

日本大学文理学部 オープンキャンパス

やさしいプログラミング実習

発展編「クリックで動きを止めよう」

日本大学 文理学部 情報科学科

実践編の復習

```
var x = 200, y = 10;
var r = 10;

function setup() {
  createCanvas(400, 400);
}

function draw() {
  background(220);
  fill(255);
  ellipse(x, y, r*2, r*2);
  y = y + 1;
  if (y > 400) {
    fill(0);
    text("Game Over", 180, 200);
  }
}
```

実践編では、こんなプログラムを作りました。

円の描画位置
を下に1ずらす

円が下まで
行ったときに実行

「Game Over」
と表示

発展編で何をするの？

落ちるボールをクリックすると、動きが止まるようにしよう。

さきほどのプログラムをまだ入力していない人は、
Webブラウザ (Google Chromeなど) で
editor.p5js.org を開いて、プログラムを打ち込もう。

3. マウスボタンをクリックしたときの 処理の書き方を学ぼう

マウスボタンをクリックしたときの処理は、
mouseClickedという新しい関数を作って、そこに書く。

```
function mouseClicked() {
```

```
    マウスボタンがクリックされたときに行う処理
```

```
}
```

アニメーションを止める ➡ noLoopという命令を使う

とりあえず、どこか(どこでもいい)がクリックされたら、
アニメーションが止まるようにしよう。

さっそくやってみよう

このプログラムの一番下に次の関数を追加しよう。

```
function mouseClicked() {  
  noLoop();  
}
```

[注意1] 必ず半角で入力すること。

[注意2] 大文字・小文字を正確に区別すること。

[注意3] 最後にセミコロンがある行は、セミコロンを忘れないこと。

The image shows a browser window with the p5.js Web Editor. The address bar shows `editor.p5js.org`. The editor has a menu bar with `File`, `Edit`, `Sketch`, and `Help`. Below the menu bar are controls for `Auto-refresh` and the user name `Trusting huckleberry`. The main area is split into a code editor on the left and a preview window on the right.

```
5  createCanvas(400, 400);
6  }
7
8  function draw() {
9    background(220);
10   fill(255);
11   ellipse(x, y, r*2, r*2);
12   y = y + 1;
13   if (y > 400) {
14     fill(0);
15     text("Game Over", 180, 200);
16   }
17 }
18
19 function mouseClicked() {
20   noLoop();
21 }
22
```

The preview window shows a gray background with a white circle at the top center. A green callout box with white text points to the circle, containing the text: `クリックすると、ボールが止まる`.

At the bottom of the code editor, a blue box highlights lines 19-21. A green callout box with white text points to this box, containing the text: `この3行を追加`.

At the bottom of the browser window, there is a `Console` area with a `Clear` button.

6. ボールをクリックしたときだけ アニメーションが止まるようにする

「○○が△△だったら□□する」は、実践編で学んだif文を使う。

```
if (  ) {  
      
}
```

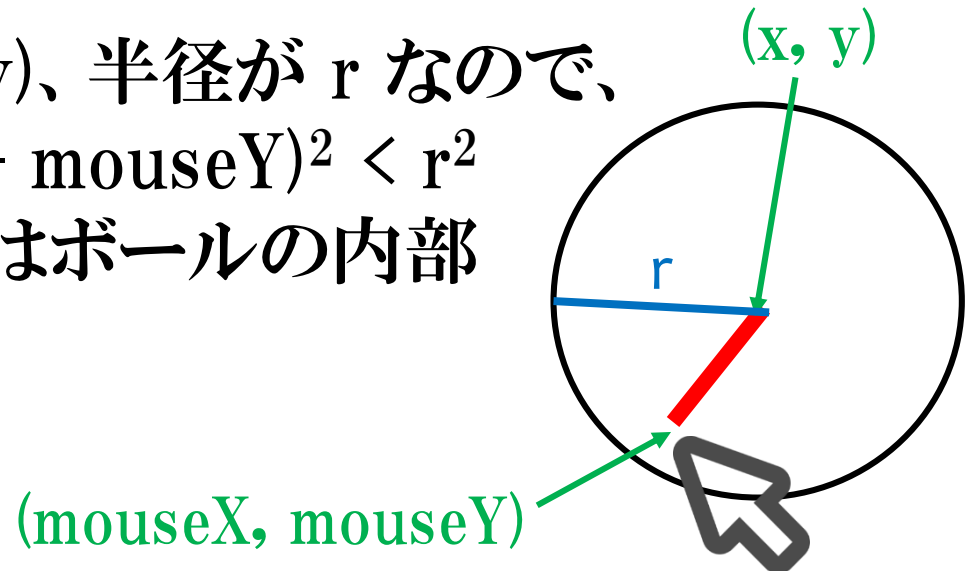
ボールの内側をクリックされたら

➡ マウスマウスカーソルの座標が、
円の領域の内部だったら

さて、どう考える？

マウ斯卡ーソルの座標と円を比べる

- * マウ斯卡ーソルの座標は、mouseX、mouseYという変数に入っている
 - * 最初からこれらの変数が定義されていて、マウ斯卡ーソルの座標が自動的に入るようになっている
- * ボールの中心が (x, y) 、半径が r なので、 $(x - \text{mouseX})^2 + (y - \text{mouseY})^2 < r^2$ なら、マウ斯卡ーソルはボールの内部



さっそくやってみよう

```
function mouseClicked() {  
  noLoop();  
}
```

この部分を次のように書き換えよう。

```
function mouseClicked() {  
  if ((x - mouseX) * (x - mouseX) +  
      (y - mouseY) * (y - mouseY) < r * r) {  
    noLoop();  
  }  
}
```

半角、大文字・小文字、注意！
セミコロン忘れ、注意！

p5.js Web Editor

editor.p5js.org

File Edit Sketch Help

Log in or Sign up

Auto-refresh Trusting huckleberry

sketch.js

```
8 function draw() {
9   background(220);
10  fill(255);
11  ellipse(x, y, r*2, r*2);
12  y = y + 1;
13 }
14
15
16 }
17
18
19 function mouseClicked() {
20   if ((x - mouseX) * (x - mouseX) +
21       (y - mouseY) * (y - mouseY) < r * r) {
22     noLoop();
23   }
24 }
25
```

Preview

ここが変更箇所

ボールの内部をクリックしたときのみ、アニメーションが停止

Console Clear

やってみよう

ボールの内部がクリックされたら、
アニメーションを止めるだけでなく、
別ウィンドウで「You won!」というメッセージを
表示するようにしてみよう。

【ヒント】 別ウィンドウで「You won!」と表示するには、
`alert("You won!");` という命令を使う。

まとめ

発展編では、

- * マウスボタンがクリックされたときの処理
- * マウスカーソルの位置が円の内部か判定する方法を学びました。

やさしいプログラミング実習は、これで終わりです。
Processingを用いたプログラミングの入門書やwebサイトがいろいろありますので、ぜひ学んでみてください。

ご覧いただき、ありがとうございました。