```java
import java.awt.*;
import java.awt.event.*;

/**
    A simple keyboard and mouse test. Displays keys pressed and released to
    the screen and shows how to watch for mouse events and get x,y mouse co-ordinates.
*/
public class EventTest extends GameCore implements KeyListener, MouseListener
{
    String  message;
    int     x,y;

    public static void main(String[] args) {
        EventTest t = new EventTest();
        t.run();
    }

     // This version of init over rides the standard GameCore version
     // and will be called from the run method present in GameCore
    public void init()
    {
         x = 20;
         y = 20;
         message = "KeyInputTest. Press Escape to exit";

        super.init();

        Window window = screen.getFullScreenWindow();

        // allow input of the TAB key and other keys normally
        // used for focus traversal
        window.setFocusTraversalKeysEnabled(false);

        // register this object as a key listener for the window
        window.addKeyListener(this);
        window.addMouseListener(this);
    }

    public void draw(Graphics2D g)
    {
        g.setFont(new Font("Dialog", Font.PLAIN, FONT_SIZE));

        if (message != null)
        {
            g.setColor(Color.blue);
            g.drawString(message,x,y);
        }
        message = null;
    }


    // The following methods are from the MouseListener interface
    // We are required to implement them since we said we implemented
    // the MouseListener interface. Empty methods will do for most of them
    public void mousePressed(MouseEvent e) { }

    public void mouseReleased(MouseEvent e) { }

    public void mouseClicked(MouseEvent e)
    {
        x = e.getX();
        y = e.getY();
    }

    public void mouseEntered(MouseEvent e) { }

    public void mouseExited(MouseEvent e) { }
    // The keyPressed method from the KeyListener interface
    public void keyPressed(KeyEvent e) {

        int keyCode = e.getKeyCode();

        // exit the program
        if (keyCode == KeyEvent.VK_ESCAPE)
             stop(); // Call the GameCore default stop method which stops the game loop
        else
        {
            message = "Pressed: " + KeyEvent.getKeyText(keyCode);
            y = y + 20;
        }
    }

    // The keyReleased method from the KeyListener interface
    public void keyReleased(KeyEvent e)
    {
        int keyCode = e.getKeyCode();

        message = "Released: " + KeyEvent.getKeyText(keyCode);
        y = y + 20;
    }

    // The keyTyped method from the KeyListener interface
    public void keyTyped(KeyEvent e)
    {
        // this is called after the key is released - ignore it
    }


}
```