Java AWT

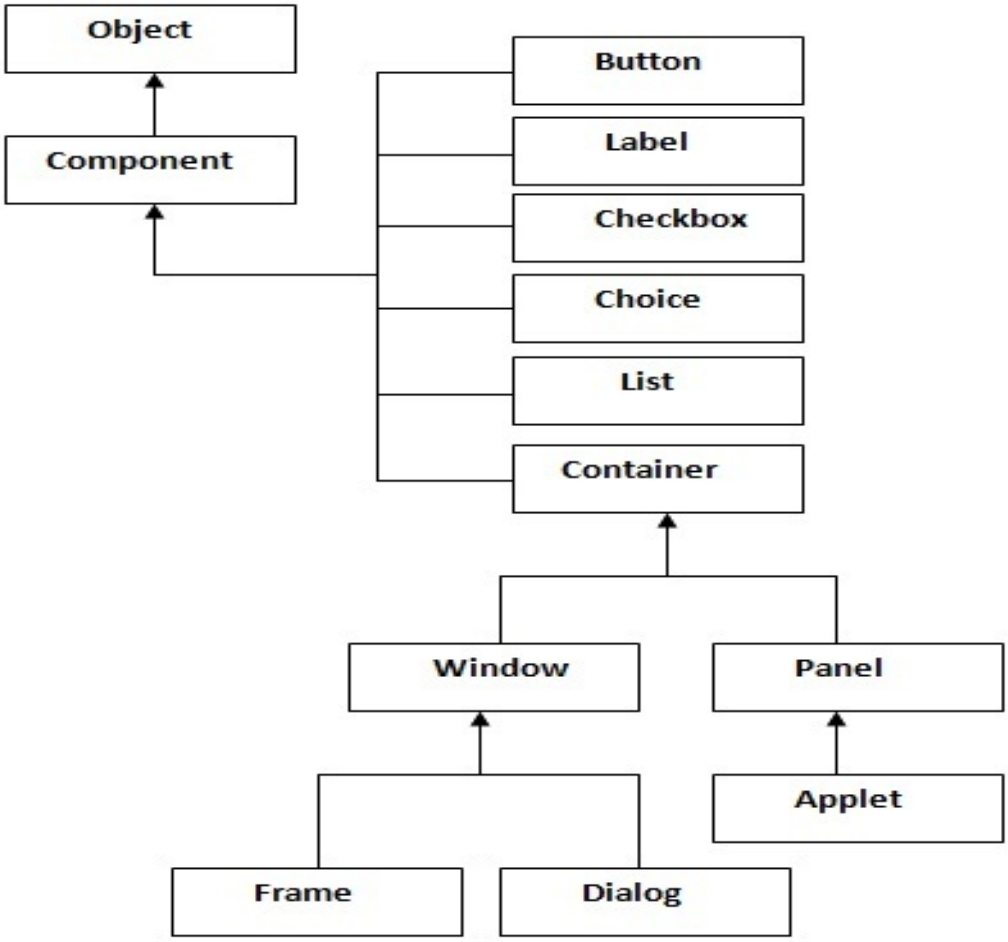
Java AWT (Abstract Windowing Toolkit) is an API to develop GUI or window-based application in java.

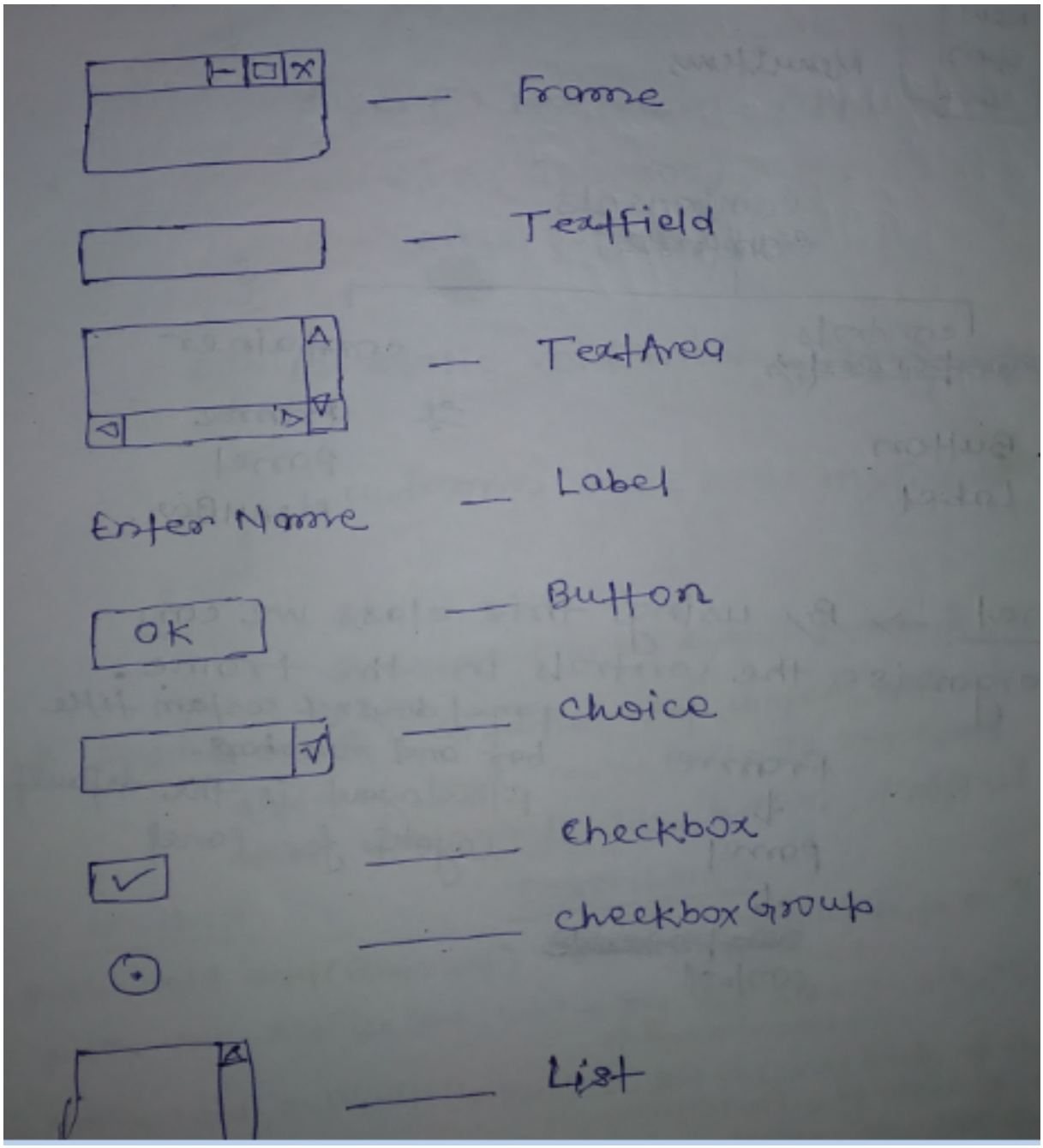
Java AWT components are platform-dependent i.e. components are displayed according to the view of operating system. AWT is heavyweight i.e. its components uses the resources of system.

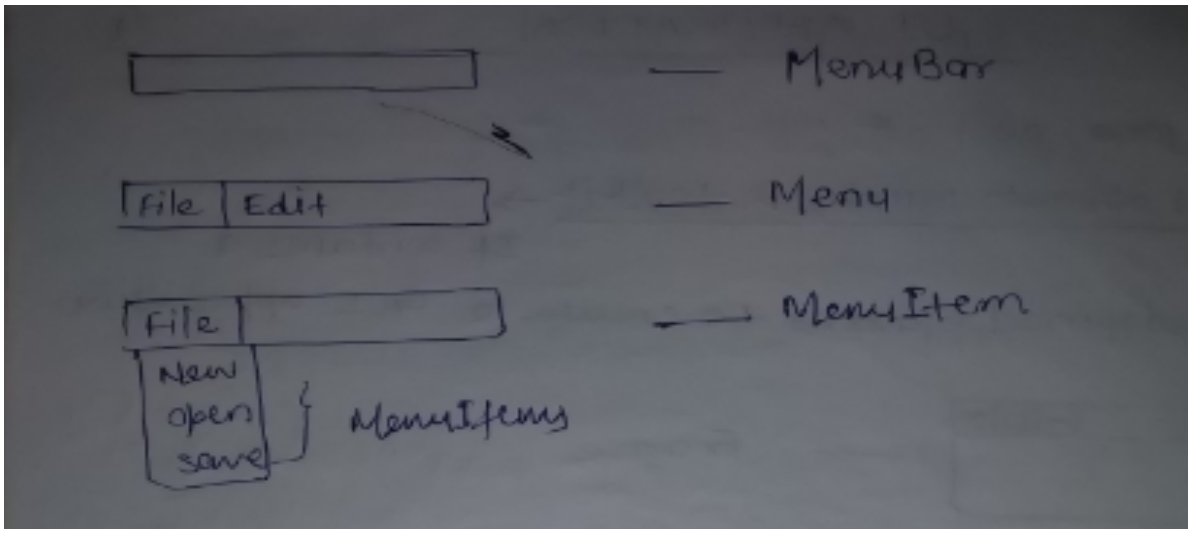
The java.awt package provides classes for AWT api such as TextField, Label, TextArea, RadioButton, CheckBox, Choice, List etc.

Java AWT Hierarchy

The hierarchy of Java AWT classes are given below.




Frame
Textfield
TextArea
Enter Name
Label
OK
Button
Choice
Checkbox
Checkbox Group
List



Container

The Container is a component in AWT that can contain another component like Buttons, TextField, Label etc.

Panel

By using "**Panel"** class we can organize the component like Button, TextField etc in the container i.e Frame or Applet. The Panel is the container that doesn't contain **title bar** and **menu bars**. "**FlowLayout"** is the default layout for Panel.

Frame

The Frame is the container that contain **title bar** and **menu bars**. It can have other components like Button, TextField etc. "BorderLayout" is the default layout for Frame.

Useful Methods of Component class

Method	Description
public void add(Component c)	inserts a component on this component.
public void setSize(int width,int height)	sets the size (width and height) of the component.
public void setLayout(LayoutManager m)	defines the layout manager for the component.
public void setVisible(boolean status)	changes the visibility of the component, by default false.

To create AWT application, you need a Frame. There are two ways to create a Frame in AWT.

1. By extending Frame class (inheritance)
2. By creating the object of Frame class

Example : By extending Frame class

```
import java.awt.*;

class DemoFrame extends Frame{

DemoFrame(){

Button b=new Button("click me");

b.setBounds(30,100,80,30);// setting button position


add(b);//adding button into frame

setSize(300,300);//frame size 300 width and 300 height

setLayout(null);//no layout manager

setVisible(true);//now frame will be visible, by default not visible

}

public static void main(String args[]){

new DemoFrame();

}}
```

Note: The **setBounds(int xaxis, int yaxis, int width, int height)** method is used to sets the position of the awt button.

Example : By creating the object of Frame class

```
import java.awt.*;

class DemoFrame{

DemoFrame{

Frame f=new Frame();

Button b=new Button("click me");

b.setBounds(30,50,80,30);

 f.add(b);

f.setSize(300,300);

f.setLayout(null);

f.setVisible(true);

}

public static void main(String args[]){

new DemoFrame();

}}
```

TextField

TextField class is a text component that allows the editing of a single line text.

Eg:

Constructor

TextField(int columns)

Constructs a new empty text field with the specified number of columns.

```
TextField t1;
t1=new TextField(20);

p1.add(t1);
```

*To accept password character in TextField

```
t1.setEchoChar('*');
```

*To retrieve the data from the TextField

```
t1.getText();
```

*To set the data in the TextField

```
t1.setText();
```

Panel

The Panel is a simplest container class. It provides space in which an application can attach any other component. Or you can say that the Panel is used to organize the controls in the Frame.

Constructor

Panel()

Creates a new panel using the default layout manager.

```
Panel p1;
p1=new Panel();     or   p1=new Panel(null);
```

*To add panel to the Frame

```
add(p1);
```

*To add other component in the Panel

```
p1.add(t1);
```

Label

Label class is a component for placing text in a container. It is used to display a single line of read only text. The text can be changed by an application but a user cannot edit it directly.

Constructor

Label(String text)

Constructs a new label with the specified string of text, left justified.

```
Eg:
Label lblrno;
lblrno=new Label("Enter the Roll Number:");
```

```
p1.add(lblrno);
```

Label()

Constructs an empty label.

Eg:

```
Label lblrno=new Label();
```

*To set the caption to Label

```
lblrno.setText("Enter the Roll Number");
```

eg:

```
Label lblrno=new Label();
lblrno.setText("Enter the Roll Number");
```

Button

The button class is used to create a labeled button. The application result in some action when the button is pushed.

Button(String text)

Constructs a new button with specified label.

Eg:

```
Button btn;
btn=new Button("OK");

p1.add(btn);
```

Button()

Constructs a button with an empty string for its label.

Eg:

```
Button btn=new Button();
```

*To set the caption to Button

```
btn.setLabel("OK");
```

eg:

```
Button btn=new Button();
btn.setLabel("OK");
```

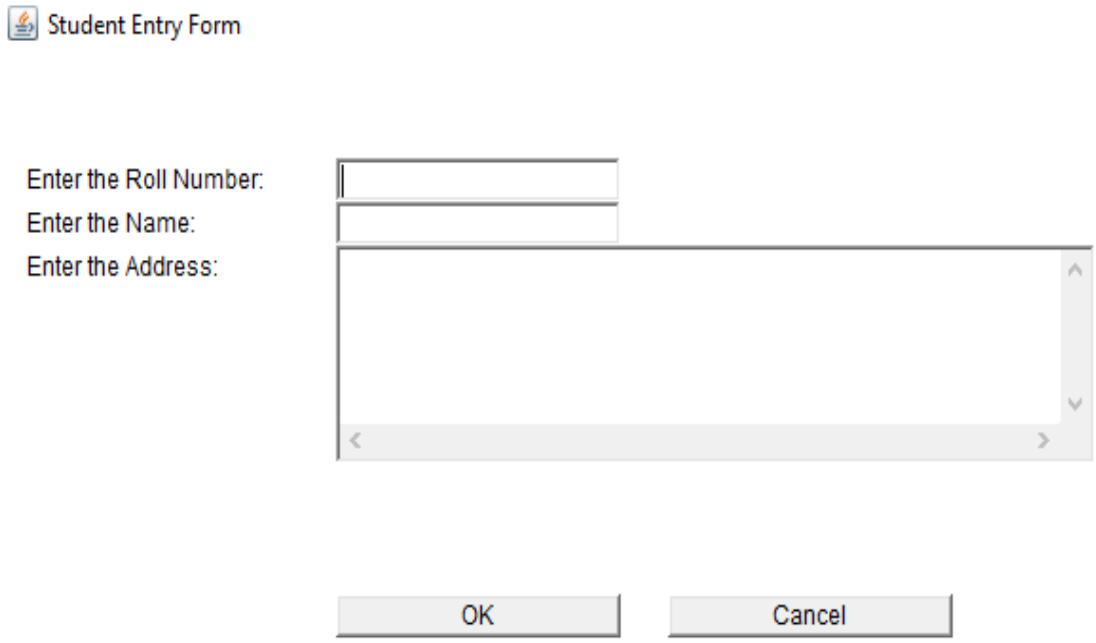
TextArea

TextArea class is a multi line region that displays text.

TextArea(int rows, int columns)
Constructs a new text area with the specified number of rows and columns and the empty string as text.

Eg:
TextArea ta;
ta=new TextArea(4,4);

Q: Demonstrate the AWT program for the following Student Entry Form using Frame



import java.awt.*;

import java.awt.event.*;

```
public class StudentForm extends Frame {
    Label lblRno,lblName,lblAddr;
    TextField txtRno,txtName;
    TextArea txtAddr;
    Button btnOk,btnCancel;
    Panel p1;
    public StudentForm()
    {
        super("Student Entry Form");
        lblRno=new Label("Enter the Roll Number:");
        lblRno.setBounds(10,50,150,20);
        lblName=new Label("Enter the Name:");
        lblName.setBounds(10,70,150,20);
        lblAddr=new Label("Enter the Address:");
        lblAddr.setBounds(10,90,150,20);
        txtRno=new TextField(20);
        txtRno.setBounds(175,50,150,20);
        txtName=new TextField(20);
        txtName.setBounds(175,70,150,20);
        txtAddr=new TextArea(5,4);
        txtAddr.setBounds(175,90,400,100);
        btnOk=new Button("OK");
        btnOk.setBounds(175,250,150,20);
        btnCancel=new Button("Cancel");
        btnCancel.setBounds(350,250,150,20);
```

```
        p1=new Panel(null);
        add(p1);
        p1.add(lblRno); p1.add(txtRno);
        p1.add(lblName); p1.add(txtName);
        p1.add(lblAddr); p1.add(txtAddr);
        p1.add(btnOk);p1.add(btnCancel);
        setSize(400, 400);
        setVisible(true);
    }
    public static void main(String[] args) {
        StudentForm sf=new StudentForm();
    }
}
```