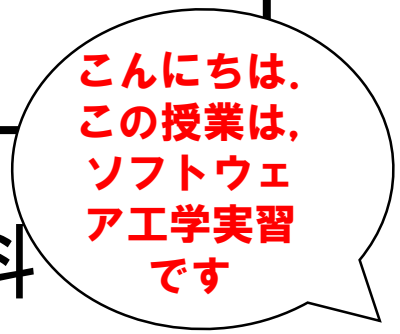
ソフトウェア工学実習
Software Engineering Practice
(第06回)

SEP07-002 Exercise（ SimpleCalc)



慶應義塾大学・理工学部・管理工学科
飯島 正
iijima@ae.keio.ac.jp

MVC (Model-View-Controller) の分離

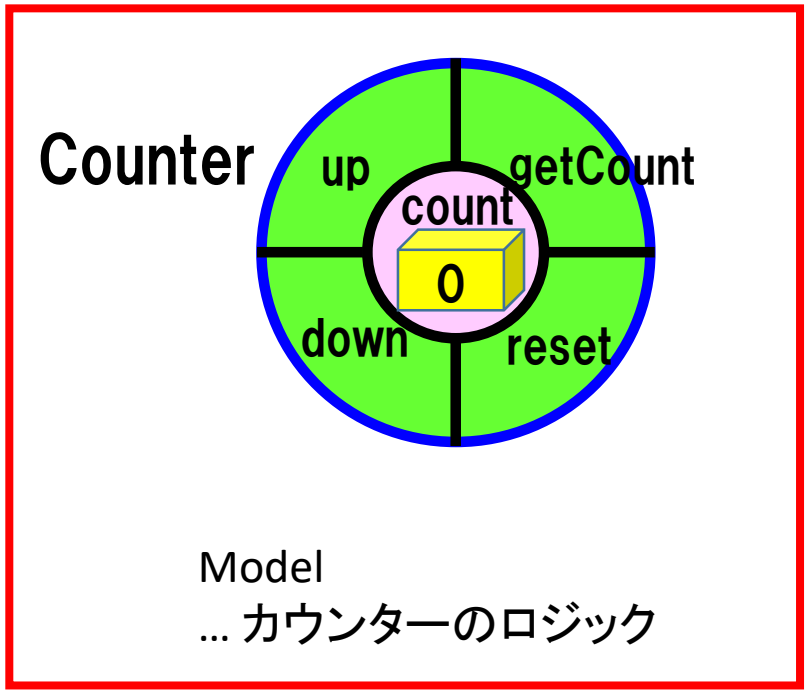
今日の話題: モデル-ビュー-コントローラ (Counterの分離)

- Model … カウンターのロジック
- View … カウンターの見た目 (表示)
- Controller …カウンターの操作

本質的なロジック

ユーザインタフェース

まず，
MVCのそれぞれで
パッケージを
分離します．



iijima@ae.keio.ac.jp

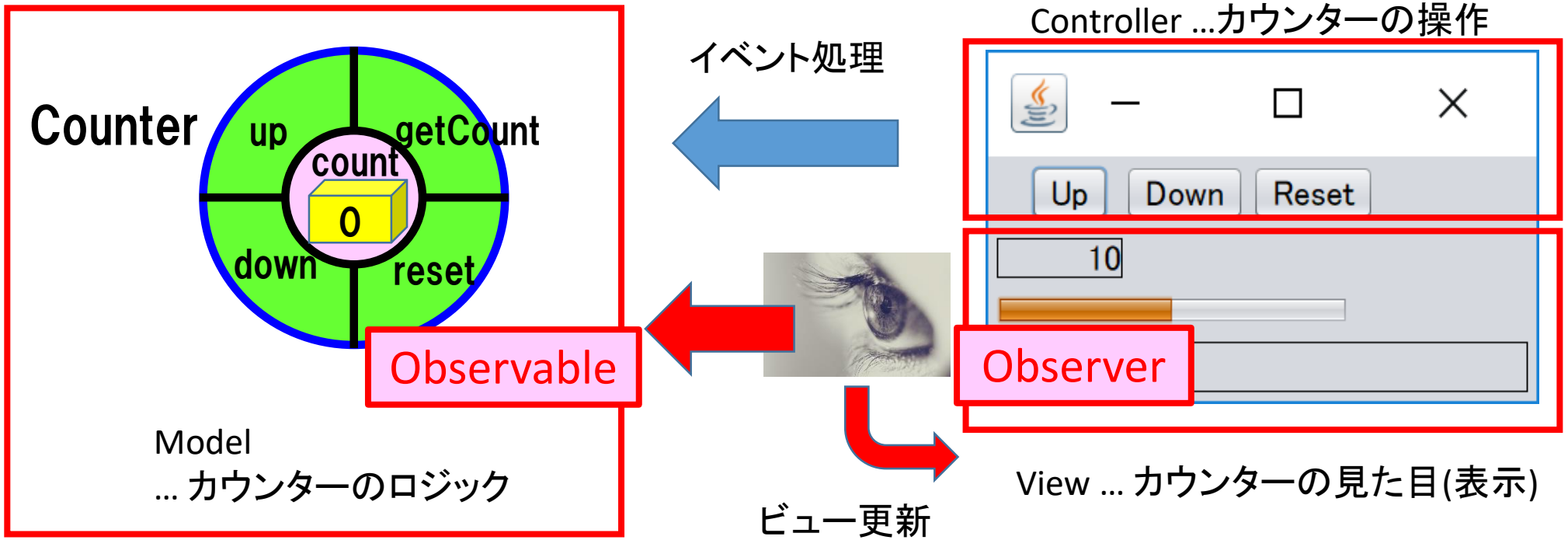
今日の話題: Observer/Observableパターンの導入

- Model … カウンターのロジック
- View … カウンターの見た目(表示)
- Controller …カウンターの操作

本質的なロジック

ユーザインタフェース

まず，
MVCのそれぞれで
パッケージを
分離します．



別の例題: 簡単な電卓（5月に入ったら，やりましょう）

- 簡単な電卓を作ってみます（5月に入ってから自力で，作っていただき，拡張していきます）
  - テキストフィールド，ボタン，ラベルを使います．
- まずは，簡単に　GUIの中に計算機能が埋め込んで作ってしまいましょう．
- 問題点→　計算が複雑になったら，ゴチャゴチャになってしまう．
- 解決→　MVC (Model-View-Controller) の考え方を導入します．

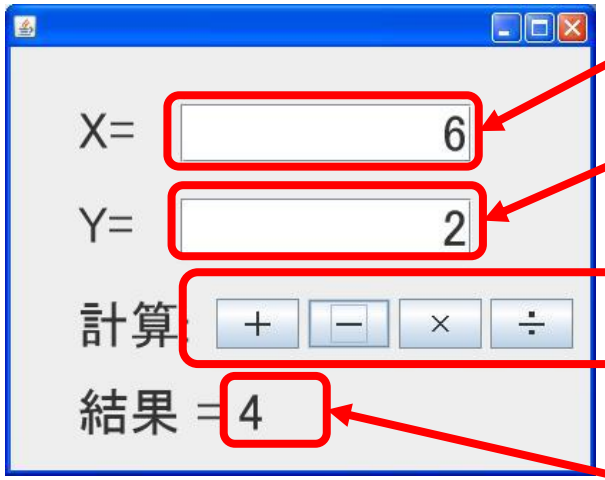
ところで，
もう一つ
例題を用意し
ます．

5月に入った
ら．これまで
の知識を使っ
て自力で
作ってみて
いただきます

iijima@ae.keio.ac.jp

別の例題: 簡単な電卓（GUIビルダですぐ作れますが…）

- NetBeansのGUIビルダで簡単に作れます.
  - 素早く試作品（プロトタイプ）を作ってしまいましょう
  - ラピッドプロトライピングといいます.



テキストフィールド(JTextField): xTextField

テキストフィールド(JTextField): yTextField

ボタン(JButton):
addButton, subButton,
mulButton, divButton

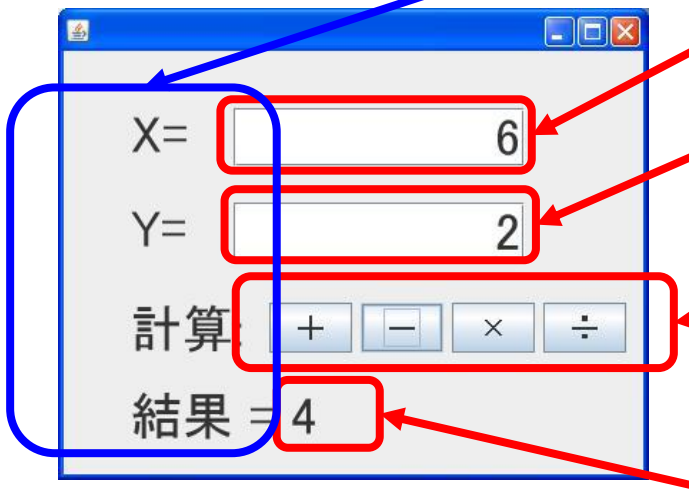
ラベル(JLabel):
resultLabel

iijima@ae.keio.ac.jp

別の例題: 簡単な電卓（GUIビルダですぐ作れますが…）

- NetBeansのGUIビルダで簡単に作れます.
  - 素早く試作品（プロトタイプ）を作ってしまいましょう
  - ラピッドプロトライピングといいます.

こちら側の見出しラベル(JLabel)の変数名は特に考えなくてもいいでしょう



テキストフィールド(JTextField): xTextField

テキストフィールド(JTextField): yTextField

ボタン(JButton):
addButton, subButton,
mulButton, divButton

ラベル(JLabel):
resultLabel

別の例題: 簡単な電卓（設計→MVCの切り分け）

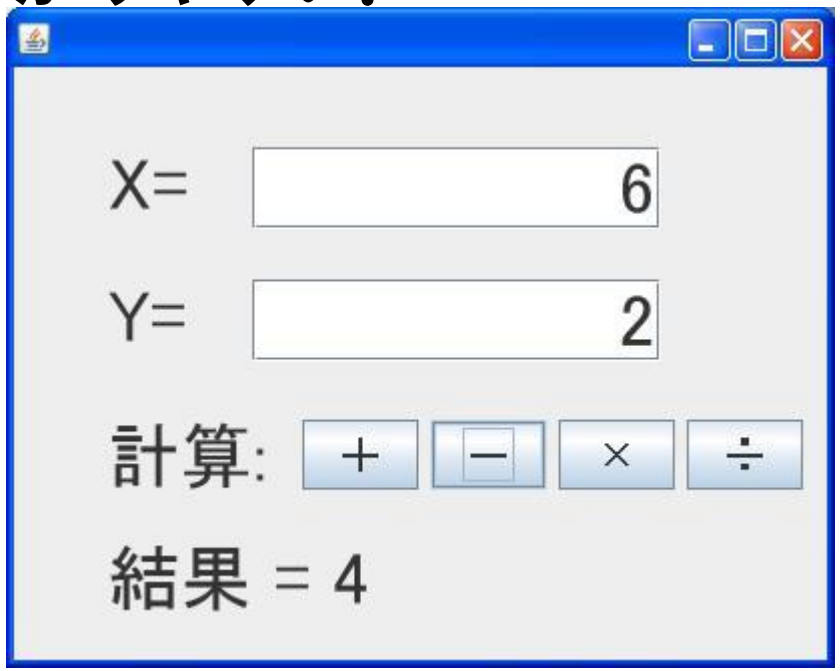
- モデルと，GUIの分離は，わかりやすい．

数値x
数値y
四則演算
例外処理
計算結果

Model

計算機能
＝モノとしての
電卓の概念

クラスSimpleCalc



View-Controller

見た目
＝結果表示

制御
＝ボタン

クラスCalcFrame

iijima@ae.keio.ac.jp

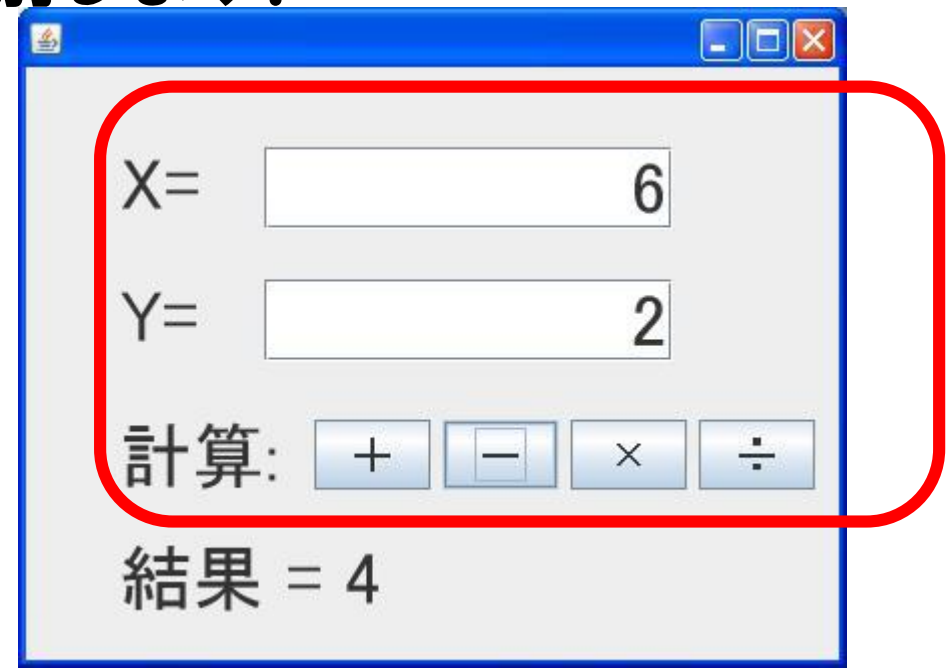
別の例題: 簡単な電卓（設計→MVCの切り分け）

- ビューとコントローラを識別します．

数値x
数値y
四則演算
例外処理
計算結果

数値x 数値y

演算の種類



ビューとコントローラを識別します

アプリケーションの動きを考えましょう

入力はコントローラの仕事です

Model

計算機能
＝モノとしての電卓の概念

クラスSimpleCalc

View-Controller

見た目
＝結果表示

制御
＝ボタン

クラスCalcFrame

※もう少し正確には，入力だけではなく，モデルを操作する（コントロールする）ロジックがコントローラの本質なのですが…

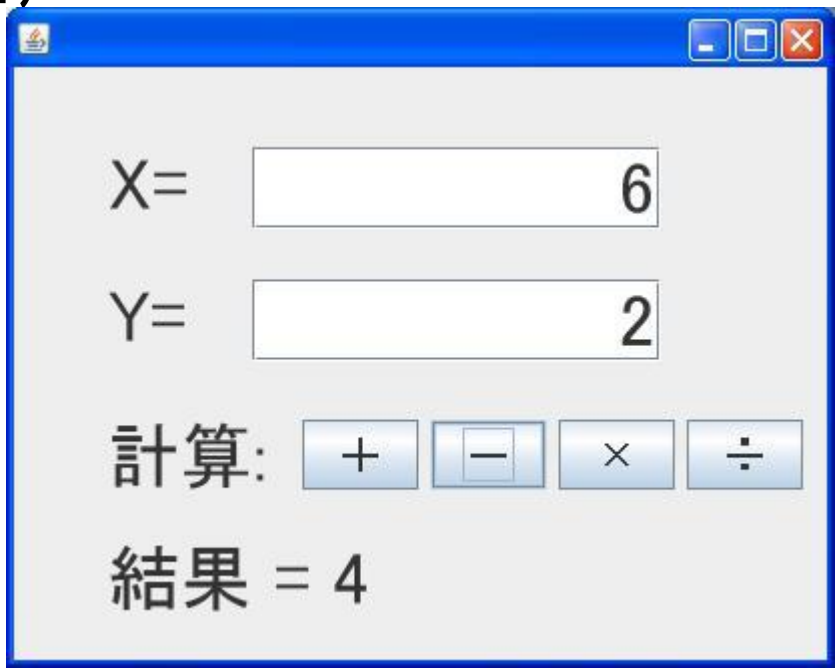
iijima@ae.keio.ac.jp

別の例題: 簡単な電卓（設計→MVCの切り分け）

- MVC (Model-View-Controller)

数値x
数値y
四則演算
例外処理
計算結果

Model



View-Controller

入力データで計算します.

計算はモデルの仕事です

計算機能
= モノとしての電卓の概念

クラスSimpleCalc

見た目
＝結果表示

制御
＝ボタン

クラスCalcFrame

iijima@ae.keio.ac.jp

別の例題: 簡単な電卓（設計→MVCの切り分け）

- MVC (Model-View-Controller)

数値x
数値y
四則演算
例外処理
計算結果

Model

計算結果

View-Controller

計算機能
= モノとしての
電卓の概念

クラスSimpleCalc

見た目
＝結果表示

制御
＝ボタン

クラスCalcFrame

iijima@ae.keio.ac.jp

別の例題: 簡単な電卓（設計→MVCの切り分け）

- MVC (Model-View-Controller)

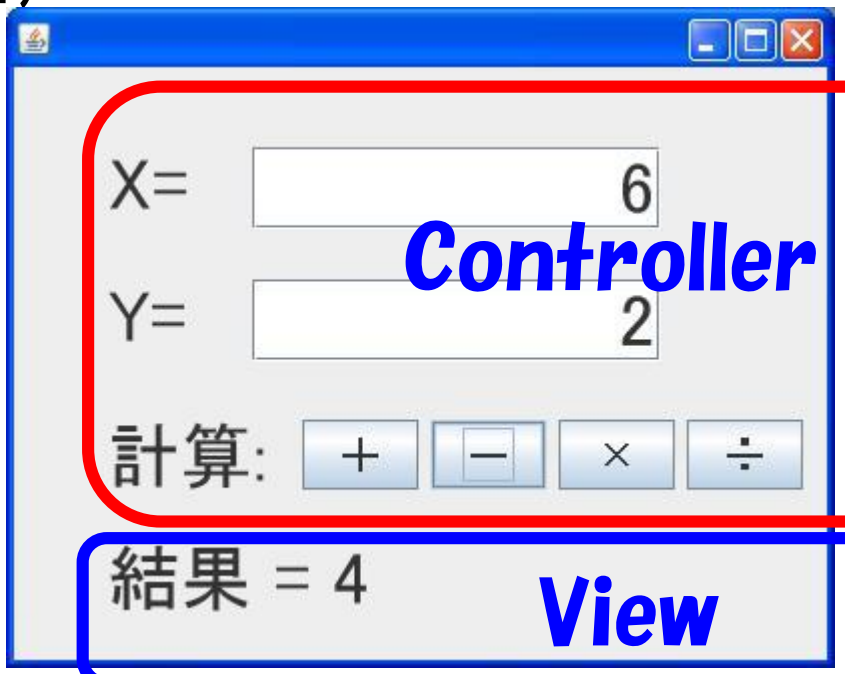
数値x
数値y
四則演算
例外処理
計算結果

Model

数値x 数値y

演算の種類

計算結果



View-Controller

※もう少し正確には，
これは，コントローラのUI
部分であって，コントローラ
のロジックではありませんが，
今のところ，ボタンの
イベントハンドラやそれに
付随するメソッドに
ロジックが定義されています．

これで，
コントローラ
と
ビューの
区別がつきま
した

計算機能
= モノとしての
電卓の概念
クラスSimpleCalc

見た目
＝結果表示

制御
＝ボタン

クラスCalcFrame

別の例題: 簡単な電卓（設計→MVCの切り分け）

- クラス構成の概要設計

クラスSimpleCalcFrame

数値x　数値y

GUI

数値x
数値y
四則演算
例外処理
計算結果

演算の種類

クラスSimpleCalcController

Controller

Model

View　クラスSimpleCalcView

計算結果

計算機能
= モノとしての
電卓の概念

クラスSimpleCalc

それぞれ
クラス名を決めて，
概要設計としま
しょう

別の例題: 簡単な電卓（クラス内部の設計）

- パッケージ構成
  - simplecalcパッケージ
    - Mainクラス
    - SimpleCalcFrameクラス
  - modelパッケージ
    - SimpleCalcクラス
  - viewパッケージ
    - SimpleCalcViewクラス
  - controllerパッケージ
    - SimpleCalcControllerクラス

数値x
数値y
四則演算
例外処理
計算結果

パッケージに
配分します

別の例題: 簡単な電卓（クラス内部の設計）

- パッケージ構成

  simplecalcパッケージ　Mainクラス

		型	名前	引数	説明
クラス（static）	メソッド	void	main	String[] args	SimpleCalcFrameを生成し表示する

別の例題: 簡単な電卓（クラス内部の設計）

- パッケージ構成

simplecalcパッケージ

SimpleCalcFrameクラス

SimpleCalcFrameクラス

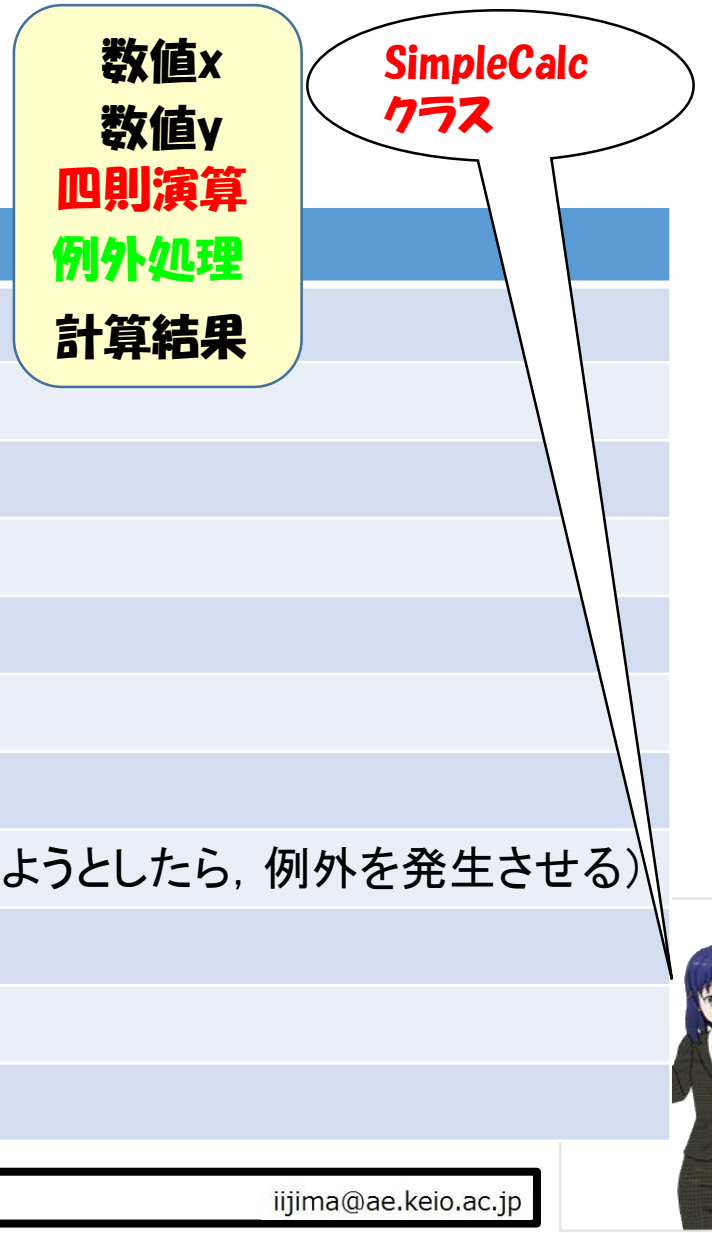
		型	名前	引数	説明
クラス（static）	属性				
	メソッド				
コンストラクタ		SimpleCalcFrame			初期化. モデルの生成, ビューとコントロールの生成と パネルの貼り付け
インスタンス	属性	SimpleCalc	model		モデルへの参照
		SimpleCalcView	view		ビューへの参照
		SimpleCalcController	SimpleCalc		コントローラへの参照
	メソッド				

別の例題: 簡単な電卓（クラス内部の設計）

- パッケージ構成

simplecalc. modelパッケージ　SimpleCalcクラス

		型	名前	引数	説明
コンストラクタ		SimpleCalc		なし	初期化
インスタンス	属性	int	x		データ
		int	y		データ
		int	result		計算結果
	メソッド	void	add	なし	加算
		void	sub	なし	減算
		void	mul	なし	乗算
		void	div	なし	除算 (ゼロで割り算をしようとしたら，例外を発生させる)
		void	setX	int x	setter
		void	setY	int y	setter
		int	getResult	なし	getter



iijima@ae.keio.ac.jp

別の例題: 簡単な電卓（クラス内部の設計）

- パッケージ構成

Simplecalc. viewパッケージ

SimpleCalcViewクラス

結果 = 4

SimpleCalcView クラス

		型	名前	引数	説明
クラス (static)	属性				
	メソッド				
コンストラクタ		SimpleCalcView			
インスタンス	属性	SimpleCalc	model		モデルへの参照
	メソッド		update		

別の例題: 簡単な電卓（クラス内部の設計）

- パッケージ構成

  Simplecalc. controllerパッケージ

  SimpleCalcControllerクラス

SimpleCalc Controller クラス

		型	名前	引数	説明
クラス（static）	属性				
	メソッド				
コンストラクタ		SimpleCalcFrame			
インスタンス	属性	SimpleCalc	model		モデルへの参照
		SimpleCalcView	view		ビューへの参照
		SimpleCalcController	SimpleCalc		コントローラへの参照
	メソッド