



Collections Java: Deviens un As
du Rangement!



Les Ingrédients du Jour:

Les Collections: C'est quoi le délire?

ArrayList: Ton sac à dos pour objets!

Ranger, chercher, trier: Deviens un expert du rangement!

For-Each: La boucle stylée pour les collections!

Inventaire et Personnages: Des exemples concrets pour t'inspirer!

Collections: Le B-A-BA du Rangement

- Collection: Un groupe d'objets, comme une équipe de choc!
- `java.util.Collection`: L'interface de base, la chef d'équipe
- `List<E>`: Une liste ordonnée, comme une file d'attente
- `Set<E>`: Un ensemble sans doublons, comme un club VIP
- `Map<K,V>`: Des associations clé-valeur, comme un dictionnaire

ArrayList: Ton Sac à Dos Java!

- ArrayList: Une liste qui grandit toute seule!
- `List<Personnage> listePerso = new ArrayList<>(); //Un sac à dos pour Personnages`
- `List<Item> listeObjets = new ArrayList<>(); //Un sac à dos pour Objets`
- Avantages: Croissance automatique, accès rapide par index, taille dynamique

ArrayList: Mode d'Emploi (Ranger)

- `ArrayList<Item> sac = new ArrayList<>(); //Crée ton sac`
- `sac.add(new Item(...)); //Ajoute un objet au sac`
- `sac.add(2, autreItem); //Insère un objet à la position 2 (attention, ça décale les autres!)`
- `Inventaire inv = new Inventaire(); inv.getObjets().add(item); //Ajoute un item à l'inventaire d'un joueur`

ArrayList: Mode d'Emploi (Chercher, Supprimer)

- `sac.remove(item); //Supprime un objet (précis)`
- `inv.retirerObjet(item); //Supprime un objet de l'inventaire`
- `Item premier = sac.get(0); //Accède au premier objet (position 0)`
- `boolean contient = sac.contains(itemRecherche); //Vérifie si le sac contient un objet`

ArrayList: Mode d'Emploi (Trier, Vider)

- `sac.size();` //Connaitre le nombre d'objet dans le sac
- `sac.clear();` //Vider le sac
- `sac.toArray(new Item[0]);` //Transformer la sac en tableau (pour les experts)

For-Each: La Boucle Stylée!

- La boucle for-each, c'est comme un 'for' amélioré, plus simple à utiliser avec les collections
- Elle parcourt tous les éléments de la collection, un par un

For-Each: Exemple en Code

- `for (Item it : inv.getObjets()) { //Pour chaque objet dans l'inventaire...`
- `System.out.println(it.getName()); //On affiche le nom de l'objet`
- `}`
- `ArrayList<Personnage> liste = new ArrayList<>(); //Une liste de personnages`
- `for (Personnage p : liste) { //Pour chaque personnage...`
- `p.afficher(); //On utilise sa méthode 'afficher' (héritée!)`
- `System.out.println(p.toJSON()); //On affiche ses infos en JSON`
- `}`

À toi de jouer! - Crée ta Team de Joueurs

- Crée une `ArrayList<Joueur>`
- Ajoute 3 joueurs différents (avec des noms cool!)
- Affiche leurs infos en JSON (pour vérifier que tout est OK)

À toi de jouer! - Inventaire en Folie

- Crée une List<Item>
- Ajoute 5 objets (épée, potion, armure... à toi de choisir!)
- Supprime l'objet qui est à la position 2
- Vérifie si ta liste contient une potion (au cas où tu aurais soif)

À toi de jouer! - Chasse aux Stats!

- Crée une liste de Personnages (Joueurs et Monstres)
- Parcours la liste et affiche seulement ceux qui sont presque morts ($\text{vieCourante} < \text{vieMax}/2$)
- Fais ça avec une boucle for-each et avec une boucle 'normale' (avec un index)

À toi de jouer! - Liste de Noms

- Crée une liste d'Item
- Récupère les noms de chaque objet et mets-les dans une nouvelle liste (de Strings)
- Affiche la liste de noms: La classe!

À toi de jouer! - Le Grand Inventaire

- Crée un Joueur (le héros c'est toi!)
- Récupère son inventaire (pour l'instant il est vide)
- Ajoute 3 objets (potion, épée, armure... ce que tu veux!)
- Calcule le prix total de son inventaire: Le héros est riche!

Besoin d'aide? Ressources Utiles

- Collections Java (Oracle) :
<https://docs.oracle.com/javase/tutorial/collections/>
- Guide List (Vogella) :
<https://www.vogella.com/tutorials/JavaCollections/article.html>
- Exemples ArrayList : <https://www.geeksforgeeks.org/arraylist-in-java/>