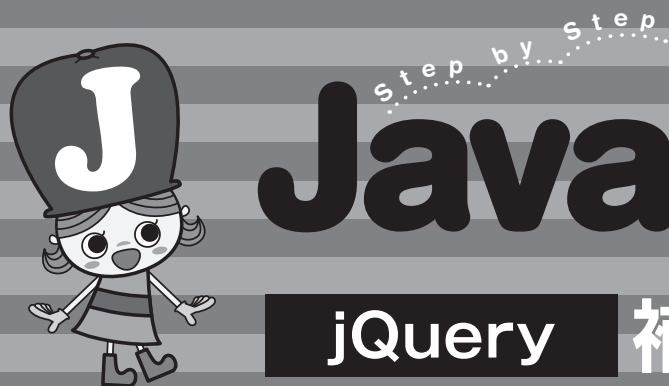




JavaScript

jQuery 補足資料

この補足資料は、「Chapter 3 次のステップのために」の「Section2 ライブラリを知ろう」(P87～)を学習する際の補足資料としてご利用ください。

「ライブラリを試してみよう」(P89)を学習する前にお読みいただくことで、jQueryを利用する目的やjQueryの基礎知識を整理できます。ぜひ、ご活用ください。

■補足資料目次

- jQueryとは……………P2
- jQueryの基礎知識……………P2
- jQueryの簡単な記述例……………P3

■補足資料の内容

- jQueryとは…………ライブラリ、フレームワーク、jQueryの意味を解説します
- jQueryの基礎知識…………jQueryを使用する際に最低限必要な知識を解説します
- jQueryの簡単な記述例…………サンプルソースで、記述方法の仕組みを解説します

※この補足資料は、テキスト「Step by Step JavaScript」内「Chapter 3 次のステップのために」の学習を行う際のjQueryについての理解を深めるための学習の参考として制作しており、「ライブラリを試してみよう」(P89)を学習する前にお読みいただくことをお勧めいたします。

※この補足資料は、2012年9月1日現在の情報により作成しております。

※この補足資料の著作権は株式会社クリーク・アンド・リバー社に帰属いたします。無断複写、転載、営利利用はできません。

jQuery とは

ライブラリとは、JavaScript でよく利用される技術をあらかじめ作成してまとめたようなイメージとして考えることができます。実行させたい目的によって多数のライブラリが存在します。ライブラリには、目的とする JavaScript の記述がまとめられているので、その部分をライブラリに任せてしまえば、自分の記述するソースコードもシンプルになります。

また、JavaScript ライブラリの中でも、目的が限定された付加的なものだけでなく、ウェブサイトで使用されるアプリケーション全体を構築できるような広範囲をカバーするものを、枠組みを提供するというような意味で、フレームワークともいいます。jQuery には、ウェブサイトでは JavaScript を使用する際に必要なさまざまな処理や効果が幅広くまとめられており、フレームワークということもできるでしょう。

jQuery を使用すれば、長いスクリプトを自分で記述することなく効率的に開発を行うことができます。jQuery のソースコード自体も軽量で扱いやすいものになっています。また、jQuery と、自分のスクリプトや jQuery 以外のライブラリも併用することができます。

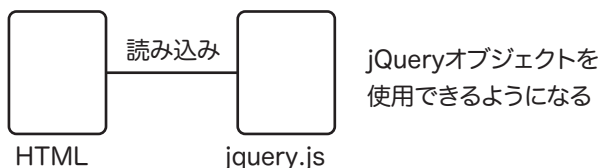
JavaScript を本格的に使用する際に避けて通れない、ブラウザの対応状況の違いも、jQuery では、現在使用されていると考えられるほとんどのブラウザに対応しており、JavaScript を記述する側で、ブラウザの問題を意識しなくて済みます。

jQuery ライブラリやプラグインは、DOM を操作するものから、ユーザーインターフェースや、効果的なエフェクトなど、JavaScript 全般を広くカバーしています。

jQuery の基礎知識

jQuery をウェブページで使用したい場合は、まず HTML の script 要素で、jQuery の JavaScript ファイルを読み込みます。次に、jQuery オブジェクトを使用し、HTML のソースコード中に、「jQuery()」と記述してから、jQuery のためのスクリプトを記述していきます。

jQuery オブジェクトの利用



ただし、jQuery を使用しているウェブページのソースコードでは、「jQuery()」という記述よりも、「\$ (ダラーマーク)」がたくさん登場することに気付くでしょう。これは、jQuery オブジェクトのショートカットといって、「jQuery()」と記述する代わりに「\$()」と記述することができるようになっているものです。\$() は、最も基本的なものであり、jQuery を使用する上で、まず初めに知っておきましょう。

次に、実際の例を見ながら基本的な仕組みを理解しましょう。



ポイント

jQuery 以外のライブラリと併用することもできますが、コンフリクトといって、お互いの記述内容が干渉しあい、うまく行かない場合もあります。このように、より応用的な使い方をする場合は、次のような記述を加えてコンフリクトしないようにすることがあります。

```
jQuery.noConflict();
```

ただし、この記述がある場合は、後ほど解説する、より効率的な記述のための「\$」という書き方は使えなくなります。

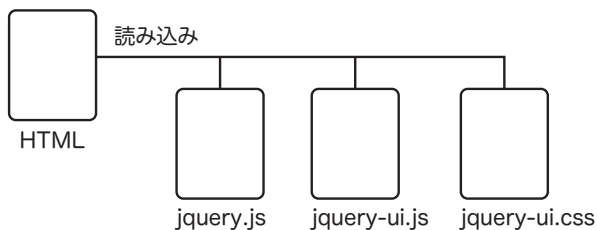
jQuery の簡単な記述例

まず、chap3sec2_fin フォルダーを開きます。このフォルダーの内容は、次に行う jQuery UI というライブラリを適用した完成見本です。詳細については、この後実際に作業を行いますので、ここでは jQuery の基本的な記述方法を説明するために、簡単に内容を把握してください。

このサンプルでは、jQuery を利用して、ウェブページのナビゲーションに、タブの表現を付け加えています。



この表現を行うために使用しているのが、「jquery.js」「jquery-ui.js」「jquery-ui.css」という3つのファイルです。「jquery.js」が jQuery 本体の JavaScript ファイル、「jquery-ui.js」がタブを使用するための JavaScript ファイル、「jquery-ui.css」がタブを表現するための CSS ファイルです。



次に、chap3sec2_fin フォルダーの、chap3sec2fin.html ファイルをテキストエディターで開いてください。head 要素の中に、次のような記述を見ることができます。

```
<link rel="stylesheet" type="text/css"
href="jquery-ui.css">
<script type="text/javascript" src="jquery.js"
"></script>
<script type="text/javascript" src="jquery-ui.
js"></script>
<script type="text/javascript">
$(function(){
    $('#tabs').tabs();
});
</script>
```

このうち、はじめの link 要素と、続く 2 つの script 要素は、jQuery 用の CSS と JavaScript ファイルを HTML に読み込むためのものです。

最後の script 要素の記述が、jQuery を利用して、このウェブページにタブの表現を付け加えている部分です。

```
<script type="text/javascript">
$(function(){
    $('#tabs').tabs();
});
</script>
```

jQuery を使用する際の、基本的な書き方は次の通りです。

```
$(function(){
    この部分に行わせたい処理を記述
});
```

jQuery オブジェクトを使用するために記述

```
<script type="text/javascript">
$(function(){
    $('#tabs').tabs(); — 行いたいjQueryの処理を記述
});
</script>
```

このように、記述することで、実際に行わせたい処理を、ウェブページがすべて読み込まれた後に実行するようになります。これは、ウェブページの内容をすべて読み込んでからでないと実行できないようなスクリプトがあった場合に、エラーになってしまうことを避ける意味があります。

次に、実際の処理である、`{ }` 内に含まれる記述を見てみましょう。

```
$('#tabs').tabs();
```

このうち、`#tabs` の部分がセレクタと呼ばれるもので、jQuery で行う処理の対象となるものを示しています。この場合は、CSS の記述と同様に、`#` (ハッシュ) を使用し、`id` で `tabs` とされた部分に対して、という意味を表しています。CSS と同様に、例えばこの後にクラスを指定するのであれば、`.` (ピリオド) でクラス名を指定することができます。

セレクタの部分の記述が終わったら、`.` (ピリオド) でつないで、メソッドを記述します。この例では、`tabs()` という jQuery UI で記述されたメソッドを実行します。

```
$('#tabs').tabs();
```

セレクタ	メソッド
対象となるもの	対象への指示

実際にどのような結果になるか、`chap3sec2fin.html` をブラウザで確認してください。

基本的な仕組みを見たところで、次に、このサンプルを実際に作成してみましょう。



JavaScript

jQuery 補足資料