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;

import java.lang.Math;

import javax.swing.JFrame;

import java.util.Random;

class ThirdGLEventListener 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(-250, -150, 250, 150);

gl.glMatrixMode(GL2.GL\_PROJECTION);

gl.glLoadIdentity();

glu.gluOrtho2D(-250.0, 250.0, -150.0, 150.0);

}

/\*\*

\* Take care of drawing here.

\*/

public void display(GLAutoDrawable drawable) {

Random random=new Random();

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

gl.glBegin(GL2.GL\_POINTS);

for(int i=1;i<=50;i++) {

int x=0;

int y=0;

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

gl.glPointSize(100.0f);

gl.glColor3d(0.5f, 1.0f, 1.0f);

gl.glVertex2d(x= random.nextInt(350)-200,y= random.nextInt(350)-200);

}

gl.glEnd();

}

//}

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 19301186\_Fahim

{

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);

ThirdGLEventListener b=new ThirdGLEventListener();

glcanvas.addGLEventListener(b);

glcanvas.setSize(400,400);

//creating frame

final JFrame frame=new JFrame("task1");

//adding canvas to frame

frame.add(glcanvas);

frame.setSize(640,480);

frame.setVisible(true);

}

}