

# Pratiquons les fonctions

[Description](#)[Sommaire](#)

## À propos de ce tutoriel

Dans ce chapitre je vous propose de faire une petite pause dans notre apprentissage et de pratiquer un peu avec des nouveaux exemples pour vérifier si les éléments que l'on a vu sont compris.

## Palindrome

00:30 - L'objectif de cet exercice est de créer une fonction `isPalindrome()` qui renverra vrai si un mot est un palindrome et false sinon. La fonction ne sera pas sensible à la casse.

```
console.log(isPalindrom('kayak')) // true
console.log(isPalindrom('SOS')) // true
console.log(isPalindrom('Kayak')) // true
console.log(isPalindrom('Bonjour')) // false
```

Pour cet exercice vous aurez besoin d'utiliser des fonctions que l'on n'a pas encore vu mais c'est l'occasion d'essayer de voir comment lire la documentation :

- [split](#)
- [reverse](#)
- [join](#)
- [toUpperCase](#)

[Voir la réponse](#)

# Moyenne

**08:03** - Dans cet exercice on souhaite modifier l'objet suivant pour ajouter une propriété `moyenne` qui contiendra la moyenne de chaque élève.

```
const students = [  
  {  
    name: 'John',  
    notes: [1, 20, 18, 19, 12]  
  },  
  {  
    name: 'Jane',  
    notes: [17, 18, 20, 13, 15]  
  },  
  {  
    name: 'Sophie',  
    notes: [17, 12, 14, 15, 13]  
  },  
  {  
    name: 'Marc',  
    notes: [2, 3, 5, 8, 9]  
  },  
  {  
    name: 'Manon',  
    notes: [18, 17, 18, 19, 12]  
  }  
]
```

Une fois cet objet modifié on créera un top 3 des élèves.

Pour cet exercice vous aurez besoin de la fonction [sort](#).

Voir la réponse

# Calculateur de fréquence

**20:50** - Nous allons créer un système capable de calculer la fréquence des mots dans une chaîne de caractère. L'objectif sera dans un premier temps d'obtenir un objet qui contiendra en propriété le mot, et en valeur le nombre d'occurrence ( `{bonjour: 3, maison: 2}` )

si je devais résumer ma vie aujourd'hui avec vous, je dirais que c'est d'abord des rer

Pour cet exercice on nettoiera la chaîne en retirant les caractères qui ne sont pas des mots (, ? : ...). Pour cela vous aurez besoin de la fonction replaceAll. Vous aurez aussi besoin de mettre la phrase en minuscule à l'aide de toLowerCase

Voir la réponse

Maintenant vous devez ajouter de la logique pour extraire le top 3 des mots les plus utilisés dans la phrase.

Voir la réponse