

# Code Match リターンズ

脱コピペ！プロが教えるJavaScript  
- パララックス・ページトップボタン編 -

2014.11.11 (株)TAM 松尾浩志

# 自己紹介

## 株式会社TAM 松尾

### 主な担当

- PHPなどのサーバサイド
- JavaScript などのフロントエンド
- HTML5 / CSS3

### パートナー型デジタルプロダクション

東京・大阪・シンガポールに111名

<http://www.tam-tam.co.jp/>



# 今日の内容

「HTML/CSSは分かるけれどJavaScriptはこれから、または独学・コピペ」という方対象

- JavaScript / jQuery 入門
- 後半は実際にみんなでコードを書いてみる  
ハンズオンです

# もくじ

1. jQueryとは？ jQuery のメリット

2. jQuery の書き方、文法

3. 開発環境について (ここまで30分ほど)

休憩 (10分)

4. ハンズオン(1) ページトップボタンを作る (30分)

5. ハンズオン(2) パララックススクロールを作ってみよう  
(45分)

jQueryとは？

# jQuery とは?

- ・ JavaScript のオープンソース・ライブラリ (便利パーツ集) です
- ・ jQuery自体、JavaScript で書かれています
- ・ つまり、



(JavaScriptでできないことはjQueryでもできません)

# JavaScriptとは？

ブラウザ上で動くスクリプト言語

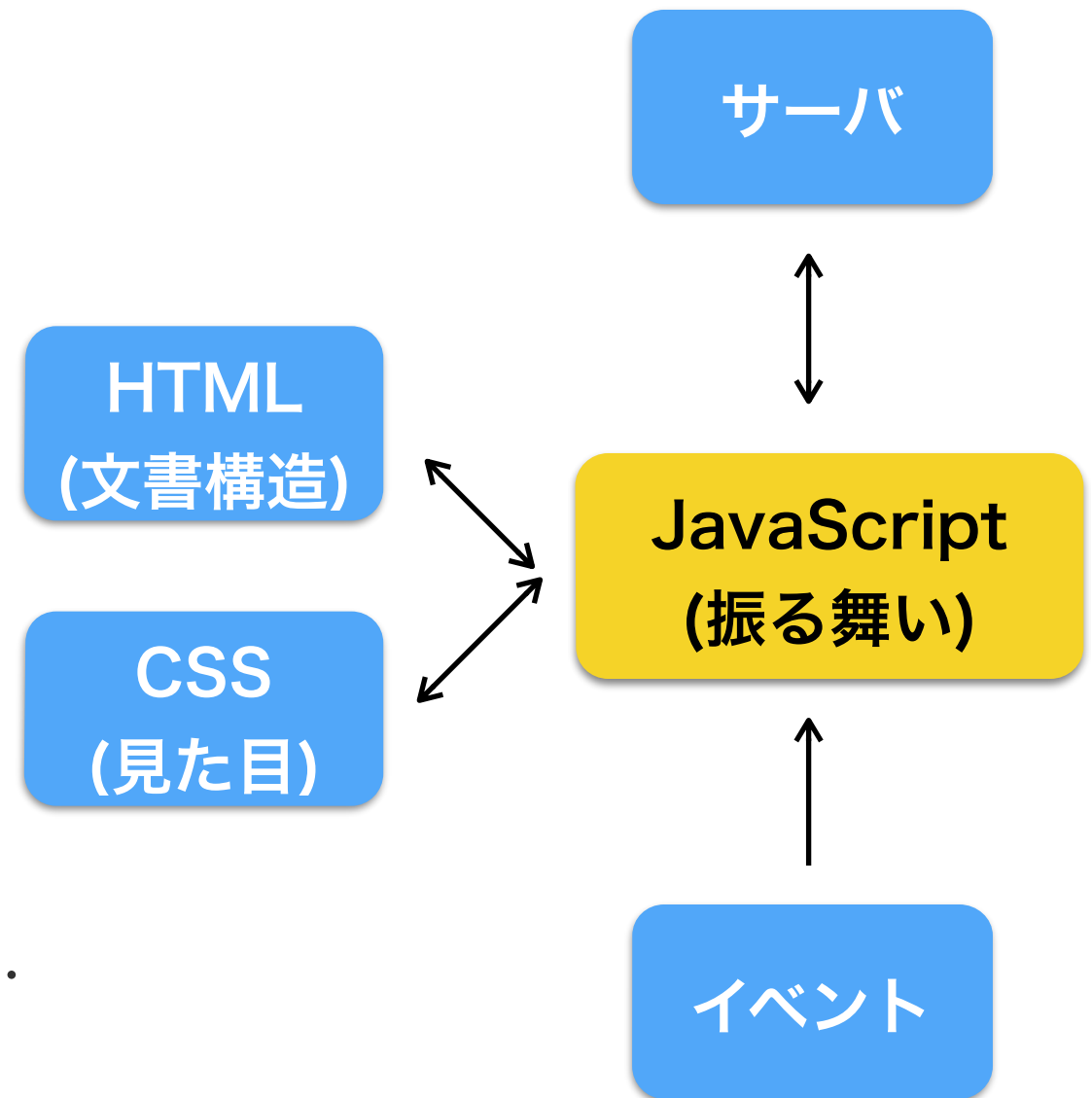
CSSを書き換えたり…

マウス操作イベントを処理したり…

アニメーションを実現したり…

サーバとデータのやりとりをしたり…

HTML + CSS だけではできない、**動的なページ**が作れます



# jQueryのメリット



# jQueryのメリット

- **write less, do more.**
  - 短く書ける！
  - CSS経験があると馴染みやすい
  - クロスブラウザ対応
  - プラグインが豊富



ラク！！

# 短く書ける！

`<img id="logo" />` を非表示にするサンプル

JavaScript の例:

```
document.getElementById('logo').style.display = 'none';
```

jQuery の例1:

```
$('#logo').css('display', 'none');
```

jQueryの 例2:

```
$('#logo').hide();
```

# 短く書ける！！！！！！

`<img id="logo" />` を2秒かけてフェードアウトさせるサンプル

## JavaScript の例:

```
var opacity = 100;
var step = 5;
var duration = 2000;
var fadeOut = function() {
    var element = document.getElementById('logo');
    opacity -= step;
    if (opacity <= 0) {
        element.style.display = 'none';
        return;
    }
    element.style.opacity = opacity / 100;
    element.style.filter = "alpha(opacity:"+opacity+")";
    setTimeout(fadeOut, duration / (100 / step));
}
fadeOut();
```

## jQuery の例:

```
$('#logo').fadeOut(2000);
```

# いろいろできます

`show()` `hide()`

`text()` `html()` `css()`

`toggle()`

`attr()` `val()`

`fadeIn()` `fadeOut()` `fadeTo()`

`addClass()` `removeClass()`

`slideDown()` `slideUp()`

`append()` `prepend()` `wrap()`

`animate()`

`after()` `before()`

`width()` `height()`

`find()` `parent()` `children()`

便利メソッドが覚えきれないほどあります、くわしくは公式ドキュメントへ。

# CSSっぽく書ける！

`$('#id')` ← ID を選択

`$('.class')` ← クラスを選択

`$('ul#id > li.class')` `$('h1, h2')` ← CSSと同じかんじ

`$('li:first')` `$('li:eq(3)')` ← jQuery独自の記法もあります

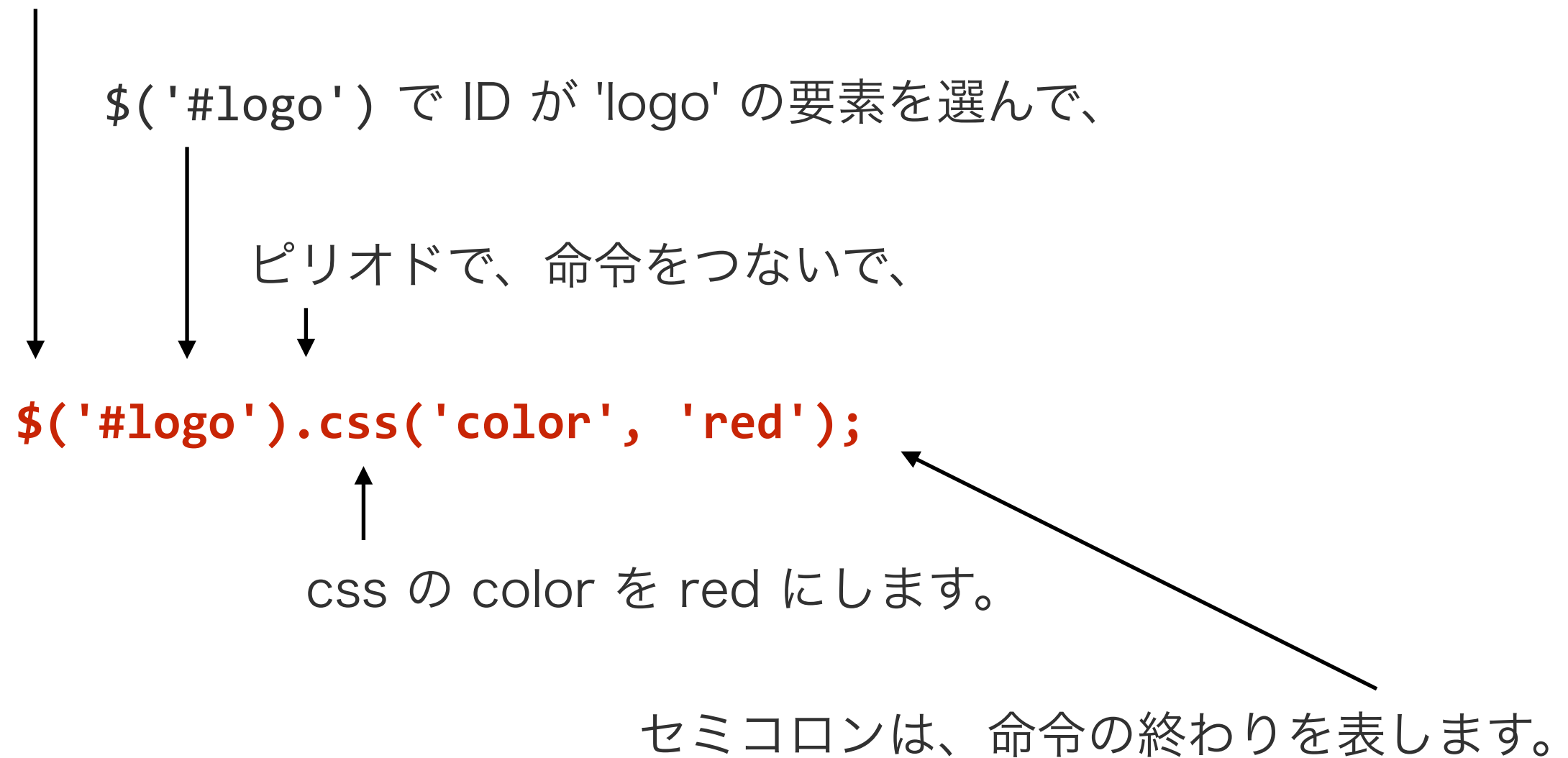
`$('input[type=checkbox]')` `$('input:not(:checked)')`

`$('CSSのセレクタ似')` なので、CSS を知っている人にはなじみやすい！

# コードの書き方、文法

# 使用例 (1)

jQuery の命令は、かならず「\$」ではじまります。



# 使用例 (2)

htmlを書き換える例：

```
$('.sample').text('好きな文字列');
```

```
<p class='sample'>ダミーの文字列</p>
```



```
<p class='sample'>好きな文字列</p>
```



# 使用例 (3)

cssを書き換える例：

```
$('.sample').css({  
    color: 'red',  
    display: 'block',  
    width: '100%'  
});
```

```
.sample {  
    color: red;  
    display: block;  
    width: 100%;  
}
```

JavaScript と CSS の文法の違いはありますが、**ほとんど一緒！**

# 使用例 (4)

ロゴをクリックしたら、ナビがフェードアウト

```
$('#logo').click(function() {  
    $('#nav').fadeOut();  
});
```

**function(){ ~~~ }** は処理のひとかたまり、ひと纏まりを表します。

`function(){ }` 内のコードはすぐには実行されず、呼び出されたタイミング（この場合は `#logo` のクリック時）にはじめて実行されます。

# 基本はこれだけ、 あとは組み合わせ（応用）

- ・ **HTML や CSS を書き換える**
- ・ **なにかイベントが起きたら、処理を実行する**  
(例：マウスのクリックやホバー、画面のスクロールなどが  
起きたら、function(){ ～ } 内のひとかたまりの処理を実行)

※ 他にはサーバとのAJAX通信や、一定時間おきに処理する、  
などのパターンもある

# JavaScriptの基本的な文法は 知っておく必要があります

- ・ jQuery は JavaScript のライブラリ、便利パーツ集
- ・ 応用には、**結局 JavaScript の文法を知らなければならない**
- ・ jQuery入門のサイトや本などでは、  
JavaScript の文法の基礎は詳しく書かれていないことも多い
- ・ JavaScript の土台の上に jQueryがあります  
「JavaScript 入門」的な本やサイトも見てみてください

# 応用の入門

- ・ プログラミングの基本は、だいたい以下の4つ
  - ・ 変数、数字や文字列の扱い
  - ・ if 文（条件判定）
  - ・ 配列とループ
  - ・ 関数（function）
- ・ このへんが理解できれば、あとはパズルのように組み合わせてだいたいことができます

# メソッドチェーン

同じものに対する命令を、ピリオドでつなげて一度に書けます

```
$('#sample').css('color', 'red');
```

```
$('#sample').text('テキスト');
```

```
$('#sample').fadeIn();
```



```
$('#sample').css('color', 'red').text('テキスト').fadeIn();
```

書きやすい、処理速度も速い、コードがわかりやすい。

# 開発環境について

# ブラウザとエディタだけ

- JavaScript は、ブラウザ上で動きます  
他の言語と違い、複雑なサーバ設定やインストール作業はいりません
- エディタはお好みで、ただし JavaScript の文法チェックをしてくれるものがおすすめ！（JSHintプラグインなど）



- 動作テストも、ブラウザ内に便利な開発ツールが備わっています

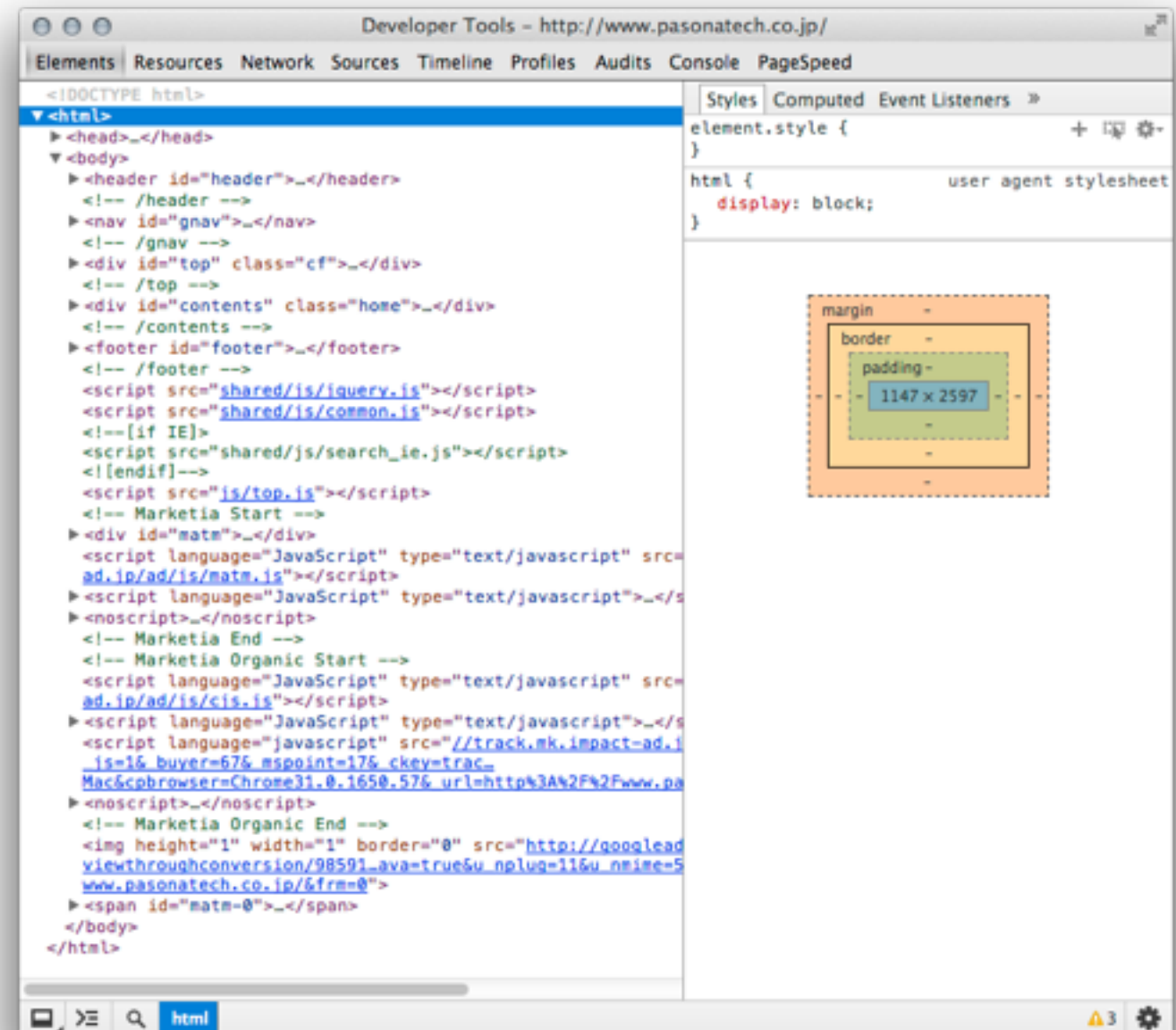


# 開発ツールべんり

## Chrome Developer Tool の例：

Chrome内蔵の開発ツール。  
HTML も CSS もその場で  
表示しながら編集できます。

JavaScript の動作を  
途中で止めたり、  
実際にコードを入力して  
テストできます。

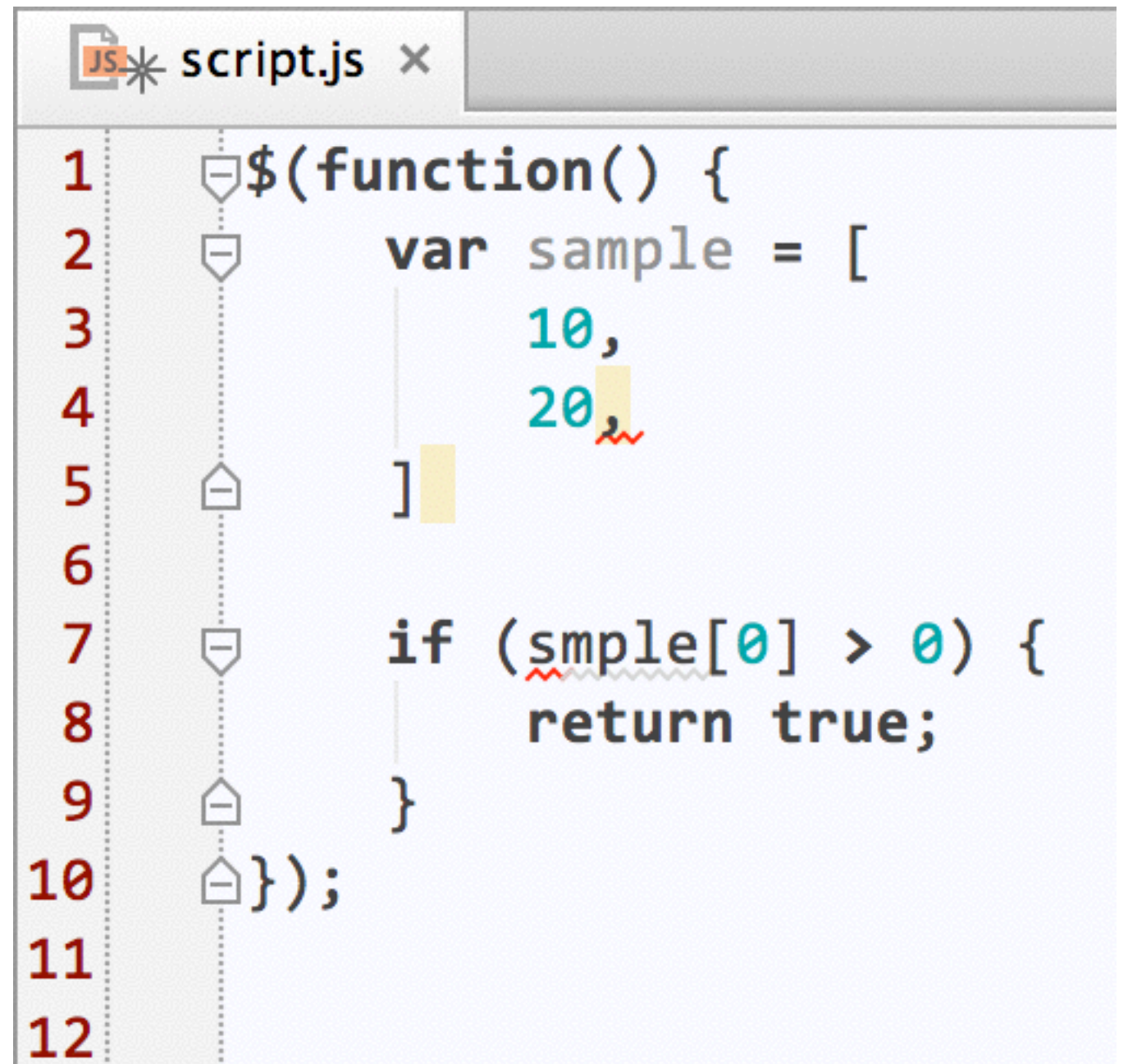


# 文法チェックべんり

## 文法チェックの動作例

エラーが 4つ、  
見つかっています

コードを書いている途中に  
ミスに気づけるので  
とても効率的です。



```
1 $(function() {
2   var sample = [
3     10,
4     20,
5   ]
6
7   if (sample[0] > 0) {
8     return true;
9   }
10  });
11
12
```

# 開発をはじめる

ブラウザとエディタだけあれば特別なセットアップはいらず、いきなり開発をはじめられます。

1. jQuery 本体を [jquery.com](http://jquery.com) からダウンロード
2. index.htmlファイルから jQuery を呼び出す
3. JavaScript を `<script>` タグ内(または外部jsファイル)に書きはじめる
4. ブラウザで実行して動作確認
5. Developer Tool でデバッグ

# jQuery のダウンロード

- ・ <http://jquery.com/> からダウンロード
- ・ jQuery にはバージョン1系 とバージョン2系があります

- ・ 1系：IE6以降に対応

- ・ 2系：IE9以降に対応  
(そのぶん軽量で高速)

- ・ 2系はIE8に対応していないので、案件に応じて選択を



# index.htmlから呼び出す

```
<html>
```

```
<head>
```

```
</head>
```

```
<body>
```

```
  ～～
```

```
    <script src="jquery-1.11.1.min.js" type="text/javascript"  
></script>
```

```
</body>
```

```
</html>
```

基本的には </body> の直前に記述します。

# 書きはじめる

<body>

～～

<script src="jquery-1.11.1.min.js" type="text/javascript"></script>

**<script>**

**\$(function() {**

**ここにコードを書く**

**});**

**</script>**

</body>

DOMの解析（HTMLの要素の展開）が終わった段階でコードが実行開始される

※ css同様、JavaScript ファイルは外部呼び出しにするのがセオリー

ハンズオン

# まずは体験、 サイトを書き換えてみよう

- ・ パソナテックさんのサイトを Google Chrome で開いてみる
- ・ Chrome Developer Tool のコンソールを開く  
**mac: [cmd] + [opt] + [J]**  
**windows: [ctrl] + [shift] + [J]**

```
$( 'body' ).fadeOut();
```

```
$( 'body' ).fadeIn(5000); ← 単位はミリ秒、5000ミリ秒 = 5秒
```

```
$( '.tagline' ).text( 'jQueryの勉強をします!!!' );
```

```
$( '.tagline' ).text( 'こんにちは！' ).css( 'color', 'red' );
```



# パララックス他のハンズオン

今回は

- ・ パララックス
- ・ TOPへ戻る

をやります



Code match

対象 特長 謝辞

こんな方を応援します。

- これからのサイト制作を一気通貫で行いたいデザイナーの方
- JavaScriptを学んで、コーディングスキルの幅を広げたいエンジニアの方
- JavaScriptを学んで、IoT(Internet of Things)な開発をしたいとお考えの方

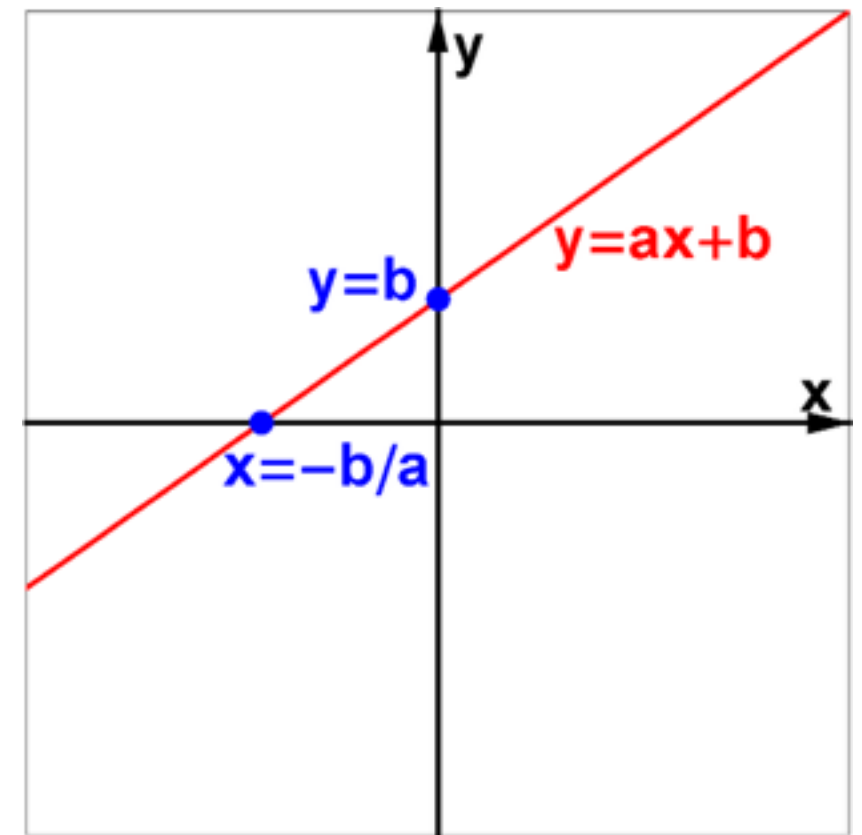
Code matchの特長

就業までをトータルサポート      ハンズオン含め無料      就業特典でさらにトレーニング

TOPへ戻る

# パララックスはややこしい？

- ・ 実はまじめにやると、一次関数 ( $y = ax + b$ ) とか数学っぽい考え方がでてくる
- ・ スクロール量に応じて変化を与える、という処理になる
- ・ まずは「トップに戻る」ボタンをやってみます



# コードのダウンロード

- ・ <http://bit.ly/codematch2014>

右下の “Download ZIP” から一式をダウンロードして、適当な場所に展開

- ・ <http://jquery.com/> から jQuery をダウンロード、/js フォルダに入れる
- ・ index.html の `</body>` の直前に以下を記述

```
<script src="jquery-1.11.1.min.js"></script>
```

```
<script>
```

```
$(function() {
```

```
});
```

```
</script>
```

# ハンズオンのカンペ (1)

クリックしたら処理を行う

```
$( '#id' ).click(function() {  
    // クリック時の処理  
});
```

スクロールのアニメーション

```
$( '#id' ).animate({  
    scrollTop: 上から何px?  
}, {  
    duration: 時間(ミリ秒)  
});
```

トップへ戻るボタンは最初には表示しないので、CSSで消しておく

スクロールしたら処理を行う

```
$(window).scroll(function()  
{  
    // スクロール時の処理  
});
```

フェードインさせる

```
$( '#id' ).fadeIn( ミリ秒 )
```

# ハンズオンのカンペ (2)

if文で条件により処理を分ける

```
if (条件を書く) {  
    // 条件が合ったとき  
}  
else {  
    // 条件が合わないとき  
}
```

今どれだけスクロールしたか？

```
$(window).scrollTop()
```

アニメーションのキャンセル

```
$('#id').stop();
```

各種ポジション取得

```
$('#id').position().top  
$(window).height()  
$(document).height()
```

変数宣言、文字と変数の連結

```
var name, var;  
test = "あ" + name + "い";
```

# ハンズオンのカンペ (3)

背景画像の位置を変える

```
$('#id').css({  
    "background-position": "center " + y + "%"  
});
```

パララックススクロールの公式（オレ流）

$$\frac{\text{scrollTop} - x1}{x2 - x1}$$

```
y = ($('#window').scrollTop() - x1) / (x2 - x1) * 100;
```

※ パーセントの場合、最後に100を掛ける

and more

# 困ったらまずは公式サイト

- ・ オフィシャルのドキュメント (英語)
- ・ <http://api.jquery.com/>
- ・ jQuery はググると情報が多い！  
でも日本語サイトだと情報が古い場合があったり、よいサンプルコードでなかったり… (もちろん情報が多いのはありがたいことです)
- ・ まずは、英語だけど、公式ドキュメントを一度は見よう



# プラグインの探し方

- ・ jQuery は便利なプラグインがたくさんあるので、有効活用しよう

- ・ **jQuery オフィシャル Plugin Registry**

<http://plugins.jquery.com/>

- ・ **jQuery UI**

<http://jqueryui.com/>

- ・ **GitHub** <http://github.com/>



- ・ ググればいっぱいいてくる。単にプラグインを紹介しているサイトよりも、実際に使ってみてデモなどを公開しているサイトの方が頼りになる
- ・ 実際にサイトのコードを開き、どんなプラグインを使っているか見る

# 人のコードを参考に

- ・ ブラウザの「ソースを見る」から、人の書いたコードは参照できます
- ・ GitHub などにもコードはたくさん公開されています。
- ・ コードの書き方には「これだ！」という正解はないことが多く、他人のコードを見ることはとても参考になります。
- ・ また、jQuery だけでなく、JavaScript の知識も深めてください。
- ・ 詳しい人に自分のコードを見てもらうのも、たいへん勉強になります！

ありがとう  
ございました