

Manipulation de fichiers

1) Écrire la fonction `sommeFichier(path)` qui retourne la somme de tous les nombres dans le fichier ayant le chemin d'accès `path`. Le fichier contient uniquement des nombres (0-9), des espaces et des fins de ligne.

2) Le fichier `"bottin.csv"` est un fichier CSV qui contient un bottin téléphonique associant à chaque nom un numéro de téléphone. Voici un exemple de fichier `"bottin.csv"`:

```
jean,450 831-5133
luc,514 824-9913
marie,819 123-4567
```

On cherche à écrire un programme interactif qui va permettre de consulter le bottin, soit en cherchant un nom (pour trouver le numéro de téléphone de la personne) ou un numéro de téléphone (pour trouver le nom de la personne), ou bien pour ajouter des personnes au bottin.

Le programme doit utiliser la fonction `"prompt"` pour demander la commande à l'utilisateur, et pour afficher le résultat. La commande `"fin"` ou une ligne vide doit terminer le programme. Lorsque le programme termine le nouveau fichier `"bottin.csv"` doit être sauvegardé.

Opérations sur les tableaux

1) Utilisez la méthode `map` sur les tableaux pour faire les traitements suivants :

- diviser par 2 chaque élément d'un tableau de nombres
- multiplier par `n` chaque élément d'un tableau de nombres
- créer une matrice 10x10 de nombres tel que le nombre en position `(i,j)` est égal à `i*j`, donc la matrice est une table de multiplication

2) Étant donné un tableau d'enregistrements contenant un champ `id` et un champ `notes`, tel que vu au cours pour la fusion de notes, utilisez la méthode `map` sur les tableaux pour faire les traitements suivants :

- créer un tableau de tous les identifiants
- créer un tableau de toutes les notes du Nieme travail (`n=0, 1, ...`)

3) Écrire une fonction `"somme"` avec la méthode `reduce` qui prend un tableau de nombres en paramètre et qui calcule la somme de ces nombres.

4) Écrire une fonction `"produit"` avec la méthode `reduce` qui prend un tableau de nombres en paramètre et qui calcule le produit de ces nombres.

5) Écrire une fonction `"maximum"` avec la méthode `reduce` qui prend un tableau non-vide de nombres en paramètre et qui retourne la plus grande valeur du tableau.

```
var max2 = function(a, b) {
```

```
    return a > b ? a : b;  
};
```

6) Écrire une fonction "myJoin" avec la méthode reduce qui prend en paramètre un tableau de strings et un élément avec lequel joindre chaque élément du tableau pour produire une string.