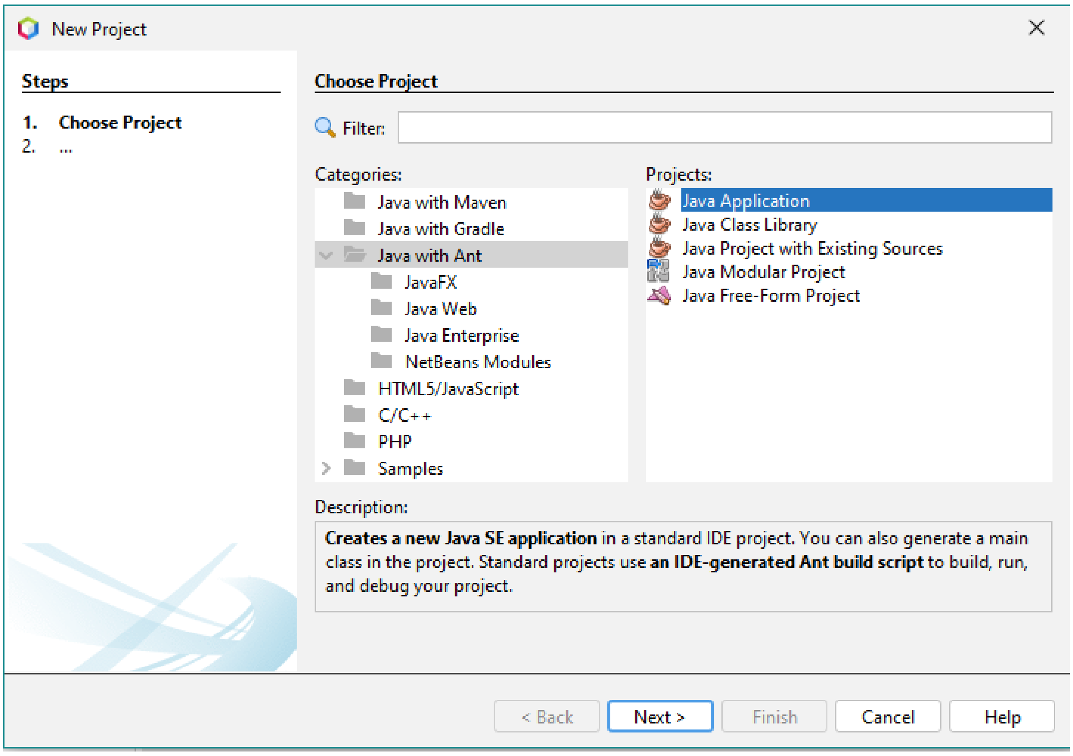
Project Documentation

Chatting Application in Java

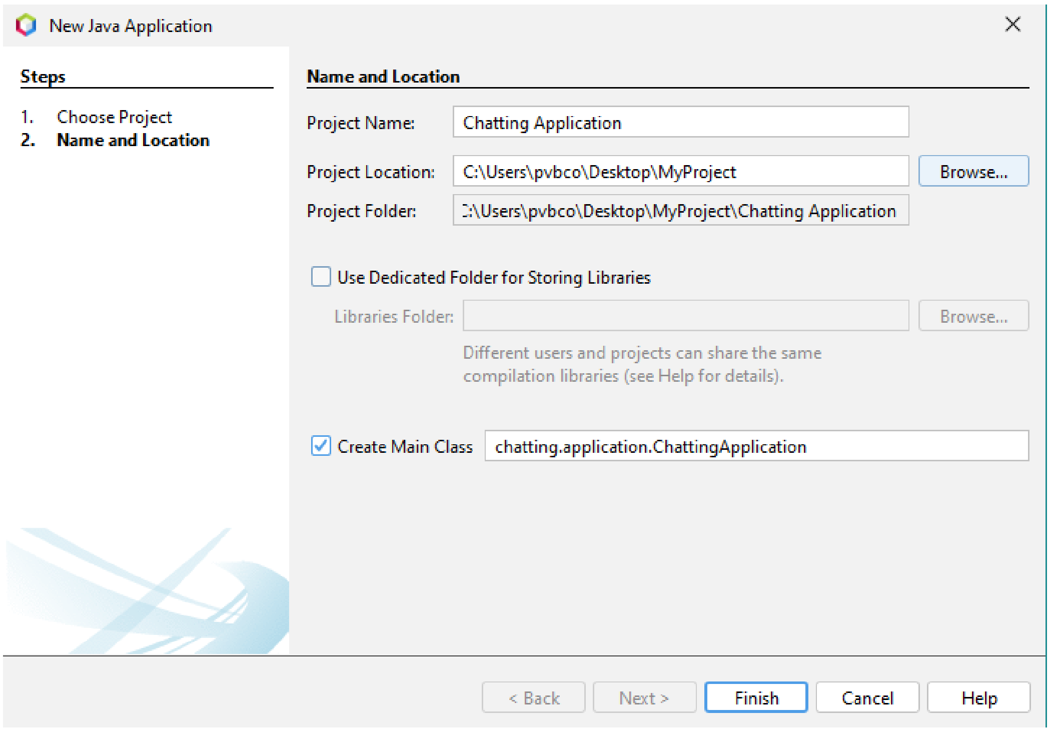
By Parth Bhalerao

Step-1

Create a new Java Project in Apache-Netbeans-12.0



Description: **Creates a new Java SE application** in a standard IDE project. You can also generate a main class in the project. Standard projects use **an IDE-generated Ant build script** to build, run, and debug your project.



Project Name: Chatting Application

Project Location: C:\Users\pvbco\Desktop\MyProject

Project Folder: :\Users\pvbco\Desktop\MyProject\Chatting Application

□ Use Dedicated Folder for Storing Libraries

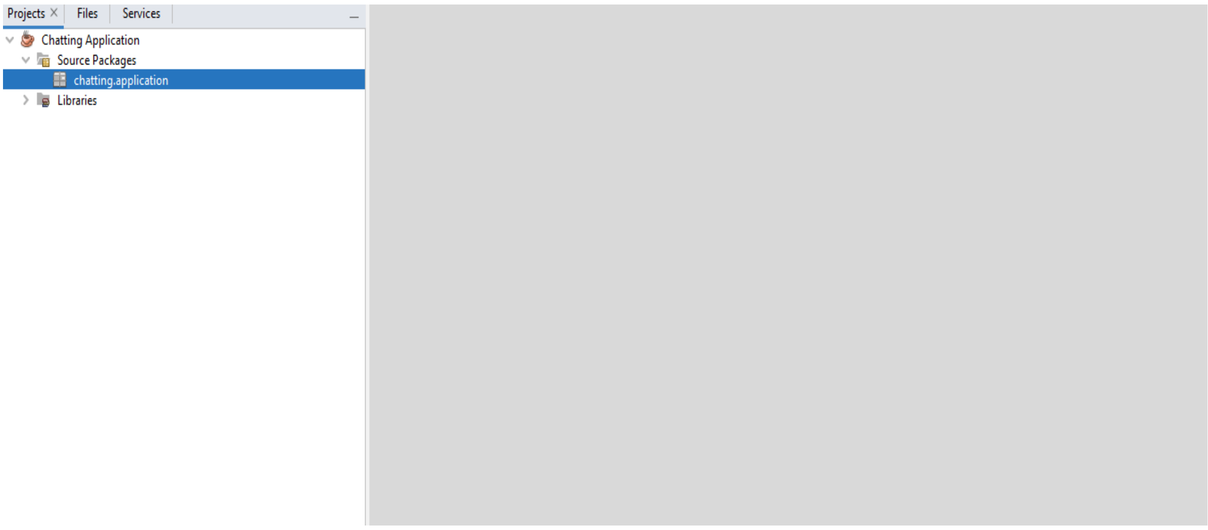
Libraries Folder:

Different users and projects can share the same compilation libraries (see Help for details).

☑ Create Main Class chatting.application.ChattingApplication

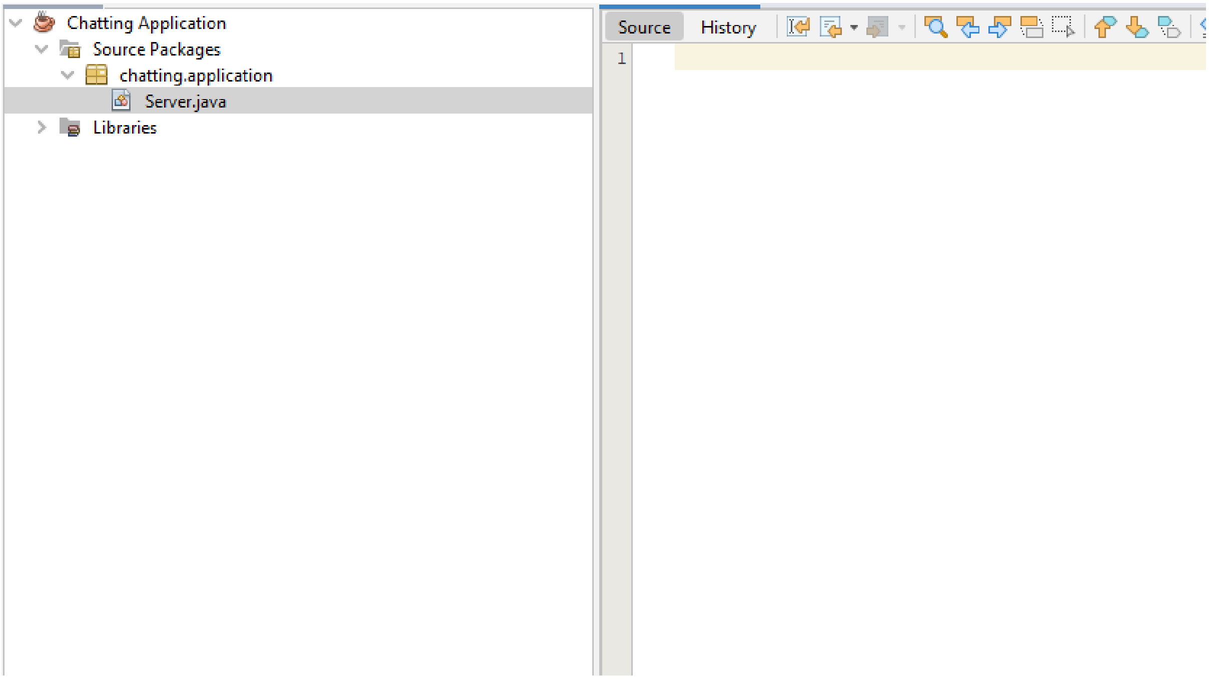
Step-2

Clean the code and start fresh and empty.



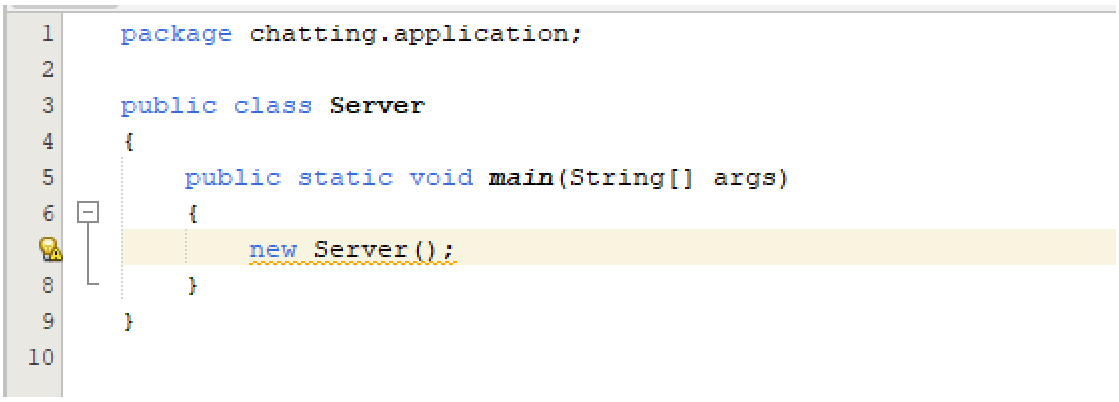
Step-3

Create File Server.java and clean code.



Step-4

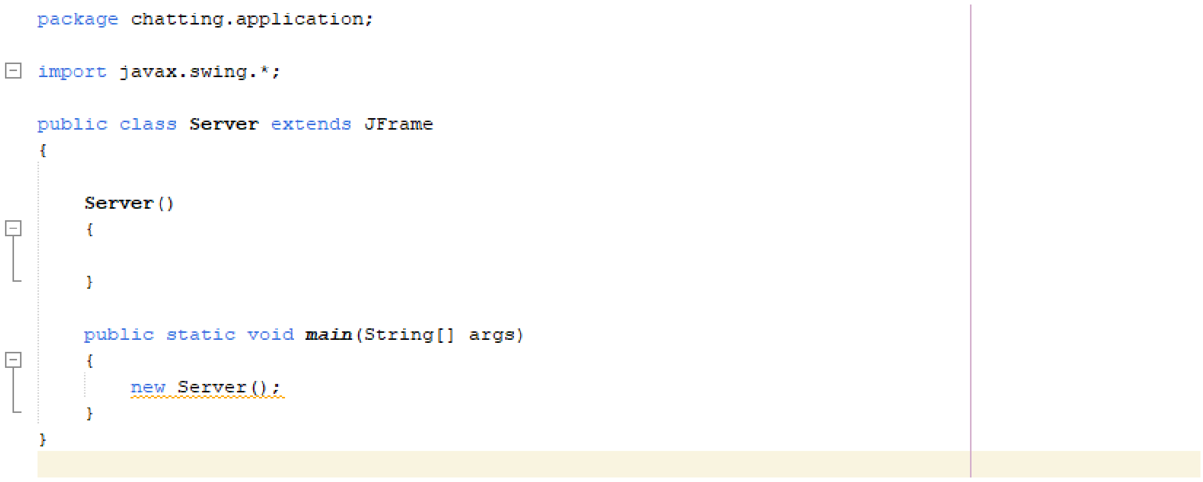
Create Public class and write PSVM and create anonymous object.



```
package chatting.application;

public class Server
{
    public static void main(String[] args)
    {
        new Server();
    }
}

```

Step-5

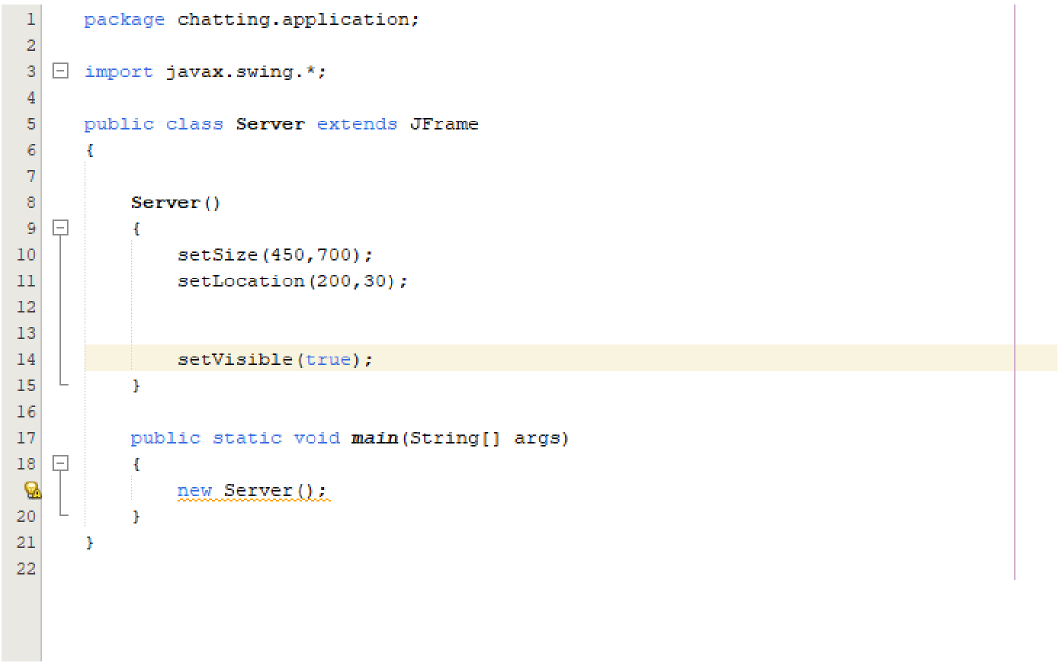
Create Constructor, Inherit JFrame and import swing library.



```
package chatting.application;

import javax.swing.*;

public class Server extends JFrame
{

    Server()
    {

    }

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

Step-6

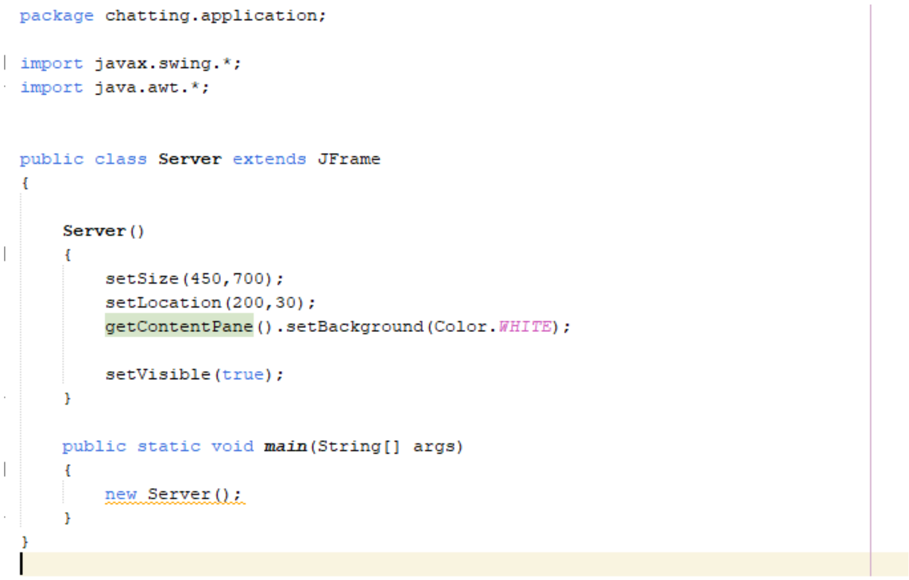
Display Frame, Set Location & Set Visible.



```java
package chatting.application;

import javax.swing.*;

public class Server extends JFrame
{

    Server()
    {
        setSize(450,700);
        setLocation(200,30);


        setVisible(true);
    }

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

```

Step-7

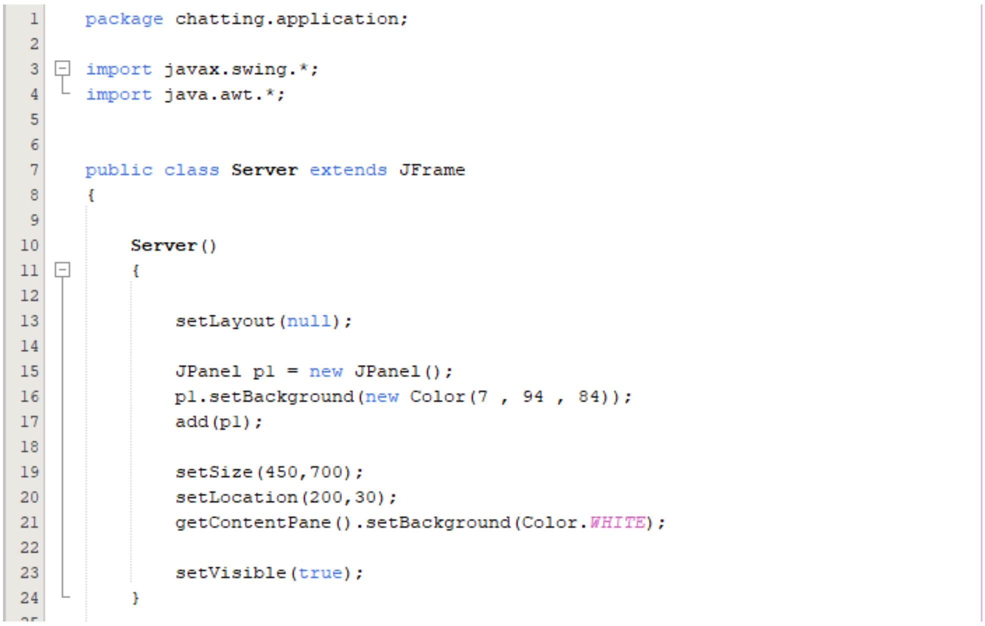
Set Background Color to White and Import awt library.



```java
package chatting.application;

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


public class Server extends JFrame
{

    Server()
    {
        setSize(450,700);
        setLocation(200,30);
        getContentPane().setBackground(Color.WHITE);

        setVisible(true);
    }

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

Step-8

Set Layout to null and create panel for Green Band.

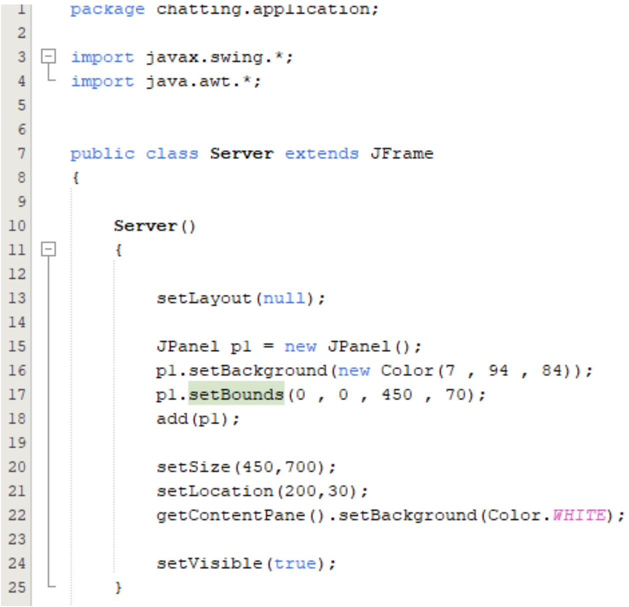


```
package chatting.application;

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


public class Server extends JFrame
{

    Server()
    {

        setLayout(null);

        JPanel p1 = new JPanel();
        p1.setBackground(new Color(7 , 94 , 84));
        add(p1);

        setSize(450,700);
        setLocation(200,30);
        getContentPane().setBackground(Color.WHITE);

        setVisible(true);
    }
```

Step-9

Set Co-ordinates and Location for the Panel p1.

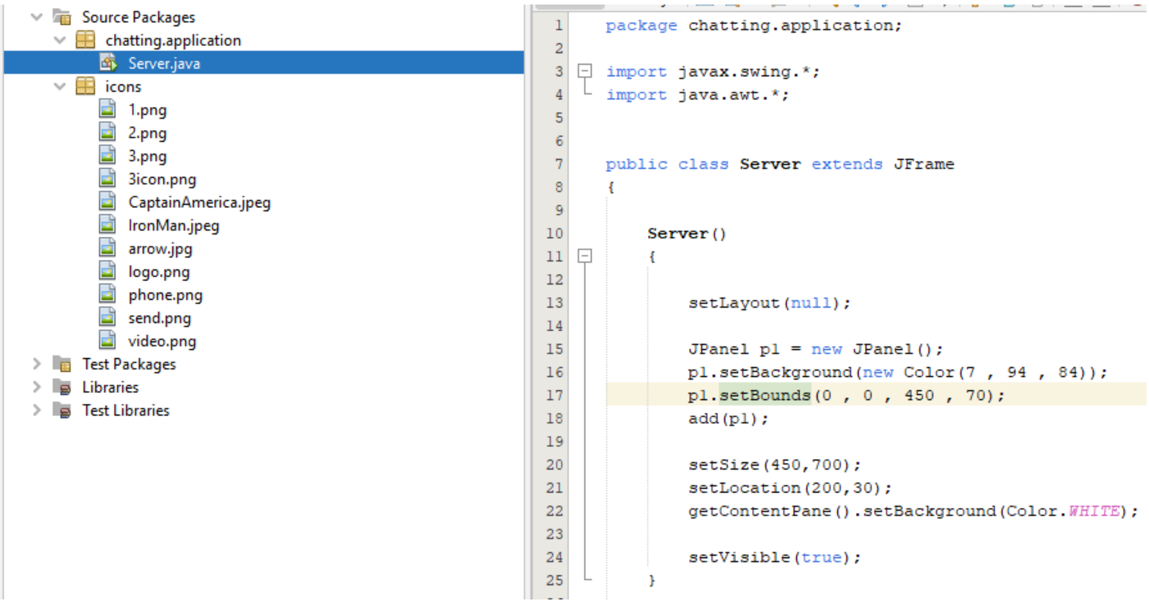
(Green Band)



```
package chatting.application;

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


public class Server extends JFrame
{

    Server()
    {

        setLayout(null);

        JPanel p1 = new JPanel();
        p1.setBackground(new Color(7 , 94 , 84));
        p1.setBounds(0 , 0 , 450 , 70);
        add(p1);

        setSize(450,700);
        setLocation(200,30);
        getContentPane().setBackground(Color.WHITE);

        setVisible(true);
    }
```

Step-10

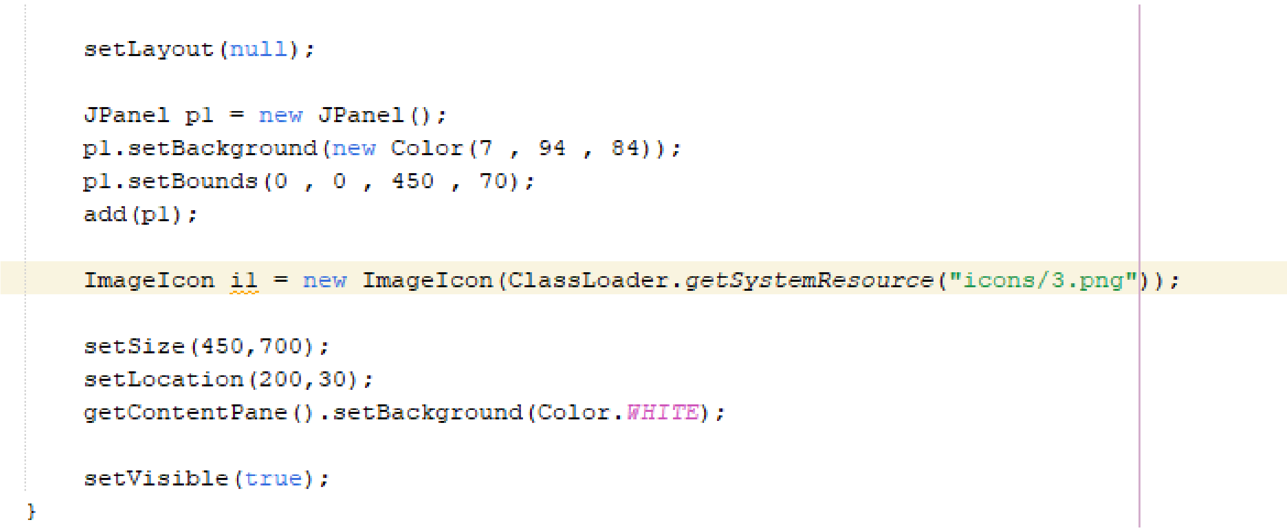
Copy Icons folder into proper path to use images later.



```
package chatting.application;

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


public class Server extends JFrame
{

    Server()
    {

        setLayout(null);

        JPanel p1 = new JPanel();
        p1.setBackground(new Color(7 , 94 , 84));
        p1.setBounds(0 , 0 , 450 , 70);
        add(p1);

        setSize(450,700);
        setLocation(200,30);
        getContentPane().setBackground(Color.WHITE);

        setVisible(true);
    }
```

Step-11

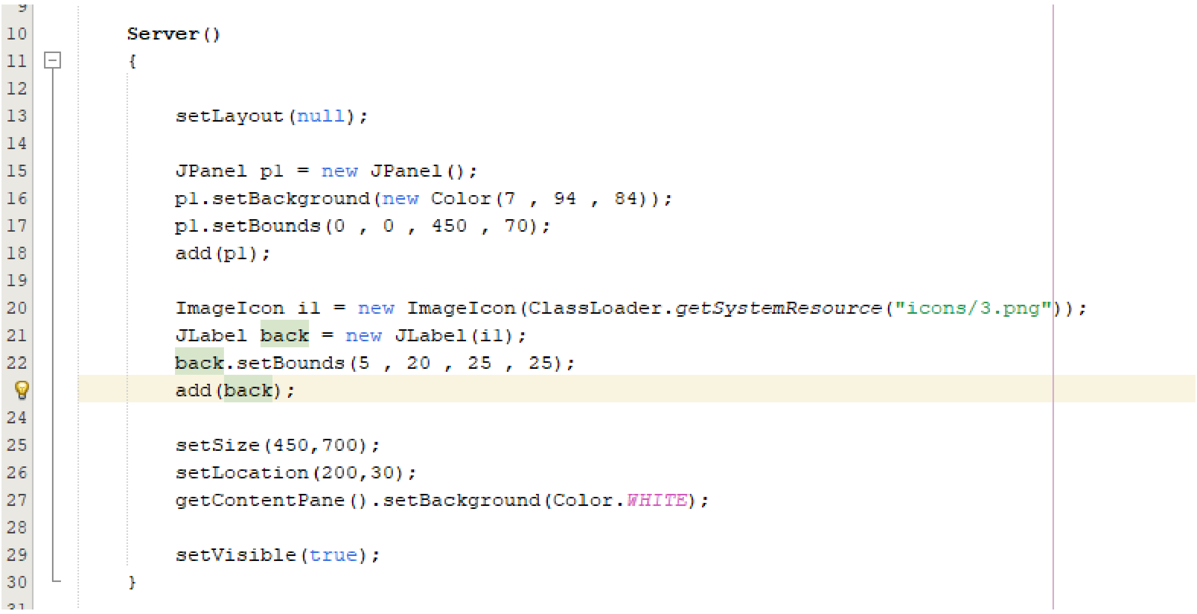
Create Image Icon Object for i1.



```
        setLayout(null);

        JPanel p1 = new JPanel();
        p1.setBackground(new Color(7 , 94 , 84));
        p1.setBounds(0 , 0 , 450 , 70);
        add(p1);

        ImageIcon i1 = new ImageIcon(ClassLoader.getSystemResource("icons/3.png"));

        setSize(450,700);
        setLocation(200,30);
        getContentPane().setBackground(Color.WHITE);

        setVisible(true);
    }
```

Step-12

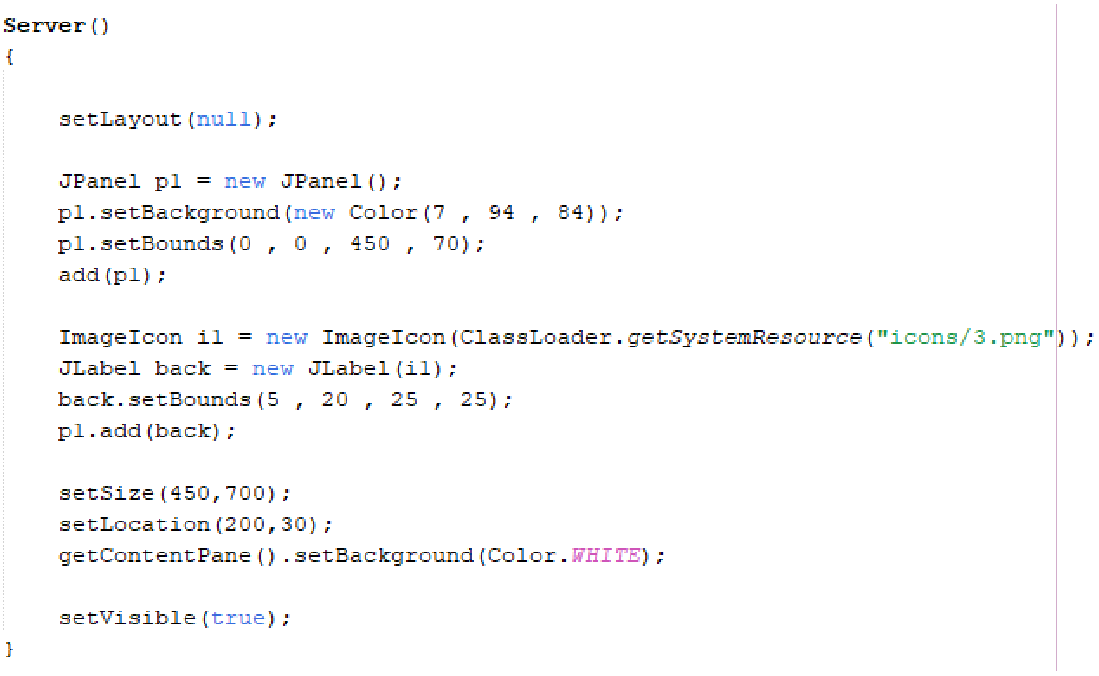
Create JLabel & SetBounds with add-back.



```
Server()
{

    setLayout(null);

    JPanel p1 = new JPanel();
    p1.setBackground(new Color(7 , 94 , 84));
    p1.setBounds(0 , 0 , 450 , 70);
    add(p1);

    ImageIcon i1 = new ImageIcon(ClassLoader.getSystemResource("icons/3.png"));
    JLabel back = new JLabel(i1);
    back.setBounds(5 , 20 , 25 , 25);
    add(back);

    setSize(450,700);
    setLocation(200,30);
    getContentPane().setBackground(Color.WHITE);

    setVisible(true);
}
```

Step-13

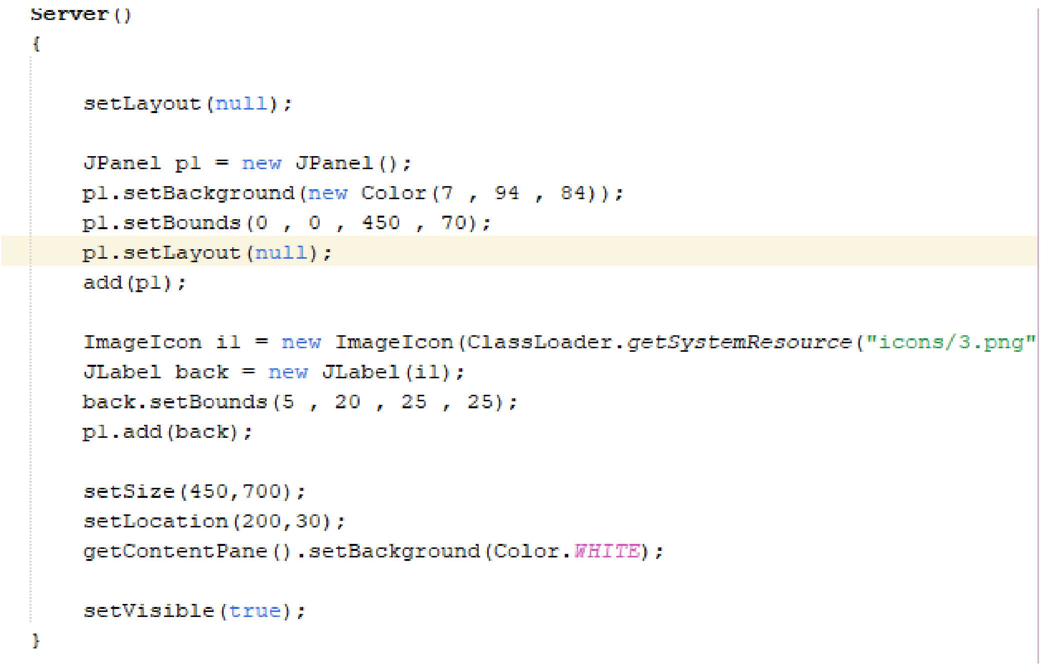
Add JLabel back on panel p1.



```
Server()
{

    setLayout(null);

    JPanel p1 = new JPanel();
    p1.setBackground(new Color(7 , 94 , 84));
    p1.setBounds(0 , 0 , 450 , 70);
    add(p1);

    ImageIcon i1 = new ImageIcon(ClassLoader.getSystemResource("icons/3.png"));
    JLabel back = new JLabel(i1);
    back.setBounds(5 , 20 , 25 , 25);
    p1.add(back);

    setSize(450,700);
    setLocation(200,30);
    getContentPane().setBackground(Color.WHITE);

    setVisible(true);
}
```

Step-14

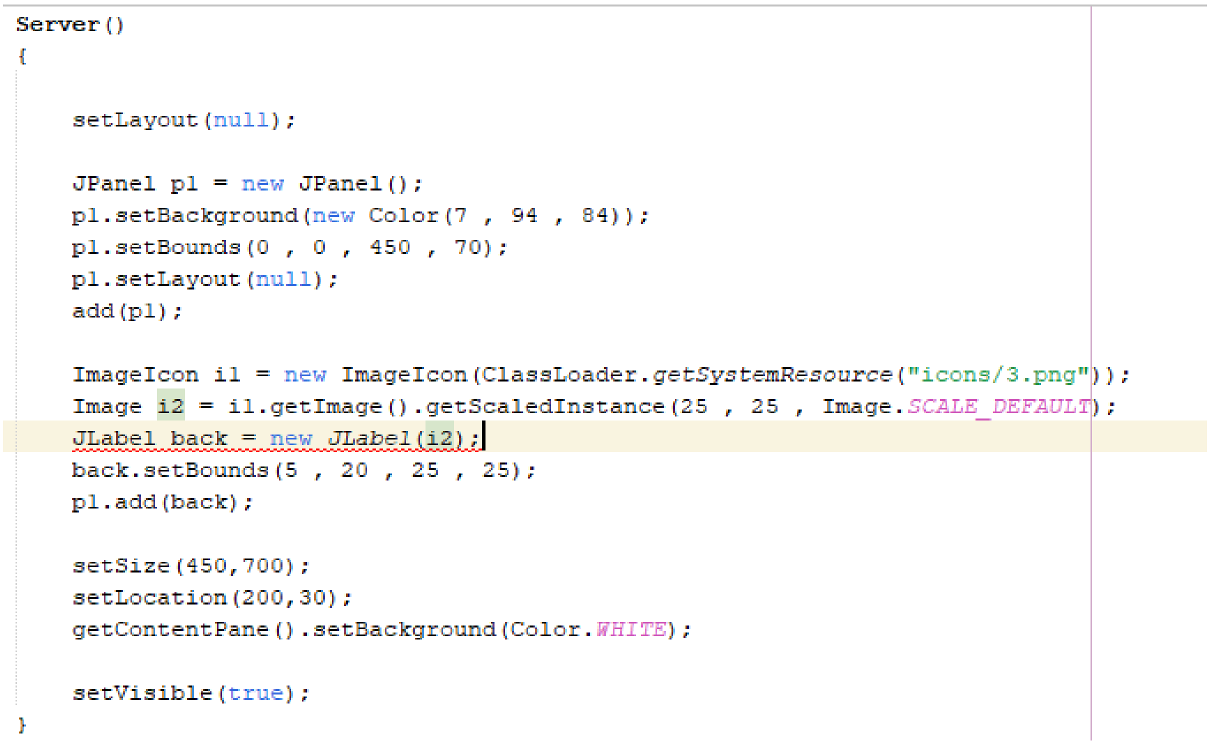
Set Layout null for panel p1.



```
Server()
{

    setLayout(null);

    JPanel p1 = new JPanel();
    p1.setBackground(new Color(7 , 94 , 84));
    p1.setBounds(0 , 0 , 450 , 70);
    p1.setLayout(null);
    add(p1);

    ImageIcon i1 = new ImageIcon(ClassLoader.getSystemResource("icons/3.png"
    JLabel back = new JLabel(i1);
    back.setBounds(5 , 20 , 25 , 25);
    p1.add(back);

    setSize(450,700);
    setLocation(200,30);
    getContentPane().setBackground(Color.WHITE);

    setVisible(true);
}
```

Step-15

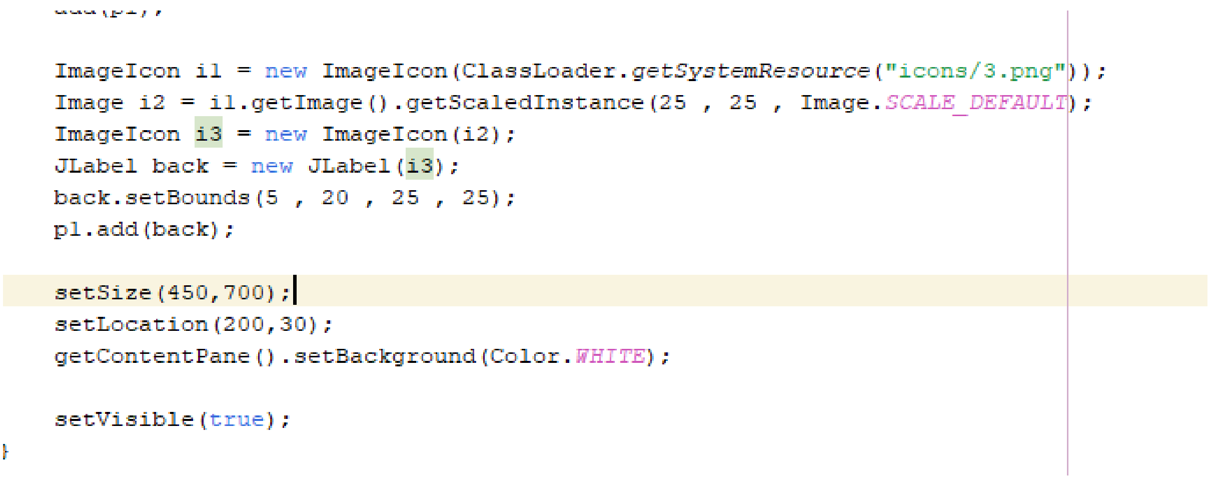
Scale Image to fix it.



```
Server()
{

    setLayout(null);

    JPanel p1 = new JPanel();
    p1.setBackground(new Color(7 , 94 , 84));
    p1.setBounds(0 , 0 , 450 , 70);
    p1.setLayout(null);
    add(p1);

    ImageIcon i1 = new ImageIcon(ClassLoader.getSystemResource("icons/3.png"));
    Image i2 = i1.getImage().getScaledInstance(25 , 25 , Image.SCALE_DEFAULT);
    JLabel back = new JLabel(i2);
    back.setBounds(5 , 20 , 25 , 25);
    p1.add(back);

    setSize(450,700);
    setLocation(200,30);
    getContentPane().setBackground(Color.WHITE);

    setVisible(true);
}
```

Step-16

Convert Scaled Image to Image Icon to solve error.

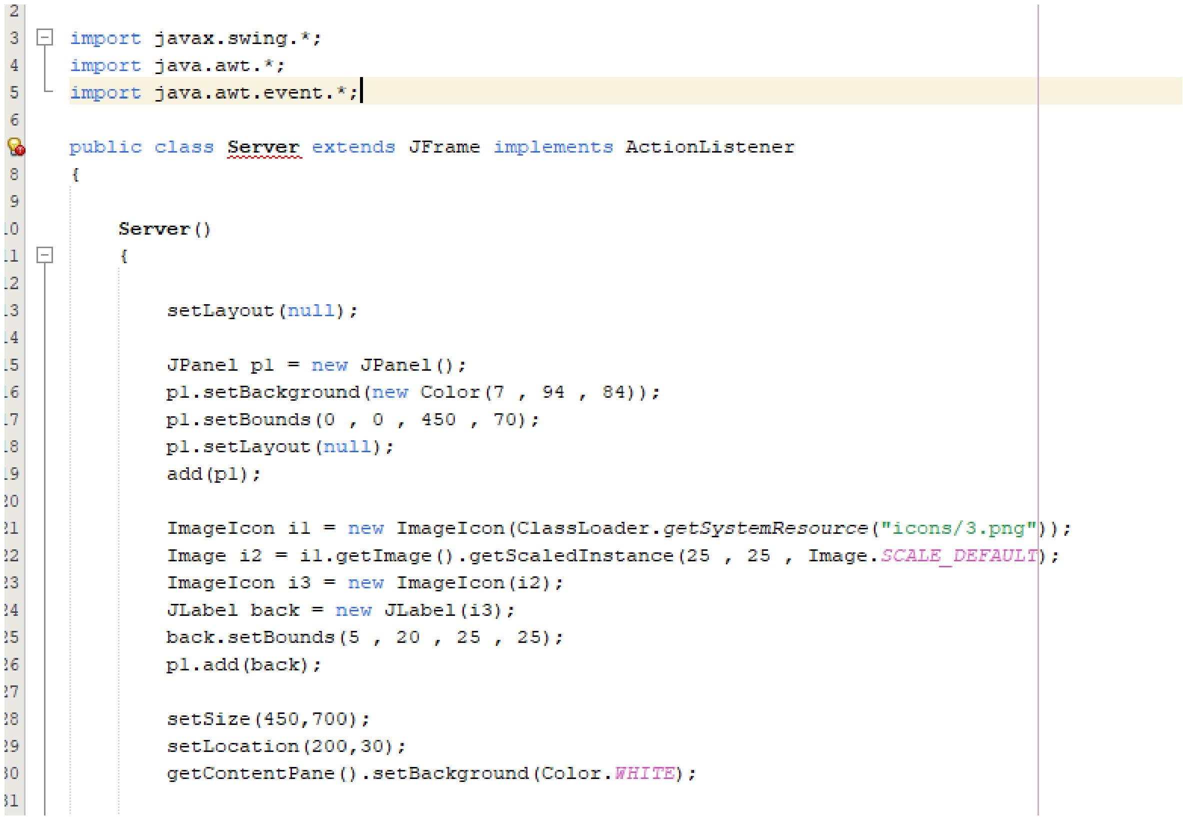


```
    add(p1);

    ImageIcon i1 = new ImageIcon(ClassLoader.getSystemResource("icons/3.png"));
    Image i2 = i1.getImage().getScaledInstance(25 , 25 , Image.SCALE_DEFAULT);
    ImageIcon i3 = new ImageIcon(i2);
    JLabel back = new JLabel(i3);
    back.setBounds(5 , 20 , 25 , 25);
    p1.add(back);

    setSize(450,700);
    setLocation(200,30);
    getContentPane().setBackground(Color.WHITE);

    setVisible(true);
}
```

Step-17

Now we require click action on back arrow.

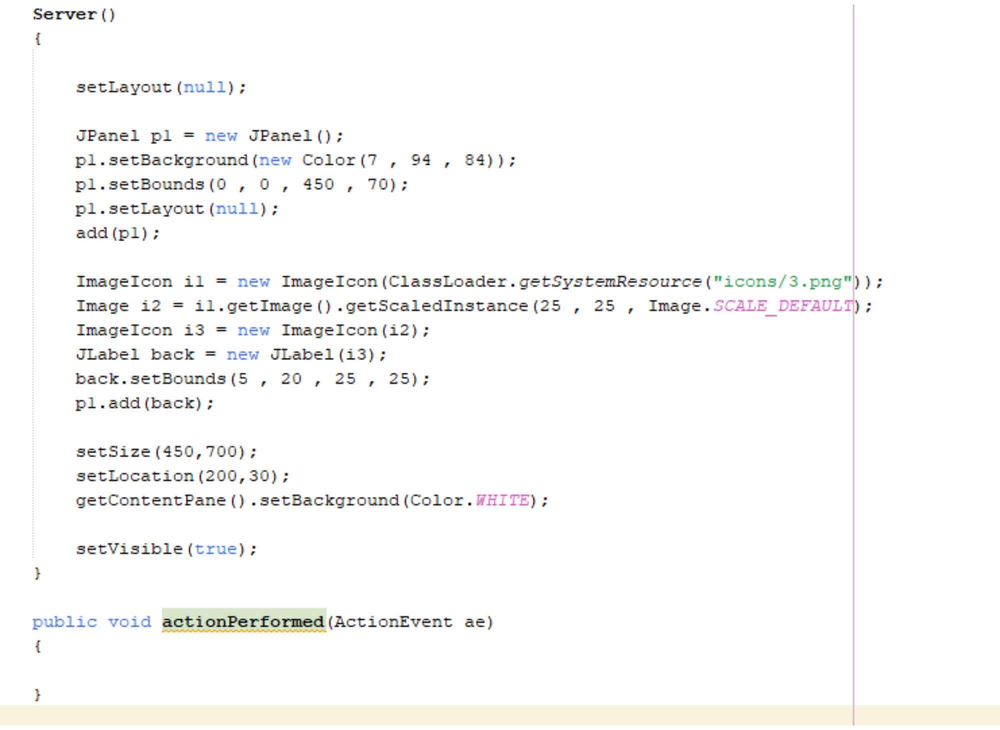
So first implement Action Listener and import library.



```

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

public class Server extends JFrame implements ActionListener
{

    Server()
    {

        setLayout(null);

        JPanel p1 = new JPanel();
        p1.setBackground(new Color(7 , 94 , 84));
        p1.setBounds(0 , 0 , 450 , 70);
        p1.setLayout(null);
        add(p1);

        ImageIcon i1 = new ImageIcon(ClassLoader.getSystemResource("icons/3.png"));
        Image i2 = i1.getImage().getScaledInstance(25 , 25 , Image.SCALE_DEFAULT);
        ImageIcon i3 = new ImageIcon(i2);
        JLabel back = new JLabel(i3);
        back.setBounds(5 , 20 , 25 , 25);
        p1.add(back);

        setSize(450,700);
        setLocation(200,30);
        getContentPane().setBackground(Color.WHITE);

```

Step-18

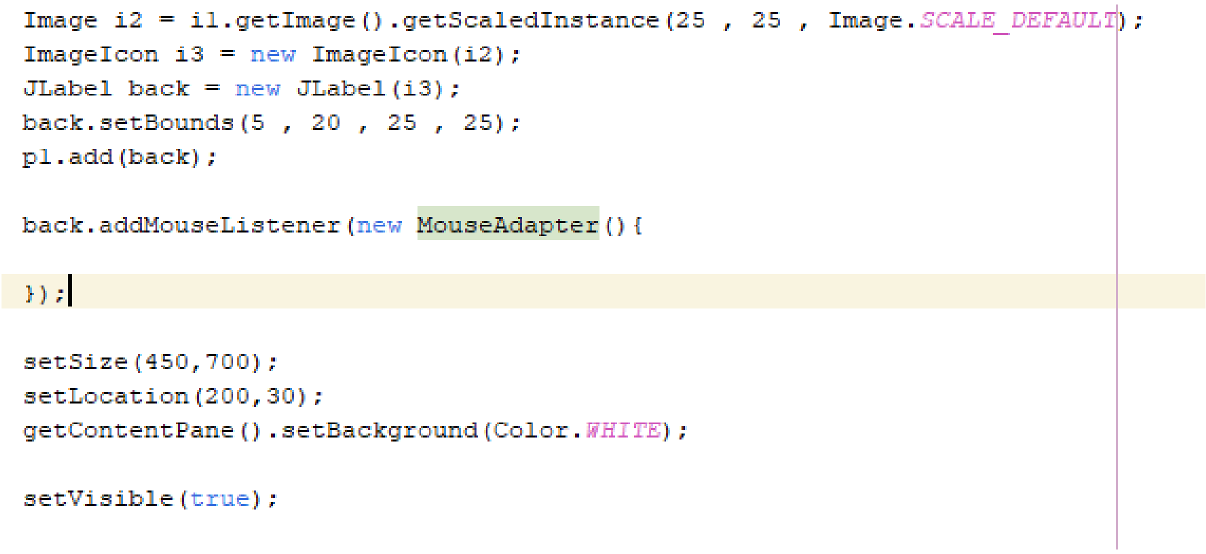
Override Abstract Method of ActionListener to solve error.



```
Server()
{

    setLayout(null);

    JPanel p1 = new JPanel();
    p1.setBackground(new Color(7 , 94 , 84));
    p1.setBounds(0 , 0 , 450 , 70);
    p1.setLayout(null);
    add(p1);

    ImageIcon i1 = new ImageIcon(ClassLoader.getSystemResource("icons/3.png"));
    Image i2 = i1.getImage().getScaledInstance(25 , 25 , Image.SCALE_DEFAULT);
    ImageIcon i3 = new ImageIcon(i2);
    JLabel back = new JLabel(i3);
    back.setBounds(5 , 20 , 25 , 25);
    p1.add(back);

    setSize(450,700);
    setLocation(200,30);
    getContentPane().setBackground(Color.WHITE);

    setVisible(true);
}

public void actionPerformed(ActionEvent ae)
{

}
```

Step-19

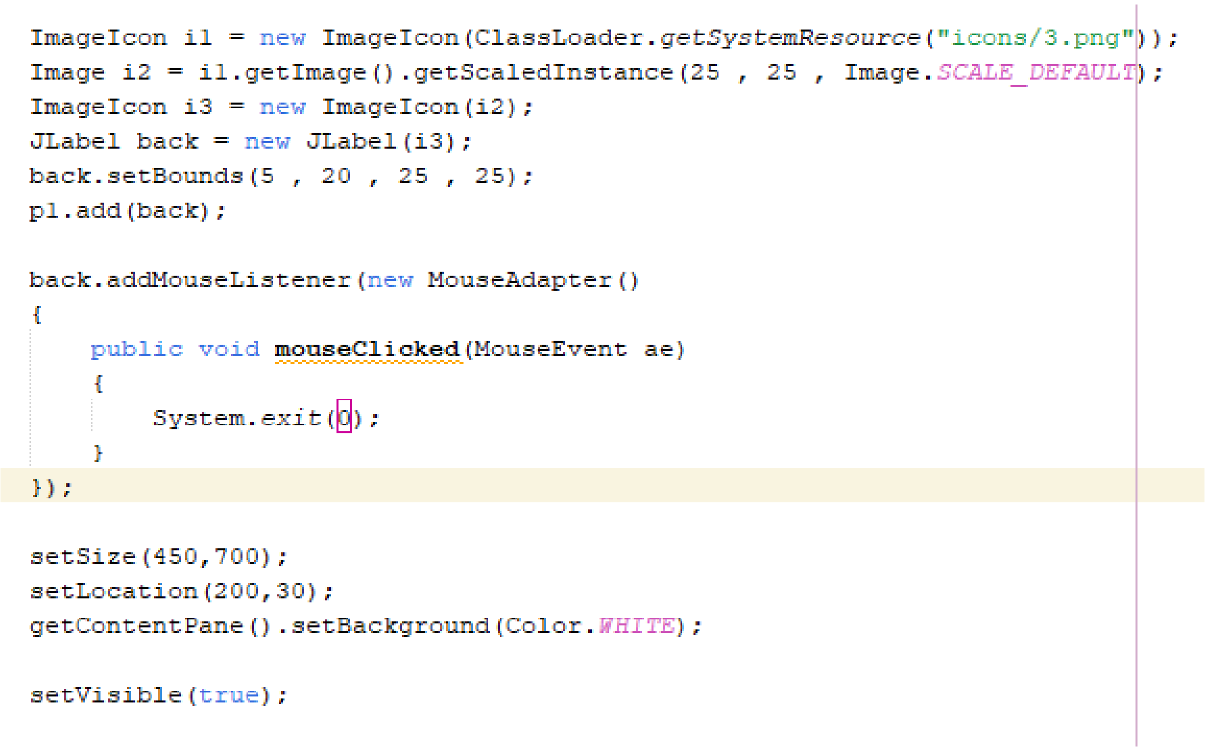
Create Mouse Listener for back button.



```
Image i2 = i1.getImage().getScaledInstance(25 , 25 , Image.SCALE_DEFAULT);
ImageIcon i3 = new ImageIcon(i2);
JLabel back = new JLabel(i3);
back.setBounds(5 , 20 , 25 , 25);
p1.add(back);

back.addMouseListener(new MouseAdapter(){

});

setSize(450,700);
setLocation(200,30);
getContentPane().setBackground(Color.WHITE);

setVisible(true);
```

Step-20

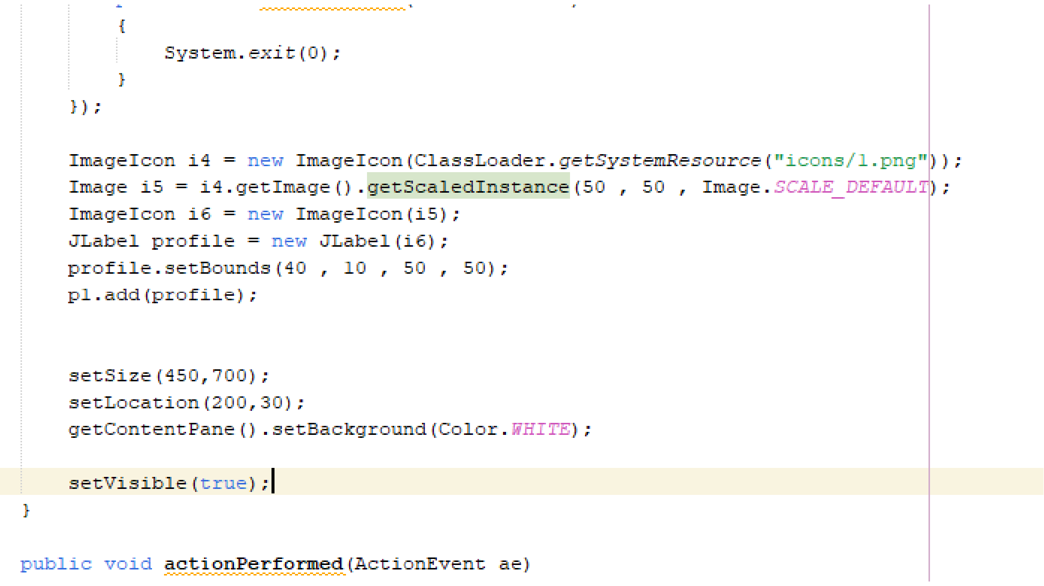
Write Mouse Clicked Method.



```java
ImageIcon i1 = new ImageIcon(ClassLoader.getSystemResource("icons/3.png"));
Image i2 = i1.getImage().getScaledInstance(25 , 25 , Image.SCALE_DEFAULT);
ImageIcon i3 = new ImageIcon(i2);
JLabel back = new JLabel(i3);
back.setBounds(5 , 20 , 25 , 25);
p1.add(back);

back.addMouseListener(new MouseAdapter()
{
    public void mouseClicked(MouseEvent ae)
    {
        System.exit(0);
    }
});

setSize(450,700);
setLocation(200,30);
getContentPane().setBackground(Color.WHITE);

setVisible(true);
```

Step-21

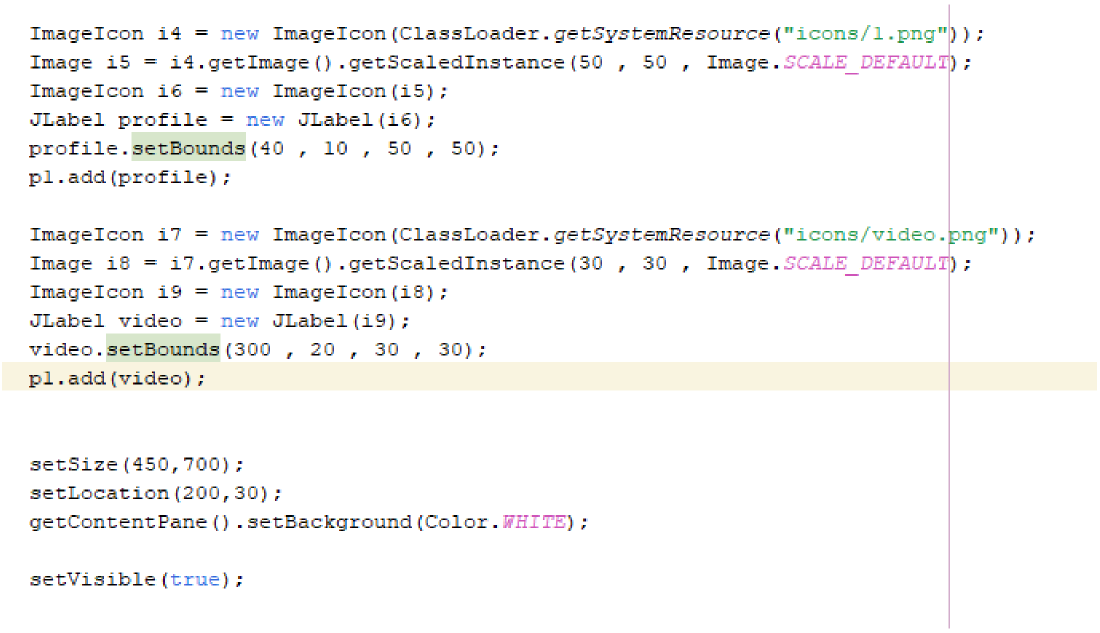
Set Profile Picture of Iron-Man.



```java
            {
                System.exit(0);
            }
        });

        ImageIcon i4 = new ImageIcon(ClassLoader.getSystemResource("icons/1.png"));
        Image i5 = i4.getImage().getScaledInstance(50 , 50 , Image.SCALE_DEFAULT);
        ImageIcon i6 = new ImageIcon(i5);
        JLabel profile = new JLabel(i6);
        profile.setBounds(40 , 10 , 50 , 50);
        p1.add(profile);


        setSize(450,700);
        setLocation(200,30);
        getContentPane().setBackground(Color.WHITE);

        setVisible(true);
    }

    public void actionPerformed(ActionEvent ae)
```

Step-22

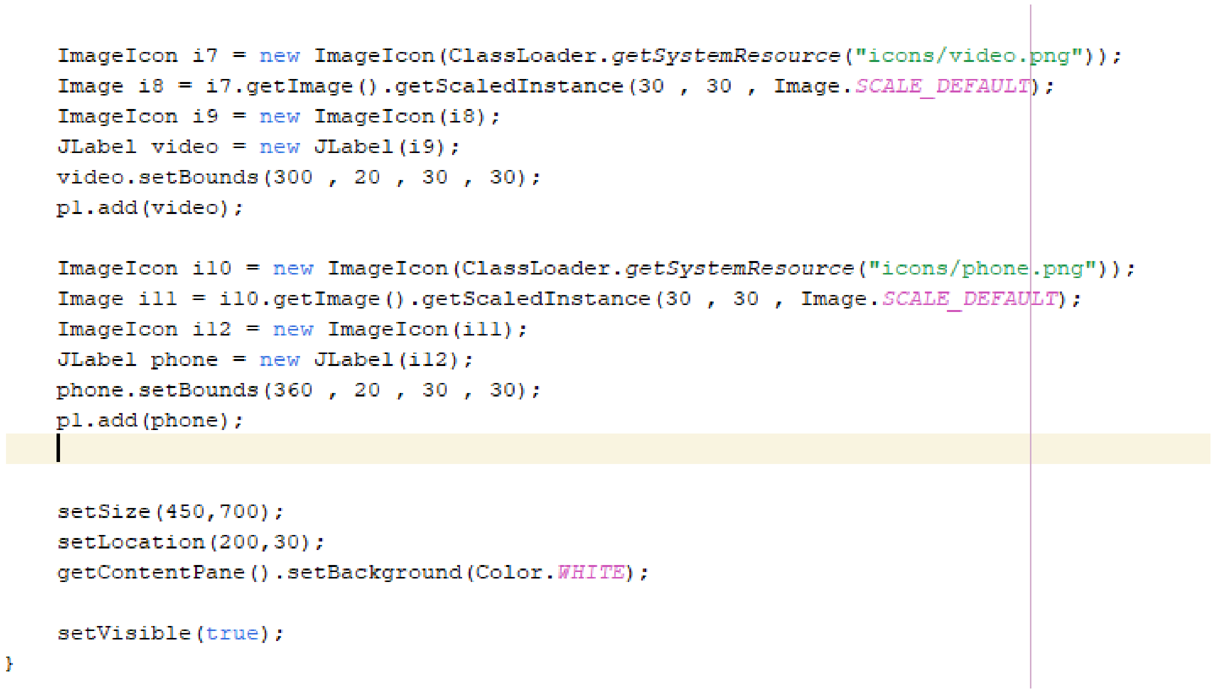
Set Picture for Video Icon.



```
ImageIcon i4 = new ImageIcon(ClassLoader.getSystemResource("icons/1.png"));
Image i5 = i4.getImage().getScaledInstance(50 , 50 , Image.SCALE_DEFAULT);
ImageIcon i6 = new ImageIcon(i5);
JLabel profile = new JLabel(i6);
profile.setBounds(40 , 10 , 50 , 50);
p1.add(profile);

ImageIcon i7 = new ImageIcon(ClassLoader.getSystemResource("icons/video.png"));
Image i8 = i7.getImage().getScaledInstance(30 , 30 , Image.SCALE_DEFAULT);
ImageIcon i9 = new ImageIcon(i8);
JLabel video = new JLabel(i9);
video.setBounds(300 , 20 , 30 , 30);
p1.add(video);


setSize(450,700);
setLocation(200,30);
getContentPane().setBackground(Color.WHITE);

setVisible(true);
```

Step-23

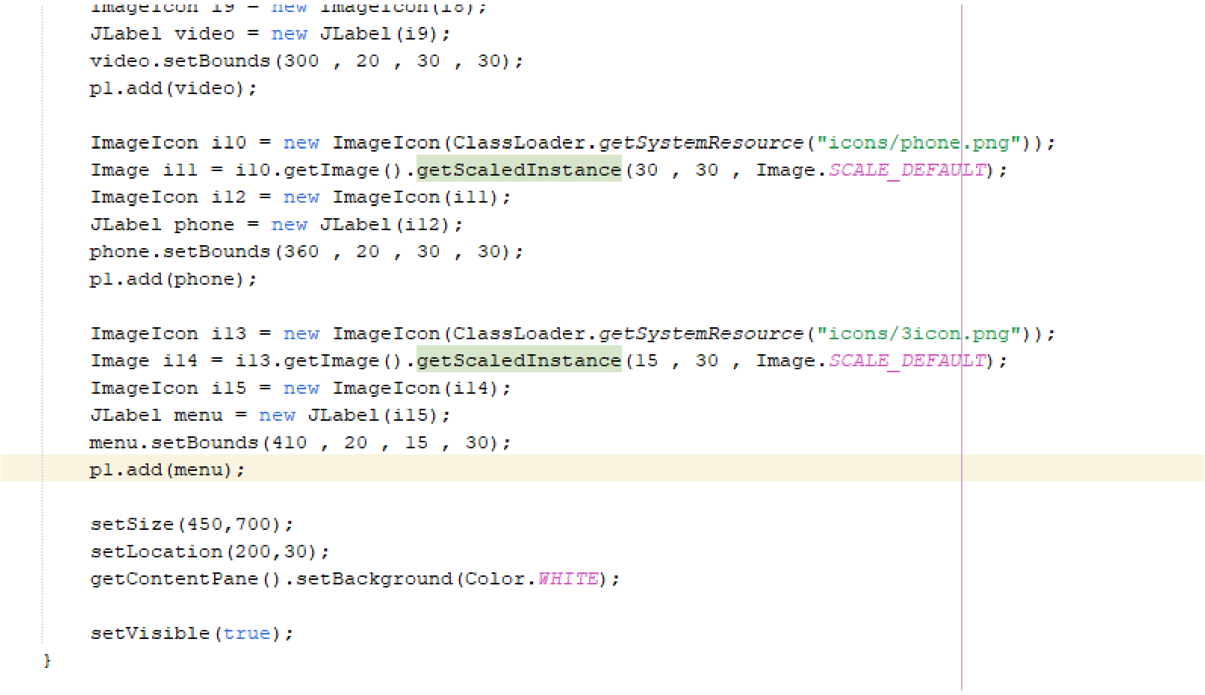
Set Picture for Call Icon.



```
    ImageIcon i7 = new ImageIcon(ClassLoader.getSystemResource("icons/video.png"));
    Image i8 = i7.getImage().getScaledInstance(30 , 30 , Image.SCALE_DEFAULT);
    ImageIcon i9 = new ImageIcon(i8);
    JLabel video = new JLabel(i9);
    video.setBounds(300 , 20 , 30 , 30);
    p1.add(video);

    ImageIcon i10 = new ImageIcon(ClassLoader.getSystemResource("icons/phone.png"));
    Image i11 = i10.getImage().getScaledInstance(30 , 30 , Image.SCALE_DEFAULT);
    ImageIcon i12 = new ImageIcon(i11);
    JLabel phone = new JLabel(i12);
    phone.setBounds(360 , 20 , 30 , 30);
    p1.add(phone);


    setSize(450,700);
    setLocation(200,30);
    getContentPane().setBackground(Color.WHITE);

    setVisible(true);
}
```

Step-24

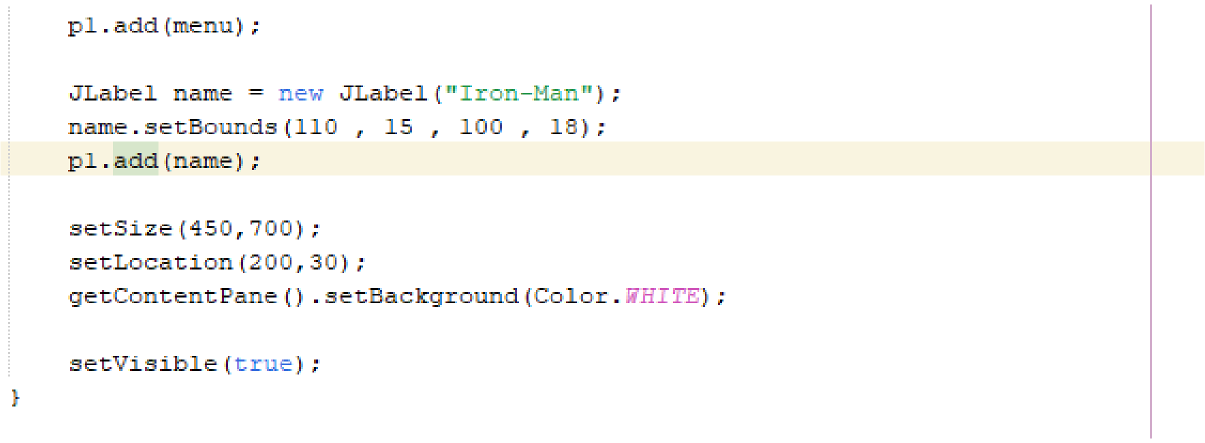
Set Picture for Menu Icon.



```
    ImageIcon i9 = new ImageIcon(i8);
    JLabel video = new JLabel(i9);
    video.setBounds(300 , 20 , 30 , 30);
    p1.add(video);

    ImageIcon i10 = new ImageIcon(ClassLoader.getSystemResource("icons/phone.png"));
    Image i11 = i10.getImage().getScaledInstance(30 , 30 , Image.SCALE_DEFAULT);
    ImageIcon i12 = new ImageIcon(i11);
    JLabel phone = new JLabel(i12);
    phone.setBounds(360 , 20 , 30 , 30);
    p1.add(phone);

    ImageIcon i13 = new ImageIcon(ClassLoader.getSystemResource("icons/3icon.png"));
    Image i14 = i13.getImage().getScaledInstance(15 , 30 , Image.SCALE_DEFAULT);
    ImageIcon i15 = new ImageIcon(i14);
    JLabel menu = new JLabel(i15);
    menu.setBounds(410 , 20 , 15 , 30);
    p1.add(menu);

    setSize(450,700);
    setLocation(200,30);
    getContentPane().setBackground(Color.WHITE);

    setVisible(true);
}
```

Step-25

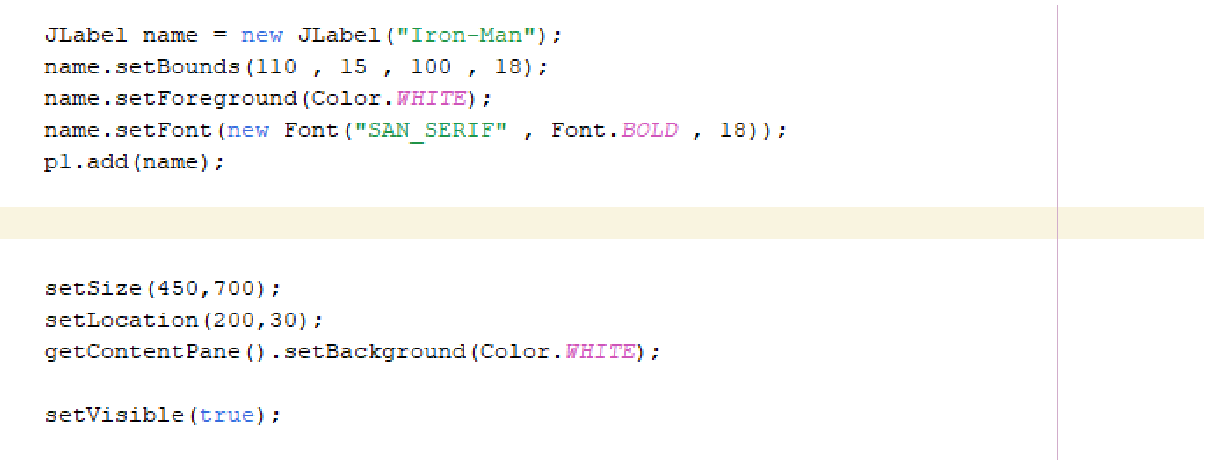
Add name besides Profile Picture.



```
    p1.add(menu);

    JLabel name = new JLabel("Iron-Man");
    name.setBounds(110 , 15 , 100 , 18);
    p1.add(name);

    setSize(450,700);
    setLocation(200,30);
    getContentPane().setBackground(Color.WHITE);

    setVisible(true);
}
```

Step-26

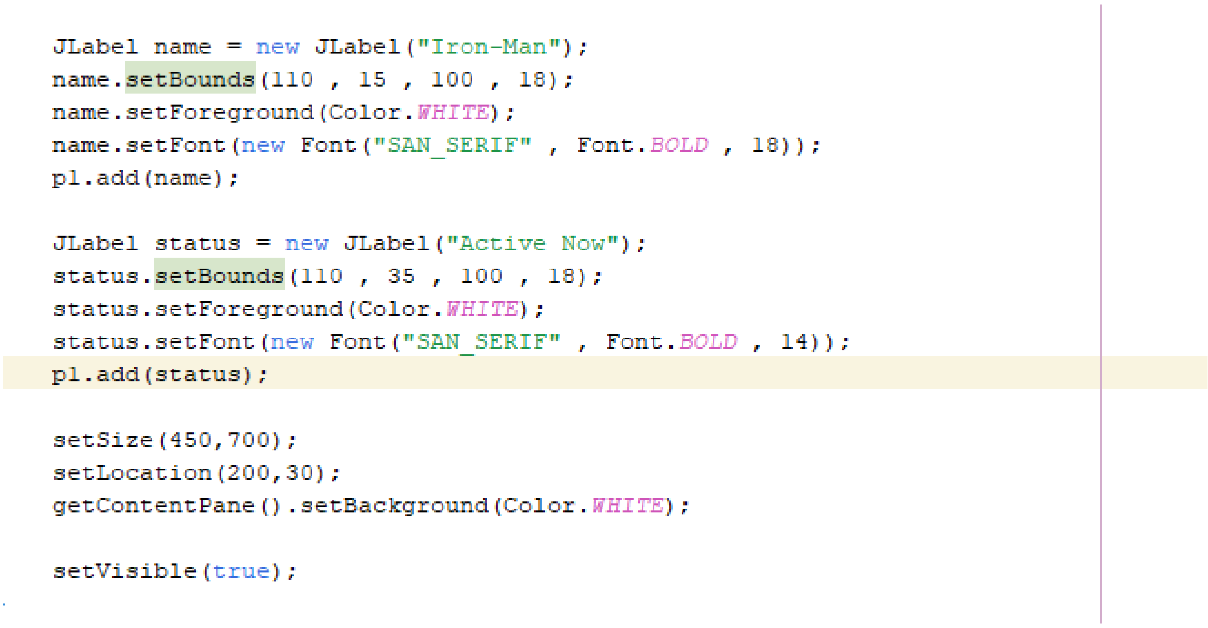
Change Text-Color and Text-Font.



```
JLabel name = new JLabel("Iron-Man");
name.setBounds(110 , 15 , 100 , 18);
name.setForeground(Color.WHITE);
name.setFont(new Font("SAN_SERIF" , Font.BOLD , 18));
p1.add(name);



setSize(450,700);
setLocation(200,30);
getContentPane().setBackground(Color.WHITE);

setVisible(true);
```

Step-27

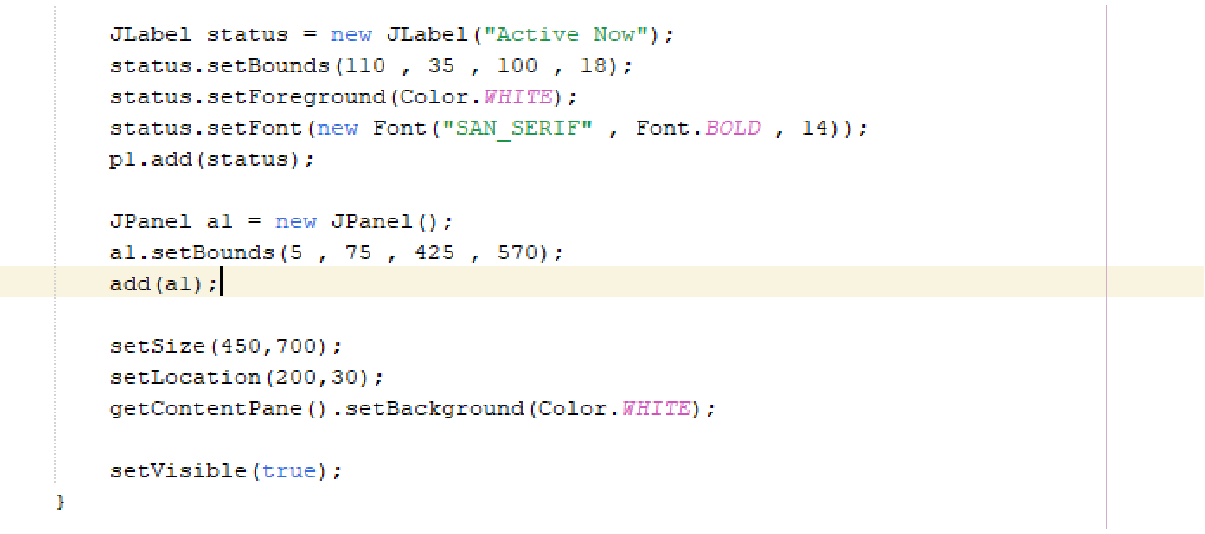
Add Status Text.



```
JLabel name = new JLabel("Iron-Man");
name.setBounds(110 , 15 , 100 , 18);
name.setForeground(Color.WHITE);
name.setFont(new Font("SAN_SERIF" , Font.BOLD , 18));
p1.add(name);

JLabel status = new JLabel("Active Now");
status.setBounds(110 , 35 , 100 , 18);
status.setForeground(Color.WHITE);
status.setFont(new Font("SAN_SERIF" , Font.BOLD , 14));
p1.add(status);

setSize(450,700);
setLocation(200,30);
getContentPane().setBackground(Color.WHITE);

setVisible(true);
```

Step-28

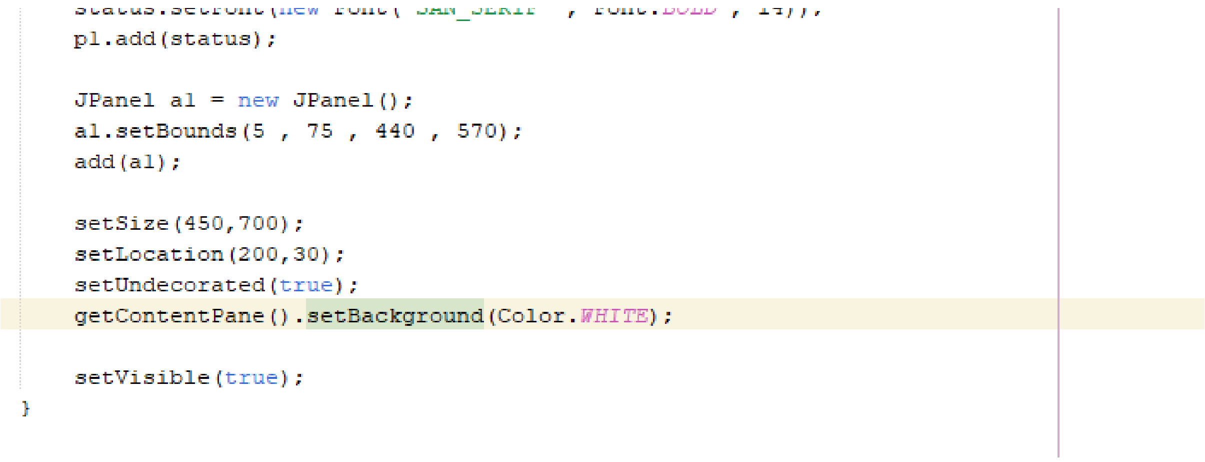
Add Panel for Text Display. (panel a1)



```
    JLabel status = new JLabel("Active Now");
    status.setBounds(110 , 35 , 100 , 18);
    status.setForeground(Color.WHITE);
    status.setFont(new Font("SAN_SERIF" , Font.BOLD , 14));
    p1.add(status);

    JPanel a1 = new JPanel();
    a1.setBounds(5 , 75 , 425 , 570);
    add(a1);

    setSize(450,700);
    setLocation(200,30);
    getContentPane().setBackground(Color.WHITE);

    setVisible(true);
}
```

Step-29

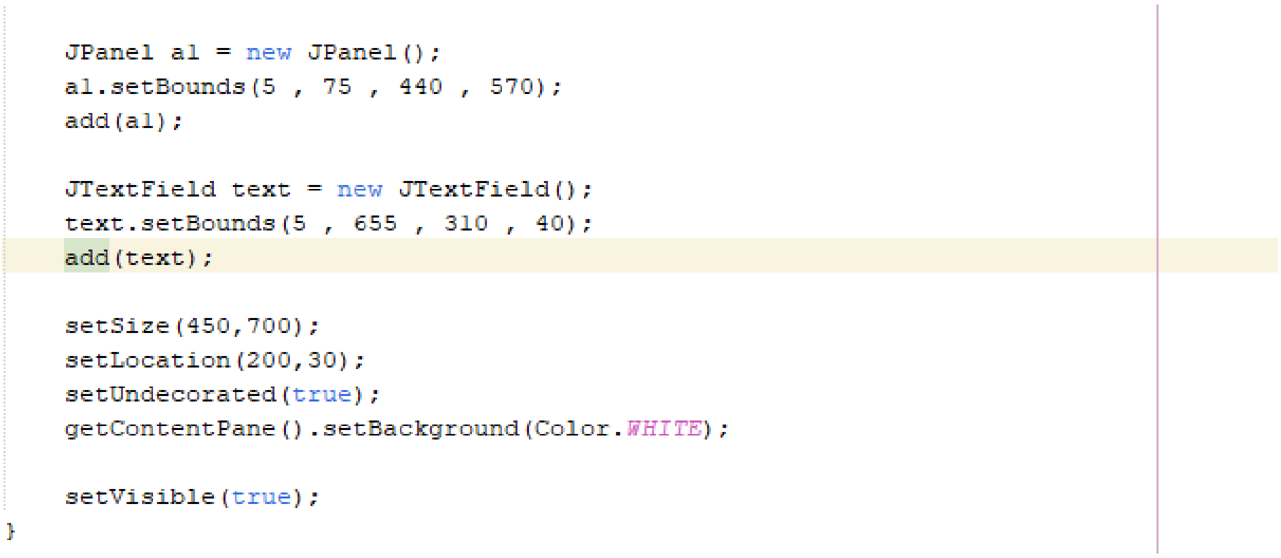
Remove Header Bar.



```
    status.setFont(new Font("SAN_SERIF" , Font.BOLD , 14));
    p1.add(status);

    JPanel a1 = new JPanel();
    a1.setBounds(5 , 75 , 440 , 570);
    add(a1);

    setSize(450,700);
    setLocation(200,30);
    setUndecorated(true);
    getContentPane().setBackground(Color.WHITE);

    setVisible(true);
}
```

Step-30

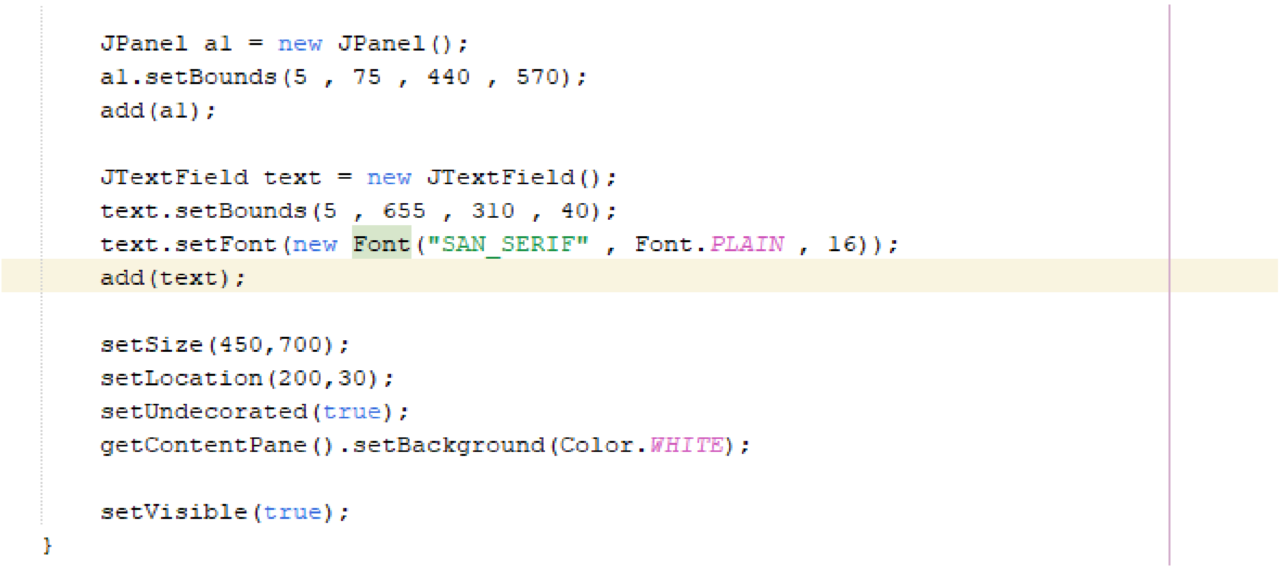
Add Text-Field.



```
    JPanel a1 = new JPanel();
    a1.setBounds(5 , 75 , 440 , 570);
    add(a1);

    JTextField text = new JTextField();
    text.setBounds(5 , 655 , 310 , 40);
    add(text);

    setSize(450,700);
    setLocation(200,30);
    setUndecorated(true);
    getContentPane().setBackground(Color.WHITE);

    setVisible(true);
}
```

Step-31

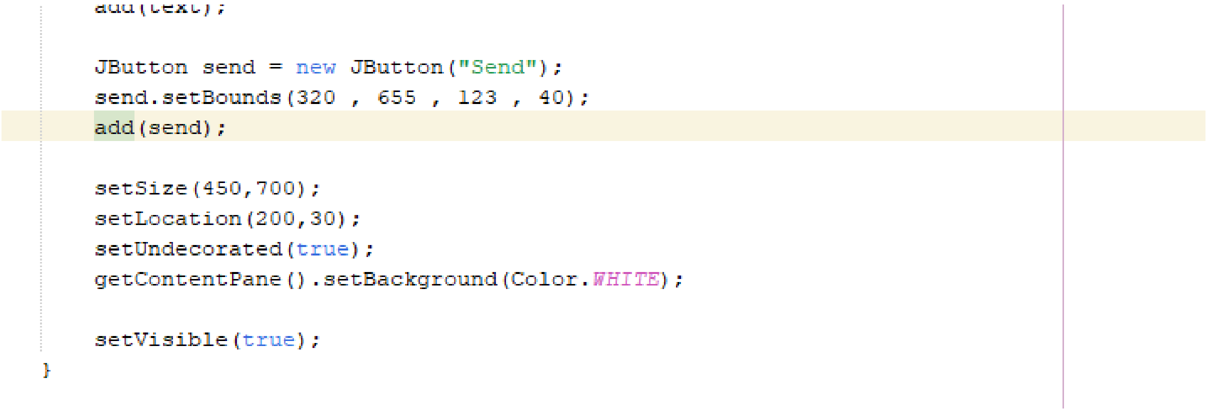
Change Text-Style.



```
    JPanel a1 = new JPanel();
    a1.setBounds(5 , 75 , 440 , 570);
    add(a1);

    JTextField text = new JTextField();
    text.setBounds(5 , 655 , 310 , 40);
    text.setFont(new Font("SAN_SERIF" , Font.PLAIN , 16));
    add(text);

    setSize(450,700);
    setLocation(200,30);
    setUndecorated(true);
    getContentPane().setBackground(Color.WHITE);

    setVisible(true);
}
```

Step-32

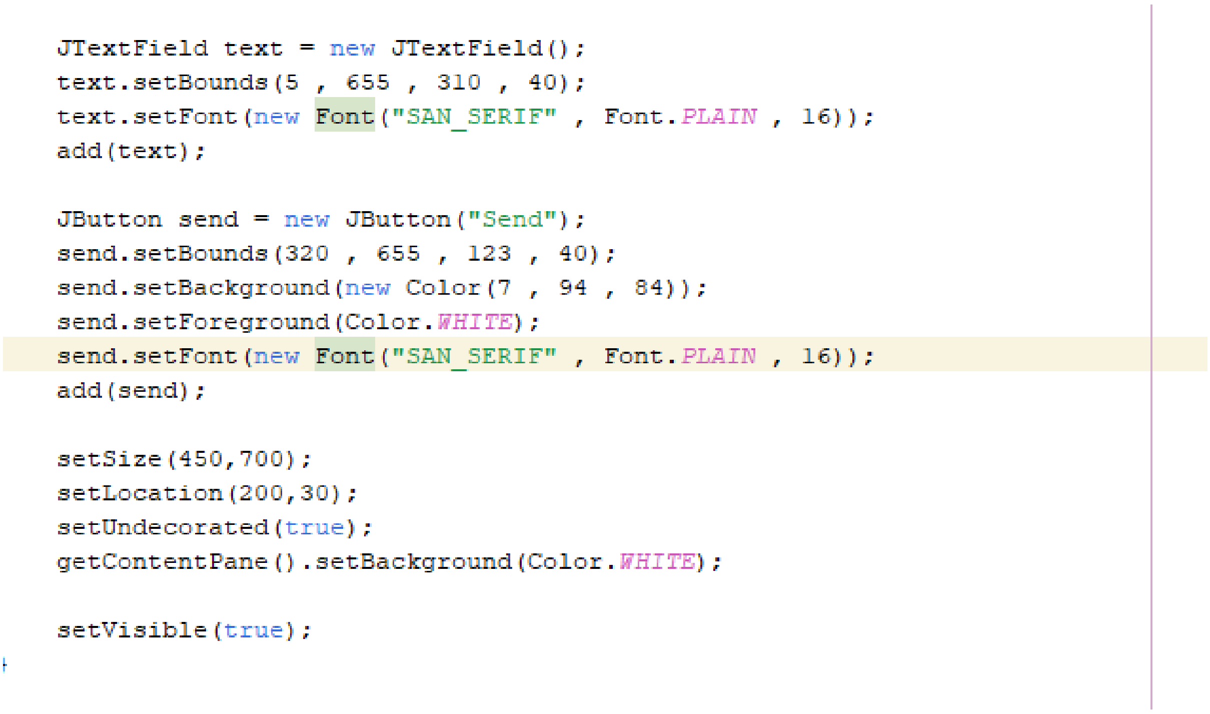
Add Send Button.



```
    add(text);

    JButton send = new JButton("Send");
    send.setBounds(320 , 655 , 123 , 40);
    add(send);

    setSize(450,700);
    setLocation(200,30);
    setUndecorated(true);
    getContentPane().setBackground(Color.WHITE);

    setVisible(true);
}
```

Step-33

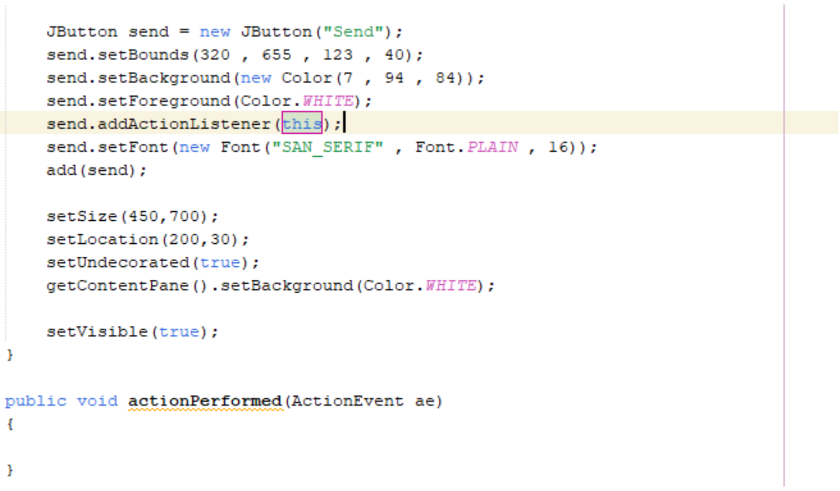
Change Styling of Send Button.



```
    JTextField text = new JTextField();
    text.setBounds(5 , 655 , 310 , 40);
    text.setFont(new Font("SAN_SERIF" , Font.PLAIN , 16));
    add(text);

    JButton send = new JButton("Send");
    send.setBounds(320 , 655 , 123 , 40);
    send.setBackground(new Color(7 , 94 , 84));
    send.setForeground(Color.WHITE);
    send.setFont(new Font("SAN_SERIF" , Font.PLAIN , 16));
    add(send);

    setSize(450,700);
    setLocation(200,30);
    setUndecorated(true);
    getContentPane().setBackground(Color.WHITE);

    setVisible(true);
}
```

Step-34

Add ActionListener to send button.



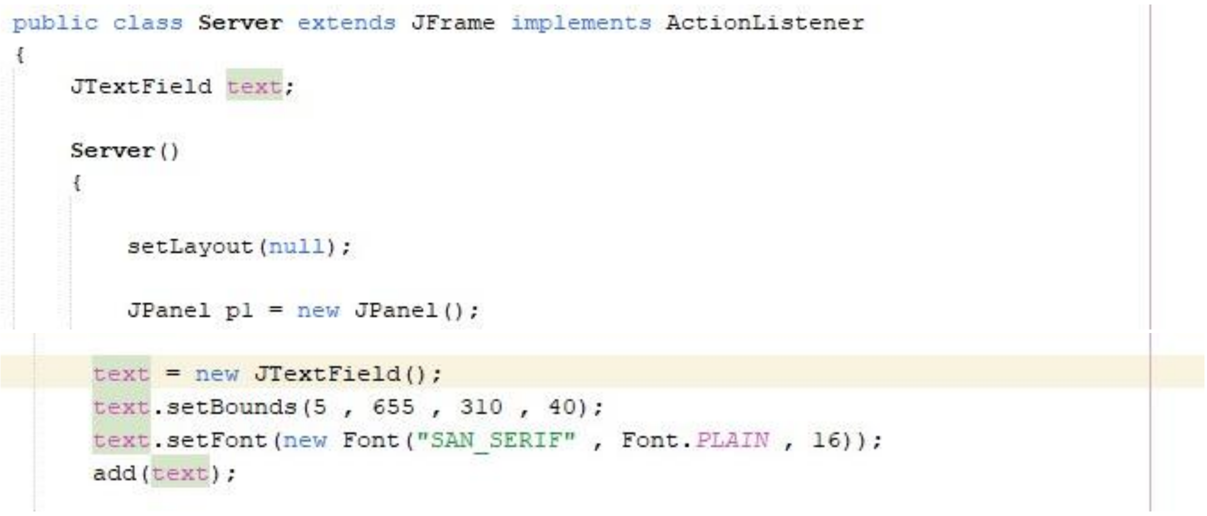
```
        JButton send = new JButton("Send");
        send.setBounds(320 , 655 , 123 , 40);
        send.setBackground(new Color(7 , 94 , 84));
        send.setForeground(Color.WHITE);
        send.addActionListener(this);
        send.setFont(new Font("SAN_SERIF" , Font.PLAIN , 16));
        add(send);

        setSize(450,700);
        setLocation(200,30);
        setUndecorated(true);
        getContentPane().setBackground(Color.WHITE);

        setVisible(true);
}

public void actionPerformed(ActionEvent ae)
{

}
```

Step-35

Now we require to fetch text from JTextFeild.

However, JTextFeild is in the scope of constructor and this does not allow us to fetch text in actionPerformed() method.

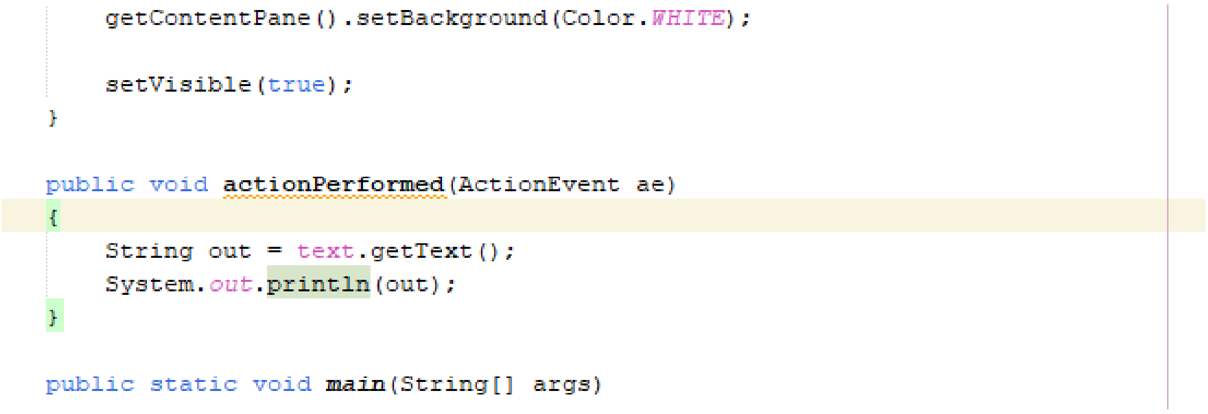
So, we need to declare JTextFeild globally.



```
public class Server extends JFrame implements ActionListener
{
    JTextField text;

    Server()
    {

        setLayout(null);

        JPanel p1 = new JPanel();

        text = new JTextField();
        text.setBounds(5 , 655 , 310 , 40);
        text.setFont(new Font("SAN_SERIF" , Font.PLAIN , 16));
        add(text);
```

Step-36

Complete actionPerformed() method to fetch text.



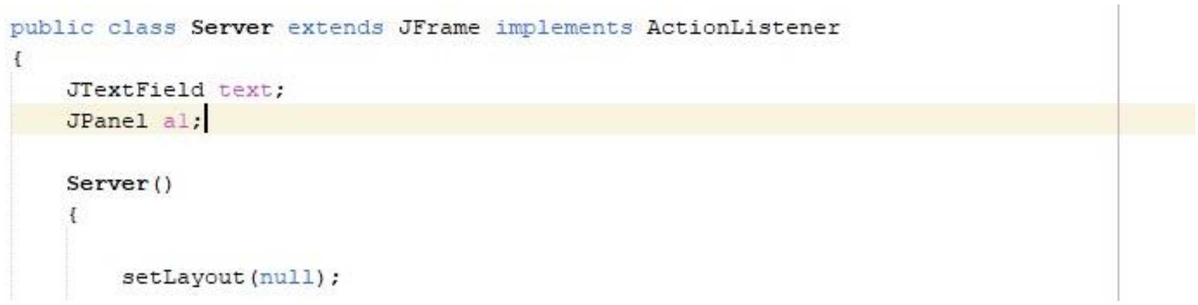
```
        getContentPane().setBackground(Color.WHITE);

        setVisible(true);
    }

    public void actionPerformed(ActionEvent ae)
    {
        String out = text.getText();
        System.out.println(out);
    }

    public static void main(String[] args)
```

Step-37

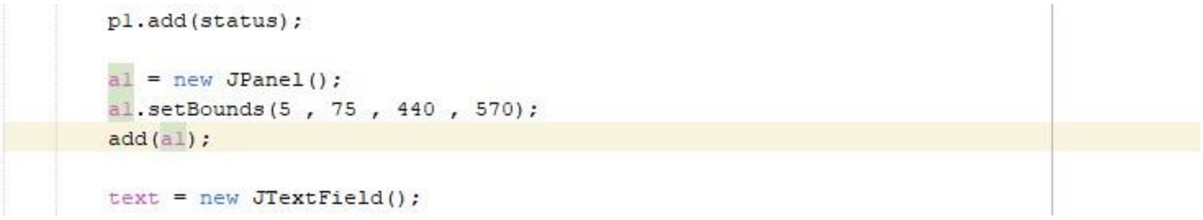
Now, fetched text needs to be displayed on panel a1.

However, panel a1 is in the scope of constructor.

So, we first need to declare panel a1 globally.



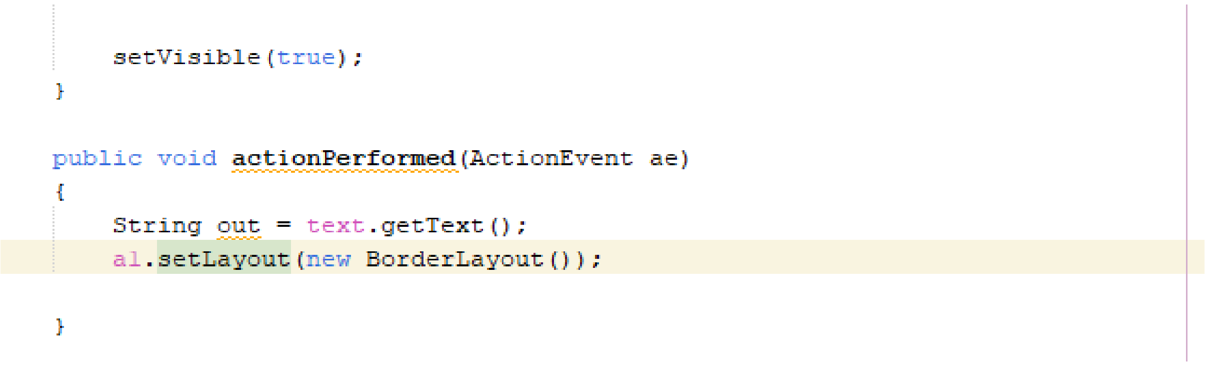
```
public class Server extends JFrame implements ActionListener
{
    JTextField text;
    JPanel a1;

    Server()
    {

        setLayout(null);
```



```
        p1.add(status);

        a1 = new JPanel();
        a1.setBounds(5 , 75 , 440 , 570);
        add(a1);

        text = new JTextField();
```

Step-38

Set Layout of a1 as border layout.

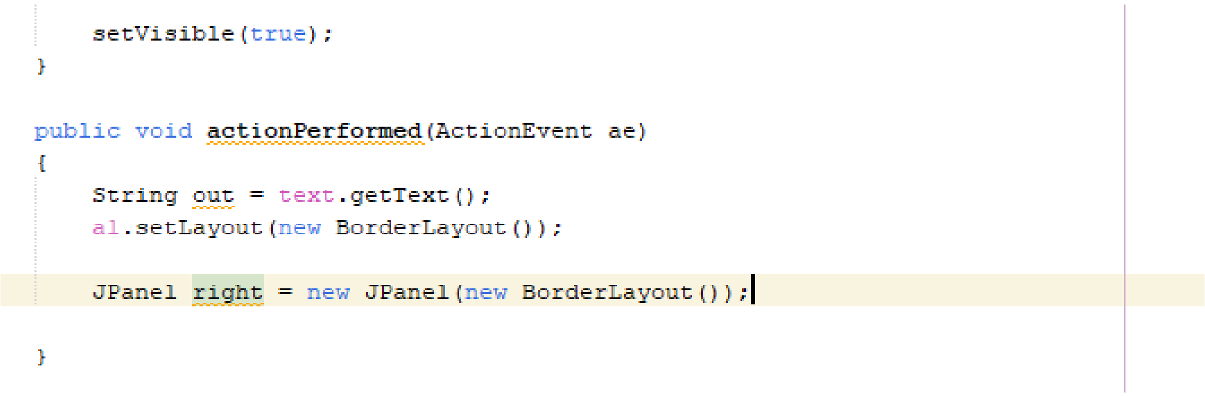


```
    setVisible(true);
}

public void actionPerformed(ActionEvent ae)
{
    String out = text.getText();
    a1.setLayout(new BorderLayout());

}
```

Step-39

Create panel to display messages to the right.

(panel right)

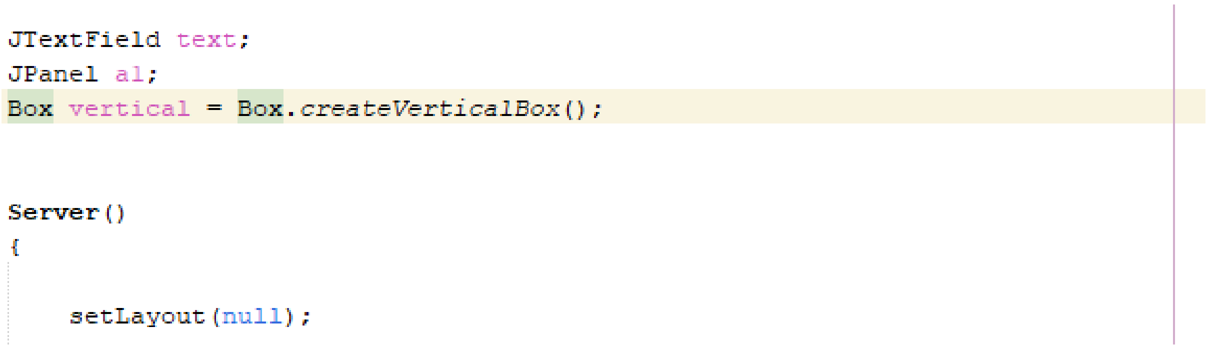


```
    setVisible(true);
}

public void actionPerformed(ActionEvent ae)
{
    String out = text.getText();
    a1.setLayout(new BorderLayout());

    JPanel right = new JPanel(new BorderLayout());

}
```

Step-40

Create a global vertical box.

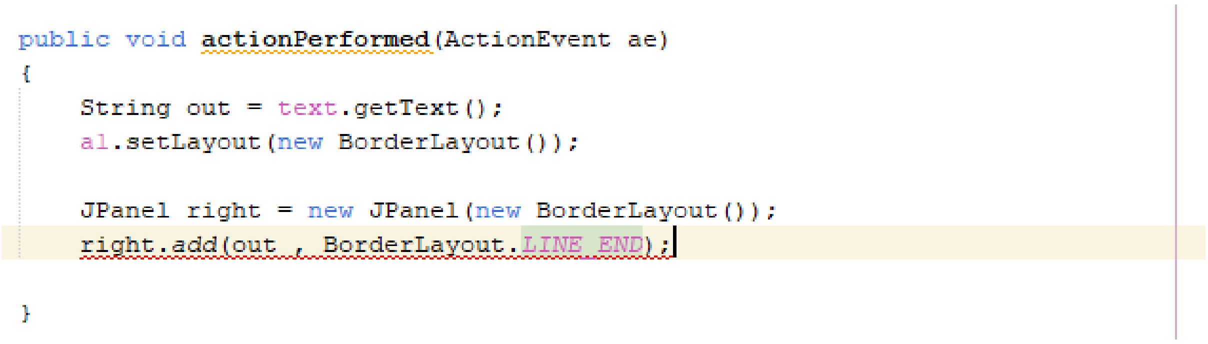
(Box is needed to display msgs vertically below each other on the right)



```
JTextField text;
JPanel al;
Box vertical = Box.createVerticalBox();


Server()
{

    setLayout(null);
```

Step-41

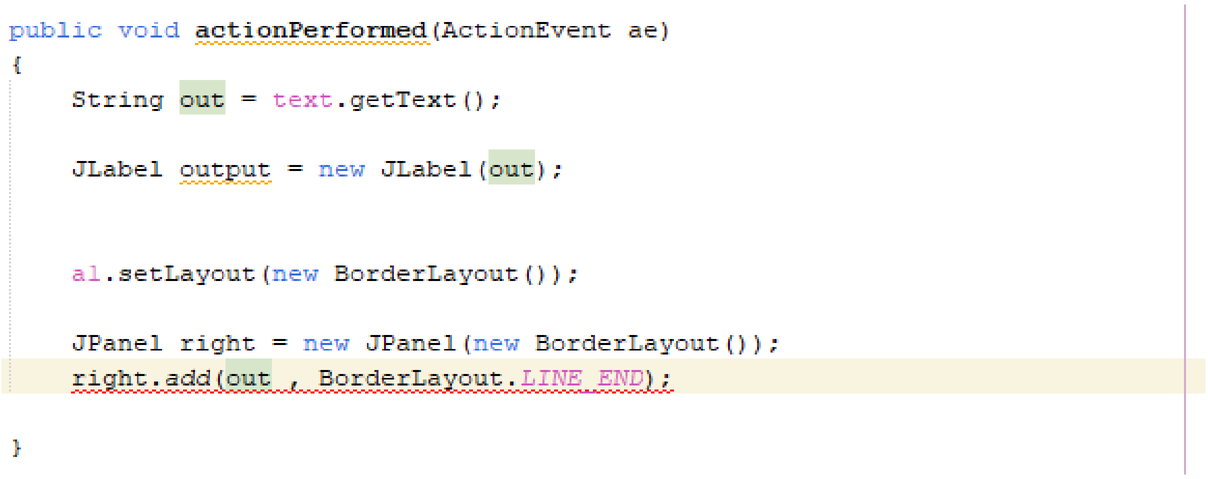
Add output text to right panel.



```
public void actionPerformed(ActionEvent ae)
{
    String out = text.getText();
    al.setLayout(new BorderLayout());

    JPanel right = new JPanel(new BorderLayout());
    right.add(out , BorderLayout.LINE_END);

}
```

Step-42

Create Label output to store string out.

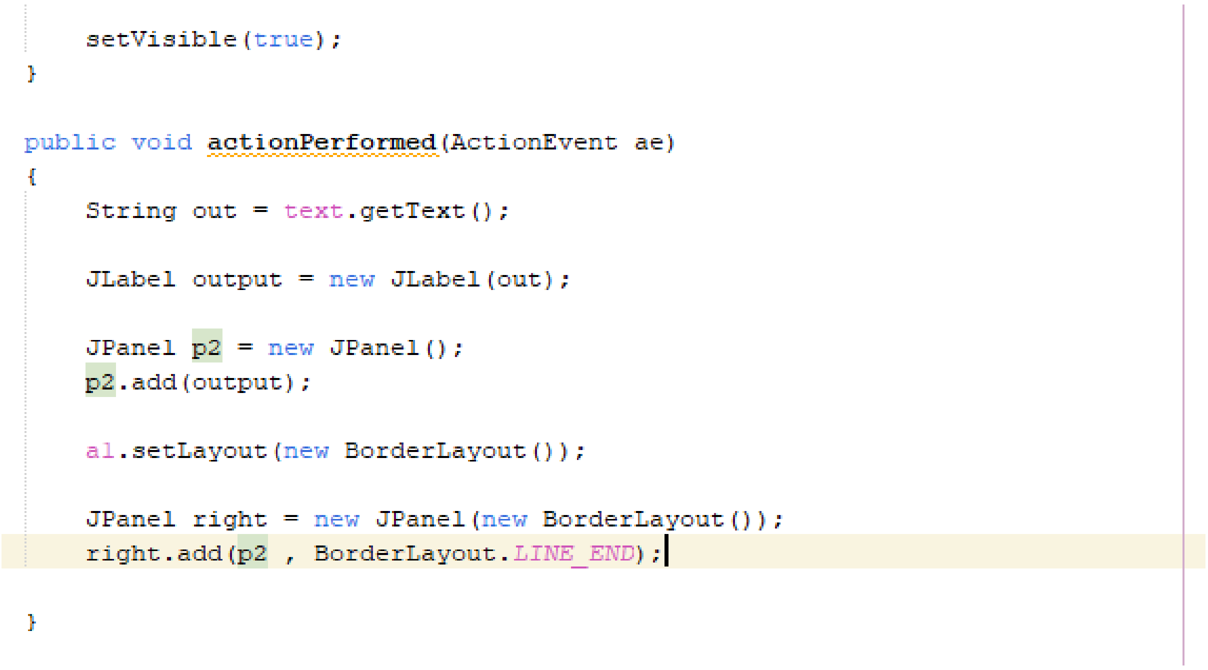


```java
public void actionPerformed(ActionEvent ae)
{
    String out = text.getText();

    JLabel output = new JLabel(out);


    al.setLayout(new BorderLayout());

    JPanel right = new JPanel(new BorderLayout());
    right.add(out , BorderLayout.LINE_END);

}
```

Step-43

Create panel p2 and add label output to it.

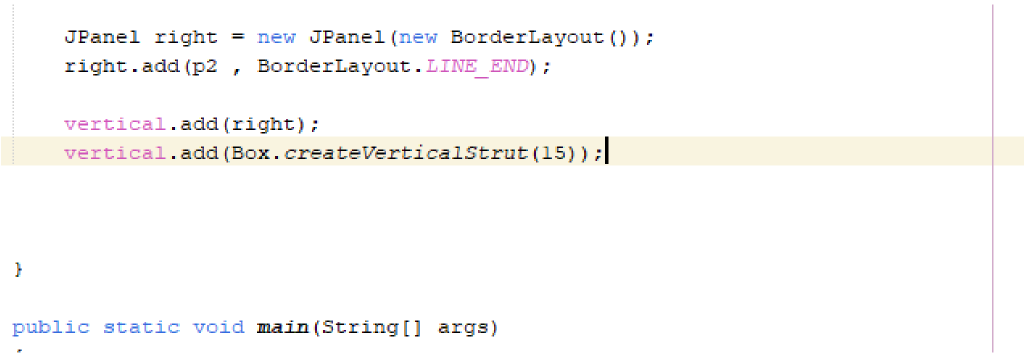
(Additionally add panel p2 to right panel and this will solve error caused in step-41)



```java
    setVisible(true);
}

public void actionPerformed(ActionEvent ae)
{
    String out = text.getText();

    JLabel output = new JLabel(out);

    JPanel p2 = new JPanel();
    p2.add(output);

    al.setLayout(new BorderLayout());

    JPanel right = new JPanel(new BorderLayout());
    right.add(p2 , BorderLayout.LINE_END);

}
```

Step-44

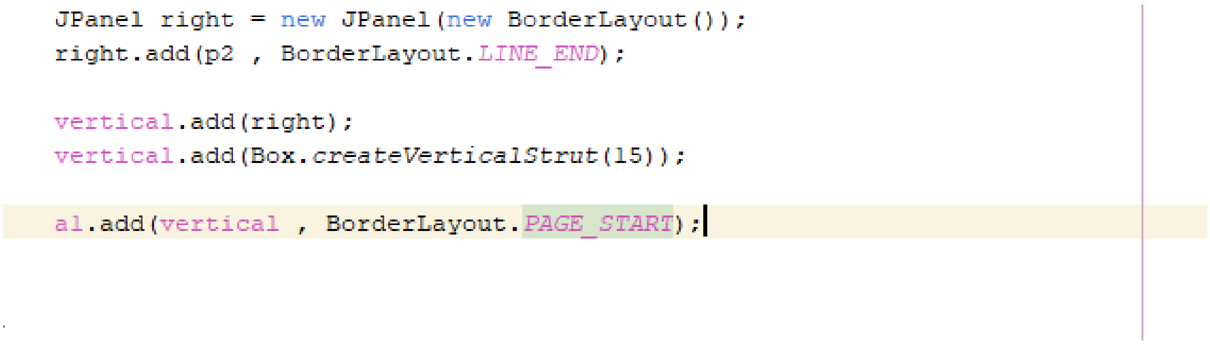
Add panel right to vertical box and create vertical struct.



```
    JPanel right = new JPanel(new BorderLayout());
    right.add(p2 , BorderLayout.LINE_END);

    vertical.add(right);
    vertical.add(Box.createVerticalStrut(15));



}

public static void main(String[] args)
```

Step-45

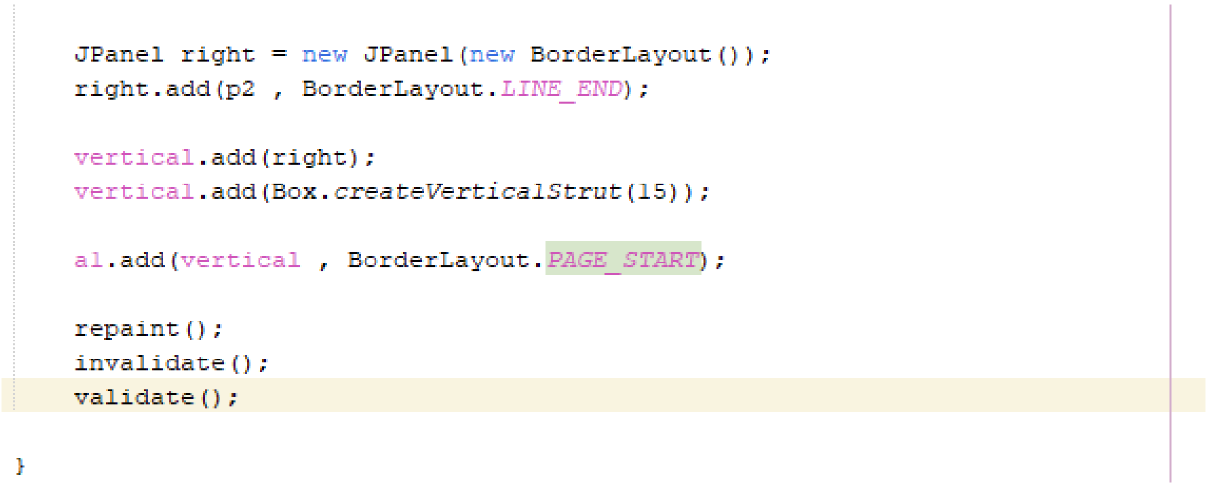
Add vertical panel to panel a1.



```
JPanel right = new JPanel(new BorderLayout());
right.add(p2 , BorderLayout.LINE_END);

vertical.add(right);
vertical.add(Box.createVerticalStrut(15));

al.add(vertical , BorderLayout.PAGE_START);
```

Step-46

Call functions to refresh *(repaint)* the frame.



```
    JPanel right = new JPanel(new BorderLayout());
    right.add(p2 , BorderLayout.LINE_END);

    vertical.add(right);
    vertical.add(Box.createVerticalStrut(15));

    al.add(vertical , BorderLayout.PAGE_START);

    repaint();
    invalidate();
    validate();

}
```

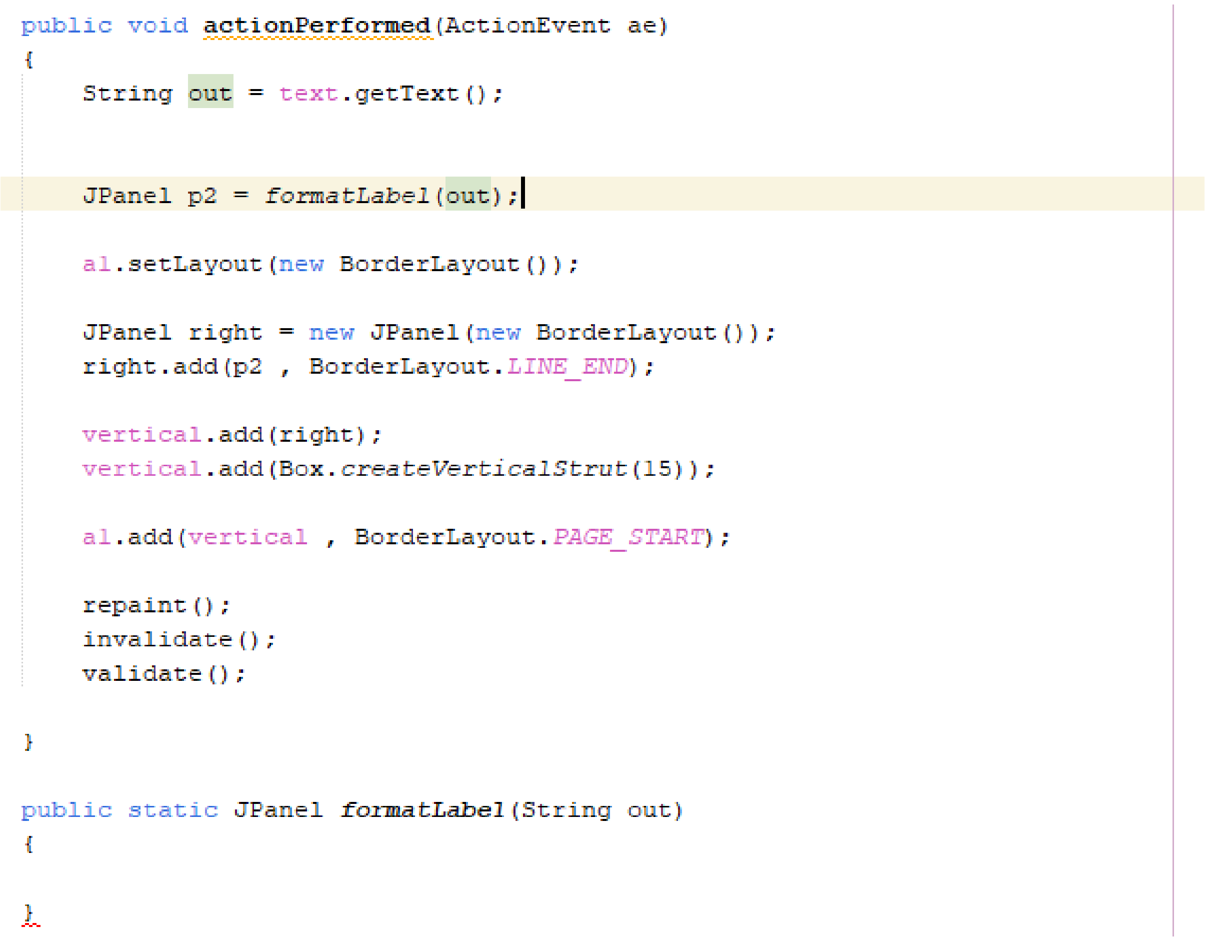
Step-47

Now, we wish to display the text in a box.

(So, we need to make changes in panel p2)

Create method formatLabel() with string out as parameter and return type of this method will be JPanel and calling this method while declaring p2 will make changes in p2.

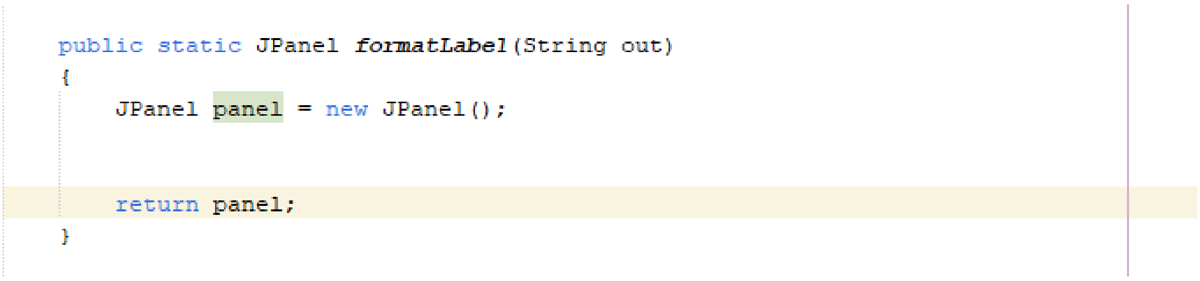
(Also no more need of JLabel output)



```
public void actionPerformed(ActionEvent ae)
{
    String out = text.getText();


    JPanel p2 = formatLabel(out);

    al.setLayout(new BorderLayout());

    JPanel right = new JPanel(new BorderLayout());
    right.add(p2 , BorderLayout.LINE_END);

    vertical.add(right);
    vertical.add(Box.createVerticalStrut(15));

    al.add(vertical , BorderLayout.PAGE_START);

    repaint();
    invalidate();
    validate();

}

public static JPanel formatLabel(String out)
{

}
```

Step-48

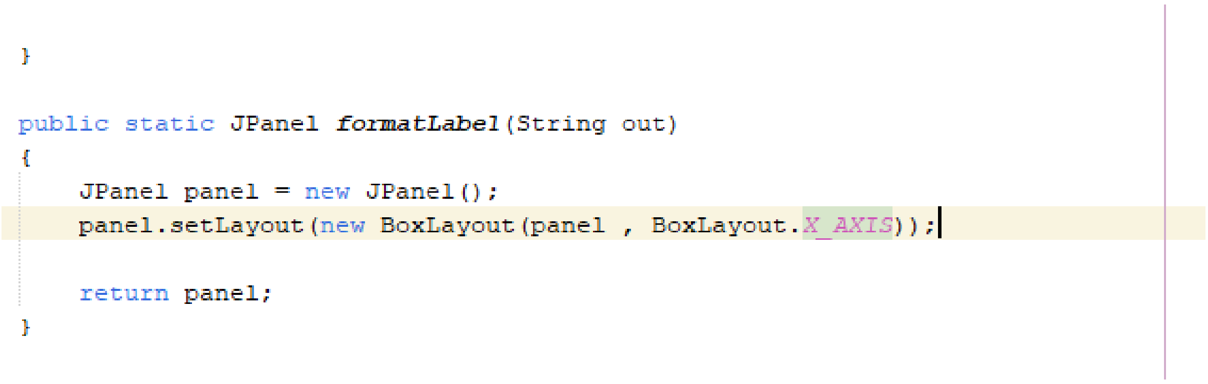
Create and return Dummy Panel inside formatLabel().



```
public static JPanel formatLabel(String out)
{
    JPanel panel = new JPanel();


    return panel;
}
```

Step-49

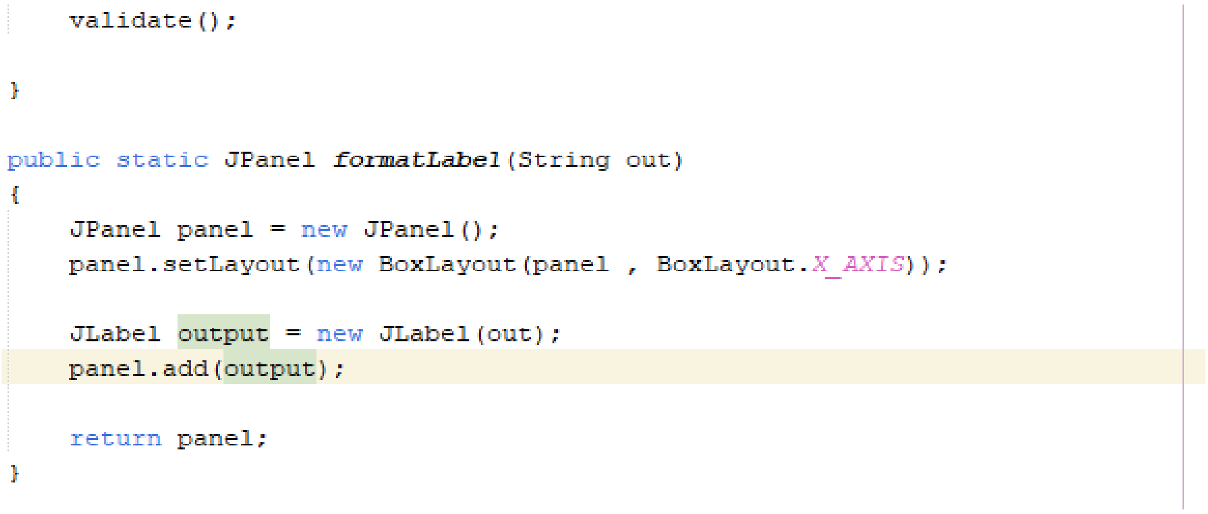
Set Layout for panel.



```
}

public static JPanel formatLabel(String out)
{
    JPanel panel = new JPanel();
    panel.setLayout(new BoxLayout(panel , BoxLayout.X_AXIS));

    return panel;
}
```

Step-50

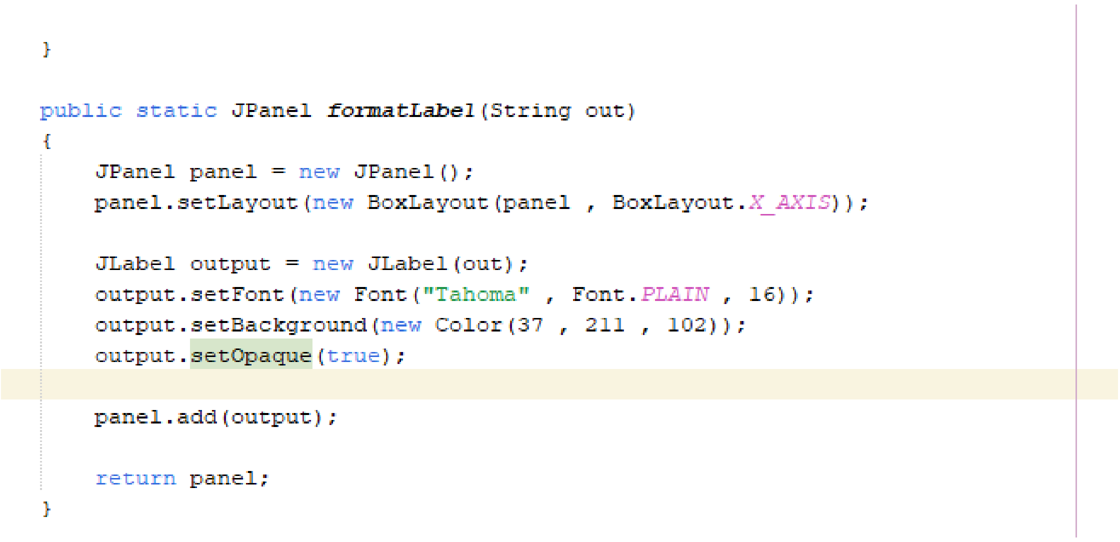
Create JLabel output and add to Dummy Panel.



```
    validate();

}

public static JPanel formatLabel(String out)
{
    JPanel panel = new JPanel();
    panel.setLayout(new BoxLayout(panel , BoxLayout.X_AXIS));

    JLabel output = new JLabel(out);
    panel.add(output);

    return panel;
}
```

Step-51

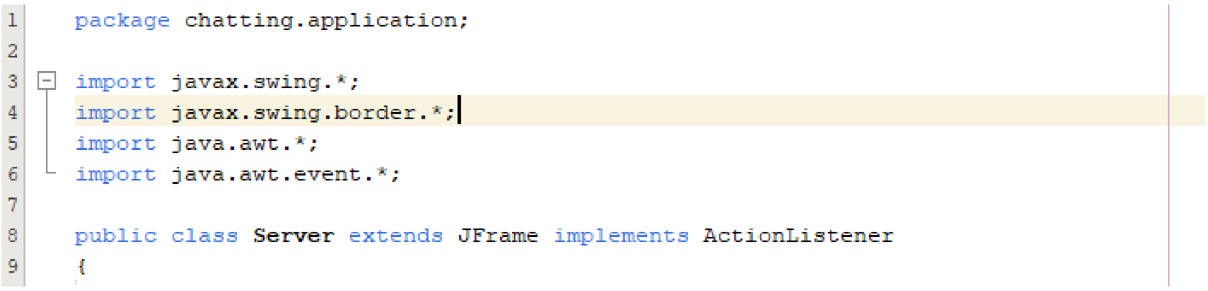
Change style and font of output text.



```
    }

    public static JPanel formatLabel(String out)
    {
        JPanel panel = new JPanel();
        panel.setLayout(new BoxLayout(panel , BoxLayout.X_AXIS));

        JLabel output = new JLabel(out);
        output.setFont(new Font("Tahoma" , Font.PLAIN , 16));
        output.setBackground(new Color(37 , 211 , 102));
        output.setOpaque(true);

        panel.add(output);

        return panel;
    }
```

Step-52

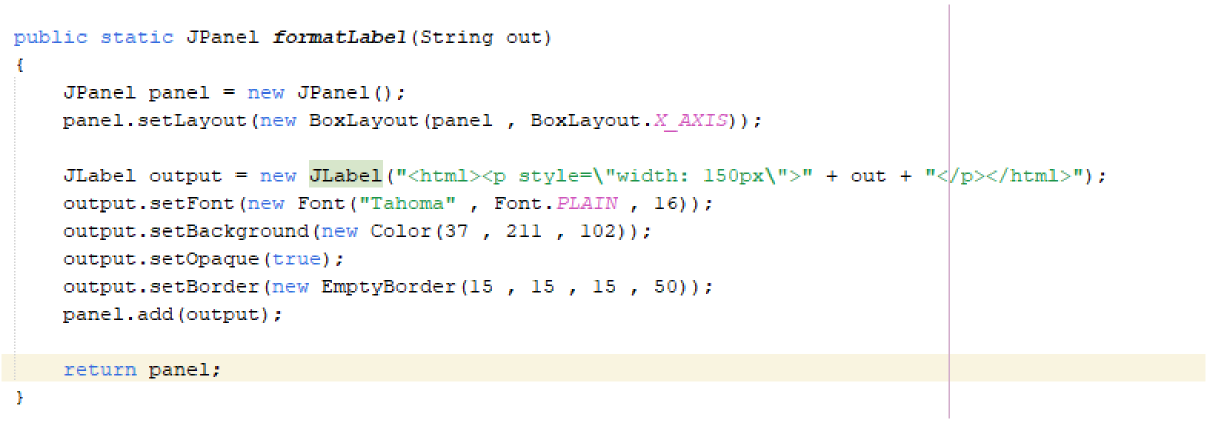
Set Border for output text and import library.



```
package chatting.application;

import javax.swing.*;
import javax.swing.border.*;
import java.awt.*;
import java.awt.event.*;

public class Server extends JFrame implements ActionListener
{
```

Step-53

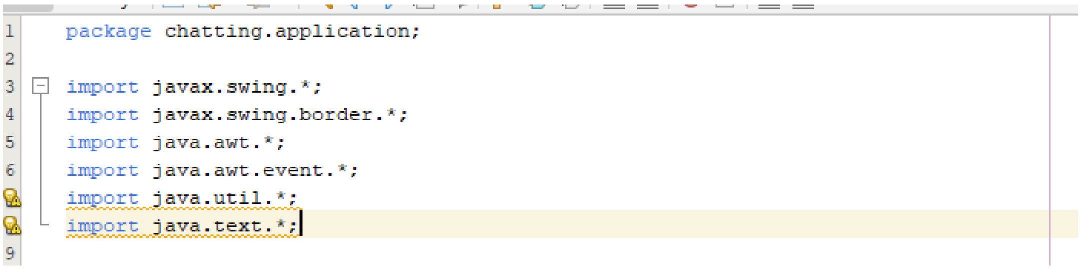
Set width for output text using Html and Css tags.



```
public static JPanel formatLabel(String out)
{
    JPanel panel = new JPanel();
    panel.setLayout(new BoxLayout(panel , BoxLayout.X_AXIS));

    JLabel output = new JLabel("<html><p style=\"width: 150px\">" + out + "</p></html>");
    output.setFont(new Font("Tahoma" , Font.PLAIN , 16));
    output.setBackground(new Color(37 , 211 , 102));
    output.setOpaque(true);
    output.setBorder(new EmptyBorder(15 , 15 , 15 , 50));
    panel.add(output);

    return panel;
}
```

Step-54

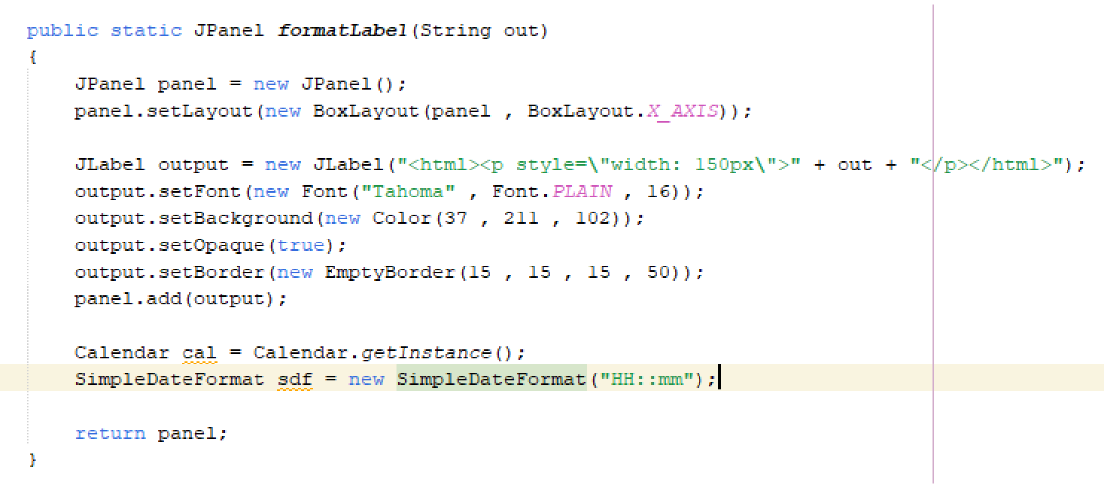
Import Libraries for calendar and text.



```
package chatting.application;

import javax.swing.*;
import javax.swing.border.*;
import java.awt.*;
import java.awt.event.*;
import java.util.*;
import java.text.*;
```

Step-55

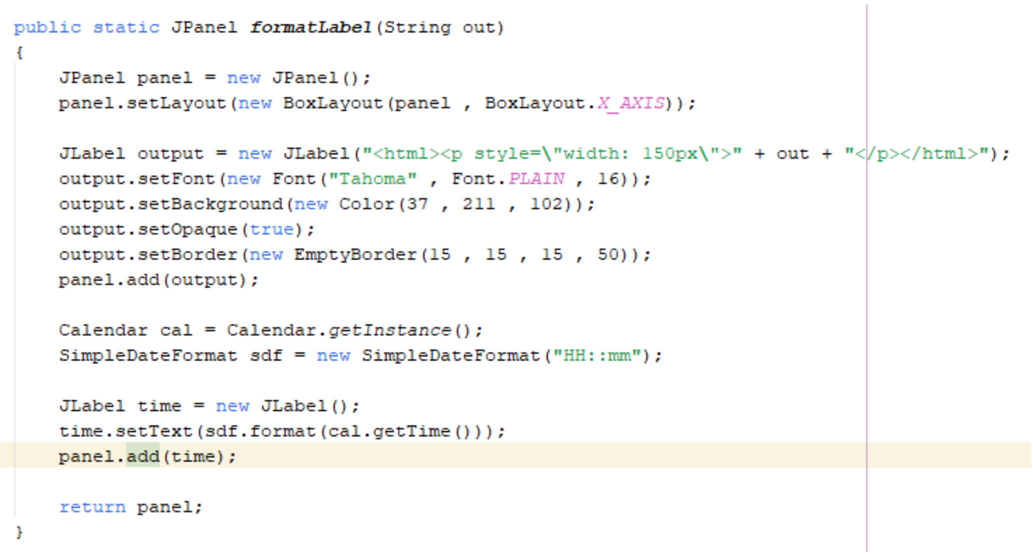
Create Objects of Calendar and Simple Date Format.



```
public static JPanel formatLabel(String out)
{
    JPanel panel = new JPanel();
    panel.setLayout(new BoxLayout(panel , BoxLayout.X_AXIS));

    JLabel output = new JLabel("<html><p style=\"width: 150px\">" + out + "</p></html>");
    output.setFont(new Font("Tahoma" , Font.PLAIN , 16));
    output.setBackground(new Color(37 , 211 , 102));
    output.setOpaque(true);
    output.setBorder(new EmptyBorder(15 , 15 , 15 , 50));
    panel.add(output);

    Calendar cal = Calendar.getInstance();
    SimpleDateFormat sdf = new SimpleDateFormat("HH::mm");

    return panel;
}
```

Step-56

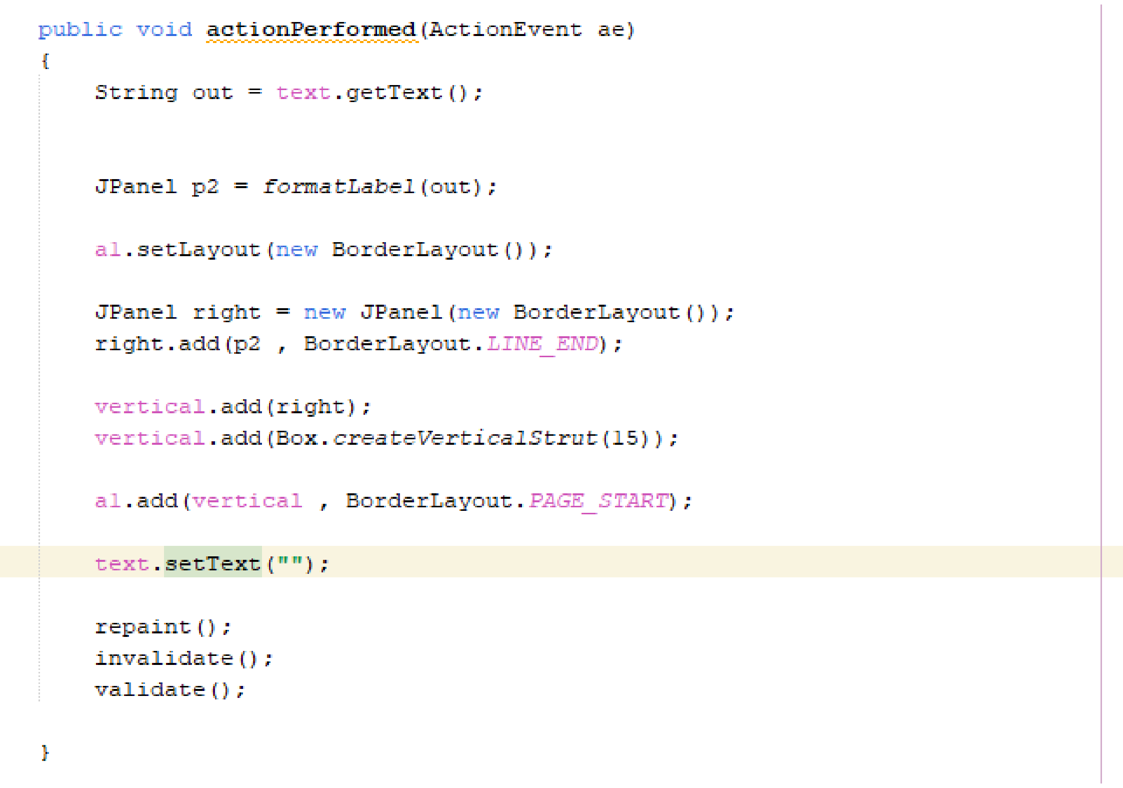
Create and Add label to display time.



```
public static JPanel formatLabel(String out)
{
    JPanel panel = new JPanel();
    panel.setLayout(new BoxLayout(panel , BoxLayout.X_AXIS));

    JLabel output = new JLabel("<html><p style=\"width: 150px\">" + out + "</p></html>");
    output.setFont(new Font("Tahoma" , Font.PLAIN , 16));
    output.setBackground(new Color(37 , 211 , 102));
    output.setOpaque(true);
    output.setBorder(new EmptyBorder(15 , 15 , 15 , 50));
    panel.add(output);

    Calendar cal = Calendar.getInstance();
    SimpleDateFormat sdf = new SimpleDateFormat("HH::mm");

    JLabel time = new JLabel();
    time.setText(sdf.format(cal.getTime()));
    panel.add(time);

    return panel;
}
```

Step-57

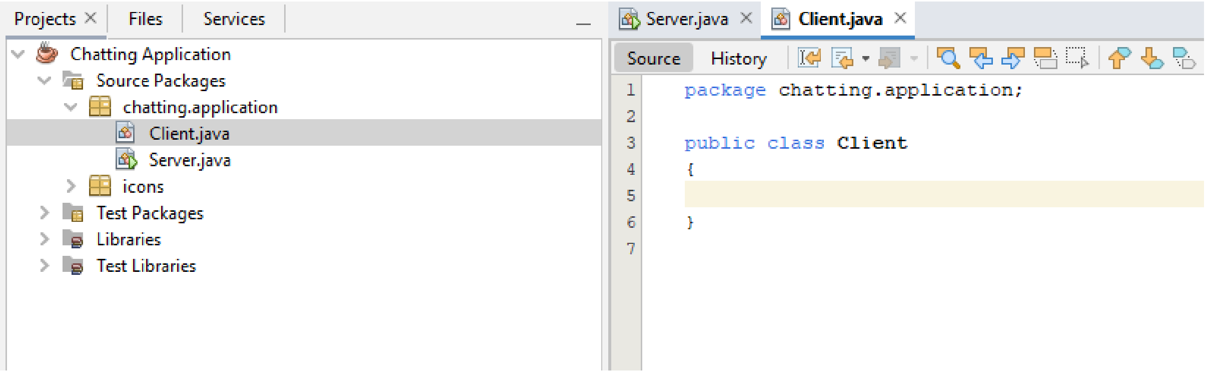
Reset Text in Text-Box to Empty.



```java
public void actionPerformed(ActionEvent ae)
{
    String out = text.getText();


    JPanel p2 = formatLabel(out);

    a1.setLayout(new BorderLayout());

    JPanel right = new JPanel(new BorderLayout());
    right.add(p2 , BorderLayout.LINE_END);

    vertical.add(right);
    vertical.add(Box.createVerticalStrut(15));

    a1.add(vertical , BorderLayout.PAGE_START);

    text.setText("");

    repaint();
    invalidate();
    validate();

}
```

Step-58

Create a file Client.java with clean code.



```java
package chatting.application;

public class Client
{

}
```

Step-59

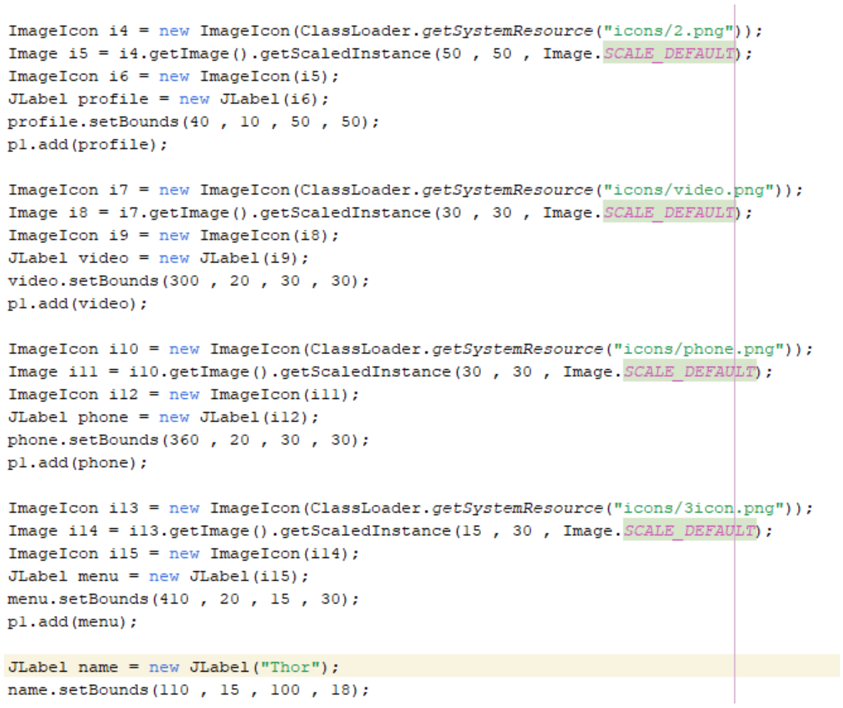
Copy complete code from server.java to client.java

Additionally make three changes

- Change name of class to public class Client
- Change name of constructor to Client
- Change name of anonymous object in psvm to Client

Step-60

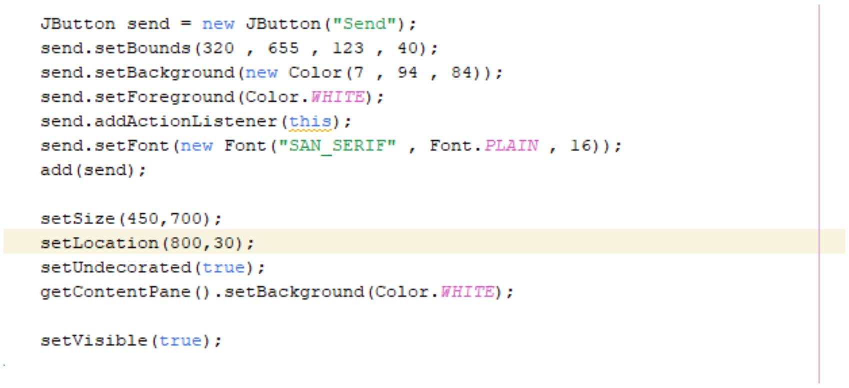
In Client.java change Profile Image and Profile name.



```
ImageIcon i4 = new ImageIcon(ClassLoader.getSystemResource("icons/2.png"));
Image i5 = i4.getImage().getScaledInstance(50 , 50 , Image.SCALE_DEFAULT);
ImageIcon i6 = new ImageIcon(i5);
JLabel profile = new JLabel(i6);
profile.setBounds(40 , 10 , 50 , 50);
p1.add(profile);

ImageIcon i7 = new ImageIcon(ClassLoader.getSystemResource("icons/video.png"));
Image i8 = i7.getImage().getScaledInstance(30 , 30 , Image.SCALE_DEFAULT);
ImageIcon i9 = new ImageIcon(i8);
JLabel video = new JLabel(i9);
video.setBounds(300 , 20 , 30 , 30);
p1.add(video);

ImageIcon i10 = new ImageIcon(ClassLoader.getSystemResource("icons/phone.png"));
Image i11 = i10.getImage().getScaledInstance(30 , 30 , Image.SCALE_DEFAULT);
ImageIcon i12 = new ImageIcon(i11);
JLabel phone = new JLabel(i12);
phone.setBounds(360 , 20 , 30 , 30);
p1.add(phone);

ImageIcon i13 = new ImageIcon(ClassLoader.getSystemResource("icons/3icon.png"));
Image i14 = i13.getImage().getScaledInstance(15 , 30 , Image.SCALE_DEFAULT);
ImageIcon i15 = new ImageIcon(i14);
JLabel menu = new JLabel(i15);
menu.setBounds(410 , 20 , 15 , 30);
p1.add(menu);

JLabel name = new JLabel("Thor");
name.setBounds(110 , 15 , 100 , 18);
```

Step-61

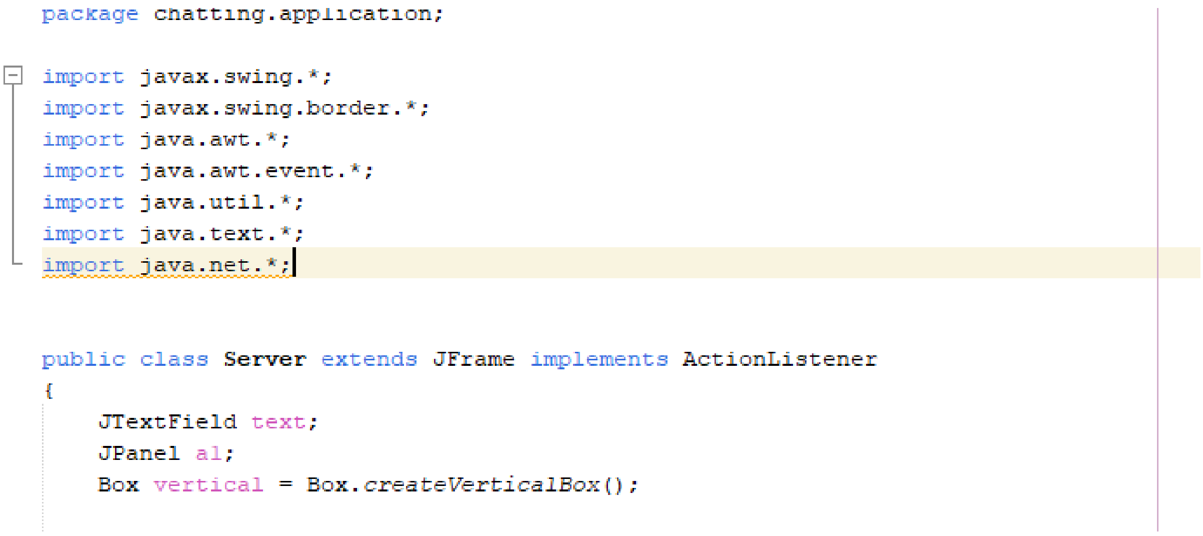
Change Location of Client Frame.



```
JButton send = new JButton("Send");
send.setBounds(320 , 655 , 123 , 40);
send.setBackground(new Color(7 , 94 , 84));
send.setForeground(Color.WHITE);
send.addActionListener(this);
send.setFont(new Font("SAN_SERIF" , Font.PLAIN , 16));
add(send);

setSize(450,700);
setLocation(800,30);
setUndecorated(true);
getContentPane().setBackground(Color.WHITE);

setVisible(true);
```

Step-62

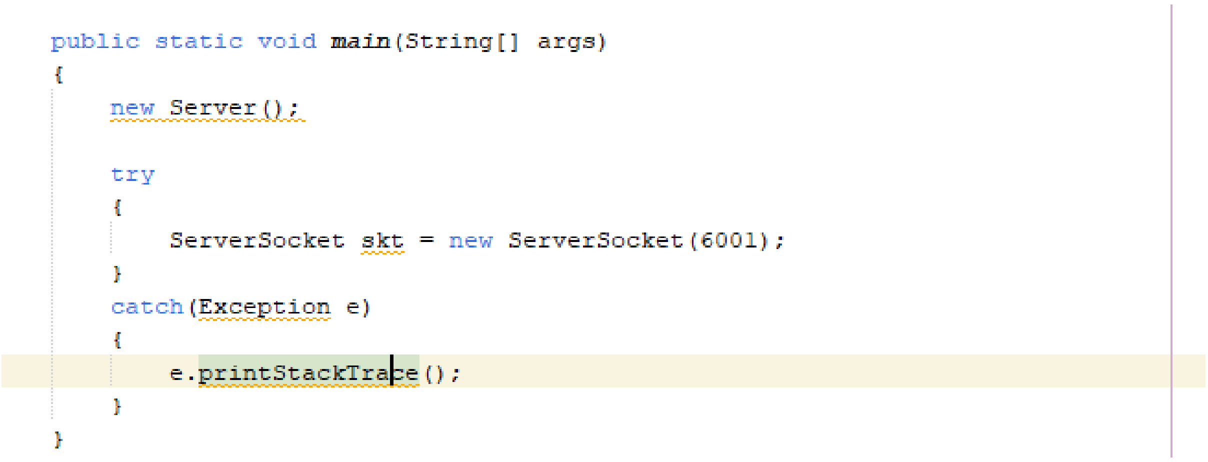
Import java-net library, which helps in creating server and socket.



```
package chatting.application;

import javax.swing.*;
import javax.swing.border.*;
import java.awt.*;
import java.awt.event.*;
import java.util.*;
import java.text.*;
import java.net.*;


public class Server extends JFrame implements ActionListener
{
    JTextField text;
    JPanel a1;
    Box vertical = Box.createVerticalBox();
```

Step-63

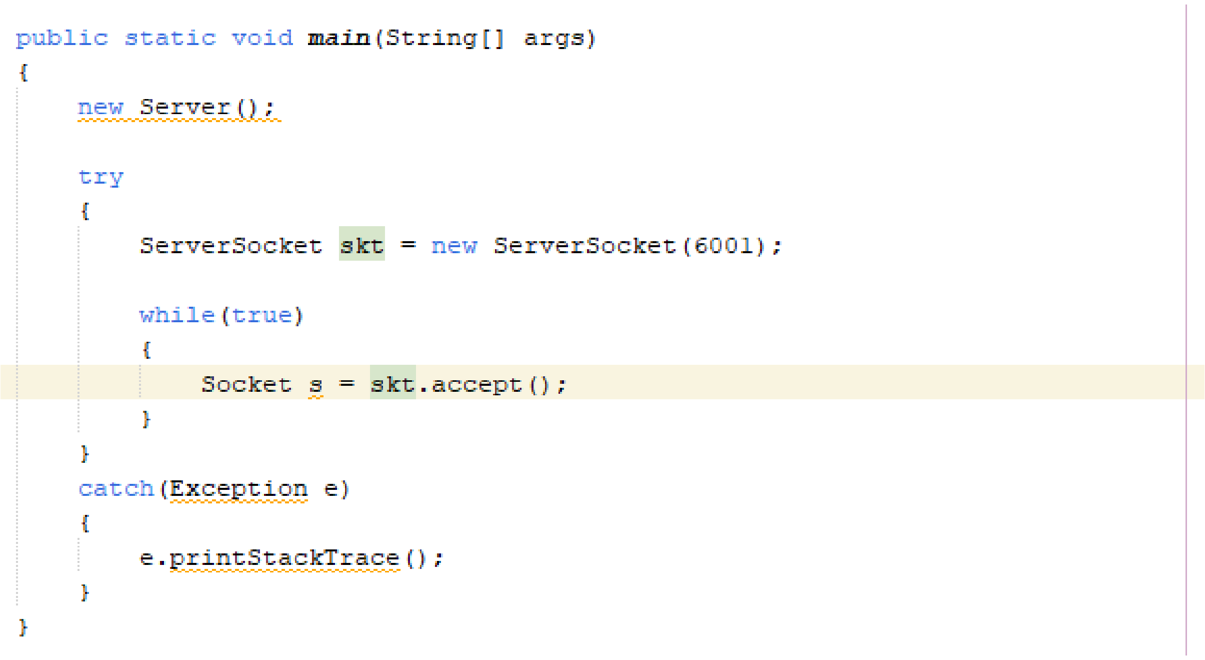
Introduce a try catch block inside with Server Socket inside psvm.



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

    try
    {
        ServerSocket skt = new ServerSocket(6001);
    }
    catch(Exception e)
    {
        e.printStackTrace();
    }
}
```

Step-64

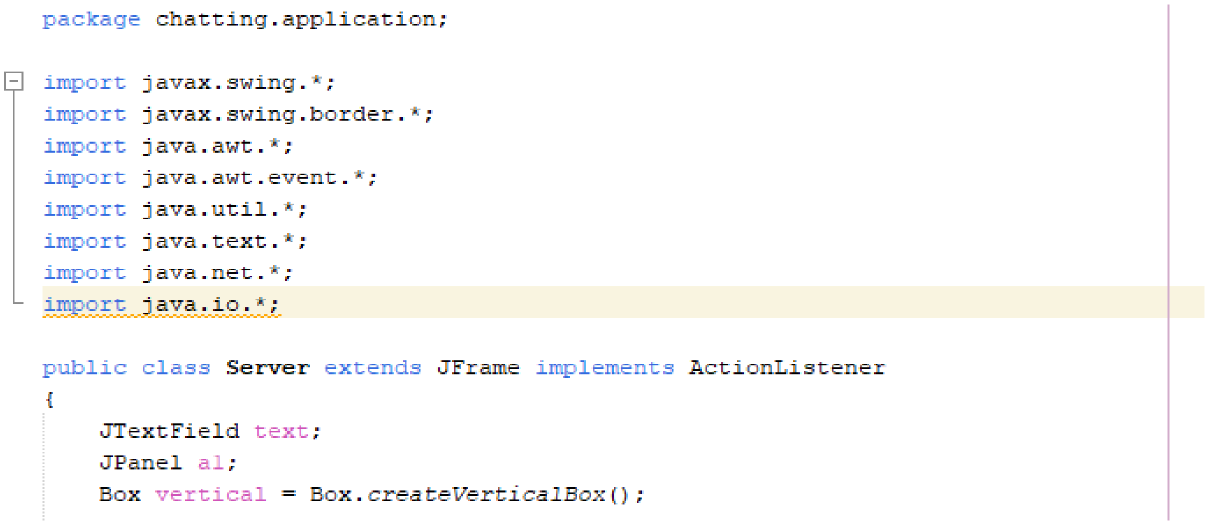
Create an infinite loop with socket s to accept msgs.



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

    try
    {
        ServerSocket skt = new ServerSocket(6001);

        while(true)
        {
            Socket s = skt.accept();
        }
    }
    catch(Exception e)
    {
        e.printStackTrace();
    }
}
```

Step-65

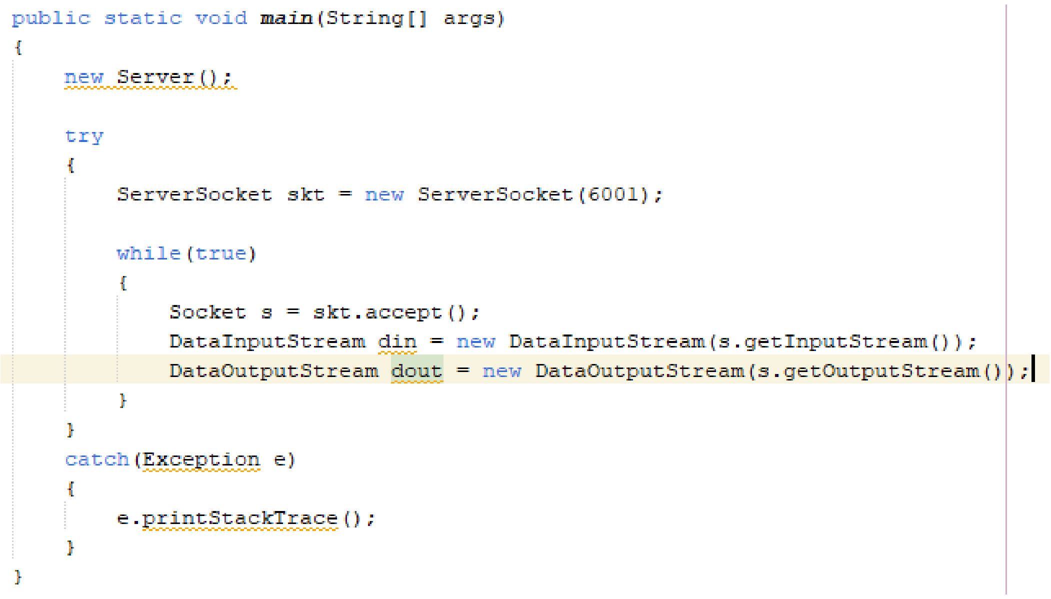
Import java input output library.



```
package chatting.application;

import javax.swing.*;
import javax.swing.border.*;
import java.awt.*;
import java.awt.event.*;
import java.util.*;
import java.text.*;
import java.net.*;
import java.io.*;

public class Server extends JFrame implements ActionListener
{
    JTextField text;
    JPanel a1;
    Box vertical = Box.createVerticalBox();
```

Step-66

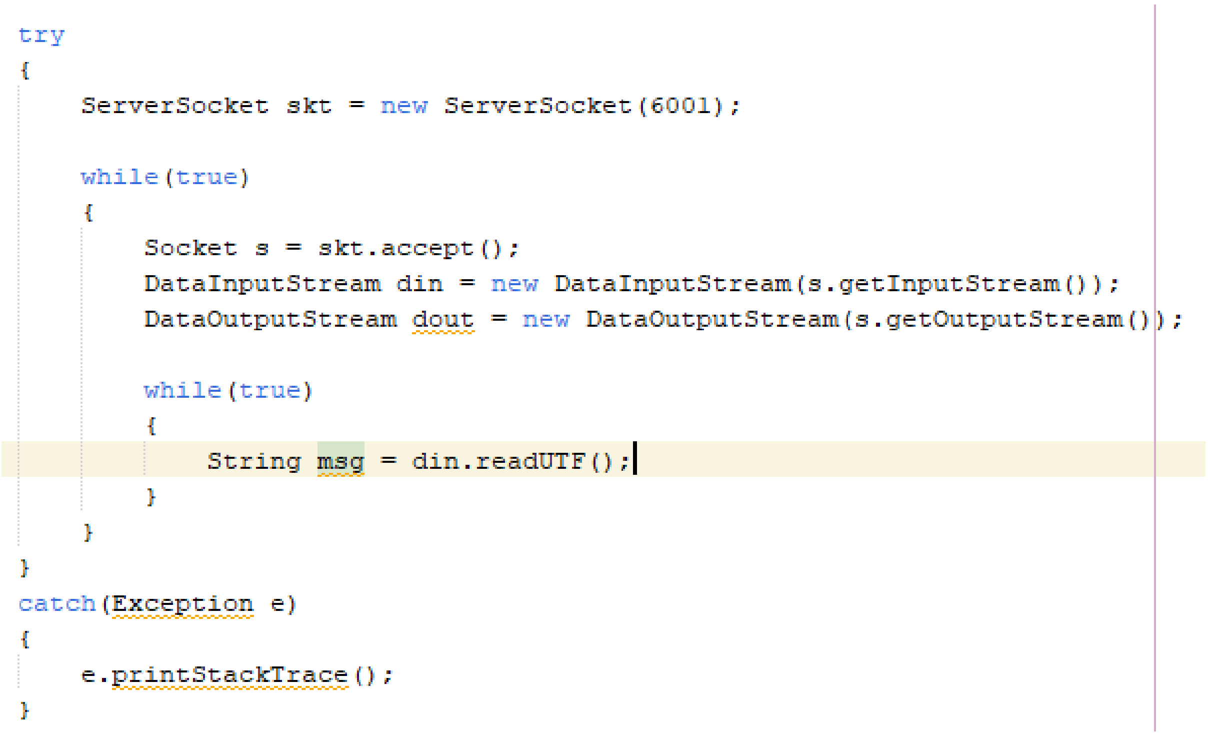
Declare data input & output stream.



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

    try
    {
        ServerSocket skt = new ServerSocket(6001);

        while(true)
        {
            Socket s = skt.accept();
            DataInputStream din = new DataInputStream(s.getInputStream());
            DataOutputStream dout = new DataOutputStream(s.getOutputStream());
        }
    }
    catch(Exception e)
    {
        e.printStackTrace();
    }
}
```

Step-67

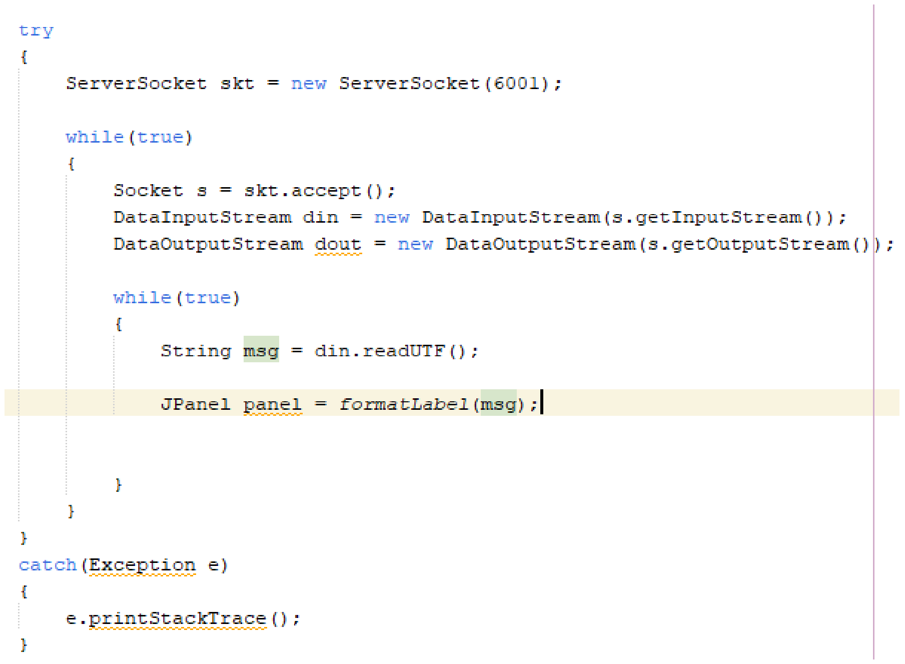
Stored the received msg into a string using UTF protocol.



```
try
{
    ServerSocket skt = new ServerSocket(6001);

    while(true)
    {
        Socket s = skt.accept();
        DataInputStream din = new DataInputStream(s.getInputStream());
        DataOutputStream dout = new DataOutputStream(s.getOutputStream());

        while(true)
        {
            String msg = din.readUTF();
        }
    }
}
catch(Exception e)
{
    e.printStackTrace();
}
```

Step-68

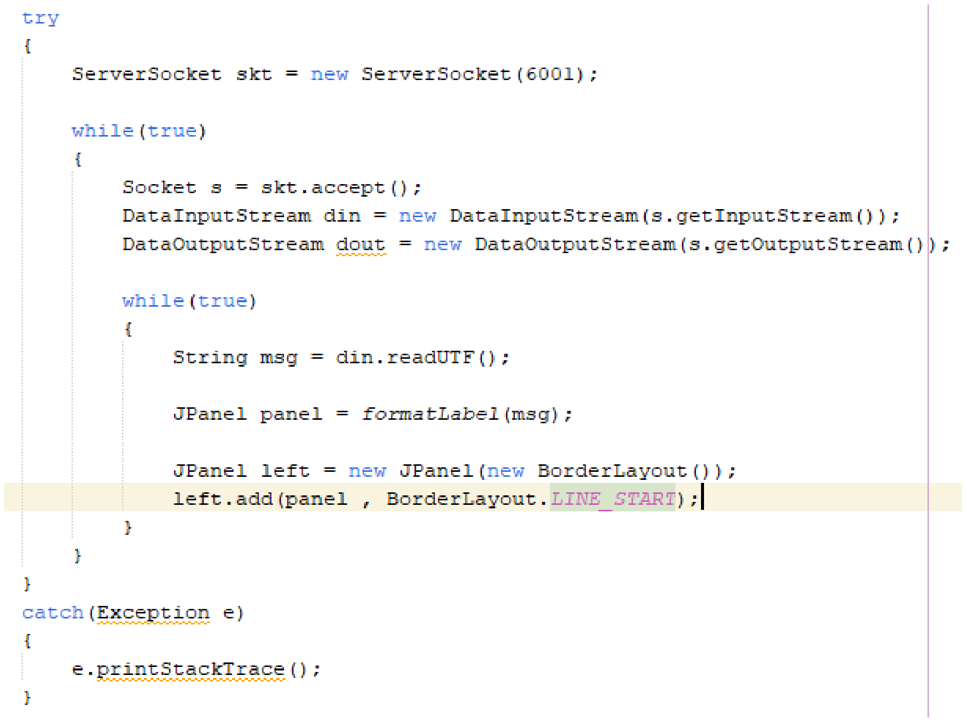
Create dummy panel for the received msg and call formatLabel() for it.



```
try
{
    ServerSocket skt = new ServerSocket(6001);

    while(true)
    {
        Socket s = skt.accept();
        DataInputStream din = new DataInputStream(s.getInputStream());
        DataOutputStream dout = new DataOutputStream(s.getOutputStream());

        while(true)
        {
            String msg = din.readUTF();

            JPanel panel = formatLabel(msg);


        }
    }
}
catch(Exception e)
{
    e.printStackTrace();
}
```

Step-69

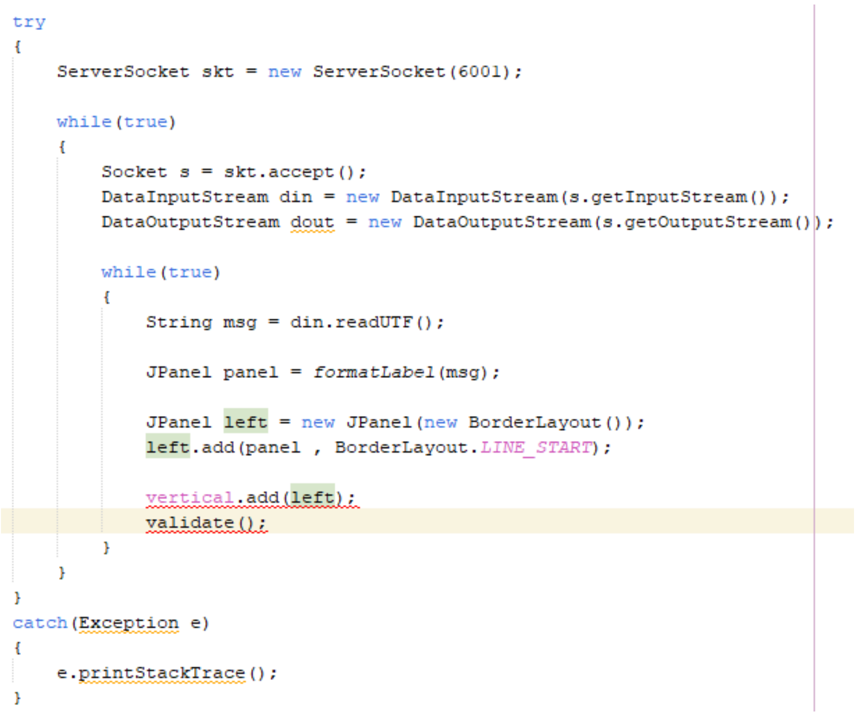
Create JPanel left for received msg along with border layout.



```
try
{
    ServerSocket skt = new ServerSocket(6001);

    while(true)
    {
        Socket s = skt.accept();
        DataInputStream din = new DataInputStream(s.getInputStream());
        DataOutputStream dout = new DataOutputStream(s.getOutputStream());

        while(true)
        {
            String msg = din.readUTF();

            JPanel panel = formatLabel(msg);

            JPanel left = new JPanel(new BorderLayout());
            left.add(panel , BorderLayout.LINE_START);
        }
    }
}
catch(Exception e)
{
    e.printStackTrace();
}
```

Step-70

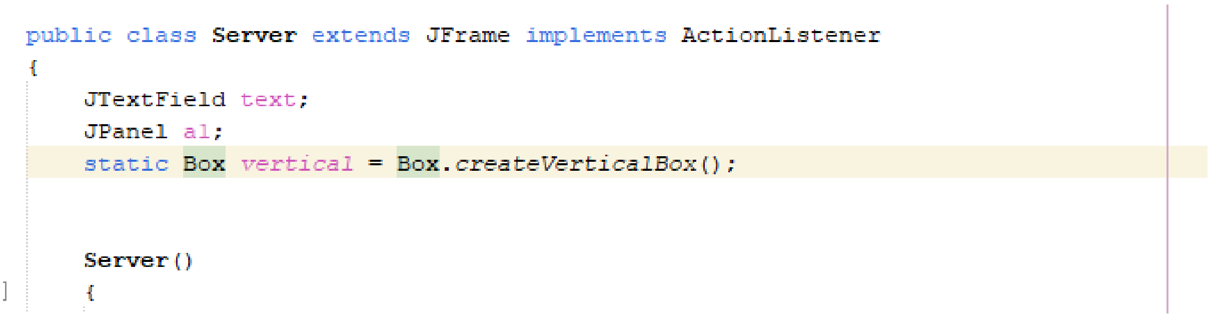
Add panel to vertical box and refresh the frame.



```
try
{
    ServerSocket skt = new ServerSocket(6001);

    while(true)
    {
        Socket s = skt.accept();
        DataInputStream din = new DataInputStream(s.getInputStream());
        DataOutputStream dout = new DataOutputStream(s.getOutputStream());

        while(true)
        {
            String msg = din.readUTF();

            JPanel panel = formatLabel(msg);

            JPanel left = new JPanel(new BorderLayout());
            left.add(panel , BorderLayout.LINE_START);

            vertical.add(left);
            validate();
        }
    }
}
catch(Exception e)
{
    e.printStackTrace();
}
```

Step-71

Declare vertical box to static to solve error.

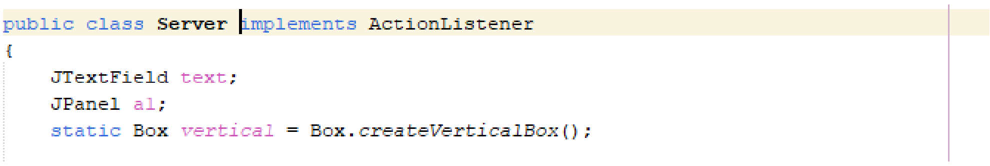


```
public class Server extends JFrame implements ActionListener
{
    JTextField text;
    JPanel a1;
    static Box vertical = Box.createVerticalBox();


    Server()
    {
```

Step-72

Now to solve the validate error, the complete frame must be static and then only the validate error can be solved.

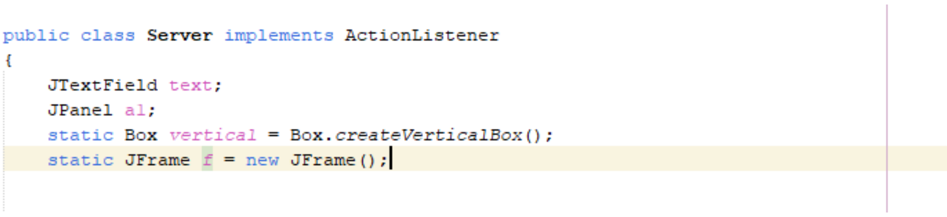
So, firstly remove extends JFrame.



```
public class Server implements ActionListener
{
    JTextField text;
    JPanel a1;
    static Box vertical = Box.createVerticalBox();
```

Step-73

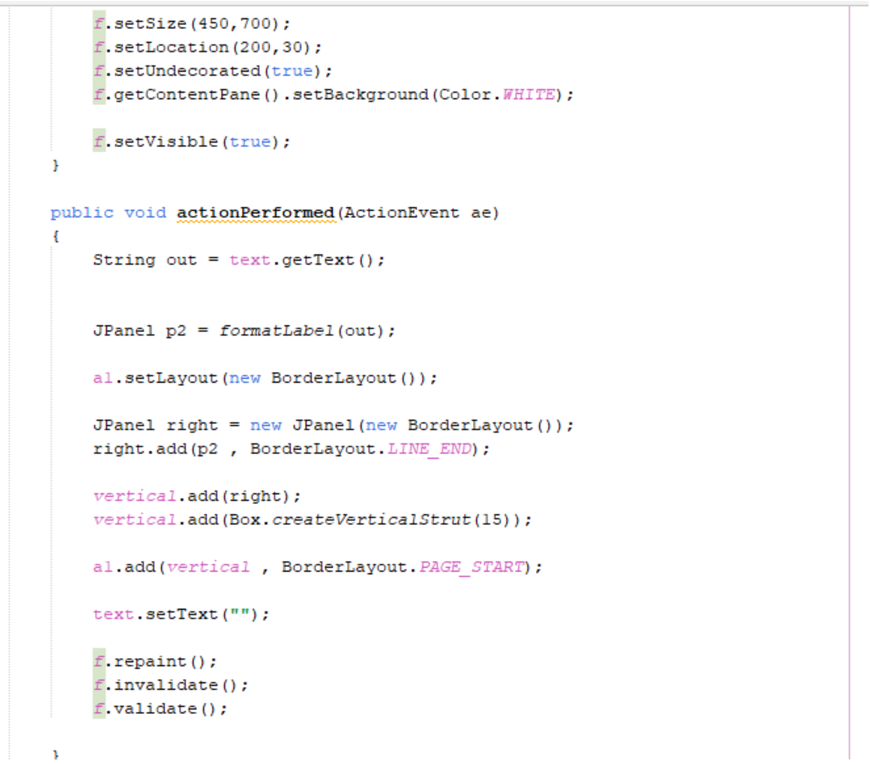
Create a static object of JFrame class and declare it globally.



```
public class Server implements ActionListener
{
    JTextField text;
    JPanel a1;
    static Box vertical = Box.createVerticalBox();
    static JFrame f = new JFrame();
```

Step-74

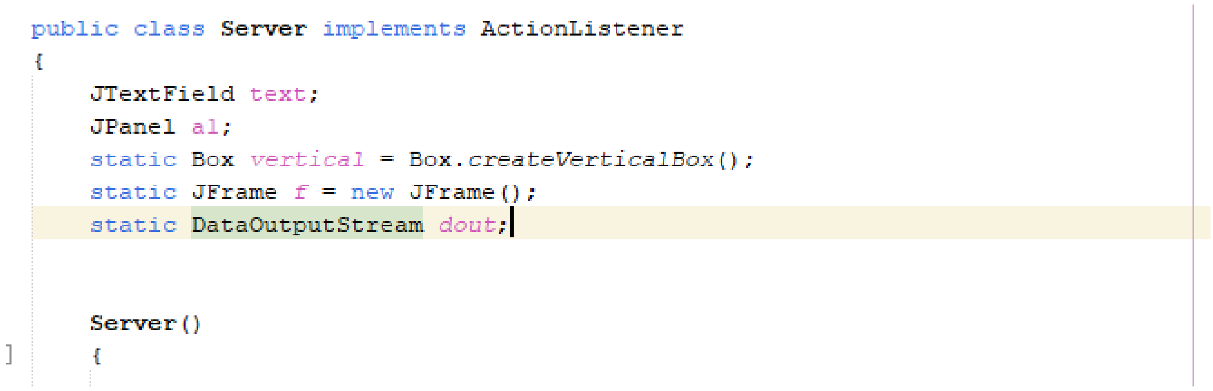
Now solve all the errors by calling the functions using object of JFrame class.



```
        f.setSize(450,700);
        f.setLocation(200,30);
        f.setUndecorated(true);
        f.getContentPane().setBackground(Color.WHITE);

        f.setVisible(true);
    }

    public void actionPerformed(ActionEvent ae)
    {
        String out = text.getText();


        JPanel p2 = formatLabel(out);

        a1.setLayout(new BorderLayout());

        JPanel right = new JPanel(new BorderLayout());
        right.add(p2 , BorderLayout.LINE_END);

        vertical.add(right);
        vertical.add(Box.createVerticalStrut(15));

        a1.add(vertical , BorderLayout.PAGE_START);

        text.setText("");

        f.repaint();
        f.invalidate();
        f.validate();

    }
```

Step-75

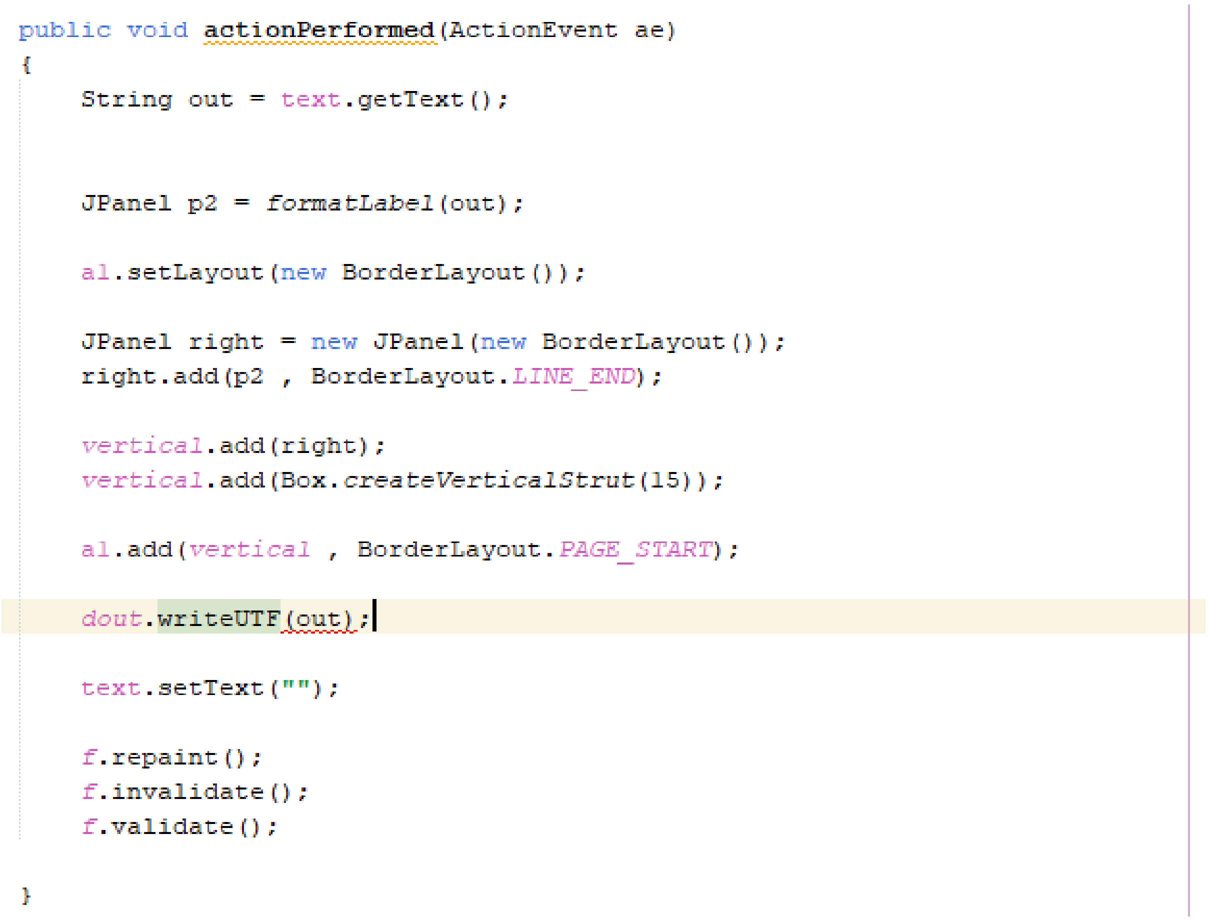
Declare Dout globally.



```
public class Server implements ActionListener
{
    JTextField text;
    JPanel a1;
    static Box vertical = Box.createVerticalBox();
    static JFrame f = new JFrame();
    static DataOutputStream dout;


    Server()
    {
```

Step-76

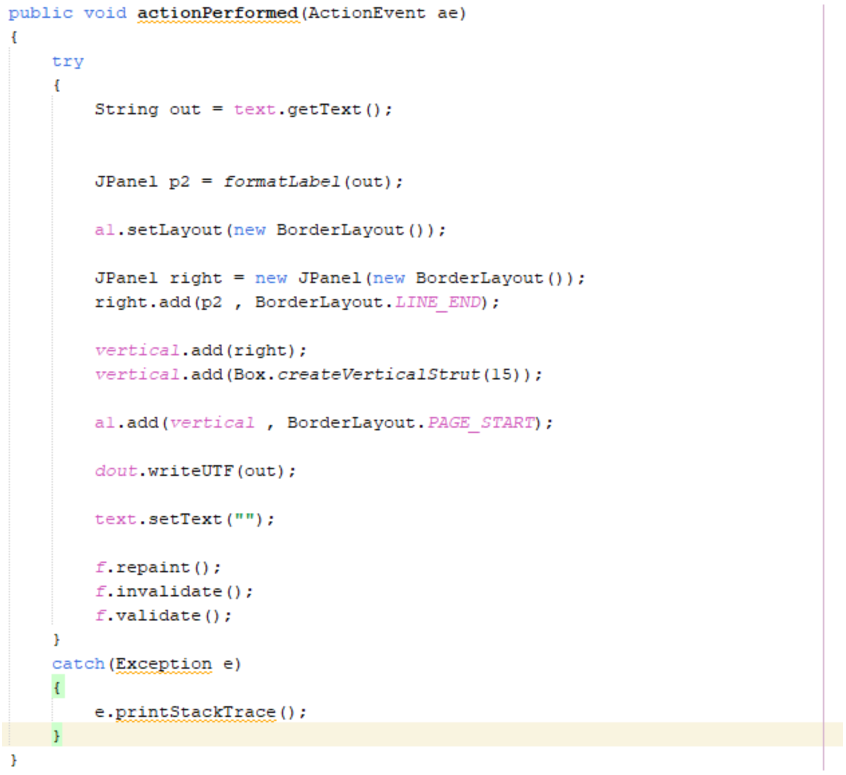
We also need to send msg to the client. So, make use of writeUTF() and this should happen after clicking send button and hence should be written inside actionPerformed() method.



```
public void actionPerformed(ActionEvent ae)
{
    String out = text.getText();


    JPanel p2 = formatLabel(out);

    a1.setLayout(new BorderLayout());

    JPanel right = new JPanel(new BorderLayout());
    right.add(p2 , BorderLayout.LINE_END);

    vertical.add(right);
    vertical.add(Box.createVerticalStrut(15));

    a1.add(vertical , BorderLayout.PAGE_START);

    dout.writeUTF(out);

    text.setText("");

    f.repaint();
    f.invalidate();
    f.validate();

}
```

Step-77

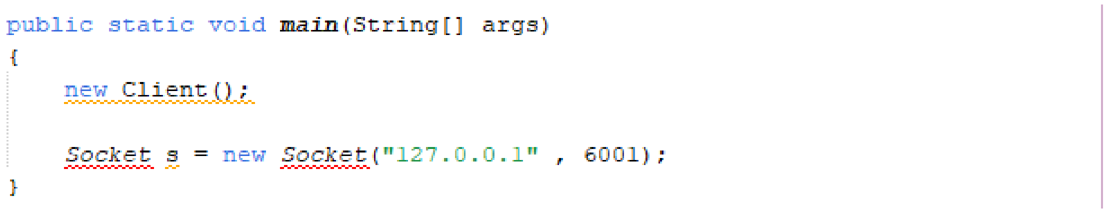
Now put the complete code of actionPerformed() method inside try-catch block. To solve the error.



```
public void actionPerformed(ActionEvent ae)
{
    try
    {
        String out = text.getText();


        JPanel p2 = formatLabel(out);

        a1.setLayout(new BorderLayout());

        JPanel right = new JPanel(new BorderLayout());
        right.add(p2 , BorderLayout.LINE_END);

        vertical.add(right);
        vertical.add(Box.createVerticalStrut(15));

        a1.add(vertical , BorderLayout.PAGE_START);

        dout.writeUTF(out);

        text.setText("");

        f.repaint();
        f.invalidate();
        f.validate();
    }
    catch(Exception e)
    {
        e.printStackTrace();
    }
}
```

Step-78

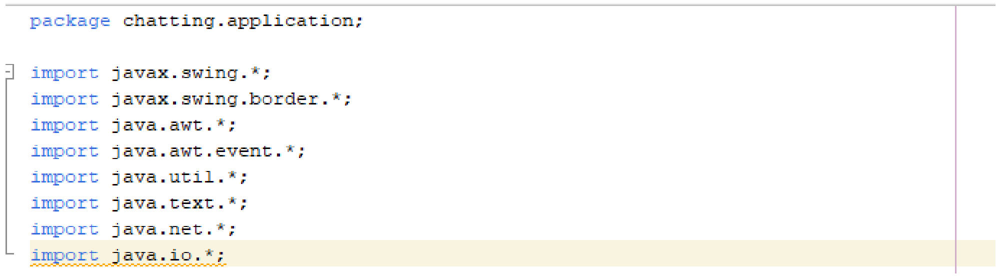
Create a socket in Client class.



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

    Socket s = new Socket("127.0.0.1" , 6001);
}
```

Step-79

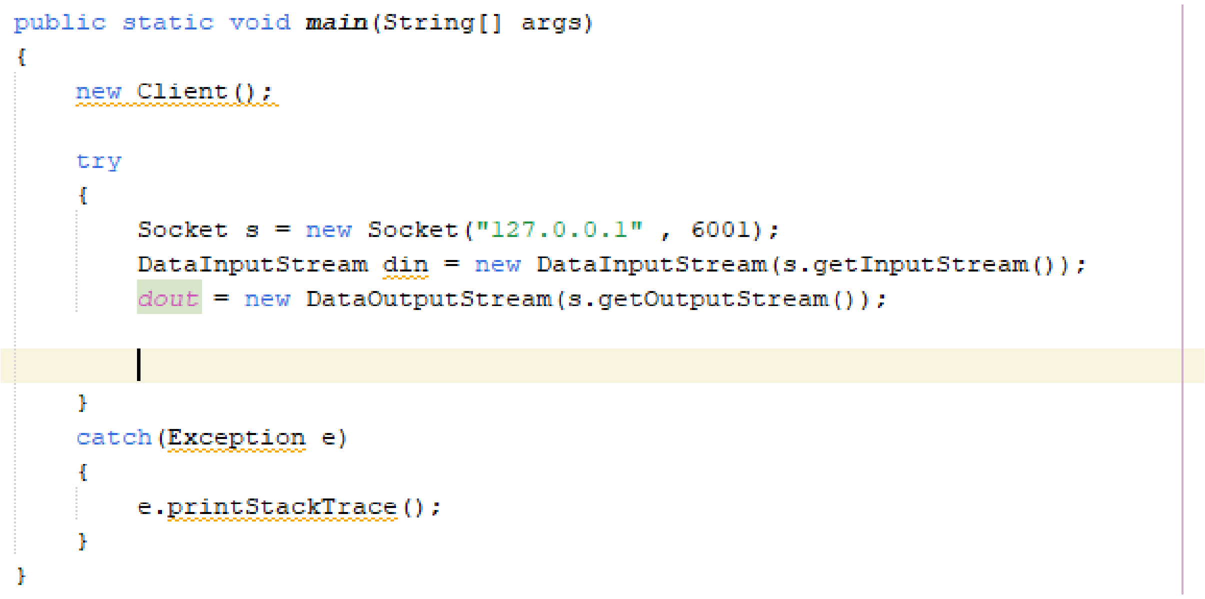
Import Libraries to solve error.



```
package chatting.application;

import javax.swing.*;
import javax.swing.border.*;
import java.awt.*;
import java.awt.event.*;
import java.util.*;
import java.text.*;
import java.net.*;
import java.io.*;
```

Step-80

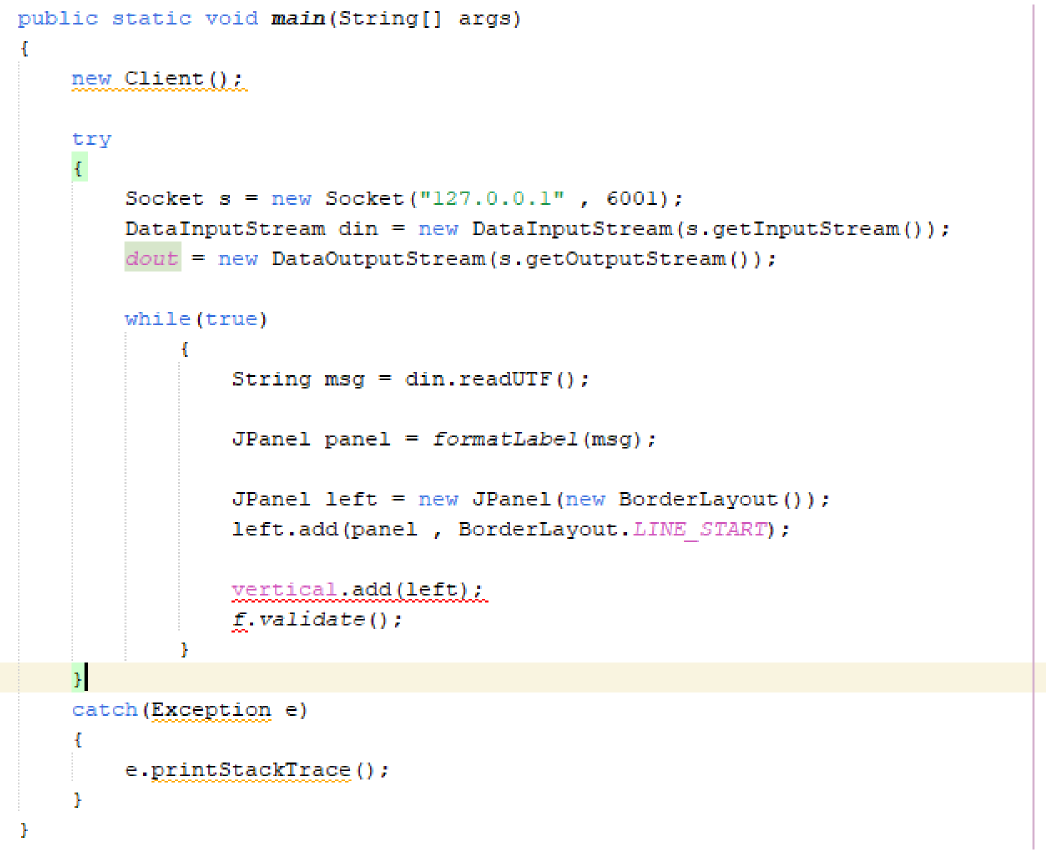
Copy din and dout from server to client and declare dout globally.



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

    try
    {
        Socket s = new Socket("127.0.0.1" , 6001);
        DataInputStream din = new DataInputStream(s.getInputStream());
        dout = new DataOutputStream(s.getOutputStream());

        
    }
    catch(Exception e)
    {
        e.printStackTrace();
    }
}
```

Step-81

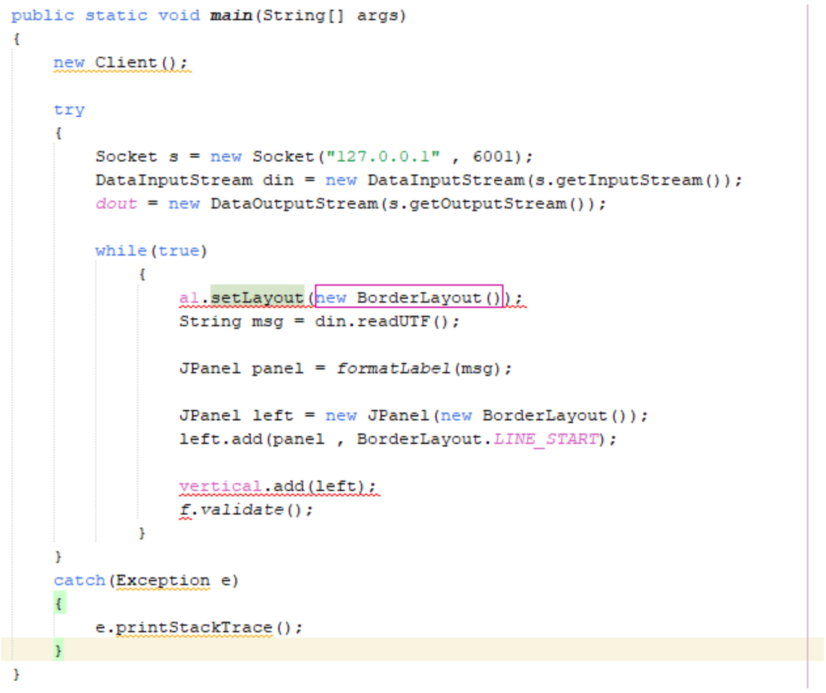
Copy readUTF() while true code from server to client.



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

    try
    {
        Socket s = new Socket("127.0.0.1" , 6001);
        DataInputStream din = new DataInputStream(s.getInputStream());
        dout = new DataOutputStream(s.getOutputStream());

        while(true)
            {
                String msg = din.readUTF();

                JPanel panel = formatLabel(msg);

                JPanel left = new JPanel(new BorderLayout());
                left.add(panel , BorderLayout.LINE_START);

                vertical.add(left);
                f.validate();
            }
    }
    catch(Exception e)
    {
        e.printStackTrace();
    }
}
```

Step-82

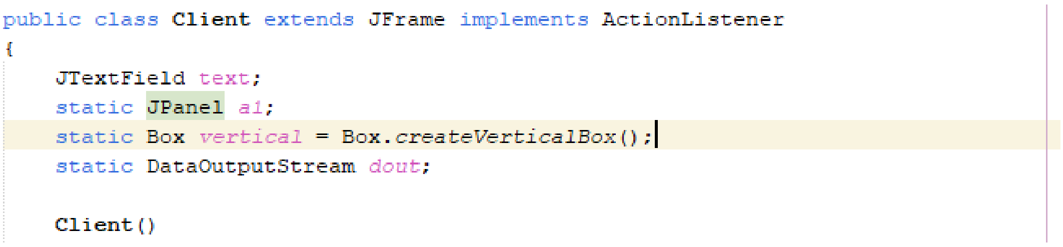
Now we need to place received msg from the server on to the left of client. So, we need to set layout for a1.



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

    try
    {
        Socket s = new Socket("127.0.0.1" , 6001);
        DataInputStream din = new DataInputStream(s.getInputStream());
        dout = new DataOutputStream(s.getOutputStream());

        while(true)
        {
            a1.setLayout(new BorderLayout());
            String msg = din.readUTF();

            JPanel panel = formatLabel(msg);

            JPanel left = new JPanel(new BorderLayout());
            left.add(panel , BorderLayout.LINE_START);

            vertical.add(left);
            f.validate();
        }
    }
    catch(Exception e)
    {
        e.printStackTrace();
    }
}
```

Step-83

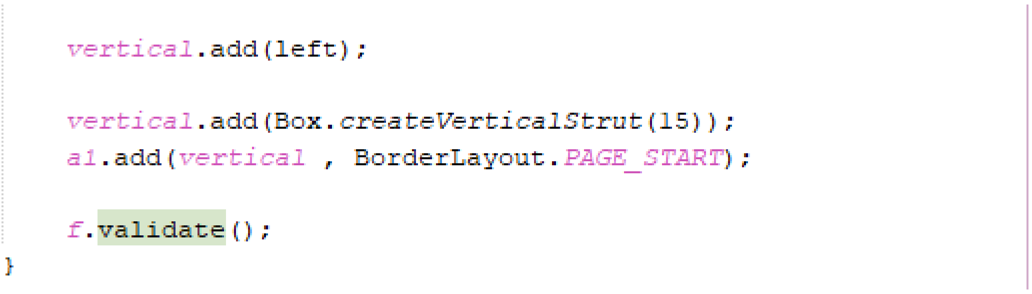
Change declaration of Box vertical and panel a1 to static.



```
public class Client extends JFrame implements ActionListener
{
    JTextField text;
    static JPanel a1;
    static Box vertical = Box.createVerticalBox();
    static DataOutputStream dout;

    Client()
```

Step-84

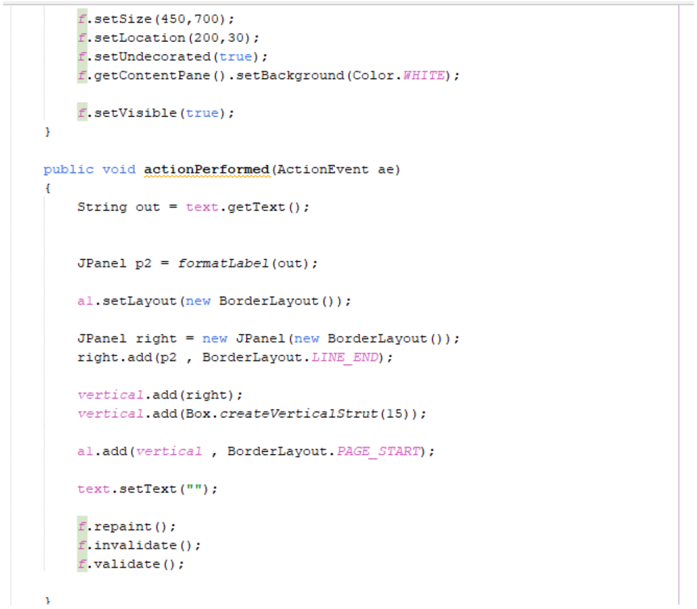
Create vertical strut and add to panel a1.



```
    vertical.add(left);

    vertical.add(Box.createVerticalStrut(15));
    a1.add(vertical , BorderLayout.PAGE_START);

    f.validate();
}
```

Step-85

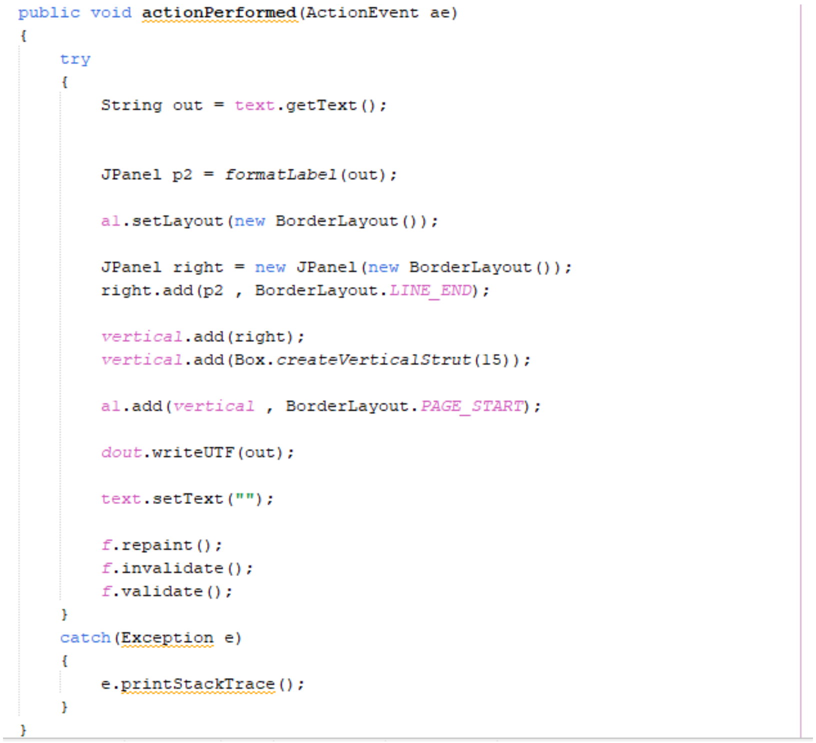
Declare static JFrame globally and solve the errors.



```
        f.setSize(450,700);
        f.setLocation(200,30);
        f.setUndecorated(true);
        f.getContentPane().setBackground(Color.WHITE);

        f.setVisible(true);
    }

    public void actionPerformed(ActionEvent ae)
    {
        String out = text.getText();


        JPanel p2 = formatLabel(out);

        a1.setLayout(new BorderLayout());

        JPanel right = new JPanel(new BorderLayout());
        right.add(p2 , BorderLayout.LINE_END);

        vertical.add(right);
        vertical.add(Box.createVerticalStrut(15));

        a1.add(vertical , BorderLayout.PAGE_START);

        text.setText("");

        f.repaint();
        f.invalidate();
        f.validate();

    }
```

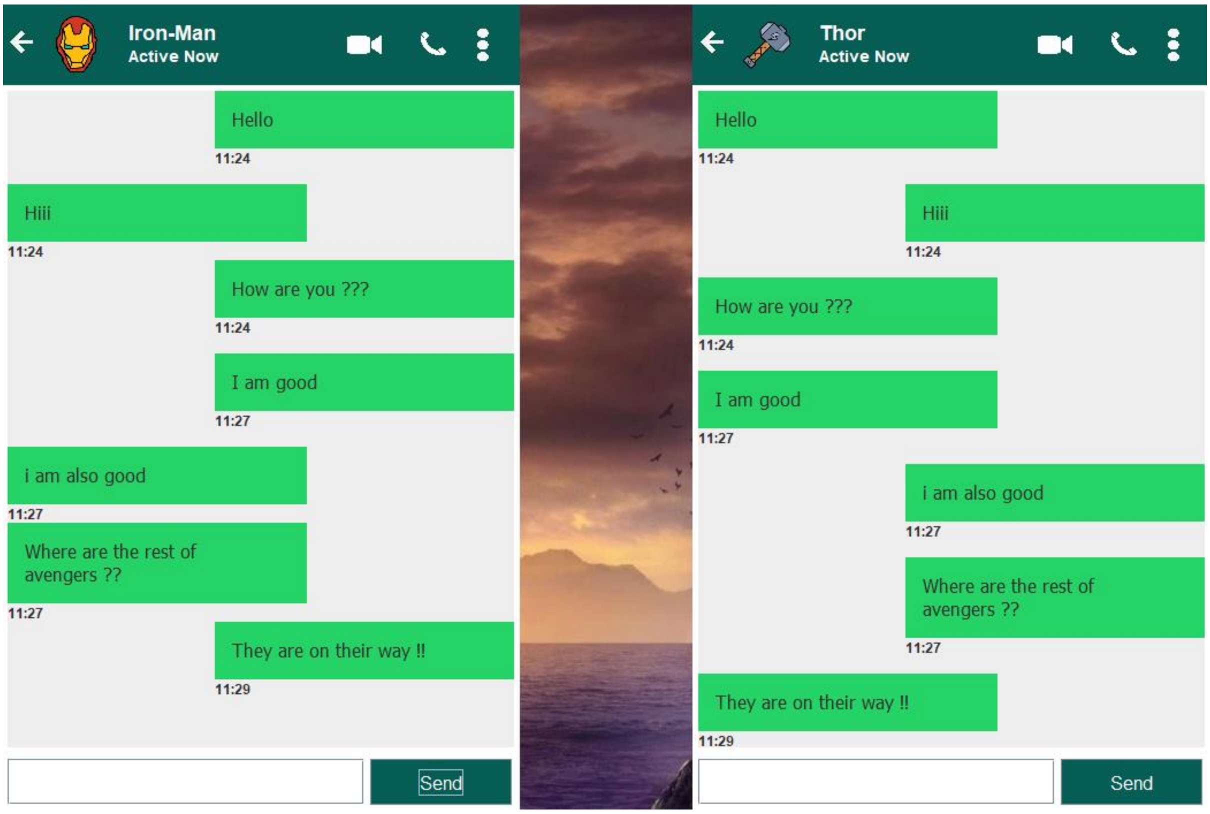
Step-86

We also need to send msg to the client. So, make use of writeUTF() and this should happen after clicking send button and hence should be written inside actionPerformed() method and put a try-catch block.



```
public void actionPerformed(ActionEvent ae)
{
    try
    {
        String out = text.getText();


        JPanel p2 = formatLabel(out);

        a1.setLayout(new BorderLayout());

        JPanel right = new JPanel(new BorderLayout());
        right.add(p2 , BorderLayout.LINE_END);

        vertical.add(right);
        vertical.add(Box.createVerticalStrut(15));

        a1.add(vertical , BorderLayout.PAGE_START);

        dout.writeUTF(out);

        text.setText("");

        f.repaint();
        f.invalidate();
        f.validate();
    }
    catch(Exception e)
    {
        e.printStackTrace();
    }
}
```

Final Output



Hello
11:24

Hiii
11:24

How are you ???
11:24

I am good
11:27

i am also good
11:27

Where are the rest of avengers ??
11:27

They are on their way !!
11:29

Hello
11:24

Hiii
11:24

How are you ???
11:24

I am good
11:27

i am also good
11:27

Where are the rest of avengers ??
11:27

They are on their way !!
11:29

GitHub Project Link

Click Here

THANK YOU !!!