

# Semaine 5 : Compléments d'information du cours

*Cours\_ UVCI  
PRW2103-1*

*DIABATE Nabègna,*  
Université Virtuelle de Côte d'Ivoire

# Table des matières



<b>I - Section 1 : Les URL en JavaScript : redirections, chemin, ancre</b>	<b>3</b>
1. 1. Les redirections en JavaScript .....	3
2. 2. Récupérer l'URL en JavaScript .....	3
3. 3. Récupérer les éléments d'une URL .....	3
<b>II - Section 2 : Ressources additionnelles</b>	<b>5</b>
1. Les Scripts JavaScript en ligne .....	5
<b>III - Section 3 : Présentation de JQuery</b>	<b>6</b>
1. 1. Qu'est ce que JQuery ? .....	6
2. 2. installation de jQuery .....	7
3. 3. Premiers pas avec JQuery .....	8
<b>IV - Conclusion</b>	<b>9</b>

# Section 1 : Les URL en JavaScript : redirections, chemin, ancre



## 1. 1. Les redirections en JavaScript

Il y a deux possibilités pour faire une redirection en JavaScript. La première est équivalente à un clic tandis que la seconde est similaire à une redirection HTTP. Il est possible d'exprimer une URL absolue comme `https://monsite.fr/chemin/page.html` ou `monsite.fr/chemin/page.html`, ou des URL relative comme `chemin/pahe.html`.

```
1 // Pareille qu'une redirection HTTP
2 window.location.replace("https://monsite.fr/");
3 // Pareil à un clic sur un lien hypertexte
4 window.location.href = "https://monsite.fr/";
```

## 2. 2. Récupérer l'URL en JavaScript

Pour pouvoir récupérer et traiter les informations relatives à l'URL, nous allons utiliser l'objet `window.location` proposé par défaut dans JavaScript.

Vous pouvez obtenir l'URL de la page sur laquelle le JavaScript est exécuté comme ci-dessous. La variable contiendra alors l'URL sous la forme `https://monsite.fr/`.

```
1 var urlActuelle = window.location.protocol + "://" + window.location.host + "/" +
  window.location.pathname;
```

## 3. 3. Récupérer les éléments d'une URL

- Pour récupérer le protocole associé à une URL en JavaScript (HTTP, HTTPS, FILE...), le code suivant est exécuté sur une page. Si la page à une adresse par exemple `https://monsite.fr`, alors la variable `MonProtocole` aura la valeur « `https` ».

```
1 var MonProtocole = window.location.protocol;
```

- Si vous voulez récupérer le nom de domaine en JavaScript et si le code suivant est exécuté sur une page ayant l'URL `https://monsite.fr/chemin/page.html`, alors la variable `LeDomaine` aura la valeur « `monsite.fr` ».

```
1 var LeDomaine = window.location.host;
```

- Si en revanche vous voulez récupérer le chemin de l'URL en JavaScript, si vous exécutez le code suivant sur une page ayant l'URL `https://monsite.fr/chemin/page.html`, alors la variable `Mypath` aura la valeur « `chemin/page.html` ».

```
1 var Mypath = window.location.pathname;
```

Vous pouvez aussi récupérer chaque éléments du chemin dans un tableau avec le code suivant.

```
1 var pathArray = window.location.pathname.split( "/" );
```

- Récupérer l'ancre passée dans l'URL en JavaScript est devenu simple. Si vous exécutez le code suivant sur une page ayant l'URL *https://monsite.fr/chemin/page.html#mon-ancre*, alors la variable *path* aura la valeur « *#mon-ancre* ». Vous pouvez retirer simplement le dièse (« *#* ») en utilisant *window.location.hash.substr(1)*.

```
1 var path = window.location.hash;
```

# Section 2 : Ressources additionnelles



## 1. Les Scripts JavaScript en ligne



### *Complément*

---

Ne dit-on généralement pas qu'il faut éviter de réinventer la roue ?

Pour ce qui est de JavaScript, après avoir suivi ce cours, vous avez assez de compétences pour réaliser des choses fabuleuses et offrir des effets agréables dans vos pages web à vos visiteurs, donnant parfois une sensation de site dynamique et suffisamment animé!

Ainsi, certaines bibliothèques de scripts déjà rédigés sont distribué gratuitement. Il vous suffit de les télécharger et de les personnaliser puis les intégrer à votre site web (c'est cela une des principale force du open-source !). Cependant, vous devez bien comprendre le code avant de l'utiliser sur votre page car il peut non seulement comporter des erreurs, mais certains malins pourraient cacher des codes malicieux dans leur script.

Vous pouvez faire des recherches sur Google pour trouver des Scripts gratuits, mais pour ma part, je vous donne quelques liens où vous pourrez en trouver :

- <http://www.ouils-web.com/page-script-javascript.asp>
- <http://scripts.toucharger.com/scripts/javascript/>
- [http://www.supportduweb.com/scripts\\_tutoriaux-4-javascript.html](http://www.supportduweb.com/scripts_tutoriaux-4-javascript.html)
- <https://www.javascriptfreecode.com/>

# Section 3 : Présentation de JQuery



## 1. 1. Qu'est ce que JQuery ?

*jQuery* est une bibliothèque JavaScript libre et multi-plateforme créée pour faciliter l'écriture de scripts côté client dans le code HTML.

La première version est lancée en janvier 2006 par *John Resig* (un petit génie du JavaScript).

C'est la bibliothèque JavaScript la plus utilisées actuellement dans le développement web. Elle est utilisée dans le cadre de simple site web, RIA (Rich Internet Applications), jeux en HTML5, etc. Sa flexibilité permet en effet à *jQuery* de s'intégrer avec d'autres bibliothèques.

Bien que les fonctionnalités possibles grâce à *jQuery* soient potentiellement infinies, nous pouvons les regrouper en 6 fonctions principale :

1. *Accéder aux éléments du DOM* : jQuery met à disposition une notation simple pour accéder aux éléments du document.
2. *Modification de l'apparence d'une page* : son intégration avec les CSS permet à jQuery d'apporter toute sorte de modification de style.
3. *Modifier le contenu d'une page* : jQuery permet de modifier le contenu d'un document, par exemple en remplaçant le texte d'un paragraphe ou la labellisation d'un bouton.
4. *Rendre la page interactive* : jQuery permet d'intercepter plusieurs types d'événements sur la page (mouvement de la souris, *click*, etc.) et par conséquent de créer des changements en fonction de ces événements.
5. *Créer des animations* : jQuery permet facilement d'animer des éléments du DOM grâce à une combinaison de CSS et changement de contenu.
6. *Fonctionnalités asynchrones (AJAX)* : jQuery permet de charger le contenu d'une page de manière asynchrone, c'est-à-dire sans la mise à jour de la page dans le navigateur.

Pour vous faire un rapide comparatif d'écriture, voici des exemples comparant jQuery à JavaScript :

- *Événements*

```
1 // jQuery
2 $(document).ready(function() {
3     // vos scripts
4 })
5
6 // Javascript
7 document.addEventListener('DOMContentLoaded', function() {
8     // vos scripts
9 })
```

```
1 // jQuery
2 $('a').click(function() {
3     // vos scripts
4 })
```

```

5
6 // Javascript
7 [].forEach.call(document.querySelectorAll('a'), function(e1) {
8     e1.addEventListener('click', function() {
9         // vos scripts
10     })
11 })

```

- *Selecteurs*

```

1 // jQuery
2 var divs = $('div')
3
4 // Javascript
5 var divs = document.querySelectorAll('div')

```

## 2. 2. installation de jQuery

jQuery est une bibliothèque JavaScript qui n'est pas installée de base dans votre navigateur. Il faut l'ajouter par le biais d'un fichier ayant pour extension `.js`.

Pour l'utiliser dans une page HTML, il faut lier le fichier jQuery par le biais de la balise `script`.

```

1 <head>
2     <script src="jquery-2.1.1.min.js"></script>
3 </head>
4

```

### - *Installation locale*

L'installation en local est idéale si vous n'avez pas toujours accès à internet ou si votre connexion n'est pas très rapide.

Nous verrons par la suite que ce n'est pas la solution optimale pour une utilisation en production.

Vous avez besoin de télécharger le fichier sur le site officiel de jQuery : [Télécharger jQuery](#)

Vous pouvez vous rendre compte qu'il existe plusieurs versions de jQuery. Mi 2013 jQuery est passé en version 2.XX laissant de côté la compatibilité avec les versions antérieures d'Internet Explorer 8, allégeant aussi son poids.

Ne prenez pas peur, vous n'aurez pas à modifier ces fichiers, vous devez juste y faire référence dans votre HTML pour l'installation.

Voici par exemple un code minimaliste d'une page HTML avec script jQuery dans le `<header>` :

```

1 <!DOCTYPE html>
2 <html lang="fr">
3 <head>
4     <meta charset="UTF-8">
5     <title>Installation de jQuery</title>
6     <link rel="stylesheet" href="css/styles.css">
7     <script src="js/jquery-2.1.1.min.js"></script>
8 </head>
9 <body>
10     <!-- Contenu de la page -->
11 </body>
12 </html>
13

```

### - Installation de jQuery à distance

La manière la plus courante et aussi la plus utilisée est de passer par un CDN.

Le fichier est mis à disposition sur de multiples serveurs à travers le monde. Ces serveurs sont souvent plus rapides et économisent aussi votre bande passante. L'autre avantage est que de très nombreux sites utilisent cette méthode, donc le fichier jQuery est sûrement déjà en cache chez votre utilisateur.

Il existe plusieurs CDN pouvant vous fournir le fichier :

- *CDN jQuery* : <http://code.jquery.com/jquery-2.1.1.js>
- *CDN Google* : <http://ajax.googleapis.com/ajax/libs/jquery/2.1.1/jquery.min.js>

Pour faire l'installation, il suffit donc de remplacer le chemin du fichier en local par l'URL du CDN.

```
1 <script src="http://ajax.googleapis.com/ajax/libs/jquery/2.1.1/jquery.min.js"
  ></script>
2
```

## 3. 3. Premiers pas avec JQuery

Nous avons vu à l'étape précédente comment installer jQuery. Maintenant, rentrons dans le vif du sujet, l'utilisation de la librairie.

Nous allons créer un nouveau fichier *app.js* contenant un script modifiant une page web. Nous utiliserons un dossier *js* pour contenir les Scripts.

Voici par exemple le contenu de la page "index.html" :

```
1 <!DOCTYPE html>
2 <html lang="fr">
3 <head>
4   <meta charset="UTF-8">
5   <title>Premiers pas avec jQuery</title>
6 </head>
7 <body>
8   <div id="text-html">Texte en HTML</div>
9   <div id="text-js"></div>
10  <!-- en dessous les scripts JS -->
11  <script src="js/jquery-2.1.1.min.js"></script>
12  <script src="js/app.js"></script>
13 </body>
14 </html>
15
```

*NB* : Vous pouvez faire l'import de *jQuery* avant la balise de fermeture *</body>* comme cela a été fait dans l'exemple ci-dessus. Cette technique améliore la vitesse de chargement du site internet.

Voici ensuite le contenu de notre script "*app.js*" :

```
1 $(function() {
2   $('#text-js').html('Texte renseigné par le biais de jQuery');
3 });
```

Vous pouvez tester la page pour voir le comportement du script avec le DOM.



# Conclusion

JQuery fera certainement l'objet d'un cours pendant votre formation, pour l'heure, vous pouvez trouver des informations pour mieux comprendre JQuery sur l'un des liens suivant :

- <http://tecfaetu.unige.ch/etu-mal/tetris/fritz0/stic-2/ex16/introduction-jquery/>
- <https://sutterlity.gitbooks.io/apprendre-jquery/content/index.html>
- <https://openclassrooms.com/fr/courses/1424551-jquery-ecrivez-moins-pour-faire-plus>

Et, comme avec JavaScript, il existe aussi des Script *jQuery* que vous pouvez télécharger et personnaliser pour usage personnel dans votre site web. Elle permettront de réaliser aussi rapidement des choses attrayantes dans votre page web telles que les sliders, les albums photos, les diaporama et autres :

- <http://www.outils-web.com/page-script-jquery.asp>
- <http://www.outils-web.com/page-script-jquery-diaporama.asp>
- <http://designspartan.com/ressources/35-plugins-jquery-gratuits-pour-rendre-votre-site-incroyable/>
- <https://fr.tuto.com/blog/2017/01/plugins-javascript-gratuits.htm>