

# Apprendre le JavaScript - Stratégie

---

## Stratégie de développement

Une bonne stratégie lorsqu'on démarre un projet et de commencer par lister les étapes en pseudo-code sous forme de commentaires. Cela permet de clarifier les étapes à suivre et de mieux comprendre le problème à résoudre.

Ensuite, on peut commencer à écrire le code en JavaScript en suivant les étapes du pseudo-code.

Voici un exemple de pseudo-code pour une application qui demande à l'utilisateur de saisir un nombre et affiche si ce nombre est pair ou impair :

```
// Demander à l'utilisateur de saisir un nombre
// Lire le nombre saisi par l'utilisateur
// Si le nombre est pair
//     Afficher "Le nombre est pair"
// Sinon
//     Afficher "Le nombre est impair"
```

Ensuite, on peut écrire le code en JavaScript en suivant les étapes du pseudo-code :

```
let nombre = prompt("Entrez un nombre");
if (nombre % 2 === 0) {
    alert("Le nombre est pair");
} else {
    alert("Le nombre est impair");
}
```

## Pseudo-code

Le pseudo-code est un langage de description des algorithmes qui permet de représenter un algorithme de manière plus lisible et plus compréhensible que le code source. Il est utilisé pour représenter un algorithme avant de le traduire dans un langage de programmation.

Le pseudo-code est composé de mots clés, de phrases et de symboles qui permettent de décrire les étapes d'un algorithme. Il est indépendant de tout langage de programmation et peut être compris par n'importe quel programmeur.

Le pseudo-code est très utile pour représenter un algorithme de manière abstraite et pour clarifier les étapes à suivre avant de commencer à écrire le code source.

Voici des exemples de mots clés utilisés en pseudo-code :

- Si : `Si condition Alors instruction`
  - serait `if (condition) { instruction }` en JavaScript
- Sinon : `Si condition Alors instruction Sinon instruction`
  - serait `if (condition) { instruction } else { instruction }` en JavaScript
- Sinon si : `Si condition Alors instruction Sinon Si condition Alors instruction`
  - serait `if (condition) { instruction } else if (condition) { instruction }` en JavaScript
- Tant que : `Tant que condition Faire instruction`
  - serait `while (condition) { instruction }` en JavaScript
- Pour : `Pour variable de début à fin Faire instruction`
  - serait `for (let i = début; i <= fin; i++) { instruction }` en JavaScript
- Lire : `Lire variable`
  - serait `let variable = prompt("Entrez une valeur")` en JavaScript
- Afficher : `Afficher variable`
  - serait `alert(variable)` en JavaScript