The operator on the right hand side needs to be given the method to calculate:

```
public void write(int value) {  
    // write data in the form of a string to the field.  
}
```

We have also used the method `calculate()` which we will define later.

```
class Display extends Panel  
{
    display debated and place on panel.  
    public Display(int value) {  
    }
    public void calculate() {  
    }
}
```

The `Display` class needs to be defined to handle the `write` and `calculate` methods.

Now let's examine our class `Display`. This class needs to display a text field, and the ability to read and write data to and from it.

```
public static void main(String[] args) {  
    initialize objects and place on default panel.  
    class Calculator extends Frame  
    {  
        public void calculate() {  
            // do calculation here  
        }
    }
}
```

The `Calculator` class needs to be defined to handle the `calculate` method.

```
public class Calculator extends Frame  
{  
    public void calculate() {  
        // do calculation here  
    }
}
```

Next, let's look at the responsibilities of each class, and try to work out the logic:

- A class to contain these components: one that will display the `Calculator` and `Display` objects and classes. We will also need the `Main` and the `Main` class.
- The possibility of using these classes to provide the GUI components.

Classes and Objects