	ROYAUME DU MAROC	
	مكتب التكوين المهني وإنعاش الشغل	
	Office de la Formation Professionnelle et de la Promotion du Travail	Formateur : HAFSA MHAND OUAMMI
	Filière : devofs 202	Niveau : 2 ^{ème} Année
	Module : Développer en FrontEnd	Année de formation: 2024/2025

TP 0 :

L'objectif : Rappel Javascript – les classes et les tableaux

Exercice 1 :


1. Créez une classe `Student` avec les propriétés suivantes :
 - o `name` (nom de l'étudiant)
 - o `age` (âge de l'étudiant)
 - o `grades` (un tableau des notes de l'étudiant)
2. Ajoutez les méthodes suivantes à la classe :
 - o `addGrade(grade)` pour ajouter une note au tableau des notes.
 - o `getAverageGrade()` pour retourner la moyenne des notes.
3. Créez un tableau d'étudiants avec plusieurs instances de `Student`.
Affichez le nom et la moyenne de chaque étudiant.

Exercice 2 :

1. Créez une classe `Book` avec les propriétés suivantes :
 - `title` (titre du livre)
 - `author` (auteur)
 - `year` (année de publication)
2. Créez une classe `Library` qui contient :
 - Une propriété `books` (un tableau contenant les livres).
 - Une méthode `addBook(book)` pour ajouter un livre à la bibliothèque.
 - Une méthode `findBooksByAuthor(author)` pour retourner tous les livres d'un auteur donné.
3. Ajoutez plusieurs livres dans la bibliothèque et testez la recherche par auteur.

Exercice 3 :

1. Créez une classe `Product` avec les propriétés suivantes :
 - `name` (nom du produit)
 - `price` (prix du produit)
 - `category` (catégorie du produit)
2. Créez une classe `Store` avec :
 - Une propriété `inventory` (un tableau de produits).

	ROYAUME DU MAROC	
	مكتب التكوين المهني وإنعاش الشغل	
	Office de la Formation Professionnelle et de la Promotion du Travail	Formateur : HAFSA MHAND OUAMMI
	Filière : devops 202	Niveau : 2 ^{ème} Année
	Module : Développer en FrontEnd	Année de formation: 2024/2025

- Une méthode `addProduct(product)` pour ajouter un produit.
- Une méthode `filterByCategory(category)` pour retourner tous les produits d'une catégorie donnée.
- Une méthode `calculateTotalInventoryValue()` pour calculer la valeur totale de l'inventaire.

Testez votre programme en créant un magasin avec plusieurs produits et en utilisant les méthodes.

Exercice 4 :

Un magasin enregistre ses ventes mensuelles sous forme de tableau. Voici un exemple de tableau contenant les chiffres d'affaires de chaque mois :

```
const monthlySales = [1200, 1500, 1700, 1300, 2100, 800, 900, 1800, 1900, 2500, 2300, 1600];
```


- Créez une fonction `calculateTotalSales(sales)` qui prend un tableau en paramètre et retourne le total des ventes.
- Créez une fonction `calculateAverageSales(sales)` qui calcule et retourne la moyenne mensuelle des ventes.
- Créez une fonction `findBestMonth(sales)` qui retourne l'indice (ou le mois correspondant) des meilleures ventes.
- Créez une fonction `filterLowMonths(sales, threshold)` qui retourne un nouveau tableau contenant les mois dont les ventes sont inférieures à une valeur seuil donnée (`threshold`).
- Créez une fonction `calculateMonthlyGrowth(sales)` qui retourne un tableau contenant la différence des ventes entre chaque mois consécutif (exemple : `[300, -400, 500, ...]`).

Exercice 5 :

Un enseignant suit les notes de ses élèves dans un tableau. Voici un exemple de tableau :

```
const grades = [12, 8, 15, 18, 10, 14, 9, 13, 17, 7, 20];
```

- Créez une fonction `findMaxGrade(grades)` qui retourne la note la plus élevée dans le tableau.
- Créez une fonction `findMinGrade(grades)` qui retourne la note la plus basse dans le tableau.
- Créez une fonction `calculateAverageGrade(grades)` qui calcule et retourne la moyenne des notes.

	ROYAUME DU MAROC	
	مكتب التكوين المهني وإنعاش الشغل	
	Office de la Formation Professionnelle et de la Promotion du Travail	Formateur : HAFSA MHAND OUAMMI
	Filière : devofs 202	Niveau : 2 ^{ème} Année
	Module : Développer en FrontEnd	Année de formation: 2024/2025

4. Créez une fonction `sortGrades(grades)` qui retourne un tableau trié dans l'ordre croissant.
5. Créez une fonction `filterPassingGrades(grades, passingGrade)` qui retourne un tableau contenant uniquement les notes supérieures ou égales à la note de passage (`passingGrade`).
6. Créez une fonction `addGrade(grades, newGrade)` qui ajoute une nouvelle note au tableau.
7. Créez une fonction `removeGrade(grades, gradeToRemove)` qui supprime une note spécifique du tableau.

Exercice 6 :

Un magasin dispose d'un inventaire sous forme de tableau, où chaque produit est représenté par un objet avec trois propriétés : `name`, `quantity`, et `price`. Voici un exemple d'inventaire :

```
const inventory = [
  { name: "Apple", quantity: 50, price: 0.5 },
  { name: "Banana", quantity: 30, price: 0.2 },
  { name: "Orange", quantity: 20, price: 0.8 },
  { name: "Mango", quantity: 15, price: 1.5 },
  { name: "Grapes", quantity: 40, price: 1.0 }
];
```

1. Créez une fonction `calculateTotalValue(inventory)` qui calcule la valeur totale de l'inventaire (quantité x prix de chaque produit).
2. Créez une fonction `sortByQuantity(inventory)` qui retourne le tableau trié par ordre croissant de la quantité.
3. Créez une fonction `filterLowStock(inventory, threshold)` qui retourne les produits dont la quantité est inférieure à un seuil donné (`threshold`).
4. Créez une fonction `restockProduct(inventory, productName, amount)` qui augmente la quantité d'un produit spécifique de `amount`.
5. Créez une fonction `findMostExpensiveProduct(inventory)` qui retourne le produit avec le prix le plus élevé.
6. Créez une fonction `removeOutOfStockProducts(inventory)` qui retourne un tableau ne contenant que les produits avec une quantité supérieure à 0.