

マイクラフト・プログラミング応用

第2回 はじめてのプラグイン

☆はじめてのJavaScript

☆はじめてのプラグイン

2018.01.22

鎌倉シチズンネット(KCN)

© 2017-2017 Kamakura Citizens Net All rights reserved

はじめてのJavaScript(0)

- JavaScript とは

JavaScript はスクリプト言語と呼ばれるプログラミング言語の一種で、Web ページを記述する言語であるHTMLの中で使われることが多い言語です。エディタ(メモ帳)で以下のHTMLを作成し(ファイル名 hello.html)、ブラウザで表示してみよう。

```
<!doctype html>
<html lang="ja">
  <head>
    <meta charset="Shift_JIS">
    <title>はじめてのJavaScript</title>
  </head>
  <body>
    <script>
      document.write( "Hello, World!" );
    </script>
  </body>
</html>
```

← これがJavaScript の文です

はじめてのJavaScript(0.1)

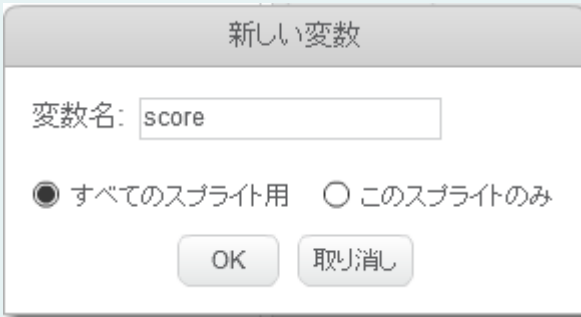



- JavaScript の解説
「とほほのWWW入門」(<http://www.tohoho-web.com/>) 中のJavaScript (<http://www.tohoho-web.com/js/index.htm>) が分かりやすいです。

基本編の「まずは始めてみよう」を読んで、「動く時計を表示してみよう」までのプログラムを作成、実行してみよう。基本編に書いてあることは理解しよう。

はじめてのJavaScript (1)



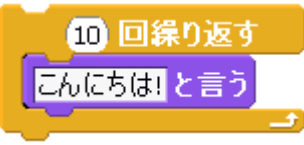




- Scratch と JavaScript (1 変数の宣言と代入文)

ScriptCraft を用いてプラグインを作るとき、JavaScript というプログラミング言語を使用します。JavaScript はHTMLと一緒に使われることが多い言語です。変数の定義と代入文の比較です。左側がScratch、右側がJavaScript です。

Scratch	JavaScript
	<pre>var score;</pre>
	<pre>score = 1;</pre>
	<pre>score = 2 + 3;</pre>
	<pre>score = score + 5;</pre>



はじめてのJavaScript (2)

- Scratch と JavaScript (2 文字列と繰り返し)

Scratch	JavaScript
	<code>message = "ハロー" + "ワールド";</code>
	<code>len = message.length;</code>
	<pre>for (var i = 0; i < 10; i++) { echo ("こんにちは"); }</pre>
   	<pre>count = 0; while (count < 10) { echo ("値は" + count); count ++; }</pre>

はじめてのJavaScript (3)

- Scratch と JavaScript (3 条件文)

Scratch	JavaScript
	<pre>if (score == 1000) { echo ("最高点です!"); }</pre> <p>(イコール「=」を2つ並べて書くことに注意)</p>
	<pre>if (score > 100) { echo ("あなたの勝ち!"); } else { echo ("あなたの負け!"); }</pre>



はじめてのJavaScript (4)

- Scratch と JavaScript (4 リスト vs 配列)

Scratch	JavaScript
	<pre>var seiseki = new Array();</pre>
	<pre>seiseki [0]</pre>
	<pre>seiseki [0] = 10;</pre>
	<pre>seiseki.length</pre>

はじめてのJavaScript (5)

- Scratch と JavaScript (4 関数)

Scratch (2.0)	JavaScript
	<pre>function shouhizei (kakaku) { var shouhizei = kakaku * 0.08; return shouhizei; }</pre>
	<pre>var zeikin = shouhizei (1000);</pre>

はじめてのJavaScript (6)

- Minecraft で JavaScript を動かしてみよう。

①Spigot サーバーを起動します

```
cd c:¥users¥{Windowsのユーザ名}¥spigot
```

```
c:¥users¥{Windowsのユーザ名}¥spigot > start.bat
```

②Minecraft を起動し、マルチプレイを選択して、サーバーアドレスとして localhost を指定し、ワールドを生成します。

③チャット欄に /js をつけて入力すると、JavaScript の文が実行されます。

```
/js 1 + 1      ----> "2" と表示される
```

```
/js var a = 1; ----> 何も表示されない
```

```
/js a          ----> "1" と表示される
```

```
/js a + 1     ----> "2" と表示される
```

```
/js var x = 'Hello ';
```

```
/js var y = 'World!';
```

```
/js x + y     ----> 何と表示されるかな？(文字列の足し算？)
```

はじめてのJavaScript (7)

④チャット欄で文字カーソル(「_」)が点滅しているとき、上矢印キー(「↑」)を入力すると以前入力した内容が表示されます。さらに上矢印キーを押すとさらにその前に入力した内容が表示されます。これを利用すれば以前入力した内容を少し修正して再入力をするなどことができます。

⑤フリー変数は値があらかじめ設定されている変数です。

__plugin、server、self、config、events がフリー変数です。

self はプレイヤーを表すフリー変数です。self はチャット欄とサーバーコンソールでのみ使用可能です。

/js self.name ---> プレイヤーの名前が表示されます。

⑥関数は一連の文をまとめたもので、関数名で呼び出すことができます。

/js echo(self, self.name) ---> プレイヤーの名前が表示されます。

/js echo(self, new Date()) ---> 現在の日付時刻が表示されます。

(new Date() でDate オブジェクトが生成されます)

/js Math.max (6, 11) ---> "11" と表示されます

/js Math.random () ---> 何と表示されるかな？

はじめてのJavaScript (8)

⑦繰り返し(P.4 参照)

```
/js for (var i = 1 ; i <= 100 ; i = i + 1) { echo(i); }
```

⑧P.7の関数(消費税を求める関数)を入力してみよう

```
/js function shouhizei ( kakaku ) { var zeikin = kakaku * 0.08; return zeikin; }
```

```
/js shouhizei ( 1000 );    ---> "80" と表示されます
```

[問題1] 消費税が10%になったら、どこをどう変えればよいかな？

[問題2] 1から100までの数の合計を計算してみよう

動きがおかしくなってしまったら、次のように入力してください。

```
/js refresh()
```

はじめてのJavaScript (9)

⑨オブジェクト

オブジェクトはプロパティ(データ)とメソッド(関数)から構成されます。

Minecraft ではプレイヤー、ワールド、ブロックなどはすべてオブジェクトです。

例: player オブジェクト

name プロパティ: プレイヤーの名前 (player.name)

chat メソッド: チャットメッセージを表示する (player.chat('Hi !'))

ScriptCraft ではJavaScript のオブジェクトとMinecraft (Spigot サーバー) のオブジェクトの両方が利用可能です。

はじめてのJavaScript (10)

⑩コメント(注釈)

「//」を記述すると、それ以降、行の終わりまでがコメントになります。

「/*」から「*/」までの間もコメントになります。

例:

```
// これはコメントです。
```

```
/* 複数行のコメントは  
   このように書くことができます。*/
```

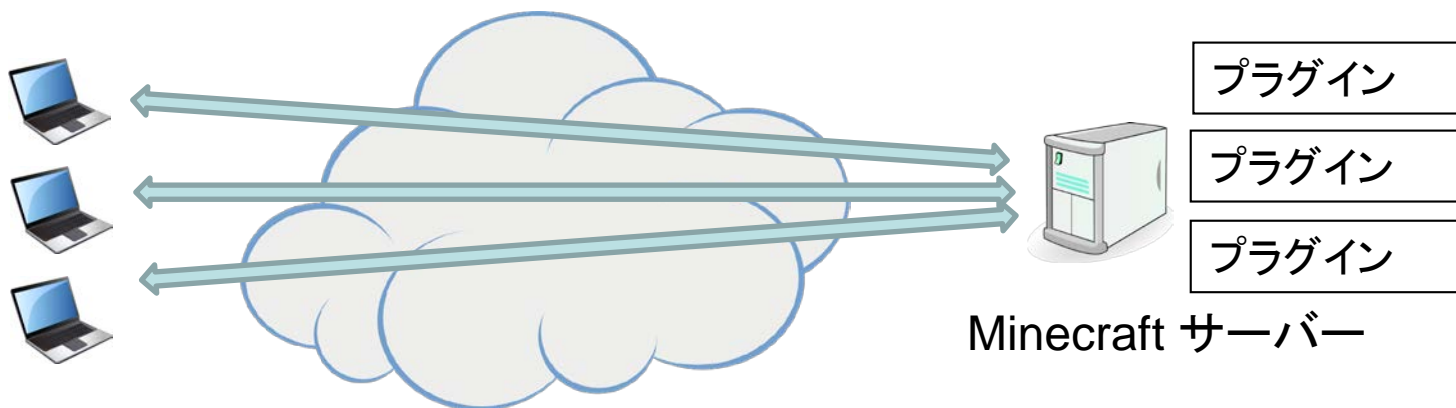
[参考] [とほほのJavaScript リファレンス](#)

はじめてのプラグイン(0)

- プラグインとは

Minecraft のプラグインは通常、サーバー側に機能を追加するソフトウェアを指し、クライアント側に機能を追加するソフトウェアをMODと呼びます。両者を総称してプラグインあるいはMODと呼ぶこともあります（Spigot はサーバーMODと呼ばれたりします）。

プラグインはサーバー側にインストールすればよく、クライアントにインストールする必要はありません。



はじめてのプラグイン (1)

- はじめてのプラグイン

下記の資料(英文)を参考にして、プラグインを作っていきます。

[The Young Person's Guide to Programming in Minecraft](#)

また、CoderDojo の次の資料(英文)も参考にしています。

[ModderDojo Athenry](#)

- ①プラグインのソースプログラムを保存するフォルダを作成します。
コマンドプロンプトで次のコマンドを入力します。

```
cd c:¥Users¥{名前}¥spigot¥scriptcraft¥plugins (注1)
```

(注1){名前} はWindows のユーザ名

```
mkdir {Minecraft で表示される自分のプレイヤー名} (注2)
```

(注2) サーバーに接続したときに表示される、xxx joined the game のxxx

はじめてのプラグイン (2)

②Windowsのメモ帳で次のコードを記述し、ファイルに保存します。

```
exports.greet = function( player ) {  
    echo( player, 'Hi ' + player.name);  
}
```

ファイル名: greet.js (ファイルの種類は「すべてのファイル」)

フォルダ: c:¥Users¥{名前}¥spigot¥scriptcraft¥plugins¥ {Minecraft で表示される自分のプレイヤー名}

上記のコードは次のように書くこともできます。

```
function greet ( player ) {  
    echo( player, 'Hi ' + player.name);  
}  
exports.greet = greet;
```

exports は関数や変数をグローバルにする(他のモジュールから参照できるようにする)ためのオブジェクトです。

はじめてのプラグイン (3)

③Minecraft (Spigot)のサーバーを起動します。

```
c:¥Users¥{名前}¥spigot¥start.bat
```

④Minecraft を起動し、「マルチプレイ」を選択、サーバーのアドレスとして、「localhost」を入力します。

⑤Minecraft のチャット欄に次のコマンドを入力します。

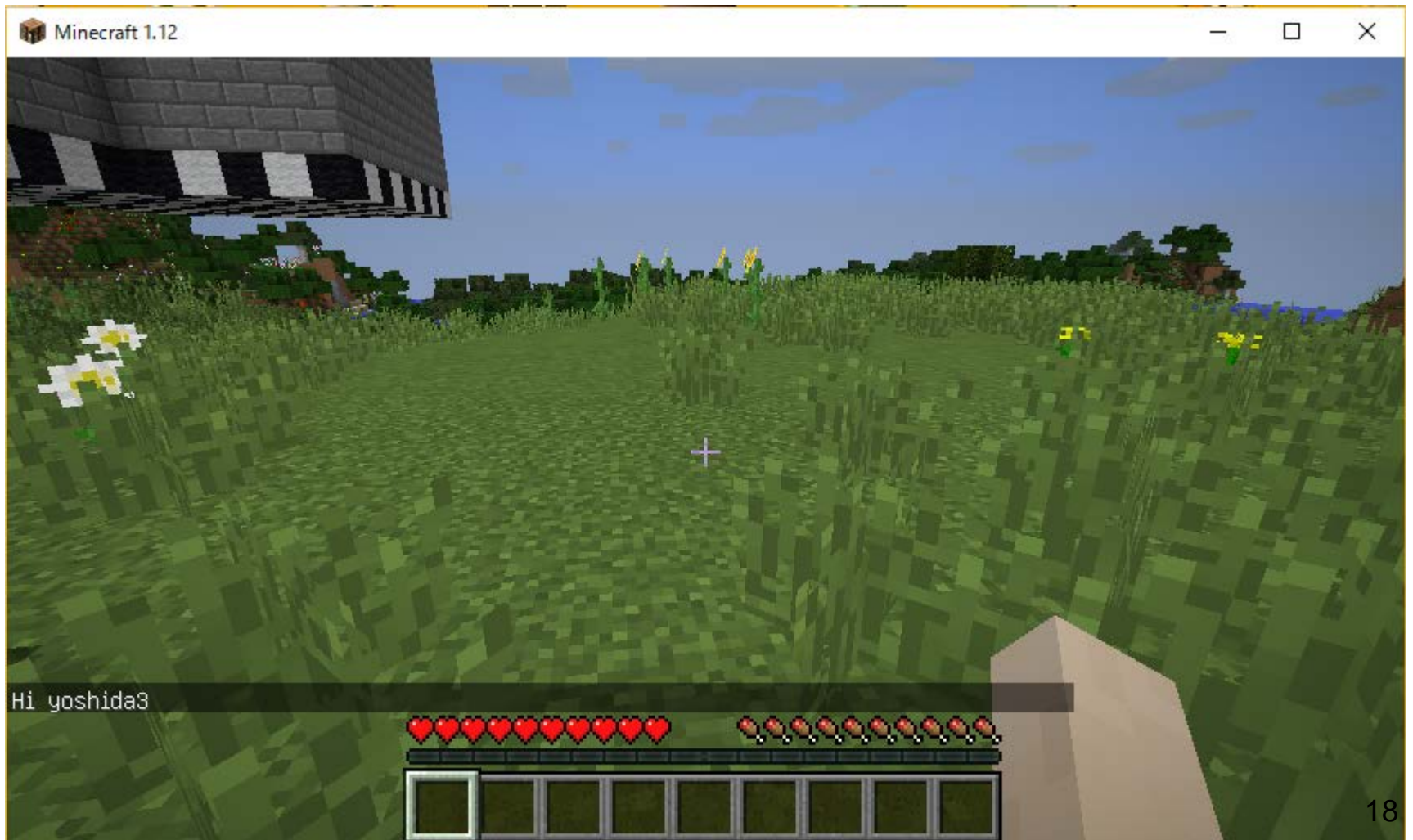
```
/js refresh()
```

```
/js greet(self)
```

次頁のように表示されたら、初めてのプラグインは成功です！

/js refresh() はプラグインのモジュールを再ロードします。プラグインのモジュールを編集したときは必ず /js refresh() を行います。

はじめてのプラグイン (4)



はじめてのプラグイン (5)

- パラメタの渡し方

greet.js で出力される「 Hi ~」のHi を任意の文字列に変更できるようにします。

例: `/js greet ('Hello ', self)`

[変更後のgreet.js]

```
exports.greet = function ( greeting , player) {  
    echo( player, greeting + player.name );  
}
```

[練習問題1] helloWorld.js

`/js helloWorld(self)` と入力したらHello World! と表示されるプラグインを作成しよう。

[練習問題2] shouhizei.js

`/js shouhizei (1000, self)` と入力したら消費税額を表示するプラグインを作成しよう。

はじめてのプラグイン (6)

- plugins フォルダと modules フォルダ
spigot のインストールフォルダの直下にあるscriptcraft フォルダの直下に、
plugins フォルダとmodules フォルダがあります。
 - ① plugins フォルダにあるモジュール(プログラム)
サーバの起動時にロードされ実行されます。(ただし、function は関数が定義されるだけで、関数が呼び出されるわけではない)
 - ② modules フォルダにあるモジュール(プログラム)
サーバの起動時にはロードされない。他のモジュールからmodules フォルダにあるモジュール利用するときは `require()` が必要です。
例: `var blocks = require('blocks');`

はじめてのプラグイン (7)

- utils モジュール

utils モジュールはScriptCraft が提供する便利な機能を集めたものです。

例: utils.players (プレイヤー全員にHi! のメッセージを送る)

```
var utils = require('utils');  
var players = utils.players();  
for (var i = 0; i < players.length; i++) {  
    echo(players[ i ], 'Hi !');  
}
```

このほか、utils.foreach、utils.array、utils.stat などがあります。

<https://github.com/walterhiggins/ScriptCraft/blob/master/docs/API-Reference.md#utilities-module>

はじめてのプラグイン (8)

- prompt と input

ScriptCraft のJavaScript には操作者に入力を求めるprompt 関数がないため、input 関数を利用します。

```
var input = require( 'input' );  
function respond ( answer, player ) {  
    echo( player, "Wow. " + answer + " that's old!" );  
}  
input ( self, ' How old are you? ', respond);
```

上記の例では操作者がデータを入力したときコールバック関数 respond が呼ばれます。input は prompt とは異なり、プログラムが入力待ちになることはありません。

はじめてのプラグイン (9)

- /jsp コマンド

/jsp コマンドは /js コマンドとは異なり、プレイヤーにオペレータ権限が無くても利用できるコマンドです。/jsp コマンドのJavaScript は次のようになります。

例:

```
var lboard = require('leaderboard');  
function leaderboard( params, sender){  
    lboard.jumps( sender );  
}  
command( leaderboard );
```

/jsp leaderboard で起動できます。

はじめてのプラグイン (10)

- ScriptCraft のAPI(アプリケーション・プログラミング・インタフェース)は下記の資料に記述されています。

[ScriptCraft API Reference](#)

また、ScriptCraft のAPI はSpigot のJavaのAPI に基づいて作成されています。

[Spigot API Reference](#)

[Spigot イベント一覧\(和文\)](#)