

Java プログラミング入門

— グラフィックの描画：四角・円 —

早稲田大学

グラフィックの描画

グラフィックの描画

- Frame 環境を用いたプログラムにより絵を描く
- 雛形の説明及び Graphics クラスに属するメソッドの紹介

雛形

- 雛形をエディタに貼り付ける
- 保存 コンパイル 実行
- 枠（フレーム）が出て来たら OK

雛形 (Hina01.java)

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

public class Hina01 extends JFrame{

    public Hina01(){
        setSize(500,500);
        setTitle("Java Programming");
        setDefaultCloseOperation(EXIT_ON_CLOSE);
        MyJPanel myJPanel= new MyJPanel();
        Container c = getContentPane();
        c.add(myJPanel);
        setVisible(true);
    }

    public static void main(String[] args){
        new Hina01();
    }

    public class MyJPanel extends JPanel{
        public MyJPanel(){

        }
        public void paintComponent(Graphics g){

        }
    }
}
```

フレームの初期設定

```
setSize(500,500);  
// 画面のサイズを設定する  
  
setTitle("Java Programing");  
// フレーム上の文字を定義する  
  
setDefaultCloseOperation(EXIT_ON_CLOSE);  
// 終了ボタンを有効にする  
  
MyJPanel myJPanel= new MyJPanel();  
// パネルを貼り付ける  
  
Container c = getContentPane();  
// コンテナの取得  
  
c.add(myJPanel);  
// パネルを貼る  
  
setVisible(true);  
// 画面を表示する
```

描画の部分

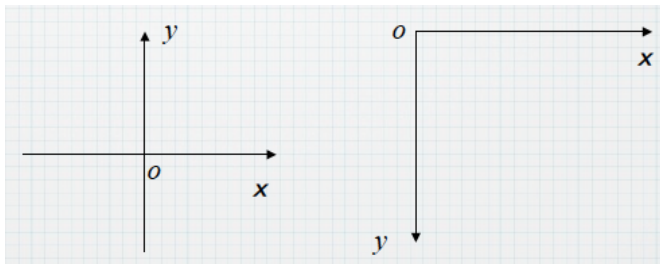
```
public void paintComponent(Graphics g) {  
  
}
```

- Graphics クラスの変数 `g` を用意している
- グラフィックスクラスには様々なメソッドが用意されている

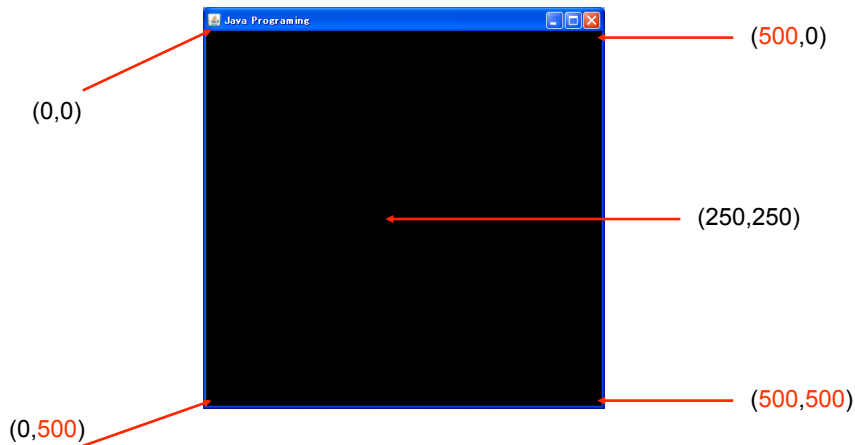
コンピュータの座標はデカルト座標とは異なる：

コンピュータの座標

- 正の整数のみで定義される
- ドット（ピクセル）単位であり，連続ではない
- y 軸の向きが逆
- 原点が左上



コンピュータの座標 (500 × 500 の場合)



利用できる範囲は実際には500より少々小さいことに注意

直線，長方形を描く

```
public void paintComponent(Graphics g) {  
    g.drawLine(100,100,200,200); //直線を引く  
    g.drawRect(100,200,200,300); //長方形を描く  
    g.fillRect(100,300,200,400); //塗りつぶした長方形を描く  
}
```

引数はそれぞれ

- g.drawLine(**開始点の座標**, **終了点の座標**);
- g.drawRect(**開始点の座標**, **幅**, **高さ**);
- g.fillRect(**開始点の座標**, **幅**, **高さ**);

楕円を描く，塗りつぶす

```
public void paintComponent(Graphics g) {  
    g.drawOval(100,100,200,200); //楕円を描く  
    g.fillOval(300,100,400,200); //塗りつぶした楕円を描く  
}
```

引数はそれぞれ

- g.drawOval(**開始点**, **幅**, **高さ**);
- g.fillOval(**開始点**, **幅**, **高さ**);

for 文を使うと

```
public void paintComponent (Graphics g) {  
    int i;  
    for (i=100;i<=300;i+=10)  
        g.drawLine(100,i,i,300);  
    }  
}
```

- このプログラムを実行する前に，結果を想像してみよう

Graphics クラス

- Graphics クラスには、どのような
 - メソッド
 - メンバ変数があるかを各自確認してみよう。

参考：

<http://docs.oracle.com/javase/1.5.0/docs/api/java/awt/Graphics.html>