

# Présentation de la librairie jQuery

# Table des matières

<b>I. Contexte</b>	<b>3</b>
<b>II. Installer jQuery</b>	<b>3</b>
<b>III. Exercice : Appliquez la notion</b>	<b>6</b>
<b>IV. L'alias \$</b>	<b>6</b>
<b>V. Exercice : Appliquez la notion</b>	<b>9</b>
<b>VI. Syntaxe de base</b>	<b>9</b>
<b>VII. Exercice : Appliquez la notion</b>	<b>12</b>
<b>VIII. La documentation</b>	<b>13</b>
<b>IX. Exercice : Appliquez la notion</b>	<b>16</b>
<b>X. Auto-évaluation</b>	<b>17</b>
A. Exercice final .....	17
B. Exercice : Défi .....	19
<b>Solutions des exercices</b>	<b>22</b>

## I. Contexte

**Durée** : 45 min

**Environnement de travail** : Visual Studio Code

**Prérequis** : HTML, CSS, JavaScript

### Contexte

JavaScript offre pléthore de solutions pour créer de l'interactivité dans nos pages web. Certaines personnes ont donc décidé de nous faciliter la tâche, comme John Resig qui, début 2006, a lancé la librairie JavaScript libre **jQuery**.

Celle-ci permet par exemple de parcourir le DOM grâce à son support de sélecteurs CSS, elle met notamment à disposition des fonctionnalités permettant de créer des animations, et est utilisée afin de manipuler le CSS de manière plus instinctive ou encore de gérer les écouteurs d'événements.

Elle est encore aujourd'hui l'une des librairies les plus utilisées du point de vue front-end.

## II. Installer jQuery

### Objectifs

- Installer jQuery
- Inclure jQuery dans son projet

### Mise en situation

Nous allons voir les différentes manières "d'installer" et d'utiliser jQuery au sein de nos projets web. Nous trouverons les sources de la librairie sur le site officiel de jQuery<sup>1</sup>.

### Définition

#### Qu'est-ce qu'une librairie ?

Celles-ci étant plutôt utilisées pour implémenter des fonctionnalités entre deux systèmes distincts, la frontière entre les librairies et les API reste relativement mince. Elles sont à ne pas confondre avec les frameworks, qui sont, eux, utilisés pour donner une architecture de base à nos applications web.

### Télécharger les sources.

Il est possible de télécharger le fichier source de jQuery sur le site officiel<sup>2</sup> dans sa version classique **jquery.js** ou dans sa version minifiée **jquery.min.js**.

Un fichier minifié a le même contenu que le fichier classique, au détail près qu'un algorithme l'a parsé pour supprimer ce qui n'est pas essentiel à la bonne exécution du code, comme :

- les espaces,
- les sauts de ligne,
- les commentaires.

---

<sup>1</sup> <https://jquery.com/>

<sup>2</sup> <https://jquery.com/download/>

jquery-3.4.1	274 Ko
jquery-3.4.1.min	87 Ko

```
1 /*!  
2  * jQuery JavaScript Library v3.4.1  
3  * https://jquery.com/  
4  *  
5  * Includes Sizzle.js  
6  * https://sizzlejs.com/  
7  *  
8  * Copyright JS Foundation and other contributors  
9  * Released under the MIT license  
10 * https://jquery.org/license  
11 *  
12 * Date: 2019-05-01T21:04Z  
13 */  
14 ( function( global, factory ) {  
15  
16     "use strict";  
17  
18     if ( typeof module === "object" && typeof module.exports === "object" ) {}
```

[illegible]

Cependant, en vue de l'utiliser au sein des scripts suivants, il faudra veiller à l'inclure avant ceux-ci.

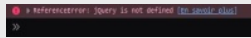
```
1 <body>
2   <script src="jquery-3.4.1.min"></script>
3   <script src="script.js"></script>
4 </body>
```



```

1 <body>
2   <script src="script.js"></script>
3   <script src="jquery-3.4.1.min"></script>
4 </body>

```



### Méthode Utiliser JQuery via un CDN (Content Delivery Network)

Un CDN, pour *Content Delivery Network* (ou Réseau de diffusion de contenu), est constitué d'ordinateurs reliés en réseau à travers Internet qui coopèrent afin de mettre à disposition du contenu ou des données à des utilisateurs.

Google, par exemple, met à disposition une interface permettant de diffuser du contenu. Concrètement, nous pouvons inclure une librairie référencée dans un CDN via une requête HTTP.

Cela nous permettra de ne pas avoir à télécharger de fichiers et de profiter des mises à jour de la librairie.

Du côté des points faibles, cela rajoute une requête HTTP à l'exécution de notre page web, ce qui peut être une source de lenteur.

Les différents CDN sur lesquels est hébergé jQuery sont répertoriés ici<sup>1</sup>.

### Exemple

```

1 <body>
2   <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js"></script>
3   <script src="script.js"></script>
4 </body>

```

Via un CDN, une librairie est incluse de la même manière qu'un script classique sauf que l'attribut `src` est une URL vers un autre site.

### Syntaxe À retenir

Il y a deux manières d'inclure jQuery dans une page :

- en téléchargeant le fichier, puis en l'incluant : `<script src="jquery3.4.1.min.js">`
- en l'incluant via un CDN :

```

<script src="https://code.jquery.com/jquery-3.6.0.min.js"
integrity="sha256-/xUj+3OJU5yExlq6GSYGHk7tPXikynS7ogEvDej/m4="
crossorigin="anonymous"></script>

```

### Complément

jQuery.com<sup>2</sup>

Minificateur de fichier (Google)<sup>3</sup>

Bien différencier librairie et framework<sup>4</sup>

1 <https://jquery.com/download/#using-jquery-with-a-cdn>

2 <https://jquery.com/>

3 <https://closure-compiler.appspot.com/home>

4 <https://waytolearnx.com/2019/03/difference-entre-framework-et-librairie.html>

### III. Exercice : Appliquez la notion

Vous disposez du fichier suivant :

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <meta charset="utf-8">
5   <meta name="viewport" content="width=device-width">
6   <title>repl.it</title>
7 </head>
8 <body>
9 <script src="script.js"></script>
10 </body>
11 </html>
12
```

#### Question

[solution n°1 p.23]

Incluez jQuery en vous basant sur la version 3.4.1 minifiée hébergée sur le CDN de Microsoft.

#### Indice :

Les différents CDN sont listés ici<sup>1</sup>.

#### Indice :

Lien direct du CDN Microsoft<sup>2</sup>.

### IV. L'alias \$

#### Objectifs

- Utiliser jQuery dans un script
- Vérifier qu'il est possible de l'utiliser

#### Mise en situation

Une fois jQuery inclus à nos pages, il ne nous reste qu'une étape avant de pouvoir utiliser ses outils. Il s'agit de faire appel à la fonction `jQuery()`.

Nous verrons que tout jQuery repose sur cette dernière. Il faut également vérifier qu'elle est bien fonctionnelle et prête à l'emploi.

#### Méthode Inclure la fonction principale de jQuery

Pour utiliser les objets, propriétés et méthodes proposés par jQuery, il est nécessaire d'inclure la fonction principale.

Le code devra obligatoirement être placé à l'intérieur de cette fonction.

1 <https://jquery.com/download/#other-cdns>

2 <https://docs.microsoft.com/en-us/aspnet/ajax/cdn/overview#jquery-version-341>

**Exemple****index.html**

```

1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8">
5     <meta name="viewport" content="width=device-width">
6     <title>repl.it</title>
7     <link href="style.css" rel="stylesheet" type="text/css" />
8   </head>
9   <body>
10    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js"></script>
11    <script src="script.js"></script>
12  </body>
13 </html>

```

**script.js**

```

1 jQuery(document).ready(() => {
2   // Notre code utilisant jQuery
3 })

```

Cette fonction prend comme paramètre **document** : il s'agit simplement de la page HTML courante et plus précisément du DOM.

Par la suite elle écoute l'événement **ready**, ce qui signifie que le code qui suivra ne sera exécuté qu'une fois notre page entièrement chargée.

Enfin, le script est placé dans une fonction anonyme.

**Méthode** **Simplifier les appels**

Il existe une syntaxe relativement plus courte pour inclure jQuery : `$ ( )`.

Le sigle "dollar" est **un alias** de la fonction jQuery. Ainsi, nous pourrions écrire le code ci-dessous :

```

1 $(document).ready(() => {
2   // Notre code utilisant jQuery
3 });

```

**Complément** **D'autres façons d'appeler jQuery**

Il est également possible d'utiliser des structures un peu plus simplifiées pour faire appel à jQuery.

**Exemple** **Sans le ready**

```

1 $(function() {
2   // Notre code utilisant jQuery
3 });

```

**Attention** **Éviter les collisions avec une autre librairie**

```

1 jQuery(function($){
2   alert('jQuery est inclus !')
3 });
4
5 // Ou encore
6
7 $(function($){

```

```
8   alert('jQuery est inclus !')
9 });
```

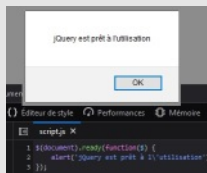
L'alias \$ peut être utilisé par d'autres librairies JS.

Afin d'éviter d'éventuels conflits entre deux librairies, nous pouvons ajouter \$ en paramètre de la fonction anonyme passée à jQuery.

### Méthode Vérifier la bonne installation de jQuery

Plaçons simplement un `alert()` ou un `console.log()` à l'intérieur de la fonction anonyme. Si une pop-up apparaît après le chargement de la page ou qu'un message s'affiche dans notre console, c'est gagné !

```
1 $(document).ready(() => {
2   alert('jQuery est prêt à l\'utilisation')
3 })
```



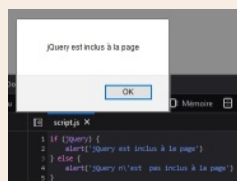
En cas d'erreur d'inclusion de jQuery, la pop-up ou le message console ne s'affichera pas et nous aurons généralement un message d'erreur nous indiquant la raison.



### Complément Test alternatif

Pour tester jQuery, nous pouvons également tester si l'objet global `jQuery` est inclus dans nos scripts.

```
1 if (jQuery) {
2   alert('jQuery est inclus à la page')
3 } else {
4   alert('jQuery n'est pas inclus à la page')
5 }
```



### Syntaxe À retenir

- La manière la plus fréquente pour initialiser jQuery est la suivante : `$(document).ready(function($){})`
- Nous pouvons l'écrire plus simplement : `$(function(){})`
- Si nous intégrons d'autres librairies utilisant l'alias \$ : `$(function($){})`
- Il est possible de tester si jQuery est prêt à l'utilisation grâce à un simple `alert` ou `console.log` à l'intérieur de la fonction anonyme passée à jQuery



```

1 $(document).ready(() => {
2     console.log('jQuery est prêt à l\'utilisation')
3 })

```

## V. Exercice : Appliquez la notion

Vous disposez du fichier suivant :

```

1 <!DOCTYPE html>
2 <html>
3 <head>
4     <meta charset="utf-8">
5     <meta name="viewport" content="width=device-width">
6     <title>repl.it</title>
7 </head>
8 <body>
9 <script src="https://ajax.aspnetcdn.com/ajax/jquery/jquery-3.4.1.min.js"></script>
10 <script src="script.js"></script>
11 </body>
12 </html>
13

```

### Question

[solution n°2 p.23]

Initialisez jQuery au sein d'un script et vérifiez son bon fonctionnement en affichant une alerte "jQuery a bien été inclus".

## VI. Syntaxe de base

### Objectifs

- Sélectionner un élément
- Appliquer une méthode à un élément

### Mise en situation

jQuery possède sa propre syntaxe. Là où JavaScript peut être verbeux, jQuery est censé nous simplifier la tâche en proposant une syntaxe plus intuitive et du code facile à écrire.

Une fois un élément du DOM sélectionné, l'objectif est de pouvoir interagir avec lui, lui affecter un comportement ou modifier son aspect.

Pour cela, jQuery propose un ensemble de méthodes.

#### Syntaxe

#### Syntaxe générale

- \$ est l'élément central. Celui-ci sera systématiquement utilisé lorsque du code sera écrit avec jQuery.
- Puis, comme en JavaScript, il est nécessaire de sélectionner l'élément du DOM sur lequel agir.
- Enfin, il faut indiquer quelle méthode utiliser afin de réaliser le traitement souhaité.

```
1 $('monSélecteur').methode()
```

### Remarque Les sélecteurs

Par convention, comme en CSS, un sélecteur par identifiant commencera toujours par "#" et un sélecteur par classe, par "." suivi du nom du sélecteur.

Rappelons-nous que la base de l'interaction entre du code HTML et du code JavaScript est le DOM. jQuery possède son propre modèle de sélecteurs et a choisi d'utiliser les sélecteurs CSS.

De la même manière qu'en CSS, il est également possible de cibler n'importe quelle balise.

### Comparatif : JavaScript / jQuery

En JavaScript pur, pour sélectionner un élément du DOM, nous pouvons utiliser la méthode `querySelector`.

```
1 <div id="content">Présentation de jQuery</div>
1 document.querySelector('#content')
```

Avec jQuery, nous utiliserons les sélecteurs CSS de la manière suivante :

```
1 <div id="content">Présentation de jQuery</div>
1 $('#content')
```

Si nous l'affichons, nous pouvons constater un objet comportant notre div.

```
> get https://api.jquery.com/jquery/3.4.1/jquery.min.js
> Object (0: divcontent, length: 1)
>
```

### Exemple Différentes façons de cibler un objet

```
1 <div id="content" class="content-div">Présentation de jQuery</div>
1 $('div') // Sélection de la div par son nom de balise.
2 $('#content') // Sélection de la div par son attribut id #content
3 $('.content-div') // Sélection des div par la classe .content-div
```

```
> get https://api.jquery.com/jquery/3.4.1/jquery.min.js
> Object (0: divcontent.content-div, length: 1, prevObject: {})
> Object (0: divcontent.content-div, length: 1)
> Object (0: divcontent.content-div, length: 1, prevObject: {})
>
```

### L'appel des méthodes

Une méthode est appelée à la suite du sélecteur : `$('selecteur').methode()`.

- `html('<span>exemple</span>')` : permet d'inclure du code HTML dans l'élément sélectionné. Équivalent d'`innerHTML` en JavaScript.
- `text('exemple')` : permet d'inclure du texte dans l'élément sélectionné. Équivalent d'`innerText` en JavaScript.
- `css('key', 'value')` : permet d'ajouter ou modifier la valeur d'un attribut CSS.
- `click()` : permet d'écouter l'événement **click** de la souris.

Il en existe beaucoup d'autres, toutes ces méthodes sont listées dans la documentation de jQuery<sup>1</sup>.

<sup>1</sup> <https://api.jquery.com/>

**Exemple**

```
1 $('#content p').html('<b>Ajoutons un élément avec du HTML</b>')
2 $('#content p').css('color', 'red') // Donnons une couleur à notre paragraphe
```

Présentation de jQuery  
Ajoutons un élément avec du HTML.

**Attention**

Lorsque l'on donne un argument aux méthodes `html` et `text`, celles-ci permettent d'initialiser le contenu d'un sélecteur avec l'argument donné en paramètre.

En revanche, si ces méthodes sont appelées sans argument, elles permettent de récupérer le contenu du sélecteur ciblé.

**Méthode** **Méthode avec fonction anonyme**

Cette syntaxe sera surtout utile lors de la mise en place d'écouteurs d'événements avec jQuery.

Il est alors possible d'y associer un script exécuté lors du déclenchement de celui-ci.

**Exemple**

```
1 <div id="content" class="content-div">
2   Présentation de jQuery
3   <p>Bonjour tout le monde !</p>
4 </div>
```

```
1 $('#content p').click(function() {
2   alert('Clique sur le paragraphe')
3 })
```

Présentation de jQuery  
Bonjour tout le monde !

Clique sur le paragraphe  
☐ Empêcher cette page d'ouvrir des dialogues supplémentaires

OK

**Complément** **Enchaîner les méthodes**

Les méthodes peuvent être enchaînées. Ceci nous offre une facilité d'écriture de nos scripts et nous permet d'associer plusieurs comportements à un élément en une seule fois.

**Exemple**

```
1 <div id="content" class="content-div">
2   Présentation de jQuery
3   <p></p>
4 </div>
```

```
1 $('#content p').text('Bonjour tout le monde !').css('color', 'blue')
```

Présentation de jQuery  
Bonjour tout le monde !

Ici nous écrivons d'abord le texte "Bonjour tout le monde !" dans le paragraphe puis nous lui attribuons la couleur bleue.

## Syntaxe À retenir

- Un sélecteur s'écrit avec l'alias **\$** suivi du nom du sélecteur entre parenthèses : `$ ('#mon-sélecteur')`
- Un sélecteur par identifiant commence par `"#"` et un sélecteur par classe commence par `"."`
- Une méthode est invoquée après le sélecteur et est séparée de celui-ci par un point `"."` : `$ ('#sélecteur').html ('<b>texte en gras</b>')`
- Les méthodes peuvent être invoquées à la chaîne : `$ ('#sélecteur').html ('<b>texte en gras</b>').css ('font-size', '2em')`

## Complément

Documentation de jQuery<sup>1</sup>

Les sélecteurs jQuery<sup>2</sup>

## VII. Exercice : Appliquez la notion

Vous disposez du fichier suivant :

```

1 <!DOCTYPE html>
2 <html>
3 <head>
4     <meta charset="utf-8">
5     <meta name="viewport" content="width=device-width">
6     <title>repl.it</title>
7 </head>
8 <body>
9 <script src="https://ajax.aspnetcdn.com/ajax/jquery/jquery-3.4.1.min.js"></script>
10 <script src="script.js"></script>
11
12 <section>
13     <article>
14         <h1></h1>
15         <div class="introduction">
16
17         </div>
18         <h2>Syntaxe générale</h2>
19         <p class="content">
20             Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
21             incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud
22             exercitation ullamco
23             laboris nisi ut aliquip ex ea commodo consequat
24         </p>
25         <h2>Les sélecteurs</h2>
26         <p class="content">
27             Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
28             incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud
29             exercitation ullamco
30             laboris nisi ut aliquip ex ea commodo consequat
31         </p>
32         <h2>Les méthodes</h2>
33         <p class="content">
```

<sup>1</sup> <https://api.jquery.com/>

<sup>2</sup> <https://api.jquery.com/category/selectors/>

```
32         Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor  
    incididunt  
33         ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud  
    exercitation ullamco  
34         laboris nisi ut aliquip ex ea commodo consequat  
35     </p>  
36  
37     <h2>Conclusion</h2>  
38 </article>  
39 </section>  
40 </body>  
41 </html>  
42  
1 $(document).ready(function() {  
2  
3 });
```

## Question

[solution n°3 p.23]

Modifiez le script précédent afin d'appliquer au document les modifications suivantes :

- Insérez le texte suivant dans l'élément h1 : "*Introduction à jQuery*".
- Au sein de l'élément div disposant de la classe `introduction`, ajoutez le code HTML suivant : `<p>Il s'agit de <strong>l'introduction</strong> </p>`  
Ce HTML devra être interprété lorsqu'il sera ajouté.
- Modifiez le CSS des paragraphes disposant de la classe `content` pour y appliquer un texte de couleur bleue.

## VIII. La documentation

### Objectif

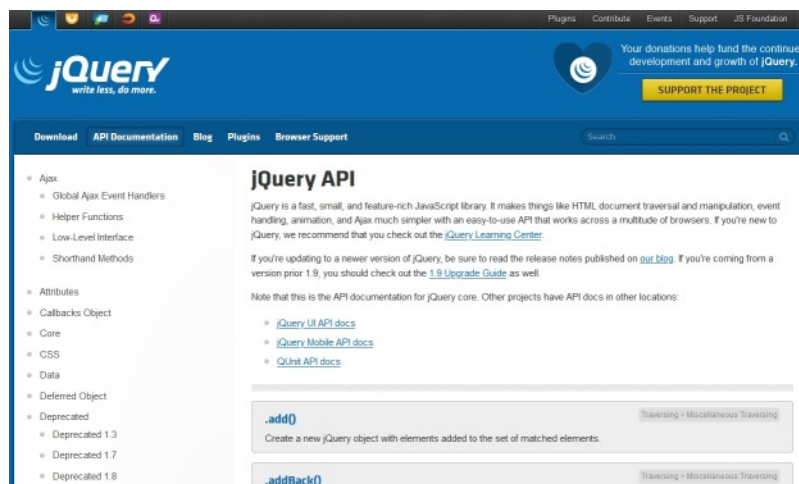
- Rechercher efficacement dans la documentation jQuery

### Mise en situation

jQuery est une librairie très complète. Pas question de parcourir son code pour connaître ses arcanes, nous utiliserons plutôt sa documentation bien fournie.

## Présentation générale

La documentation de la librairie<sup>1</sup> est accessible directement sur le site officiel de jQuery, dans le menu **API Documentation**.



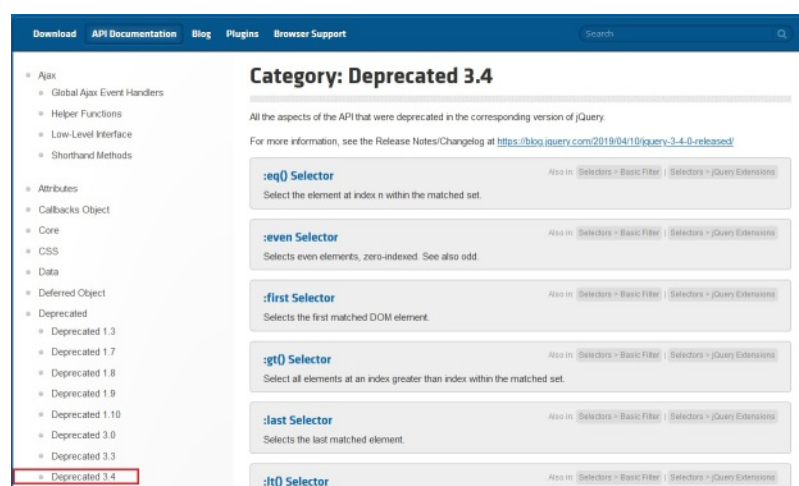
On y retrouve :

- un **panel central** listant par défaut l'ensemble des méthodes disponibles dans jQuery, comme `addClass()`, `html()`, `click()`.
- un **menu de gauche** permettant de filtrer les méthodes par catégories comme les sélecteurs, les événements, les formulaires.
- une **zone de recherche** en haut à droite permettant la recherche d'une méthode par son nom.

## Identifier les variantes de fonctionnalités

Le menu de gauche, section **deprecated**, permet également de lister les éléments et méthodes obsolètes selon la version de la librairie.

Cela peut s'avérer très utile pour savoir ce qu'il faudra mettre à jour dans un script durant une montée de version, par exemple.



Un peu plus bas, dans la section **version**, on retrouve les méthodes ajoutées aux différentes versions de jQuery.

Toujours dans le cas d'une montée de version, cela permet de connaître les nouveautés apportées.

<sup>1</sup> <https://api.jquery.com/>



## La recherche d'information

La barre de recherche située à droite dans le menu du haut permet de rechercher un élément.

Les résultats de recherche sont mis à jour au fil de la saisie.

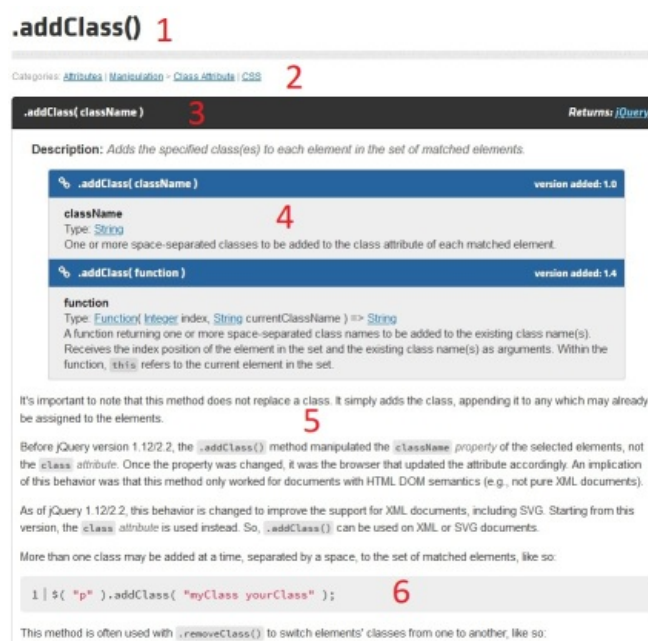


## Lire la documentation d'un élément

Prenons l'exemple de la méthode `addClass()`<sup>1</sup>.

La documentation d'une méthode se décompose ainsi :

1. Rappel du nom de la méthode
2. Catégories auxquelles elle appartient : permet de nous donner des indications du contexte dans lequel l'utiliser
3. Signature de la méthode, c'est-à-dire son nom, ses éventuels paramètres et ce qu'elle retourne
4. Une description précise des paramètres attendus
5. Des indications sur l'utilisation de la méthode selon les versions de jQuery
6. Des exemples de syntaxe
7. Des exemples d'application avec le résultat escompté



<sup>1</sup> <https://api.jquery.com/addClass/#addClass-className>

**Examples:**

Add the class "selected" to the matched elements.

```

1 <!doctype html>
2 <html lang="en">
3 <head>
4   <meta charset="utf-8">
5   <title>addClass demo</title>
6   <style>
7     p {
8       margin: 8px;
9       font-size: 16px;
10    }
11    .selected {
12      color: blue;
13    }
14    .highlight {
15      background: yellow;
16    }
17  </style>
18  <script src="https://code.jquery.com/jquery-3.4.1.js"></script>
19 </head>
20 <body>
21
22  <p>Hello</p>
23  <p>and</p>
24  <p>Goodbye</p>
25
26  <script>
27    $( "p" ).last().addClass( "selected" );
28  </script>
29
30 </body>
31 </html>

```

Demo:

Hello  
and  
Goodbye

**Syntaxe**    **À retenir**

- Dans le cadre d'une librairie aussi fournie que jQuery, la documentation est notre meilleure alliée.
- Les exemples d'application sont nombreux, ce qui est un plus dans l'apprentissage d'une technologie.

## IX. Exercice : Appliquez la notion

Vous venez de voir la documentation de la méthode `addClass`<sup>1</sup> et disposez du code suivant :

```

1 <!DOCTYPE html>
2 <html>
3 <head>
4   <meta charset="utf-8">
5   <meta name="viewport" content="width=device-width">
6   <title>repl.it</title>
7 </head>
8 <body>
9   <script src="https://ajax.aspnetcdn.com/ajax/jquery/jquery-3.4.1.min.js"></script>
10  <script src="script.js"></script>
11
12  <section>
13    <article>
14      <h1>Introduction à jQuery</h1>
15
16      <h2>Syntaxe générale</h2>
17      <p class="content">
18        Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
19        incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud
20        exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat
21      </p>
22      <h2>Les sélecteurs</h2>
23      <p class="content">
24        Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
25        incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud
26        exercitation ullamco

```

<sup>1</sup> <https://api.jquery.com/addClass/#addClass-className>



```

26         laboris nisi ut aliquip ex ea commodo consequat
27     </p>
28     <h2>Les méthodes</h2>
29     <p class="content">
30         Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
31     incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud
32     exercitation ullamco
33         laboris nisi ut aliquip ex ea commodo consequat
34     </p>
35     <h2>Conclusion</h2>
36 </article>
37 </section>
38 </body>
39 </html>
40
1 $(document).ready(function () {
2     $('p.content').css('color', 'blue');
3 });

```

### Question

[solution n°4 p.23]

En vous aidant de la documentation, identifiez comment supprimer la classe `content` du premier paragraphe.

#### Indice :

Essayez de regarder du côté de la méthode `removeClass`.

#### Indice :

Dans l'exemple de la méthode `addClass`, on remarque une méthode `last`, peut-être existe-t-il une méthode `first` ?

## X. Auto-évaluation

### A. Exercice final

#### Exercice 1

[solution n°5 p.24]

Exercice

Qu'appelle-t-on une bibliothèque ?

- ☐ Un ensemble d'objets et de méthodes mis à disposition des développeurs pour agrémenter leur code
- ☐ Un système permettant d'implémenter des fonctionnalités et une communication entre deux systèmes distincts
- ☐ Un système permettant de donner une architecture de base à une application

Exercice

jQuery est considéré comme...

- ☐ Un framework
- ☐ Une librairie
- ☐ Une API

Exercice

Quelles sont les différences notables d'un fichier minifié par rapport à un fichier classique ?

- ☐ Les espaces sont supprimés
- ☐ Les sauts de lignes sont supprimés
- ☐ La logique est supprimée
- ☐ Les variables et fonctions sont renommées
- ☐ Les commentaires sont supprimés
- ☐ Il est plus lourd

#### Exercice

Lorsque l'on fait appel à un CDN pour intégrer une librairie, les sources sont...

- ☐ Hébergées localement
- ☐ Hébergées à distance

#### Exercice

Que permet le code ci-dessous ?

```
1 if (jQuery) {  
2     // code à exécuter  
3 }
```

- ☐ Il permet d'initialiser jQuery
- ☐ Il permet de vérifier que l'objet global jQuery est présent

#### Exercice

Quel est l'équivalent de `document.querySelector('.content')` en jQuery ?

- ☐ `$('content')`
- ☐ `$(document).('content')`
- ☐ `$('.content')`
- ☐ `$('#content')`

#### Exercice

Qu'effectue le code suivant ?

```
1 $(document).ready(function () {  
2     $('.content').css('color', 'blue');  
3 });  
4
```

- ☐ Il cible tous les éléments de la classe `content` et leur applique une couleur bleue
- ☐ Il cible l'ensemble des éléments de la classe `content` s'ils sont affichés en bleu

#### Exercice

Parmi ces méthodes, la/lesquelles pourrait-on utiliser pour cibler le premier élément de la classe `content` ?

- ☐ `$('#p.content:first-child').css('color', 'blue');`
- ☐ `$('#p.content::first').css('color', 'blue');`
- ☐ `$('#p.content').first().css('color', 'blue');`
- ☐ `$('#p.content').css('color', 'blue').first();`

#### Exercice

Quel sera le contenu de `#introduction` si on y applique la méthode suivante ?

```
1 $('#introduction').text('<p>Introduction</p>');
```

#### Exercice

L'appel suivant permet...

```
1 $('#introduction').html();
```

- ☐ De récupérer le contenu HTML de `#introduction`
- ☐ De vider le contenu HTML de `#introduction`

### B. Exercice : Défi

Aujourd'hui, c'est le grand jour, vous avez décidé de vous lancer dans la mise en ligne de votre premier blog.

Pour la mise en page, n'étant pas très à l'aise pour le moment, vous avez choisi d'utiliser la librairie Bootstrap<sup>1</sup>, qui intègre de nombreuses fonctionnalités. La preuve, vous avez réussi à mettre en place un diaporama permettant de mettre en lumière vos articles les plus consultés, ainsi qu'un bouton de partage sur les réseaux sociaux.

Au premier coup d'œil, tout semble parfait, pourtant... il semblerait que quelque chose cloche. La mise en page est conforme à vos espérances, mais le diaporama ne fonctionne pas, et le bouton "dropdown" reste désespérément immobile.

```
1 <!DOCTYPE html>
2 <html lang="fr">
3 <head>
4   <meta charset="utf-8"/>
5 </head>
6 <body>
7 <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css"
8   rel="stylesheet" integrity="sha384-1BmE4kWBq78iYhFdvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3" crossorigin="anonymous">
9 <main class="container">
10   <header class="jumbotron">
11     <h1>Mon blog personnel</h1>
12     <p class="lead">
13       Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
14       incididunt ut labore et dolore magna aliqua.
15       Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip
16       ex ea commodo consequat
17     </p>
18   </header>
19   <section>
20     <h2>Articles vedettes</h2>
21     <div class="bd-example">
22       <div id="carouselExampleCaptions" class="carousel slide" data-ride="carousel">
23         <ol class="carousel-indicators">
```

<sup>1</sup> <https://getbootstrap.com/>

```

21         <li data-target="#carouselExampleCaptions" data-slide-to="0"
class="active"></li>
22     <li data-target="#carouselExampleCaptions" data-slide-to="1"></li>
23 </ol>
24 <div class="carousel-inner">
25     <div class="carousel-item active">
26         
27         <div class="carousel-caption d-none d-md-block">
28             <h5>First slide label</h5>
29             <p>Nulla vitae elit libero, a pharetra augue mollis interdum.</p>
30         </div>
31     </div>
32     <div class="carousel-item">
33         
34         <div class="carousel-caption d-none d-md-block">
35             <h5>Second slide label</h5>
36             <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
37         </div>
38     </div>
39 </div>
40 <a class="carousel-control-prev" href="#carouselExampleCaptions" role="button"
data-slide="prev">
41     <span class="carousel-control-prev-icon" aria-hidden="true"></span>
42     <span class="sr-only">Previous</span>
43 </a>
44 <a class="carousel-control-next" href="#carouselExampleCaptions" role="button"
data-slide="next">
45     <span class="carousel-control-next-icon" aria-hidden="true"></span>
46     <span class="sr-only">Next</span>
47 </a>
48 </div>
49 </div>
50 <hr/>
51 </section>
52 <section class="list-group">
53     <h2 class="p-3">Tous mes articles</h2>
54     <article class="list-group-item">
55         <h3 class="text-muted">Lorem ipsum</h3>
56         <p>
57             Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore
58             eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia
59             deserunt mollit anim id est laborum
60         </p>
61         <div class="btn-group">
62             <button type="button" class="btn btn-success dropdown-toggle" data-
toggle="dropdown" aria-haspopup="true" aria-expanded="false">
63                 Partager sur les réseaux!
64             </button>
65             <div class="dropdown-menu">
66                 <a class="dropdown-item" href="#">Partager sur Twitter</a>
67                 <a class="dropdown-item" href="#">Partager sur Facebook</a>
68                 <a class="dropdown-item" href="#">Partager par e-mail</a>
69             </div>
70         </div>
71     </article>
72     <article class="list-group-item">
73         <h3 class="text-muted">Lorem ipsum</h3>
74         <p>

```

```

74      Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore
    eu fugiat nulla pariatur.
75      Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia
    deserunt mollit anim id est laborum
76      </p>
77      <div class="btn-group">
78      <button type="button" class="btn btn-success dropdown-toggle" data-
toggle="dropdown" aria-haspopup="true" aria-expanded="false">
    Partager sur les réseaux!
79      </button>
80      <div class="dropdown-menu">
81      <a class="dropdown-item" href="#">Partager sur Twitter</a>
82      <a class="dropdown-item" href="#">Partager sur Facebook</a>
83      <a class="dropdown-item" href="#">Partager par e-mail</a>
84      </div>
85      </div>
86      </article>
87      <article class="list-group-item">
88      <h3 class="text-muted">Lorem ipsum</h3>
89      <p>
90      Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore
    eu fugiat nulla pariatur.
91      Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia
    deserunt mollit anim id est laborum
92      </p>
93      <div class="btn-group">
94      <button type="button" class="btn btn-success dropdown-toggle" data-
toggle="dropdown" aria-haspopup="true" aria-expanded="false">
    Partager sur les réseaux!
95      </button>
96      <div class="dropdown-menu">
97      <a class="dropdown-item" href="#">Partager sur Twitter</a>
98      <a class="dropdown-item" href="#">Partager sur Facebook</a>
99      <a class="dropdown-item" href="#">Partager par e-mail</a>
100      </div>
101      </div>
102      </article>
103      </section>
104      </main>
105      <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js"
integrity="sha384-ka7Sk0Gln4gmtz2MlQnikT1wXgYsOg+OMhuP+ILRH9sENB00LRn5q+8nbTov4+1p"
crossorigin="anonymous"></script>
106      </body>
107      </html>
108
109
110

```

### Question 1

[solution n°6 p.26]

Dans votre enthousiasme, il semblerait que vous n'ayez pas lu la documentation, et que la librairie jQuery n'ait pas été incluse sur votre page avant d'intégrer la librairie Bootstrap. Elle est pourtant nécessaire au bon fonctionnement du projet, rectifiez donc cela en appelant la librairie depuis un CDN.

Profitez-en également pour vérifier la bonne définition de jQuery en affichant un message dans la console.

### Question 2

[solution n°7 p.26]

Et si ce test n'était pas suffisant ? Par acquis de conscience, vérifiez tout de même que la librairie est fonctionnelle.

- Appliquez au titre de premier niveau la couleur verte
- Le premier titre de second niveau sera quant à lui affiché en italique
- Modifiez le titre "Articles vedettes" pour afficher "Mes meilleurs articles" en enchaînant cet appel avec le précédent

**Solutions des exercices**

### p. 6 Solution n°1

```

1 <!DOCTYPE html>
2 <html>
3 <head>
4   <meta charset="utf-8">
5   <meta name="viewport" content="width=device-width">
6   <title>repl.it</title>
7 </head>
8 <body>
9 <script src="https://ajax.aspnetcdn.com/ajax/jquery/jquery-3.4.1.min.js"></script>
10 <script src="script.js"></script>
11 </body>
12 </html>
13

```

### p. 9 Solution n°2

```

1 $(document).ready(function() {
2   alert('jQuery a bien été inclus')
3 });

```

### p. 13 Solution n°3

```

1 $(document).ready(function () {
2   $('h1').text('Introduction à jQuery');
3   $('div.introduction').html('<p>Il s\'agit de <strong>l\'introduction</strong> </p>');
4   $('p.content').css('color', 'blue');
5 });

```

### p. 17 Solution n°4

La méthode `first`<sup>1</sup> permet de sélectionner le premier élément, la méthode `removeClass`<sup>2</sup> permet de supprimer une classe.

```

1 $(document).ready(function () {
2   $('p.content').first().removeClass('content');
3   $('p.content').css('color', 'blue');
4 });

```

Avez-vous remarqué que, si vous exécutez d'abord la modification de CSS avant de supprimer la classe, celui-ci reste pourtant appliqué ? C'est normal !

Inspectez votre DOM. Vous remarquerez que la méthode `css` ajoute un attribut `style inline` à chaque paragraphe disposant de la classe `content`.

La seconde instruction demande à ce que l'on supprime la classe. Le CSS n'y étant pas associé directement, il n'a pas disparu.

<sup>1</sup> <https://api.jquery.com/first/#first>

<sup>2</sup> <https://api.jquery.com/removeClass/>

```

@{System.out.println("2/2");
}

<div class="text" style="color: blue;">
    <p>
        Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed euismod tempor incididunt
        ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco
        laboris nisi ut aliquip ex ea commodo consequat
    </p>
</div>

<div class="lecteurs">
    <div class="content" style="color: blue;">
        Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed euismod tempor incididunt
        ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco
        laboris nisi ut aliquip ex ea commodo consequat
    </div>
</div>

```

### Exercise p. 17 Solution n°5

## Exercise

## Qu'appelle-t-on une bibliothèque ?

- ☒ Un ensemble d'objets et de méthodes mis à disposition des développeurs pour agréger leur code
- ☐ Un système permettant d'implémenter des fonctionnalités et une communication entre deux systèmes distincts
- ☐ Un système permettant de donner une architecture de base à une application

## Exercise

## jQuery est considéré comme...

- ☐ Un framework
- ☒ Une librairie
- ☐ Une API

## Exercise

Quelles sont les différences notables d'un fichier minifié par rapport à un fichier classique ?

- ☒ Les espaces sont supprimés
- ☒ Les sauts de lignes sont supprimés
- ☐ La logique est supprimée
- ☒ Les variables et fonctions sont renommées
- ☒ Les commentaires sont supprimés
- ☐ Il est plus lourd

## Exercise

Lorsque l'on fait appel à un CDN pour intégrer une librairie, les sources sont...

- ☐ Hébergées localement
- ☒ Hébergées à distance

## Exercise

Que permet le code ci-dessous ?

```
1 if (jQuery) {
2     // code à exécuter
3 }
```



- ☐ Il permet d'initialiser jQuery
- ☒ Il permet de vérifier que l'objet global `jQuery` est présent

### Exercice

Quel est l'équivalent de `document.querySelector('.content')` en jQuery ?

- ☐ `$('content')`
- ☐ `$(document).('.content')`
- ☒ `$('.content')`
- ☐ `$('#content')`

### Exercice

Qu'effectue le code suivant ?

```
1 $(document).ready(function () {  
2   $('.content').css('color', 'blue');  
3 });  
4
```

- ☒ Il cible tous les éléments de la classe `content` et leur applique une couleur bleue
- ☐ Il cible l'ensemble des éléments de la classe `content` s'ils sont affichés en bleu

### Exercice

Parmi ces méthodes, la/lesquelles pourrait-on utiliser pour cibler le premier élément de la classe `content` ?

- ☒ `$('p.content:first-child').css('color', 'blue');`  
*On peut utiliser le pseudo-element `first`, comme en CSS.*
- ☐ `$('p.content::first').css('color', 'blue');`  
*`::first` ne permet pas de cibler le premier élément d'un attribut, même en CSS.*
- ☒ `$('p.content').first().css('color', 'blue');`  
*La méthode `first` permet de filtrer les éléments.*
- ☐ `$('p.content').css('color', 'blue').first();`  
*On doit d'abord filtrer avant d'appliquer.*

### Exercice

Quel sera le contenu de `#introduction` si on y applique la méthode suivante ?

```
1 $('#introduction').text('<p>Introduction</p>');
```

`<p>Introduction</p>`

### Exercice

L'appel suivant permet...

```
1 $('#introduction').html();
```

- ☒ De récupérer le contenu HTML de `#introduction`
- ☐ De vider le contenu HTML de `#introduction`

Q Si l'on souhaite vider le contenu de la balise, il faut faire `$('#introduction').html('') ;`.

#### p. 21 Solution n°6

Voici ce que vous avez dû ajouter avant l'appel au CDN de Bootstrap :

```
1 <script src="https://ajax.aspnetcdn.com/ajax/jquery/jquery-3.4.1.min.js"></script>
2 <script src="script.js"></script>
```

Et le contenu de votre script JS :

```
1 if (jQuery) {
2   console.log('jQuery est correctement défini')
3 } else {
4   console.log('Il y a un souci dans la définition de jQuery')
5 }
```

#### p. 21 Solution n°7

Tout fonctionne, le premier test était donc correct !

Voici ce que vous avez dû produire :

```
1 if (jQuery) {
2   console.log('jQuery est correctement défini')
3 }
4 $(document).ready(() => {
5   $('h1').css('color', 'green')
6   $('h2').first().css('font-style', 'italic').text('Mes meilleurs articles')
7 })
8 } else {
9   console.log('Il y a un souci dans la définition de jQuery')
10 }
11
12
13
```