

## SERIE D'EXERCICE en TypeScript

### Exercice 1 :

Etant donné le tableau suivant qui représente les données personnelles des employés d'une entreprise ainsi que leurs villes d'intervention :

```
const personnes = [
  {type: 'user', nom: 'Max Mustermann', age: 25, villes: ['Marseille', 'Lyon', 'Paris']},
  {type: 'admin', nom: 'John Wick', age : 45, villes : ['Paris']},
  {type: 'user', nom: 'Kate Muller', age : 23, villes : ['Nantes', 'Lyon', 'Lille', 'Nice']},
  {type: 'admin', nom: 'Bruce Willis', age : 64, villes : ['Paris', 'Nantes']},
  {type: 'user', nom: 'Jack Wilson', age : 35, villes : ['Marseille', 'Lyon', 'Montpellier']},
  {type: 'admin', nom: 'Carol Smith', age : 23, villes : ['Marseille', 'Nice', 'Montpellier']}
];
```

Répondez aux questions suivantes en utilisant **forEach**, **map**, **filter**, **reduce**, **every** et **some** :

1. Ecrivez une fonction **getByType(type: string): Array<Object>** qui retourne la liste des personnes selon le type passé en paramètre.
2. Ecrivez une fonction **getByVille(ville: string): Array<Object>** qui retourne la liste des personnes ayant dans villes la ville passée en paramètre.
3. Ecrivez une fonction **getOtherVilleThan(ville: string): Array<Object>** qui retourne la liste des personnes n'ayant pas dans villes la ville passée en paramètre.
4. Ecrivez une fonction **countVilles(): Array<Object>** qui retourne un tableau d'objets : chaque objet contient le nom d'une personne ainsi que son nombre de villes (voir ci-dessous).

```
[
  { nom: 'Max Mustermann ', nbrVilles : 3 },
  { nom: 'John Wick ', nbrVilles : 1 },
  { nom: 'Kate Muller ', nbrVilles : 4 },
  { nom: 'Bruce Willis ', nbrVilles : 2 },
  { nom: 'Jack Wilson ', nbrVilles : 3 },
  { nom: 'Carol Smith ', nbrVilles : 3 }
]
```
5. Ecrivez une fonction **getByVillesNumber(nbr: number): Array<Object>** qui retourne les personnes dont le nombre de villes d'intervention correspond au paramètre nbr.
6. Ecrivez une fonction **countCharacterInVilles(): Array<Object>** qui retourne un tableau d'objets : chaque objet contient le nom d'une personne ainsi que le nombre total de caractères de ses villes (voir ci-dessous).

```
[
  { nom: 'Max Mustermann ', total : 18 },
  { nom: 'John Wick ', total : 5 },
  { nom: 'Kate Muller ', total : 19 },
  { nom: 'Bruce Willis ', total : 11 },
  { nom: 'Jack Wilson ', total : 24 },
  { nom: 'Carol Smith ', total : 24 }
]
```

7. Ecrivez une fonction **countByVille(ville: string): number** qui retourne le nombre de personnes qui interviennent dans la ville passée en paramètre (ici c'est 3 pour Marseille par exemple)
8. Ecrivez une fonction **findHavingMaxVille(): number** qui permet de retourner le nombre max de villes d'intervention de toutes les personnes (ici c'est 4)

## Exercice 2 :

Considérons une classe **Stagiaire** ayant les attributs suivants :

- nom : un attribut privé de type chaîne de caractères
  - notes : un attribut privé de type tableau de nombres
1. Créez la classe **Stagiaire** et définissez un constructeur avec deux paramètres
  2. définissez les getters et setters des deux attributs.
  3. Ecrivez la méthode **calculerMoyenne()** qui permet de retourner la moyenne de notes d'un stagiaire
  4. Ecrivez les méthodes **trouverMax()** et **trouverMin()** qui permettent de retourner respectivement les notes max et min d'un stagiaire.

Considérons maintenant une classe appelée **Formation** ayant les attributs suivants :

- intitulé : un attribut privé de type string
  - nbrJours : un attribut privé de type number
  - stagiaires : un tableau d'objets de type **Stagiaire**
1. Créez la classe **Formation**, Définissez les getters et setters de ses attributs, et définissez le constructeur **Formation** (intitulé: string, nbrJours: number, stagiaires: **Stagiaire**[])
  2. Ecrivez une méthode **calculerMoyenneFormation()** qui retourne la moyenne d'un objet de type formation (la moyenne des moyennes des stagiaires)
  3. Ecrivez une méthode **getIndexMax()** qui retourne l'indice du stagiaire dans le tableau stagiaires ayant la meilleure moyenne de la formation.
  4. Ecrivez une méthode **afficherNomMax()** qui affiche le nom du premier stagiaire ayant la meilleure moyenne d'une formation.
  5. Ecrivez une méthode **afficherMinMax()** qui affiche la note minimale du premier stagiaire ayant la meilleure moyenne d'une formation.
  6. Ecrivez une méthode **trouverMoyenneParNom(nom: string)** qui affiche la moyenne du premier stagiaire dont le nom est passé en paramètre.

Dans un fichier **main.ts**, testez toutes les méthodes réalisées dans les questions précédentes (créez par exemple trois objets `Stagiaire` et affectez-les à une même formation et faites appel aux quatre dernières méthodes que vous avez implémentées).

