**Les méthodes indispensables pour manipuler les tableaux en JavaScript**

**1. forEach()**

**À quoi ça sert ?** Exécuter une action sur chaque élément d’un tableau. Ne retourne rien.

**Exemple :**

const fruits = ["pomme", "banane", "kiwi"];

fruits.forEach(fruit => {

console.log("J'aime le fruit :", fruit);

});

**2. map()**

**À quoi ça sert ?** Créer un nouveau tableau en transformant chaque élément.

**Exemple :**

const nombres = [1, 2, 3];

const doubles = nombres.map(nombre => nombre \* 2);

console.log(doubles); // [2, 4, 6]

**3. filter()**

**À quoi ça sert ?** Créer un nouveau tableau avec seulement les éléments qui passent un test.

**Exemple :**

const nombres = [1, 2, 3, 4, 5];

const pairs = nombres.filter(n => n % 2 === 0);

console.log(pairs); // [2, 4]

**4. find()**

**À quoi ça sert ?** Trouver le premier élément qui correspond à un test. Retourne l'élément ou undefined.

**Exemple :**

const utilisateurs = [

{ id: 1, nom: "Alice" },

{ id: 2, nom: "Bob" },

];

const user = utilisateurs.find(u => u.id === 2);

console.log(user); // { id: 2, nom: "Bob" }

**5. reduce()**

**À quoi ça sert ?** Réduire un tableau à une seule valeur (nombre, string, objet, etc).

**Syntaxe :**

array.reduce((accumulateur, valeurActuelle) => {

// traitement

return nouvelleValeur;

}, valeurInitiale);

* accumulateur : la valeur accumulée à chaque étape (au début, c'est valeurInitiale)
* valeurActuelle : l'élément du tableau en cours de traitement
* valeurInitiale : la valeur de départ pour l'accumulation

**Exemple simple :**

const nombres = [1, 2, 3, 4];

const somme = nombres.reduce((acc, val) => acc + val, 0);

console.log(somme); // 10

**6. some()**

**À quoi ça sert ?** Tester si au moins un élément satisfait une condition.

**Exemple :**

const notes = [10, 12, 15];

const aBonneNote = notes.some(note => note >= 14);

console.log(aBonneNote); // true

**7. every()**

**À quoi ça sert ?** Tester si **tous** les éléments satisfont une condition.

**Exemple :**

const notes = [15, 16, 18];

const tousBons = notes.every(note => note >= 10);

console.log(tousBons); // true

**8. includes()**

**À quoi ça sert ?** Vérifier si un tableau contient une valeur.

**Exemple :**

const fruits = ["pomme", "kiwi"];

console.log(fruits.includes("kiwi")); // true

**9. sort() et reverse()**

**Trier ou inverser un tableau**

const noms = ["Zoe", "Alice", "Bob"];

console.log(noms.sort()); // ["Alice", "Bob", "Zoe"]

console.log(noms.reverse()); // ["Zoe", "Bob", "Alice"]

*Attention :* sort() modifie le tableau original.

**10. slice() et splice()**

* slice(start, end) : extrait une partie du tableau, sans le modifier
* splice(start, deleteCount, ...items) : modifie le tableau original

**Exemple :**

const animaux = ["chat", "chien", "lion", "zèbre"];

console.log(animaux.slice(1, 3)); // ["chien", "lion"]

animaux.splice(2, 1); // supprime "lion"

console.log(animaux); // ["chat", "chien", "zèbre"]

**Récapitulatif rapide**

|  |  |  |
| --- | --- | --- |
| **Méthode** | **Sert à...** | **Modifie le tableau ?** |
| forEach | Faire une action pour chaque élément | ❌ |
| map | Transformer chaque élément | ❌ |
| filter | Garder certains éléments | ❌ |
| find | Trouver un élément | ❌ |
| reduce | Réduire en une seule valeur | ❌ |
| some | Au moins un qui passe le test | ❌ |
| every | Tous doivent passer le test | ❌ |
| includes | Contient une valeur | ❌ |
| sort | Trier | ✅ |
| reverse | Inverser | ✅ |
| slice | Extraire sans modifier | ❌ |
| splice | Supprimer/Ajouter | ✅ |