## **Final 3D Scene Design Reflection**

For my CS 330 final project, I created a simple kitchen counter scene with a cutting board and a knife. I used the skills I picked up throughout the milestones—like shape creation, texture mapping, camera movement, and lighting—and brought them all together in one final project. I also made sure to apply the feedback I received, especially when it came to lighting and overall scene clarity.

### **Why I Chose This Scene**

I wanted to build something that felt grounded and easy to relate to, so I went with a kitchen counter setup. It gave me a good way to show off different shapes and textures while keeping the scene clean and simple. The knife is made of two parts—a box for the handle and a tapered cylinder for the blade—and the cutting board is just a scaled box. These shapes were easy to build but still gave me a chance to use more than one texture and make it look like a realistic object.

I used wood, metal, and handle textures to make each part stand out, and I adjusted the UV scale so none of them looked stretched or pixelated. The cutting board, for example, uses a high-res wood texture that tiles just enough to avoid that “pasted on” look.

### **Lighting and Visual Effects**

Lighting gave me the most trouble throughout the project. In earlier milestones, the scene was way too dark, even though the textures and objects were rendering fine. I fixed this by adjusting the directional light to shine straight down and increasing the ambient and diffuse values. Once I did that, everything in the scene became more visible without looking washed out.

I made the plane (the table) respond to lighting so it would reflect the light properly, but I left the knife parts to just show textures—no lighting—since they’re smaller and didn't need highlights.

### **Camera Movement and Controls**

For the camera, I used the standard WASD keys for forward/back and left/right movement, plus Q and E to go up and down. The mouse lets you look around, and the scroll wheel changes how fast you move. It works like a basic first-person camera, so it feels natural to explore the scene from any angle.

You can also hit “P” to switch to perspective view or “O” to go to orthographic. This was useful for checking alignment and scale during testing, and I left it in so users can look at the scene in both 2D and 3D views.

### **Custom Functions and How I Organized Things**

To keep my code clean, I wrote some custom helper functions that I reused across the project. For example, SetTransformations() handles scaling, rotation, and position in one go, so I didn’t have to write out full transformation chains every time I placed an object. I also used SetShaderColor() to easily change object colors and FindTextureID() to manage which texture goes where.

The main RenderScene() function is broken up by object, with toggles for whether each one uses lighting or just texture. This helped me keep everything readable and easy to update.

### **Final Thoughts**

Overall, I’m proud of how this turned out. It’s a simple scene, but it shows that I understand how to work with shapes, textures, lighting, and camera movement in OpenGL. I had some trouble with lighting at first, but I was able to fix it and make the scene look much better. This class helped me feel more comfortable with 3D graphics programming, and I’m looking forward to building more complex scenes in the future.