

コーディング規約 / 06_JavaScript

- JavaScript
 - 基本
 - 記述ルール
 - - strictモード
 - - グローバル変数
 - - var
 - - 命名規則
 - jQuery
 - - 変数宣言 (jQueryオブジェクト)
 - - メソッドチェーン
 - - イベントバインド
 - - イベントバインドと処理の分離

JavaScript

基本

- JavaScriptは動的な表現が必要な場合と、外部APIなどを使用する場合に使用する
- JavaScriptはできる限り外部ファイル化する。但し、必要に応じてHTMLファイル内に直接買い込むことも可とする
- JavaScriptプラグインなどを利用する際に、読み込みの方法に指定がある場合はそれに従う **(極力プラグインを使わない方向性で)**
- jQueryなどのJavaScriptライブラリの使用は可とする

記述ルール

可読性、メンテナンス性、汎用性確保のため、以下の通りに記述してください。

- strictモード

的確なエラーチェックを行なえる為、strictモードを使用してください。

```
// ファイルの先頭
"use strict";
```

- グローバル変数

基本的にグローバル変数の使用は禁止です。（即時関数などでローカルスコープを使用してください。）

```
(function() {  
  var hoge = 'ほげだよ';  
  function hoge() {  
    alert('ほげ')  
  }  
})();
```

- var

変数宣言時、必ずvarをつけてください。

```
var hoge = 'ほげだよ';
```

- 命名規則

変数

キャメルケースで命名してください。

```
var isTrue = true;
```

定数

大文字 + _ で命名してください。

```
var TIMER_NUM = 1000;
```

関数

キャメルケース、動詞 + 名詞で命名してください。

```
function getUserData() {  
  // 処理  
}
```

コンストラクタ

コンストラクタ（クラス）関数は、大文字で始めてください。

```
function App(name) {  
  this.name = name;  
  this.init();  
}  
  
var app = new App('maison');
```

jQuery

- 変数宣言（jQueryオブジェクト）

jQueryオブジェクトを変数にする場合、先頭に\$をつけてください。

```
var $photoItem = $('.photo__item');
```

- メソッドチェーン

処理高速化のため、メソッドチェーンを使用してください。（可読性が悪くならない程度）

```
$photoItem.find('img').addClass('is-active');
```

- イベントバインド

イベントバインドにはonを使用してください。

```
$photoItem.on('click', function() {  
    alert('click!!')  
});
```

- イベントバインドと処理の分離

イベントバインドと処理はわけてください。

```
$objA.on('click', function() {  
    funcA();  
});  
$objB.on('click', function() {  
    funcB();  
});  
.  
.  
.  
function funcA() {  
    // objAの処理  
}  
function funcB() {  
    // objBの処理  
}
```

添付ファイル (0)



白藤 卓也によって 2020/10/26 19:15:49 に作成されました。