# **Doing it the AWT Way**

## **Introduction**

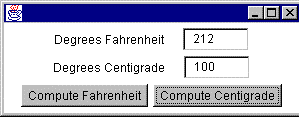
Hang onto your socks.

"AWT" stands for Abstract Windowing Toolkit. Here is a native AWT implementation of the temperature conversion application. It is quite lengthy and involves a number of classes. There are actually several different ways that this program could have been written, but they are all comparable in length and complexity. You can, and probably should, skip reading this listing, but if you cannot resist, go ahead.

Remember, this listing is included here only for the sake of comparison; however, if you are ever going to be a full fledged Java programmer, you will eventually need to master the AWT.

## **The interface**

Here is a snapshot of the interface. Although it looks about the same as the interface shown before, it is actually a little less sophisticated because it uses a flow layout rather than an a grid bag layout. As a consequence, if the user resizes the window, the components in the window will not necessarily remain in their current relative positions. However, if you are unfamiliar with the AWT these remarks do not make much sense, so onward.



## **The code**

**File ConvervsionWithAWT.java**

import java.awt.\*;

public class ConversionWithAWT extends Frame {

private Label fahrenheitLabel;

private TextField fahrenheitField;

private Label centigradeLabel;

private TextField centigradeField;

private Button fahrenheitButton;

private Button centigradeButton;

public ConversionWithAWT(){

fahrenheitLabel = new Label ("Degrees Fahrenheit");

fahrenheitField = new TextField ("212", 6); // 6 columns wide

centigradeLabel = new Label ("Degrees Centigrade");

centigradeField = new TextField ("100", 6); // 6 columns wide

fahrenheitButton = new Button ("Compute Fahrenheit");

centigradeButton = new Button ("Compute Centigrade");

FlowLayout layout = new FlowLayout();

setLayout (layout);

add (fahrenheitLabel);

add (fahrenheitField);

add (centigradeLabel);

add (centigradeField);

add (fahrenheitButton);

add (centigradeButton);

fahrenheitButton.addActionListener

(new FahrenheitButtonListener (this));

centigradeButton.addActionListener

(new CentigradeButtonListener (this));

fahrenheitField.addKeyListener (new DigitKeyListener());

centigradeField.addKeyListener (new DigitKeyListener());

addWindowListener (new GenericWindowListener());

}

public void computeFahrenheit(){

String str = centigradeField.getText().trim();

int centigrade = (new Integer (str)).intValue();

int fahrenheit = centigrade \* 9 / 5 + 32;

fahrenheitField.setText ("" + fahrenheit);

}

public void computeCentigrade(){

String str = fahrenheitField.getText().trim();

int fahrenheit = (new Integer (str)).intValue();

int centigrade = (fahrenheit - 32) \* 5 / 9;

centigradeField.setText ("" + centigrade);

}

public static void main (String[] args){

Frame frm = new ConversionWithAWT();

frm.setSize (250, 150);

frm.setVisible (true);

}

}

**File CentigradeButtonListener.java**

import java.awt.event.\*;

public class CentigradeButtonListener implements ActionListener{

private ConversionWithAWT theGUI;

public CentigradeButtonListener (ConversionWithAWT gui){

theGUI = gui;

}

public void actionPerformed (ActionEvent e){

theGUI.computeCentigrade();

}

}

**File FahrenheitButtonListener.java**

import java.awt.event.\*;

public class FahrenheitButtonListener implements ActionListener{

private ConversionWithAWT theGUI;

public FahrenheitButtonListener (ConversionWithAWT gui){

theGUI = gui;

}

public void actionPerformed (ActionEvent e){

theGUI.computeFahrenheit();

}

}

**File DigitKeyListener.java**

import java.awt.event.\*;

public class DigitKeyListener extends KeyAdapter{

public void keyTyped(KeyEvent e){

char ch = e.getKeyChar();

if (! Character.isDigit(ch))

e.consume();

}

}

**File GenericWindowListener**

import java.awt.event.\*;

public class GenericWindowListener implements WindowListener{

public void windowClosing (WindowEvent e){

System.exit(0);

}

public void windowActivated (WindowEvent e){}

public void windowClosed (WindowEvent e){}

public void windowDeactivated (WindowEvent e){}

public void windowDeiconified (WindowEvent e){}

public void windowIconified (WindowEvent e){}

public void windowOpened (WindowEvent e){}

}

That's it.