【Monaca問題集】オンラインランキング機能を作ってみよう！「連打ゲーム」

2017/05/17作成 (2017/05/18修正)

- Monacaデバッガー使用時(iPhone6)の画面例です

GitHub

https://github.com/natsumo/MonacaFirstApp

コンテンツ概要

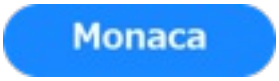
- ニフティクラウドmobile backend の機能『データストア』を学習するための問題集です
  - ニフティクラウドmobile backend の利用登録（無料）が必要です。
- 問題用プロジェクトにはオンラインランキング機能が実装されていない状態の「連打ゲーム」です
  - 既に実装済みのニフティクラウドmobile backend を利用するための準備（SDK導入など）方法の詳細はこちらをご覧ください。
    http://mb.cloud.nifty.com/doc/current/introduction/quickstart_ios.html

問題について

- 問題は２問あります
- ２問クリアすると「連打ゲーム」にオンラインランキング機能を実装したアプリが完成します
- Monaca（ハイブリットアプリ開発環境）を使用してアプリを作成します
  https://ja.monaca.io/
  - アカウントをお持ちでない場合は会員登録（無料）が必要です。
- ブラウザ環境にInternet Explorerは使用できません
  - Google Chrome（推奨）をご使用ください。

問題に取り組む前の準備

Monacaでプロジェクトインポートしてアプリを起動



▼問題用プロジェクト▼

https://github.com/natsumo/MonacaFirstApp/archive/Question.zip

1. Monacaにログインします
   https://ja.monaca.io/
2. 左上の「Import Project」をクリックします
3. 「プロジェクト名」を入力します　例）「連打アプリ」
4. 「インポート方法」の「URLを指定してインポート」をチェックし、上記リンク（問題用プロジェクト）をコピーし、貼り付けます
5. 「インポート」をクリックするとインポートされたプロジェクトが作成されます
6. 作成されたプロジェクトを「開く」をクリックして開きます
7. プロジェクトが開き、プレビュー画面が表示されます
   - プレビュー画面で遊んでみましょう！
   - Monacaデバッガーも使用可能です
     https://ja.monaca.io/debugger.html

「連打ゲーム」の操作方法

1. 「Start」ボタンをタップします
2. 「3」,「2」,「1」とカウントダウンし、「スタート！」から「タイムアップ！」の10秒間「◎」の部分がタップできるようになります
3. 10秒間の間に何回タップできるかを競う単純なゲームです
4. 10秒経つと名前を入力するアラートが表示されますので、入力し「OK」をクリックします
5. 画面に名前とスコアが表示されます
    - 4.で「キャンセル」をクリックした場合は「保存がキャンセルされました」と表示されます

※ **注意**：問題に取り組む前の状態では「ランキングを見る」ボタンをタップしてもランキングは表示されません

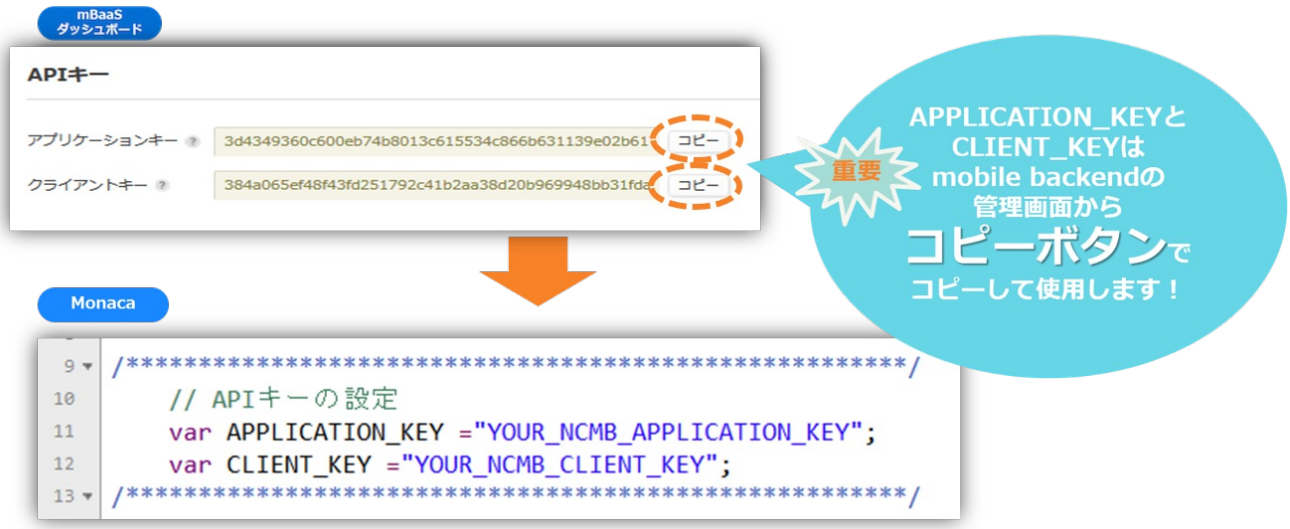
アプリの新規作成とAPIキーの設定

mBaaS ダッシュボード

- ニフティクラウドmobile backend にログインしアプリの新規作成を行います
  - アプリ名はわかりやすいものにしましょう。例）「renda」
- アプリが作成されるとAPIキーが２種類（アプリケーションキーとクライアントキー）発行されます
  - 次で使用します。

Monaca

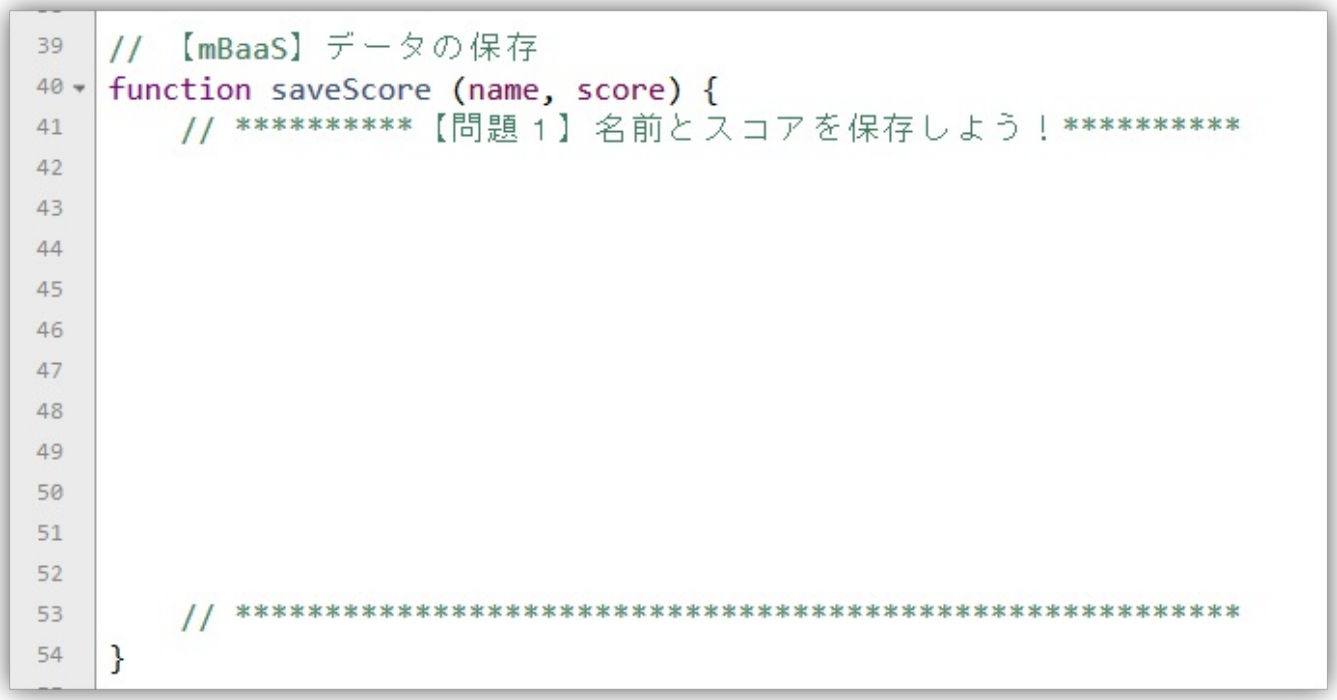
- `js/tapGame.js` を編集します
- 先程ニフティクラウドmobile backendのダッシュボード上で確認したAPIキーを、それぞれ `YOUR_NCMB_APPLICATION_KEY` と `YOUR_NCMB_CLIENT_KEY` に貼り付けます



- このとき、ダブルクォーテーション（`"`）を消さないように注意してください！

【問題１】
名前とスコアの保存をしてみよう！

`js/tapGame.js` を開きます。下図の **`saveScore`** メソッドを編集し、引数の **`name`** （アラートで入力した名前）と **`score`** （連打ゲームでタップした回数）の値をmBaaSに保存する処理をコーディングしてください



```
// 【mBaaS】データの保存
function saveScore (name, score) {
    // **********【問題１】名前とスコアを保存しよう！**********




    // ************************************************************
}
```

- データストアに保存先クラスを作成します
  - クラス名は「`GameScore`」としてください
- `name` を保存するフィールドを「`name`」、`score` を保存するフィールドを「`score`」として保存してください

ヒント

- ニフティクラウドmobile backend のドキュメントページをご活用ください
  http://mb.cloud.nifty.com/doc/current/datastore/basic_usage_monaca.html

コーディング後の作業

問題１のコーディングが完了したら、下記の作業を行います

【作業1-1】

それぞれ該当する箇所に以下の処理を追記して、実行時にコンソールにログを表示できるようにします

- 保存に成功した場合の処理を行う箇所に追記

```
// 保存に成功した場合の処理
console.log("保存に成功しました。");
```

- 保存に失敗した場合の処理を行う箇所に追記

```
// 保存に失敗した場合の処理
console.log("保存に失敗しました。エラー:" + error);
```

コンソールログの確認方法（ブラウザのコンソール表示）

- プレビュー画面の場合：【Windows】→「F12」キーまたは「Ctrl+Shift+K」、【Mac】「Command+Option+I」で表示されます
- デバッガーの場合：画面のアイコンをタップし、「!」マークのアイコンをクリックすると「App Log」画面に表示されます

※ **注意**：入力が完了したら必ず**「保存」をクリック**してプロジェクトを保存してください！Windowsの場合「Ctrl+S」、iOSの場合「Command+S」でも保存可能です。

【作業1-2】

プレビュー画面あるいはデバッガーで実行し、「Start」ボタンを押してゲームを遊びます

- 名前を入力し、「OK」をクリックすると【問題１】で作成した`saveScore`メソッドが呼ばれ、データが保存されます
- このとき下記のいずれかのログが出力されます
  - 保存成功時：「`保存に成功しました。`」
  - 保存失敗時：「`保存に失敗しました。エラー:**********`」

【問題１】答え合わせ

ニフティクラウドmobile backend上での確認

mBaaS
ダッシュボード

- 保存されたデータを確認しましょう
    - 「データストア」をクリックすると、「`GameScore`」クラスにデータが登録されていることが確認できます。

- 上図はスコアが35連打で名前を「あいうえお」とした場合の例です。

コードの答え合わせ

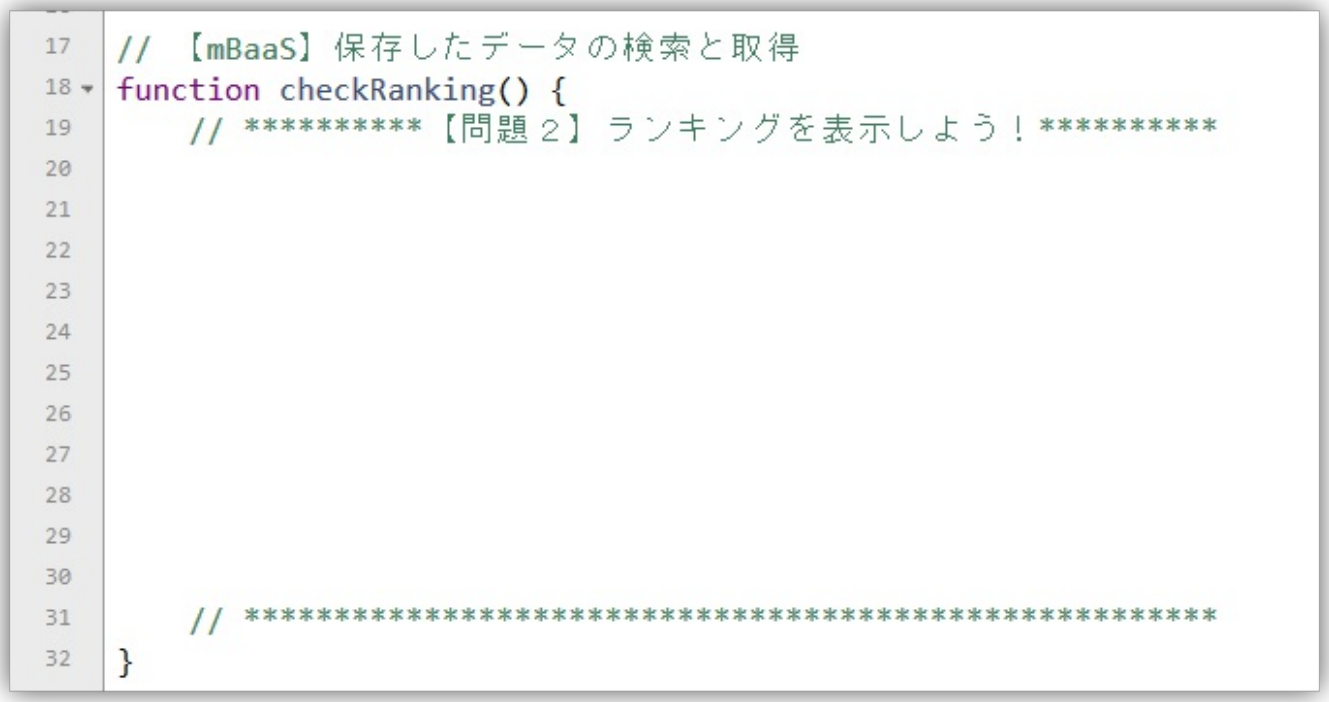
Monaca

- 模範解答は以下です

```
// **********【問題1】名前とスコアを保存しよう!**********
// 保存先クラスを作成
var GameScore = ncmb.DataStore("GameScore");
// クラスインスタンスを生成
var gameScore = new GameScore();
// 値を設定
gameScore.set("name", name);
gameScore.set("score", score);
// 保存を実施
gameScore.save()
         .then(function (){
             // 保存に成功した場合の処理
             console.log("保存に成功しました。");
         })
         .catch(function (error){
             // 保存に失敗した場合の処理
             console.log("保存に失敗しました。エラー:" + error);
         });
// **************************************************
```

【問題２】
ランキングを表示しよう！

`js/Ranking.js`を開きます。下図の`checkRanking`メソッドを編集し、データストアの`GameScore`クラスに保存した`name`と`score`のデータを`score`の降順（スコアの高い順）で検索・取得する処理をコーディングしてください



```
// 【mBaaS】保存したデータの検索と取得
function checkRanking() {
    // **********【問題２】ランキングを表示しよう！**********









    // **************************************************
}
```

- 検索データ件数は５件とします

ヒント

- ニフティクラウドmobile backend のドキュメントページをご活用ください
  http://mb.cloud.nifty.com/doc/current/datastore/basic_usage_monaca.html
  http://mb.cloud.nifty.com/doc/current/datastore/ranking_monaca.html

コーディング後の作業

問題２のコーディングが完了したら、下記の作業を行います

【作業2-1】

それぞれ該当する箇所に以下の処理を追記して、実行時にコンソールにログを表示できるようにします

- 検索に成功した場合の処理を行う箇所に追記

```
// 検索に成功した場合の処理
console.log("検索に成功しました。");
```

- 検索に失敗した場合の処理を行う箇所に追記

```
// 検索に失敗した場合の処理
console.log("検索に失敗しました。エラー:" +error);
```

コンソールログの確認方法（ブラウザのコンソール表示）

- プレビュー画面の場合：【Windows】→「F12」キーまたは「Ctrl+Shift+K」、【Mac】「Command+Option+I」で表示されます
- デバッガーの場合：画面のアイコンをタップし、「!」マークのアイコンをクリックすると「App Log」画面に表示されます

※ **注意**：入力が完了したら必ず**「保存」をクリック**してプロジェクトを保存してください！Windowsの場合「Ctrl+S」、iOSの場合「Command+S」でも保存可能です。

【作業2-2】

プレビュー画面あるいはデバッガーで実行し、「ランキングを見る」ボタンをタップします

- 画面起動後、`checkRanking`メソッドが呼ばれ、【問題１】で保存されたデータが検索・取得されます
- このとき下記のいずれかのログが出力されます
    - 保存成功時：「`保存に成功しました。`」
    - 保存失敗時：「`検索に失敗しました。エラー:******`」
- 検索の状態（成功・失敗）に関係なく、「ランキングを見る」ボタンをタップしても、まだランキングは表示されません

【作業2-3】

検索に成功したら、該当する箇所に以下の処理を追記して、取得した値から必要なデータを取り出し、ランキング画面へ反映させる`setData`メソッドを呼びます

- 検索に成功した場合の処理を行う箇所に追記

```
// テーブルにデータをセット
setData(results);
```

【作業2-4】

プレビュー画面あるいはデバッガーで実行し、「ランキングを見る」ボタンを押します

- 先ほどのスコアが表示されれば完成です！おめでとうございます★

【問題２】答え合わせ

ランキング画面の確認

Monaca

- ランキング画面を確認しましょう
    - アプリで「ランキングを見る」をタップすると以下のようにランキングが表示されます

- 上図は３回遊んだ場合の例です。複数回遊んで、ランキングが表示されることを確認しましょう！

コードの答え合わせ

Monaca

- 模範解答は以下です

```
// **********【問題2】ランキングを表示しよう!**********
// 保存先クラスを作成
var highScore = ncmb.DataStore("GameScore");
// scoreの降順でデータ5件を取得するように設定する
highScore.order("score", true)
         .limit(5)
         .fetchAll()
         .then(function(results){
             // 検索に成功した場合の処理
             console.log("検索に成功しました。");
             // テーブルにデータをセット
             setData(results);
         })
         .catch(function(error){
             // 検索に失敗した場合の処理
             console.log("検索に失敗しました。エラー:" +error);
         });
// **************************************************
```

参考

- 問題の回答を実装した完全なプロジェクトをご用意しています

▼完成版プロジェクト▼
https://github.com/natsumo/MonacaFirstApp/archive/AnswerProject.zip

- APIキーを設定してご利用ください