FAIRE UN PLUGIN EN JQUERY

**Sites visités :**

[www.grafikart.fr](http://www.grafikart.fr)

[www.pierre-giraud.com](http://www.pierre-giraud.com)

[www.w3shool.com](http://www.w3shool.com)

**Méthode**

1.Créer un dossier monplugin.js

exemple de saisi de plug in <https://openclassrooms.com/courses/1259101-developpez-votre-propre-plug-in-jquery> :

(*function*($)

{

$.*fn*.notreSuperPlugIn=*function*(*parametres*)

{

//Code de notre plug-in ici

};

})(jQuery);

A l’intérieur on saisit les données dynamiques de notre plugIn

Ici, nous voulons appliquer notre plug-in sur tous les paragraphes. En fait, cela revient à appeler une méthode de nos paragraphes. Si vous avez des notions en POO, vous savez que pour accéder à l'élément appelant la méthode, il faut utiliser le mot-clé **this**. Si vous n'avez pas de notions en POO, sachez simplement que pour accéder à l'élément appelant la méthode, il faut utiliser le mot-clé **this**.

Ainsi, le mot-clé **this** dans notre plug-in fait référence à l'objet **jQuery** appelant notre plug-in ! De petits exemples pour que vous compreniez bien :

$('div').notreSuperPlugIn(); //this fera référence à l'objet jQuery $('div')

$('#header').notreSuperPlugIn(); //this fera référence à l'objet jQuery $('#header')

$('table.commentaire').notreSuperPlugIn(); //this fera référence à l'objet jQuery $('table.commentaire')

Pour respecter cette philosophie, notre plug-in doit permettre de s'ajouter dans cette chaîne, comme ceci :

$('p').css('color','red').notreSuperPlugIn().appendTo($('div')).show();

Pour ce faire, rien de plus simple, il suffit de renvoyer l'objet **jQuery** appelant notre plug-in... Vous savez, notre fameux mot-clé **this**.

(*function*($)

{

$.*fn*.notreSuperPlugIn=*function*(*parametres*)

{

this.each(*function*()

{

//Code de notre plug-in ici

});

return this;

};

})(jQuery);

Bien entendu, la fonction each() renvoie elle aussi l'objet **jQuery** l'ayant appelé. Ainsi, il est également possible d'écrire ceci :

(*function*($)

{

$.*fn*.notreSuperPlugIn=*function*(*parametres*)

{

return this.each(*function*()

{

//Code de notre plug-in ici

});

};

})(jQuery);

Ceci sera donc notre code de base pour tous nos futurs plug-ins. Ce n'est pas bien compliqué en fait, il faut juste le savoir, c'est tout )

**Voici un exemple :**

(function($) {

// jQuery plugin definition

$.fn.reverseText = function(params) {

// merge default and user parameters

params = $.extend( {minlength: 0, maxlength: 99999}, params);

// traverse all nodes

this.each(function() {

// express a single node as a jQuery object

var $t = $(this);

// find text

var origText = $t.text(), newText = '';

// text length within defined limits?

if (origText.length >= params.minlength && origText.length <= params.maxlength) {

// reverse text

for (var i = origText.length-1; i >= 0; i--) newText += origText.substr(i, 1);

$t.text(newText);

}

});

// allow jQuery chaining

return this;

};

})(jQuery);

2.Insérer le script ‘monPlugin’ entre le script bibliothèque jQuery et le script.Js