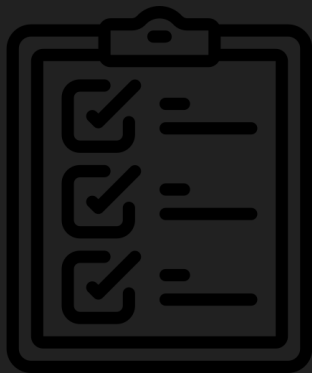


jQuery

Simplifiez la manipulation du DOM et les interactions JavaScript

Plan du cours



- Introduction à jQuery
- Sélecteurs jQuery
- Manipulation du DOM
- Gestion des événement
- Effets et Animations

jQuery



Introduction à jQuery

Qu'est-ce que jQuery ?

- Bibliothèque JavaScript
- Écriture simplifiée et concise du code JavaScript
- Facilite la manipulation du DOM et la gestion des événements

Introduction à jQuery

Qu'est-ce que jQuery ?

Une bibliothèque en informatique est un ensemble de fonctions, de procédures ou de modules pré écrits qui sont regroupés pour être réutilisés dans le développement de logiciels.

Les bibliothèques informatiques permettent aux développeurs de gagner du temps et d'optimiser leur travail en utilisant du code préexistant plutôt que de devoir tout développer eux-mêmes. Elles sont conçues pour accomplir des tâches courantes et récurrentes.

Introduction à jQuery

Avantages de l'utilisation de jQuery

- Sélection simplifiée des éléments du DOM
- Manipulation facile du contenu, des styles et des attributs des éléments
- Gestion simplifiée des événements
- Effets et animations prédéfinis pour des interfaces dynamiques
- Prise en charge d'Ajax pour des mises à jour asynchrones

Introduction à jQuery

Intégration de jQuery dans un projet

- Téléchargement de la bibliothèque jQuery depuis le site officiel
- Inclusion de jQuery dans la page HTML via une balise `<script>`
- Utilisation d'une version locale de jQuery ou d'un CDN

Introduction à jQuery

Intégration de jQuery dans un projet

```
<!DOCTYPE html>
<html lang="en">
<head>
  <script src="https://cdn.jsdelivr.net/npm/jquery@3.6.0/dist/jquery.min.js"></script>
  <title>Document</title>
</head>

<body>
  <script>
    // votre script ici
  </script>
</body>
```


Les sélecteurs

Premiers pas avec jQuery

- Sélection d'éléments avec des sélecteurs de base
- Utilisation de `$()` ou `jQuery()` pour sélectionner des éléments
- Exemples de sélecteurs d'éléments par ID, classe et balise

En utilisant le symbole "\$", ou le mot jQuery vous pouvez accéder aux fonctionnalités de la bibliothèque jQuery et manipuler le DOM de manière concise.

Les sélecteurs

Sélection par l'id

```
<body>
  <form>
    <input type="text" id="nom">
  </form>
  <script>
    // récupère l'élément qui a l'id nom
    let element = $("#nom");
    console.log(element); // afficher l'élément récupéré
  </script>
</body>
```

Les sélecteurs

Sélection par classe

```
<body>
  <p class="para">Lorem ipsum...</p>
  <div class="para">Lorem ipsum dolor sit...</div>
  <script>
    // récupère tous les éléments avec la classe para
    let elements = jQuery(".para");
    console.log(elements); // affiche la liste des éléments récupérés
  </script>
</body>
```

Les sélecteurs

Sélection par balise

```
<body>
  <p>Lorem ipsum...</p>
  <p>Lorem ipsum dolor sit...</p>
  <script>
    // récupère tous les balise p
    let elements = jQuery("p");
    console.log(elements); // affiche la liste des éléments récupérés
  </script>
</body>
```

Les sélecteurs

Recuperer la valeur d'un input

```
<body>
  <form>
    <input id="nom">
  </form>
  <script>
    // récupère la valeur du input
    let elements = $("#nom").val();
    console.log(elements); // affiche une chaîne de caractère vide
  </script>
</body>
```

Les sélecteurs

Modifier la valeur d'un input

```
<body>
  <form>
    <input id="nom">
  </form>
  <script>
    // modifie la valeur du input
    let elements = $("#nom").val("la valeur");
  </script>
</body>
```

Manipulation du DOM

Manipulation du contenu des éléments

- Accès et modification des propriétés textuelles des éléments
- Modification du contenu HTML des éléments
- Ajout, suppression et modification des classes CSS des éléments

Manipulation du DOM

Pour accéder et modifier les propriétés textuelles des éléments en jQuery, vous pouvez utiliser les méthodes `text()` et `html()`. Voici comment les utiliser :

```
<body>
  <p id="paragraphe">Lorem ipsum...</p>
  <script>
    var text = $('#paragraphe').text(); // récupérer le contenu de
l'élément
    console.log(text); // affiche "Lorem ipsum..."
  </script>
</body>
```

Dans cet exemple, la méthode `text()` est utilisée pour récupérer le contenu textuel de l'élément avec l'id "paragraphe". La valeur textuelle est stockée dans la variable `text`.

Manipulation du DOM

Ce code remplace le contenu textuel de l'élément avec l'id "paragraphe" par le texte "Nouveau texte". La méthode text() est utilisée avec un argument pour définir la nouvelle valeur textuelle.

```
<body>
  <p id="paragraphe">Lorem ipsum...</p>
  <script>
    // remplace le texte "Lorem ipsum..." par "Nouveau texte"
    $('#paragraphe').text('Nouveau texte');
    let text = $('#paragraphe').text();
    console.log(text); // affiche "Nouveau texte"
  </script>
</body>
```

Manipulation du DOM

Ce code remplace le contenu HTML de l'élément avec l'id "paragraphe" par la balise `Nouveau contenu`. La méthode `html()` est utilisée avec un argument pour définir le nouveau contenu HTML.

```
<body>
  <p id="paragraphe">Lorem ipsum...</p>
  <script>
    // créer une balise strong avec le texte "Nouveau texte" qu'il injecte dans le paragraphe
    $('#paragraphe').html('<strong>Nouveau contenu</strong>');
    let contenu = $('#paragraphe').html();
    console.log(contenu); // affiche "<strong>Nouveau contenu</strong>"
  </script>
</body>
```

Manipulation du DOM

Pour ajouter, supprimer et modifier les classes CSS des éléments en jQuery, vous pouvez utiliser les méthodes `addClass()`, `removeClass()`, `toggleClass()` et `hasClass()`. Voici comment les utiliser :

Manipulation du DOM

Dans cet exemple, la méthode `addClass()` est utilisée pour ajouter la classe CSS "maClasse" à l'élément avec l'id "para". L'élément sélectionné aura désormais la classe "maClasse" ajoutée à sa liste de classe.

```
<body>
  <p id="para">Lorem ipsum...</p>
  <script>
    // ajoute la classe "maClasse" au paragraphe
    $('#para').addClass('maClasse');
  </script>
</body>

// au final on aura une balise p comme ça:
<p id="para" class="maClasse">Lorem ipsum...</p>
```

Manipulation du DOM

La méthode `removeClass()` est utilisée pour supprimer la classe CSS "maClasse" à l'élément avec la l'id "para". L'élément sélectionné n'aura plus la classe "maClasse" dans sa liste de classe.

```
<body>
  <p id="para" class="maClasse">Lorem ipsum...</p>
  <script>
    // retire la classe "maClasse" au paragraphe
    $('#para').removeClass('maClasse');
  </script>
</body>

// au final on aura une balise p comme ça:
<p id="para" class="">Lorem ipsum...</p>
```

Manipulation du DOM

La méthode `toggleClass()` alterne l'ajout et la suppression de la classe CSS "maClasse" sur l'éléments avec l'id "para". Si l'éléments a déjà la classe "maClasse", elle sera supprimée ; sinon, elle sera ajoutée.

```
<body>
  <p id="para" class="">Lorem ipsum...</p>
  <script>
    // retire la classe "maClasse" au paragraphe si elle existe sinon il l'ajoute
    $('#para').toggleClass('maClasse');
  </script>
</body>
```

Manipulation du DOM

La méthode `hasClass()` permet de vérifier si au moins l'un des éléments avec la classe CSS "monElement" a la classe CSS "maClasse". Elle renvoie `true` si la classe est présente et `false` sinon.

```
<body>
  <p id="para">Lorem ipsum...</p>
  <script>
    if ($('#para').hasClass('maClasse')) {
      console.log("la classe existe")
    }else{
      console.log("la classe n'existe pas")
    }
  </script>
</body>
```

Gestion des événements

Pour gérer les événements avec jQuery, vous pouvez utiliser la méthode `on()`. Elle permet d'attacher des gestionnaires d'événements à des éléments sélectionnés. Voici comment utiliser `on()` pour gérer les événements :

```
$ ( '#monElement' ) .on ( 'click', function () {  
    // Code à exécuter lors du clic sur l'élément  
} ) ;
```


Gestion des événements

```
<body>  
  <button id="para">click moi</button>  
  <script>  
    $('#para').on('click', function() {  
      console.log("click détecté sur le bouton")  
    });  
  </script>  
</body>
```

Effets et Animations

Les effets et animations sont des fonctionnalités de jQuery qui permettent d'ajouter des transitions et des animations visuelles à un site web.

Avec jQuery, vous pouvez ajouter différents effets et animations à vos éléments pour améliorer l'expérience utilisateur.

Voici quelques exemples d'effets et d'animations couramment utilisés avec jQuery :

Effets et Animations

Afficher et masquer des éléments :

- La méthode `.show()` pour afficher un élément avec une transition

```
$('#monElement').show(1000);
```

- La méthode `.hide()` pour masquer un élément avec une transition

```
$('#monElement').hide(2000);
```

Effets et Animations

Afficher et masquer des éléments :

- La méthode `.fadeIn()` pour afficher un élément avec effet fondu

```
$ ( '#monElement' ) .fadeIn (1000) ;
```

- La méthode `.fadeOut()` pour masquer un élément avec effet fondu

```
$ ( '#monElement' ) .fadeOut (2000) ;
```

Effets et Animations

Afficher et masquer des éléments :

- La méthode `.slideUp()` pour faire glisser vers le haut

```
$('#monElement').slideUp(1000);
```

- La méthode `.slideDown()` pour faire glisser vers le bas

```
$('#monElement').slideDown(2000);
```

Effets et Animations

Animation personnalisée :

jQuery permet également de créer des animations personnalisées en utilisant la fonction `animate()`. Cette fonction permet de spécifier les propriétés CSS à animer, la durée de l'animation et d'autres options.

Effets et Animations

Afficher et masquer des éléments :

```
$("#monElement").animate({  
    opacity: 0.5, // Change l'opacité à 0.5  
    fontSize: '3em', // Modifie la taille de police à 20 pixels  
}, 4000); // Durée de l'animation : 4 seconde
```

Effets et Animations

Animation de glissement : la méthode `slideToggle`

La méthode `slideToggle()` de jQuery est utilisée pour afficher ou masquer un élément en utilisant une animation de glissement (slide). Son rôle principal est de basculer entre l'état affiché et l'état masqué de l'élément, en utilisant une animation fluide.

Effets et Animations

```
<body>
  <div class="myDiv">
    <h3>Titre</h3>
    <p>Contenu initial</p>
  </div>
  <button id="toggleButton">Basculer</button>
  <script>
    $("#toggleButton").click(function () {
      $(".myDiv").slideToggle();
    });
  </script>
</body>
```

Effets et Animations

Animation de glissement : la méthode `slideToggle`

Lorsque `slideToggle()` est appelé sur un élément visible, il le masque avec une animation de glissement vers le haut ou vers le bas, selon l'état actuel de l'élément. Si l'élément est masqué, `slideToggle()` l'affiche en effectuant une animation de glissement vers le bas.

Effets et Animations

Animation de glissement : la méthode `each`

La méthode `each()` de jQuery est utilisée pour itérer sur une collection d'éléments et exécuter une fonction de rappel sur chaque élément de la collection.

Effets et Animations

```
<body>
  <ul>
    <li>Élément 1</li>
    <li>Élément 2</li>
    <li>Élément 3</li>
  </ul>
  <script>
    $("li").each(function (index, element) {
      $(this).click(() => {
        console.log("Index : " + index);
        console.log("Contenu : " + $(element).text());
      })
    });
  </script>
</body>
```

Liens utils

- <https://jquery.com/>
- <https://www.w3schools.com/js/default.asp>
- <https://devdocs.io/jquery/>