Coordinate Systems

|  |
| --- |
| Matrix4 model = Matrix4.Identity;  model = Matrix4.CreateRotationX(MathHelper.DegreesToRadians(-55.0f)); |

|  |
| --- |
| Matrix4 view = Matrix4.Identity;  view = Matrix4.CreateTranslation(0.0f, 0.0f, -3.0f); |

|  |
| --- |
| Matrix4 projection;  projection = Matrix4.CreatePerspectiveFieldOfView(MathHelper.DegreesToRadians(45.0f), 800.0f / 600.0f, 0.1f, 100.0f); |

shader.vert

|  |
| --- |
| #version 330 core  layout (location = 0) in vec3 aPos;  layout (location = 1) in vec3 aColor;  layout (location = 2) in vec2 aTexCoord;  out vec3 ourColor;  out vec2 TexCoord;  uniform mat4 model;  uniform mat4 view;  uniform mat4 projection;  void main() {  gl\_Position = projection \* view \* model \* vec4(aPos, 1.0);  ourColor = aColor;  TexCoord = vec2(aTexCoord.x, aTexCoord.y);  } |

|  |
| --- |
| int modelLoc = GL.GetUniformLocation(ourShader.ID, "model");  GL.UniformMatrix4(modelLoc, false, ref model);  int viewLoc = GL.GetUniformLocation(ourShader.ID, "view");  GL.UniformMatrix4(viewLoc, false, ref view);  int projectionLoc = GL.GetUniformLocation(ourShader.ID, "projection");  GL.UniformMatrix4(projectionLoc, false, ref projection); |

Interface gráfica do usuário

Descrição gerada automaticamente com confiança baixa

More 3D

|  |
| --- |
| float[] vertices = {  -0.5f, -0.5f, -0.5f, 0.0f, 0.0f,  0.5f, -0.5f, -0.5f, 1.0f, 0.0f,  0.5f, 0.5f, -0.5f, 1.0f, 1.0f,  0.5f, 0.5f, -0.5f, 1.0f, 1.0f,  -0.5f, 0.5f, -0.5f, 0.0f, 1.0f,  -0.5f, -0.5f, -0.5f, 0.0f, 0.0f,  -0.5f, -0.5f, 0.5f, 0.0f, 0.0f,  0.5f, -0.5f, 0.5f, 1.0f, 0.0f,  0.5f, 0.5f, 0.5f, 1.0f, 1.0f,  0.5f, 0.5f, 0.5f, 1.0f, 1.0f,  -0.5f, 0.5f, 0.5f, 0.0f, 1.0f,  -0.5f, -0.5f, 0.5f, 0.0f, 0.0f,  -0.5f, 0.5f, 0.5f, 1.0f, 0.0f,  -0.5f, 0.5f, -0.5f, 1.0f, 1.0f,  -0.5f, -0.5f, -0.5f, 0.0f, 1.0f,  -0.5f, -0.5f, -0.5f, 0.0f, 1.0f,  -0.5f, -0.5f, 0.5f, 0.0f, 0.0f,  -0.5f, 0.5f, 0.5f, 1.0f, 0.0f,  0.5f, 0.5f, 0.5f, 1.0f, 0.0f,  0.5f, 0.5f, -0.5f, 1.0f, 1.0f,  0.5f, -0.5f, -0.5f, 0.0f, 1.0f,  0.5f, -0.5f, -0.5f, 0.0f, 1.0f,  0.5f, -0.5f, 0.5f, 0.0f, 0.0f,  0.5f, 0.5f, 0.5f, 1.0f, 0.0f,  -0.5f, -0.5f, -0.5f, 0.0f, 1.0f,  0.5f, -0.5f, -0.5f, 1.0f, 1.0f,  0.5f, -0.5f, 0.5f, 1.0f, 0.0f,  0.5f, -0.5f, 0.5f, 1.0f, 0.0f,  -0.5f, -0.5f, 0.5f, 0.0f, 0.0f,  -0.5f, -0.5f, -0.5f, 0.0f, 1.0f,  -0.5f, 0.5f, -0.5f, 0.0f, 1.0f,  0.5f, 0.5f, -0.5f, 1.0f, 1.0f,  0.5f, 0.5f, 0.5f, 1.0f, 0.0f,  0.5f, 0.5f, 0.5f, 1.0f, 0.0f,  -0.5f, 0.5f, 0.5f, 0.0f, 0.0f,  -0.5f, 0.5f, -0.5f, 0.0f, 1.0f  }; |

|  |
| --- |
| GL.VertexAttribPointer(0, 3, VertexAttribPointerType.Float, false, 5 \* sizeof(float), 0);  GL.EnableVertexAttribArray(0);  GL.VertexAttribPointer(1, 3, VertexAttribPointerType.Float, false, 5 \* sizeof(float), 3 \* sizeof(float));  GL.EnableVertexAttribArray(1);  GL.VertexAttribPointer(2, 2, VertexAttribPointerType.Float, false, 5 \* sizeof(float), 3 \* sizeof(float));  GL.EnableVertexAttribArray(2); |

O do meio nem importa tanto já que o infeliz que fez esse tutorial (e que não foi eu) tirou as cores dos vertices

|  |
| --- |
| GL.DrawArrays(PrimitiveType.Triangles, 0, 36);  ~~GL.DrawElements(PrimitiveType.Triangles, indices.Length, DrawElementsType.UnsignedInt, 0);~~ |

|  |
| --- |
| model = Matrix4.CreateRotationX(MathHelper.DegreesToRadians((float)GLFW.GetTime() \* 50.0f)); |

Link: <https://learnopengl.com/video/getting-started/coordinate_system_no_depth.mp4>

OnLoad()

|  |
| --- |
| GL.Enable(EnableCap.DepthTest); |

OnRenderFrame()

|  |
| --- |
| GL.Clear(ClearBufferMask.ColorBufferBit | ClearBufferMask.DepthBufferBit); |

Link: <https://learnopengl.com/video/getting-started/coordinate_system_depth.mp4>

More cubes!

|  |
| --- |
| Vector3[] cubePositions = {  new Vector3( 0.0f, 0.0f, 0.0f),  new Vector3( 2.0f, 5.0f, -15.0f),  new Vector3(-1.5f, -2.2f, -2.5f),  new Vector3(-3.8f, -2.0f, -12.3f),  new Vector3( 2.4f, -0.4f, -3.5f),  new Vector3(-1.7f, 3.0f, -7.5f),  new Vector3( 1.3f, -2.0f, -2.5f),  new Vector3( 1.5f, 2.0f, -2.5f),  new Vector3( 1.5f, 0.2f, -1.5f),  new Vector3(-1.3f, 1.0f, -1.5f)  }; |

|  |
| --- |
| GL.BindVertexArray(VAO);  for(int i = 0; i < 10; i++) {  Matrix4 model = Matrix4.Identity;  model = Matrix4.CreateTranslation(cubePositions[i]);  float angle = 20.0f \* i;  model \*= Matrix4.CreateRotationX(MathHelper.DegreesToRadians(angle));  model \*= Matrix4.CreateRotationY(MathHelper.DegreesToRadians(angle \* 0.3f));  model \*= Matrix4.CreateRotationZ(MathHelper.DegreesToRadians(angle \* 0.5f));  GL.UniformMatrix4(GL.GetUniformLocation(ourShader.ID, "model"), false, ref model);  GL.DrawArrays(PrimitiveType.Triangles, 0, 36);  } |

Imagem de jogo de vídeo game

Descrição gerada automaticamente com confiança baixa