



#### **UP Web**

## **jQuery**

#### Année universitaire 2019-2020









## Plan

- Introduction
- Bibliothèque JQUERY
- La fonction jQuery()
- Les évènements
- Quelques effets
- Manipulation du DOM
- JQuery UI
- jQuery Plugins





## De JavaScript... à jQuery

- La naissance de JavaScript
  - 1995 : **Brendan Eich** développe pour Netscape Communications Corporation, un langage de script, appelé Mocha, puis LiveScript et finalement JavaScript
  - Javascript est intégré dans le navigateur Netscape 2. Succès immédiat.
- La guerre des navigateurs
  - Netscape et Microsoft (avec JScript dans Internet Explorer) ont développé leur propre variante de JavaScript avec des fonctionnalités supplémentaires et incompatibles, notamment dans la manipulation du DOM (modèle objet du navigateur WEB)







Simplifier votre code Javascript avec Jquery Utiliser des bibliothèques prédéfinis pour simplifier votre code.





## **Exemple**

#### Avec du JavaScript pure

var divs=document.getElementById('div');
divs.style.display='none';

#### **Avec jQuery**

\$('#div').hide();





### **Exemple**

#### Avec du JavaScript pure

```
function toggle() {
  var ele = document.getElementById(" form1");
  if(ele.style.display == "none") {
      ele.style.display = "block";}
  else { ele.style.display = "none"; }
  Avec jQuery
  $('#form1').toggle();
```



# Bibliothèque JQUERY(1/3)

- Une bibliothèque javascript open-source et cross-browser
- Elle permet de traverser et manipuler très facilement l'arbre DOM des pages web à l'aide d'une syntaxe fortement similaire à celle d'XPath.
- JQuery permet par exemple de changer/ajouter une classe CSS, créer des animations, modifier des attributs, etc.
- Gérer les événements javascript
- Faire des requêtes AJAX simplement.



## Bibliothèque JQUERY(2/3)

Disponible sur le site de Jquery

```
http://jquery.com/:
```

2 formats sont disponibles en téléchargement : l'un compressé (.min) dit de production et l'autre non compressé. <script type="text/javascript" src="jquery.js"></script>

Ou directement sur Google code

```
<script type="text/javascript"
src="http://ajax.googleapis.com/ajax/libs/jquery/1/jquery.min.js">
</script>
```



# Bibliothèque JQUERY(3/3)



## La fonction jQuery() (1/5)

```
$('selecteur').action();

- Sélecteur natif (H1,H2,p...)
- Sélecteur prédéfini (Id, class)

- Keypress }
- ...
```

```
Function() {
    $('selecteur').action();
}
```



```
La fonction jQuery() (2/5)
```

```
1 jQuery(document).ready(function() {
2 // Ici, le DOM est entièrement défini
3 });
```

```
1 $(document).ready(function() {
2 // Ici, le DOM est entièrement défini
3 });
```

```
1 $(function() {
2 // Ici, le DOM est entièrement défini
3 });
```



## La fonction jQuery() (3/5)

• jQuery repose sur une seule fonction : jQuery() ou \$()

\$(Selector).Action();

- C'est une fonction JavaScript
- Elle accepte des paramètres
- Elle retourne un objet

Find Something, Do Something (actions)



# La fonction jQuery() (4/5)

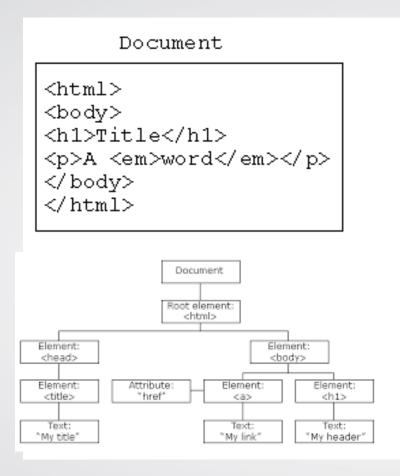
- \$ accepte un sélecteur CSS en argument
- \$ accepte des ID :
  - \$('#nomID') retourne **un** élément Equivalent à document.getElementById
- \$ accepte des classes :
  - \$('.nomClasse') retourne tous les éléments qui correspondent à cette classe
- \$ accepte plusieurs sélecteurs
  - \$('.article, .nouvelles, .edito')

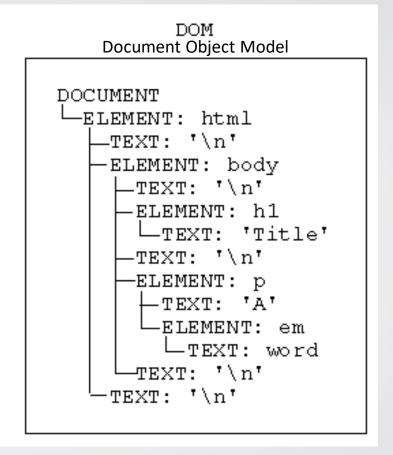
#### id ou class?

- id: pour distinguer un unique élément
- class: pour distinguer plusieurs éléments avec des caractéristiques communes



## La fonction jQuery() (5/5)







### Callback



#### **Callbacks**

Une fonction de callback est exécutée après que l'effet courant soit terminée.

Syntaxe typique: \$(selector).hide(speed,callback);

```
Exp1:
```

```
$("button").click(function(){
    $("p").hide("slow",function(){
    alert("Le paragraphe est maintenat caché"); }); });

Exp2:
$("button").click(function(){
    $("p").hide(1000);
alert("Le paragraphe est maintenat caché");
});
```



## Chainage



#### Chainage

La technique appelée chaînage nous permet d'exécuter des commandes multiples jQuery, l'un après l'autre, sur le même élément

#### Exp:

\$("#p1").css("color","red").slideUp(2000).slideDown(2000);



# Modifier les attributs et les propriétés (1/2)

Les attributs (valeurs initiales présentes dans le HTML)

```
$("#id_element").attr("title","toto");
a = $("#id_element").attr("title");
$("#id_element").removeAttr("title");
```

Les propriétés (valeurs courantes dans le DOM)

```
$("#id_element").prop("checked",true);
a = $("#id_element").prop("checked");
$("#id_element").removeProp("toto");
```

• La valeur :

```
$("#id_element").val("coucou");
v = $("#id_element").val();
```

• Modifier le style: changer la couleur des textes de liens \$("a").css("color","red")



# Modifier les attributs et les propriétés (2/2)

Modifier la visibilité: Cacher, Montrer, Cacher/Montrer

```
$("div.toto").hide()
$("div.toto").show()
$("div.toto").toggle()
```

- Modifier le contenu: Ajouter du texte... à la fin au début \$("div.toto").append("fin div") \$("div.toto").prepend("debut div")
- Modifier un attribut: changer la source de l'image d'id fleurs \$("img #fleurs").attr("src","images/tulipe.png")
- Récupérer un attribut : récupérer l'attribut alt de l'image d'id fleurs

```
$("img # fleurs ").attr("alt")
```

 Ajouter une classe : ajouter la classe important à l'élément d'id toto

\$("div#toto").class("important")



## Parcourir les éléments



```
Appliquer une fonction sur un ensemble d'éléments : $("#list").find("li").each(function(i) { $(this).append( "Je suis le numero " + i); });

Récupérer la liste des fils d'un élément : $('#list').children().css('background-color', 'red');

Récupérer le premier élément : $('#list').children().first().css('background-color', 'red');

Récupérer le dernier élément : $('#list').children().last().css('background-color', 'red');
```



## Ajouter, supprimer des éléments

#### Modifier et récupérer le contenu des éléments :

```
$("#list").find("li").html("<b>toto</b>");
$("#list").find("li").text("<b>toto</b>");
c_html = $("#list").find("li").html(); c_text = $("#list").find("li").text();
```

#### Ajouter dans des éléments :

```
$("#list").find("li").append("toto");
```

#### Supprimer un ensemble d'éléments du DOM :

```
$("#list").find("li").remove();
```

#### Insérer avant et après des éléments :

```
$('#test').children().before("avant chaque element);
$('#test').children().after("apres chaque element);
```



## jQuery et CSS



#### Les classes css:

```
r = $("#id_element").hasClass("maClasseCss");
$("#id_element").removeClass("maClasseCss");
// La classe est ajoutée si non présente ou supprimée si déjà
présente
Jouer avec le style :
$("#id_element").css('background-color', 'red');
color = $("#id_element").css('background-color');
Jouer avec la hauteur et la largeur :
$("#id_element").height(100);
h = $("#id_element").height();
$("#id_element").width(100);
w = (\#id_element).width();
Connaître la position d'un élément :
p = $("#id_element").offset(); // par rapport au document
alert(p.left+" "+p.top);
```



## Les événements (1/3)

Un événement est une action remplie par le navigateur, ou plus généralement l'utilisateur.

#### L'écoute sur la souris

Fonctionnement	Action
Clic	click()
Double-clic	dblclick()
Passage de la souris	hover()
Rentrer dans un élément	mouseenter()
Quitter un élément	mouseleave()
Presser un bouton de la souris	mousedown()
Relâcher un bouton de la souris	mouseup()
Scroller (utiliser la roulette)	scroll()



## Les événements (2/3)



#### L'écoute sur le clavier

Fonctionnement	Action
Se lance lorsqu'une touche du clavier est enfoncée	Keydown()
se lance lorsqu'on maintient une touche enfoncée	Keypress()
se lance lorsqu'on relâche une touche préalablement enfoncée.	Keyup()

#### Le cas des formulaires

Action	Fonction
Focalisation	focus()
Sélection (p.e. dans une liste)	select()
Changement de valeur	change()
Envoi du formulaire	submit()



## Les événements



#### Exemple:

```
<html>
<head>
<style> div { color: red; } </style> <script src="iguery-min.is"></script>
</head>
<body>
<select name="pays" multiple="multiple">
<option>France
<option selected="selected">U.S.A</option>
<option>Burkina Fasso
<option selected="selected">Japon</option> <option>Mexique</option>
<option>Afrique du Sud
<option>Tahiti
</select> <div></div>
// On attache un événement change au select, qui permet de // récupérer chaque option
sélectionnée pour les afficher // dans la div. Puis, on déclenche l'évenement.
<script>
$( "select" )
.change(function (){
var str = "":
$( "select option:selected" ).each(function() {
   str += $(this).text() + "";
});
$( "div" ).text( str ); })
.change();
```



</script>

</body> </html>

## Autres fonctions



**jQuery.each()** : une fonction générique d'itération qui permet d'itérer sur des objets ou des tableaux.

**jQuery.parseHTML()**: une fonction qui transforme une chaine en un tableau de noeuds du DOM

**jQuery.parseJSON()**: une fonction qui prend une chaine JSON et retourne l'objet JavaScript.



# Quelques effets (1/2)

#### Apparition, disparition

Action	Fonction	
show() hide()	Afficher et cacher des éléments	
show(vitesse) hide(vitesse)	Afficher et cacher des éléments avec une vitesse.	
	Cette vitesse est indiquée par des mots-clefs	
	("slow","normal" ou "fast") ou le nombre de	
	millisecondes que doit durer l'animation.	
Toggle() toggle()	Permettent de basculer d'un mode d'affichage à un	
	autre (un élément caché devient visible, ou un	
	élément visible devient caché).	
Envoi du formulaire	submit()	



## Quelques effets (2/2)

- slideDown() et slideUp() permettent de faire apparaître (respectivement disparaître) un élément à la manière d'un store se déroulant ou s'enroulant.
- slideToggle() permet de basculer d'un mode d'affichage à un autre.
- fadeIn(vitesse) et fadeOut(vitesse) permettent de faire progressivement apparaître (ou disparaître) un élément en jouant sur sa transparence.
- **slideDown**() et **slideUp**() permettent de faire apparaître (respectivement disparaître) un élément à la manière d'un store se déroulant ou s'enroulant.
- slideToggle() permet de basculer d'un mode d'affichage à un autre.
- fadeIn(vitesse) et fadeOut(vitesse) permettent de faire progressivement apparaître (ou disparaître) un élément en jouant sur sa transparence.



## Récapitulatif (1/3)



#### Obtenir/Modifier du contenu HTML

- text () : Définit ou retourne le contenu du texte de certains éléments
- html () : Définit ou retourne le contenu des éléments sélectionnés (y
- compris les balises HTML)
- val () : Définit ou retourne la valeur de champs de formulaire

#### Ajouter du contenu HTML

```
append ()
```

prepend ()

after ()

before ()

#### Suppression d'élément ou de contenu

remove()

empty()



## Récapitulatif (2/3)



#### **Manipulation CSS**

addClass(): Ajout

removeClass(): Suppression

toggleClass(): Inversion de Ajout/Suppression de classes

css(): Positionner un attribut de style

#### Jouer avec la visibilité

fadeIn(): \$(selector).fadeIn(speed,callback);

fadeOut() : \$(selector).fadeOut(speed,callback);

fadeToggle() : \$(selector).fadeToggle(speed,callback);

fadeTo() : \$(selector).fadeTo(speed,opacity,callback); (opacité

comprise entre 0 et 1)

#### Jouer avec le sliding

slideDown() : \$(selector).slideDown(speed,callback);

slideUp() : \$(selector).slideUp(speed,callback);

slideToggle() : \$(selector).slideToggle(speed,callback);

#### Animations (effets de déplacement)

animate() : \$(selector).animate({params},speed,callback);



## Récapitulatif (3/3)



#### Voici quelques événements communs du DOM:

Souris	Clavier	Formulaire	Document/Window
click	keypress	submit	Load
dblclick	keydown	change	resize
mouseenter	keyup	focus	scroll
mouseleave		blur	unload



## jQuery UI(1/2)



- Un ensemble de composants graphiques téléchargeable à l'adresse http://jqueryui.com/download.
- jQuery est la bibliothèque centrale., jQueryUI est construit au- dessus de lui. Si vous utilisez jQueryUI, vous devez également inclure jQuery.

Des « comportements » (interactions)

- draggable : pour glisser-déplacer un élément <u>http://jqueryui.com/demos/draggable/</u>
- droppable : pour « déposer » un élément
   <a href="http://jqueryui.com/demos/droppable/">http://jqueryui.com/demos/droppable/</a>
- resizable : pour redimensionner un élément <u>http://jqueryui.com/demos/resizable/</u>
- selectable : pour sélectionner des éléments à la souris http://jqueryui.com/demos/selectable/



## jQuery UI(2/2)



#### Des « widgets »

- « calendrier » <a href="http://jqueryui.com/demos/datepicker/">http://jqueryui.com/demos/datepicker/</a>
- « accordéon <u>»Http://jqueryui.com/de</u>« accordéon
   »Http://jqueryui.com/dem<u>os/accordion/</u>
- boîte de dialogue <a href="http://jqueryui.com/demos/dialog/">http://jqueryui.com/demos/dialog/</a>
- barre de progression <a href="http://jqueryui.com/demos/progressbar/">http://jqueryui.com/demos/progressbar/</a>
- curseur <a href="http://jqueryui.com/demos/slider/">http://jqueryui.com/demos/slider/</a>
- onglets <a href="http://jqueryui.com/demos/tabs/">http://jqueryui.com/demos/tabs/</a>
- des « effets » http://jqueryui.com/demos/effect/
- Clignotement, disparition, apparition, éclatement, transition...
- Une collection de thèmes
   <a href="http://jqueryui.com/themeroller/">http://jqueryui.com/themeroller/</a>



## jQuery Plugins



Un plugin jQuery est tout simplement une nouvelle méthode que nous utilisons pour étendre l'objet prototype de jQuery. En étendant l'objet prototype vous activez tous les objets jQuery pour hériter de toutes les méthodes que vous ajoutez.

https://plugins.jquery.com/

- <u>ui</u> (542)
- jquery (482)
- <u>form</u> (285)
- <u>animation</u> (273)
- input (252)
- <u>image</u> (210)
- <u>responsive</u> (184)
- slider (172)
- <u>ajax</u> (154)
- scroll (140)



## Références



- jquery.com
- https://api.jquery.com
- jqueryui.com
- jqueryPlugins
- plugins.jquery.com

