

[Welcome home !](#)[Règlement intérieur](#)[Poster une issue](#)[JavaScript](#)[Environnement de dev](#)[👉 Semaine 1](#)[Terminal](#)[Variables](#)[Conditions](#)[Boucles](#)[👉 Semaine 2](#)[Manipuler des chaînes de caractères](#)[Fonctions](#)[Tableaux](#)[📺 Vidéos](#)[Introduction et utilisation](#)[Manipulation des tableaux](#)[Parcourir un tableau](#)[👉 Semaine 3](#)[Déroulement d'un programme](#)[Les booléens](#)[Les valeurs truthy et falsy](#)[Notion de scope](#)[Conditions d'existence](#)[Conditions imbriquées](#)[👉 Semaine 4](#)[Objets](#)[Requêtes à API](#)[Backend & Strapi](#)[HTML / CSS](#)

# Passer en revue les éléments d'un tableau

Pour parcourir un tableau, le principe est le même que pour les chaînes de caractères !

✓ Souvenez-vous : dans un tableau, chaque élément porte un **index** : le premier a l'index 0, le second l'index 1, etc... jusqu'au dernier élément, qui possède comme **index** la longueur du tableau - 1.

Une boucle **for** faisant varier la valeur de **i**, de l'index du premier élément à l'index du dernier élément, est donc de nouveau l'outil le plus adapté :

```
const tab = ["Xavier", "Farid", "Bastien"]

for (let i = 0; i < tab.length; i++) {
  // la boucle démarre à 0, index du premi
  // et termine à tab.length - 1, index du
  console.log(tab[i]);
}
```

Le terminal affichera :

```
Xavier
Farid
Bastien
```

Dans ce cas précis (un tableau composé exclusivement de chaînes de caractères), vous pouvez non seulement passer en revue les éléments du tableau, mais aussi passer en revue chacune des lettres de chaque élément !

✓ Si vous considérez bien la boucle faite sur le tableau, **tab[i]** est une chaîne de