

Web Designing Web マスター
足立裕司の

リニューアル覚え書き

Web Designing 公式サイトをリニューアルした際のちょっとしたポイントを紹介

サイトも併せて見てください



text Web マスター
足立裕司 (9031) <http://9031.com/>

フリーランスの Web デザイナー。本誌01号(2001年)から現在まで Web Designing 公式サイトの Web マスターを担当。また本誌2001年から2003年までは「Web SPICES」を連載。初のiOSアプリ「Will Return」を8月にリリース。<http://willreturn.at/>



Web Designing の公式サイトを、2012年7月18日に全面リニューアルしました。「今どきのサイトの標準的仕様+α」をコンセプトに、デザイン、使い勝手、コーディング、スマートフォン対応など、さまざまなアップデートを行っています。
<http://book.mynavi.jp/wd/>

Vol.3 jQuery の利用と細部のインタラクション

近年、JavaScript 使用時のフレームワークは、jQuery がほぼスタンダードになっています。しかしながら前回の Web Designing 公式サイトリニューアル時(約7年半前)の JavaScript には、まだそういったフレームワークはなく、ポップアップウィンドウを開く程度にしか使用していませんでした。派手なアニメーション的動作については、Flash が全盛の時代でした。今回のリニューアルでは、遅ればせながら、やっと jQuery をサイト全体に渡って使用することになりました。とはいえ、あまり派手な演出をするようなコンテンツではないので、必要性和効率重視の方向で取り入れています。プラグインの使用は意外と少なく、本サイト専用で書いているスクリプトが多くなっています。

既存プラグインとして基本的なところでは、アニメーションのイージングのバリエーションを増やすための Easing プラグイン、スクロールの動きをスムーズに行うための Smooth Scroll プラグインを使用しています。目立つところでは、毎号の誌面を拡大・スライドショーで表示する誌面レビューに、定番の jQuery 版 LightBox を使用しています。

LightBox へのトリガーとなる、トップページの誌面サムネイルのランダム表示は、プラグインではなく、本サイト専用で jQuery で作成しています。また、リニューアル前は Flash 形式であった、毎月の掲載サイトを自動で切替表示するバナーも、jQuery ベースにリニューアルしました。こちらもプラグインではなく、本サイト専用で作成しています。

そのほか細かなところでは、アコーディオン式の折り畳み表示や、ポップアップ・ドロップダウンメニューなどにも jQuery を使用しています。これらには、記述も簡単な jQuery のイージングアニメーションや class のトグル切替を利用しています。

誌面スペースの都合もあり、ソースは記述例として短く抜粋してまとめています。複雑なものについては、仕様の概略を紹介していきます。

横スライド・アニメーション表示



トップページの誌面サムネイル・ランダム切替
誌面サムネイルは、それぞれ「jQuery LightBox」(<http://leandrovieira.com/projects/jquery/lightbox/>) プラグインで該当ページを表示するためのトリガーとなっています。また、これらのサムネイルは、詳細ページのサムネイルと同一の画像なので、詳細ページ画像のプリロード機能も兼ねています。仕組みとしては、サムネイル画像の URL を格納した配列からランダムに1つを選択、3段のカラムのうちランダムに選んだ1段に追加し、数秒ごとに animate() で段の left 座標を移動しています。移動し終えたら左端に消えた1枚を remove() で削除します



旧 Flash バナーの jQuery 化
リンク先のリストをブロックレベル化して横1列に並べ、リストの項目数分の切替ボタンを自動生成、数秒ごとに animate() でリストの left 座標を移動しています

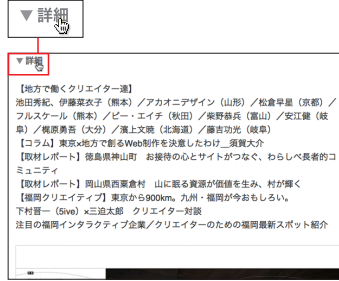
```
$("#element").animate({ // 要素のスタイル値をアニメーションさせながら可変
  left: ... // left 座標の値を
}, 1000, // 1秒かけて
function() { // アニメーションが完了したら以下を実行
  ...
});
```

アコーディオン式展開・折り畳み表示

展開前



展開後



アコーディオン表示は、プラグインを使わずとも数行のjQueryで簡単に実装できます。トリガーとした要素をクリックした際に、指定の要素をスライドダウン、またはスライドアップで、表示・非表示をトグル的に切り替えます。開閉時のアニメーションは、「jQuery Easing」(<http://gsgd.co.uk/sandbox/jquery/easing/>)プラグインを使えば動きのバリエーションを増やすこともできます。実際のサイトでは「詳細を表示」ボタンもjQueryからappend()で追加しています

```
<p class="more-trigger">詳細を表示</p>
<div class="more-content">
  <p>...</p>
</div>
```

```
$(".more-trigger").toggle(function(){ // トリガーをクリックしたら以下の動作を交互に実行
  $(this).addClass("open").next(".more-content").stop().slideDown("normal","easeInQuad"
); // スライドダウンで表示
}, function(){
  $(this).removeClass("open").next(".more-content").stop().
  slideUp("normal","easeInQuad"); // スライドアップで非表示
});
```

ポップアップメニュー

展開前



展開後



ポップアップメニューやドロップダウンメニューは、閉じている状態と展開している状態それぞれのスタイルをCSSで用意しておき、トリガーとなる要素をクリックしたときにclassをトグル的に切り替え、スタイル(表示・非表示など)が変更されるようにして実装しています。メニューを展開した後は、ページ上の任意の場所をクリックしたらメニューを閉じる動作を追加します。開閉時にアニメーション効果を用いたい場合には、前述のアコーディオンのような方法やanimate()を使った方がいいと思います

```
<ul id="nav-lang">
  <li><a href="#">Japanese</a></li>
  <li><a href="#">English</a></li>
</ul>
```

```
$("#nav-lang a").click(function(event){ // メニューをクリックしたら
  if (!$("#nav-lang").hasClass("open")) { // メニューがまだ展開状態でないなら
    event.preventDefault(); // a要素のデフォルトの動作を止め
    $("#nav-lang").addClass("open"); // メニュー展開時のスタイル(class)を適用
    $("body").bind("click.nav-lang",function(){ // ページ上の任意の位置をクリックしたら
      $("#nav-lang").removeClass("open"); // メニュー展開時のスタイル(class)を削除
      $(this).unbind("click.nav-lang"); // ページ上のクリック監視も削除
    });
  }
});
```

ページ内スクロール



既成の「ページトップに戻る」プラグインも多数ありますが、このサイトの場合は、「ページトップに戻る」以外に「特集記事」「連載記事」の見出しの計3カ所を移動できるように作成しています。ウィンドウのスクロールイベントを監視して、スクロール量が発生していればボタンをフェードイン、そうでなければフェードアウト、という動作で簡単に実装できます。ページ内アンカーのスクロール時の動きには「jQuery Smooth Scroll」(<https://github.com/kswedberg/jquery-smooth-scroll>)プラグインを使用しています。ここで、さきほどのjQuery Easingプラグインと組み合わせると、スクロール時の動きにバリエーションを持たせることもできます

```
$(window).scroll(function(){ // ウィンドウをスクロールしたら
  if ($(window).scrollTop()>10 && !$("#totop:visible").length) { // ウィンドウのスクロール量
    10px以上で
    $("#totop").fadeIn("fast"); // ボタンをフェードインで表示
  } else if ($(window).scrollTop()<=10 && $("#totop:visible").length) { // またはウィンドウの
    スクロール量10px以下で
    $("#totop").fadeOut("fast"); // ボタンをフェードアウトで非表示
  }
});
$("a[href^='#']").click(function(event){ // 同一ページ内アンカーをクリックしたら
  event.preventDefault(); // a要素のデフォルトの動作を止め
  $.smoothScroll({ scrollTarget: this.hash, easing: "jswing" }); // ターゲットの位置までSmooth
  Scroll
});
```