

Ejercicio Agregaciones y FP

Preguntas: conteste en Java8 y Kotlin donde corresponda usando estilo FP y operaciones de agregación

Ejercitamos facilidades para operaciones de agregación (alias *collectors*) usando FP en Java8 y Kotlin.

Considere y estudie el código en directorios `work/src/java`. En `data/persons.json` está un archivo ficticio de personas que incluye nombre (de pila y apellidos), edad y un identificador único para cada persona. Ver `java/src/ReadPersons.java` para detalles sobre su lectura. El resto son clases para leer ese archivo usando [GSON](#); algo que no es tan relevante al problema de esta práctica pero le da ejemplos de lectura/escritura usando FP que puede serle útil revisar.

Parte I (Java)

Estudie este link de [Java8 Collectors](#).

Para sus respuestas en Java use como punto de partida la clase `Classifier.java`.

Compilación y corrida (sin que funcione aun lo que se pide más adelante). La salida fue truncada

```
PP: javac -d classes -cp lib\* src\java\*.java
PP: java -cp classes;lib\* eif400.files.ReadPersons | more +5
{"name":{"first":"Mateo","middle":"Adrian","last":"Fernandez","veryLast":"Navarro"},"age":21,"_id":"6"}
{"name":{"first":"David","middle":"Adrian","last":"Martin","veryLast":"Moreno"},"age":22,"_id":"7"}
{"name":{"first":"Martin","middle":"Adrian","last":"Diaz","veryLast":"Blanco"},"age":22,"_id":"8"}
{"name":{"first":"Mateo","middle":"Adrian","last":"Gomez","veryLast":"Dominguez"},"age":24,"_id":"9"}
{"name":{"first":"Tomas","middle":"Adrian","last":"Perez","veryLast":"Rubio"},"age":23,"_id":"10"}
{"name":{"first":"Nicolas","middle":"Adrian","last":"Martinez","veryLast":"Ferrer"},"age":20,"_id":"11"}
{"name":{"first":"Jeronimo","middle":"Adrian","last":"Rodriguez","veryLast":"Sanz"},"age":20,"_id":"12"}
{"name":{"first":"David","middle":"Adrian","last":"Ruiz","veryLast":"Ferrer"},"age":22,"_id":"13"}
{"name":{"first":"Jeronimo","middle":"Adrian","last":"Fernandez","veryLast":"Medina"},"age":22,"_id":"14"}
```

Preguntas

- 1) Calcule la edad promedio de las personas
- 2) Llamemos una familia a un grupo de personas que tienen los mismos apellidos en el mismo orden.
 - a. Imprima el número de familia distintas
 - b. Imprima los apellidos de cada familia y el número de miembros

Parte II (Kotlin)

Estudie estos links de [GroupingBy](#) y de [Grouping](#).

Preguntas

Cree `src/kotlin/classifier.kt` un equivalente de su `Classifier.java` anterior y resuelva las mismas preguntas anteriores en Kotlin