Working with Listeners and Events

Basic Concepts/Points to remember

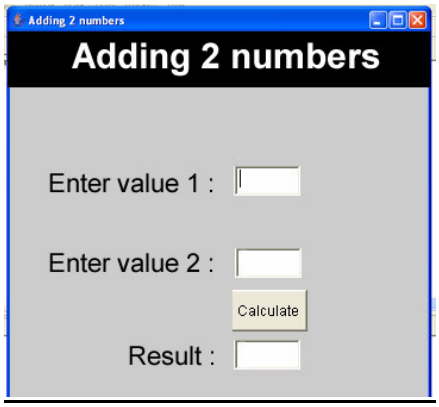
- All code should go in the constructor function of the class (that is the function which has the same name as the class). main( ) will just call this constructor by the use of "new" keyword to create an object of type class.

  note: If your program has a constructor, then it is initiated/called/activated every time you make an object of that class.

- All windows components (Buttons, Labels, CheckBoxes, TextFields etc) need to be declared as private (to the class they are in) and should be outside all functions within that class.

- What this method does is the following:
  - Avoids the need to keep using the object of type class to invoke AWT functions.
  - Gives programmer complete access to all the private variables from anywhere within the class.

JAVA Program to design a simple addition



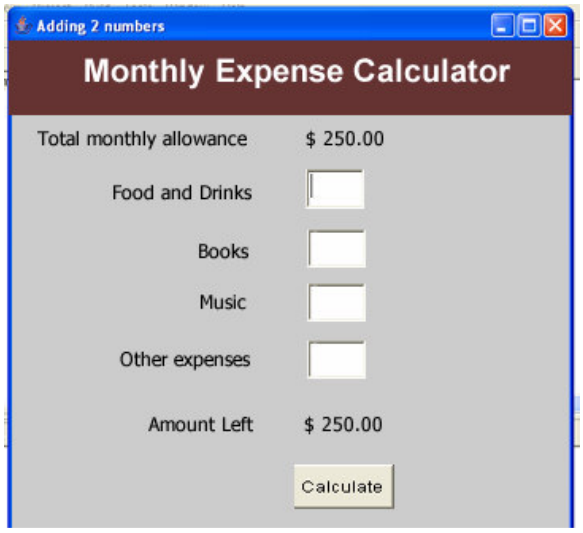
Source Code

```
//program to display a simple addition calculator
//AddCommand.java (c) 2005
//S Krishna

import java.awt.*;
import java.awt.event.*;


public class AddCommand extends Frame implements ActionListener
{

//variables that are private to the class they are declared in.
private TextField tf1;
private TextField tf2;
private TextField result;


//this is the constructor function within which everything is added to
//the frame
   public AddCommand()
   {
      setSize(500,500);
      setTitle("Adding 2 numbers");
      setLayout(null);
      //making sure Java does not implement any default layout style


      tf1 = new TextField("",20);
      tf1.setBounds(250,100,55,25);
      add(tf1);

      tf2 = new TextField("",20);
      tf2.setBounds(200,200,55,25);
```

```
        add(tf2);


        Button addButton = new Button("Calculate");
        addButton.setBounds(270,175,65,35);
        add(addButton);
        addButton.addActionListener(this);


        Button exitButton = new Button("Quit");
        exitButton.setBounds(350,200,65,35);
        add(exitButton);
        exitButton.addActionListener(new ActionListener()
        {
              public void actionPerformed(ActionEvent evt)
              {
                 System.exit(0);
              }
        });

        result = new TextField("",20);
        result.setBounds(300,300,55,25);
        add(result);

        show();
    }

    public static void main(String[] args)
    {
       AddCommand ac = new AddCommand();
    }

    public void actionPerformed(ActionEvent evt)
    {
        int numA,numB;

        numA = (new Integer(tf1.getText()).intValue());
        numB = (new Integer(tf2.getText()).intValue());

        result.setText(String.valueOf(numA+numB));

    }

}
```

Class Activity

Using the code you have, design a program as given below:



Validations/Parameters

- Do not accept negative values in the fields.
- Display an appropriate message when a negative value is entered.
- Keep the “Amount Left” value updated every time “Calculate” is pressed.