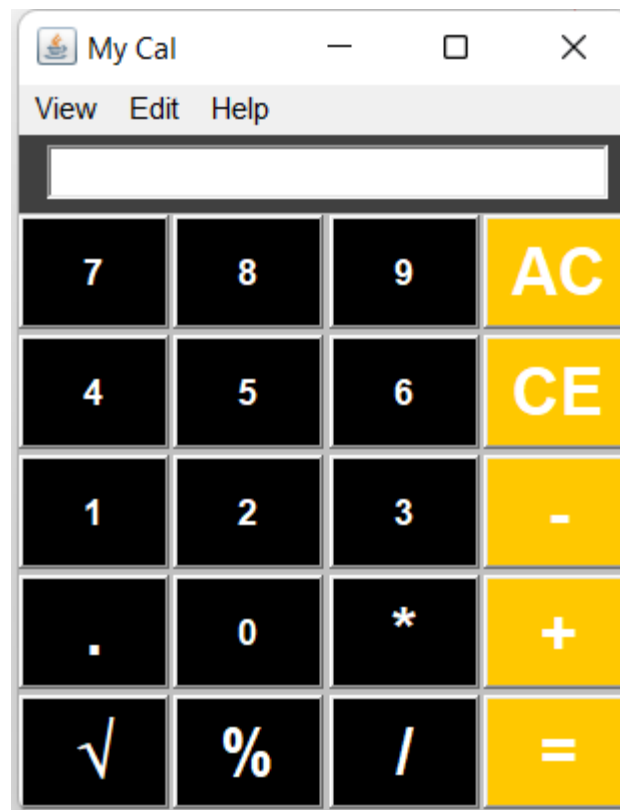




Java Institute for Advanced Technology
SOFTWARE APPLICATION DEVELOPMENT
H7E1 04
H7E1 04/AS/01

VERANGI KAUSHIKA HITIBANDARA
200170001405
GAMPAHA





```
package cal;
```

```
import java.awt.BorderLayout;
```

```
import java.awt.Button;
```

```
import java.awt.Color;
```

```
import java.awt.Font;
```

```
import java.awt.Frame;
```

```
import java.awt.GridLayout;
```

```
import java.awt.Menu;
```

```
import java.awt.MenuBar;
```

```
import java.awt.MenuItem;
```

```
import java.awt.Panel;
```

```
import java.awt.TextField;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.awt.event.WindowAdapter;

import java.awt.event.WindowEvent;
```

```
/**
```

```
*
```

```
* @author Verangi
```

```
*/
```

```
class callisner extends WindowAdapter {
```

```
    @Override
```

```
    public void windowClosing(WindowEvent e) {
```

```
        System.exit(0);
```

```
    }
```

```
}
```

```
public class cal implements ActionListener{
```

```
    Button b0, b1, b2, b3, b4, b5, b6, b7, b8, b9, b11, b12, b13, b14, b15, b16, b17, b18, b19, b20;
```

```
    TextField tf;
```

```
    String fv, sv, op;
```

```
    double fdv, sdv, tot;
```

```
    int f, l;
```

```
cal() {  
  
    Frame f1 = new Frame();  
  
    f1.setBackground(Color.DARK_GRAY);  
  
    f1.addWindowListener(new callisner());  
  
    f1.setBounds(500, 250, 260, 330);  
  
    f1.setTitle("My Cal");  
  
  
    f1.setVisible(true);  
  
  
    MenuBar mbar = new MenuBar();  
  
    MenuItem ma1 = new MenuItem("Standard");  
  
    MenuItem ma2 = new MenuItem("Scientific");  
  
    MenuItem ma3 = new MenuItem("Copy");  
  
    MenuItem ma4 = new MenuItem("View Help F1");  
  
    MenuItem ma5 = new MenuItem("About Calculator");  
  
  
    Menu m1 = new Menu("View");  
  
    m1.add(ma1);  
  
    m1.add(ma2);  
  
    Menu m2 = new Menu("Edit");  
  
    m2.add(ma3);  
  
    Menu m3 = new Menu("Help");  
  
    m3.add(ma4);  
  
    m3.add(ma5);  
}
```

```
mbar.add(m1);
```

```
mbar.add(m2);
```

```
mbar.add(m3);
```

```
f1.setMenuBar(mbar);
```

```
Panel p1 = new Panel();
```

```
Panel p2 = new Panel();
```

```
GridLayout g1 = new GridLayout(5, 4, 2, 2);
```

```
b0 = new Button("0");
```

```
b1 = new Button("1");
```

```
b2 = new Button("2");
```

```
b3 = new Button("3");
```

```
b4 = new Button("4");
```

```
b5 = new Button("5");
```

```
b6 = new Button("6");
```

```
b7 = new Button("7");
```

```
b8 = new Button("8");
```

```
b9 = new Button("9");
```

```
b11 = new Button("+");  
b12 = new Button("-");  
b13 = new Button("/");  
b14 = new Button("*");  
b15 = new Button("CE");  
b16 = new Button("%");  
b17 = new Button(".");  
b18 = new Button("=");  
b19 = new Button("AC");  
b20 = new Button("v");
```

```
b0.setBackground(Color.BLACK);  
b1.setBackground(Color.BLACK);  
b2.setBackground(Color.BLACK);  
b3.setBackground(Color.BLACK);  
b4.setBackground(Color.BLACK);  
b5.setBackground(Color.BLACK);  
b6.setBackground(Color.BLACK);  
b7.setBackground(Color.BLACK);  
b8.setBackground(Color.BLACK);  
b9.setBackground(Color.BLACK);
```

```
b11.setBackground(Color.ORANGE);  
b12.setBackground(Color.ORANGE);  
b13.setBackground(Color.BLACK);  
b14.setBackground(Color.BLACK);  
b15.setBackground(Color.ORANGE);  
b16.setBackground(Color.BLACK);  
b17.setBackground(Color.BLACK);  
b18.setBackground(Color.ORANGE);  
b19.setBackground(Color.ORANGE);  
b20.setBackground(Color.BLACK);
```

```
b0.setForeground(Color.WHITE);  
b1.setForeground(Color.WHITE);  
b2.setForeground(Color.WHITE);  
b3.setForeground(Color.WHITE);  
b4.setForeground(Color.WHITE);  
b5.setForeground(Color.WHITE);  
b6.setForeground(Color.WHITE);  
b7.setForeground(Color.WHITE);  
b8.setForeground(Color.WHITE);  
b9.setForeground(Color.WHITE);  
b11.setForeground(Color.WHITE);  
b12.setForeground(Color.WHITE);
```

```
b13.setForeground(Color.WHITE);  
b14.setForeground(Color.WHITE);  
b15.setForeground(Color.WHITE);  
b16.setForeground(Color.WHITE);  
b17.setForeground(Color.WHITE);  
b18.setForeground(Color.WHITE);  
b19.setForeground(Color.WHITE);  
b20.setForeground(Color.WHITE);
```

```
Font fon1 = new Font("Calibri", Font.BOLD, 14);  
Font fon2 = new Font("Calibri", Font.BOLD, 26);
```

```
b1.setFont(fon1);  
b2.setFont(fon1);  
b3.setFont(fon1);  
b4.setFont(fon1);  
b5.setFont(fon1);  
b6.setFont(fon1);
```



```
b7.setFont(fon1);
```

```
b8.setFont(fon1);
```

```
b9.setFont(fon1);
```

```
b0.setFont(fon1);
```

```
b11.setFont(fon2);
```

```
b12.setFont(fon2);
```

```
b13.setFont(fon2);
```

```
b14.setFont(fon2);
```

```
b15.setFont(fon2);
```

```
b16.setFont(fon2);
```

```
b17.setFont(fon2);
```

```
b18.setFont(fon2);
```

```
b19.setFont(fon2);
```

```
b20.setFont(fon2);
```

```
tf = new TextField(25);
```

```
p2.add(b7);
```

```
p2.add(b8);
```

```
p2.add(b9);
```

```
p2.add(b19);
```

```
p2.add(b4);
```

```
p2.add(b5);  
p2.add(b6);  
p2.add(b15);  
p2.add(b1);  
p2.add(b2);  
p2.add(b3);  
p2.add(b12);  
p2.add(b17);  
p2.add(b0);  
p2.add(b14);  
p2.add(b11);  
p2.add(b20);  
p2.add(b16);  
p2.add(b13);  
p2.add(b18);
```

```
p1.add(tf);
```

```
f1.add(p1, BorderLayout.NORTH);  
f1.add(p2, BorderLayout.CENTER);  
p2.setLayout(g1);  
p2.setBackground(Color.LIGHT_GRAY);
```

```
b1.addActionListener(this);
```

```
b2.addActionListener(this);  
b3.addActionListener(this);  
b4.addActionListener(this);  
b5.addActionListener(this);  
b6.addActionListener(this);  
b7.addActionListener(this);  
b8.addActionListener(this);  
b9.addActionListener(this);  
b0.addActionListener(this);  
b11.addActionListener(this);  
b12.addActionListener(this);  
b13.addActionListener(this);  
b14.addActionListener(this);  
b15.addActionListener(this);  
b16.addActionListener(this);  
b17.addActionListener(this);  
b18.addActionListener(this);  
b19.addActionListener(this);  
b20.addActionListener(this);  
  
}
```

```
@Override
```

```
public void actionPerformed(ActionEvent e) {  
  
    Object o = e.getSource();
```

```
if (o.equals(b0)) {  
    tf.setText(tf.getText() + b0.getLabel());  
} else if (o.equals(b1)) {  
    tf.setText(tf.getText() + b1.getLabel());  
} else if (o.equals(b2)) {  
    tf.setText(tf.getText() + b2.getLabel());  
} else if (o.equals(b3)) {  
    tf.setText(tf.getText() + b3.getLabel());  
} else if (o.equals(b4)) {  
    tf.setText(tf.getText() + b4.getLabel());  
} else if (o.equals(b5)) {  
    tf.setText(tf.getText() + b5.getLabel());  
} else if (o.equals(b6)) {  
    tf.setText(tf.getText() + b6.getLabel());  
} else if (o.equals(b7)) {  
    tf.setText(tf.getText() + b7.getLabel());  
} else if (o.equals(b8)) {  
    tf.setText(tf.getText() + b8.getLabel());  
} else if (o.equals(b9)) {  
    tf.setText(tf.getText() + b9.getLabel());  
} else if (o.equals(b17)) {  
    tf.setText(tf.getText() + b17.getLabel());  
}
```

```
} else if (o.equals(b11)) {
```

```
    fv = tf.getText();
```

```
    tf.setText("");
```

```
    op = b11.getLabel();
```

```
} else if (o.equals(b12)) {
```

```
    fv = tf.getText();
```

```
    tf.setText("");
```

```
    op = b12.getLabel();
```

```
} else if (o.equals(b13)) {
```

```
    fv = tf.getText();
```

```
    tf.setText("");
```

```
    op = b13.getLabel();
```

```
} else if (o.equals(b15)) {
```

```
    fv = tf.getText();
```

```
    tf.setText("");
```

```
    op = b15.getLabel();
```

```
} else if (o.equals(b14)) {
```

```
    fv = tf.getText();
```

```
    tf.setText("");
```

```
    op = b14.getLabel();
```

```

} else if (o.equals(b16)) {

    fdv = Double.parseDouble(tf.getText());

    tf.setText(Double.toString(fdv / 100));


} else if (o.equals(b19)) {

    int l = tf.getText().length();

    int f = l - 1;


    if (l > 0) {

        StringBuilder back = new StringBuilder(tf.getText());

        back.deleteCharAt(f);

        tf.setText(back.toString());

    } else {

        tf.setText("");

    }


} else if (o.equals(b20)) {

    fdv = Double.parseDouble(tf.getText());

    fdv = Math.sqrt(fdv);

    tf.setText(String.valueOf(fdv));


} else if (o.equals(b18)) {

    sv = tf.getText();

    fdv = Double.parseDouble(fv);

    sdv = Double.parseDouble(sv);

```

```
    if (op.equals("+")) {  
        tot = fdv + sdv;  
        tf.setText(tot + "");  
  
    } else if (op.equals("-")) {  
        tot = fdv - sdv;  
        tf.setText(tot + "");  
  
    } else if (op.equals("/")) {  
        tot = fdv / sdv;  
        tf.setText(tot + "");  
  
    } else if (op.equals("*")) {  
        tot = fdv * sdv;  
        tf.setText(tot + "");  
  
    } else if (op.equals("%")) {  
        tot = fdv % sdv;  
        tf.setText(tot + "");  
  
    }  
}  
  
}
```

```
}
```

```
class CreateCal {
```

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

```
        new cal();
```

```
    }
```

```
}
```