# **Java AWT Button**

# The button class is used to create a labeled button that has platform independent implementation. The application result in some action when the button is pushed.

1. **public** **class** Button **extends** Component **implements** Accessible
2. **import** java.awt.\*;
3. **public** **class** ButtonExample {
4. **public** **static** **void** main(String[] args) {
5. Frame f=**new** Frame("Button Example");
6. Button b=**new** Button("Click Here");
7. b.setBounds(50,100,80,30);
8. f.add(b);
9. f.setSize(400,400);
10. f.setLayout(**null**);
11. f.setVisible(**true**);
12. }
13. }

## **Java AWT Button Example with ActionListener**

1. **import** java.awt.\*;
2. **import** java.awt.event.\*;
3. **public** **class** ButtonExample {
4. **public** **static** **void** main(String[] args) {
5. Frame f=**new** Frame("Button Example");
6. **final** TextField tf=**new** TextField();
7. tf.setBounds(50,50, 150,20);
8. Button b=**new** Button("Click Here");
9. b.setBounds(50,100,60,30);
10. b.addActionListener(**new** ActionListener(){
11. **public** **void** actionPerformed(ActionEvent e){
12. tf.setText("Welcome to Javatpoint.");
13. }
14. });
15. f.add(b);f.add(tf);
16. f.setSize(400,400);
17. f.setLayout(**null**);
18. f.setVisible(**true**);
19. }
20. }

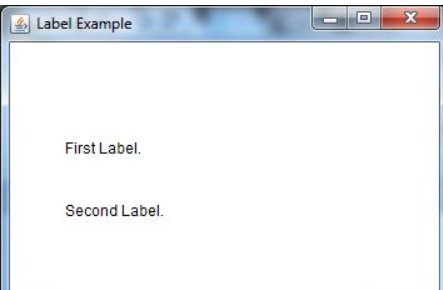
# **Java AWT Label**

The [object](https://www.javatpoint.com/object-and-class-in-java) of Label class is a component for placing text in a container. It is used to display a single line of read only text. The text can be changed by an application but a user cannot edit it directly.

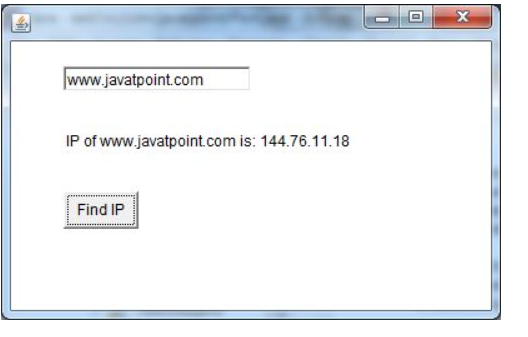
## **AWT Label Class Declaration**

1. **public** **class** Label **extends** Component **implements** Accessible

## **Java Label Example**

1. **import** java.awt.\*;
2. **class** LabelExample{
3. **public** **static** **void** main(String args[]){
4. Frame f= **new** Frame("Label Example");
5. Label l1,l2;
6. l1=**new** Label("First Label.");
7. l1.setBounds(50,100, 100,30);
8. l2=**new** Label("Second Label.");
9. l2.setBounds(50,150, 100,30);
10. f.add(l1); f.add(l2);
11. f.setSize(400,400);
12. f.setLayout(**null**);
13. f.setVisible(**true**);
14. }
15. }  

## **Java AWT Label Example with ActionListener**

1. **import** java.awt.\*;
2. **import** java.awt.event.\*;
3. **public** **class** LabelExample **extends** Frame **implements** ActionListener{
4. TextField tf; Label l; Button b;
5. LabelExample(){
6. tf=**new** TextField();
7. tf.setBounds(50,50, 150,20);
8. l=**new** Label();
9. l.setBounds(50,100, 250,20);
10. b=**new** Button("Find IP");
11. b.setBounds(50,150,60,30);
12. b.addActionListener(**this**);
13. add(b);add(tf);add(l);
14. setSize(400,400);
15. setLayout(**null**);
16. setVisible(**true**);
17. }
18. **public** **void** actionPerformed(ActionEvent e) {
19. **try**{
20. String host=tf.getText();
21. String ip=java.net.InetAddress.getByName(host).getHostAddress();
22. l.setText("IP of "+host+" is: "+ip);
23. }**catch**(Exception ex){System.out.println(ex);}
24. }
25. **public** **static** **void** main(String[] args) {
26. **new** LabelExample();
27.     }
28. }

# **Java AWT TextField**

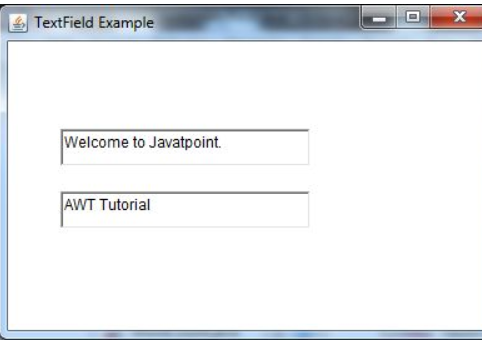
The [object](https://www.javatpoint.com/object-and-class-in-java) of a TextField class is a text component that allows the editing of a single line text. It inherits TextComponent class.

## **AWT TextField Class Declaration**

1. **public** **class** TextField **extends** TextComponent

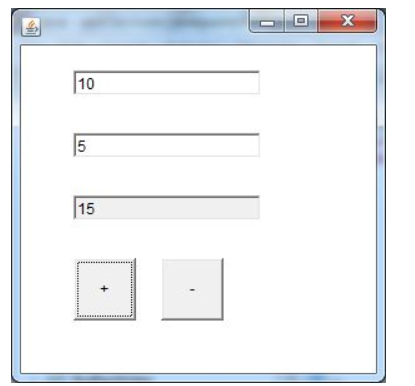
## **Java AWT TextField Example**

1. **import** java.awt.\*;
2. **class** TextFieldExample{
3. **public** **static** **void** main(String args[]){
4. Frame f= **new** Frame("TextField Example");
5. TextField t1,t2;
6. t1=**new** TextField("Welcome to Javatpoint.");
7. t1.setBounds(50,100, 200,30);
8. t2=**new** TextField("AWT Tutorial");
9. t2.setBounds(50,150, 200,30);
10. f.add(t1); f.add(t2);
11. f.setSize(400,400);
12. f.setLayout(**null**);
13. f.setVisible(**true**);
14. }
15. }



## **Java AWT TextField Example with ActionListener**

1. **import** java.awt.\*;
2. **import** java.awt.event.\*;
3. **public** **class** TextFieldExample **extends** Frame **implements** ActionListener{
4. TextField tf1,tf2,tf3;
5. Button b1,b2;
6. TextFieldExample(){
7. tf1=**new** TextField();
8. tf1.setBounds(50,50,150,20);
9. tf2=**new** TextField();
10. tf2.setBounds(50,100,150,20);
11. tf3=**new** TextField();
12. tf3.setBounds(50,150,150,20);
13. tf3.setEditable(**false**);
14. b1=**new** Button("+");
15. b1.setBounds(50,200,50,50);
16. b2=**new** Button("-");
17. b2.setBounds(120,200,50,50);
18. b1.addActionListener(**this**);
19. b2.addActionListener(**this**);
20. add(tf1);add(tf2);add(tf3);add(b1);add(b2);
21. setSize(300,300);
22. setLayout(**null**);
23. setVisible(**true**);
24. }
25. **public** **void** actionPerformed(ActionEvent e) {
26. String s1=tf1.getText();
27. String s2=tf2.getText();
28. **int** a=Integer.parseInt(s1);
29. **int** b=Integer.parseInt(s2);
30. **int** c=0;
31. **if**(e.getSource()==b1){
32. c=a+b;
33. }**else** **if**(e.getSource()==b2){
34. c=a-b;
35. }
36. String result=String.valueOf(c);
37. tf3.setText(result);
38. }
39. **public** **static** **void** main(String[] args) {
40. **new** TextFieldExample();
41. }
42. }



# **Java AWT TextArea**

The [object](https://www.javatpoint.com/object-and-class-in-java) of a TextArea class is a multi line region that displays text. It allows the editing of multiple line text. It inherits TextComponent class.

## **AWT TextArea Class Declaration**

1. **public** **class** TextArea **extends** TextComponent

## **Java AWT TextArea Example**

