

jQuery Tips

[jQuery][JavaScript]

- ・リファレンス http://docs.jquery.com/Main_Page
- ・リファレンス (日本語) <http://semooh.jp/jquery/>
- ・入門 http://www.openspc2.org/JavaScript/Ajax/jquery_study/ver1.3.1/index.html

ロジック

文字列操作

トリミング

```
var s = $.trim(s);
```

繰り返し

each() からの脱出

オブジェクトの内容が空白か判定

- ・ each() からは、return false で脱出できる (関数は抜けない)

```
function isBlank(input) {  
    if (input == null) {  
        return true;  
    }  
    if ($.type(input) == "string") {  
        return ($.trim(input) == "");  
    }  
    var flg = true;  
    $(input).each(function(){  
        var obj = $(this);  
        if ($.trim(obj.val()) != "" || $.trim(obj.text()) != "") {  
            flg = false;  
            return false;  
        }  
    });  
    return flg;  
}
```

機能

読み込み中メッセージを表示

html

```
<div id="loading" style="display:none"> 計算中・・・</div>
```

style

```
<style type="text/css">  
#loading {  
    position: absolute;  
    top:200px;
```

```

left:200px;
padding: 8px;
background-color:lightyellow;
color: steelblue;
font-size:small;
text-align:center;
width:100px;
}
</style>

```

表示

```
$("#loading").show("normal");
```

消す

```
$("#loading").hide("normal");
```

例

COCOMO計算式	
ズ (KLOC)	
PM)	
l (M)	計算中...
l (P)	
OC/PM)	
工数の局面への分布	

配列

配列に値が存在するか調べる

- <http://api.jquery.com/jquery.inArray>
- 存在しない場合、-1

```
$.inArray("val",array);
```

配列を変換

- [jQuery.map](#)

```

var newArray = $.map(array, function(val,index){
return parseInt(val.trim());
});

```

Ajax

エラーの場合、レスポンスボディ (たとえば XML) は jqXHR.responseText から取得

```
$.ajax({
url:url,
```

```

success: function(data, textStatus, jqXHR) {
  :
},
error: function(jqXHR, textStatus, errorThrown) {
  var xml = jqXHR.responseText;
  if (xml != null) {
    alert(xml);
  } else {
    alert(errorThrown);
  }
}
:

```

ただし、XML が文字列として取得されるので、jQuery で XML として利用するには、DOM への変換が必要

XML 文字列を DOM に変換するライブラリ

- <http://outwestmedia.com/jquery-plugins/xmldom/>

よくわからない HTTP ステータスコードでエラーとなることがある

- WinInet のエラーの可能性

<http://support.microsoft.com/kb/193625>

```

12029      ERROR_INTERNET_CANNOT_CONNECT
           he attempt to connect to the server failed.

```

Http リクエストヘッダーを設定する

- <http://api.jquery.com/jquery.ajax/>
- beforeSend を利用する

```

$.ajaxSetup({
  beforeSend: function(xhr) {
    xhr.setRequestHeader("If-Modified-Since", "Thu, 01 Jun 1970 00:00:00 GMT");
  }
});

```

UI

Dialog を自動で閉じる

- <http://jqueryui.com/demos/dialog/>

```

var dialog = $("#dlg_message");
dialog.dialog("open");
setTimeout(function(){
  if (dialog.dialog("isOpen")) {
    dialog.dialog("close");
  }
}, 800);

```

Accordion UI をすべて展開状態にする

Accordion UI をすべて展開状態にすることには対応していないため、独自に実装する

- <http://docs.jquery.com/UI/API/1.8/Accordion>

例

- <http://jsbin.com/eqape>

```
<div id="hoge_panel">
  <h3><a href="#"> タイトル 1</a></h3>
  <div>test1</div>
  <h3><a href="#"> タイトル 2</a></h3>
  <div>test2</div>
  <h3><a href="#"> タイトル 3</a></h3>
  <div>test3</div>
  <h3><a href="#"> タイトル 4</a></h3>
  <div>test4</div>
</div>

$("#hoge_panel").addClass("ui-accordion ui-accordion-icons ui-widget ui-helper-reset")
    .find("h3").addClass("ui-accordion-header ui-helper-reset ui-state-default ui-corner-top
ui-corner-bottom")
    .hover(function() { $(this).toggleClass("ui-state-hover"); })
    .prepend('<span class="ui-icon ui-icon-triangle-1-s"></span>')
    .click(function() {
        $(this)
            .toggleClass("ui-accordion-header-active ui-corner-bottom")
            .find("> .ui-icon").toggleClass("ui-icon-triangle-1-e ui-icon-triangle-1-s").end()
            .next().toggleClass("ui-accordion-content-active").slideToggle();
        return false;
    }).next().addClass("ui-accordion-content ui-helper-reset ui-widget-content
ui-corner-bottom").show()
;
;
```

Accordion UI 内容の高さを制御する

- heightStyle を設定する。
- <http://api.jqueryui.com/accordion/#option-heightStyle>

```
$(".selector").accordion({"heightStyle":"content"});
```

ローディングアイコンをなるべく画面の中央に表示

```
$("#loading").css({
    function() { return $(window).innerWidth() / 2; },
    top : function() {
        return $(document).scrollTop() + $(window).innerHeight() / 2;
    }
});
$("#loading").show();
```

DOM

jQuery オブジェクトから DOM オブジェクトを得る

```
var domObj = jsonObj.get(0);
```

HTML

選択されたラジオボタンの value を取得する。

```
var sel_val = $("input:radio[@name='radio_button_name']:checked").val();
```

複数マッチした要素から、特定の要素を取得する。

- <http://api.jquery.com/category/traversing/>

```
var first_h2 = $("h2").first();
```

IE で動的にセレクトボックスの項目を変更できない

IE で正しく動作しない

```
$("#option", "#selectboxId").each(function(){  
    if ($(this).val() == targetCode) {  
        $(this).text(newName);  
    }  
});
```

DOM を操作する (IE でも OK)

```
var selectBox = document.getElementById("selectboxId");  
for (var i=0; i<selectBox.options.length; i++) {  
    if (selectBox.options[i].value == targetCode) {  
        selectBox.options[i] = new Option(newName, targetCode);  
    }  
}
```

HTML(テーブル)

選択された行のセルのテキストを取得

```
var tr = $("<tr>")  
tr.click(function(){  
    var cols = $(this).children();  
    var col0_txt = $(cols[0]).text();  
    var col1_txt = $(cols[1]).text();  
    var col2_txt = $(cols[2]).text();  
});
```

テーブルの行が選択された時にラジオボタンを ON にする。

```
var tr = $("<tr>")  
tr.click( function () {  
    $("#projects_table tbody tr").removeClass("selected_row");  
    $(this).addClass("selected_row");  
    $(this).children("td").children("input:radio").attr("checked", "checked");  
});
```

new project	
System Name	Application Name
<input checked="" type="radio"/> test1	test2
<input type="radio"/> test1	test2

属性の設定

- ・ 'q' という id の value に test を設定

```
$('#q').attr('value', 'test');  
// OR  
$('#q').val('test')
```

テーブルの操作

Script

```
// $.getJSON callback  
function reload_date(json) {  
  if ($("#calc_expr_table tbody").length > 0) {  
    $("#calc_expr_table tbody").remove();  
  }  
  $("#calc_expr_table").append("<tbody></tbody>");  
  $("#calc_expr_table tbody").append(  
    "<tr><th> プログラムサイズ (KLOC)</th><td>" + json.kdsi + "</td></tr>" +  
    "<tr><th> 工数 (PM)</th><td>" + json.effort + "</td></tr>" +  
    "<tr><th> 開発期間 (M)</th><td>" + json.tdev + "</td></tr>" +  
    "<tr><th> プログラムサイズ (KLOC)</th><td>" + json.fsp + "</td></tr>" +  
    "<tr><th> 生産性 (KLOC/PM)</th><td>" + json.prod + "</td></tr>"  
  );  
}
```

HTML

```
<table id="calc_expr_table"></table>
```

CSS

クラスの追加

- ・ num_value id の要素のスタイルクラスを num_field に設定

```
$("#num_value").addClass("num_field");
```

XML

XML のタグ名を取得する

- ・ <http://blog.livedoor.jp/luna666luna/archives/3447449.html>

```
$(xml).each(function(){  
  $(this).get(0).tagName;  
});
```

プラグイン

- ・ <http://plugins.jquery.com/>

数値書式 jquery-numberformatter

- <http://code.google.com/p/jquery-numberformatter/>

```
$("#num_field").format({format:"#,###.00", locale:"us"});
```

角を丸くする jQuery Corner

- <http://malsup.com/jquery/corner/>

```
$("#message_area").corner();
```

テーブル操作

- <http://www.designwalker.com/2009/09/jquery-table.html>
- <http://d.hatena.ne.jp/cyokodog/20080720/1216569757>

Cookie

- <http://archive.plugins.jquery.com/project/Cookie>

有効期限を設定する

```
var serviveDays = 30;
var date=new Date();
date.setTime(date.getTime()+(serviveDays*24*60*60*1000));
$.cookie("key_foo", "value_bar", {expires:date});
```

API Memo

html()

- innerHTML と同じ

addClass(class), removeClass(class)

- 指定した要素に CSS クラスを追加 / 削除

parent(expr), children(expr)

- parent : 対象の要素の親 (先祖をさかのぼる) を取得
- children : 対象の要素の子 (直下の子供のみ) を取得