

JQUERY

1-Présentation JQuery

- Bibliothèque Javascript
- Navigateurs supportés:
 - Chrome, Firefox: \geq version courante – 1
 - Safari: 5.1+
 - Opera: 12.1x ou \geq v courante -1
 - iOS: 6.1+
 - Android: 2.3, 4.0+
 - IE:
 - JQuery 1.x (IE 6+)
 - JQuery 2.x (IE 9+)

jquery

Terme de recherche

Dojo

Terme de recherche

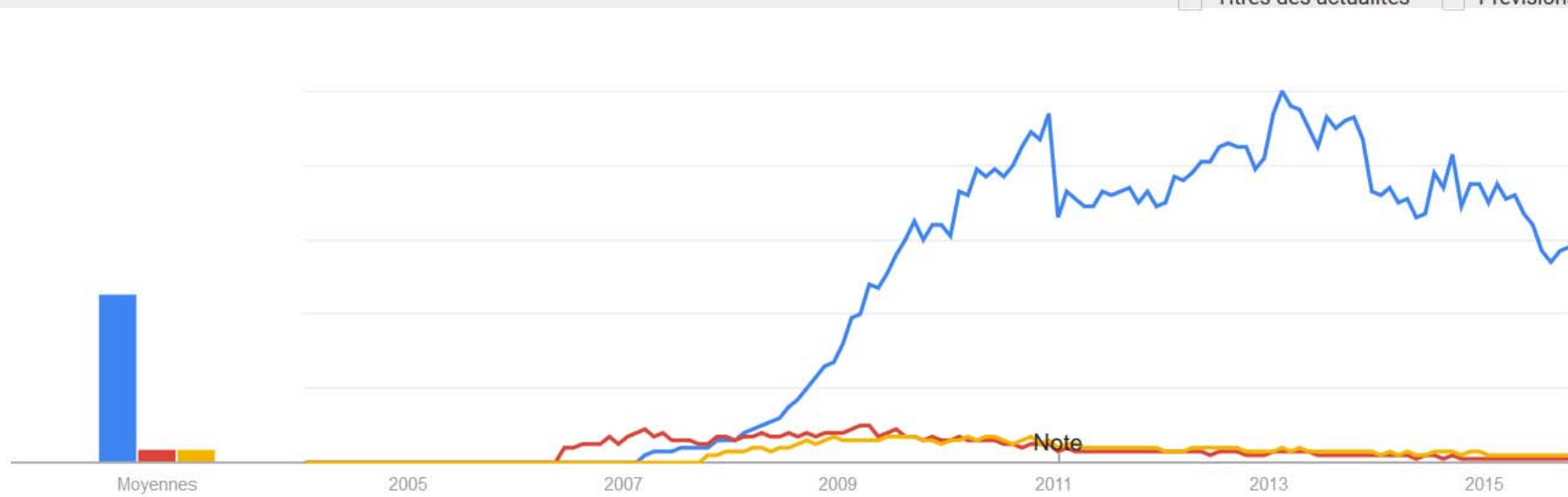
ExtJS

Terme de recherche

+ Ajouter un terme

Évolution de l'intérêt pour cette recherche ?

Titres des actualités Prévisions ?



</>

2- JQuery: Caractéristiques

- Gestion HTML/DOM
- CSS
- Événements HTML
- Effets et animations
- AJAX
- Utilitaires
- Support des plugins

3- Intégrer JQuery à une application

- Chargement de la bibliothèque à partir d'un site distant

- **Microsoft:**

- `<script src="http://ajax.aspnetcdn.com/ajax/jQuery/jquery-2.1.1.min.js"> </script>`

- **Google**

- `<script src="ajax.googleapis.com/ajax/libs/jquery/2.1.1/jquery.min.js "> </script>`

- `<script src="https://code.jquery.com/jquery-2.1.1.min.js"> </script>`

- Chargement local:

- `<script src="jquery-2.1.1.js"></script>` chargement des éléments de base de la bibliothèque.

4- Syntaxe JQuery

- `$(sélecteur).action()`: un élément (ou une collection d'éléments) est sélectionné à l'aide de la fonction `$`, puis l'action « `action()` » lui est appliquée.
- `$` est un raccourci de la fonction `jquery`.
- JQuery utilise les sélecteurs css pour sélectionner les éléments, exemples:
 - `$(this).hide()` // cache l'élément en cours
 - `$("p").hide()` // masque tous les éléments `<p>`
 - `$(".c1").hide()` // masque tous les éléments de la classe `c1`
 - `$("#id1").hide()` // masque l'élément dont l'id est `id1`

5- Les fonctions de base

- `$(document).ready(function () {
 // ...
});`

- **Ou bien**

- `$(function () {
 // ...
});`

- Exécute la fonction anonyme une fois la page est complètement chargée.

Sélection des éléments avec css et xpath

- JQuery utilise une combinaison des langages css et xpath pour sélectionner les éléments.
- Exemples:
 - `$("*")` // Sélection de tous les éléments
 - `$("p.intro")` //Sélection de tous les éléments p de la classe intro
 - `$("p:first")` //sélection du premier élément <p>
 - `$("ul li:first")` //Sélection du premier élément li dans le premier élément ul.
 - `$("ul li:first-child")` // Sélectionne le premier élément de chaque

- `$("[required]")` //Sélectionne tous les éléments ayant un attribut `required`
- `$("a[target='_blank']")` //Sélectionne tous les éléments `<a>` ayant l'attribut `target = _blank`
- `$("a[target!='_blank']")`
- `$(":button")` //Sélectionne tous les éléments `<button>` et `<input>` de type `button`
- `$("tr:even")` // Sélectionne tous les élément `<tr>` pairs
- `$("tr:odd")` // Sélectionne tous les élément `<tr>` impairs
- `$("p[@class]")` //Sélectionne tous les éléments `p` ayant un attribut `class`.
- `$("input[@name=txtNom]")`
- `$("li:contains('Marrakech')");` // Sélectionne les éléments `li` contenant le mot 'Marrakech'.

5- les événements

- JQuery Simplifie la gestion des événements de JavaScript
- connecter une fonction à un événement DOM, exemple: définir l'événement clic de tous les paragraphes :

```
$("#p").click(function () {  
    // ...  
});  
$("#p").dblclick(function () {  
    $(this).hide();  
});  
$("#input").focus(function () { $(this).css("background-color",  
"#cccccc"); });  
$("#input").blur(function () {$(this).css("background-color",  
"#ffffff");  
});
```

6- Ajax

- La fonction load
 - `$(selecteur).load(URL [, parametres, fonction_rappel]);`
 - La fonction load charge les données à partir de l'url « URL » et les affiche dans l'élément identifié par « selecteur ».
 - parametres: données à envoyer en paramètre de la requete Ajax.
 - fonction_rappel: fonction à appeler après le chargement des données.
- La fonction de rappel peut avoir les paramètres suivants (reponse, statut, xhr):
 - reponse: résultat de la requête Ajax
 - statut: statut de l'appel, peut avoir l'une des valeurs: success ou error.
 - xhr: l'objet xhr.
- Exemple:
 - `$("#div1").load("contenu.txt");`

\$.get() et \$.post()

- \$.get(URL, fonction_rappel,[type_donnees]);
 - La fonction de rappel possède deux paramètres (donnees, statut)
 - \$.post(URL, parametres, fonction_rappel, type_donnees);
 - donnees: Les données téléchargées par la requête Ajax.
 - statut: statut de l'exécution de la requête Ajax.
 - type_donnees: html,text,xml,json

\$.ajax

```
var req = $.ajax({
  type: "POST",
  dataType: "json",
  url:"clients.aspx",
  data: "'ID': '123'",
  contentType:
    "application/json; charset=utf-8",
  success: function (res) {
    alert("Nom du client: " + res.nom);},
  error: function (xhr, textStatus, errorThrown) {
    alert(xhr.status);
    alert(textStatus);
    alert(thrownError);
  },
  async:true });
```

7- Effets et animations

- `hide()` et `show()`
 - Syntaxe: `$(selecteur).hide(vitesse,rappel);`
 - Vitesse: `slow`, `fast` ou une valeur en ms.
- `toggle()`
 - Syntaxe: `$(selecteur).toggle (vitesse,rappel);`
 - Si l'élément est affiché il sera masqué et inversement.
- `Fading()`
 - `fadeIn(): $(selecteur).fadeIn(vitesse,rappel);`
 - `fadeOut(): $(selecteur).fadeOut(vitesse,rappel);`
 - `fadeToggle(vitesse, rappel)`
 - `fadeTo(vitesse,opacité, rappel):` opacité entre 0 et 1.

Exemple FadeIn et FadeOut

Script

```
$("#cmdMasquer").click(function () {  
    $("#div1").fadeOut();  
  
    $("#div2").fadeOut("slow");  
  
    $("#div3").fadeOut(3000);  
        });  
  
$("#cmdAfficher").click(function () {  
    $("#div1").fadeIn();  
  
    $("#div2").fadeIn("slow");  
  
    $("#div3").fadeIn(3000);  
        });
```

HTML

```
<p>FadeIn et FadeOut</p>  
<button  
    id="cmdMasquer">Masquer</button><butt  
on id="cmdAfficher">Afficher</button>  
<br><br>  
<div id="div1"  
    style="width:80px;height:80px;backgro  
und-color:red;"></div><br>  
<div id="div2"  
    style="width:80px;height:80px;backgro  
und-color:green;"></div><br>  
<div id="div3"  
    style="width:80px;height:80px;backgro  
und-color:blue;"></div>
```

Sliding

- Fonctions
 - slideDown()
 - slideUp()
 - slideToggle()
- Les fonctions acceptent les paramètres vitesse, et rappel.

Exemple SlideDown

Script

```
$("#p1").click(function () {  
  
    $("#p2").slideDown("slow");  
});
```

Style

```
#p1,#p2  
{  
padding:5px;  
text-align:center;  
background-color:#e5eccc;  
border:solid 1px #c3c3c3;  
}  
  
#p2
```

```
{  
padding:50px;  
display:none;  
}  
HTML  
<div id="p1">Cliquez ici</div>  
<div id="p2">Un panneau</div>
```

animate

- Syntaxe: `$(selecteur).animate({params}, vitesse, rappel);`
- params: propriétés css à animer

Exemple Animate

Script

```
$("#button").click(function () {  
    $("#div").animate({  
        left: '250px',  
        opacity: '0.5',  
        height: '150px',  
        width: '150px'  
    });  
});  
});  
}
```

HTML

```
<button>Animer</button>
```

```
<div
```

```
style="background:#98bf21;height:100p  
x;width:100px;position:absolute;">  
</div>
```

8-Fonctions HTML

Accès au contenu d'un élément HTML

- `text()`: obtenir ou modifier le contenu d'un élément html
- `html()`: obtenir ou modifier le contenu d'un élément html (les balises de l'élément sont comprises)
- `val()`: obtenir ou modifier la valeur d'un champ de formulaire
- `attr(nom_attribut,valeur)`: lire ou modifier la valeur d'un attribut.

Exemple:

Script

```
<script>
    $(//document.ready
        function () {
            // attacher un gestionnaire à
l'événement clic de b1
            $("#b1").click(function ()
{
// Lecture du contenu des éléments html
alert("La fonction val:" + $("#t1").val());
            alert("La fonction text:" +
$("#d1").text());
            alert("La fonction html:" + $("#d1").html());
//Modification
var contenu = prompt("Modifier le contenu de
l'élément div d1");
$("#d1").text(contenu);
var contenu = prompt("Modifier l'élément div
d2");
$("#d2").html(contenu);
var contenu = prompt("Modifier la valeur de
t1");
$("#t1").val(contenu);
// Lecture et modification de l'attribut size
            alert("La fonction
attr,size:" + $("#t1").attr("size"));

```

```
        var contenu =
prompt("Modifier la valeur de l'attribut size");
        $("#t1").attr("size",contenu);
        $("#t1").val(contenu);
    });
}
};
</script>
```

HTML

```
<div id="d1" > <p>DIV d1
    Les fonctions html DOM</p>
</div>
<input id="t1" type="text" value=""
size="55"/>
<input id="b1" type="button" value="button"
/>
<div id="d2"> DIV d2
    Les fonctions html DOM
</div>
```

Ajouter des éléments DOM

- appendTo() : ajoute le contenu à la fin de l'élément sélectionné.
 - Exemple: `$("#p").append("contenu");`
- prepend() : ajoute le contenu au début de l'élément sélectionné.
- after() : insère le contenu après l'élément sélectionné.
- before() : insère le contenu avant l'élément sélectionné

Supprimer des éléments DOM

- remove(): supprime l'élément sélectionné.
 - Exemples:
 - `$("#p").remove();`
 - `$("#p").remove(".c1");`
- empty(): efface le contenu de l'élément sélectionné.

9- JQuery css

- `addClass()`: associer une classe css à la sélection
 - `$("#div").addClass("c1");`
- `removeClass()`: supprimer une classe de la sélection
 - `$("#div").removeClass("c1");`
- `toggleClass()`: bascule entre `addClass` et `removeClass()`
 - `$("#div").toggleClass("c1");`
- `css()`: obtenir ou modifier la valeur d'une propriété css.
 - Lecture: `var couleur = $("#p").css("background-color");`
 - Modification:
 - `$("#p").css("background-color", "yellow");`
 - `$("#p").css({ "background-color": "yellow", "font-size": "200%" });`

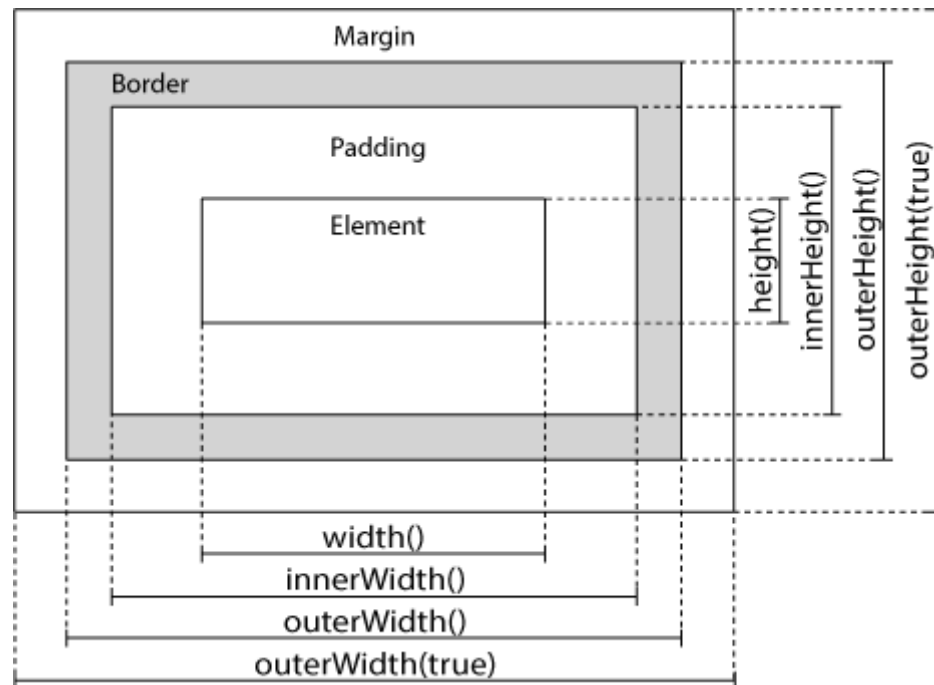
9- Dimensions JQuery

- Lecture/Modification

- width()
- height()

- Lecture

- innerWidth()
- innerHeight()
- outerWidth()
- outerHeight()



Exemple

Script

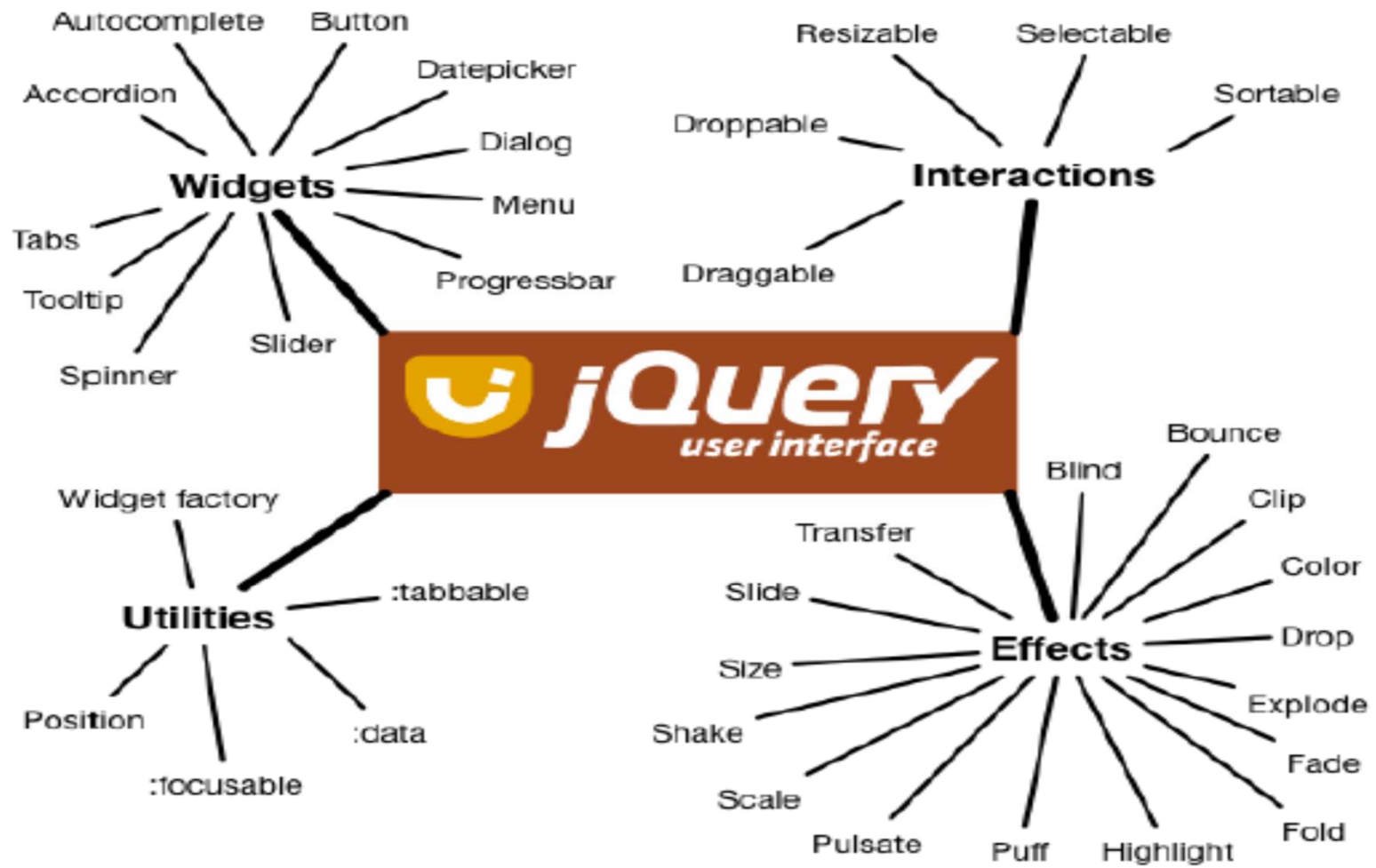
```
$("#b1").click(function () {  
    var lar = $("#t1").width();  
    alert(lar);  
    $("#t1").width(lar + 10);  
});
```

HTML

```
<input id="t1" type="text" />  
<input id="b1" type="button" value="size" />
```

10- JQuery UI

- JQuery UI basée sur JQuery est structurée en quatre catégories d'éléments
 - Widgets: composants de l'interface utilisateurs (Menu, Dialog,...)
 - Effets: ensemble d'animations et de transitions des éléments DOM
 - Interactions: les interactions permettent à l'utilisateur d'interagir avec les éléments DOM de la page web (draggable, sortable, droppable).
 - Utilitaires: ensemble d'outils utilisés en interne par JQuery UI
- Intégration dans une page
 - `<link href="jquery-ui.css" rel="stylesheet" />`
 - `<script src="external/jquery/jquery.js"></script>`
 - `<script src="jquery-ui.js"></script>`



Accordéon

- Chaque panneau est composé d'un élément contenant le titre du panneau et un autre le contenu.

Script

```
$(function () {
```

```
$("#ac").accordion({active:1 });  
});
```

Html

```
<div id="ac" style="width:200px">  
  <h3>Panneau 1</h3>  
<div id="p1" >  
Panneau 1</div>  
  <h3>Panneau 2</h3>  
<div id="p2" >
```

```
Panneau 2</div>
```

```
  <h3>Panneau 3</h3>
```

```
<div id="p3" >
```

```
Panneau 3</div>
```

```
</div>
```

Onglets

Script

```
$("#t").tabs();
```

Html

```
<div id="t">
  <ul>
    <li><a href="#t1">Onglet
1</a></li>
    <li><a href="#t2">Onglet 2</a></li>
    <li><a href="#t3">Onglet
3</a></li>
  </ul>
  <div id="t1" >
Onglet 1
</div>
```

```
<div id="t2">
Onglet 2
</div>
<div id="t3" >
Onglet 3
</div>
</div>
```

Menus

Script

```
$("#menu").menu()
```

Style

```
.ui-menu {width:180px}
```

Html

```
<ul id="menu">
<li><a href="#">Menu 1</a></li>
<li><a href="#">Menu 2</a></li>
<li><a href="#">Menu 3</a></li>
  <li> <a href="#">Menu 3 1</a>
  <ul>
    <li><a href="#">Menu 3 1
1</a></li>
    <li> <a href="#">Menu 3 1 2</a>
```

```
</ul>
    <li><a
href="#">Menu 3 1 2 1</a></li>
    <li><a
href="#">Menu 3 1 2 2</a></li>
  </ul>
</li>
</ul>
</li>
</ul>
```

Boîtes de dialogues

Script

```
$("#dialog").dialog();
```

Html

```
<div id="dialog" title="Un exemple  
de boîte de dialogue" class="ui-  
dialog-title">  
    <p class="ui-dialog-  
content">Contenu</p>  
</div>
```

Exemple 2

Script

```
$("#dialog").dialog({
```

```
    modal:true,  
    dialogClass: "no-close",  
    buttons: [{  
        text: "OK",  
        click: function () {  
            $(this).dialog("close");  
        }  
    }  
    ]  
});
```