

DOM

JS



DOM

MODIFICATION DE PAGES WEB



DOM

ORGANISER LE CODE JS



- Ou on code ?

DOM

ORGANISER LE CODE JS



<> index.html × JS main.js

<> index.html > html

```
1  <!DOCTYPE html>
2  <html lang="fr">
3  <head>
4    <meta charset="UTF-8">
5    <title>Tuto</title>
6  </head>
7  <body>
8
9  <!-- Code HTML -->
10
11 <script type="text/javascript" src="main.js"></script>
12 </body>
13 </html>
```

< > C | http://127.0.0.1

Elements Console

top

true

DOM

ORGANISER LE CODE JS



The screenshot shows a code editor on the left and a web browser on the right. The code editor displays the HTML file `index.html` with the following content:

```
<!DOCTYPE html>
<html lang="fr">
<head>
  <meta charset="UTF-8">
  <title>Tuto</title>
</head>
<body>
  <!-- Code HTML -->
  <!-- <script type="text/javascript" src="main.js"></script> -->
  <script>
    alert("Salut on se marre");
  </script>
</body>
</html>
```

The web browser on the right shows the URL `http://127.0.0.1:5500`. A message box is displayed with the text:

127.0.0.1:5500 indique
Salut on se marre

The browser's developer tools are open, showing the **Console** tab. The console contains a single log entry with a blue arrow icon, indicating the execution of the JavaScript code.

DOM

DOCUMENT MODELING OBJECT



- Le navigateur va lire notre document HTML
- Il en fait une « traduction » pour l'afficher d'une certaine manière.
- JS va pouvoir interagir avec ce Document crée par le navigateur
 - On va pouvoir sélectionner, modifier des éléments, en ajouter, en supprimer...

DOM

DOCUMENT MODELING OBJECT



```
index.html > html
1 <!DOCTYPE html>
2 <html lang="fr">
3 <head>
4   <meta charset="UTF-8">
5   <title>Tuto</title>
6 </head>
7 <body>
8
9 <!-- Code HTML -->
10 <h1>Hello</h1>
11 <p>Lorem ipsum dolor sit amet
12   consectetur, adipisicing elit.
13   Mollitia, magnam quibusdam
14   reprehenderit cum, sint aspernatur
15   dolore eveniet quidem, praesentium
16   enim accusantium odit rem ad earum
17   tempore illo cumque saepe quos.</p>
18 <script type="text/javascript" src="main.js">
19
20
21 </script>
22 </body>
23 </html>
```

```
JS main.js
1 // Code JavaScript ici
2
3
4 // alert("Salut");
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
```

Hello

Lorem ipsum dolor sit amet consectetur, adipisicing elit. magnam quibusdam reprehenderit cum, sint aspernatur quidem, praesentium enim accusantium odit rem ad earum illo cumque saepe quos.

Elements Console >>

top

> document

< ▼ #document

```
<!doctype html>
<html lang="fr">
  > <head>...</head>
  ▼ <body cz-shortcut-listen="true">
    <!-- Code HTML -->
    <h1>Hello</h1>
    > <p>...</p>
```

DOM

DOCUMENT MODELING OBJECT



The screenshot displays a web development environment with three main panels:

- Left Panel (index.html):** Contains the HTML structure. It includes a DOCTYPE declaration, a lang attribute set to "fr", a meta charset of "UTF-8", a title "Tuto", and a body containing a heading "Hello" and a paragraph of Lorem Ipsum text. A script tag at the bottom loads "main.js".
- Middle Panel (JS main.js):** Contains JavaScript code. It starts with a comment "// Code JavaScript ici" and includes a commented-out line "// alert('Salut');".
- Right Panel (Browser):** Shows the rendered page with the heading "Hello" and the Lorem Ipsum paragraph. Below the page content is the browser's developer console, which shows the DOM tree structure for the body element, including the injected HTML and the loaded script.

```
index.html < html
1 <!DOCTYPE html>
2 <html lang="fr">
3 <head>
4   <meta charset="UTF-8">
5   <title>Tuto</title>
6 </head>
7 <body>
8
9 <!-- Code HTML -->
10 <h1>Hello</h1>
11 <p>Lorem ipsum dolor sit amet
12   consectetur, adipisicing elit.
13   Mollitia, magnam quibusdam
14   reprehenderit cum, sint aspernatur
15   dolore eveniet quidem, praesentium
16   enim accusantium odit rem ad earum
17   tempore illo cumque saepe quos.</p>
18 <script type="text/javascript" src="main.js"></script>
19
20
21 </body>
22 </html>
```

```
JS main.js
1 // Code JavaScript ici
2
3
4 // alert("Salut");
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
```

Browser Console (Elements tab):

```
> document.body
< body cz-shortcut-listen="true">
  <!-- Code HTML -->
  <h1>Hello</h1>
  <p>...</p>
  <script type="text/javascript" src="main.js"></script>
  <!-- Code injected by live-server -->
  <script type="text/javascript">...</script>
```


DOM

TROUVER UN ÉLÉMENT



- Comment trouver un élément dans notre code HTML via JS ?
- On va stocker un élément du DOM dans une variable.
- On va utiliser `getElementsByTagName ()`
(récupérer un élément selon le nom de son tag)
(C'est le nom des balises html p, h1, body, script etc ...)
- Par défaut `getElementsByTagName ()` récupère des données et les stocke sous formes de tableau.
(même vide si il ne trouve rien)

DOM

TROUVER UN ÉLÉMENT



The image shows a web browser window on the right and a code editor on the left. The code editor has two tabs: 'index.html' and 'main.js'. The 'main.js' tab is active, showing the following JavaScript code:

```
1 // Code JavaScript ici
2
3
4 const titre = document.getElementsByTagName("h1");
5
6 console.log(titre);
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
```

The browser window on the right displays the rendered page. It has a title bar with 'Tuto' and standard navigation buttons. The page content includes a large heading 'Hello' and a paragraph of Lorem Ipsum text. Below the page content, the browser's developer tools are open, showing the 'Elements' and 'Console' panels. The 'Console' panel is active, displaying the output of the JavaScript code from 'main.js:6':

```
▼ HTMLCollection [h1] ⓘ
  length: 1
  ▼ 0: h1
    align: ""
    title: ""
    lang: ""
    translate: true
    dir: ""
    hidden: false
    accessKey: ""
```

DOM

TROUVER UN ÉLÉMENT



- Comment trouver un élément dans notre code HTML via JS ?
- On peut aussi chercher un élément html par son id
- On va utiliser `getElementById ()`
(récupérer un élément selon le nom de son tag)
(C'est le nom de l'id des balises html)
- `getElementById ()` renvoi directement les données elles mêmes
et non pas sous forme de tableau.

DOM

TROUVER UN ÉLÉMENT

The image is a screenshot of a web browser and a code editor. The code editor on the left shows a file named 'main.js' with the following JavaScript code:

```
1 // Code JavaScript ici
2
3
4 const titre = document.getElementById("titre");
5 // const titre = document.body.getElementsByTagName("h1");
6
7 console.log(titre);
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
```

The browser on the right shows a page with the title 'Hello' and a paragraph of Lorem ipsum text. The browser's developer tools are open, showing the 'Elements' and 'Console' panels. The 'Elements' panel shows the HTML structure, and the 'Console' panel shows the output of the JavaScript code: `<h1 id="titre">Hello</h1>` from `main.js:7`.