**TYPESCRIPT**

**1. Création des Interfaces**

1. **Définis une interface** ILivre pour représenter un livre. Cette interface doit inclure :
   * titre (de type string)
   * auteur (de type string)
   * anneePublication (de type number)
2. **Définis une interface** IMembre pour représenter un membre de la bibliothèque. Cette interface doit inclure :
   * nom (de type string)
   * age (de type number)
   * listeLivresEmpruntes (un tableau de type ILivre[] pour stocker les livres empruntés)

**2. Création de Classes**

1. **Crée une classe** Livre qui implémente l'interface ILivre. Elle doit avoir un constructeur pour initialiser les valeurs titre, auteur et anneePublication.
2. **Crée une classe** Membre qui implémente l'interface IMembre. Elle doit avoir un constructeur pour initialiser nom, age et un tableau vide listeLivresEmpruntes.

**3. Gestion de la Bibliothèque**

1. **Crée une classe** Bibliotheque qui gère les membres et les livres disponibles.
   * Un tableau livresDisponibles (de type Livre[]) qui contient tous les livres disponibles dans la bibliothèque.
   * Un tableau membres (de type Membre[]) qui contient tous les membres inscrits.
2. **Ajoute des méthodes** dans Bibliotheque :
   * ajouterLivre(livre: Livre): ajoute un livre à livresDisponibles.
   * ajouterMembre(membre: Membre): ajoute un membre à membres.
   * emprunterLivre(nomMembre: string, titreLivre: string): cette méthode vérifie si le livre est disponible. Si oui, il est ajouté au listeLivresEmpruntes du membre et retiré de livresDisponibles.
   * retournerLivre(nomMembre: string, titreLivre: string): cette méthode permet à un membre de rendre un livre. Le livre est retiré de listeLivresEmpruntes et ajouté à livresDisponibles.

**4. Implémentation du Polymorphisme avec Héritage**

1. **Crée une classe** LivreRoman qui hérite de Livre. Ajoute une propriété genre (de type string) pour spécifier le genre littéraire du roman (par exemple, "Fiction", "Histoire", etc.).
2. **Crée une classe** LivreScience qui hérite également de Livre. Ajoute une propriété domaine (de type string) pour spécifier le domaine scientifique (par exemple, "Physique", "Chimie", etc.).
3. Dans la classe Bibliotheque, utilise le polymorphisme pour que la méthode ajouterLivre puisse ajouter aussi bien des LivreRoman que des LivreScience.

**5. Tests et Boucles**

1. **Crée une fonction** afficherLivresDisponibles(bibliotheque: Bibliotheque) qui utilise une boucle pour afficher tous les livres disponibles dans la bibliothèque.
2. **Crée une fonction** afficherLivresEmpruntes(membre: Membre) qui utilise une boucle pour afficher tous les livres empruntés par un membre spécifique.

À rendre sous forme de lien vers le dépôt du projet sur GitHub par email.  
Délai : 31/10/2024 à 10h.

Email : sow1998dara@gmail.com