

Guia #1

Programación Funcional

En base al código proporcionado junto a este trabajo práctico, realizar el filtrado de los datos solicitados utilizando las expresiones lambda aprendidas y la API Streams.

1. Filtrar números pares

- Dada una lista de números enteros, utiliza `filter` para obtener solo los números pares y guárdalos en una nueva lista.

2. Transformar una lista de nombres a mayúsculas

- Usa `map` para convertir cada nombre de una lista en su versión en mayúsculas.

3. Ordenar una lista de números

- Usa `sorted` para ordenar una lista de números enteros de menor a mayor.

4. Contar elementos mayores a un valor dado

- Dada una lista de números, usa `filter` y `count` para contar cuántos valores son mayores que 7.

5. Obtener los primeros 5 elementos de una lista

- Usa `limit` para extraer solo los primeros 5 elementos de una lista de números.

6. Convertir una lista de palabras en su longitud

- Usa **map** para transformar una lista de palabras en una lista con la longitud de cada palabra.

7. Concatenar nombres con una separación

- Dada una lista de nombres, usa **reduce** para concatenarlos en un solo String separados por comas.

8. Eliminar duplicados de una lista

- Usa **distinct** para remover duplicados de una lista de números enteros.

9. Obtener los 3 números más grandes de una lista

- Usa **sorted** y **limit** para encontrar los 3 números más grandes en una lista de enteros.

10. Agrupar palabras por su longitud

- Usa **Collectors.groupingBy** para agrupar una lista de palabras según su cantidad de caracteres.

11. Encontrar el producto de todos los números de una lista

- Usa **reduce** para calcular el producto de todos los números de una lista.

12. Obtener el nombre más largo de una lista

- Usa **reduce** para encontrar el nombre con más caracteres en una lista de nombres.

13. Convertir una lista de enteros en una cadena separada por guiones

- Usa `map` y `Collectors.joining` para convertir una lista de enteros en una cadena con valores separados por `-`.

14. **Agrupar una lista de números en pares e impares**

- Usa `Collectors.partitioningBy` para separar los números de una lista en pares e impares.

15. **Obtener la suma de los cuadrados de los números impares**

- Usa `filter`, `map` y `reduce` para obtener la suma de los cuadrados de los números impares de una lista.