Java Programming

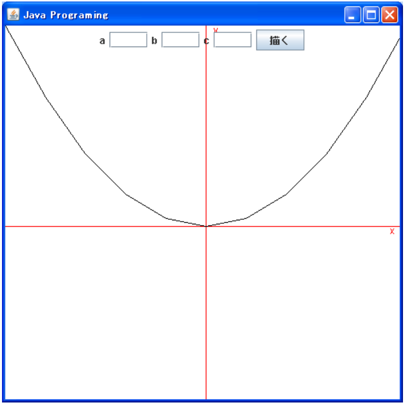
— Graphics: Labels · Texts —

Waseda University

Today's topic

Write a program of GUI

Draw the graph of $y = ax^2 + bx + c$, where a, b, c are constants. These are inputted from the screen.



Swing • AWT

Create a GUI

- To creat a GUI (Graphical User Interface), the following libraries are useful:

  AWT (Abstract Window Toolkit):
  AWT is Java's original windowing, graphics, and user-interface widget toolkit for providing a graphical user interface (GUI).

  Swing:
  Swing is also a GUI widget toolkit for Java. It provides a more sophisticated set of GUI components than AWT.

JTextField

JTextField class

- To create three text fields on the screen, we will use JTextField class, which is a lightweight component that allows the editing of a single line of text.

See also:
`https://docs.oracle.com/javase/7/docs/api/javax/swing/JTextField.html`

JTextField class

- Declare three variables of the JTextField class:
  After the line “public class MyJPanel”,

```
JTextField textField1,textField2, textField3;
```

- Create instances: In the constructor “public MyJPanel(){}”

```
textField1 = new JTextField(“ 1.0 ”,4);
textField2 = new JTextField(“ 0.0 ”,4);
textField3 = new JTextField(“ 0.0 ”,4);
```

Add textFields on the frame

```
add(textField1);
add(textField2);
add(textField3);
```

Then, three text fields appear on the screen.

Label

JLabel class

- Prepare labels that explain the role of each text field
- Declare three variables of the JLabel class:
  After the line “public class MyJPanel”,

```
JLabel label1, label2, label3;
```

- Create instances: In the constructor “public MyJPanel(){}”,

```
label1 = new JLabel("a");
label2 = new JLabel("b");
label3 = new JLabel("c");
```

JLabel class

- Add three labels on the frame.

```
add(label1);
add(label2);
add(label3);
```

Remark: an order of the add methods should be

```
add(label1);
add(textField1);
add(label2);
add(textField2);
add(label3);
add(textField3);
```

Button

JButton class

- Add a button on the frame by using JButton class
- After entering numbers in the text fields, the program starts drawing when the button is pushed.

See also:
`https://docs.oracle.com/javase/jp/6/api/javax/swing/JButton.html`

JButton class

- Declare a variable of the JButton class

```
JButton beginButton;
```

- Create an instance:

```
beginButton = new JButton("Draw");
```

- Add the button on the frame:

```
add(beginButton);
```

ActionListener

What is ActionListener?

- ActionListener is the listener interface for receiving action events. When the action event occurs (e.g, click a button), the actionPerformed method is invoked.
- Declare MyJPanel class and specify that this class implements an ActionListener interface:

```
public class MyJPanel ... implements ActionListener
```

- Register the instance of the draw button by the following statement:

```
beginButton.addActionListener(this);
```

When the user clicks the draw button, the actionPerformed method is invoked (event processing).

Event processing

Event processing

```
public void actionPerformed(ActionEvent e) {
  // write a program that reacts to the action
}
```

- The actionPerformed method is invoked when the registered event handler occurs (the user clicks the draw button).

Event processing

Event processing

```
public void actionPerformed(ActionEvent e) {
  if(e.getSource()==beginButton){
    a=Double.parseDouble(textField1.getText());
    b=Double.parseDouble(textField2.getText());
    c=Double.parseDouble(textField3.getText());
    repaint();
  }
}
```

Remark that we should declare

```
  double a,b,c;
```

in the MyJPanel class.

Exercise

Write a program

Draw the graph of $y = ax^2 + bx + c$, where a, b, c are constants. These are inputted from the screen.

- Edit the paintComponent method

  For example, the part

  ```
  x1=x;
  y1=x1*x1;
  x2=(x+incx);
  y2=x2*x2;
  ```

  when $y = x^2$ should be changed by using coefficients a, b, c.
- Declare the coefficients a, b, c before the constructor of MyJPanel.