

Rappels divers JavaScript

I. RAPPELS SUR LA STRUCTURE D'UNE BALISE HTML



II. ACCES AU CONTENU D'UNE BALISE ET MODIFICATION

1) Quelques évènements

- `onclick` : clic sur l'élément
- `onmouseover` : survol de l'élément par le pointeur de la souris
- `onmouseout` : le pointeur de la souris sort de l'élément
- `oninput` : création d'une glissière dans le fichier HTML :

```
<input id="entree" type="range" oninput="maFonction();" min="0" max="100">
```

La fonction est appelée lorsque l'utilisateur déplace le curseur (`oninput`)

On accède ensuite à la valeur de la position du curseur (ici entre 0 et 100) :

```
let x = document.getElementById("entree")
```

`x.value`

Attention : cette valeur est de type string.

2) Accès à un élément, la méthode `getElementById()`

```
let x = document.getElementById("identifiant");
// la variable x accède à l'élément dont l'id est "identifiant"
let y = document.getElementsByTagName("select");
// y récupère la liste des éléments de balise "select"
let z = y[0];
// pour récupérer le 1er élément d'une liste
let z = document.querySelector("select");
// récupère directement le 1er élément de balise "select"
```

ou encore :

- `document.getElementsByClassName()`
- `document.querySelectorAll()`

3) Accès à un élément, relatif à un autre élément

Sur un élément `elt` identifié, on peut utiliser des méthodes ou des attributs donc le nom est assez explicite :

- `elt.childNodes` donne la liste de tous les nœuds-enfants
- `elt.firstChild` renvoie le premier nœud-enfant
- `elt.lastChild` renvoie le dernier nœud-enfant
- `elt.lastElementChild` renvoie le dernier nœud-enfant qui est un élément (pas un attribut ou du texte)
- `elt.previousSibling` renvoie le nœud-frère précédent
- `elt.nextSibling` renvoie le prochain nœud-frère
- `elt.nextElementSibling` renvoie le prochain nœud-frère qui est élément

4) Modifier le style d'un élément

```
x.style.color= "#0000FF";  
// modifie la couleur du texte en bleu (codage hexadécimal)  
x.style.color= "rgb(0,0,255)"; en décimal  
x.style.background= "#FFFF00";  
// modifie la couleur de fond de l'élément en jaune  
x.style.width=100+ "px"; fixe la largeur de l'élément à 100 pixels
```

5) Modifier le contenu d'un élément ou modifier des nœuds

On utilise la propriété `innerHTML` :

```
x.innerHTML = "nouveau contenu";
```

Sur un élément `elt` identifié, on peut utiliser des méthodes donc le nom est assez explicite :

- `elt.remove()`
- `elt.removeAttribute()`
- `elt.removeChild()`
- `elt.appendChild()`
- `elt.insertChild()`
- `elt.setAttribute()`
- `elt.createElement()`
- `elt.createTextNode()`

6) Des fonctions utiles

- `prompt("saisie de texte");` : Ouvre une boîte de dialogue avec une zone de saisie utilisateur.
- `typeof(a)` ; : affiche type de la variable `a`
- `Math.random()` ; : Renvoie un flottant pseudo-aléatoire compris dans l'intervalle $[0,1[$