OpenGL Final Project Desk Render Reflection & Screenshot

For my 3D scene, I carefully selected objects that would create a small, realistic desktop environment, in attempts to recreate a photo of a desk. I chose a desk, a pen, a notebook, and a mug because these items are familiar and recognizable. I wanted the scene to feel cohesive and real, and I thought about how the objects would interact spatially. For example, the pen lies on the desk, the notebook sits nearby, and the mug is placed slightly to the side. This spatial organization was intentional because it creates a natural composition and makes the scene easier for the viewer to interpret. Each object also allows me to demonstrate different geometry types, such as planes for the desk, cylinders for the pen and mug, a cone for the pen tip, and a torus for the mug handle. Using a variety of shapes helped me show my ability to model multiple object types using OpenGL primitives.

When it came to programming the scene, I had to consider several pieces of functionality. First, I wanted to apply textures to objects to make them appear more realistic. For example, the desk uses a wood texture, the pen uses metal textures, and the notebook uses leather and paper textures. Applying these textures required me to set up texture loading functions, bind textures to the objects, and scale the UV coordinates correctly. I also implemented lighting in the scene. I used multiple light sources to illuminate the objects and create depth and realism. This included a primary directional light, a secondary point light, and a third warm light to simulate a desk lamp. By adjusting ambient, diffuse, and specular lighting values in the shader, I could control how each object reacted to light, and I also added a global ambient component to ensure the darker areas of the scene were visible. This decision was key in preventing the spotlight effect from making parts of the scene completely black.

Navigating the 3D scene is designed to be intuitive. I programmed camera controls so the user can move around the environment and view the scene from multiple angles. Using keyboard inputs, the user can move forward, backward, left, and right, while mouse movements allow the camera to rotate and look around. This setup mimics standard first-person or free-look camera controls used in many 3D applications and games. By decoupling the camera controls from the object rendering functions, I ensured that the user could explore the scene without affecting object positions or scene integrity. This made the experience more interactive and allowed me to demonstrate both rendering and user input handling in a cohesive way.

To keep my code modular and organized, I developed several custom functions. Functions like SetTransformations handle object scaling, rotation, and positioning in a single place, which I then reuse for every object. This prevents me from repeating the same code for each mesh and allows me to adjust transformations easily. Another function, SetShaderMaterial, encapsulates setting all material properties at once. By calling this function whenever I switch objects or materials, I avoid repetitive code and reduce the chance of errors. Functions for setting textures and UV scaling also make it simple to apply different textures across objects consistently. Overall, these custom functions serve as reusable building blocks that keep my program clean, make debugging easier, and allow me to expand the scene later without rewriting existing code.

In conclusion, the development choices for this 3D scene were driven by both visual and programming considerations. Object selection, lighting, and textures all contribute to a visually realistic and organized environment. At the same time, the programming structure, including camera controls and reusable functions ensures that the scene is interactive, maintainable, and flexible. Each decision, from object placement to function design, was made to create a scene that is both engaging for the user and well-structured from a coding perspective. Not to mention, I enjoyed recreating (or attempting at least) a real picture and I learned a lot while doing so.

Here's the final render, you can see the pen and also the edges of the mug/notebook have an extra bright/warm light on them illustrating the lamp.

A screenshot of a computer

AI-generated content may be incorrect.