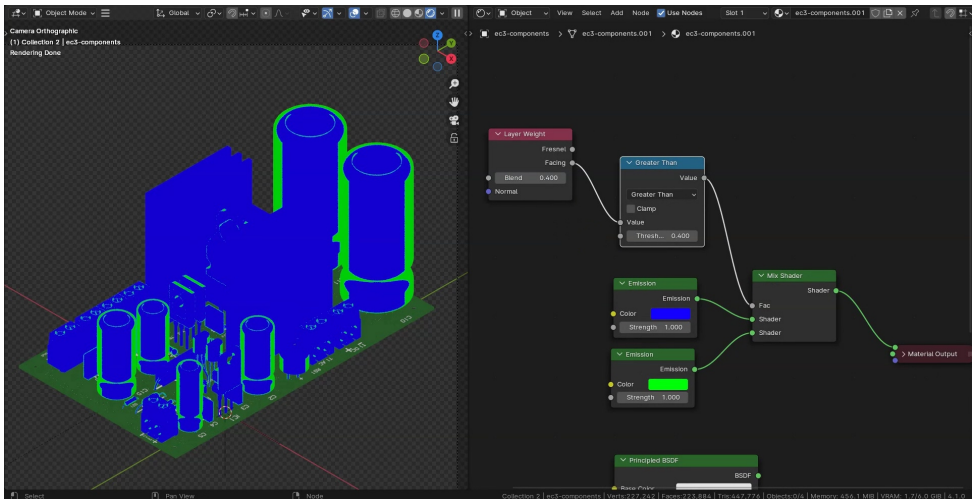
Be sure to connect the **Facing** output of the **Layer Weight** node, **not the Fresnel output**, to the Factor input of the Mix Shader node.

As you can see, by modifying the Blend value, we can use **different shaders depending on the orientation of the faces** relative to the camera.

However, the **transition** between the two shaders is **not very defined**. To achieve **a sharper transition**, we insert a Math node in **Greater Than** mode between the Facing and Factor.

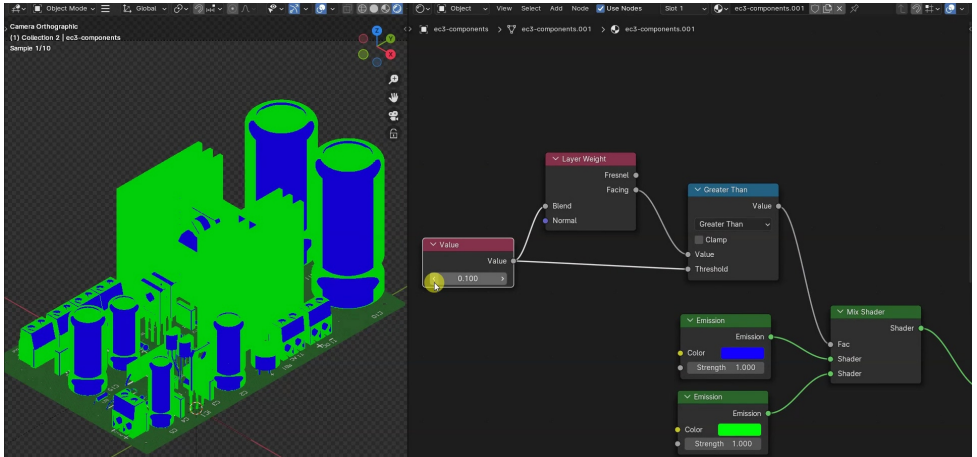


The second parameter of the Math node will serve as **the threshold value to determine the switching** from one shader to another.



To obtain the same value **for both the Blend parameter** of the Layer Weight node and the second Value input of the Greater Than node, we insert **a Value node** from the Input group into the setup and connect it to those two inputs.

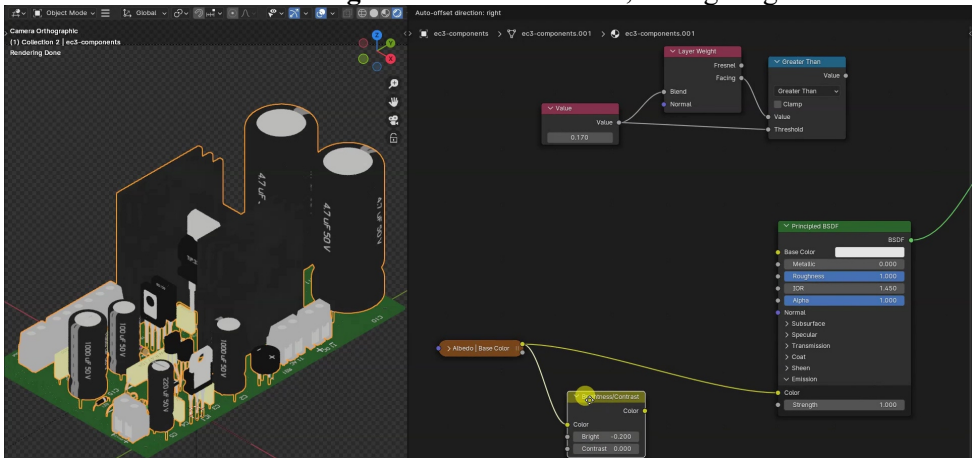
By modifying this value, even with very small variations, we can achieve a good mixing factor to add non-gradient shading to the faces of the objects.



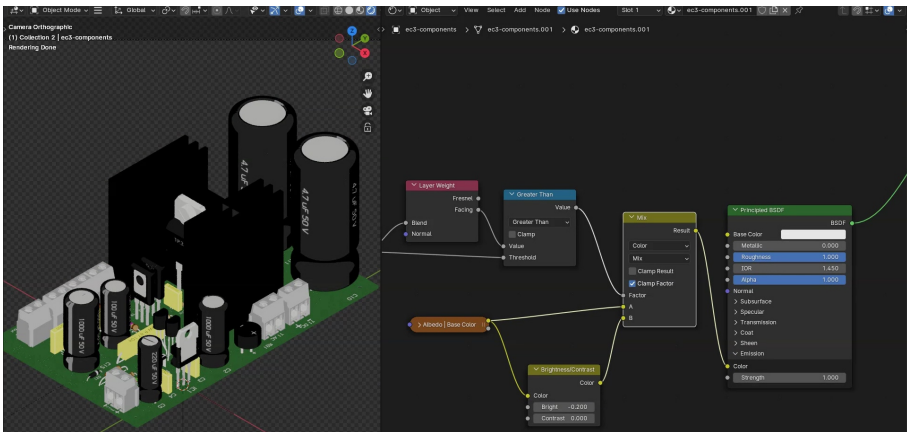
We will use this mixing factor to create a **slightly darker version of the Albedo Texture** based on the orientation of the object faces.

Now, **remove the two Emission nodes** and the **Mix Shader node** from the setup, reconnecting the Principled node to the Material Output.

To get a darker version of the Albedo Texture, connect the **Color output of the Base Color Texture** to a **Brightness Contrast** node, setting **Brightness** to **-0.2**.

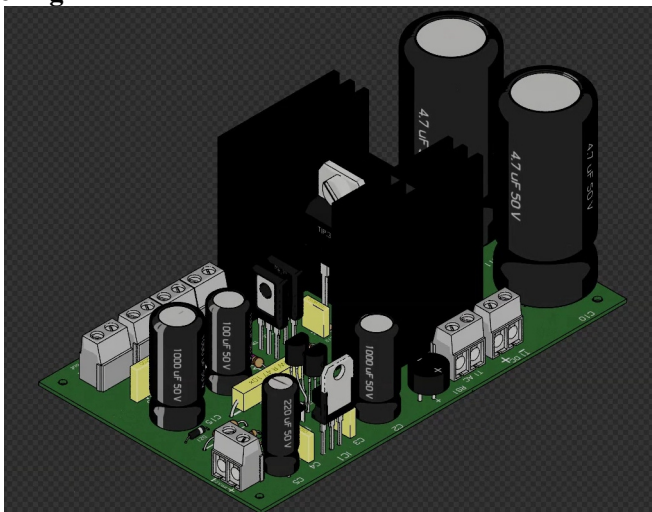


Insert a Color Mix node and connect the Color output of the Albedo Texture to one of the image inputs and the output of the Brightness Contrast node to the other image input. Connect the output of the Greater Than node to the Factor input of the Color Mix node just added to the setup. Finally, connect the output of the Color Mix node to the Emission Color input of the Principled node.



This setup needs to be replicated **for all the Materials available in the scene**. However, the preview visible in a 3D Viewport in Rendered mode is not yet the final one, as **it does not include the Freestyle effect**.

Start a rendering to observe the true final result.



PROS AND CONS OF THE NEW SHADING

The new shading has both positive and negative aspects.

The positive aspect is that, thanks to Freestyle and the Facing node, the objects are no longer too flat, making it easier to perceive their volumes.

The negative aspect is that **this shading changes significantly based on the orientation of the objects** relative to the camera, so it may be necessary to spend some time **adjusting the Value node** or the orientation of the objects.

It should also be noted that the **Greater Than node**, implemented here as a threshold node to sharply separate the color based on the orientation of the faces, might produce **an unappealing result in animations**. In this case, consider the possibility of **removing the Greater Than node**, thus achieving smoother transitions.

Well, that's all for this tutorial! I hope you found it helpful! See you next time!