

Monacaオンライン講座テスト

～Monacaでプログラミング入門～

アシアル株式会社
アシアル情報教育研究所
岡本 雄樹



自己紹介

■名前

- ・ 岡本雄樹(アシアル情報教育研究所 所長)

■著書

- ・ イラストでよくわかるPHP
- ・ WordPressプロフェッショナル養成読本
- ・ Monacaで学ぶはじめてのプログラミング

■メッセージ

- ・ 「コンピューター」「インターネット」「プログラミング」
- ・ 私は高校生の時にそれらと出会うことで人生が拓けました。
- ・ 先生方とMonacaによるアプリ開発を通じて、情報技術の活用方法や作品作りの楽しさを広めてまいります。



アシアル株式会社について

- **2002年**

- 代表の田中正裕が本郷の地で創業（当時20才）
- PHP言語に関する雑誌発刊・教育・コンサルティング

- **2010年**

- アシアルPHPスクールのマネージャーに岡本が就任

- **2012年**

- アプリ開発ツール「Monaca」をリリース
- JavaScript言語とHTML5による複数OS向けアプリ開発環境を提供

- **2015年**

- MonacaEducation事業がスタート

- **2020年**

- アシアル情報教育研究所設立

次期学習指導要領とプログラミング

- 小学校(2020～)
 - 既存の教科の中で実施
- 中学校(2021～)
 - 技術・家庭科の中で実施
- 高校(2022～)
 - 教科「情報」で実施
 - 共通必修修科目「情報Ⅰ」が2単位(70時間)

中学校のプログラミング教育

- 現行

- プログラムによる計測・制御

- 次期

- プログラムによる計測・制御
- 「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」

高校のプログラミング教育

- 現行

- 教科「情報」は2003年度から存在する
- 原稿科目は「社会と情報」と「情報の科学」の選択必修
- プログラミングを扱うのは「情報の科学」
- 「情報の科学」は2割程度の学校が履修

- 次期

- 新科目「情報Ⅰ」に一本化されプログラミングが必ず行われる
- 更に新科目「情報Ⅱ」が選択科目として追加される
- (「総合的な探究の時間」でSTEM教育が取り入れられるため、ここでもプログラミングが行われる可能性がある。)

情報 I

情報に関する科学的な見方・考え方を働かせ、情報技術を活用して問題の発見・解決を行う学習活動を通して、問題の発見・解決に向けて情報と情報技術を適切かつ効果的に活用し、情報社会に主体的に参画するための資質・能力を次のとおり育成することを目指す。

- (1) 情報社会の問題解決
- (2) コミュニケーションと情報デザイン
- (3) コンピュータとプログラミング
- (4) 情報通信ネットワークとデータの活用

JavaScript入門



幾つかの言語と成り立ち

- **BASIC言語**
 - Beginner向けの言語として1964年に登場
- **C言語**
 - 主にOS (UNIX) 開発のため1972年代初頭に登場
- **VBA**
 - MS Office上で動作する言語として1993年に登場
- **JavaScript**
 - ブラウザ上で動くプログラミング言語として1995年に登場

言語も、何らかの課題を解決するために登場しています

JavaScriptは何故生まれたか？

- ブラウザ上でプログラムを動かしたかったから

- HTML言語だけではブラウザ上のコンテンツ(画像や表など)を静的に表示することしかできない
- JavaScriptがあれば画面に動きをつけることができる
- 外部と通信できれば (AJAX)、ブラウザ上で動作する高度なアプリケーションも開発が可能に
 - 初期の代表例はGoogle Maps
- もはやブラウザさえあればOS不要では？
 - そしてChrome OSが誕生

JavaScriptの存在意義

- **クロスプラットフォーム**

- ブラウザさえあればどのOSでも利用できる
 - 言語の文法も「Ecma」という標準化団体が標準化済み
 - 後方互換性も維持されている

- **ブラウザがOSを代替する？**

- パソコンの世界
 - 「Windows」 vs 「Mac」 vs 「Chrome OS」
- スマートフォンの世界
 - 「Android OS」 vs 「iOS」 vs 「ブラウザアプリ」



ブラウザがOSを代替すると誰が何で得する？

- アプリ開発者

- OSのストアを経由せずにアプリを配信できる
- 一つの言語で複数OSにアプリを出せる

- OSメーカー

- 他社のアプリ開発者を取り込める(かもしれない)

- 利用者

- インストール無しで気軽にサービスを利用できる

ブラウザで動作する言語

- **Flash(ActionScript言語)**
 - iOSで排除されたため、ほぼ終了
- **JavaScript以外の言語は動作不可？**
 - JavaScriptに変換すれば動く
 - 内部でJavaScriptに変換できれば動作可能
 - 変換できるものとできないものがある
 - サーバー側で実行して結果だけHTMLで返せば...
 - この方式ならどの言語でもOK
 - 厳密にはブラウザで動作していない、サーバーも必要
 - 実はWebAssemblyという仕組みの開発が進んでいます
 - 将来的には様々な言語が普通に利用できる可能性も

教育現場におけるJavaScriptのメリット

- **生徒の反応が良い**
 - 画像や音を簡単に扱える
- **学校の外でも気軽に取り組める**
 - 生徒が自宅で予習や復習しやすい
 - 研修もブラウザベースの方が組みやすい
- **教材の寿命が長い**
 - 指導案やプリントを使い続けられる

Monacaで学ぶJavaScript



Monacaの特徴

ブラウザだけで動作するクラウドIDE(統合開発環境)

The screenshot displays the Monaca web IDE interface, which is a cloud-based development environment accessible via a browser. The interface is divided into several key sections:

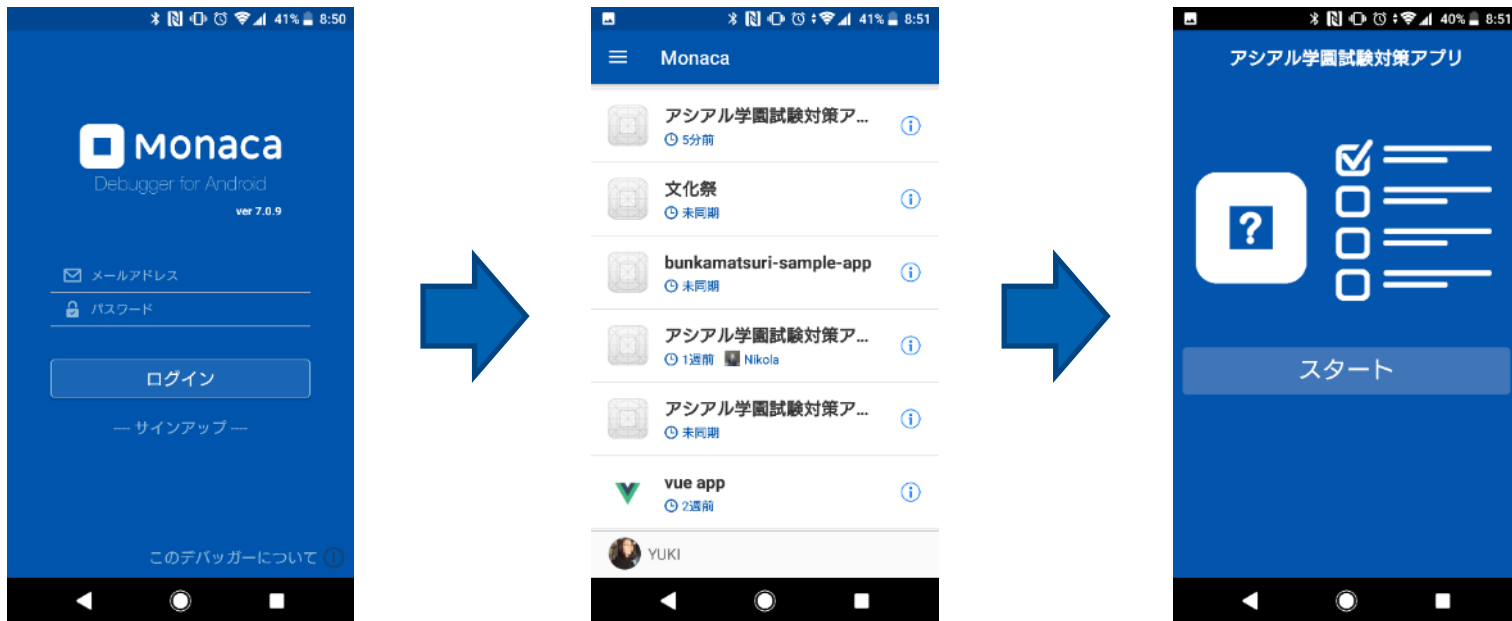
- メニューバー (Menu Bar):** Located at the top, it contains various icons and labels for file management, editing, execution, and settings.
- プロジェクトパネル (Project Panel):** Located on the left side, it shows a hierarchical view of the project files and folders, including 'node_modules', 'ros', 'www', 'assets', 'questions.csv', 'components', 'css', 'images', 'js', 'app.js', 'index.html', 'quiz.html', '.gitignore', 'config.xml', 'LICENSE', 'package-lock.json', 'package.backup.json', and 'package.json'.
- コードエディター (Code Editor):** The central area where the code is written. It shows the HTML structure of a page, including the head and body tags, and the main content area. A callout indicates that this is where file editing is performed.
- デバッグパネル (Debugger Panel):** Located at the bottom, it provides a console for viewing error messages and other debugging information.
- ライブプレビュー (Live Preview):** On the right side, it shows a real-time preview of the application running on a simulated iPhone 8 device. The preview shows a blue screen with a white question mark icon, a list of items, and a 'スタート' (Start) button.

Callouts in the image highlight the following features:

- メニューバー (ささまざまな機能呼び出せます):** Menu bar (can call out various functions)
- コードエディター (ファイルの編集を行います):** Code editor (allows editing of files)
- プロジェクトパネル (ファイルの管理などを行います):** Project panel (allows management of files, etc.)
- デバッグパネル (エラー情報などを表示します):** Debugger panel (displays error information, etc.)
- ライブプレビュー (動作確認を行います):** Live preview (allows confirmation of operation)

Monacaの特徴

Monaca for Study (アプリ) でスマホやタブレットが教材に



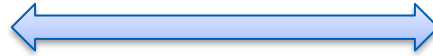
MonacaIDE上で記述したソースコードが手元で動き出す！
QRコードの読み込みも可能（2020年度より）

Monacaの特徴

外部のWebAPIやmBaasとも連携可能



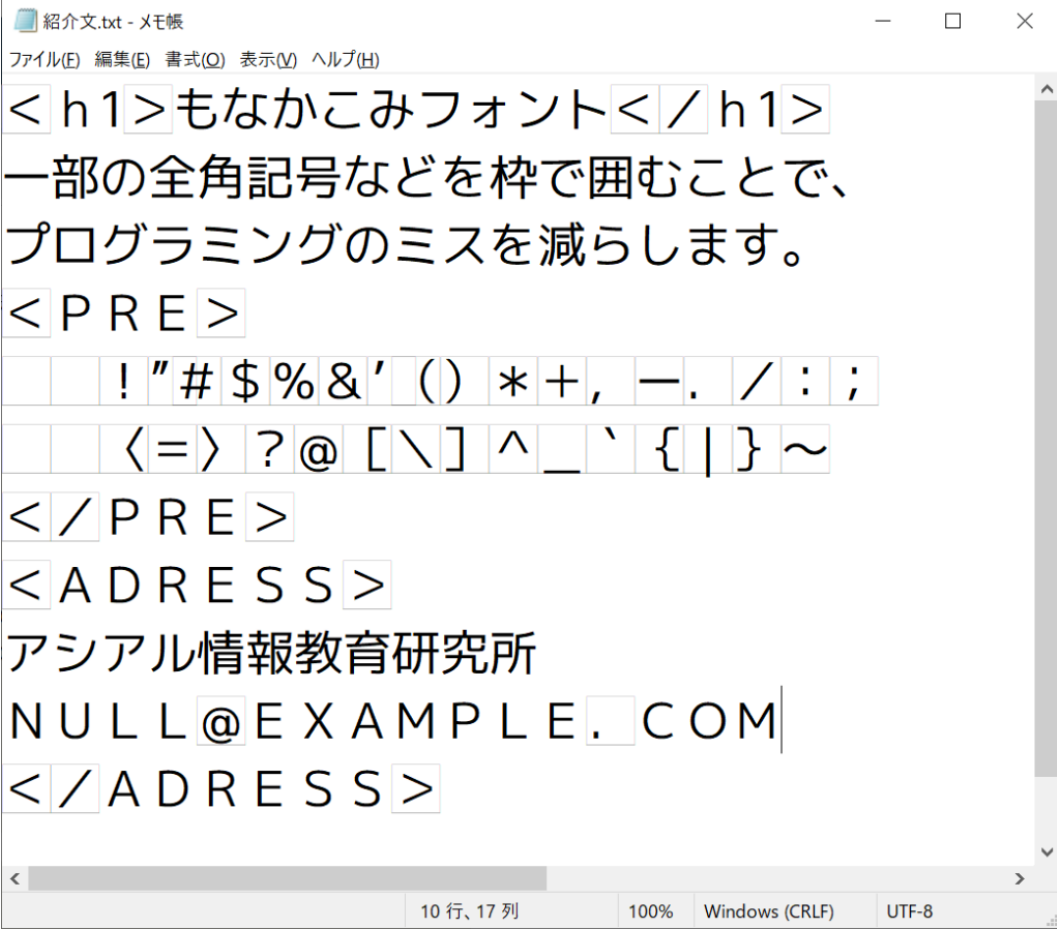
JavaScriptによる
データの送受信



- ・プッシュ通知
- ・クラウドデータベース
- ・ユーザー認証
- ・IoT連携

外部のWebAPIと連携して天気や地図情報を表示したり、mBaasと連携してクラウドデータベースの情報を参照したりすることができます。

もなかこみフォント搭載(2020年度より)



```
紹介文.txt - メモ帳
ファイル(F) 編集(E) 書式(O) 表示(V) ヘルプ(H)
< h 1 > もなかこみフォント < / h 1 >
一部の全角記号などを枠で囲むことで、
プログラミングのミスを減らします。
< P R E >
! " # $ % & ' ( ) * + , - . / : ;
< = > ? @ [ \ ] ^ _ ` { | } ~
< / P R E >
< A D R E S S >
アシアル情報教育研究所
N U L L @ E X A M P L E . C O M
< / A D R E S S >
```

10 行, 17 列 100% Windows (CRLF) UTF-8

Monacaで簡単なアプリを動かす



ダッシュボード

- 開発中のアプリはプロジェクト単位で管理します
- 新しいプロジェクトを作ると一覧に並びます



新しいプロジェクトを作る

- 今回は「ブロック崩し」を選択して作ってみましょう
- プロジェクト名は自由に設定できます

⊗ 新しいプロジェクトを作る

1 テンプレート



2 プロジェクトの情報

⊗ 新しいプロジェクトを作る

1 テンプレート
ブロック崩し

2 プロジェクトの情報

プロジェクト名
ブロック崩し

説明

作成

プロジェクトを開く

- 一覧からプロジェクトを選択して下さい
- 「クラウドIDEで開く」を選択して下さい



TIPS

無限ループなどで開けなくなったプロジェクトはセーフモードで開けます。

クラウドIDEで開く

セーフモードで開く

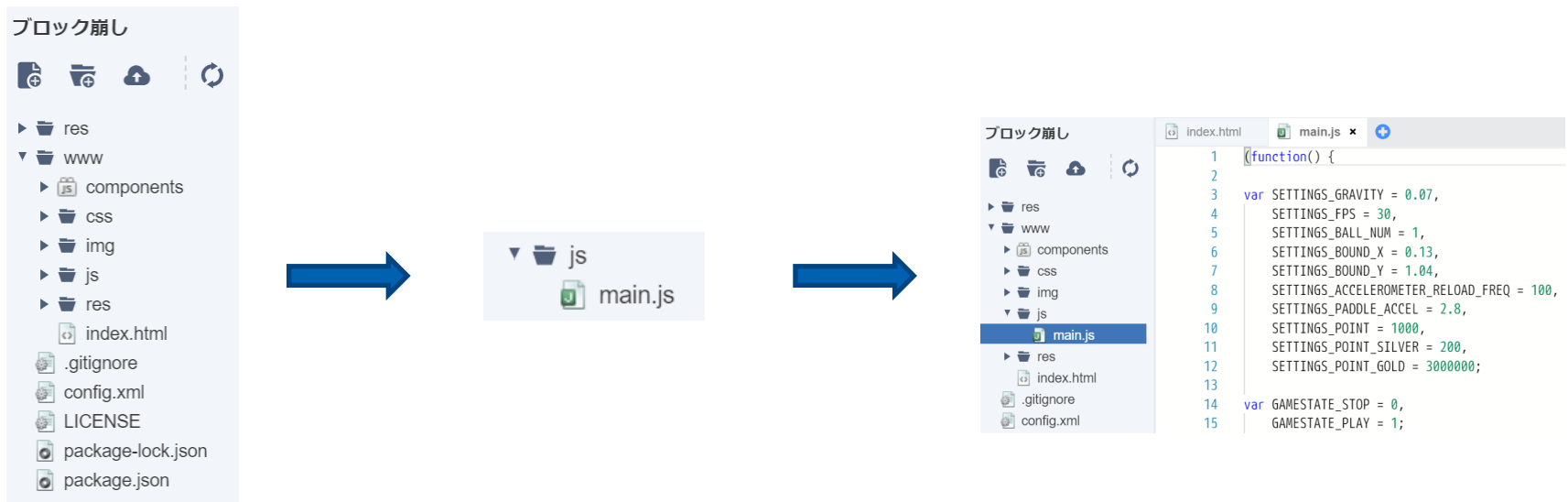
MonacaIDE(統合開発環境)

- ファイル管理・編集・動作確認などを行えます



フォルダとファイルを開く

プロジェクトパネルのファイルツリーを操作して下さい



フォルダの▶をクリック

ファイルを
ダブルクリック

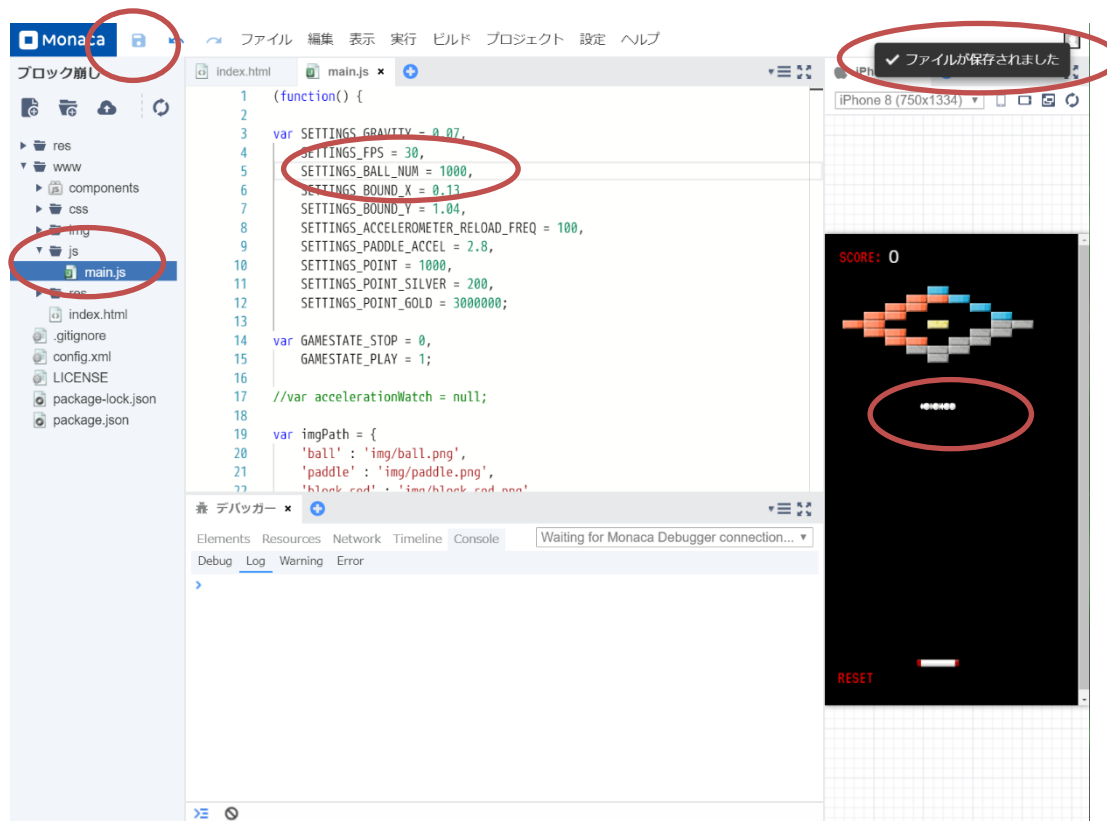
エディタ側の
タブが開きます

プログラムの変更（玉の数を増やす）

- ブロック崩しプログラムの本体は「app.js」ファイルです
- プロジェクトパネルのファイルツリーからapp.jsを探して「ダブルクリック」して下さい
- 5行目の変数SETTINGS_BALL_NUMを1000に「修正」して下さい
- app.jsの変更を「保存」して下さい。
- プレビュー上の玉の数が増えれば成功です

玉が増えれば成功

「保存」はボタンかショートカットキー (Ctrl + s)



応用：ブロックの配置を変える

- ブロックは二次元配列で管理されています
- Monaca入門のタイピングで配列の予習が可能
- 'red'や'gold'は文字なので「'」か「"」で囲む

```
index.html  main.js x +
43  setMap: function() {
44      var blockMap = [
45          [null, null, null, null, 'blue', null, null, null, null],
46          [null, null, null, null, 'red', 'red', 'blue', null, null],
47          [null, null, null, 'red', 'red', null, null, 'blue', null],
48          [null, null, 'red', 'red', null, null, null, 'blue', null],
49          [null, 'red', 'red', null, null, 'gold', null, null, 'silver', 'silver'],
50          [null, null, 'red', 'red', null, null, null, 'silver', 'silver', null],
51          [null, null, null, 'red', 'red', null, 'silver', 'silver', null, null],
52          [null, null, null, null, 'silver', 'silver', 'silver', null, null, null],
53          [null, null, null, null, null, 'silver', null, null, null, null]
54      ];
55
56      for(j = 0; j < blockMap.length; j++) {
57          for(i = 0; i < blockMap[j].length; i++) {
58              if(blockMap[j][i] !== null) {
59                  var block = BB.addBlock(10 + (30 * i), 80 + (12 * j), blockMap[j][i]);
60              }
61          }
62      }
63  },
```

応用：玉の色を変える(簡易編集)

- 画像ファイルをダブルクリックすると編集できます
- ペンで塗りつぶして下さい
- 保存後に閉じない場合は手動で×をクリック

