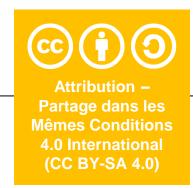
Programmation Web - Avancé

JavaScript & Node.js Génération dynamique d'HTML





Presentation template by SlidesCarnival



Génération dynamique d'HTML

J'aimerais écrire de l'HTML de manière dynamique côté client...



Génération dynamique d'HTML côté client : les possibilités

- Option A : génération à partir de String représentant l'HTML
- Option B : génération à partir d'éléments du DOM créés



 Utilisation de la propriété innerHTML d'un élément HTML DOM

```
let htmlText = `<h6>1. List through a for loop and use of innerHTML</h6>
`;
// dynamically writing HTML code through a classical loop
for (let i = 0; i < LIBRARIES.length; i++) {
   htmlText += `<li>${LIBRARIES[i]}`;
}
htmlText += "";
// setting the html content of the p#page element to add the prepared list
page.innerHTML = htmlText;
```



Quand ajouter un écouteur d'événement ?

```
// setting the content of the div#page element to add the prepared list
page.innerHTML += htmlText;
// setting event listeners on the  can only be done after that the
// ul elements have been rendered (after page.innerHTML)
const uls = document.querySelectorAll("ul");
uls.forEach((ul) => {
  ul.addEventListener("click", () => {
    alert("you click on me !");
  });
});
```





 Danger N°1 % à la position de l'ajout d'écouteurs d'événements : oubli de la MàJ du DOM

```
// Example of badly positionned listener : adding an event listener on a
list that has not yet been added to the DOM !
...
const itemList = document.querySelector("ul");
itemList.addEventListener("click", () =>
    alert("badly positionned listener will never be called")
);
// update to innerHTML reset the attached event listeners !
page.innerHTML = htmlText;
```





 Danger N°2 associé à la position de l'ajout d'écouteurs d'événements : MàJ du DOM

```
page.innerHTML = htmlText;
// example of badly positionned listener : adding an event listener on a
list that will be reset by the next DOM update of page element !
const itemList = document.querySelector("ul");
itemList.addEventListener("click", () =>
    alert("badly positionned listener will never be called")
);
...
// update to innerHTML reset the attached event listeners !
page.innerHTML += htmlText;
```



Option B : génération à partir d'éléments du DOM créés

- Création d'un élément HTML : document.createElement()
- Attacher l'élément au sein du « DOM tree » : appendChild()

```
// Get reference to the parent element
let page = document.getElementById("page");
// Create the child element
let list = document.createElement("ul");
// Attach the child element to its parent
page.appendChild(list);
```



Option B : génération à partir d'éléments du DOM créés

Quand ajouter un écouteur d'événement ?

```
let list = document.createElement("ul");
for (let i = 0; i < LIBRARIES.length; i++) {
   let item = document.createElement("li");
   item.innerText = LIBRARIES[i];
   item.addEventListener("click", onClickHandler);
   list.appendChild(item);
}
page.appendChild(list);</pre>
```



Génération dynamique d'HTML côté client & gestion d'événements



DEMO : Génération dynamique d'HTML Différents types de boucles pour générer des listes.