Creating User Interfaces

B.Tech. (IT), Sem-5,
Core Java Technology (CJT)

Dharmsinh Desai University
Prof. (Dr.) H B Prajapati

1

Dialog

- Dialog box is used as a temporary window to receive additional information from the user.
- It is also used to provide notification about some event has occurred.
- Dialog class (subclass of Window) is used to create a dialog box.
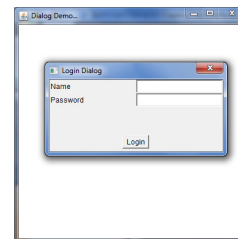- A dialog box must be associated with a frame

2

Dialog

- Constructor:
  - Dialog(Frame parent, String title, boolean modal)
- First parameter, parent, is compulsory
- Third parameter, modal, indicates whether other windows can be accessed without the Dialog box is dismissed.
- Dialog box is made visible using setVisible() method, like done with Frame.

3

Example: Dialog

4

Example: Dialog

```
import java.awt.*;
import java.awt.event.*;
class DialogDemo extends Frame{
    MyDialog md;
    String name="Sem-5";
    String password="Sem-5";
    DialogDemo(String title){
    super(title);
    setSize(400,400);
    setVisible(true);
```

5

Example: Dialog

```
    MyDialog md=new MyDialog(this,"Login Dialog",false);
    md.setSize(300,150);
    md.setVisible(true);
    md.setResizable(false);
}
public static void main(String [] args){
    DialogDemo d=new DialogDemo("Dialog Demo...");
}
```

6

Example: Dialog

```
public boolean login(String name,String password){
        if(this.name.equals(name) &&
this.password.equals(password))
                return true;
        else
                return false;
    }
}
```

7

Example: Dialog

```
class MyDialog extends Dialog implements ActionListener{
    DialogDemo d;
    TextField name,passwd;
    Button ok;
    public MyDialog(DialogDemo d,String title,boolean modal){
        super(d,title,modal);
        this.d=d;
        Panel p1=new Panel();
        p1.setLayout(new GridLayout(2,2));
```

8

Example: Dialog

```
        p1.add(new Label("Name"));
        p1.add(name=new TextField(10));
        p1.add(new Label("Password"));
        p1.add(passwd=new TextField(10));
        passwd.setEchoChar('*');
        add("North",p1);

        Panel p3=new Panel();
        ok=new Button("Login");
        p3.add(ok);
        ok.addActionListener(this);
        add("South",p3);
    }
```
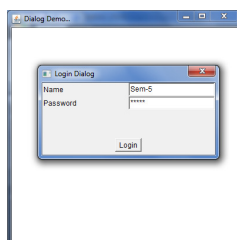
9

Example: Dialog

```
    public void actionPerformed(ActionEvent ae){
        if(d.login(name.getText(),passwd.getText())){
            this.dispose();
        }
    }
}
```
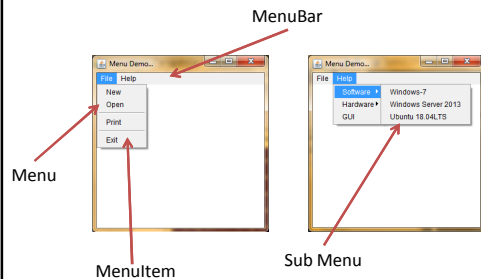
10

Example: Dialog

11

Menu

- Menu can allow selection of action to be performed in easier way.
- Menu's components do not consume space of frame. I.e., they are shown only when a menubar is activated.
- Menu can be created only on Frame.
- Menu is created using the following three classes:
  - Menu
  - MenuBar
  - MenuItem

12

2

Menu, MenuBar, and MenuItem

MenuBar

Menu

MenuItem

Sub Menu

13

Menu

- Steps of using Menu
  1. Create a MenuBar and set it for the Frame
  2. Create Menu(s) and add them in MenuBar
  3. Create MenuItem(s) and add them in Menu.
  4. Write event handling code for MenuItems

14

Menu

Step-1: Create MenuBar and set it for the frame
- MenuBar mb=new Menubar();
- frame.setMenuBar(mb);

15

Menu

Step-2: Create Menu and add them in the MenuBar.
- Constructor to create a Menu
  - public Menu(String label, boolean tearOff)
- The constructor will create a new Menu instance with the specified label and tearOff.
- The tearOff enables the programmer to create a menu that displays even when the mouse button is released.
- public Menu(String label) create a menu with the specified label. It is equivalent to Menu(label, false)

16

Menu

Step-2: Create Menu and add them in the MenuBar.
Menu fileMenu=new Menu("File", true);
Menu helpMenu=new Menu("Help", true);
mb.add(fileMenu);
mb.add(helpMenu);

17

Menu

Step-3: Create MenuItems and add them in Menu(s).
fileMenu.add(new MenuItem("New"));
fileMenu.add(new MenuItem("Open"));
fileMenu.add(new MenuItem("-"));
fileMenu.add(new MenuItem("Print"));
fileMenu.add(new MenuItem("-"));
fileMenu.add(new MenuItem("Exit"));

The character – specified in the MenuItem separates menu items. We can also use the addSeparator() method
fileMenu.addSeparator() to separate menu items.

18

Menu

Step-3: Create MenuItems and add them in Menu(s).
We can also add Menu(s) inside another Menu.
Menu swHelpSubMenu = new Menu("Software");
Menu hwHelpSubMenu = new Menu("Hardware");
helpMenu.add(swHelpSubMenu);
helpMenu.add(hwHelpSubMenu);
swHelpSubMenu.add(new MenuItem("Windows-7"));
swHelpSubMenu.add(new MenuItem("Windows Server 2013"));
swHelpSubMenu.add(new MenuItem("Ubuntu 18.04LTS"));

19

Menu

Step-3: Create MenuItems and add them in Menu(s).
- We can add CheckboxMenuItem to a Menu.
- CheckboxMenuItem is a subclass of MenuItem
- It adds a boolean state to the MenuItem, and displays a check when its state is true.
- We can click the menu item to turn it on and off.
helpMenu.add(new CheckboxMenuItem("GUI"));

20

Menu

Step-4: Write event handling code for MenuItems
- MenuItems generate ActionEvent object
- Register all MenuItems for ActionListener
- Handle events similar to that of Button

21

Example: Menu Demo

```
import java.awt.*;
import java.awt.event.*;
class MenuDemo extends Frame{
    MenuBar mb;
    Menu fileMenu;
    Menu helpMenu;
    public static void main(String[] args){
        Frame f=new MenuDemo("Menu Demo...");
        f.setSize(300,300);
        f.setVisible(true);
    }
```

22

Example: Menu Demo

```
MenuDemo(String title){
    super(title);
    mb=new MenuBar();
    setMenuBar(mb);

    Menu fileMenu=new Menu("File", true);
    Menu helpMenu=new Menu("Help", true);
    mb.add(fileMenu);
    mb.add(helpMenu);
```

23

Example: Menu Demo

```
    fileMenu.add(new MenuItem("New"));
    fileMenu.add(new MenuItem("Open"));
    fileMenu.add(new MenuItem("-"));
    fileMenu.add(new MenuItem("Print"));
    fileMenu.add(new MenuItem("-"));
    fileMenu.add(new MenuItem("Exit"));
```

24

Example: Menu Demo

```
    Menu swHelpSubMenu = new Menu("Software");
    Menu hwHelpSubMenu = new Menu("Hardware");
    helpMenu.add(swHelpSubMenu);
    helpMenu.add(hwHelpSubMenu);
    swHelpSubMenu.add(new MenuItem("Windows-7"));
    swHelpSubMenu.add(new MenuItem("Windows Server
2013"));
    swHelpSubMenu.add(new MenuItem("Ubuntu
18.04LTS"));
    helpMenu.add(new CheckboxMenuItem("GUI"));
  }
}
```

25

Example: GUI Calculator
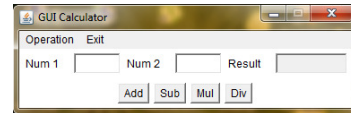
26

Example: GUI Calculator

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

class GUIMenuCal extends Frame implements ActionListener{
    TextField tf1;
    TextField tf2;
    TextField tf3;
    Menu mo;
    Menu me;
    MenuBar mb;
    MenuItem madd,msub,mmul,mdiv,mexit;
    Button add,sub,mul,div;
```

27

Example: GUI Calculator

```
public static void main(String[] args){
    Frame f=new GUIMenuCal("GUI Calculator");
    f.pack();
    f.setVisible(true);
}
GUIMenuCal(String title){
    super(title);
    tf1=new TextField(4);
    tf2=new TextField(4);
    tf3=new TextField(8);
    tf3.setEditable(false);
```

28

Example: GUI Calculator

```
    Panel p=new Panel();
    p.add(new Label("Num 1"));
    p.add(tf1);
    p.add(new Label("Num 2"));
    p.add(tf2);
    p.add(new Label("Result"));
    p.add(tf3);

    Panel pb=new Panel();
    pb.add(add=new Button("Add"));
    pb.add(sub=new Button("Sub"));
    pb.add(mul=new Button("Mul"));
    pb.add(div=new Button("Div"));
```

29

Example: GUI Calculator

```
    add("North",p);
    add("South",pb);
    mb=new MenuBar();
    setMenuBar(mb);
    mo=new Menu("Operation");
    me=new Menu("Exit");
    madd=new MenuItem("Add");
    msub=new MenuItem("Sub");
    mmul=new MenuItem("Mul");
    mdiv=new MenuItem("Div");
    mexit=new MenuItem("Exit");
```

30

Example: GUI Calculator

```
mo.add(madd);
mo.add(msub);
mo.add(mmul);
mo.add(mdiv);
me.add(mexit);
mb.add(mo);
mb.add(me);

add.addActionListener(this);
sub.addActionListener(this);
mul.addActionListener(this);
div.addActionListener(this);
```

31

Example: GUI Calculator

```
madd.addActionListener(this);
msub.addActionListener(this);
mmul.addActionListener(this);
mdiv.addActionListener(this);
mexit.addActionListener(this);

setResizable(false);
}
```

32

Example: GUI Calculator

```
public void actionPerformed(ActionEvent e){
        String arg=e.getActionCommand();
        int no1,no2;
        no1=Integer.parseInt(tf1.getText().trim());
        no2=Integer.parseInt(tf2.getText().trim());
        if(arg.equals("Add")){
                int result=no1+no2;
                tf3.setText(""+result);
        }
```

33

Example: GUI Calculator

```
else if(arg.equals("Sub")){
        int result=no1-no2;
        tf3.setText(""+result);
}
else if(arg.equals("Mul")){
        int result=no1*no2;
        tf3.setText(""+result);
 }
```
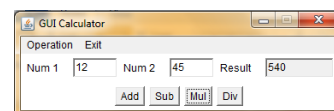
34

Example: GUI Calculator

```
else if(arg.equals("Div")){
        double result=((no1*1.0)/no2);
        tf3.setText(""+result);
}
else
        System.exit(0);
  }
}
```

35

Example: GUI Calculator

36