

静岡北中学校 SSHインセンティブレクチャー

Processingでつくる Webサイト

2023年9月9日（土）10:30～12:00, 13:00～14:30

@教育棟・508教室

静岡理工科大学

情報学部 コンピュータシステム学科

幸谷（こうや） 智紀（ともりのり）

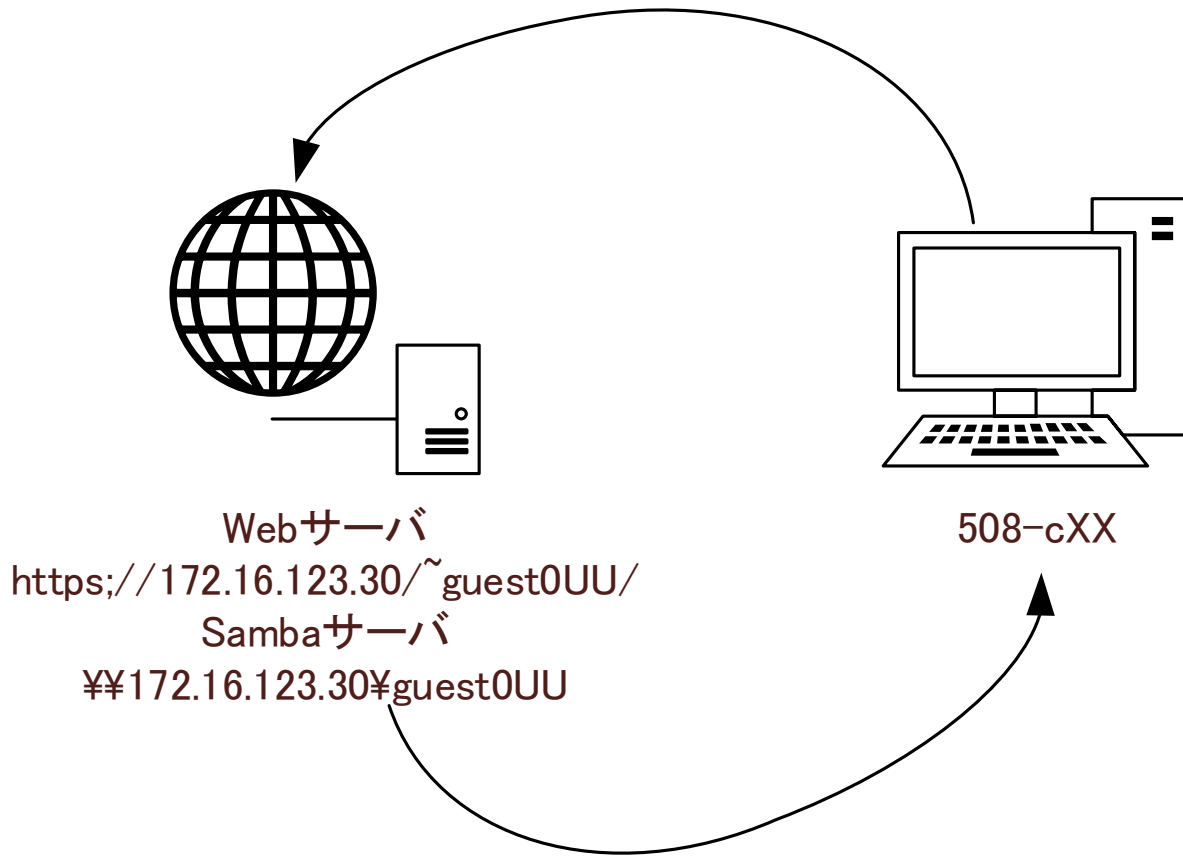
<https://cs-tklab.na-inet.jp/>



本日のメニュー

1. 実習環境と今日の目標
2. HTMLで目次を作ろう
3. サンプルアプリをながめてみよう
4. Processingで図形を描いてみよう
5. Processingでアニメーションを作ろう
6. Processingでカメラを作ってみよう
7. Processingでアルバムを作ってみよう
8. 今日のまとめ

1. 実習環境と今日の目標(1/2)



• 実習環境

- Webサーバ・・・ブラウザからアクセスしてWebページ用のデータを送付する

`https://172.16.123.30/~guest0UU/`

- Sambaサーバ・・・ファイルを共有して読み書きできるようにする

`¥¥172.16.123.30¥¥guest0UU`

• 今日の目標

- Processing(プロセッシング) というコンピュータ言語を使って、写真を集めたアルバムを作る。

1. 実習環境と今日の目標(2/2)

- 目次

自分の名前のWebページ

2023年9月9日(土)

1. [サンプルページ](#)
2. [絵を描く](#)
3. [アニメを描く](#)
4. [カメラ](#)
5. [アルバム](#)

参考 : p5js.org

自分の名前(ローマ字で) (c) 2023

- アルバム

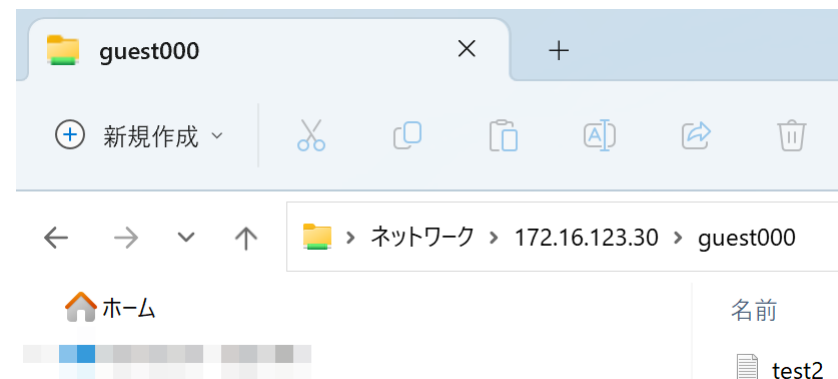


今日使うもの：まず使ってみよう

① ファイルマネージャ

→ ¥¥172.16.123.30¥guest0UU

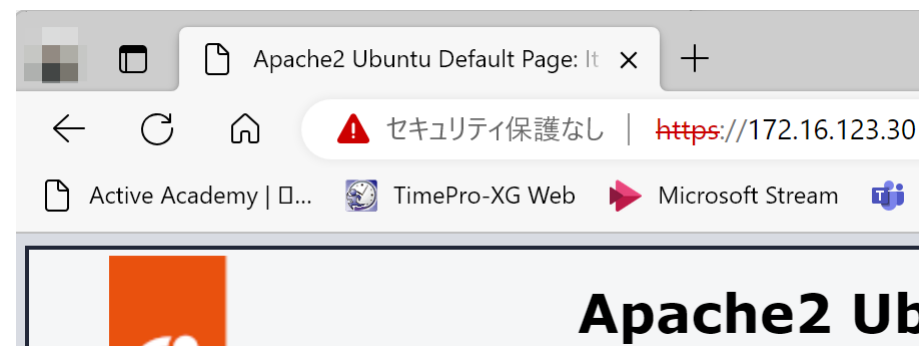
にアクセスできるか確認



② (Web) ブラウザ: Edge

→ <https://172.16.123.30/>

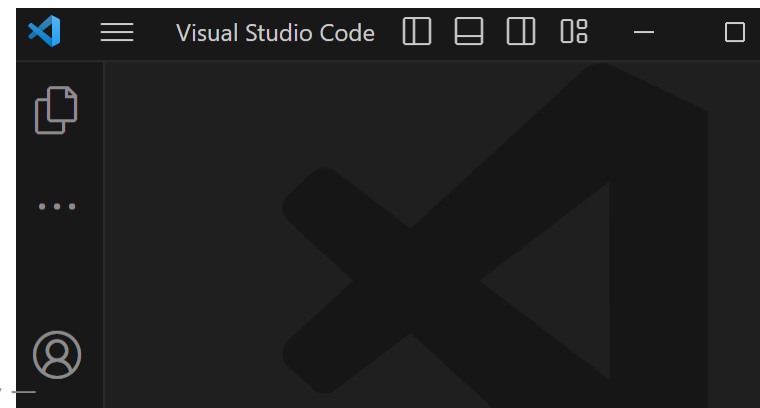
にアクセスできるか確認



③ (テキスト) エディタ: Visual Code Edit

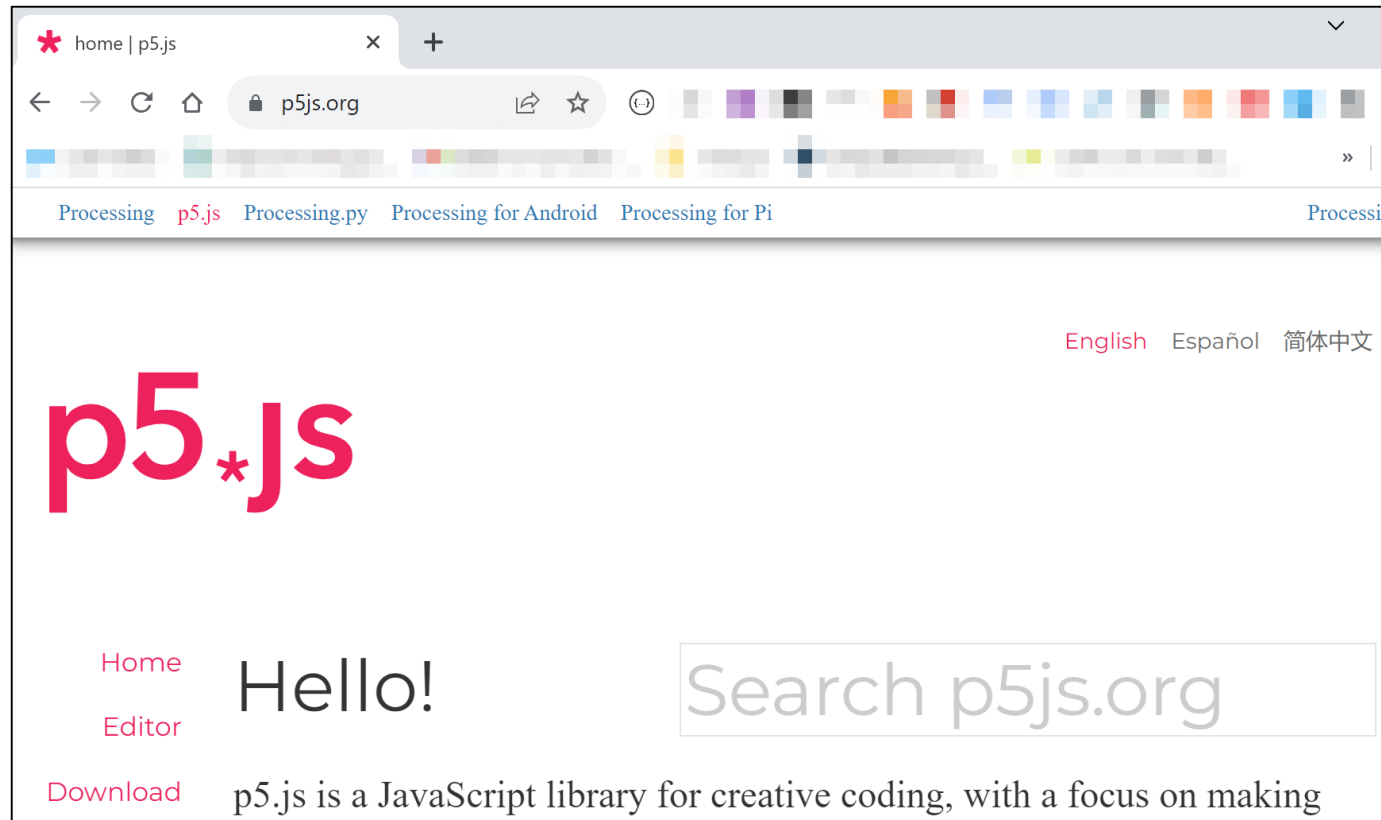


をクリックして開くことを確認



p5.jsの本家をブックマークしておこう

1. ブラウザのアドレス欄に「<https://p5js.org/>」と打ち込んでp5.jsのページを開き, 「ブックマーク」しておく



2. HTMLで目次を作ろう(1/2)

1. ファイルマネージャから「public_html」フォルダを作成
2. ブラウザからp5js.orgにアクセス
3. サイドバーの「Download」をクリック
4. 「p5.js Complete」でp5.zipをダウンロード
5. ファイルマネージャでp5.zipの中身を「public_html」にコピー
6. エディタで「index.html」を作成する→次のスライドへ

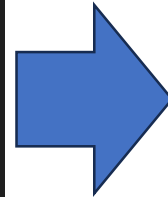
172.16.123.30 > guest000 > public_html	
名前	更新日時
 addons	2023/09/01 17:
 empty-example	2023/09/01 17:
 index	2023/09/01 17:
 p5	2023/07/10 17:
 p5.min	2023/07/10 17:
 README	2023/07/10 17:

2. HTMLで目次を作ろう(2/2)

エディタ

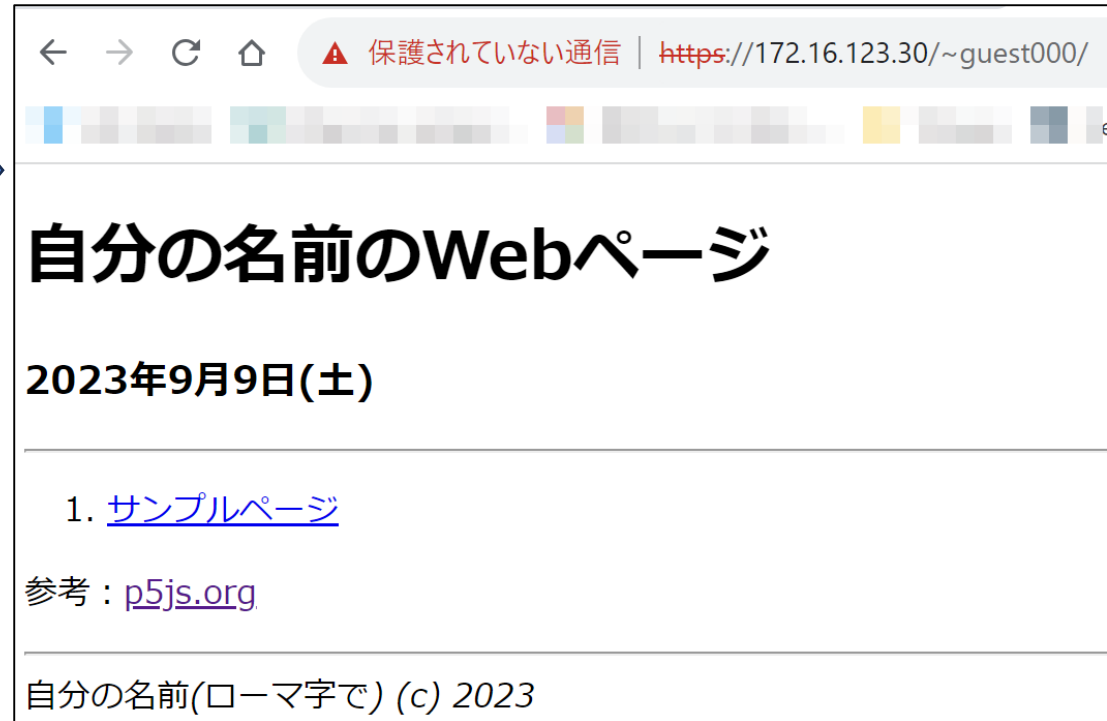
¥¥172.16.123.30¥guest0UU¥public_html¥index.html

```
1 <!DOCTYPE html>
2 <html lang="ja">
3   <head>
4     <meta charset="UTF-8" />
5     <title>自分の名前のWebページ</title>
6   </head>
7   <body>
8     <h1>自分の名前のWebページ</h1>
9     <h3>2023年9月9日(土)</h3>
10    <hr />
11    <ol>
12      <li><a href="empty-example/">サンプルページ</a></li>
13    </ol>
14    <p>参考 : <a href="https://p5js.org/">p5js.org</a></p>
15    <hr />
16    <address>自分の名前(ローマ字で) (c) 2023</address>
17  </body>
18 </html>
```



ブラウザ

https://172.16.123.30/~guest0UU/

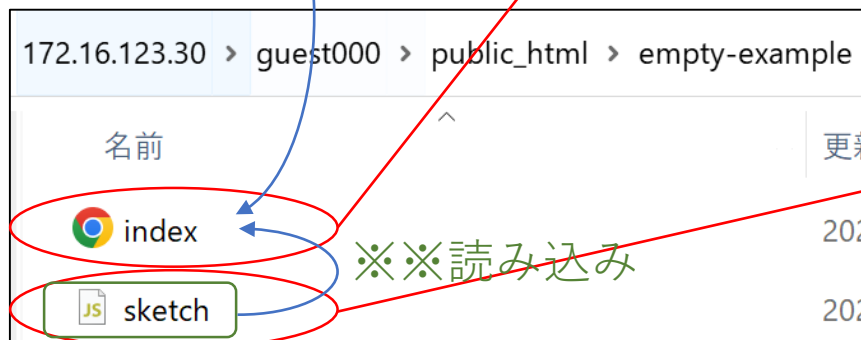


3. サンプルアプリをながめてみよう(1/4)

- ファイルマネージャで「empty-example」フォルダを開く



※読み込み



※※読み込み

```
1 <!DOCTYPE html>
2 <html lang="">
3
4 <head>
5   <meta charset="utf-8">
6   <meta name="viewport"
7     content="width=device-width, initial-
8     scale=1.0">
9   <title>p5.js example</title>
10  <style>
11    body {
12      padding: 0;
13      margin: 0;
14      background-color: white;
15    }
16  </style>
17  <script src="../p5.min.js"></script>
18  <!-- <script src="../addons/p5.sound.js"></script> -->
19  <script src="sketch.js"></script>
20 </head>
21 <body>
22   <main>
23   </main>
24 </body>
25 </html>
```

※読み込み

※※読み込み

```
1 function setup() {
2   // put setup code here
3 }
4
5 function draw() {
6   // put drawing code here
7 }
```

これが

「Processing」スクリプト

①setup関数(function)

②draw関数

の順で実行される。

3. サンプルアプリをながめてみよう(2/4)

- ブラウザで

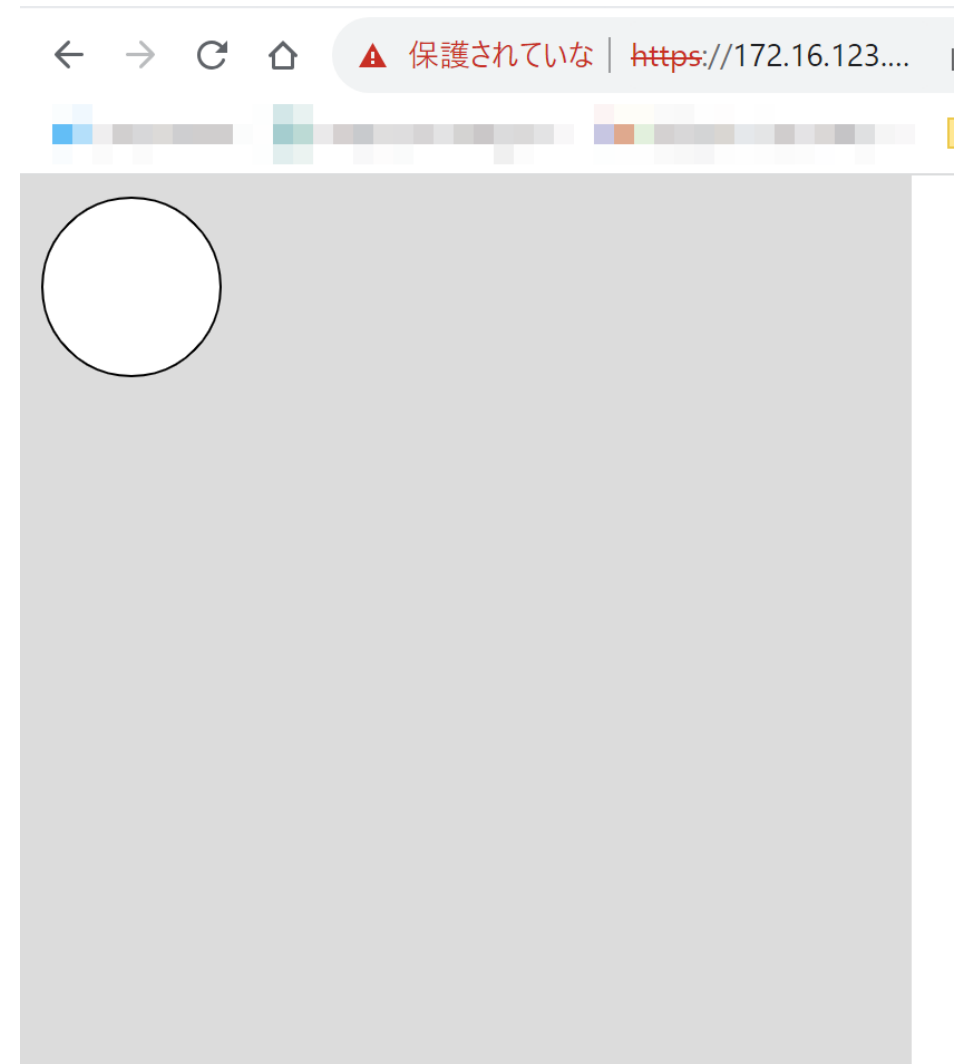
`https://172.16.123.30/~guest0UU/empty-example/`
をアクセスして動作確認すること。



3. サンプルアプリをながめてみよう(3/4)

次にsketch.jsをエディタで開き

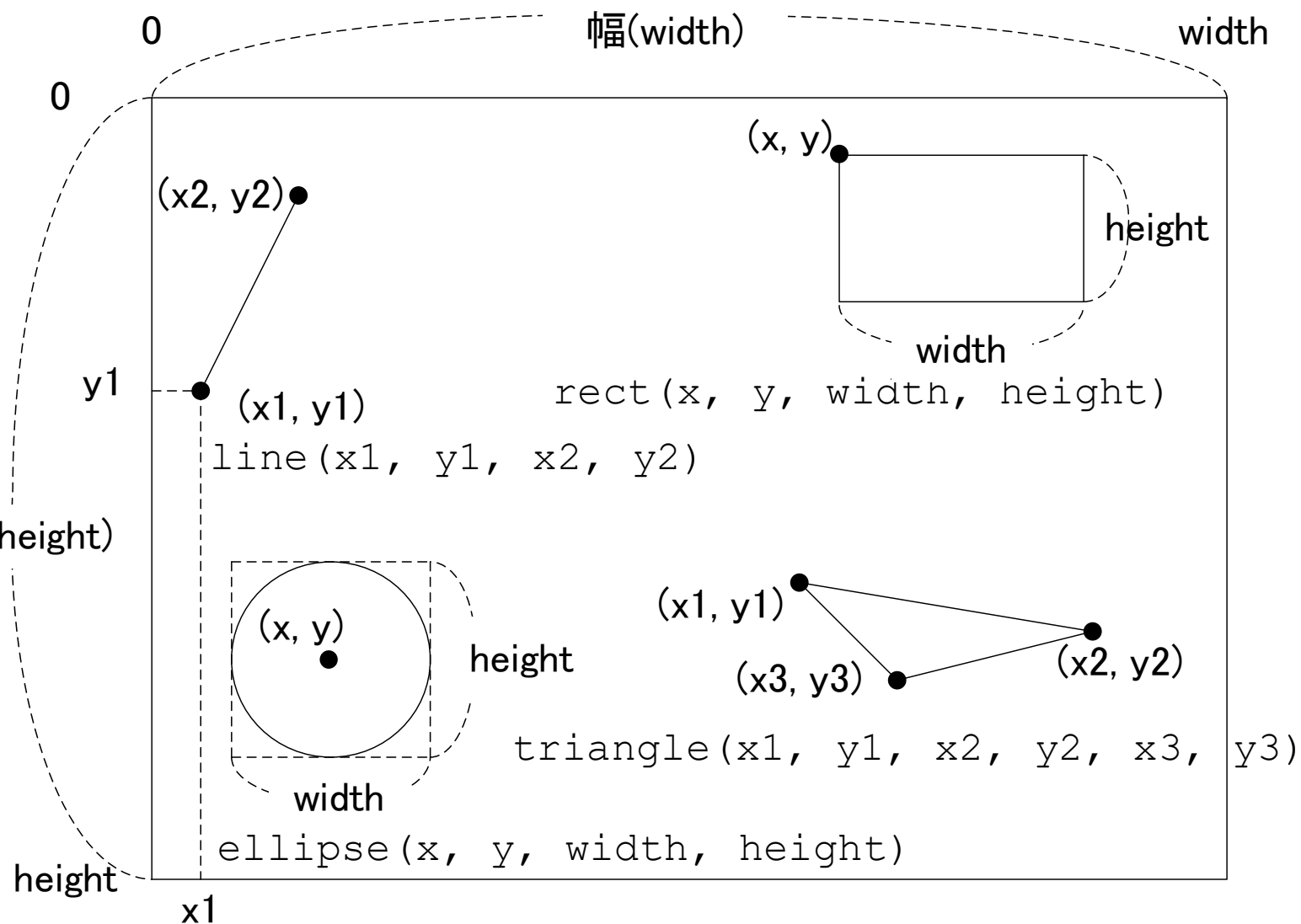
```
1  function setup() {  
2    // put setup code here  
3    createCanvas(400, 400);  
4  }  
5  
6  function draw() {  
7    // put drawing code here  
8    background(220);  
9    ellipse(50, 50, 80, 80);  
10 }
```



を追加して
もう一度ブラウザでアクセス
(リロードでもよい)
して動作確認してみよう。

図形の書き方

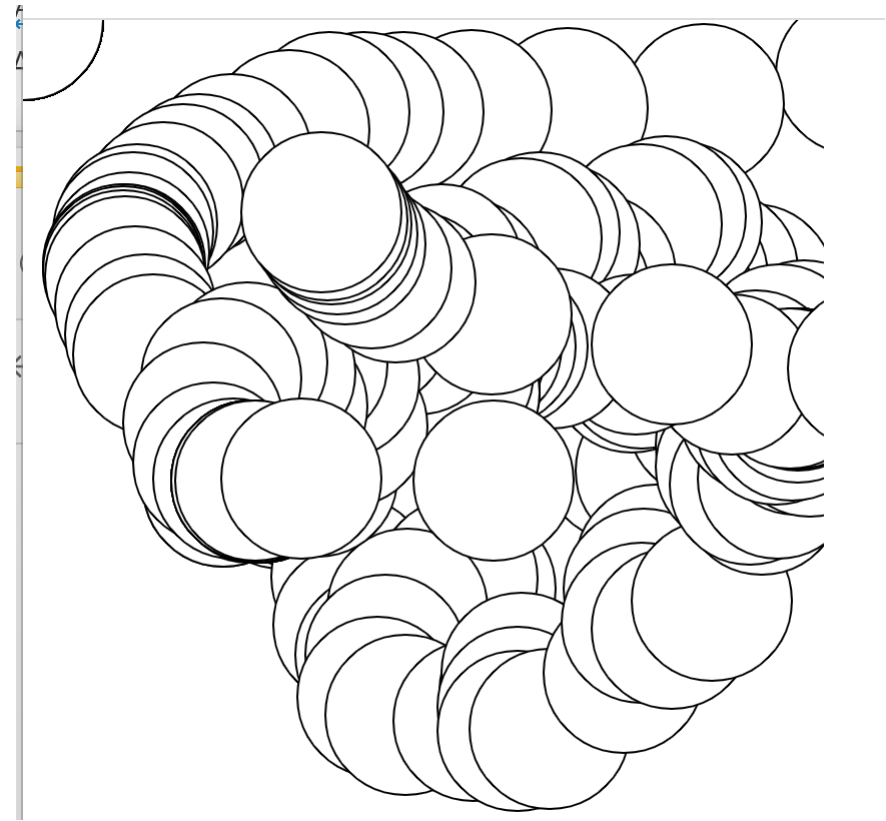
- 線: line関数
- 長方形: rect関数
- 楕円: ellipse関数
- 三角形: triangle関数
- 線の変更
 - 太さ: strokeWidth(太さ)
 - 両端の形: strokeCap
 - 角の形: strokeJoin
- 背景色変更: background
- 色変更: fill
 - fill(0) 黒 ~ fill(255) 白
 - fill(赤, 緑, 青) それぞれ 0 ~ 255



サンプルアプリをながめてみよう(4/4)

- マウスが動くと円が動くようにしよう。

```
1 function setup() {  
2   // put setup code here  
3   createCanvas(400, 400);  
4 }  
5  
6 function draw() {  
7   // put drawing code here  
8   //background(220);  
9   //ellipse(50, 50, 80, 80);  
10  if(mouseIsPressed) {  
11    fill(0);  
12  } else {  
13    fill(255);  
14  }  
15  ellipse(mouseX, mouseY, 80, 80);  
16 }
```



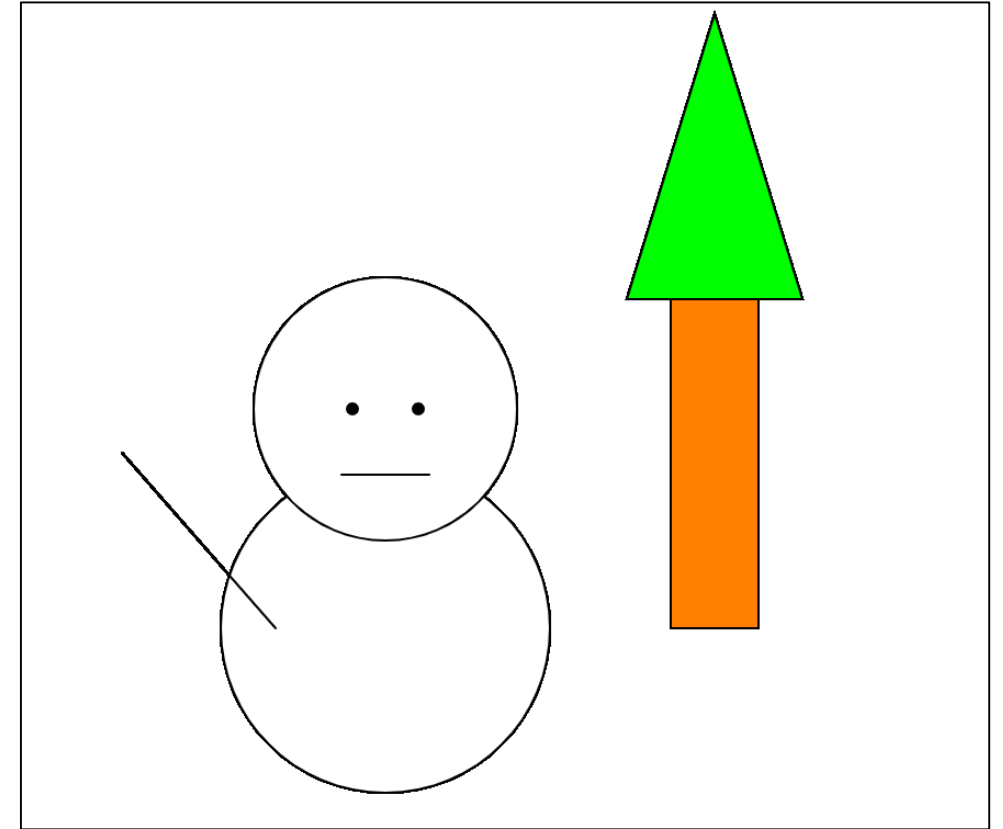
Processingアプリの作り方

1. 「public_html」の下に半角英数字名のフォルダ(例えば「draw」とする)を作り, 「empty-example」の中身(index.htmlとsketch.js)を全部コピーする。
2. コピー先のフォルダ「draw」にあるsketch.jsをエディタで開いてプログラムを作る。
3. ブラウザで
<https://172.16.123.30/~guest0UU/draw/>
にアクセスして動作確認する。

4. Processingで図形を描いてみよう

- 「draw」アプリを作ってみる。

```
1  function setup() {  
2      // put setup code here  
3      createCanvas(400, 400);  
4  }  
5  
6  function draw() {  
7      // 背景色を白に  
8      background(255);  
9      // 木  
10     fill(0, 255, 0); // 緑で塗りつぶし  
11     triangle(350, 20, 390, 150, 310, 150); // 枝  
12     fill(255, 128, 0); // 茶  
13     rect(330, 150, 40, 150); // 胴体  
14  
15     // 雪だるま  
16     fill(255); // 白で塗りつぶす  
17     ellipse(200, 300, 150, 150); // 胴体  
18     ellipse(200, 200, 120, 120); // 頭  
19     fill(0); // 黒  
20     ellipse(185, 200, 5, 5); // 左目  
21     ellipse(215, 200, 5, 5); // 右目  
22     line(180, 230, 220, 230); // 口  
23     line(80, 220, 150, 300); // 右手  
24 }
```



public_html/index.htmlに目次を追加する

1. [¥¥172.16.123.30¥guest0UU¥public_html¥index.html](http://172.16.123.30/guest000/public_html/index.html)をエディタで開く
2. 項目を追加

```
<ol>  
  <li><a href="empty-example/">サンプルページ</a>  
  <li><a href="draw/">絵を描く</a></li>  
</ol>
```

3. ブラウザをリロードして追加されたことを確認する

自分の名前のWebページ

2023年9月9日(土)

1. [サンプルページ](#)

2. [絵を描く](#)

参考: p5js.org

自分の名前(ローマ字で) (c) 2023

5. Processingでアニメーションを作ろう

- 「draw」アプリの内容をコピーして「anime」アプリを作ってみる。→手を振るだけの動作を追加。

```
1  function setup() {  
2    // put setup code here  
3    createCanvas(400, 400);  
4    // 一秒間に表示するフレーム数  
5    frameRate(2); // 2フレーム/秒  
6  }  
7  
8  // フレーム番号  
9  let no_frame = 0;  
10
```

```
11 function draw() {  
12   console.log('no_frame = ', no_frame);  
13   background(255); // 背景色を白に
```

```
28  
29   if(no_frame % 2 == 0) // フレーム番号が偶数の時  
30     line(80, 220, 150, 300); // 右手  
31   else  
32     line(50, 300, 150, 300); // 右手  
33  
34   //フレーム番号を一つ増やす  
35   no_frame = no_frame + 1;  
36 }
```

自由課題

自分で描いた絵を動かしてみよう！→「jiyuu」アプリとする

1. 「draw」アプリを参考に絵を描く
2. 「anime」アプリを参考に，絵の一部を動かしてみる
3. 「jiyuu」アプリをindex.htmlに追加して目次の項目を増やす

6. Processingでカメラを作ってみよう(1/2)

- 「camera」アプリを作り，自分の顔が映るかどうか確認する。

```
1  function setup() {  
2    // put setup code here  
3    createCanvas(400, 400);  
4    capture = createCapture(VIDEO);  
5    capture.hide();  
6  }  
7  
8  function draw() {  
9    // put drawing code here  
10   image(capture, 0, 0, width, width  
11   * capture.height / capture.width);  
12   //console.log('video running...');  
13 }
```



6. Processingでカメラを作ってみよう(2/2)

```
1  let main_canvas;  
2  
3  function setup() {  
4    // put setup code here  
5    main_canvas = createCanvas(400,  
6    400);  
7    capture = createCapture(VIDEO);  
8    capture.hide();  
9  
10   button = createButton('シャッター');  
11   button.position(400, 400, 50);  
12   button.mousePressed(GazouSave);  
13 }  
14  
15 function draw() {  
16   // put drawing code here  
17   image(capture, 0, 0, width, width *  
18   capture.height / capture.width);  
19   //console.log('video running...');  
20 }  
21  
22 function GazouSave(){  
23   saveCanvas(main_canvas, 'watashi',  
24   'png');  
25 }  
26 }
```

- シャッター機能を付けて，静止画がダウンロードされることを確認する。



7. Processingでアルバムを作ってみよう(1/3)

- 「album」アプリを作成する。「watashi.png」もコピーしておくこと。

```
1  let img;  
2  
3  function preload() {  
4    |  img = loadImage('watashi.png');  
5  }  
6  
7  function setup() {  
8    |  // put setup code here  
9    |  createCanvas(400, 400);  
10   |  image(img, 0, 0);  
11  }
```



- 見切れてしまうので、次に修正する。

7. Processingでアルバムを作ってみよう(2/3)

```
1  let img;  
2  
3  function preload() {  
4    |  img = loadImage('watashi.png');  
5  }  
6  
7  function setup() {  
8    |  // put setup code here  
9    |  //createCanvas(400, 400);  
10   |  createCanvas(img.width, img.height);  
11   |  image(img, 0, 0);  
12  }
```



7. Processingでアルバムを作ってみよう(3/3)

```
1 let maisuu = 10; // 最大写真数
2 let canvas_width = 1000, canvas_height = 1000; // カンバスの大きさ
3 let yoko_maisuu = 3; // 一行分の最大写真枚数
4 //let img;
5 let img = [];
6
7 function preload() {
8   let i;
9   img[0] = loadImage('watashi.png');
10  console.log('watashi.png', img[0]);
11  for(i = 1; i < maisuu; i++)
12  {
13    filename = 'watashi (' + i + ').png';
14    img[i] = loadImage(filename);
15    console.log(filename, img[i]);
16  }
17 }
```

```
18 function setup() {
19   // put setup code here
20   //createCanvas(400, 400);
21   //createCanvas(img.width, img.height);
22   createCanvas(canvas_width, canvas_height);
23   x = 0; y = 0;
24   x_step = canvas_width / yoko_maisuu;
25   y_step = img[0].height * (x_step / img[0].width);
26   tate_maisuu = maisuu / yoko_maisuu;
27   index = 0;
28   for(i = 0; i < tate_maisuu; i++)
29   {
30     y = i * y_step;
31     for(j = 0; j < yoko_maisuu; j++)
32     {
33       x = j * x_step;
34       image(img[index], x, y, x_step, y_step);
35       index = index + 1;
36       if(index >= maisuu)
37         break;
38     }
39   }
40 }
```

アルバム完成！

時間があれば次のことに
チャレンジ

- 10枚以上表示できる？
- 横の隙間を作る
- バックの色を変える
等々 . . .



8. 今日のまとめ

- 今日作ったProcessingアプリケーション
 1. サンプルページ
 2. 絵を描く
 3. アニメ
 4. カメラ
 5. アルバム

自分の名前のWebページ

2023年9月9日(土)

1. [サンプルページ](#)
2. [絵を描く](#)
3. [アニメを描く](#)
4. [カメラ](#)
5. [アルバム](#)

参考 : p5js.org

自分の名前(ローマ字で) (c) 2023

- 余裕があれば他の人のWebページへのリンクを追加してみよう