



SNS COLLEGE OF ENGINEERING



Kurumbapalayam(Po), Coimbatore - 641 107

Accredited by NAAC-UGC with 'A' Grade

Approved by AICTE, Recognized by UGC & Affiliated to Anna University, Chennai

Department of Information Technology

19CS204 OBJECT ORIENTED PROGRAMMING

I YEAR /II SEMESTER

Topic - Java Applets





Applet

- Applet is a special type of java program that is embedded in the webpage to generate the dynamic content.
- It runs inside the browser and works at client side.
- It works at client side so less response time.
- Secured
- It can be executed by browsers running under many platforms, including Linux, Windows, Mac Os etc.



Applet



- An applet is a Java class that extends the `java.applet.Applet` class.
- A `main()` method is not invoked on an applet, and an applet class will not define `main()`.
- Applets are designed to be embedded within an HTML page.
- When a user views an HTML page that contains an applet, the code for the applet is downloaded to the user's machine.

Applet

- Applet is initialized.
- Applet is started.
- Applet is painted.
- Applet is stopped.
- Applet is destroyed





Applet

- **init** – This method is intended for whatever initialization is needed for your applet. It is called after the param tags inside the applet tag have been processed.
- **start** – This method is automatically called after the browser calls the init method. It is also called whenever the user returns to the page containing the applet after having gone off to other pages.
- **stop** – This method is automatically called when the user moves off the page on which the applet sits. It can, therefore, be called repeatedly in the same applet.
- **destroy** – This method is only called when the browser shuts down normally. Because applets are meant to live on an HTML page, you should not normally leave resources behind after a user leaves the page that contains the applet.
- **paint** – Invoked immediately after the start() method, and also any time the applet needs to repaint itself in the browser. The paint() method is actually inherited from the java.awt.



Applet

```
import java.applet.Applet;  
import java.awt.Graphics;  
public class First extends Applet{  
  
    public void paint(Graphics g){  
        g.drawString("welcome",150,150);  
    }  
  
}
```

```
<html>  
<body>  
<applet code="First.class" width="300" height="300">  
</applet>  
</body>  
</html>
```

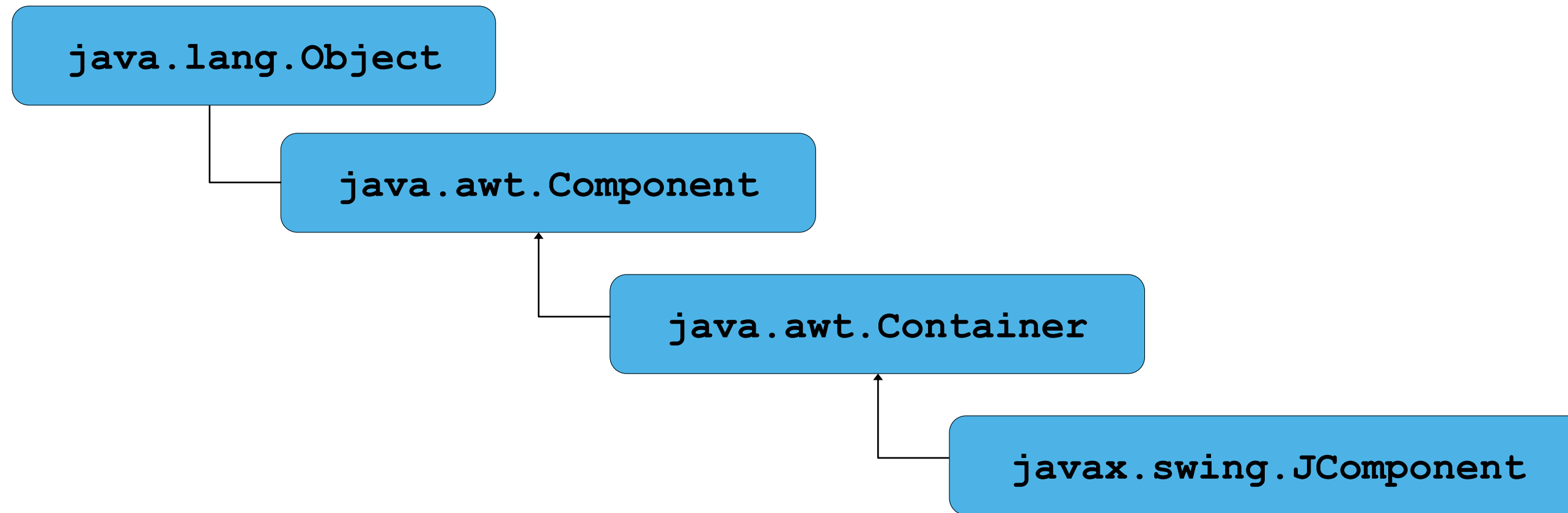



Swing



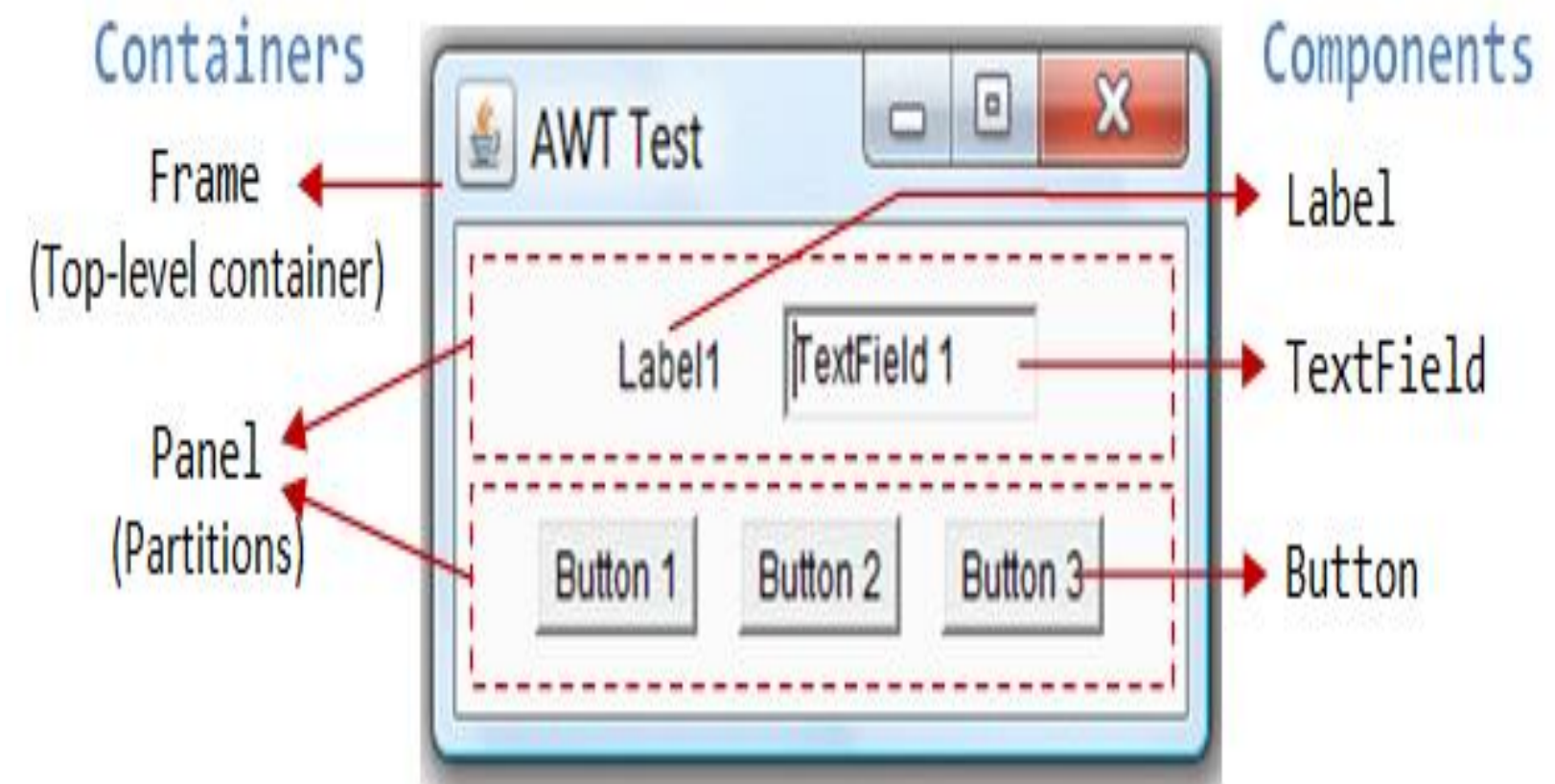
- Swing in java is part of Java foundation class which is lightweight and platform independent.
- It is used for creating window based applications. It includes components like button, scroll bar, text field etc. Putting together all these components makes a graphical user interface.
- Defined in package javax.swing
- Original GUI components from AWT in java.awt
- Heavyweight components - rely on local platform's windowing system for look and feel
- Swing components are lightweight
- Not weighed down by GUI capabilities of platform
- More portable than heavyweight components
- Swing components allow programmer to specify look and feel
- Can change depending on platform
- Can be same across all platforms

Swing

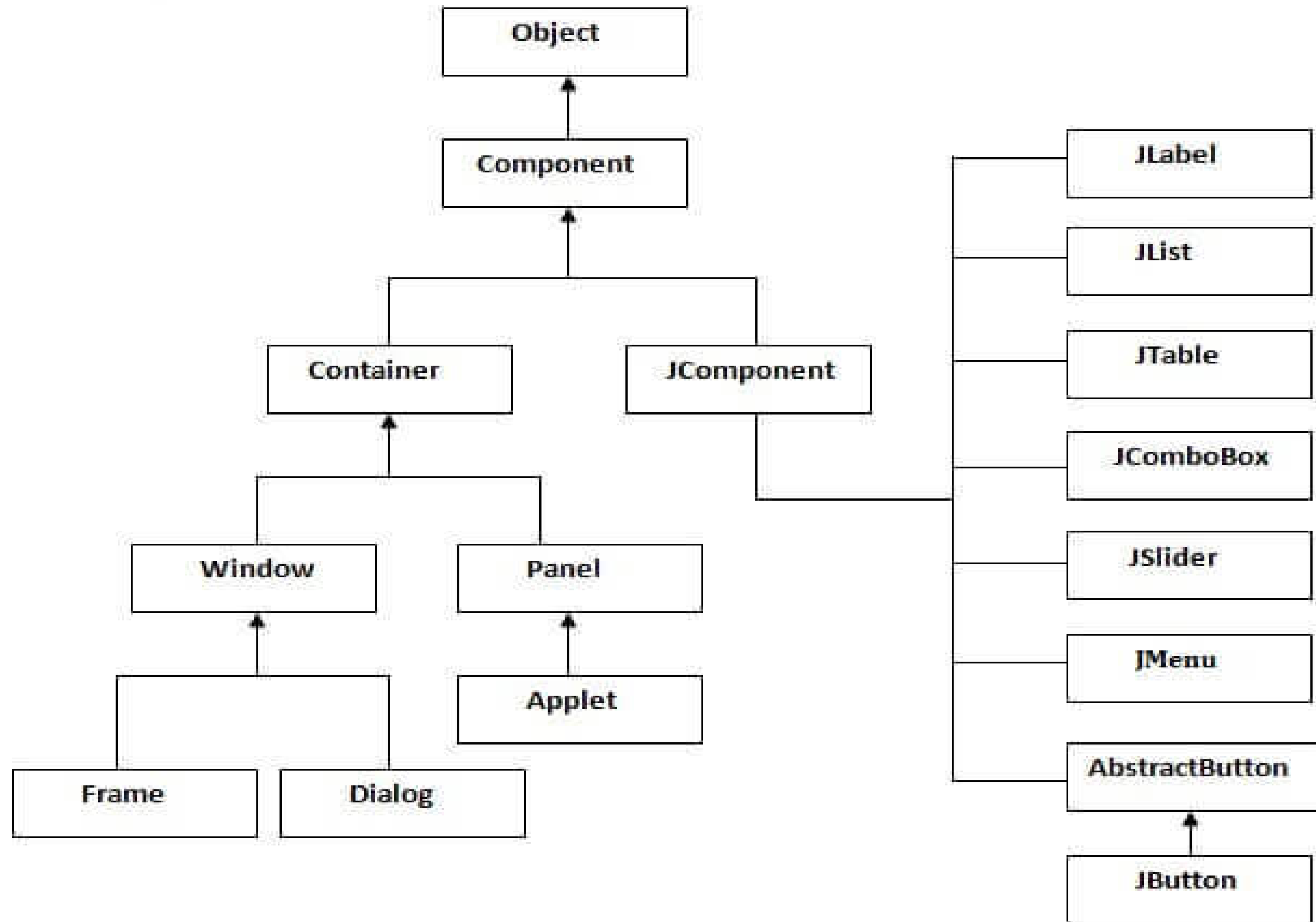


Swing

- **Component** defines methods used in its subclasses (for example, **paint** and **repaint**)
- **Container** - collection of related components
 - When using **JFrame**, add components to content pane (a **Container**)
- **JComponent** - superclass to most Swing components



Swing





Swing



Jbutton Class

```
import javax.swing.*;
public class Simple {
    JFrame f;
    Simple(){
        f=new JFrame();//creating instance of JFrame

        JButton b=new JButton("click");//creating instance of JButton
        b.setBounds(130,100,100, 40);

        f.add(b);//adding button in JFrame

        f.setSize(400,500);//400 width and 500 height
        f.setLayout(null);//using no layout managers
        f.setVisible(true);//making the frame visible
    }

    public static void main(String[] args) {
        new Simple();
    }
}
```



Swing



JTextField Class

It inherits the JTextComponent class and it is used to allow editing of single line text.

Example

```
import javax.swing.*.*;
public class Text{
public static void main(String args[]) {
JFrame a = new JFrame("Textfield");
JTextField b = new JTextField("Example for text field");
b.setBounds(50,100,200,30);
a.add(b);
a.setSize(300,300);
a.setLayout(null);
a.setVisible(true);
}
}
```



THANK YOU