

TP 6 listes (ArrayList, LinkedList)

Exercice 1 : Décodage de messages

Faire un programme qui lit un texte (comprenant des mots et des entiers) mis dans une chaîne de caractères.

Le message à décoder se trouve dans le fichier `message.txt` mis sur ENT. Vous le copier coller dans la chaîne de caractères que vous créez.

Dans le but de déchiffrer le message codé de cette chaîne de caractères, le programme doit :

- au fur et à mesure de la lecture de la chaîne de caractères :
 - mettre les nombres entiers dans une liste (ArrayList) d'entiers
 - et mettre les mots dans une liste (ArrayList) de mots.
- puis afficher le message en affichant les mots d'indice les entiers contenus dans la liste de nombres.

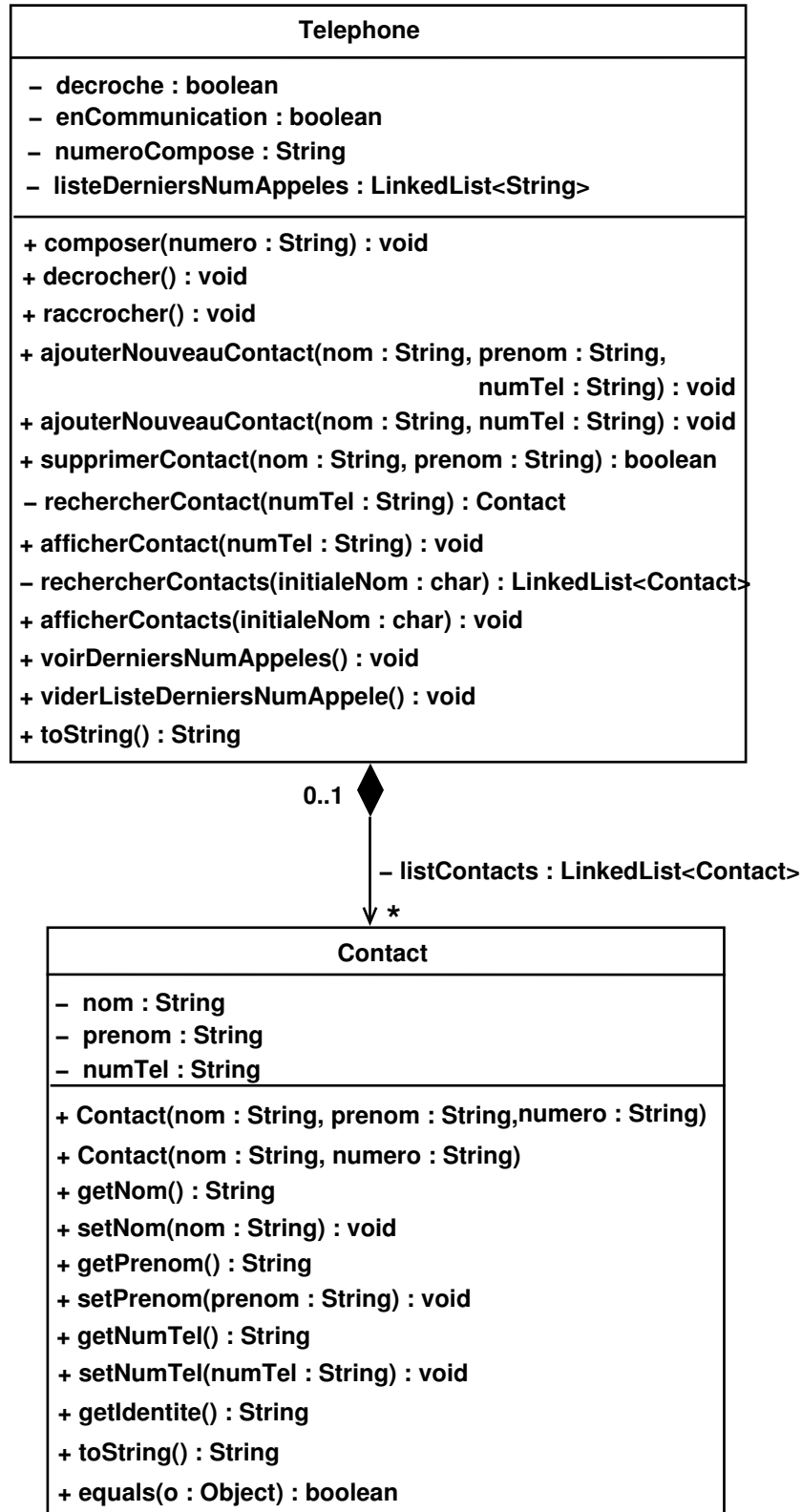
Indications : Il est possible de lire les mots et les nombres qui composent une chaîne de caractères. Pour cela on va utiliser la classe `Scanner` et ses méthodes :

- `Scanner(laChaineAscanner : String)` : constructeur d'un `Scanner` qui va permettre de lire la chaîne
- `hasNext() : boolean` : qui dit s'il reste quelque chose à lire dans la chaîne
- `hasNextInt() : boolean` : qui dit s'il reste un nombre à lire dans la chaîne
- `nextInt() : int` : qui lit et retourne le nombre lu
- `next() : String` : qui lit et retourne le prochain « mot » (de la tête de lecture au prochain espace ou à la fin de la chaîne)
- `close() :` fermeture du scanner.

Quel est le message caché ?

Exercice 2 : Le Téléphone

Codez les classes `Contact` et `Telephone` comme indiqué ci-dessous :



Précisions :

- dans la classe `Contact` :
 - le nom ou le numéro de téléphone d'une personne ne peut être vide. Sinon déclenchement d'exceptions.
- la méthode `getIdentite() : String` retourne la chaîne formée par le nom suivi du prénom si il y a un prénom ou simplement le nom sinon.

- la méthode `toString():String` retourne une chaîne de la forme :
Contact : Durand Pierre
0908070605
- dans la classe Telephone :
 - Lorsque l'on compose un numéro de téléphone celui-ci est « enregistré ». Attention, il n'est pas possible de composer un numéro si le téléphone est décroché ;
 - Lorsque l'on décroche on est mis en communication avec le numéro appelé si un numéro a été composé au préalable. Sinon le dernier numéro appelé est proposé comme numéro composé (et si on redécroche ce sera ce numéro avec lequel on sera mis en communication). Dans ce dernier cas, le téléphone est considéré décroché mais pas en communication. Ce qui fait que suite à cette action si l'utilisateur veut composer un nouveau numéro (celui proposé ne lui convenant pas) il faudra tout d'abord qu'il raccroche. Dans le cas où il n'y avait pas de numéro composé et que la liste des derniers numéros appelés est vide, le téléphone n'est pas considéré comme décroché et un message est affiché indiquant "aucun numéro à appeler" ;
 - Lorsque l'on raccroche, la communication est arrêtée et le numéro composé effacé ;
 - Lorsque l'on ajoute un nouveau contact donné par son nom, prénom et numéro de téléphone (ou simplement par son nom et son numéro de téléphone), celui-ci est créé et ajouté à la liste des contacts sauf s'il y est déjà présent. Dans ce dernier cas on lance une Exception. Si le nom ou le numéro de téléphone sont null alors on laisse remonter l'exception lancée par la création du contact.
 - Lorsque l'on veut supprimer un contact donné par son nom et son prénom (son prénom pouvant être null), on cherche ce contact dans la liste des contacts puis on le supprime si on le trouve. Cette méthode retourne un booléen indiquant si la suppression a bien eu lieu ou si le contact n'a pas été trouvé.
 - rechercher un contact par son numéro de téléphone : retourne null si non trouvé ou le contact.
 - afficher le contact correspondant au numéro de téléphone donné c'est afficher l'identité du contact trouvé, ou afficher le numéro de téléphone si non trouvé.
 - rechercher les contacts dont la lettre donnée est l'initiale de leur nom c'est retourner la liste de ces contacts (elle peut être vide)
 - afficher les contacts dont le nom commence par une lettre donnée c'est afficher les contacts de la liste trouvée suite à une recherche appropriée.
 - voir les derniers numéros appelés permet d'afficher les numéros de cette liste ou alors le message "liste vide".
 - vider les derniers numéros appelés permet de vider cette liste.
 - la méthode `toString` retourne la description du téléphone, à savoir : s'il est "décroché", "décroché et en communication avec le 0102030405" (si numéro inconnu), "décroché et en communication avec Dupond Pierre" (si numéro connu) ou "raccroché".