

Présentation d'une librairie (jquery)

Table des matières

I. Contexte	3
II. Les bases de JQuery	3
A. Introduction à JQuery	3
B. Intégrer JQuery	4
C. Les sélecteurs JQuery	4
D. Manipulation de contenu Html	6
E. Exercice : Quiz	9
III. Approfondir JQuery	10
A. Gestion des événements	10
B. Effets et animations	11
C. JQuery et AJAX	12
D. Exercice : Quiz	13
IV. Essentiel	14
V. Auto-évaluation	15
A. Exercice	15
B. Test	18
Solutions des exercices	18

I. Contexte

Durée : 1 h

Environnement de travail : Replit

Contexte

Vous avez probablement déjà entendu parler de jQuery. Cette bibliothèque JavaScript est utilisée par de nombreux développeurs à travers le monde pour simplifier la manipulation du DOM et la gestion des événements dans les pages web.

Dans ce cours, nous allons vous guider à travers les fonctionnalités de base de jQuery et vous montrer comment vous pouvez l'utiliser pour améliorer vos projets web. Nous allons commencer par comprendre comment installer jQuery dans votre projet web, puis nous allons étudier le système permettant de sélectionner des éléments HTML à l'aide de sélecteurs CSS. Enfin, nous verrons comment manipuler ces éléments à l'aide de méthodes jQuery.

Nous aborderons également la gestion des événements, l'ajout d'effets et d'animations, et l'interaction avec des serveurs web à l'aide de requêtes AJAX. À la fin de ce cours, vous aurez une compréhension solide de la librairie jQuery et de ses fonctionnalités de base, ce qui vous permettra de l'utiliser pour améliorer vos projets web et devenir un développeur web plus efficace.

Attention

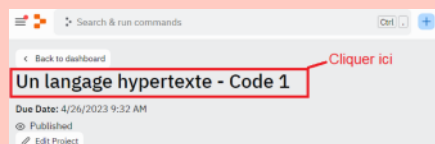
Pour avoir accès au code et à l'IDE intégré de cette leçon, vous devez :

- 1) Vous connecter à votre compte sur <https://replit.com/> (ou créer gratuitement votre compte)
- 2) Rejoindre la Team Code Studi du module via ce lien : <https://replit.com/teams/join/kffyqqqiagcezrjugixbetjackpqhpggo-javascript-studi>

Une fois ces étapes effectuées, nous vous conseillons de rafraîchir votre navigateur si le code ne s'affiche pas.

En cas de problème, redémarrez votre navigateur et vérifiez que vous avez bien accepté les cookies de connexion nécessaires avant de recommencer la procédure.

Pour accéder au code dans votre cours, cliquez sur le nom du lien Replit dans la fenêtre. Par exemple :



II. Les bases de JQuery

A. Introduction à JQuery

Définition JQuery

jQuery est une librairie JavaScript populaire qui facilite le développement web en JavaScript. jQuery a été créée en 2006 par John Resig et est maintenant utilisé par de nombreux développeurs web à travers le monde.

La principale fonctionnalité de jQuery est la sélection d'éléments HTML à l'aide de sélecteurs CSS et la manipulation de ces éléments à l'aide de méthodes JavaScript. jQuery fournit également des fonctionnalités pour gérer les événements, ajouter des effets et des animations, et interagir avec des serveurs web à l'aide de requêtes AJAX.

Remarque JQuery est mort ?

jQuery était la bibliothèque JavaScript la plus populaire au monde pendant une grande partie des années 2010. Cependant, aujourd'hui, jQuery n'est plus aussi largement utilisée qu'auparavant, et il y a plusieurs raisons à cela :

- La plupart des fonctionnalités de jQuery ont été intégrées aux navigateurs web modernes, ils ont maintenant des API natives pour la manipulation du DOM, l'interaction avec les événements, les animations, les effets, etc. Cela signifie que de nombreuses fonctionnalités que jQuery fournissait autrefois sont maintenant intégrées aux navigateurs, rendant jQuery moins nécessaire.
- Les frameworks JavaScript modernes tels que React, Vue.js et Angular ont leurs propres bibliothèques pour la manipulation du DOM et la gestion des événements, qui sont plus performantes et plus optimisées pour le rendu des applications web complexes.
- jQuery est une bibliothèque relativement lourde, ce qui signifie que l'inclure dans votre application peut ajouter une charge inutile en termes de taille de fichier et de temps de chargement.

Cependant, cela ne signifie pas que jQuery est complètement obsolète ou qu'elle n'a plus d'utilité. Elle est toujours utilisée dans beaucoup de projets web, notamment dans les sites web plus anciens qui ont été créés avant l'émergence des frameworks JavaScript modernes. De plus, les développeurs web peuvent toujours utiliser jQuery s'ils le souhaitent et s'ils trouvent que cela convient mieux à leurs besoins. Bien que peu utilisée dans de nouveaux projets, elle est encore très présente, et est donc une compétence fondamentale du développement web.

B. Intégrer JQuery

Méthode

Inclure JQuery revient à inclure le fichier JS de JQuery, tout comme vous pourriez inclure un fichier JS que vous avez codé vous-mêmes. Deux options s'offrent à vous :

1. Télécharger la bibliothèque jQuery à partir du site web officiel de jQuery

Voici le lien pour télécharger JQuery, vous pouvez télécharger la version 3.6.4, via le lien 'Download the compressed, production jQuery 3.6.4' : [jquery¹](https://jquery.com/download/)

```
1 <!-- Inclure la bibliothèque jQuery depuis les sources que j'ai téléchargé dans un dossier lib -->
2 <script src="/lib/jquery-3.6.0.min.js"></script>
```

2. Utiliser une URL vers la bibliothèque hébergée sur un CDN (content delivery network)

```
1 <!-- Inclure la bibliothèque jQuery de puis un cdn -->
2 <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
```

C. Les sélecteurs JQuery

Méthode

jQuery offre une syntaxe simple pour sélectionner des éléments HTML en utilisant des sélecteurs CSS. Par exemple, pour sélectionner tous les éléments de paragraphe dans une page, vous pouvez utiliser le sélecteur \$('p').

Vous pouvez également combiner des sélecteurs pour sélectionner des éléments plus spécifiques. Par exemple, pour sélectionner tous les éléments de paragraphe qui ont la classe highlight, vous pouvez utiliser le sélecteur \$('p.highlight').

¹ <https://jquery.com/download/>

Une fois que vous avez sélectionné des éléments HTML, vous pouvez utiliser les méthodes de jQuery pour manipuler ces éléments.

Exemple

Voici le code que l'on va ajouter à l'`index.html`.

```

1 <!DOCTYPE html>
2 <html>
3
4 <head>
5   <meta charset="utf-8">
6   <meta name="viewport" content="width=device-width">
7   <title>Présentation d'une librairie (jquery)</title>
8   <!-- Inclusions d'un fichier CSS que nous écrirons à la main -->
9   <link href="style.css" rel="stylesheet" type="text/css" />
10 </head>
11
12 <body>
13   <p id="mon-id">Je suis le paragraphe qui contient l'id #mon-id</p>
14   <p class="ma-classe">Je suis le paragraphe qui contient la classe .ma-classe</p>
15   <div>Je suis la div</div>
16   <!-- Inclusion , fichiers Javascript toujours juste avant la fermeture de la balise body-->
17   <!-- Ce code permet d'intégrer la lib jquery-->
18   <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
19   <!-- Nous écrirons notre code dans un fichier script.js -->
20   <script src="script.js"></script>
21
22 </body>
23
24 </html>

```

Quant au code Javascript, nous rédigerons le code suivant qui affichera sur notre page les phrases :

- Je suis le paragraphe qui contient l'id #mon-id
- Je suis le paragraphe qui contient la classe .ma-classe
- Je suis la div

```

1 // Sélectionner un élément par son ID
2 $('#mon-id').html();
3
4 // Sélectionner un élément par sa classe
5 $('.ma-classe').html();
6
7 // Sélectionner un élément par son type
8 $('div').html();

```

[cf.]

Remarque

Vous pouvez également sélectionner des éléments HTML en fonction de leur position dans le DOM, comme les éléments enfants, les éléments frères, etc.

```

1 <!DOCTYPE html>
2 <html>
3
4 <head>
5   <meta charset="utf-8">

```

```

6  <meta name="viewport" content="width=device-width">
7  <title>Présentation d'une librairie (jquery)</title>
8  <link href="style.css" rel="stylesheet" type="text/css" />
9  </head>
10
11 <body>
12   <div>
13     Je suis le premier enfant de body
14     <div>Je suis un enfant</div>
15   </div>
16
17   <div>
18     Je suis le second enfant de body
19   </div>
20   <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
21   <script src="script.js"></script>
22
23 </body>
24
25 </html>

```

```

1 // Sélectionner les frères d'un élément
2 $('div').siblings().css({
3   "border": "2px solid orange"
4 });
5 // Sélectionner les enfants d'un élément
6 $('div').children().css({
7   "border": "4px solid pink"
8 });

```

[cf.]

Dans cet exemple, les div du parent “Body” seront bordés d’orange tandis que les enfants des div seront bordés de rose.

D. Manipulation de contenu Html

Méthode .html()

jQuery offre plusieurs méthodes pour manipuler le contenu HTML d'un élément sélectionné. Par exemple, vous pouvez utiliser la méthode .html() pour récupérer ou définir le contenu HTML d'un élément.

```

1 <!DOCTYPE html>
2 <html>
3
4 <head>
5   <meta charset="utf-8">
6   <meta name="viewport" content="width=device-width">
7   <title>Présentation d'une librairie (jquery)</title>
8   <link href="style.css" rel="stylesheet" type="text/css" />
9 </head>
10
11 <body>
12   <div>Contenu HTML</div>
13   <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
14   <script src="script.js"></script>
15
16 </body>
17

```

```

18 </html>

1 // Récupérer le contenu HTML d'un élément
2 $('div').html();
3
4 // Définir le contenu HTML d'un élément
5 $('div').html('<p>Nouveau contenu HTML</p>');

```

[cf.]

Méthode .text()

Vous pouvez également utiliser la méthode .text() pour récupérer ou définir le contenu texte d'un élément.

```

1 <!DOCTYPE html>
2 <html>
3
4 <head>
5   <meta charset="utf-8">
6   <meta name="viewport" content="width=device-width">
7   <title>Présentation d'une librairie (jquery)</title>
8   <link href="style.css" rel="stylesheet" type="text/css" />
9 </head>
10
11 <body>
12   <div>Contenu texte</div>
13   <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
14   <script src="script.js"></script>
15
16 </body>
17
18 </html>

1 // Récupérer le contenu texte d'un élément
2 $('div').text();
3
4 // Définir le contenu texte d'un élément
5 $('div').text('Nouveau contenu texte');

```

[cf.]

Méthode .val()

Enfin, la méthode .val() peut être utilisée pour récupérer ou définir la valeur d'un élément de formulaire, comme un champ de saisie de texte.

```

1 <!DOCTYPE html>
2 <html>
3
4 <head>
5   <meta charset="utf-8">
6   <meta name="viewport" content="width=device-width">
7   <title>Présentation d'une librairie (jquery)</title>
8   <link href="style.css" rel="stylesheet" type="text/css" />
9 </head>
10
11 <body>
12   <input type="text" id="monInput" />
13   <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
14   <script src="script.js"></script>
15

```

```

16 </body>
17
18 </html>

1 // Récupérer la valeur d'un champ de saisie de texte
2 $('#monInput').val();
3
4 // Définir la valeur d'un champ de saisie de texte
5 $('#monInput').val('Nouvelle valeur');

```

[cf.]

Méthode .attr() et removeAttr

La méthode .attr() vous permettra de récupérer ou de définir la valeur d'un attribut. La méthode .removeAttr() quant à elle, vous permettra de supprimer un attribut.

```

1 <!DOCTYPE html>
2 <html>
3
4 <head>
5   <meta charset="utf-8">
6   <meta name="viewport" content="width=device-width">
7   <title>Présentation d'une librairie (jquery)</title>
8   <link href="style.css" rel="stylesheet" type="text/css" />
9 </head>
10
11 <body>
12   <input type="text"/>
13   <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
14   <script src="script.js"></script>
15
16 </body>
17
18 </html>

1 // Récupérer la valeur de l'attribut href d'un lien
2 $('a').attr('href');
3
4 // Définir la valeur de l'attribut href d'un lien
5 $('a').attr('href', 'https://www.example.com');
6
7 //Supprimer l'attribut title d'un lien
8 $('a').removeAttr('title');

```

[cf.]

Méthode .addClass() et .removeClass()

Il existe également des méthodes pour ajouter et supprimer des classes CSS d'éléments sélectionnés. Vous pouvez utiliser la méthode .addClass() pour ajouter une classe à un élément, et la méthode .removeClass() pour supprimer une classe d'un élément.

```

1 <!DOCTYPE html>
2 <html>
3
4 <head>
5   <meta charset="utf-8">
6   <meta name="viewport" content="width=device-width">

```



```
7 <title>Présentation d'une librairie (jquery)</title>
8 <link href="style.css" rel="stylesheet" type="text/css" />
9 </head>
10
11 <body>
12 <input type="text"/>
13 <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
14 <script src="script.js"></script>
15
16 </body>
17
18 </html>

1 // Ajouter la classe 'active' à un élément
2 $('div').addClass('active');
3
4 // Supprimer la classe 'active' d'un élément
5 $('div').removeClass('active');
```

[cf.]

E. Exercice : Quiz

[solution n°1 p.19]

Question 1

Quelle est la fonction principale de la librairie JQuery ?

- ☐ Manipuler le DOM
- ☐ Manipuler les tableaux
- ☐ Manipuler les chaînes de caractères

Question 2

Comment inclure JQuery dans une page HTML ?

- ☐ En utilisant la balise script avec un lien vers un fichier jquery.js
- ☐ En utilisant la balise script avec un lien vers un fichier style.css
- ☐ En utilisant la balise div avec un lien vers un fichier jquery.js

Question 3

Quel est le caractère spécial utilisé dans les sélecteurs JQuery, pour récupérer des éléments via l'identifiant Html ?

- ☐ \$
- ☐ #
- ☐ %

Question 4

Quelle est la méthode de JQuery utilisée pour sélectionner des éléments ?

- ☐ getElementById()
- ☐ querySelectorAll()
- ☐ \$()

Question 5

Quelle est la méthode de JQuery utilisée pour ajouter une classe à un élément ?

- ☐ addClass()
- ☐ setClass()
- ☐ createClass()

III. Approfondir JQuery

A. Gestion des événements

Méthode .on()

jQuery permet de gérer facilement les événements de la page web, comme les clics de souris, les saisies de clavier, etc. Vous pouvez utiliser la méthode .on() pour attacher un gestionnaire d'événements à un élément.

```

1 <!DOCTYPE html>
2 <html>
3
4 <head>
5   <meta charset="utf-8">
6   <meta name="viewport" content="width=device-width">
7   <title>Présentation d'une librairie (jquery)</title>
8   <link href="style.css" rel="stylesheet" type="text/css" />
9 </head>
10
11 <body>
12 <button>Cliquez moi</button>
13 <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
14 <script src="script.js"></script>
15
16 </body>
17
18 </html>

1 // Attacher un gestionnaire d'événements au clic de souris sur un bouton
2 $('button').on('click', function() {
3   alert('Le bouton a été cliqué');
4 });

```

[cf.]

Méthode **.hover()**

Il existe également des méthodes raccourcies pour les événements courants, comme `.click()`, `.hover()`, `.keypress()`, etc.

```

1 <!DOCTYPE html>
2 <html>
3
4 <head>
5   <meta charset="utf-8">
6   <meta name="viewport" content="width=device-width">
7   <title>Présentation d'une librairie (jquery)</title>
8   <link href="style.css" rel="stylesheet" type="text/css" />
9 </head>
10
11 <body>
12 <a href="google.fr">Survolez moi pendant que vous inspectez mon élément</a>
13 <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
14 <script src="script.js"></script>
15
16 </body>
17
18 </html>

```

```

1 // Attacher un gestionnaire d'événements au survol d'un élément
2 $('a').hover(function() {
3   //au hover, la classe 'hovered' sera ajoutée à l'élément
4   $(this).addClass('hovered');
5 }, function() {
6   //Lors de la sortie du hover, la classe 'hovered' sera retirée à l'élément
7   $(this).removeClass('hovered');
8 });

```

[cf.]

B. Effets et animations

Méthode **Les effets d'animations**

jQuery offre des méthodes pour ajouter des effets et des animations à votre page web, comme la dissimulation et l'affichage des éléments, la modification des dimensions des éléments, etc.

```

1 <!DOCTYPE html>
2 <html>
3
4 <head>
5   <meta charset="utf-8">
6   <meta name="viewport" content="width=device-width">
7   <title>Présentation d'une librairie (jquery)</title>
8   <link href="style.css" rel="stylesheet" type="text/css" />
9 </head>
10
11 <body>
12 <p>Bonjour à tous</p>
13 <div>Je suis une div</div>
14 <button>Bouton</button>
15 <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
16 <script src="script.js"></script>
17
18 </body>

```

```

19
20 </html>

1 // Cacher un élément avec un effet de fondu
2 $('p').fadeOut();
3
4 // Afficher un élément avec un effet de fondu
5 $('div').fadeIn();
6
7 // Modifier la hauteur d'un élément avec une animation
8 $('button').animate({ height: '200px' });

```

[cf.]

Voici une liste des animations que vous pouvez utiliser avec jQuery :

- .animate() : permet d'animer des propriétés CSS, comme la position, la taille, la couleur, l'opacité, etc.
- .fadeIn() : fait apparaître progressivement un élément en le rendant de plus en plus opaque.
- .fadeOut() : fait disparaître progressivement un élément en le rendant de plus en plus transparent.
- .slideDown() : fait apparaître un élément en le faisant glisser vers le bas.
- .slideUp() : fait disparaître un élément en le faisant glisser vers le haut.
- .slideToggle() : alterne entre slideDown() et slideUp().
- .show() : affiche un élément en le rendant visible.
- .hide() : cache un élément en le rendant invisible.
- .toggle() : alterne entre show() et hide().
- .stop() : arrête une animation en cours sur un élément.

C. JQuery et AJAX

Définition Les requêtes Ajax

Les requêtes AJAX (Asynchronous JavaScript and XML) permettent de communiquer avec un serveur web en arrière-plan, sans avoir à recharger la page entière. Cela permet de charger du contenu dynamiquement et d'améliorer l'expérience utilisateur en évitant des rechargements de page inutiles. La bibliothèque jQuery fournit une interface simplifiée pour effectuer des requêtes AJAX de manière asynchrone. Dans ce cours, nous allons voir comment utiliser jQuery pour effectuer des requêtes AJAX et mettre à jour dynamiquement le contenu de notre page.

Méthode \$.ajax()

La méthode \$.ajax() peut être utilisée pour récupérer des données d'un serveur. Voici un exemple de code qui effectue une requête GET pour récupérer des données au format JSON :

```

1 $.ajax({
2   url: "https://monserveur.com/api/data",
3   dataType: "json",
4   success: function(data) {
5     // Le résultat de la requête est stocké dans la variable data
6     console.log(data);
7   },
8   error: function(jqXHR, textStatus, errorThrown) {
9     console.log(textStatus, errorThrown);
10  }

```

```
11 });
```

[cf.]

Le code utilise la méthode `"ajax"` de JQuery qui prend un objet de configuration comme paramètre construit comme ceci :

- `"url"` : c'est l'URL à partir de laquelle les données doivent être récupérées.
- `"dataType"` : c'est le type de données que l'on s'attend à recevoir en réponse. Dans ce cas, il est défini sur `"json"` car les données sont au format JSON.
- `"success"` : c'est une fonction de rappel qui est appelée lorsque la requête AJAX réussit et renvoie des données. La réponse de la requête est passée à cette fonction en tant que paramètre (ici, la réponse est stockée dans la variable `"data"`).
- `"error"` : c'est une fonction de rappel qui est appelée lorsque la requête AJAX échoue. Les détails de l'erreur sont passés à cette fonction en tant que paramètres. Les trois paramètres sont :
 - `"jqXHR"` : c'est un objet qui représente la requête AJAX qui a échoué. On peut l'utiliser pour obtenir des informations sur la requête, telles que le statut de la réponse HTTP et le message d'erreur renvoyé par le serveur.
 - `"textStatus"` : c'est une chaîne de caractères qui décrit le type d'erreur qui s'est produit. Les valeurs possibles sont `"timeout"`, `"error"`, `"abort"` et `"parsererror"`.
 - `"errorThrown"` : c'est une chaîne de caractères qui contient un message d'erreur détaillé. Cette valeur dépend du type d'erreur qui s'est produit.

Il faut savoir que cette manière de faire les requêtes Ajax est très courante. Si vous faites dans votre vie professionnelle du JQuery, vous verrez ce genre de requête.

Complément Les callbacks avec \$.Ajax

La méthode `$.Ajax()` de JQuery permet également d'utiliser des callbacks pour gérer les résultats de la requête. Voici un exemple de code qui utilise les callbacks `beforeSend`, `complete`, `success` et `error` :

```
1 $.Ajax({
2   url: "https://example.com/api/data",
3   beforeSend: function() {
4     console.log("Avant l'envoi de la requête");
5   },
6   complete: function() {
7     console.log("Après la fin de la requête");
8   },
9   success: function(response) {
10    console.log("La requête a réussi :", response);
11  },
12  error: function(jqXHR, textStatus, errorThrown) {
13    console.log("La requête a échoué :", textStatus, errorThrown);
14  }
15 });
```

[cf.]

D. Exercice : Quiz

[solution n°2 p.20]

Question 1

La méthode `.animate()` de JQuery permet de créer des animations sur des propriétés CSS.

- ☐ Vrai
- ☐ Faux

Question 2

Quel événement est déclenché lorsque l'utilisateur clique sur un élément HTML avec la classe `.btn` ?

- ☐ click
- ☐ hover
- ☐ scroll

Question 3

Quel est le paramètre de la méthode `.Ajax()` de JQuery qui permet de spécifier le type de requête HTTP à envoyer ?

- ☐ type
- ☐ url
- ☐ data

Question 4

méthode `.fadeIn()` de JQuery permet de masquer un élément HTML avec une animation de fondu.

- ☐ Vrai
- ☐ Faux

Question 5

Quel est l'évènement JQuery déclenché lorsqu'un formulaire est soumis ?

- ☐ submit
- ☐ change
- ☐ keydown

IV. Essentiel

Nous avons vu que la bibliothèque JQuery est une bibliothèque JavaScript populaire qui facilite la manipulation du DOM, la gestion des événements et l'animation des pages web. Étant une des technologies majoritaires pendant plus de 10 ans, cette technologie est présente dans énormément de projets actuellement.

Nous savons maintenant comment utiliser les sélecteurs pour localiser des éléments HTML, modifier leur contenu et changer leurs styles à l'aide des méthodes de manipulation DOM de JQuery.

Nous avons également vu comment gérer les événements à l'aide des méthodes d'écoute d'événements de JQuery.

Ensuite, nous avons exploré les fonctions d'animation de JQuery telles que `.hide()`, `.show()`, `.fadeOut()`, `.fadeIn()`, et plus encore.

Enfin, nous avons découvert la méthode `$.Ajax()` de JQuery, qui permet de faire des requêtes AJAX de manière asynchrone.

V. Auto-évaluation

A. Exercice

Vous travaillez pour une entreprise de e-commerce qui souhaite améliorer l'expérience utilisateur de son site web en utilisant la librairie jQuery. Votre tâche consiste à ajouter une fonctionnalité permettant de filtrer les produits par catégorie en utilisant des boutons de sélection. Vous disposez déjà du Html, il vous faut ajouter la librairie JQuery, et coder le Javascript pour répondre aux deux problématiques.

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Mon site e-commerce</title>
5   <style>
6     body {
7       font-family: Arial, sans-serif;
8       margin: 0;
9       padding: 0;
10    }
11
12    header {
13      background-color: #333;
14      color: #fff;
15      padding: 20px;
16    }
17
18    h1 {
19      margin: 0;
20    }
21
22    nav ul {
23      list-style-type: none;
24      margin: 0;
25      padding: 0;
26    }
27
28    nav li {
29      display: inline-block;
30      margin-right: 20px;
31    }
32
33    nav a {
34      color: #fff;
35      text-decoration: none;
36    }
37
38    main {
39      padding: 20px;
40    }
41
42    #categories {
43      margin-bottom: 20px;
44    }
45
46    #categories button {
47      background-color: #333;
48      color: #fff;
49      border: none;
50      padding: 10px;
```

```
51         margin-right: 10px;
52         cursor: pointer;
53     }
54
55     #categories button:hover {
56         background-color: #666;
57     }
58
59     #products {
60         display: flex;
61         flex-wrap: wrap;
62     }
63
64     .product {
65         width: 200px;
66         background-color: #eee;
67         padding: 20px;
68         margin-right: 20px;
69         margin-bottom: 20px;
70     }
71
72     .product h2 {
73         margin-top: 0;
74     }
75
76     .product p {
77         margin: 0;
78         font-weight: bold;
79     }
80
81     .clothes {
82         border: 1px solid blue;
83     }
84
85     .electronics {
86         border: 1px solid green;
87     }
88
89     .books {
90         border: 1px solid orange;
91     }
92
93     footer {
94         background-color: #333;
95         color: #fff;
96         padding: 20px;
97         text-align: center;
98     }
99
100 </style>
101
102 </head>
103 <body>
104     <header>
105         <h1>Mon site e-commerce</h1>
106         <nav>
107             <ul>
108                 <li><a href="#">Accueil</a></li>
```



```
109     <li><a href="#">Produits</a></li>
110     <li><a href="#">Panier</a></li>
111     <li><a href="#">Contact</a></li>
112 </ul>
113 </nav>
114 </header>
115
116 <main>
117     <div id="categories">
118         <button data-category="product">Tous les produits</button>
119         <button data-category="clothes">Vêtements</button>
120         <button data-category="electronics">Electronique</button>
121         <button data-category="books">Livres</button>
122     </div>
123
124     <div id="products">
125         <div class="product clothes">
126             <h2>Chemise</h2>
127             <p>Prix : 29,99€</p>
128         </div>
129         <div class="product electronics">
130             <h2>Téléviseur</h2>
131             <p>Prix : 499,99€</p>
132         </div>
133         <div class="product books">
134             <h2>Romans</h2>
135             <p>Prix : 9,99€</p>
136         </div>
137         <div class="product clothes">
138             <h2>Pantalon</h2>
139             <p>Prix : 39,99€</p>
140         </div>
141         <div class="product electronics">
142             <h2>Ordinateur portable</h2>
143             <p>Prix : 899,99€</p>
144         </div>
145         <div class="product books">
146             <h2>Livres pour enfants</h2>
147             <p>Prix : 4,99€</p>
148         </div>
149     </div>
150 </main>
151
152 <footer>
153     <p>Tous droits réservés</p>
154 </footer>
155
156 </body>
157 <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
158 <script src="script.js"></script>
159
160 </html>
```

Question 1

[solution n°3 p.21]

Lorsque l'utilisateur clique sur un bouton de sélection, cachez tous les produits, et affichez tous les produits correspondant à la catégorie à afficher.

Indice :

C'est l'attribut 'data-category' qui vous permettra de récupérer la classe à afficher.

Question 2

[solution n°4 p.21]

Cherchez ensuite à animer les produits lorsqu'ils se masquent et s'affichent.

B. Test

Exercice 1 : Quiz

[solution n°5 p.21]

Question 1

Qu'est-ce que la librairie JQuery ?

- ☐ Un langage de programmation
- ☐ Un éditeur de code
- ☐ Une bibliothèque de fonctions JavaScript

Question 2

JQuery est encore largement utilisée aujourd'hui.

- ☐ Vrai
- ☐ Faux

Question 3

Quel est l'avantage de l'utilisation de la méthode "on" de JQuery pour attacher des événements à des éléments HTML ?

- ☐ Elle permet d'attacher des événements à des éléments dynamiquement ajoutés à la page
- ☐ Elle permet d'attacher des événements de façon synchrone
- ☐ Elle permet d'attacher des événements sans avoir à spécifier le type d'événement

Question 4

Quel est le rôle de la méthode .Ajax() de jQuery ?

- ☐ Elle permet de créer des animations sur une page web
- ☐ Elle permet d'envoyer des requêtes HTTP asynchrones au serveur web
- ☐ Elle permet de créer des effets visuels sur une page web

Question 5


Quel est le rôle de la méthode .animate() de jQuery ?

- ☐ Elle permet de créer des animations sur une page web
- ☐ Elle permet de valider les formulaires sur une page web
- ☐ Elle permet de manipuler le style CSS d'un élément HTML

Solutions des exercices


Exercice p. 9 Solution n°1**Question 1**

Quelle est la fonction principale de la librairie JQuery ?

- ☒ Manipuler le DOM
- ☐ Manipuler les tableaux
- ☐ Manipuler les chaînes de caractères
-  La librairie JQuery est principalement utilisée pour manipuler le Document Object Model (DOM) d'une page web.


Question 2

Comment inclure JQuery dans une page HTML ?

- ☒ En utilisant la balise script avec un lien vers un fichier jquery.js
- ☐ En utilisant la balise script avec un lien vers un fichier style.css
- ☐ En utilisant la balise div avec un lien vers un fichier jquery.js
-  En utilisant la balise script avec un lien vers un fichier jquery.js. Pour utiliser la librairie JQuery dans une page HTML, il faut inclure le fichier jquery.js via la balise script.


Question 3

Quel est le caractère spécial utilisé dans les sélecteurs JQuery, pour récupérer des éléments via l'identifiant Html ?

- ☐ \$
- ☒ #
- ☐ %
-  Le caractère # est utilisé dans les sélecteurs JQuery pour sélectionner un élément via son identifiant unique.


Question 4

Quelle est la méthode de JQuery utilisée pour sélectionner des éléments ?

- ☐ getElementById()
- ☐ querySelectorAll()
- ☒ \$()
-  La méthode \$() de JQuery est utilisée pour sélectionner des éléments du DOM.

Question 5


Quelle est la méthode de JQuery utilisée pour ajouter une classe à un élément ?

- ☒ `addClass()`
- ☐ `setClass()`
- ☐ `createClass()`
-  La méthode `addClass()` de JQuery est utilisée pour ajouter une classe à un élément.

Exercice p. 13 Solution n°2


Question 1

La méthode `.animate()` de JQuery permet de créer des animations sur des propriétés CSS.

- ☒ Vrai
- ☐ Faux
-  La méthode `.animate()` de JQuery permet de créer des animations sur des propriétés CSS en utilisant des valeurs numériques et des durées d'animation spécifiées en millisecondes.


Question 2

Quel événement est déclenché lorsque l'utilisateur clique sur un élément HTML avec la classe `.btn` ?

- ☒ `click`
- ☐ `hover`
- ☐ `scroll`
-  L'événement `click` est déclenché lorsque l'utilisateur clique sur un élément HTML, y compris ceux avec la classe `.btn`.


Question 3

Quel est le paramètre de la méthode `.Ajax()` de JQuery qui permet de spécifier le type de requête HTTP à envoyer ?

- ☒ `type`
- ☐ `url`
- ☐ `data`
-  Le paramètre `type` de la méthode `.Ajax()` permet de spécifier le type de requête HTTP à envoyer, tel que GET, POST, PUT, DELETE, etc.

Question 4

méthode `.fadeIn()` de JQuery permet de masquer un élément HTML avec une animation de fondu.

- ☐ Vrai
- ☒ Faux
-  La méthode `.fadeIn()` de JQuery permet de faire apparaître un élément HTML en utilisant une animation de fondu.


Question 5

Quel est l'évènement JQuery déclenché lorsqu'un formulaire est soumis ?

☒ submit

☐ change

☐ keydown

 L'événement submit est déclenché lorsqu'un formulaire est soumis, que ce soit en cliquant sur un bouton de soumission ou en appuyant sur la touche Entrée lorsque le focus est sur un champ de formulaire.

p. 18 Solution n°3

```
1 // Attend que la page soit chargée
2 $(document).ready(function() {
3
4     // Ajoute un gestionnaire d'événement pour le clic sur un bouton de catégorie
5     $('#categories button').click(function() {
6         // Récupère la valeur du bouton cliqué
7         var category = $(this).attr('data-category');
8
9         // Masque tous les produits
10        $('.product').hide();
11
12        // Affiche seulement les produits correspondant à la catégorie sélectionnée
13        $('.' + category).show();
14    });
15 });
```

[cf.]

p. 18 Solution n°4


```
1 // Attend que la page soit chargée
2 $(document).ready(function() {
3
4     // Ajoute un gestionnaire d'événement pour le clic sur un bouton de catégorie
5     $('#categories button').click(function() {
6         // Récupère la valeur du bouton cliqué
7         var category = $(this).attr('data-category');
8
9         // Masque tous les produits
10        $('.product').slideUp();
11
12        // Affiche seulement les produits correspondant à la catégorie sélectionnée
13        $('.' + category).slideDown();
14    });
15 });
```

[cf.]

Exercice p. 18 Solution n°5


Question 1

Qu'est-ce que la librairie JQuery ?

- ☐ Un langage de programmation
- ☐ Un éditeur de code
- ☒ Une bibliothèque de fonctions JavaScript
-  La librairie JQuery est une bibliothèque de fonctions JavaScript permettant de simplifier l'écriture de code pour manipuler le HTML, les événements, les animations, les requêtes Ajax, etc.


Question 2

JQuery est encore largement utilisée aujourd'hui.

- ☒ Vrai
- ☐ Faux
-  Bien que d'autres bibliothèques JavaScript comme React ou Angular aient gagné en popularité, JQuery est toujours largement utilisée aujourd'hui.


Question 3

Quel est l'avantage de l'utilisation de la méthode "on" de JQuery pour attacher des événements à des éléments HTML ?

- ☒ Elle permet d'attacher des événements à des éléments dynamiquement ajoutés à la page
- ☐ Elle permet d'attacher des événements de façon synchrone
- ☐ Elle permet d'attacher des événements sans avoir à spécifier le type d'événement
-  L'avantage de la méthode "on" de JQuery est qu'elle permet d'attacher des événements à des éléments dynamiquement ajoutés à la page, contrairement aux méthodes d'attache d'événements classiques qui ne fonctionnent que sur les éléments existants au chargement de la page.

Question 4

Quel est le rôle de la méthode .Ajax() de jQuery ?

- ☐ Elle permet de créer des animations sur une page web
- ☒ Elle permet d'envoyer des requêtes HTTP asynchrones au serveur web
- ☐ Elle permet de créer des effets visuels sur une page web
-  La méthode .Ajax() de jQuery permet d'envoyer des requêtes HTTP asynchrones au serveur web et de recevoir les données renvoyées.

Question 5

Quel est le rôle de la méthode .animate() de jQuery ?

- ☒ Elle permet de créer des animations sur une page web
- ☐ Elle permet de valider les formulaires sur une page web
- ☐ Elle permet de manipuler le style CSS d'un élément HTML

Q La méthode `.animate()` de jQuery permet de créer des animations sur une page web en modifiant progressivement les propriétés CSS d'un élément HTML.