Vulkan Program Flow – the Setup

Create a GLFW Vulkan Window

Query the Physical Devices and Choose (1 in our case)

Decide on the Extensions and Layers You Want

Create the Logical Device

Create the Queue(s) (1 in our case)

Allocate and Fill memory for the Vertices and Indices

Allocate and Fill memory for the Uniform Buffers

Create the Command Buffers (3 in our case)

If using Textures, create the Sampler, Read the Texture, and move it to Device Local Memory

Create the Swap Chain (2 images in our case)

Be sure you have Compiled the Shaders into .spv files

Create the Descriptor Set Data Structures

Create the Graphics Pipeline Data Structure Layout(s)

Fill the Graphics Pipeline Data Structure(s)
Vulkan Program Flow – the Rendering Loop

while( the GLFW Window should not close )
{
    UpdateScene( )
    RenderScene( )
}

- Create the Transformations
- Fill the Uniform Buffers

- Acquire the Next Swap Chain Image
- Begin its Command Buffer
- Create the RenderPass with the Framebuffer information
  - for( all the different Graphics Pipeline Data Structures being used )
    - Bind that Graphics Pipeline Data Structure
    - Set any Dynamic State Variables
    - Bind the Proper Descriptor Set Values
      - Do the Drawing
  - End the RenderPass
- End the Command Buffer
- Submit the Command Buffer to a Queue
- Wait for the Queue to Finish Submitting
- Present the Image to the Viewer