

モバイルシステム演習

Processing

- 簡単な図形描画 -

株式会社GOCCO.

watanabe@gocco.co.jp

サイズを指定する

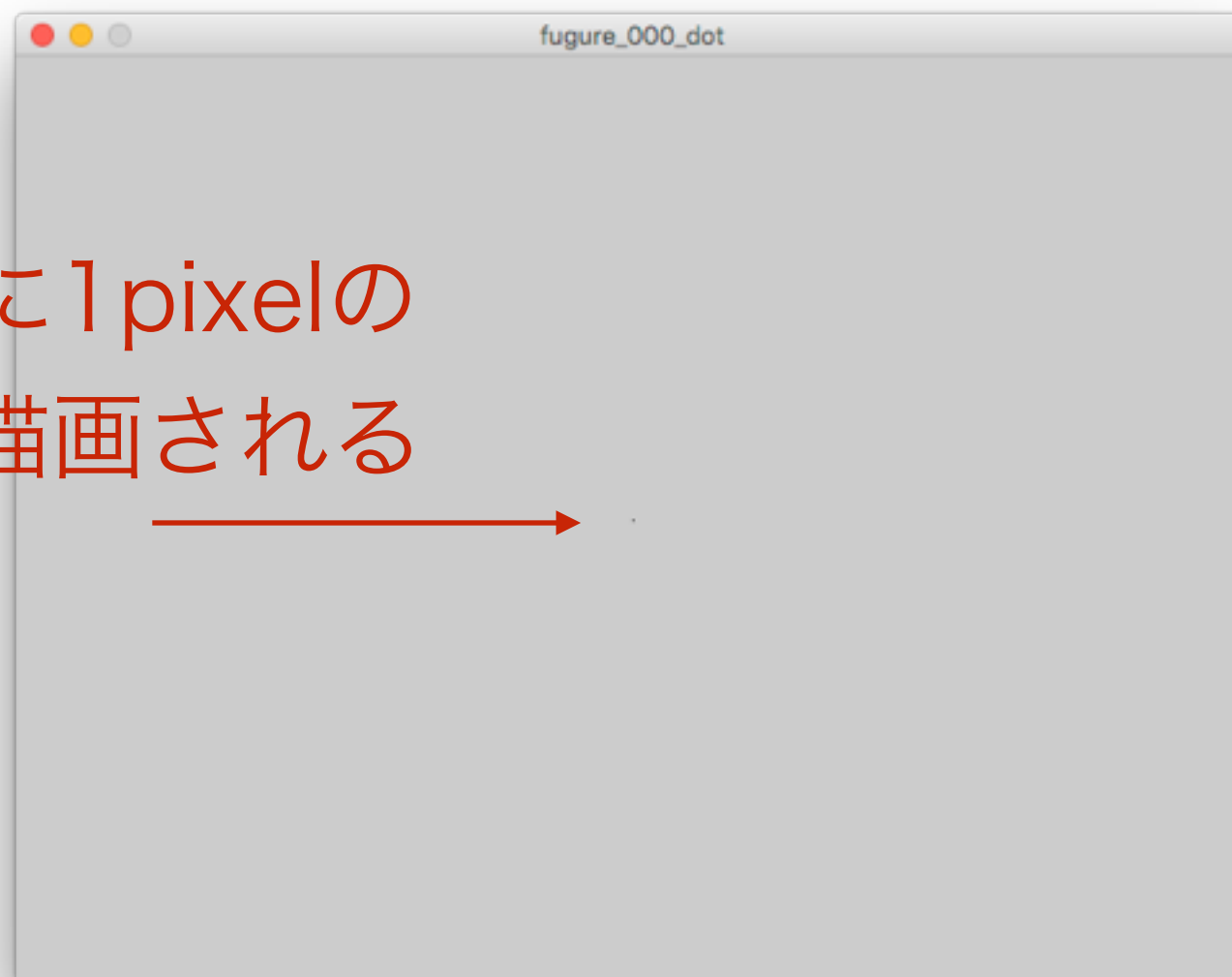
```
size( 横幅, 縦幅 );
```



点を描画する

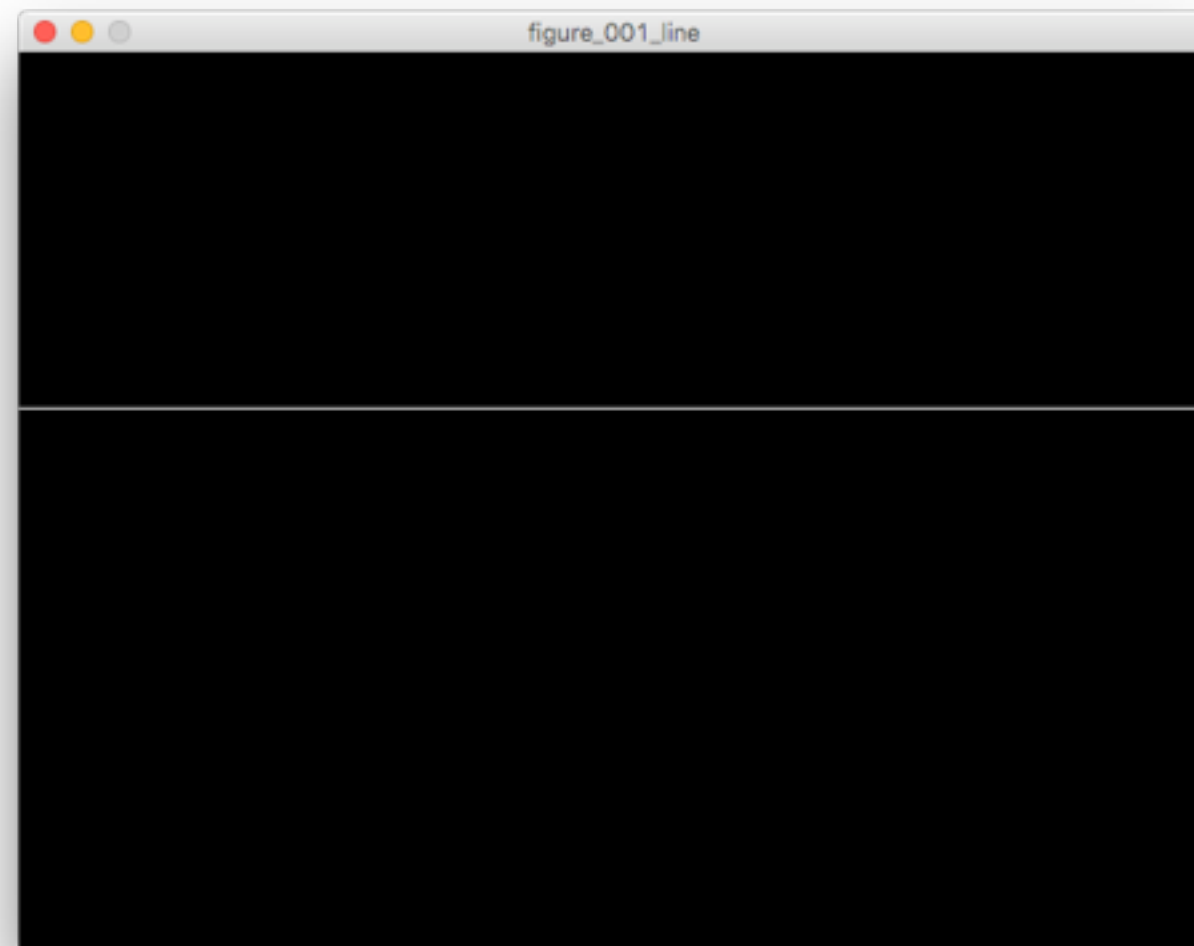
```
point(x座標, y座標);
```

画面中央に1pixelの
ドットが描画される



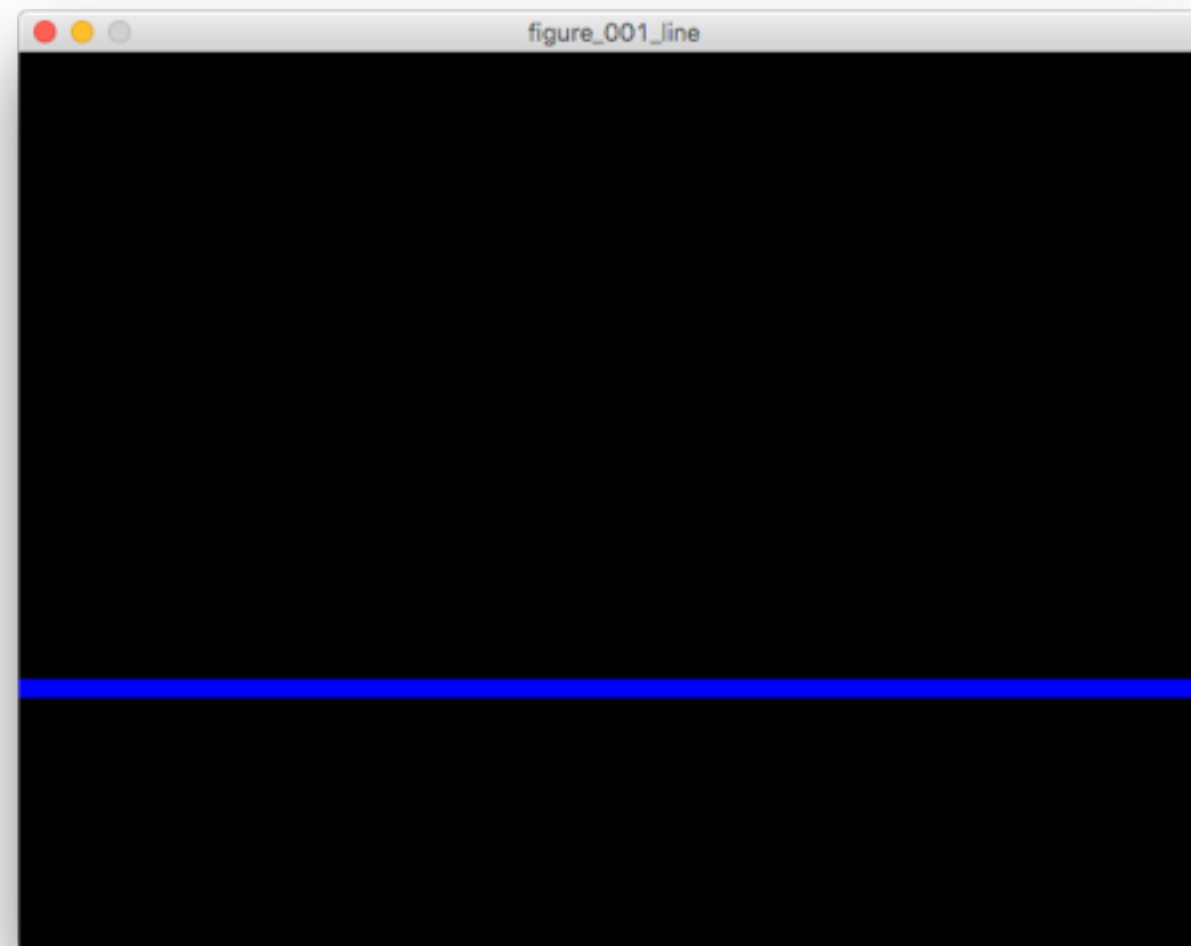
線を描画する

`line(x座標始点, y座標始点, x座標終点, y座標終点);`



線を描画する

```
strokeWeight(10); // 線の太さ  
stroke(0, 0, 255); // 線の色  
line(0, height / 2 + 100, width, height / 2 + 100);
```



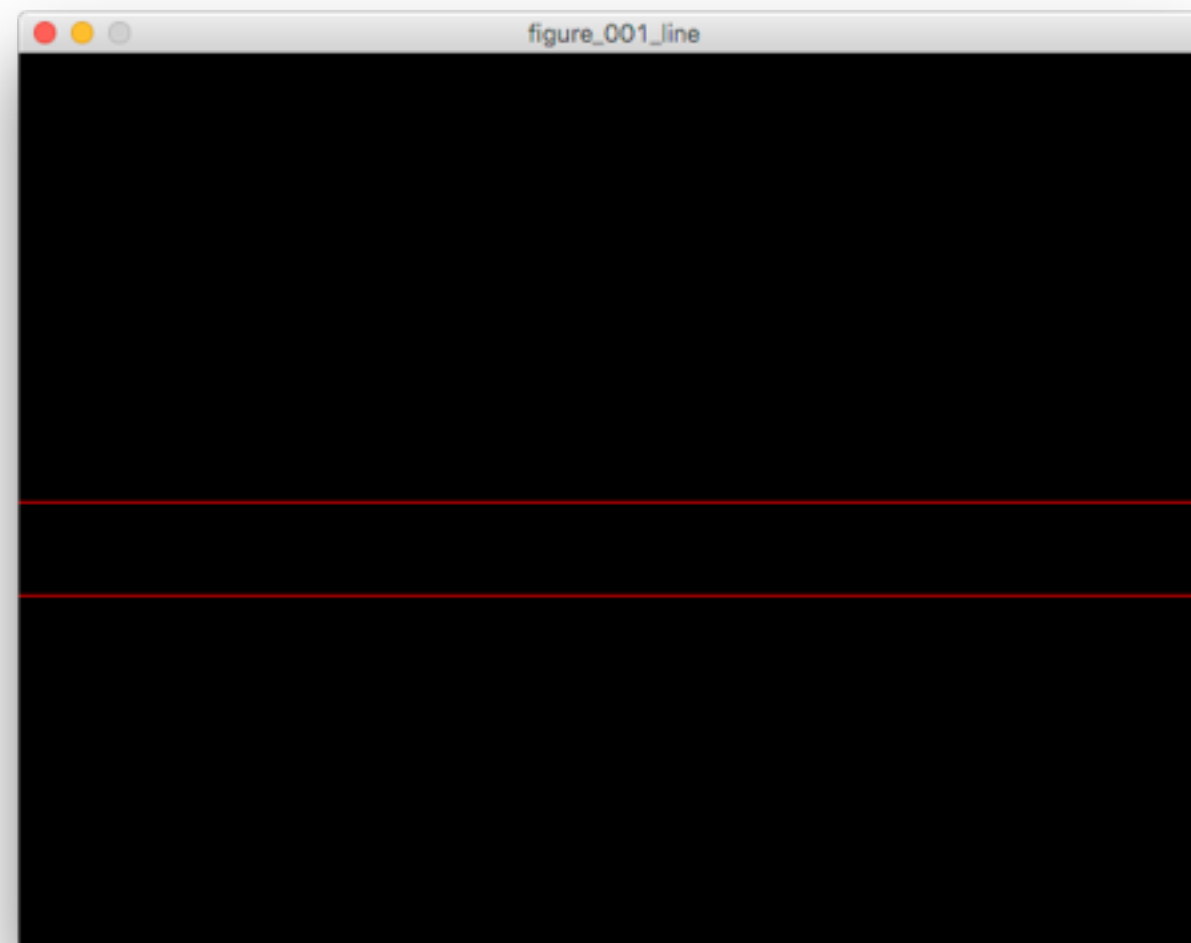
線を描画する

// 色や太さの設定はその後描画するすべての図形に影響する

```
stroke(255, 0, 0);
```

```
line(0, height / 2, width, height / 2);
```

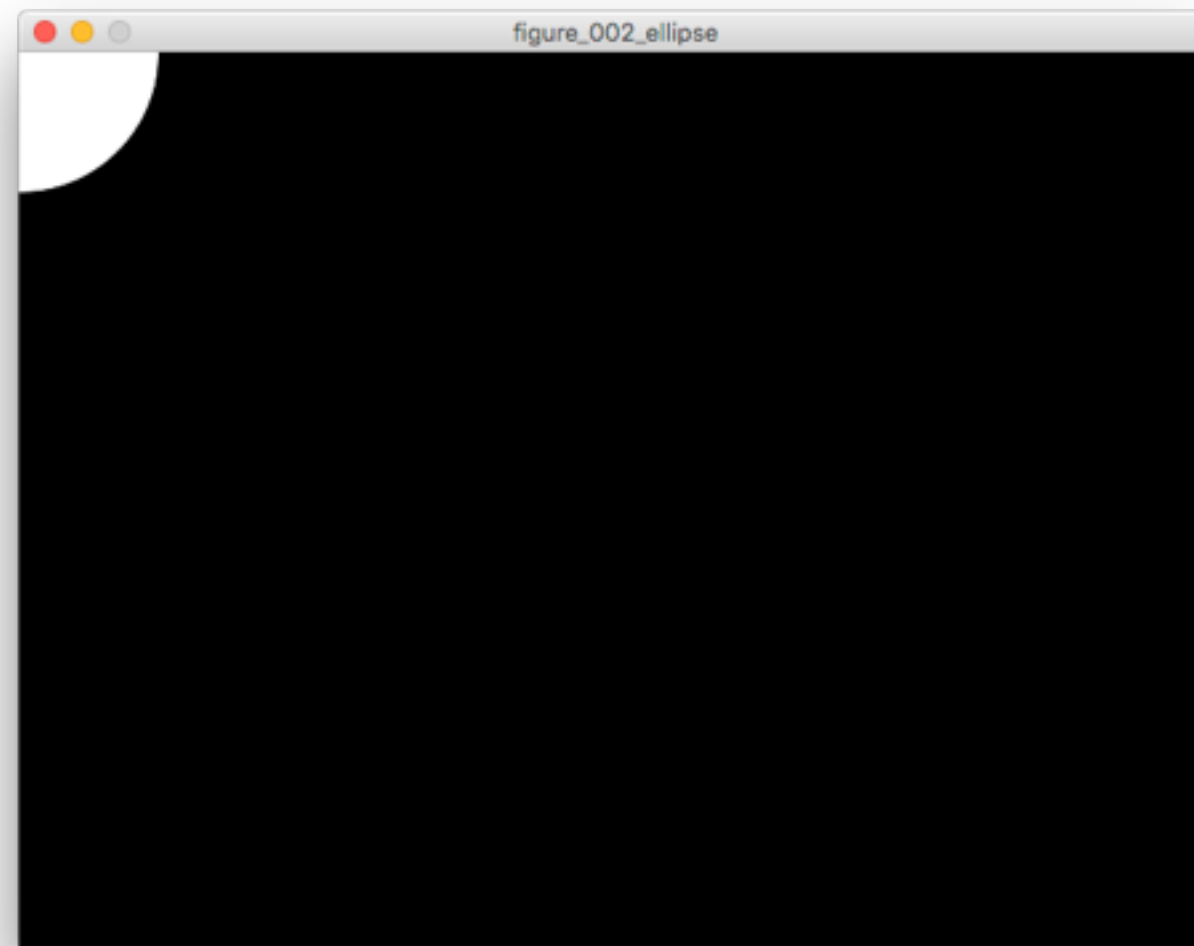
```
line(0, height / 2 + 50, width, height / 2 + 50);
```



円を描画する

```
ellipse(x座標、y座標, 横幅, 縦幅);
```

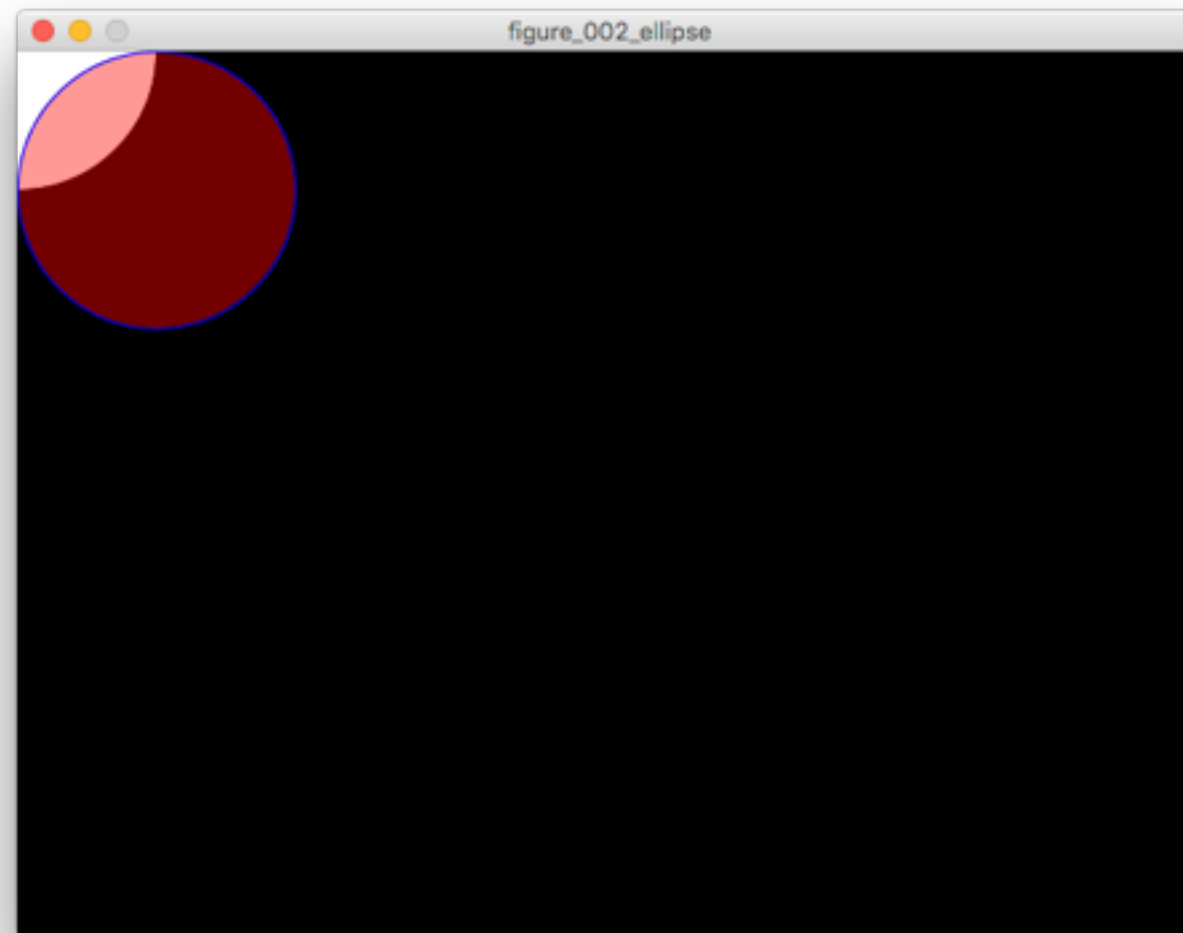
→ デフォルトは円の中心座標を設定



円を描画する

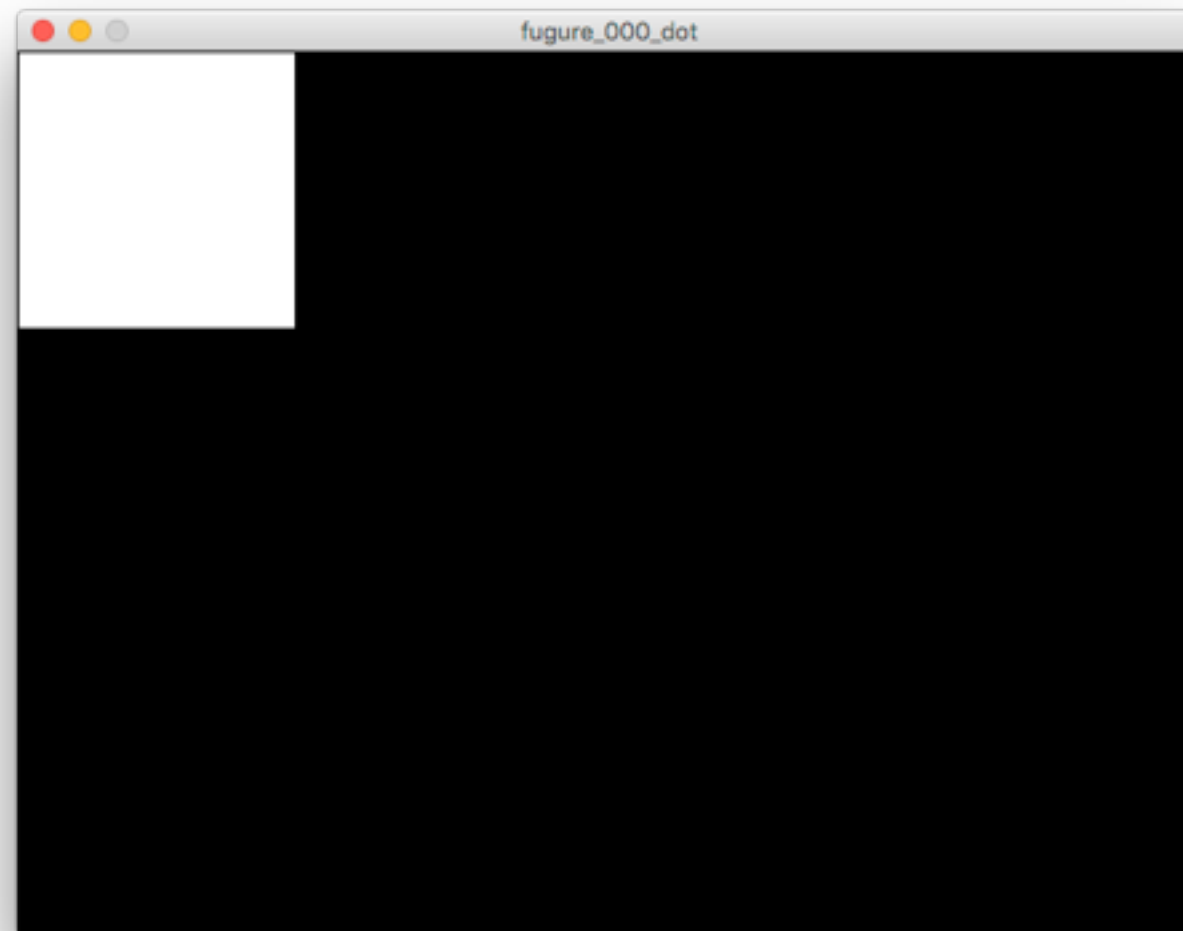
円の左上がアンカーポイントになる

```
ellipseMode(CORNER); // アンカーポイントの設定  
fill(Rの値, Gの値, Bの値); // 0~255の256段階
```



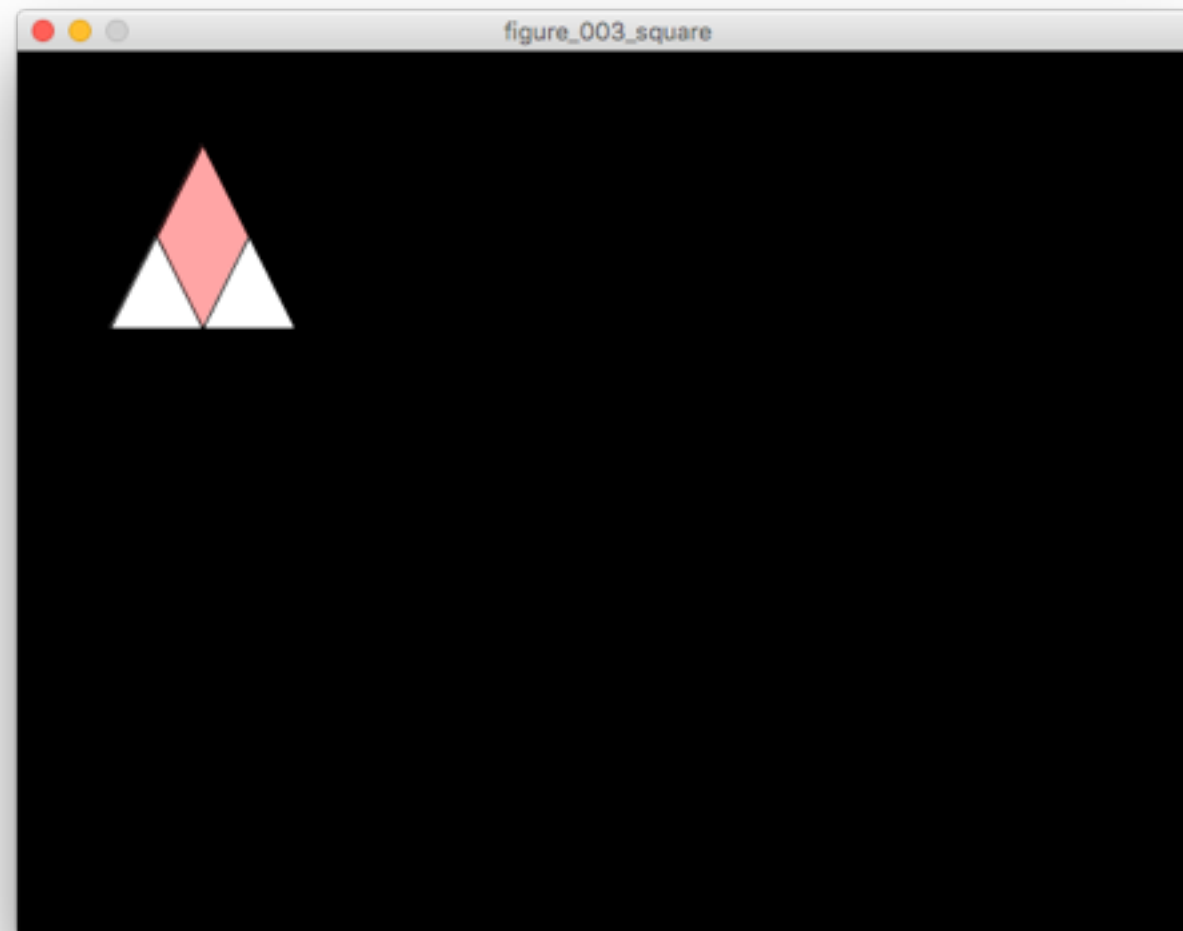
四角形を描画する

`rect`(左上角のx座標, 左上角のy座標, 矩形の幅, 矩形の高さ);



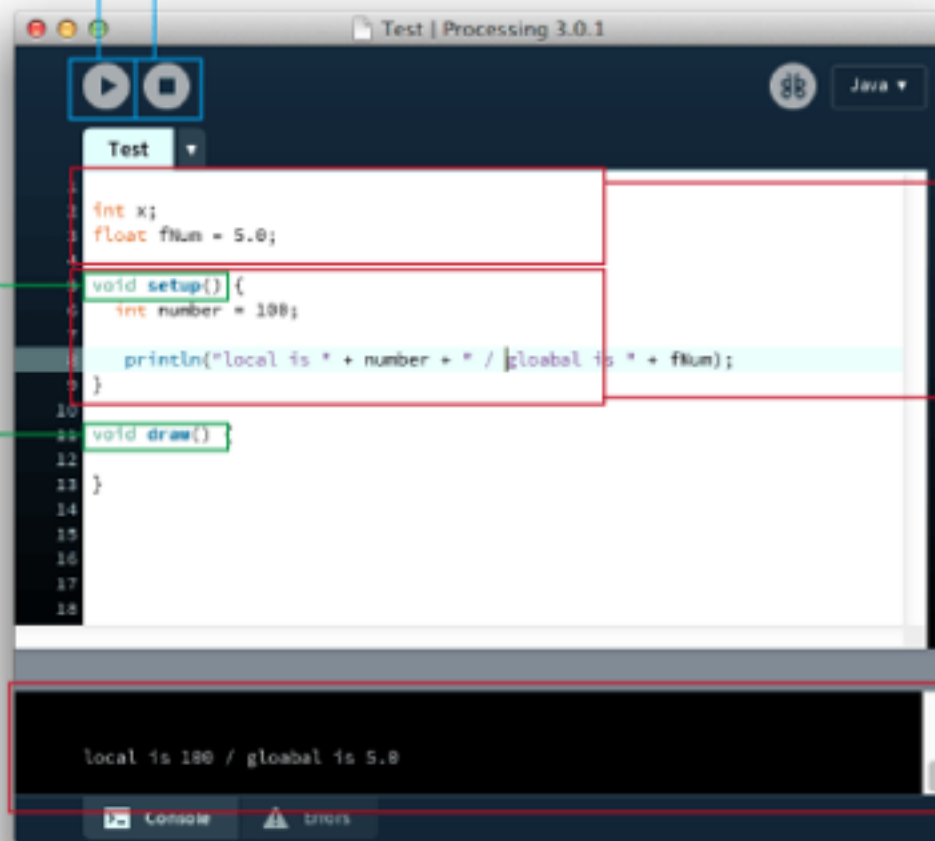
その他の図形を描画する

```
quad(角1のx座標, 角1のy座標, 角2のx座標, 角2のy座標, 角3のx座標, 角3のy座標, 角4のx座標, 角4のy座標);  
triangle(角1のx座標, 角1のy座標, 角2のx座標, 角2のy座標, 角3のx座標, 角3のy座標);
```



**点、線、円、四角形、その他の図形を使って、
デザインしてみよう**

プログラムの実行ボタン
プログラムの停止ボタン



グローバル変数領域

ここで宣言された変数は、他の関数の中でも使用できる。
この領域で宣言された変数をグローバル変数と呼ぶ。

関数領域

{ } で囲われた領域内で宣言された変数は、その領域内でのみしか
使用することが出来ない。この領域内で宣言された変数をローカル変数と呼ぶ。

コンソール

print 関数でログを出力した場合、ここに内容が表示される

setup 関数 (予約関数)

初期化を行う関数で、プログラム実行後に一番最初に 1 度だけ処理が行われる関数。

draw 関数 (予約関数)

プログラム実行後、毎フレーム処理が行われる関数。描画を更新したり、アニメーションさせたりするための関数。
プログラムの心臓部分