

## GUI プログラミング 第1回ノート

2020 年 9 月 24 日

必要な部分を埋めて完成させること。自分用のメモは自由に記入してよい。

### GUI プログラミング はじめに

Q. GUI は何の略か?

Graphical User Interfaces)

Q. WIMP はそれぞれ何の頭文字か?

- W : window
- I : icon
- M : menu
- P : pointing device

### 直接操作インタフェース (Direct Manipulation)

- 対話と関係のあるオブジェクト(対象)を常に画面に表示。
- 実際にオブジェクトに働きかける動作や、メニュー選択によりユーザの指示を入力。
- これらの操作は速く、可逆的で、結果は 直ちにオブジェクトの変化として見るができる。

Q. メタファ (隠喩) を自分の言葉で例を挙げて説明しなさい。

世の中の現象や物事をより具体的なイメージに置き換えて理解を促進するような認知方式の一つ

### ルックアンドフィール (Look and Feel)

Q. ルックアンドフィールとは何か、自分の言葉で説明しなさい。

アプリケーションの操作部分や、デザインのこと

ルック (Look)見かけ

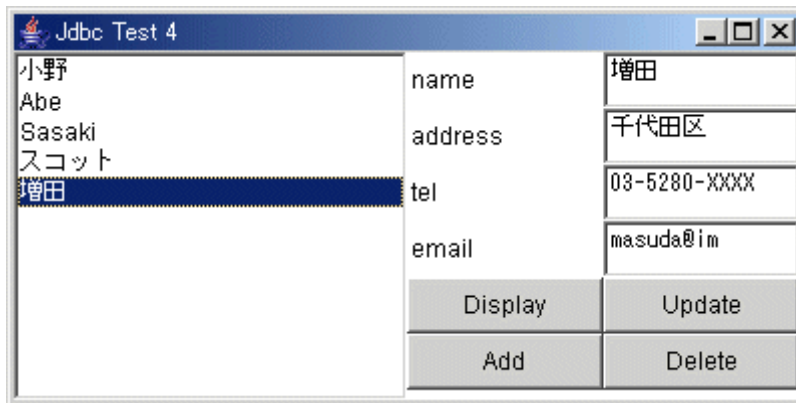
フィール (Feel)操作感

## ユーザインタフェースの作成

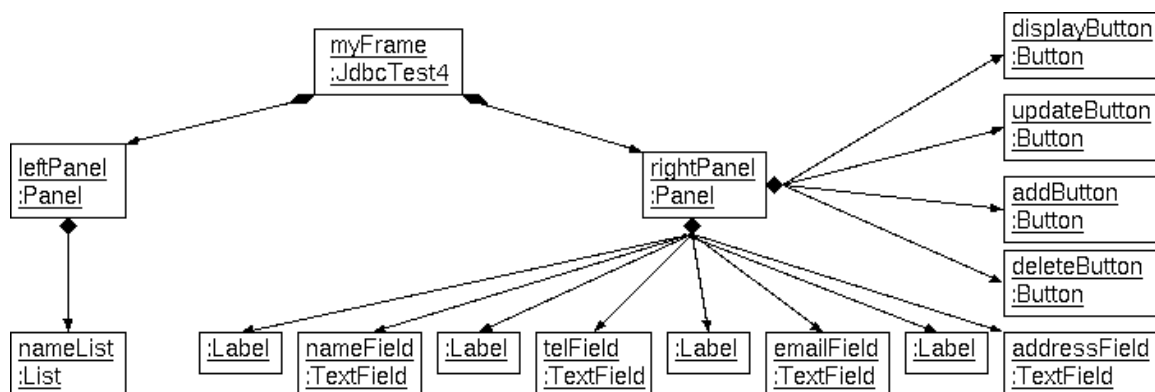
定型的な UI 部品(Component)はあらかじめ用意することができる。 オブジェクト指向の応用の成功例の一つ。 ↓

- ボタン
- メニュー
- リスト
- テキスト
- ダイアログ などなど・・・

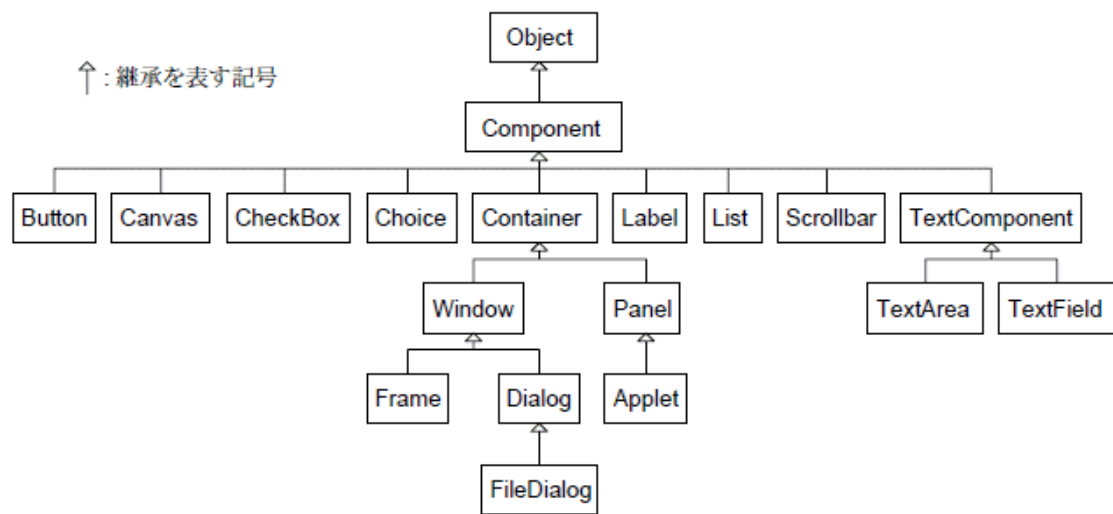
部品(コンポーネント)を組み合わせて UI を構成する。一つのウィンドウにはたくさんの部品が組み合わされている。 パネルという台紙にコンポーネントを貼り付けていくというイメージである。



このウィンドウは以下の構成で作られている。



Java では Component クラス配下にさまざまな GUI 部品が用意されている。 以下は AWT (Abstract Windowing Toolkit) のクラス階層の一部である。



目的に応じて既存の GUI 部品を選んで組み合わせる。

Q. コンポーネントに関する以下の表を完成させなさい。

用途	クラス
ウィンドウ枠	Frame
ダイアログ	Dialog, FileDialog
アプレット	Applet
複数部品の管理	Panel
コマンド実行	Button
グラフィック描画	Canvas
二値選択	Checkbox
択一選択	Choise
択一または複数選択	Choise, List
テキスト表示	Label
数値入力	Scrollbar
複数行テキスト入出力	TextArea
単一行テキスト入出力	TextField

## Java を利用した GUI プログラミング

Q. GUI の作成とは、作成する GUI に対して、以下を行うことである。空欄を埋めよ。

1. 部品の選択
2. 部品の配置
3. AP（アプリケーション）との結び付け

## アプレット (Applet)

let とは小さいものを意味する。Web ブラウザ上で動作する Java プログラムを Java アプレットと呼ぶ。Web サーバから Java のクラスファイルを Web ブラウザ上にダウンロードして動作する。

## イベント駆動型プログラム (Event-Driven Program)

Q. イベント駆動型とは何か、自分の言葉で説明しなさい。

イベントが発生するとプログラムが動く。今までの main() メソッドの先頭から最後まで実行して終わるプログラムとは異なる。

## イベントの例

- マウスボタンが押された。
- マウスボタンが離された。
- マウスカーソルが領域に入った。
- キーボードが押された。
- コンポーネントが作られた。
- 一定時間が過ぎた。
- 何も起こらなかった。

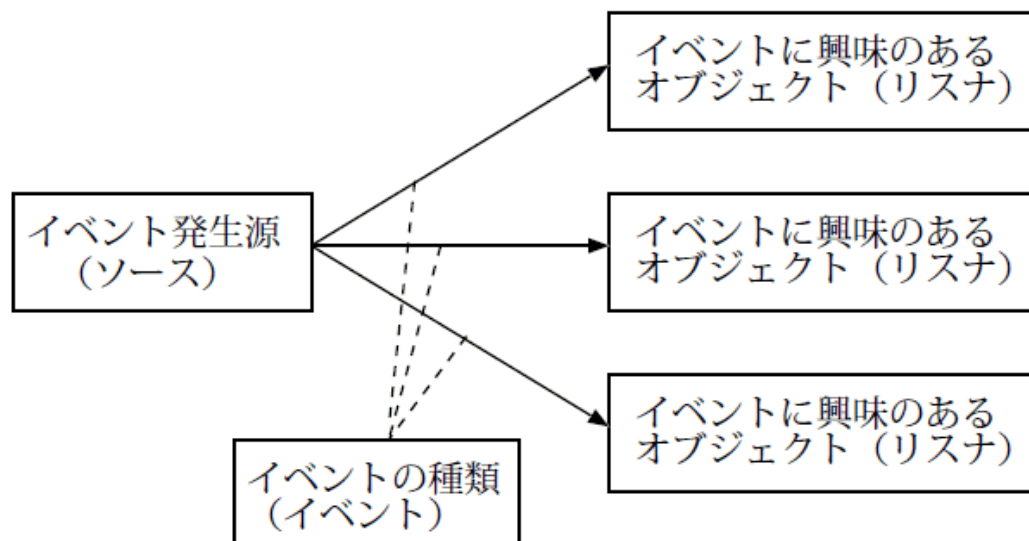
などなど・・・、実際にプログラムが動いてみないとどのようなイベントがどのような順番で発生するかわからない。

以下にイベントの例を示す。

操作の種類	イベント	リスナ	メソッド
コマンド実行	ActionEvent	ActionListener	void actionPerformed(ActionEvent e)
フォーカス移動	FocusEvent	FocusListener	void focusGained(FocusEvent e) void focusLost(FocusEvent e)
項目の選択操作	ItemEvent	ItemListener	void itemStateChanged(ItemEvent e)
マウスボタン操作  およびマウス移動	MouseEvent	MouseListener  MouseMotion- Listener	void mouseClicked(MouseEvent e) void mousePressed(MouseEvent e) void mouseReleased(MouseEvent e) void mouseDragged(MouseEvent e) void mouseMoved(MouseEvent e)
テキストの変更	TextEvent	TextEventListener	void textValueChanged(TextEvent e)

## イベントの取得とメッセージの分配

プログラムが動いている時は常にイベント待ちの状態。あるイベントが発生したら何をするのかという組を必要だけ用意しておく。



イベントが発生すると対応するコンポーネントがリスナにメッセージを送る。

#### イベントハンドラ（リスナ側の処理）

発生したイベントに応じて、コンポーネントが実行すべき処理を登録する。

例えば

ボタンコンポーネントの上でマウスボタンがクリックされた。



リスナのボタンが押された時の処理が呼び出される。

```

void actionPerformed(ActionEvent e)
{
    処理;
}
  
```

ただし、あらかじめコンポーネントとリスナを結びつけておく必要がある。Java では各種リスナはインタフェースとして定義されている。自分が作成したクラスに必要なインタフェースを実装する。

#### 一般形

登録	void addリスナ型 (リスナ型 el)
削除	void removeリスナ型 (リスナ型 el)

#### ボタンクリックの例

登録	void addActionListener(ActionListener al)
削除	void removeActionListener(ActionListener al)

Q. イベントとリスナの関係について自分の言葉で説明しなさい。

## Java でのプログラミング

今までの `main()` メソッドを使ったプログラムと違い、

1. コンポーネント(フレーム、パネル)上に必要な部品を配置する。
2. 各部品のプロパティを設定する。
3. 各部品のリスナ(イベントハンドラ)を登録する。
4. Java でメソッド、リスナを書く

## イントロのおわりに

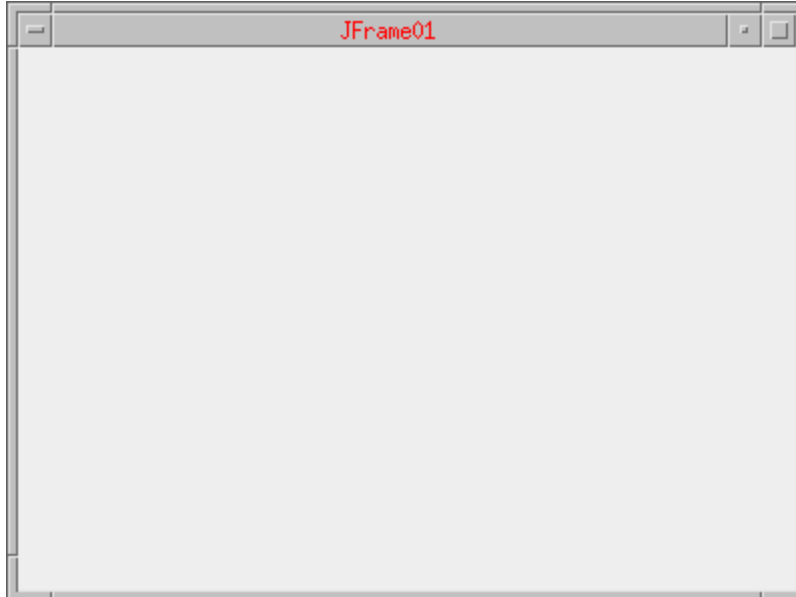
- オブジェクト指向
- クラス-インスタンス
- 属性と操作  
プロパティと関数、手続き
- インヘリタンス(継承)  
スーパークラスとサブクラス
- ポリモフィズム(多態性)
- メッセージ通信

今後の GUI プログラミングでは **Swing** を利用していく。

オブジェクト指向の基本についてはオブジェクト指向プログラミングで行っている。

## ウィンドウ JFrame

### 1-1 最も簡単な GUI アプリケーション



import javax.swing.\*; // Swing を利用するので

```
class JFrame01 {  
    public static void main(String[] args) {  
        JFrame frame = new JFrame("JFrame01"); // JFrame のインスタンス生成  
        frame.setSize(400, 300); // サイズ設定  
        frame.setVisible(true); // 画面上に表示  
    }  
}
```

### ウィンドウクローズの際の動作を指定する

setDefaultCloseOperation() メソッドを使用する。

パラメータ	動作
JFrame.DO_NOTHING_ON_CLOSE	何もしない。AWT の Frame と同じ
JFrame.HIDE_ON_CLOSE	ウィンドウを隠す(Swing でのデフォルト)
JFrame.DISPOSE_ON_CLOSE	ウィンドウを破棄し、アプリケーションを終了する
JFrame.EXIT_ON_CLOSE	ウィンドウを破棄し、アプリケーションを終了する

```
import javax.swing.*;

class JFrame02 {
    public static void main(String[] args) {
        JFrame frame = new JFrame( "JFrame02" );
        frame.setDefaultCloseOperation( JFrame.EXIT_ON_CLOSE ); /*** 閉じる
ボタンの設定
        frame.setSize(400, 300);
        frame.setVisible(true);
    }
}
```

以上

学籍番号: 18nc006

氏名: 石澤芳瑠