

Apprendre le JavaScript - Modifier les attributs

En JavaScript, vous pouvez modifier les attributs des éléments HTML. Vous pouvez ajouter, supprimer ou modifier des attributs. Ceci permet de modifier le comportement ou l'apparence des éléments HTML en fonction des actions de l'utilisateur.

Par exemple, au clic, vous pouvez ajouter une classe à un élément pour changer sa couleur.

Modifier le texte

Nous avons vu comment modifier le texte d'un élément HTML avec la propriété `textContent`. Vous pouvez également modifier le texte d'un élément HTML avec la propriété `innerText`. La différence entre les deux est que `textContent` renvoie tout le texte, y compris le texte caché, tandis que `innerText` renvoie le texte visible.

```
let element = document.getElementById("my-element");
element.innerText = "Nouveau texte";
```

Meilleures pratiques

Il est préférable d'utiliser `textContent` pour modifier le texte d'un élément car il est plus rapide que `innerText`.

Modifier la classe

Chaque élément HTML possède un attribut `classList` qui est une liste des classes de l'élément. C'est un objet qui possède des méthodes pour ajouter, supprimer ou vérifier une classe.

```
const element = document.querySelector("h1");
let variableBooleene = true;

element.classList.add("my-class"); // Ajoute la classe "my-class"
element.classList.remove("my-class"); // Retire la classe "my-class"
element.classList.toggle("my-class"); // Ajoute la classe "my-class" si elle
n'existe pas, sinon la retire
element.classList.toggle("my-class", variableBooleene); // Ajoute la classe "my-
class" si la variable est vraie, sinon la retire

element.classList.contains("my-class"); // Vérifie si la classe "my-class" existe
```

Modifier le dataset

Les attributs `data-*` sont des attributs personnalisés que vous pouvez ajouter à un élément HTML. Ils sont stockés dans un objet `dataset` qui est une liste des attributs `data-*` de l'élément.

```
<div id="my-element" data-name="Malek"></div>
```

```
let element = document.getElementById("my-element");
element.dataset.name = "John"; // Modifier l'attribut data-name
element.dataset.age = 25; // Ajouter un attribut data-age
delete element.dataset.name; // Supprimer l'attribut data-name
```

Dataset

Le dataset est un objet qui contient tous les attributs `data-*` d'un élément. Chaque attribut `data-*` est stocké dans le dataset sous forme de clé-valeur. Par exemple, si vous avez un attribut `data-name`, vous pouvez le modifier avec `element.dataset.name`.

Le dataset est généralement utilisé pour stocker l'état d'un élément ou ajouter des données supplémentaires à un élément. À la différence des classes, les attributs `data-*` ne sont pas utilisés pour le style, mais pour le comportement.

Ajouter un autre attribut

D'autres attributs existe sur certains éléments HTML précis:

- `href` pour les liens
- `src` pour les images
- `type`, `value`, `disabled` pour les champs de formulaire
- `checked` pour cocher une case à cocher
- `selected` pour sélectionner un élément dans une liste déroulante

On peut accéder et modifier ces attributs de la même manière que les autres attributs en utilisant la notation pointée.

```
let link = document.querySelector("a");
link.href = "https://www.example.com"; // Modifier l'attribut href
console.log(link.href); // Affiche la valeur de l'attribut href

let image = document.querySelector("img");
image.src = "image.jpg"; // Modifier l'attribut src
console.log(image.src); // Affiche la valeur de l'attribut src

let input = document.querySelector("input");
input.type = "text"; // Modifier l'attribut type
input.value = "Hello, world!"; // Modifier l'attribut value
input.disabled = true; // Modifier l'attribut disabled
console.log(input.type); // Affiche la valeur de l'attribut type
```

```
console.log(input.value); // Affiche la valeur de l'attribut value
console.log(input.disabled); // Affiche la valeur de l'attribut disabled

let checkbox = document.querySelector("input[type=checkbox]");
checkbox.checked = true; // Cocher la case à cocher
console.log(checkbox.checked); // Affiche la valeur de l'attribut checked
```

Retirer un attribut

Pour retirer un attribut, vous pouvez utiliser la méthode `removeAttribute`. Cette méthode prend en paramètre le nom de l'attribut à retirer.

```
let link = document.querySelector("a");
link.removeAttribute("href"); // Retire l'attribut href

let checkbox = document.querySelector("input[type=checkbox]");
checkbox.removeAttribute("disabled"); // Retire l'attribut checked
```