

Bài 1: Tìm kiếm và phân tích tần suất từ khóa trong file log lớn

Mô tả:

Bạn có một file log rất lớn (`application.log`) chứa nhiều dòng log từ hệ thống. Nhiệm vụ của bạn là tìm ra **tần suất xuất hiện của mỗi từ khóa** trong file này. Bạn cần tối ưu bộ nhớ và thời gian xử lý, vì file log có thể lên tới vài GB.

Yêu cầu:

- Đọc file log lớn dòng theo dòng.
- Tìm tần suất của mỗi từ trong file.
- Ghi kết quả ra một file mới, với mỗi từ và số lần xuất hiện.
- Không load toàn bộ file vào bộ nhớ.

Hướng dẫn:

1. Sử dụng `BufferedReader` để đọc file dòng theo dòng.
 2. Dùng `Map<String, Integer>` để lưu trữ số lần xuất hiện của từng từ.
 3. Sử dụng `split()` để phân tách mỗi dòng thành các từ và tính toán tần suất.
 4. Ghi kết quả ra một file `keyword_frequency.txt`.
-

Bài 2: Đọc và phân tích dữ liệu JSON trong file lớn

Mô tả:

Bạn có một file JSON lớn (khoảng 2-3GB) chứa thông tin của hàng triệu người dùng (`users.json`). Mỗi người dùng có tên, tuổi, địa chỉ và email. Nhiệm vụ của bạn là:

- Tìm tất cả các người dùng có độ tuổi trên 30.
- Ghi các thông tin của những người này vào một file CSV `filtered_users.csv`.

Yêu cầu:

- Đọc và phân tích file JSON lớn mà không nạp toàn bộ vào bộ nhớ.

- Đảm bảo chương trình có thể xử lý hàng triệu đối tượng mà không gặp vấn đề bộ nhớ.
- Sử dụng thư viện JSON như Jackson hoặc Gson để phân tích dữ liệu.

Hướng dẫn:

1. Sử dụng `BufferedReader` kết hợp với thư viện Jackson/Gson để phân tích JSON từng phần.
 2. Lọc người dùng theo độ tuổi.
 3. Ghi kết quả vào file CSV.
-

Bài 3: Phân tích và tạo báo cáo thống kê từ file log

Mô tả:

Bạn có một file log rất lớn (`server.log`) chứa các dòng dữ liệu từ một ứng dụng web (bao gồm các sự kiện như `INFO`, `ERROR`, `WARN`). Nhiệm vụ của bạn là:

- Đọc toàn bộ file log và phân tích dữ liệu theo từng loại sự kiện (`INFO`, `ERROR`, `WARN`).
- Tính số lần mỗi sự kiện xuất hiện.
- Tính thời gian trung bình giữa các sự kiện `INFO` và `ERROR` (dựa trên timestamp).
- Ghi báo cáo thống kê vào file `log_report.txt`.

Yêu cầu:

- Đọc file log lớn theo dòng.
- Phân loại và đếm các sự kiện theo từng mức độ (`INFO`, `ERROR`, `WARN`).
- Tính thời gian trung bình giữa các sự kiện `INFO` và `ERROR`.
- Ghi kết quả vào một file báo cáo.

Hướng dẫn:

1. Sử dụng `BufferedReader` để đọc từng dòng trong file log.

2. Lọc các dòng **INFO**, **ERROR**, **WARN** và tính toán thống kê.
3. Tính thời gian giữa các dòng **INFO** và **ERROR** dựa trên timestamp.
4. Ghi kết quả vào file báo cáo **log_report.txt**.