

高等学校〈共通教科情報科〉  
プログラミング教育  
学習指導案





## はじめに

和歌山県では、コンピュータやネットワークの仕組み、これらを活用し進展している社会の動きについて、基礎的な知識を身に付けさせるとともに、これらに親しんで、使いこなすことができる力を育むことができるよう、県内全ての小・中・高・特別支援学校において、発達の段階に応じた体系的なICT教育を実施します。(次ページの体系図参照)

### 高等学校におけるプログラミング教育

- 高等学校におけるプログラミング教育は、プログラミング学習応用期と捉え、中学校までにビジュアル言語などを通して学んだことを発展・活用させ、テキスト言語を用いたプログラミングを学ぶことで、コンピュータやネットワークを活用する知識と技能を身に付ける。
- 様々な事象を、プログラミングを用いてモデル化し、評価・改善することで、生活や社会の中にある問題の発見と解決に取り組む姿勢や論理的な思考力を身に付ける。
- 共通教科情報科必履修科目において年間20時間程度のプログラミング教育を実施する。

各学校においては、本指導案の内容を生かしつつ、学校の特質や教育課程の実態に応じて効果的なプログラミング学習が行えるよう創意・工夫した授業に取り組んでください。その際、学校間で取組状況などの情報を交換し、その後の授業に生かすことも大切です。プログラミングを学ぶことで、生徒がコンピュータを使いこなし、生活や社会の中でコンピュータを活用し、課題を発見し解決する力を身に付けるなど、これからの社会をより良く生きていく力を育まれることを期待します。

平成31年3月

和歌山県教育委員会

# きのくにICT教育の体系

校 種	学 年	時 間	コンピュータを使いこなす力 ※プログラミング・AIを通じて、コンピュータに人間の意図した処理を行わせることができる能力	コンピュータやネットワーク等を活用する力			
			集中的に力をつける		1年を通じて力をつける		
			時 間	能 力	時 間	能 力	
高 校 (応用期)		70時間	20/70時間 (情報科)	<ul style="list-style-type: none"> <li>テキスト言語によるプログラミングを用いて事象をモデル化する力</li> <li>生活や社会の問題を、コンピュータで解決しようとする力</li> <li>AI技術を体験し、活用しようとする力</li> </ul>	50/70時間 (教科書)	<div style="display: flex; justify-content: space-between;"> <div style="writing-mode: vertical-rl; transform: rotate(180deg);">アプリケーション操作スキル</div> <div style="border: 1px solid gray; padding: 5px;"> <p>【高校】</p> <p>&lt;専門科&gt; CAD、デザイン、文書処理などのスキルの習得 等</p> <p>&lt;普通科&gt; 資料の収集や処理、発信 等</p> <p>【中・高】 統計教育での表計算ソフトの活用とスキルの習得 等</p> <p>【中学校】 調べたことのまとめを、プレゼンで表す技術の習得 等</p> <p>【小学校】 文字入力やマウス操作の習得 等</p> <div style="border: 1px solid gray; padding: 5px; margin-top: 10px;">各教科等にて、実施</div> </div> </div>	
	中 学 校 (基礎期)	3	20時間	10/17.5時間 技術・家庭科(技術分野)	<ul style="list-style-type: none"> <li>計測・制御のプログラミングで問題を解決する力</li> <li>プログラミング(ビジュアル言語)でロボットに意図した動きをさせる力</li> <li>テキスト言語の体験</li> </ul>		10時間
		2	20時間	10/35時間 技術・家庭科(技術分野) (※別途、情報の技術10時間)	<ul style="list-style-type: none"> <li>ネットワークを利用した双方向性のあるコンテンツのプログラミング(ビジュアル言語)で問題を解決する力</li> </ul>		10時間
1		15時間	5/35時間 技術・家庭科(技術分野)	<ul style="list-style-type: none"> <li>ビジュアル言語での基礎的なプログラミングをする力</li> <li>プログラミングで小型コンピュータに意図した動きをさせる力</li> </ul>	10時間		
小 学 校 (体験期)	6	18時間	8時間 総合的な学習の時間や、各教科等からまとまった時間を確保	<ul style="list-style-type: none"> <li>ビジュアル言語でプログラミングに慣れ親しむ</li> <li>プログラミングで、ロボット教材に、意図した動きをさせる力(初級)</li> </ul>	10時間		
	5	18時間	8時間 総合的な学習の時間や、各教科等からまとまった時間を確保		10時間		
	4 3 1		コンピュータの活用や、アンプラグドでプログラミング的思考を育む				

## 目次

	ページ
指導用教材・資料	4
授業実施にあたって	5
アプリプログラミングシート使用例	6
<b>Monaca で学ぶはじめてのプログラミング ～ モバイルアプリ入門編 ～ 指導案</b>	
第 1 章 はじめてのアプリ開発 ～ 導入／開発環境の使い方を学ぼう ～	8
第 2 章 はじめてのHTML	11
第 3 章 はじめてのCSS	14
第 4 章 JavaScript 入門	17
第 5 章 条件分岐	19
第 6 章 関数	21
第 7 章 イベント	24
第 8 章 DOM	26
第 9 章 フォーム	28
第 10 章 いろいろな演算子	31
第 11 章 配列	34
第 12 章 繰り返し	37
<b>アプリプログラミングシート学習段階表</b>	40
<b>アプリプログラミングシート 指導案</b>	
1 HTML／CSS を使ったアプリ(図鑑)の作成	41
2 条件による分岐を使ったアプリ(おみくじ)の作成	48
3 繰り返し処理を使ったアプリ(貯金シミュレーションアプリ)の作成	58
4 配列と繰り返しを使ったアプリ(DX 図鑑)の作成	68
5 センサーとオープンデータを使ったアプリ(地図)の作成	78

## 指導用教材・資料

### 1 クラウド型アプリ開発環境「Monaca」

ログイン先：<https://monaca.education/ja/login>

動作環境：インターネット接続及び Google Chrome (Windows、Mac 共通)。

ただし、android、iOS 上で動作するモバイル版は動作対象外。

アカウント：一人につき 1 アカウントを使用。

### 2 書籍「Monaca で学ぶはじめてのプログラミング～モバイルアプリ入門編～」

生徒が使用するテキストブック。

Monaca を使用してモバイルアプリを作成するための基本的な学習教材。

### 3 授業用スライド (Monaca で学ぶはじめてのプログラミング～モバイルアプリ入門編～)

テキストに合わせた授業用スライド (パワーポイントにより作成)。

※ サンプルプログラム及び授業用スライドは下記 URL よりダウンロード

<https://edu.monaca.io/wakayama>

### 4 学習用教材 (アプリプログラミングシート)

A 3 用紙でプリントアウトすることを想定し、生徒に配付して学習する教材として作成したもの。

学習段階に合わせ、適切なものを選択して使用する。

和歌山県教育センター学びの丘／きのくに学習館.NET / きのくに e ラーニングよりダウンロード。

内容：図鑑

おみくじ

貯金シミュレーションアプリ

D X 図鑑

地図 (基本編)

地図 (オープンデータ編)

### 5 タブレット PC 「iPad」

制作したアプリの動作確認用に使用。「Monaca for Study (教育版 Monaca デバッガー)」を使用して確認。

## 授業実施にあたって

### 1 事前準備

- (1) Google Chrome のインストール
- (2) 教員用・生徒用アカウントの取得・配付（和歌山県教育委員会が一括で取得し、各学校へ配付。）
- (3) 学習用教材（アプリプログラミングシート）の印刷  
（A 3用紙での印刷および PowerPoint データについては環境によって、一部、画面表示と印刷が異なる場合があるので PDF データでの印刷を推奨。）
- (4) iPad の動作確認および「Monaca for Study（教育版 Monaca デバッガー）」のインストール

### 2 授業展開例

効果的なプログラミング教育を実施するために、指導用教材・資料 2 および 4 を各学校の実状に合わせて適切に組み合わせる。その際、他分野の学習内容との関わりに留意し、効率的な学習が行えるよう計画的に進める。また、生徒が主体的で対話的な深い学びやアクティブラーニングによる協働を行うことで、課題解決に向けて自ら考えられるように授業を展開する。

#### 【授業展開例 1】

時間	段階	教材	主な内容
1 時間目 ～ 1 2 時間目	導入／ 基礎	Monaca で学ぶはじめてのプログラミング ～モバイルアプリ入門編～	第 1 章 アプリ開発入門 ～第 1 2 章 繰り返し
1 3 時間目 ～ 1 6 時間目	発展	アプリプログラミングシート 地図アプリ（基本編）	関数の使用とセンサーによる情報の取得
1 7 時間目 ～ 2 0 時間目	発展	アプリプログラミングシート 地図アプリ（オープンデータ編）	オープンデータの活用

#### 【授業展開例 2】

時間	段階	教材	主な内容
1 時間目	導入	Monaca で学ぶはじめてのプログラミング ～モバイルアプリ入門編～	第 1 章 アプリ開発入門
2 時間目 ～ 4 時間目	基礎 1	アプリプログラミングシート 図鑑アプリ	HTML の基本的な構造と ハイパーリンク
5 時間目 ～ 8 時間目	基礎 2	アプリプログラミングシート おみくじ	条件による分岐 乱数の取得
9 時間目 ～ 1 2 時間目	基礎 3	アプリプログラミングシート 貯金シミュレーションアプリ	繰り返し フォームの値の取得
1 3 時間目 ～ 1 6 時間目	基礎 4	アプリプログラミングシート DX 図鑑アプリ	配列 繰り返し
1 7 時間目 ～ 2 0 時間目	発展	アプリプログラミングシート 地図アプリ（センサー活用編）または 地図アプリ（オープンデータ編）	関数の使用とセンサーによる情報の取得またはオープンデータの活用

## アプリプログラミングシート使用例

### 1 準備

(1) 下記HPよりデータをダウンロードする。

和歌山県教育センター学びの丘／きのくに学習館.NET／きのくにeラーニングよりダウンロード

(2) A3用紙（推奨）で印刷する。

[ 両面印刷 / 片面印刷 ] は授業の実態に合わせる。

[ 学習集団全員に配付 / グループに1枚配付 ] は生徒の実態に合わせ、必要枚数を印刷する。

### 2 授業展開

① 3～4人の学習グループを作る。

② グループ毎にアプリプログラミングシートを配布する。

全員に配付するか、グループに1枚配付するかは生徒の実態に合わせる。

③ 一人一人の生徒が、アプリプログラミングシート1ページ目の指示に沿って、サンプルアプリをインポートし、実際にアプリを動作させ、教師が解説をしながら理解に導く。

その際、できるだけ生徒個人または生徒同士の教え合いの活動で理解できるよう促す。

④ アプリプログラミングシート2ページ目以降のカスタマイズを、グループで助け合いながら実習する。

⑤ グループ内で協働しながら、「課題にチャレンジしてみよう」に取り組み、制作したアプリをクラス内で互いに発表を行う。(グループで一つの制作を行ってもよい(プロジェクトの共有機能を使う等)。ただし、グループ内での個人の作業や学習活動に偏りが出ないように配慮する必要がある。)



Monacaで学ぶはじめてのプログラミング

～モバイルアプリ入門編～ 指導編

## 第1章：はじめてのアプリ開発 ～ 導入/開発環境の使い方を学ぼう

### 学習目標

(知識及び技能)

- プログラムを記述するためにはファイルの編集や保存が必要であることがわかる。
- 統合開発環境を使ったファイルの編集や保存を行うことができる。

(思考力, 判断力, 表現力)

- アプリを自分たちで作成できることに気づく。
- アプリを作成するためには適切な開発ツールを選択し、組み合わせる必要があることを理解できる。

(学びに向かう力, 人間性等)

- デジタルの消費者から生産者となる視点を持つ。
- 統合開発環境が提供する諸機能に興味を持つとする。
- 他の種類のプログラミングツールがあることに気づく。

### 前提条件


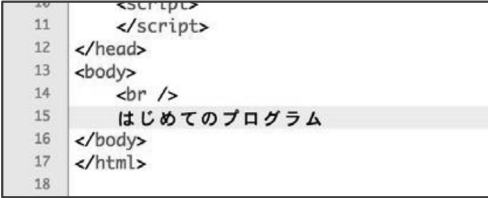
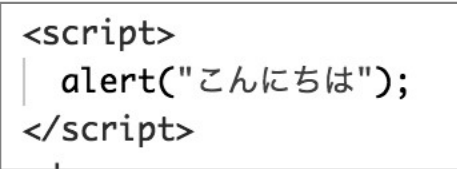
コンピュータの基本的な操作（クリック、ドラッグ、文字入力）ができる。  
ブラウザで Web サイトを利用することができる。

### 準備

Google Chrome のインストール、Monaca アカウントの作成、書籍の配付、MonacaEducation 教材紹介ページ(html)の配付

授業用サンプルプロジェクト：Monaca 標準の『最小限のテンプレート』

指導過程 第1章：はじめてのアプリ開発 ～ 開発環境の使い方を学ぼう

指導過程	教師の働きかけ	備考	書籍	スライド
学習課題の把握	<ul style="list-style-type: none"> <li>自分たちが使うアプリにはどのようなものがあるか考えさせる。</li> </ul>	<ul style="list-style-type: none"> <li>アプリにはジャンルがある。 →生活便利系、SNS系、ゲーム系、写真系</li> <li>デジタルの消費者から生産者となる視点を意識させる。</li> </ul>		
基本操作から、最初のプロジェクトを作成する。	<ul style="list-style-type: none"> <li>Monaca のサイトにアクセスし、配付したユーザ名とパスワードを使ってログインさせる。</li> <li>新規プロジェクトを作成させる。</li> </ul> 	<ul style="list-style-type: none"> <li>パスワード入力時に大文字小文字の区別を注意させる。</li> <li>Monaca のプロジェクトを開いた際に、ソースコードの中で使われているプログラミング言語の名前や特徴について大まかに説明する。</li> </ul> <p>(説明内容の例)</p> <ul style="list-style-type: none"> <li>html ファイル (表示内容そのもの)</li> <li>css ファイル (画面の装飾用)</li> <li>jpg 等の画像ファイル</li> <li>各種設定ファイル(歯車アイコン)</li> </ul>	10 -11	8 -9
プロジェクトを編集する。	<ul style="list-style-type: none"> <li>ソースコードを編集して保存させる。</li> <li>リロードについて説明する。</li> </ul> 	<ul style="list-style-type: none"> <li>コードエディターの拡大縮小の機能を確認 (Ctrl+<input type="text"/> または <input type="text"/>)</li> <li>プレビュー画面への反映を確認 (反映されない場合再読み込み)</li> <li>未保存のファイルを編集しているエディターのタブには『*』記号が付くことを説明する (保存忘れによるトラブルを未然に防ぐ)。</li> </ul>	12 -15	15 -17
簡単なプログラムを実行する	<ul style="list-style-type: none"> <li>script タグの間に記述したソースコードがプログラムとして実行されることを説明する。</li> <li>書籍に従って様々な文字をダイアログ表示するプログラムを記述するように指示する。</li> </ul> 	<ul style="list-style-type: none"> <li>script タグの外に記述した場合は正しく実行されない。</li> <li>ダブルクォート(")で文字列を囲むことやセミコロン(;)を付けることに注意を促す。</li> <li>初めてのプログラミングをしたことに言及する。</li> </ul>	15 -16	18 -19
【発展1】デバッガーを活用する	<ul style="list-style-type: none"> <li>Monaca で作成したプロジェクトをタブレットで動作させてデバッグしながらプログラミングができることを説明する。</li> <li>デバッガーから代表者のユーザ名とパスワードを使ってログインさせる。</li> </ul>	<ul style="list-style-type: none"> <li>一つのサービスをパソコンとスマートデバイスの両方から利用できることを元に、クラウド型サービスの特徴を説明する。</li> </ul>	17 -18	21 -22
【発展2】他のサンプルアプリを開き動作を変更する	<ul style="list-style-type: none"> <li>「ブロック崩しゲーム」等を開く</li> <li>1 つの変数の値を変更するとアプリの動作が大きく変わることを示し、体験させる。</li> </ul>		-	-

まとめ	<ul style="list-style-type: none"><li>● 簡単なアプリを作成したこと、アプリ作成にはHTML/JavaScript等の各種ファイルが集まったプロジェクトが必要であることを確認し、次回以降の活動への意欲を喚起する。</li></ul>		-	-
-----	---	--	---	---

## 第 2 章 : はじめての HTML

### 学習目標

(知識及び技能)

- HTML が Web ページ・Web アプリ作成に使用されていることを理解する。
- HTML の基本的な構造を理解する。
- URL やファイル・フォルダの構造を理解する。
- ファイルの種類(テキスト、画像)、拡張子について理解する。

(思考力, 判断力, 表現力)

- 一つのアプリには複数のファイルが必要であり、それらが関連づけられていることを理解できる。
- 作成した HTML をインターネット上で公開する手段を考え、サーバの必要性について理解できる。
- HTML を活用して情報発信する方法を理解し、その意味を考えることができる。

(学びに向かう力, 人間性等)

- 興味のある Web ページのソースコードを実際に確認する。
- 学校や部活動の Web ページを企画して実際に作成する。


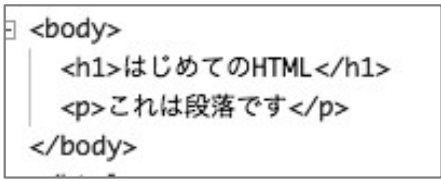

### 前提条件

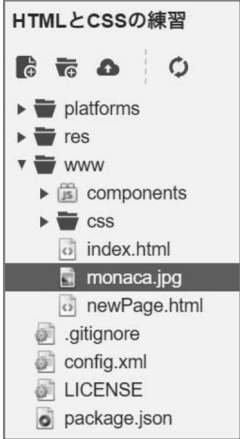
コンピュータの基本的な操作 (クリック, ドラッグ, 文字入力) ができる。  
ブラウザで Web サイトを利用することができる。

### 準備

Google Chrome のインストール、Monaca アカウントの作成、書籍の配付  
授業用サンプルプロジェクト : 第 2 章 HTML 入門 (ひな型)

指導過程 第2章：はじめてのHTML

指導過程	教師の働きかけ	備考	書籍	スライド
学習課題の把握	<ul style="list-style-type: none"> <li>● 自分たちが使う Web ページにはどのような要素(文字や画像等)があるか考えさせる。</li> <li>● 「HTMLを使用し最初のページを作成しよう」</li> </ul>	<ul style="list-style-type: none"> <li>● Web ページでは、文字だけでなく、画像・動画・音声を扱うことができる(情報のデジタル化の学習内容と関連づけることも可能)。</li> <li>● リンクによって他のWebページへ遷移できる。</li> <li>● HTMLはWebページだけでなく、アプリの作成にも使えることを示す。</li> </ul>	20	-
HTML 概要の学習	<ul style="list-style-type: none"> <li>● 書籍・スライドを使用し、HTMLの概要を説明する。</li> <li>● 自校のWebページ等を使用し、ソースコードを見せる。</li> </ul>	<ul style="list-style-type: none"> <li>● 「HTMLという言葉聞いたことあるか」、「HTMLについて知っていることはあるか」、「HTMLを見たことはあるか」、「HTMLを作ることができると思うか」等の発問・質問で意欲を喚起する。</li> </ul>	19 -26	
開発環境でプロジェクトを開く	<ul style="list-style-type: none"> <li>● Monacaのサイトにアクセスし、配付したユーザー名とパスワードを使ってログインさせる。</li> <li>● MonacaEducation 教材紹介ページ (<a href="https://edu.monaca.io/wakayama">https://edu.monaca.io/wakayama</a>)内の第二章 ひな形 から開き、プロジェクトをインポートさせる。</li> </ul> 	<ul style="list-style-type: none"> <li>● プロジェクト内のフォルダ構造については生徒の理解度に応じて適宜解説する。</li> </ul>		
HTML の編集	<ul style="list-style-type: none"> <li>● &lt;body&gt;内に次のHTMLタグ及びテキストを入力して保存 (Ctrl+s)し、プレビュー画面に表示されることを確認する。</li> </ul> 	<ul style="list-style-type: none"> <li>● タグを使用せず、テキストのみを入力して保存し、表示させると、結果の違いを確認できる。</li> <li>● 生徒の理解度に応じて、使用するタグの種類を調整する。</li> <li>● タグは半角で入力するよう指示。(全角で入力すると、そのまま表示される)</li> </ul>	20 -25	
リンクの作成を通じ、タグの種類や属性の仕組みを学ぶ	<ul style="list-style-type: none"> <li>● HTMLの主要なタグの用途や属性の仕組みを説明する。</li> <li>● リンクタグ&lt;a&gt;&lt;/a&gt;を通じてパスやURLの仕組み、属性の書き方を指導する。</li> </ul> 	<ul style="list-style-type: none"> <li>● URLについて復習する。</li> <li>● 属性値はダブルクォートかで囲むかシングルクォートで囲むことに注意を促す。</li> <li>● プレビュー画面での『外部サイト』の表示は、ブラウザの制限があるため事前に表示が可能なサイトかをチェックしておく。</li> </ul>	27 -31	31 -34

<p>画像を表示する</p>	<ul style="list-style-type: none"> <li>● プロジェクトの中に画像ファイルが準備されていることを確認させる。</li> </ul>  <pre data-bbox="328 734 754 864"> 13 &lt;body&gt; 14 &lt;img src="monaca.jpg"&gt; 15 &lt;/body&gt; </pre> <ul style="list-style-type: none"> <li>● <code>img</code> タグ <code>&lt;img&gt;</code> と属性を記述させ、画像が表示されることを確認させる。</li> </ul>	<ul style="list-style-type: none"> <li>● 左のプロジェクトパネルでファイルの管理を行う。画像のアイコンをダブルクリックすると画像の詳細を確認できる。またフォルダ左部の三角をクリックすることでフォルダの開閉を行える。</li> <li>● <code>src</code> 属性の値に外部サイトの画像も指定できることを説明する。</li> </ul>	<p>32 -33</p>	<p>43 -47</p>
<p>【発展1】オリジナルページの作成</p>	<ul style="list-style-type: none"> <li>● 書籍で紹介されているタグを活用してオリジナルのページ作成を促す。</li> </ul>	<ul style="list-style-type: none"> <li>● 自己紹介ページなど、何かを紹介するページが考えられる。また、複数のHTML ファイルをリンクさせることにより図鑑アプリで学習を深めるのも良い。</li> </ul>	<p>-</p>	<p>-</p>
<p>まとめ</p>	<ul style="list-style-type: none"> <li>● HTML を用いて Web ページを作成し、公開することで、情報発信できることや、パスとリンクの概念を確認し、Web ページの仕組みを理解することで次回以降の活動への意欲を喚起する。</li> </ul>		<p>-</p>	<p>-</p>

## 第 3 章 : はじめての CSS

### 学習目標

(知識及び技能)

- Web ページの画面を装飾するためには CSS が使用できることを理解する。
- 基本的なセレクタやプロパティを使用し CSS を編集することができる。

(思考力, 判断力, 表現力)

- 自分の意図する画面の装飾をするために、適切なセレクタやプロパティを設定することができる。
- Web ページの利用者の立場に立って、より使いやすいものにするためには、どのような作りがよいか考えられる。

(学びに向かう力, 人間性等)

- 自分の考えた画面の装飾をより使いやすいものにするため、試行錯誤しながら自主的に改良しようとする。

### 前提条件

コンピュータの基本的な操作（クリック、ドラッグ、文字入力）ができる。  
ブラウザで Web サイトを利用することができる。  
HTML の仕組みや記述方法を体験している。

### 準備

Google Chrome のインストール、Monaca アカウントの作成、書籍の配付  
授業用サンプルプロジェクト：第 2 章 HTML 入門（ひな型）の継続使用  
または、授業用サンプルプロジェクト：第 2 章 HTML 入門（完成版）



指導過程 第3章：はじめてのCSS

指導過程	教師の働きかけ	備考	書籍	スライド
学習課題の把握	<ul style="list-style-type: none"> <li>自分たちが普段目にする Web ページにはどのような装飾が施されているか考えさせる。</li> <li>「CSS を使用し、前の授業で作成したページを装飾しよう」</li> </ul>	<ul style="list-style-type: none"> <li>CSS による装飾の設定が文字色や背景色、ボタンや表の枠線などに適用されていることをヒントとして示す。</li> </ul>	-	-
簡単な CSS を記述する	<ul style="list-style-type: none"> <li>学校の Web ページなどの CSS ソースコードを示し、Web ページの装飾が CSS で記述されていることを示しつつ、CSS の記述方法を指導する。</li> <li>前回のプロジェクトを開き、プロジェクトパネルの CSS ファイルを編集させる。</li> </ul> 	<ul style="list-style-type: none"> <li>プロジェクトパネルを使ってフォルダ内のファイルを開く方法を改めて説明する。</li> </ul>	36 -37	49 -54
CSS セレクタやプロパティの仕組みを学ぶ	<ul style="list-style-type: none"> <li>CSS 記述の下記 3 要素を説明する。 <ol style="list-style-type: none"> <li>装飾する対象を決めるセレクタがある。</li> <li>①のサイズや色、余白等の様々な設定ができる。(プロパティという)</li> <li>プロパティはピクセル数、カラーコード等の値で指定する。</li> </ol> </li> </ul> 	<ul style="list-style-type: none"> <li>IDセレクタやCSSセレクタに『#』や『.』が付くのは、コンピュータがタグ名と区別するためであることを説明する。</li> <li>各種のサイズ、色、余白等をプロパティで指定する事が可能。</li> <li>色の指定を書籍ではカラーネーム主体で行っているが、カラーコードを使って16進数を教えることもできる。</li> <li>1つのセレクタに対して{から}の間で複数のプロパティ：値の組み合わせを指定することも可能。</li> </ul>	38, 44 -46	55 -56, 64 -68
文字列や画像要素にCSSを適用する	<ul style="list-style-type: none"> <li>書籍の実習に従ってCSSを適用させる。</li> </ul> 	<ul style="list-style-type: none"> <li>CSS のプロパティの値を変更して入力し、学習を深める（極端に大きな文字サイズなどを入れると楽しみながら学習できる）。</li> <li>早く終えた生徒にはその他のプロパティの値を変更するといった試行をさせると良い。</li> <li>適宜、タブレットで表示を確認する。</li> </ul>	39 -43, 46 -47	57 -63, 69 -71
【発展1】オリジナルページの装飾	<ul style="list-style-type: none"> <li>オリジナルのページを作成して装飾を施すように促す。見出しの h1 タグなどに背景色を設定し、重要な情報を&lt;strong&gt;タグで囲んだ上でCSSによる強調のための装飾を施すことなどを検討させる。</li> </ul>	<ul style="list-style-type: none"> <li>画像に width プロパティを適用すると、画像のサイズをリサイズできるのでページを作りやすくなる。</li> <li>CSS プロパティや HTML のタグを調べるリファレンスサイト例 『http://www.htmq.com/』</li> </ul>	-	-

まとめ	<ul style="list-style-type: none"> <li>● HTMLにCSSで装飾を行うときには、タグの名前だけでなくHTMLのid属性やclass属性でも指定できることを確認する。</li> <li>● HTMLやCSSを活用することで情報を視覚的に分かりやすく発信できることを理解し、ユーザビリティの視点を持たせるなど、情報リテラシーの習得や、情報技術の学習への意欲を喚起する。</li> </ul>		-	-
-----	---	--	---	---

## 留意事項

### CSS について

- 以前はHTMLによって装飾を行っていたが、CSSの登場によってデータと装飾を分離することが可能となり、情報を検索エンジン等のプログラムで扱いやすくなった。
- CSSも改良が進み、CSS3以降は角丸やグラデーション・簡単なアニメーション表現なども行えるようになった。

### CSS での色指定について

- 色の表現方法は、「red」や「blue」などの色の名称(カラーネーム)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

# ff 00 ff

赤の強さ 緑の強さ 青の強さ

## 第 4 章 : JavaScript 入門

### 学習目標

(知識及び技能)

- Web ページを動的なものにするためには JavaScript を使用することを理解する。
- 簡単な JavaScript を記述することが出来る (変数・Date)。

(思考力, 判断力, 表現力)

- 自分の意図した処理を JavaScript によって実行させることができる。
- 複雑な計算や処理などをプログラムすることで自動化できることを理解し、その利用価値について考えることができる。

(学びに向かう力, 人間性等)

- 自分の意図する処理を実行するために試行錯誤しながら継続的に改良を行う。

### 前提条件

コンピュータの基本的な操作 (クリック, ドラッグ, 文字入力) ができる。


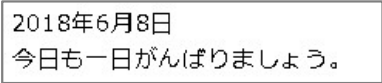
ブラウザで Web サイトを利用することができる。

HTML の仕組みや記述方法を体験している。

### 準備

Google Chrome のインストール、Monaca アカウントの作成、書籍の配付  
授業用サンプルプロジェクト : Monaca 標準の『最小限のテンプレート』

指導過程 第4章：JavaScript 入門

指導過程	教師の働きかけ	備考	書籍	スライド
学習課題の把握	<ul style="list-style-type: none"> <li>自分たちが普段使う Web ページにはどのようなプログラムが備わっているか考えさせる。</li> <li>JavaScript を学んで動的な Web ページや Web アプリとして動作するプログラムを作成しよう。</li> </ul>	<ul style="list-style-type: none"> <li>Web ページや Web アプリのプログラムにはブラウザ上で動くものと、サーバ側で動くものの 2 種類が存在する。JavaScript はどちらのタイプも対応できるが、この授業ではブラウザ上で動作するプログラムについて学習することを説明する。</li> </ul>	-	-
プログラムの記述方法を学ぶ	<ul style="list-style-type: none"> <li>ソースコードの記述で必要となる各記号の役割を説明する。</li> <li>ソースコードにインデントを施したりコメントを付けることの必要性を考えさせる。</li> </ul> 	<ul style="list-style-type: none"> <li>インデントをつける習慣を身につけると、あとで登場する条件分岐や繰り返しの授業が進めやすい。</li> <li>コメントを一行毎に書く必要は無い。全体の説明の他、プログラムの記述では説明できないことなどをメモするために書くとうわかりやすい。</li> </ul>	50 -52	73 -78
変数の概念を理解する	<ul style="list-style-type: none"> <li>プログラムで計算などを行うときには、データを一時的に記憶する必要があることを説明する。</li> <li>変数を用意することで数値や文字列を保持し、プログラム中で繰り返し利用できることを理解させる。</li> </ul>	<ul style="list-style-type: none"> <li>書籍のサンプル通りにプログラムが動いたら、変数名や代入する値を変えて試行することで学習を深める。</li> <li>変数に格納した値を複数回参照したり、別の値を代入して変数の動きを確認させると良い。</li> </ul>	53 -55	79 -82
今日の日付を画面に出力する	<ul style="list-style-type: none"> <li>プログラミング言語にはあらかじめ便利な命令が用意されていることを説明する。</li> <li>日付を扱う『Date』を通じて、命令の呼び出し方を理解させる。</li> </ul> 	<ul style="list-style-type: none"> <li>書籍ではオブジェクトという単語はあえて避けているが、『new』という記述で『Date』から日付オブジェクトを作成している。</li> <li>オブジェクトは『値』や『関数』を複数格納できる。用意されているオブジェクトを使うだけなら簡単であるが、オブジェクトを一から作る方法を教えるのは 11 章の配列を学習した後のほうが容易である。</li> </ul>	56 -60	83 -87
【発展 1】日付の応用	<ul style="list-style-type: none"> <li>『Date』の機能を調べて、時分秒も出力できることを確認する。</li> <li>『Date』以外の便利な命令を調べて動作させてみる。</li> <li>日付情報を任意の箇所に表示させるにはどうしたらよいかを検討させる。</li> </ul>	<ul style="list-style-type: none"> <li>Date 以外の便利な命令には、Math など存在する。Math が持つ命令は表計算ソフトの関数にあるようなもの(四捨五入の round 等)が多いので教えやすい。</li> <li>日付を任意の箇所に表示するためには 8 章の DOM という技術が必要になる。</li> </ul>	-	-
まとめ	<ul style="list-style-type: none"> <li>プログラミング言語を使うことで Web ページ内で動的に文字を出力できることを確認する。</li> <li>プログラムで情報の処理を行えることを踏まえ、プログラミング学習への意欲を喚起する。</li> </ul>	<ul style="list-style-type: none"> <li>今回学んだ変数の応用として、利用者に情報を入力させて情報を処理するような例を話すと分かりやすい。具体的には身長と体重を入力させて BMI 値を結果として返すプログラムなど。</li> <li>利用者に情報を入力させる方法としては、9 章のフォームや prompt() という命令が利用できる。</li> </ul>	-	-

## 第 5 章 : 条件分岐

### 学習目標

(知識及び技能)

- プログラムの実行を、指定した条件によって振り分けることができる。
- 比較演算子による真偽の判定をすることができる。

(思考力, 判断力, 表現力)

- 日常生活において条件の判定によって異なる結果を求める事例をあげることができる。
- 日常生活における条件の判定によって異なる結果を求める事例を比較演算子による真偽の判定に置き換え、プログラム化することができる。

(学びに向かう力, 人間性等)

- 複雑な条件をプログラムで処理する方法を検討する。
- 複数の分岐や変数を活用したプログラムの作成に積極的に取り組もうとする。

### 前提条件

コンピュータの基本的な操作（クリック、ドラッグ、文字入力）ができる。

ブラウザで Web サイトを利用することができる。

HTML の仕組みや記述方法を体験している。

変数の仕組みを理解している。

Date オブジェクトの概要を理解している。

### 準備

Google Chrome のインストール、Monaca アカウントの作成、書籍の配付

授業用サンプルプロジェクト : Monaca 標準の『最小限のテンプレート』をもとに、第 4 章で作成したプロジェクトの継続使用

または、授業用サンプルプロジェクト : 第 4 章 JavaScript 入門 (完成版)

指導過程 第5章：条件分岐

指導過程	教師の働きかけ	備考	書籍	スライド																												
学習課題の把握	<ul style="list-style-type: none"> <li>条件によって処理を振り分けるプログラムのケースを考えさせる。</li> <li>曜日毎に異なる文字を表示するプログラムを作成することを説明する。</li> </ul>	<ul style="list-style-type: none"> <li>書籍で曜日による分岐を扱っているため、曜日や時刻を例にすると実習させやすい。学校生活を例とした場合、朝は学校に行く・土曜日と日曜日は学校が休みなので行かない・しかし部活がある場合は学校に行くなどが考えられる。</li> <li>フローチャートを提示し、図解することが望ましい。</li> </ul>	-	-																												
if 文の記述法を学ぶ	<ul style="list-style-type: none"> <li>条件分岐の基本となるif文とelse文の文法と役割を説明する。</li> <li>条件式の書き方を説明する。</li> </ul> <p>比較演算子の種類</p> <table border="1"> <thead> <tr> <th>演算子</th> <th>概要</th> <th>条件式の例</th> <th>結果</th> </tr> </thead> <tbody> <tr> <td>==</td> <td>左辺と右辺が等しい場合は正しい</td> <td>1 == 1</td> <td>正しい</td> </tr> <tr> <td>!=</td> <td>左辺と右辺が等しくない場合は正しい</td> <td>1 != 2</td> <td>正しい</td> </tr> <tr> <td>&lt;</td> <td>左辺が右辺より小さい場合は正しい</td> <td>1 &lt; 1</td> <td>正しくない</td> </tr> <tr> <td>&lt;=</td> <td>左辺が右辺以下の場合は正しい</td> <td>1 &lt;= 1</td> <td>正しい</td> </tr> <tr> <td>&gt;</td> <td>左辺が右辺より大きい場合は正しい</td> <td>1 &gt; 1</td> <td>正しくない</td> </tr> <tr> <td>&gt;=</td> <td>左辺が右辺以上の場合は正しい</td> <td>1 &gt;= 1</td> <td>正しい</td> </tr> </tbody> </table>	演算子	概要	条件式の例	結果	==	左辺と右辺が等しい場合は正しい	1 == 1	正しい	!=	左辺と右辺が等しくない場合は正しい	1 != 2	正しい	<	左辺が右辺より小さい場合は正しい	1 < 1	正しくない	<=	左辺が右辺以下の場合は正しい	1 <= 1	正しい	>	左辺が右辺より大きい場合は正しい	1 > 1	正しくない	>=	左辺が右辺以上の場合は正しい	1 >= 1	正しい	<ul style="list-style-type: none"> <li>if 文の条件を満たさないときの処理がある場合は else を使う。さらに、別の条件による分岐がある場合には else if を使う。</li> <li>比較は『==』だが代入は『=』なので間違えやすく、また文法エラーでもないので誤りに気づきづらく、注意が必要である。</li> <li>『&gt;』や『&lt;』の活用例として、日付が月の上旬・中旬・下旬のどれであるかを振り分けるプログラムが考えられる。</li> </ul>	62 -63	89 -92
演算子	概要	条件式の例	結果																													
==	左辺と右辺が等しい場合は正しい	1 == 1	正しい																													
!=	左辺と右辺が等しくない場合は正しい	1 != 2	正しい																													
<	左辺が右辺より小さい場合は正しい	1 < 1	正しくない																													
<=	左辺が右辺以下の場合は正しい	1 <= 1	正しい																													
>	左辺が右辺より大きい場合は正しい	1 > 1	正しくない																													
>=	左辺が右辺以上の場合は正しい	1 >= 1	正しい																													
曜日チェックプログラムの作成	<ul style="list-style-type: none"> <li>前回のプロジェクトを編集し、if 文を追記して今日が日曜日かどうかをチェックするプログラムを記述させる。</li> </ul>	<ul style="list-style-type: none"> <li>if 文ではインデントが重要なので繰り返し指導を行う。</li> <li>例では日曜日を扱っているが、別の曜日でもよい。</li> </ul>	63 -65	93 -95																												
曜日別メッセージプログラムの作成	<ul style="list-style-type: none"> <li>else if 文の文法と役割を説明する。</li> <li>前回のプロジェクトを編集し、else if 文を追記して曜日毎に異なるメッセージを出力するプログラムを記述する。</li> </ul> 	<ul style="list-style-type: none"> <li>JavaScript で else if を記述する場合は間に空白スペースが必要なので注意する。</li> <li>Date の曜日番号を元に今日の曜日を日本語で出力するようなプログラムは配列でも実現できる。11 章に配列で実現したプログラムを掲載している。</li> </ul>	66 -68	96 -100																												
【発展1】オリジナル条件分岐の実装	<ul style="list-style-type: none"> <li>サンプルを応用して、授業開始時に検討したような条件振り分けプログラムを作成する。</li> <li>if 文をネスト（入れ子）にして複雑な条件を扱う。</li> </ul>	<ul style="list-style-type: none"> <li>『&gt;』や『&lt;』を活用したサンプルを提示することで、応用の幅を広げることができる。</li> <li>コンピュータは深いネストでも処理できるが、可読性が下がるので 3 階層程度に留めるのが一般的。</li> </ul>	-	-																												
まとめ	<ul style="list-style-type: none"> <li>if 文による条件分岐と比較演算子の仕組みを確認する。</li> <li>日常生活の至る所で条件分岐が行われていることを理解し、プログラミングを身近なものに感じさせる。</li> </ul>	<ul style="list-style-type: none"> <li>『条件分岐』はプログラミングの中で最も重要な基礎項目の一つであるため、重点的に復習するように指導する。</li> <li>同様に重要な基礎項目である『繰り返し』や『配列』を学習する 10 章までに完全な理解を目指す。</li> <li>『&gt;』や『&lt;』を活用したサンプルを提示することで、応用の幅を広げることができる。</li> </ul>	-	-																												

## 第 6 章：関数

### 学習目標

(知識及び技能)

- 複数行にわたるプログラムを一つの関数にまとめる方法を学ぶ。
- 関数に引数や戻り値を設定できることを理解する。

(思考力, 判断力, 表現力)

- プログラムの一部を関数化する意義を考えることができる。
- これまでに作成したプログラムの一部を関数化するべきかを検討・試行することができる。
- 一つの関数に複数の引数を持たせたり、戻り値で複雑な値を扱うなどの方法を検討することができる。

(学びに向かう力, 人間性等)

- 便利な関数を作成し、グループやクラスのメンバーで共有しようとする。

### 前提条件

コンピュータの基本的な操作（クリック、ドラッグ、文字入力）ができる。

ブラウザで Web サイトを利用することができる。

HTML の仕組みや記述方法を体験している。

変数の仕組みを理解している。

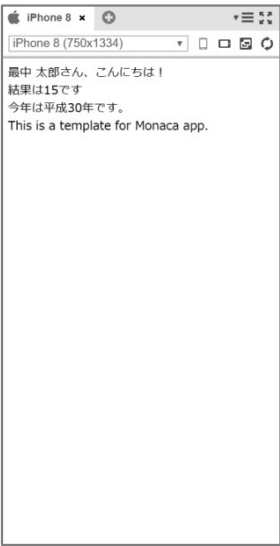
### 準備

Google Chrome のインストール、Monaca アカウントの作成、書籍の配付  
授業用サンプルプロジェクト：Monaca 標準の『最小限のテンプレート』

指導過程 第6章：関数

指導過程	教師の働きかけ	備考	書籍	スライド
学習課題の把握	<ul style="list-style-type: none"> <li>● 複数行にわたるプログラムを一つの関数にまとめる必要性を説明する。</li> <li>● 例えば西暦→和暦変換について、「今年は何年か?」「2000年は何年?」「1964年は何年?」「次の元号に変わったらどうする?」等の投げかけをし、「よく使う一連の処理をまとめることのメリット」を考えさせる。</li> <li>● さらに、そのようなメリットを持つ処理は、「具体的にどのようなものが考えられるか」を検討させる。</li> </ul>	<ul style="list-style-type: none"> <li>● 書籍第6章の冒頭にあるとおり、長いプログラムでは関数化による可読性の向上が重要となる。</li> <li>● 関数化して便利なものの例として、曜日を日本語で返す関数が考えられる。</li> <li>● また、消費税の計算関数やBMI値の計算関数、生年月日→年齢/干支/星座/学年の計算関数などが考えられる。</li> </ul>	69	-
関数の記述法を学ぶ①	<ul style="list-style-type: none"> <li>● function による関数定義の文法を示し、サンプルプログラムを記述させる。 下図：引数も戻り値も無い簡単な例</li> </ul> <pre> &lt;script&gt;   function sayHello() {     document.writeln("こんにちは!");   }    sayHello(); &lt;/script&gt; </pre>	<ul style="list-style-type: none"> <li>● 関数は引数と戻り値を使うことができるが、省略も可能である。書籍では、引数と戻り値のない簡単な関数から一つずつ学習していく。</li> <li>● 関数は一度作成すれば何度でも呼び出せることを確認する。また、それによって、効率よくプログラムを作成できることを説明する。</li> </ul>	70 -73	102 -107
関数の記述法を学ぶ②	<ul style="list-style-type: none"> <li>● 単純に「こんにちは」と表示するだけでなく、名前も表示させる方がより使い勝手の良い関数になりそうだと示唆する。</li> <li>● 引数のある関数定義の文法を教え、サンプルプログラムを記述させる。 下図：引数が1つある例</li> </ul> <pre> &lt;link rel="stylesheet" href="css/style.css"&gt; &lt;script&gt;   function sayHello(name) {     document.writeln(name + "さん、こんにちは!");   }    sayHello("山本");   sayHello("田中"); &lt;/script&gt; </pre>	<ul style="list-style-type: none"> <li>● 書籍の例では引数は一つだが、複数の引数を受け取る関数も定義できる。また、引数に渡せる情報は文字列や数値だけでなく、配列やオブジェクトも渡すことができる。</li> </ul>	74 -75	108 -112
関数の記述法を学ぶ③	<ul style="list-style-type: none"> <li>● 戻り値のある関数定義の文法を教え、サンプルプログラムを記述させる。 下図：戻り値がある例</li> </ul> <pre> &lt;script&gt;   function calc() {     var num = 1 + 2 + 3 + 4 + 5;     return num;   }    var result = calc();   document.writeln("結果は" + result + "です。"); &lt;/script&gt; </pre>	<ul style="list-style-type: none"> <li>● 戻り値は一つしか返せない。例えば関数の中で if 文を記述して2通りの return を設定することはできるが、戻り値で返せる値は一つだけである。ただし、戻す値を配列やオブジェクトにすることは可能である。</li> <li>● 生徒の理解度が高ければ、引数と戻り値両方がある例も可(消費税計算等)。</li> </ul>	76 -78	113 -117



<p>西暦・和暦変換プログラムの作成</p>	<ul style="list-style-type: none"> <li>● 西暦を和暦に変換するプログラムを作成する。</li> </ul> 	<ul style="list-style-type: none"> <li>● 西暦・和暦変換プログラムの応用例として、江戸時代から数えるプログラムなども考えられる。</li> </ul>	<p>79 -80</p>	<p>118 -119</p>
<p>【発展1】オリジナル関数の実装</p>	<ul style="list-style-type: none"> <li>● 授業開始時に検討したようなメリットを持つ処理のプログラムを実際に作成する。</li> </ul>	<ul style="list-style-type: none"> <li>● BMI 値の計算は 10 章で扱っているので、予習を兼ねてここで検討できると理解に繋げやすい。</li> </ul>	<p>-</p>	<p>-</p>
<p>まとめ</p>	<ul style="list-style-type: none"> <li>● 前回まで利用した日付オブジェクトの命令群も関数の仲間であることを説明する。</li> <li>● 用意された関数を組み合わせることで効率的にプログラミングできることを確認する。</li> <li>● その他に関数として用意されていたら便利なものを検討させる。</li> </ul>	<ul style="list-style-type: none"> <li>● 実際のシステム開発でも用意された関数などを組み合わせる作業はとても多い。</li> <li>● アルゴリズムを考える力と、用意されたアルゴリズムを組み合わせる力のどちらも必要な力である。</li> <li>● オブジェクトが持つ命令群はメソッドと呼ぶが、実質は関数である。</li> </ul>	<p>-</p>	<p>-</p>

## 第7章：イベント

### 学習目標

(知識及び技能)

- プログラムを任意の機会で行う方法を学ぶ。
- イベントの概念を理解する。

(思考力, 判断力, 表現力)

- 日常生活で普段利用している機器やソフトウェアがどのような機会で行っているか、イベントとプログラムの実行を関連づけて考えることができる。

(学びに向かう力, 人間性等)

- 特定のイベントに応じて実行されるプログラムの例を説明し、グループやクラスのメンバーと共有する。

### 前提条件

コンピュータの基本的な操作（クリック、ドラッグ、文字入力）ができる。

ブラウザで Web サイトを利用することができる。

HTML の仕組みや記述方法を体験している。

変数の仕組みを理解している。

### 準備

Google Chrome のインストール、Monaca アカウントの作成、書籍の配付

授業用サンプルプロジェクト：第7章 イベント（ひな形）

指導過程 第7章：イベント

指導過程	教師の働きかけ	備考	書籍	スライド												
学習課題の把握	<ul style="list-style-type: none"> <li>これまで作ってきた静的な画面とスマホ等で普段使う動的なアプリの違いを示す(タブレットで何らかのボタンを押すと動作する様子を示す等)。</li> <li>イベントの概念を理解させる。</li> <li>ブラウザ上での発生が想定できるイベント群を検討させる。</li> </ul>	<ul style="list-style-type: none"> <li>アプリを例示する際は、ボタンを押すと即座に反応があるものがわかりやすい。</li> <li>パソコンであればキーボードの入力がイベントとして取得できる、また、タブレット等であれば『タッチ』もイベントとして取得できる。</li> <li>上記イベントを応用すればタイピングゲームや指でなぞるお絵かきアプリなども作成できる。</li> </ul>	-	-												
イベントの種類を学ぶ	<ul style="list-style-type: none"> <li>イベントの種類を学び、クリックイベントで関数を呼び出すサンプルプログラムを記述させる。</li> </ul> <table border="1" data-bbox="325 757 1166 1059"> <thead> <tr> <th>イベント名</th> <th>概要</th> </tr> </thead> <tbody> <tr> <td>click</td> <td>HTML 要素がクリック、またはタップされたタイミングで発生します。</td> </tr> <tr> <td>load</td> <td>最初にページが表示されたタイミングで発生します。このイベントは &lt;body&gt; 要素に対して付与します。</td> </tr> <tr> <td>touchstart</td> <td>モバイル端末の画面に指が触れたタイミングで発生します。</td> </tr> <tr> <td>touchmove</td> <td>モバイル端末の画面に指を触れたまま上下左右に動かしている間、複数回発生します。</td> </tr> <tr> <td>touchend</td> <td>モバイル端末の画面から指を話したタイミングで発生します。</td> </tr> </tbody> </table>	イベント名	概要	click	HTML 要素がクリック、またはタップされたタイミングで発生します。	load	最初にページが表示されたタイミングで発生します。このイベントは <body> 要素に対して付与します。	touchstart	モバイル端末の画面に指が触れたタイミングで発生します。	touchmove	モバイル端末の画面に指を触れたまま上下左右に動かしている間、複数回発生します。	touchend	モバイル端末の画面から指を話したタイミングで発生します。	<ul style="list-style-type: none"> <li>タブレットでボタンをタップした場合もクリックイベントが発生する。</li> </ul>	82 -84	121 -123
イベント名	概要															
click	HTML 要素がクリック、またはタップされたタイミングで発生します。															
load	最初にページが表示されたタイミングで発生します。このイベントは <body> 要素に対して付与します。															
touchstart	モバイル端末の画面に指が触れたタイミングで発生します。															
touchmove	モバイル端末の画面に指を触れたまま上下左右に動かしている間、複数回発生します。															
touchend	モバイル端末の画面から指を話したタイミングで発生します。															
load イベントを学ぶ	<ul style="list-style-type: none"> <li>load イベントを説明し、サンプルプログラムを記述させる。</li> </ul> <pre data-bbox="339 1189 719 1223">&lt;body onload="loadMessage()"&gt;</pre>	<ul style="list-style-type: none"> <li>load イベントは8章のDOMでも使われており、応用的なアプリ作成を進めていく上では重要なイベントである。</li> </ul>	84 -85	124 -126												
【発展 1】複数の関数とイベントを扱う	<ul style="list-style-type: none"> <li>複数の関数とボタンを用意し、押されたボタンによって異なる関数が呼び出されるようなプログラムを作成する。</li> </ul>		-	-												
まとめ	<ul style="list-style-type: none"> <li>イベントの概念を用いることで、プログラムの実行を制御できることを確認する。</li> <li>普段の生活の中でもイベントに応じて動作する機器やプログラムがないかを検討させる。</li> </ul>	<ul style="list-style-type: none"> <li>人の動きでイベントを発生させて動作するシステムの例として、自動ドアや自動改札機などが上げられる。</li> <li>押されたボタンの種類に応じて動作する自動販売機なども、イベントに応じて動作する機器の一つである。</li> </ul>	-	-												

## 第 8 章 : DOM

### 学習目標

(知識及び技能)

- HTML で定義した表示内容を動的に変更する方法を学ぶ。

(思考力, 判断力, 表現力)

- 身の回りのスマホアプリなどがどのような条件で表示内容を変更しているかを検討することができる。
- プログラムで CSS を切り替え、文字や背景の色などを目的に応じて変更することができる。
- イベントや条件分岐に応じた、適切な処理を考え、選択することができる。

(学びに向かう力, 人間性等)

- 「おみくじアプリ」などの既存のプログラムの設計や工夫を読み解く。

### 前提条件

コンピュータの基本的な操作（クリック、ドラッグ、文字入力）ができる。

ブラウザで Web サイトを利用することができる。

HTML の仕組みや記述方法を体験している。


変数の仕組みを理解している。

### 準備

Google Chrome のインストール、Monaca アカウントの作成、書籍の配付

授業用サンプルプロジェクト：第 8 章 DOM（ひな形）

指導過程 第8章：DOM

指導過程	教師の働きかけ	備考	書籍	スライド
学習課題の把握	<ul style="list-style-type: none"> <li>これまで静的な画面または単純な表示(alert())を行うプログラムを作成してきたが、実際に使うアプリと同様に、プログラムを動かしながら表示を変える必要性を示す。</li> <li>JavaScript のプログラムで HTML を書き換える必要性を説明する。</li> <li>DOM の概念を理解させる。</li> </ul>	<ul style="list-style-type: none"> <li>画面の変更が分かりやすいWebサイトの例として、地図サイトなどがある。</li> <li>動作の例示は、メッセージアプリで送信ボタンを押すとメッセージが自分の送信済みエリアに表示される等のシンプルで親しみのあるものがよい。</li> <li>JavaScript で HTML 要素を DOM を使って取得することで様々な編集を行える。</li> </ul>	88 -89	128 -130
DOM による表示内容の変更	<ul style="list-style-type: none"> <li>DOM 操作により HTML 内に文字表示を行うサンプルプログラムを記述させる。</li> </ul> <pre> &lt;script&gt; function greet() {   // 現在の時刻を取得   var date = new Date();   var hour = date.getHours();    if(hour &gt;= 5 &amp;&amp; hour &lt;= 10) {     document.getElementById("message").innerHTML = "おはよう！";   } else if(hour &gt;= 11 &amp;&amp; hour &lt;= 18) {     document.getElementById("message").innerHTML = "こんにちは！";   } else {     document.getElementById("message").innerHTML = "こんばんは！";   } } &lt;/script&gt; </pre>	<ul style="list-style-type: none"> <li>『getElementById』の element とは HTML 要素のことを指している。説明の仕方としては「DOM を使って、ID で指定された HTML 要素を取得している。」と説明できる。</li> <li>このプログラムは関数の復習と条件分岐の復習を兼ねているため、時間を掛けて取り組む必要がある。</li> </ul>	90 -93	131 -133
DOM による画像や背景の変更	<ul style="list-style-type: none"> <li>DOM 操作により HTML 中に画像表示や背景の変更を行うサンプルプログラムを記述させる。</li> </ul> 	<ul style="list-style-type: none"> <li>画像を扱う&lt;img&gt;タグは src 属性で画像の場所を参照している。src 属性の値を DOM で書き換えることで、表示する画像を変更できる。</li> </ul>	94 -97	134 -139
【発展 1】DOM 操作の理解を深める	<ul style="list-style-type: none"> <li>朝と昼と夜で背景色が変わるプログラムに変更させる。</li> <li>時刻を表示するように変更させる。</li> </ul>	<ul style="list-style-type: none"> <li>DOM を含めた 8 章までの技術があれば、動きのある実用的なプログラムを作成することができる。</li> <li>「おみくじアプリ」は、8 章までの技術で作られたサンプルアプリのため、必要に応じて活用する。</li> <li>11 章の配列と 12 章の繰り返しは難易度が高いため、そこに進むまでに理解度や関心を高めておきたい。</li> </ul>	-	-
まとめ	<ul style="list-style-type: none"> <li>Web ページや Web アプリの表示内容を動的に変更するときには DOM という技術が使われている。</li> <li>DOM を扱うためには、HTML タグや属性の理解が必要となる。</li> </ul>	<ul style="list-style-type: none"> <li>HTML タグや属性の種類についての復習を促す。</li> </ul>	-	-

## 第9章：フォーム

### 学習目標

(知識及び技能)

- 様々な入力フォームを用意し、任意の値をプログラムに渡す方法を学ぶ。

(思考力, 判断力, 表現力)

- 各フォーム部品の特性について検討し、入力される値を想定することができる。
- プログラムの目的や、入力される値の特性に応じたフォームを選択し、設計することができる。

(学びに向かう力, 人間性等)

- 普段利用している Web サイトやアプリではどのようなフォーム部品が使われているかを調査し、なぜそのフォームが使われているか検討しようとする。

### 前提条件

コンピュータの基本的な操作（クリック，ドラッグ，文字入力）ができる。

ブラウザで Web サイトを利用することができる。

HTML の仕組みや記述方法を体験している。

変数の仕組みを理解している。

### 準備

Google Chrome のインストール、Monaca アカウントの作成、書籍の配付

授業用サンプルプロジェクト：第9章 フォーム（ひな形）

指導過程 第9章：フォーム

指導過程	教師の働きかけ	備考	書籍	スライド
学習課題の把握	<ul style="list-style-type: none"> <li>SNS 等のアプリを実際に示し、利用者にとって使いやすいプログラムを作成するためには、データを入力するためのフォームが必要であることを示す。</li> <li>HTML でフォームを作成する方法を理解させる。</li> <li>フォーム部品の値をJavaScriptで取得する方法を理解させる。</li> </ul>	<ul style="list-style-type: none"> <li>HTML にはテキストデータや選択肢を入力するための汎用的なフォーム部品が一通り用意されており、それらを組み合わせることでアプリに必要な入力画面を作ることができる。</li> </ul>	-	-
フォーム部品の確認	<ul style="list-style-type: none"> <li>書籍で示されているフォーム部品の説明を行い、役割の違いを理解させる。</li> <li>普段利用しているWebサイトやアプリで用意されているフォーム部品の例を検討させる。</li> </ul> <div style="border: 1px solid black; padding: 5px; margin: 5px 0;"> <p>例 テキストボックスの記述例</p> <pre>&lt;input type="text" value="テスト"&gt;</pre> </div> <div style="border: 1px solid black; padding: 5px; margin: 5px 0;"> <p>例 ドロップダウンメニューの記述例</p> <pre>&lt;select&gt;   &lt;option value="1"&gt;AAA&lt;/option&gt;   &lt;option value="2"&gt;BBB&lt;/option&gt;   &lt;option value="3"&gt;CCC&lt;/option&gt;   &lt;option value="4"&gt;DDD&lt;/option&gt; &lt;/select&gt;</pre>  </div>	<ul style="list-style-type: none"> <li>1 行のテキストデータを入力する場合には&lt;input type="text"&gt;を利用する。書籍では触れていないが、複数行にわたる場合は&lt;textarea&gt;を利用できる。</li> <li>選択肢の中から選ぶ場合にはテキストデータを入力する必要が無いため、選択肢のフォーム部品を使う。</li> <li>複数の選択肢から一つだけ選ぶ場合は &lt;input type="radio"&gt; か &lt;select&gt;を使う。radio はワンアクションで選択肢を選べるため、使いやすいが、表示領域を取ってしまうため、選択肢が多い場合には select を利用の方がよい場合がある。</li> <li>例として、性別には radio、都道府県には select を使うことが一般的である。</li> <li>フォーム部品は利用者の使い勝手を考慮して選択する。</li> </ul>	100 -101	141 -145
フォームの値を取得する	<ul style="list-style-type: none"> <li>書籍の実習に従ってJavaScriptを記述し、フォームの値を取得する。</li> </ul> <div style="border: 1px solid black; padding: 5px; margin: 5px 0;"> <pre>// テキストボックスの入力値を取得 var name = document.getElementById("name").value;</pre> </div> <div style="border: 1px solid black; padding: 5px; margin: 5px 0;"> <pre>// ドロップダウンメニューのうち選択されている値を取得 var language = document.getElementById("languageList").value;</pre> </div>	<ul style="list-style-type: none"> <li>textの値を取得するときにはinput要素のvalue属性値を参照する。</li> <li>checkboxの値を取得するときにはinput要素のchecked属性値を参照する。checked属性値はtrue/falseの真偽値を返す。</li> <li>select(ドロップダウン)の値を取得するときにはselect要素のvalue属性値を参照する。なおselectはoption要素にvalue値が記載されているときにはvalueを返し、valueが値が記載されていないときには要素の内容を値として返す。</li> </ul>	102 -104	146 -147
【発展1】オリジナルフォームの作成	<ul style="list-style-type: none"> <li>Webサイトやアプリを想定して、特定の目的に沿った入力フォームを選択し、設計させる。</li> </ul>	<ul style="list-style-type: none"> <li>フォームの改造例として、学校生活に関するアンケートなどが考えられる。好きな教科を全て選ぶような設問であればcheckboxを利用できる。</li> </ul>	-	-
【発展2】オリジナルフォームの活用	<ul style="list-style-type: none"> <li>オリジナル入力フォームの値をJavaScriptで取得して活用させる。</li> </ul>	<ul style="list-style-type: none"> <li>フォームを活用したアプリの例として、10章では身長と体重を入力するフォームを用意してBMI値を計算する実習を用意している。この指導は10章の発展として引き続き設定することも可能である。</li> </ul>	-	-

まとめ	<ul style="list-style-type: none"> <li>フォームは情報デザインの種類であり、プログラムの目的は入力データの特性等に応じて、適切な部品を選ぶことでアプリの使いやすさにつながる。</li> </ul>	<ul style="list-style-type: none"> <li>すでに多くの開発者が作成した汎用的なフォーム部品が Web 上には多数存在しており、それらを取得し、組み合わせることで、より適切なフォームを比較的容易に作成することができる。</li> </ul>	-	-
-----	---	---	---	---

留意事項

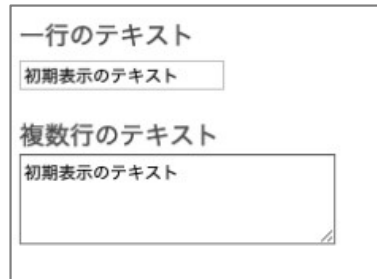
複数行テキストの入力フォーム

- 書籍で説明しているテキストボックスは一行しか入力できない。  
 textarea タグを使用することで複数行のテキストも扱うことができる。
  - 初期表示のテキストは value ではなく、開始タグと終了タグの間の文字列となる。
  - cols で幅(半角文字数)を rows で行数を指定することができる。

```

<p>
  一行のテキスト<br/>
  <input type="text" value="初期表示のテキスト">
</p>
<p>
  複数行のテキスト<br/>
  <textarea cols="30" rows="4">初期表示のテキスト</textarea>
</p>

```



ラジオボタンかセレクトボックスか

- ラジオボタンの方がセレクトボックスよりも一覧性が良く、ワンアクションで選べる。

高校の種類：  県立  市立  私立

高校の種類：    
 県立   
 市立   
 私立

- しかし、選択肢が多い場合は画面を大きく専有してしまい、ラジオボタンは好ましくない(下図)。

北海道  青森県  岩手県  宮城県  
 秋田県  山形県  福島県  茨城県  
 栃木県  群馬県  埼玉県  千葉県  
 東京都  神奈川県  新潟県  富山県  
 石川県  福井県  山梨県  長野県  
 岐阜県  静岡県  愛知県  三重県  
 滋賀県  京都府  大阪府  兵庫県  
 奈良県  和歌山県  鳥取県  島根県  
 岡山県  広島県  山口県  徳島県  
 香川県  愛媛県  高知県  福岡県  
 佐賀県  長崎県  熊本県  大分県  
 宮崎県  鹿児島県  沖縄県



## 第 10 章： いろいろな演算子

### 学習目標

(知識及び技能)

- 目的に応じた各種の演算子があることを学ぶ。
- プログラム内で演算子を利用して計算や処理をする方法を学ぶ。

(思考力, 判断力, 表現力)

- 数学などで学んだことのある演算子や数式との違いを比較しながら考えることができる。
- 目的に合わせた演算子を選択し、使用することができる。

(学びに向かう力, 人間性等)

- 端数処理などのさらに高度な計算処理をプログラム内で行う方法を考えようとする。
- サンプルアプリを応用し、課題解決につながるアプリを開発しようとする。

### 前提条件

コンピュータの基本的な操作（クリック、ドラッグ、文字入力）ができる。

ブラウザで Web サイトを利用することができる。

HTML の仕組みや記述方法を体験している。


変数の仕組みを理解している。

### 準備

Google Chrome のインストール、Monaca アカウントの作成、書籍の配付

授業用サンプルプロジェクト：第 10 章 いろいろな演算子（ひな形）

指導過程 第10章：いろいろな演算子

指導過程	教師の働きかけ	備考	書籍	スライド												
既習事項の確認、学習課題の把握	<ul style="list-style-type: none"> <li>第5章で学習した比較演算子、第6章で学習した単純な計算を確認する。</li> <li>既習の単純な計算だけでなく、複雑な計算（積み立て貯金の利息等）をコンピュータでどのように解決するかフローチャート等で示し興味を喚起する。</li> <li>演算子を用いることで様々な計算をプログラムで実行できることを理解させる。</li> <li>論理演算という考え方があることを理解させる。</li> </ul>	<ul style="list-style-type: none"> <li>ここでは演算子を用いた計算方法を扱う。なお、JavaScriptではMathという計算用のオブジェクトを使うことで乱数処理や端数処理などの演算を扱うことができる。難易度はDateオブジェクトよりも低いため、生徒の興味に合わせて紹介を行う。</li> </ul>	-	-												
演算子の種類を学習する	<ul style="list-style-type: none"> <li>四則演算子と論理演算子の説明を行う。また、複合代入演算子を使うことで代入を伴う四則演算を簡潔に記述できることを理解させる。</li> </ul> <table border="1" data-bbox="312 878 801 969"> <thead> <tr> <th>演算子</th> <th>概要</th> <th>使用例</th> <th>結果</th> </tr> </thead> <tbody> <tr> <td>&amp;&amp;</td> <td>2つの条件式がどちらもtrueの場合はtrue</td> <td>1 == 1 &amp;&amp; 2 == 2</td> <td>true</td> </tr> <tr> <td>  </td> <td>2つの条件式のどちらかがtrueの場合はtrue</td> <td>1 == 2    2 == 2</td> <td>true</td> </tr> </tbody> </table>	演算子	概要	使用例	結果	&&	2つの条件式がどちらもtrueの場合はtrue	1 == 1 && 2 == 2	true		2つの条件式のどちらかがtrueの場合はtrue	1 == 2    2 == 2	true	<ul style="list-style-type: none"> <li>四則演算子は理解しやすいが、複合代入演算子の『++』と『--』は生徒にとって馴染みが薄いので注意が必要。12章の繰り返し処理で丁寧な説明を行う。</li> <li>論理演算の用途として、実習で扱うBMI計算プログラムのサンプルコードを併用して説明すると分かりやすい。</li> </ul>	106 -107	149 -153
演算子	概要	使用例	結果													
&&	2つの条件式がどちらもtrueの場合はtrue	1 == 1 && 2 == 2	true													
	2つの条件式のどちらかがtrueの場合はtrue	1 == 2    2 == 2	true													
BMI計算プログラムの作成	<ul style="list-style-type: none"> <li>BMI値など、公式で計算できる値はプログラムで自動的に計算できることを理解させる。</li> <li>サンプルプログラムから書籍に従ってBMI計算プログラムを完成させる。</li> </ul> 	<ul style="list-style-type: none"> <li>if文による条件分岐で比較演算を利用したことに触れ、比較演算が返す結果は真偽値であることを改めて学習させる。真偽値同士を論理演算することで複雑な条件を一つの条件式で扱えることを学習させる。</li> </ul>	108 -112	154 -156												
【発展1】BMI計算プログラムの改良	<ul style="list-style-type: none"> <li>BMI値が低体重もしくは肥満となる場合に警告を出すための装飾や処理を検討する。</li> <li>BMI値の小数点以下の値を端数処理する方法を検討する。</li> <li>BMI値以外の計算プログラムを検討して作成する。</li> </ul>	<ul style="list-style-type: none"> <li>警告の装飾を行う方法として、8章で扱った背景色を変えるDOMの技術を活用できる。</li> <li>端数処理にはMath.floor()などの命令を使うことができる。</li> <li>計算プログラムの例として面積を求めるプログラムや消費税を求めるプログラムなどが考えられる。</li> </ul>	-	-												
まとめ	<ul style="list-style-type: none"> <li>フォームと演算子を組み合わせることで計算プログラムを作成することができることを理解する。</li> <li>アプリの中には、演算子や計算用の命令を組み合わせることで複雑な計算を処理していることを理解する。</li> </ul>		-	-												

## 留意事項

算術演算子よりも高度な計算は Math オブジェクトを使用する事により可能である。

- `Math.PI` : 円周率を返す。
- `Math.abs(x)` : 引数として与えた数の絶対値を返す。
- `Math.floor(x)` : 引数として与えた数以下の最大の整数を返す。
- `Math.ceil(x)` : 引数として与えた数以上の最小の整数を返す。
- `Math.random()` : 0 以上 1 未満の乱数を返す。
- `Math.round(x)` : 引数として与えた数を四捨五入して、近似の整数を返す。
- `Math.sin(x)` : 引数として与えた数のサイン (正弦) を返す。
- `Math.sqrt(x)` : 引数として与えた数の平方根を返す。

参照 : [https://developer.mozilla.org/ja/docs/Web/JavaScript/Reference/Global\\_Objects/Math](https://developer.mozilla.org/ja/docs/Web/JavaScript/Reference/Global_Objects/Math)

## 第 11 章 : 配列

### 学習目標

(知識及び技能)

- 扱うデータの数が多の場合、配列の利用が適していることを理解する。
- 配列を利用して複数の情報をまとめて扱う方法を学ぶ。

(思考力, 判断力, 表現力)

- プログラムをトレースし、配列で保持する正しい値を導き出すことができる。
- 配列の特性を生かし、活用できる場面と方法を検討することができる。

(学びに向かう力, 人間性等)

- 座標などの複数の値で構成されるデータを配列に格納するにはどうしたら良いかを検討しようとする。

### 前提条件

コンピュータの基本的な操作 (クリック, ドラッグ, 文字入力) ができる。

ブラウザで Web サイトを利用することができる。

HTML の仕組みや記述方法を体験している。

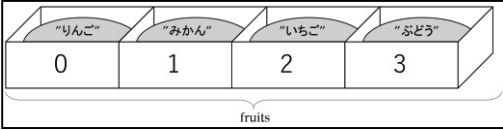
変数の仕組みを理解している。

### 準備

Google Chrome のインストール、Monaca アカウントの作成、書籍の配付

授業用サンプルプロジェクト : 第 11 章 配列 (ひな形)

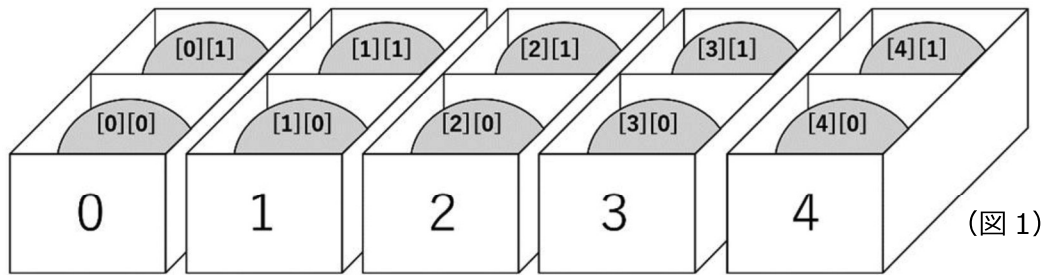
指導過程 第 11 章：配列

指導過程	教師の働きかけ	備考	書籍	スライド
既習事項の確認、学習課題の把握	<ul style="list-style-type: none"> <li>● 配列を用いることで複数の値を一つにまとめられることを理解させる。</li> <li>● これまで扱ってきた曜日のようなデータも、配列を使うことで簡潔に表現できることを理解させる。</li> <li>● 曜日を出力する場合、if 文を用いるよりも配列を利用するほうが簡潔にプログラムが記述できることを示す。</li> </ul> 	<ul style="list-style-type: none"> <li>● 配列を何かに例えるなら、タンスや本棚などが考えられる。</li> <li>● Date オブジェクトでは曜日情報が 0～6 の数字として管理されており、配列と組み合わせやすい。コンピュータで情報を扱うときには、このように情報がコード化されていると処理や容量の面で都合が良い。</li> </ul>	115 -116	-
配列の概念を学習する	<ul style="list-style-type: none"> <li>● 配列の記述方法を説明する。</li> <li>● 配列に格納した値を参照する方法を説明する。</li> </ul> <pre>var fruits = [ "りんご", "みかん", "いちご", "ぶどう" ]; alert(fruits[2]);</pre>	<ul style="list-style-type: none"> <li>● 配列のインデックスは 0 から 1 ずつ増える整数で与えられるが、length プロパティは要素数を返すので注意を促す。例えば配列の要素数が 4 のときに、4 番目の要素を参照したい場合は、0 番を考慮して 3 を指定する必要がある。</li> </ul>	114 -116	158 -162
心理テストプログラムの作成	<ul style="list-style-type: none"> <li>● 書籍に従って心理テストプログラムを作成する。</li> </ul> <div data-bbox="309 1115 611 1406" style="border: 1px solid black; padding: 5px;"> <p><b>心理テスト</b></p> <p>あなたが森の中を一人で散歩していると、ある生き物に出会いました。その生き物はどれですか？</p> <p><input type="text" value="鳥"/> <input type="button" value="回答する"/></p> <p>この心理テストは、あなたが一番大事にしているものを表しています。 あなたが一番大事にしているものは、「友情」です。</p> </div>	<ul style="list-style-type: none"> <li>● このプログラムでは HTML 側で select を使って選択枝を用意し、JavaScript 側で配列を使って結果を用意している。select 側の値と配列のインデックス値を対応させることで連携を行っている。</li> </ul>	116 -119	163 -164
【発展 1】心理テストプログラムの改良	<ul style="list-style-type: none"> <li>● 心理テストの結果と選択枝を追加する。</li> </ul>	<ul style="list-style-type: none"> <li>● select 側の選択枝を増やす場合は配列の要素も増やす必要がある。</li> </ul>	-	-
まとめ	<ul style="list-style-type: none"> <li>● コンピュータで複数の情報を扱うときには曜日のように番号で管理すると扱いが容易になる。また、配列の仕組みを使うことで複数の値をまとめて管理できる。配列を作成すると自動的にインデックスが振られる、また、インデックスを使って要素の値を参照できるようになる。</li> </ul>		-	-

## 留意事項

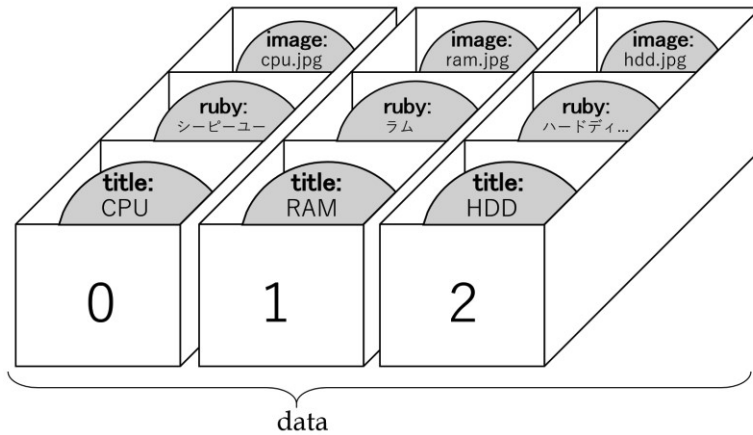
配列の各要素に格納する値に構造を持たせることで、より複雑な情報を取り扱うことが可能になる。

例えば、二次元配列と呼ばれるデータ構造では、配列の各要素に配列を格納することで、横×縦の2次元の表を扱うことができる（図1）。



(図1)

また、アプリプログラミングシートDX図鑑においては、配列を使って複数の図鑑項目を保持する際に、個々の項目に連想配列（キーを使って参照する特殊な配列）を使っている（図2）。



(図2)

※P73に関連する説明あり。

## 第 12 章：繰り返し

### 学習目標

(知識及び技能)

- 繰り返しの制御文を理解し、共通の処理を簡潔に記述する方法を学ぶ。

(思考力, 判断力, 表現力)

- 配列と繰り返しの関係性を理解し、繰り返し処理を用いて配列の値を効率的に参照できることを理解する。
- 日常生活において、繰り返しが利用されている場面を検討する。

(学びに向かう力, 人間性等)

- 配列以外の例で、繰り返しを活用できる用途を検討して記述する。

### 前提条件

コンピュータの基本的な操作方法（クリック，ドラッグ，文字入力）ができる。

ブラウザで Web サイトを利用することができる。

HTML の仕組みや記述方法を体験している。

変数の仕組みを理解している。

条件分岐の仕組みを理解している。

配列の仕組みを理解している。

### 準備

Google Chrome のインストール、Monaca アカウントの作成、書籍の配付

授業用サンプルプロジェクト：第 12 章繰り返し（ひな形）

指導過程 第12章：繰り返し

指導過程	教師の働きかけ	備考	書籍	スライド												
既習事項の確認、学習課題の把握	<ul style="list-style-type: none"> <li>繰り返しの制御文を用いることで共通の処理を簡潔に記述できることを理解させる。</li> <li>日常生活の中で普段から繰り返し行っていることを検討させる。</li> </ul>	<ul style="list-style-type: none"> <li>コンピュータは単純な計算を高速に処理することで便利な機能を実現している。また、プログラムを自分で作成するときにも、コンピュータに実行させたいことを単純な処理の繰り返しや分岐に置き換えて記述するような発想が求められる。</li> <li>毎日学校に通うのは繰り返し処理であり、土日は学校が休みというのは条件分岐である。時間割を配列に格納して for 文で参照することも可能である。</li> </ul>	-	-												
繰り返しを実現する方法を学習する	<ul style="list-style-type: none"> <li>for 文の中では、初期化式・継続条件式・増減式の 3 つの処理が 1 行にまとまっていることと、それぞれの式はこれまで学習した内容の延長線上であることを説明する。</li> </ul> <table border="1" data-bbox="295 891 805 1008"> <thead> <tr> <th>式の種類</th> <th>概要</th> <th>実行するタイミング</th> </tr> </thead> <tbody> <tr> <td>初期化式</td> <td>カウンタ変数を初期化する代入式</td> <td>最初の 1 回だけ実行される</td> </tr> <tr> <td>継続条件式</td> <td>繰り返しを継続する条件式</td> <td>1 回分の繰り返し処理の最初に毎回実行される</td> </tr> <tr> <td>増減式</td> <td>カウンタ変数の値を増減させる計算式</td> <td>1 回分の繰り返し処理の最後に毎回実行される</td> </tr> </tbody> </table>	式の種類	概要	実行するタイミング	初期化式	カウンタ変数を初期化する代入式	最初の 1 回だけ実行される	継続条件式	繰り返しを継続する条件式	1 回分の繰り返し処理の最初に毎回実行される	増減式	カウンタ変数の値を増減させる計算式	1 回分の繰り返し処理の最後に毎回実行される	<ul style="list-style-type: none"> <li>〇回繰り返すという場合には for 文が適しているが、〇〇という状態までの間繰り返すという場合には while 文が適している。</li> </ul>	122 -123	166 -168
式の種類	概要	実行するタイミング														
初期化式	カウンタ変数を初期化する代入式	最初の 1 回だけ実行される														
継続条件式	繰り返しを継続する条件式	1 回分の繰り返し処理の最初に毎回実行される														
増減式	カウンタ変数の値を増減させる計算式	1 回分の繰り返し処理の最後に毎回実行される														
配列のデータを繰り返しで処理する	<ul style="list-style-type: none"> <li>繰り返しを活用することで、配列の値を順番に処理できることを理解させる。</li> <li>配列に格納されている値を元に画像を複数表示するプログラムを作成する。</li> </ul> 	<ul style="list-style-type: none"> <li>for 文のカウンタ変数と配列のインデックスを対応させることで、それらを効果的に活用できる。</li> <li>配列には 0 番の要素があるので、for 文で配列を処理するときには、初期化式でカウンタ変数に 0 を代入し、継続条件式には『&lt;』や『&gt;』を用いて配列の要素数と比較することが多い。</li> </ul>	123 -126	169 -171												
【発展 1】繰り返しプログラムの改良	<ul style="list-style-type: none"> <li>サンプルを改良して追加のフルーツ画像を表示するために必要なことを検討させる。</li> </ul>	<ul style="list-style-type: none"> <li>このサンプルでは画像ファイルを扱っているため、フルーツの種類を増やすためには画像ファイルの用意が必要となる。</li> </ul>	-	-												
まとめ	<ul style="list-style-type: none"> <li>同じ処理を何度も実行したい場合には繰り返しの制御文を利用できる。</li> <li>for 文では変数の初期化や比較演算・代入演算など、これまで学習してきた知識が必要となる。</li> <li>配列や繰り返しを学ぶことで、現実の情報をデータとして変数に格納してプログラムで処理することが容易になる。</li> </ul>		-	-												



## 留意事項

### 繰り返しの記述方法（for 文と while 文）

プログラムを作成するときには「同じ処理を何回も繰り返し処理したい」という場面がしばしば発生する。プログラミング言語には繰り返しを表現する構文が複数用意されており、代表的なものとして for 文や while 文が存在する。

繰り返しの回数が事前に決められる場合には for 文、事前の繰り返し回数が決まっておらず「特定の条件を満たしている間繰り返す」場合には while 文の使用が適している。

for 文と while 文の記述方法は以下の通りである。

#### ○ for 文の記述方法

```
for ( 初期化式; 継続条件式; 増減式 ) {  
    処理  
}
```

#### 変数 year の回数繰り返す例

```
for ( var i = 1; i <= year; i++ ) {  
    処理  
}
```

初期化式                      継続条件式                      増減式

#### ○ while 文の記述方法

```
while ( 継続条件式 ) {  
    処理  
}
```

#### 変数 amount が変数 target より小さい間繰り返す例

```
while ( amount < target ) {  
    処理  
}
```

継続条件式

for 文は初期化式でカウンタ変数(本事例では変数 i )を定義し、継続条件式で繰り返しの回数を指定する。また、増減式は繰り返しのたびに処理されるため、繰り返しの度にカウンタ変数が一ずつ増えるように記述する。

例えば、毎年 10 万円積み立てて、利息を複利で 2%受け取った際の 10 年後の貯金額を求めるといったケースでは for 文の使用が適している。

while 文には初期化式や増減式が存在しないため、繰り返し処理の中で継続条件式を満たす状態となるように継続条件式に関する変数の値を増減させたり、変化させたりするなどの処理を記述する必要がある。

例えば、積立金額が 100 万円に達するまで毎年 10 万円ずつ積み立てるといったケースでは while 文の使用が適している。

## アプリプログラミングシート学習段階表

学習段階	1		2		3		4	
	導入／基礎 1		基礎 2		基礎 3		基礎 4	
時間数	4時間程度		4時間程度		4時間程度		4時間程度	
学習内容	【導入】 ツールの使い方 プログラミング概要 【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。		【主】 条件による分岐の方法 【主2】 乱数の取得		【主】 繰り返しの処理 【副】 フォームの値を変数に取得 する 変数の値をHTMLに反映す る		【主】 繰り返しの処理と配列	
教材アプリ	図鑑		おみくじ		貯金シミュレーション アプリ		DX図鑑	
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。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 でページが移動できるアプ リ。		ボタンを押すと、「大吉」 「中吉」「凶」などの結果 をランダムに表示するアプ リ。		フォームに毎年積み立てる 金額と利率を入力し、計算 ボタンを押すことで、1年 毎の合計金額が表示される アプリ。		メニューボタンを押すと、 メニューの項目が表示さ れ、項目を選ぶと画像や説 明文が表示されるアプリ。	
カスタマイズ①	トップページの項目と新し いページの追加		おみくじの結果を追加する		積み立ての年数を変更でき るようにする		DX図鑑の項目を追加する	
カスタマイズ②	配色の変更		メッセージを追加する		while文を使用して、目標 金額に達するために必要な 年数を求める		DX図鑑の各項目のタイト ルにルビを追加する	
備考	HTMLやCSS、Webサー ビスは他分野との関わりを 示しながら指導できる。		カスタマイズによって独自 性を発揮しやすい内容であ るので、学習に前向きに取 り組む姿勢を育てたい。		複利計算の仕組みについて の知識を学ぶことができ る。		画像の扱い方等を通して、 著作権や情報モラル等につ いての学習に発展させるこ ともできる。	
学習段階	5							
	発展 1				発展 2			
時間数	4時間程度				4時間程度			
学習内容	【主】 関数の使用とセンサーによ る情報の取得				【主】 配列と関数およびオープン データの活用			
教材アプリ	地図アプリ（基本編）				地図アプリ （オープンデータ編）			
アプリの概要	和歌山城を中心とした地図 が表示され、下には自分の 現在地に関する情報が表示 されるアプリ。和歌山城や その周辺にマーカーが表示 される。				和歌山城を中心とした地図 が表示されるアプリ。和歌 山城やその周辺にマーカー が表示され、その位置や表 示内容をプログラムによっ て変更する。			
カスタマイズ①	地図の中心と拡大率を変え る。				紹介したい地点の緯度、経 度を調べ、地図上の複数の 地点にマーカーを立てる。			
カスタマイズ②	地磁気センサーの値を取得 し、地図を回転させる。				オープンデータを活用して マーカーを立てる。			
備考	地域社会や自分の生活に関 わる問題の発見と解決に生 かすことができる。				地域社会や自分の生活に関 わる問題の発見と解決に生 かすことができる。			

アプリプログラミングシート 指導案

HTML / CSS を使ったアプリ(図鑑)の作成

# HTML / CSS を使ったアプリ(図鑑)の作成

## 学習目標

(知識及び技能)

- アプリを自分たちで作成できることに気づく。
- プログラムを記述するためには、ファイルの編集や保存が必要であることがわかる。
- 統合開発環境を使ったファイルのインポート、編集や保存を行うことができる。
- HTML、CSS の基本的な使い方がわかる。

(思考力, 判断力, 表現力)

- 自分が扱いたい情報を HTML によって表現することができる。
- 自分の意図する HTML ページの見た目 (装飾) を、CSS を用いて表現することができる。

(学びに向かう力, 人間性等)

- デジタルの消費者から生産者となる視点を持つ。
- ユーザビリティやユニバーサルデザインといった視点で見やすい情報として表現しようとする。

## 前提条件

コンピュータの基本的な操作方法 (クリック, ドラッグ, 文字入力) ができる。

ブラウザでウェブサイトを利用することができる。

Monaca 開発環境の基本的な使い方を理解している (ログイン, インポート, 編集)。

## 準備

Google Chrome のインストール、Monaca アカウントの作成、書籍の配付、MonacaEducation 教材紹介ページ(html)の配付

授業用サンプルプロジェクト：図鑑

## 単元の流れ 指導過程 HTML/CSS を使ったアプリ(図鑑)の作成

本指導案は最短となる 3 時間で作成しているが、授業の実態に合わせて 3 時間以上の取組とすることも可能である。

また、各時の展開方法についても一般的な方法を示したが、授業の実態に合わせて適切な方法を選択することが推奨される。

時	学習活動の概要	各時の展開				備考
		導入	展開 1	展開 2	まとめ	
1	図鑑アプリを <ul style="list-style-type: none"> <li>┌ Monaca で開く</li> <li>├ 読み解く</li> <li>└ 動作させる</li> </ul>	課題提示 <ul style="list-style-type: none"> <li>・学習内容</li> <li>・制作課題</li> </ul>	【確認】 アプリをインポートし動作させる	【実践】 関係ファイルの一覧を確認し、コードを読む	次回の授業実施内容の提示	
2	アプリプログラミングシートに沿って図鑑アプリをカスタマイズする	・前時までの復習 ・課題提示	【活動①】 カスタマイズ① 図鑑項目の追加	【活動②】 カスタマイズ② 色の変更	本時の総括	カスタマイズの①と②をどのように行うかは生徒の実態に合わせる
3	理論(HTML / CSS)を理論化する	前時までの復習	【理論①】 HTML の記述を学ぶ	【理論②】 CSS の記述を学ぶ	単元の総括	

## 第 1 時 図鑑アプリを開く・読む・動かす

第 1 時においては、単元での学習内容の提示と共に、アプリのプログラムのインポート、動作の確認、プログラムの読み込みを行う。

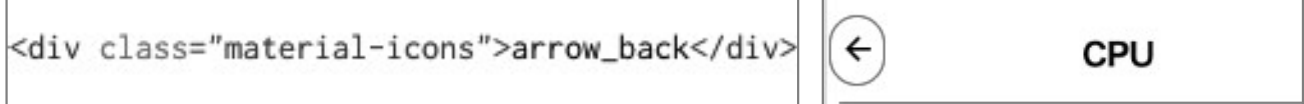
指導過程	教師の働きかけ	備考	シート
導入	<b>【課題提示】</b> <ul style="list-style-type: none"> <li>┌ ・学習課題：HTML / CSS)</li> <li>└ ・制作課題：図鑑</li> </ul>	教師の発言例 「図鑑アプリの制作を通じて、HTML / CSS について学んでいきましょう」	
展開 1	<b>【確認】</b> ・Monaca にログイン後、右記の「サポート web」から、図鑑アプリをインポートする ・クラウド IDE（統合開発環境）で開くと、画面右側にプレビュー画面が表示されることを確認する <b>【トップページ】 ⇔ 【個別項目ページ】</b> のリンクについて、双方向で移動できることを確認する。 ・HTML のタグに挟まれた文字を打ち替え、保存するとプレビュー画面に変更が反映されることを確認する	「サポート Web」の URL <a href="https://edu.monaca.io/wakayama">https://edu.monaca.io/wakayama</a>  例：トップページの「CPU」を「シーピーユー」に変更 <pre style="border: 1px solid black; padding: 5px; font-family: monospace;">           &lt;div id="menu"&gt;             &lt;a href="cpu.html"&gt;&lt;button&gt;シーピーユー&lt;/button&gt;&lt;/a&gt;             &lt;a href="ram.html"&gt;&lt;button&gt;RAM&lt;/button&gt;&lt;/a&gt;             &lt;a href="hdd.html"&gt;&lt;button&gt;HDD&lt;/button&gt;&lt;/a&gt;             &lt;a href="ssd.html"&gt;&lt;button&gt;SSD&lt;/button&gt;&lt;/a&gt;           &lt;/div&gt;           </pre>	P.1
展開 2	<b>【実践】</b> ・関係ファイルの一覧を確認する ・指導上のポイントがあることを確認する <ul style="list-style-type: none"> <li>┌ html ファイル（ページの内容そのもの）</li> <li>├ css ファイル（ページの装飾用）</li> <li>├ jpg 等の画像ファイル</li> <li>└ 各種設定ファイル(アイコンが歯車マーク)</li> </ul> ・index.html や cpu.html 等を読み込む ・<a>タグ及び<img>タグについては、この後使うので必ず説明する	教師からの提示とするか、各生徒の作業とするかは、授業の実態に合わせて適宜選択する。 （例）一連を教員が提示した後、各生徒で作業させ、動作等の確認を行う （例）教師の提示と各生徒の作業及び確認を細かく分けて実施する  <a>タグ <a>～</a>でハイパーリンクを設置 <img>タグ <img>によって画像を表示	P.1
まとめ	本時は、HTML 及び CSS についてのインプット中心の授業であり、次時にはカスタマイズしていくことを示し、生徒の意欲を喚起する。	アプリプログラミングシートの配付について （例）両面刷りで本時に配付 （例）片面刷りで本時、次時に分けて配付などは、授業の展開に合わせる。	

## 第1時 留意事項

### ① 文字列 `arrow_back` が矢印アイコンになる

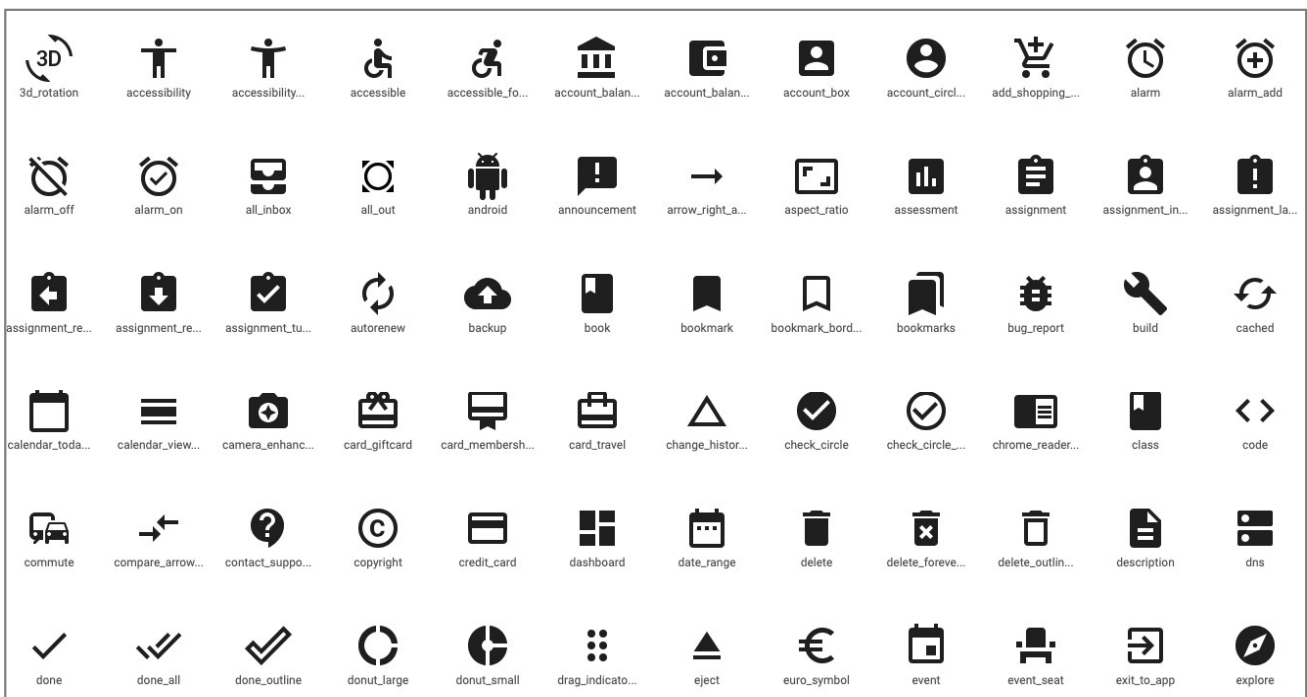
※ 本単元の学習の内容と直接関係しないので、詳細な指導・解説は省略可能である。

下記の記述により、「`arrow_back`」という文字列ではなく、左向きの矢印がアイコンで表示される。



これは Google の提供する Material icons というライブラリ (<https://material.io/tools/icons/>) を使用しており、下図のような数百のアイコンが簡単な記述により使用可能となる。

使用可能なアイコン(一部抜粋) <https://google.github.io/material-design-icons/>



Material icons は下記の CSS を HTML の head の中で読み込み、

```
<link href="https://fonts.googleapis.com/icon?family=Material+Icons" rel="stylesheet">
```

下記のように `material-icons` クラスの中でアイコンの名称の文字列(`arrow_back` 等)を指定するとアイコンの表示が簡易に実現可能である。

```
<div class="material-icons">arrow_back</div>
```

## 第 2 時 アプリプログラミングシートに沿って図鑑アプリをカスタマイズする

第 2 時においては、アプリプログラミングシート P.2 の内容を各生徒共通に実施する。

指導過程	教師の働きかけ	備考	シート
導入	<p>【課題の提示】</p> <p>前時の学習内容の確認</p> <ul style="list-style-type: none"> <li>・ログインする</li> <li>・プロジェクトを開く</li> <li>・html / css 等の各種ファイルを確認する html 内の&lt;a&gt;,&lt;img&gt;タグ など</li> </ul> <p>本時の課題の提示</p> <p>アプリプログラミングシート(P.2)の「図鑑の項目と配色をカスタマイズしてみよう」など</p>	<p>アプリプログラミングシートには、①ページの追加、②色の変更 の 2 事例があるが、学習課題としては①を優先とし、授業の実態に合わせて②の実施については調整する。</p> <p>(例) 全員で①、②を実施【本指導案】</p> <p>(例) 全員で①を実施し、②については早く①が終わった生徒が実施するなど</p>	P.2
展開	<p>【活動】カスタマイズ①</p> <ol style="list-style-type: none"> <li>1 「図鑑に『SD カード』のページを追加する」というカスタマイズを行う課題を提示する</li> <li>2 必要な画像やテキスト文章を用意する</li> <li>3 ssd.html をコピーし、sd.html とする</li> <li>4 sd.html の h1,説明,ライセンスの内容を変更する</li> <li>5 sd.html の画像を変更する</li> <li>6 index.html にリンクを追加する</li> <li>7 変更を確認する</li> <li>8 同様の改良を生徒が実施するよう指示</li> </ol> <p>※ 机間指導し、生徒の理解促進を図る</p>	<p>教師の提示と生徒の実習・確認をどのように切り替えるかは、授業の実態に合わせて行う。</p> <p>早く完了した生徒は、周囲の生徒のサポートや別項目（別のページ）の追加をしてみるように指示する。</p>	P.2 左
展開	<p>【活動】カスタマイズ②</p> <ol style="list-style-type: none"> <li>1 css/style.css を開く</li> <li>2 アプリプログラミングシートのように背景色を変更し、反映されることを示す</li> <li>3 同様の改良を生徒が実施するよう指示する</li> </ol> <p>※机間指導し、生徒の理解促進を図る</p>	<p>早く完了した生徒は、他の項目を変更して様々な変化を試してみるように指示する。</p> <p>またユニバーサルデザインや誰が見ても見やすい情報として表現できるように考え、工夫してみるように指示をする。</p>	P.2 右
まとめ	<p>【確認】</p> <p>HTML と CSS、画像ファイルを組み合わせることで、自分たちが扱いたい情報を表現できたことを確認する。</p>		



第3時 理論(HTML/CSS)を学ぶ

第3時においては、ここまでの内容について書籍(第2,3章 / P.20-)を活用して理論的な理解を図る。

指導過程	教師の働きかけ	備考	書籍
導入	<p>【前時までの学習内容確認】</p> <ul style="list-style-type: none"> <li>・ログイン方法について</li> <li>・プロジェクトを開く方法について</li> <li>・html / css 等の各種ファイルの確認</li> <li>・html 内の&lt;a&gt;,&lt;img&gt;タグ など</li> </ul> <p>本時のめあての確認 作成したアプリについて、HTML を読み、表示内容やリンク等について確認する。</p>	<p>教師の発言例 「前回までに作成したアプリがどのように動いているか詳しく見ていきましょう」 など</p>	
展開 1	<p>【理論】 HTML について確認する。 指導上のポイント</p> <ul style="list-style-type: none"> <li>・タグの開始と終了について</li> <li>・各タグの属性について</li> <li>・タグの入れ子構造について</li> <li>・リンクタグ(&lt;a&gt;タグ)について</li> <li>・画像(&lt;img&gt;タグ)について</li> </ul>	<p>HTML は表示内容そのものを扱う点も確認する。 1 つの html ファイルだけでなく、他の html, css, 画像ファイル等が組み合わさり Web サイトが構築されていることも確認する。</p>	P.20 -31
展開 2	<p>【理論】 CSS について確認する。 指導上のポイント</p> <ul style="list-style-type: none"> <li>・セレクタ タグ、id、クラスセレクタ</li> <li>・プロパティ サイズ、位置の指定</li> <li>・値 色、ピクセル数値の指定</li> </ul>	<p>CSS は HTML で作られたページに装飾をしていることを確認する。</p>	P.36 -46
まとめ	<p>【単元の総括】 本単元では、図鑑アプリの制作を通して各種ファイルから Web サイトやアプリが構成されていること、また、HTML で内容を、CSS で装飾を指定していることを確認する。</p>		

## アプリプログラミングシート 指導案

条件による分岐を使ったアプリ(おみくじ)の作成

## 条件分岐を使ったアプリ(おみくじ)の作成

### 学習目標

(知識及び技能)

- 任意の条件によってプログラムの実行を振り分ける方法を理解することができる。
- 比較演算子による真偽の判定方法を理解することができる。

(思考力, 判断力, 表現力)

- 自分が意図する判定のための条件を比較演算子を用いて表現することができる。

(学びに向かう力, 人間性等)

- より複雑な判定を条件による分岐を用いた、プログラムによって行う方法を検討しようとする。

### 前提条件

コンピュータの基本的な操作方法（クリック，ドラッグ，文字入力）ができる。

ブラウザで Web サイトを利用することができる。

Monaca 開発環境の基本的な使い方を理解している（ログイン，インポート，編集）。

HTML の仕組みや記述方法を体験している（文字列と画像の表示を含む）。

変数の仕組みを理解している。

### 準備

Google Chrome のインストール、Monaca アカウントの作成、書籍の配付、MonacaEducation 教材紹介ページ(html)の配付

授業用サンプルプロジェクト：おみくじ

## 単元の流れ 指導過程 条件分岐を使ったアプリ(おみくじ)の作成

本指導案は、最短で4時間として作成しているが、授業の実態に合わせて4時間以上の取組とすることも考えられる。また、各時の展開方法についても一例を示したが、授業の実態に合わせて適切な方法に変更し実践することが推奨される。

時	学習活動の概要	各時の展開				備考
		導入	展開 1	展開 2	まとめ	
1	おみくじアプリを <ul style="list-style-type: none"> <li>┌ Monaca で開く</li> <li>├ 読み解く</li> <li>└ 動作させる</li> </ul> (フローチャートを含む)	課題提示 ・学習内容 ・制作課題	【設計】 フローチャートで 検討 (条件分岐の動作を含む)	【実践】 アプリをインポート し実行する コードを読み解く	次回の実施内容の提示	
2	アプリプログラミングシートに沿っておみくじアプリをカスタマイズする	前時の復習 課題提示	【活動】 カスタマイズを行う 早く完了した個人またはグループは、他の人の作業を助けるなどサポートを行う		本時の総括	カスタマイズ① または②をどのように行うかは授業の実態に合わせる
3	理論(条件による分岐他)を学ぶ	前時までの復習	【理論】 if 文を学ぶ if / else / else if 等	【理論】 比較演算子を用いた条件式について学ぶ	次回への意欲喚起	
4	設定した課題にチャレンジする	前時までの復習 課題提示	【活動】	【共有】	単元の総括	

※自由制作に応じ、4時間を超えて実施することも考えられる

第1時 おみくじアプリを開く・読む・動かす

第1時においては、「単元の学習内容」の提示とともに、アプリの要件をフローチャートで考えた後、アプリのプログラムの読み込みと動作の確認を行う。

指導過程	教師の働きかけ	備考	シート
導入	<p>【課題提示】</p> <ul style="list-style-type: none"> <li>・単元の学習内容：条件による分岐の理解</li> <li>・制作課題：おみくじアプリの両方を示し、導入とする。</li> </ul>	<p>教師の発言例</p> <p>「おみくじアプリの制作を通じて、条件による分岐の処理を学んでいきましょう」</p>	P.1 左上
展開1	<p>【設計】</p> <p>アプリの流れ(下記 1~3)をフローチャートにて示す(次ページ留意事項参照)。</p> <ol style="list-style-type: none"> <li>1 おみくじの結果(大吉等)は乱数で求めた値をもとにして決めることとする</li> <li>2 乱数で求めた値に応じて、異なる画像を変数に格納(代入)する</li> <li>3 おみくじの結果の画像を表示する</li> <li>4 生徒の理解に沿って次を示してもよい ※ <ul style="list-style-type: none"> <li>・結果ダイアログの表示</li> <li>・次のおみくじが引けるよう表示 など</li> </ul> </li> </ol>	<p>フローチャートについては、</p> <p>(例) アプリプログラミングシートのフローチャートを示す</p> <p>(例) 板書等で作成過程から示す</p> <p>(例) 様々なことをフローチャートで書かせる検討をさせる</p> <p>など、授業の実態に応じ、適宜行う。</p> <p>※ 4については、「よりよいアプリの作成」のためには必要な視点である。ただし、今回の単元の学習内容となる「条件による分岐」の本質的な部分ではないため、生徒に示す場合は留意する</p>	P.1 左下
展開2	<p>【実践】</p> <ul style="list-style-type: none"> <li>・Monaca にログインする</li> <li>・アプリを右記 URL よりインポートする</li> <li>・実際のアプリの動作を確認する</li> <li>・プログラムをフローチャートと対応させながら、プログラムの流れを理解する</li> </ul> <p>※ 生徒の理解に沿って次項を示してもよい</p> <ul style="list-style-type: none"> <li>・乱数を alert() で表示させるなどのインタラクティブな動作確認も実施する</li> </ul>	<p>サポート Web URL</p> <p><a href="https://edu.monaca.io/wakayama">https://edu.monaca.io/wakayama</a></p> <p>教師からの提示とするか、各生徒の作業とするかは、授業の実態に合わせる。</p> <p>(例) 一連を教員が提示した後、各生徒で作業させ、動作等の確認を行う</p> <p>(例) 教師の提示と各生徒の作業及び確認を細かく分けて実施する</p>	P.1 右
まとめ	<p>本時は条件による分岐の知識についてインプットすること中心であり、次時にはカスタマイズしていくことを示し、生徒の意欲を喚起する。</p>	<p>アプリプログラミングシートの配付については、授業の展開に合わせる。</p> <p>(例) 両面刷りで本時に配付</p> <p>(例) 片面刷りで本時、次時に分けて配付</p>	

## 第1時 留意事項

- ① おみくじの結果は乱数で求めた値をもとに決めることとする
- ※ おみくじの結果の決め方（基準となる値の決め方など）については、乱数以外の方法もあり得るが今回の例では乱数を用いる。
  - ※ 本単元の学習内容と直接関係しないので、詳細な指導・解説は省略可能である。

Math.random()

0 以上 1 未満 (0.0000...~0.9999...) のランダムな小数値(乱数)を生成する。

Math.random() \* 5

乱数値を 5 倍することにより最小値 0 以上、最大値 5 未満の小数値になる。

最小値 0.0000... \* 5 = 0      最大値 0.9999... \* 5 = 4.9999...

Math.floor()

小数値の切り捨てを行う。

「ランダムな小数値を 5 倍した値」の小数点以下を切り捨てることにより、0、1、2、3、4 のいずれかの整数値をランダムに求めている。

```
function play() {  
    // 0~4の範囲のランダムな整数の値を得る  
    var num = Math.floor(Math.random() * 5);  
}
```

- ② 乱数によって求められた値に応じて、画像 (daikichi.png 等) を変数に格納 (代入) する
- ①で求められた値に応じて、結果となる画像のファイル名を変数(image\_name)に格納する。
- フローチャート及びソースコードを提示しながら、「乱数により求められた値(num)が1である場合、どの命令が実行される(どの画像が表示される)か」などの提示を行い、条件による判定によって命令が分岐されることについて、理解できるように指導する。
- ③ 結果となる画像を表示する
- ※ 本単元の学習の内容と直接関係しないので、詳細な指導・解説は省略可能である。
- ②にて変数(image\_name)に格納した画像のファイル名が、<img id="omikuji" />の参照先の画像ファイル名となることで画像が表示される。

## 第2時 アプリプログラミングシートに沿っておみくじアプリをカスタマイズする

第2時においては、アプリプログラミングシート P.2 の内容を各生徒共通で実施する。

指導過程	教師の働きかけ	備考	シート
導入	<p>【課題の提示】</p> <p>前時の学習内容確認</p> <ul style="list-style-type: none"> <li>・ログイン方法について</li> <li>・プロジェクトを開く方法について</li> <li>・フローチャートについて</li> </ul> <p>(乱数によってもととなる値を求める → 条件による分岐で画像ファイル名を変数に格納 → 格納された画像ファイルの表示)</p> <p>本時の課題の提示 アプリプログラミングシート(P.2)の 「おみくじの結果を追加してみよう」など</p>	<p>アプリプログラミングシートの①結果の追加、②メッセージの追加のそれぞれをどのように実施するかは授業の実態により適宜選択する。</p> <p>(例) 全員で①と②を一律に実施 (例) 全員で①を実施し、早く終わった場合②に進む【本指導案の方法】 (例) 全員で①を実施し、完了した(理解ができた)生徒から、他の生徒のサポートを行い、学習補助をする</p>	
展開	<p>【活動】</p> <ol style="list-style-type: none"> <li>1 「おみくじの結果に『大凶』を追加する」カスタマイズ課題を提示する</li> <li>2 フローチャートについて必要な変更を考察させる(発問等で生徒に働きかける)</li> <li>3 プログラムの変更を実演する</li> <li>4 プログラムの動作の確認をする ※ 動かない場合はデバッグする</li> <li>5 同様の変更を生徒が実施するよう指示し、変更及び動作確認を行わせる ※ 机間指導を行い、理解促進を図る</li> <li>6 課題を達成した生徒には、もう一方の「メッセージの追加」のカスタマイズ課題に取り組んでみるよう指示する</li> </ol>	<p>本指導案では、</p> <ol style="list-style-type: none"> <li>[1] フローチャートの変更：発問しクラス内で考える</li> <li>[2] プログラムの変更：教師が示す</li> <li>[3] 動作の確認：教師が示す</li> <li>[4] 同様の変更：各生徒が実施</li> </ol> <p>としているが、教師の提示と生徒の実習・確認をどのように組み合わせるかは授業の実態に合わせて選択する(一連を提示した後に各生徒での確認 / 提示と確認を細かく分けて実施等)</p>	P.2
まとめ	<p>【確認】</p> <p>本時での次の点の一致について確認</p> <ul style="list-style-type: none"> <li>・変更の意図について</li> <li>・フローチャートの変更部分について</li> <li>・プログラムの変更について</li> <li>・アプリの挙動の変化について</li> </ul>		

## 第2時 留意事項

- ① 「おみくじの結果に『大凶』を追加する」ために必要なフローチャートの変更について考察する
- ・おみくじの結果のもとになる値（乱数を用いて求める値）の範囲拡大（5=daikyo）
  - ・変数(num)が追加された値（5）であった場合に、変数 image\_name に daikyou.png の画像ファイル名を格納する必要があること（ image\_name="daikyo.png" ）
- の2点を変更すべき点となる。

- ② プログラムの別の変更方法について

※ 本単元の学習の内容と直接関係しないので、詳細な指導・解説は省略可能である。

下記のように、else は「凶」のまま変更せず、num == 5 の else if を追加し大凶を表示するというプログラムの変更を行った場合でも同様の動作を達成することができる。

ただし、さらにプログラムのカスタマイズ（さらなる結果の追加など）を行う場合のプログラムの読みやすさを考えると、アプリプログラミングシートに掲載したカスタマイズ方法の方がより良いと考えられる。

また、生徒が自ら考察を行い、下記のようなカスタマイズを行った場合には、「間違いである」として指摘することのないように留意する必要がある。プログラムで行いたい結果を達成させる方法については、多様性がある。

```
} else if (num == 3) {  
    image_name = "suekichi.png";  
    message = "末吉です";  
} else if (num == 5) {  
    image_name = "daikyou.png";  
    message = "大凶です";  
} else {  
    image_name = "kyou.png";  
    message = "凶です";  
}
```



### 第3時 アプリプログラミングシートに沿っておみくじアプリをカスタマイズする

第3時においては、第2時までの内容について書籍(第5章/P.62-)を活用して理論的な理解を図る。  
また、他の比較演算子を学ぶことで第4時の活動の幅を広げる。

指導過程	教師の働きかけ	備考	書籍																												
導入	<p>【前時までの学習内容確認】</p> <ul style="list-style-type: none"> <li>・ログインの方法について</li> <li>・プロジェクトを開く方法について</li> <li>・フローチャートについて</li> </ul> <p>(乱数によって値を求める →条件による分岐で画像ファイル名を変数に格納 →格納した画像ファイルの表示)</p> <p>本時のめあての確認：プログラムの動作について理論的な視点で読む。</p>	<p>教師の発言例</p> <p>「前回までに作成したプログラムがどのように動いているのか詳しく見ていきましょう」</p>																													
展開 1	<p>【理論】</p> <p>if 文の記述方法と実行のされ方を確認する。 if / else / else if</p> <pre style="border: 1px solid black; padding: 5px;"> if (条件式) {   条件式が正しい場合に実行する処理 } else {   条件式が正しくない場合に実行する処理 }                     </pre>	<p>【指導の留意点】</p> <ul style="list-style-type: none"> <li>・比較は『==』(等値演算子)</li> <li>・代入は『=』</li> </ul> <p>これらは間違えやすく、また文法エラーとならないので気がつきにくいいため留意させる。</p>	P.62 -68																												
展開 2	<p>【理論】</p> <p>様々な比較演算子を示し、意味と条件式の例などについて学ぶ。</p> <table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr> <th>演算子</th> <th>概要</th> <th>条件式の例</th> <th>結果</th> </tr> </thead> <tbody> <tr> <td>==</td> <td>左辺と右辺が等しい場合は正しい</td> <td>1 == 1</td> <td>正しい</td> </tr> <tr> <td>!=</td> <td>左辺と右辺が等しくない場合は正しい</td> <td>1 != 2</td> <td>正しい</td> </tr> <tr> <td>&lt;</td> <td>左辺が右辺より小さい場合は正しい</td> <td>1 &lt; 1</td> <td>正しくない</td> </tr> <tr> <td>&lt;=</td> <td>左辺が右辺以下の場合は正しい</td> <td>1 &lt;= 1</td> <td>正しい</td> </tr> <tr> <td>&gt;</td> <td>左辺が右辺より大きい場合は正しい</td> <td>1 &gt; 1</td> <td>正しくない</td> </tr> <tr> <td>&gt;=</td> <td>左辺が右辺以上の場合は正しい</td> <td>1 &gt;= 1</td> <td>正しい</td> </tr> </tbody> </table>	演算子	概要	条件式の例	結果	==	左辺と右辺が等しい場合は正しい	1 == 1	正しい	!=	左辺と右辺が等しくない場合は正しい	1 != 2	正しい	<	左辺が右辺より小さい場合は正しい	1 < 1	正しくない	<=	左辺が右辺以下の場合は正しい	1 <= 1	正しい	>	左辺が右辺より大きい場合は正しい	1 > 1	正しくない	>=	左辺が右辺以上の場合は正しい	1 >= 1	正しい	<p>・おみくじにおいては、== のみ使用しているが、他の演算子についてもこの時間に学習する。</p>	P.63
演算子	概要	条件式の例	結果																												
==	左辺と右辺が等しい場合は正しい	1 == 1	正しい																												
!=	左辺と右辺が等しくない場合は正しい	1 != 2	正しい																												
<	左辺が右辺より小さい場合は正しい	1 < 1	正しくない																												
<=	左辺が右辺以下の場合は正しい	1 <= 1	正しい																												
>	左辺が右辺より大きい場合は正しい	1 > 1	正しくない																												
>=	左辺が右辺以上の場合は正しい	1 >= 1	正しい																												
まとめ	<p>本時は理論についてのインプットが中心であり、次時には個々のアイデアでカスタマイズを行ってもらうことを示し、意欲を喚起する。</p>																														

### 第3時 留意事項

① if 文の記述方法と実行のされ方を確認する (if 文のプログラムを読む)

- ・ アプリプログラミングシート P.1 などのフローチャートを示し、処理の流れを確認する。
- ・ プログラムを示し、プログラムの実行のされ方を確認する。

┌ if のみ  
├ else を加える  
└ else if を加える

などの、説明する段階の組み方が考えられる。

② 【間違いやすいポイント】 == と = の間違い

- ・ 比較演算子においては「==」を使用するが、「=」と誤った場合も「変数への代入」となるために「文法エラーとはならない」ので誤りに気づきにくい。

③ 他の比較演算子

- ・ 各生徒が、次時でプログラムのカスタマイズを行う課題に向けて指導する。
- ・ 本プログラムにおいて比較演算子を使用する場合は  
「num==3 を num>=3 に変更した場合」の動作を予想させる  
「変数 (num) が 3 以上の場合」などとしてプログラムを変更させてみる  
といった学習方法が想定される。

## 第4時 アプリプログラミングシートに沿っておみくじアプリをカスタマイズする

第4時においては、第3時までの学習内容をもとに各生徒の意欲に沿って制作活動を行う。

指導過程	教師の働きかけ	備考	シート
導入	<p>【前時までの学習内容確認】</p> <ul style="list-style-type: none"> <li>┌ ・条件に応じて結果（画像）を表示するアプリ</li> <li>└ ・画像だけでなく文字列等も表示可能</li> </ul> <p>if / else / else if や様々な比較演算子を用いることにより柔軟な（任意の）動作をさせることができる。</p>		
展開 1	<p>【活動】 アプリプログラミングシートにある3つの中からどの活動をするか選択する。</p> <ol style="list-style-type: none"> <li>① 自分のアイデアでさらにカスタマイズしてみよう</li> <li>② おみくじアプリをはじめから作ってみよう</li> <li>③ おみくじアプリの仕組みを使って、別のアプリを作ってみよう</li> </ol> <p>本時においてもこれまでと同様に</p> <ol style="list-style-type: none"> <li>1 自分(たち)の作成の意図を明らかにする</li> <li>2 フローチャートで流れを考察する</li> <li>3 プログラムを記述する</li> <li>4 動作させてみる</li> <li>5 プログラムを実行し、意図したことと違う場合、デバッグを繰り返し、修正する</li> </ol> <p>この流れを大切にできるよう指導する（特に、①と③の活動の場合は必ず指導する）。</p>	<p>生徒の実態に合わせて活動する集団（クラス単位/グループ単位/個人単位）を決めて行う。</p> <p>活動内容については授業の実態に合わせて適宜選択する。</p> <p>（例）学級全員が①を実施する （例）原則として①を実施し、意欲のある生徒は③を実施する など</p>	P.2 右
展開 2	<p>【共有】 各生徒が作成したものをクラス全体、グループ内等で共有する。</p> <ul style="list-style-type: none"> <li>・自分が意図したこと</li> <li>・フローチャートのポイントについて</li> <li>・プログラムのポイント（工夫点など）について</li> <li>・実際の作品の動作、実演</li> <li>・更なる課題点や修正したい点について</li> </ul> <p>などを可能な限り共有するように指導し、他者及び自分の作品の評価を行う。</p>		
まとめ	<p>本単元では、おみくじアプリの制作を通して、条件によって結果が変わるアプリを作成したことを確認する。</p> <p>また、if / else 文や比較演算子の組み合わせを柔軟に使うことにより、様々なプログラムが実現可能となる。</p>	<p>条件により、プログラムを分岐させることで、求められる結果を変えることができることを確認する。</p>	

## アプリプログラミングシート 指導案

繰り返し処理を使ったアプリ(貯金シミュレーションアプリ)の作成

## 繰り返し処理を使ったアプリ(貯金シミュレーションアプリ)の作成

### 学習目標

(知識及び技能)

- 同じ処理を何度も実行する場合は、繰り返し処理にすることで効率化できることを理解する。
- 繰り返し文を使い、何度も繰り返す処理を効率的にプログラムで記述する方法を理解する。
- 算術演算子を使い、数値の計算処理をプログラムで記述する方法を理解する。
- 複合代入演算子を使い、数値の計算処理を効率的にプログラムで記述する方法を理解する。

(思考力, 判断力, 表現力)

- 自分の意図する一連の動作から、複数回、同様の処理を行う必要がある冗長的な部分や共通して繰り返される部分を判断し、繰り返し文を用いてその処理を表現することができる。
- 自分の意図する処理について、適切な演算子を組み合わせて表現することができる。

(学びに向かう力, 人間性等)

- 日常で繰り返し発生する冗長的な計算等の処理について、自動化して効率化しようと考察することができる。

### 前提条件

コンピュータの基本的な操作方法（クリック，ドラッグ，文字入力）ができる。

ブラウザで Web サイトを利用することができる。

Monaca 開発環境の基本的な使い方を理解している（ログイン，インポート，編集）。

HTML の仕組みや記述方法を体験している。

変数の仕組みを理解している。

### 準備

Google Chrome のインストール、Monaca アカウントの作成、書籍の配付、MonacaEducation 教材紹介ページ(html)の配付

授業用サンプルプロジェクト：貯金シミュレーションアプリ

単元の流れ 指導過程 繰り返し/演算子を使ったアプリ(貯金シミュレーションアプリ)の作成

本指導案は、最短で4時間として作成しているが、授業の実態に合わせて4時間以上の取組とするとも考えられる。また、各時の展開方法についても一例を示したが、授業の実態に合わせて適切な方法に変更し実践することが推奨される。

時	学習活動の概要	各時の展開				備考
		導入	展開 1	展開 2	まとめ	
1	貯金シミュレーションアプリを <ul style="list-style-type: none"> <li>┌ Monaca で開く</li> <li>├ 読み解く</li> <li>└ 動作させる</li> </ul> (フローチャートを含む)	課題提示 ・学習内容 ・制作課題	【設計】 フローチャートで検討	【実践】 アプリをインポートし実行する コードを読み解く	次回実施内容の提示	
2	アプリプログラミングシートに沿って貯金シミュレーションアプリをカスタマイズする	前時の復習 課題提示	【活動】 カスタマイズ① 積み立てる期間を入力して処理するアプリにする	【活動】 カスタマイズ② 目標金額を設定し、何年かかるかを求めるアプリにカスタマイズする	本時の総括	カスタマイズ①、②は授業の実態に合わせて、2時間に分けて行ってもよい
3	理論(各演算子及び繰り返しの処理)を学ぶ	前時までの復習	【理論】 繰り返し処理について学ぶ	【理論】 算術演算子(+、-、*、/)及び複合代入演算子(++、+=など)について学ぶ	次回への意欲喚起	
4	設定した課題にチャレンジする	前時までの復習 課題提示	【活動】	【共有】	単元の総括	※

※第2時のカスタマイズ及び第4時の自由制作について4時間を超えて実施することも考えられる

## 第 1 時 貯金シミュレーションアプリを開く・読む・動かす

第 1 時においては、単元での学習内容の提示と共に、アプリのプログラムのインポート、動作の確認、プログラムの読み込みを行う。

指導過程	教師の働きかけ	備考	シート
導入	<b>【課題提示】</b> ・単元の学習内容：繰り返しの処理(と各演算子) ・制作課題：貯金シミュレーションアプリの両方を示す。	教師の発言例 「貯金シミュレーションアプリの制作を通じて、繰り返しの処理と各演算子について学んでいきましょう」	
展開 1	<b>【設計】</b> アプリの流れ(下記)をフローチャートにて示す。 1 積み立てる金額と利率について、入力された値を取得する 2 合計値を初期化 3 繰り返しの処理 (10 回) (1) 合計値に積み立てる額を加算代入する (2) 合計値に利息を計算し加算代入する (3) 年数と合計値を表示する 4 生徒の理解に沿って次を示しても良い ※ ・表示の初期化	フローチャートについては、 (例) アプリプログラミングシートのフローチャートを示す (例) 板書等で作成過程から示す (例) 生徒にフローチャートを検討させるなど、授業の実態に応じ、適宜行う。  ※ 4 については、今回の単元の学習内容となる「繰り返し処理」の本質的な部分ではないため、生徒に示す場合は留意する。	P.1 左
展開 2	<b>【実践】</b> ・Monaca にログインする ・アプリを右記 URL よりインポートする ・実際のアプリの動作を確認する ・フローチャートと対応させながら、プログラムの流れを理解する  生徒の理解に沿って、合計値の変数 amount の値を alert() で表示させるなどのインタラクティブな動作確認を実施するのもよい。	サポート Web URL <a href="https://edu.monaca.io/wakayama">https://edu.monaca.io/wakayama</a>  教師からの提示とするか、各生徒の作業とするかは、授業の実態に合わせて行う。 (例) 一連を教師が提示した後、各生徒で作業させ、動作等の確認を行う (例) 教師の提示と各生徒の作業及び確認を細かく分けて実施する	P.1 右
まとめ	本時は、繰り返しの処理(と各演算子)についてインプット中心であり、次時にはカスタマイズしていくことを示し、生徒の意欲を喚起する。	アプリプログラミングシートの配付についても、授業の実態に合わせて適宜行う。 (例) 両面刷りで本時にて配付 (例) 片面刷りで本時、次時に分けて配付	

## 第1時 留意事項

### ① 貯金シミュレーションアプリで行われる計算（複利法による計算）について

複利法とは、元金（積み立てる金額）によって生じた利息を次期の元金に組み入れる方式のこと。この方式では、元金だけでなく、それまでに生じた利息に対しても次期の利息がつき、各期の利息が次第に増加していく。

(例)	毎年積み立てる金額 10000 円	(年)利率：2%	
1 年目(期末)	元金：10000 円	利息 200 円	期末元利合計：10200 円
2 年目(期末)	元金：20200 円	利息 404 円	期末元利合計：20604 円
3 年目(期末)	元金：30604 円	利息 312.08 円	期末元利合計：30916.08 円
...			



## 第2時 アプリプログラミングシートに沿って貯金シミュレーションアプリをカスタマイズする

第2時においては、アプリプログラミングシート P.2 の内容を全生徒が共通に実施する。

指導過程	教師の働きかけ	備考	シート
導入	<p>【課題の提示】</p> <p>前時の学習内容確認</p> <ul style="list-style-type: none"> <li>・ログイン方法について</li> <li>・プロジェクトを開く方法について</li> <li>・フローチャートについて</li> </ul> <p>(積み立てる金額と、利率として入力された値を取得する → 合計値を初期化する → 繰り返し文で計算し、結果を表示する)</p> <p>本時の課題の提示</p> <p>アプリプログラミングシート(P.2)の「積み立てる年数を変更できるようにする」及び「目標の金額まで積み立てていくアプリに変更する」</p>	<p>アプリプログラミングシートの①年数の変更、②目標金額の設定のそれぞれは授業の実態に合わせ適宜行う。</p> <p>(例) 【本指導案の方法】全員で①と②を一律に実施する</p> <p>(例) 全員で①を実施し、次の時間に②を行う。完了した生徒から、他の生徒のサポートを行い、学習補助をする</p>	P.2
展開	<p>【活動①】</p> <ol style="list-style-type: none"> <li>1 「積み立てる年数を変更できるようにする」というカスタマイズ課題を提示する</li> <li>2 フローチャートについて必要な変更を考察させる(発問等で生徒に働きかける)</li> <li>3 プログラム及びHTMLの変更を実演する</li> <li>4 プログラムの動作の確認をする</li> <li>5 同様の変更を生徒が実施するよう指示し、変更及び動作確認を行わせる</li> </ol> <p>※ 机間指導を行い、理解促進を図る</p>	<p>for 文を変更する際に必要であれば書籍 P.122 も活用する。</p> <p>(初期化式/継続条件式/増減式)</p> <p>教師の提示と生徒の実習・確認をどのように組み合わせるかは授業の実態に合わせて選択する(一連を提示した後に各生徒での確認 / 提示と確認を細かく分けて実施等)。</p>	P.2 左
展開	<p>【活動②】</p> <ol style="list-style-type: none"> <li>1 「目標の金額まで積み立てていくアプリに変更する」カスタマイズ課題を提示し、この場合は for 文ではなく while 文を使うことを説明する</li> <li>2 以降は、上の【活動①】の流れに準じる</li> </ol>	<p>for 文では初期化式・増減式で変数 i を使ったが、while 文では年数を表示するための変数 i を別途宣言する必要がある。変数 i への加算処理及び繰り返しを終了するために必要な処理を while 文内に記述する必要がある。</p>	P.2 右
まとめ	<p>本時における、次の点について確認</p> <ul style="list-style-type: none"> <li>・アプリの変更の意図について</li> <li>・フローチャートの変更部分について</li> <li>・プログラムの変更について</li> <li>・アプリの動作の変化について</li> </ul>		

## 第2時 留意事項

### ① 繰り返しの記述方法 (for 文と while 文)

プログラムを作成するときには「同じ処理を何回も繰り返し処理したい」という場面がしばしば発生する。積立計算を10年(10回)行うことや一定金額に達するまで毎年積立を行うことも、繰り返し処理の一つである。

プログラミング言語には繰り返しを表現する構文が複数用意されており、代表的なものとして for 文や while 文が存在する。

繰り返しの回数が事前に決められる場合には for 文、事前の繰り返し回数が決まっておらず「特定の条件を満たすまで繰り返す」場合には while 文の使用が適している。

for 文と while 文の記述方法は以下の通りである。

#### ○ for 文の記述方法

```
for ( 初期化式; 継続条件式; 増減式 ) {  
    処理  
}
```

本アプリシート内での例

```
for ( var i = 1; i <= year; i++ ) {  
    処理  
}
```

初期化式                      継続条件式                      増減式

#### ○ while 文の記述方法

```
while ( 継続条件式 ) {  
    処理  
}
```

本アプリシート内での例

```
while ( amount < target ) {  
    処理  
}
```

継続条件式

for 文は初期化式でカウンタ変数(本課題内では変数*i*)を定義し、継続条件式で繰り返しの回数を指定する。また、増減式は繰り返しのたびに処理されるため、繰り返しのたびにカウンタ変数が1ずつ増えるように記述する。

while 文には初期化式や増減式が存在しないため、繰り返し処理の中で継続条件式を満たす状態となるように継続条件式に関する変数の値を増減させたり、変化させたりするなどの処理を記述する必要がある。

例えば、積立金額が100万円に達するまで毎年10万円ずつ積み立てるといったケースでは while 文の使用が適している。

第3時 理論(繰り返しの処理及び各演算子)を学ぶ

第3時においては、第2時までの内容について書籍(第10, 12章 / P.106-, 122-)を活用して理論的な理解を図る。

指導過程	教師の働きかけ	備考	書籍												
導入	<p>【前時の学習内容確認】</p> <p>フローチャート</p> <ul style="list-style-type: none"> <li>・入力フォームから積立額と利率を取得</li> <li>・合計値を初期化</li> <li>・繰り返し文で計算し、表示</li> </ul> <p>本時のめあての確認：プログラムの動作について理論的な視点で読む。</p>	<p>「変数」の理解についても再度確認を行う。</p> <p>教師の発言例 「前回までに作成したプログラムがどのように動いているか詳しく見ていきましょう」等</p>													
展開1	<p>【理論】</p> <p>書籍を使用し、繰り返し文について確認する。</p> <p>for文のポイント:初期化式、継続条件式、増減式</p> <table border="1" data-bbox="280 967 943 1133"> <thead> <tr> <th>式の種類</th> <th>概要</th> <th>実行するタイミング</th> </tr> </thead> <tbody> <tr> <td>初期化式</td> <td>カウンタ変数を初期化する代入式</td> <td>最初の1回だけ実行される</td> </tr> <tr> <td>継続条件式</td> <td>繰り返しを継続する条件式</td> <td>1回分の繰り返し処理の最初に毎回実行される</td> </tr> <tr> <td>増減式</td> <td>カウンタ変数の値を増減させる計算式</td> <td>1回分の繰り返し処理の最後に毎回実行される</td> </tr> </tbody> </table> <p>アプリの繰り返しの処理の中で alert(i)や alert(amount)などを表示させることも考えられる。</p> <pre data-bbox="288 1227 758 1478"> //1年ずつ計算 for (var i = 1; i &lt;= 10; i++){     alert(amount);     amount += addition;     amount += amount * rate / 100;     : </pre>	式の種類	概要	実行するタイミング	初期化式	カウンタ変数を初期化する代入式	最初の1回だけ実行される	継続条件式	繰り返しを継続する条件式	1回分の繰り返し処理の最初に毎回実行される	増減式	カウンタ変数の値を増減させる計算式	1回分の繰り返し処理の最後に毎回実行される		P. 122 -126
式の種類	概要	実行するタイミング													
初期化式	カウンタ変数を初期化する代入式	最初の1回だけ実行される													
継続条件式	繰り返しを継続する条件式	1回分の繰り返し処理の最初に毎回実行される													
増減式	カウンタ変数の値を増減させる計算式	1回分の繰り返し処理の最後に毎回実行される													
展開2	<p>【理論】</p> <p>書籍を使用し、各演算子について確認する。</p> <p>算術演算子：一般的な計算とほぼ同様の使い方であるが、文字列に対して“+”を使用すると連結になる点に注意する。</p> <p>論理演算子：本アプリでは使用しないが、if文の条件の判定において使用する。</p> <p>複合代入演算子：代入と算術演算を同時に行う処理はしばしば必要になり、複合代入演算子を使用すると簡潔に表現できることを提示する。</p>		P. 106 -112												
まとめ	<p>本時は理論のインプット中心であり、次時には生徒の自由なカスタマイズを行ってもらうことを示し、意欲を喚起する。</p>														

① parseInt ( ) について

※ やや発展的な内容かつ JavaScript 特有の事項になるため、詳細な指導・解説は省略可能。

input 要素 (入力された値) から取得した値は文字列として認識されるため、計算処理を行うには数値に変換する必要がある。

```
var addition = parseInt(document.getElementById("addition").value);
```

文字列を + で扱おうと「文字列の結合」となってしまう(右①)。

数値の加算として扱いたい場合(右②)は、文字列を数値に変換するために、parseInt 関数を用いて実行する必要がある(右③)。

①	> "2" + "2"; < "22"
②	> 2+2; < 4
③	> parseInt("2") + parseInt("2"); < 4

## 第 4 時 アプリプログラミングシートに沿って貯金シミュレーションアプリをカスタマイズする

第 4 時においては、第 3 時までの学習内容をもとに各生徒の意欲に沿って制作活動を行う。

指導過程	教師の働きかけ	備考	シート
導入	<p>【前時までの学習内容確認】</p> <ul style="list-style-type: none"> <li>・繰り返し文により一連の処理を効率的に処理が可能</li> <li>・演算子により様々な計算の処理が可能</li> </ul>		
展開 1	<p>【活動】</p> <p>アプリプログラミングシートにある下記 2 つの中からどの活動をするか選択する。</p> <p>① 積み立てる金額と利息を分けて表示されるようにしてみよう</p> <p>② 身の回りの計算をプログラムで求めてみよう</p> <p>本時においてもこれまでと同様に</p> <ol style="list-style-type: none"> <li>1 自分(たち)の作成の意図を明らかにする</li> <li>2 フローチャートで流れを考察する</li> <li>3 プログラムを記述する</li> <li>4 動作させてみる</li> <li>5 プログラムを実行し、意図したことと違う場合、デバッグを繰り返し、修正する</li> </ol> <p>という流れを大切にしよう指導する。</p>	<p>生徒の実態に合わせて活動する集団（クラス単位/グループ単位/個人単位）を決めて行う。</p> <p>活動内容についても授業の実態に合わせて適宜行う。</p> <p>（例）学級全員が①を実施する （例）原則として①を実施し、意欲のある生徒は②を実施する （例）学級全員が②を実施する など</p>	P.2 右
展開 2	<p>【共有】</p> <p>各生徒が作成したものをクラス全体、グループ内等で共有する。</p> <ul style="list-style-type: none"> <li>・自分が意図したこと</li> <li>・フローチャートのポイントについて</li> <li>・プログラムのポイント（工夫点など）について</li> <li>・実際の作品の動作、実演</li> <li>・更なる課題点や修正したい点について</li> </ul> <p>などを可能な限り共有するように指導し、他者及び自分の作品の評価を行う。</p>		
まとめ	<p>【確認】</p> <p>本単元では、貯金シミュレーションアプリの制作を通じて、繰り返し文や演算子を使用し、何度も繰り返す処理を効率的に記述する方法を学び、その方法を活用したアプリを作成したことを確認する。</p>		

## アプリプログラミングシート 指導案

配列と繰り返しを使ったアプリ(DX 図鑑)の作成

## 配列と繰り返しを使ったアプリ(DX 図鑑)の作成

### 学習目標

(知識及び技能)

- 配列を使った情報について理解できる。
- 配列の各要素に対して、繰り返しの処理を用い、データを効率的に格納する処理をプログラムにすることができる。

(思考力, 判断力, 表現力)

- 自分が扱いたい情報を構造的に捉え、配列の形で適切に表現することができる。
- 自分の意図した動作を繰り返しの処理を用いて表現することができる。

(学びに向かう力, 人間性等)

- 情報を構造的に認識し、効率的に取り扱おうと試みる。

### 前提条件

コンピュータの基本的な操作方法（クリック，ドラッグ，文字入力）ができる。

ブラウザで Web サイトを利用することができる。

Monaca 開発環境の基本的な使い方を理解している（ログイン，インポート，編集）。

HTML/CSS の仕組みや記述方法を体験している。

変数・関数・イベント・DOM の仕組みを理解している。

### 準備

Google Chrome のインストール、Monaca アカウントの作成、書籍の配付、Monaca Education 教材紹介ページ(html)の配付

授業用サンプルプロジェクト：DX 図鑑

単元の流れ 指導過程 配列と繰り返しを使ったアプリ(DX 図鑑)の作成

本指導案は最短となる4時間で作成しているが、授業の実態に合わせて4時間以上で実施してもよい。また、各時の展開方法についても一般的な一例を示したが、授業の実態に合わせて適切な方法を選択することが推奨される。

時	学習活動の概要	各時の展開				備考
		導入	展開 1	展開 2	まとめ	
1	DX 図鑑アプリを Monaca で開く 読み解く 動作させる (フローチャートを含む)	課題提示 ・学習課題 ・制作課題	【設計】 フローチャートで 検討 (繰り返しの処理 の動作を含む)	【実践】 アプリをインポート し実行する コードを読む	次回実施内容 の提示	
2	アプリプログラミングシートに 沿って、DX 図鑑アプリをカ スタマイズする	前時までの復習 課題提示	【活動】カスタマイズ 早く仕上げた個人またはグループ は、他の人の作業を助けるなど、サ ポートを行う		本時の総括	カスタマイズする ①と②をどのよ うに行うかは授 業の実態に合 わせて選択
3	理論(配列と繰り返しの処 理)について学ぶ	前時までの復習	【理論】 配列の記述につ いて学ぶ	【理論】 繰り返しの処理 を用いて配列を 扱う方法を学ぶ	次回への意欲喚 起	
4	設定した課題にチャレンジ する	前時までの復習 課題提示	【活動】	【共有】	単元の総括	

※自由制作について、4 時間を超えて実施することも考えられる。



## 第 1 時 DX 図鑑アプリを開く・読む・動かす

第 1 時においては、単元での学習内容の提示と共に、アプリの要件をフローチャートで考えた後に、アプリのプログラムの読み込みと動作の確認を行う。

指導過程	教師の働きかけ	備考	シート
導入	<b>【課題提示】</b> 単元の学習内容：配列と繰り返しの処理 制作課題：図鑑 の両方を示し、導入する。	教師の発言例 「DX 図鑑アプリの制作を通じて、配列と繰り返しの処理について学んでいきましょう」	
展開 1	<b>【設計】</b> アプリの流れ(下記 1～2)をフローチャートにて示す。 1 図鑑の配列を取得する 2 menu 要素に対し、繰り返し文を使い、図鑑の配列の各要素を追加する	フローチャートについては、 (例) アプリプログラミングシートのフローチャートを示す (例) 板書等で作成過程から示す (例) 生徒に考察させる など を授業の実態に合わせて適宜行う。	P.1 左
展開 2	<b>【実践】</b> ・Monaca にログインする ・アプリを右記 URL よりインポートする ・実際のアプリの動作を確認する ・プログラムをフローチャートと対応させながら、プログラムの流れを理解する ・プログラムの途中の要素を alert() で表示させるなどのインタラクティブな動作確認を実施する ・data.js の内容と配列の表についても示す  <pre style="border: 1px solid black; padding: 5px;">                     var data = [                     {                         title: 'CPU',                         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。',                         image: 'cpu.jpg',                         license: '&lt;a href="https://commons.wikimedia.org/wiki/File:Intel_80486DX2_bottom.jpg"&gt;Photo&lt;/a&gt;'                     },                     {                         title: 'RAM',                     }                 ]                     </pre>	サポート Web URL <a href="https://edu.monaca.io/wakayama">https://edu.monaca.io/wakayama</a>  教師からの提示とするか、各生徒の作業とするかは、授業の実態に合わせて効果的に行う。 (例) 一連を教師が提示した後、各生徒で作業させ、動作等の確認を行う (例) 教師の提示と各生徒の作業及び確認を細かく分けて実施する	P.1 右
まとめ	<b>【意欲喚起】</b> 本時だけではインプット中心であり、次時にはカスタマイズしていくことを示し生徒の意欲を喚起する。	アプリプログラミングシートの配付については、授業の実態に合わせて適宜行う。 (例) 両面刷りで本時に配付 (例) 片面刷りで本時、次時に分けて配付	

## 第1時 留意事項

### 本アプリの特徴

メニューをプログラムで動的に生成していることと、メニューをクリックすると画面が書き換わることの二つが本アプリの特徴である。メニューの動的生成は『DOM』と『繰り返し』を応用して実現している。また、クリックによる画面の書き換えは、メニューを動的生成する際に onclick イベントを記述することで実現している。

### 全体のプログラム

```
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);
  }
}
```

① for 文で  
図鑑データの  
配列 data を  
対象に繰り返  
す

② li タグ内の文字列を図鑑項目の title にする <li> CPU </li> 等

③ カスタムデータ属性の index に i (配列の何番要素か) を設定する  
<li data-index="0"> CPU </li> 等

④ li がクリック  
された時の  
挙動を定義  
(詳細下図)

### ④ onclick イベントの定義

```
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;
}
```

④-1 key に (③で設定した) 配列 data の何番要素かを取り出す

④-2 pure-drawer メニューを非表示にする

④-3 対応する図鑑項目の  
title, image 等を HTML に  
挿入する。

この onclick では関数の引数 event にてクリックされたメニュー項目の情報を受け取ることができる。この情報を利用して、対応する図鑑項目に応じた情報を画面に表示している。

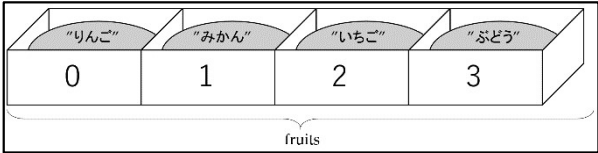
## 第2時 アプリプログラミングシートに沿って DX 図鑑アプリをカスタマイズする

第2時においては、アプリプログラミングシート P.2 の内容を各生徒共通に実施する。

指導過程	教師の働きかけ	備考	シート
導入	<p>【課題の提示】</p> <p>前時の学習内容確認</p> <ul style="list-style-type: none"> <li>・ログインの方法について</li> <li>・プロジェクトを開く方法について</li> <li>・フローチャートについて</li> </ul> <p style="margin-left: 20px;">図鑑配列を取得 → menu 要素に対し、繰り返し文を使い、図鑑配列の各要素を追加する</p> <p>本時の課題の提示 アプリプログラミングシート(P.2)の「DX 図鑑に項目を追加してみよう」</p>	<p>アプリプログラミングシートには、①項目の追加、②ルビの追加の2事例があるが、学習課題としては①が優先となるので、授業の実態に合わせて②の実践方法を調整する。</p> <p>(例) ①については全員が実施し、②については早く終わった生徒が実施する【本指導案】</p> <p>(例) 全員で①、②を実施</p>	P.2
展開1	<p>【活動】</p> <ol style="list-style-type: none"> <li>1 「DX 図鑑に項目を追加する」というカスタマイズを提示する</li> <li>2 必要な画像やテキストを用意する</li> <li>3 配列に要素を追加する</li> <li>4 プログラムの動作の確認をする 動かない場合デバッグを行う</li> <li>5 同様の改良を実施するよう指示し、動作確認を行わせる</li> </ol> <p>※ 机間指導し、理解促進を図る</p>	<p>教師の提示と生徒の実習・確認をどのように組み合わせるかは授業の実態に合わせて選択する。</p> <p>早く完成した生徒は、カスタマイズ②の「ルビの追加」に取り組むよう指示する。</p>	
展開2	<p>【確認】</p> <p>画像等の著作権については、一定の条件の下で自由に使用可能なライセンス(クリエイティブ・commons・ライセンス等)であることを確認する。</p>	<p>情報に関する法規や制度及びマナーの意義、情報モラル等に関連づけて学習することもできる。</p> <p>※アプリプログラミングシートでは、Wikimedia を使用している</p>	
まとめ	<p>【確認】</p> <p>配列を表示するプログラムができていれば、要素を追加するだけで、簡単に項目が増やせることを確認する。</p>		

### 第3時 アプリプログラミングシートに沿って DX 図鑑アプリをカスタマイズする

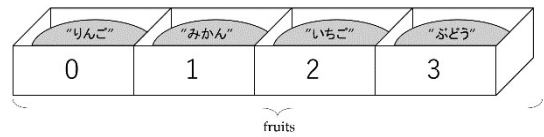
第3時においては、第2時までの内容について書籍(第11,12章 / P.114-)を活用して理論的な理解を図る。

指導過程	教師の働きかけ	備考	書籍
導入	<p>【前時の学習内容確認】</p> <ul style="list-style-type: none"> <li>・ログインの方法について</li> <li>・プロジェクトを開く方法について</li> <li>・フローチャートについて</li> </ul> <p>図鑑の配列を取得 → menu 要素に対して、繰り返し文を使い、図鑑配列の各要素を追加する</p> <p>本時のめあての確認：プログラムの動作について理論的な視点で読む。</p>	<p>必要に応じて、「変数」についての理解も確認する。</p> <p>教師の発言例 「前回までに作成したプログラムがどのように動いているか詳しく見ていきましょう」</p>	
展開 1	<p>【理論】 配列について確認する。</p> 		P. 114 -119
展開 2	<p>【理論】 繰り返し文を用いて配列を処理する方法について確認する。</p> <p>抜粋例</p> <pre>for (var i = 0; i &lt; data.length; i++) {   alert(data[i].title); }</pre>		P. 122 -126
まとめ	<p>本時は、配列及び繰り返しの処理についてのインプット中心の内容であり、次時には個人でカスタマイズしていくことを示し意欲を喚起する。</p>		

### 第3時 留意事項

#### ① JavaScript の連想配列について

JavaScript には、右図のような単純な配列(インデックスが数値)だけでなく、文字列をキーとして、アクセスできる連想配列(\*)も使用可能である。( \*文字列から値(中身)が連想可能な配列であるため)

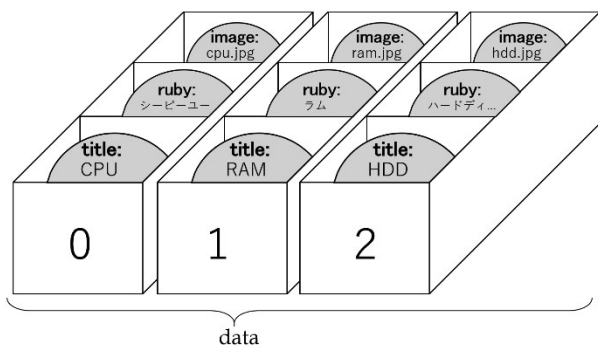


```
{
  title: 'CPU',
  explain: 'Central Processing Unitの略、日本語にすると中央演算処理装置です。演算処理を行います。',
  image: 'cpu.jpg',
  license: '<a href="https://commons.wikimedia.org/wiki/File:Intel_80486DX2_bottom.jpg">Photo</a> by Andrew Dunn / cc-by-sa-2.0',
},
```

DX 図鑑の data.js では、1つの図鑑の項目を下記のように連想配列で扱っている。

そのため、index.htmlで扱う際にも、下記のように .title、.image、.explain、.license と何を代入しているのかコードを読むだけで想像が付きやすい。

```
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;
```



連想配列を使わずに、同様のプログラムを書くと下図のように、何を代入しているのか分かりにくくなり、後でコードを修正する際に混乱する可能性がある。

```
document.getElementById('pure-toggle-left').checked = false;
document.getElementById('title').innerHTML = data[key][0];
document.getElementById('photo').src = data[key][1];
document.getElementById('explain').innerHTML = data[key][2];
document.getElementById('license').innerHTML = data[key][3];
```

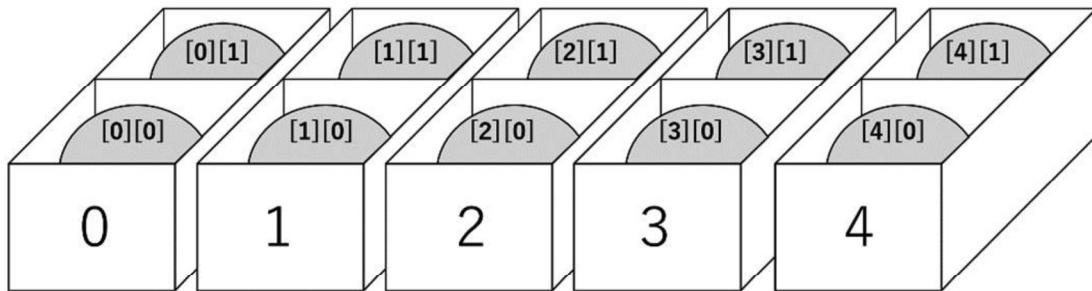
また、カスタマイズ②のようにルビ('ruby')等の項目を追加するといった変更の際も、下図のように追加した項目以降のインデックス全てを変更する必要がでてきてしまう。

```
document.getElementById('pure-toggle-left').checked = false;
document.getElementById('title').innerHTML = data[key][0];
document.getElementById('ruby').innerHTML = data[key][1];
document.getElementById('photo').src = data[key][2];
document.getElementById('explain').innerHTML = data[key][3];
document.getElementById('license').innerHTML = data[key][4];
```

## ② 二次元配列について

配列の各要素に格納する値に構造を持たせることで、より複雑な情報を取り扱うことが可能になる。

例えば、二次元配列と呼ばれるデータ構造では、配列の各要素に配列を格納することで、横×縦の2次元の表を扱うことができる。



## ③ 配列と繰り返しの処理 (for 文) について

li 要素への代入等を除いた単純な繰り返し文で、下図のように表現できる。

```
for (var i = 0; i < data.length; i++) {
  alert(data[i].title);
}
```

1 行目の for 文では data.length(=要素の個数) 回、処理を繰り返すことを、i=0 から i++(インクリメント:1 を加える操作)することで指示している。

2 行目では data[i]で図鑑のインデックス(=i)の項目を取り出している。

## 第 4 時 配列と繰り返しの理論を学ぶ

第 4 時においては、第 3 時までの学習内容をもとに、各生徒の意欲に沿って制作を進める。

指導過程	教師の働きかけ	備考	シート
導入	<p><b>【前時までの学習内容確認】</b></p> <p>・配列と繰り返しの処理を用いることにより、効率的に情報を扱うことができるアプリ</p>		
展開 1	<p><b>【活動】</b></p> <p>アプリプログラミングシートにある下記 3 つの中からどの活動をするか選択する。</p> <p>① オリジナルの図鑑アプリを考えてみよう</p> <p>② 配列と繰り返しの処理を活用したプログラムの例を調べてみよう</p> <p>③ 著作権について調べてみよう</p> <p>本時においても、これまでと同様に</p> <ol style="list-style-type: none"> <li>1 取り扱う対象の情報を構造的に認識する</li> <li>2 フローチャートで流れを考える</li> <li>3 プログラムを記述する</li> <li>4 動作させる</li> </ol> <p>※ 作成時は、著作権に留意する</p>	<p>生徒の実態に合わせて活動する集団（クラス単位/グループ単位/個人単位）を決めて行う。</p> <p>活動内容について授業の実態に合わせて適宜行う。</p> <p>（例）学級全員が①を実施する</p> <p>（例）原則として①を全員で行い、意欲のある生徒は③を実施する など</p>	P.2 右
展開 2	<p><b>【共有】</b></p> <p>各生徒が作成したものを学級全体、グループ内で共有する。</p> <p>・自分(たち)が意図したことについて</p> <p>・フローチャートのポイントについて</p> <p>・プログラムのポイント（工夫点など）について</p> <p>・実際の作品の動作、実演</p> <p>・さらなる課題点や修正したい点について</p> <p>などを可能な限り共有するように指導し、他者及び自分の作品評価を行う。</p>		
まとめ	<p><b>【確認】</b></p> <p>本単元では、DX 図鑑の制作を通して、配列や繰り返しの処理を用いることで、多くの情報を扱うアプリを効率的に作成したことを確認する。</p>		

## アプリプログラミングシート 指導案

地磁気センサーとオープンデータを使ったアプリ(地図)の作成



## 地磁気センサーとオープンデータを使ったアプリ(地図)の作成

### 学習目標

(知識及び技能)

- センサー(地磁気センサー)の値をプログラムに反映できる。
- オープンデータの有用性に気づく。
- 自分たちが普段使っているアプリでも様々なセンサーやデータが利用されていることに気づく。

(思考力, 判断力, 表現力)

- 自分の意図した通りにセンサーからの値を反映するプログラムを表現することができる。
- 自分の設定した課題に沿った形でオープンデータを選択・整形し、表現することができる。

(学びに向かう力, 人間性等)

- センサーからの情報を含め、プログラムで身近な課題を解決するよう考察できる。
- オープンデータを活用し、社会に役立つシステムを作るよう考察できる。

### 前提条件

コンピュータの基本的な操作方法 (クリック, ドラッグ, 文字入力) ができる。

ブラウザで Web サイトを利用することができる。

Monaca 開発環境の基本的な使い方を理解している (ログイン, インポート, 編集)。

HTML・CSS の仕組みや記述方法を体験している。

変数・関数・イベント・配列・繰り返しの仕組みを理解している。

### 準備

Google Chrome のインストール、Monaca アカウントの作成、書籍の配付、MonacaEducation 教材紹介ページ(html)の配付

授業用サンプルプロジェクト：地図

単元の流れ 指導過程 地磁気センサーとオープンデータを使ったアプリ(地図)の作成

本指導案は最短となる 5 時間で作成しているが、授業の実態に合わせて 5 時間以上の取り組みとすることが考えられる。また、各時の展開方法についても一例を示したが、基本編のみの展開とするなど、授業の実態に合わせて適切な展開とすることが推奨される。

時	学習活動の概要	各時の展開				備考
		導入	展開 1	展開 2	まとめ	
1	地図アプリを <ul style="list-style-type: none"> <li>┌ Monaca で開く</li> <li>├ 読み解く</li> <li>└ 動作させる</li> </ul> (フローチャートを含む)	課題提示 ・学習内容 ・制作課題	【設計】 フローチャートで 検討 (マーカーの動作 を含む)	【実践】 アプリをインポート し動かす コードを読む	次回実施内容 の提示	
2	アプリプログラミングシート (基本編)に沿って地図ア プリをカスタマイズする	前時までの復習 課題提示	【活動】カスタマイズ 早く仕上げた個人またはグループ は、他の人の作業を助けるなど、サ ポートを行う		本時の総括	カスタマイズ①ま たは②をどのよう に行うかは授業 の実態に合わせ て選択
3	アプリプログラミングシート (オープンデータ編)に沿っ てプログラムを読み、複数 地点にマーカーを立てる	複数地点の情報 扱うアプリを提示	【確認】 フローチャートで 複数のマーカーを 扱うことなどを確 認	【実践】 自分で選んだ 地点を配列に して反映	次時は、外部か らデータを取得 することを伝える	
4	オープンデータについて学 び、地図アプリに反映	地点情報のソース について考察	【理論】 オープンデータと は何かを学ぶ	【実践】 オープンデータ を変換し、アプリ に反映	次時は、自分た ちで地図アプリを 作成することを 伝える	
5	設定した課題にチャレンジ する	前時までの復習 課題提示	【活動】	【共有】	単元の総括	

※5 時間を超えて実施することも考えられる。また、自由制作を省略する場合も考えられる。

## 第 1 時 地図アプリを開く・読む・動かす

第 1 時においては、単元での学習内容の提示と共に、アプリの要件をフローチャートで考えた後に、アプリのプログラムの読み込みと動作の確認を行う。

指導過程	教師の働きかけ	備考	シート
導入	<p>【課題提示】</p> <ul style="list-style-type: none"> <li>・学習課題：センサー、オープンデータ</li> <li>・制作課題：地図アプリ</li> </ul> <p>の両方を示す。</p>	<p>教師の発言例</p> <p>「地図アプリの制作を通じて、センサーとオープンデータについて学んでいきましょう」</p>	<p>P.1 左上</p>
展開 1	<p>【設計】</p> <p>アプリの流れ(下記)をフローチャートにて示す。</p> <ol style="list-style-type: none"> <li>1 基本となる地図のレイヤーを作成する</li> <li>2 中心マーカーのレイヤーを作成する</li> <li>3 samples マーカーのレイヤーを作成する</li> </ol>	<p>レイヤーの考え方についてはアプリプログラミングシート P.1 右中段を使用し説明する。</p>	<p>P.1 右上</p>
展開 2	<p>【実践】</p> <ul style="list-style-type: none"> <li>・Monaca にログインする</li> <li>・アプリを右記 URL よりインポートする</li> <li>・実際のアプリの動作を確認する</li> </ul> <p>PC 端末での確認後、タブレット端末の Monaca デバッガーでも検証する。</p> <p>タブレットを回転させると方位の数値が変わることを確認する。</p>	<p>サポート Web URL</p> <p><a href="https://edu.monaca.io/wakayama">https://edu.monaca.io/wakayama</a></p> <p>教師の提示及び各生徒の作業については、授業の実態に合わせ効果的に行う。</p> <p>(例) 教師が一連を提示した後、各生徒で作業させ、動作等の確認を行う</p> <p>(例) 教師の提示と各生徒の作業及び確認を細かく分けて実施する など</p>	<p>P.1 左下</p>
まとめ	<p>【意欲喚起】</p> <p>次時には、最後に示したタブレットの方位の値を地図に反映させることを伝え、生徒の意欲を喚起する。</p>		

## 第2時 アプリプログラミングシート(基本編)に沿って地図アプリをカスタマイズする

第2時においては、アプリプログラミングシート P.2 の内容を各生徒共通で実施する。

指導過程	教師の働きかけ	備考	シート
導入	<p>【課題の提示】</p> <p>前時の学習内容確認</p> <ul style="list-style-type: none"> <li>・ログインの方法について</li> <li>・プロジェクトを開く方法について</li> <li>・フローチャートについて               <ul style="list-style-type: none"> <li>基本となる地図のレイヤーを作成する</li> <li>→中心マーカーのレイヤーを作成する</li> <li>→samples マーカーのレイヤーを確認する</li> </ul> </li> </ul> <p>本時の課題の提示</p> <p>アプリプログラミングシート(P.2)の「地図アプリをカスタマイズしてみよう」など</p>	<p>アプリプログラミングシートには、①中心点と拡大率の変更、②地磁気センサーの利用の2事例があるが、学習課題としては②の内容が中心となるので、授業の実態に合わせて①の実践方法を調整する。</p> <p>(例) 全員で①及び②を実施する【本指導演案】</p> <p>(例) ①については教師が示すことで時間を省略し、②を全員で行い、なるべく②に時間を割くなど</p>	P.2
展開1	<p>【活動1】</p> <ol style="list-style-type: none"> <li>1 「地図の中心と拡大率を変えてみよう」というカスタマイズを提示する</li> <li>2 プログラムの変更点を確認する</li> <li>3 緯度経度取得方法を示す</li> <li>4 プログラムの変更を実演する</li> <li>5 動作を確認する</li> <li>動かない場合はデバッグを行う</li> <li>6 同様の変更を生徒が実施するよう指示する</li> </ol> <p>※ 机間指導し、生徒の理解促進を図る</p>	<p>緯度経度は様々なwebサイトで取得可能であるが、アプリプログラミングシートでは、国土地理院地図を推奨している。</p> <p><a href="https://maps.gsi.go.jp/">https://maps.gsi.go.jp/</a></p>	P.2 左
展開2	<p>【活動2】</p> <ol style="list-style-type: none"> <li>1 「地磁気センサーで地図を回転させよう」というカスタマイズを提示する</li> <li>2 フローチャートを確認する</li> <li>3 プログラムの変更を実演する</li> <li>4 タブレット端末にて、動作を確認する</li> <li>動かない場合はPCでデバッグを行う</li> <li>5 同様の変更を生徒が実施するよう指示する</li> </ol> <p>※ 机間指導し、生徒の理解促進を図る</p>	<p>これまで変更してきた config.js を変更するのではなく、app.js を変更することに注意する。</p> <p>センサーの仕様による値の範囲(0～359度)とオープンストリートマップの仕様による値の範囲(0～<math>2\pi</math>)が異なるため、値を変換する必要がある点に注意するとともに、デバイスの向きと反対側に回転させる必要がある - 1 を乗じる必要があることに注意する。</p>	P.2 右
まとめ	<p>【確認】 本時では地図アプリの初期設定と地磁気センサーの値を反映させるカスタマイズを実施したことを確認する。</p>		

### 第3時 アプリプログラミングシート(オープンデータ編)のプログラムを読み、カスタマイズする

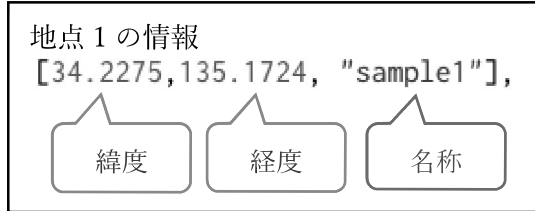
第3時においては、アプリプログラミングシート P.3 の内容を読み、複数地点へマーカーを立てる地図アプリを作成する。

指導過程	教師の働きかけ	備考	シート
導入	<p>【課題の提示】</p> <p>前時の学習内容確認</p> <ul style="list-style-type: none"> <li>・ログインの方法について</li> <li>・プロジェクトを開く方法について</li> <li>・config.js のフローチャートについて</li> </ul> <p>(中心点の緯度経度を定める →samples の緯度経度を定める →初期の拡大率を定める)</p> <p>本時の課題の提示 アプリプログラミングシート(P.3,4)の「複数地点にマーカーを立ててみよう」等</p>	<p>飲食店口コミサイトや地図アプリの検索結果などを用いて、地図に複数地点が表示されるアプリを例示することなどが考えられる。</p>	P.3 ,4
展開 1	<p>【確認】</p> <p>複数地点にマーカーを立てるフローチャートを確認する。</p> <p>config.js の samples の [ 緯度, 経度, 名称 ] という配列の全ての値を、繰り返し文を使って、地図アプリ上で扱える値に変換する必要がある。</p>		P.4 左
展開 2	<p>【活動】</p> <ol style="list-style-type: none"> <li>1 config.js の samples 変数を自分たちの紹介したい点(複数)に差し替える</li> <li>2 上記 1 の地点が収まるように center と zoom も調整する(基本編参照)</li> </ol> <p>※ 机間指導し、生徒の理解促進を図る</p>	<p>自分たちの紹介したい地点の緯度経度については前時と同様、国土地理院地図を使用する。</p> <p>samples 変数は 配列となっている。</p>	P.4 右  P.2 左
まとめ	<p>本時では、地図アプリに自分の紹介したい点を掲載した(マーカーを立てた)が、次時は、行政機関が公開している公共性の高い場所を地図内に掲載する(マーカーを立てる)ことを伝える。</p>		

### 第3時 留意事項

#### ① 複数地点にマーカーを立てるフローチャート

config.js では 地点の情報を配列の形で保持している。



しかし、地図アプリ上の位置としてこの情報を扱うためには、地図ライブラリの形式に変換する必要がある( array2feature 関数 )。

```
for (var i = 0; i < samples.length; i++) {  
  var feature = array2feature(samples[i]);  
  features.push(feature);  
}
```

今回は複数の地点を扱うため、features 配列に追加( features.push() )するようループで処理している。

#### 第4時 アプリプログラミングシート(オープンデータ編)に沿って地図アプリをカスタマイズする

第4時においては、和歌山県が提供しているオープンデータを自分たちの地図アプリに取り込む活動を通じて、『オープンデータとは、[①国や地方公共団体が公開する行政情報等を、②コンピュータで扱いやすいデータ形式で、③二次利用が可能である]という性質を持つ』ということを理解し、体験する。

指導過程	教師の働きかけ	備考	シート
導入	<p>地点情報のソース(出所)について考察するために、オープンデータの性質を示す。</p> <ul style="list-style-type: none"> <li>・[前時] 自分が列挙したもの</li> <li>・[民間] 企業が収集した飲食店情報等</li> <li>・[行政] 行政機関が収集した避難地点情報等</li> </ul> <p>のように、地点情報には様々なソース(出所)のものがあり、情報の性質により二次利用の可否も異なる。</p>	<p>[オープンデータの性質①について考察]</p> <ul style="list-style-type: none"> <li>・自分で集めた情報：自分の意思で公開/非公開を決められる</li> <li>・民間で集めた情報：外部への発信等(二次利用)は不可であることが多い</li> <li>・行政機関が集めた情報：次の展開で議論する</li> </ul>	
展開1	<p>【理論】</p> <p>コンピュータで扱いやすいデータ形式について考察する。</p>	<p>[オープンデータの性質②について考察]</p> <p>教師の発問の例</p> <p>「どういったデータ形式であれば、アプリで利用しやすいだろうか」</p>	P.3 右下
展開2	<p>【実践】</p> <p>オープンデータを変換し、アプリに反映する。</p> <ol style="list-style-type: none"> <li>1 GitHub から地点情報を扱うものを探す</li> <li>2 CSV を選択する</li> <li>3 Raw で名前を付けて保存する</li> <li>4 ダウンロードファイルを Excel で開く</li> </ol> <p>ここまでの作業後、行政機関が集めた情報が公開されていることを示す。</p> <ol style="list-style-type: none"> <li>5 Excel で対象列をコピーし、オープンデータ変換用.xlsx ファイルの「緯度」・「経度」・「名称」の各列にペーストする</li> <li>6 JavaScript の配列形式の列ができる</li> <li>7 config.js の配列 samples を置換する</li> <li>8 地図に反映されることを確認する</li> </ol>	<p>どのデータを使用するかは各学校で選択可である(アプリプログラミングシートでは「福祉避難所マップ」を使用)</p> <p>活動の単位(個人/ペア/グループ等)は生徒の実態に合わせて決定する</p> <p>一連の流れを提示してから、生徒たちに作業を行うように指示するが、授業の実態に合わせて、適宜、提示と作業のタイミングを調整しながら行う。</p>	P. 5
まとめ	<p>【確認】</p> <p>行政機関が CSV 形式で公開している情報は、様々なアプリやプログラムで扱える形に変換することができ、ユーザ(自分たち)が作ったアプリに取り込むことができること(二次利用によるメリット)について確認する。</p>	<p>[オープンデータの性質③について考察]</p>	

## 第4時 留意事項

### ① GitHub とは

ソフトウェアを開発するプロジェクトのためのソースコード管理サービス。公開されているソースコードの閲覧や簡単なバグ管理機能、SNS の機能を備えており、開発者に広く利用されている。

和歌山県においても、オープンデータをより多くのソフトウェア開発者がアクセスしやすくするため、GitHub で公開している。

### ② データ形式について

- ・ CSV 形式は多くのソフトウェアで利用可能であり汎用性が高く、表計算ソフトでも扱いやすく、二次利用しやすいテキスト型のデータ形式である。
- ・ xls 形式、xlsx 形式は、利用に特定のソフトを必要とし再利用しにくいことがある。
- ・ PDF 形式はデータ構造が再利用しにくい形式となっている。

なお、和歌山県の GitHub で公開されている他の形式 (KML 形式、SHAPE 形式、JSON 形式) は専用のソフトウェアに適した形ではあるが、表計算ソフトでの表現にはなじまないため、本指導案では利用しない。



## 第 5 時 アプリプログラミングシートに沿って地図アプリをカスタマイズする


第 5 時においては第 4 時までの学習内容をもとに各生徒の興味・関心に合わせて実施する。

指導過程	教師の働きかけ	備考	シート
導入	<p>【前時までの学習内容の確認】</p> <ul style="list-style-type: none"> <li>・地図アプリを通じて①センサーと②オープンデータを中心に緯度経度、拡大率、レイヤー等について学習したことを確認する</li> </ul>		
展開 1	<p>【活動】</p> <p>アプリプログラミングシートにある下記 2 つの中からどの活動をするか選択する。</p> <ul style="list-style-type: none"> <li>① 他のオープンデータから地図を作ってみよう</li> <li>② 自分だけのリストから地図を作ってみよう</li> </ul> <p>オープンデータについては、和歌山県だけでなく、政府のオープンデータカタログサイトの活用も考えられる。  <a href="https://www.data.go.jp/">https://www.data.go.jp/</a></p>	<p>生徒の実態に合わせて、[クラス単位/グループ単位/個人単位]での実施を決定する。</p> <p>活動内容についても</p> <p>(例) 学級全員が①を実施する</p> <p>(例) 原則とし①を全員で実施し、意欲のある生徒は②を実施するなど</p> <p>を授業の実態に合わせ、適宜効果的に行う。</p>	P.5 右
展開 2	<p>【共有】</p> <p>各生徒が作成したものをクラス全体/グループ内等で共有する。</p> <p>共有する際に、</p> <ul style="list-style-type: none"> <li>・自分(たち)が意図したことについて</li> <li>・さらなる課題点や修正したい点について</li> <li>・表示するデータのポイント(要点や意図など)についても共有するよう指導する。</li> </ul>		
まとめ	<p>本単元では、地図アプリの制作を通じて、センサーとオープンデータについて学習したことを確認する。またオープンデータの性質などについても確認する。</p>		

高等学校<共通教科情報科>  
きのくにICT教育  
プログラミング教育学習指導案

発行：2019年（平成31年） 第1版  
和歌山県教育委員会学校教育局県立学校教育課





きのくにICT教育  
高等学校〈共通教科情報科〉  
プログラミング教育  
学習指導案

発行：2019(平成31)年3月

和歌山県教育庁学校教育局県立学校教育課  
〒640-8585  
和歌山県和歌山市小松原通一丁目1番地  
TEL. 073-441-3681 / FAX. 073-424-8877