Nom : RAHELIARISOA

Prénoms : Andriamasy Lorraine Agnès

Filière : Logiciels et Systèmes Intelligents

Atelier 3 : Javascript et Functional Programming

**Objectif :** L’objectif principal de ce Travail pratique et de se familiariser avec les concepts de la programmation fonctionnelle de langage javascript.

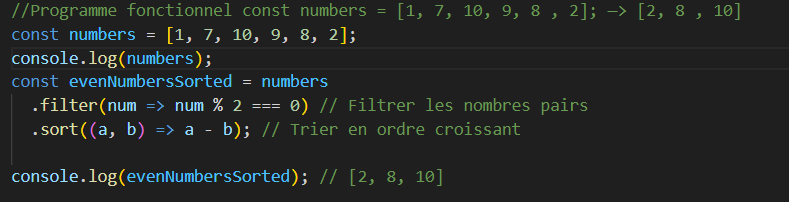
# Exercice 1 :

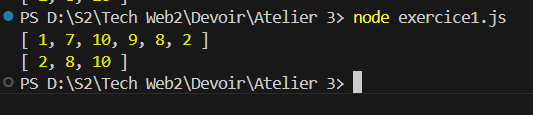
Ecrire un programme fonctionnel qui prend une liste puis renvoie un autre liste selon la logique suivante :

const numbers = [1, 7, 10, 9, 8 , 2]; —> [2, 8 , 10]  
Les critères sont :

- qu’il soit un nombre pair

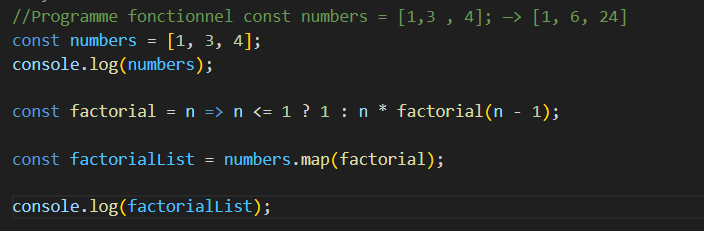
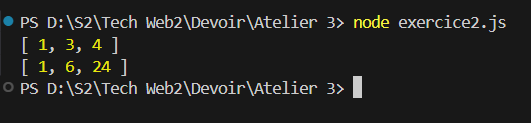
-Ranger par ordre croissant  
On utilise la fonction d’ordre supérieur filter() car elle crée un nouveau tableau en filtrant les éléments respectant une condition(par exemple ici, on garde les nombres pairs, c’est-à-dire le reste de la division par 2 est égal à zéro.)

Aussi, pour ranger par ordre croissant, on utilise sort() qui trie en ordre croissant.  




# Exercice 2 :

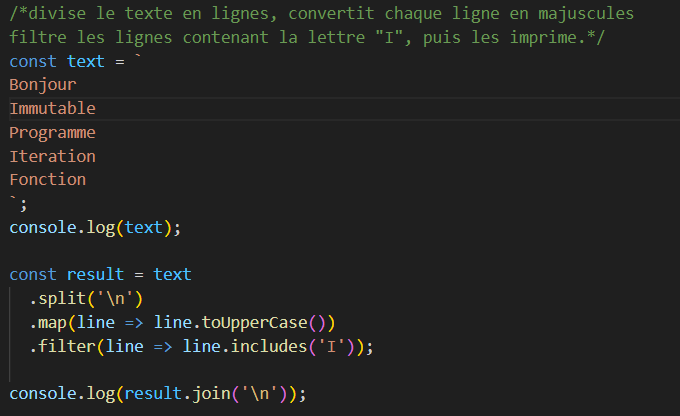
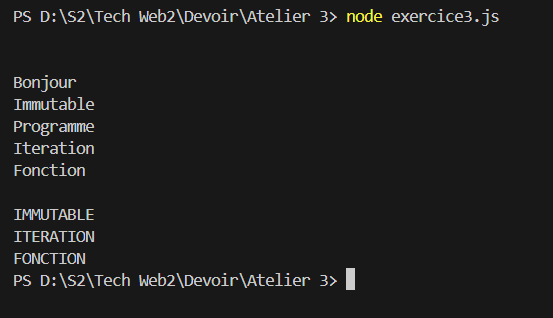
Ecrire un programme fonctionnel qui prend une liste puis renvoie un autre liste selon la logique suivante :

const numbers = [1,3 , 4]; —> [1, 6, 24]  
D’après l’analyse, elle renvoie une autre liste qui est leur factorielle.  
On utilise la fonction d’ordre supérieur map() qui applique une fonction à chaque élément du tableau et donne un nouveau tableau.  
  


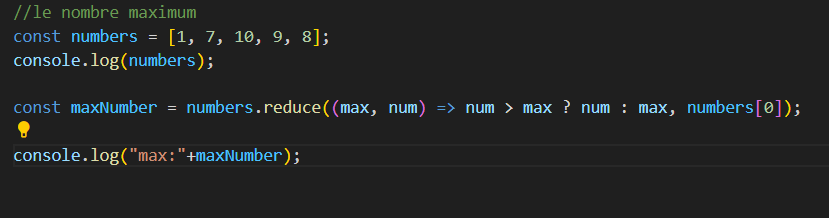
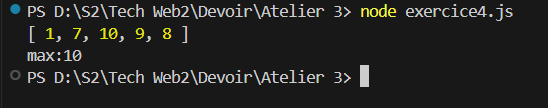
# Exercice 3 :

Ecrire un programme fonctionnel qui divise le texte en lignes, convertit chaque ligne en majuscules, filtre les lignes contenant la lettre "I", puis les imprime.

* split('\n') divise le texte en lignes.
* toUpperCase() transforme chaque ligne en majuscules.
* includes('I') filtre les lignes contenant la lettre "I" avec la fonction filter()

# Exercice 4 :

A partir d'une liste de nombres, trouver le nombre maximum dans cette liste. const numbers = [1, 7, 10, 9, 8];  
On utilise la fonction reduce() qui applique une fonction de réduction pour accumuler une valeur unique et ayant 2 paramètres : ce qui était avant et le paramètre courant.  
On va comparer chaque élément pour trouver le plus grand.  
  


# Exercice 5 :

Ecrire un programme fonctionnel qui prend un tableau d'objets représentant des produits, et calcul le prix total de tous les produits TTC (25 % TVA ).

const products = [

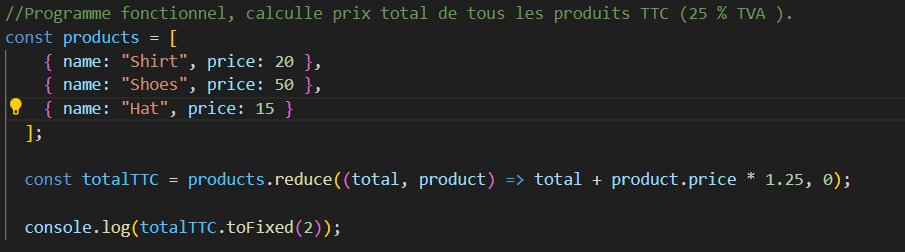
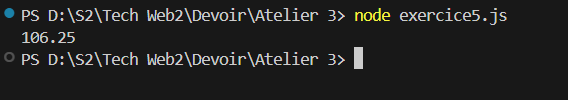
{ name: "Shirt", price: 20 },

{ name: "Shoes", price: 50 },

{ name: "Hat", price: 15 }

];

On va utiliser reduce() qui additionne le prix TTC (prix \* 1.25) et toFixed(2) formate le résultat à deux décimales.

# Exercice 6 :

Traitement des données

Vous êtes chargé de traiter un grand ensemble de données contenant des informations sur des produits. Chaque produit a des attributs tels que le nom, le prix, la catégorie et la quantité en stock.

L'objectif est d'effectuer diverses opérations sur l'ensemble de données, telles que le filtrage des produits par catégorie, le calcul des prix totaux et la recherche des produits en faible stock.

Nous devons créer une application web qui permet de traiter un ensemble de données de produits et d'afficher les résultats de différentes opérations, telles que le filtrage par catégorie, le calcul du prix total et la recherche des produits en faible stock.

Les produits sont stockées dans un fichier json selon la forme suivante :

{ name: 'Ordinateur portable', price: 999, category: 'Électronique', stock: 10 },

1. Filtrer les produits par catégorie
2. Calculer le prix total des produits
3. Trouver les produits en faible stock
4. Trier des produits par prix ou par stock : Ajoutons la possibilité de trier les produits affichés par prix ou par quantité en stock.
5. Barre de recherche : Intégrons une barre de recherche pour permettre à l'utilisateur de rechercher des produits par nom.

4. En utilisant chart.js afficher les graphes (Dashboard).

Note : pour lire depuis un fichier local : https://[www.geeksforgeeks.org/how-to-read-a-local-text-](http://www.geeksforgeeks.org/how-to-read-a-local-text-) file-using-javascript/