package com.mycompany.it;

import java.util.Vector;

import com.codename1.ui.layouts.BorderLayout;

import com.codename1.ui.plaf.Border;

import com.codename1.charts.util.ColorUtil;

import com.codename1.ui.Container;

import com.codename1.ui.Form;

import com.codename1.ui.events.ActionEvent;

import com.codename1.ui.layouts.BorderLayout;

public class ObjectSelectionForm extends Form {

private Vector<GeometricShape> worldShapes = new Vector<GeometricShape>();

CustomContainer centerContainer = new CustomContainer(worldShapes);

public ObjectSelectionForm() {

setLayout(new BorderLayout());

centerContainer.getAllStyles().setBgTransparency(255);

centerContainer.getAllStyles().setBgColor(ColorUtil.LTGRAY);

centerContainer.getAllStyles().setBorder(Border.createLineBorder(4,

ColorUtil.MAGENTA));

add(BorderLayout.CENTER,centerContainer);

worldShapes.addElement(new MyRect(100, 100, 50, 50, ColorUtil.BLACK));

worldShapes.addElement(new MyCir(0 + getWidth()/2, 0 + getHeight()/2, 50, ColorUtil.GREEN));

worldShapes.addElement(new MyLine(0 + centerContainer.getX(), 0 + centerContainer.getY(), 0 + getWidth()/2, 0 + getHeight()/2, ColorUtil.BLUE));

this.show();

}

}

package com.mycompany.it;

import com.codename1.ui.Graphics;

import com.codename1.ui.geom.Point;

public class MyCir extends GeometricShape {

int iShapeX;

int iShapeY;

int width, height, radius;

int color;

public MyCir(int iShapeX, int iShapeY, int radius, int color) {

this.iShapeX = iShapeX;

this.iShapeY = iShapeY;

this.width = 2\*radius;

this.height = 2\*radius;

this.color = color;

}

public void draw(Graphics g, Point pCmpRelPrnt) {

// TODO Auto-generated method stub

g.setColor(color);

int xLoc = pCmpRelPrnt.getX()+ iShapeX;// shape location relative

int yLoc = pCmpRelPrnt.getY()+ iShapeY;// to parents origin

if(isSelected()) {

g.fillArc(xLoc, yLoc, width, height, 0, 360);

}

else{

g.drawArc(xLoc, yLoc, width, height, 0, 360);

}

}

public boolean contains(Point pPtrRelPrnt, Point pCmpRelPrnt) {

int px = pPtrRelPrnt.getX(); // pointer location relative to

int py = pPtrRelPrnt.getY(); // parents origin

int xLoc = pCmpRelPrnt.getX()+ iShapeX;// shape location relative

int yLoc = pCmpRelPrnt.getY()+ iShapeY;// to parents origin

if ( (px >= xLoc) && (px <= xLoc+width)

&& (py >= yLoc) && (py <= yLoc+height) )

return true;

else

return false;

}

}

