

COLLECTIONS (SANS LA NOTION DE GÉNÉRIQUE)¹- SUITE

Écrire une classe `MyList` déclarant des méthodes permettant de manipuler une liste d'objets : ajout/suppression d'un élément à la fin de la liste (`add/removeLast`), ajout/suppression d'un élément en tête de liste (`add/removeFirst`), renvoi du premier/dernier élément (`getFirst/Last`), recherche d'un élément (`contains`), suppression de tous les éléments de la liste (`clear`), représentation de la liste sous forme de chaîne de caractères (`toString`), taille de la liste (`size`) et test de liste vide (`isEmpty`). Pour cela, on va utiliser un chaînage simple au moyen d'une paire (`Cell`) constituée d'un élément (appelé `data`) et d'une référence vers la paire suivante (appelée `next`). Utilisez le squelette de la classe `Cell` suivant :

```
class Cell {
    private Object data;
    private Cell next;

    public Cell () {...}
    public Cell (Object data){...}
    public Cell (Object data, Cell next){...}
    public Object getData(){...}
    public Cell getNext(){...}
    public void setData(Object data){...}
    public void setNext(Cell next){...}
}
```

La classe `MyList` devra implémenter l'interface `Iterable` (ce qui veut dire que notre classe contient un ensemble d'éléments que l'on peut parcourir). Vous devez déclarer et implémenter une unique méthode, `iterator()`, qui renvoie un `Iterator` sur l'ensemble des éléments. Pour cela, il vous faudra définir votre propre classe `MyIterator` qui implémente l'interface `Iterator`, de la manière suivante.

```
class MyIterator implements Iterator{
    private Cell current;

    public MyIterator (Cell first){...}
    public boolean hasNext() {...}
    public Object next() {...}
    public void remove() {
        throw new UnsupportedOperationException();
    }
}
```

Écrivez une classe `Main` contenant une méthode `main` pour tester les méthodes de la classe `MyList`.

NOTION DE PAQUETAGE.

Modifiez les fichiers sources des classes `MyList`, `Cell` et `MyIterator` pour qu'elles se trouvent dans un même paquetage, `myList`. Laissez la classe `Main` en dehors du paquetage `myList`.

- Que se passe-t-il si vous essayez de compiler le fichier `Main`? Corrigez le code si vous considérez que cela est nécessaire.

1. En TP, vous pouvez choisir d'utiliser la version 1.4 de Java (`Project/Properties/Java Compiler`)

- Modifiez les fichiers sources des classes se trouvant dans le paquetage `myList` pour qu'il n'y ait qu'une seule classe publique, en occurrence la classe `MyList` (i.e. le mot clé `public` devant la déclaration de la classe). Commentez les effets d'une telle configuration.