import javax.swing.\*;  
import java.awt.\*;  
import java.awt.event.MouseEvent;  
  
*/\*\*  
 \* A MushroomBodyhoz tartozó swinges nézet, amely az objektumot piros háromszögként jeleníti meg.  
 \*/*public class SwingMushroomBody extends JPanel implements Updatable {  
  
 private MushroomBodyView mbv;  
 private JPopupMenu mushroomBodyPopupMenu;  
 private static final int *TRIANGLE\_MARGIN* = 5;  
  
 */\*\*  
 \* Konstruktor.  
 \* Beállítja az egérfigyelőt, létrehozza a popup menüt és frissíti a tooltipet.  
 \*  
 \* @param mushroomBody A MushroomBody példány, amelyhez a SwingMushroomBody objektum létrehozásra kerül.  
 \*/* public SwingMushroomBody(MushroomBody mushroomBody) {  
 this.mbv = mushroomBody;  
  
 addMouseListener(new MushroomBodyMouseAdapter(this));  
 update();  
  
 mushroomBodyPopupMenu = new JPopupMenu();  
  
 mushroomBodyPopupMenu.add("MushroomBody: " + ObjectRegistry.*lookupName*(mushroomBody));  
  
 JButton ejectButton = new JButton("Eject spores");  
 ejectButton.addActionListener(new EjectSporesButtonListener(mushroomBody));  
 mushroomBodyPopupMenu.add(ejectButton);  
 }  
  
 */\*\*  
 \* Tooltip frissítése a megmaradt spórakilövések számával.  
 \*/* @Override  
 public void update() {  
 setToolTipText("Remaining ejects: " + mbv.getRemainingEjects());  
 }  
  
 */\*\*  
 \* Megjeleníti a popup menüt.  
 \*  
 \* @param e MouseEvent objektum.  
 \*/* public void showPopupMenu(MouseEvent e) {  
 mushroomBodyPopupMenu.show(e.getComponent(), e.getX(), e.getY());  
 }  
  
 */\*\*  
 \* A MushroomBody grafikus megjelenítése piros háromszög megrajzolásával.  
 \*  
 \* @param g Grafikus objektum.  
 \*/* @Override  
 protected void paintComponent(Graphics g) {  
 super.paintComponent(g);  
 Graphics2D mbTriangle = (Graphics2D) g;  
  
 int w = getWidth();  
 int h = getHeight();  
  
 int[] xPoints = { w / 2, w - *TRIANGLE\_MARGIN*, *TRIANGLE\_MARGIN* };  
 int[] yPoints = { *TRIANGLE\_MARGIN*, h - *TRIANGLE\_MARGIN*, h - *TRIANGLE\_MARGIN* };  
  
 mbTriangle.setColor(Color.*RED*);  
 mbTriangle.fillPolygon(xPoints, yPoints, 3);  
 }  
}