**Task 3: Calculator**

Code:

import java.awt.\*;

import java.awt.event.\*;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

public class MyCalculator extends Frame

{

public boolean setClear=true;

double number, memValue;

char op;

String digitButtonText[] = {"7", "8", "9", "4", "5", "6", "1", "2", "3", "0", "+/-", "." };

String operatorButtonText[] = {"/", "sqrt", "\*", "%", "-", "1/X", "+", "=" };

String memoryButtonText[] = {"MC", "MR", "MS", "M+" };

String specialButtonText[] = {"Backspc", "C", "CE" };

MyDigitButton digitButton[]=new MyDigitButton[digitButtonText.length];

MyOperatorButton operatorButton[]=new MyOperatorButton[operatorButtonText.length];

MyMemoryButton memoryButton[]=new MyMemoryButton[memoryButtonText.length];

MySpecialButton specialButton[]=new MySpecialButton[specialButtonText.length];

Label displayLabel=new Label("0",Label.RIGHT);

Label memLabel=new Label(" ",Label.RIGHT);

final int FRAME\_WIDTH=325,FRAME\_HEIGHT=325;

final int HEIGHT=30, WIDTH=30, H\_SPACE=10,V\_SPACE=10;

final int TOPX=30, TOPY=50;

///////////////////////////

MyCalculator(String frameText)//constructor

{

super(frameText);

int tempX=TOPX, y=TOPY;

displayLabel.setBounds(tempX,y,240,HEIGHT);

displayLabel.setBackground(Color.BLUE);

displayLabel.setForeground(Color.WHITE);

add(displayLabel);

memLabel.setBounds(TOPX, TOPY+HEIGHT+ V\_SPACE,WIDTH, HEIGHT);

add(memLabel);

// set Co-ordinates for Memory Buttons

tempX=TOPX;

y=TOPY+2\*(HEIGHT+V\_SPACE);

for(int i=0; i<memoryButton.length; i++)

{

memoryButton[i]=new MyMemoryButton(tempX,y,WIDTH,HEIGHT,memoryButtonText[i], this);

memoryButton[i].setForeground(Color.RED);

y+=HEIGHT+V\_SPACE;

}

//set Co-ordinates for Special Buttons

tempX=TOPX+1\*(WIDTH+H\_SPACE); y=TOPY+1\*(HEIGHT+V\_SPACE);

for(int i=0;i<specialButton.length;i++)

{

specialButton[i]=new MySpecialButton(tempX,y,WIDTH\*2,HEIGHT,specialButtonText[i], this);

specialButton[i].setForeground(Color.RED);

tempX=tempX+2\*WIDTH+H\_SPACE;

}

//set Co-ordinates for Digit Buttons

int digitX=TOPX+WIDTH+H\_SPACE;

int digitY=TOPY+2\*(HEIGHT+V\_SPACE);

tempX=digitX; y=digitY;

for(int i=0;i<digitButton.length;i++)

{

digitButton[i]=new MyDigitButton(tempX,y,WIDTH,HEIGHT,digitButtonText[i], this);

digitButton[i].setForeground(Color.BLUE);

tempX+=WIDTH+H\_SPACE;

if((i+1)%3==0){tempX=digitX; y+=HEIGHT+V\_SPACE;}

}

//set Co-ordinates for Operator Buttons

int opsX=digitX+2\*(WIDTH+H\_SPACE)+H\_SPACE;

int opsY=digitY;

tempX=opsX; y=opsY;

for(int i=0;i<operatorButton.length;i++)

{

tempX+=WIDTH+H\_SPACE;

operatorButton[i]=new MyOperatorButton(tempX,y,WIDTH,HEIGHT,operatorButtonText[i], this);

operatorButton[i].setForeground(Color.RED);

if((i+1)%2==0){tempX=opsX; y+=HEIGHT+V\_SPACE;}

}

addWindowListener(new WindowAdapter()

{

public void windowClosing(WindowEvent ev)

{System.exit(0);}

});

setLayout(null);

setSize(FRAME\_WIDTH,FRAME\_HEIGHT);

setVisible(true);

}

//////////////////////////////////

static String getFormattedText(double temp)

{

String resText=""+temp;

if(resText.lastIndexOf(".0")>0)

resText=resText.substring(0,resText.length()-2);

return resText;

}

////////////////////////////////////////

public static void main(String []args)

{

new MyCalculator("Calculator - JavaTpoint");

}

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

class MyDigitButton extends Button implements ActionListener

{

MyCalculator cl;

//////////////////////////////////////////

MyDigitButton(int x,int y, int width,int height,String cap, MyCalculator clc)

{

super(cap);

setBounds(x,y,width,height);

this.cl=clc;

this.cl.add(this);

addActionListener(this);

}

////////////////////////////////////////////////

static boolean isInString(String s, char ch)

{

for(int i=0; i<s.length();i++) if(s.charAt(i)==ch) return true;

return false;

}

/////////////////////////////////////////////////

public void actionPerformed(ActionEvent ev)

{

String tempText=((MyDigitButton)ev.getSource()).getLabel();

if(tempText.equals("."))

{

if(cl.setClear)

{cl.displayLabel.setText("0.");cl.setClear=false;}

else if(!isInString(cl.displayLabel.getText(),'.'))

cl.displayLabel.setText(cl.displayLabel.getText()+".");

return;

}

int index=0;

try{

index=Integer.parseInt(tempText);

}catch(NumberFormatException e){return;}

if (index==0 && cl.displayLabel.getText().equals("0")) return;

if(cl.setClear)

{cl.displayLabel.setText(""+index);cl.setClear=false;}

else

cl.displayLabel.setText(cl.displayLabel.getText()+index);

}//actionPerformed

}//class defination

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

class MyOperatorButton extends Button implements ActionListener

{

MyCalculator cl;

MyOperatorButton(int x,int y, int width,int height,String cap, MyCalculator clc)

{

super(cap);

setBounds(x,y,width,height);

this.cl=clc;

this.cl.add(this);

addActionListener(this);

}

///////////////////////

public void actionPerformed(ActionEvent ev)

{

String opText=((MyOperatorButton)ev.getSource()).getLabel();

cl.setClear=true;

double temp=Double.parseDouble(cl.displayLabel.getText());

if(opText.equals("1/x"))

{

try

{double tempd=1/(double)temp;

cl.displayLabel.setText(MyCalculator.getFormattedText(tempd));}

catch(ArithmeticException excp)

{cl.displayLabel.setText("Divide by 0.");}

return;

}

if(opText.equals("sqrt"))

{

try

{double tempd=Math.sqrt(temp);

cl.displayLabel.setText(MyCalculator.getFormattedText(tempd));}

catch(ArithmeticException excp)

{cl.displayLabel.setText("Divide by 0.");}

return;

}

if(!opText.equals("="))

{

cl.number=temp;

cl.op=opText.charAt(0);

return;

}

// process = button pressed

switch(cl.op)

{

case '+':

temp+=cl.number;break;

case '-':

temp=cl.number-temp;break;

case '\*':

temp\*=cl.number;break;

case '%':

try{temp=cl.number%temp;}

catch(ArithmeticException excp)

{cl.displayLabel.setText("Divide by 0."); return;}

break;

case '/':

try{temp=cl.number/temp;}

catch(ArithmeticException excp)

{cl.displayLabel.setText("Divide by 0."); return;}

break;

}//switch

cl.displayLabel.setText(MyCalculator.getFormattedText(temp));

//cl.number=temp;

}//actionPerformed

}//class

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

class MyMemoryButton extends Button implements ActionListener

{

MyCalculator cl;

/////////////////////////////////

MyMemoryButton(int x,int y, int width,int height,String cap, MyCalculator clc)

{

super(cap);

setBounds(x,y,width,height);

this.cl=clc;

this.cl.add(this);

addActionListener(this);

}

////////////////////////////////////////////////

public void actionPerformed(ActionEvent ev)

{

char memop=((MyMemoryButton)ev.getSource()).getLabel().charAt(1);

cl.setClear=true;

double temp=Double.parseDouble(cl.displayLabel.getText());

switch(memop)

{

case 'C':

cl.memLabel.setText(" ");cl.memValue=0.0;break;

case 'R':

cl.displayLabel.setText(MyCalculator.getFormattedText(cl.memValue));break;

case 'S':

cl.memValue=0.0;

case '+':

cl.memValue+=Double.parseDouble(cl.displayLabel.getText());

if(cl.displayLabel.getText().equals("0") || cl.displayLabel.getText().equals("0.0") )

cl.memLabel.setText(" ");

else

cl.memLabel.setText("M");

break;

}//switch

}//actionPerformed

}//class

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

class MySpecialButton extends Button implements ActionListener

{

MyCalculator cl;

MySpecialButton(int x,int y, int width,int height,String cap, MyCalculator clc)

{

super(cap);

setBounds(x,y,width,height);

this.cl=clc;

this.cl.add(this);

addActionListener(this);

}

//////////////////////

static String backSpace(String s)

{

String Res="";

for(int i=0; i<s.length()-1; i++) Res+=s.charAt(i);

return Res;

}

//////////////////////////////////////////////////////////

public void actionPerformed(ActionEvent ev)

{

String opText=((MySpecialButton)ev.getSource()).getLabel();

//check for backspace button

if(opText.equals("Backspc"))

{

String tempText=backSpace(cl.displayLabel.getText());

if(tempText.equals(""))

cl.displayLabel.setText("0");

else

cl.displayLabel.setText(tempText);

return;

}

//check for "C" button i.e. Reset

if(opText.equals("C"))

{

cl.number=0.0; cl.op=' '; cl.memValue=0.0;

cl.memLabel.setText(" ");

}

//it must be CE button pressed

cl.displayLabel.setText("0");cl.setClear=true;

}//actionPerformed

}//class

Output:

