



Chrystel Moreau

- **jQuery** est une bibliothèque JavaScript qui va vous simplifier :
 - le parcours et la modification du DOM
 - la gestion des événements
 - l'implémentation d'effets visuels / animations
 - la manipulation du CSS
 - l'implémentation AJAX
- **jQueryUI** (User Interface) est une librairie de composants permettant d'ajouter des interactions et des widgets.
- **jQuery Plugins** (validation de formulaire, ...)

Comparatif jQuery / JavaScript



	jQuery	JavaScript
Sélecteurs	<pre>var divs=\$('div');</pre>	<pre>var divs=document.querySelectorAll('div');</pre>
Evènement	<pre>\$('a').click(function () { // code });</pre>	<pre>[].forEach.call(document.querySelectorAll('a'), function(el) { el.addEventListener('click', function () { // code }) });</pre>
Manipulation	<pre>\$('body').append('<p>');</pre>	<pre>document.body.appendChild(document.createElement('p'));</pre>

jQuery repose sur une seule fonction **jQuery()**, notation abrégée **\$()** qui permettra de sélectionner des éléments dans votre page web.

Syntaxe :

`$("sélecteur de balise"[,contexte]). méthode`

Exemple :

- **`jQuery("h1").hide();` ou `$("h1").hide();`**
- **`var context = $("h1");`
`$("a", context).hide();`**

Sélecteurs CSS `$("élément")`

- `$("*")` : tous les éléments HTML
- `$("x")` : élément x
- `$("x y")` : éléments y descendants de x
- `$("x > y")` : éléments y fils direct de x
- `$("x + y")` : éléments y qui suivent x
- `$("x,y")` : éléments x et y
- `$("#valID")` : élément avec id="#valID"
- `$(".classe")` : élément avec class="classe"

Les sélecteurs permettent de sélectionner un ou plusieurs éléments du DOM pour les manipuler.

Sélecteurs d'attributs `$("élément[attribut]")`

- `$("elt[attr]")` : tous les éléments **elt** ayant l'attribut **attr**
- `$("elt[attr1][attr2]")` : tous les éléments **elt** ayant les attributs **attr1** et **attr2**
- `$("elt[attr]=val")` : tous les éléments **elt** ayant l'attribut **attr** dont la valeur est exactement **val**
- `$("elt[attr]^=val")` : tous les éléments **elt** ayant l'attribut **attr** commençant par **val**

...

Sélecteurs de position `$("élément:pseudo-classes")`

- `$("x:first")` : 1er élément x du document
- `$("x:first-child")` : 1er fils de l'élément x du document
- `$("x:nth-child")` : nième fils de l'élément x du document
- `$(":eq(n)")` : élément dont l'index est n

...

Sélecteurs de formulaire `$("élément:pseudo-classes")`

- `$(":input")` : éléments `<input>`, `<textarea>`, `<select>`, `<button>`
- `$(":checkbox")` : élément de type checkbox
- `$(":checked")` : éléments (checkbox et radio) positionnés à **checked**
- `$(":checkbox:checked:enabled")` : éléments checkbox positionnés à **checked** et **actifs**

...

Les méthodes permettent d'appliquer des actions, modifications ou simples lectures d'attributs ou propriétés d'un objet

Utilisation : `$("#selecteur").methode();`

Manipulation du DOM

`$("#element").html('texte');` : remplace le contenu d'un élément

`$("#element").text();` : récupère le texte d'un élément

`$("#element").append('texte');` : ajoute du contenu à un élément

`$("#element").prepend('texte');` : ajoute du contenu au début d'un élément

`$("#element").remove();` : supprimer un élément

`$("#element").empty();` : vider un élément

`$("#element").attr("href");` : récupère la valeur d'un attribut d'un élément

`$("#element").attr("href","http://...");` : modifie la valeur d'un attribut d'un élément

...

Modifier le style

`$("#element").css('background', 'gold');` : remplace le contenu d'un élément

`$("#element").addClass('myclass');` : ajoute une classe de style à un élément

`$("#element").removeClass('myclass');` : supprime une classe de style à un élément

...

Animer un élément

`$("#element").hide();` : cacher un élément

`$("#element").show();` : montrer un élément

`$("#element").fadeIn();` : faire apparaître un élément avec animation de fondu

...

Associer une action à un élément => `$("#element").event(<function>);`

`$("document").ready(function() { ...});` : quand le DOM est prêt

`$("#element").click(function() { alert(0)});` : click de la souris

`$("#element").mouseover(function() { $(this).css('color','red'); });` : passage de la souris

les évènements les plus utiles :

blur, change, click, dblclick, focus, hover, keydown, keyup, mousedown, mouseup, ready, resize, scroll, submit

Parcourir un Objet jQuery

`$('.selecteur').each(function() { //votre code });` : parcours un objet

`$("#p").eq(n).hide();` : cache le contenu d'un élément <p> à l'index n

- **\$.load()** permet de récupérer du contenu à insérer dans notre page.

```
$("#div_result").load('result.txt'); // chargement de result.txt dans la balise #div_res
```

- **\$.ajax()** permet d'envoyer/recevoir du contenu et aussi de laisser le choix des actions à faire.

```
$("#bouton").click(function() {  
    $.ajax({  
        url : 'ressource.php', // la ressource cible coté serveur  
        type : 'POST', // type de la requête HTTP (GET/POST)  
        data : 'e1='+val1+'&e2='+val2, // éléments envoyés  
        dataType : 'html', // données à recevoir (PHP,HTML,JSON,XML)  
        success : function (data) {}, // fonction à appeler si la requête abouti  
        error : function () {} // fonction à appeler si erreur  
    });  
});
```

- **\$.get()** et **\$.post()** les raccourcis de \$.ajax

```
$("#bouton").click(function() {  
    $.get(  
        'ressource.php', // la ressource cible coté serveur  
        'false', // pas d'élément envoyé  
        'json', // données à recevoir de type JSON  
        function (data,status) { // code } // fonction à effectué en cas de succès  
    });  
});
```


- Déclarer un plugin

```
(function($) {  
    $.fn.monPlugin = function ( paramètre, ... ) {  
        // Ici vos scripts  
    };  
})(jQuery);
```



création du Plugin

- Appeler un plugin

```
$('#element').monPlugin();
```



appel du Plugin

1. Inclure la **bibliothèque**
2. Inclure l'exécution de la **fonction jQuery(...)** ou **\$(...)** à la **fin** du chargement du **DOM**

```
<html>  
<head>  
  <title>test jquery</title>
```

1 `<script type="text/javascript" src="jquery/js/jquery.js"> </script>`

```
<script type="text/javascript">
```

2 `$(document).ready(function() {`

```
  // la fonction de rappel (ou callback) associée à la méthode ready est exécutée  
  // seulement quand le document est chargé
```

```
  });
```

```
</script>
```

```
</head>
```

```
...
```

1. Inclure la **bibliothèque**
2. Inclure l'exécution de la **fonction jQuery(...)** ou **\$(...)** à la **fin** du chargement du **DOM**
3. Agir sur le document

```
<html>  
<head>  
  <title>test jquery</title>
```

```
  <script type="text/javascript" src="jquery/js/jquery.js"> </script>
```

```
  <script type="text/javascript">
```

```
    $(document).ready( function() {
```

```
      $('h1').css('color', 'red');
```

```
    });
```

```
  </script>
```

```
</head>
```

```
...
```

← modifie le style des `<h1>`
et les colorie en rouge

méthode **.get()** :
\$.get(url [,data] [,function(..)] [,dataType]);

```
<script type="text/javascript">
```

```
$(document).ready( function() {
```

```
    $('#regions').change( function () {
```

```
        $.get ( 'chef_lieu.php',  
              {region : $('#selected').text()},
```

```
              function (data) { $('#chef_lieu').html(data); }
```

```
        );
```

```
    });
```

```
});
```

```
</script>
```

```
...
```

```
<body>
```

```
  pays
```

```
  <select id="regions">
```

```
    <option value='France'> PACA </option>
```

```
    <option value='France'> Ile de France </option>
```

```
  </select>
```

```
  <span id="chef_lieu"> </span>
```

```
</body>
```

// URL vers laquelle est envoyé la demande
// données à envoyer vers le serveur
// fonction à exécuter quand la méthode est terminée

jQuery UI est une surcouche de jQuery qui propose des outils pour créer des interfaces graphiques interactives. Ses fonctionnalités se divisent en différentes catégories :

□ **Interactions**

- Donner des propriétés à des éléments HTML => drag, drop, resize, ...

□ **Widgets**

- Objets "clé en main " qui proposent des fonctionnalités de haut niveau => Accordion, Tabs, Dialog, DatePicker ...

Interaction: **draggable**

1. Inclure les **bibliothèques**
2. Inclure le **style CSS**

```
<html>
<head>
  <title>test jquery</title>
  <script type="text/javascript" src="jquery/js/jquery.js"></script>
  <script type="text/javascript" src="jquery/js/jquery-ui.js"></script>
  <link type="text/css" href="jquery/css/ui-lightness/jquery-ui-1.8.19.custom.css" rel="stylesheet">
</head>
<body>
  <div style="border: 1px solid gray; width: 100px; height: 100px; display: inline-block; vertical-align: middle; margin-right: 20px; text-align: center; line-height: 100px;">déplace moi</div>
  <div style="border: 1px solid gray; width: 100px; height: 100px; display: inline-block; vertical-align: middle; text-align: center; line-height: 100px;">déplace moi</div>
</body>
</html>
```



Interaction: **draggable**

1. Inclure les **bibliothèques**
2. Inclure le **style CSS**
3. Définir un **attribut id** pour le champ cible
4. Associer la fonction **draggable**

```
<html>
<head>
  <title>test jquery</title>
  <script type="text/javascript" src="jquery/js/jquery.js"></script>
  <script type="text/javascript" src="jquery/js/jquery-ui.js"></script>

  <link type="text/css" href="jquery/css/ui-lightness/jquery-ui-1.8.19.custom.css" rel="stylesheet">

  <script type="text/javascript">
    $(document).ready(function() {
      $("#objdrag").draggable();
    });
  </script>
</head>
<body>
  <div id="objdrag" > <p>déplace moi</p></div>
</body>
</html>
```

4

3



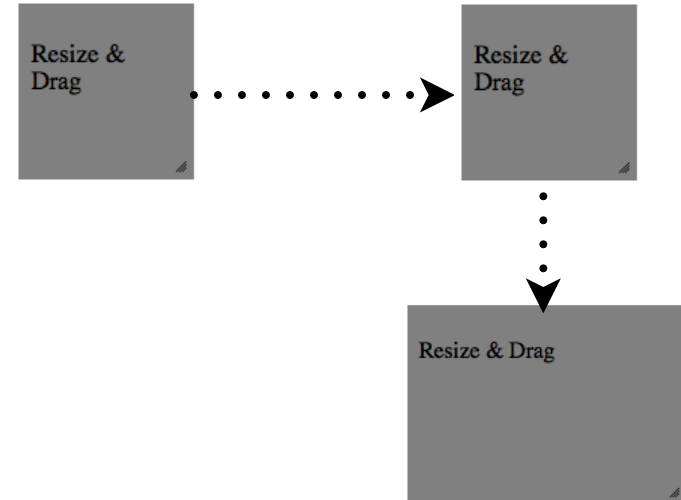
Interaction: **resizable**

4

```
<script type="text/javascript">
  $(document).ready(function() {
    $("#objet").draggable();
    $("#objet").resizable();
  });
</script>
</head>
```

3

```
<body>
<div id="objet" > <p>Resize & Drag</p></div>
</body>
</html>
```



Widget Accordion

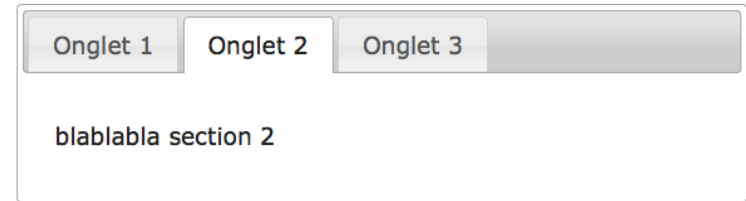
4

```
<script type="text/javascript">
  $(document).ready ( function() {
    $("#accordeon").accordion();
  })
</script>
</head>
<body>
  <div id="accordeon">
    <h3>Section 1</h3> <div> <p>bla bla bla bla section 1</p> </div>
    <h3>Section 2</h3> <div> <p>bla bla bla bla section 2</p> </div>
    <h3>Section 3</h3> <div> <p>bla bla bla bla section 3</p> </div>
    <h3>Section 4</h3> <div> <p>bla bla bla bla section 4</p> </div>
  </div>
</body>
```

3



Widget Tabs



4

```
<script type="text/javascript">
  $(document).ready(function() {
    $("#onglets").tabs();
  });
</script>
</head>
```

3

```
<body>
  <div id="onglets">
    <ul>
      <li><a href="#onglet1">Onglet 1</a></li>
      <li><a href="#onglet2">Onglet 2</a></li>
      <li><a href="#onglet3">Onglet 3</a></li>
    </ul>
    <div id="onglet1">blablabla section 1</div>
    <div id="onglet2">blablabla section 2</div>
    <div id="onglet3">blablabla section 3</div>
  </div>
</body>
```

Widget Dialog

4

```
...  
<script type="text/javascript">  
    $(document).ready ( function() {  
        $("#dialogue").dialog();  
    });  
</script>  
</head>
```

```
<body>
```

3

```
<div id="dialogue" title="Basic dialog">  
    <p>Boîte de dialogue</p>  
</div>
```

```
</body>  
</html>
```



Widget DatePicker

...

```
<script type="text/javascript">  
  $(document).ready(function() {  
    $("#date1").datepicker();  
  });  
</script>  
</head>
```

```
<body>
```

```
Saisir une date: <input type="text" name="date" id="date1" >
```

```
</body>
```

```
</html>
```

Saisir une date:



4

3

Widget DatePicker

Saisir une date: =>



```
...  
<script type="text/javascript">  
  $(document).ready(function() {  
    $("#date1").datepicker({  
      dateFormat: "yy-mm-dd",  
      altField: "#Altdate",  
      altFormat: "DD, d MM, yy"  
    });  
  });  
</script>  
</head>
```

```
<body>  
Saisir une date: <input type="text" name="date" id="date1" >  
&nbsp;=> <input type="text" id="Altdate" size=30>  
</body>  
</html>
```

4

3

Widget slider

VARIATION [3]



...

```
<script type="text/javascript">
  $(document).ready(function() {
    $("#monslider").slider ( {
      min : 0,
      max : 10,
      value : 0,
      slide : function(event, ui) { $("#variation").html(ui.value); }
    });
  });
</script>
</head>
<body>
<b>VARIATION</b> &nbsp;   [ <span id="variation"> 0 </span> ]
<div id="monslider"></div>
</body>
</html>
```



4

3

jQuery permet d'intégrer des modules additionnels qui vous faciliteront la vie :

- validation de formulaire
- manipulation de tableau
- manipulation de graphe

Plugin : validate



Vérification de formulaire

1. Inclure les **bibliothèques**
2. Définir le **style** pour les erreurs

```
<html>
<head>
  <title>test jquery</title>
  <style type="text/css">
    .error {color:red }
  </style>

  <script type="text/javascript" src="jquery/js/jquery.js"></script>
  <script type="text/javascript" src="jquery/js/jquery.validate.js"></script>
```

```
<body>
<form>
  Saisir votre Nom: <input type='text' name='t1'> <br>
  <input type="submit">
</form>
</body>
</html>
```

Saisir votre Nom:

Valider

Vérification de formulaire

1. Inclure les bibliothèques
2. Définir le style pour les erreurs
3. Ajouter un **attribut id** au formulaire
4. Associer la **vérification** au formulaire


```
<html>
<head>
  <title>test jquery</title>
  <style type="text/css">
    .error {color:red }
  </style>

  <script type="text/javascript" src="jquery/js/jquery.js"></script>
  <script type="text/javascript" src="jquery/js/jquery.validate.js"></script>
  <script type="text/javascript">
    $(document).ready(function() {
      $("#formtest").validate();
    });
  </script>
```

4

```
<body>
<form id="formtest">
  Saisir votre Nom: <input type='text' name='t1'> <br>
  <input type="submit">
</form>
</body>
</html>
```

3



Saisir votre Nom:

Valider

Vérification de formulaire

```
<html>
<head>
  <title>test jquery</title>
  <style type="text/css">
    .error {color:red }
  </style>

  <script type="text/javascript" src="jquery/js/jquery.js"></script>
  <script type="text/javascript" src="jquery/js/jquery.validate.js"></script>

  <script type="text/javascript">
    $(document).ready(function() {
      $("#formtest").validate ( {
        rules: {t1: { required : true} },
        messages: {t1: { required : "entrer un nom" } }
      });
    });
  </script>
</head>

<form id="formtest" >
  Saisir votre Nom: <input type='text' name='t1'> <br>
  <input type="submit">
</form>
```

1. Inclure les bibliothèques
2. Définir le style pour les erreurs
3. Ajouter un attribut id au formulaire
4. Associer la vérification au formulaire
5. définir les **règles** et les **messages**

5

Saisir votre Nom: entrer un nom

Valider

Plugin : dataTable



1. Inclure les **bibliothèques**
2. Inclure **le style CSS**
3. Ajouter un **attribut id** au tableau

```
<html>
<head>
  <title>test jquery</title>
  <script type="text/javascript" src="jquery/js/jquery.js"> </script>
  <script type="text/javascript" src="jquery/js/jquery.dataTables.js"> </script>
  <link rel="stylesheet" type="text/css" src="jquery/css/jquery.dataTables.css"> </script>
</head>
<body>
  <table id="tableau">
    <thead>
      <tr><th>Commune</th>
        <th>Code postal</th>
        <th>Canton</th>
        <th>Population 2012</th>
      </tr>
    </thead>
    <tbody>
      ...
    </tbody>
  </table>
</body>
```

Plugin : dataTable



1. Inclure les **bibliothèques**
2. Inclure **le style CSS**
3. Ajouter un **attribut id** au tableau
4. Associer la fonction **dataTable** au tableau

```
<html>
<head>
  <title>test jquery</title>
  <script type="text/javascript" src="jquery/js/jquery.js"></script>
  <script type="text/javascript" src="jquery/js/jquery.dataTables.js"></script>
</script>
  $(document).ready(function() {
    $("#tableau").dataTable();
  });
</script>
```

4

```
<body>
  <table id="tableau">
    ...
  </table>
</body>
```

Commune	Code postal	Canton	Population 2012
Aubagne	13400	94	46100
Cassis	13260	54	7800
Marseille	13001	99	851400

Showing 1 to 3 of 3 entries

Previous 1 Next

1. Inclure les **bibliothèques**
2. Inclure **le style CSS**
3. Ajouter un **attribut id** au graphe
4. Associer la fonction **jqplot**

1

```
<script type="text/javascript" src="js/jquery.js"></script>  
<script type="text/javascript" src="js/jquery.jqplot.js"></script>  
<script type="text/javascript" src="js/jqplot.pieRenderer.min.js"></script>
```

2

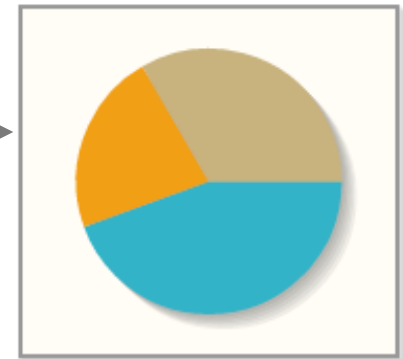
```
<link rel="stylesheet" href="css/jquery.jqplot.css" type="text/css" media="screen" />
```

4

```
<script>  
  $(document).ready( function() {  
    var data = [['G1', 40],['G2', 20],['G3', 30]];  
    $jqplot ("pie1", [data], {  
      seriesDefaults: {renderer: jQuery.jqplot.PieRenderer}  
    });  
  });  
</script>
```

3

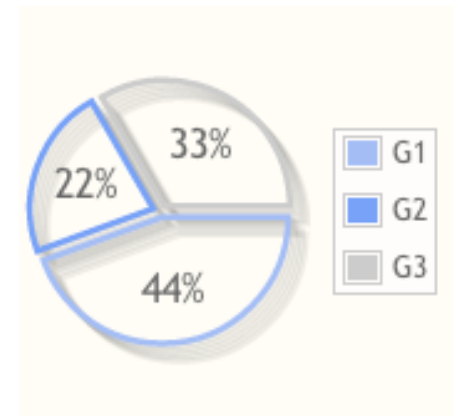
```
<body>  
<div id="pie1" style="text-align:center;height:200px;width:200px"> </div>  
</body>
```



1. Inclure les bibliothèques
2. Inclure le style CSS
3. Ajouter un attribut id au graphe
4. Associer la fonction du plot
5. Compléter la **fonction** du plot

```
<script>
  $(document).ready( function() {
    var data = [['G1', 40],['G2', 20],['G3', 30]];
    $jqplot ("pie1", [data], {
      seriesDefaults: {
        renderer: jQuery.jqplot.PieRenderer,
        rendererOptions: { showDataLabels: true,fill: false,sliceMargin: 2,padding : 4},
        grid : {borderWidth:0,shadow: false,
        legend: { show:true, location: 'e' },
        seriesColors: [ "#A9BBF4", "#829DF6", "#CCCCCC",]
      }
    });
  });
</script>
```

```
<body>
<div id="pie1" style="text-align:center;height:200px;width:200px"> </div>
</body>
```



- <https://jquery.com/>
- <https://jqueryui.com/>

- <http://jqueryvalidation.org/validate/>
- <https://www.datatables.net/>
- <http://www.jqplot.com/>