

Les événements en JS



La dynamisation du front est l'un des usages fondamental du javascript. Pour cela, nous allons étudier dans cette mission une méthode native fondamentale de Javascript : .addEventListener()

Comment ça fonctionne????

Plutôt que des grandes phrases, on va directement voir un exemple:

Le HTML

Lorem ipsum dolor sit amet consectetur adipisicing elit. Qui suscipit eum labore
debitis, voluptas consectetur officia nam saepe beatae, laborum molestias. Eveniet soluta recusandae
deserunt ea odio incidunt voluptates perferendis?

<button id="myBtn">Changer couleur</button>



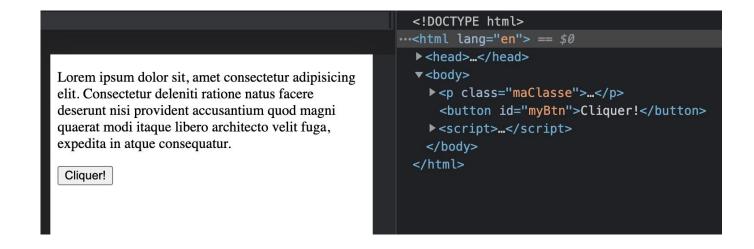
```
Le script
```

```
let btn = document.querySelector('#myBtn');
let para = document.addEventListener('.classe');
btn.addEventListener('click', function() {
    para.classList.add('couleur');
})
```

Le CSS



Le rendu: AVANT CLICK





Le rendu : APRÈS CLICK





Comme vous le voyez, avec un combo JS/CSS il est possible de réaliser des effets plus poussés qu'en pur CSS.

Je ne vais pas vous lister tous les événements disponibles en JS, la documentation officielle est là pour ça.

Ceux que vous verrez le plus souvent seront click, blur, mouseover ou encore scroll, mais cette liste n'est encore une fois pas fermée.



Nous allons à présent effectuer une mission assez longue qui va mêler habillement CSS et JS afin de réaliser le mini site que vous trouverez ici :

https://youtu.be/aNS3B3nGJeY

Dans cet exercice, vous allez devoir gérer les clicks sur des boutons, l'arrivée et la sortie de la souris sur des images avec l'ajout et la suppression de classes à la volée, modifier des attributs de balises html dynamiquement... L'exercice est assez complet et complexe, n'oubliez pas de créer des fonctions pour centraliser des comportements.



Je vous conseille de d'abord faire le html CSS, sans ça le javascript ne va pas servir à grand chose.

Je vous joins les images et la police dans les documents de la mission.

Droppez votre travail au format zip sous le nom:

Nom_Prenom_Oscars.zip