Shaders

Mais atributos

|  |
| --- |
| float[] vertices = {  // positions // colors  -0.5f, -0.5f, 0.0f, 0.0f, 1.0f, 0.0f, // bottom left  -0.5f, 0.5f, 0.0f, 0.0f, 0.0f, 1.0f, // top left  0.5f, 0.5f, 0.0f, 0.0f, 0.0f, 1.0f, // top right  0.5f, -0.5f, 0.0f, 1.0f, 0.0f, 0.0f // bottom right  }; |

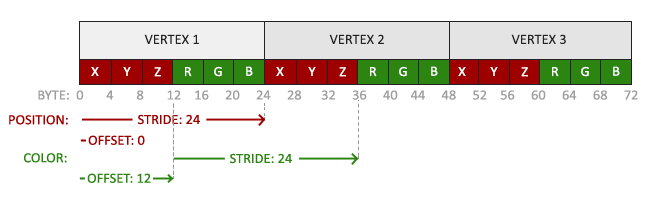
shader.vert

|  |
| --- |
| #version 330 core  layout (location = 0) in vec3 aPos; // a variável position possui o atributo position 0  layout (location = 1) in vec3 aColor; // a variável color tem posição de atributo 1  out vec3 ourColor; // envia uma cor para o shader de fragmento  void main() {  gl\_Position = vec4(aPos, 1.0);  ourColor = aColor; // define ourColor para a cor de entrada que obtivemos dos dados do vértice  } |

Nota: não colocar comentários em português em shaders, então bugando a shader. Não sei pq.

shader.frag

|  |
| --- |
| #version 330 core  out vec4 FragColor;  in vec3 ourColor;  void main() {  FragColor = vec4(ourColor, 1.0f);  } |



|  |
| --- |
| // position attribute  GL.VertexAttribPointer(0, 3, VertexAttribPointerType.Float, false, 6 \* sizeof(float), 0);  GL.EnableVertexAttribArray(0);  // color attribute  GL.VertexAttribPointer(1, 3, VertexAttribPointerType.Float, false, 6 \* sizeof(float), 3 \* sizeof(float));  GL.EnableVertexAttribArray(1); |

