

Apprendre JavaScript - Générer du HTML

Introduction

Le HTML est simplement une longue chaîne de caractères. Vous pouvez générer du HTML en utilisant des chaînes de caractères, des templates de chaînes de caractères ou des fonctions de templates.

La propriété `innerHTML`

La propriété `innerHTML` permet de récupérer ou de définir le contenu HTML d'un élément. Si vous définissez `innerHTML` sur un élément, le contenu HTML de cet élément sera remplacé par la nouvelle valeur.

```
let element = document.querySelector("p");
let variable = "Hello";
element.innerHTML = `${variable}</strong>`;
```

Pour ajouter une liste d'éléments HTML à un élément, vous pouvez utiliser la propriété `innerHTML` avec une boucle et l'opérateur `+=`. À noter qu'à chaque itération de la boucle, le contenu HTML de l'élément sera écrasé. Les éléments sont réécrits à chaque itération. Si vous aviez mis des événements sur ces éléments, ils seront perdus.

```
let conteneurListeHTML = document.querySelector("ul");

let itemsTableau = ["Un", "Deux", "Trois"];

for (let i = 0; i < itemsTableau.length; i++) {
  conteneurListeHTML.innerHTML += `- ${itemsTableau[i]}</li>`;
}

// Si vous voulez ajouter des écouteurs d'événements, vous devez le faire après la
// boucle
const elementsListe = conteneurListeHTML.querySelectorAll("li");
elementsListe.forEach(function (element) {
  element.addEventListener("click", fonctionAExecuterAuClic);
});

```

Vider rapidement un élément HTML de son contenu

Pour vider rapidement un élément HTML de son contenu, vous pouvez simplement définir `innerHTML` sur une chaîne de caractères vide. Cela permet de réinitialiser le contenu de l'élément. Les éléments enfants de l'élément HTML seront supprimés.

```
let element = document.querySelector("ul");
element.innerHTML = "";
```

La méthode `insertAdjacentHTML`

La méthode `insertAdjacentHTML` insère un texte HTML spécifié dans une position spécifiée. Cette méthode n'écrase pas le contenu existant, mais ajoute le nouveau contenu à l'endroit spécifié. En terme de performance, `insertAdjacentHTML` est plus rapide que `innerHTML`.

```
let conteneurListeHTML = document.querySelector("ul");

let itemsTableau = ["Un", "Deux", "Trois"];

for (let i = 0; i < itemsTableau.length; i++) {
  // Ajoute le contenu à la fin de l'élément
  conteneurListeHTML.insertAdjacentHTML("beforeend", `<li>${itemsTableau[i]}</li>`);
  const elementAjout = conteneurListeHTML.querySelector("li:last-child");
  elementAjout.addEventListener("click", fonctionAExecuterAuClic);
}
```

Il existe plusieurs autres façons de générer du HTML en JavaScript, mais `innerHTML` et `insertAdjacentHTML` sont les plus couramment utilisées. Nous verrons le reste dans les prochains cours.