**GUI - Component Arrays**

Reuse the previous program we created to make a Fancy calculator. We will modify this program to make it even fancier.

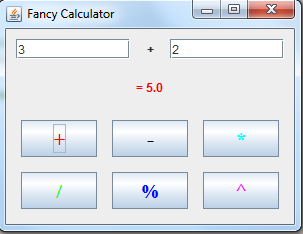
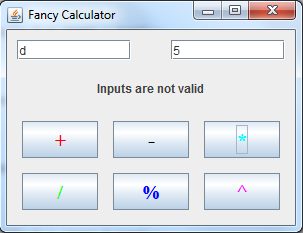
This calculator will accept all number values.

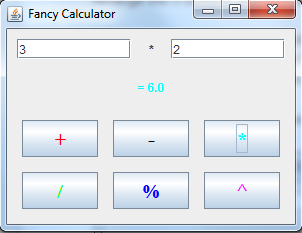
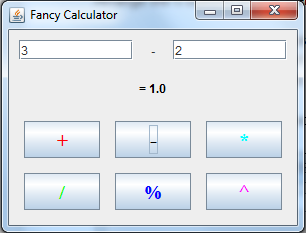
Instead of two buttons, you will create an **array of 6 buttons**. Each button will represent one of the following operators: add, subtract, multiply, divide, remainder, and exponent. Like an actual calculator, label each button with the operator symbol.

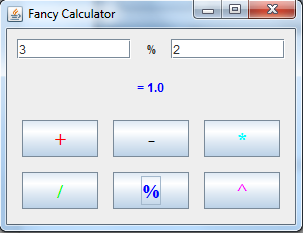
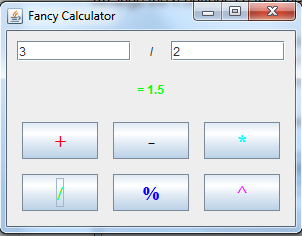
Arrange the 6 buttons so that there is two rows of three buttons each. Make sure to make it look nice with a good layout choice.

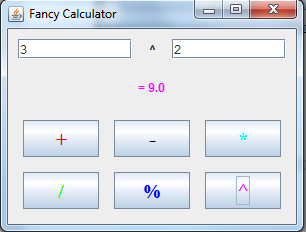
When each button is pressed, just like the original fancy calculator the operator label should change to have an appropriate symbol, and the answer label will show the calculation result in the matching colour. Do not disable any of the buttons this time, they should all be active.

Remember to use loops when working with arrays.









|  |
| --- |
| import javax.swing.event.\*;  import javax.swing.\*;  import java.awt.\*;  import java.awt.event.\*;  public class GUI implements ActionListener  {  JFrame f;  Container container;  JButton[] btn;  JTextField txtA, txtB;  JLabel lblA, lblB;  boolean valid;    public static double numA, numB;  public static String signs = "+-\*/%^", message;    public static Color[] color = {Color.RED, Color.BLACK, Color.MAGENTA, Color.GREEN, Color.BLUE, Color.PINK};    JPanel txtPanel, btnPanel, msgPanel, mainPanel, calcPanel;    public GUI()  {  f = new JFrame("Fancy Calculator");    mainPanel = new JPanel();  mainPanel.setLayout(new BoxLayout(mainPanel, BoxLayout.Y\_AXIS));    calcPanel = new JPanel(new BorderLayout());  calcPanel.setBorder(BorderFactory.createEmptyBorder(10,10,10,10));    txtPanel = new JPanel();    msgPanel = new JPanel(new BorderLayout());  msgPanel.setBorder(BorderFactory.createEmptyBorder(15,15,15,15));    btnPanel = new JPanel(new GridLayout(2, 3, 15, 15));    btn = new JButton[6];    txtA = new JTextField(10);  txtB = new JTextField(10);  lblA = new JLabel(" ", JLabel.CENTER);  lblB = new JLabel("Message", JLabel.CENTER);    //Text Panel  txtPanel.add(txtA);  txtPanel.add(lblA);  txtPanel.add(txtB);    //Message Panel  msgPanel.add(lblB);    //Button Panel with all the buttons added to it  for(int x = 0; x < btn.length; x++)  {  btn[x] = new JButton("" + signs.charAt(x));  btn[x].addActionListener(this);  btn[x].setForeground(color[x]);  btnPanel.add(btn[x]);  }    //Add the Components to the mainPanel  mainPanel.add(txtPanel);  mainPanel.add(msgPanel);  mainPanel.add(btnPanel);    //add all the components to the calcPanel and center it  calcPanel.add(mainPanel);    f.add(calcPanel);    f.pack();  f.setLocationRelativeTo(null);  f.setVisible(true);  }    /\*\*  \* Method Name : checkInput  \* @param JTextField txtA - stores the first number  \* @param JTextField txtB - stores the second number  \* @param JLabel label - message Label output  \* \*\*/  public static boolean checkInput(JTextField txtA, JTextField txtB)  {    //Try to convert the strings of numbers to double  try  {  numA = Double.parseDouble(txtA.getText());  numB = Double.parseDouble(txtB.getText());    return true;    //label.setText("<html> Error 1 <br/> Error 2 </html>");  }    //Invalid Inputs given  catch(NumberFormatException e)  {  message = "Inputs are not valid";    }//end try/catch    return false;    }//end checkInput(JTextField, JTextField)    public void actionPerformed(ActionEvent e)  {  message = "= ";  lblB.setForeground(Color.BLACK);    //Check whether the number are valid or not  valid = checkInput(txtA, txtB);    if(valid)  {  for(int x = 0; x < btn.length; x++)  {  if(e.getSource() == btn[x])  {  lblB.setForeground(color[x]);  lblA.setText(" " + signs.charAt(x) + " ");    //Addition  if(x == 0)  message += (numA + numB);    //Subtraction  else if(x == 1)  message += (numA - numB);    //Multiplication  else if(x == 2)  message += (numA \* numB);    //Quotient  else if(x == 3)  message += (numA / numB);    //Remainder  else if(x == 4)  message += (numA % numB);    //Power  else  message += Math.pow(numA, numB);  }  }  }    //Print the result on the screen  lblB.setText(message);  }    //Main  public static void main(String[] args)  {  new GUI();  }  } |
| Screenshots: |