import java.awt.\*;

import java.applet.\*;

import java.awt.event.\*;

/\*<applet code= "NotepadDemo" width=400 height=400>

</applet>\*/

//create subclass of dialog

class DialogDemo extends Dialog implements ActionListener

{

Button b;

String msg;

DialogDemo(Frame parent, String title) //constructor

{

super(parent,title,true);

setLayout(new FlowLayout(FlowLayout.RIGHT)); //set layout

this.setSize(300,300);

b = new Button("Cancel"); //obj of button

add(b); //add control

b.addActionListener(this); //register

}

public void actionPerformed(ActionEvent ae)

{

dispose(); //close

}

public void paint(Graphics g)

{

msg = "This is in the dialog box";

g.drawString(msg, 100,200);

}

}

class FileDialogDemo extends FileDialog

{

FileDialog fd;

FileDialogDemo(Frame parent, String title, int how)

{

super(parent,title,how);

}

}

//Create subclass of frame

class MenuFrame extends Frame

{

String msg = "";

CheckboxMenuItem debug,test;

MenuFrame(String title)

{

super(title);

//create menu bar and add it to the frame

MenuBar mbar = new MenuBar();

setMenuBar(mbar);

//create menu items - file

Menu file = new Menu("File");

MenuItem item1,item2,item3,item4,item5,item6,item7;

file.add(item1 = new MenuItem("New"));

file.add(item2 = new MenuItem("Open"));

file.add(item3 = new MenuItem("Save"));

file.add(item4 = new MenuItem("Close"));

file.add(item5 = new MenuItem("Help"));

file.add(item6 = new MenuItem("-"));

file.add(item7 = new MenuItem("Quit"));

mbar.add(file);

//create menu items - edit

MenuItem item8,item9,item10;

Menu edit = new Menu("Edit");

edit.add(item8 = new MenuItem("Cut"));

edit.add(item9 = new MenuItem("Copy"));

edit.add(item10 = new MenuItem("Paste"));

//checkable menu items

debug = new CheckboxMenuItem("Debug");

test = new CheckboxMenuItem("Test");

edit.add(debug);

edit.add(test);

mbar.add(edit);

//object to handle action and item event

MyMenuHandler handler = new MyMenuHandler(this);

//register

item1.addActionListener(handler);

item2.addActionListener(handler);

item3.addActionListener(handler);

item4.addActionListener(handler);

item5.addActionListener(handler);

item6.addActionListener(handler);

item7.addActionListener(handler);

item8.addActionListener(handler);

item9.addActionListener(handler);

item10.addActionListener(handler);

debug.addItemListener(handler); //ItemListener

test.addItemListener(handler); //ItemListener

//object to handle window events

MyWindowAdapter adapter = new MyWindowAdapter(this);

//register

addWindowListener(adapter);

}

public void paint(Graphics g)

{

g.drawString(msg, 10,200);

if(debug.getState())

{

g.drawString("Debug is on...",10,220);

}

else

{

g.drawString("Debug is off...",10,220);

}

if(test.getState())

{

g.drawString("Test is on...",10,240);

}

else

{

g.drawString("Test is off...",10,240);

}

}

}

class MyWindowAdapter extends WindowAdapter

{

MenuFrame menuFrame;

public MyWindowAdapter(MenuFrame menuFrame)

{

this.menuFrame = menuFrame;

}

public void windowClosing(WindowEvent we)

{

menuFrame.dispose();

}

}

class MyMenuHandler implements ActionListener, ItemListener

{

MenuFrame menuFrame;

public MyMenuHandler(MenuFrame menuFrame)

{

this.menuFrame = menuFrame;

}

//handle action events

public void actionPerformed(ActionEvent ae)

{

String msg= "You Selected: ";

String arg = (String) ae.getActionCommand();

//activate dialog box

if(arg.equals("New"))

{

msg +="New";

DialogDemo d = new DialogDemo(menuFrame, "New Dialog box");

d.setVisible(true);

}

//activate filedialog box

if(arg.equals("Open"))

{

msg +="Open";

FileDialogDemo fd = new FileDialogDemo(menuFrame, "OPEN",FileDialog.LOAD);

fd.setVisible(true);

}

//activate filedialog box

if(arg.equals("Save"))

{

msg +="Save";

FileDialogDemo fd = new FileDialogDemo(menuFrame, "SAVE",FileDialog.SAVE);

fd.setVisible(true);

}

//close help quit cut copy paste

menuFrame.msg = msg;

menuFrame.repaint();

}

public void itemStateChanged(ItemEvent ie)

{

menuFrame.repaint();

}

}

//create frame window

public class NotepadDemo extends Applet

{

Frame f;

public void init()

{

f = new MenuFrame("Menu Demo");

f.setSize(400,400);

f.setVisible(true);

}

public void start()

{

f.setVisible(true);

}

public void stop()

{

f.setVisible(false);

}

}









