

SOMMAIRE

Fonction	5
Hide()	5
Show()	5
addClass()	5
Filtres	5
:header	5
:Parent(parent d"éléments)	5
:has(contient un élément)	5
:contains(texte)(contient un texte précis)	5
:empty(élément vide)	5
:not(negation)	5
:only-child(enfant unique)	5
\$("input[type=submit]") ;	5
: input : text : checkbox : radio	5
:selected :checked :disabled	5
Evénement	5
Bind()	5
Unbind()	6
Propriété des événements	6
Toggle ()	6
Ready()	6
Event.preventDefault()	6
\$(this)	6
HTML ET JQUERY	6
Html ()	6
Text()	6
Maniuplation des attributs	6
Val()	6
Attr()	7
RemoveAttr ()	7
Jquery et CSS	7
addClass()	7
hasClass()	7
removeClass()	7

toggleClass()	7
Manipulations CSS	7
Css()	7
Dimension	8
Height() Width()	8
Height(valeur) Width(valeur)	8
innerHeight() innerWidth()	8
outerHeight() outerWidth()	8
• Positionnements des objets	8
Offset ()	8
OffsetParent ()	8
Position () Position (top,left)	8
ScrollTop ()	8
ScrollLeft ()	8
DOM (Document Object Model) Et Manipulation	8
Append () Prepend()	8
AppendTo() prependTo()	8
Before() after()	8
InsertAfter () insertBefore ()	8
Clone ()	8
ReplaceWith ()	9
ReplaceAll ()	9
wrap()	9
Unwrap()	9
WrapAll ()	9
WrapInner ()	9
Empty()	9
Remove ()	9
detach()	9
Animations et effets	9
• Effets	9
Hide()	9
hide(1000 ou slow ou fast)	9
Show()	9
Toggle()	9
Toggle(true ou false)	10
Delay()	10
• Callback	10
Mouvement et transitions	10
slideDown() slideUp() slideToggle()	10

fadeIn() fadeOut() fadeTo()	10
fadeTo("slow",0.5)	10
animate()	10
Stop()	10
Filtres.....	11
Eq() first() last() filter() is() not () slic()	11
\$(p).eq(-2)	11
Slice(debut,fin)	11
Filter()	11
Not().....	11
Recherche.....	11
Children() closest() contents() siblings() next()	11
Find ()	11
Next() prev()	11
nextAll() prevAll()	11
closest()	11
Parent ()	11
children()	11
Parents ()	11
Siblings ()	11
Parcours.....	11
Index()	11
Get()	12
Each()	12
\$.each()	12
.length.....	12
Size ()	12
Plugins et jQuery	12
Ajax et jQuery.....	12
\$("#maListe ").load(" page.html ") ;	12
\$("#maListe ").load(" page.html #monId ") ;	13
Load avec paramètres.....	13
\$.ajax	13
Evénements.....	13
ajaxComplete	13
ajaxError	13
ajaxSend.....	13
ajaxStart.....	13
ajaxStop.....	13
ajaxSuccess.....	13

\$.post.....	13
\$.get.....	14

Fonction

Hide()

cache un contenu

Show()

afficher un contenu

addClass()

ajouter un classe

EXEMPLE :

```
$("#a").hide()  
$("#a").hide("slow")
```

Filtres

:header

Les titres h1 à h6

:Parent(parent d"éléments)

:has(contient un élément)

:contains(texte)(contient un texte précis)

:empty(élément vide)

EXEMPLE :

```
Ex : <div></div>
```

:not(negation)

:only-child(enfant unique)

\$("#input[type=submit]");

```
! not  
^ debut  
$ finit  
* contient la valeur
```

: input : text : checkbox : radio etc ...

:selected :checked :disabled

Événement

EXEMPLE :

```
$("#a").click(function(){  
});
```

Bind()

Lie une fonction à un ou plusieurs événements survenant pour un ou plusieurs éléments

EXEMPLE :

```
$("#a").bind("click",function(){
}) ;

$("#a").bind("click mouseenter",function(){
}) ;
```

Unbind()

Supprimer les événements d'un élément

Propriété des événements

EXEMPLE :

```
$("# p ").bind(" click ",function("e "){
Alert(e.pageX + e.pageY) ;
}) ;
```

E variable rempli automatiquement par l'événement

On associe un événement à un élément pour s'exécuter une seule fois

Toggle ()

simule le clic alternatif par la souris

EXEMPLE :

```
$.toggle(fonctionPaire,fonctionImpaire) ;
```

Ready()

Associe une fonction exécutée lorsque le DOM est prêt à être manipulé

Event.preventDefault()

Désactiver le comportement normal d'un élément comme return false

\$(this)

réfère à l'objet ayant appelé la méthode ou à celui recevant l'événement

HTML ET JQUERY

Html ()

Renvoie le contenu innerHTML d'un élément

- Modification

EXEMPLE:

```
$("#id").html("<p>etc</p>") ;
```

Text()

Renvoie le contenu texte (sans html)

- Modifier le texte tel quelle

EXEMPLE:

```
$("#p3").text("<p>hello</p>");
```

Manipulation des attributs

Val()

Renvoie et modifier la valeur d'un élément

Attr()

Renvoie la valeur d'un attribut

EXEMPLE :

```
$("#img :first").attr("alt") ;
```

- Modification

EXEMPLE :

```
$("#img :first").attr("alt","modification") ;
```

RemoveAttr ()

Supprimer un attribut

Propriétés multiple

EXEMPLE :

```
$(menu a).attr({  
  Href : "www.test.com",  
  Rel : "nofollow"  
});
```

RESULTAT :

```
<p id="menu">  
<a Href="www.test.com"  
Rel="nofollow">go</a></p>
```

Jquery et CSS

addClass()

Ajouter une classe à certains éléments

hasClass()

Tester la présence d'une classe renvoie true ou false

removeClass()

Supprimer une classe

toggleClass()

Ajouter une classe ou plus si elle est absente et retire la classe si elle est présente

Manipulations CSS

Css()

Retourne la valeur de la propriété CSS

MODIFICATION :

```
$(".css("color","red") ;
```

Modifier une ou plusieurs propriétés

```
Var tabcss={  
  "color" : "red",  
  "font-weight", "bold"  
} ;  
$(".p").css(tabcss) ;
```

CHAINAGE :

Permet de chaîner des fonctions au lieu de répéter le sélecteur

```
$("#id").css("color","red") ;  
$("#id").addClass("maClasse") ;  
$("#id").css("color","red").addClass("maClasse") ;
```

Dimension

Height() Width()

Obtenir les dimensions :

Height(valeur) Width(valeur)

Modifier les dimensions :

innerHeight() innerWidth()

Dimensions internes (padding inclus) :

outerHeight() outerWidth()

Dimensions externes (avec border)

- **Positionnements des objets**

Offset ()

Position sur la page

OffsetParent ()

Premier parent positionné

Position () Position (top,left)

Position relative à l'offsetParent

ScrollTop ()

Retourne ou modifie la valeur de scroll par rapport au haut de la page

ScrollLeft ()

Retourne ou modifie la valeur de scroll par rapport au gauche

DOM (Document Object Model) Et Manipulation

Append () Prepend()

Ajoutent du contenu à l'intérieur des éléments

AppendTo() prependTo()

Ajoutent les éléments sélectionnés présents dans le DOM à l'intérieur de la cible indiquée

Before() after()

Insérer du contenu avant/après les éléments sélectionnés

InsertAfter () insertBefore ()

Insèrent le contenu indiqué présent dans le DOM avant/après les éléments sélectionnés

Clone ()

Duplique un nœud

EXEMPLE :


```
$("#p").clone().appendTo("#cible") ;
```

ReplaceWith ()

Remplace l'élément par du code HTML

ReplaceAll ()

Remplace tous les éléments par celui sélectionné

wrap()

Entoure chaque élément avec le nouveau

EXEMPLE :

```
$("#p").wrap("<div></div>") ;
```

RESULTAT :

```
<div><p></p></div>
```

Unwrap()

Supprime le conteneur de l'ensemble des éléments sélectionnés

WrapAll ()

Entoure l'ensemble des éléments sélectionnés avec le nouveau indiqué

WrapInner ()

Entoure le contenu des éléments sélectionnés

Empty()

Supprime tous les enfants

Remove ()

Supprime du DOM les éléments ciblés

detach()

Détache temporairement les éléments(les supprime du DOM) pour réutilisation future

EXEMPLE :

```
var $images=$("# images ").detach() ;  
images.insertAfter("p") ;
```

Animations et effets

- Effets

Hide()

Masque un élément

hide(1000 ou slow ou fast)

Effet avec animation

Show()

Afficher un élément

Toggle()

Affiche ou masque un élément selon sa visibilité

Toggle(true ou false)

VITESSES

Slow -> 600ms
Normal -> 400ms
Fast -> 200ms

Delay()

Ajouter un délai d'exécution dans la file d'attente de l'animation(en ms)

EXEMPLE :

```
$("#p").hide("slow").delay(3000).show()
```

- **Callback**

Exécuté après la fin de l'animation

EXEMPLE :

```
$("#p").hide("slow",  
Function(){  
    Alert("animation terminée") //fonction de callback  
});
```

Mouvement et transitions

slideDown() slideUp() slideToggle()

Révèle/masque un élément en ajustant sa hauteur

fadeIn() fadeOut() fadeTo()

Révèle/masque un élément grace à l'opacité

fadeTo("slow",0.5)

Avec un niveau d'opacité

animate()

agit sur un ensemble de propriétés css notées en camelCase(ex :marginLeft,width) dont les valeurs sont numériques

EXEMPLE :

```
$(cible).animate(  
{  
    Opacity :0.5,  
    Width :50%,  
    marginLeft :20px,  
    fontSize :4em  
},"slow"  
);
```

Stop()

Stop l'animation d'un ou plusieurs éléments

Filtres

Eq() **first()** **last()** **filter()** **is()** **not ()** **slic()**

\$(p).eq(-2)

À partir de la fin

Slice(debut,fin)

Tranche ou intervalle

Filter()

EXEMPLE :

```
$(p).filter(".info")
$(p).filter(":first-child")
```

Not()

Cible les éléments contraires au filtre

Recherche

Children() **closest()** **contents()** **siblings()** **next()** ...

Find ()

EXEMPLE :

```
$(div).find("img")
```

Next() **prev()**

Frères suivants et précédents directs

nextAll() **prevAll()**

Tous les frères suivants et précédents directs

closest()

Le plus proche parent

Parent ()

Retourne le parent direct

children()

Retourne le fils direct

Parents ()

Retourne tous les parents (jusqu'à body et html)

Siblings ()

Retourne tous les frères

Parcours

Index()

Retourne la pos d'un élément dans la liste par rapport à ses frères

Get()

Retourne les éléments DOM

Each()

Itère sur les éléments d'une collection

EXEMPLE :

```
$("#p").each(function(index,element){  
    Console.log(index + " = " + element) ;  
});
```

\$.each()

EXEMPLE :

```
Var test=Array(" kiwi ", " mangue ") ;  
$.each(test,function(){  
    Console.log(this) ;  
});
```

.length

Propriété permet de connaître le nombre d'éléments sélectionnés, ou de vérifier si un élément existe

Size ()

Fonction équivalente

EXEMPLE :

```
$("#*").size () ;
```

Plugins et jQuery

- Un plugin est une fonction
- \$.fn.extend étend les fonctions internes de jQuery

EXEMPLE :

```
$.fn.faireqqChose=function(arg,..){  
    Return $(this) ;  
};
```

EXEMPLE :

```
$.fn.test=function($width,$height){  
    console.log($width + " " + $height);  
    console.log($(this));  
    return $(this);  
}
```

EXECUTION :

```
$("#h1").test(100,200);
```

Ajax et jQuery

\$("#maListe").load("page.html") ;

Charger une page ou une partie de code HTML

`$("#maListe").load(" page.html #monId ");`

Ou en précisant un identifiant

Load avec paramètres

- Paramètres passés en POST (facultatif)
- Callback facultatif

EXEMPLE :

```
$("#maListe").load(" glaces.php ",  
{  
    "parfums[]" :["chocolate","kiwi"]  
});
```

```
$("#flux").load(" rss.php ",  
{limit :10}  
Function(){alert(" chargement ok ") ;}  
);
```

\$.ajax

EXEMPLE :

```
$.ajax(  
{  
    url : " inscription ",  
    type= " POST ",  
    data= " nom= "+nom+"& " email= "+email,  
    success :function(resultat){  
        console.log(resultat) ;  
    }  
});
```

Evénements

S'applique à l'ensemble du document pour toute requête effectuée par le navigateur

ajaxComplete → Requête terminé

ajaxError → Erreur

ajaxSend → Avant l'envoi de la requête

ajaxStart → Au début de l'envoi

ajaxStop → Quand toutes les requêtes ont pris fin

ajaxSuccess → Succès !

EXEMPLE :

```
$(" p ").ajaxComplete(function(){  
    Console.log(" requete ajax terminée ") ;  
});
```

\$.post

EXEMPLE :

```
$(#ajax-button).click(function(){  
$.post(  

```

```
    " ajax-post.php ",
    {
        Login :$(" #login ").val(),
        Password : $(#password ).val()
    },
    Function(data){
        Console.log(data) ;
    }, " text "
) ;
}) ;
```

\$.get

Cas d'un fichier XML

EXEMPLE:

```
$.get(
    " fichier.xml ",
    False,
    Fonction-retour,
    " xml "
) ;
```

```
Function Fonction-retour (donnee-xml){
    Console.log(donnee-xml) ;
}
```