import java.awt.\*;

import java.awt.datatransfer.Clipboard;

import java.awt.datatransfer.DataFlavor;

import java.awt.datatransfer.StringSelection;

import java.awt.datatransfer.Transferable;

import java.awt.event.\*;

import java.io.BufferedReader;

import java.io.FileReader;

import java.io.FileWriter;

import java.io.IOException;

public class MyNotePadD{

public static void main(String[] args) {

MyNotePadD ntpd = new MyNotePadD();

ntpd.prepareGUI();

}

Frame mainframe;

private MenuBar mb;

private Menu file,edit, format,view,help;

private MenuItem save, open, newFile, exit;

private MenuItem cut,copy,paste,font,statusBar,aboutNP,aboutMe;

TextArea textArea;

Window Event Listener wListener = new WindowEventListner();

MyNotePadD(){

}

public void prepareGUI(){

mainframe = new Frame();

mainframe.setTitle("NotePad");

mainframe.setBounds(400,100,400,400);

mainframe.setLayout(null);

mainframe.addWindowListener(wListener);

mainframe.setVisible(true);

mainframe.setResizable(false);

textArea = new TextArea();

textArea.setName("textArea");

textArea.setBounds(10,60,380,360);

MB =new MenuBar();

file = new Menu("File");

edit = new Menu("Edit");

format = new Menu("Format");

view = new Menu("View");

help = new Menu("Help");

newFile = new MenuItem("New");

save = new MenuItem("Save");

open = new MenuItem("Open");

exit = new MenuItem("Exit");

cut = new MenuItem("Cut");

copy = new MenuItem("Copy");

paste = new MenuItem("Paste");

font = new MenuItem("Font");

statusBar = new MenuItem("Status Bar");

aboutNP = new MenuItem("About NotePad");

aboutMe = new MenuItem("About Me");

mainframe.setMenuBar(mb);

mb.add(file);

mb.add(edit);

mb.add(format);

mb.add(view);

mb.add(help);

file.add(newFile);

file.add(save);

file.add(open);

file.add(exit);

edit.add(cut);

edit.add(copy);

edit.add(paste);

format.add(font);

view.add(statusBar);

help.add(aboutNP);

help.add(aboutMe);

mainframe.add(textArea);

ActionListener al = new MyActionListener();

open.addActionListener(al);

newFile.addActionListener(al);

save.addActionListener(al);

exit.addActionListener(al);

cut.addActionListener(al);

copy.addActionListener(al);

paste.addActionListener(al);

font.addActionListener(al);

statusBar.addActionListener(al);

aboutNP.addActionListener(al);

aboutMe.addActionListener(al);

/\*

textArea.addCaretListener(//Works ONLY for JTextArea

new CaretListener()

{

public void caretUpdate(CaretEvent e)

{

int lineNumber=0, column=0, pos=0;

try

{

pos=textArea.getCaretPosition();

lineNumber=textArea.getLineOfOffset(pos);

column=pos-textArea.getLineStartOffset(lineNumber);

}catch(Exception excp){}

if(ta.getText().length()==0){lineNumber=0; column=0;}

statusBar.setText("|| Ln "+(lineNumber+1)+", Col "+(column+1));

}

});

}\*/

}

private class WindowEventListner implements WindowListener {

public void windowClosing(WindowEvent we){

System.out.println(we.toString());

System.exit(0);

}

public void windowActivated(WindowEvent we){}

public void windowClosed(WindowEvent we){}

public void windowDeactivated(WindowEvent we){}

public void windowDeiconified(WindowEvent we){}

public void windowIconified(WindowEvent we) {}

public void windowOpened(WindowEvent we){}

}

public class MyActionListener extends javax.swing.JFrame implements ActionListener{

Clipboard clipboard = getToolkit().getSystemClipboard();

String filename;

@Override

public void actionPerformed(ActionEvent ev) {

MenuItem item = (MenuItem)(ev.getSource());

if(item.getLabel().equals("New")) {

textArea.setText("");

mainframe.setTitle(filename);

}

else if (item.getLabel().equals("Save")){

FileDialog fd = new FileDialog(mainframe,"Save File",FileDialog.SAVE);

fd.setVisible(true);

if(fd.getFile() != null){

filename = fd.getDirectory() + fd.getFile();

mainframe.setTitle(filename);

} try{

FileWriter fw = new FileWriter(filename);

fw.write(textArea.getText());

mainframe.setTitle(filename);

fw.close();

}catch(IOException ioe){

System.out.println("File not found");

}

}

else if (item.getLabel().equals("Open")){

FileDialog fd = new FileDialog(mainframe, " Open File", FileDialog.LOAD);

fd.setVisible(true);

if(fd.getFile() != null){

filename= fd.getDirectory() + fd.getFile();

mainframe.setTitle(filename);

}try{

BufferedReader reader = new BufferedReader(new FileReader(filename));

StringBuilder sb = new StringBuilder();

String line =null;

while((line = reader.readLine()) != null){

sb.append(line + "\n");

textArea.setText(sb.toString());

}

reader.close();

}catch(IOException ioe){

System.out.println("File not found");

}

}

else if (item.getLabel().equals("Exit")){

System.exit(0);

}

else if (item.getLabel().equals("Cut")){

String cutString = textArea.getSelectedText();

StringSelection cutSelection = new StringSelection(cutString);

clipboard.setContents(cutSelection, cutSelection);

textArea.replaceRange("", textArea.getSelectionStart(),textArea.getSelectionEnd());

}

else if (item.getLabel().equals("Copy")){

String copyText = textArea.getSelectedText();

StringSelection copySelection = new StringSelection(copyText);

clipboard.setContents(copySelection, copySelection);

}

else if (item.getLabel().equals("Paste")){

try{

Transferable pasteText = clipboard.getContents(MyNotePadD.this);

String sel = (String) pasteText.getTransferData(DataFlavor.stringFlavor);

textArea.replaceRange(sel, textArea.getSelectionStart(), textArea.getSelectionEnd());

}catch(Exception ioe){

System.out.println("Didn't Work");

}

}

else if (item.getLabel().equals("Font") ||item.getLabel().equals("Status Bar") ){

Frame f2 = new Frame("404");

Label lb = new Label(" Button couldn't be constructed");

f2.setBounds(450,400,200,150);

f2.add(lb);

f2.setBackground(Color.yellow);

f2.addWindowListener(new WindowAdapter() {

public void windowClosing(WindowEvent windowEvent){

f2.dispose();

}

});

f2.setVisible(true);

f2.setResizable(false);

}

else if (item.getLabel().equals("About NotePad") ){

Font font =new Font("Arial", Font.ITALIC,20);

Frame f2 = new Frame("Intro");

Label l1 = new Label("Very Simple Simple Simple Notepad");

l1.setFont(font);

f2.setBounds(440,400,360,200);

f2.add(l1);

f2.setBackground(Color.yellow);

f2.addWindowListener(new WindowAdapter() {

public void windowClosing(WindowEvent windowEvent){

f2.dispose();

}

});

f2.setVisible(true);

f2.setResizable(false);

}

else if (item.getLabel().equals("About Me") ){

Font font =new Font("Arial", Font.PLAIN,20);

Frame f2 = new Frame("Intro");

Label l1 = new Label("JNC :D I know I'm not yet good at this :(");

l1.setFont(font);

f2.setBounds(440,400,380,200);

f2.add(l1);

f2.setBackground(Color.CYAN);

f2.addWindowListener(new WindowAdapter() {

public void windowClosing(WindowEvent windowEvent){

f2.dispose();

}

});

f2.setVisible(true);

f2.setResizable(false);

}

}

}

}