

# Revision Algo

## Exercice 1 :

**Objectif :** Créer un tableau sans doublons, mais cette fois-ci, les éléments à supprimer sont des chaînes de caractères dont la longueur est identique.

## Description :

On vous fournit un tableau contenant uniquement des chaînes de caractères. Vous devez créer un nouveau tableau en éliminant les chaînes ayant la même longueur. Si plusieurs chaînes ont la même longueur, seule la première occurrence doit être conservée.

## Exemple :

```
const input = ['chat', 'chien', 'poisson', 'lion', 'dauphin', 'poney'];
```

Votre réponse :

```
console.log(dedupe); // ['chat', 'poisson', 'dauphin']
```

## Exercice 2 :

**Objectif :** Créer un tableau sans doublons, mais cette fois-ci en filtrant par la première lettre des chaînes.

## Description :

On vous fournit un tableau contenant uniquement des chaînes de caractères. Vous devez créer un nouveau tableau en éliminant les chaînes qui commencent par la même lettre. Si plusieurs chaînes commencent par la même lettre, seule la première occurrence doit être conservée.

## Exemple :

```
const names = ['Alice', 'Bob', 'Annie', 'John', 'Bert', 'Charlie'];
```

Votre réponse :

```
console.log(dedupe); // ['Alice', 'Bob', 'John']
```

## Exercice 2

### Description :

On vous donne un tableau de chaînes de caractères. Vous devez créer un nouveau tableau qui contient **uniquement les chaînes dont les lettres sont uniques**. Autrement dit, les chaînes contenant des lettres répétées doivent être éliminées.

### Exemple :

```
const words = ['apple', 'banana', 'cherry', 'orange', 'grape']
```

Votre réponse :

```
console.log(dedupe); // ['cherry', 'orange', 'grape']
```

### Explication de l'algorithme attendu :

Dans chaque exercice, vous devrez itérer sur le tableau d'origine, puis tester si un élément doit être ajouté au nouveau tableau ou non en fonction des conditions spécifiques (longueur des chaînes, première lettre, ou lettres uniques dans la chaîne). Vous devrez veiller à ne pas utiliser de méthodes natives qui pourraient faciliter l'élimination des doublons, comme `sort()`, `indexOf()`, ou d'autres méthodes déjà toutes faites.