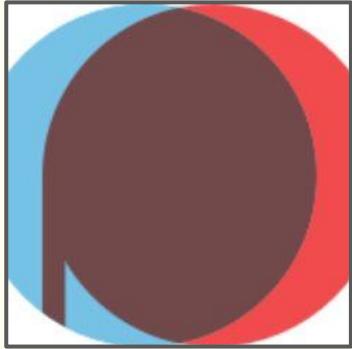


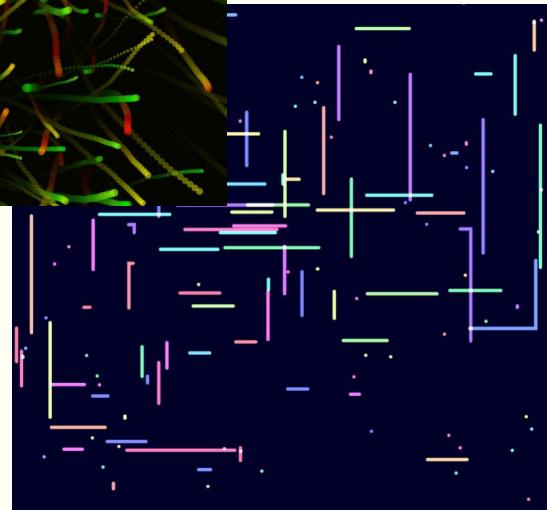
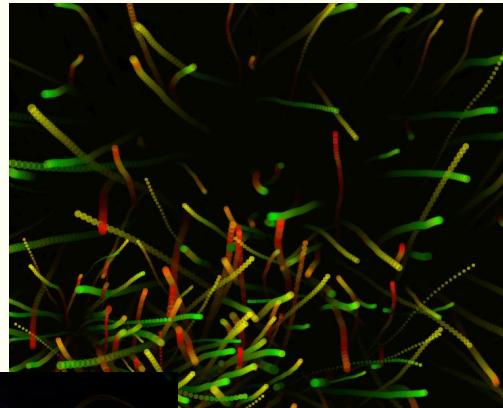
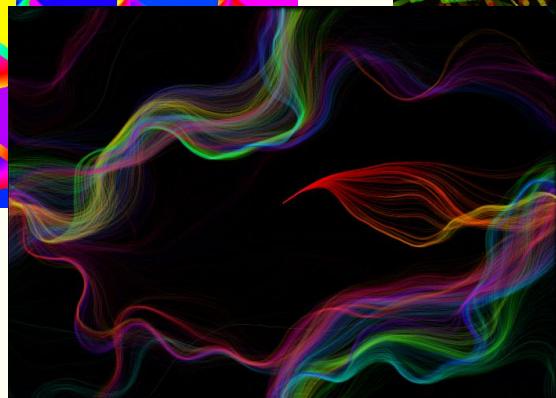
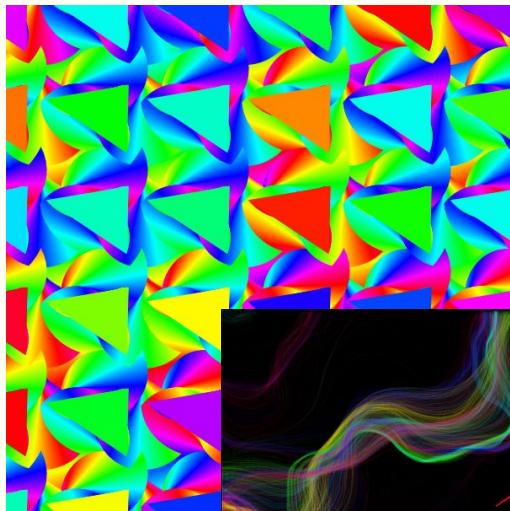
# OpenProcessing体験

# OpenProcessingって何？

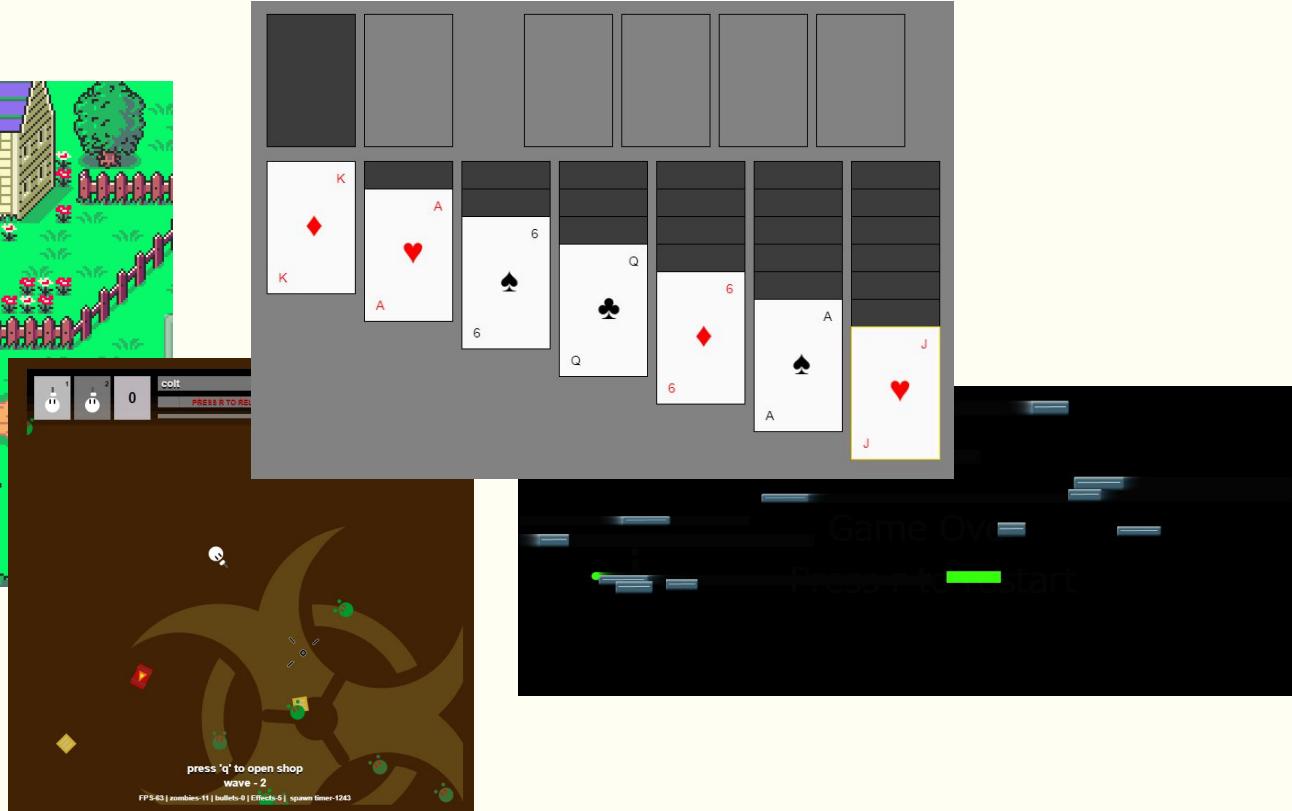


OpenProcessingは、  
ブラウザ上で簡単に  
作る  
遊ぶ  
見せる  
ことができるプログラミング環境です

こんな感じのデジタルアートや



# こんな感じのゲームが作れちゃいます



なんと環境構築

必要なし!!

※メアドでアカウントを作る手間はあります

携帯でも、

作品を遊ぶ

作品の共有

コーディング ※

ができちゃいます！

※携帯のキーボードはコーディングに向いてない

# OpenProcessingの始め方

まずは以下のURLにアクセス

<https://www.openprocessing.org>



# Joinを押して登録開始

16:03 19%

openprocessing.org

OpenProcessing

## Creative Coding

for the curious mind

The community of designers, artists, educators and everyday coders, experimenting on algorithmic design.

[Join](#)

[Sign in](#)

Process4.reload() by Alessandro Valentino

Connect and Inspire

Join 100,000 creative coders and follow their work. Add your ideas with forks, and find your inspiration for your next coding challenge.

[Explore sketches](#)

< >

OpenProcessing - Creative Code

https://www.openprocessing.org

OpenProcessing

## Creative Coding

for the curious mind

The community of designers, artists, educators and everyday coders, experimenting on algorithmic design.

[Join](#)

[Sign in](#)

Frozen brush by Jason Labbe

Connect and Inspire

Join 100,000 creative coders and follow their work. Add your ideas with forks, and find your inspiration for your next coding challenge.

[Explore sketches](#)

No Hassle Coding

Code away your ideas in seconds with intuitive editor and **psjs**, enable common libraries with ease, and share your work with the community.

[Create a sketch](#)

Teach and Learn

Teachers exclusive: Invite your students to study coding, give assignments and collect student work on your class page.

[Learn more](#)

# 必要事項を入力して登録

docomo 16:04 19% 

openprocessing.org

○Join the Creative Coders on OpenProcessing

NAME \_\_\_\_\_

EMAIL \_\_\_\_\_

PASSWORD \_\_\_\_\_

STAY SIGNED IN  X

私はロボットではありません   
reC  
プライバシー

By clicking Join, you agree to the [Terms of Service](#).

**Join**

Join - OpenProcessing

https://www.openprocessing.org/home/join

○

Join the Creative Coders on OpenProcessing

NAME \_\_\_\_\_

EMAIL \_\_\_\_\_

PASSWORD \_\_\_\_\_

STAY SIGNED IN X

私はロボットではありません   
reCAPTCHA  
プライバシー・利規約

By clicking Join, you agree to the [Terms of Service](#).

**Join**

Have an account already? [Sign in](#).

# 登録したらマイページに到着



## Things happened



Welcome to the world of creative coding!

Here are super awesome things you can do:

- Start coding by clicking Create a Sketch button above.

## Things happened



Welcome to the world of creative coding!

- Here are super awesome things you can do:
- Start coding by clicking Create a Sketch button above.
  - Browse around the site, follow other users and check out their sketches
  - Comment and ⌘ sketches to show your appreciation.
  - Make a change in the code on any sketch, and save your changes as a fork.
  - If you are teaching coding, [create a class](#) to collaborate with students.

You will see more tips and ideas here to get inspired!

## Events coming up



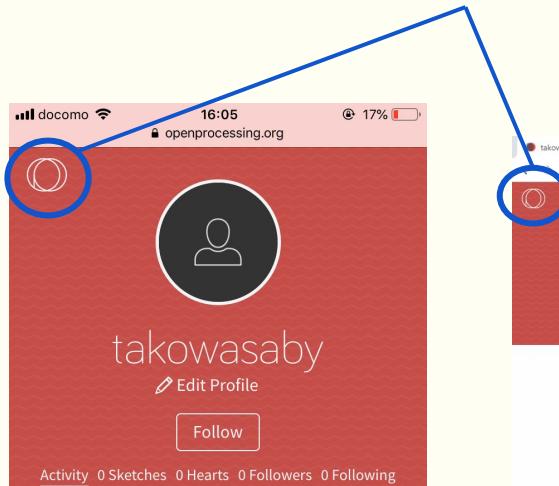
CODE • ART • DATA • DESIGN  
EYEO Festival 2019  
Join the conversation.  
Converge to Inspire.  
Tickets on sale now.

## Coding Challenge: Tiny Sketch

Execution memory {  
  // what can you create...  
}  
  
Execution draw() {  
  // ...in 200 characters!  
}  
  
Join the challenge to win a free ticket to EYEO Festival in June.

## Consideration by

# 左上アイコンでメニューへ



Things happened

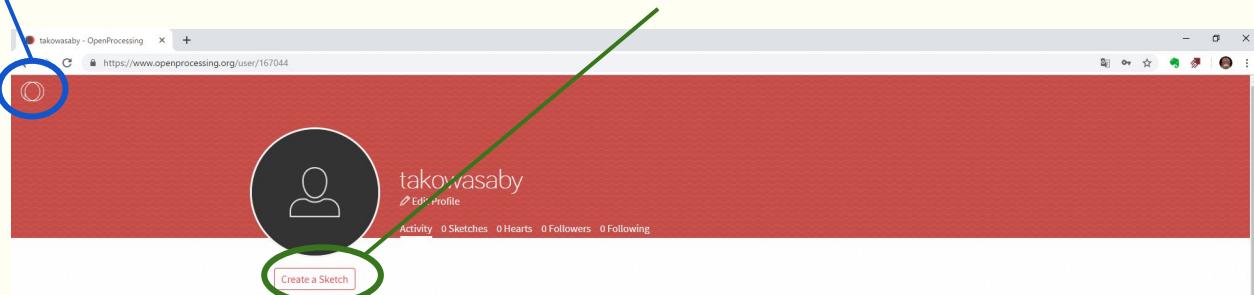


Welcome to the world of creative coding!

Here are super awesome things you can do:

- Start coding by clicking Create a Sketch button above.

# Create a Sketchで作品を作る



Things happened



Welcome to the world of creative coding!

Here are super awesome things you can do:

- Start coding by clicking Create a Sketch button above.
- Browse around the site, follow other users and check out their sketches
- Comment and ⌂ sketches to show your appreciation.
- Make a change in the code on any sketch, and save your changes as a fork.
- If you are teaching coding, [create a class](#) to collaborate with students.

You will see more tips and ideas here to get inspired!

Coding Challenge: Tiny Sketch

```
function setup() {  
  // What can you create...?  
}  
  
function draw() {  
  // ...in 200 characters!  
}
```

Can you push your limits to create a sketch under 200 characters with P5.js?  
Join the challenge to win a free ticket to EYEO Festival in June.

Consideration by

# メニューからは作品の検索ができます

This screenshot shows the mobile version of the OpenProcessing website. At the top, there's a header bar with the URL "openprocessing.org" and a battery icon showing 16% remaining. Below the header is the main navigation menu:

- OpenProcessing
- My Sketches
- Classes
- My Feed
- Edit Profile
- Get Plus+ Membership
- Sign out

Below the menu is a search bar with the placeholder "Search". Underneath the search bar are links for "physics", "game", and "/Recent".

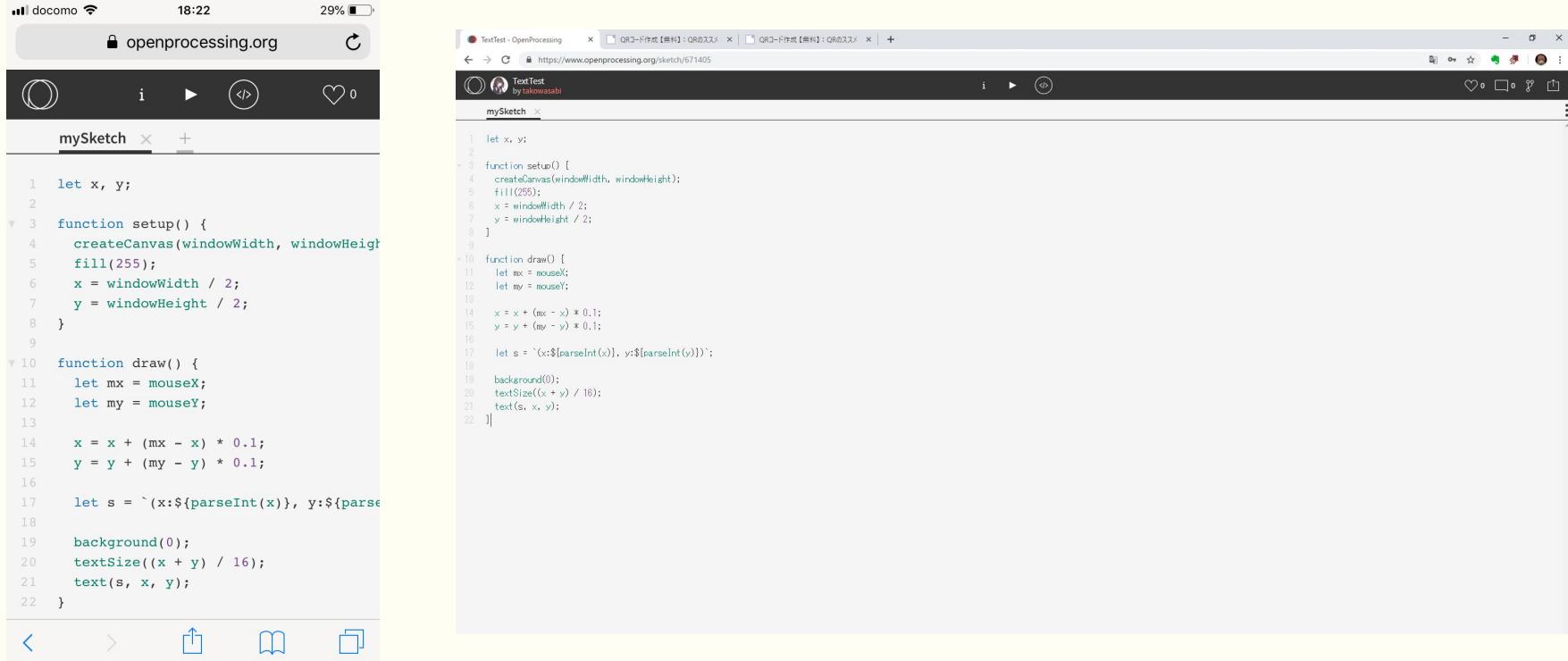
At the bottom of the page are footer links: "info@openprocessing.org", "Twitter", "Credits", "Community Guidelines", "Terms of Service", and "Privacy Policy". There are also standard mobile navigation icons at the very bottom.

This screenshot shows the desktop version of the OpenProcessing website. The URL in the address bar is "https://www.openprocessing.org/user/167044".

The page includes the following sections:

- My Sketches**: Links to "My Feed", "Edit Profile", "Get Plus+ Membership", and "Sign out".
- Classes**: Links to "Create a Class" and "View All Classes".
- Search**: A search bar with suggestions like "physics", "game", "visualization", "particles", "color", "evolution", "circle", and "lines".
- Things happened**: A section showing a sequence of icons (a face, a star, a speech bubble) followed by the text "Welcome to the world of creative coding!".
- Welcome to the world of creative coding!**: A list of things you can do:
  - Start coding by clicking Create a Sketch button above.
  - Browse around the site, follow other users and check out their sketches
  - Comment and sketches to show your appreciation.
  - Make a change in the code on any sketch, and save your changes as a fork.
  - If you are teaching coding, [create a class](#) to collaborate with students.
- Events coming up**: An event card for "EYEO Festival 2019" with the text "Join the conversation. Converge to Inspire. Tickets on sale now."
- Coding Challenge: Tiny Sketch**: A challenge card with the text "Can you push your limits to create a sketch under 200 characters with P5.js? Join the challenge to win a free ticket to EYEO Festival in June."
- Consideration by**: A section with a red horizontal bar.

# 作品の編集画面はこんな感じ



The screenshot shows the OpenProcessing.org sketch editor interface. At the top, there's a header with a signal icon, 'docomo', time '18:22', battery level '29%', and a refresh button. Below the header, the URL 'openprocessing.org' is displayed. The main area has tabs for 'mySketch' and 'QRコード作成【無料】: QRの2次元コード', 'QRコード作成【無料】: QRのスクリプト', and 'QRコード作成【無料】: QRのスケッチ'. The 'mySketch' tab is active, showing the following P5.js code:

```
let x, y;

function setup() {
  createCanvas(windowWidth, windowHeight);
  fill(255);
  x = windowHeight / 2;
  y = windowHeight / 2;
}

function draw() {
  let mx = mouseX;
  let my = mouseY;

  x = x + (mx - x) * 0.1;
  y = y + (my - y) * 0.1;

  let s = `x:${parseInt(x)}, y:${parseInt(y)}`;

  background(0);
  textSize((x + y) / 16);
  text(s, x, y);
}
```

At the bottom of the code editor, there are navigation icons: back, forward, upload, and download.

# 再生ボタンで作品を見る

The screenshot shows a web browser displaying the openprocessing.org website. The URL is https://www.openprocessing.org/sketch/671405. The page title is "TextTest - OpenProcessing". The main content is a Processing sketch titled "mySketch". The code is as follows:

```
1 let x, y;
2
3 function setup() {
4   createCanvas(windowWidth, windowHeight);
5   fill(255);
6   x = windowHeight / 2;
7   y = windowHeight / 2;
8 }
9
10 function draw() {
11   let mx = mouseX;
12   let my = mouseY;
13
14   x = x + (mx - x) * 0.1;
15   y = y + (my - y) * 0.1;
16
17   let s = `x:${parseInt(x)}, y:${parseInt(y)} `;
18
19   background(0);
20   textSize((x + y) / 16);
21   text(s, x, y);
22 }
```

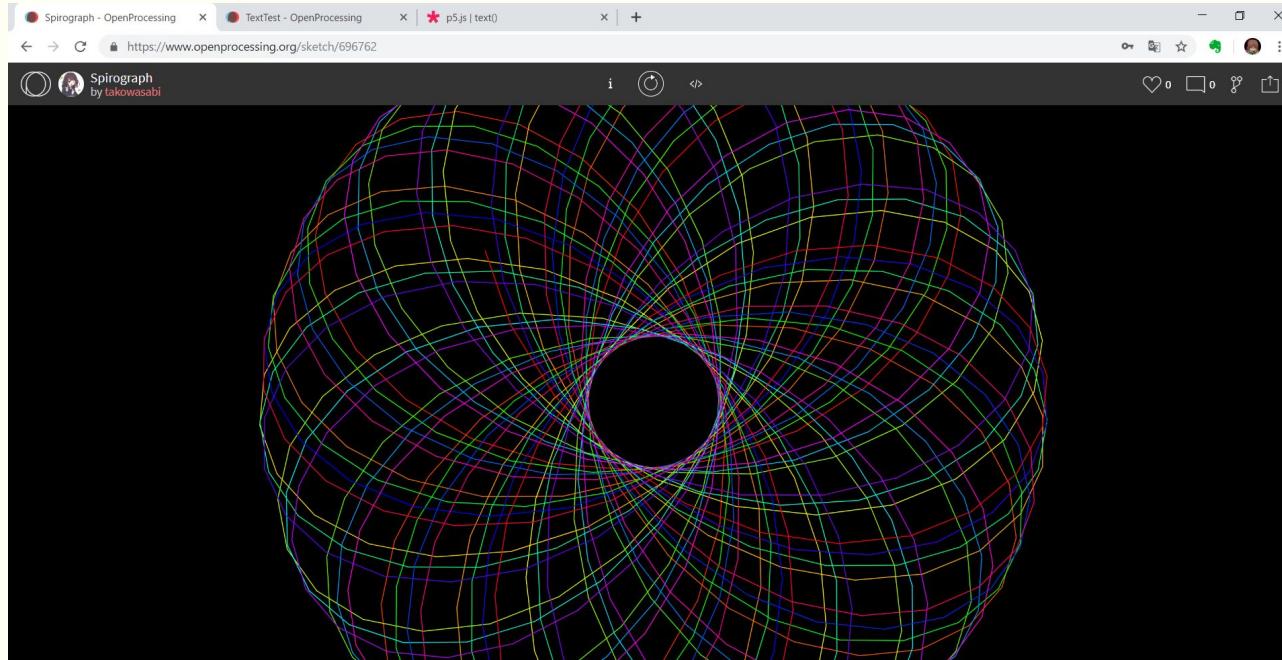
The browser's address bar shows the URL. The top right corner of the browser window displays the time (18:22), battery level (29%), and signal strength. The browser's header includes a play button, which is highlighted with a blue circle. A blue line connects this play button to another play button located in the code editor's header, which also has a blue circle around it.

**作品制作開始**

# 本日の作品

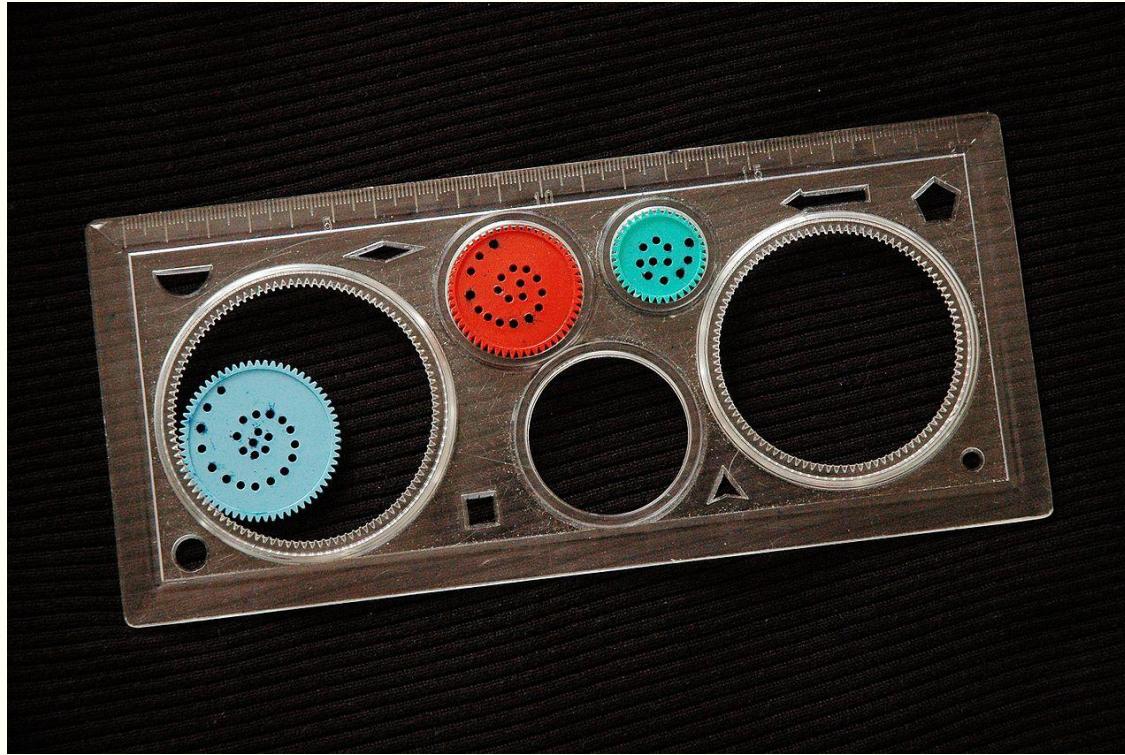
# 本日の作品

## スピログラフ (Spirograph)



スピログラフって何？

# スピログラフって何？



# スケッチブックを作ろう

まずは新規の  
スケッチブック(作品)  
を作成

The screenshot shows a user profile on the OpenProcessing website. The profile belongs to 'takowasaby'. It features a large circular placeholder for a profile picture. Below the placeholder is a 'Create a Sketch' button, which is highlighted with a green oval. To the right of the button, there's a section titled 'Things happened' with two small icons: a blue circle with a dot and a black star. Next to these icons is the text 'Welcome to the world of creative coding!'. Below this, there's a list of things you can do:

- Start coding by clicking Create a Sketch button above.
- Browse around the site, follow other users and check out their sketches
- Comment and ⌂ sketches to show your appreciation.
- Make a change in the code on any sketch, and save your changes as a fork.
- If you are teaching coding, create a class to collaborate with students.

At the bottom of this section, it says 'You will see more tips and ideas here to get inspired!'. To the right of this main content area, there's a sidebar titled 'Events coming up' featuring the EYEO festival logo and information about the EYEO Festival 2019.

**Events coming up**

**EYEO FESTIVAL**  
CODE • ART • DATA • DESIGN  
EYEO Festival 2019  
Join the conversation.  
Converge to Inspire.  
Tickets on sale now.

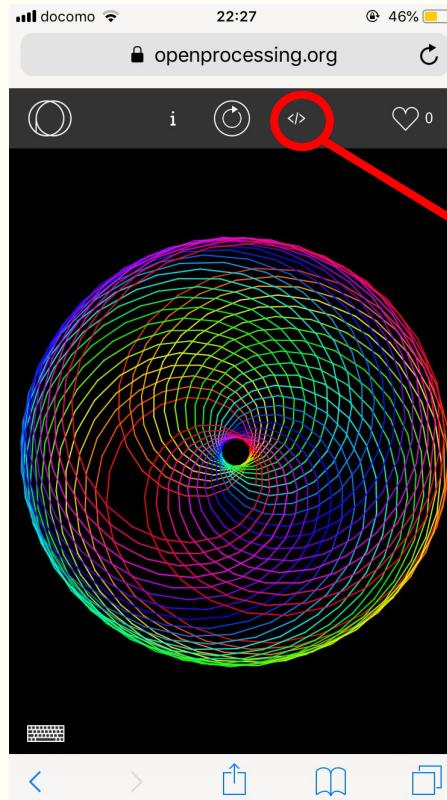
**Coding Challenge: Tiny Sketch**

Execution memory: {  
 //what can you create...  
}  
Execution done: {  
 //...in 200 characters  
}  
Can you push your limits to create a sketch under 200 characters with P5.js?  
Join the challenge to win a free ticket to EYEO Festival in June.

Consideration by

# 携帯の方はコードを追っていきましょう

<https://www.openprocessing.org/sketch/696762>



ここから  
コードが見れます

どうやって再現するの？

プログラムで再現したい

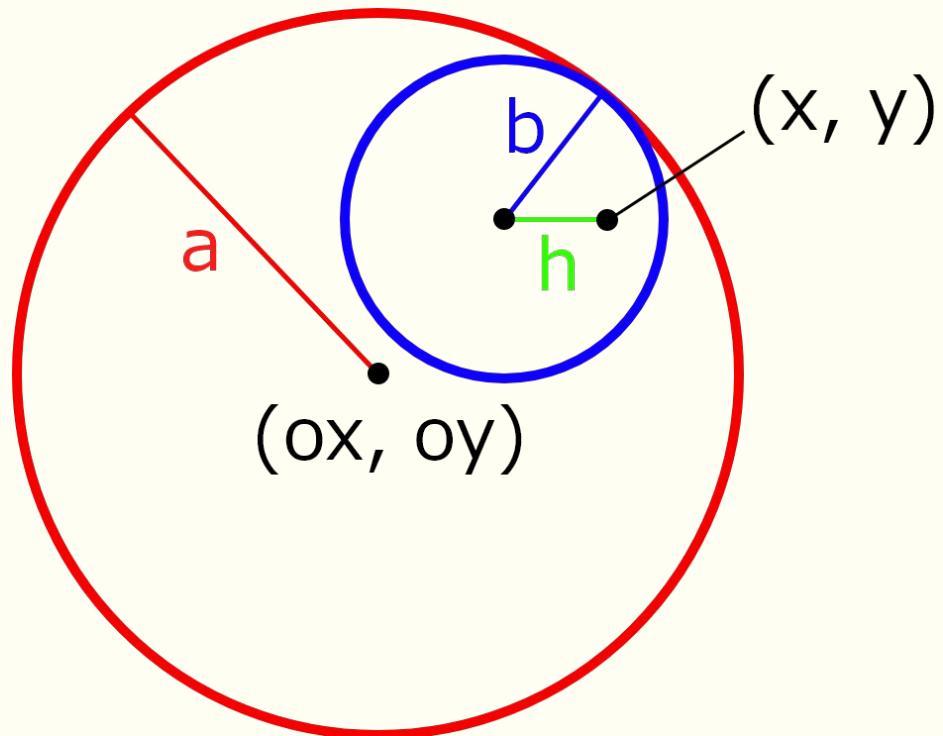


数式があるとプログラムが書きやすい



数式を立てます

# まずは変数定義



数式はこんな感じ

$$\begin{cases} x = (a - b) \cos \theta - h \cos \left( \frac{a+b}{b} \theta \right) \\ y = (a - b) \sin \theta - h \sin \left( \frac{a+b}{b} \theta \right) \end{cases}$$

# プログラム書きます

この状態がデフォルト

```
function setup() {  
    createCanvas(windowWidth, windowHeight);  
    background(100);  
}  
  
function draw() {  
    ellipse(mouseX, mouseY, 20, 20);  
}
```

# プログラム書きます

この状態がデフォルト

```
function setup() {  
    createCanvas(windowWidth, windowHeight);  
    background(100);  
}
```

始めにやる処理

```
function draw() {  
    ellipse(mouseX, mouseY, 20, 20);  
}
```

描画の処理

# 要らない部分は消しましょう

```
function setup() {  
    createCanvas(windowWidth, windowHeight);  
    background(100);  
}  
  
function draw() {  
    ellipse(mouseX, mouseY, 20, 20),  
}
```

# こうなればOK

```
function setup() {  
    createCanvas(windowWidth, windowHeight);  
    background(100);  
}  
  
function draw() {  
}
```

# プログラムでも変数を作ります

```
let a = 0.0;
```

```
function setup() {  
    createCanvas(windowWidth, windowHeight);  
    background(100);  
}
```

```
function draw() {  
    a = mouseX;  
}
```

# 変数の定義 (JavaScript)

let a = 0.0 ;  
変数 a は 0.0 です

# 変数への値の変更(JavaScript)

a    =    mouseX;

a    は    mouseX(変数)と同じ値 です

mouseX:マウスのX座標を表す

# 他の変数も同じ場所で定義

```
let a = 0.0;  
let b = 0.0;  
let h = 0.0;  
let x1 = 0.0;  
let y1 = 0.0;  
let x2 = 0.0;  
let y2 = 0.0;  
let theta1 = 0.0;  
let theta2 = 0.0;  
...
```

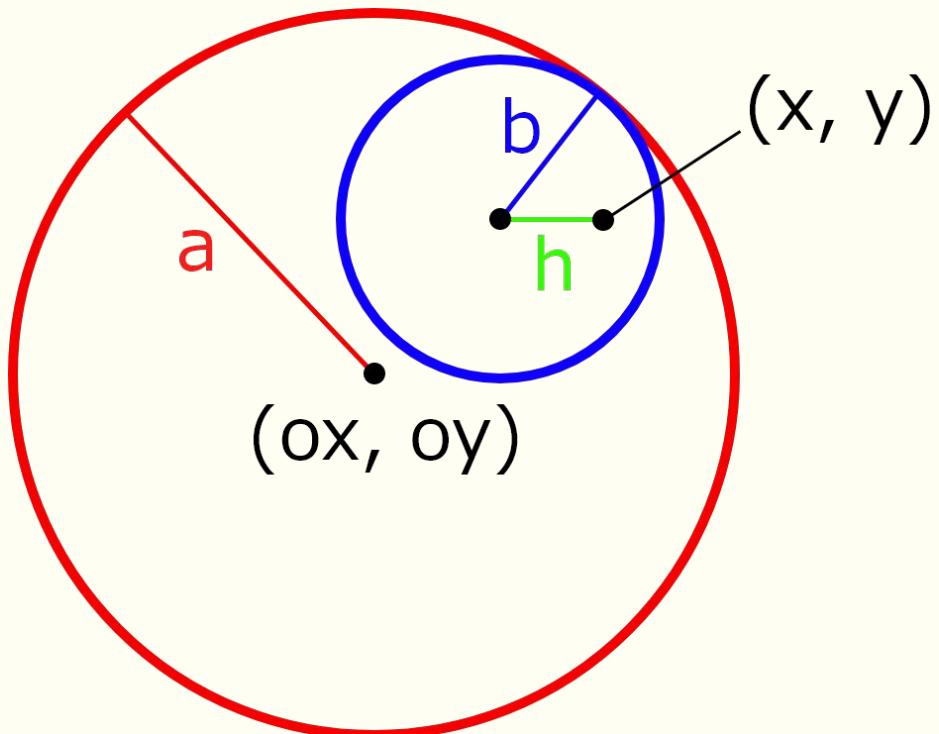
# a, b, hの値を設定

```
...
function draw() {
    a = mouseX;
    b = windowWidth / 2;
    h = mouseY;
}
```

mouseY:マウスのY座標を表す

windowWidth:画面の横幅

# 変数の意味を再確認



# 曲線の描画

プログラムで曲線を描くのは大変



たくさん線分を書いて曲線に近づける



少しずつ  $\theta$  をずらして線分を引いていく

たくさん線分を描くために

プログラムでは、たくさんの処理を

繰り返し行うときに for文を使う

```
for (let i = 0; i < 1024; i++) { }
```

# 部分ごとの意味

i : 繰り返した回数を数えるカウンタ

for (let i = 0; i < 1024; i++)

カウンタを用意    カウンタが1024以下なら繰り返す

正の方向にカウントしていく

# たくさん線分を描くためには

...

```
function draw() {  
    a = mouseX;  
    b = ox;  
    h = mouseY;  
    for (let i = 0; i < 1024; i++)  
    {
```

ここに、少しずつ線分を描く処理を書く

```
}
```

```
}
```

$i$ を用いて $\theta_1$ を決定し、ずらしたものを $\theta_2$ とする

```
theta1 = (i / 32) * 3 * PI;  
theta2 = ((i + 1) / 32) * 3 * PI;
```

PI:いわゆる  $\pi$  を表す

iを用いてθ1を決定し、ずらしたものをθ2とする

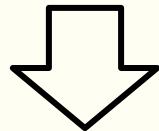
...

```
for (let i = 0; i < 1024; i++)  
{  
    theta1 = (i / 32) * 3 * PI;  
    theta2 = ((i + 1) / 32) * 3 * PI;  
}
```

...

$\theta$ をもとにx, yを求める

$$\begin{cases} x = (a - b) \cos \theta - h \cos \left( \frac{a + b}{b} \theta \right) \\ y = (a - b) \sin \theta - h \sin \left( \frac{a + b}{b} \theta \right) \end{cases}$$



```
x = (a - b) * cos(theta1) + h * cos(((a - b) / b) * theta);  
y = (a - b) * sin(theta1) + h * sin(((a - b) / b) * theta);
```

# $\theta_1, \theta_2$ でそれぞれ(x1, y1), (x2, y2)を求める

```
...
for (let i = 0; i < 1024; i++)
{
    theta1 = (i / 32) * 3 * PI;
    theta2 = ((i + 1) / 32) * 3 * PI;

    x1 = (a - b) * cos(theta1) + h * cos(((a - b) / b) * theta1);
    x2 = (a - b) * cos(theta2) + h * cos(((a - b) / b) * theta2);

    y1 = (a - b) * sin(theta1) + h * sin(((a - b) / b) * theta1);
    y2 = (a - b) * sin(theta2) + h * sin(((a - b) / b) * theta2);
}

...
```

線分を描くためには

```
line(x1, y1, x2, y2);
```

座標(x1, y1)の点と座標(x2, y2)の点の間に線を引く

# 線分を描く部分を追加

```
...
for (let i = 0; i < 1024; i++)
{
    theta1 = (i / 32) * 3 * PI;
    theta2 = ((i + 1) / 32) * 3 * PI;

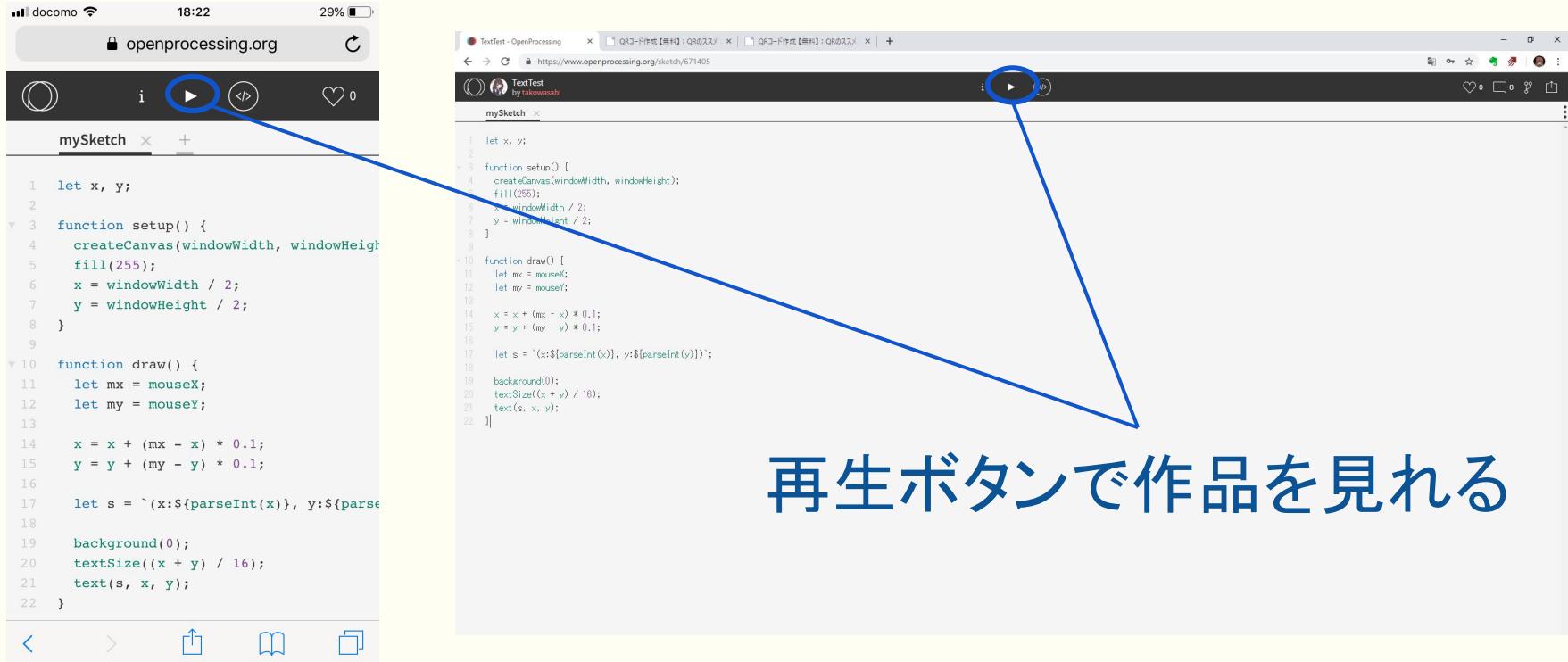
    x1 = (a - b) * cos(theta1) + h * cos(((a - b) / b) * theta1);
    x2 = (a - b) * cos(theta2) + h * cos(((a - b) / b) * theta2);

    y1 = (a - b) * sin(theta1) + h * sin(((a - b) / b) * theta1);
    y2 = (a - b) * sin(theta2) + h * sin(((a - b) / b) * theta2);

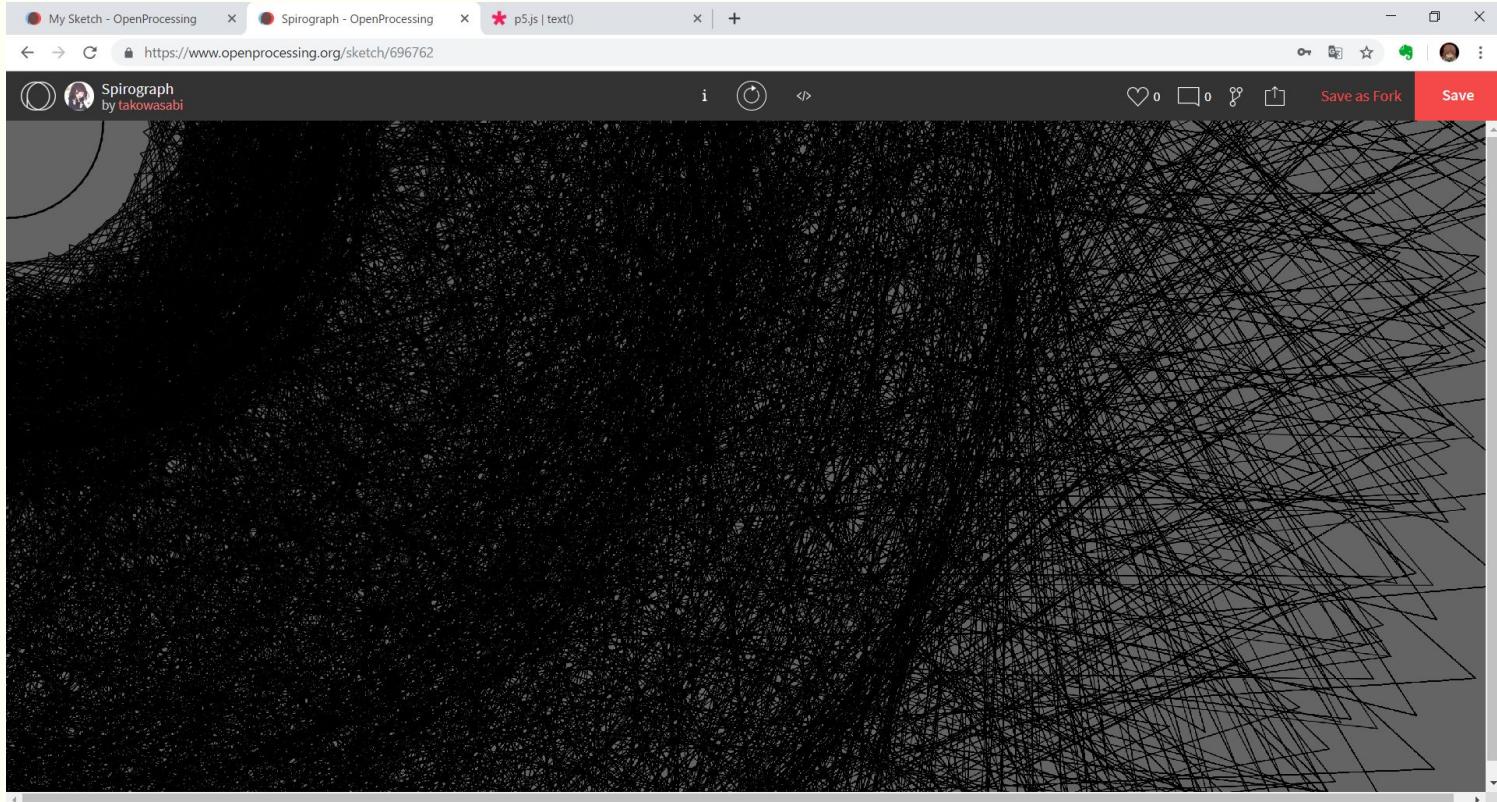
    line(x1, y1, x2, y2);
}
...

```

# ちょっと動かしてみよう



# こんな感じになるはず



# 完成版と何が違うのか？

- 左上が中心になってしまっている
- 書いた線が消えない
- 色がついていない

# 完成版と何が違うのか？

- 左上が中心になってしまっている
- 書いた線が消えない
- 色がついていない

# processingでの座標



実はprocessingでは原点が左上にある  
このため、中心に動かす場合は描画するときに原点を移動する必要がある

# 画面の真ん中を原点とする

```
...
let theta2 = 0.0;
let ox;
let oy;

function setup() {
...
}

...
function draw() {
    ox = windowWidth / 2;
    oy = windowHeight / 2;
...
}
```

# 描画の時に原点の分ずらす

```
line(x1, y1, x2, y2);
```



```
line(ox + x1, oy + y1, ox + x2, oy + y2);
```

# 完成版と何が違うのか？

- 左上が中心になってしまっている
- 書いた線が消えない
- 色がついていない

何故線がどんどん増えていくのか？

マウスの座標が変わるたびに  
キャンバスの上に線を何度も描いている感じ



線を描くたびにキャンバスを綺麗にする

# background(0)で背景を黒く塗りつぶす

...

```
function draw() {
```

```
    background(0);
```

```
a = mouseX;
```

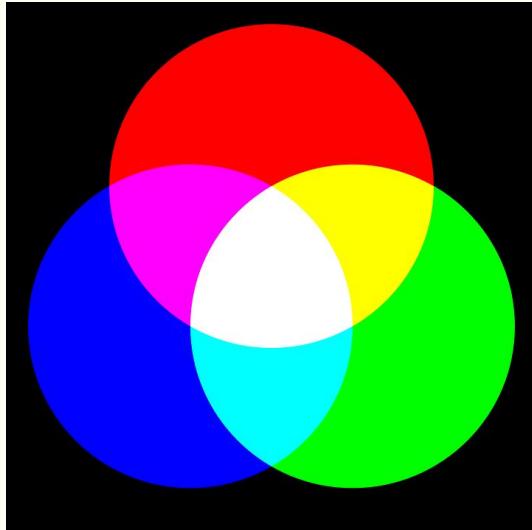
...

# 完成版と何が違うのか？

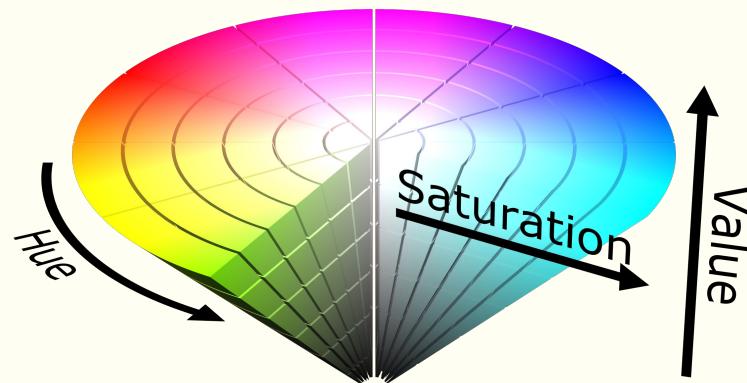
- 左上が中心になってしまっている
- 書いた線が消えない
- 色がついていない

# 色の表し方

RGB



HSB



# 色の指定の形式を変更

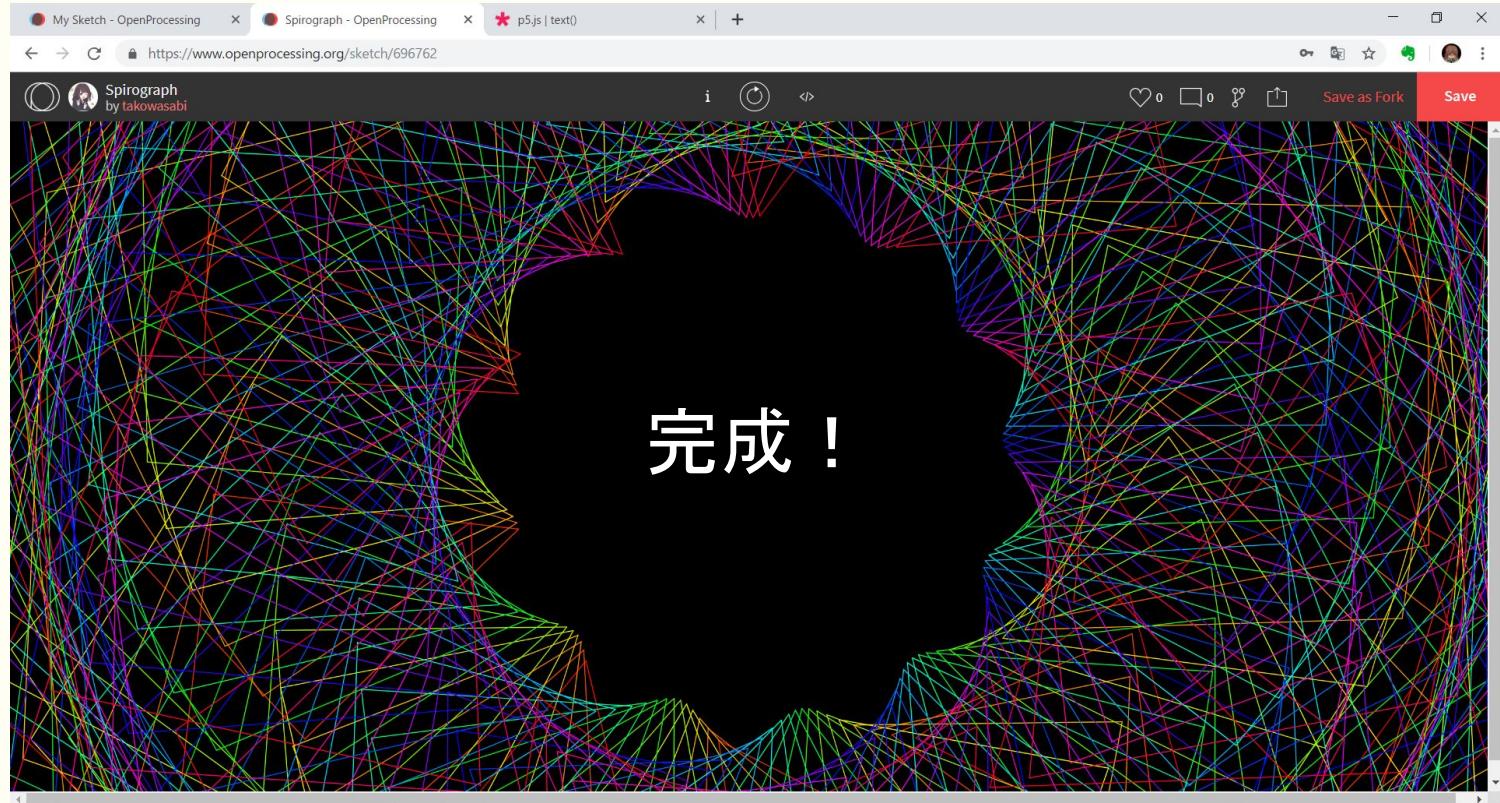
```
...  
function setup() {  
    createCanvas(windowWidth, windowHeight);  
    colorMode(HSB, 1023);  
}  
...
```

## 描画の前にStrokeで色を指定

...

```
stroke(i, 1023, 1023);  
                    
line(ox + x1, oy + y1, ox + x2, oy + y2);
```

...



おまけ

さらに詳しく知りたい方は

以下のリファレンスを参照してください

<https://p5js.org/reference/>

※リファレンスにあるサンプルコードはなんと

書き換え可能です、リファレンス見てるだけで楽しい！

# 実際に作品をアップロードしてみよう in PC

まずは新規の  
スケッチブック(作品)  
を作成

The screenshot shows a user profile page on the OpenProcessing website. The profile belongs to 'takowasaby'. At the top, there is a large circular placeholder for a profile picture. Below it is a smaller circular placeholder with a person icon. To the right of the placeholders, the username 'takowasaby' is displayed, along with a link to 'Edit Profile'. Below the username, there are statistics: 'Activity 0 Sketches 0 Hearts 0 Followers 0 Following'. A green line with an arrow points from the text 'まずは新規のスケッチブック(作品)を作成' to the 'Create a Sketch' button, which is highlighted with a red oval.

Things happened

Welcome to the world of creative coding!

Here are super awesome things you can do:

- Start coding by clicking Create a Sketch button above.
- Browse around the site, follow other users and check out their sketches
- Comment and ⌂ sketches to show your appreciation.
- Make a change in the code on any sketch, and save your changes as a fork.
- If you are teaching coding, [create a class](#) to collaborate with students.

You will see more tips and ideas here to get inspired!

Events coming up

**EYEOSF 2015**

CODE • ART • DATA • DESIGN  
EYE Festival 2015  
Join the conversation.  
Converge to Inspire.  
Tickets on sale now.

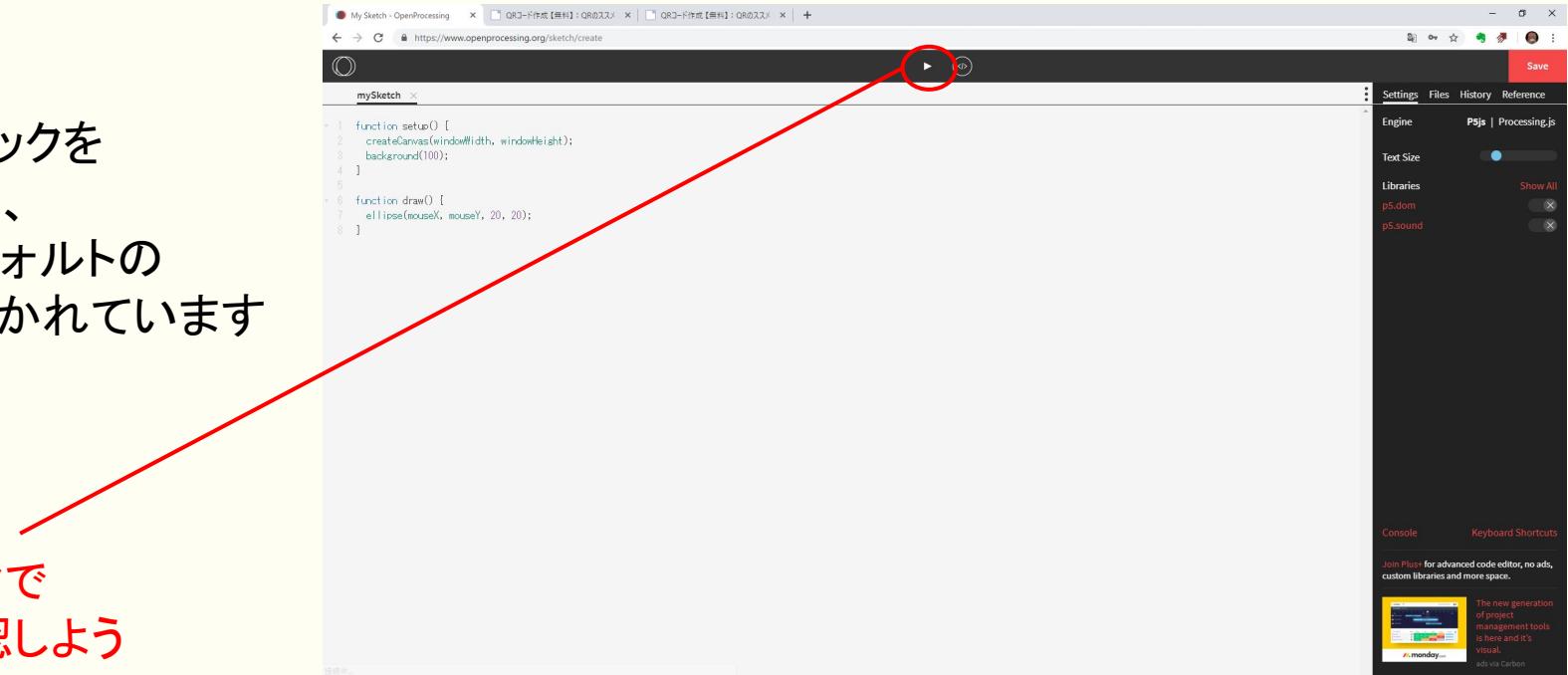
Coding Challenge: Tiny Sketch

Execution memory: {  
 //what can you create...  
}  
Execution done: {  
 //...in 200 characters  
}  
Can you push your limits to create a sketch under 200 characters with P5.js?  
Join the challenge to win a free ticket to EYE Festival in June.

Consideration by

# 実際に作品をアップロードしてみよう in PC

スケッチブックを作成すると、すでにデフォルトのコードが書かれています

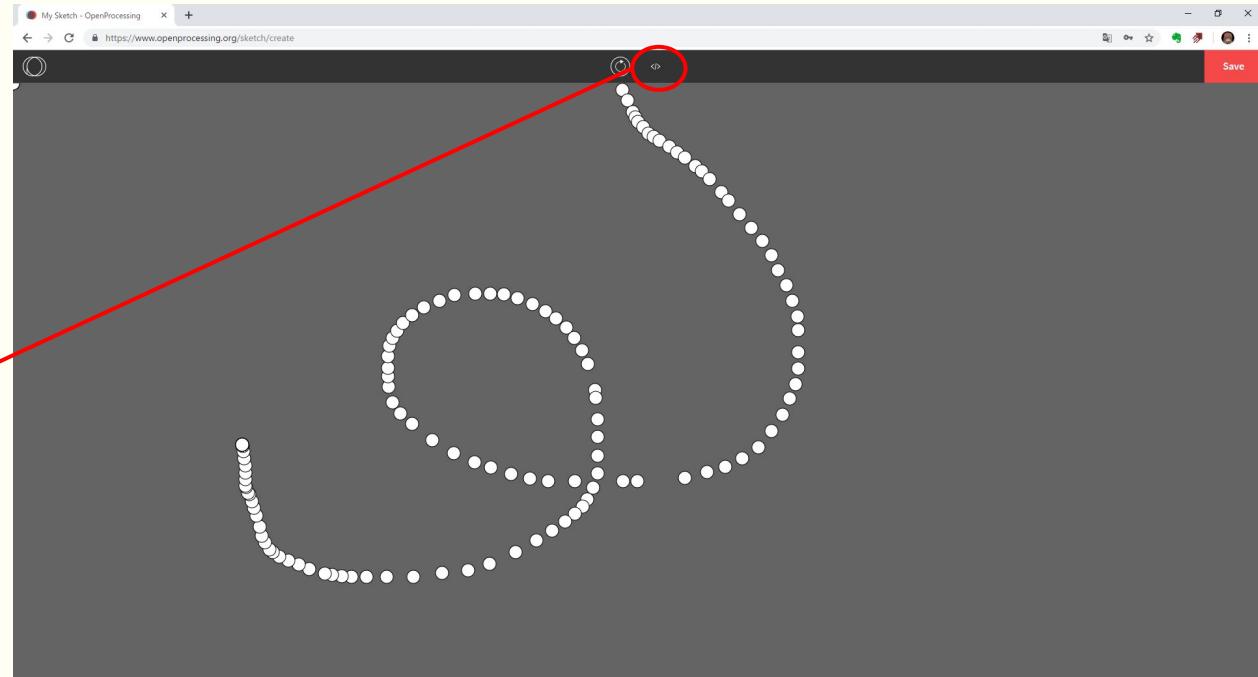


再生マークで挙動を確認しよう

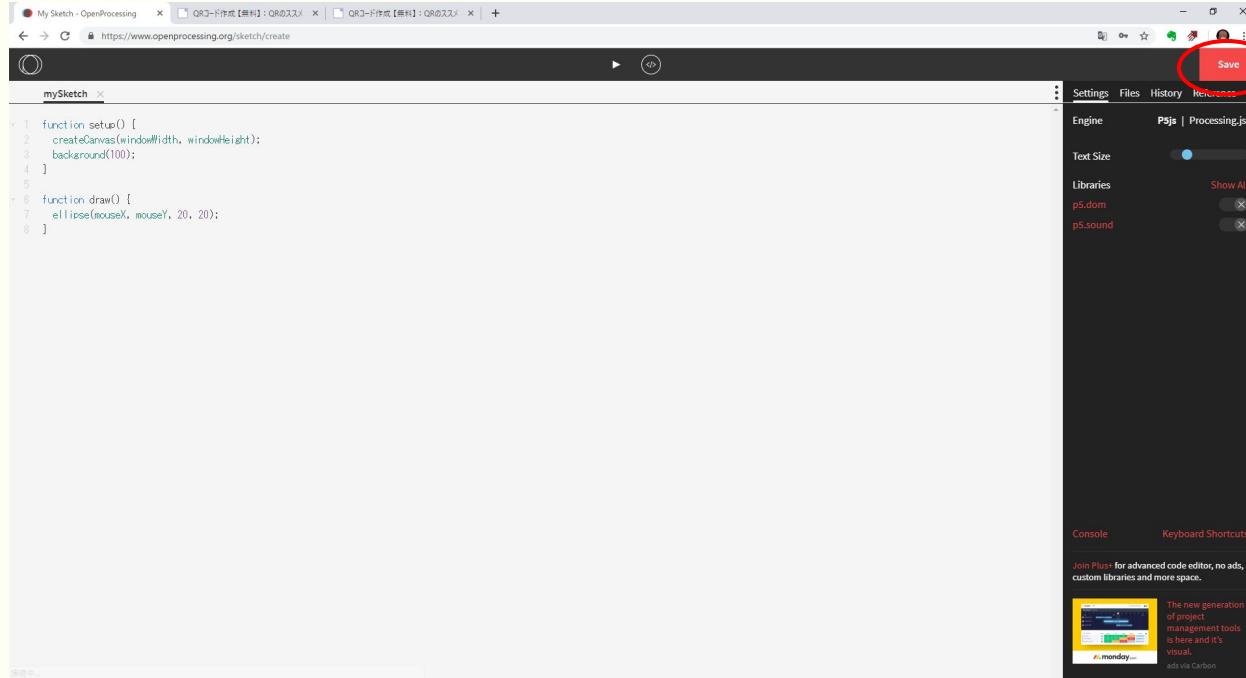
# 実際に作品をアップロードしてみよう in PC

マウスの動きに合わせて  
白丸が出来ます

確認したら  
〈/〉マークで  
編集画面に戻ろう



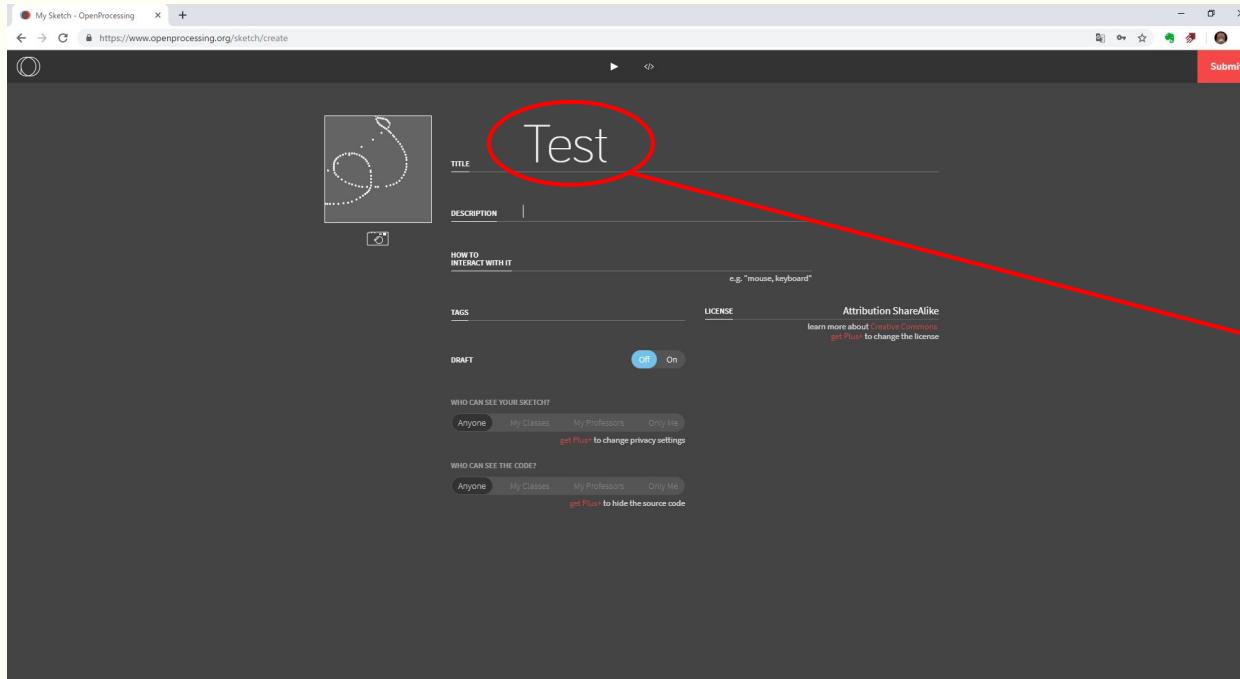
# 実際に作品をアップロードしてみよう in PC



では、作品をアップロード  
していきましょう

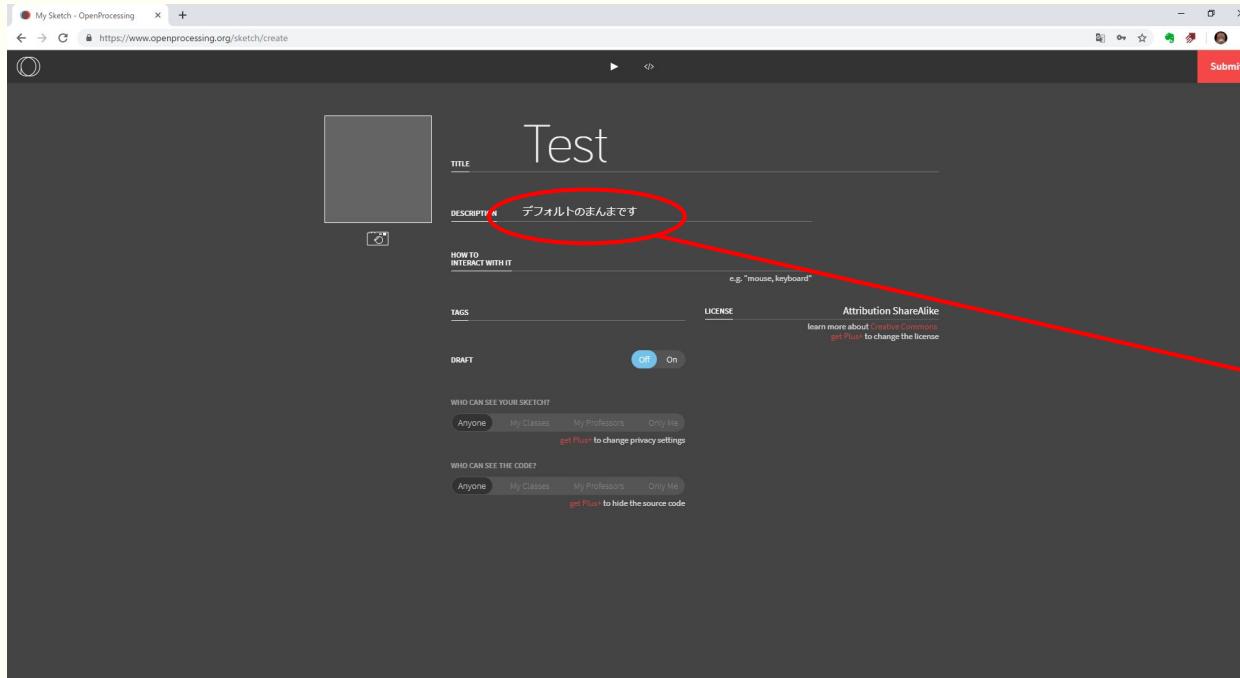
「Save」を押して  
アップロード画面へ

# 実際に作品をアップロードしてみよう in PC



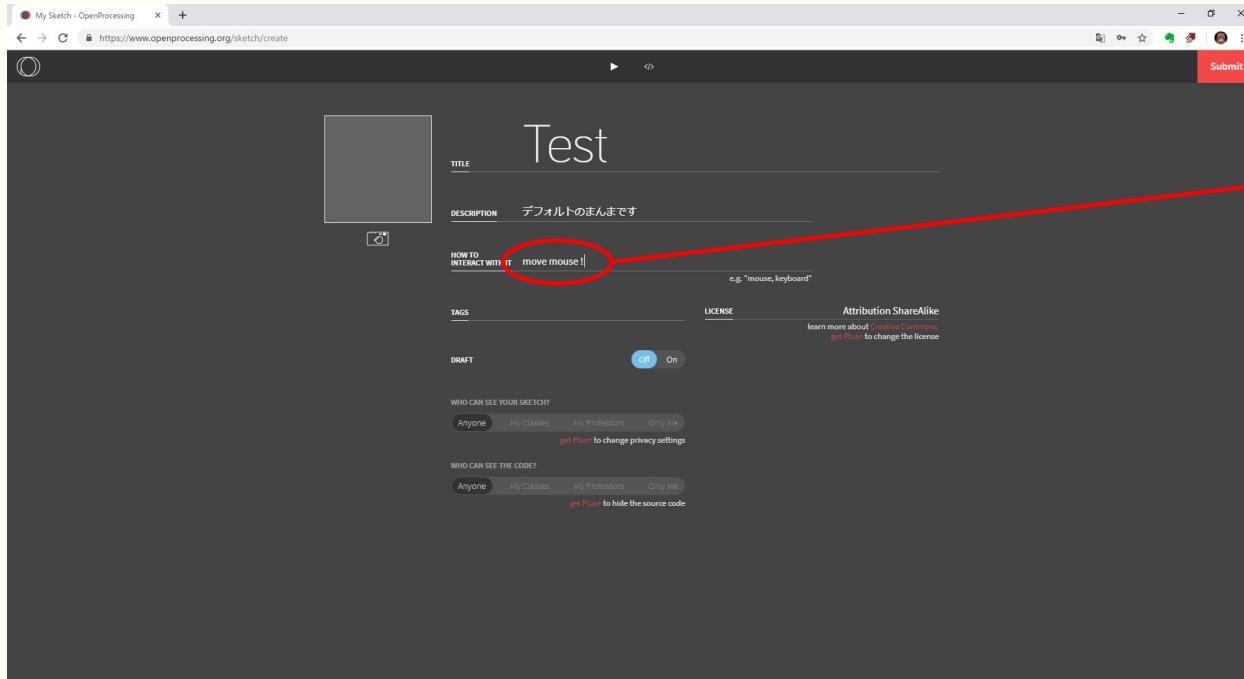
ここに作品の  
タイトルを入力します

# 実際に作品をアップロードしてみよう in PC



ここに作品の  
説明を書きます

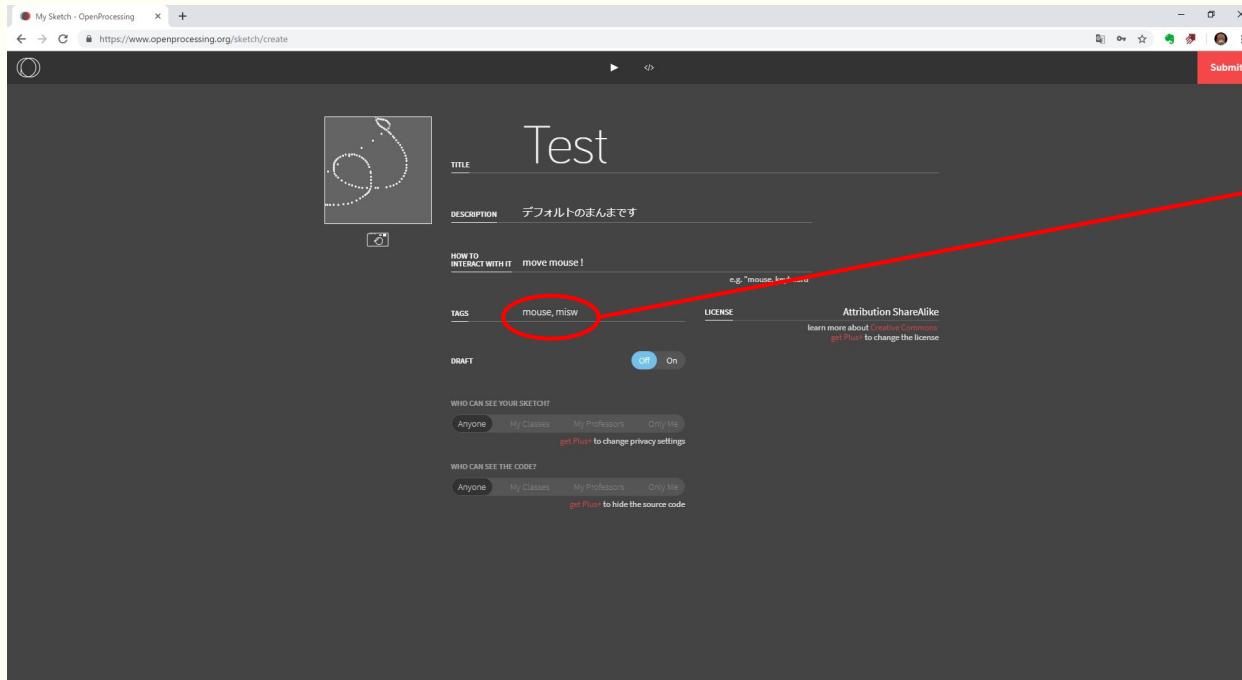
# 実際に作品をアップロードしてみよう in PC



ここに作品の  
操作説明を書きます

この説明文は作品を見る  
ときに表示されるようになっています

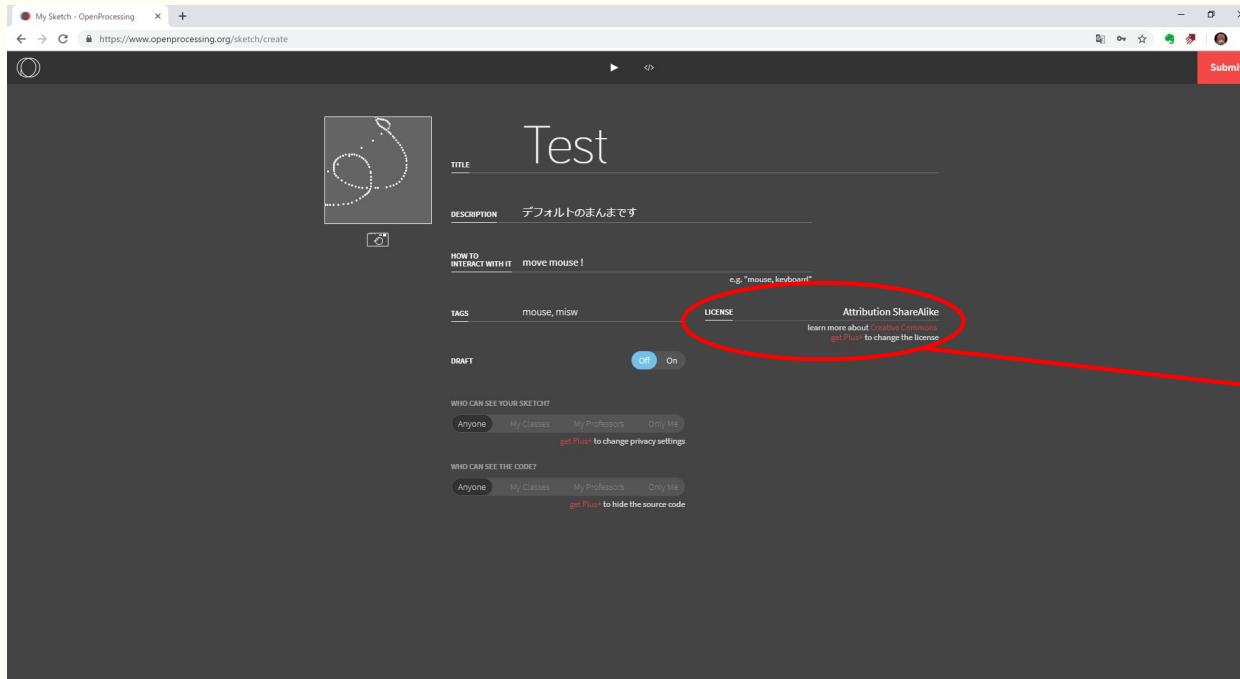
# 実際に作品をアップロードしてみよう in PC



ここに作品に  
つけたいタグを  
書きます

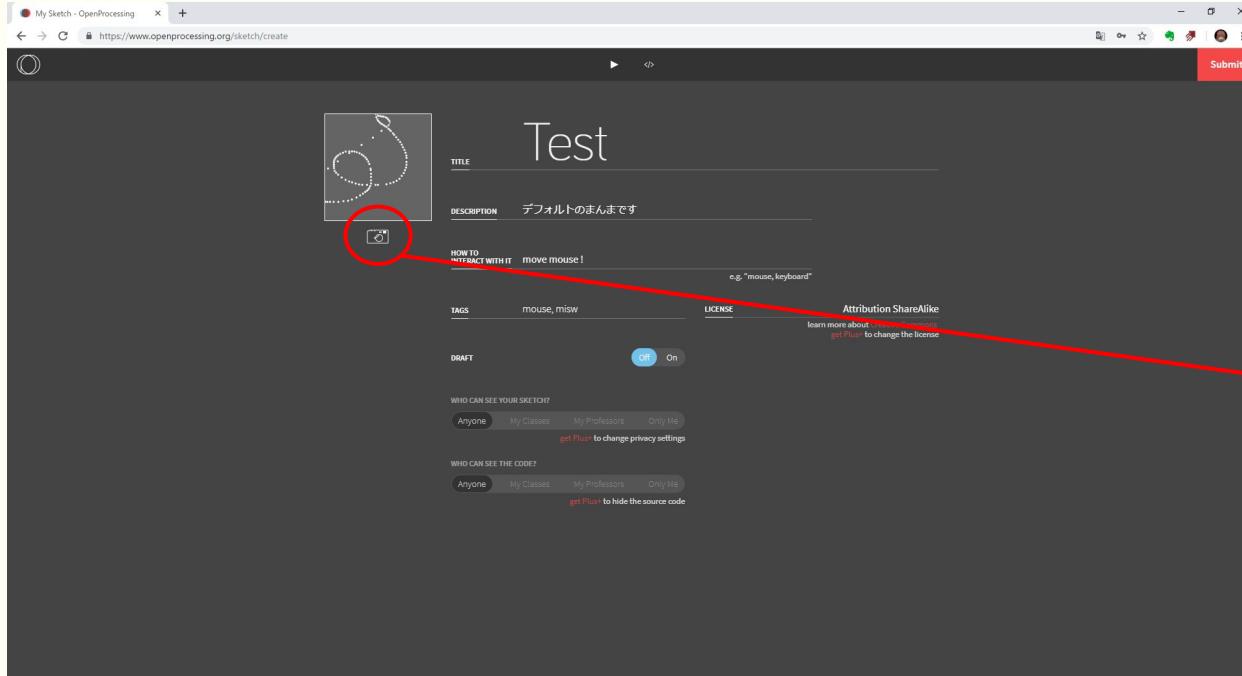
「misw」タグを  
お忘れなく

# 実際に作品をアップロードしてみよう in PC



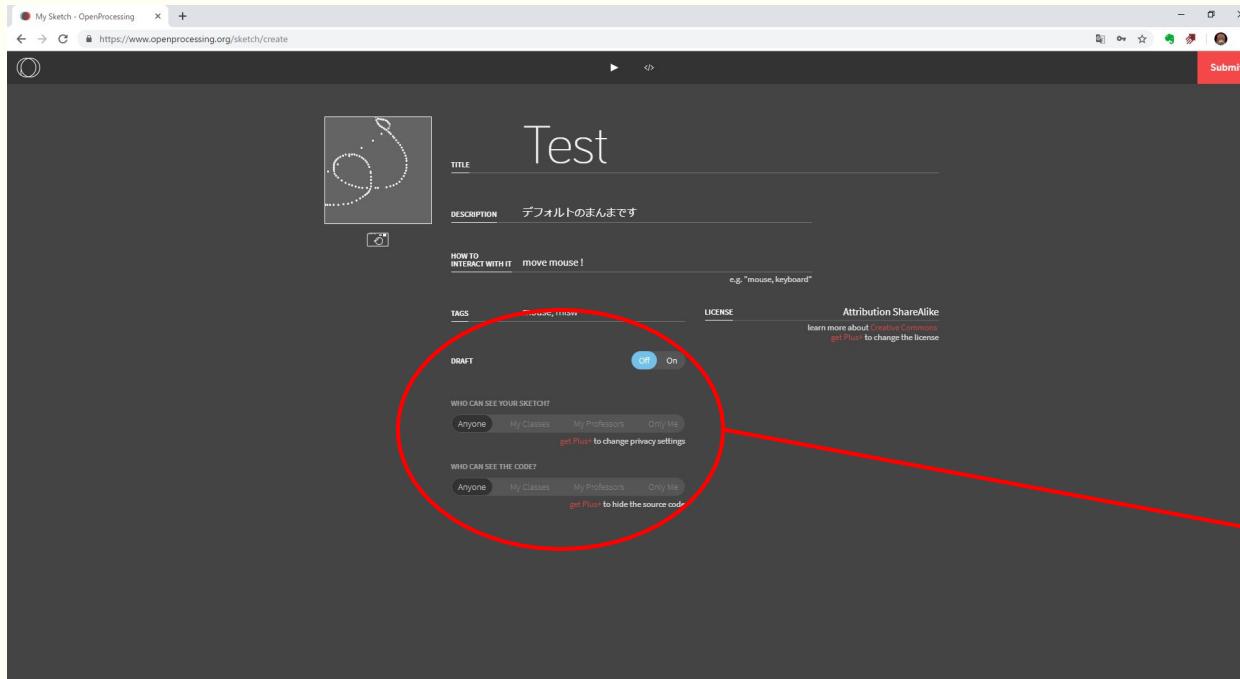
LICENSEは  
とりあえず気にしなくて大  
丈夫です

# 実際に作品をアップロードしてみよう in PC



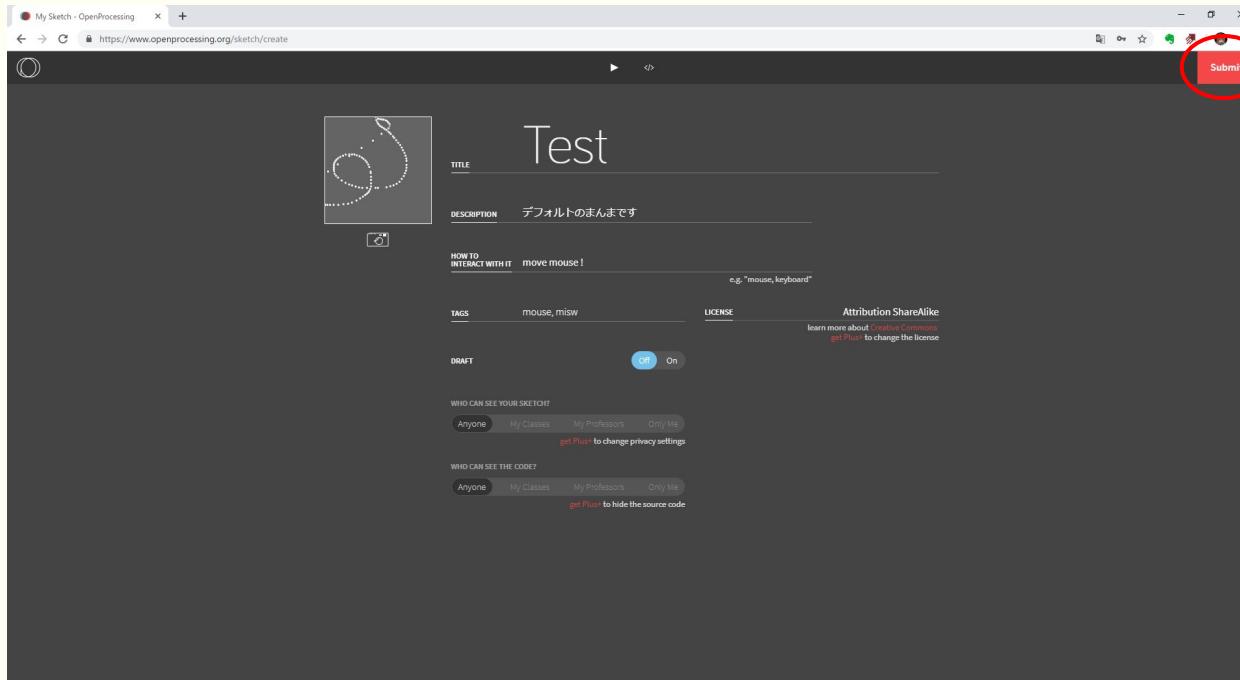
Cameraボタンで  
スクリーンショットを  
撮り直せます

# 実際に作品をアップロードしてみよう in PC



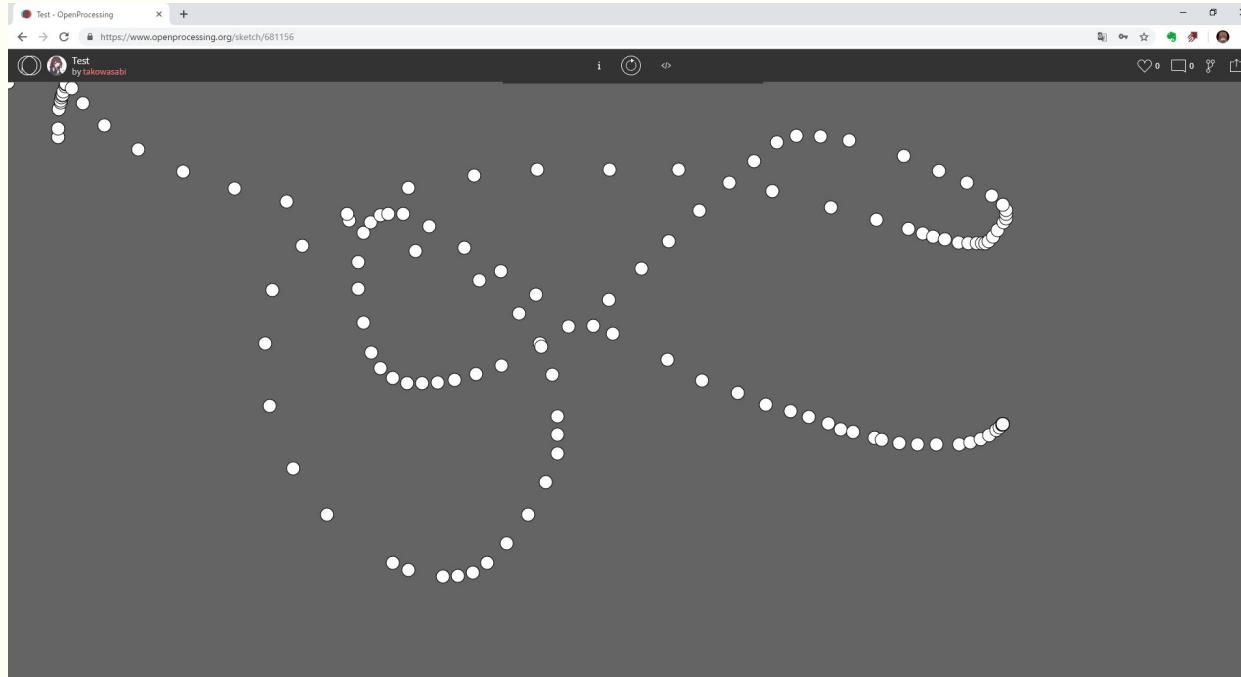
これら辺で  
アクセス範囲を  
指定できます

# 実際に作品をアップロードしてみよう in PC



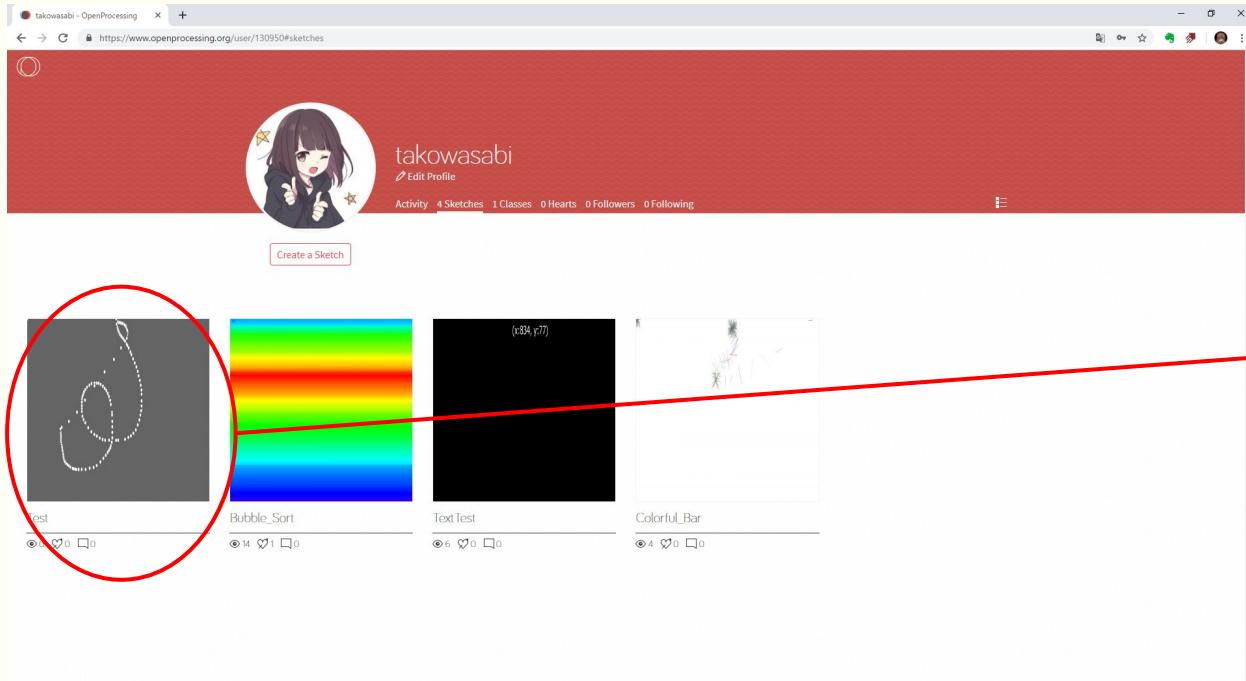
入力が終わったら  
「Submit」で公開！

# 実際に作品をアップロードしてみよう in PC



作品の視聴画面に行けたら公開完了です！！！

# 実際に作品をアップロードしてみよう in PC



公開した作品はマイページのSketchesから確認できます