

操作ガイド

(©2024 PPAP

- Piano Practice
Application Project)

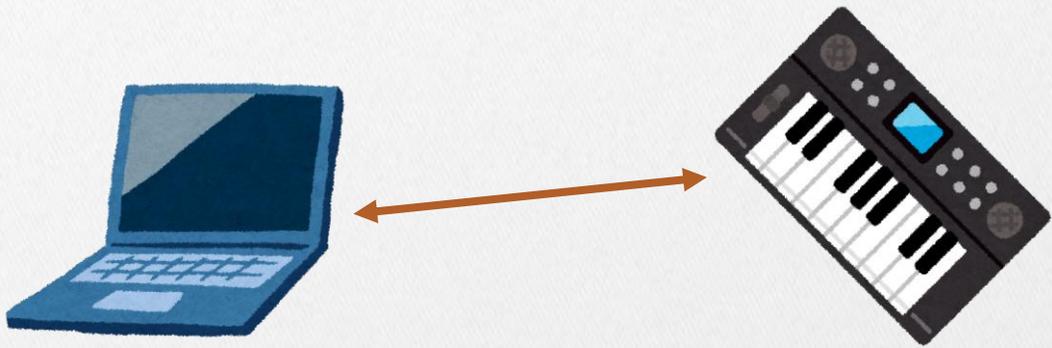
はじめに

このアプリは電子ピアノと接続してピアノ練習を行うことができるウェブアプリケーションです。ぜひ手元の電子ピアノとUSBケーブルと接続して実行してみてください。

楽譜が読めない方でも鍵盤を目で追いながら練習することができます。

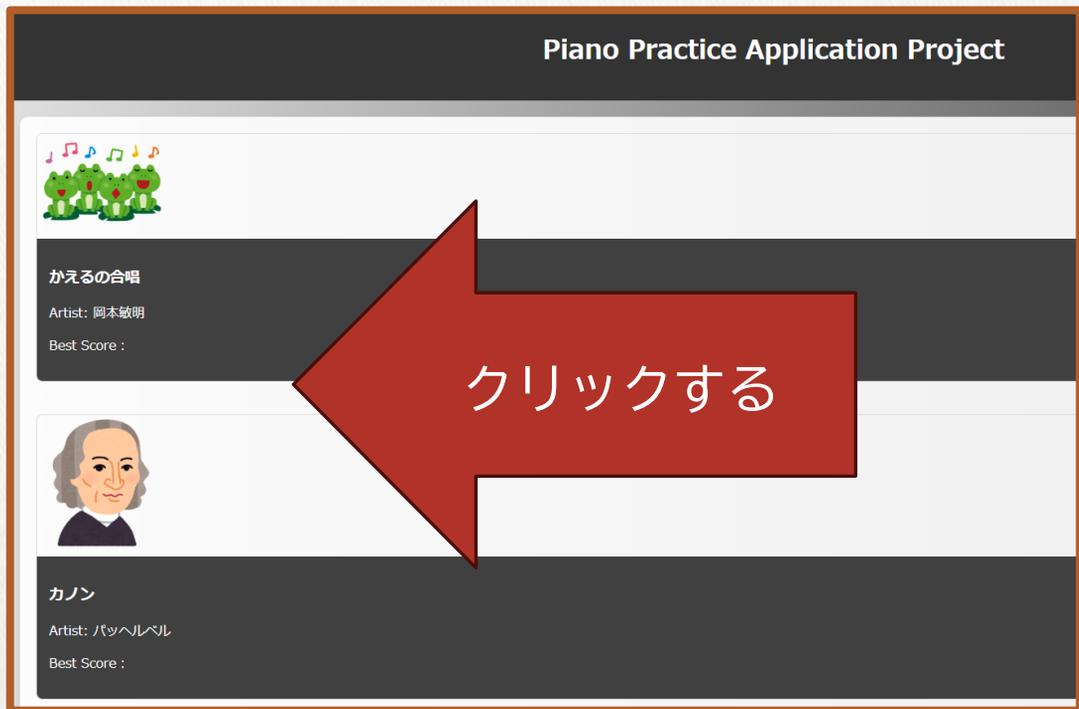
楽曲は随時追加予定です。また気が向けばアップデートを行います。

接続方法



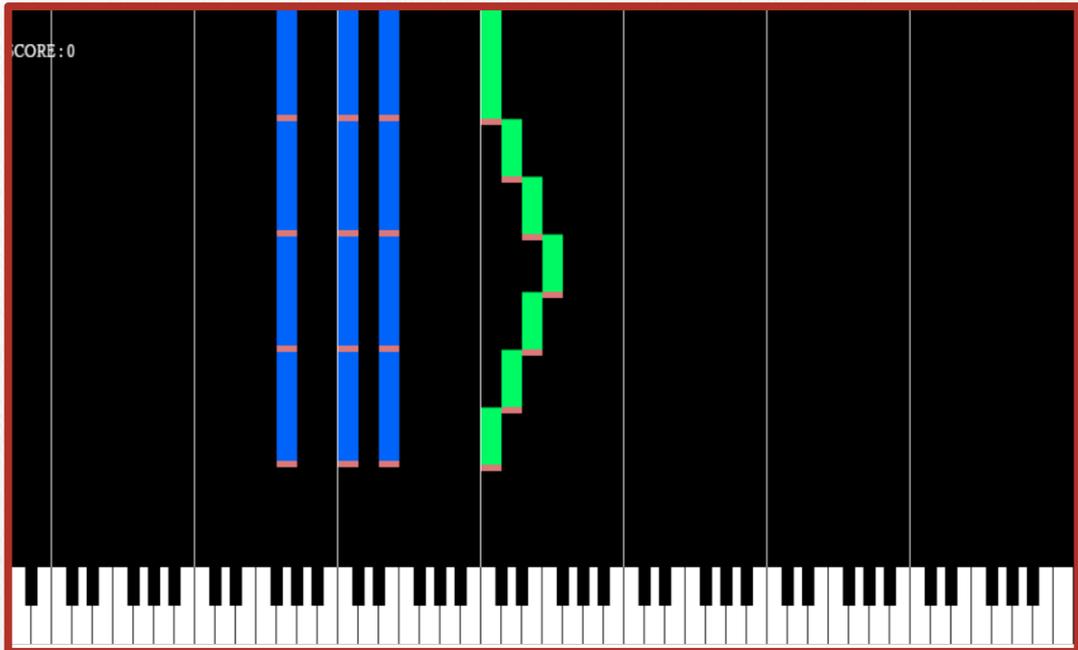
USBケーブルを使ってパソコンと電子ピアノを接続するだけでOKです。
うまくいかない場合は、アプリを開く前に接続するか、接続した後ブラウザを更新するようにしてください。

メニュー画面



弾きたい楽曲を選択したらクリックして演奏画面に入ります。

ゲーム画面(演奏)



ゲーム画面ではノーツが落ちてきます。ノーツが鍵盤に触れるタイミングで対応する場所を電子ピアノの鍵盤で弾くと、ポイントが加算されます。

緑色は右手、青色は左手で弾くのが上手に演奏するコツです。

練習画面

フレーズ: フレーズ1 | ジャンプ | 開始 | 120 BPM

Right:1 | ◀▶

ソ

Left:1 | ◀▶

シ♭ ド ミ♭ ソ シ♭ ド ミ♭ ソ

© 2024 PPAP - Piano Practice Application Project -

練習画面では自分のペースでゆっくりと練習ができます。

光る鍵盤に対応する場所を電子ピアノで弾くと次の音がガイドで表示されていき、これを繰り返すことで曲を弾くことができます。

練習モード 楽曲の追加方法 (次ページから)

自分で楽曲を
追加してみたい！



楽曲の追加方法

①: 楽譜を作成する

ファイル 編集 表示

```
const right = [  
  60, 62, 64, 65, 0, //ドレミファミレド  
  64, 62, 60, 0,  
  64, 65, 67, 69, 0,  
  67, 65, 64, 0,  
  60, 60, 0,  
  60, 60, 0,  
  60, 60, 62, 62, 64, 64, 65, 65, 0,  
  64, 62, 60, 0  
];  
  
const left = [  
  60, 60, 0, //ドド  
  60, 60, 0,  
  60, 60, 0,  
  60, 60, 0,  
  60, 60, 0,  
  60, 60, 0,  
  60, 60, 0,  
  60, 60, 0,  
  60, 60, 0,  
];
```

テキストエディタで以下のように、
`const right[]`に右手、
`const left[]`に左手の楽譜を入力します。

真ん中のドが60になるのでそれを目安に一音ずつ入力してください。

現在のバージョンでは右手、左手各一音ずつまで楽譜に保存できます。

楽譜の小節終わりの部分には0を入れてください。

楽曲の追加方法

②: jsファイルを用意する。

hosomichi.js	2024/01/24 13:30	JSFile	9 KB
kaeru.js	2024/01/23 10:58	JSFile	8 KB
kirisame.js	2024/01/20 13:14	JSFile	10 KB
main.js	2024/01/23 14:27	JSFile	31 KB
main-kanon.js	2024/01/23 14:27	JSFile	36 KB
metronome.js	2024/01/24 12:01	JSFile	2 KB
phraseController.js	2024/01/24 13:28	JSFile	4 KB
play(保存用).js	2024/01/24 13:30	JSFile	9 KB



コピーして
(曲名).html
として新た
に保存

play(保存用).html
をコピーして新曲
用のjsファイルを
用意します。そし
てファイルを
Visual Studio
Codeで開きます。

楽曲の追加方法

③: htmlファイルを用意する。

名前	更新日時
 hosomichi.html	2024/01/24 13:46
 index.html	2024/01/24 11:48
 kaeru.html	2024/01/24 11:46
 kirisame.html	2024/01/24 11:49
 play(保存用).html	2024/01/20 14:01



次にjsファイルに対応するhtmlファイルを作成します。

jsファイルと同名のhtmlファイルを作成し、Visual Studio Codeで開きます。

楽曲の追加方法

③: htmlファイルを用意する。

```
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-wid
<link rel="stylesheet" href="css/style.css">
<link rel="stylesheet" href="css/metronome.css"
<title>PPAP</title>

</head>
<body>

<!--ここに楽曲のjsファイル-->
<script src=""></script>

<!--メトロノーム
<script src=""></script>

<!--フレーズ
<script src=""></script>
```

ここに作成したjsファイル
のURLを入れる。
(例:../js/(曲名).html)

楽曲の追加方法

④: index.htmlに作成した
htmlファイルへのリンクを作成する。

名前	更新日時	種類	サイズ
 hosomichi.html	2024/01/24 13:46	Chrome HTML Do...	7 KB
 index.html	2024/01/24 11:48	Chrome HTML Do...	2 KB
 k...	2024/01/24 11:46	Chrome HTML Do...	1 KB
 kir...e.html	2024/01/24 11:49	Chrome HTML Do...	7 KB
 pl... (保存).html	2024/01/24 15:33	Chrome HTML Do...	7 KB

次にindex.htmlをVisual Studio Codeで開きます。

楽曲の追加方法

④:index.htmlに作成した
htmlファイルへのリンクを作成する。

```
<!--  
コピー用(追加時コピーして使用)  
<a href=".html">  
  <div class="musicCard">  
      
    <div class="musicInfo">  
      <h3>(ここに曲名)</h3>  
      <p>Artist:(ここにアーティスト名)</p>  
      <p>Best Score : </p>  
    </div>  
  </div>  
</a>  
-->
```

必要な情報を入力します。

を指定し、には楽曲のサムネイルを指定することができます。サムネイルに指定する画像はimageフォルダに入れましょう。

楽曲の追加方法

④: index.htmlで追加した楽曲が選べる
確認する。



index.htmlファイルを開き、追加した楽曲が選択できるかどうかを確認します。もしエラーが起きたり、アクセスできないときはhtmlファイルのパスや、楽曲のhtmlでjsファイルのパスがただしく指定されているかを確認しましょう。