

JavaScript fournit un ensemble de méthodes intégrées pour manipuler les chaînes de caractères. Ces fonctions permettent de rechercher, modifier, diviser, remplacer et comparer du texte. Voici un tour d'horizon des principales fonctions avec des exemples concrets.

1. Obtenir la longueur d'une chaîne

Méthode : `.length`

- Permet d'obtenir le nombre de caractères dans une chaîne (y compris les espaces).

Exemple :

```
let texte = "Bonjour, monde!";  
console.log(texte.length); // 15
```

2. Accéder à un caractère d'une chaîne

Méthodes : `.charAt(index)`, `[]`

- Permet d'extraire un caractère à une position donnée.

Exemple :

```
let texte = "JavaScript";  
console.log(texte.charAt(4)); // 'S'  
console.log(texte[4]);      // 'S'
```

3. Extraire une sous-chaîne

Méthodes : `.slice(start, end)`, `.substring(start, end)`, `.substr(start, length)`

Méthode	Description
<code>.slice(start, end)</code>	Extrait une partie de la chaîne entre start et end (exclu). Supporte les indices négatifs.
<code>.substring(start, end)</code>	Comme slice, mais ne supporte pas les indices négatifs.
<code>.substr(start, length)</code>	Extrait length caractères à partir de start.

Exemple :

```
let texte = "Bonjour, monde!";  
  
console.log(texte.slice(0, 7)); // "Bonjour"  
  
console.log(texte.substring(0, 7)); // "Bonjour"  
  
console.log(texte.substr(9, 5)); // "monde"
```

4. Convertir la casse d'une chaîne

Méthodes : .toUpperCase(), .toLowerCase()

- Permet de transformer une chaîne en majuscules ou en minuscules.

Exemple :

```
let texte = "Hello World";  
  
console.log(texte.toUpperCase()); // "HELLO WORLD"  
  
console.log(texte.toLowerCase()); // "hello world"
```

5. Rechercher une sous-chaîne

Méthodes : .indexOf(), .lastIndexOf(), .includes(), .startsWith(), .endsWith()

Méthode	Description
.indexOf("str")	Retourne l'index de la première occurrence de "str" ou -1 si non trouvé.
.lastIndexOf("str")	Retourne l'index de la dernière occurrence.
.includes("str")	Retourne true si "str" est présent dans la chaîne.
.startsWith("str")	Vérifie si la chaîne commence par "str".
.endsWith("str")	Vérifie si la chaîne se termine par "str".

Exemple :

```
let texte = "JavaScript est génial!";  
  
console.log(texte.indexOf("Script")); // 4
```

```
console.log(texte.lastIndexOf("e")); // 17
console.log(texte.includes("génial")); // true
console.log(texte.startsWith("Java")); // true
console.log(texte.endsWith("!")); // true
```

6. Remplacer une sous-chaîne

Méthodes : `.replace()`, `.replaceAll()`

- Permet de remplacer du texte dans une chaîne.

Exemple :

```
let texte = "Bonjour le monde!";
console.log(texte.replace("le monde", "tout le monde")); // "Bonjour tout le monde!"
console.log(texte.replaceAll("o", "0")); // "B0nj0ur le m0nde!"
```

7. Séparer une chaîne en tableau

Méthode : `.split(separateur)`

- Découpe une chaîne en un tableau selon un séparateur.

Exemple :

```
let texte = "pomme,banane,orange";
let fruits = texte.split(",");
console.log(fruits); // ["pomme", "banane", "orange"]
```

8. Supprimer les espaces inutiles

Méthode : `.trim()`, `.trimStart()`, `.trimEnd()`

- Permet de supprimer les espaces au début et à la fin d'une chaîne.

Exemple :

```
let texte = "  JavaScript  ";
```

```
console.log(texte.trim()); // "JavaScript"
console.log(texte.trimStart()); // "JavaScript  "
console.log(texte.trimEnd()); // "  JavaScript"
```

9. Répéter une chaîne

Méthode : .repeat(n)

- Répète la chaîne n fois.

Exemple :

```
let texte = "Hello ";
console.log(texte.repeat(3)); // "Hello Hello Hello "
```

10. Concaténer des chaînes

Méthodes : .concat(), +

- Permet d'assembler plusieurs chaînes en une seule.

Exemple :

```
let part1 = "Bonjour";
let part2 = "tout le monde!";
console.log(part1.concat(" ", part2)); // "Bonjour tout le monde!"
console.log(part1 + " " + part2); // "Bonjour tout le monde!"
```

11. Interpolation de chaînes (Template Literals)

Syntaxe : `Texte \${variable} texte`

- Permet d'insérer des variables dans une chaîne.

Exemple :

```
let nom = "Alice";
let age = 25;
console.log(`Bonjour, je m'appelle ${nom} et j'ai ${age} ans.`);
```

```
// "Bonjour, je m'appelle Alice et j'ai 25 ans."
```

12. Vérifier si une chaîne est vide

Méthode : Comparaison avec ""

- Vérifie si une chaîne ne contient aucun caractère.

Exemple :

```
let texte = "";  
if (texte === "") {  
    console.log("La chaîne est vide.");  
} else {  
    console.log("La chaîne contient du texte.");  
}
```

13. Inverser une chaîne

Méthode : split(), reverse(), join()

- Permet d'inverser l'ordre des caractères.

Exemple :


```
let texte = "JavaScript";  
let inverse = texte.split("").reverse().join("");  
console.log(inverse); // "tpircSavaJ"
```


Exercices d'application sur les dates et les chaînes de caractères

Exercice 1 :

Entrer deux dates et afficher la différence en jours, heures et minutes.

Calcul du Temps Écoulé

Date 1 : 

Date 2 : 

Temps écoulé : 6 jours, 0 heures et 0 minutes

Exercice 2 :

Entrer un montant en euros et convertir en plusieurs devises.

Convertisseur de Devises

Montant en € :

Devise : ▼

100 EUR = 108.00 USD

Exercice 3 :

Entrer un prix minimum pour filtrer et trier les produits. (afficher les produits dont le prix est supérieur ou égale au prix saisi)

```
const produits = [ { nom: "Clavier", prix: 40 }, { nom: "Souris", prix: 25 }, { nom: "Casque", prix: 60 }, { nom: "Chargeur", prix: 15 } ];
```

Filtrage et Tri des Produits

Prix Minimum :

- Clavier - 40€
- Casque - 60€