プログラミング概論

第14回　2023年12月20日

JavaScriptによるアプリ開発

今回の授業内容

- JavaScriptによるアプリの開発
  - ピアノアプリの開発（＋作業時間の計測）
- AppInventorで改めてピアノをつくって作業時間を計測
- JavaScriptとAppInventorで太鼓をつくって入力時間を比較

今回計測したタイムは次のレポートを書く際に必要になります．必ず記録し無くさないように注意すること．

作業時間と入力時間の違い

- 作業時間とは
  - 入力開始から完成までにかかった時間
  - 入力し終わるまででではなく，修正時間も含む
  - AppInventorの場合はデザイナ画面での作業も含む
- 入力時間とは
  - 特定の箇所を入力するのにかかった時間
  - AppInventorの場合はブロックの組み立てにかかった時間（デザイナ画面の作業は含まない）
  - 修正時間は含まない

JAVASCRIPTによるアプリの開発

開発から実行までの流れ

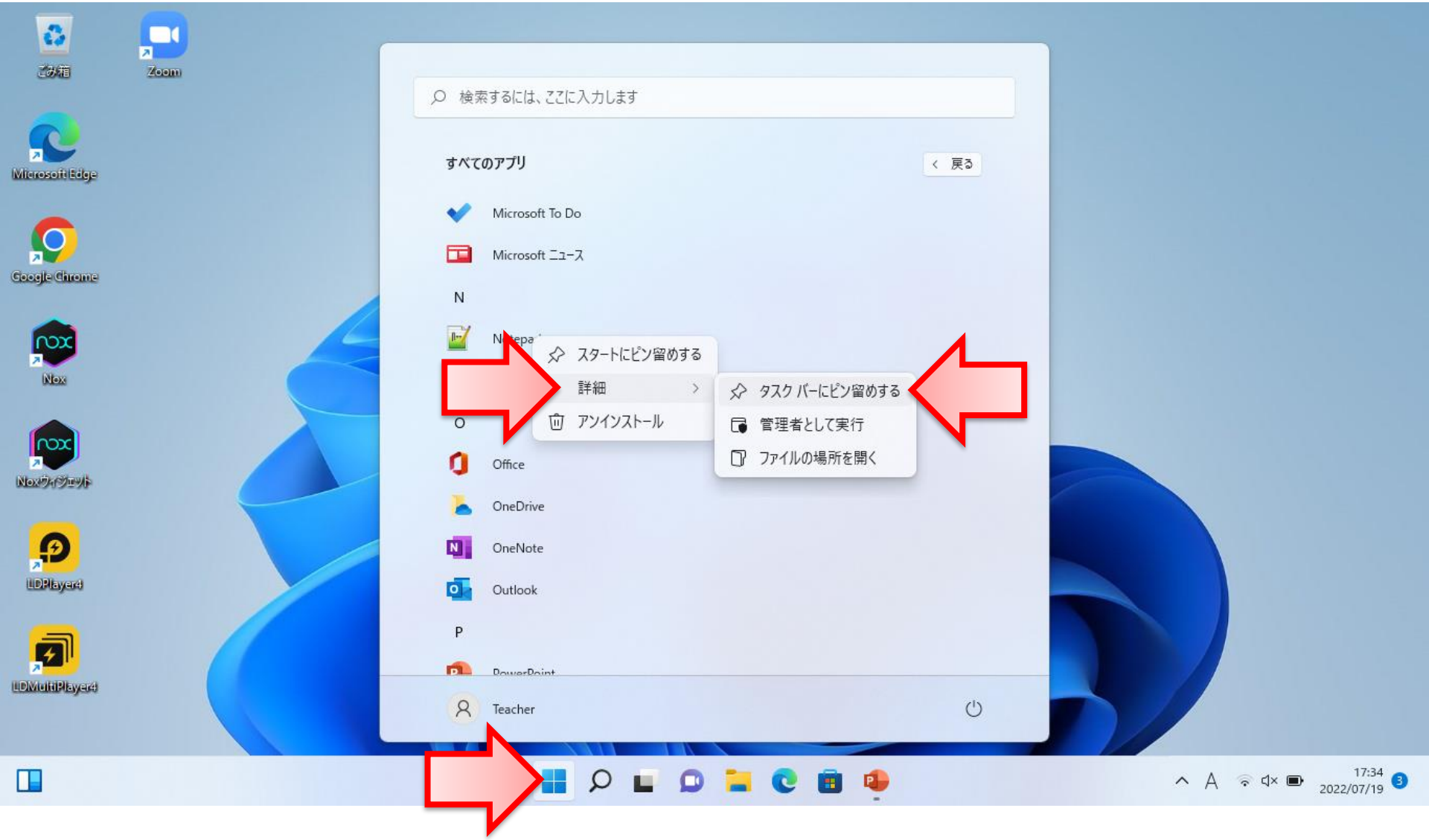
1. Notepad++を起動する
2. プログラムを慎重に入力する
   ＆入力間違いがないか何度も見直す
3. 保存する
4. アップロードする
5. 自分のスマートフォンで実行する

今回はウェブアプリなのでiPhoneでも動きます！

今回の作業の進め方の注意

- まずはピアノアプリをJavaScriptで作成する
- ピアノアプリは作業時間を計測するため作業を中断するときにはストップウォッチを一時停止する
- ピアノアプリが正しく動くまでは太鼓アプリの作成に取りかからない

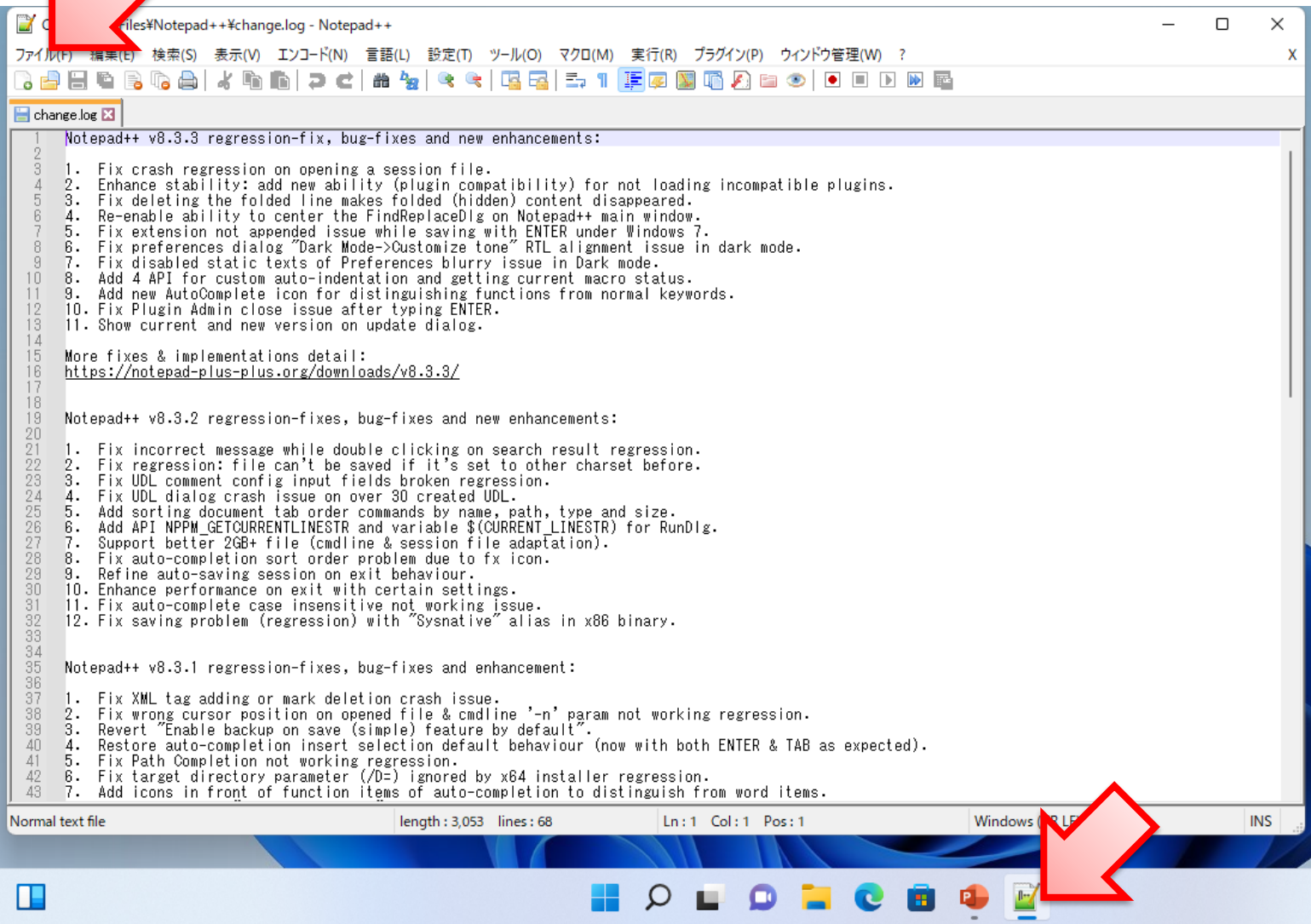
Notepad++をタスクバーにピン留めする



スタートメニュー→［すべてのアプリ］→［詳細］→［タスクバーにピン留めする］

Notepad++を起動する

［ファイル］→［新規作成］



プログラムを慎重に入力する

入力開始から完成までにかかった時間を計測する
（ストップウォッチのウェブサイトやスマホアプリ等を利用して）

*新規1 - Notepad++

ファイル(F) 編集(E) 検索(S) 表示(V) エンコード(N) 言語(L) 設定(T) ツール(O) マクロ(M) 実行(R) プラグイン(P) ウィンドウ管理(W) ?

change.log 新規1

```
<meta charset="utf-8">
<meta name="viewport" content="width=320, user-scalable=no">
<link rel="stylesheet" href="common.css" type="text/css" media="all">
</head>
<body>
<canvas id="cvs" width="320" height="320"></canvas>
<script type="text/javascript">
var cv = document.getElementById("cvs");
var ctx = cv.getContext("2d");
img = new Image();
img.onload = function() {
    ctx.drawImage(img, 0, 0, 320, 320);
};
img.src = "piano.png";
cv.addEventListener("touchstart", touchHandler, false);
var sound = [];
var filename = ["C3","D3","E3","F3","G3","A3","B3"];
for (i = 0; i < filename.length; i++) {
    sound[i+1] = new Audio("sound/"+filename[i]+".mp3");
}
function touchHandler(e) {
    x = e.touches[0].clientX;
    y = e.touches[0].clientY;
    if (x < 45)
        sound[1].play();
    else if (x < 90)
        sound[2].play();
    else if (x < 135)
        sound[3].play();
    else if (x < 180)
        sound[4].play();
    else if (x < 225)
        sound[5].play();
    else if (x < 270)
        sound[6].play();
    else
        sound[7].play();
    e.preventDefault();
}
</script>
</body>
</html>
```

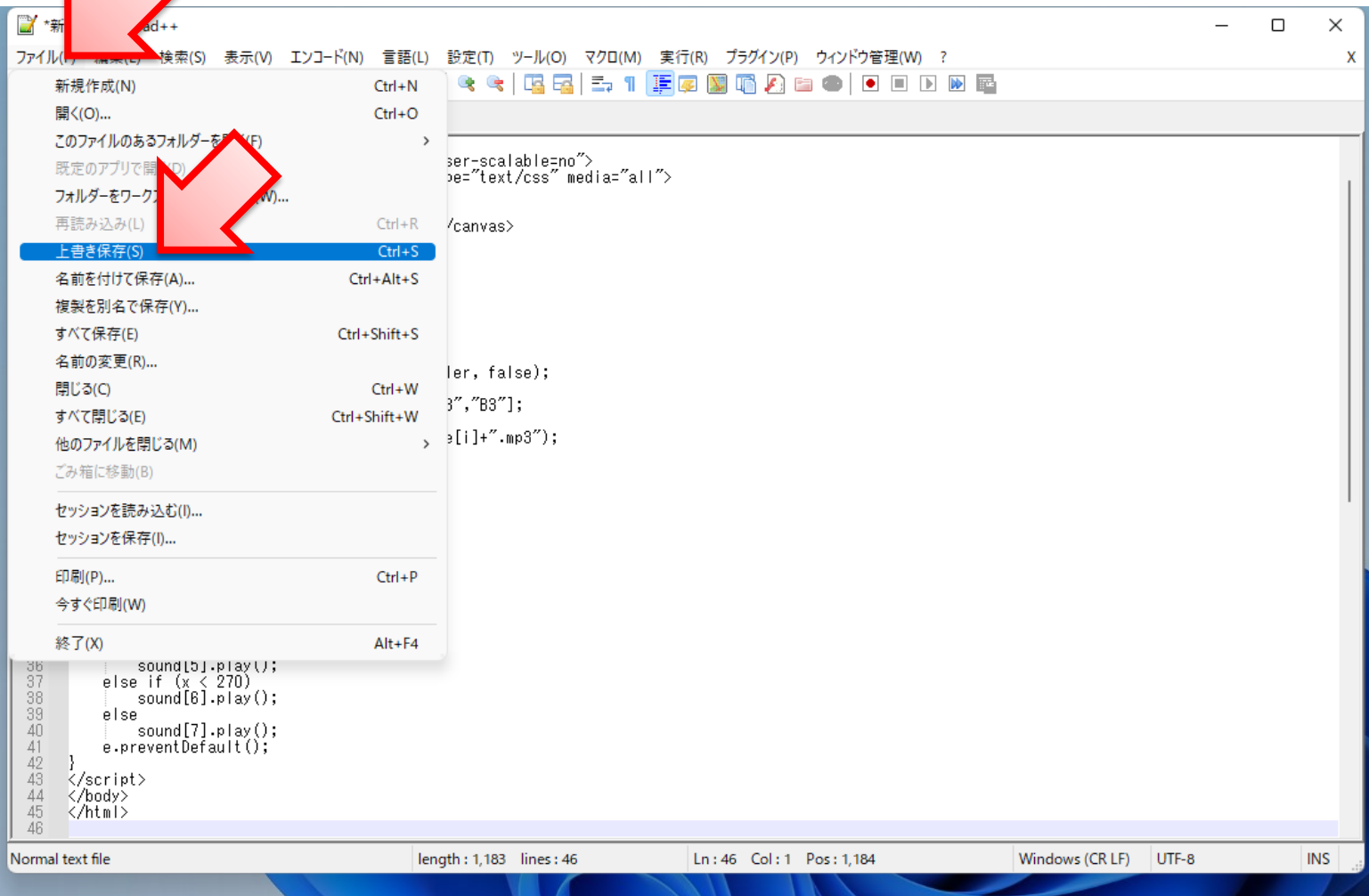
Normal text file | length : 1,183 lines : 46 | Ln : 46 Col : 1 Pos : 1,184 | Windows (CR LF) | UTF-8 | INS

【入力上の注意】
- すべて半角で打つ
  （日本語入力モードにしない）
- 大文字／小文字を正確に打つ

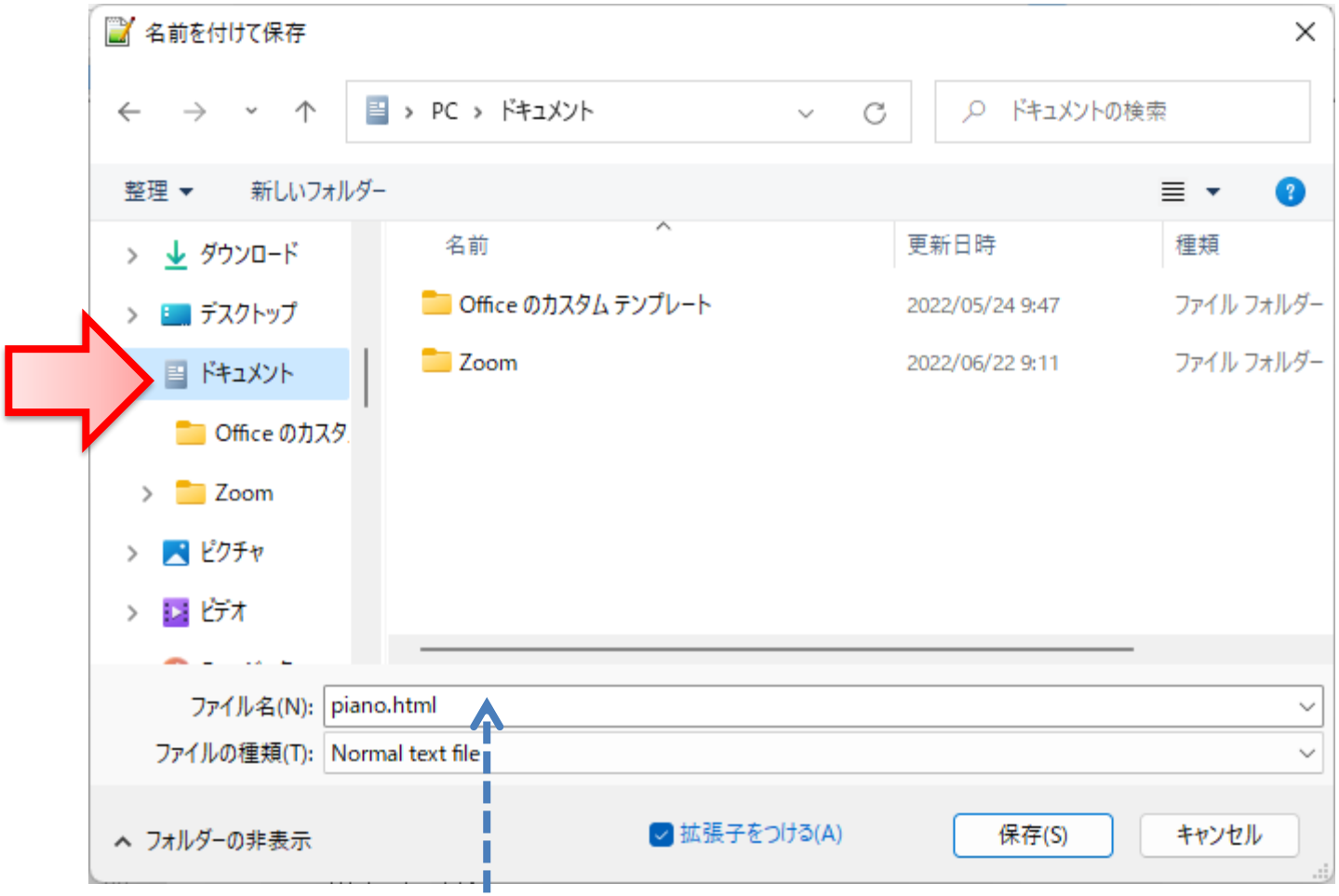
ピアノアプリ：　　分　　秒

保存する

［ファイル］→［上書き保存］



保存する



ピアノアプリ　：piano.html
太鼓アプリ　　：taiko.html

アップロードする

保護されていない通信 | daisy.fwex.tohoku-gakuin.ac.jp/progintro.html

レポートについて

- レポートサンプル
- 優秀レポート紹介 new!
- よくある質問

課題の提出サイト

- 課題１ 楽器アプリの制作 （〆切：11月28日23:59）
- 課題２ 的当てゲームの制作（〆切：12月19日23:59）
- 課題３ 天気予報アプリの制作 （〆切：1月9日23:59）

- JavaScript実行用

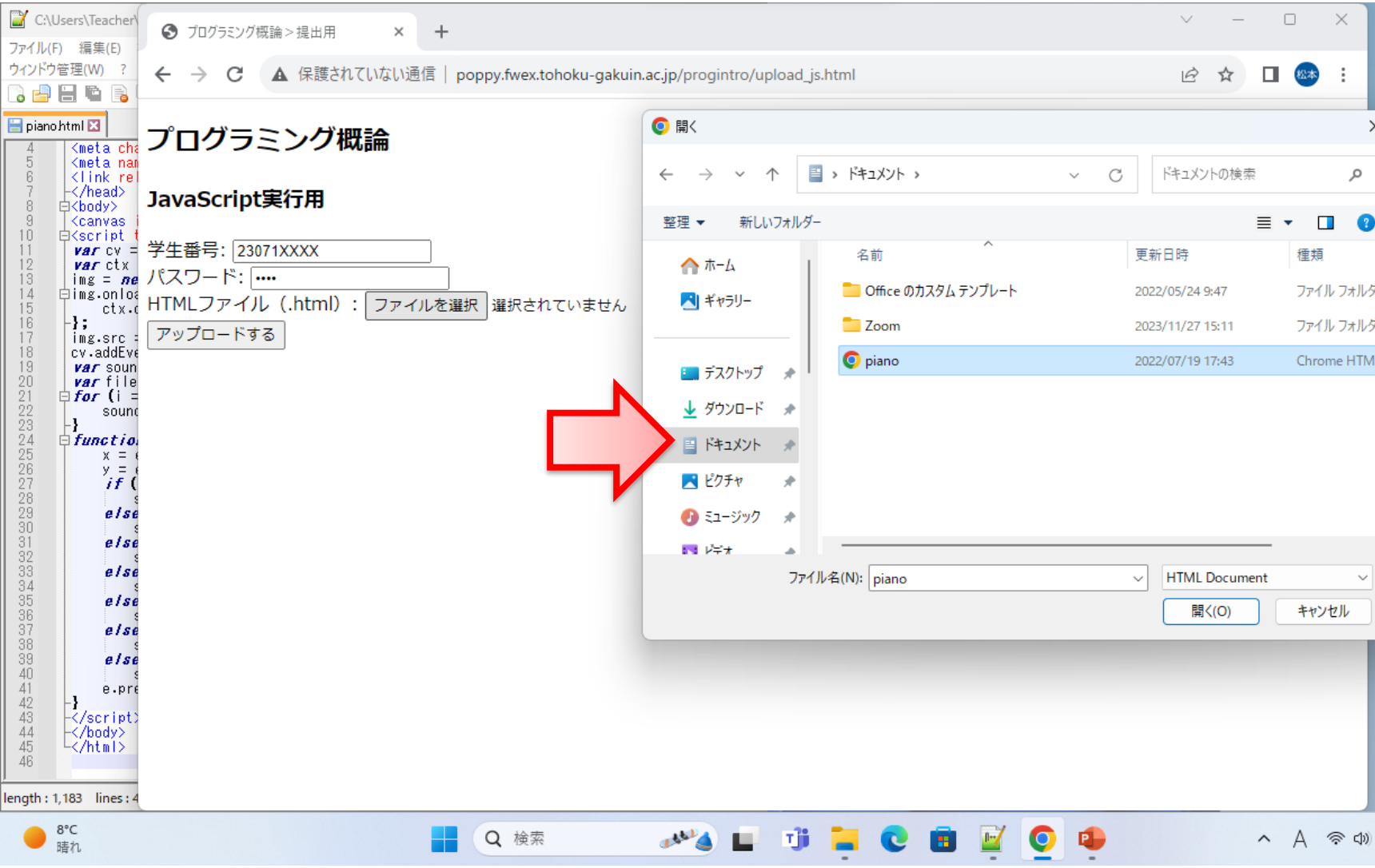
フリー素材へのリンク

- いらすとや（おみくじの検索結果）
- 音楽研究所（森のくまさん）

素材

- ピアノ

アップロードする


プログラミング概論＞提出用
保護されていない通信 | poppy.fwex.tohoku-gakuin.ac.jp/progintro/upload_js.html
プログラミング概論
JavaScript実行用
学生番号: 23071XXXX
パスワード: ••••
HTMLファイル（.html）：
ファイルを選択
選択されていません
アップロードする
開く
ドキュメント
ドキュメントの検索
整理
新しいフォルダー
ホーム
ギャラリー
デスクトップ
ダウンロード
ドキュメント
ピクチャ
ミュージック
名前
更新日時
種類
Office のカスタム テンプレート
2022/05/24 9:47
ファイル フォルダー
Zoom
2023/11/27 15:11
ファイル フォルダー
piano
2022/07/19 17:43
Chrome HTML D
ファイル名(N): piano
HTML Document
開く(O)
キャンセル

打ち込みミスの有無を確認

poppy.fwex.tohoku-gakuin.ac.jp

保護されていない通信 | poppy.fwex.tohoku-gakuin.ac.jp/progintro/

```
OK!: <!DOCTYPE html>
OK!: <html>
OK!: <head>
OK!: <meta charset="utf-8">
OK!: <meta name="viewport" content="width=320, user-scalable=no">
OK!: <link rel="stylesheet" href="common.css" type="text/css" media="all">
OK!: </head>
OK!: <body>
OK!: <canvas id="cvs" width="320" height="320"></canvas>
OK!: <script type="text/javascript">
OK!: var cv = document.getElementById("cvs");
OK!: var ctx = cv.getContext("2d");
OK!: img = new Image();
OK!: img.onload = function() {
OK!:     ctx.drawImage(img, 0, 0, 320, 320);
OK!: };
OK!: img.src = "piano.png";
OK!: cv.addEventListener("touchstart", touchHandler, false);
OK!: var sound = [];
OK!: var filename = ["C3","D3","E3","F3","G3","A3","B3"];
OK!: for (i = 0; i < filename.length; i++) {
OK!:     sound[i+1] = new Audio("sound/"+filename[i]+".mp3");
OK!: }
OK!: function touchHandler(e) {
OK!:     x = e.touches[0].clientX;
OK!:     y = e.touches[0].clientY;
OK!:     if (x < 45)
OK!:         sound[1].play();
OK!:     else if (x < 90)
OK!:         sound[2].play();
OK!:     else if (x < 135)
OK!:         sound[3].play();
OK!:     else if (x < 180)
OK!:         sound[4].play();
OK!:     else if (x < 225)
OK!:         sound[5].play();
OK!:     else if (x < 270)
OK!:         sound[6].play();
OK!:     else
OK!:         sound[7].play();
OK!:     e.preventDefault();
OK!: }
OK!: </script>
OK!: </body>
OK!: </html>
```

アップロードが完了しました.

すべての行がOK!に
なっているか確認する

poppy.fwex.tohoku-gakuin.ac.jp

保護されていない通信 | poppy.fwex.tohoku-gakuin.ac.jp/progintro/

```
OK!: <!DOCTYPE html>
OK!: <html>
OK!: <head>
OK!: <meta charset="utf-8">
OK!: <meta name="viewport" content="width=320, user-scalable=no">
OK!: <link rel="stylesheet" href="common.css" type="text/css" media="all">
OK!: </head>
OK!: <body>
OK!: <canvas id="cvs" width="320" height="320"></canvas>
OK!: <script type="text/javascript">
OK!: var cv = document.getElementById("cvs");
OK!: var ctx = cv.getContext("2d");
OK!: img = new Image();
OK!: img.onload = function() {
Alert!:     ctx.drawImage(img, 0, 0, 320, 320);
OK!: };
OK!: img.src = "piano.png";
OK!: cv.addEventListener("touchstart", touchHandler, false);
OK!: var sound = [];
OK!: var filename = ["C3","D3","E3","F3","G3","A3","B3"];
OK!: for (i = 0; i < filename.length; i++) {
OK!:     sound[i+1] = new Audio("sound/"+filename[i]+".mp3");
OK!: }
OK!: function touchHandler(e) {
OK!:     x = e.touches[0].clientX;
OK!:     y = e.touches[0].clientY;
OK!:     if (x < 45)
OK!:         sound[1].play();
OK!:     else if (x < 90)
OK!:         sound[2].play();
OK!:     else if (x < 135)
OK!:         sound[3].play();
OK!:     else if (x < 180)
OK!:         sound[4].play();
OK!:     else if (x < 225)
OK!:         sound[5].play();
OK!:     else if (x < 270)
OK!:         sound[6].play();
OK!:     else
OK!:         sound[7].play();
OK!:     e.preventDefault();
OK!: }
OK!: </script>
OK!: </body>
OK!: </html>
```

アップロードが完了しました.

Alert!の行があったら
打ち込みミスがある
可能性が高い

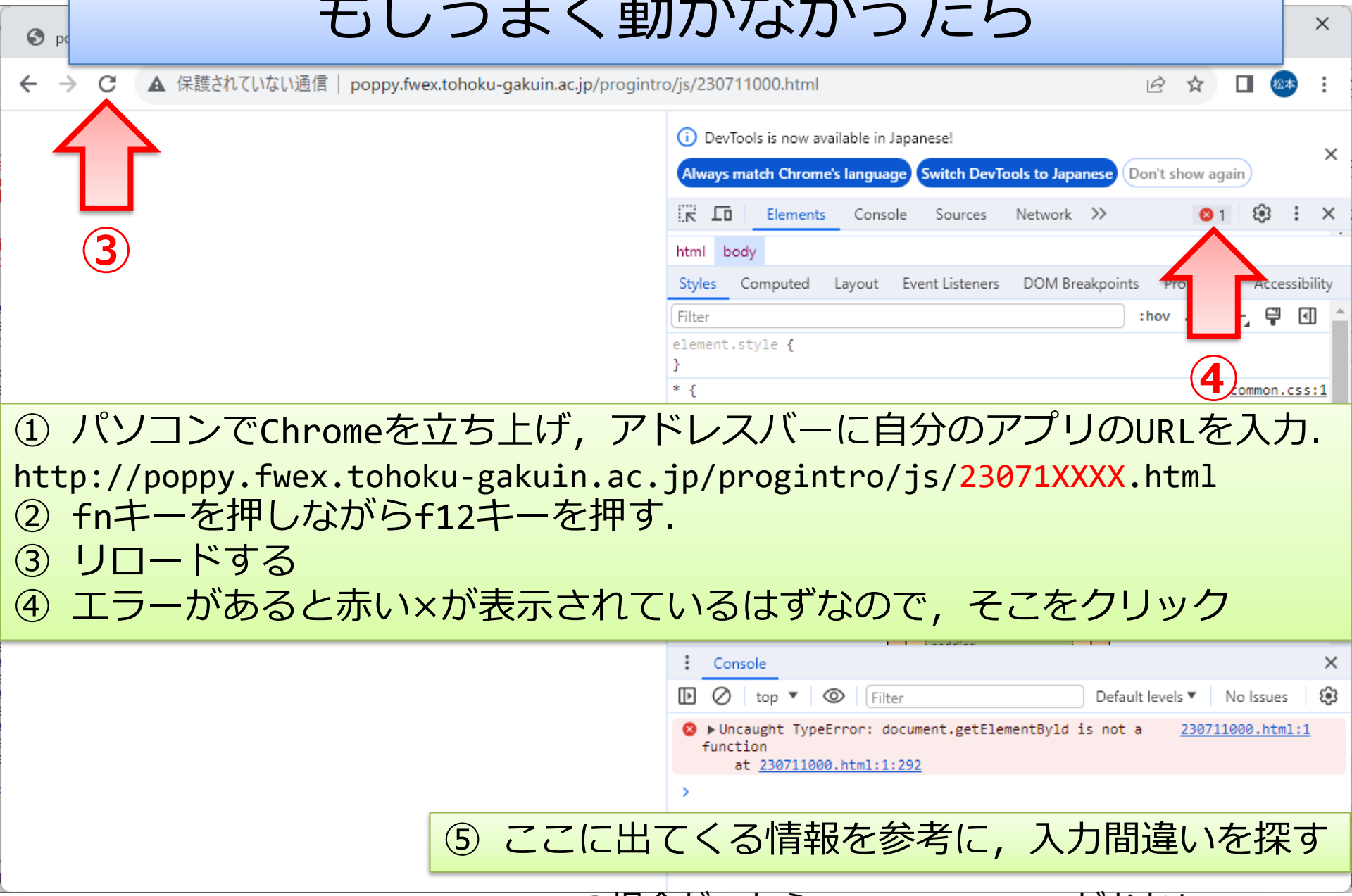
自分のスマートフォンで実行する

http://poppy.fwex.tohoku-gakuin.ac.jp/progintro/js/230710000.html

自分の学生番号に変更する

起動してすぐは反応が鈍くても，しばらく
いじっているうちに改善する可能性があります

もしうまく動かなかったら



① パソコンでChromeを立ち上げ，アドレスバーに自分のアプリのURLを入力.
`http://poppy.fwex.tohoku-gakuin.ac.jp/progintro/js/23071XXXX.html`
② fnキーを押しながらf12キーを押す.
③ リロードする
④ エラーがあると赤い×が表示されているはずなので，そこをクリック

⑤ ここに出てくる情報を参考に，入力間違いを探す

→ この場合だったら`getElementByld`がおかしい

APPINVENTORで改めてピアノをつくって作業時間を計測

改めて最初から作り直して時間を計測する

プロジェクトを新規作成してから正しく動くことを確認するまでにかかった時間

分　　秒

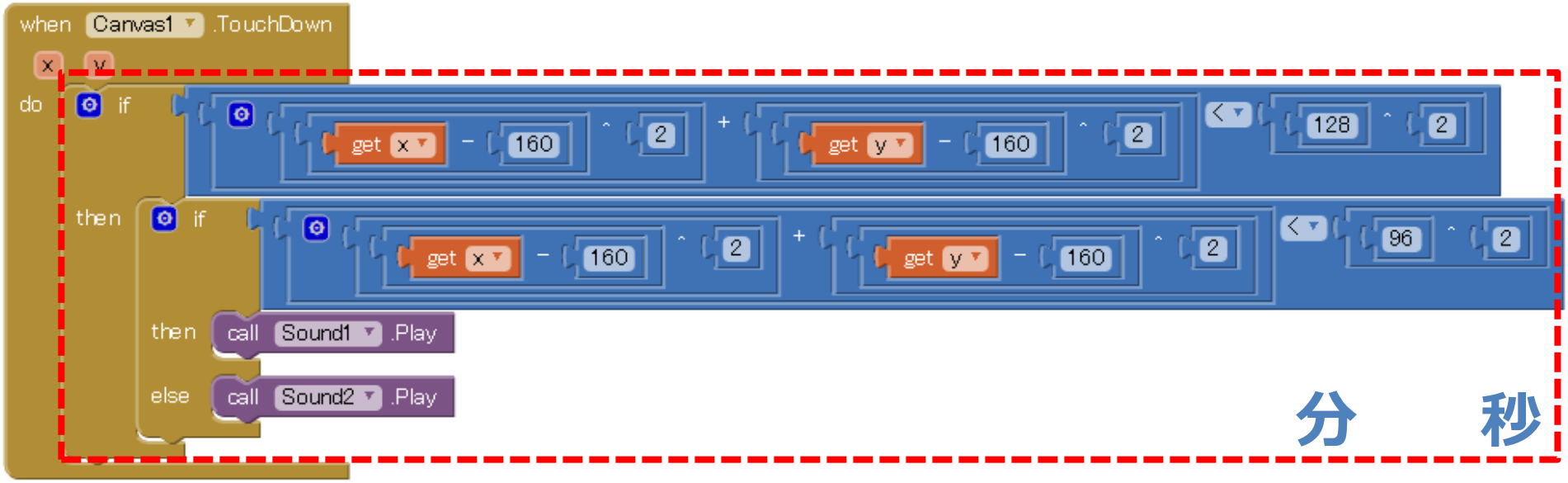
JAVASCRIPTとAPPINVENTORで太鼓をつくって入力時間を比較

入力時間の比較を行う

- 最初から最後まででではなく，一部の「文字入力」と「ブロック入力」時間のみの比較を行う
  （動作確認時間や修正時間を含まない）

デザイナ画面での作業時間は含まない

点線内の入力時間をそれぞれ計測する




```
function touchHandler(e) {
    x = e.touches[0].clientX;
    y = e.touches[0].clientY;
    if ((x - 160) * (x - 160) + (y - 160) * (y - 160) < 128 * 128)
        if ((x - 160) * (x - 160) + (y - 160) * (y - 160) < 96 * 96)
            sound[1].play();
        else
            sound[2].play();
    e.preventDefault();
}
```

分　秒

授業中に完成しなかったら

- タイムの計測中に授業の終了時刻になってしまったら，タイマーを止め，ここまでのタイムを記録する．時間があるときに作業の続きをおこない，タイムを合算すればよい.
- 授業中にすべての作業が終わらなかった場合は，次回の授業までに作業の続きをおこない，タイムを計測して記録しておくこと.