```
import java.awt.*;

public class calc extends Frame{
        Button[] b=new Button[25];
        Pan tf=new Pan (1);
        Font f = new Font("TimesRoman", Font.BOLD, 14);
        double n = 0;
        int op;
        boolean flag = false;
        Toolkit tool;

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

public calc(){
    setTitle("Calculator");
    tool=getToolkit();
    setBackground(new Color(38, 104, 165));
    setForeground(new Color(255,255,255));
    setResizable(false);
    setIconImage(tool.getImage(GetResources("ico.gif")));
    resize(350,400);
    setLayout (null);
    add(tf);
    tf.setBounds (50,50,240,25);
    tf.setFont(f);
    tf.setForeground (new Color(0,0,0));
    tf.setBackground (new Color(150,150,255));
    tf.setText("0");
    for (int i=0;i<25;i++){
        String s=""+i;
        if (i>=10)
        switch (i) {
              case 10: s="+";break;
              case 11: s="-";break;
              case 12: s="*";break;
              case 13: s="/";break;
              case 14: s="=";break;
              case 15: s="C";break;
              case 16: s="sqrt";break;
              case 17: s="%";break;
              case 18: s="sin";break;
              case 19: s="cos";break;
              case 20: s="tan";break;
              case 21: s="Exp";break;
              case 22: s="Log";break;
              case 23: s="";break;
              case 24: s="";break;

        }
        b[i]=new Button (s);
        b[i].setFont(new Font ("TimesRoman",1,20));
        b[i].setForeground (new Color(0,0,0));
        b[i].setBackground (new Color(255,255,222));
        add(b[i]) ;

        if (i<5)
                b[i].setBounds(50+50*i,100,40,40);
        else if (i<10)
                b[i].setBounds(50+50*(i-5),150,40,40);
        else if (i<15)
                b[i].setBounds(50+50*(i-10),200,40,40);
        else if (i<20)
                b[i].setBounds(50+50*(i-15),250,40,40);
        else
                b[i].setBounds(50+50*(i-20),300,40,40);
    }

    Dimension res=tool.getScreenSize();
    move((int)((res.width-400)/2+100),(int)((res.height-400)/2+100));
    setVisible(true);
}

public java.net.URL GetResources(String s) {return this.getClass().getResource(s);}



public boolean handleEvent(Event e){
  if(e.id==Event.WINDOW_DESTROY) dispose();
  if (e.id==Event.ACTION_EVENT){
                for (int i=0;i<10;i++)
                        if ((e.target).equals(b[i])){
                                String s = tf.getText();
                                if(s.equals("0")) s = "" + i;
                                else if(flag) {s = "" + i; flag=false;}
                                else s += i;
                                tf.setText(s);
                                return true;
                        }
                if ((e.target).equals(b[10])) {
                        n = Double.parseDouble(tf.getText());
                        op=10; flag=true;
                        return true;
```

```
}
if ((e.target).equals(b[11])) {
        n = Double.parseDouble(tf.getText());
        op=11; flag=true;
        return true;
}
if ((e.target).equals(b[12])) {
        n = Double.parseDouble(tf.getText());
        op=12; flag=true;
        return true;
}
if ((e.target).equals(b[13])) {
        n = Double.parseDouble(tf.getText());
        op=13; flag=true;
        return true;
}
if ((e.target).equals(b[14])) {
        switch(op){
                case 10:
                        n += Double.parseDouble(tf.getText());
                        break;
                case 11:
                        n -= Double.parseDouble(tf.getText());
                        break;
                case 12:
                        n *= Double.parseDouble(tf.getText());
                        break;
                case 13:
                        n /= Double.parseDouble(tf.getText());
                        break;
                case 17:
                        n %= Double.parseDouble(tf.getText());
                        break;

        }
        String ss = "" + n;
        if(ss.endsWith(".0")) ss = ss.substring(0, ss.length()-2);
        tf.setText(ss);
        return true;
}
if ((e.target).equals(b[15])) {tf.setText("0");  flag = false; return true;}
if ((e.target).equals(b[16])) {
        double d = Double.parseDouble(tf.getText());
        if(d>=0){
                n = Math.sqrt(d);
                String ss = "" + n;
                if(ss.endsWith(".0")) ss = ss.substring(0, ss.length()-2);
                tf.setText(ss);
        }
        return true;
}
if ((e.target).equals(b[17])) {
        n = Double.parseDouble(tf.getText());
        op=17; flag=true;
        return true;
}
if ((e.target).equals(b[18])) {
        double d = Double.parseDouble(tf.getText());
        n = Math.sin(d);
        String ss = "" + n;
        if(ss.endsWith(".0")) ss = ss.substring(0, ss.length()-2);
        tf.setText(ss);
        return true;
}
if ((e.target).equals(b[19])) {
        double d = Double.parseDouble(tf.getText());
        n = Math.cos(d);
        String ss = "" + n;
        if(ss.endsWith(".0")) ss = ss.substring(0, ss.length()-2);
        tf.setText(ss);
        return true;
}
if ((e.target).equals(b[20])) {
        double d = Double.parseDouble(tf.getText());
        n = Math.tan(d);
        String ss = "" + n;
        if(ss.endsWith(".0")) ss = ss.substring(0, ss.length()-2);
        tf.setText(ss);
        return true;
}
if ((e.target).equals(b[21])) {
        double d = Double.parseDouble(tf.getText());
        n = Math.exp(d);
        String ss = "" + n;
        if(ss.endsWith(".0")) ss = ss.substring(0, ss.length()-2);
        tf.setText(ss);
        return true;
}
if ((e.target).equals(b[22])) {
        double d = Double.parseDouble(tf.getText());
        if(d>0){
                n = Math.log(d);
                String ss = "" + n;
                if(ss.endsWith(".0")) ss = ss.substring(0, ss.length()-2);
                tf.setText(ss);
            }
            return true;
        }
    }
    return false;
}

}

//================================================


class Pan extends Panels {

  public Font f=new Font("Helvetica", 1, 18);
  FontMetrics fm = getFontMetrics(f);
  public String s="";
  private int tip;

   public Pan() {
    super();
   }

   public Pan(int tip) {
    this();
    this.tip = tip;
   }

   public void setText(String s){
    this.s=s;
    repaint();
   }

   public String getText(){
    return s;
   }

   public void paint(Graphics g) {
    super.paint(g);
    g.setFont(f);

    if(tip==1){
        g.setColor(Color.white);
```

```
                  g.drawString(s, size().width-fm.stringWidth(s)-6, 20);
         }else{
                  g.setColor(Color.black);
                  g.drawString(s, 10, 20);
         }
   }


}


//=========================================================


class Panels extends Panel {

   public Image im, im1;

   public Panels(Image im) {this.im = im;}

   public Panels() {}

   public void update(Graphics g) {paint(g);}

   public void paint(Graphics g) {
         super.paint(g);
         Dimension dimension = size();
         im1 = createImage(dimension.width, dimension.height);
         pan(im1.getGraphics());
         g.drawImage(im1, 0, 0, this);
   }

   public void pan(Graphics g) {
         Dimension dimension = size();
         int w = dimension.width;
         int h = dimension.height;
      Color color = getBackground();
      g.setColor(color);
      g.fillRect(0, 0, w, h);
      if(im!=null){
               for(int k = 0; k < w; k += im.getWidth(this))
                           for(int l = 0; l < h; l += im.getHeight(this))
                           g.drawImage(im, k, l, this);
      }
      g.setColor(color.brighter());
      g.drawRect(1, 1, w - 2, h - 2);
```

```
        g.setColor(color.darker());
        g.drawRect(0, 0, w - 2, h - 2);
    }
}
```