3 méthodes : Html avec balises script, dans un fichier js, dans un serveur externe

Syntaxe html

$(‘’sélecteur’’).nomdelafonction() ;

Le sélecteur c’est comme ce qu’on faisait sur CSS pour désigner une classe ou une balise

Ex : div .entête {…}

* Pour les ciblés, on peut leur attribuer une classe, un id, la balise ou une pseudo classe
* On peut aussi mettre un évènement et le rattacher à un bouton.

Exemple : function hideLiCacher() {

$(".cacher").hide();

}

Rattacher à ce bouton

<button onclick="hideLiCacher()" type="button" class="btn btn-outline-dark">Dark</button>

Deuxième syntaxe

$(document).ready(function(){

$(‘’h1’’).click(function(){

Alert(‘’h1 va disparaitre’’) ;

$(this).hide() ;

}) ;

}) ;

This représente h1

On met tout l’évènement sur la même formule

Ne pas se tromper sur le délire des parenthèses.

Celle-ci est plus recommander quand on utilise This sinon c’est l’autre qui est la plus utilisé.

La nomination des évènement JQuery et JS sont différente

On va voir la différence ensemble, exemple :

.click() sur jQuery onclick sur JS

https://www.w3schools.com/jquery/jquery\_events.asp

<https://www.w3schools.com/js/js_events.asp>

* Pour utiliser plusieurs évènements dans la même fonction (jQuary):

$(document).ready(function(){

$("p").on({

mouseenter: function(){

$(this).css("background-color", "lightgray");

},

mouseleave: function(){

$(this).css("background-color", "lightblue");

},

click: function(){

$(this).css("background-color", "yellow");

}

});

});

* Pour Javascript les plus utiliser, sont :

onchange Un élément HTML a été modifié

onclick L'utilisateur clique sur un élément HTML

onmouseover L'utilisateur déplace la souris sur un élément HTML

onmouseout L'utilisateur éloigne la souris d'un élément HTML

onkeydown L'utilisateur appuie sur une touche du clavier

onload Le navigateur a fini de charger la page

Effet JQuery

Le paramètre callback : pour que plusieurs effets soient appliqués. (pas besoin de le mentionné sur JQuery mais sur JS on les mets)

https://www.w3schools.com/jquery/jquery\_callback.asp

Fade JQuery

Pour tous les effets fondus sur les évènements

<https://www.w3schools.com/jquery/jquery_fade.asp>

Effect Slide JQuery

slideDown() Slide vers le bas

<https://www.w3schools.com/jquery/jquery_slide.asp>

Animations JQuery

<https://www.w3schools.com/jquery/jquery_animate.asp>

animsta.net = très important

JQuery Stop

La méthode stop() jQuery est utilisée pour arrêter une animation ou un effet avant qu'il ne soit terminé.

Chaine JQuery

Plusieurs animations à la fois.

<https://www.w3schools.com/jquery/jquery_chaining.asp>

Très intéressent aussi

Manipulation du DOM dans JQuery

Trois méthodes jQuery simples mais utiles pour la manipulation DOM sont :

GET (on ne met pas de paramètre, sa récupère seulement les valeurs)

Exemple :

$(document).ready(function(){

$("#btn1").click(function(){

alert("Text: " + $("#test").text());

});

$("#btn2").click(function(){

alert("HTML: " + $("#test").html());

});

});

.text = va nous afficher tout le text pur qui est dans ma balise ciblée.

.html = va nous afficher tout le contenu y compris le ‘’bold’’.

<p id="test">This is some <b>bold</b> text in a paragraph.</p>

text() - Définit ou renvoie le contenu du texte des éléments sélectionnés

(affiche seulement le contenu textuel, sans les balises)

html() - Définit ou renvoie le contenu des éléments sélectionnés (y compris le balisage HTML)

val() - Définit ou renvoie la valeur des champs de formulaire

La méthode attr() jQuery est utilisée pour obtenir des valeurs d'attribut.

SET(on attribue une nouvelle valeur)

Exemple :

$(document).ready(function(){

$("#btn1").click(function(){

$("#test1").text("Hello world!");

$(‘’#test1’’).text($(‘’#test4’’)).text() ;

});

$("#btn2").click(function(){

$("#test2").html("<b>Hello world!</b>");

});

$("#btn3").click(function(){

$("#test3").val("Dolly Duck");

});

});

</script>

</head>

<body>

<p id="test1">This is a paragraph.</p>

<p id="test2">This is another paragraph.</p>

<p id="test4">NOUVEAUTE.</p>

<p>Input field: <input type="text" id="test3" value="Mickey Mouse"></p>

<button id="btn1">Set Text</button>

<button id="btn2">Set HTML</button>

<button id="btn3">Set Value</button>

Le texte en marron c’est la possibilité de changer la formule qu’on applique à la valeur pour obtenir un changement.

En tapant la formule, on obtient la valeur du test4.

Pour inclure un sélecteur dans jQuery, on met notre signe $(‘’….’’ )

<https://www.w3schools.com/jquery/jquery_selectors.asp>

Il y a un sélecteur universel (\*) qui cible tous les éléments de la page

Piqure de rappel sur les pseudo-classes :

https://www.w3schools.com/css/css\_pseudo\_classes.asp

Ajouter contenu JQuery

Nous examinerons quatre méthodes jQuery utilisées pour ajouter du nouveau contenu :

append() - Insère du contenu à la fin des éléments sélectionnés

Exemple :

$(document).ready(function(){

$("#btn1").click(function(){

$("p").append(" <b>Appended text</b>.");

});

Cela ajoutera le texte ‘’appended text’’ à la fin de toutes les balises P.

$(document).ready(function(){

$("#btn1").click(function(){

$("p").prepend("<b>Prepended text</b>. ");

Cela ajoutera le texte ‘’prepended text’’ au début de toutes les balises P.

prepend() - Insère du contenu au début des éléments sélectionnés

after() - Insère du contenu après les éléments sélectionnés

before() - Insère du contenu avant les éléments sélectionnés

function appendText() {

var txt1 = "<p>Text.</p>"; // Création du texte avec HTML

var txt2 = $("<p></p>").text("Text."); // Création du texte avec jQuery

var txt3 = document.createElement("p");

txt3.innerHTML = "Text."; // Création du texte avec DOM

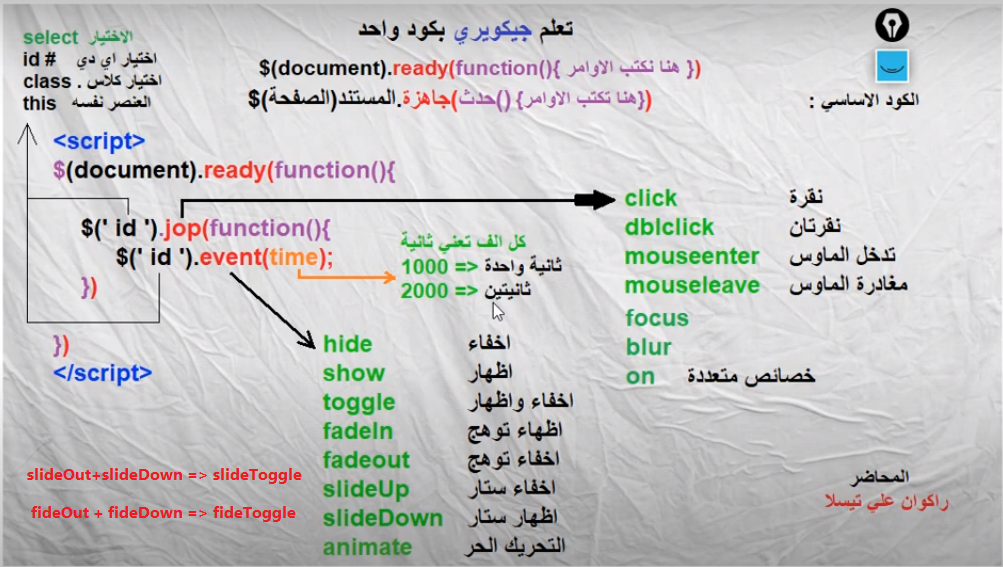
$("body").append(txt1, txt2, txt3); // Ajouter de nouveaux éléments

}

Var txt1 = entre guillemet, on met le contenu du texte avec la balise. (valeur dur)

Var txt2 = $(‘’…’’) = syntaxe pour sélecteur JQuery et après .text= on rajoute le texte ‘’…’’.

Var txt3 = On créer une fonction avec comme paramètre ‘’p’’, en lui mettant document.createElement, elle va créer la balise ‘’p’’ . Et avec le texte=’’…’’



https://www.youtube.com/watch?v=6xsDGM6Zhbs

Sélecteurs jQuery

|  |  |  |
| --- | --- | --- |
| **Syntax** | **Description** | **Example** |
| $("\*") | Selects all elements | [Try it](https://www.w3schools.com/jquery/tryit.asp?filename=tryjquery_sel_all2) |
| $(this) | Selects the current HTML element | [Try it](https://www.w3schools.com/jquery/tryit.asp?filename=tryjquery_sel_this) |
| $("p.intro") | Selects all <p> elements with class="intro" | [Try it](https://www.w3schools.com/jquery/tryit.asp?filename=tryjquery_sel_pclass) |
| $("p:first") | Selects the first <p> element | [Try it](https://www.w3schools.com/jquery/tryit.asp?filename=tryjquery_sel_pfirst) |
| $("ul li:first") | Selects the first <li> element of the first <ul> | [Try it](https://www.w3schools.com/jquery/tryit.asp?filename=tryjquery_sel_ullifirst) |
| $("ul li:first-child") | Selects the first <li> element of every <ul> | [Try it](https://www.w3schools.com/jquery/tryit.asp?filename=tryjquery_sel_ullifirstchild) |
| $("[href]") | Selects all elements with an href attribute | [Try it](https://www.w3schools.com/jquery/tryit.asp?filename=tryjquery_sel_hrefattr) |
| $("a[target='\_blank']") | Selects all <a> elements with a target attribute value equal to "\_blank" | [Try it](https://www.w3schools.com/jquery/tryit.asp?filename=tryjquery_sel_hrefattrblank) |
| $("a[target!='\_blank']") | Selects all <a> elements with a target attribute value NOT equal to "\_blank" | [Try it](https://www.w3schools.com/jquery/tryit.asp?filename=tryjquery_sel_hrefattrnotblank) |
| $(":button") | Selects all <button> elements and <input> elements of type="button" | [Try it](https://www.w3schools.com/jquery/tryit.asp?filename=tryjquery_sel_button2) |
| $("tr:even") | Selects all even <tr> elements | [Try it](https://www.w3schools.com/jquery/tryit.asp?filename=tryjquery_sel_even) |
| $("tr:odd") | Selects all odd <tr> elements | [Try it](https://www.w3schools.com/jquery/tryit.asp?filename=tryjquery_sel_odd) |

Utilisez notre [testeur de sélecteur jQuery](https://www.w3schools.com/jquery/trysel.asp) <https://www.w3schools.com/jquery/trysel.asp>

 pour démontrer les différents sélecteurs.

# Méthodes d'événement jQuery

Déférence nomination entre javasecript et jqeury

Voici quelques événements DOM courants:

|  |  |  |  |
| --- | --- | --- | --- |
| **Mouse Events** | **Keyboard Events** | **Form Events** | **Document/Window Events** |
| click | keypress | submit | load |
| dblclick | keydown | change | resize |
| mouseenter | keyup | focus | scroll |
| mouseleave |  | blur | unload |

<https://www.w3schools.com/jquery/jquery_events.asp>

Méthodes d'événement jQuery couramment utilisées

$ (document) .ready ()

La $(document).ready()méthode nous permet d'exécuter une fonction lorsque le document est complètement chargé. Cet événement est déjà expliqué dans le chapitre [Syntaxe jQuery](https://www.w3schools.com/jquery/jquery_syntax.asp) .

**Cliquez sur()**

La click()méthode attache une fonction de gestionnaire d'événements à un élément HTML.

La fonction est exécutée lorsque l'utilisateur clique sur l'élément HTML.

L'exemple suivant indique: Lorsqu'un événement de clic se déclenche sur un <p>élément; masquer l' <p>élément courant :

Exemple

$("p").click(function(){  
  $(this).hide();  
});

**Syntaxe:**

$(*selector*).hide(*speed,callback*);  
  
$(*selector*).show(*speed,callback*);

Le paramètre optionnel de vitesse spécifie la vitesse de masquage / affichage et peut prendre les valeurs suivantes: "lent", "rapide" ou millisecondes.

Le paramètre de rappel facultatif est une fonction à exécuter une fois la méthode hide()ou show()terminée (vous en apprendrez plus sur les fonctions de rappel dans un chapitre ultérieur).

L'exemple suivant montre le paramètre de vitesse avec hide():

Exemple

$("button").click(function(){  
  $("p").hide(1000);  
});

# Effets jQuery - Glissement

<https://www.w3schools.com/jquery/jquery_slide.asp>

# Effets jQuery – Animation

## jQuery animate () - Manipuler plusieurs propriétés

Notez que plusieurs propriétés peuvent être animées en même temps:

### Exemple

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

# jQuery Callback Functions

## Fonctions de rappel jQuery

Les instructions JavaScript sont exécutées ligne par ligne. Cependant, avec les effets, la ligne de code suivante peut être exécutée même si l'effet n'est pas terminé. Cela peut créer des erreurs.

Pour éviter cela, vous pouvez créer une fonction de rappel.

Une fonction de rappel est exécutée une fois l'effet en cours terminé.

Syntaxe typique: $ ( *sélecteur* ) .hide ( *vitesse, rappel* );

Exemples

L'exemple ci-dessous a un paramètre de rappel qui est une fonction qui sera exécutée une fois l'effet de masquage terminé :

### Exemple avec rappel

$("button").click(function(){  
  $("p").hide("slow", function(){  
    alert("The paragraph is now hidden");  
  });  
});

# https://www.w3schools.com/jquery/jquery\_callback.asp

# jQuery - Chaînage

**Astuce: De** cette façon, les navigateurs n'ont pas à trouver le ou les mêmes éléments plus d'une fois.

Pour enchaîner une action, il vous suffit d'ajouter l'action à l'action précédente.

Les chaînes exemple suivant ainsi les css(), slideUp()et les slideDown() méthodes. L'élément "p1" passe d'abord au rouge, puis il glisse vers le haut, puis il glisse vers le bas:

### **Exemple**

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

# jQuery - Obtenir du contenu et des attributs

**DOM = Document Object Model**  
  
Le DOM définit une norme pour accéder aux documents HTML et XML:  
  
*"Le Document Object Model (DOM) du W3C est une plate-forme et une interface indépendante du langage qui permet aux programmes et scripts d'accéder dynamiquement et de mettre à jour le contenu, la structure et style d'un document. "*

## Obtenir du contenu - text (), html () et val ()

Trois méthodes jQuery simples mais utiles pour la manipulation DOM sont:

* text() - Définit ou renvoie le contenu du texte des éléments sélectionnés
* html() - Définit ou renvoie le contenu des éléments sélectionnés (y compris le balisage HTML)
* val() - Définit ou renvoie la valeur des champs de formulaire

L'exemple suivant montre comment obtenir du contenu avec jQuery text()et les html()méthodes:

### Exemple

$("#btn1").click(function(){  
  alert("Text: " + $("#test").text());  
});  
$("#btn2").click(function(){  
  alert("HTML: " + $("#test").html());  
});

<https://www.w3schools.com/jquery/jquery_dom_get.asp>

# jQuery - Obtenir du contenu et des attributs

<https://www.w3schools.com/jquery/jquery_dom_get.asp>

### **Exemple**

$("#btn1").click(function(){  
  alert("Text: " + $("#test").text());  
});  
$("#btn2").click(function(){  
  alert("HTML: " + $("#test").html());  
});

## Définir le contenu - text (), html () et val ()

Nous utiliserons les trois mêmes méthodes de la page précédente pour **définir le contenu** :

* text() - Définit ou renvoie le contenu du texte des éléments sélectionnés
* html() - Définit ou renvoie le contenu des éléments sélectionnés (y compris le balisage HTML)
* val() - Définit ou renvoie la valeur des champs de formulaire

L'exemple suivant montre comment définir le contenu avec le jQuery text(), html()et val()méthodes:

### Exemple

$("#btn1").click(function(){  
  $("#test1").text("Hello world!");  
});  
$("#btn2").click(function(){  
  $("#test2").html("<b>Hello world!</b>");  
});  
$("#btn3").click(function(){  
  $("#test3").val("Dolly Duck");  
});

https://www.w3schools.com/jquery/tryit.asp?filename=tryjquery\_dom\_html\_set

## Toutes les pseudo-classes

<https://www.w3schools.com/css/css_pseudo_classes.asp>

jQuery - Ajouter des éléments

## Méthodes jQuery after () et before ()

La after()méthode jQuery insère le contenu APRÈS les éléments HTML sélectionnés.

La before()méthode jQuery insère le contenu AVANT les éléments HTML sélectionnés.

Pour (befor ) et (after) les ellement ajouter se trouve a cote du selecteur (ils seront ces frere)

Exemple :

https://www.w3schools.com/jquery/tryit.asp?filename=tryjquery\_html\_after

## Ajouter un nouveau contenu HTML

Nous examinerons quatre méthodes jQuery utilisées pour ajouter du nouveau contenu:

* append() - Insère du contenu à la fin des éléments sélectionnés
* elle ajoute à l’intérieur du body
* prepend() - Insère du contenu au début des éléments sélectionnés …
* append et prepend s’ajoutent à l’intérieur du sélecteur (élément ajouter sera le fils du sélecteur)
* after() - Insère du contenu après les éléments sélectionnés
* before() - Insère du contenu avant les éléments sélectionnés
* Pour before et after, les éléments ajouter, se trouvent à côté du sélecteur (ils seront ses frères)

### Exemple

$("p").append("Some appended text.");

JSON = Notation d’objet de JavaScript

Le ‘’style’’ c’est exclusivement pour les slider, pour le reste c’est déconseiller

## jQuery remove (), méthode

La remove()méthode jQuery supprime le ou les éléments sélectionnés et leurs éléments enfants.

Exemple (elle supprime l’élément et ces fils)

<https://www.w3schools.com/jquery/tryit.asp?filename=tryjquery_dom_remove>

## jQuery empty (), méthode

La empty()méthode jQuery supprime les éléments enfants du ou des éléments sélectionnés.

Exemple (elle supprime le contenu de div) tu le trouve même pas

<https://www.w3schools.com/jquery/tryit.asp?filename=tryjquery_dom_empty>

jQuery Manipulation du CSS

jQuery a plusieurs méthodes de manipulation CSS. Nous examinerons les méthodes suivantes:

1. addClass() - Ajoute une ou plusieurs classes aux éléments sélectionnés
2. removeClass() - Supprime une ou plusieurs classes des éléments sélectionnés
3. toggleClass() - Bascule entre l'ajout / la suppression de classes des éléments sélectionnés
4. css() - Définit ou renvoie l'attribut de style

Exemple

1. <https://www.w3schools.com/jquery/tryit.asp?filename=tryjquery_dom_addclass2>
2. <https://www.w3schools.com/jquery/tryit.asp?filename=tryjquery_dom_removeclass>
3. <https://www.w3schools.com/jquery/tryit.asp?filename=tryjquery_dom_toggleclass>
4. <https://www.w3schools.com/jquery/tryit.asp?filename=tryjquery_css_getcolor> // iterable il contient plusieurs élément (c’est pour sa il a applique que pour le premier élément psk ya pas une alert

notation d’un objet JavaScript

**$("p").css({"background-color": "yellow", "font-size": "200%"});**

Méthodes de dimension jQuery

jQuery propose plusieurs méthodes importantes pour travailler avec les dimensions:

* width()
* height()
* innerWidth()
* innerHeight()
* outerWidth()
* outerHeight()

Dimensions jQuery 

## Méthodes jQuery width () ET height ()

La width()méthode définit ou renvoie la largeur d'un élément (exclut le remplissage, la bordure et la marge).

La height()méthode définit ou renvoie la hauteur d'un élément (exclut le remplissage, la bordure et la marge).

L'exemple suivant renvoie la largeur et la hauteur d'un <div> élément spécifié :

### **Exemple**

$("button").click(function(){  
  var txt = "";  
  txt += "Width: " + $("#div1").width() + "</br>";  
  txt += "Height: " + $("#div1").height();  
  $("#div1").html(txt);  
});

<https://www.w3schools.com/jquery/tryit.asp?filename=tryjquery_dim_width_height>

Méthodes jQuery innerWidth () et innerHeight ()

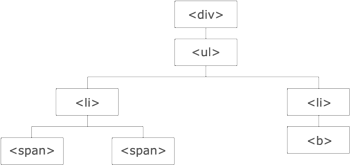
https://www.w3schools.com/jquery/jquery\_dimensions.asp

# jQuery Traversing

## Qu'est-ce que Traversing?

jQuery traversing, qui signifie "se déplacer", est utilisé pour "trouver" (ou sélectionner) des éléments HTML en fonction de leur relation avec d'autres éléments. Commencez par une sélection et parcourez cette sélection jusqu'à ce que vous atteigniez les éléments que vous désirez.

L'image ci-dessous illustre une page HTML sous forme d'arborescence (arbre DOM). Avec jQuery traversing, vous pouvez facilement vous déplacer vers le haut (ancêtres), vers le bas (descendants) et sur le côté (frères et sœurs) dans l'arborescence, à partir de l'élément sélectionné (actuel). Ce mouvement est appelé traverser - ou se déplacer à travers - l'arborescence DOM.



# jQuery Traversing - Ancêtres

Avec jQuery, vous pouvez parcourir l'arborescence DOM pour trouver les ancêtres d'un élément.

Un ancêtre est un parent, un grand-parent, un arrière-grand-parent, etc.

## Traversée de l'arborescence DOM

Trois méthodes jQuery utiles pour parcourir l'arborescence DOM sont:

* parent()
* parents()
* parentsUntil()

## jQuery parent (), méthode

La parent()méthode renvoie l'élément parent direct de l'élément sélectionné.

Cette méthode ne traverse qu'un seul niveau dans l'arborescence DOM.

L'exemple suivant renvoie l'élément parent direct de chaque <span> élément:

### **Exemple**

$(document).ready(function(){  
  $("span").parent();  
}); // $("span").parent().css({"color": "red", "border": "2px solid red"});

<https://www.w3schools.com/jquery/tryit.asp?filename=tryjquery_parent>

La parents()méthode retourne tous les éléments ancêtres de l'élément sélectionné, tout le chemin jusqu'à l'élément racine du document ( <html>).

L'exemple suivant renvoie tous les ancêtres de tous les <span>éléments:

### **Exemple**

$(document).ready(function(){  
  $("span").parents();  
}); ////parentS

<https://www.w3schools.com/jquery/tryit.asp?filename=tryjquery_parents>

The parentsUntil() method returns all ancestor elements between two given arguments.

The following example returns all ancestor elements between a <span> and a <div> element:

### **Example**

$(document).ready(function(){  
  $("span").parentsUntil("div");  
});

<https://www.w3schools.com/jquery/tryit.asp?filename=tryjquery_parentsuntil>

## jQuery children() Method

The children() method returns all direct children of the selected element.

This method only traverses a single level down the DOM tree.

The following example returns all elements that are direct children of each <div> elements:

### **Example**

$(document).ready(function(){  
  $("div").children();  
});

You can also use an optional parameter to filter the search for children.

The following example returns all <p> elements with the class name "first", that are direct children of <div>:

### **Example**

$(document).ready(function(){  
  $("div").children("p.first");  
});

<https://www.w3schools.com/jquery/tryit.asp?filename=tryjquery_children2>

## jQuery find (), méthode

La find()méthode retourne les éléments descendants de l'élément sélectionné, jusqu'au dernier descendant.

L'exemple suivant retourne tous les <span>éléments qui sont les descendants de <div>:

### **Exemple**

$(document).ready(function(){  
  $("div").find("span");  
});

L'exemple suivant renvoie tous les descendants de <div>:

### **Exemple**

$(document).ready(function(){  
  $("div").find("\*");  
});

<https://www.w3schools.com/jquery/tryit.asp?filename=tryjquery_find2>

## jQuery siblings (), méthode

La siblings()méthode renvoie tous les éléments frères de l'élément sélectionné.

L'exemple suivant renvoie tous les éléments frères de <h2>:

### **Exemple**

$(document).ready(function(){  
  $("h2").siblings();  
});

Vous pouvez également utiliser un paramètre facultatif pour filtrer la recherche des frères et sœurs.

L'exemple suivant renvoie tous les éléments frères de <h2>qui sont des <p> éléments:

### **Exemple**

$(document).ready(function(){  
  $("h2").siblings("p");  
});

<https://www.w3schools.com/jquery/tryit.asp?filename=tryjquery_siblings2>

## jQuery next (), méthode

La next()méthode retourne l'élément frère suivant de l'élément sélectionné.

L'exemple suivant renvoie le frère suivant de <h2>:

### **Exemple**

$(document).ready(function(){  
  $("h2").next();  
});

## jQuery nextAll (), méthode

La nextAll()méthode renvoie tous les éléments frères suivants de l'élément sélectionné.

L'exemple suivant renvoie tous les éléments frères suivants de <h2>:

### **Exemple**

$(document).ready(function(){  
  $("h2").nextAll();  
});

**https://www.w3schools.com/jquery/tryit.asp?filename=tryjquery\_nextall**

## jQuery nextUntil (), méthode

La nextUntil()méthode renvoie tous les éléments frères suivants entre deux arguments donnés.

L'exemple suivant renvoie tous les éléments frères entre a <h2>et un <h6>élément:

### **Exemple**

$(document).ready(function(){  
  $("h2").nextUntil("h6");  
});

https://www.w3schools.com/jquery/tryit.asp?filename=tryjquery\_nextuntil

## Méthodes jQuery prev (), prevAll () et prevUntil ()

Les méthodes prev(), prevAll()et prevUntil()fonctionnent exactement comme les méthodes ci-dessus mais avec une fonctionnalité inverse: elles renvoient les éléments frères précédents (parcourent les éléments frères et sœurs dans l'arborescence DOM, au lieu d'avancer).

## jQuery first (), méthode

La first()méthode renvoie le premier élément des éléments spécifiés.

L'exemple suivant sélectionne le premier <div> élément:

### **Exemple**

$(document).ready(function(){  
  $("div").first();  
});

## jQuery last (), méthode

La last()méthode renvoie le dernier élément des éléments spécifiés.

L'exemple suivant sélectionne le dernier <div> élément:

### **Exemple**

$(document).ready(function(){  
  $("div").last();  
});

## [**https://www.w3schools.com/jquery/tryit.asp?filename=tryjquery\_last**](https://www.w3schools.com/jquery/tryit.asp?filename=tryjquery_last)

## Méthode jQuery eq ()

La eq()méthode renvoie un élément avec un numéro d'index spécifique des éléments sélectionnés.

Les numéros d'index commencent à 0, donc le premier élément aura le numéro d'index 0 et non 1. L'exemple suivant sélectionne le deuxième <p>élément (numéro d'index 1):

**Exemple**

$(document).ready(function(){  
  $("p").eq(1);  
});

## jQuery filter (), méthode

La filter()méthode vous permet de spécifier un critère. Les éléments qui ne correspondent pas aux critères sont supprimés de la sélection et ceux qui correspondent sont renvoyés.

L'exemple suivant renvoie tous les <p>éléments avec le nom de classe "intro":

### **Exemple**

$(document).ready(function(){  
  $("p").filter(".intro");  
});

## Méthode jQuery eq ()

La eq()méthode renvoie un élément avec un numéro d'index spécifique des éléments sélectionnés.

Les numéros d'index commencent à 0, donc le premier élément aura le numéro d'index 0 et non 1. L'exemple suivant sélectionne le deuxième <p>élément (numéro d'index 1):

### **Exemple**

$(document).ready(function(){  
  $("p").eq(1); // **Premier élément 0**  
});

## jQuery filter (), méthode

La filter()méthode vous permet de spécifier un critère. Les éléments qui ne correspondent pas aux critères sont supprimés de la sélection et ceux qui correspondent sont renvoyés.

L'exemple suivant renvoie tous les <p>éléments avec le nom de classe "intro":

### **Exemple**

$(document).ready(function(){  
  $("p").filter(".intro");  
});

## jQuery not (), méthode

La not()méthode renvoie tous les éléments qui ne correspondent pas aux critères.

**Astuce:** la not()méthode est l'opposé de filter().

L'exemple suivant renvoie tous les <p>éléments qui n'ont pas de nom de classe "intro":

### **Exemple**

$(document).ready(function(){  
  $("p").not(".intro");  
});

<https://www.w3schools.com/jquery/tryit.asp?filename=tryjquery_not>

# Sélecteurs jQuery

<https://www.w3schools.com/jquery/jquery_ref_selectors.asp>

# Méthodes d' événement jQuery

<https://www.w3schools.com/jquery/jquery_ref_events.asp>

# Méthodes d’effet jQuery

<https://www.w3schools.com/jquery/jquery_ref_effects.asp>

# Méthodes jQuery HTML / CSS

<https://www.w3schools.com/jquery/jquery_ref_html.asp>

# Méthodes de parcours jQuery

<https://www.w3schools.com/jquery/jquery_ref_traversing.asp>