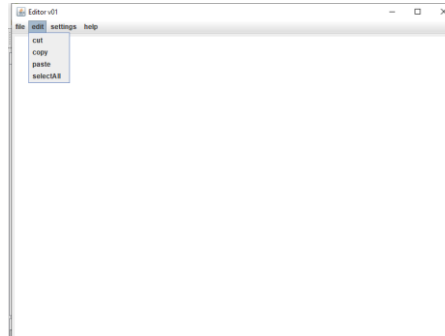


# Φύλλο εργασίας 17 για το μάθημα «ΕΙΔΙΚΑ ΘΕΜΑΤΑ ΣΤΟΝ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟ ΥΠΟΛΟΓΙΣΤΩΝ ,Java”

Εκπαιδευτικός: Παπαστεργίου Κωνσταντίνος, Καθηγητής Πληροφορικής

## Εφαρμογή MyEditor



```
package editor;
```

```
import javax.swing.*;*;  
import java.awt.*;*;  
import java.awt.event.*;*;  
import java.io.*;*;  
import java.io.File; // Import the File class  
import java.io.IOException; // Import the IOException class to handle errors
```

```
public class Editor implements ActionListener{
```

```
    private static JMenu file,edit,help,settings;  
    private static JMenuItem nnew,nopen,nsave,nprint,cut,copy,paste,selectAll,font,desktop;  
    private static JMenuBar mb;  
    private static JTextArea t,tset;  
    private static JLabel tname,tcolor,tfont,tface;  
    private static JButton tb,tblack,tblue,tyellow,tred;  
    private static JCheckBox box1,box2,box3;  
    private static JRadioButton bb1,bb2,bb3;
```

```
    private static int v=12;  
    private static Color mycolor = Color.black;  
    private static Color bgcolor = Color.gray;  
    private static int FontType = Font.PLAIN;  
    private static String FontFace= "Verdana";
```

```
    public static void main(String args[])throws IOException {
```

```
        JFrame frame= new JFrame("Editor v01");
```

```
        mb=new JMenuBar();  
        file=new JMenu("file");  
        edit=new JMenu("edit");  
        settings=new JMenu("settings");  
        help=new JMenu("help");
```

```
nnew=new JMenuItem("new");
nnew.addActionListener(new Editor());

nopen=new JMenuItem("open");
nopen.addActionListener(new Editor());
nsave=new JMenuItem("save");

nsave.addActionListener(new Editor());
nprint=new JMenuItem("print");
nprint.addActionListener(new Editor());
cut=new JMenuItem("cut");
cut.addActionListener(new Editor());
copy=new JMenuItem("copy");
copy.addActionListener(new Editor());
paste=new JMenuItem("paste");
paste.addActionListener(new Editor());
selectAll=new JMenuItem("selectAll");
selectAll.addActionListener(new Editor());
font=new JMenuItem("font");
font.addActionListener(new Editor());
desktop=new JMenuItem("desktop");
desktop.addActionListener(new Editor());
```

```
file.add(nnew);
file.add(nopen);
file.add(nsave);
file.add(nprint);
edit.add(cut);
edit.add(copy);
edit.add(paste);
edit.add(selectAll);
settings.add(font);
settings.add(desktop);
```

```
mb.add(file);
mb.add(edit);
mb.add(settings);
mb.add(help);
```

```
t=new JTextArea();
t.setBounds(5,5,795,595);
t.setFont(new Font(FontFace, FontType, v));
t.setForeground(mycolor);
```

```
frame.add(mb);
frame.add(t);
frame.setJMenuBar(mb);
```

```
frame.setLayout(null);
frame.setSize(800,600);
frame.setVisible(true);
```

```
}
```

```
public void actionPerformed(ActionEvent e) {
    if(e.getSource()==nnew){
```

```

    t.setText("");
}

if(e.getSource()==nsave){
    JFileChooser j = new JFileChooser("C:");

    // Invoke the showsSaveDialog function to show the save dialog
    int r = j.showSaveDialog(null);

    if (r == JFileChooser.APPROVE_OPTION) {

        // Set the label to the path of the selected directory
        File fi = new File(j.getSelectedFile().getAbsolutePath());

        try {
            // Create a file writer
            FileWriter wr = new FileWriter(fi, false);

            // Create buffered writer to write
            BufferedWriter w = new BufferedWriter(wr);

            // Write
            w.write(t.getText());

            w.flush();
            w.close();
        }
        catch (Exception evt) {
            JOptionPane.showMessageDialog(null, evt.getMessage());
        }
    }
    // If the user cancelled the operation
    else
        JOptionPane.showMessageDialog(null, "the user cancelled the operation");
}

```

```

if(e.getSource()==nopen){
    // Create an object of JFileChooser class
    JFileChooser j = new JFileChooser("c:");

    // Invoke the showsOpenDialog function to show the save dialog
    int r = j.showOpenDialog(null);

    // If the user selects a file
    if (r == JFileChooser.APPROVE_OPTION) {
        // Set the label to the path of the selected directory
        File fi = new File(j.getSelectedFile().getAbsolutePath());

        try {
            // String
            String s1 = "", sl = "";

            // File reader
            FileReader fr = new FileReader(fi);

            // Buffered reader

```

```

BufferedReader br = new BufferedReader(fr);

// Initialize sl
sl = br.readLine();

// Take the input from the file
while ((s1 = br.readLine()) != null) {
    sl = sl + "\n" + s1;
}

// Set the text
t.setText(sl);
}
catch (Exception evt) {
    JOptionPane.showMessageDialog(null, evt.getMessage());
}
}
// If the user cancelled the operation
else
    JOptionPane.showMessageDialog(null, "the user cancelled the operation");
}

```

```

if(e.getSource()==nprint){
    try {
        // print the file
        t.print();
    }
    catch (Exception evt) {
        JOptionPane.showMessageDialog(null, evt.getMessage());
    }
}

```

```

if(e.getSource()==cut){
    t.cut(); }
if(e.getSource()==paste) {
    t.paste(); }
if(e.getSource()==copy) {
    t.copy(); }
if(e.getSource()==selectAll) {
    t.selectAll(); }

```

```

if(e.getSource()==font){
    JFrame frame2= new JFrame("font");
    frame2.setLayout(null);
    frame2.setSize(400,300);
    frame2.setVisible(true);

    tname= new JLabel();
    tname.setText("Επίλεξε μέγεθος");
    tname.setBounds(10, 20, 120, 20);
    tname.setForeground(Color.black);
    tname.setFont(new Font("Verdana", Font.BOLD, 11));

    tset=new JTextArea();

```

```
tset.setBounds(30,50,30,20);
tset.setFont(new Font("Verdana", Font.PLAIN, 11));

tb=new JButton("ok");
tb.addActionListener(new Editor());
tb.setBounds(20, 80, 50, 30);

tcolor= new JLabel();
tcolor.setText("Επίλεξε χρώμα");
tcolor.setBounds(200, 20, 120, 20);
tcolor.setForeground(Color.black);
tcolor.setFont(new Font("Verdana", Font.BOLD, 11));

tblack=new JButton("");
tblack.addActionListener(new Editor());
tblack.setBounds(210, 60, 25, 25);
tblack.setBackground(Color.black);

tblue=new JButton("");
tblue.addActionListener(new Editor());
tblue.setBounds(245, 60, 25, 25);
tblue.setBackground(Color.blue);

tyellow=new JButton("");
tyellow.addActionListener(new Editor());
tyellow.setBounds(280, 60, 25, 25);
tyellow.setBackground(Color.yellow);

tred=new JButton("");
tred.addActionListener(new Editor());
tred.setBounds(315, 60, 25, 25);
tred.setBackground(Color.red);

tfont= new JLabel();
tfont.setText("Επίλεξε γραμματοσειρά");
tfont.setBounds(10, 120, 150, 20);
tfont.setForeground(Color.black);
tfont.setFont(new Font("Verdana", Font.BOLD, 11));

box1 = new JCheckBox("Verdana");
box1.addActionListener(new Editor());
box1.setBounds(20, 135, 120, 50);
box1.setSelected(true);

box2 = new JCheckBox("TAHOMA");
box2.addActionListener(new Editor());
box2.setBounds(20, 170, 120, 50);
box2.setSelected(false);

box3 = new JCheckBox("Arial");
box3.addActionListener(new Editor());
box3.setBounds(20, 205, 120, 50);
box3.setSelected(false);

tface= new JLabel();
```

```

tface.setText("Επιλεξε ιδότητα");
tface.setBounds(230, 120, 150, 20);
tface.setForeground(Color.black);
tface.setFont(new Font("Verdana", Font.BOLD, 11));

bb1 = new JRadioButton("Normal");
bb1.addActionListener(new Editor());
bb1.setBounds(220, 135, 120, 50);
bb1.setSelected(true);

bb2 = new JRadioButton("Bold");
bb2.addActionListener(new Editor());
bb2.setBounds(220, 170, 120, 50);
bb2.setSelected(false);

bb3 = new JRadioButton("Italic");
bb3.addActionListener(new Editor());
bb3.setBounds(220, 205, 120, 50);
bb3.setSelected(false);

frame2.add(tname);
frame2.add(tset);
frame2.add(tb);
frame2.add(tcolor);
frame2.add(tblack);
frame2.add(tblue);
frame2.add(tyellow);
frame2.add(tred);
frame2.add(tfont);
frame2.add(box1);
frame2.add(box2);
frame2.add(box3);

frame2.add(tface);
frame2.add(bb1);
frame2.add(bb2);
frame2.add(bb3);
}

if(e.getSource()==tb) {
    String w1=tset.getText();
    v= Integer.parseInt(w1);
    t.setFont(new Font(FontFace, FontType, v));
    t.setForeground(mycolor);
}

if(e.getSource()==tblack) {
    mycolor=Color.black;
    t.setFont(new Font(FontFace, FontType, v));
    t.setForeground(mycolor);
}

if(e.getSource()==tblue) {
    mycolor=Color.blue;
    t.setFont(new Font(FontFace, FontType, v));
}

```

```

    t.setForeground(mycolor);
}

if(e.getSource()==tyellow) {
    mycolor=Color.yellow;
    t.setFont(new Font(FontFace, FontType, v));
    t.setForeground(mycolor);
}

if(e.getSource()==tred) {
    mycolor=Color.red;
    t.setForeground(mycolor);
}

if(box1.isSelected()){
    FontFace="Verdana";
    t.setFont(new Font(FontFace, FontType, v));
}

if(box2.isSelected()){
    FontFace="TAHOMA";
    t.setFont(new Font(FontFace, FontType, v));
}

if(box3.isSelected()){
    FontFace="Arial";
    t.setFont(new Font(FontFace, FontType, v));
}

if(bb1.isSelected()){
    FontType = Font.PLAIN;
    t.setFont(new Font(FontFace, FontType, v));
}

if(bb2.isSelected()){
    FontType = Font.BOLD;
    t.setFont(new Font(FontFace, FontType, v));
}

if(bb3.isSelected()){
    FontType = Font.ITALIC;
    t.setFont(new Font(FontFace, FontType, v));
}

if(bb3.isSelected() && bb2.isSelected()){
    FontType = Font.BOLD + Font.ITALIC;
    t.setFont(new Font(FontFace, FontType, v));
}

}

}

```