Dylan Olthoff, Rylan Casanova

CST-310

Professor Citro

##### **Project 2: Your Surrounding World (Documentation)**

In this project, we were tasked with getting 6 different pictures from 3 different scenes(2 per scene), of scenes that we would want to replicate in openGL. Below we have our scenes which include a wide variety of different objects that could be interesting to replicate in openGL. With whichever scene gets selected to be our official scene, we will be committing to that said scene for the rest of the projects so it is very important that we pick our scenes carefully. Down below are our included 6 different pictures of our 3 different scenes.

***Everything in this document is our own work***

**Scene 1:**





This scene was chosen as one of our options because it has a lot of shapes that could be easily made using OpenGL while there are also some other complex shapes included which would be more of a challenge to replicate. Some examples of the shapes that would be easy would include the desk itself because it is simply made of mostly squares and rectangles. Some of the harder objects to replicate might include the keyboard because it has so many pieces or the circle stuffed animal on the desk by the speaker because it is such a unique object.

**Scene 2:**





Scene 2 includes both easy and difficult to replicate shapes. For instance, the table and chairs are mostly geometric in shape, made up of rectangles and lines with little curvature. However, the plates and paper towel roll are more round in shape which may be more challenging to replicate using OpenGL - but not impossible. We also added in some cups to the scene which will help add some color. Lastly, the lighting in this scene will be mostly easy to replicate, with only a few cast shadows and some tint on the chair cushions and silverware.

**Scene 3:**





Scene 3 includes more geometric-shaped objects than the previous scene, however the lighting may be more difficult to accurately demonstrate in OpenGL. The shelf, boxes, and TV are all very geometric in shape which should be easy to draw. Also, the same cups are used to add some color to the scene. In the first image, a reflection in the TV can be seen which may prove to be very difficult to accurately depict in OpenGL. In the second image, all of the shelves have unique cast shadows and lighting which also may be tedious to draw. Some of the details, such as the wrinkles in the middle box, the label on the bottom left box and cups, and the texture of the paper towels may take some time to recreate using OpenGL, but shouldn’t be too difficult.

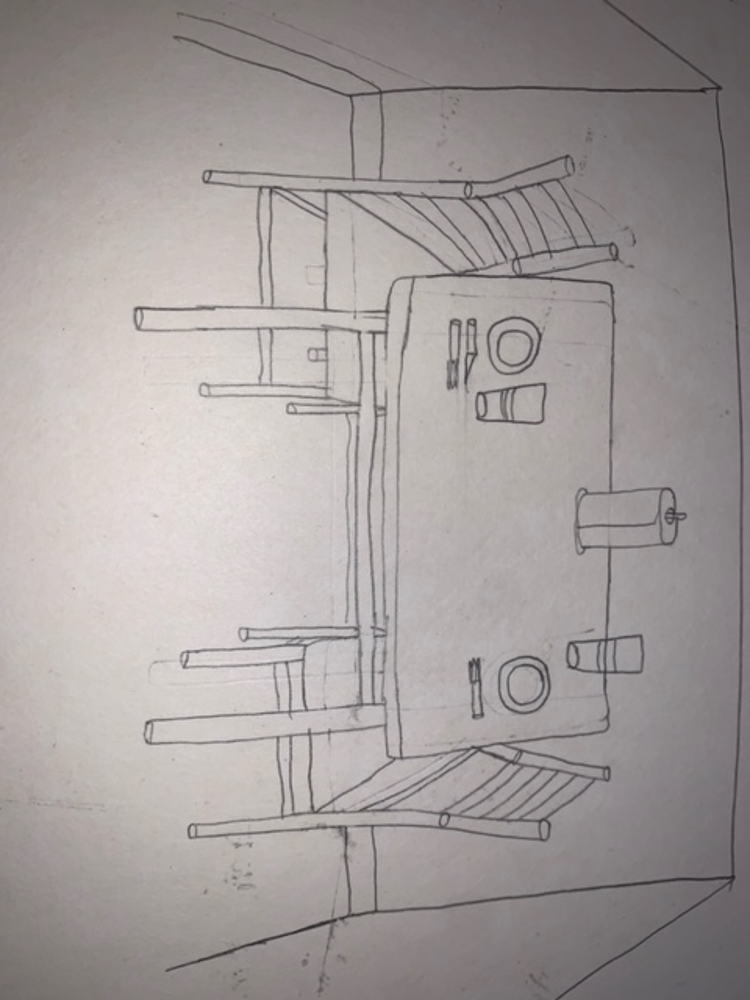
Dylan Olthoff, Rylan Casanova

CST-310

Professor Citro

##### **Project 3: Your Surrounding World (Documentation)**

The Image that got chosen to be our scene for this and the upcoming projects was the first picture from our second scene which is shown below. We were tasked in this project to recreate this scene in a geometrical drawing which can also be seen below right beside the original scene picture.



1. **Listing objects in the foreground, background, and in between**

Objects in foreground:

* Cups (x2)
* Plates (x2)
* Silverware (x3)
* Paper towels

Objects in between:

* Table

Objects in background:

* Wall
* Floor
* Chairs (x2)

1. **Main objects in the scene**

* Table
* Cups
* Plates
* Chairs
* Paper towels

1. **Key characteristics of the scene**

* Light source coming from directly above the scene
* Shadows underneath the table
* Very plain colors aside from the two colored cups

1. **Method for rendering each object in the scene**

Cups:

* Trapezoidal quads

Plates:

* Oblong discs (ovals)

Silverware:

* Long and skinny rectangle with shape on the end depending on utensil:
  + Spoon: circle
  + Fork: three long rectangles + triangles on each end
  + Knife: long rectangle + triangle on end

Paper towels:

* Cylinder with a quad for the towel on the end

Table:

* Trapezoid (tabletop)
* Long rectangles (table legs)

Wall:

* Large rectangle + triangle

Floor:

* Large rectangle + triangle

Chairs:

* Trapezoid (chair cushion)
* Long rectangles (chair legs + back)

1. **Difficulty ranking of each object in the scene (easiest to hardest)**

* Wall
* Floor
* Cups (x2)
* Paper towels
* Table
* Silverware (x3)
* Plates (x2)
* Chairs (x2)

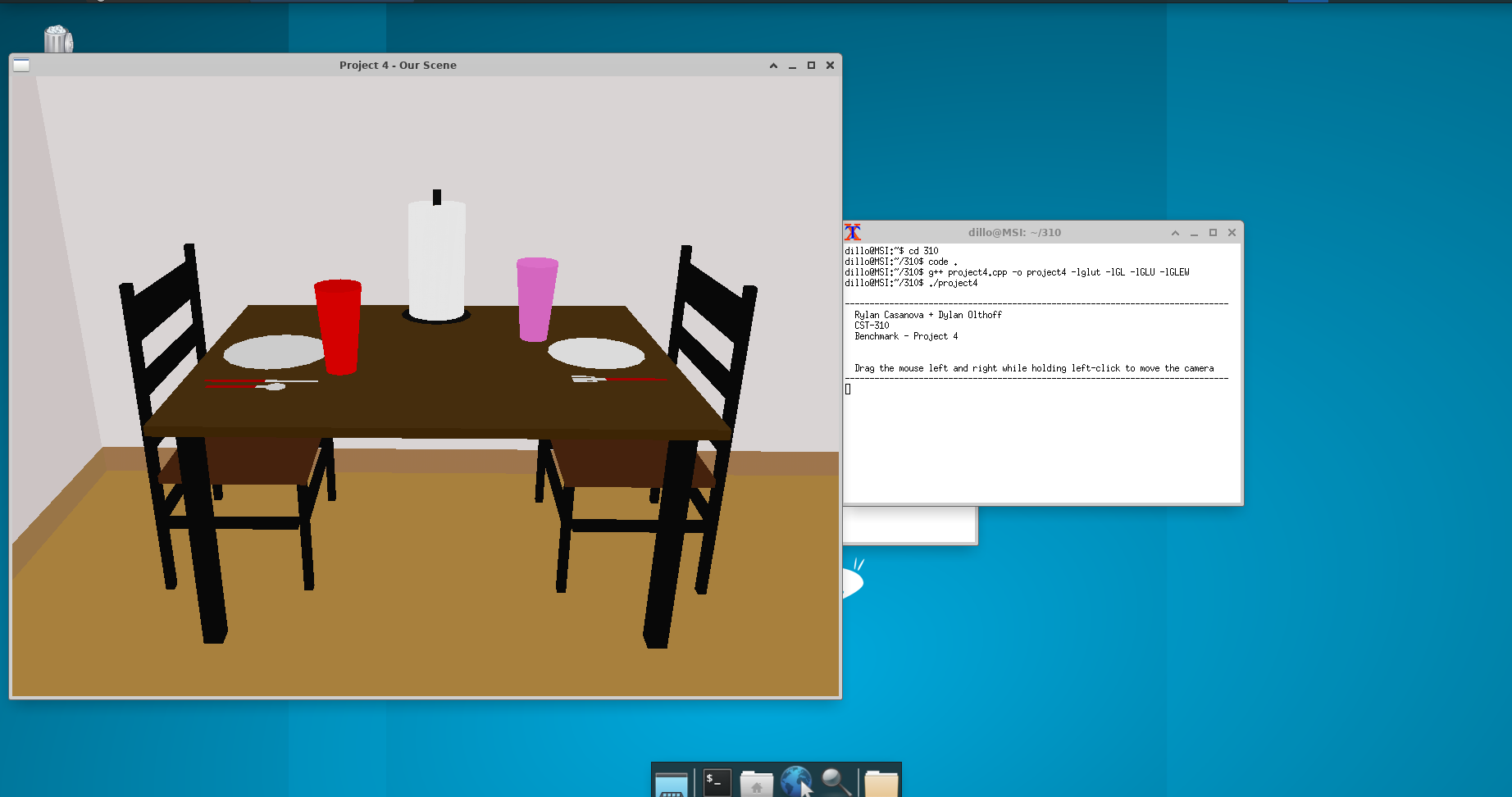
Dylan Olthoff, Rylan Casanova

CST-310

Professor Citro

##### **Project 4: Render Your Scene With Primitives**

In this project we were tasked with recreating the drawn scene from the last project in openGL with primitives. Also, in this project we are also tasked with creating a camera as well for the scene. This camera is used to capture the perspective of the drawn scene and to allow someone looking at the scene to be able to move the camera and look around at the scene at different angles. For our scene we used primitives to create each of the objects and the background, and then we used a camera that can be moved with the mouse to capture the perspective of the scene. We also changed the color of each object to capture the look of the scene even more and to give the scene colors that match the original scene photo that was selected. For each object listed below, there will be listed the mathematical characteristics, the primitives used to render the scene, any transformations that were used, and if any shaders were used for each object created. Shaders are user defined programs that can be used that give developers more control over the graphics pipeline. Two basic types of shaders are fragment shaders and vertex shaders. Vertex shaders are used to manipulate vertices and fragment shaders are used to make the pixels in between the vertices and control how they look.



**Chairs**:

To create the chairs in the scene we used the quads to create each of the pieces for the chair to create what is shown in the scene above. Each quad is made up of four points that are each plotted in the 3d plane using the xyz axis. When they are plotted they then create a full quad using those four points. The chair legs, cushion, and backrest were all made up of quads and colored correctly to match the colors in the original scene. No shaders were needed in order to create the chairs in the scene.

**Table**:

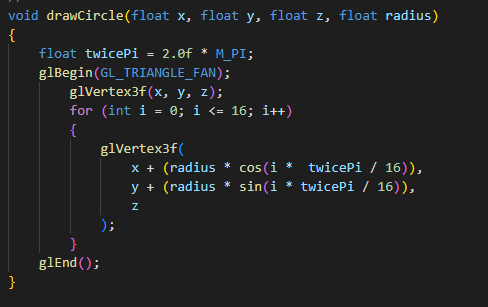
To create the table in the scene, we used quads. Like stated above in how the chairs were made, each quad consists of 4 points that in return create a quad in which we can plot in the xyz plane to create the desired image. The table legs and table top were both created using quads and were both colored correctly according to the image. No shaders were needed in order to make the table.

**Ground**:

To create the ground we used a single quad. This single quad is placed at a base level and everything else will be built on top of it. It was then colored to match the scene. No shaders were needed to make the floor

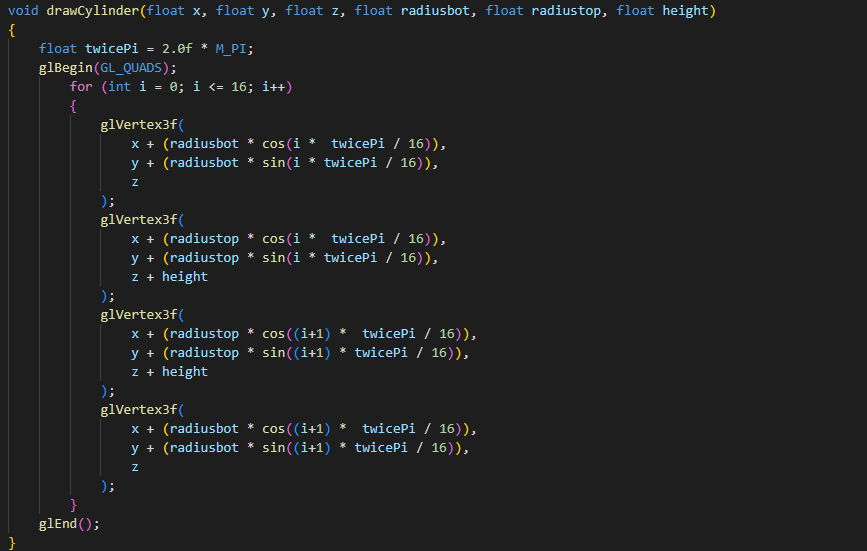
**Walls**:

To create the walls we also used quads. For the wall we created the two walls and then also created the wall trims by also using smaller quads. Both the wall and the wall trim were colored to match the scene. No shaders were needed to create this in the scene.



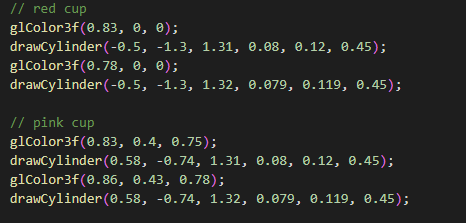
**Plates**:

To create the plates in the scene we created our drawCircle function which takes in an x,y, and z coordinate value and a radius value. In the function we use the primitive glBegin(GL\_TRIANGLE\_FAN) function to begin drawing the triangles for the circle. Then we create a for loop which loops 16 times and does x + (radius \* cos(i \* twicePI / 16)), y + (radius \* sin(i \* twicePi/16)), z). This loop and math for the circle using pi and the radius creates the circle of points or triangles which end up making a circle of triangles which is why the gl function is called the triangle fan. Then after doing this we colored the plates to match the scene. No shaders were needed in making this object.



**Paper Towel:**

To create the paper towel and the stand for the paper towel in the scene we used the drawCircle function we created and another function we created called drawCylinder(). This function uses the gl function for quads and has a for loop that loops 16 times like the circle function. The difference here is that using the radius and pi variables, it creates a 3d cylinder shape whereas the circle function just created a flat circle. The pole is also created using the drawCylinder() function we created and the base of the paper towel stand is created using our drawcircle function. Each part was colored accordingly to match the scene and no shaders were needed for this object.



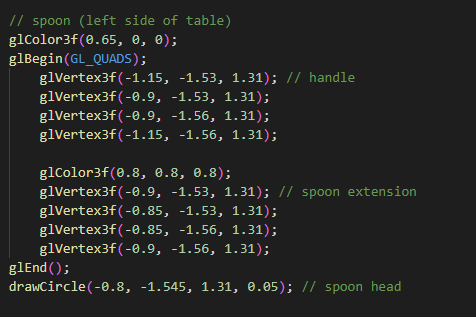
**Cups**:

Each of the cups were created using the drawCylinder() function we created. With the values we passed through the function we were able to create the quads that would give us the right shape for the cup. After this we colored the cups and no shaders were needed.



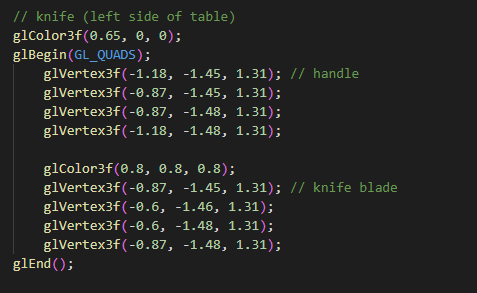
**Fork**:

The fork was created using multiple quads. After using quads to create the fork we colored it to match the scene picture. No shaders were needed for this object.



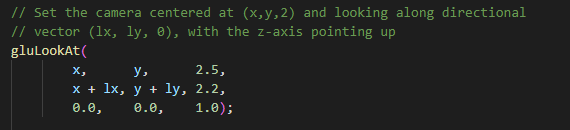
**Spoon**:

The spoon was created by using quads but for the head of the spoon we used the drawCircle function to create that round shape that a spoon would have. After this we colored the spoon to complete the look. There were no shaders needed to make this object.



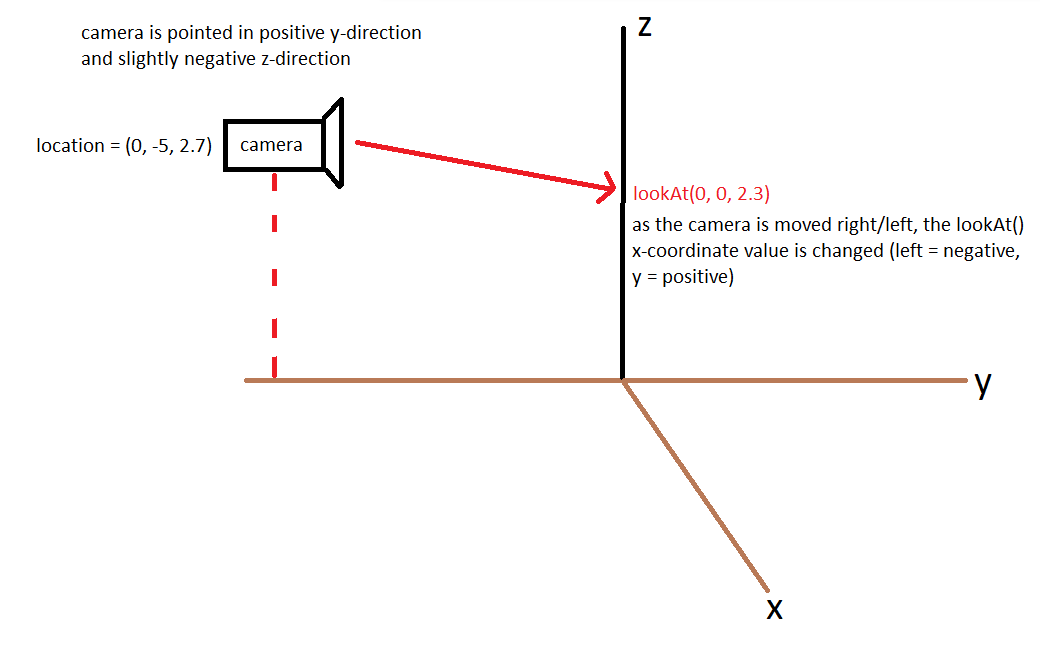
**Knife**:

The knife was created using quads. Then after creating the knife we colored it to give it the colors to match its counterpart in the scene picture. No shaders were needed for this object.



**Camera**:

The camera that we created in the scene is one that is able to be moved by the user's mouse so that they can get different perspectives of the scene. The camera is able to move if the user moves their mouse left and right while holding left-click. The starting angle that we set the camera at is right in front of the scene angled a bit above the table looking down at the scene. This angle matches the drawing and the chosen picture of the scene perfectly. The camera is centered at (0, -5, 2.7) and looking along the vector (0 + lx, 0 + ly, 2.3) and the z-axis is pointing up. Having a camera that can move via the mouse really allows for more angles to be looked at and allows for the scene to feel more 3d and real. The 3d diagram for the camera is pasted below.



Dylan Olthoff, Rylan Casanova

CST-310

Professor Citro

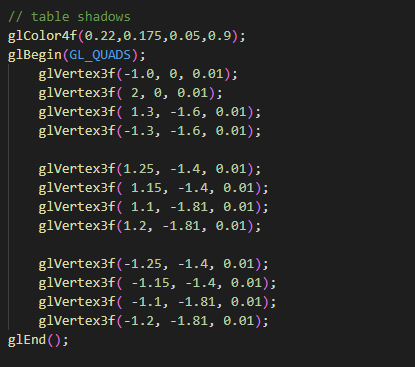
##### **Project 5: Render Your Scene With Primitives**

For Project 5 we are tasked to use the scene that we previously created in project 4 and improve on the quality of the scene and to add anything that is missing from the drawing to the scene. We are tasked to improve each object in the scene to make them look more realistic and to use complex 3D models instead of just primitives for each object. To improve the rendering of the scene we can add meshes to improve the overall look of each object and we are also able to add textures to the objects to make them seem even more realistic and look like the original scene picture that was chosen. Like in project 4, we are tasked to create a camera to represent our scene for the project. We will be using the same camera from project 4 because it fits our scene perfectly and imitates the angle of the original scene picture that was chosen. We are then tasked to give an update on each object that is rendered in the scene and give an update on the mathematical characteristics, primitives, transformations, and if any shaders were used for each object in the scene to give a more realistic look to the objects.



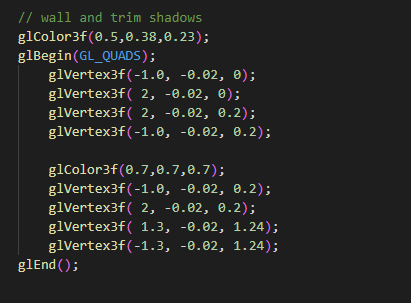
**Chairs:**

For this project we did not touch the chairs and everything having to do with the chairs is what we created in project 4.

****

**Table:**

In this project to update the last project we added a table shadow to give more detail and depth to the overall scene. To create this shadow effect for the table we used the glColor4f() function which allows us to make the color of the shadow transparent and then quads for the shape of the shadow. No other shaders were needed to create this shadow.

****

**Walls:**

For the walls in this function it falls under the category of the table shading because the shadow from the table goes onto the ground. So the shadows that would be cast on the walls and the wall trim reflect the table shadows. This gives the image the appearance of the full table shadow.

**Plates:**

Nothing was changed for the plates in this iteration and everything having to do with the plates is what we created in project 4

**Cups:**

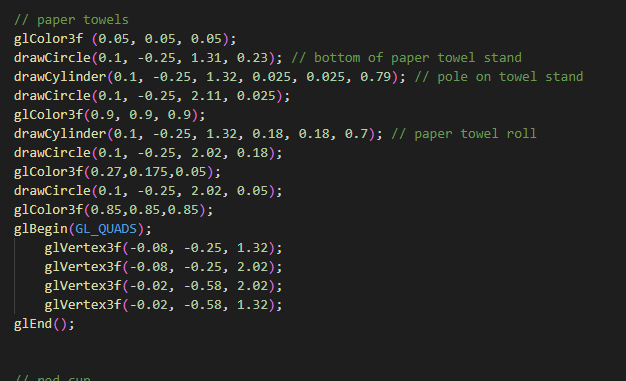
Nothing was changed for the cups in this iteration and everything having to do with the cups is what we created in project 4

**Knife:**

Nothing was changed for the knife in this iteration and everything having to do with the knife is what we created in project 4

**Spoon:**

Nothing was changed for the spoon in this iteration and everything having to do with the spoon is what we created in project 4

****

**Paper Towel:**

In this project we changed the paper towel to add the paper towel piece on the side. This gives the paper towel more detail and makes it more recognizable as a paper towel roll compared to what it looked like in project 4. To do this we used quads and changed the color of it to make it stand out. We also moved it slightly to the right to align it better with our photo of the scene.

**Camera:**

Since we created the camera in the last project we did not update it further. Our camera is a click and hold camera set at an angle. This allows the user to hold their mouse down to rotate the scene and look around.

