

プログラム問題集(課題)

プログラミング入門2 第8回

■ 問題11

次のクラスMarkを作成せよ。

インスタンス変数

変数の型と名前	初期値	説明
String name	無し	学生の名前
int mark	無し	試験の得点

コンストラクタ

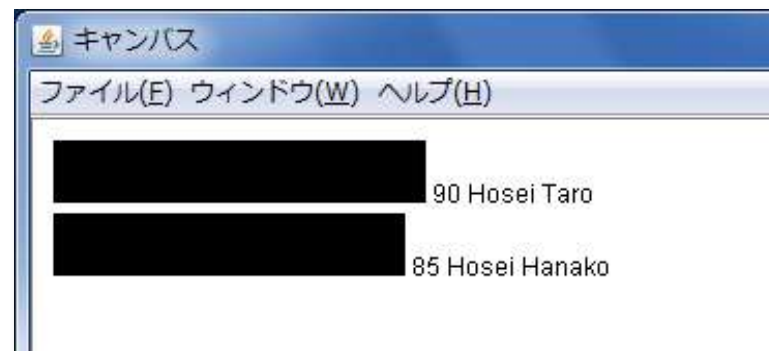
コンストラクタ(引数)	機能
Mark(String name, int mark)	Markクラスのインスタンス変数の値がそれぞれ引数の値であるMarkインスタンスを作成する。

(クラス名 : Mark)

■ 問題12

問題: 実行例のように得点を長方形で表して、データをキャンバス上に表示する Mark のインスタンスメソッドを作成せよ。

返り値の型	メソッド名(引数)	機能
void	showMark(int x, int y, int height)	実行例のようにMarkのインスタンスを元にして表示する。



(クラス名: Q12ShowMark)

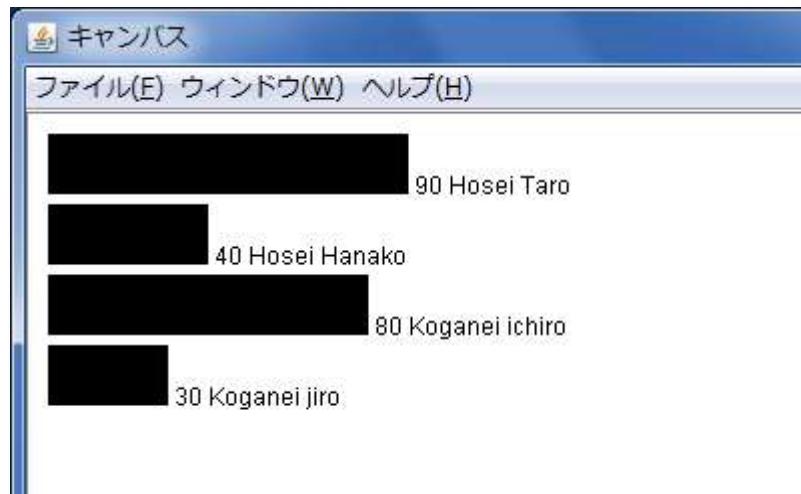
■ 問題13(0)

問題: Markのインスタンスを配列に格納するようにさせよ(この配列を得点リストと呼ぶ)。

(クラス名: Q13ShowMarkList)

■ 問題13(1)

問題：得点リストに含まれるデータを実行例のようにグラフとして表示せよ。



(クラス名 : Q13ShowMarkList)

■ 問題14(1)

次のメソッドgetMarkを、Q14MarkListの中に作成し、動作を確認せよ。

返り値の型	メソッド名(引数)	機能
Mark	getMark(Mark [] list, String name)	得点リストlist中に格納されているMark インスタンスのうち名前nameを持つも のを一つ返す。該当する名前が見つか らない場合は、nullを返す。

(クラス名 : Q14MarkList)

■ 問題14(2)

ユーザーが入力した名前の得点を表示せよ。ただし、該当する名前がリスト内に見つからない場合には、メッセージを出すようにせよ。

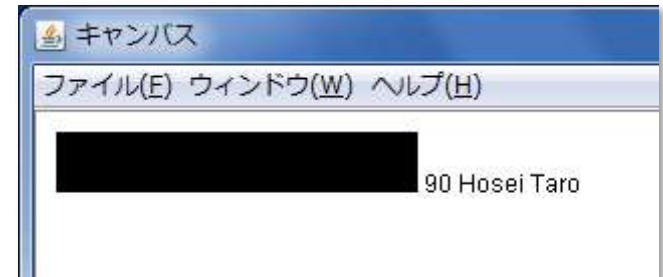


入力

名前を入力して下さい

Hosei Taro

了解 取消し



キャンバス

ファイル(E) ウィンドウ(W) ヘルプ(H)

[Redacted] 90 Hosei Taro

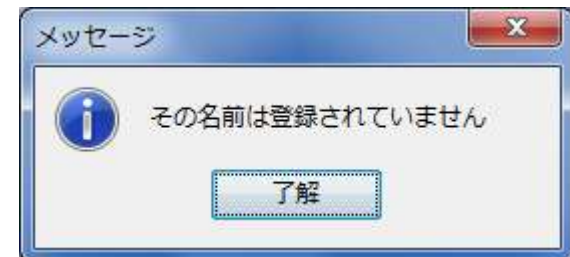


入力

名前を入力して下さい

Hosei Hosei

了解 取消し



メッセージ

その名前は登録されていません

了解

(クラス名: Q14MarkList)

■ 問題14(3)

ユーザーが入力した名前の人得点を表示せよ。ここでは、連続で入力できるようにせよ。該当する名前がリスト内に見つからない場合に、入力を終了するものとする。

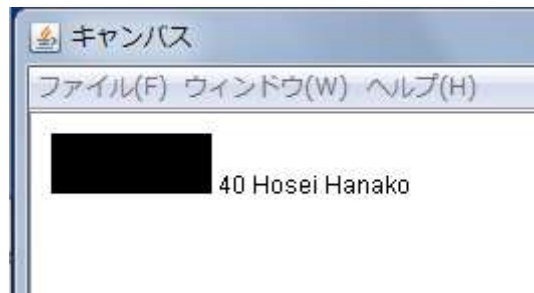


入力

名前を入力して下さい

Hosei Hanako

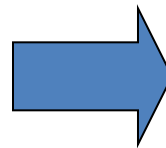
了解 取消し



キャンパス

ファイル(F) ウィンドウ(W) ヘルプ(H)

40 Hosei Hanako

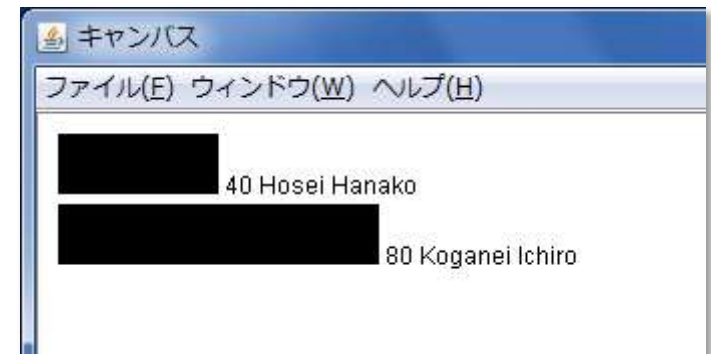


入力

名前を入力して下さい

Koganei Ichiro

了解 取消し



キャンパス

ファイル(E) ウィンドウ(W) ヘルプ(H)

40 Hosei Hanako

80 Koganei Ichiro

(クラス名: Q14MarkList)

■ 問題21(1) (以降は余力のあるもの向き)

NamedRectangleクラスを作成せよ。

インスタンス変数

変数の型と名前	初期値	説明
String name	無し	長方形の名前
Int x	無し	X座標
Int y	無し	Y座標
Int width	無し	幅
Int height	無し	高さ
int red	無し	赤色成分
int green	無し	緑色成分
int blue	無し	青色成分

(クラス名: NamedRectangle)

コンストラクタ

コンストラクタ(引数)	機能
NamedRectangle(String name, int x, int y, int width, int height, int red, int green, int blue)	NamedRectangleクラスのインスタンス変数の値がそれぞれ引数の値であるNamedRectangleインスタンスを作成する。

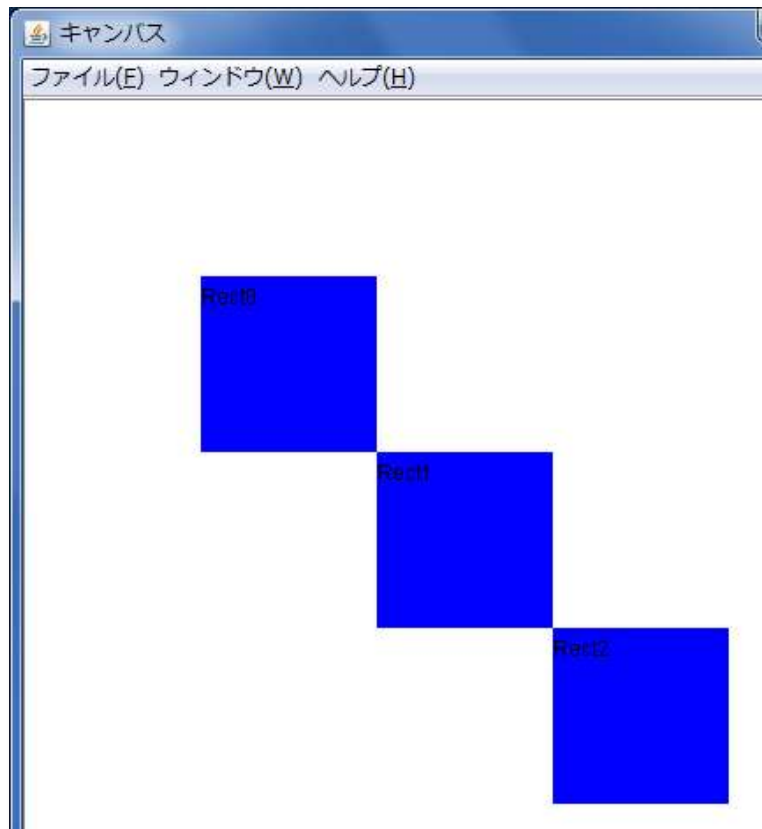
インスタンスメソッド

返り値の型	メソッド名(引数)	機能
void	show()	キャンバス上に色付きの四角形を表示する。名前も表示する。(問題21(2)の実行例を参考にせよ。)

(クラス名 : NamedRectangle)

■ 問題21(2)

NamedRectangleのインスタンスを配列に格納し、そのインスタンスを実行例のように表示せよ。



見にくいですが、それぞれの四角形の左上に名前が書いてある。

この例では左上から、
Rect1
Rect2
Rect3
である。

(クラス名: Q21RectangleList)

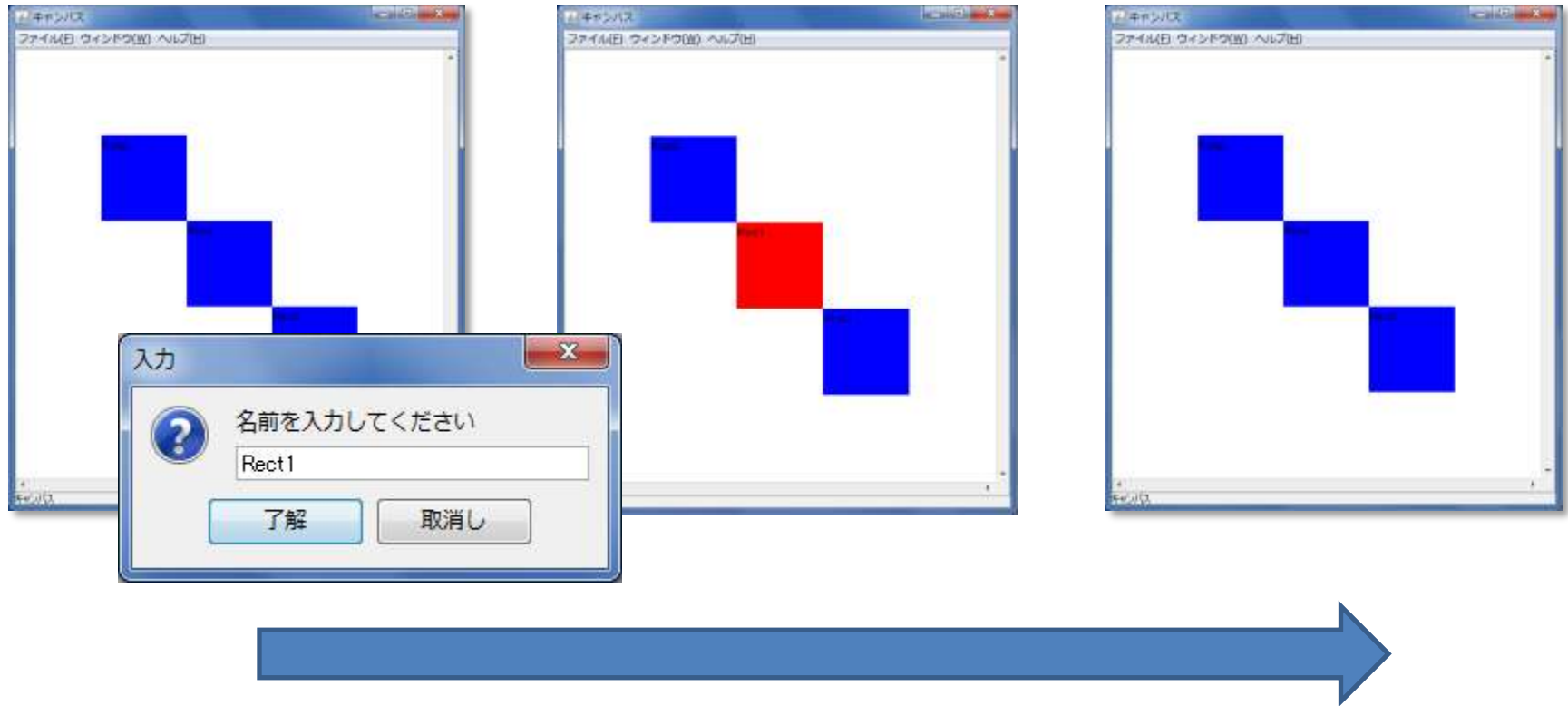
■ 問題31(1)

以下のgetメソッドをQ31RectangleListの中に作成せよ。
またユーザに名前の入力で長方形を指定させ、指定した長方形を赤色でフラッシュさせよ。

戻り値の型	メソッド名(引数)	機能
NamedRectangle	get(NamedRectangle [] list, String name)	長方形リストlist中に格納されているNamedRectangleインスタンスのうち名前nameを持つものを一つ返す。該当する名前が見つからない場合は、nullを返す。

(クラス名: Q31RectangleList)

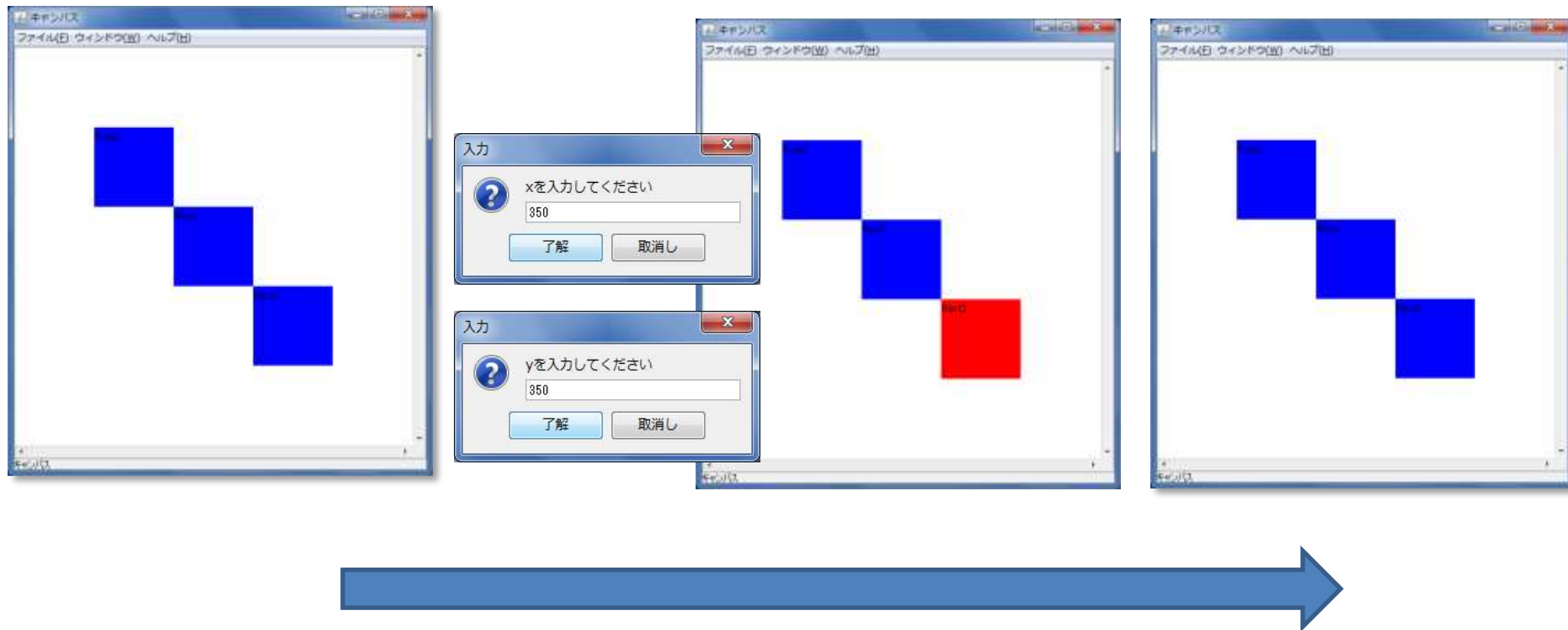
問題31(1)実行例



(クラス名: Q31RectangleList)

■ 問題31(2)

ユーザーに座標を指定させ、その座標を四角形の中に含む長方形を赤色にフラッシュさせよ。



(クラス名: Q31RectangleList)

■ 問題31(3)

マウスでクリックした長方形を赤色にフラッシュさせよ。
マウスからの入力に関しては、次頁を参考にせよ。

(クラス名: Q31RectangleList)

マウスからの入力を取得(例1)

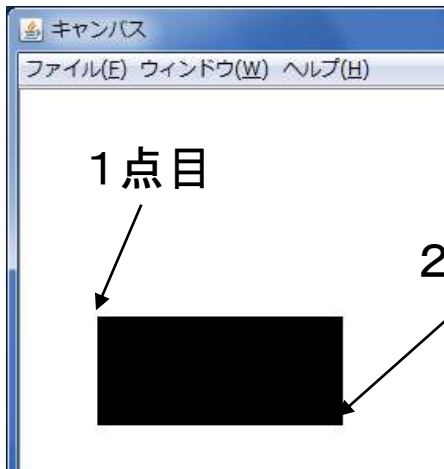
```
1 package j2.lesson08;
2
3 import gpjava.Canvas;
4
5 public class MouseInputExample1 {
6     public static void main(String[] args) {
7         new MouseInputExample1().start();
8     }
9
10    void start() {
11        Canvas.show();
12        Canvas.waitForPoint("キャンバス内で、マウスをクリックしてください。");
13        int x = Canvas.getPointedX();
14        int y = Canvas.getPointedY();
15        Canvas.drawString(x, y, "(" + x + ", " + y + ")");
16    }
17 }
```

「キャンバス内で、マウスをクリックしてください。」というメッセージがコンソールに表示され、キャンバス上でマウスがクリックされるのを待つ。(クリックされると、先に進む)

Canvas.getPointedX()と、Canvas.getPointedY()で直前に押されたキャンバス上での座標を取得できる。この場合の表示例は、
(65,60)

マウスからの入力を取得 (例2)

```
1 package j2.lesson08;
2
3 import gpjava.Canvas;
4
5 public class MouseInputExample2 {
6     public static void main(String[] args) {
7         new MouseInputExample2().start();
8     }
9
10    void start() {
11        Canvas.show();
12        Canvas.waitForPoint("1点目を入力してください");
13        int x = Canvas.getPointedX();
14        int y = Canvas.getPointedY();
15        Canvas.waitForPoint("2点目を入力してください");
16        int width = Canvas.getPointedX() - x;
17        int height = Canvas.getPointedY() - y;
18        Canvas.fillRect(x, y, width, height);
19    }
20 }
```



マウスのクリック待ちの際に、コンソールに表示したいメッセージを引数で与える。