

```
import java.awt.Color;
import java.awt.Graphics;
import java.awt.event.MouseEvent;
import java.awt.event.MouseListener;
import java.awt.event.MouseMotionListener;

import javax.swing.JFrame;
import javax.swing.JPanel;

public class GUI extends JFrame {

    private Board board;

    public GUI() {

        board = new Board();
        this.setContentPane(board);

        //Ακροατής σε συμβάντα τύπου 'κλικ' στο ποντίκι
        //MouseListener listener = new MouseClickListener();
        //board.addMouseListener(listener);

        //Ακροατής σε συμβάντα τύπου 'drag & drop' στο ποντίκι
        MouseMoveListener listener = new MouseMoveListener();
        board.addMouseMotionListener(listener);

        this.setVisible(true);
        this.setSize(400, 400);
    }

    //Χειρισμός συμβάντων κλικ ποντικιού
    // Βιβλιοθήκη MouseListener
    // Εσωτερική της GUI για να μπορώ να βλέπω το board
    class MouseClickListener implements MouseListener {

        // Αφηρημένες μεθόδους

        @Override
        public void mouseClicked(MouseEvent event) {

            // Παίρνω συντεταγμένες που ο χρήστης έκανε κλικ με το ποντίκι
            int x = event.getX();
            int y = event.getY();

            board.setXYCoordinates(x, y); // Θέτει τις συντεταγμένες στην σκακιέρα
            board.repaint(); //Ξανασχεδιάζει τον εαυτό της
        }

        @Override
        public void mouseEntered(MouseEvent arg0) {
            // TODO Auto-generated method stub
        }

        @Override
        public void mouseExited(MouseEvent arg0) {
            // TODO Auto-generated method stub
        }

        @Override
        public void mousePressed(MouseEvent arg0) {
            // TODO Auto-generated method stub
        }

    }
}
```

```
@Override
public void mouseReleased(MouseEvent arg0) {
    // TODO Auto-generated method stub

}
```

```
}
```

```
class MouseMoveListener implements MouseMotionListener {
```

```
    // Αφηριμένες μεθόδους
```

```
@Override
public void mouseDragged(MouseEvent event) {
```

```
    // Παίρνω συντεταγμένες που ο χρήστης έκανε κλικ με το ποντίκι
```

```
    int x = event.getX();
    int y = event.getY();
```

```
    board.setXYCoordinates(x, y); // Θέτει τις συντεταγμένες στην σκακιέρα
    board.repaint(); // Ξανασχεδιάζει τον εαυτό της
```

```
}
```

```
@Override
public void mouseMoved(MouseEvent arg0) {
    // TODO Auto-generated method stub
```

```
}
```

```
}
```

```
}
```

```
class Board extends JPanel {
```

```
    // Συντεταγμένες που θα βγαίνει στο πιόνι
```

```
    private int xCoord = 0;
    private int yCoord = 0;
```

```
    // Θέτει τις τιμές που θα σχεδιαστεί ο δίσκος (μέθοδος set)
```

```
    // μέθοδος που θέτει συντεταγμένες στο 'πιόνι' που θα σχεδιαστεί
```

```
    public void setXYCoordinates(int x, int y) {
        xCoord = x;
        yCoord = y;
    }
```

```
    // Επικάλυψη του painComponent και ζωγραφίζουμε σκακιέρα
```

```
    public void paintComponent(Graphics g) {
        super.paintComponent(g);
        int sqSize1 = this.getWidth() / 8;
        int sqSize2 = this.getHeight() / 8;
```

```
        int sqSize = sqSize1;
        if(sqSize2 < sqSize1)
            sqSize = sqSize2;
```

```
        for(int i=0; i<8; i++) {
```

```
            for(int j=0; j<8; j++) {
                int x = j * sqSize;
                int y = i * sqSize;
                //g.drawRect(x, y, sqSize, sqSize);
                if( (i+j)%2 != 0) {
                    g.setColor(Color.BLUE);
```

```
    g.fillRect(x, y, sqSize, sqSize);
}
else {
    g.setColor(Color.RED);
    g.fillRect(x, y, sqSize, sqSize);
}
}
```

```
//Σχεδίαση πιονιού. Αλλάζω το χρώμα της γραφίδας
//προκειται για οβάλ (κύκλος) που σχεδιάζεται εντός ορθογωνίου (τετραγώνου)
```

```
g.setColor(Color.GREEN);
```

```
// Ζωγραφίζω ένα οβάλ x,y εκεί που ξεκινάει και έχει μέγεθος ίσο με το μέγεθος του τετραγωνιδίου
// (sqSize/2 για να μην βγαίνει από σκακιέρα).
```

```
// η μετατόπιση κατά sqSize/2 γίνεται ώστε το κλίκ να αντιστοιχεί στο κέντρο του κύκλου
```

```
g.fillOval(xCoord-sqSize/2, yCoord-sqSize/2, sqSize, sqSize);
```

```
}
```

```
}
```