

情報リテラシー実習

Exercise in Information Literacy

□プログラミングの基礎

本日は
プログラミングの基礎を
Processingを使って学びます



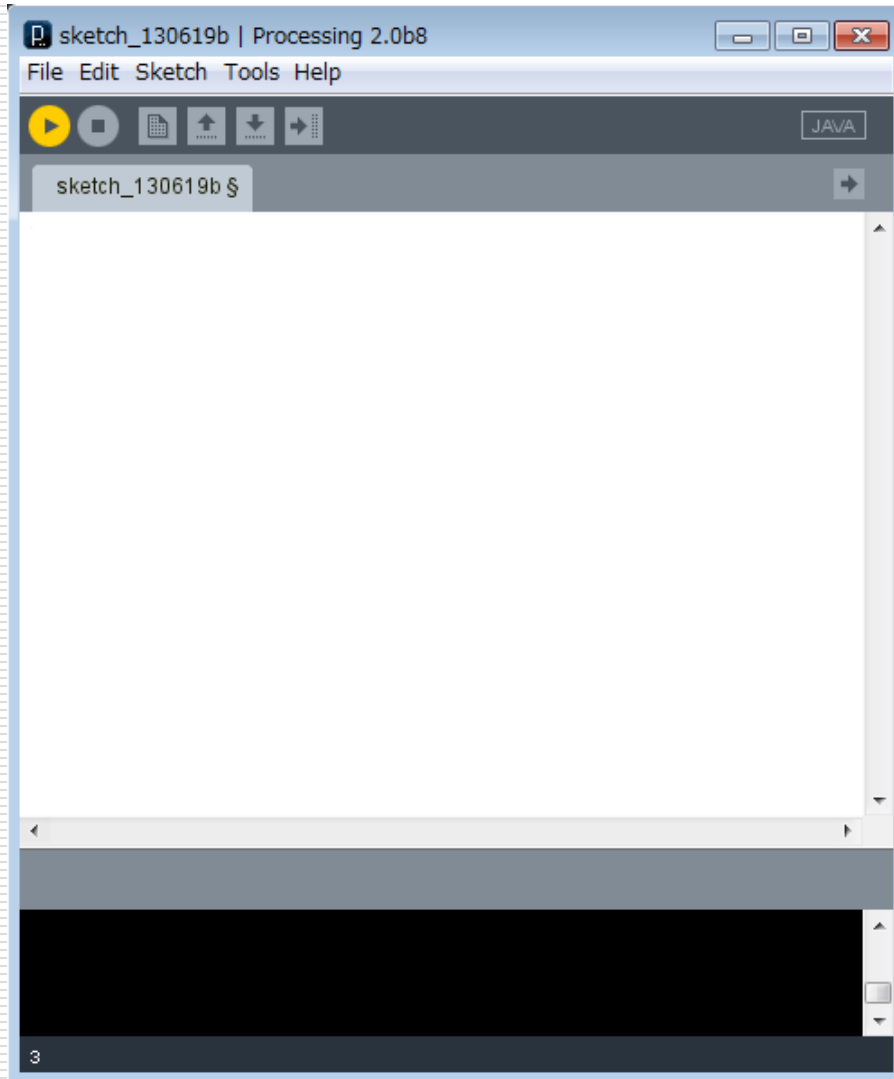
学ぶ項目

- ☐ 逐次
 - ☐ 変数
 - ☐ 分岐
 - ☐ 反復
-

準備

set up

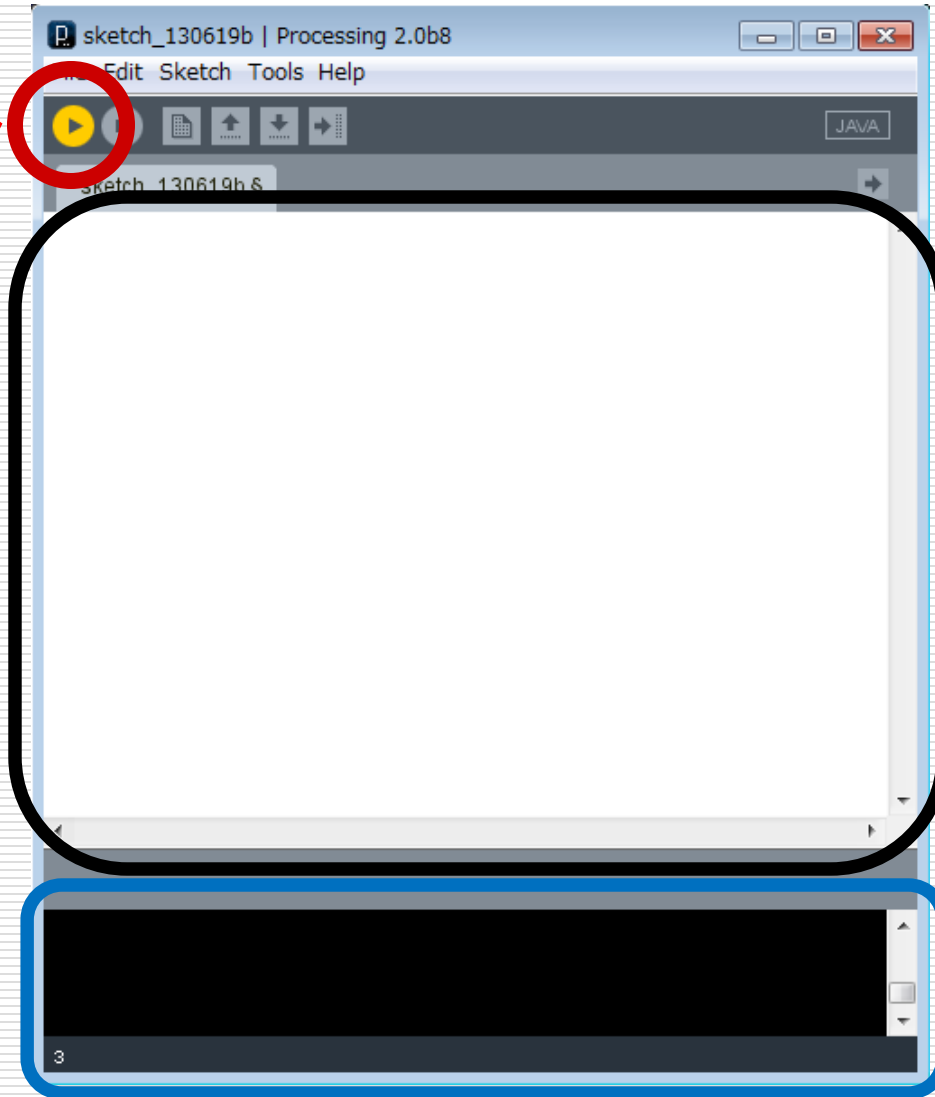
起動



こんなのが
出る・・・はず

部分説明

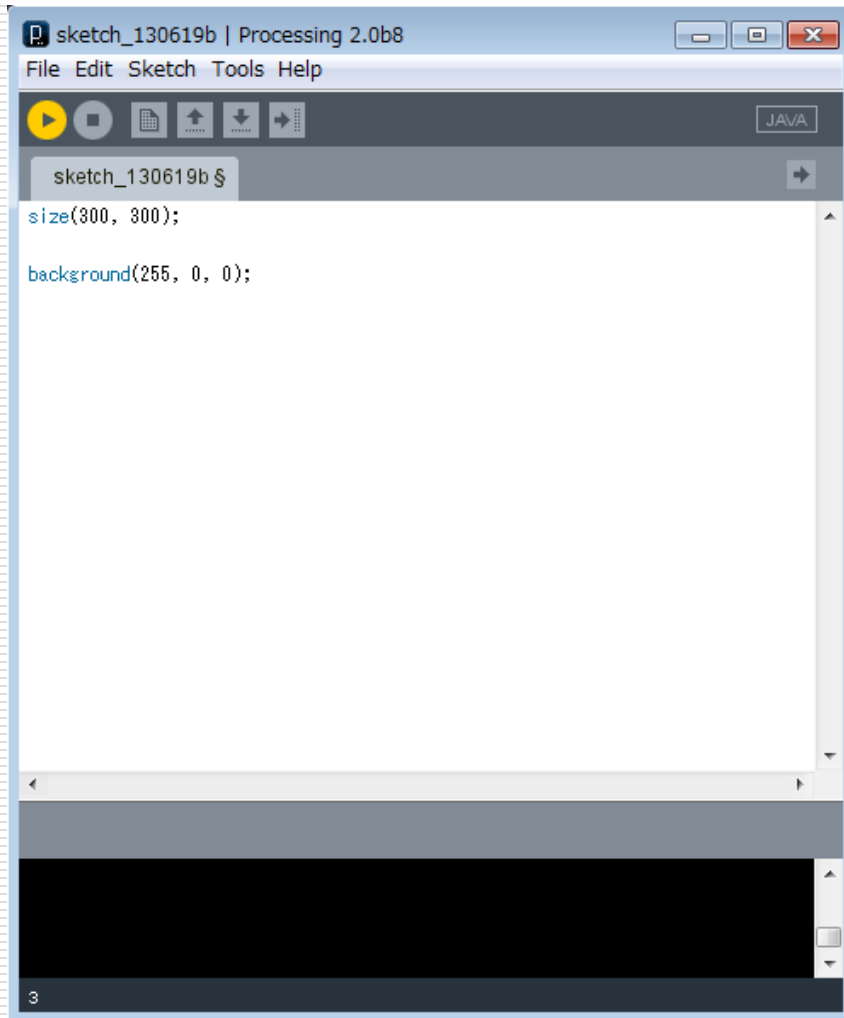
Runボタン



エディタ

テキスト
エリア

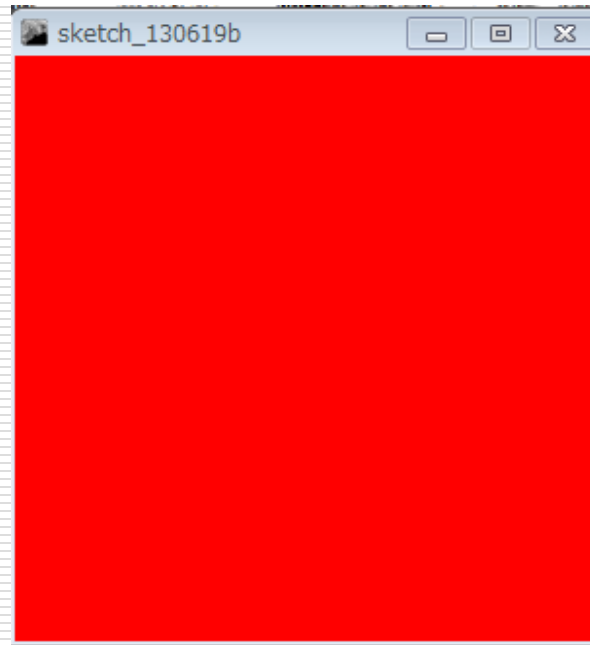
はじめてのプログラム



`size(300, 300);`

`background(255, 0, 0);`

実行結果



ディスプレイウインドウ

プログラムの説明

```
size(300, 300);  
background(255, 0, 0);
```

ディスプレイウィンドウを
300x300のサイズに！

背景の色を赤に！

println

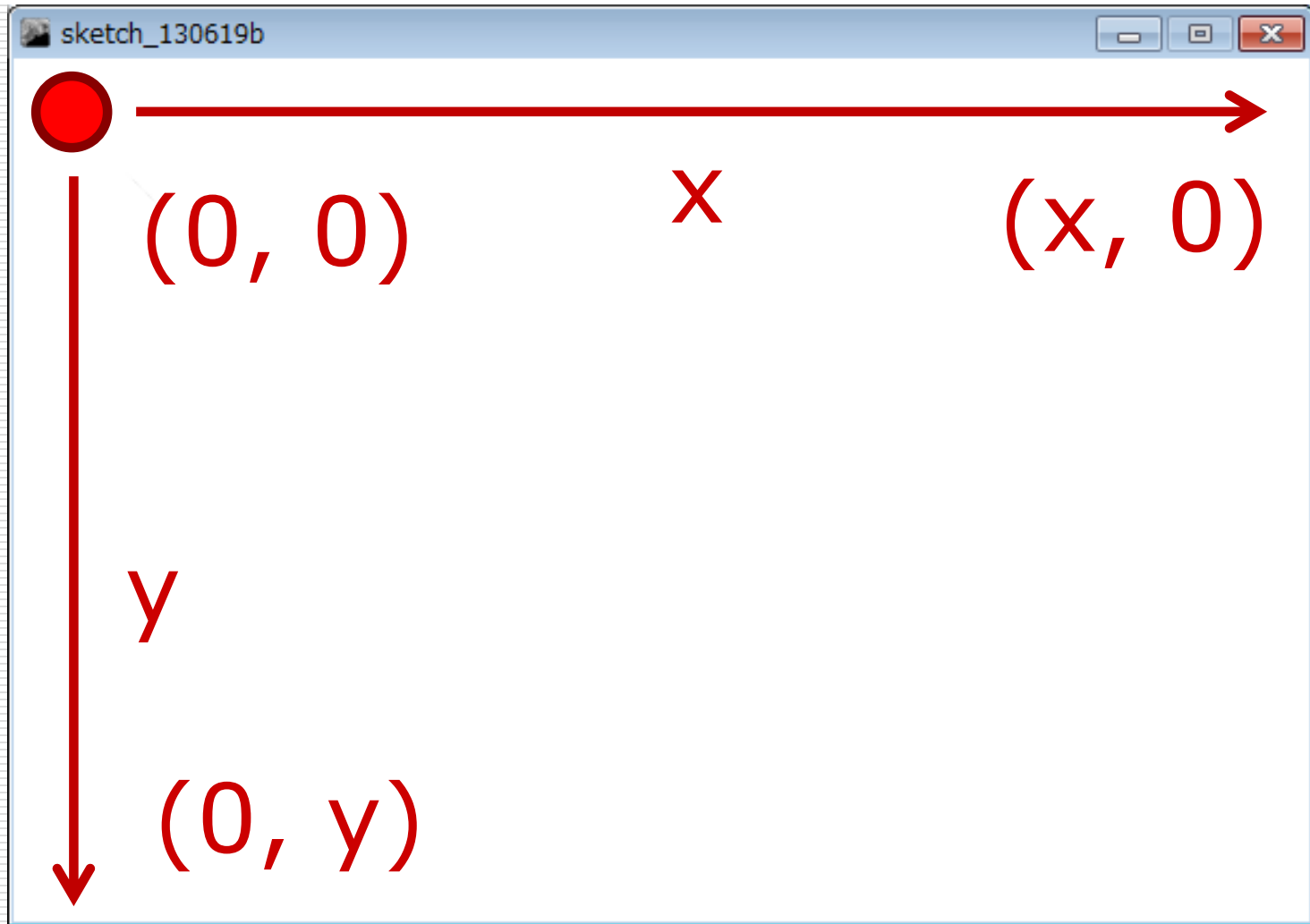
```
println(0);
```

テキストエリアに0が表示される

四則演算

```
println(1 + 2);  
println(3 - 4);  
println(5 * 6);  
println(7 / 3);
```

座標の話



線を引いてみましょう

```
size(600, 400);  
background(255, 255, 255);  
  
line(10, 10, 100, 100);
```

始点の座標

終点の座標

線を引いてみましょう



もっと線を引いてみましょう


準備のまとめ

- Processingの起動方法
 - println
 - 四則演算
 - 座標
 - 線の使いかた
-

逐次

sequence

命令は順番に



```
size(600, 400);  
background(255, 255, 255);  
  
rect(20, 20, 80, 80);  
ellipse(60, 60, 60, 60);  
line(10, 10, 110, 110);  
line(110, 10, 10, 110);
```

上から順番に命令が実行される

命令の順番を変えると？

逐次のまとめ

□ 命令は上から下へ

変数

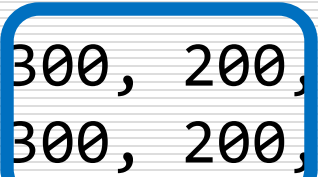
value

変数って？

値を**保存**するのに使う

円を真ん中に2つ書くプログラム

```
size(600, 400);  
background(255, 255, 255);  
  
ellipse(300, 200, 100, 100);  
ellipse(300, 200, 50, 50);
```



ここ同じだし、繰り返して
書きたくないな・・・

円を真ん中に2つ書くプログラム

```
size(600, 400);  
background(255, 255, 255);
```

```
int x = 300;  
int y = 200;
```

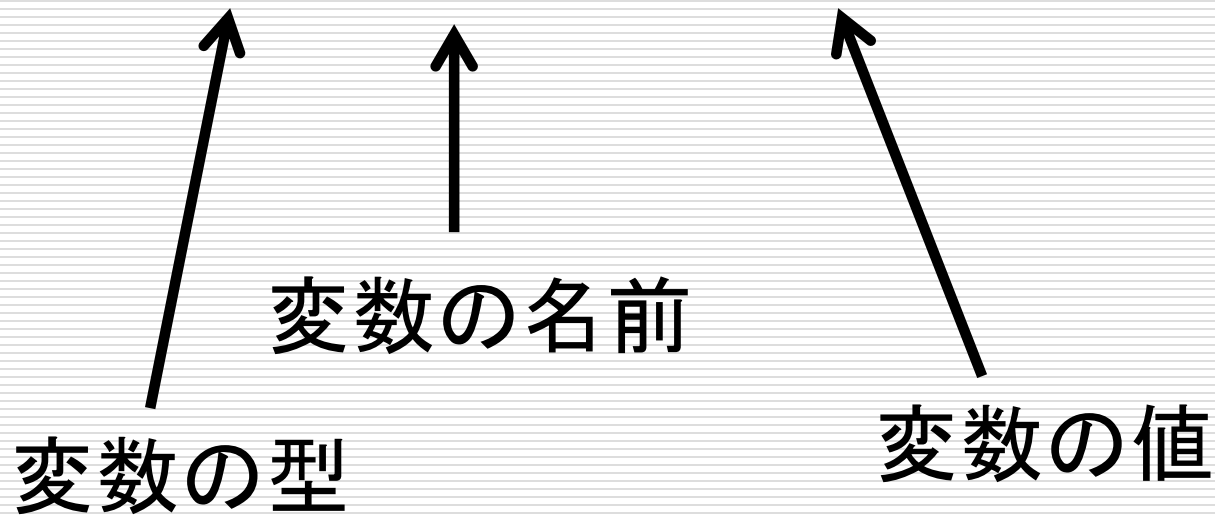
```
ellipse(x, y, 100, 100);  
ellipse(x, y, 50, 50);
```

変数を使う

xに300, yに200という数字が保存

変数の使いかた

int x = 300 ;



変数の使いかた(別バージョン)

`int x;` ← 変数を使うによって宣言

`x = 300;` ← 変数に値を代入

型の種類と再代入

```
size(600, 400);  
background(255, 255, 255);
```

```
int x = 100;  
double y = 3.1415;  
String z = "こんにちは";
```

```
x = 300; ←
```

再代入

```
println(x);  
println(y);  
println(z);
```

変数のまとめ

- 変数は値を保存
 - 変数には型がある
 - 何度でも変数に値を入れられる
-

分歧

branch

条件によって処理を変える

```
size(600, 400);  
background(255, 255, 255);  
  
int x = 10;  
  
if (x % 2 == 0) {  
    ellipse(300, 200, 100, 100);  
}  
else {  
    rect(300, 200, 100, 100);  
}
```

xが偶数か奇数で表示される図が変わる

比較

☐ ==

☐ !=

☐ <

☐ >

☐ <=

☐ >=

☐ かつ(&&)

☐ または(||)

分岐のまとめ

- 分岐は命令の流れを変える

反復

loop

繰り返し...


```
size(600, 400);  
background(255, 255, 255);
```

```
line(100, 10, 500, 10);  
line(100, 60, 500, 60);  
line(100, 110, 500, 110);  
line(100, 160, 500, 160);  
line(100, 210, 500, 210);
```

50ずつ離れた線を引く

while

```
size(600, 400);  
background(255, 255, 255);  
  
int i = 10;  
  
while (i <= 210) {  
  line(100, i, 500, i);  
  i = i + 50;  
}
```



条件が成り立ってる間繰り返す

反復のまとめ

- ある条件が成り立ってる間は命令を繰り返す

演習

exercise

処理を組み合わせてみる

```
size(400, 400);  
background(255, 255, 255);  
  
noFill();  
  
int i = 100, j = 1;  
  
while (i <= 300) {  
  if (j % 2 == 0) {  
    stroke(255, 0, 0);  
  }  
  else {  
    stroke(0,0,255);  
  }  
  ellipse(i, i, 100, 100);  
  i += 20; j += 1;  
}
```

ランダム

```
size(200, 200);  
background(255, 255, 255);
```

```
int i = 0;
```

```
while (i < 1000) {  
  float x = random(200);  
  float y = random(200);
```

```
  point(x, y);
```

```
  i++;  
}
```

かなり高度

```
void setup() {
  size(600,600);
  textFont(createFont("SansSerif",35));
  colorMode(HSB,360,100,100);
}

void draw() {
  translate(width/2,3*height/4);
  background(180,0,75);
  for (int i=0; i<61; i+=4) {
    float x = 0.25 * (-pow(i,2) + 40*i + 1200)*sin((PI*i)/180);
    float y = -0.25 * (-pow(i,2) + 40*i + 1200)*cos((PI*i)/180);
    fill((x +frameCount) % 360,100,100);
    text("♥",x,y);
    text("♥",-x,y);
  }
}
```

調べていろいろやってみよう！

- やりたいこと+processingでググる
- Processing 入門 とかでググる

作品提出方法

masato@cs.kumamoto-u.ac.jpまで
代表者がメールする



メールの内容

- 件名:リテラシー10回のレポート
- グループの学籍番号(半角英数字)と名前
- どこを頑張ったかアピール
- 実行するためのファイル

締切り:次回授業開始まで
