

TD 9 Collections : Map

1 Rappel de Cours

Dans ce TD, nous allons nous entraîner à utiliser les collections de type `Map` c'est à dire les tables de correspondances d'objets qui associent une valeur (un objet) à une clé, avec au plus une valeur par clé (voir la [documentation](#)).

Pour rappel, lors du TP 6-7 vous implémentez ce type de modèles (table d'association format liste `ATList`, et format arbre `ATTree`). Ces implémentations et d'autres existent via des bibliothèques JAVA telles que :

- `HashMap<K, V>` : table de hachage,
- `LinkedHashMap<K, V>` : table de hachage + liste pour l'ordre,
- `TreeMap<K, V>` : arbre.

Toute classe implémentant l'interface `Map` doit définir les méthodes suivantes :

- `V put(K key, V value)` : associe key à value. Si l'association existait déjà, l'ancienne valeur est écrasée dans la Map, et est retournée par la méthode
- `V get(Object key)` : renvoie la valeur associée à la clé key, ou null
- `Set<K> keySet()` : renvoie l'ensemble des clés de la map

2 Exercices

Exercice 1.

On veut réaliser une application d'annuaire téléphonique pour un smartphone. La spécification informelle de cette classe est :

- Elle doit stocker des contacts, chaque contact possède un nom, un numéro de téléphone et un e-mail.
- Quand l'utilisateur tape le nom d'un contact, l'annuaire téléphonique renvoie le contact complet.

Question 1. Via un diagramme de classe (cf. description à base de boîtes), modéliser la classe `Annuaire` ainsi que la classe `Contact` qu'elle utilise.

Vous devez préciser le type de chaque attribut de chaque classe. En particulier, quelle structure de données (= quelle collection) allez-vous utiliser pour stocker les contacts ?

Question 2. Implémentez la classe `Contact` stockant les informations d'un contact.

Question 3. Implémentez la classe `Annuaire` qui stocke la liste des contacts et permet de l'interroger. Vous devez implémenter les fonctions spécifiées ci-après :

- Le constructeur d'un annuaire (initialement l'annuaire est vide) ;
- Une méthode d'ajout d'un contact dont la signature est `void ajouter(Contact c)` ;
- Une méthode de recherche d'un contact dont la signature est `Contact lookupContact(String nom)`.

Votre smartphone, quand vous recevez un appel, affiche le nom du contact associé au numéro de l'appelant.

Question 4. Comment modifieriez-vous `Annuaire` pour qu'elle puisse fournir la fonction :

```
String lookupByNumero(String numTel)
```

Cette fonction renvoie le nom du contact trouvé ou `null` s'il n'est pas trouvé.

Exercice 2.

On vous donne une chaîne de caractères stockée dans une variable `str`. Écrivez une fonction qui pour chaque lettre de l'alphabet présente dans `str` renvoie sa (ses) position(s) dans `str`.

Exemple : Avec `str = "Coucou monde"`; le résultat est :

`c -> [0, 3] ; o -> [1, 4, 8] ; u -> [2, 5] ; m -> [7] ; n -> [9] ; d -> [10] ; e -> [11]`