

# ArrayList

## Nombres aléatoires

Créez une liste de 30 nombres entiers aléatoires différents compris entre 1 et 100.  
Créez ensuite une deuxième liste contenant également 30 entiers aléatoires différents compris entre 1 et 100. Les nombres de la deuxième liste seront également différents de ceux de la première liste.

Triez les listes et affichez-les à l'aide de la méthode toString().

## Opérations ensemblistes sur les listes.

Créez deux arrayList {"George", "Jim", "John", "Blake", "Kevin", "Michael"} et {"George", "Katie", "Kevin", "Michelle", "Ryan"}

Trouvez leur union, différence et intersection.

Vous pouvez cloner les listes pour préserver les listes originales de toute modification.

Recherchez les méthodes les plus pertinentes dans les différentes classes de l'API afin de réaliser efficacement les opérations demandées.

## Polygone

La classe Polygone contient une suite de Points.

Un polygone peut avoir un nombre de sommets variable.

Utilisez ArrayList pour stocker un ensemble de Points d'un polygone.

Dans la classe Polygone, vous devez implémenter les méthodes suivantes :

- void ajouter(Point p)

Cette méthode ajoute le Point (passé en paramètre) comme un point supplémentaire à la fin de ArrayList de Polygone.

- void enlever(int ind)

Cette méthode enlève le point identifié par l'index passé en paramètre.

- void changer(int ind, int x, int y)

Cette méthode permet de changer les coordonnées du point de l'index ind à (x,y).

- int diagonales(): Calcule le nombre de diagonales fourni par la formule  $\frac{n(n-3)}{2}$  où n représente le nombre de côtés du polygone.

Après chaque modification, vous devez afficher les sommets du polygone.

## Agenda de personnes

Définissez une application qui permette de gérer un agenda où sont répertoriés des personnes avec leur nom, leur adresse et leur numéro de téléphone.

L'application doit proposer un menu qui permette à l'utilisateur de choisir un traitement parmi les suivants : ajout d'une personne (pas de doublon sur les noms), affichage complet, recherche d'une personne pour afficher son adresse et son numéro de téléphone, modification ou suppression d'une personne à partir du nom de cette personne. Définissez une classe `Personne` et une classe `Agenda` structurée par une variable de type `ArrayList<Personne>` (version 5 de Java).

Ajouter l'option « Trier la liste » et utilisez la méthode `Collections.sort(...)`.