# **Tutorial 6 - Shaders and Uniforms**

In this exercise you are going to examine a few shaders and then answer a few questions. The goal of this tutorial is to help you understand how to integrate shaders with your existing code base.

When reading the shaders you should answer the following questions

1. How many input parameters does the shader contain?
2. How would your Vertex Structure in the C++ changes to match these inputs?
3. What would the generic attribute look like to support this vertex structure(glEnableVertexAttribArray & glVertexAttribPointer)?
4. How many uniforms does the shader contain?
5. What would be the matching types on the Application(C++) side? Remember sampler2D are represented on the C++ as Texture Units (ints).
6. How would you retrieve the location of the uniform using glGetUniformLocation?
7. What glUniform\* call could send the variable across? The following link to OpenGL documentation may help - <https://www.opengl.org/sdk/docs/man/html/glUniform.xhtml>
8. Where in the Application side would you put the code to support the Uniforms? E.g. MyGame, GameObject or a new class?

**You should upload you answers to the Tutorial 6 Submission link on GCU Learn before next week's lab.**

## Shaders

1. SimpleVS.glsl and TextureFS.glsl
2. LightVS.glsl and LightFS.glsl
3. ParallaxMappingVS.glsl and ParallaxMappingFS.glsl