

プログラミング入門

第3回の補足資料

理工学部 情報科学科 隅田 康明

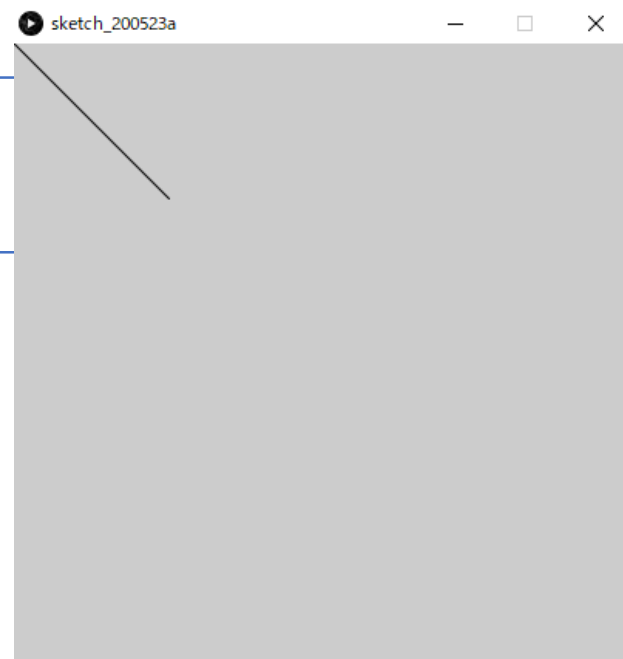
pp@is.kyusan-u.ac.jp
(質問はここに)

sumida@ip.kyusan-u.ac.jp
(出席などについての問い合わせはこちら)

Processing最初の一步

- 下記のプログラムを入力して実行

```
size(400,400);  
line(0, 0, 100, 100);
```



線が一本引かれるのを確認したら次へ

線が引かれなければ打ち間違っている、よく見直そう

プログラムの説明 : `size(400,400);`

- `size(横のピクセル数, 縦のピクセル数);`
 - Processingを実行した時の、
実行画面の大きさを設定する命令

今書いたプログラムの解説

```
line( 0 , 0 , 100 , 100 );
```

線を引く

始点の座標

終点の座標

一つの命令
の終わり

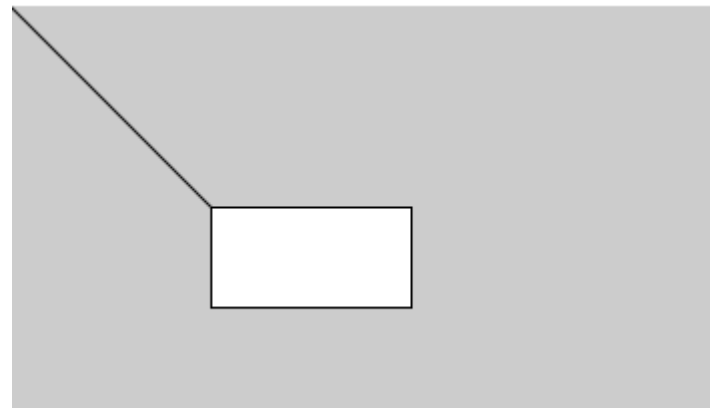
- 意味：(0, 0) から (100,100)をつなぐ線を引く
- 一つの命令を書いたら、[;](セミコロン)を記述
 - 文章を。で区切るようなイメージ
- 今回使う命令は、全て“メソッド”という種類の命令

命令には、それぞれ意味があり、厳格にルールが決められている

他の図形も描いてみる

- `line(0, 0, 100, 100);` の下に長方形を描いてみる

```
line( 0, 0, 100, 100 );  
rect(100,100,100,50);
```



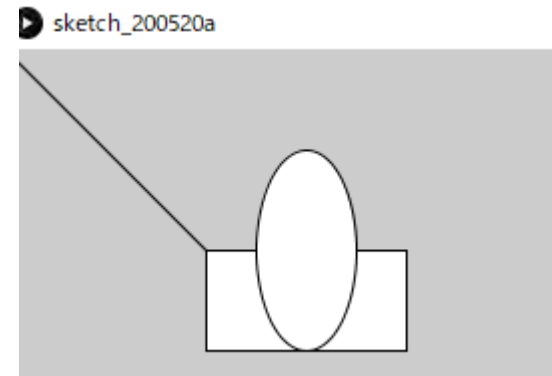
- `rect(100,100,100,50);`
 - `rect(左上x,左上y,幅,高さ);` は長方形を描く命令
 - `(100,100)`を左上に
横100、縦50の大きさの長方形を描く
 - 縦の方が小さいので、横長の四角形になる

数値を変えて、図形の形や位置がどう変わるのか試してみよう

更に図形を追加してみる

- 円を追加

```
line( 0, 0, 200, 200 );  
rect(100,100,100,50);  
ellipse(150,100,50,100);
```
- `ellipse(150,100,50,100);`
 - `ellipse(中心x,中心y,幅,高さ);` は円を描く命令
 - (150,100)を左上に
横50、縦100の大きさの長方形を描く
 - 横の方が小さいので、縦長の楕円になる



位置の指定：座標

```
line(100,100,200,200);  
rect(200,200,100,100);
```

こんな命令をしたとする

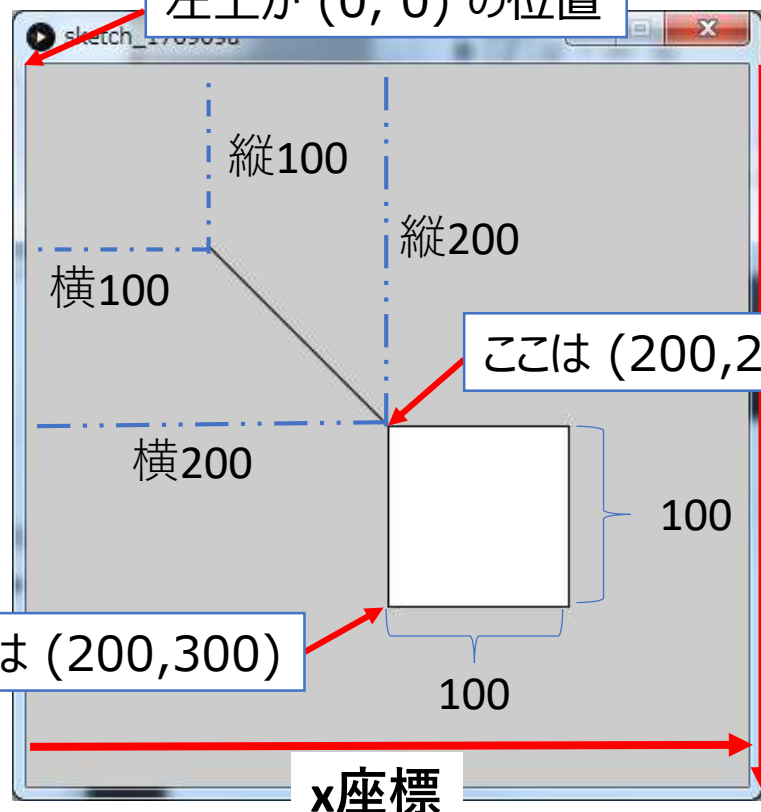
左上が (0, 0) の位置

x軸は横、y軸は縦（数学のグラフと同じ）
（ただし、yの向きは逆（数値が大きいほど下））

一番左が x座標 0
一番上が y座標 0

Processingでは、
位置は0から数え始める

右に何歩、下に何歩と
移動してから描くイメージ



（例） 楕円を描く命令

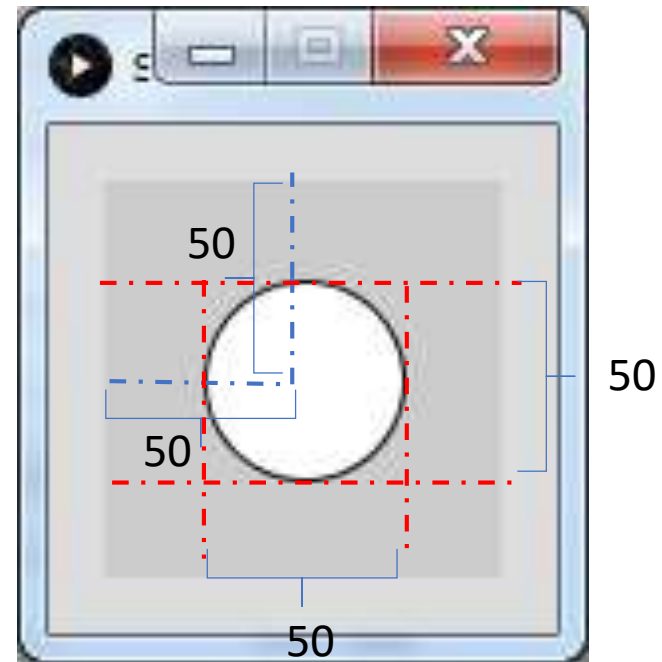
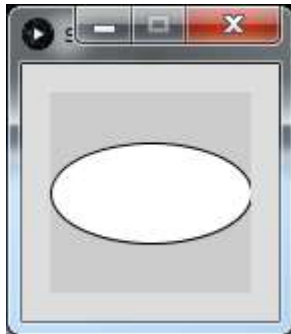
ellipse(中心のx,中心のy,幅,高さ);

ellipse(50,50,50,50);

(50,50)を中心に、幅が50、高さが50の円を描く

ellipse(50,50,100,50);

幅が100、高さが50の円 = 楕円



命令にはそれぞれ意味がある

- `line(100,100,100,100);`
`rect(100,100,100,100);`
`ellipse(100,100,100,100);`
- カッコ内の数値は , (カンマ) で区切って4つずつ
 - 命令によって、いくつ値を書いていいかも決まっている
 - しかし、描かれる図形は全く違う
- `line(始点x, 始点y, 終点x, 終点y);`
`rect(左上x, 左上y, 幅, 高さ);`
`ellipse(中心x, 中心y, 幅, 高さ);`

このプログラムは書かなくてもいい（既にあるellipseの命令を書き換えてみよう）

プログラムの**逐次実行**

- プログラムは 1 行目から順々に実行(例外もある)

size(400,400);

line(100,100,200,200);

ellipse(200,200,100,100);

rect(200,200,100,100);

画面の大きさを変える

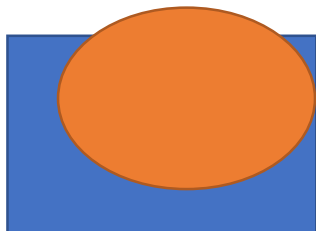
線を引く

丸を描く

四角を描く

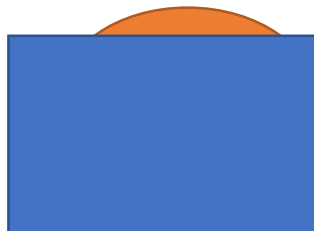
1.四角を描く

2.丸を描く



1.丸を描く

2.四角を描く

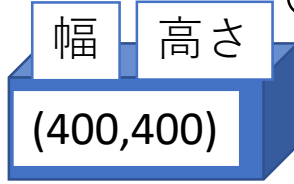


下に書いた命令の図形が、
上に重なっていく

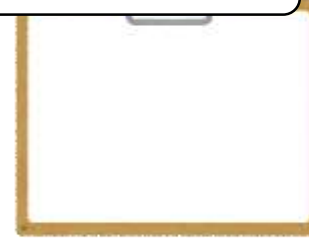
上から順に、一つずつ実行していく

```
void setup(){  
  size(400,400);  
  line(0,0,50,50);  
  rect(50,50,50,100);  
}
```

sizeさん



(400,400) のキャンバス作ったよ



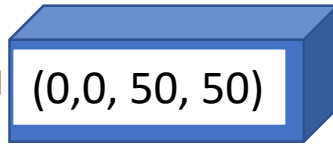
次よろしく

幅と高さの2つの数値を受け取ると、
その幅高さの描画領域を作る

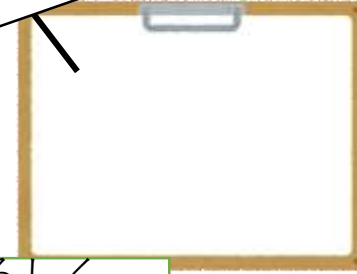
lineさん



(始点x, 始点y, 終点x, 終点y)



(0,0) から (50,50) に線を引いたよ



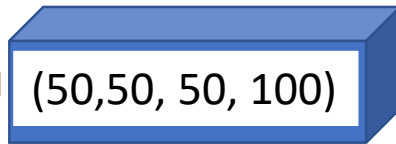
次よろしく

始点座標と終点座標を受け取ると、
その2点を結ぶ線を描いてくれる

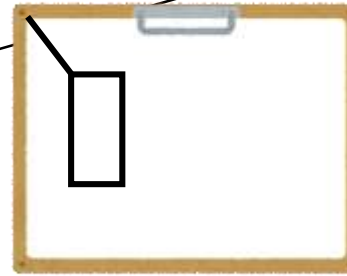
rectさん



(左上x, 左上y, 幅, 高さ)

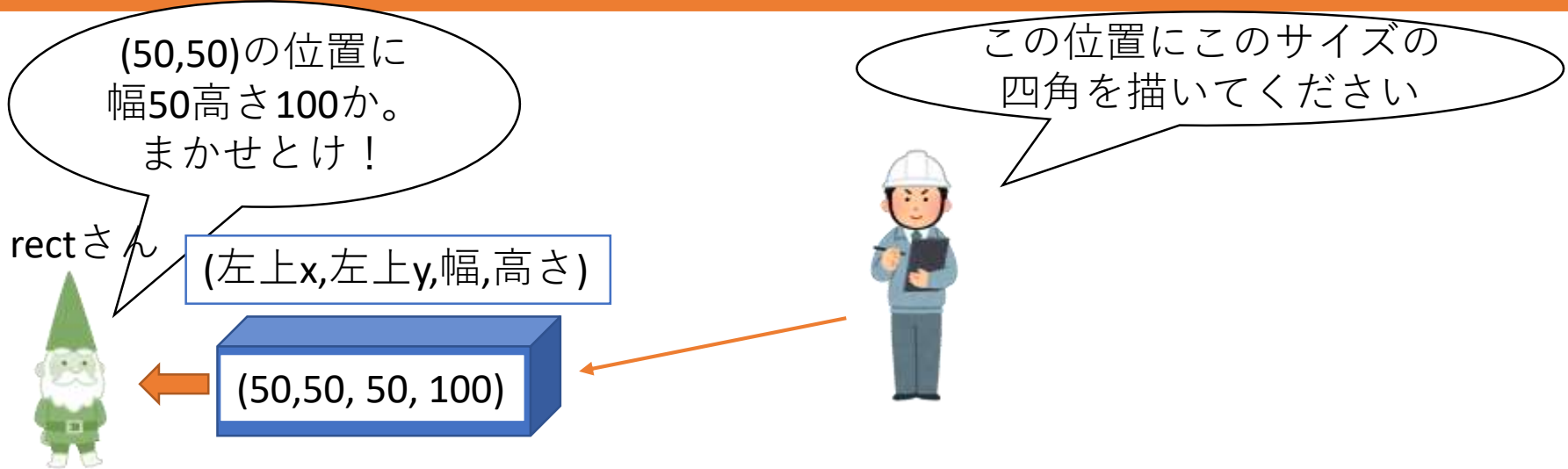


(50,50) に幅50, 高さ100の四角を描いたよ

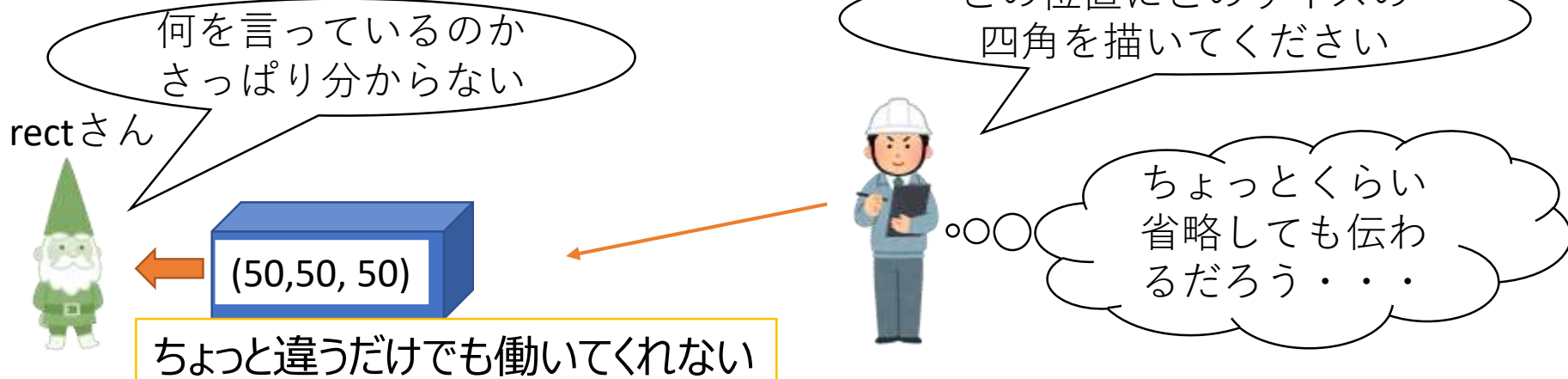


左上の頂点の座標と幅と高さを受け取ると、
左上座標を基点に、その幅と高さの長方形を描いてくれる

メソッドを書く(呼び出す)ときの決まり： ルールを守って命令を出そう



命令の出し方を間違えると・・・

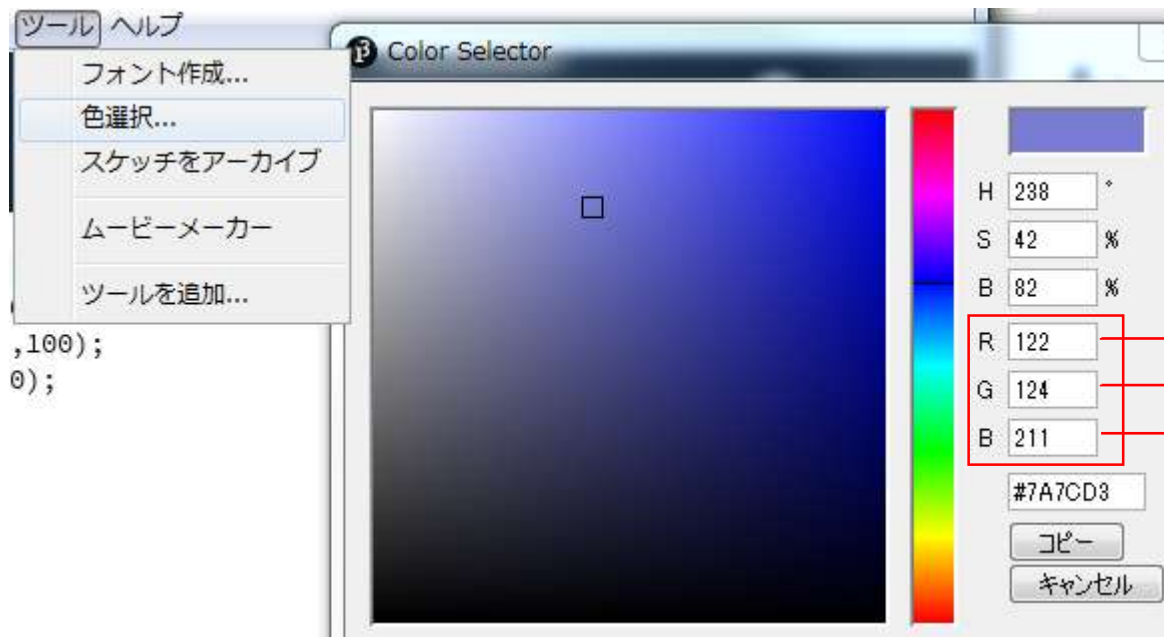


図形に色を付ける

- 何も指示しないと、線の色は黒色、塗りつぶしは白色になる
- 色を変えたければ、何色にするか**命令する**必要がある
- 線の色を変える命令：**stroke(赤, 緑, 青);**
 - それぞれ、0～255の数値で入力する
- 塗りつぶしの色を変える命令：**fill(赤, 緑, 青);**
 - それぞれ、0～255の数値で入力する
- 線を書かない：noStroke();
- 塗りつぶさない：noFill();

思い通りの色を作れない(PCの場合) カラーセレクトで色を選ぶ

- RGBの数値だけだと、何色になるのか分かりにくい
- [ツール]メニューの「色選択…」



`stroke(122,124,211);`

思い通りの色を作れない(その他の場合) 色の見本から選ぶ

- 色の見本から選ぶ
 - Webで使えるカラーネームと、そのカラーコード・RGB値一覧
 - <https://www.webcreatorbox.com/webinfo/color-name>
 - 注意：上記の参考サイトに書いてある、
rgb(255, 255, 224) などは、
Processingのプログラムではありません。
 - 参考にする場合には、R,G,Bの数字だけを、
stroke() や fill() の中に入れて使いましょう

図形を描く手順

1. strokeやfillで線、塗りつぶしの色を指定
2. 図形を描画
3. この繰り返し

```
stroke(255,0,0); //赤色のペンを持つ  
fill(0,0,255); //青色の塗りつぶしブラシを持つ  
rect(10,10,100,50); //長方形を描く
```

1個目の図形

```
stroke(0,0,255); //青色のペンを持つ  
fill(0,255,0); //緑色の塗りつぶしブラシを持つ  
ellipse(100,50,80,50); //楕円を描く
```

2個目の図形

3個目、4個目、5個目・・・と増やしていく

その他の命令（メソッド）

- Processingには、ここまでで紹介した以外にも沢山の命令がある
- 次のページに、よく使う【図形を描く命令】を載せておくので、
 - 命令を実際に書いて、
 - 数値を変えるとどう変わるのか確かめながら、
 - 命令の意味を理解していこう
- 打ち間違い、カッコの中の数値の数、に特に気をつける

手を動かして、書いて、実行してみて、理解する。
最初は上手く行かなくてもいい、まず試してみよう！

命令の意味	命令文	例
HSBモードにする		
線の色を設定する	stroke(赤,緑,青);	stroke(255,0,0);
透明度も設定する	stroke(赤,緑,青,透明度);	stroke(255,0,0,128);
線の太さを設定する	strokeWeight(線の太さ);	strokeWeight(5);
塗り潰しの色を設定	fill(赤,緑,青);	fill(0,0,255);
透明度も設定する	fill(赤,緑,青,透明度);	fill(0,0,255,128);
線を引かない	noStroke();	
塗り潰さない	noFill();	
線を引く	line(始点x, 始点y, 終点x, 終点y);	line(100, 100, 200, 200);
長方形を描く	rect(左上のx,左上のy, 幅, 高さ);	rect(100,100,50,80);
三角形を描く	triangle(x1,y1,x2,y2,x3,y3);	triangle(10,50,30,40,70,90);
四角形を描く	quad(x1, y1, x2, y2, x3, y3, x4, y4);	quad(30, 30, 80, 20, 70, 60, 30, 80);
円を描く	ellipse(中心のx,中心のy, 幅, 高さ);	ellipse(100, 100, 50, 80);
円弧を描く	arc(中心x,中心y,幅,高さ,開始角,終了角)	arc(100,100,50,50,0,PI/2);
文字を書く	text("書きたい文字列",左上のx,左上のy)	text("Hellow",100,50)
文字の大きさを設定	textSize(フォントサイズ);	textSize(24);
点を打つ	point(x座標,y座標);	point(10,10);

プログラムを作っていく上で

- **細かく実行**して、**動くことを確かめる**
 - 1行書いたら実行して確かめる
 - 1行変更したら実行して確かめる
- 少し書いたらすぐ実行して確かめる
 - よくある悪い例
 - 一気に10行くらい書いて実行したら、エラーが沢山出てどこが悪いのか分からなくなった
- **細かく、細かく、実行**してエラーなく動くか確認していこう