

# アプリプログラミングシート ~ 図鑑

▶ 第3章まで終了した学習者向け

## アプリの概要

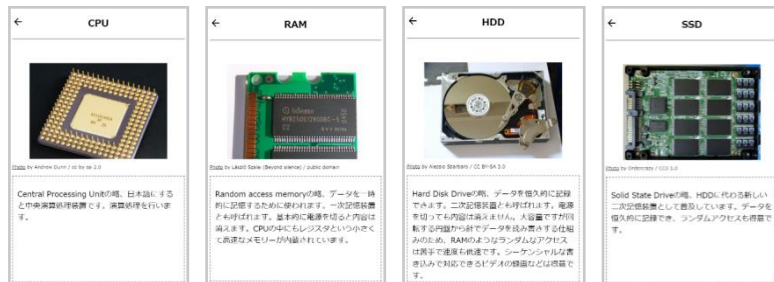
コンピュータ用語の図鑑アプリを作成する。このアプリは、トップページと各項目のページそれぞれのボタンを押すことでページ移動ができるアプリである。

使われているHTML、CSSでのページの作成方法とページ間を移動する方法（ハイパーリンク）について学ぶ。

### トップページ



### 個別項目ページ



## 学習内容

【主】 HTMLの基本的な構造とハイパーリンク(リンク)で別ページを表示する方法

【副】 その他関連する書籍対応表 (右表)

書籍	関連する内容
第1章 アプリ開発入門	Monacaで図鑑プロジェクトを編集する
第2章 HTML入門	ボタンはHTMLで用意されている
第3章 CSS入門	ボタンの色や配置を設定している

## 図鑑アプリを動かしてみよう

見本のDL (↓クリック)



上記から 図鑑アプリの完成版を『インポート』をクリック



Monacaプロジェクトにインポート

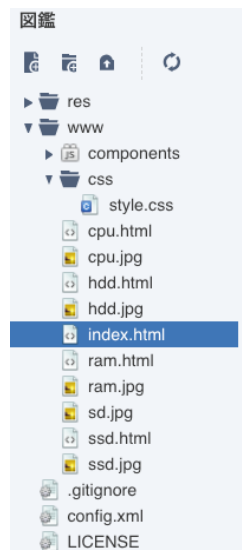
矢印を押してトップページに戻る



ボタンを押して、各項目のページを表示する

## ファイル一覧

トップページを含むすべてのページは、HTMLで作成されている。



## index.htmlを読んでみよう

```
<body>
  <nav>
    <div>&nbsp;</div>
    <h1>コンピュータ図鑑</h1>
    <div>&nbsp;</div>
  </nav>
  <div id="menu">
    <a href="cpu.html"><button>CPU</button></a>
    <a href="ram.html"><button>RAM</button></a>
    <a href="hdd.html"><button>HDD</button></a>
    <a href="ssd.html"><button>SSD</button></a>
  </div>
</body>
```

①ナビゲーション部分

②メニュー部分

③ボタンにリンクを設定

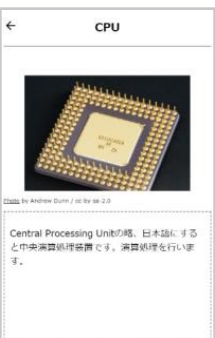


## cpu.htmlを読んでみよう

```
<body>
  <nav>
    <a href="index.html">
      <div class="material-icons">
        arrow_back
      </div>
    </a>
    <h1>CPU</h1>
    <div></div>
  </nav>
  <div id="mount">
    
  </div>
  <p id="license">
    <a href="https://commons.wikimedia.org/wiki/File:Intel_80486DX2_bottom.jpg">Photo</a>
    by Andrew Dunn / cc-by-sa-2.0</p>
  <div id="explain">
    <p>
      Central Processing Unitの略、日本語にすると中央演算処理装置です。
      演算処理を行います。
    </p>
  </div>
</body>
```

④ CSSで arrow\_back をアイコンに変換

⑤ CPUの画像を表示



# カスタマイズ① トップページの項目と新しいページを追加をしてみよう

# カスタマイズ② 図鑑アプリの配色を変更をしてみよう

## 目録 カスタマイズの概要

例：「SD Card」のページを追加してみよう

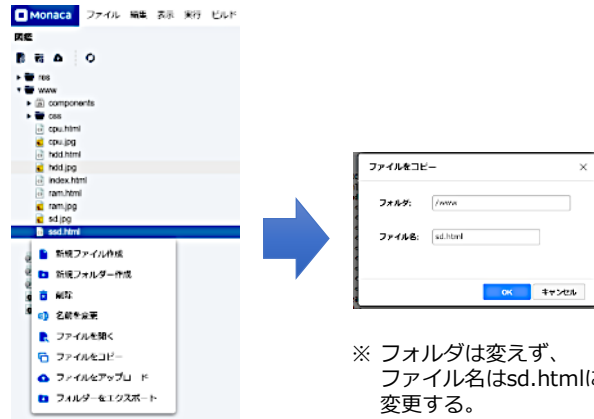


### 変更点

- sd.htmlページを作成する
  - ※ 適当なファイル(例ではssd.html)をコピーし、その内容を変更して作成する。
- index.htmlにボタンを追加する

## 目録 ファイルのコピー方法

ファイル一覧で、コピー元ファイルを右クリックすることでコピーできる。



※ フォルダは変えず、ファイル名はsd.htmlに変更する。

## 目録 index.htmlの変更例

```
<div id="menu">
  <a href="cpu.html"><button>CPU</button></a>
  <a href="ram.html"><button>RAM</button></a>
  <a href="hdd.html"><button>HDD</button></a>
  <a href="ssd.html"><button>SSD</button></a>
  <a href="sd.html"><button>SD Card</button></a>
</div>
```

ボタンとリンクの追加

## 目録 新しい項目ページ作成時の変更箇所と変更例

```
<nav>
  <a href="index.html"><div class="material-icons">arrow_back</div></a>
  <h1>SSD</h1>
  <div></div>
</nav>
<div id="mount">
  
</div>
<p id="license">
  <a href="https://commons.wikimedia.org/wiki/File:Embedded_World_2014_SSD.jpg">
    Photo
  </a>
  by Ordercrazy / CC0 1.0
</p>
<div id="explain">
  <p>
    Solid State Driveの略、HDDに代わる新しい二次記憶装置として普及しています。
    データを恒久的に記録でき、ランダムアクセスも得意です。
  </p>
</div>
```

h1(大見出し)の変更

大見出しの変更例  
<h1>SD Card</h1>

画像の変更

画像の変更例  


ライセンス表記の変更

ライセンス表記の変更例  
<a href="https://commons.wikimedia.org/wiki/File:SD\_Cards.jpg">  
Photo  
</a>  
by Rstoplabe14 / public domain

※今回の画像は、著作権が放棄された画像(パブリックドメイン)のため、ライセンス表記は削除可。

説明文の変更

説明文の表記例  
挿抜を容易に行えるカード型の補助記憶装置です。

⇒ 他にも自分でページを追加してみよう。

## 目録 カスタマイズの概要



CSSを使ってトップページの配色を変更する。

### 変更点

- 背景色を『cyan』に変更
- ボタンの背景色を『red』に変更
- ボタンの文字色を『white』に変更

## 目録 css/style.cssの変更例

```
body{
  background-color: cyan;
}

#menu button {
  width:100%;
  height:50px;
  padding:10px;
  margin-bottom:10px;
  border-radius:10px;
  color: white;
  background-color: red;
}
```

全体(bodyタグ内)の背景色を変更

ボタンの文字色を変更

ボタンの背景色を変更

※ 色は好きな色に変更して構いません。

## 目録 カラーコードによる色指定

色の表現方法は、「red」や「blue」などの色の名称を指定する方法のほかに、カラーコードと呼ばれる色の指定方法がある。

カラーコードによる例



赤がff (255)、緑が00 (0)、青がff (255) なので、光の三原色の赤と青を混ぜた色(マゼンタ)が色指定されます。

※16進数等、詳しくは書籍を参考

# アプリプログラミングシート ~DX図鑑~

▶ 第12章まで終了した学習者向け

## アプリの概要



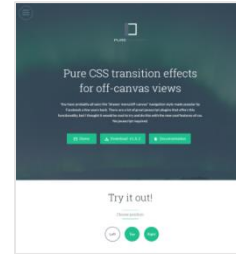
メニューボタン(☰)を押すと、メニューの項目が表示される。  
項目を選ぶと画像や説明文が表示される。

## 学習内容

- 【主】 繰り返しの処理と配列
- 【副】 その他関連する内容(書籍対応表)

書籍	関連する内容
第1章 アプリ開発入門	MonacaでDX図鑑プロジェクトを編集する
第2章 HTML入門	画像やボタンはHTMLで用意されている
第3章 CSS入門	画像や文章の表示領域を整えている
第4章 JavaScript入門	図鑑配列を変数に代入している
第6章 関数	全体をonLoad()関数としてまとめている
第7章 イベント	onloadイベントでonLoad()関数を呼び出している
第8章 DOM	メニューがクリックされると図鑑の画像や文章が書き換わる
第11章 配列	図鑑の内容を配列形式でまとめている
第12章 繰り返し	配列の内容をDOMでli要素として追加する処理を繰り返しで行っている

## メニューの仕組みについて

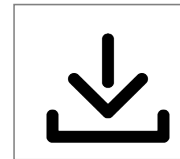


このアプリのメニューは、インターネット上で公開され、無料で自由に使用できる『Pure Drawer』というツールで作成されている。(※1)  
『Pure Drawer』自体はCSSだけで記述されており、HTML側に特定のタグを記述するだけで典型的なデザインのメニューを実装することができる。  
CSSの内容はcssフォルダ内の『pure-drawer.css』ファイルで確認できる。

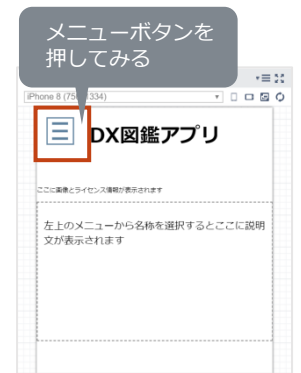
参考: <http://mac81.github.io/pure-drawer/>  
※1 MITライセンスという制限がほぼ無いライセンス

## DX図鑑を動かしてみよう

見本のDL (↓クリック)



上記から DX図鑑アプリの『インポート』をクリック



## DX図鑑のフローチャート(簡易)を確認しよう



## プログラムを読んでみよう

`<script src="data.js"></script>` } ① 図鑑配列を取得  
`<script>`

```
function onLoad(){
  var menu = document.getElementById("menu");

  for (var i = 0; i < data.length; i++) {
    var li = document.createElement("li");
    li.innerHTML = data[i].title;
    li.dataset.index = i;
    li.onclick = function (event) {
      key = event.target.dataset.index;

      document.getElementById('pure-toggle-left').checked = false;
      document.getElementById('title').innerHTML = data[key].title;
      document.getElementById('photo').src = data[key].image;
      document.getElementById('explain').innerHTML = data[key].explain;
      document.getElementById('license').innerHTML = data[key].license;
    }
    menu.appendChild(li);
  }
}
```

`</script>`

④ メニューの項目がクリックされたときの処理

} ⑤ メニュー項目を追加  
} } ⑥ 繰り返し処理の終わり

# カスタマイズ① DX図鑑に項目を追加してみよう

# カスタマイズ② DX図鑑のタイトルにルビを追加してみよう

## 目 カスタマイズの概要

DX図鑑の内容「SD Card」を加えてみよう。



### 変更点

- 配列 data.js にSDカードの情報を追加

## 📌 画像の探し方

画像を利用するには著作権を考慮する必要があり、インターネット上で公開されている画像であっても自由に使えるわけではない。作品に組み込める画像の入手先として、例えば『Wikimedia Commons』などがあり、条件付きで自由に利用可能(クリエイティブ・コモンズ・ライセンス)な画像や著作権が放棄された(パブリックドメイン)画像が公開されている。



引用：  
[https://commons.wikimedia.org/wiki/File:SD\\_Cards.JPG](https://commons.wikimedia.org/wiki/File:SD_Cards.JPG)

## 📄 JavaScriptの変更

要素を追記しよう

```

1 var data = [
2   {
3     title: 'CPU',
4     explain: 'Central Processing Unitの略、日本語にすると中央演算処理装置です。演算処理を行います。',
5     image: 'cpu.jpg',
6     license: '<a href="https://commons.wikimedia.org/wiki/File:Embedded_World_2014_SSD.jpg">Photo</a> by Ordercrazy / CC0 1.0',
7   },
8   {
9     title: 'RAM',
10    explain: 'Random Access Memoryの略、データを記憶するための記憶装置とも呼ばれます。',
11    image: 'ram.jpg',
12    license: '<a href="https://commons.wikimedia.org/wiki/File:SD_Cards.JPG">Photo</a> by Rstoplabe14 / public domain',
13  },
14 ],
15 ];

```

カンマ注意

図鑑情報を入力

## 目 カスタマイズの概要

タイトルにルビを加えてみよう



## 📄 配列の変更

data.jsの各要素にruby情報(ルビの情報)を加える

```

1 var data = [
2   {
3     title: 'CPU',
4     ruby: 'シーピーユー',
5     explain: 'Central Processing Unitの略、日本語にすると中央演算処理装置です。演算処理を行います。',
6     image: 'cpu.jpg',
7     license: '<a href="https://commons.wikimedia.org/wiki/File:Intel_80486DX2_bottom.jpg">Photo</a> by Andrew Dunn / cc-by-sa-2.0',
8   },
9   {
10    title: 'RAM',
11    ruby: 'ラム',
12    explain: 'Random Access Memoryの略、データを記憶するための記憶装置とも呼ばれます。',
13    image: 'ram.jpg',
14    license: '<a href="https://commons.wikimedia.org/wiki/File:Embedded_World_2014_SSD.jpg">Photo</a> by Ordercrazy / CC0 1.0',
15  },
16 ],
17 ];

```

ruby(ルビ)追加



課題にチャレンジしてみよう

- ① オリジナルの図鑑アプリを考えてみよう。
- ② 配列と繰り返しを活用したプログラムの例を調べてみよう。
- ③ 著作権について調べてみよう。

## 📄 JavaScriptの変更

ruby付きでタイトルを出力するプログラム

```

document.getElementById('title').innerHTML =
  "<ruby>" + data[key].title + "</rt>"
  + data[key].ruby + "</rt>" + "</ruby>";

```

- ※ 実際には1行で記述して構わない。
- ※ しっかりプログラムを読んでみよう！