

BASE JAVASCRIPT

ARRAYS



- En JS on a un type d'objet particulier
- Les arrays, un objet pour stocker des données A LA MANIÈRE d'un tableau
- On peut y stocker plusieurs données, des nombres, des chaines de caractères, on peut même stocker des tableaux dans des tableaux
- Cela permet de stocker plusieurs informations dans 1 Variable
- En JS on utilise les crochets [] que l'on assigne à une variable.
- On sépare chaque élément du tableau par une ,

BASE JAVASCRIPT

ARRAYS



```
4  
5  
6  
7 var name = "José";  
8 var age = 44;  
9 var array = [];  
10 console.log(array);  
11  
12  
13  
14  
15  
16  
17  
18
```



▶ []

BASE JAVASCRIPT

ARRAYS



main.js > ...

```
1 // Code JavaScript ici
2
3
4
5
6
7 var name = "José";
8 var age = 44;
9 var array = [12, "Salut", name, age];
10 console.log(array);
11
12
13
14
15
16
17
18
19
20
21
22
23
24
```

Elements Console Sources Network >>

top Filter All levels

▼ (4) [12, "Salut", "José", 44] ⓘ

- 0: 12
- 1: "Salut"
- 2: "José"
- 3: 44
- length: 4
- ▶ __proto__: Array(0)

>

BASE JAVASCRIPT

ARRAYS



The image shows a web browser window and a code editor. The code editor displays the following JavaScript code in `main.js`:

```
1 // Code JavaScript ici
2
3
4
5
6
7 var name = "José";
8 var age = 44;
9 var passions = ["Boxe", "Fleurs"];
10 var array = [name, age, passions];
11 // var array = [12, "Salut", name, age];
12 console.log(array);
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
```

The browser window shows the URL `http://127.0.0.1:5`. The console displays the output of `console.log(array)`:

```
(3) ["José", 44, Array(2)]
  0: "José"
  1: 44
  2: Array(2)
    0: "Boxe"
    1: "Fleurs"
    length: 2
    __proto__: Array(0)
  length: 3
  __proto__: Array(0)
```

BASE JAVASCRIPT

ARRAY : ACCÈS AUX DONNÉES



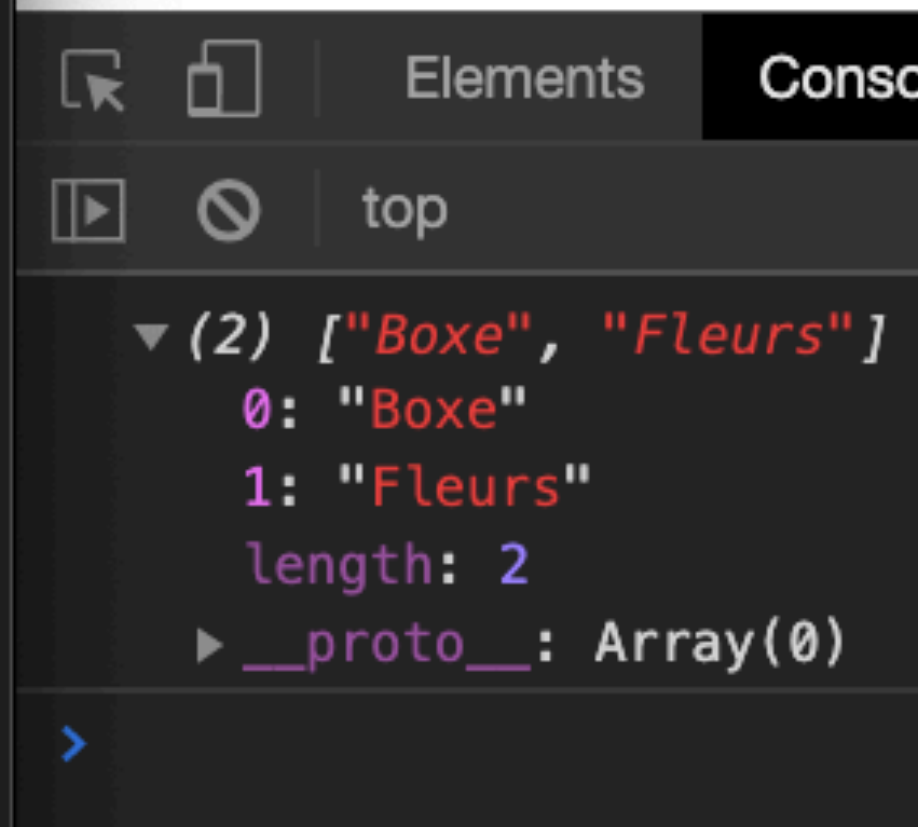
- On peut accéder spécifiquement à une données contenue dans un array.
- On utilise encore une fois les crochets [] et on renseigne un index [1]

BASE JAVASCRIPT

ARRAY : ACCÈS AUX DONNÉES



```
var name = "José";  
var age = 44;  
var passions = ["Boxe", "Fleurs"];  
var array = [name, age, passions];  
console.log(array[2]);
```

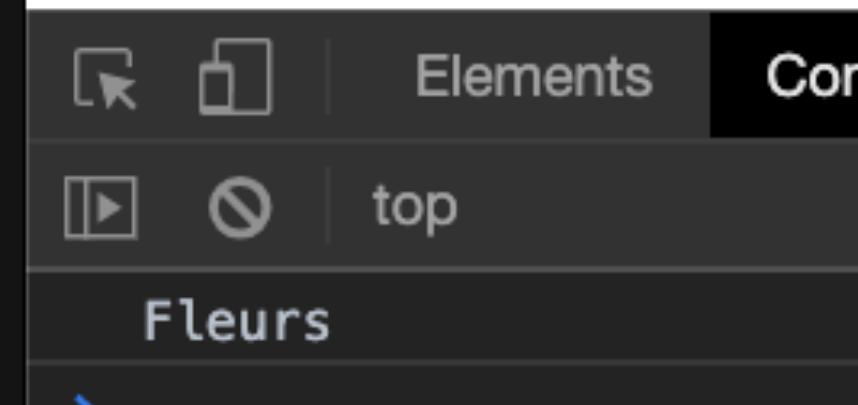


BASE JAVASCRIPT

ARRAY : ACCÈS AUX DONNÉES



```
main.js > ...  
1 // Code JavaScript ici  
2  
3  
4  
5  
6 var name = "José";  
7 var age = 44;  
8 var passions = ["Boxe", "Fleurs"];  
9 var array = [name, age, passions];  
10 console.log(array[2][1]);  
11  
12  
13  
14  
15  
16  
17  
18
```

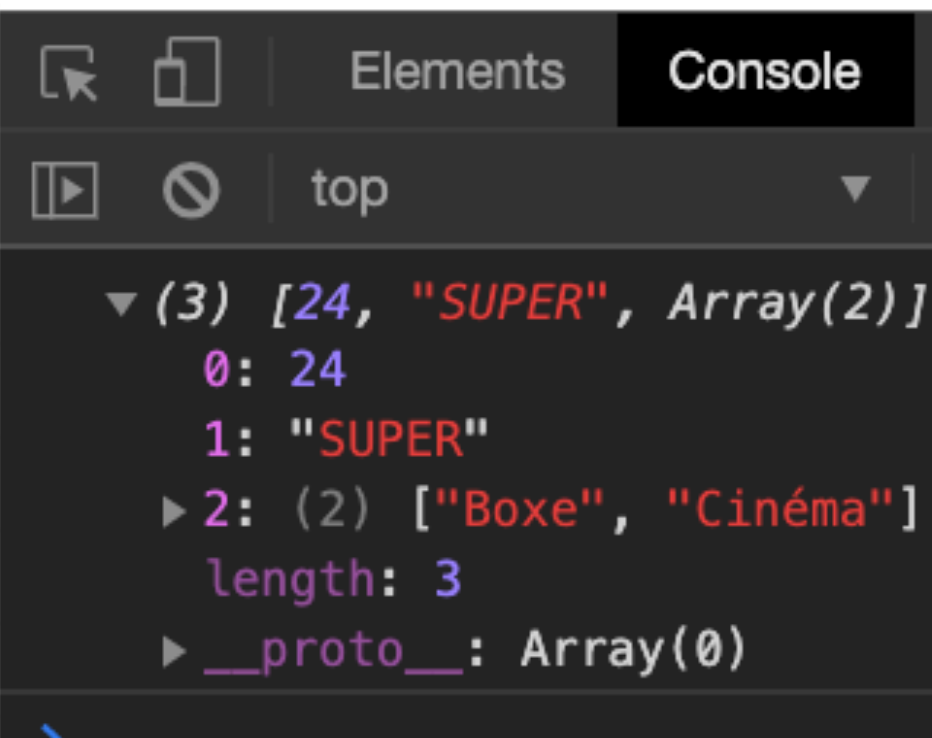


BASE JAVASCRIPT

ARRAY : ACCÈS AUX DONNÉES



```
3
4
5
6 var name = "José";
7 var age = 44;
8 var passions = ["Boxe", "Fleurs"];
9 var array = [name, age, passions];
10 array[0] = 24;
11 array[1] = "SUPER";
12 array[2][1] = "Cinéma";
13 console.log(array);
14
15
16
17
18
19
20
21
22
23
```



BASE JAVASCRIPT

ARRAY : TROUVER LA TAILLE



- On utilise length

BASE JAVASCRIPT

ARRAY : TROUVER LA TAILLE



```
JS main.js > ...  
1  // Code JavaScript ici  
2  
3  
4  
5  
6  
7  
8  var name = "José";  
9  var age = 44;  
10 var passions = ["Boxe", "Fleurs"];  
11 var array = [name, age, passions];  
12 array[0] = 24;  
13 array[1] = "SUPER";  
14 array[2][1] = "Cinéma";  
15 console.log(array[2].length);  
16  
17  
18
```

Elements Console
top
2

BASE JAVASCRIPT

ARRAY : TROUVER LA TAILLE



```
// Code JavaScript ici
```

```
var name = "José";  
var age = 44;  
var passions = ["Boxe", "Fleurs"];  
var array = [name, age, passions];  
array[1] = "SUPER";  
array[2][1] = "Cinéma";  
console.log(array[0].length);
```



Elements

Console



top



4

>

BASE JAVASCRIPT

ARRAY : AJOUTER/SUPPR DES DONNÉES



- On l'a vu les tableau permettent de stocker facilement plusieurs variables
- C'est utiles pour regrouper des variables qui ont le même contexte d'utilisation. Par exemple toutes les variable d'un utilisateur
(nom, prénom, âge, mail, mot de passe, etc...)
- Il peut s'avérer très utile de pouvoir ajouter ou supprimer des éléments d'un tableau.
(supprimer âge et ajouter date de naissance à la place)
- On va utiliser la fonction push()

BASE JAVASCRIPT

ARRAY : AJOUTER/SUPPR DES DONNÉES



5 main.js > ...

1 // Code JavaScript ici

2

3

4

5

6

7 var monTab = [100, 200, 300];

8 monTab.push("HEHE");

9 console.log(monTab);

10

11

12

13

14

15

16

17

18

19



Elements

Console



top



▶ (4) [100, 200, 300, "HEHE"]



BASE JAVASCRIPT

ARRAY : AJOUTER/SUPPR DES DONNÉES



- On peut également supprimer des données
- On va utiliser la fonction Pop()
- Par défaut pop() va supprimer le dernier élément du tableau.

BASE JAVASCRIPT

ARRAY : AJOUTER/SUPPR DES DONNÉES



JS main.js > ...

```
1 // Code JavaScript ici
2
3
4
5
6
7 var monTab = [100, 200, 300];
8 monTab.pop();
9 console.log(monTab);
10
11
12
13
14
15
16
17
18
19
20
21
```

Elements Console Sources

top

▶ (2) [100, 200]

>

BASE JAVASCRIPT

ARRAY : AJOUTER/SUPPR DES DONNÉES



main.js > ...

```
1 // Code JavaScript ici
```

```
2
```

```
3
```

```
4
```

```
5
```

```
6
```

```
7
```

```
8
```

```
9
```

```
10
```

```
11
```

```
12
```

```
13
```

```
14
```

```
15
```

```
16
```

```
17
```

```
18
```

```
19
```

```
var monTab = [100, 200, 300];
```

```
var suppr = monTab.pop();
```

```
console.log(monTab);
```

```
console.log(suppr);
```



Elements

Console



top



► (2) [100, 200]

300

BASE JAVASCRIPT

EXO: TROUVER LES INITIALES



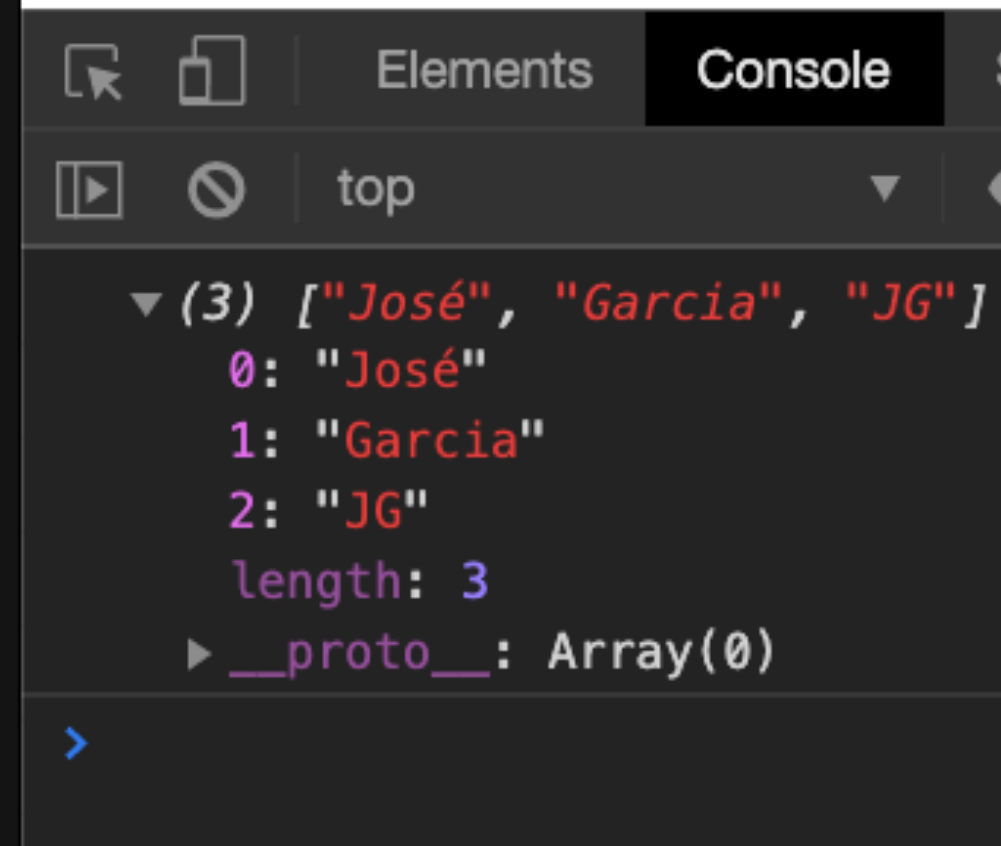
- Créer 2 variables
 - nom
 - prénom
- Créer un tableau phrase et on y ajoute
 - Le nom
 - Le prénom
 - Les initiales
- Afficher le tableau dans la console
le nom le prénom et les initiales

BASE JAVASCRIPT

EXO: TROUVER LES INITIALES



```
5  var firstName = "José";
6  var name = "Garcia";
7  var initials = firstName[0] + name[0];
8
9  var phrase = [];
10
11  phrase.push(firstName);
12  phrase.push(name);
13  phrase.push(initials);
14
15  console.log(phrase);
16
17
18
19
20
21
22
23
24
```



BASE JAVASCRIPT

EXO: TROUVER LES INITIALES



The screenshot shows a web browser window on the right and a code editor on the left. The browser's address bar shows the URL `http://127.0.0.1:5`. The code editor has two tabs: `index.html` and `JS main.js`. The `JS main.js` tab is active, showing the following JavaScript code:

```
1  // Code JavaScript ici
2
3
4
5  var firstName = "José";
6  var name = "Garcia";
7  var initials = firstName[0] + name[0];
8
9  var phrase = [];
10
11 phrase.push(firstName, name, initials);
12
13 console.log(phrase);
14
15
16
17
18
```

The browser's developer tools are open at the bottom, with the 'Console' tab selected. It shows a log entry: `(3) ["José", "Garcia", "JG"]`, indicating that the `console.log` statement executed successfully and logged the array `["José", "Garcia", "JG"]`.