A shader is a program run on your GPU. Sometimes, we may find it’s easy to do some work in CPU and sent the data to GPU. But typically, GPU is way faster than CPU.

For this episode, for most programing, we will focus on two types of shaders: 1. Vertex shaders, 2. Fragment (pixel) shaders. There are many different shaders, but we will talk it in the further course. For 90% shader you will be dealing with these 2 shaders.

The vertex shaders will be called 3 times for drawing one triangle since there are 3 points. It will take in all vertex attribute in the buffer

A picture containing chart

Description automatically generated

It can access to this data in shaders because we specify this in

Text

Description automatically generated

The position 0 is used by shader

If you have a camera in somewhere in 3D, what is the camera move? We deal with it in vertex shader about the transformation if necessary.

In Fragment (pixel) shaders, it does rasterization, to fill in the triangle. This shader is to define which color it supposed to be. It may be called thousand of times in this case, since it has thousands of data in this triangle

Fragment shaders is one of the most important staff. A shader could be thousands of lines of code for a nice graph.

Finally, OpenGL is a state machine, you need to enable your shader