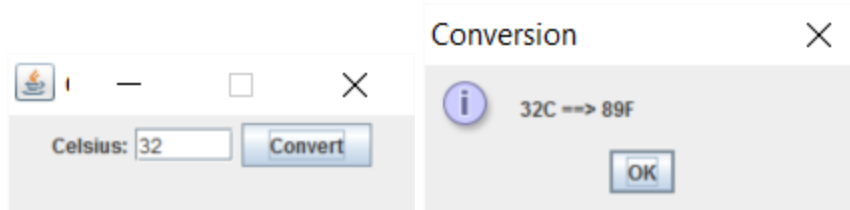# ActionListener Interface

In class activity

1. Implementation

a.

b. Implement a small app that converts Celsius to Fahrenheit. The GUI is shown above.

c. The temperature conversion formula is: F = C * 1.8 + 32

d. Design constraints:

    i. You must use an anonymous class to handle events.

    ii. The conversion is displayed using a modal dialog box.

e. Paste your source code below:

```java
package Week5;

import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.text.DecimalFormat;

/**
 * Created by marquettenj on 1/15/2016.
 */
public class CelsiusConverter extends JFrame {

    private final int WIDTH = 300;
    private final int HEIGHT = 125;
    private JLabel celsius;
    private JTextField input;
    private JButton convert;


    public static void main(String[] args) {
        CelsiusConverter demo = new CelsiusConverter();
    }


    public CelsiusConverter(){
        setTitle("Celsius converter");
        setSize(WIDTH, HEIGHT);
        setLayout(new FlowLayout());
        setDefaultCloseOperation(WindowConstants.EXIT_ON_CLOSE);
        celsius = new JLabel("Celsius: ");
        add(celsius);
        input = new JTextField("32.0", 5);
        add(input);
        convert = new JButton("convert");
        convert.addActionListener(new ActionListener() {
            @Override
            public void actionPerformed(ActionEvent e) {
                double numberInput = Double.parseDouble(input.getText());
                double fahrenheit = (numberInput * 1.8) + 32;
                JOptionPane.showMessageDialog(null, (int)numberInput + "C ==> " + (int)fahrenheit + "F");
            }
        });
        add(convert);
        setVisible(true);
    }
}
```

2. Discussion Questions

a. What is the role of an event listener? What is one example from your app? What is another example?

    An object that does nothing until a specific action occurs in an interface. Our event was a click of the JButton named "Convert." Another example of an event is a user-defined keystroke.

b. If you wanted to validate that the degrees entered really was a number before converting it when the focus was lost on the text field, how would you do so?

    We would add a focus listener and override the focuslost() method to check the validity of the input using Double.parseDouble(String), and re prompt the user if it doesn't find a double or an int.. The focusgained() method would do nothing.

c. What is another way to handle the convert event and why would that be better? In what way could that be worse?

    With the use of a separate inner class. This would be better because we could clean up the code so that there isn't so much going on in the single method, we could just call the inner class, "buttonHandler," for example.
    This could potentially be worse because this GUI only has one event source that doesn't need to be repeated, and creating the inner class is meant to allow your code to be more reusable, at the cost of more space taken in your code.