JavaKit Doc

About

Javakit is about faster development in desktop applications while creating less code. With this library you can create simple applications within minutes.

Docs

Components

Window(JFrame)

The first 5 parameters are all about the window or JFrame .

1. X
2. Y
3. Width
4. Height
5. Title

The last 5 parameters are all about the content of the window.

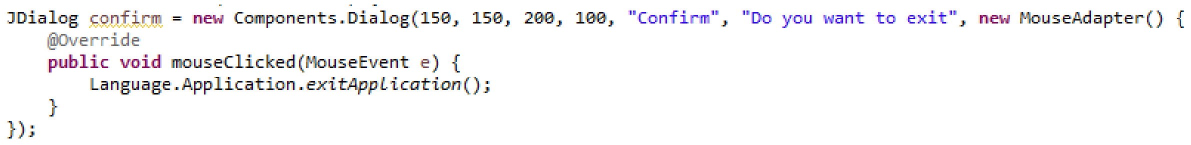
1. North Panel
2. East Panel
3. South Panel
4. West Panel
5. Center Panel

```
JFrame win = new Components.Window(0, 0, 1000, 600, "Title", DataEntry, new JPanel(), new
JPanel(), new JPanel(),DataTable);
```

Dialog(JDialog)

Note: There are downsides to this component, but still very useful.

There are two ways to use this component. The code looks different from the Window component, because it has a mouse event(s) attached to it.



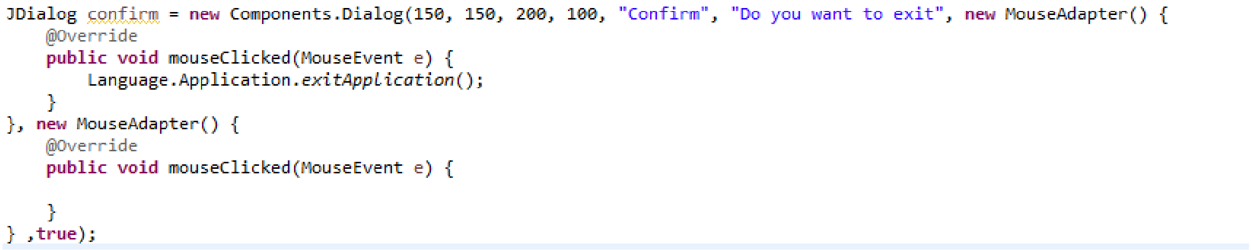
```
JDialog confirm = new Components.Dialog(150, 150, 200, 100, "Confirm", "Do you want to exit", new MouseAdapter() {
    @Override
    public void mouseClicked(MouseEvent e) {
        Language.Application.exitApplication();
    }
});
```

The first 6 parameters are almost like the Window component

1. X
2. Y
3. Width
4. Height
5. Title
6. Message

In the first image there is only one mouse event which is the 7th parameter. In the image below there is two more parameters

1. Another mouse event
2. Whether or not to close the Dialog component



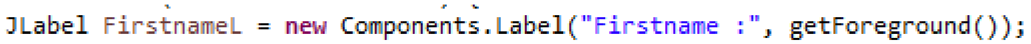
```
JDialog confirm = new Components.Dialog(150, 150, 200, 100, "Confirm", "Do you want to exit", new MouseAdapter() {
    @Override
    public void mouseClicked(MouseEvent e) {
        Language.Application.exitApplication();
    }
}, new MouseAdapter() {
    @Override
    public void mouseClicked(MouseEvent e) {

    }
} ,true);
```

The only downside is that if the last parameter is FALSE, there is no efficient way to dispose of it. With this said there are other properties in this component, that I will not go through because this is a starting guide, a more in-depth version will go into these factors.

Label(JLabel)

There are only two parameters to this component.

1. Text
2. Foreground – Text color

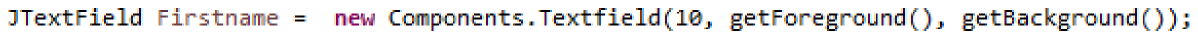


```
JLabel FirstnameL = new Components.Label("Firstname :", getForeground());
```

Textfield(JTextField)

There are only three parameters to this component.

1. Columns
2. Foreground – text color
3. Border color



```
JTextField Firstname =  new Components.Textfield(10, getForeground(), getBackground());
```

Textarea(JTextArea)

There are only 4 parameters to this component.

1. Columns
2. Rows
3. Foreground – text color
4. Border color

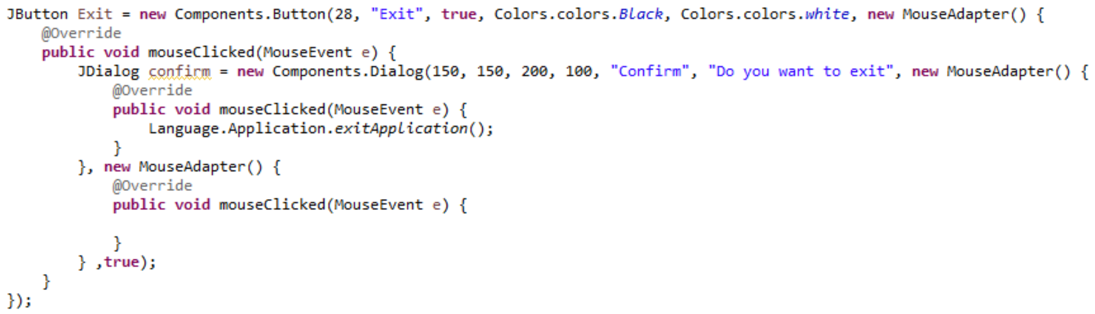
Sorry there is no image for this component.

Button(JButton)

Honestly, this component is fun to use because of the events you can do with just this one component. This component has a custom graphics that are pre-defined. There are only six parameters to this component.

1. Width

2. Text
3. Gradient Boolean
4. Foreground – text color
5. Border color
6. Mouse event

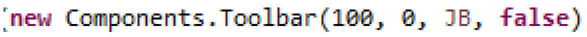


```
JButton Exit = new Components.Button(28, "Exit", true, Colors.colors.Black, Colors.colors.white, new MouseAdapter() {
    @Override
    public void mouseClicked(MouseEvent e) {
        JDialog confirm = new Components.Dialog(150, 150, 200, 100, "Confirm", "Do you want to exit", new MouseAdapter() {
            @Override
            public void mouseClicked(MouseEvent e) {
                Language.Application.exitApplication();
            }
        }, new MouseAdapter() {
            @Override
            public void mouseClicked(MouseEvent e) {

            }
        } ,true);
    }
});
```

Toolbar(JToolBar)

This component has more to it than you may think, but a very powerful way to have a better application. This component only has 4 parameters.

1. Width
2. Height
3. JButton or Button array
4. Mouse event



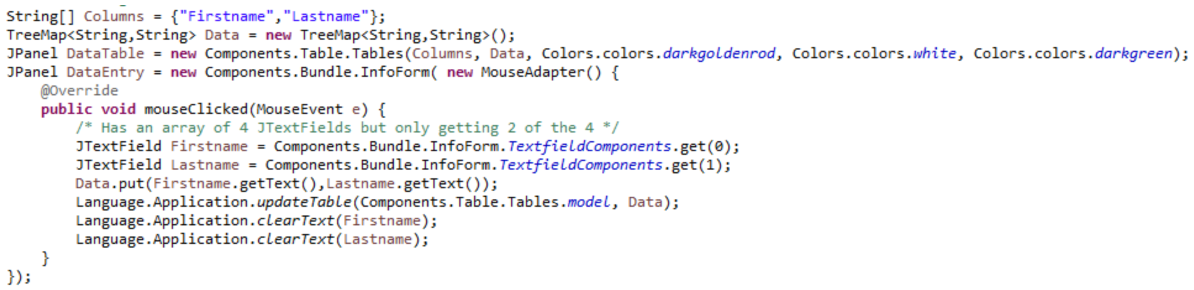
```
new Components.Toolbar(100, 0, JB, false)
```

Tabs

Table(JTable)

The nice thing about this component is it's endless use. There are 5 parameters to this component.

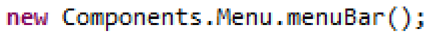
1. Columns text array
2. TreeMap<String,String> for data
    a. A String array for a third column. Note: the is no image for the three column table
3. Color for the grid
4. Color for the background
5. Color for the foreground



```
String[] Columns = {"Firstname","Lastname"};
TreeMap<String,String> Data = new TreeMap<String,String>();
JPanel DataTable = new Components.Table.Tables(Columns, Data, Colors.colors.darkgoldenrod, Colors.colors.white, Colors.colors.darkgreen);
JPanel DataEntry = new Components.Bundle.InfoForm( new MouseAdapter() {
    @Override
    public void mouseClicked(MouseEvent e) {
        /* Has an array of 4 JTextFields but only getting 2 of the 4 */
        JTextField Firstname = Components.Bundle.InfoForm.TextfieldComponents.get(0);
        JTextField Lastname = Components.Bundle.InfoForm.TextfieldComponents.get(1);
        Data.put(Firstname.getText(),Lastname.getText());
        Language.Application.updateTable(Components.Table.Tables.model, Data);
        Language.Application.clearText(Firstname);
        Language.Application.clearText(Lastname);
    }
});
```

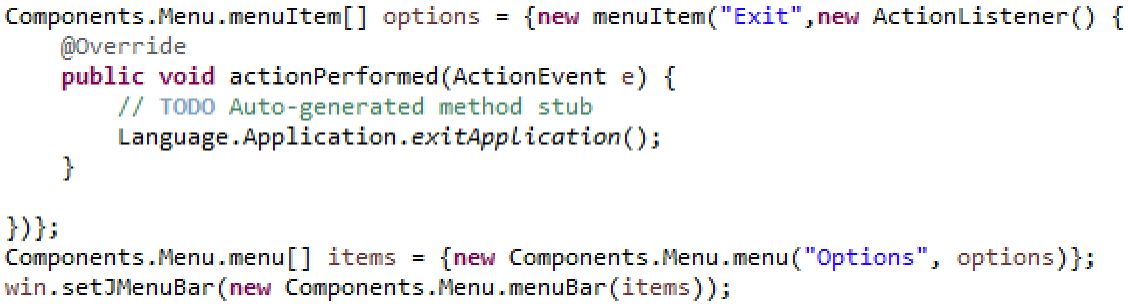
Menu (JMenuBar)

There are two ways you can use this component and I'll cover both ways.

1. 

   ```
   new Components.Menu.menuBar();
   ```

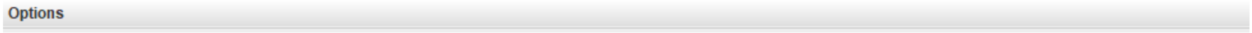
   This method will only create a options dropdown with a exit item to end the application.
2. The second way has more too it but, useful and the image really covers it, but I'll explain it anyway



```
Components.Menu.menuItem[] options = {new menuItem("Exit",new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent e) {
        // TODO Auto-generated method stub
        Language.Application.exitApplication();
    }

})};
Components.Menu.menu[] items = {new Components.Menu.menu("Options", options)};
win.setJMenuBar(new Components.Menu.menuBar(items));
```

Options is an array that has the dropdown item. With that said, there is 2 parameters a String and an ActionListener. For every dropdown item you will need to do this.

Items is the array that basically list the items of the menu. This image will show the basic output except the exit button.



I find the second way is the best way to do it.

Bundles

This section is just to cover something I added that is going to be expanded. It's really just a bunch of these components combined in to just one big whole component. You may have to add a mouse listener which you have already done if you use any of these components already.

Colors and Language

Colors

It has all the colors you may need, it's still expanding and needs work for names, but I've been using these colors for all my applications no matter what.

Application and Console

Basic functions for console and application that I will go over in the next document.