

Cours n°15

Programme du cours

Activité	Durée
Initiation au javascript	10m
Sélecteurs basiques	15m
Evènement basiques	15m
Exercice Todo List	30m

Initiation au Javascript

Javascript c'est quoi ?

JavaScript est un langage de programmation principalement utilisé dans le développement web pour améliorer l'interactivité et fournir des fonctionnalités riches côté client. Dans le contexte d'ASP.NET Core MVC, JavaScript est souvent utilisé pour ajouter une interactivité côté client aux vues MVC.

Les points clés

Interactivité : JavaScript permet d'ajouter de l'interactivité à vos pages web, comme des animations, des manipulations d'éléments DOM, le traitement des événements utilisateur, etc.

Asynchrone : Avec JavaScript, vous pouvez effectuer des opérations asynchrones, comme des requêtes AJAX, qui permettent à une page web d'interagir avec un serveur web sans nécessiter de rafraîchissement complet de la page.

Manipulation du DOM : JavaScript vous permet de manipuler le Document Object Model (DOM), ce qui signifie que vous pouvez ajouter, modifier ou supprimer des éléments HTML et des attributs CSS à la volée.

Base

Le code javascript peut s'inclure de plusieurs façons dans une vue HTML. Par exemple nous pouvons placer le code entre des balises `<script>` `</script>`. Le code javascript peut avoir des fonctions. Elles sont déclarées comme suit (il n'y a pas de notion d'accessibilité) :

```
function MaSuperFonction()
{
}
```

Sélecteurs basiques

Pour sélectionner un élément du DOM en javascript, nous utilisons les fonctionnalités dans l'élément « document ». Par exemple, pour sélectionner un élément HTML par son ID, nous utiliserons le sélecteur « `getElementById` » comme ci-dessous.

```
var element = document.getElementById("MonId");
```

Pour sélectionner une liste d'éléments par class ou par nom nous devons utiliser le sélecteur « `getElementsByClassName` » et « `getElementsByName` » respectivement.

Pour créer un élément HTML via JavaScript, nous utilisons la méthode « `createElement` » de « document »

```
var newButton = document.createElement("button");
```

Une fois l'élément créer, il faut l'ajouter un DOM. Pour ça sélectionner un élément qui obtiendra ce « nouvelle enfant » puis utilisez pour injecter l'élément HTML dans l'élément sélectionné

```
element.appendChild(newButton);
```

Si l'on veut agir directement sur le contenu d'une balise HTML, nous pouvons utiliser la propriété « innerHTML »

```
element.innerHTML= "Le nouveau text du bouton";
```

Evènement basiques

En JavaScript nous utilisons des événements pour gérer les interactions avec l'utilisateur. [Liste des évènements](#). Par exemple nous avons l'évènement « onclick » qui permet d'effectuer une action lors d'un clique utilisateur.

```
<button onclick="MonActionAuClique()">Ma super action</button>

<script>
    function MonActionAuClique(){
        alert('Hello');
    }
</script>
```

Note : la fonction « alert() » permet d'afficher un message à l'utilisateur.

Debugger

Pour déboguer nous allons utiliser la console de notre navigateur appuyez sur f12 ou clique droite sur une page puis « console développeur ». Les erreurs sont affichées dans la section « console ». Nous pouvons aussi afficher du texte dans la console en utilisant la méthode « console.log('Ma Log') » dans notre code Javascript ; cela affichera le texte dans la console, ce qui peut être utile pour déboguer le code. Sinon nous pouvons aussi placer l'instruction « debugger » dans notre code. Cela aura pour effet de stopper le code sur la ligne et nous donne la possibilité de déboguer via le navigateur.

Exercice Todo list

Objectif

L'objectif est de créer une todo liste.

- Nous devons avoir la possibilité d'ajouter un nombre illimité d'élément à la liste.
- Un bouton doit afficher le nombre de tâche actuellement dans la liste via un message à l'utilisateur
- Un bouton doit permettre de supprimer tous les éléments de la liste

Maquette

TodoList Home Privacy

Ma Liste de Tâche !

Ajouter une tâche	Afficher le nombre de tâches	Supprimer tout
<input type="text" value="Nom de la tâche"/>		
<input type="text" value="Nom de la tâche"/>		
<input type="text" value="Nom de la tâche"/>		