



jQuery

Jean-Christophe DUBOIS

jean-christophe.dubois@univ-rennes.fr

jQuery



- Bibliothèque JavaScript libre et multi-plateforme créée pour faciliter l'écriture de scripts côté client
- Créée par John Resig : “write less, do more”
- Simplification de JavaScript
- Mise à disposition des fonctionnalités suivantes :
 - Manipulation du DOM
 - Gestion d'événement
 - Manipulation des attributs et propriétés CSS
 - Effet visuel et animation
 - Requête AJAX

jQuery



- Site officiel : <https://jquery.com/>
- Syntaxe
 - `window.jQuery()`, `jQuery()` ou plus simplement `$()` pour faire appel à la librairie jQuery
- Installation
 - En local :

```
<script src="./jquery/jquery-3.6.3.min.js"></script>
```

- Via un CDN (Content Delivery Network) :

```
<script src="https://code.jquery.com/jquery-3.6.3.min.js"  
integrity="sha256-pvPw+upLPUjgMXy0G+8O0xUf+/Im1MZjXxxgOcBQBxu="   
crossorigin="anonymous"></script>
```

jQuery



- Vérification du bon chargement du DOM

- En JavaScript

```
document.addEventListener('DOMContentLoaded', function() {  
    // Le DOM est complet et prêt à être modifié  
})
```

- En jQuery

```
$(document).ready(function() {  
    // Le DOM est complet et prêt à être modifié  
});
```

- Ou plus simplement

```
$(function() {  
    // Le DOM est complet et prêt à être modifié  
});
```

jQuery



- Vérification du bon chargement de la page
 - En JavaScript

```
window.addEventListener("load", function() {  
    // La page est entièrement chargée (images, styles CSS...)  
})
```

- En jQuery

```
$(window).on("load",function() {  
    // La page est entièrement chargée (images, styles CSS...)  
});
```

Sélecteurs d'éléments

- Sélecteurs

- Balise

```
let elems = $("p"); // document.getElementsByTagName( "p");
```

- Classe

```
let elems = $(".maClasse"); // document.getElementsByClassName("maClasse");
```

- Identifiant

```
let elem = $("#monId"); // document.getElementById("monId");
```

- :not la négation

```
let elems = $(":not(p)"); // tout sauf p
```

Sélecteurs d'éléments

- Combinaison de plusieurs sélecteurs

```
let elems = $(".maClasse1, .maClasse2");  
            // éléments porteurs de .maClasse1 ou .maClasse2
```

- Sélecteur de contexte
 - Pour éviter les répétitions

```
let elems = $(".maClasse1, .maClasse2", "#monId");  
            // élénts avec .maClasse1 ou .maClasse2  
            // dans un élément avec #monId
```

Sélecteurs d'éléments

- Sélecteur d'attribut

- [attribut]/[attribut="valeur"] // porteur d'un attribut

```
let elems = $("a[href]"); // élnts a porteurs de l'attribut href
```

- [attribut^="valeur"] // attribut débutant par

```
let elems = $("a[href^='https']"); // élnts a avec href débutant par https
```

- [attribut\$="valeur"] // attribut finissant par

```
let elems = $("a[href]$='.fr']"); // élnts a avec href finissant par .fr
```

- [attribut*="valeur"] // attribut comportant

```
let elems = $("a[href]*='BUT']"); // élnts a avec href comportant BUT
```


Sélecteurs d'éléments

- Sélecteurs hiérarchiques

- > descendant direct
- + frère direct
- ~ frère indirect

```
let elems = $("main > p");           // p enfant main
// équivalent à document.querySelectorAll("main > p");
```

- :first-child/:last-child // premier/dernier enfant direct

```
let elem = $("li:first-child");       // 1er li premier enfant direct
```

- :nth-child // n^{ième} enfant direct

```
let elem = $("li:nth-child(2)");      // 2ème enfant direct li
let elems = $("li:nth-child(even)");  // 2èmes li enfants directs pairs
```

Sélecteurs d'éléments

- Sélecteurs de contenu

- :contains("unTexte")

```
let elems = $("p:contains('unTexte')"); // p contenant 'unTexte'
```

- :has("sélecteur")

```
let elems = $("p:has(span)"); // p contenant une balise span
```

- :parent

```
let elems = $("p:parent"); // p avec un contenu
```

- :empty

```
let elems = $("p:empty"); // p sans contenu
```

Sélecteurs d'éléments

- Sélecteurs de filtrage

- :first

```
let elem = $("p:first");           // premier p équiv. à $("p:eq(0)")
```

- :last

```
let elem = $("p:last");           // dernier p
```

- :eq(indice) // equal

```
let elem = $("p:eq(1)");          // 2ème p
```

- :gt(indice) // greater than

```
let elems = $("p:gt(3)");         // p au delà du 4ème p
```

- :lt(indice) // lower than

```
let elems = $("p:lt(3)");         // p en deçà du 4ème p
```

Sélecteurs d'éléments

- Sélecteurs de filtrage

- :header

```
let elems = $("header");           // toutes les balises h1...h6
```

- :even

```
let elems = $("p:even");           // p pairs
```

- :odd

```
let elems = $("p:odd");             // p impairs
```

- :visible

```
let elems = $("p:visible");         // p visibles
```

- :hidden

```
let elems = $("p:hidden");          // p masqués
```

Sélecteurs d'éléments

- **Exercice** : Instructions pour accéder aux éléments suivants :
 - tous les **.bloc**
 - tous les **titres enfants directs** de **#un**
 - tous les paragraphes **contenant** le texte **"CSS"**
 - **dernier** paragraphe contenu dans **#un**
 - tous les enfants de **#deux** **sauf** les paragraphes

Pour appliquer une couleur sur les éléments sélectionnés :

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

Sélecteurs d'éléments

- **Exercice** : Instructions pour accéder aux éléments suivants :
 - tous les **.bloc**
 - tous les **titres enfants directs** de **#un**
 - tous les paragraphes **contenant** le texte **"CSS"**
 - **dernier** paragraphe contenu dans **#un**
 - tous les enfants de **#deux** **sauf** les paragraphes

```
<body>
  <h1>CV</h1>
  <article id="un" class="bloc">
    <h2>Diplômes</h2>
    <p>Bac S</p>
    <p>BUT</p>
  </article>
  <article id="deux" class="bloc">
    <h2>Langages</h2>
    <p>CSS</p>
    <p>SCSS</p>
  </article>
</body>
```

Appliquez vos
instructions sur ce
code html

cf. CV.html

Accès aux éléments

- Méthodes de filtrage d'éléments

- .first() // équiv. au sélecteur :first

```
$("#p").first().addClass("rouge"); // premier p
```

- .last() // équiv. au sélecteur :last

```
$("#p").last().addClass("rouge"); // dernier p
```

- .eq(indice) // équiv. au sélecteur :eq()

```
$("#p").eq(2).addClass("rouge"); // 3ème p (de 0 à ... n)
```

- .filter(sélecteur)

```
$("#p").filter(".maClasse").addClass("rouge"); // p avec .maClasse
```

- .not(sélecteur) // contraire de filter()

```
$("#p").not(".maClasse").addClass("rouge") // p sans .maClasse
```

Accès aux éléments

- Méthodes d'accès aux ascendants

- `.parent()/.parent("sélecteur")` // parent direct

```
$("#p").parent().addClass("rouge"); // chaque parent direct d'un p
$("#p").parent(".maClasse").addClass("rouge");
// chaque parent direct d'un p porteur de .maClasse
```

- `.parents()/.parents("sélecteur")` // ascendants

```
$("#p").parents("div").addClass("rouge"); // tous les ascendants div
```

- `.closest("sélecteur")` // ascendant le + proche

```
$("#p").closest("div").addClass("rouge"); // ascendant div de p
// le plus proche
```


Accès aux éléments

- Méthodes d'accès aux descendants

- `.children()/children("sélecteur")` // enfants directs

```
$("#div").children("p").addClass("rouge"); // les enfants p directs
```

- `.find("sélecteur")` // enfants directs ou non

```
$("#div").find("*").addClass("rouge"); // tous les descendants  
// directs ou indirects
```

Méthodes d'accès aux éléments

- Méthodes d'accès aux frères

- `.siblings()/.siblings("sélecteur")` // tous les frères

```
$("#p").siblings("h3").addClass("rouge"); // tous les frères h3
```

- `.next()/.prev()` // frère suivant/précédent

```
$("#p").next().addClass("rouge"); // frère suivant
```

- `.nextAll()/.prevAll()` // tous les frères suivants/précédents

```
$("#p").nextAll().addClass("rouge") // tous les frères suivants
```

- `.nextUntil("sélecteur")/.prevUntil("sélecteur")`

```
$("#p").nextUntil("h4").addClass("rouge") // tous les frères après p  
// situés avant le prochain h4
```

Modification du DOM

- Méthodes de modification de contenu

- .text()/text("NvTexte") retourne ou met à jour le contenu

```
$("#nom").text(); // retourne le contenu textuel de #nom
```

- .html()/html("NvCode") retourne ou m.à.j. le code HTML

```
$("#nom").html("<em>Ok ?</em>"); // mise à jour du contenu html
```

- .val()/val("NvVal") retourne ou m.à.j. la valeur d'un champ

```
$("#nom").val(); // retourne la valeur du champ #nom
```

- .attr("attrib")/attr("attrib","NvVal") retourne ou m.à.j.

```
$("#nom").attr("href"); // retourne la valeur de l'attribut href  
$("#nom").attr("href","http://www.univ-rennes.fr"); m.à.j. de href
```

Modification du DOM

- Méthodes d'ajout

- .append("texte/code html") // ajout à la fin de l'élément

```
$("#nom").append("<em>Ok ?</em>"); // ajout à la fin de #nom
```

- .prepend("texte/code html") // ajout au début de l'élément

```
$("#nom").prepend("<em>Ok ?</em>"); // ajout au début de #nom
```

- .before("texte/code html") // ajout avant l'élément

```
$("#nom").before("<em>Ok ?</em>"); // ajoute avant #nom
```

- .after("texte/code html") // ajout après l'élément

```
$("#nom").after("<em>Ok ?</em>"); // ajoute après #nom
```

Modification du DOM

- Méthodes de suppression

- .remove()

```
$("#div").remove(); // supprime la balise
```

- .empty()

```
$("#div").empty(); // supprime le contenu de la balise
```

Modification d'un élément

- Méthodes de manipulation du CSS

- .addClass("classe")

```
$("#p").addClass("rouge"); // ajout de la classe rouge
```

- .removeClass("classe")/.removeClass()

```
$("#p").removeClass("rouge"); // suppression de la classe rouge
```

- .toggleClass("classe")

```
$("#p").toggleClass("rouge"); // ajout d'une classe si non présente  
// ou suppression si présente
```

- .css("prop","val")/.css({"prop1":"val1","prop2":"val2"});

```
$("#p").css("color","red"); // définition d'un style CSS
```

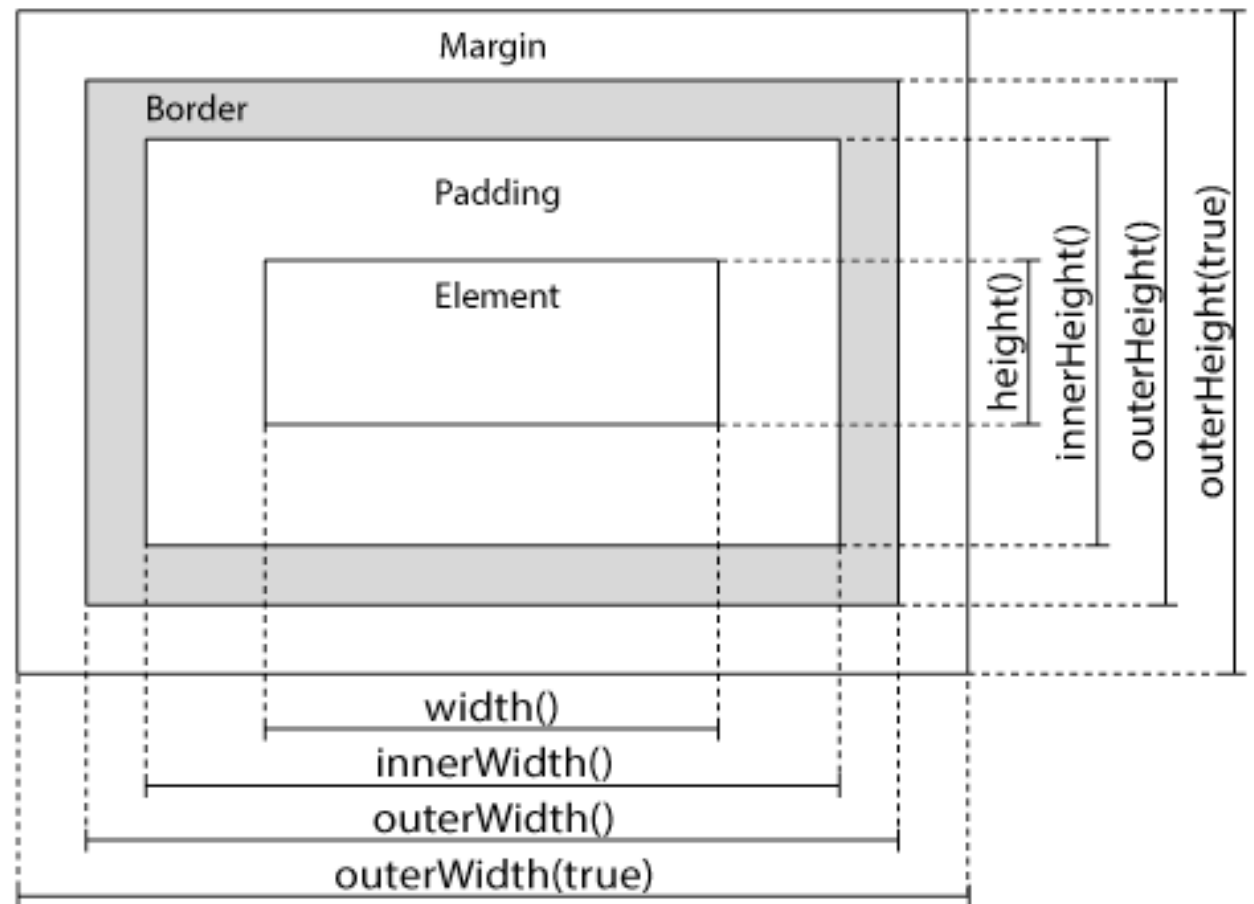
```
$("#p").css({"color":"red","font-size":"2em"}); // ajout avec un objet
```

Modification d'un élément

- Méthodes de dimensionnement des éléments

- .height()
- .width()
- .innerHeight()
- .innerWidth()
- .outerHeight()
- .outerWidth()
- .outerHeight(true)
- .outerWidth(true)

Retourne ou définit
une dimension





- Chaînage de méthodes

- Appels successifs :

- `$(selecteur).methode1();`
 - `$(selecteur).methode2();`
 - `$(selecteur).methode3();`

```
$("#header").addClass("maClasse1");  
$("#header").toggle("maClasse2");
```

- Appels chaînés :

- `$(selecteur).methode1().methode2().methode3();`

```
$("#header").addClass("maClasse1").toggle("maClasse2");
```


Modification de Document

- **Exercice** : Instructions pour réaliser les opérations suivantes :
 - Mettre tous les frères de l'élément porteur de **#un** en rouge
 - Supprimer tous les paragraphes enfants de l'élément porteur de **#deux**
 - Ajouter 2 p contenant *JS* et *jQuery* à l'élément porteur de **#deux**
 - Remplacer la classe **.bloc** par la classe **.cv** pour les parents directs de **h2**
 - Modifier l'adresse mail du webmaster.
L'accès à l'élément à modifier se fera en prenant en compte le fait que l'attribut **href** débute par la chaîne **"mailto:"**

Gestion d'évènement

- Gestionnaires d'évènements liés à la souris
 - Ajout d'un évènement :
 - .click()/.dblclick()/.mousedown()/.mouseup()
 - .scroll()/.mousemove()/.mouseover()/.mouseout()
 - .mouseenter()/.mouseleave() : mode **bouillonnement exclusif**

```
$("#p").on("click", function() {           // ou $("#p").click(function()  
    // Déclenchement lors d'un clic sur p  
});
```

- Association de plusieurs évènements à une même fonction

```
$("#p").on("click dblclick", function() {  
    // Déclenchement lors d'un clic ou d'un double-clic sur p  
});
```

Gestion d'évènement

- Gestionnaires d'évènements liés à la souris
 - Suppression d'un événement :
click/dblclick/mousedown/mouseup/scroll/mousemove
mouseover/mouseout/mouseenter/mouseleave

```
$("#p").off("click"); // suppression d'un évènement lié à un clic sur p
```

```
$("#p").off("click dblclick"); // suppression de plusieurs évènements
```

```
$("#p").off(); // suppression de tout évènement lié à p
```

- Arrêt de la propagation

```
$("#p").on("click", function(event) {  
    event.stopPropagation(); // Arrêt de la propagation  
});
```

Gestion d'évènement

- Exercice :
 - Ajouter un bouton "Sports" en fin de page
 - Programmer ce bouton "Sports" afin qu'il permette :
 - au 1^{er} clic : l'ajout, avant le footer, d'un élément article contenant :
 - un titre h2 "Sports"
 - un p contenant un sport saisi au clavier
 - aux clics suivants : l'ajout d'un seul p contenant un sport saisi au clavier

Effet visuel

- Effets de masquage ou d'affichage

- .hide()/.hide(durée)/.hide(durée, callBack) // masquage

```
$("#p").on("click", function() {  
    $("#strong").hide();           // masquage  
    $("#em").hide(1000);           // délai en millisecondes  
    $("#span").hide("slow");       // slow, normal ou fast  
    $("#mark").hide(1000, function() {  
        alert("Élément masqué");  
    });  
});
```

- .show()/.show(durée)/.show(durée, callBack) // affichage

Effet visuel

- Effets de fondu entrant ou sortant

- .fadeIn()/fadeIn(durée)/fadeIn(durée, callBack) // fondu

```
$("#p").on("click", function() {  
    $("#strong").fadeIn();           // affichage avec fondu  
    $("#em").fadeIn(1000);           // délai en millisecondes  
    $("#span").fadeIn("slow");       // slow, normal ou fast  
    $("#mark").fadeIn(1000, function() {  
        alert("Élément masqué avec fondu");  
    });  
});
```

- .fadeOut()/fadeOut(durée)/fadeOut(durée, callBack)
// fondu sortant

Effet visuel

- Effet de fondu entrant ou sortant selon le besoin
 - `.fadeToggle()/.fadeToggle(durée)`
`.fadeToggle(durée, callBack)` // fondu entrant ou sortant

```
$("#p").on("click", function() {  
    $("#strong").fadeToggle(); // affichage/masquage avec fondu  
    $("#em").fadeToggle(1000); // délai en millisecondes  
    $("#span").fadeToggle("slow"); // slow, normal ou fast  
    $("#mark").fadeToggle(1000, function() {  
        alert("Élément masqué ou affiché avec fondu");  
    });  
});
```

Effet visuel

- Effet de modification de l'opacité
 - .fadeTo(durée, opacité, callBack) // opacité paramétrée

```
$("#p").on("click", function() {  
    $("#strong").fadeTo(1000, 0.5);    // opacité à 50%  
    $("#span").fadeTo("slow", 0.5);    // slow, normal ou fast  
    $("#mark").fadeTo(1000, 0.5, function() {  
        alert("Opacité abaissée à 50%"); // avec callBack  
    });  
});
```


Effet visuel

- Effet de masquage ou d'affichage en accordéon
 - .slideDown()/slideDown(durée, opacité, callBack)

```
$("#p").on("click", function() {  
    $("#strong").slideDown();           // affichage en accordéon  
    $("#em").slideDown(1000);           // avec délai  
    $("#span").slideDown("slow");       // slow, normal ou fast  
    $("#mark").slideDown(1000, function() {  
        alert("Affichage en accordéon"); // avec callBack  
    });  
});
```

- .slideUp()/slideUp(durée, opacité, callBack)
- .slideToggle()/slideToggle(durée, opacité, callBack)

Animation

- Effet d'animation

- .animate({paramètres}, durée, callBack)

```
$("#p").on("click", function() {  
    $("#p").animate({top:"50px"}),      // déplacement vers le bas  
    $("#p").animate({left:"50px"}, 1000); // vers la droite + délai  
    $("#p").animate({width:"+=50px",height:"+=50px"}, "slow");  
        // 2 animations simultanées, 2 agrandissements  
    $("#p").animate ({left:"+50px"}, 1000, function() {  
        alert("Déplacement à droite");    // avec callBack  
    });  
});
```

Animation

- Effet d'animation
 - Animations successives

```
$("#p").on("click", function() {  
    $("#p").animate({width:"+=50px",height:"+=50px"}, "slow")  
    .animate ({left:"+50px"}, 1000);  
    // 2 agrandissements puis 1 déplacement  
});
```

- Animations simultanées

```
$("#p").on("click", function() {  
    $("#p").animate({width:"+=50px",height:"+=50px"}, "slow")  
    .animate ({left:"+50px"}, {"queue": false, "duration": "slow"});  
    // 2 agrandissements et 1 déplacement simultanés  
});
```

Animation

- Effet d'animation
 - .delay(durée)

```
$("#p").on("click", function() {  
    $("#p").animate({width:"+=50px",height:"+=50px"}, "slow")  
    .delay(2000)  
    .animate ({left:"+50px"}, 1000);  
    // 2 agrandissements, 1 pause, 1 déplacement  
});
```