



#プログラミング #draw 関数 #random 関数 #変数  
#アニメーション

## Processing でプログラミングしよう(3)

図形が動くアニメーションをつくりましょう。

### 材料(必要なもの)

Processing

#### 1 関数

関数は Processing のプログラムの基本的な構成要素です。今まで使用していた `size()`、`line()`、`fill()` など実は関数なのです。これらの関数には役割があり、関数を組み合わせることで1つのプログラムをつくっていたのです。

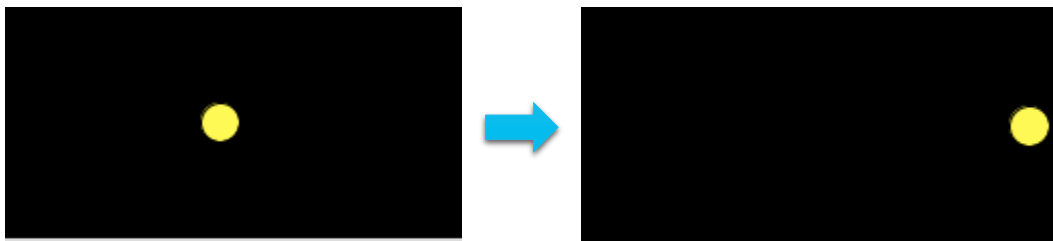
#### 2 draw 関数

`draw` 関数は、描画を毎秒約 60 回実行する役割を持っています。つまり、アニメーションを扱う関数です。`draw()` の後の`{}`内に記述したコードが、アニメーションとして描画されます。

なお、アニメーションのフレームレートを調整する命令は「`frameRate(数値)`」です。

#### 3 2点間を移動するアニメーションを描きましょう

次のように左から右へと円が移動するアニメーションをつくりましょう。円を描いて、`x`座標の位置を変数とし、変数の値を例えば1ずつ増加するように、どんどんと値を大きくするだけです。この計算を `draw` 関数の中で行えば、どんどんと増やすことが可能になります。



```
1 void setup(){
2   size(240,120);
3   frameRate(24);
4 }
5
6 int x = -20;
7
8 void draw(){
9   background(0);
10  x += 1 ;
11  fill(255,255,0);
12  ellipse(x,60,20,20);
13 }
```

## (1) setup 関数

setup 関数は、draw 関数を補完する役割を持っています。draw 関数は、2 で述べたように何回も繰り返して実行されます。一方で、キャンパスのサイズやフレームレートの指定は 1 回だけ実行するコードです。そこで、これらのコードを setup 関数の中に記述します。

## (2) 変数の有効範囲と draw 関数内の変数

変数は、原則として宣言している関数の中だけで有効となります。しかし、変数によっては特定の関数の中だけでなく、プログラム全体の中で用いたい場合もあるでしょう。その場合には、関数の外で変数を宣言するとその有効範囲が限定されなくなります。そうすることで、プログラムの中のどこでも使うことができます。

今回使用する変数は、円の x 座標の位置として変数 x を用いることとしました。それをどんどんと変化させるため、draw 関数の中で計算しています。draw 関数は毎秒約 60 回実行するので、何回も計算を繰り返してくれます。しかし、draw 関数の中で変数 x を宣言して初期値を設定してしまうと、初期化も繰り返してしまいます。

そこで今回は、変数 x の宣言と初期値の設定を関数の外で行い、1 回だけ初期化するようにしています。

## 4 問題演習

- (1) 2 点間を移動するスピードを上げてみましょう。
- (2) 垂直方向に移動するように変えてみましょう。
- (3) 左下から右上へ斜めに移動するように変えてみましょう。

## 5 random 関数

先に取り上げた 2 点間の移動は、x 座標の値がどんどんと大きく（または小さく）なる規則的な動きです。規則的な動きは美しいものですが、不規則な動きも意外性があり面白いものです。

そこで、乱数を利用して不規則な動きをつくりましょう。乱数を発生させるには random 関数を用います。命令は「random(下限値, 上限値)」です。ここで設定した上限値未満と下限値以上の数値をランダムに返します。なお、下限値を省略すると「0」として扱います。数値は整数型だけでなく浮動小数点数型も扱うことができます。

## 6 問題演習

2 点間を移動するプログラムを改造し、次のようなプログラムをつくりましょう。

- (1) x 座標の位置を乱数に置き換えて、円が横軸方向にランダムに点滅するようにしましょう。
- (2) x 座標だけでなく、y 座標も乱数に置き換えて、円があちこちにランダムに点滅するようにしましょう。
- (3) 円の色が黄色だけでなく、いろいろな色で点滅するようにしましょう。
- (4) 円が点滅するのではなく、次々と表示されるようにしましょう。

### コツ(留意点)

図形等を動かすときは、キャンパスの大きさを setup 関数、動かす図形等を draw 関数の中に記述しましょう。また、座標には変数や乱数を用いましょう。

### 作成者

北海道帯広緑陽高等学校 前田健太郎  
k\_maeda@hokkaido-c.ed.jp