Tutorial of Swing

> source code are based on lab materials of "sustech-teaching group"
>
> Document designed by ZHU Yueming in 2023. May. 6th

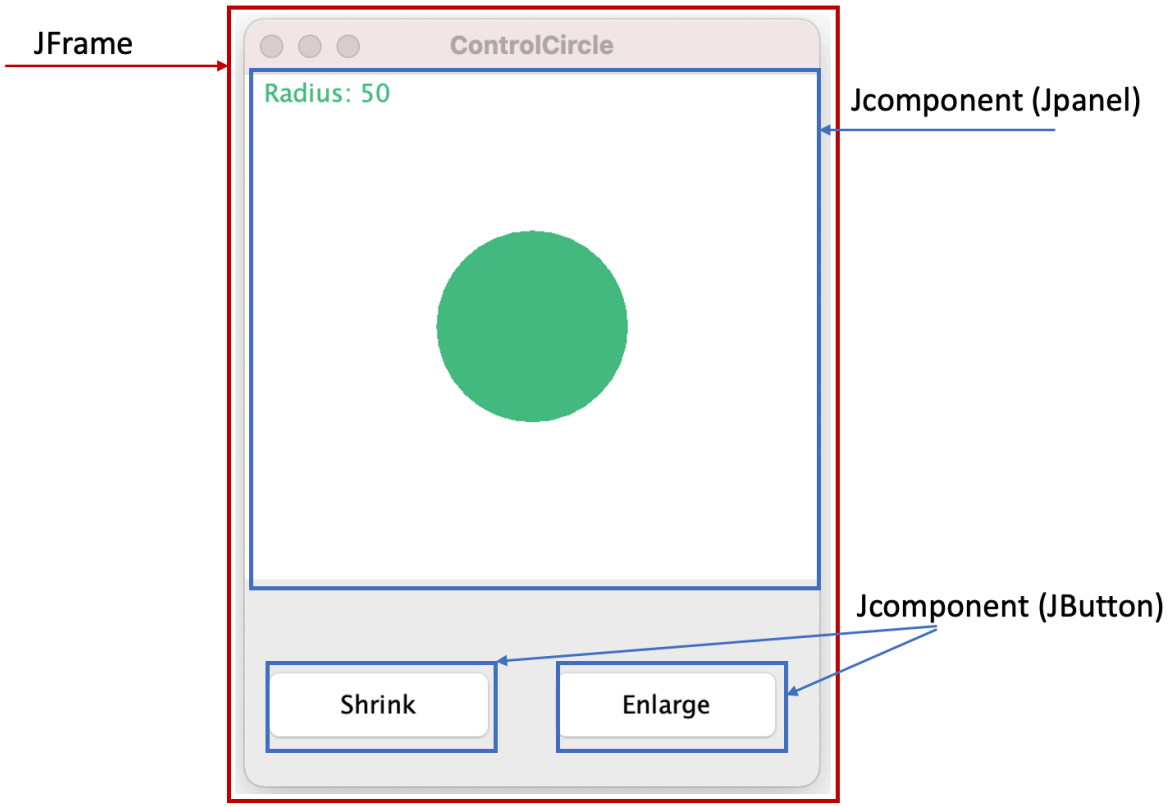
Objective

- Understand JFrame and JComponent
- Understand x and y coordinate in swing

Introduction

1. JFrame and JComponent

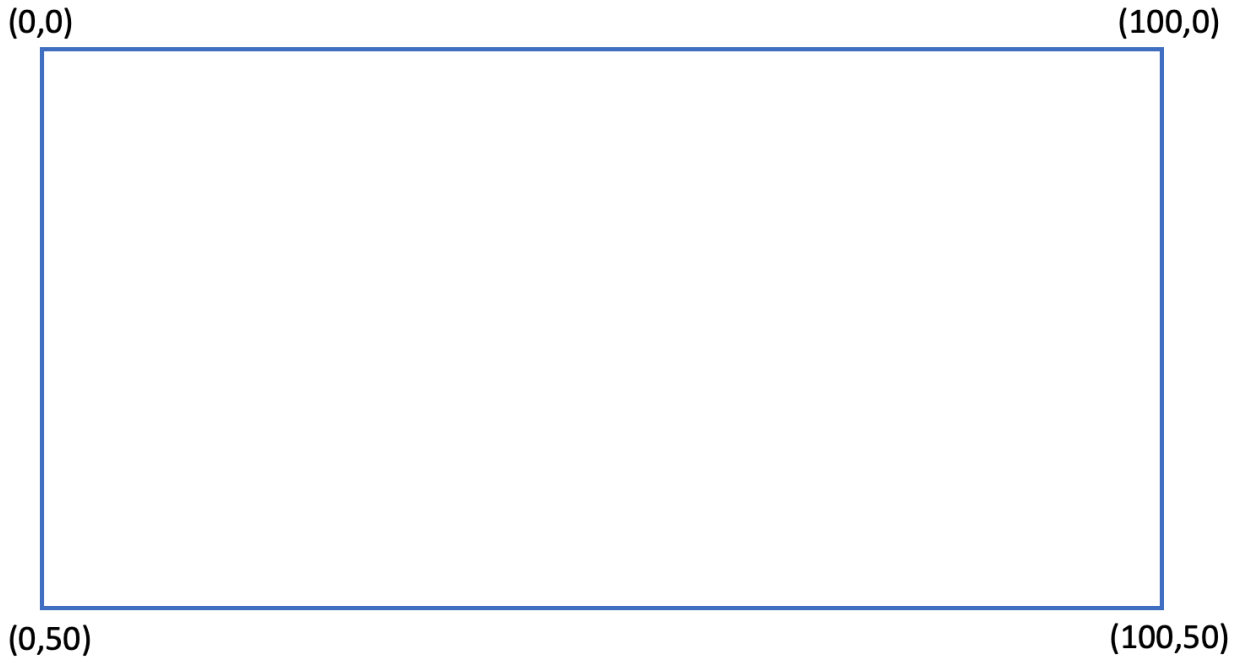
JFrame represents the form of the desktop application, and the components in the form are called JComponent.



2. x and y coordinate

x and y coordinates

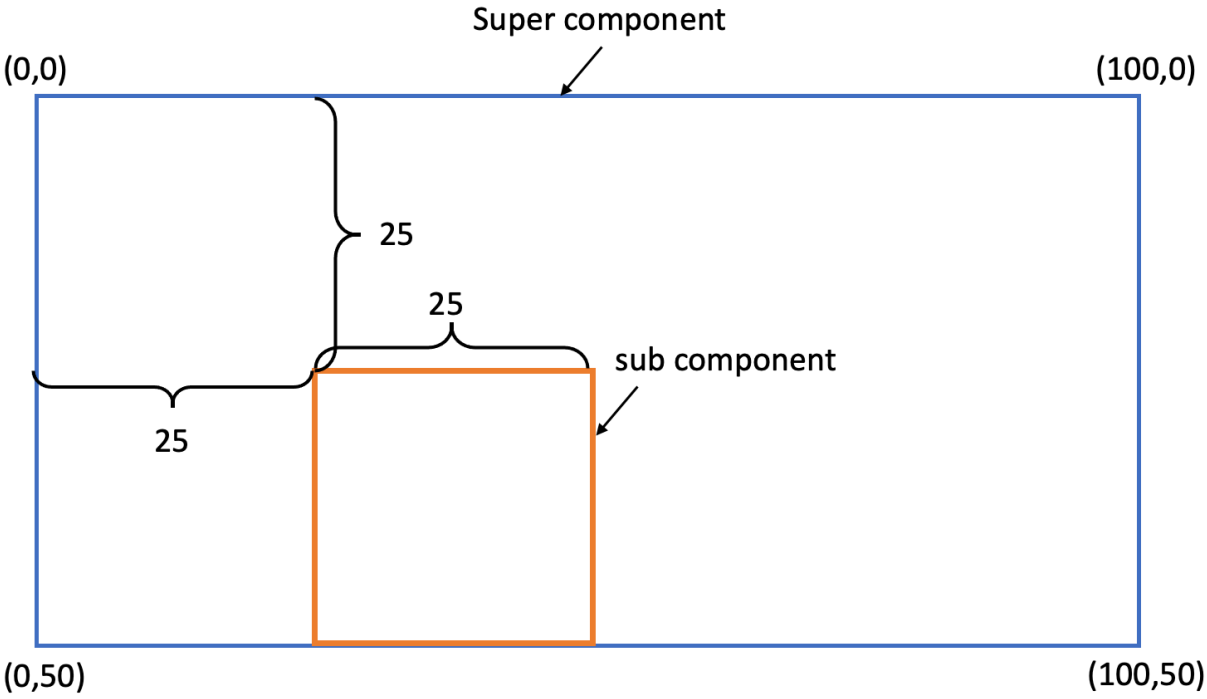
In Swing, the origin of x and y coordinates is in the upper left corner. For example, if the width and height are 100,50 respectively, the x and y coordinate of the four corner should be



location

In Swing, the location of current component represents the its upper left corner on the x and y coordinate of its super component.

For example:



The location of sub component in super component is: (25,25)

If we want the components to be displayed on it superComponent normally, we need to add the components to the super component and place them in a suitable location.

For example:

```
superComponent.setSize(50,100);
subComponet.setSize(25,25);
subComponet.setLocation(25,25);
superComponent.add(subComponent);
```

Example Code:

```
import javax.swing.*;
import java.awt.*;

public class DisplayJpg {

    public static void main(String[] args) {
        JFrame frame = new JFrame();  //create a JFrame
        //A way to load image
        Image image =
Toolkit.getDefaultToolkit().getImage("picture.jpg").getScaledInstance(100, 100,
Image.SCALE_FAST);
```

```
        JComponent imageComponent = new ImageComponent(image);// create an
instance of ImageComponent
        imageComponent.setSize(100, 100);// set the size of image component
        System.out.printf("imageComponent [%d,%d]\n",
imageComponent.getWidth(), imageComponent.getHeight());

        frame.setLocationRelativeTo(null);
        imageComponent.setLocation(50, 50); // set absolute location
        frame.setLayout(null);   //set layout with null to suit absolute
location
        frame.setVisible(true);  //Set the window to visible
        frame.add(imageComponent);// add Jcomponent into Jframe
        //set the size of the window
        frame.setSize(imageComponent.getWidth() + 100, 200);
        System.out.printf("window Frame [%d,%d]\n", frame.getWidth(),
frame.getHeight());
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);  //let the window
can be close by click "x"
    }

// design a user defined component, which can draw an image
    static class ImageComponent extends JComponent {
        Image paintImage;

        public ImageComponent(Image image) {
            this.setLayout(null);
            this.setFocusable(true);//Sets the focusable state of this
Component to the specified value. This value overrides the Component's default
focusability.
            this.paintImage = image;
            repaint();//execute the paintComponent method
        }

        // the method describes how to paint, and being invoked by repaint()
method.
        @Override
        protected void paintComponent(Graphics g) {
            super.paintComponent(g);
            g.drawImage(paintImage, 0, 0, paintImage.getWidth(this),
paintImage.getHeight(this), this);
        }
    }

}
```