

Différence ENTRE LES DEUX MÉTHODES DE PROGRAMMATION FONCTIONNELLE ET NATIVE :

Fonction de recherche (Programmation fonctionnelle) :

1. Utiliser la méthode `filter()` pour filtrer la liste et renvoyer une nouvelle liste contenant uniquement les éléments qui correspondent à l'élément recherché.
2. Renvoyer la nouvelle liste.

Boucles `for` (Approche native) :

1. Initialiser une variable pour stocker l'index de l'élément recherché.
2. Parcourir la liste en utilisant une boucle `for`.
3. Pour chaque élément de la liste, comparer l'élément à l'élément recherché.
4. Si l'élément est trouvé, stocker l'index de l'élément dans la variable et sortir de la boucle.
5. Renvoyer l'index de l'élément recherché.

COMPARAISON :

1. La programmation fonctionnelle est plus concise et plus lisible que les boucles `for`.
2. Les deux approches ont la même complexité temporelle, mais la programmation fonctionnelle peut être légèrement plus rapide dans certains cas.

CHOIX RECOMMANDÉ :

La programmation fonctionnelle est le meilleur choix pour rechercher des éléments dans une liste.

AVANTAGES DE LA PROGRAMMATION FONCTIONNELLE :

1. Plus concise : La méthode `filter()` simplifie le processus de recherche en quelques lignes de code.
2. Plus lisible : Le code de programmation fonctionnelle est souvent plus clair et déclaratif.
3. Peut être légèrement plus rapide dans certains cas : Bien que la différence de performance soit généralement minime, la programmation fonctionnelle peut offrir une légère amélioration dans certains scénarios.

RÉSULTATS DES TESTS DE PERFORMANCE

* Programmation fonctionnelle : 128543 opérations par seconde (M ops/s)

* Boucles natives : 83160 opérations par seconde (M ops/s)

CONCLUSION

La programmation fonctionnelle est une meilleure option que les boucles `for` pour rechercher des éléments dans une liste. Elle est plus concise, plus lisible et peut être légèrement plus rapide dans certains cas.

<https://jsben.ch/nMmS7>

