

オブジェクト指向プログラミング 第9 回

ボタンとアクションリスナー

担当：高橋、佐藤聖也

1 準備

最初に、第 3 回の講義で取り上げた JFrameSample01.java を思い出しましょう。

JFrameSample01.java (再録)

```
import javax.swing.*;

class JFrameSample01{
    public static void main(String [] args){
        //JFrame クラスのインスタンスを生成
        JFrame fr = new JFrame();
        //フレームサイズの指定 (横, 縦) [ピクセル]
        fr.setSize(500, 400);
        //フレームタイトルの表示
        fr.setTitle("Frame title");
        //×ボタンをおすとプログラムを終了します
        fr.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        //フレームをディスプレイ画面に表示します
        fr.setVisible(true);
    }
}
```

これを、コンストラクタを使って書きなおすと、以下の JFrameSample01b.java のようにできます。

JFrameSample01b.java

```
import javax.swing.*;

class JFrameSample01b extends JFrame{
    JFrameSample01b(){
        // this. は省略することができます。
        //フレームサイズの指定 (横, 縦) [ピクセル]
        this.setSize(500, 400);
        //フレームタイトルの表示
        this.setTitle("Frame title");
        //×ボタンをおすとプログラムを終了します
        this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    }
    public static void main(String [] args){
        //JFrame クラスのインスタンスを生成
        JFrameSample01b fr = new JFrameSample01b();
        //フレームをディスプレイ画面に表示
    }
}
```

```
        fr.setVisible(true);  
    }  
}
```

JFrameSample01b.java を Eclipse で作成するとエラーメッセージが出ます。黄色のアンダーラインがついたクラス名をマウスで選択した後、CTRL+1 を押して「生成シリアル・バージョン ID の追加」を押し、3 行目「import java.io.Serializable;」と 5 行目「implements Serializable」を追加することで、エラーメッセージを取り除くことができます。

JFrameSample01c.java

```
import javax.swing.*;  
import java.io.Serializable;  
  
class JFrameSample01c extends JFrame implements Serializable{  
    private static final long serialVersionUID = 3985606183151864772L;  
  
    JFrameSample01c(){  
        //フレームサイズの指定（横，縦）[ピクセル]  
        this.setSize(500, 400);  
        //フレームタイトルの表示  
        this.setTitle("Frame title");  
        //×ボタンをおすとプログラムを終了します  
        this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);  
    }  
  
    public static void main(String [] args){  
        //JFrame クラスのインスタンスを生成  
        JFrameSample01c fr = new JFrameSample01c();  
        //フレームをディスプレイ画面に表示  
        fr.setVisible(true);  
    }  
}
```

2 ボタンとアクションリスナー

2.1 レイアウトマネージャー：FlowLayout クラス

JButtonSampleFlowLayout.java

```
import javax.swing.*; // JFrame クラスを含む
import java.awt.event.*; // ActionListener インターフェースを含む
import java.awt.*; // Container クラス, FlowLayout クラスを含む

class JButtonSampleFlowLayout extends JFrame implements ActionListener{

    public void actionPerformed(ActionEvent e){
        System.out.println("Button was pushed!");
    }

    JButtonSampleFlowLayout(){
        setSize(500, 400);
        setTitle("Frame title");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        //レイアウトマネージャーによるコンポーネントの配置
        Container contentPane = this.getContentPane();
        contentPane.setLayout(new FlowLayout() );

        JButton btn1 = new JButton("Button-1");
        btn1.addActionListener(this);
        contentPane.add(btn1);

        JButton btn2 = new JButton("Button-2");
        btn2.addActionListener(this);
        contentPane.add(btn2);

        JButton btn3 = new JButton("Button-3");
        btn3.addActionListener(this);
        contentPane.add(btn3);
    }

    public static void main(String [] args){
        //JFrame クラスのインスタンスを生成
        JButtonSampleFlowLayout fr = new JButtonSampleFlowLayout();
        //フレームをディスプレイ画面に表示します
        fr.setVisible(true);
    }
}
```

2.2 レイアウトマネージャー：BorderLayout クラス

JButtonSampleBorderLayout.java

```
import javax.swing.*; //JFrame クラスを含む
import java.awt.event.*; //ActionListener インターフェースを含む
import java.awt.*; //Container クラス, BorderLayout クラスを含む

class JButtonSampleBorderLayout extends JFrame implements ActionListener{
    public void actionPerformed(ActionEvent e){
        System.out.println("Button was pushed!");
    }

    JButtonSampleBorderLayout(){
        setSize(500, 400);
        setTitle("Frame title");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        //レイアウトマネージャーによるコンポーネントの配置
        Container contentPane = this.getContentPane();
        contentPane.setLayout(new BorderLayout() );

        //ボタンの生成
        JButton btn1 = new JButton("Button-NORTH");
        btn1.addActionListener(this);
        contentPane.add(btn1, BorderLayout.NORTH);

        JButton btn2 = new JButton("Button-EAST");
        btn2.addActionListener(this);
        contentPane.add(btn2, BorderLayout.EAST);

        JButton btn3 = new JButton("Button-WEST");
        btn3.addActionListener(this);
        contentPane.add(btn3, BorderLayout.WEST);

        JButton btn4 = new JButton("Button-SOUTH");
        btn4.addActionListener(this);
        contentPane.add(btn4, BorderLayout.SOUTH);

        JButton btn5 = new JButton("Button-CENTER");
        btn5.addActionListener(this);
        contentPane.add(btn5, BorderLayout.CENTER);
    }

    public static void main(String [] args){
        //JFrame クラスのインスタンスを生成
        JButtonSampleBorderLayout fr = new JButtonSampleBorderLayout();
        //フレームをディスプレイ画面に表示します
        fr.setVisible(true);
    }
}
```

2.3 レイアウトマネージャー：GridLayout クラス

JButtonSampleGridLayout.java

```
import javax.swing.*; //JFrame クラスを含む
import java.awt.event.*; //ActionListener インターフェースを含む
import java.awt.*; //Container クラス, GridLayout クラスを含む

class JButtonSampleGridLayout extends JFrame implements ActionListener{
    public void actionPerformed(ActionEvent e){
        System.out.println("Button was pushed!");
    }

    JButtonSampleGridLayout(){
        setSize(500, 400);
        setTitle("Frame title");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        //レイアウトマネージャーによるコンポーネントの配置
        Container contentPane = this.getContentPane();
        contentPane.setLayout(new GridLayout(3,2) );

        JButton btn1 = new JButton("Button-1");
        btn1.addActionListener(this);
        contentPane.add(btn1);

        JButton btn2 = new JButton("Button-2");
        btn2.addActionListener(this);
        contentPane.add(btn2);

        JButton btn3 = new JButton("Button-3");
        btn3.addActionListener(this);
        contentPane.add(btn3);

        JButton btn4 = new JButton("Button-4");
        btn4.addActionListener(this);
        contentPane.add(btn4);

        JButton btn5 = new JButton("Button-5");
        btn5.addActionListener(this);
        contentPane.add(btn5);
    }

    public static void main(String [] args){
        //JFrame クラスのインスタンスを生成
        JButtonSampleGridLayout fr = new JButtonSampleGridLayout();
        //フレームをディスプレイ画面に表示します
        fr.setVisible(true);
    }
}
```

2.4 パネル：JPanel クラス

JPanelSampleBorderLayout.java

```
import javax.swing.*; //JFrame クラスを含む
import java.awt.*; //Container クラス, BorderLayout クラスを含む

class JPanelSampleBorderLayout extends JFrame{

    JPanelSampleBorderLayout(){
        setSize(500, 400);
        setTitle("Frame title");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        //レイアウトマネージャーによるコンポーネントの配置
        Container contentPane = this.getContentPane();
        contentPane.setLayout(new BorderLayout() );

        JPanel panel1 = new JPanel();
        panel1.setBackground(Color.BLUE);
        contentPane.add(panel1,BorderLayout.NORTH);

        JPanel panel2 = new JPanel();
        panel2.setBackground(Color.RED);
        contentPane.add(panel2,BorderLayout.EAST);

        JPanel panel3 = new JPanel();
        panel3.setBackground(Color.GREEN);
        contentPane.add(panel3,BorderLayout.WEST);

        JPanel panel4 = new JPanel();
        panel4.setBackground(Color.ORANGE);
        contentPane.add(panel4,BorderLayout.SOUTH);

        JPanel panel5 = new JPanel();
        panel5.setBackground(Color.white);
        contentPane.add(panel5,BorderLayout.CENTER);
    }

    public static void main(String [] args){
        //JFrame クラスのインスタンスを生成
        JPanelSampleBorderLayout fr = new JPanelSampleBorderLayout();
        //フレームをディスプレイ画面に表示します
        fr.setVisible(true);
    }
}
```

2.5 ボタンと getSource メソッド

JButtonSample02.java

```
import javax.swing.*; // JFrame クラスを含む
import java.awt.event.*; // ActionListener インターフェースを含む
import java.awt.*; // Container クラス, FlowLayout クラスを含む

class JButtonSample02 extends JFrame implements ActionListener{

    JButton btn1 = new JButton("Button-1");
    JButton btn2 = new JButton("Button-2");
    JButton btn3 = new JButton("Button-3");

    public void actionPerformed(ActionEvent e){
        //押されたボタンの参照を selectedButton に代入する
        //Object obj = e.getSource();
        //JButton selectedButton = (JButton)obj;
        JButton selectedButton = (JButton)e.getSource();

        if(selectedButton==btn1){
            System.out.println("Button-1 was pushed!");
        }else if(selectedButton==btn2){
            System.out.println("Button-2 was pushed!");
        }else{
            System.out.println("Button-3 was pushed!");
        }
    }

    JButtonSample02(){
        setSize(500, 400);
        setTitle("Frame title");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        //レイアウトマネージャーによるコンポーネントの配置
        Container contentPane = this.getContentPane();
        contentPane.setLayout(new FlowLayout() );

        btn1.addActionListener(this);
        contentPane.add(btn1);

        btn2.addActionListener(this);
        contentPane.add(btn2);

        btn3.addActionListener(this);
        contentPane.add(btn3);
    }

    public static void main(String [] args){
        JButtonSample02 fr = new JButtonSample02();
        fr.setVisible(true);
    }
}
```

2.6 パネルとボタン

JButtonSample03.java

```
import javax.swing.*; //JFrame クラスを含む
import java.awt.event.*; //ActionListener インターフェースを含む
import java.awt.*; //Container クラス, FlowLayout クラスを含む

class JButtonSample03 extends JFrame implements ActionListener{
    JButton btn1 = new JButton("Button-1");
    JButton btn2 = new JButton("Button-2");
    JButton btn3 = new JButton("Button-3");

    public void actionPerformed(ActionEvent e){
        //押されたボタンの参照を selectedButton に代入する
        JButton selectedButton = (JButton)e.getSource();

        if(selectedButton==btn1){
            System.out.println("Button-1 was pushed!");
        }else if(selectedButton==btn2){
            System.out.println("Button-2 was pushed!");
        }else{
            System.out.println("Button-3 was pushed!");
        }
    }

    JButtonSample03(){
        setSize(500, 400);
        setTitle("Frame title");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setLayout(new FlowLayout());

        JPanel panel1 = new JPanel();
        panel1.setPreferredSize(new Dimension(200, 100));
        panel1.setLayout(new BorderLayout());

        btn1.addActionListener(this);
        panel1.add(btn1, BorderLayout.NORTH);

        btn2.addActionListener(this);
        panel1.add(btn2, BorderLayout.CENTER);

        btn3.addActionListener(this);
        panel1.add(btn3, BorderLayout.SOUTH);

        getContentPane().add(panel1);
    }

    public static void main(String [] args){
        JButtonSample03 fr = new JButtonSample03();
        fr.setVisible(true);
    }
}
```


課題 1

下の Figure1 のようなボタン付きフレームを表示するプログラム `JButtonSample04_00**000.java` を作成せよ。ただし、以下の条件を満たすものとする。また、`00**000`は各自の学籍番号とする。

- 各ボタンをクリックした際に、ボタンに応じた標準出力がなされるようにする。
- フレームサイズは変更できないようにする。

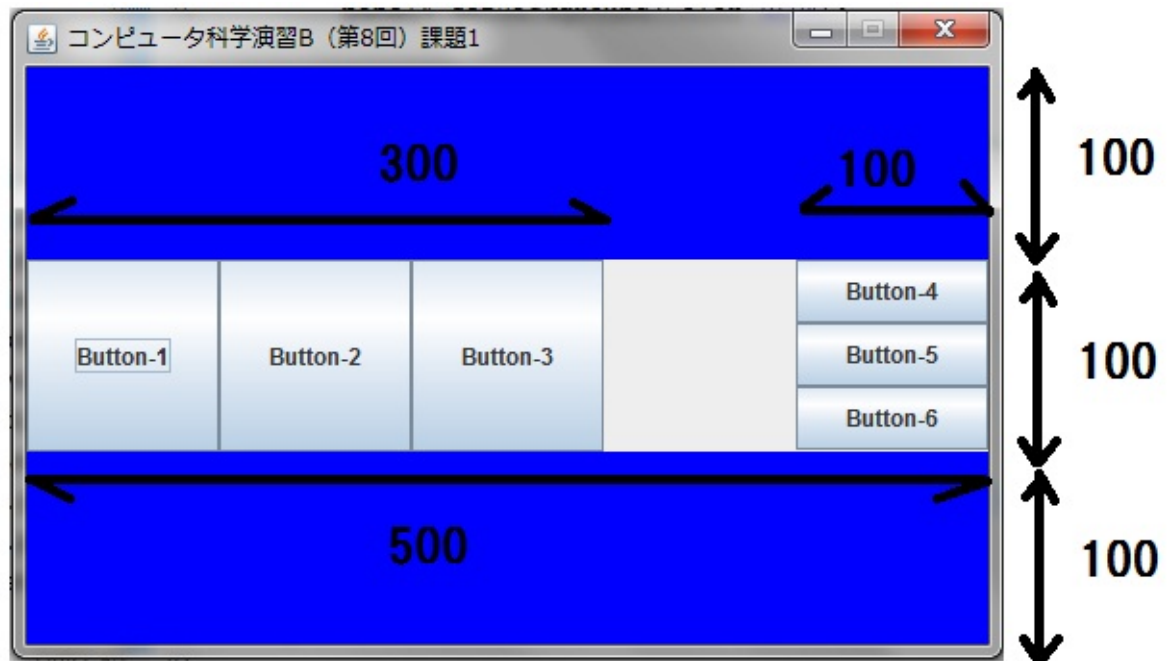


Figure 1: 課題 1 の実行結果

提出先: <https://tdu.app.box.com/f/97bd8063aba44cd1beac9240e024095c>

提出期限: 7/12 23:59