**芸工情報処理基礎　第６回　2024年5月24日　　西尾**

**【音声ファイル１】**

**注意　この授業は第1回目から指示通りに学習・実行をしないと動きません。**

**途中からこの授業をとる人は、第1回目の内容から順番に進めてください。**

**途中で飛ばしたりすると、全く動作しません。**

**（少しは動くが、表示がまともに出なくなる）**

**必ず、第1回から飛ばさず順にやってください。**

**この注意に従わず、やっていない場合、サポートできません。**

**また、プログラミングのコードは文章ではありません。**

**動作を確かめながら、少しずつ積み上げるものです。**

**一気に打ち込んで、動かないのは当然です。気を付けましょう。**

**１．前回の内容**

教材をインターネット上のニシオマトリックスから「プログラミングサンプル３からの続き」をダウンロードして、デスクトップに作成したフォルダa2020prog内に入れた。ZIPファイルを展開（解凍、Unzip）して動作を確認した。

VSCodeを開き、教材のフォルダをドラッグし、編集（エディット）を行い、JavaScriptのコードを変更し、プログラムを変更した結果をGoogleChromeの画面で確認した。

**２．今回の内容**

概要

　前回まで描画を四角形（矩形）で行ってきたが、折れ線を使い多角形で描画する。

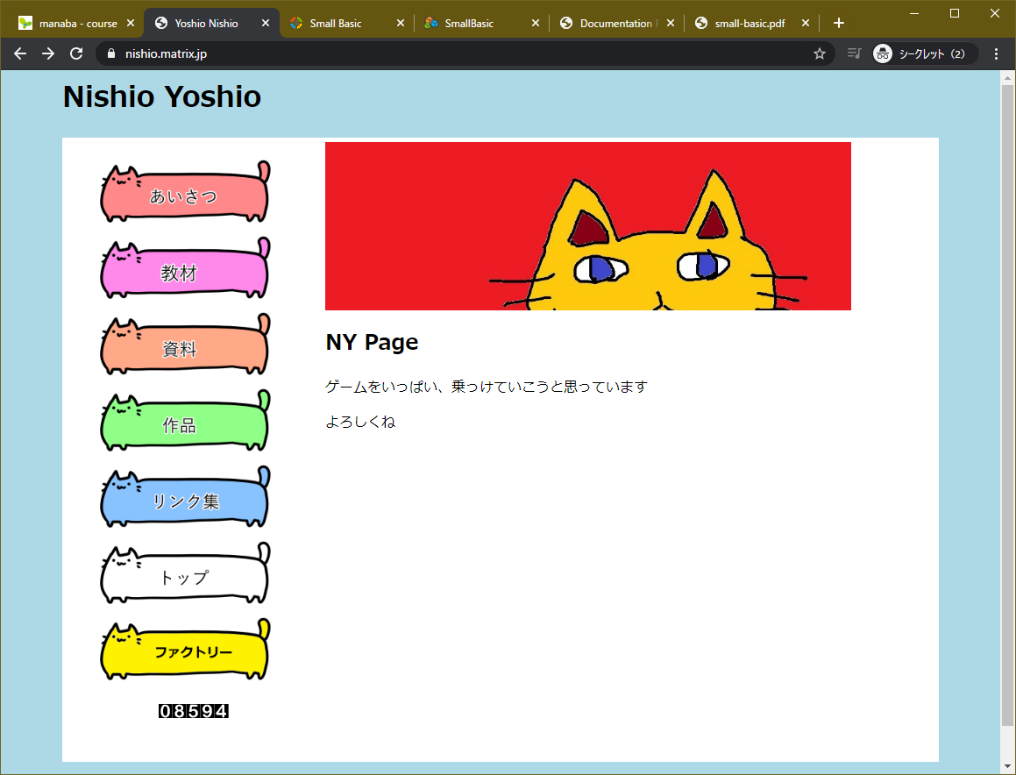
教材「プログラミングAサンプル５」をダウンロードしてVSCodeを使用し、プログラムのJavaScriptのコードを変更する。

**【音声ファイル２】**

**３．「プログラミングAサンプル５」を表示する**

　まず、ニシオマトリックスに入る

https://nishio.matrix.jp/



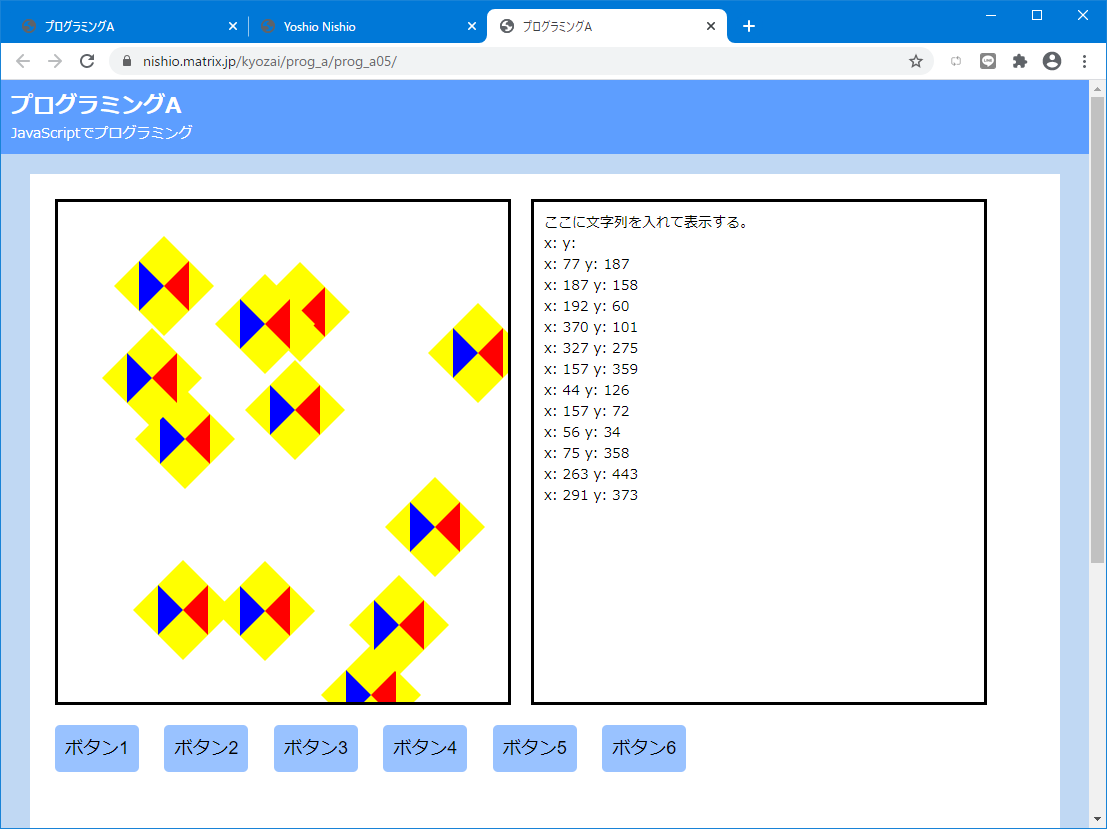
「教材」を押す



「プログラミングA」を押す



**「プログラミングAサンプル５」を表示** を押す　マウスで何回かクリックしてください

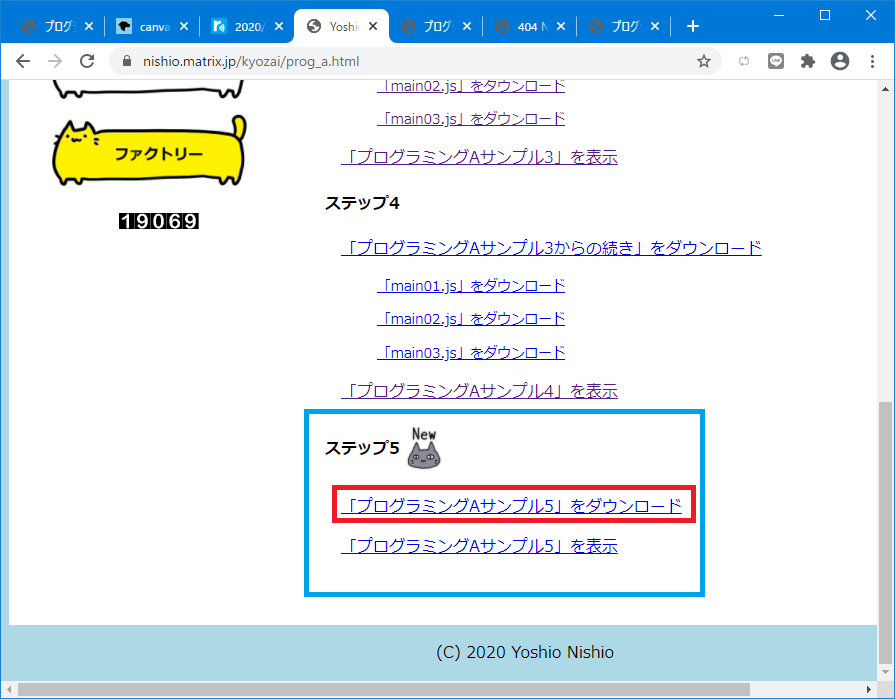


最後に✖で閉じておきましょう。（閉じていないと勘違いの元になります）

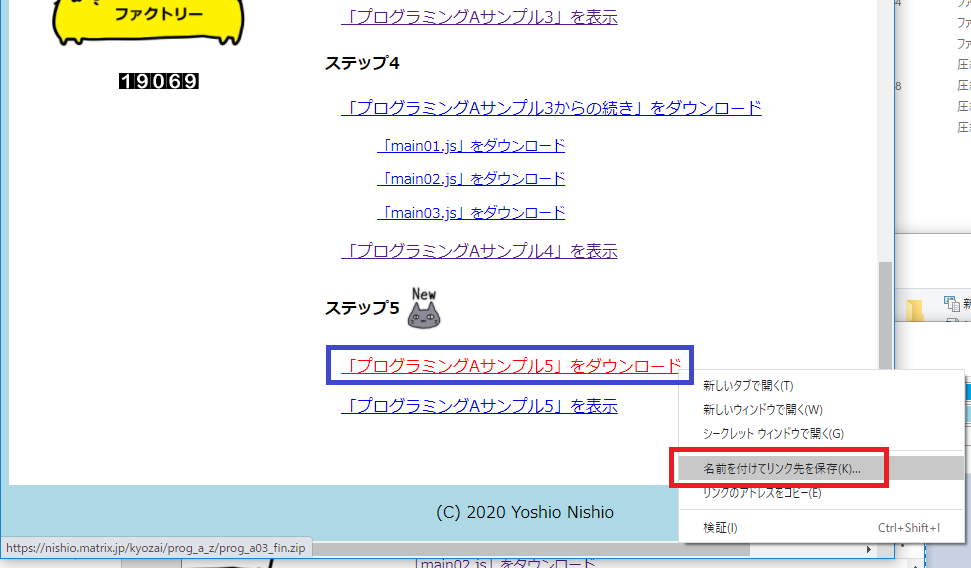
**【音声ファイル３】**

**４．「プログラミングAサンプル５」のダウンロード**

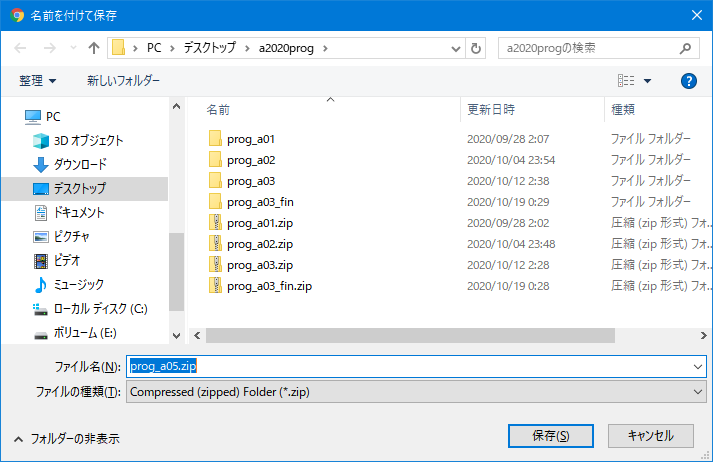
ニシオマトリックスの「教材」の「プログラミングA」に行きます



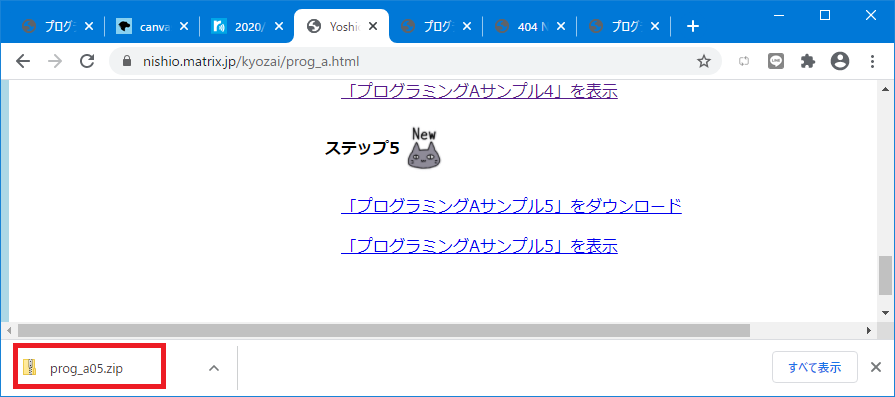
**「プログラミングAサンプル５」をダウンロード**を右ボタンクリック／副ボタンクリックすると、プルダウンメニューが出てくるので**「名前を付けてリンクを保存」**をクリックし選択する



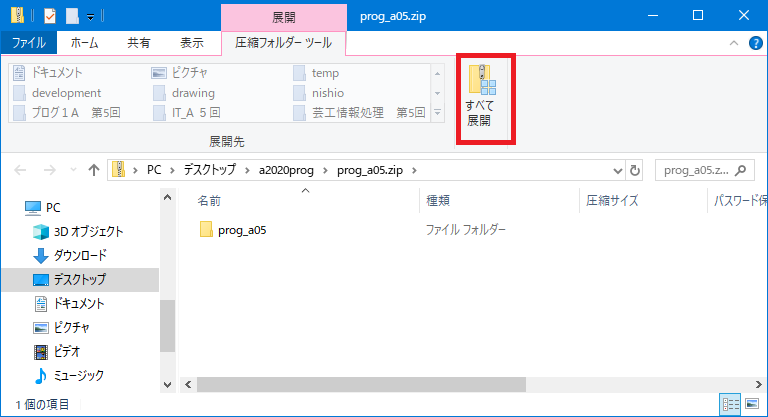
保存先の候補が表示される

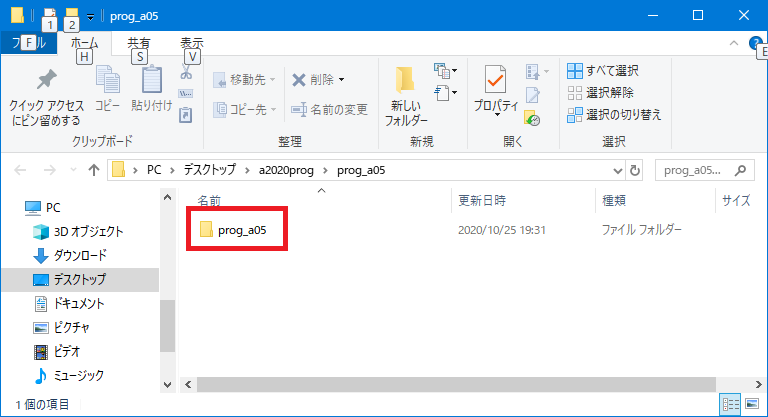
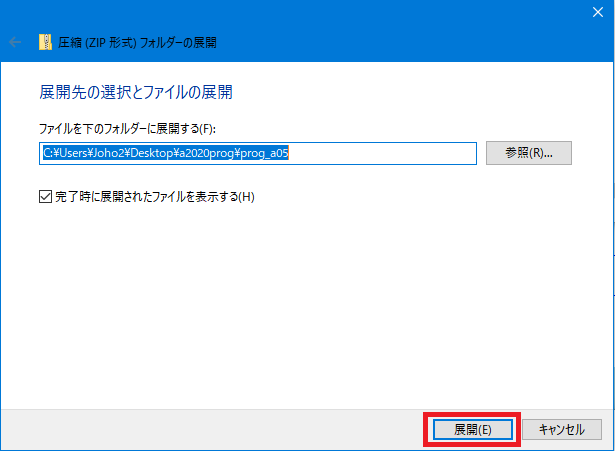


下図のようにウインドウの左下に出てくるのでクリックする

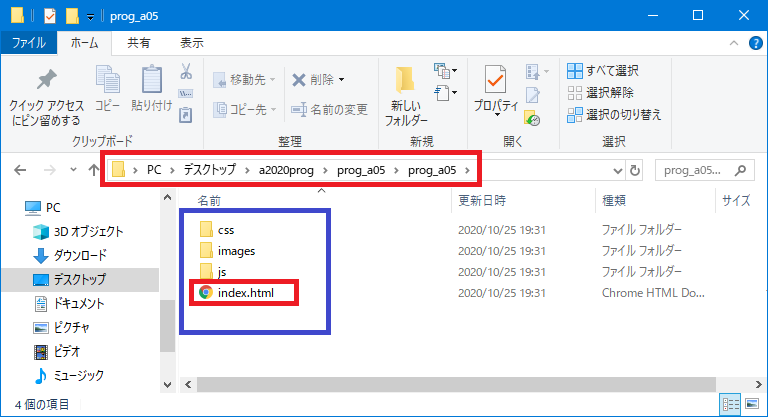


前回と同様にして展開（解凍）します

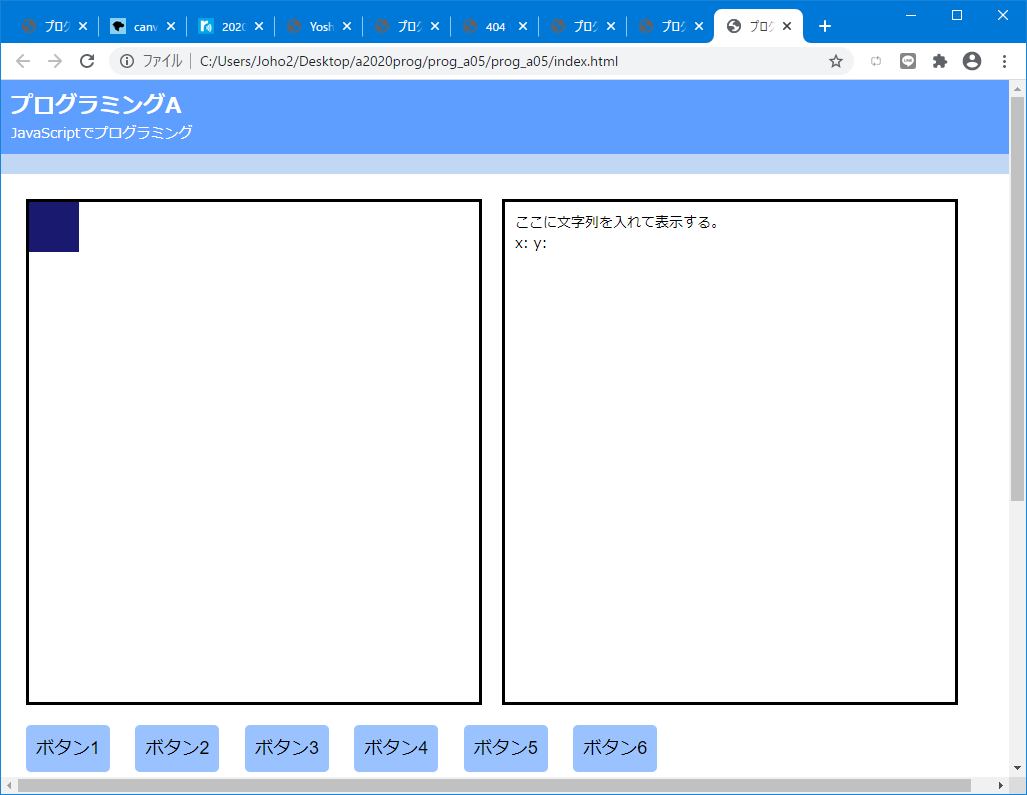




ちゃんと解凍したものがあるかチェックします



index.htmlをクリックして、ちゃんと入ったかを確認します



今回はここから初めて、

[「プログラミングAサンプル5」を表示](https://nishio.matrix.jp/kyozai/prog_a/prog_a05)

で見たプログラムに発展させていきます。

**【音声ファイル４】**

**５．VSCodeでjsファイルを編集する準備**

　JavaScriptファイルのことを略してjsファイルと言います。

VSCodeを立ち上げて、教材からダウンロードしたjsファイルを編集する準備をしましょう。

（１）VSCodeを立ち上げる

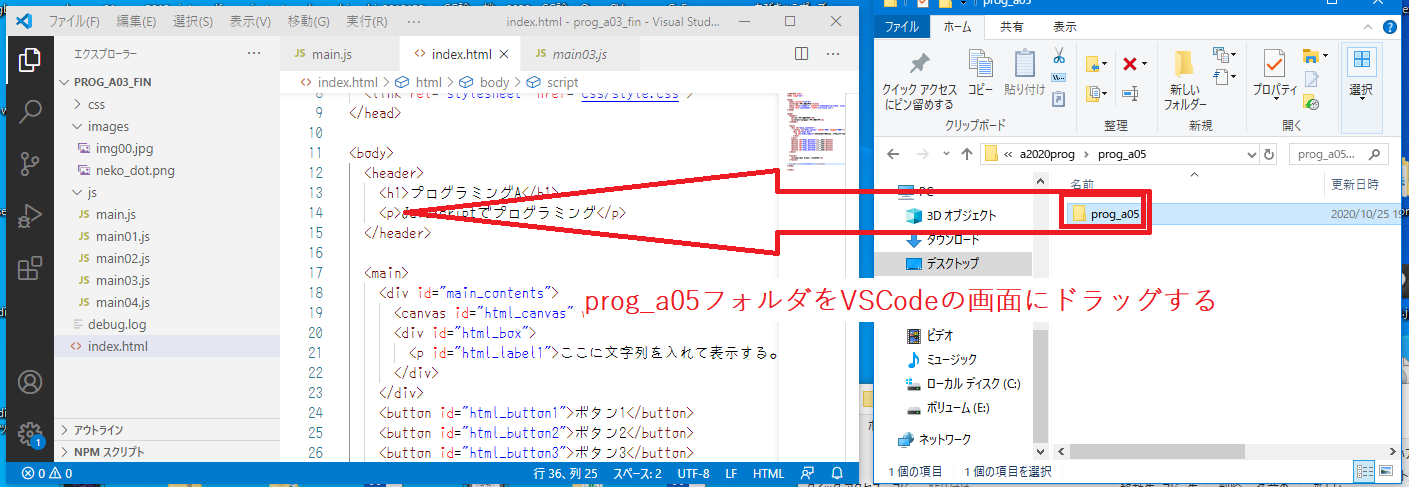
　VSCodeを立ち上げる



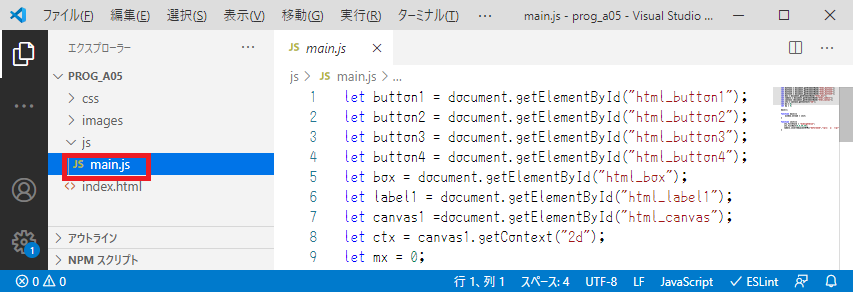
VSCodeを立ち上げます



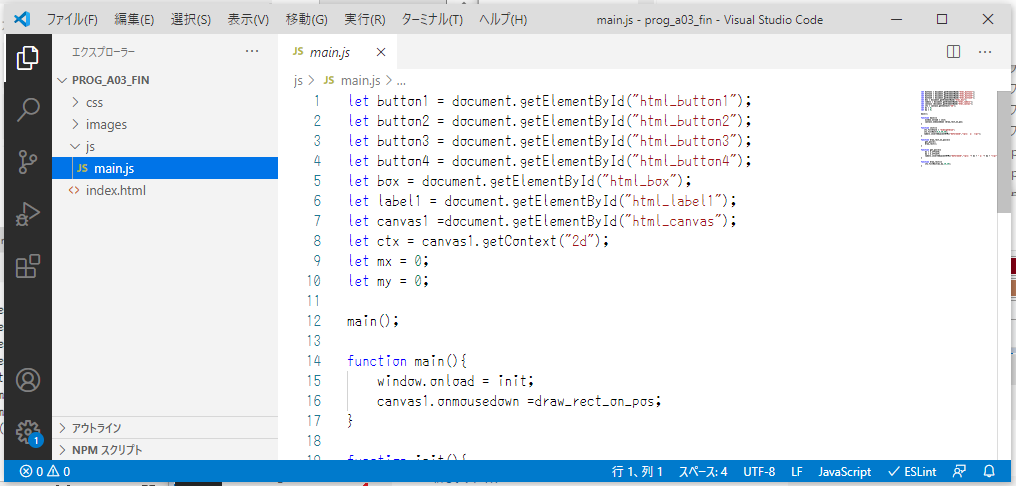
「prog\_a05」フォルダ（Windowsでは２重になっているので、下の方のフォルダ）をVSCodeのウインドウに向かってドラッグする。



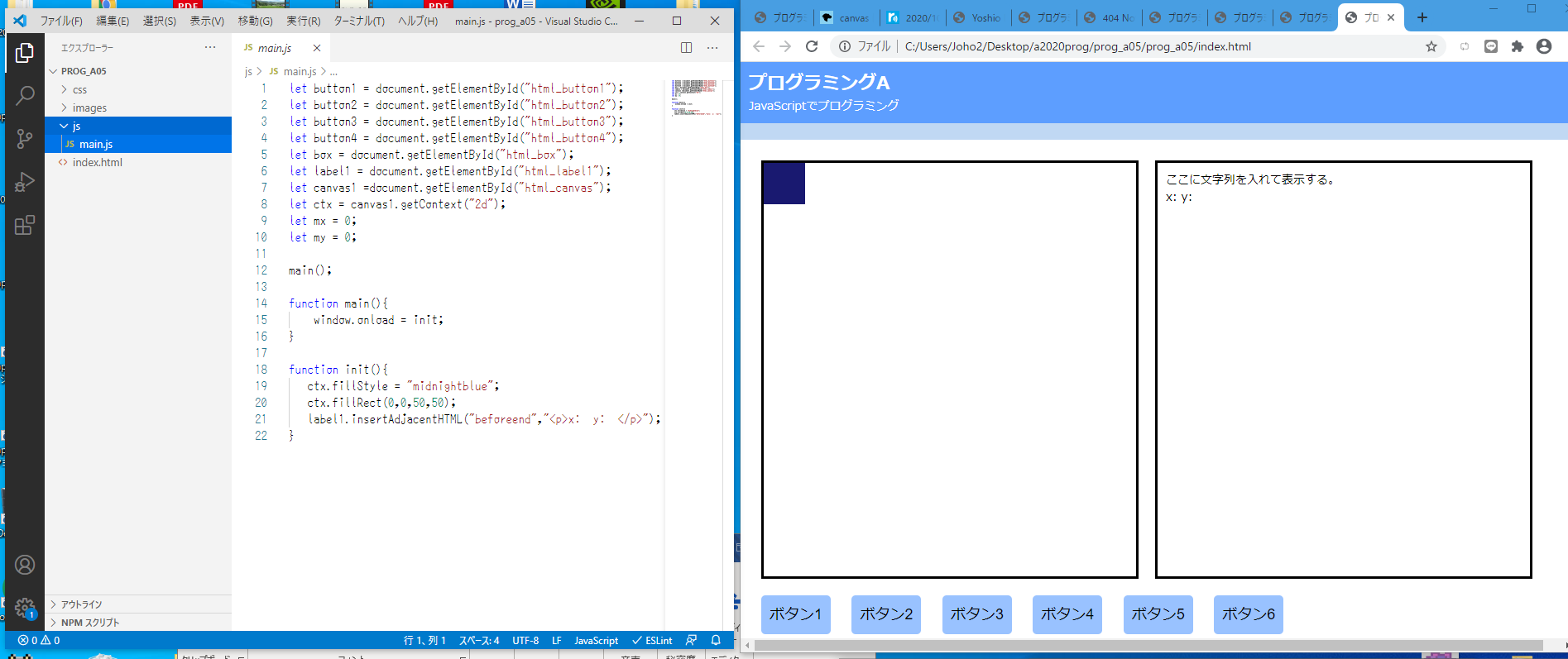
「エクスプローラー」を表示させる。「main.js」をクリックする



main.jsファイルを編集する準備が出来ました。



**画面の配置とプログラミングの流れ　スムーズに作業が進むように配置を整えましょう**



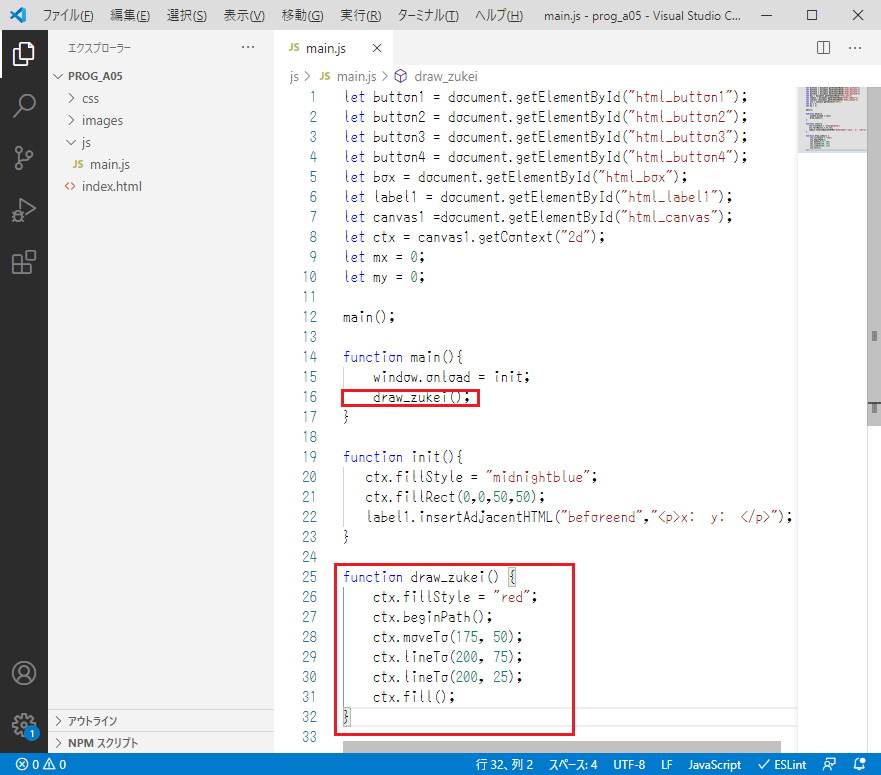
**prog\_a05をVSCodeの画面で表示　　　　　　prog\_a05の中のindex.htmlをダブルクリックして表示**

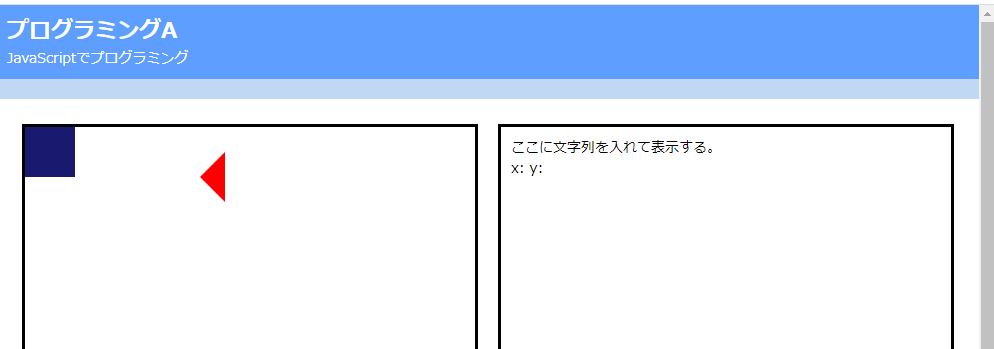
**【音声ファイル５】**

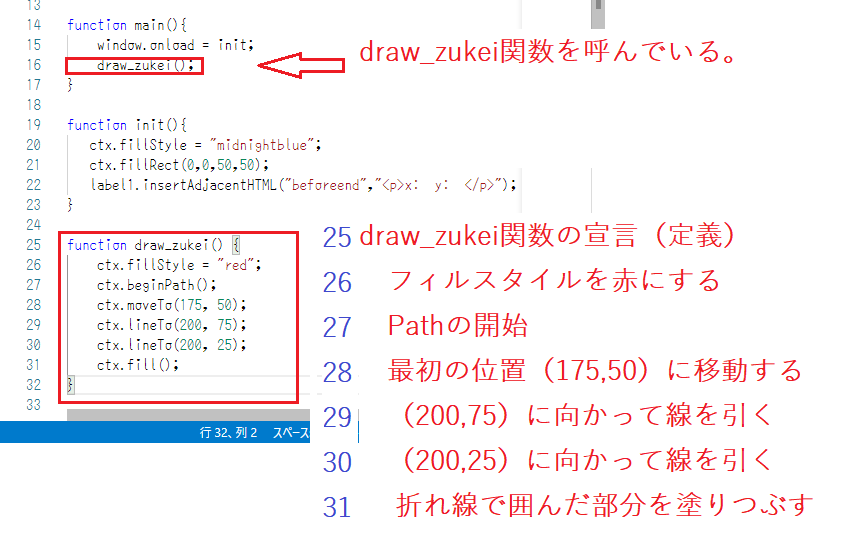
**６．main.jsファイルのプログラムを変更する**

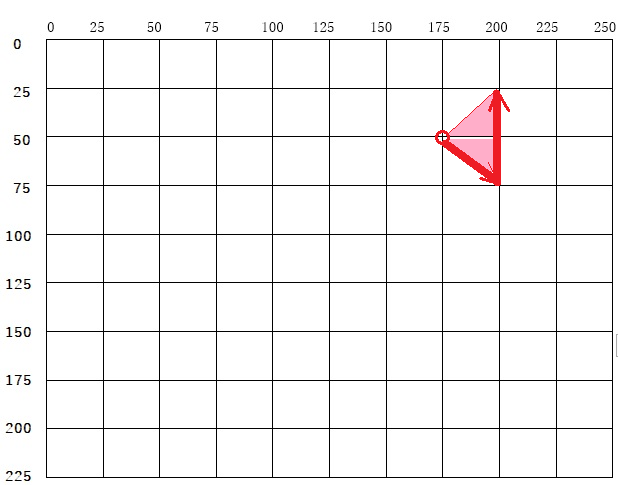
draw\_zukei関数を作り呼び出します。

変更後（赤い四角で囲んだ部分を入力する）







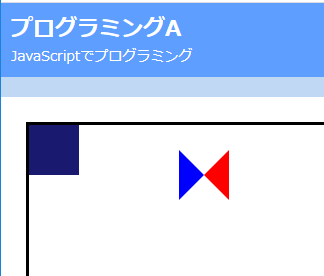
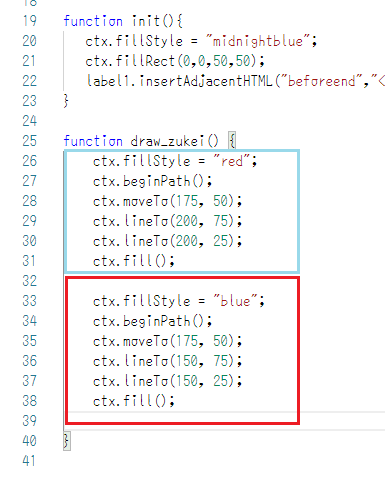


**【音声ファイル６】**

**７．main.jsファイルのプログラムを変更する**

draw\_zukei関数を改良します。

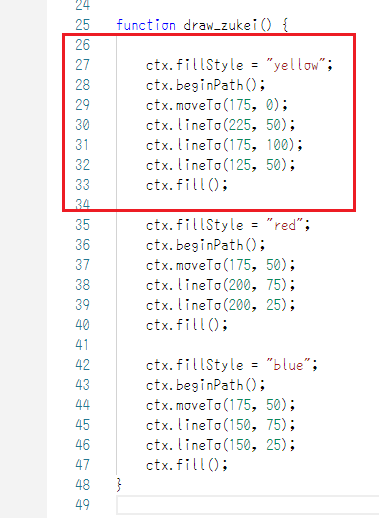
変更後（赤い四角で囲んだ部分を入力する）



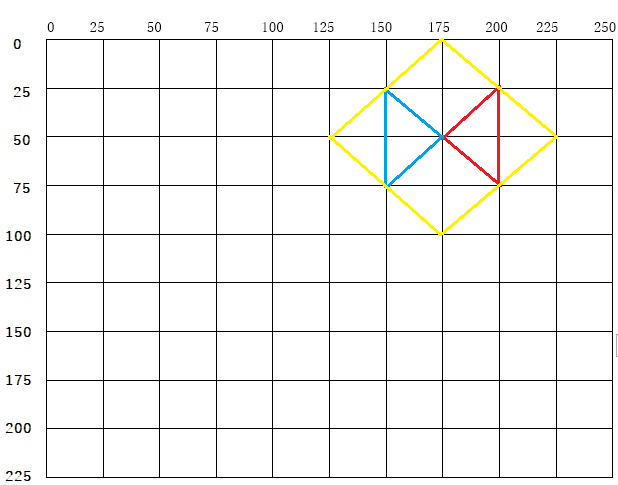
三角形ばかり描きましたが、多角形も描けます。

draw\_zukei関数を改良します。

変更後（赤い四角で囲んだ部分を入力する）







図形の描き方は理解できましたか。

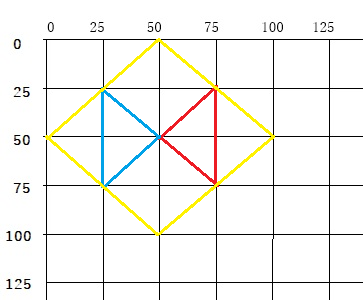
自分で図形を描いてみましょう。

**【音声ファイル７】**

**８．main.jsファイルのプログラムを変更する**

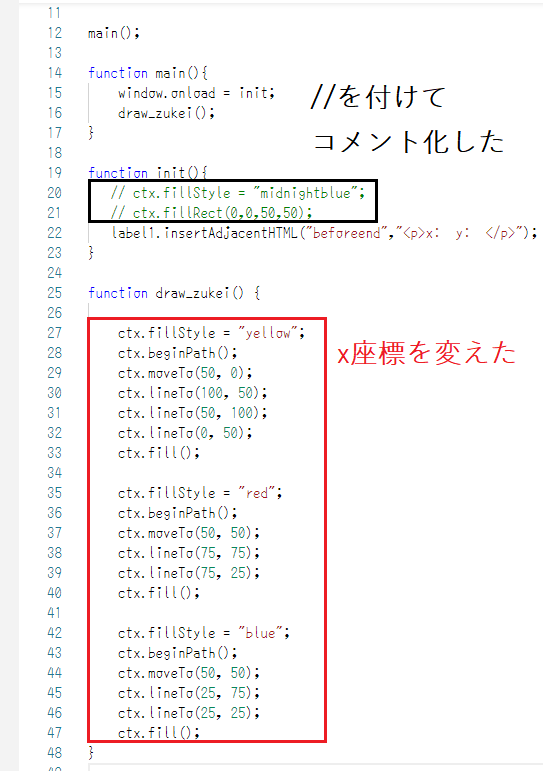
draw\_zukei関数を改良します

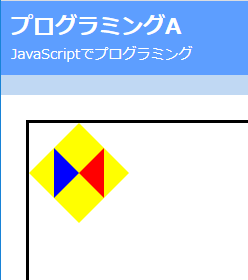
変更後（赤い四角で囲んだ部分を入力する）



使いやすいように原点（０，０）の近くに移動して考えます

この場合プログラムは





原点（０，０）のところを点P（px, py）として、関数も引数（ひきすう）をともない

関数もdraw\_zukei( px, py)と変更する。

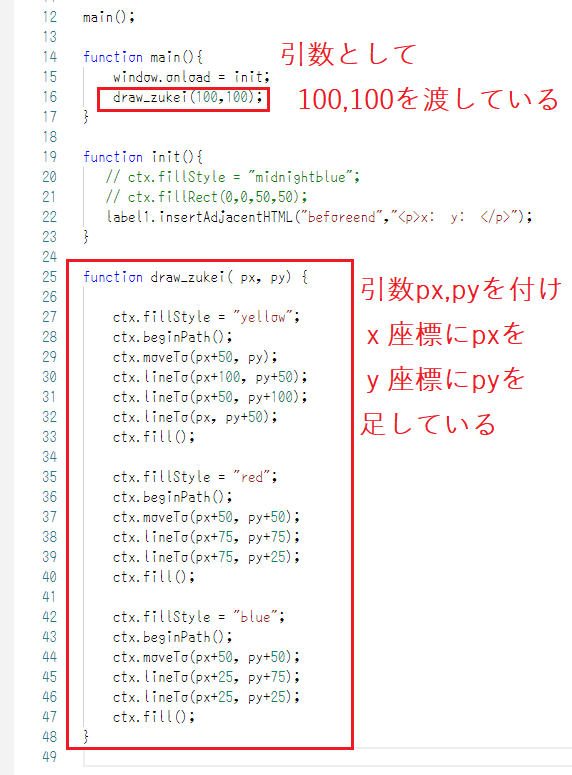
呼ぶ側も

draw\_zukei( 100,100)

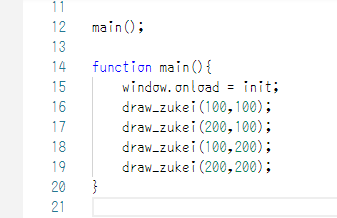
として、引数（ひきすう）ｐｘとｐｙにそれぞれ１００と１００を値として渡しています

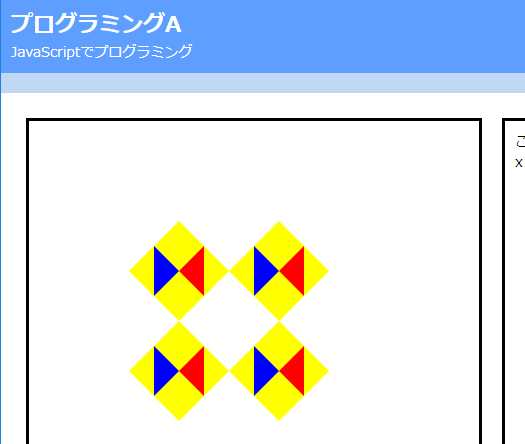


図形も（１００，１００）の位置にずれている



呼ぶ側を増やしてみた

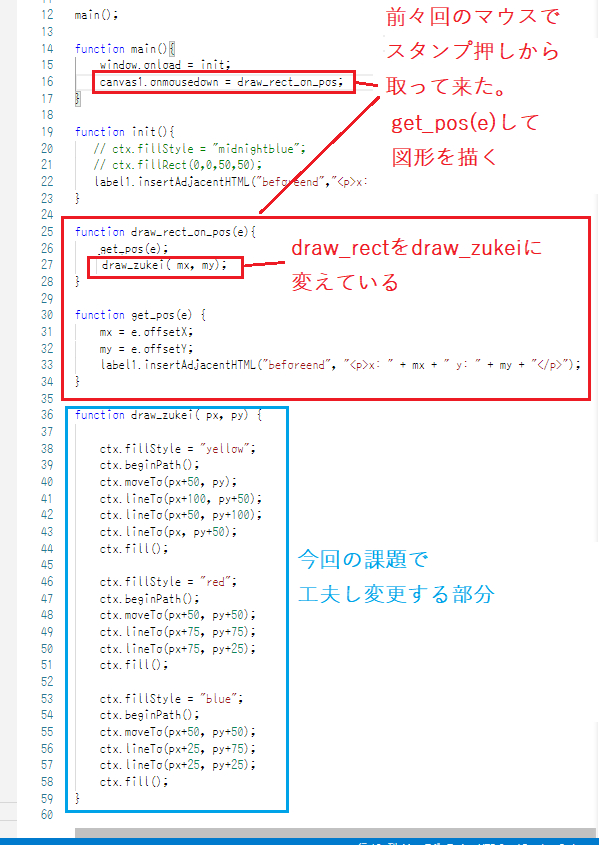




**【音声ファイル８】**

**９．マウスによるスタンプ**

マウスでスタンプ出来るように、前々回のプログラムの一部を取ってきて使います。



main.jsファイルのコードを提供します、コピー&ペーストして使いましょう

let button1 = document.getElementById("html\_button1");

let button2 = document.getElementById("html\_button2");

let button3 = document.getElementById("html\_button3");

let button4 = document.getElementById("html\_button4");

let box = document.getElementById("html\_box");

let label1 = document.getElementById("html\_label1");

let canvas1 =document.getElementById("html\_canvas");

let ctx = canvas1.getContext("2d");

let mx = 0;

let my = 0;

main();

function main(){

    window.onload = init;

    canvas1.onmousedown = draw\_rect\_on\_pos;

}

function init(){

   // ctx.fillStyle = "midnightblue";

   // ctx.fillRect(0,0,50,50);

   label1.insertAdjacentHTML("beforeend","<p>x:  y:  </p>");

}

function draw\_rect\_on\_pos(e){

    get\_pos(e);

     draw\_zukei( mx, my);

}

function get\_pos(e) {

    mx = e.offsetX;

    my = e.offsetY;

    label1.insertAdjacentHTML("beforeend", "<p>x: " + mx + " y: " + my + "</p>");

}

function draw\_zukei( px, py) {

    ctx.fillStyle = "yellow";

    ctx.beginPath();

    ctx.moveTo(px+50, py);

    ctx.lineTo(px+100, py+50);

    ctx.lineTo(px+50, py+100);

    ctx.lineTo(px, py+50);

    ctx.fill();

    ctx.fillStyle = "red";

    ctx.beginPath();

    ctx.moveTo(px+50, py+50);

    ctx.lineTo(px+75, py+75);

    ctx.lineTo(px+75, py+25);

    ctx.fill();

    ctx.fillStyle = "blue";

    ctx.beginPath();

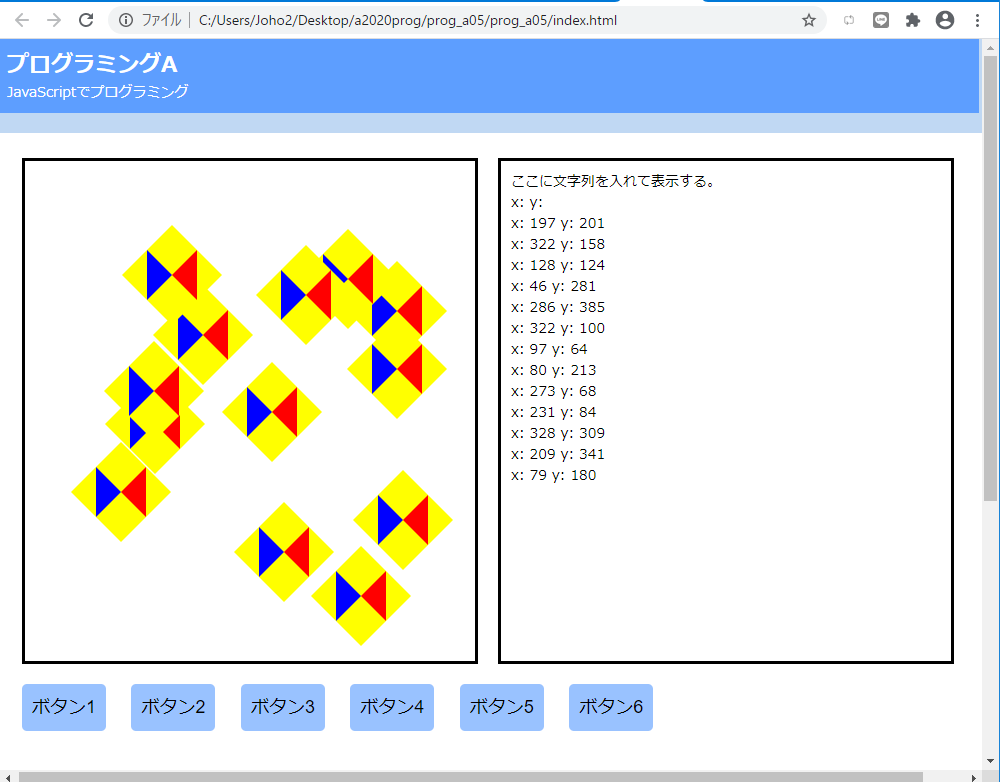
    ctx.moveTo(px+50, py+50);

    ctx.lineTo(px+25, py+75);

    ctx.lineTo(px+25, py+25);

    ctx.fill();

}

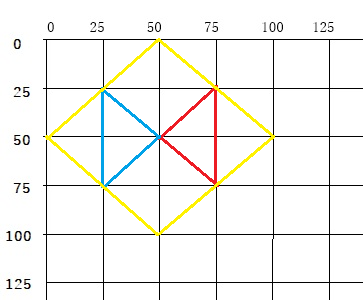


**【音声ファイル９】**

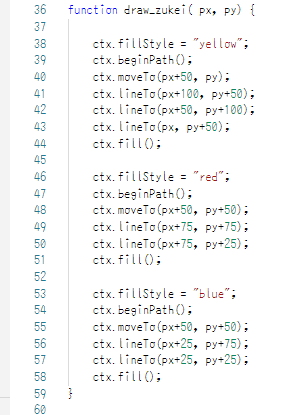
**１０．課題**

draw\_zukei関数で出来る図形を自分で作りましょう。

参考になる部分を再び挙げます



この図形を作るコードは



です。

このコードを自分の考えた図形にするように改良し、前節で提供したmain.jsに組み込み実行し、

マウスでいくつも図形を表示してできた画像をレポートとして提出してください。

感想・コメントを３行以上付けてください

この課題レポートの提出期限は、１週間後の金曜日の２３時５５分です。

学情システムで受け付けています。

提出例

**芸工情報処理基礎　2024年5月24日**



**【音声ファイル10】**