En vue de la gestion d’une médiathèque, on vous demande d’écrire une application pour gérer des médias de natures diverses : CdMusical, CdVariete et Dvd.

Tous les médias ont un numéro d’enregistrement unique (un entier) et un titre (une chaîne de caractères).

**Classe Media** :

*Attributs* : titre (String), prixJournalierLocation (double), disponible (boolean). L’attribut disponible indique si le média est libre (true) ou s’il est déjà loué (false).

*Méthodes* :

toString : méthode abstraite qui renverra toutes les informations du média

getPrixJournalierLocation : méthode qui renvoie le prix journalier de location

equals : deux médias sont les mêmes s’ils ont le même titre

**Classe Dvd** : sous-classe de Media, prixJournalierLocation = 0.40€

*Attributs* : genre (int) (0=DOCUMENTAIRE, 1=FILM, 2=SERIE)

**Classe CdMusical** : sous-classe de Media, prixJournalierLocation = 0.34€

*Attributs* : auteur (String), nombrePistes (int)

**Classe CdVariete** : sous-classe de CdMusical, prixJournalierLocation = 0.30€

*Attributs* : disqueDor (boolean)

**Classe Location** :

*Attributs* : media (Media), date (String), nombreJours (int)

**Classe Mediatheque** :

*Attributs* : nom (String), listeMembres (tableau de maximum 100 membres), listeMedias (tableau de maximum 5000 medias)

*Méthodes* :

getMedias : renvoie la liste (sous forme de chaîne de caractères) des médias de la médiathèque et leur disponibilité

getMediasDisponibles : renvoie la liste (sous forme de chaîne de caractères) des médias disponibles

calculerPrixLocations : renvoie le prix totale des locations d’un membre transmis en argument

ajouterMembre : ajoute un membre transmis en argument (renvoie false si le membre est déjà présent ou s’il n’y a plus de place dans le tableau)

ajouterMedia : ajoute un média transmis en argument (renvoie false si le membre est déjà présent ou s’il n’y a plus de place dans le tableau)

retirerMembre : retire le membre transmis en argument de la liste des membres (renvoie false si le membre n’est pas présent). Il ne peut pas y avoir d’emplacement libre dans le tableau entre deux membres.

toString : renvoie le contenu des deux listes (utiliser infoMin pour les membres)

**Classe Membre** :

*Attributs* : nom (String), prenom (String), numeroMembre (int), listeLocations (tableau de maximum 5 locations)

*Méthodes* :

ajouterLocation : effectue la location d’un média à partir d’une date pour un certain nombre de jours

infoMin : retourne uniquement le nom, le prénom et le numéro du membre

equals : deux membres sont identiques s’ils ont le même nom et le même prénom

Pour les locations de 7 jours ou plus, une réduction de 20% du prix est octroyée.

NB : **d’autres méthodes** que celles mentionnées ci-dessus peuvent s’avérer **nécessaires** à la réalisation du programme.

Veillez à respecter les **principes de réutilisation du code et d’encapsulation**. Ajoutez des getters et setters si nécessaire.