

Collections et Streams

Exercice 1 : classe Collections

Une voiture est caractérisée par sa marque et sa couleur.

Construire une liste de voiture.

A l'aide de la classe Collections, afficher le min et le max de la liste de voitures de deux façons différentes :

- Version 1 : La classe voiture implémente l'interface Comparable
- Version 2 : La classe voiture n'implémente pas l'interface Comparable

NB : Une voiture v1 est plus grande qu'une voiture v2 si sa marque m1 est après la marque m2 de la voiture v2 dans l'ordre alphabétique.

Exemple de sortie console :

```
Liste des voitures
Car [brand=Citroën, color=gris métallisé]
Car [brand=Peugeot, color=Rouge]
Car [brand=Mini, color=Jaune]
Car [brand=Dacia, color=Noir]
max V1
Car [brand=Peugeot, color=Rouge]
max V1
Car [brand=Citroën, color=gris métallisé]
max V2
Voiture [brand=Peugeot, color=Rouge]
min V1
Voiture [brand=Citroën, color=gris métallisé]
```

Exercice 2 : forEach

Construire une liste d'animaux et la parcourir à l'aide d'un forEach. Afficher le nom de l'animal et son poids en kg.

Ne pas hésiter à faire la version avec une lambda.

Un animal est caractérisé par son cri, son nom, son poids et sa couleur.

Afficher l'animal qui a le poids le plus élevé. Puis afficher l'animal qui a le poids le plus faible.

Exemple de sortie console

```
Animal [name=Mouton, weight=99, color=brun, cry=bêlement]
Animal [name=Chat, weight=4, color=roux, cry=miaulement]
Animal [name=Vache, weight=620, color=beige, cry=beuglement]
Animal [name=Chien, weight=30, color=blanc, cry=abolement]
----- Avec lambda -----
Mouton 99kg
Chat 4kg
Vache 620kg
Chien 30kg
Animal le plus lourd: Animal [name=Vache, weight=620, color=beige, cry=beuglement]
Animal le plus léger: Animal [name=Chat, weight=4, color=roux, cry=miaulement]
```

Exercice 3 : Map

Dans l'entreprise Biomérieux, les employés sont identifiés par un numéro unique sur 6 chiffres commençant par 9.

Ex : 900256

Construire une table associative d'employés dont la clé est ce numéro unique.

On définira un employé avec sa clé, son prénom, son nom et son adresse mail.

Afficher les employés par ordre décroissant du numéro d'identification (*du plus grand au plus petit*)

Parcourir la liste et n'afficher que les employés dont l'adresse mail comporte .fr

Exemple de sortie console

```
Employee [identifiant=987000, lastname=bilel, firstname=gharbi, mail=bilel.gharbi@hotmail.fr]
Employee [identifiant=956780, lastname=odile, firstname=marc, mail=modile@hotmail.fr]
Employee [identifiant=956777, lastname=kaoucar, firstname=gharbi, mail=kaoucar@gmail.com]
Employee [identifiant=956760, lastname=odile, firstname=marc, mail=modile@hotmail.fr]
Employee [identifiant=900877, lastname=safia, firstname=gharbi, mail=safia.gharbi@gmail.com]
-----Employés dont les adresses mail sont en ".fr"-----
Employee [identifiant=987000, lastname=bilel, firstname=gharbi, mail=bilel.gharbi@hotmail.fr]
Employee [identifiant=956780, lastname=odile, firstname=marc, mail=modile@hotmail.fr]
Employee [identifiant=956760, lastname=odile, firstname=marc, mail=modile@hotmail.fr]
```

Exercice 4 : Stream

Soit employees une Collection d'Employés.

Un employé est caractérisé par son nom, son prénom, son genre (*F ou H*) et sa taille en mètres.

A l'aide d'un Stream et des méthodes qui en découlent (*filter, mapToDouble, average...*) calculer la moyenne de la taille des personnes de sexe masculin. Faire de même pour les personnes de sexe féminin.

Exemple de sortie console

Liste des employés

```
Employee [lastname=Gharbi, firstname=Safia, gender=F, size=1.72]
Employee [lastname=LeBron, firstname=James, gender=H, size=2.06]
Employee [lastname=Elqotbi, firstname=Mehdi, gender=H, size=1.92]
Employee [lastname=Gharbi, firstname=Bilel, gender=H, size=1.85]
Employee [lastname=Gharbi, firstname=Aïcha, gender=F, size=1.73]
Taille moyenne des hommes: 1.9433333333333334 mètres
Taille moyenne des femmes: 1.725 mètres
```

Exercice 5 : ToArray et collect

Considérons une collection d'animaux. Un animal est caractérisé par son nom, sa couleur, son cri et son poids en kg.

- Afficher le nom des animaux qui commence par la lettre 'c', trié par ordre alphabétique
- Récupérer dans un tableau de chaînes de caractères le nom des animaux dont le poids est inférieur à 30 kg
- Récupérer dans une chaîne de caractères séparés par un | le nom des animaux dont la couleur commence par b

Exemple de sortie console

Liste des animaux

```
Animal [name=sheep, color=brown, cry=bleating, weight=100.0]
```

```
Animal [name=cow, color=mahogany, cry=moo, weight=720.0]
```

```
Animal [name=dog, color=black, cry=barking, weight=40.0]
```

```
Animal [name=cat, color=white, cry=miaowing, weight=4.5]
```

```
Animal [name=mouse, color=grey, cry=tickle, weight=0.04]
```

Nom des animaux commençant par la lettre c dans l'ordre alphabétique

```
cat
```

```
cow
```

Nom des animaux avec un poids < 30 kilos

```
[cat, mouse]
```

Nom des animaux dont la couleur commence par la lettre b

```
sheep | dog
```

Exercice 6 : Optional

Considérons une collection de voyages. Un voyage est caractérisé par son nom et une liste d'escales.

Une escale est caractérisée par un nom et un nombre de kilomètres.

A partir de la collection de voyages, récupérer sous forme de liste, pour chaque voyage, le nombre de kilomètres le plus faible parmi toutes ses escales.

Exemple de sortie console

Liste des voyages

```
Trip [name=Algeria, stopovers=[Stopover [name=Marseille, nbKm=300], Stopover [name=Alicante, nbKm=500]]]
```

```
Trip [name=Australia, stopovers=[Stopover [name=London, nbKm=500], Stopover [name=Emirates, nbKm=2000]]]
```

```
Trip [name=India, stopovers=[Stopover [name=Frankfurt, nbKm=800], Stopover [name=Emirates, nbKm=2000], Stopover [name=Pune, nbKm=1000]]]
```

Nombre de kilometre le plus faible parmi toutes les escales pour chaque voyage

```
[300, 500, 800]
```

Exercice 7 : reduce

Considérons une collection de chocolat. Un chocolat est caractérisé par son nom, son type (blanc, noir ou au lait) et par sa marque.

En utilisant la méthode reduce, récupérer dans une chaîne de caractère, le nom des chocolats séparés par un \.

Exemple de sortie console

Liste des chocolats

```
Chocolate [name=Milka Oreo, type=MILK, brand=Milka]
```

```
Chocolate [name=Rocher, type=BLACK, brand=Suchar]
```

```
Chocolate [name=Galak, type=WHITE, brand=Nestlé]
```

```
Chocolate [name=Côte d'or, type=MILK, brand=Côte d'or]
```

```
Chocolate [name=Lindt, type=BLACK, brand=Lindt]
```

```
\Milka Oreo\Rocher\Galak\Côte d'or\Lindt\
```

Exercice 8 : sorted

Implémenter la classe `Document` suivante :

```
# esgi.exo8
v Document
  title : String
  author : String
  size : int
  creationDate : LocalDate
  Document(String, String, int, LocalDate)
  getTitle() : String
  getAuthor() : String
  getSize() : int
  getCreationDate() : LocalDate
  setTitle(String) : void
  setAuthor(String) : void
  setSize(int) : void
  setCreationDate(LocalDate) : void
  toString() : String
```

Un document est caractérisé par un titre, un auteur (*prénom nom*), une taille en Ko et une date de création.

Implémenter une classe `DocumentManager` comme suit :

```
# esgi.exo8
v DocumentManager
  documents : List<Document>
  DocumentManager()
  sortDocumentsByTitle() : List<String>
  sortDocumentsByReverseSize() : List<Document>
  sortDocumentsByAuthor() : List<String>
  getDocuments() : List<Document>
```

La classe `DocumentManager` alimente une liste documents avec minimum 5 documents (*vous êtes libres dans le choix des titres, auteurs...*)

```
# esgi.exo8
v DocumentManager
  documents : List<Document>
  DocumentManager()
  sortDocumentsByTitle() : List<String>
  sortDocumentsByReverseSize() : List<Document>
  sortDocumentsByAuthor() : List<String>
  getDocuments() : List<Document>
```

La méthode `List<String> sortDocumentsByTitle()` permet de récupérer la liste des titres des documents dans l'ordre alphabétique.

La méthode `List<Document> sortDocumentsByReverseSize()` permet de récupérer la liste des documents triés par leur taille (de la plus grande à la plus petite)

La méthode `List<String> sortDocumentsByAuthor()` permet de récupérer le titre/auteur des documents triés par ordre alphabétique des auteurs.

Implémenter une classe `Main` pour tester ces 3 méthodes.

Exemple d'exécution de la classe `Main` :

```
Console × Document.java DocumentManager.java Main.java
<terminated> Main (4) [Java Application] D:\java\jdk-19.0.2\bin\javaw.exe (14 sept. 2023, 11:04:52 - 11:04:53) [pid: 7492]

-----Liste des documents sans tri-----
Document [title=CV Product Owner, author=Safia Gharbi, size=150, creationDate=2023-09-13]
Document [title=Spécifications techniques projet MyRecrutement, author=Ibrahim Gharbi, size=233, creationDate=2022-10-05]
Document [title=MSPR Base de données, author=Claire potiche, size=250, creationDate=2021-01-05]
Document [title=Spécifications fonctionnelles projet APICE, author=Dora l'exploratrice, size=99, creationDate=1999-03-08]
Document [title=Inscription à l'activité Théâtre, author=Marie Oury, size=120, creationDate=2005-03-22]
-----Liste des titres des documents dans l'ordre alphabétique-----
CV Product Owner
Inscription à l'activité Théâtre
MSPR Base de données
Spécifications fonctionnelles projet APICE
Spécifications techniques projet MyRecrutement
-----Liste des documents triés de la plus grande taille à la plus petite taille-----
Document [title=MSPR Base de données, author=Claire potiche, size=250, creationDate=2021-01-05]
Document [title=Spécifications techniques projet MyRecrutement, author=Ibrahim Gharbi, size=233, creationDate=2022-10-05]
Document [title=CV Product Owner, author=Safia Gharbi, size=150, creationDate=2023-09-13]
Document [title=Inscription à l'activité Théâtre, author=Marie Oury, size=120, creationDate=2005-03-22]
Document [title=Spécifications fonctionnelles projet APICE, author=Dora l'exploratrice, size=99, creationDate=1999-03-08]
-----Liste des couples titre/auteur des documents triés par ordre alphabétique des auteurs-----
MSPR Base de données/Claire potiche
Spécifications fonctionnelles projet APICE/Dora l'exploratrice
Spécifications techniques projet MyRecrutement/Ibrahim Gharbi
Inscription à l'activité Théâtre/Marie Oury
CV Product Owner/Safia Gharbi
```