

*Java講座第6回

○×ゲームを作ろう！

* 2次元配列

* array[3][3]

[0][0] [0][1] [0][2]

[1][0] [1][1] [1][2]

[2][0] [2][1] [2][2]

* インスタンス変数

```
1 package java_lectures;
2
3+ import javax.swing.JOptionPane;
4
5
6
7 public class CircleAndX {
8
9     int[][] cOrX = new int[3][3];
10    int turn = 1;
11    boolean result = true;
12
13- public static void main(String[] args){
14    new CircleAndX().start();
15
16    }
17- void start(){
```

- * ローカル変数（メソッド内で宣言した変数）は宣言したメソッド内でのみ使える
- * インスタンス変数は同じパッケージ内なら使える

メソッドに関係なく、クラス内で共有して使える変数！

*準備

- * クラス名は「CircleAndX」
- * 今回使用するインスタンス変数

```
int[][] cOrX = new int[3][3];  
int turn = 1;
```

```
1 package java_lectures;  
2  
3⊕ import javax.swing.JOptionPane;⊔  
6  
7 public class CircleAndX {  
8  
9     int[][] cOrX = new int[3][3];  
10    int turn = 1;  
11  
12⊖ public static void main(String[] args){⊔  
13        new CircleAndX().start();  
14  
15    }⊔
```

*今回使うインスタンス変数

*int turn

現在のターン数。クリック終了後に+1をしていく。
最初は1ターン目なので1としておく。

*int cOrX[3][3]

1ならば○、0ならば×、-1ならば入力なし

右の状態では

$cOrX[0][0] = 1$

$cOrX[0][1] = 0$

その他は-1になっている。

○	×	

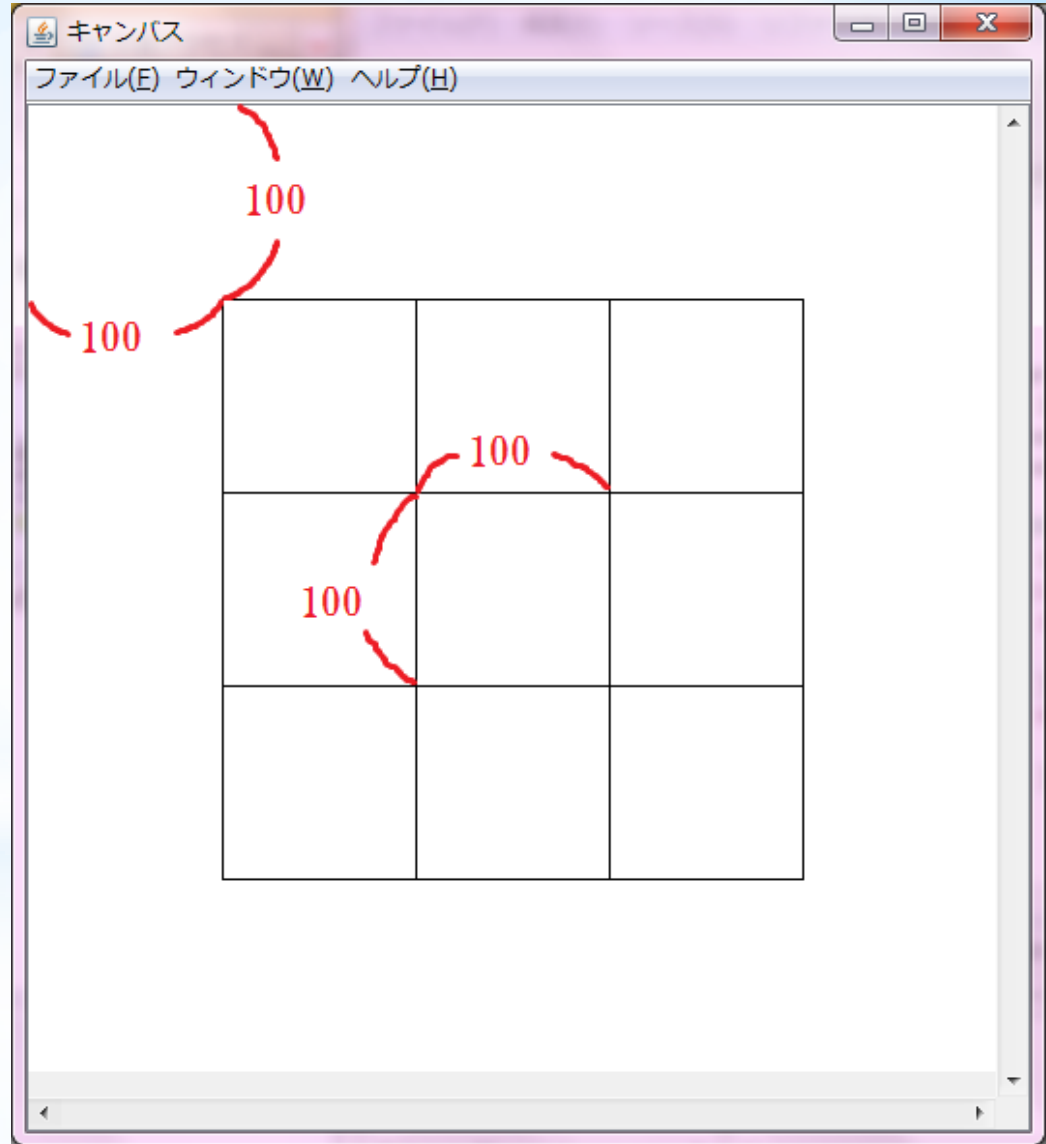
* ボードを描く

* void型メソッド「drawBoard」

※直線を描く

Canvas.drawLine(始点x,
始点y, 終点x, 終点y)

* 終わってますよね～？



* ○と×を描く

- * void型メソッド「drawCOrX(int x, int y, boolean cOrX)」
- * 引数xとy ○か×を描くX,Y座標（配列のインデックス）
(xとyは0~2の値)
- * cOrXによって○か×を描く
trueならば○、falseならば×

※文字を描く

Canvas.drawStringCenter(x座標, y座標, 表示する文字)

* キャンバスのクリック

* キャンバスをクリックして、そのポイントが枠内ならば対応した配列のインデックスを表示させる。

* void型メソッド「click()」

* `Canvas.waitForPoint("")`;

※藤田さんCanvasでは""はいらない
キャンバスがクリックされるまで停止。

* `int x = Canvas.getPointedX()`;

`int y = Canvas.getPointedY()`;

クリックされた座標を返す。

* 枠の中をクリックしたときのみ実行

- ・ 配列のインデックスに対応させたint型変数XとYを使う

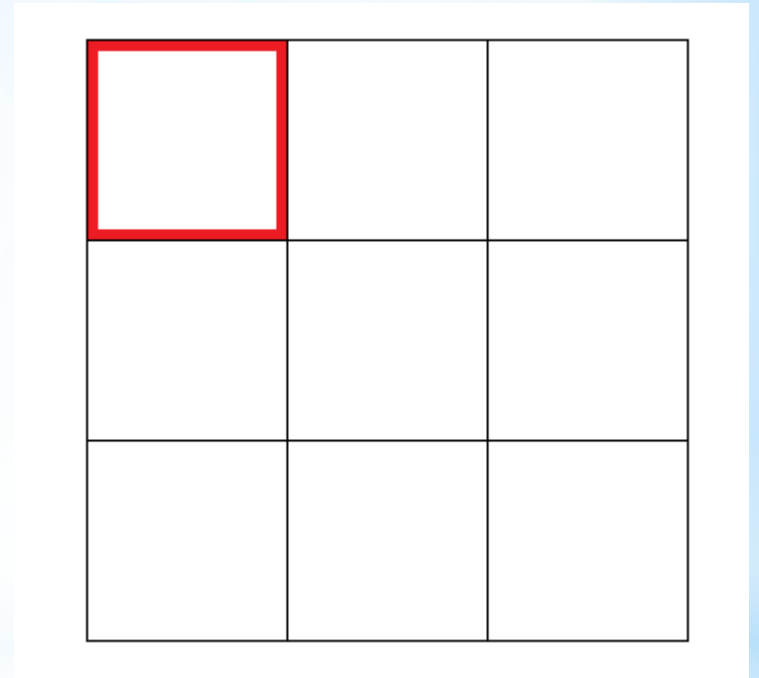
* `System.out.println(X + “,” + Y)`

コンソールに括弧内を表示。

赤枠内をクリックすると

0,0

と表示される。



条件 1

枠内(板状)をクリックしたとき

それ以外は何の実行もしない (このメソッド内のみ)

条件 2

クリックされた配列に○も×も書かれていないとき

それ以外は条件 1 と同様にこれ以降は実行しない

条件 3

○のターンのときか×のターンのときか

配列に○か×の情報を保存

`drawCOrX`を実行

ターン数を1増やす

*実行してみる前に : : :

*配列cOrXに何の値も入っていない . . . !

→clickメソッドのif文でcOrXに値がないとエラーが出てしまうのでは . . . ?

*startメソッドで初期化をしよう!

(全てのcOrXに-1を入れる)

*Canvas.show()でキャンバスを表示

*藤田さんCanvasを使っている人は

Canvas.setColor(0, 0, 0)で黒色を指定しておく!

* 勝敗の確認

- * Boolean型メソッド「check()」
 - * 縦、横、斜めで○か×が揃ったらfalseを返す
 - * また9ターン目が終了したらfalseを返す
 - * それ以外はtrueを返す
-
- * 余裕がある人はヒントを見ずにやってみよう！

* 結果の表示

- * メソッド「showMessage()」
- * 引き分けか○or×の勝利と表示する
- * どれを表示するかはターン数から考える
- * 表示の方法はダイアログでもキャンバスに直接でも好きな方で

Canvas.drawString(400,100,メッセージ)

または

JOptionPane.showMessageDialog(null, メッセージ)

*startメソッドを仕上げる

- *最後に今まで作ったメソッドをstartメソッド内に組み込めば完成！
- *一定条件を満たすまではclickなどは繰り返さなくてはいけない
- *もちろん勝敗が付いたらそこで終了できるように