**Justify development choices for your 3D scene.**

I chose to create an image that contained a desk that held four distinct items. These items were a book, lamp, coffee mug, and a computer monitor. The reason I chose these objects is because It resembles my own work desk at home. It is a sight I am familiar with due to my many nights completing schoolwork.

For the desk object, I chose to use a box mesh. I found this choice to be effective as it provided the scene with a base that had depth. For the lighting and texture, I chose to resemble a wooden environment. This would allow me to create contrast within the scene without a great deal of reflection, as wood in the natural world does not offer much in the way of reflection. This rationale was also used to implement the wall that served as a background. The big difference being that a plane was used instead of a box for the shape mesh.

For the monitor present in my scene I used two different textures. The first is a silver texture that reflects light in the way that metal would. The second is the black screen. I took into account that most screens are made of glass, so the lighting/texture reflected that fact.

The coffee mug that I included in the scene comprises two different lighting and texture combinations. The handle for the mug is a torus shape that is meant to resemble a porcelain-like material that is commonly found in most finished pottery. It is highly-reflective to any light that strikes it. The other component of the mug is the cup itself. I chose a cylinder shape that was tall and resembled a cup. The texture and lighting combination I chose was similar to my implementation of the desk’s texture. The mug’s texture and lighting did not have a glossy appearance like the handle, but rather a matte clay-like finish.

For the book object, I used three box shapes. The outer boxes are of the same dimensions as they represent the front and back covers of the book object. The box between the two covers are the pages. The box is slightly smaller width-wise to demonstrate the disparity in width between the three components. The lighting and texture were both fabric-like. Meaning that they did not reflect much light and have a matte texture.

The lamp in the image is composed of two cylindrical shapes and one cone shape. The two cylinders act as the base. The first is short and wide, while the cylinder sitting atop of the base is narrow and tall. The lamp shade is a cone that gradually tapers as it reaches its top top point. The lighting and texture combination is a mix of previously implemented ones. The two cylinders utilize the metal texture and lighting implemented in the monitor’s base and bezels, so they can reflect light properly. The lamp shade utilizes the texture and shading combination seen in the book cover and pages. It does not reflect any incoming light.

I programmed these implementations using the APIs provided by OpenGL. This streamlined development, as it provided a high-level implementation of the following concepts: lighting, texture, shading, scale, and position. On top of this, I chose to re-use texture and lighting, as I chose objects that behaved similarly when exposed to light.

**Explain how a user can navigate your 3D scene**

There are many ways a user can navigate my 3D scene. The first grouping of key movement are the WASD keys. The W key handles zoom-in, the A key handles movement to the left, the S key handles zoom-out, and the D key handles the movement to the right. The Q and E keys handle vertical movements. The Q key handles upward movement and the E key handles downward movement. The mouse’s scroll wheel allows the user to customize the rate of speed the movements will be performed.

**Explain the custom functions in your program that you are using to make your code more modular and organized.**

I used custom functions for three features. These three features were the texture, lighting, and materials. These methods were modular and reusable. This meant that I was able to make a function call and initialize the values within these functions. It then allowed me to pass in the lighting material and texture values into the SetShaderTexture and SetShaderMaterial methods. It made texture and lighting material usage easy to reuse. The objects themselves were rendered in the RenderScene method. I chose to hard code them all in.