

On récupère des éléments que l'on met dans une variable `elements`.

On itère sur les éléments pour leur donner un évènement en leur passant la fonction `addEventListener('click')`.

Pour finir on appelle une fonction anonyme en lui passant en paramètre `i` qui vaut le paramètre (`local cpt`)

et cela nous retourne une autre fonction anonyme qui nous affiche ce que l'on désire en l'occurrence la valeur de `i` au moment où on click sur un `Li`.

```
HTML ▼
1 <ul>
2 <li>Li 1</li>
3 <li>Li 2</li>
4 <li>Li 3</li>
5 </ul>

CSS ▼
1
2

JavaScript + No-Library (pure JS) ▼
1 var elements =
  document.getElementsByTagName('li');
2
3 for(var i = 0; i < elements.length; i++){
4   elements[i].addEventListener('click',
5     function(local_cpt){
6       return function(){
7         console.log(local_cpt);
8       };
9     }(i));
10
11
12
13
```

```
0
1
2
```

- Li 1
- Li 2
- Li 3

```
1 var elements = document.getElementsByTagName('li');
2
3 for(var i = 0; i < elements.length; i++){
4   elements[i].addEventListener('click', (function(local_cpt){
5     return function(){
6       console.log(elements[local_cpt].innerText);
7     }
8   })(i));
9 }
10
11
```

Façon de récupérer le innerText



# Les fonctions call-back

Fonction anonyme que l'on met dans la variable show qui effectue un affichage.

Fonction sum() qui prend 3 paramètres.

La fonction sum prend en 3ème paramètre la fonction anonyme mise dans la variable show.

la fonction anonyme prend en paramètre (c)

La fonction anonyme "show" se charge d'afficher le résultat.

JavaScript + No-Library (pure JS) ▼

```
1 var show = function(a){
2   console.log('show : '+a);
3 }
4
5 function sum(a,b,affiche){
6   var c = a+b;
7   affiche(c);
8   return c;
9 }
10
11 var result = sum(1,2,show);
12
13
14
```