|  |  |
| --- | --- |
| **Exp.no:**10 | **Design of scientific calculator** |
| **Date:**12-09-19 |

**Aim:**To design a calculator using event-driven programming paradigm of Java with Decimal and Scientific manipulations.

**Algorithm:**

Step 1: Declare a package calc.

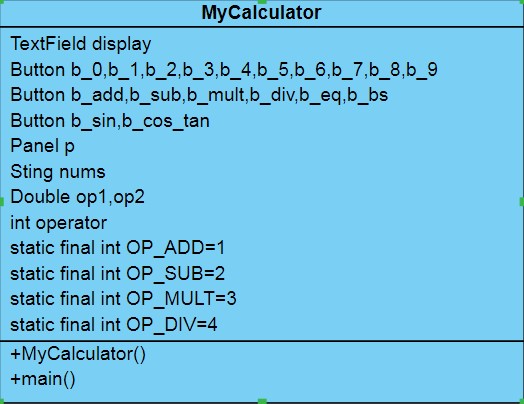
Step 2: Declare the class MyCalculator that extends Frame and implements WindowListener and Action Listener.

Step 3: Add textfield, required buttons, panel, and static members.

Step 4: In the constructor, link WindowListener, ActionListener and Panel to the class and add buttons to the panel.

Step 5: Use ActionListener to perform the required actions.

**Class Diagram:**



**Program:**

/\*\*\*

* created by P.sumanth
* 212217105047
* Eee.b
* \*/

package calculator;

import java.awt.\*; import java.awt.event.ActionEvent; import java.awt.event.ActionListener; import java.awt.event.WindowEvent; import java.awt.event.WindowListener;

public class MyCalculator extends Frame implements

WindowListener,ActionListener{

TextField display;

Button b\_0,b\_1,b\_2,b\_3,b\_4,b\_5,b\_6,b\_7,b\_8,b\_9;

Button b\_add,b\_sub,b\_mul,b\_div,b\_eq,b\_sin,b\_cos,b\_tan;

Panel p;

String nums; Double op1,op2; int operator; static final int OP\_ADD=1; static final int OP\_SUB=2; public MyCalculator()

{ this.addWindowListener(this); this.setLayout(new GridLayout(2,1)); nums="0";

display=new TextField("0"); display.setEditable(false); this.add(display);

p=new Panel();

p.setLayout(new GridLayout(3,2)); this.add(p);

b\_0=new Button("0"); b\_0.addActionListener(this); p.add(b\_0);

b\_1=new Button("1"); b\_1.addActionListener(this); p.add(b\_1);

b\_2=new Button("2"); b\_2.addActionListener(this); p.add(b\_2);

b\_3=new Button("3"); b\_3.addActionListener(this); p.add(b\_3);

b\_4=new Button("4"); b\_4.addActionListener(this); p.add(b\_4);

b\_5=new Button("5"); b\_5.addActionListener(this); p.add(b\_5);

b\_6=new Button("6"); b\_6.addActionListener(this); p.add(b\_6);

b\_7=new Button("7"); b\_7.addActionListener(this); p.add(b\_7);

b\_8=new Button("8"); b\_8.addActionListener(this); p.add(b\_8);

b\_9=new Button("9"); b\_9.addActionListener(this); p.add(b\_9);

b\_add=new Button("+"); b\_add.addActionListener(this); p.add(b\_add);

b\_sub=new Button("-"); b\_sub.addActionListener(this); p.add(b\_sub);

b\_eq=new Button("="); b\_eq.addActionListener(this); p.add(b\_eq);

b\_div=new Button("/"); b\_div.addActionListener(this); p.add(b\_div);

b\_mul=new Button("\*"); b\_mul.addActionListener(this); p.add(b\_mul);

b\_sin=new Button("sin"); b\_sin.addActionListener(this); p.add(b\_sin);

b\_cos=new Button("cos"); b\_cos.addActionListener(this); p.add(b\_cos);

b\_tan=new Button("tan"); b\_tan.addActionListener(this); p.add(b\_tan); }

public static void main(String[] args) { MyCalculator mc;

mc=new MyCalculator(); mc.setSize(300,250); mc.setTitle("calculator"); mc.setVisible(true);

}

@Override

public void windowOpened(WindowEvent e) { // TODO Auto-generated method stub

}

@Override

public void windowClosing(WindowEvent e) { System.exit(0);

}

@Override

public void windowClosed(WindowEvent e) { // TODO Auto-generated method stub

}

@Override

public void windowIconified(WindowEvent e) { // TODO Auto-generated method stub

}

@Override

public void windowDeiconified(WindowEvent e) { // TODO Auto-generated method stub

}

@Override

public void windowActivated(WindowEvent e) { // TODO Auto-generated method stub

}

@Override

public void windowDeactivated(WindowEvent e) { // TODO Auto-generated method stubb

}

@Override

public void actionPerformed(ActionEvent e) { // TODO Auto-generated method stub if(e.getSource()==b\_0)

{ nums=nums+"0";

display.setText(nums);

}else if(e.getSource()==b\_1)

{ nums=nums+"1";

display.setText(nums); }else if(e.getSource()==b\_2)

{ nums=nums+"2";

display.setText(nums); }else if(e.getSource()==b\_3)

{ nums=nums+"3";

display.setText(nums); }else if(e.getSource()==b\_4)

{ nums=nums+"4";

display.setText(nums); }else if(e.getSource()==b\_5)

{ nums=nums+"5";

display.setText(nums); }else if(e.getSource()==b\_6)

{ nums=nums+"6";

display.setText(nums); }else if(e.getSource()==b\_7)

{ nums=nums+"7";

display.setText(nums); }else if(e.getSource()==b\_8)

{ nums=nums+"8";

display.setText(nums); }else if(e.getSource()==b\_9)

{ nums=nums+"9";

display.setText(nums);

}else if(e.getSource()==b\_add)

{ op1=Double.parseDouble(nums); nums="0"; display.setText(nums); operator=OP\_ADD;

}else if(e.getSource()==b\_eq)

{ switch(operator)

{ case OP\_ADD:

op2=Double.parseDouble(nums); nums=""+(op1+op2); display.setText(nums); break;

}

}else if(e.getSource()==b\_sub)

{ op1=Double.parseDouble(nums); nums="0"; display.setText(nums); operator=OP\_SUB;

}else if(e.getSource()==b\_eq)

{ switch(operator) { case OP\_SUB: op2=Double.parseDouble(nums); nums=""+(op1-op2); display.setText(nums); break;

}

}else if(e.getSource()==b\_sin)

{ op1=Double.parseDouble(nums); nums=""+Math.sin(op1\*Math.PI/180); display.setText(nums);

}else if(e.getSource()==b\_cos)

{ op1=Double.parseDouble(nums); nums=""+Math.cos(op1\*Math.PI/180); display.setText(nums);

}else if(e.getSource()==b\_tan)

{ op1=Double.parseDouble(nums); nums=""+Math.tan(op1\*Math.PI/180); display.setText(nums);

}

}

}

Output:



**Result:**

Thus a java console application that uses event-driven programming paradigm of Java to design a calculator with decimal and scientific manipulations is verified.