

# Graphics with Processing



2010-01 Processing

<http://vilab.org>

塩澤秀和

# 1.1 Processing

---

## Processingとは？

- 特徴
  - グラフィックに適した手軽な言語 (C/C++でCGをやるより簡単)
  - 芸術家などコンピュータの専門家でない人に配慮
  - 文法はJavaとほとんど同じ (実体はJava+“便利な機能”)
- Webサイト
  - <http://processing.org>
  - Exhibition(展示室)は必見
- 今後役に立つの？
  - OpenGL, DirectX (C/C++)でも考え方は同じ
  - Javaのライブラリとして使える

## 参考資料

- インストール方法
  - 1.4 演習課題の手順に従うこと
- サンプルプログラム集
  - File → Examples
- マニュアル
  - Help → Reference
  - 日本語訳  
<http://www.technotype.net/processing/reference/>
- 参考文献
  - 『Built with Processing デザイン/アートのためのプログラミング入門』, 前川, 田中著, (株)BNN
  - 『ビジュアルライジング・データ — Processingによる情報視覚化手法』, Ben Fry著, オライリージャパン

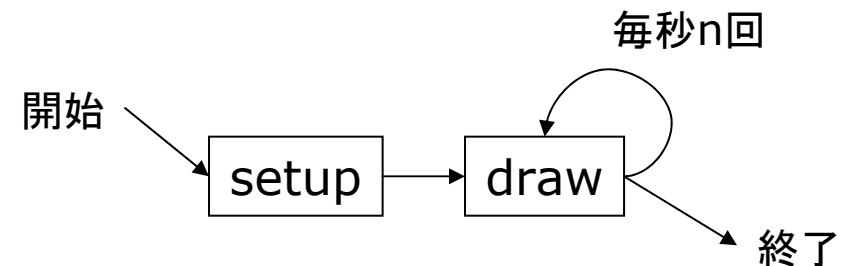
## 1.2 簡単なプログラム

### プログラムの基本構造

- サンプルプログラム
  - Examples → Basics → Structure → SetupDraw
  - コメントはC++やJavaと同じ
- void setup()
  - 初期設定関数(メソッド)
  - プログラム開始時に、自動的に1回だけ実行される
- void draw()
  - 描画関数(メソッド)
  - 画面上で描画をする必要があるときに自動的に実行される
  - アニメーション(loop)の場合、繰り返し何回も実行される
  - 他にも再描画が必要なとき実行

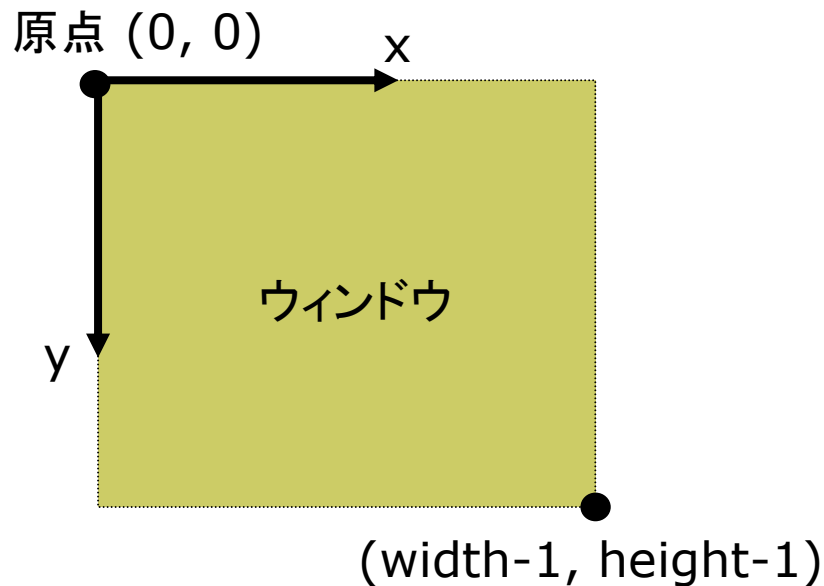
### 基本的なsetup処理

- size(幅, 高さ)
  - ウィンドウを開く
  - setupの最初に必要
- frameRate(回数)
  - 1秒間あたりの描画コマ数
  - アニメーション(動画)に利用
  - 省略した場合は毎秒60コマ
- noLoop()
  - アニメーションを無効化する
  - loop()で解除できる



# 1.3 基本的な描画

## Processingの座標系



## 色の数値表現

- 白黒(グレー)
  - 0~255の整数
- カラー
  - 0~255の整数(RGB) × 3個
  - 例: `stroke(r, g, b)`

## 描画の準備

- `background(色)`
  - 背景色でウィンドウを塗りつぶす
  - 通常, `draw`の最初にやる
- `stroke(色)`
  - 線の色を指定する
- `strokeWeight(太さ)`
  - 線の太さを指定する

## 基本図形

- `point(x座標, y座標)`
  - 点を打つ
- `line(x1, y1, x2, y2)`
  - 直線を引く
- `rect(x, y, 幅, 高さ)`
  - 長方形(矩形)を描く

## 1.4 演習課題

---

### Processingのインストール

- ダウンロード
  - <http://vilab.org/cg2010/>
  - Javaセット版(推奨)  
processing-1.2.1.zip
- 起動
  - ZIPを適当な場所に展開して、  
processing.exe を実行
  - ユーザ名が日本語だとダメかも
- 機能
  - (>)ボタンでプログラム実行
  - [↓]=保存, [↑]=読込など
- 課題
  - サンプルプログラムを参考にして、  
簡単な図形を描いてみよう

### 課題の提出練習

- 提出URL  
<http://vilab.org/upload/cg-upload.html>
- 提出方法
  - フォームにクラス・番号等を入力
  - 参照でソースファイルを選択  
(マイ)ドキュメント¥Processing  
¥スケッチ名¥スケッチ名.pde
  - ファイルの種類は「pde」のまま
  - [送信]を押して提出
- 確認方法
  - フォームにクラス・番号等を入力
  - ファイル欄を空欄のまま[送信]
  - 提出したプログラムがあればOK