package Lab1;

import java.util.Random;

import javax.swing.JFrame;

import com.jogamp.opengl.GL2;

import com.jogamp.opengl.GLAutoDrawable;

import com.jogamp.opengl.GLCapabilities;

import com.jogamp.opengl.GLEventListener;

import com.jogamp.opengl.GLProfile;

import com.jogamp.opengl.awt.GLCanvas;

import com.jogamp.opengl.glu.GLU;

class FirstGLEventListener implements GLEventListener {

/\*\*

\* Interface to the GLU library.

\*/

private GLU glu;

/\*\*

\* Take care of initialization here.

\*/

public void init(GLAutoDrawable gld) {

GL2 gl = gld.getGL().getGL2();

glu = new GLU();

gl.glClearColor(0.0f, 0.0f, 0.0f, 1.0f);

gl.glViewport(-5, -5, 5, 5);

gl.glMatrixMode(GL2.***GL\_PROJECTION***);

gl.glLoadIdentity();

glu.gluOrtho2D(-5.0, 5.0, -5.0, 5.0);

}

/\*\*

\* Take care of drawing here.

\*/

public void display(GLAutoDrawable drawable) {

GL2 gl = drawable.getGL().getGL2();

gl.glClear(GL2.***GL\_COLOR\_BUFFER\_BIT***);

// for drawing individual pixels

gl.glBegin(GL2.***GL\_POINTS***);

for (int i = 0; i < 100; i++) {

float x = randNumb();

float y = randNumb();

gl.glColor3f(1,0.5f,0);

gl.glVertex2f(x, y);

}

gl.glEnd();

}

/\*\*

\* Generate a random floating point number between -1 and 1.

\*/

public float randNumb() {

int max = 1;

int min = -1;

Random random = new Random();

float randomNum = (random.nextFloat() \* (max - min)) + min;

return randomNum;

}

public void reshape(GLAutoDrawable drawable, int x, int y, int width, int height) {

}

public void displayChanged(GLAutoDrawable drawable, boolean modeChanged, boolean deviceChanged) {

}

public void dispose(GLAutoDrawable arg0) {

}

}

public class Task1 {

public static void main(String args[]) {

// getting the capabilities object of GL2 profile

final GLProfile profile = GLProfile.*get*(GLProfile.***GL2***);

GLCapabilities capabilities = new GLCapabilities(profile);

// The canvas

final GLCanvas glcanvas = new GLCanvas(capabilities);

FirstGLEventListener b = new FirstGLEventListener();

glcanvas.addGLEventListener(b);

glcanvas.setSize(400, 400);

// creating frame

final JFrame frame = new JFrame("Basic Frame");

// adding canvas to frame

frame.add(glcanvas);

frame.setSize(640, 480);

frame.setVisible(true);

}

}