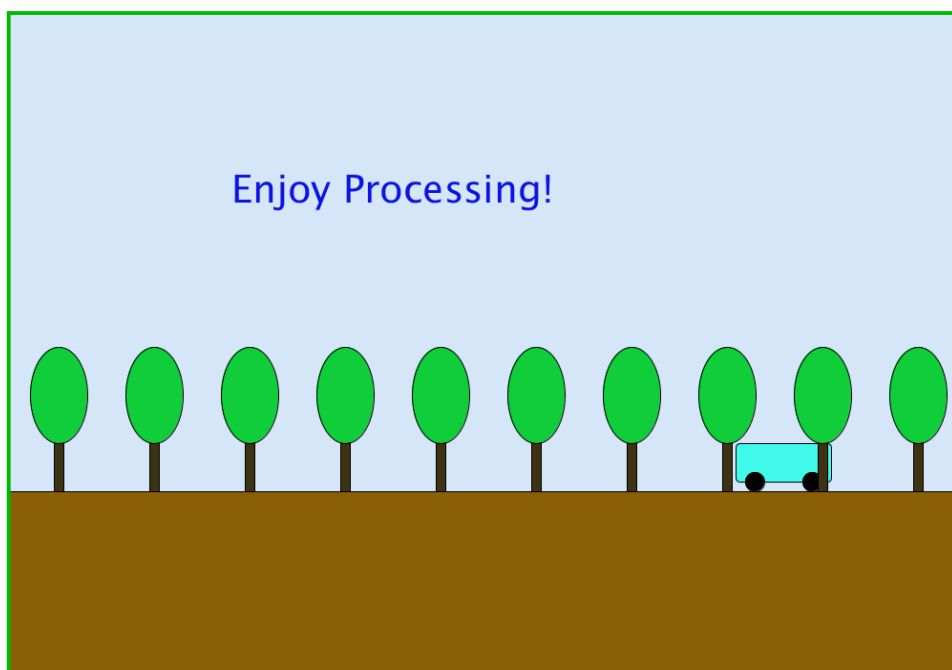


Processing

第8回



松田小学校 / 寄小学校

8 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 float a;
2 void setup() {
3   size(700, 700);
4 }
5 void draw() {
6   background(200);
7   translate(350, 350);
8   rotate(a);
9   fill(255, 0, 0);
10  circle(300, 0, 30);
11  fill(0, 0, 255);
12  circle(200, 0, 20);
13  fill(255, 255, 0);
14  rect(0, 0, 50, 50);
15  a=a+0.01;
16 }
```

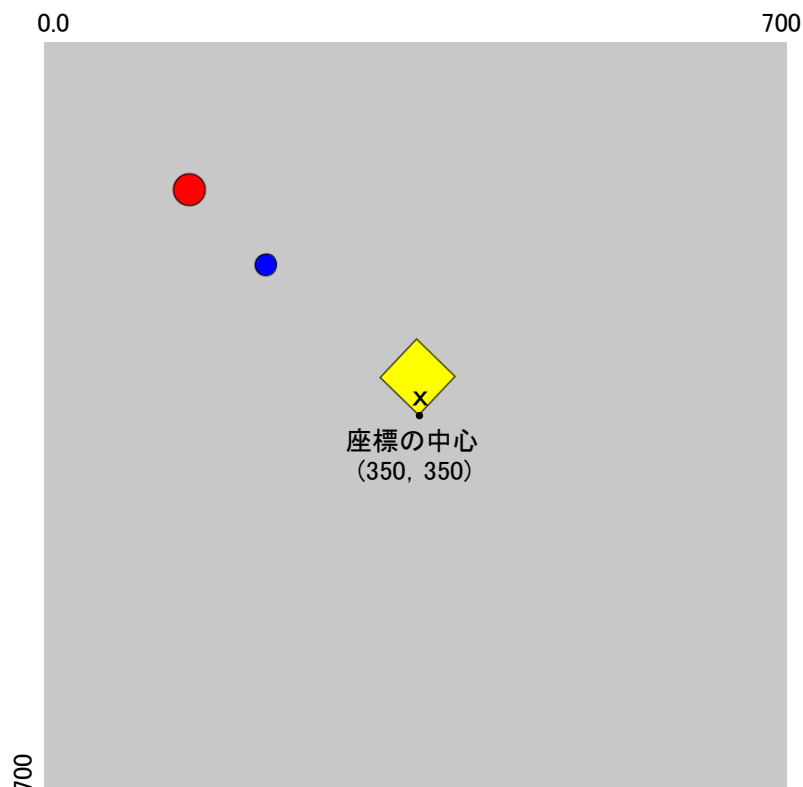
変数に小数を使いたいから、float だ。

コピペが使えるよ。

めちゃくちゃ重要

void セットアップ
void setup() { } は
{ }内を最初に一度だけ実行する。

void ドロー
void draw() { } は
{ }内を何度もくりかえす。



ミッション

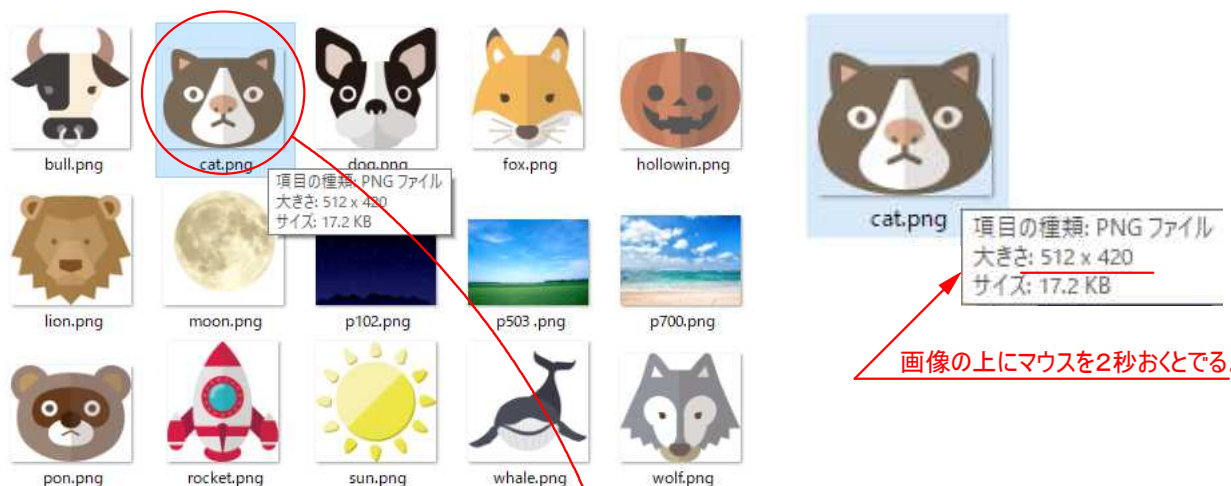
1. 丸の色や大きさ、回転半径、回転の速さなどを変えてみよう。
2. 四角形はなぜX点を中心に廻っているのだろう？

保存は不要だよ。できたらステップ1に進もう。

8 - ステップ 1 : 画像 (image=イメージ) を取り入れよう

画像の取り入れ方

- ① デスクトップにあるDATA(データ)ファイルを、ダブルクリックして開く。
- ② 取り入れる画像を選んで、マウスでプロセッシングに引きずってくる。
- ③ 「1個のファイルがスケッチに追加されました」と出たら、成功だよ。



著作権について

絵やカット、写真、音楽などの作品を創作した者がもつ権利を著作権といいます。お金をかせぐことを目的として、著作権者に無断で作品を使うことはできません。ただし、著作権者が権利を放棄している場合は無断で使ってもかまいません。

8 - ステップ 2 : 画像 (image=イメージ) を表そう

画像を表すための準備

画像を扱うには、PImage (ピーイメージ) という変数が必要だ。

Picture Image (ピクチャー イメージ)

画像を読み込むには、画像名=loadImage ("画像名.png");

load Image(ロードイメージ)

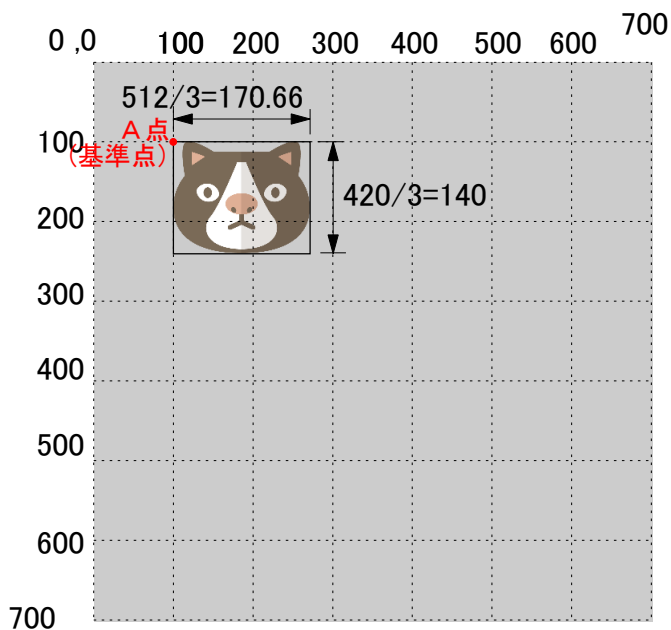
画像を表示するには、image (画像名、X座標、Y座標、横幅、縦の高さ)

```
1 PImage cat; ← cat を使うと変数を宣言する
2
3 size(700, 700); ○ は大文字だよ
4 cat = loadImage("cat.png");
5 image(cat, 100, 100, 512/3, 420/3);
```



項目の種類: PNG ファイル
大きさ: 512 x 420
サイズ: 17.2 KB

A 点の X 座標 ↑
A 点の Y 座標 ↑
大きすぎるので 1/3 にした。

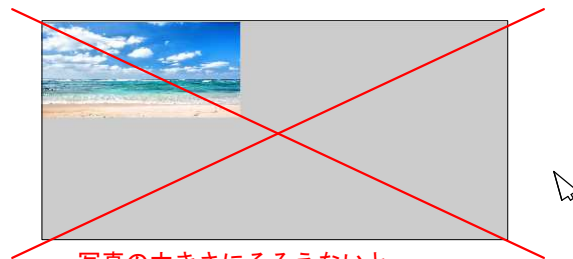
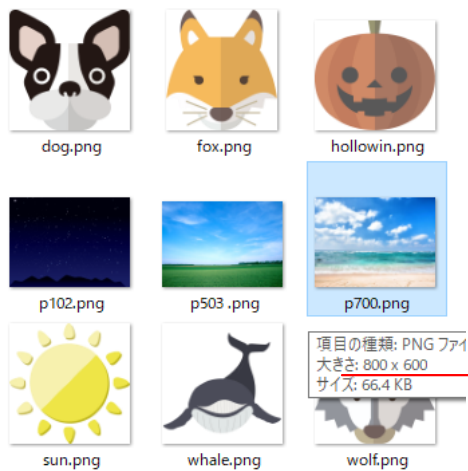


ミッション

- ① 5行目の /3 をとって、実行してみよう。
- ② 画像の大きさを変えてみよう。/3 を、/2 とか /5 にすると変わるよ。
- ③ /3 に戻して、画像の位置を変えてみよう。中央に位置できるかな？
- ④ 他の画像も取り入れてみよう。2行目に PImage なまえ の宣言を追加してね。次に 4、5行目を 6、7行目にコピーして、取り入れる名前に変えよう。

8 ステップ 3 : 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。



写真の大きさにそろえないと、
上のようになってしまうよ。

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

写真の大きさにそろえる。

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

size(); と写真の大きさが同じ時は書かなくてもOK。

size(); の数字と写真の大きさが同じ時は、大きさを書かなくても、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。

8 ステップ 4 : 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 Image p700, whale; ← 背景と鯨をつかう宣言
2 int a, b=1;
3
4 void setup() {
5   size(800, 600); ← 〇は大文字だよ
6   p700=loadImage("p700.png"); ← 背景を読み込む
7   whale=loadImage("whale.png"); ← 鯨を読み込む
8 }
9
10 void draw() {
11   image(p700, 0, 0);
12   image(whale, a, 250, 512/3, 512/3); ← 鯨のX座標に変数 a をつかう
13                                     ← 鯨が大きすぎるので、1/3にした。
14   a=a+b;
15   if( a<=0 ) { b=1; } ← 右に行く
16   if(a>width-512/3) { [ ]; } ← 左に行く
17 }
```



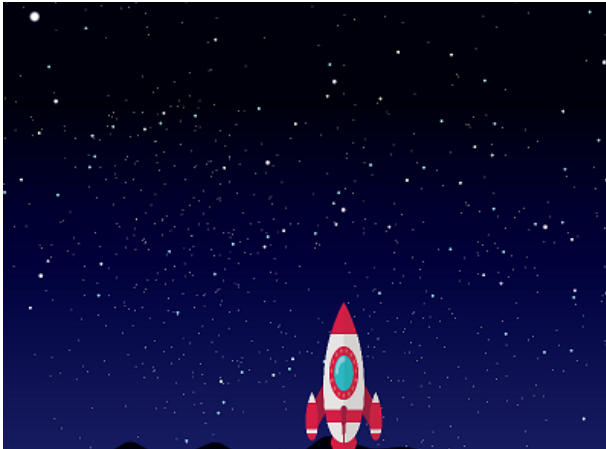
できたら whale で保存しよう。

8 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  ← int a = 最初のロケットの位置;
3
4 void setup() {
5   size(800, 600);
6   p102 = loadImage("p102.png");
7   rocket = ("rocket.png");
8 }
9
10 void draw() {
11   image(p102, 0, 0);
12   image(, 400, a, 100, 200);
13   a=a-1;
14 }
```



ミッション

- ① ロケットを上昇させてみよう。2行目に動かすための変数 `a` を宣言してから、`void setup()`; と `void draw()`; を使うんだよ。できたら `rocket` で保存しよう。
- ② 月を追加して、月 (`moon.png`) を横切らせよう。ロケットと月が交差するかな？
`moon` で保存しよう。

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

信号機を描いてみよう。
緑色→黄色→赤色と点灯するように、プログラミングしてみよう。そして、自分だけのオリジナル信号機にしてみよう。



Bye-bye!