

不透明度、文字列、画像ファイル、座標変換

基礎グラフィック

PROCESSING FRIENDS

今週までの内容

データ

乱数

変数

setupとdraw

色

制御構造

逐次処理

分岐処理

繰り返し処理

今日の内容

色の不透明度

文字列の表示

画像ファイルの保存

座標変換

時間があれば
一部だけでも

色の指定

colorMode(RGB)

コンピュータから見ると
RGBによる指定が自然

R:赤 (Red)
G:緑 (Green)
B:青 (Blue)

colorMode(HSB, 359, 99, 99)

人間から見ると
HSBによる指定が自然

H:色相 (Hue)
S:彩度 (Saturation)
B:明度 (Brightness)

色相、彩度、明度

色相(Hue)

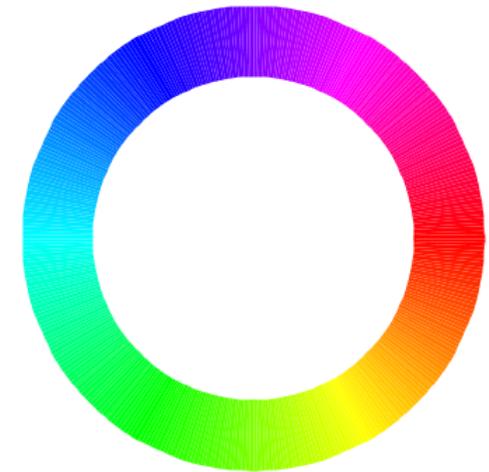
- 赤、黄、青などの「色合い」
- 0～359

彩度(Chroma or Saturation)

- 「あざやかさ」の度合い
- 0～99

明度(Value or Brightness)

- 「明るさ」の度合い
- 0～99



colorMode

colorMode(RGB) <- デフォルトの色指定

- RGBの値を0～255
- colorMode(RGB,100) <- 0～100の数値で指定

colorMode(HSB,359,99,99);

- H:0～359
- S,B : 0～99
- colorSelectorの数値
- colorMode(HSB) <- HSBの値を0～255で指定

デフォルト (default) : 何も指定しないときに使われる値など

色を表すデータ型

color型 (RGBでもHSBでもどちらでも利用できる)

```
color c1 = color(10,100,255);
```

```
color c2 = color(10,255,255);
```

こんな感じで利用可能

```
background(c1);  
fill(c2);  
stroke(c2);  
c2 = c1;
```

色の指定に利用可能

代入もできる

不透明度（ α チャンネル）

色の情報に不透明度の情報を追加すると、
下の色が透けて見えるようになる

Processingでは0から255の数値で表す



不透明度小

不透明度大

色指定の際の4番目の値が不透明度

不透明度の利用

色々な面白い効果を実現することができる

残像表現

重ね塗り表現

文字列

①

アルファベット、数字や記号などの並び

②

“で始まり”で終わる

③

この”と”の間が処理対象の文字

④

文字列を変数に保存するためには、**String**型を利用する

⑤

半角英数の1文字はchar型として使用することが出来る

文字列

"Kait"

"20180522"

"Kait-1201"

"日本語もOK?"

"readily available"

"!123/*+@AsD"

""

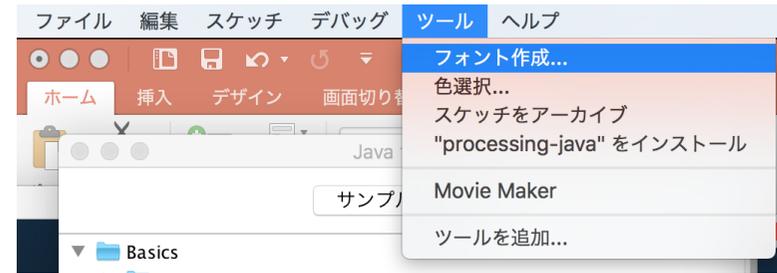
""

Processingのプログラム内で文字列を表示するためには

フォント作成で表示する
フォントの情報を作る

表示するフォントの情報を
PFont型変数に読み込む

表示に利用するフォントを
指定する

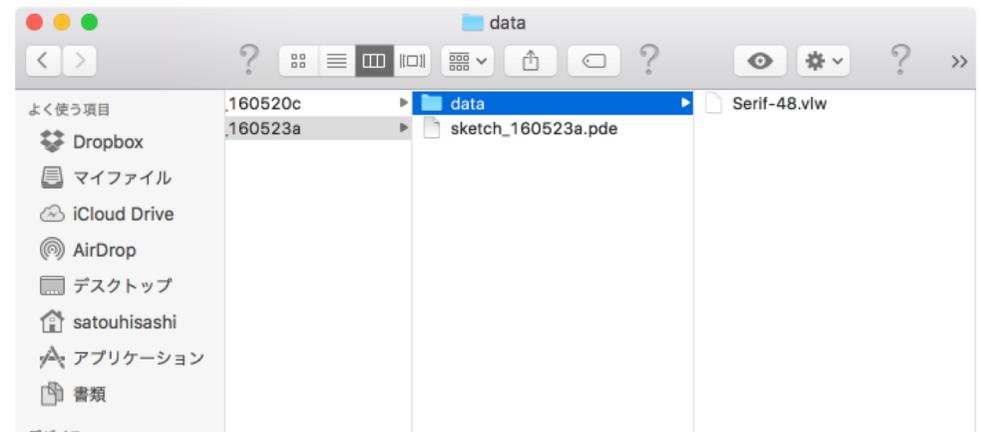
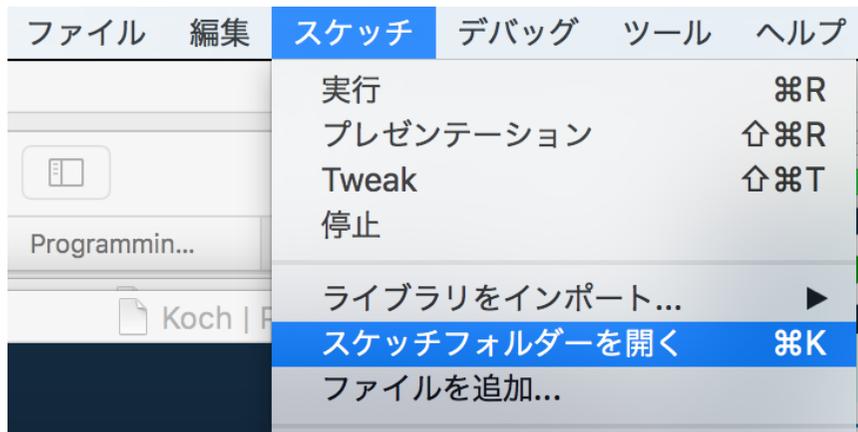


loadFont関数を使用

textFont関数を使用

「スケッチフォルダーを開く」で表示されるフォルダ内のdataに保存されている

ファイル名は「フォント名-サイズ.vlw」



通常、このdataフォルダ内にスケッチ内で使用する画像ファイルなども保存する。

関数名など	説明
PFont	フォント情報を格納するデータ型
String	文字列を格納するデータ型
loadFont(file)	引数fileで指定されたvlwファイルを読み込む関数
textFont(f)	PFont型の引数fで指定したフォントを表示に利用する
textFont(f,size)	PFont型の引数fで指定したフォントを大きさsizeで表示に利用する
text(str,x,y)	引数strで指定された文字列を位置(x,y)に表示する関数
text(str,x,y,w,h)	引数strで指定された文字列を位置(x,y)、幅w、高さhの長方形の内部に表示する関数
textSize(size)	表示に利用するフォントの大きさをsizeに設定する関数
createFont(fname,size)	大きさsizeで引数fnameで指定したフォント情報を作成する。

textAscent

Anegasaki

textDescent

textWidth



関数名など	説明
<code>textAlign(CENTER)</code>	指定した長方形領域に、文字列を中央揃えで表示するようにする。
<code>textAlign(LEFT)</code>	指定した長方形領域に、文字列を左揃えで表示するようにする。
<code>textAlign(RIGHT)</code>	指定した長方形領域に、文字列を右揃えで表示するようにする。
<code>textWidth(str)</code>	現在の表示文字設定で、文字列strを表示した時の幅を求める関数
<code>textDescent()</code>	現在の表示文字設定で、文字列を表示した時のベースラインからどれだけ下に表示されるかを求める関数。
<code>textAscent()</code>	現在の表示文字設定で、文字列を表示した時のベースラインからどれだけ上に表示されるかを求める関数。

簡単な文字列演算

Str関数で文字列に変換

文字列同士の+演算は文字列の連結（結合）

ウィンドウの内容を保存

save関数

save関数を実行する時点でのウィンドウの内容を画像ファイルとして保存する

saveFrame関数

draw関数を実行するたびに連番の画像ファイルとしてウィンドウの内容を保存する

画像ファイルの表示

①

画像ファイルの読み込み
loadImage関数を使う

②

画像データはPImage型の変数に保存する

③

画像ファイルの表示
image関数を使う

image関数

image(PImage型変数,表示位置のX座標,表示位置のY座標)

image(PImage型変数,X座標,Y座標,幅,高さ)

imageMode関数 : rectModeと同じ、画像を表示する際の座標指定を変更

PImage型変数.width : 画像の幅

PImage型変数.height : 画像の高さ

 PImage型変数.width : 画像の幅

 PImage型変数.height : 画像の高さ

今日やった内容

HSBと不透明度

文字列の表示

画像ファイルの保存と表示

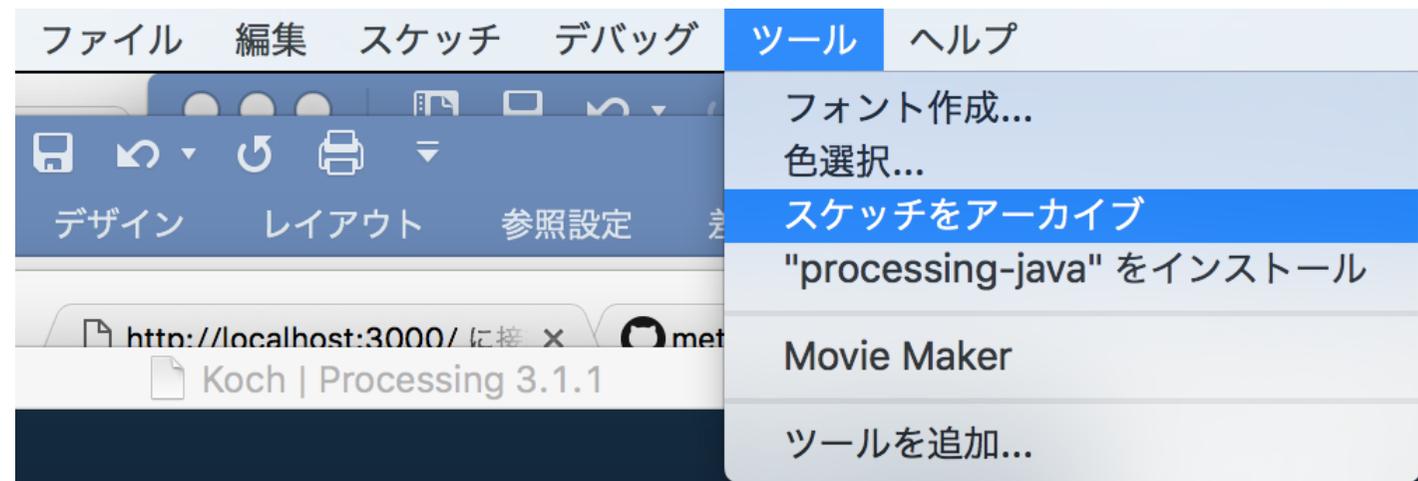


今日の内容は完全に理解できますよね。

クイズの提出に関して

キャリアポートフォリオ上で解答して下さい。

問1、2は「スケッチをアーカイブ」で保存されるファイルをアップロードして下さい。



授業時に配布した資料

<http://www.sato-lab.jp/imfu/index.html>

にあります。