

Course Name: A Level (2nd Sem)

Topic: Swing

Subject: JAVA

Date: 17-06-20

Swing:

Swing is the principal GUI toolkit for the Java programming language. It is a part of the JFC (Java Foundation Classes), which is an API for providing a graphical user interface for Java programs. It is completely written in Java. Swing, which is an extension library to the AWT, includes new and improved components that enhance the look and functionality of GUIs. It employs MVC (model-view-controller). Swing is more portable and more flexible than AWT.

What is MVC?

The MVC is an architectural design pattern that separates an application into three main logical components ModelViewController.

Model: Model represents the shape of the data. A class in Java is used to describe a model. Model objects store data retrieved from the database.

View: View in MVC is a user interface. View displays model data to the user and also enables them to modify them. View makes it easy to communicate with the model and the controller.

Controller: The controller handles the user request. Typically, the user uses the view and raises any request, which is handled by the controller. The controller processes the request and returns the appropriate view as a response.

What is container class?

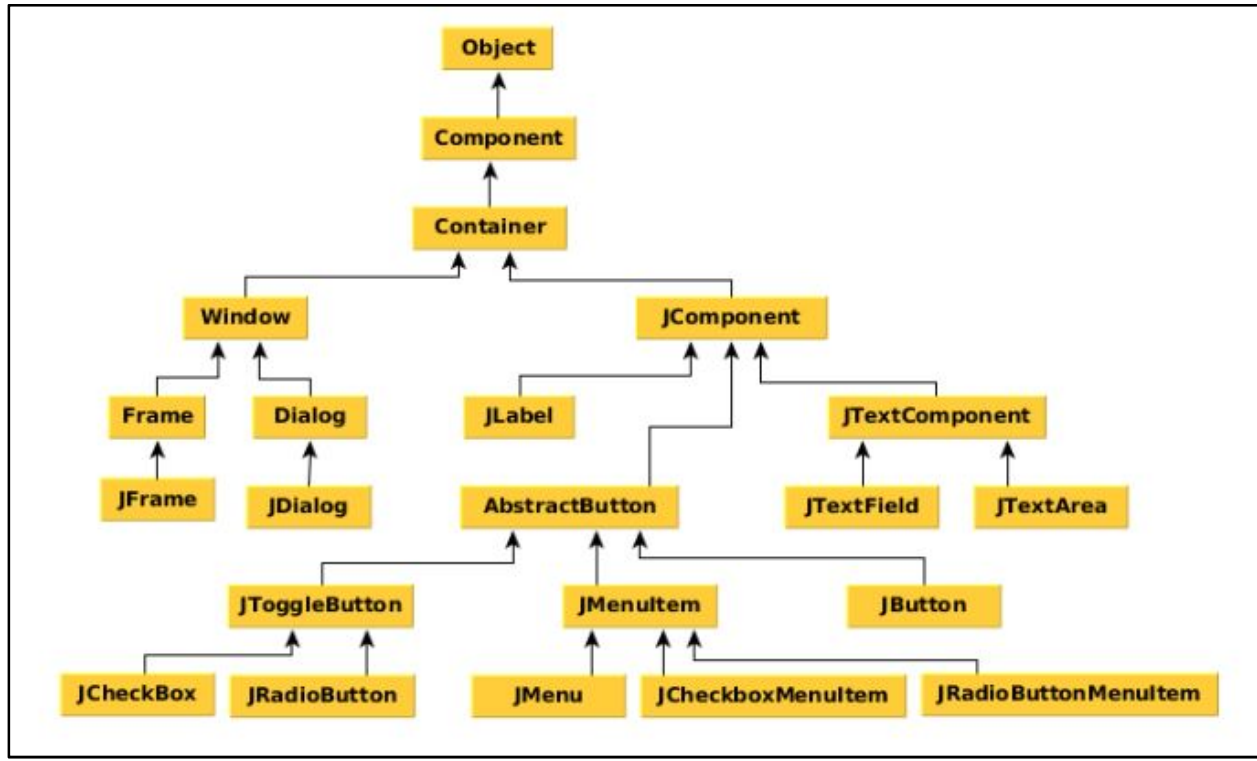
Container classes are classes that can have other components on it. So for creating a GUI, we need at least one Container object. There are three types of containers:

1) Panel : It is a pure container and is not a window in itself. The sole purpose of a Panel is to organize the components on to a window.

2) Frame : It is a fully functioning window with its own title and icons.

3) Dialog : It can be thought of as a pop-up window that pops out when a message has to be displayed. It is not a fully functioning window like the Frame.

SWING Hierarchy:



1)SWING BUTTON

```
import javax.swing.*;
```

```
public class gui {  
    public static void main(String[] args) {  
        JFrame f=new JFrame("Buttonex");  
        JButton b=new JButton("Click Here");  
        b.setBounds(50,100,95,30);  
        f.add(b);  
        f.setSize(400,400);  
        f.setLayout(null);  
        f.setVisible(true);  
    }  
}
```

2)Adding Icon to Jbutton

```
import javax.swing.*;
public class IconButton{
    IconButton(){
        JFrame f=new JFrame("Button Example");
        JButton b=new JButton(new ImageIcon("D:\\A.JPG"));
        b.setBounds(100,100,100, 40);
        f.add(b);
        f.setSize(300,400);
        f.setLayout(null);
        f.setVisible(true);
        f.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    }
    public static void main(String[] args) {
        new IconButton();
    }
}
```

← It directly closes the frame

3)JpasswordField

```
import javax.swing.*;

public class JPWD extends JFrame {
    public static void main(String[] args) {
        JFrame.setDefaultLookAndFeelDecorated(true);
        JFrame frame = new JFrame();
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.setTitle("JTextField Test");
        frame.setLayout(new GridLayout(2, 2));
        JLabel label = new JLabel("User Name:");
        JLabel label2 = new JLabel("Password:");
        JTextField un = new JTextField(20);
        JPasswordField jp = new JPasswordField();
        frame.add(label);
        frame.add(un);
        frame.add(label2);
        frame.add(jp);
        frame.setSize(200, 70);
        frame.setVisible(true);
    }
}
```

4)Java JTextArea

```
import javax.swing.*;
public class Ta
{
    Ta(){
        JFrame f= new JFrame();
        JTextArea area=new JTextArea("A LEVEL CLASS");
        area.setBounds(10,30, 200,200);
        f.add(area);
        f.setSize(300,300);
        f.setLayout(null);
        f.setVisible(true);
    }
    public static void main(String args[])
    {
        new Ta();
    }
}
```

5)Java Jtable

```
import javax.swing.*;
public class TE {
    JFrame f;
    TE (){
        f=new JFrame();
        String data[][]={ {"007","James Camroon","Avatar"},
                           {"006","Tim Burton","Batman"},
                           {"009","Nolan","inception"} };
        String column[]={"ID","NAME","Movie"};
        JTable jt=new JTable(data,column);
        jt.setBounds(30,40,200,300);
        JScrollPane sp=new JScrollPane(jt);
        f.add(sp);
        f.setSize(300,400);
        f.setVisible(true);
    }
    public static void main(String[] args) {
        new TE ();
    }
}
```

Table data is entered in the form of 2d array

Column name

It provides scrollable view of a component