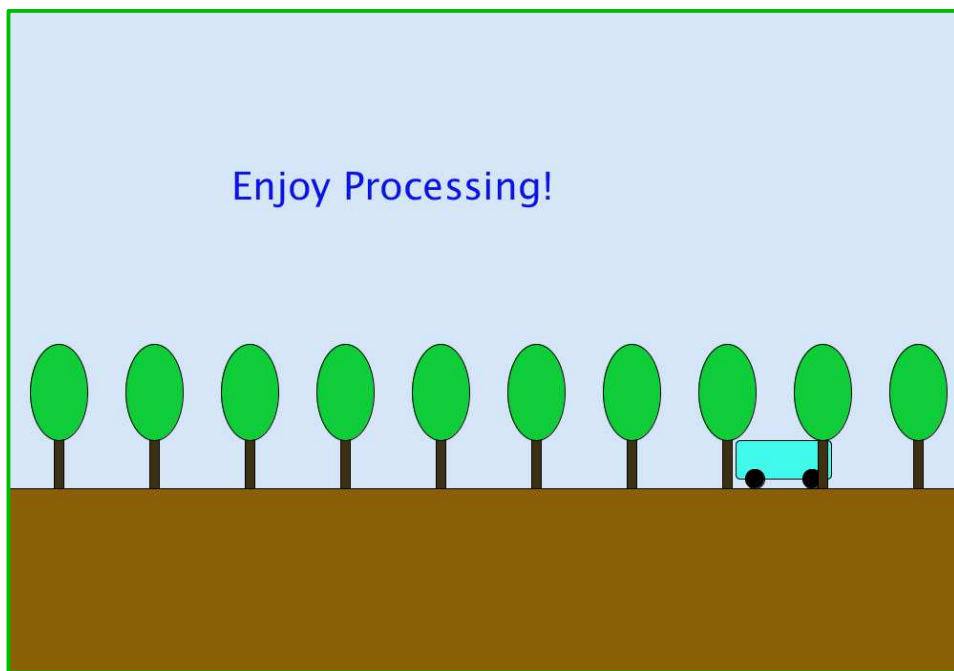


Processing

第7回



松田小学校 / 寄小学校

7 - ステップ 0 : 前回の復習だよ

新規のファイルを開いて、下のコードを打って実行してみよう。

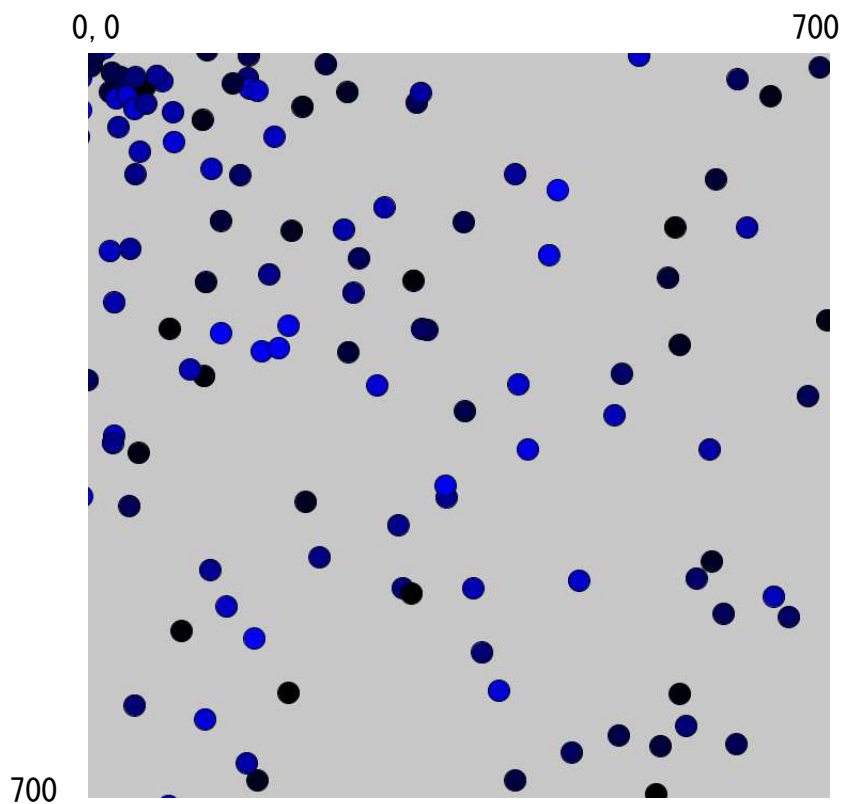
```
1 void setup() {  
2   size(700, 700);  
3   frameRate(10);  
4 }  
5  
6 void draw() {  
7   background(200);  
8   for(float a=0; a<700; a=a+0.5) {  
9     fill(0, 0, random(256));  
10    ellipse(random(-5, 5)*a, random(-5, 5)*a, 20, 20);  
11  }  
}}
```

実行回数のコントロールで、1秒間に10回を表す

random(-5, 5) は、-5 から 5 未満の数字を不規則に発生させる。

X座標とY座標は、randomをつかって -5 から 5 未満の数字を発生させて、それに 0 から700 までの数字 a をかけて決めている。

for() のなかに、a=a+0.5 という小数があるから、
int a=0; ではなく、float a=0; なんだ。



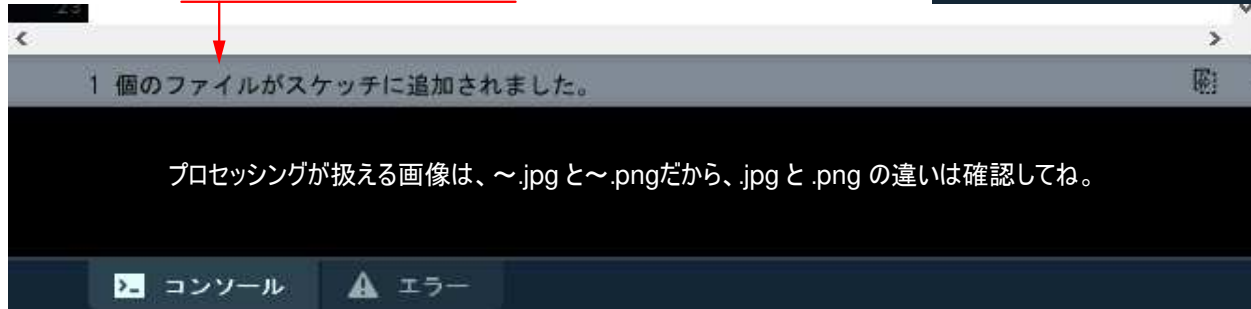
保存は不要だよ。できたらステップ1に進もう。

7ステップ 1 : 画像を取り入れよう

ファイルから新規を開いてみよう。

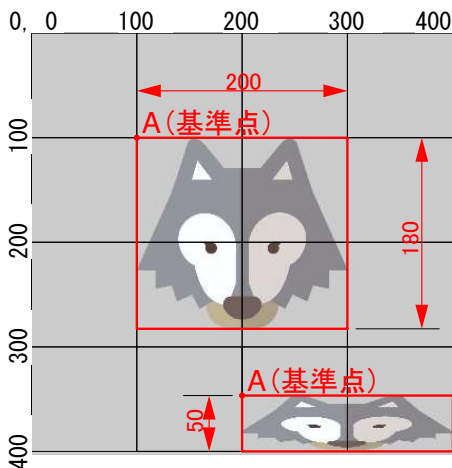
デスクトップに DATA というフォルダーがあるから、それをダブルクリックして開いてみよう。フォルダーの中から、wolf.png を選んで画像の上で左クリックしたまま、白いテキスト・エディターへと引きずってみよう。

ここに下の文ができれば、成功だ！



下のコードを打ってみよう。

```
1 PImage wolf; // PImage wolf; と画像のための変数を宣言する
2
3 size(400, 400); // 変数に画像名 wolf を代入 .png を確認
4 wolf = loadImage("wolf.png"); // 画像のための変数宣言は PImage だよ。
5 image(wolf, 100, 100, 200, 180); // wolf はオオカミという意味で、
// 画像の名前として使っている。
// 画像の名前 画像のA点の位置 画像の大きさ
```



横幅と縦の長さの比率を無視すると、下の図のようになってしまうよ。縦横比は次のステップで詳しく説明するね。

では、他の画像を取り込んだりして、数字を変化させて大きさや位置を変えて楽しんでみよう。

画像の縦横比を無視するとこうなる！

上のコードとどこを変えたかな？
実行してみよう。

```
1 PImage wolf; // 〇は大文字だよ
2
3 size(400, 400); // .png を確認
4 wolf = loadImage("wolf.png");
5 image(wolf, 50, 50, 300, 270);
```

300:270=200:180=10:9 で同じ比率にしてある。

for 文を組み込んでみよう。

```
1 PImage wolf; // 〇は付けてないけど、
2 // どれが大文字か確認してね。
3 size(400, 400);
4 wolf = loadImage("wolf.png");
5 for(int a=1; a<400; a=a+100){
6 image(wolf, a, 75, 100, 90);}
```

wolf で保存しよう。

7ステップ 2 : 写真を取り込んでみよう

ファイルから新規を開いてみよう。画像と同じように写真も取り込むことができる。ふつう写真は背景に取り込むよね。そこで大きさや縦横比が大きな影響を与えるんだ。



① DATA のホルダーをダブル・クリックして開いて、使いたい画像や写真の上にマウスを置くと、ファイルの種類や大きさが表示されるんだ。p700.jpg は800×600 だね。これを書き留めて、コードの中に入れておくんだ。
写真を左クリックして引きずってくるのは、画像と同じだよ。

② 写真を背景に使う場合は、背景はウィンドウ全体に広がっていないとおかしいよね。
背景のサイズと、size(,); の数字をそろえないと、左のような状態になってしまうんだ。p700.jpg の写真を背景に使うなら、size(800,600); にするとOKだ。写真の基準点は、左上だから写真の位置は 0, 0, だ。



```

1 PImage p700;
2
3 size(800, 600);
4 p700=loadImage("p700.jpg");
5 image(p700, 0, 0, 800, 600);
    
```

写真の大きさを消して実行してみよう。

画像の縦横の比率の話し=相似(そうじ)ということ

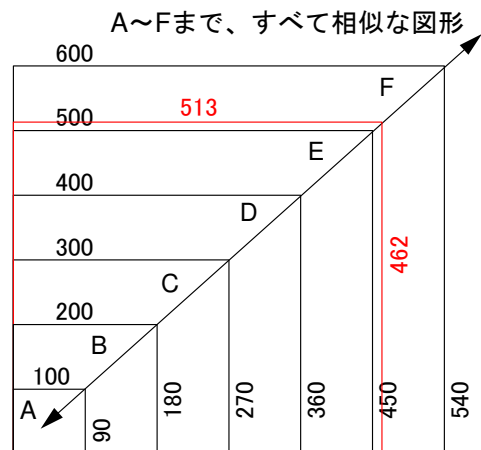
縦横の比率が等しい図形を<相似>とって、下の図のような関係があるんだ。相似の図形は、大きくなっても小さくなくても、横：縦または縦：横の比率が一定なのだ。
この図だと、600：540 も 100：90 も比率は 1.11 だし、513：462 も 1.11 だね。540：600 も 360：400 も比率は 0.9 だし、462：513 も 0.9 だね。

```

1 PImage p700;
2
3 size(800, 600);
4 p700=loadImage("p700.jpg");
5 image(p700, 0, 0);
    
```

写真の大きさを書かないと、プロセッシングが自動的に判断して、ウィンドウ一杯に表してくれるんだ。だから写真の場合は、大きさを書かない方が良くもね。

できたら foto で保存しよう。



7 ステップ 3 : 写真の背景に画像をいれてみよう

背景にキレイな写真が入ったので、次にはこの背景に画像を加えてみよう。
whale(くじら)を、テキストエリアに引きずって(ドラッグ&ドロップという)こよう。

```
1 loadImage p700, whale;
2 int a=1;
3 int b=1;
4
5 void setup() {
6   size(800, 600);
7 }
8
9 void draw() {
10  a=a+b;
11  if( a<=0 ) { b=1; }
12  if(a>width-180) { }
13
14  p700 = loadImage("p700. jpg");
15  image(p700, 0, 0);
16  whale = loadImage("whale. png");
17  image(whale, a, 250, 180, 180);
18 }
```

背景と鯨をつかう宣言

○ は大文字だよ

右に行く

左に行く

鯨が左右に動く if 条件文

鯨の X 座標に変数 a をつかう



できたら whale で保存しよう。

7-ステップ 4 : 画像(ufo1)を動かそう

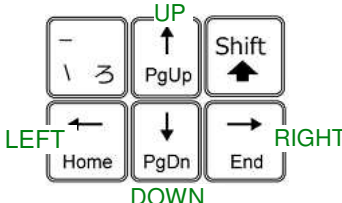
ファイルから新規を開いてみよう。画像を自分で動かすことは出来ないだろうか？
ufo1 をテキストエリアに引きずってこよう。そして if 文と組み合わせてみるよ。

```
1 Image ufo1; // ← ufo1 という画像を使う宣言。
2 int x=0; // X方向に動かすための変数
3 int y=0; // Y方向に動かすための変数
4
5 void setup() {
6   size(800, 800);
7   imageMode(CENTER); // ← 基準点は画像の中心だね。
8 }
9
10 void draw() {
11   background(255); // .png を確認
12   ufo1 = loadImage("ufo1.png");
13   image(ufo1, x, y, 150, 100);
14 }
15
16 void keyPressed() {
17   if(keyCode==LEFT) {x=x-1;}
18   if(keyCode==UP) {y=y-1;}
19   if(keyCode==RIGHT) {x=x+1;}
20   if(keyCode==DOWN) {y=y+1;}
21 }
```

○は大文字だよ

何も指定しないと、画像の基準点は左上だけど、
imageMode(CENTER); を加えると、中心になるんだ。

コピー&ペーストだよ、
手直しはその後でね。



keyPressed は mousePressed と同じで、キーが押されたという意味だけれど、キーはたくさんあるから if 文を使って命令を細かく指定してやるんだね。
17行目は、もし、keyCode が LEFT なら x に x-1 を代入しなさい、という命令だから、画像は左へと動いていくんだ。上の図はキーボード右下を切り取ったんだけど、上のコードを実行したらキーを押してみよう。

ミッション

矢印に従って動くけど、メチャクチャ遅いね。
下のヒントを使って、早く動くようになおしてみよう。

ヒント

速さを制御しているのは、 $x=x-1$ の -1 だった。この数字を大きくすれば早くなる。でも4つもあるから、1つ1つ変更するのは大変だ。速さの変数 `int go;` を作ろう。`int go=10;` で試してみよう。上のコードの17~20行目までの `1` は `go` に変えるんだ。

ufo1 で保存しよう。

7ステップ 5 : チャレンジだ、写真と画像の続き

新規のファイルを開いて、右の図になるコードを書こう。

画像や写真はDATAファイルから引きずってくるんだ。

```
1 PImage p102, rocket;  
2  
3 size(800, 600);  
4  
5 p102 = loadImage("p102.png");  
6 image(p102, 0, 0);  
7 rocket = loadImage("rocket.png");  
8 image( , 400, 400, 100, 200);
```



ミッション：難問だよ。

②は新規のファイルを開いてね

- ① ロケットを上昇させてみよう。できたら **rocket-2** で保存しよう。
- ② 同じ星空を背景にして、月 (moon.png) を横切らよう。 **moon-1** で保存しよう。
- ③ 同じ星空を背景にして、月を左下から登らせて、真ん中まで来たら、右下へと沈んで行かせよう。できたら **moon-2** で保存しよう。if 文を使うのがヒントだ。

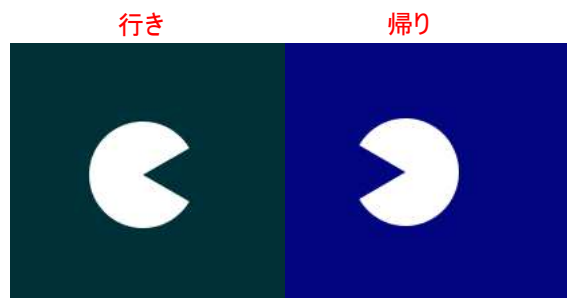
画像や写真は下記のサイトから借用しました。

FLAT ICON DESIGN <http://flat-icon-design.com/>
photoAC <https://www.photo-ac.com/>



次回の予告

円弧を知っているか？ 円の一部を円弧と言うんだね。半円は中心角が 180° の円弧だし、上弦の月も下弦の月も円弧だよ。次回は、円弧の描き方から、方向変換などをやろう。



画像や図形の往復はできるようになったね。今までは行ったままの姿勢で戻ってきていた。今度はキチンと進行方向を向いて往復させたいね。

Bye-bye!