

# p5.jsでアプリを作ってみよう

---

## 今回作るアプリ

くじアプリを作ります。

画面をクリックすると「当たり、ハズレ、もう一回」のいずれかが表示される。

## 完成のソースコード

先に完成のソースコードを書いておきます。一通り目を通しておいってください。

- sketch.js

```
// 項目の内容
const textList = ["当たり", "ハズレ", "もう一回"];
// 選ばれた項目
let textResult;

// --- 初回更新 -----
function setup() {
  // 表示するキャンバスを全画面に
  createCanvas(windowWidth, windowHeight);

  // 文字の基準点を上下左右の中央に
  textAlign(CENTER, CENTER);
  // 文字の大きさの設定
  textSize(200);

  // 項目の結果をランダムにセット
  textResult = random(textList);
}

// --- 1秒間に60回更新 -----
function draw() {
  // 背景を赤色に
  background("#e02020");
  // 塗りつぶしは白色に
  fill("#FFFFFF");
  // 境界線も白色に
  stroke("#FFFFFF");

  // 選ばれた項目の表示
  text(textResult, width / 2, height / 2);

  // クリックされている間の処理
  if (mouseIsPressed == true) {
    // 項目の結果をランダムにセット
    textResult = random(textList);
  }
}
```

以下の2つのファイルは前回のファイルをコピーしてください。

- index.html
- style.css

## フォルダの作成

前回作ったフォルダの中に新しいフォルダを作ります。名前は「app」としてください。

## ファイルの作成

作った「app」フォルダ内に

- index.html
- style.css

2つのファイルをコピーしておきます。

次に完成のソースコードを模写して

- sketch.js

のファイルを作成しておいてください。

以上で準備は完了です。

## ソースコードの説明

模写したソースコードの各部の説明をしていきます。

### コメント

// はそれ以降の文字がコメントになります。

コメントはプログラムとは関係無く、人間のために書きます。

自分が忘れないようにメモ代わりに使ったり、他の人にソースコードを見せるときにわかりやすいように、何を書いているのか？を説明するために使うことが多いです。

### 変数

プログラミングでは数字や文字や画像など、色々なデータを変数に入れて一時的に保存します。

書き方は `let 好きな変数名;` と書きます。

また、`const 好きな変数名;` と書くこともあります。これは一度この変数にデータを入れてしまうと別のデータに変更出来なくなります。

そういった変数は一般的に「定数」と呼ばれます。

```
// 項目の内容
const textList = ["当たり", "ハズレ", "もう一回"];
// 選ばれた項目
let textResult;
```

今回はくじの「当たり」「ハズレ」「もう一回」という3種類の項目を保存しておく定数と、その3種類の項目から選ばれた一つを保存する変数の2つを作っています。

## 関数

関数とはいくつかの処理をまとめたものです。

書き方は

```
function 好きな関数名() {  
    // ここにいくつかの処理を書く  
}
```

と書きます。

今回のアプリでは `setup` と `draw` という2つの関数を作っています。

```
function setup() {  
    // 処理は省略  
}  
  
function draw() {  
    // 処理は省略  
}
```

`setup` はプログラム起動時に一度だけ実行されます。

最初の設定などに使います。

`draw` は1秒間に60回実行されます。

ここで文字の変更などを行うことで、動いて見えるという仕組みです。

### setupの処理について

`setup` で行っている処理はどんなものがあるのでしょうか。確認していきましょう。

```
// 表示するキャンバスを全画面に  
createCanvas(windowWidth,windowHeight);  
  
// 文字の基準点を上下左右の中央に  
textAlign(CENTER, CENTER);  
// 文字の大きさの設定  
textSize(200);  
  
// 項目の結果をランダムにセット  
textResult = random(textList);
```

### drawの処理について

`draw`で行っている処理も確認しておきましょう。

```
// 背景を赤色に
background("#e02020");
// 塗りつぶしは白色に
fill("#FFFFFF");
// 境界線も白色に
stroke("#FFFFFF");

// 選ばれた項目の表示
text(textResult, width / 2, height / 2);

// クリックされている間の処理
if (mouseIsPressed == true) {
  // 項目の結果をランダムにセット
  textResult = random(textList);
}
```