**Singly Linked List (SLL) – Lab1**

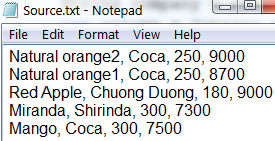
**Mục tiêu**:

1. Luyện tập cách tự xây dựng một SLL giúp quản lý một danh sách đối tượng cụ thể (ở đây là Flower).
2. Ôn tập cách thao tác danh sách với tập tin văn bản
3. Làm quen với một dạng đề thi về SLL

**Cấu trúc bài thực hành**

|  |  |
| --- | --- |
|  | Một danh sách các chai nước ngọt được quản lý bằng một SLL.  Các lớp   * SoftDrink : Class for a soft drink. * SLL\_Node: Class for a node in a SLL. * Sll\_SoftDrink: Class for a SLL of soft drinks. * SLL\_SoftDrink\_Tester: Class for testing   Các file dữ liệu  **Source.txt**: Text file chứa một số soft drink ban đầu  **Results\_text.txt**: Text file chứa kết quả ghi từ chương trình sử dụng lớp PrintWriter  **Results\_bin.txt**: Binary file chứa kết quả ghi từ chương trình sử dụng lớp RandomAccessFile nhưng dữ liệu ghi vào lại là mã ASCII nên file này cũng là filew văn bản |

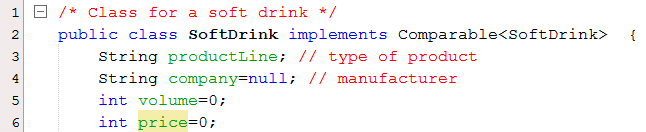
**Nội dung Source.txt** (File này cần tạo trước khi chạy chương trình- nên làm trước khi viết code)

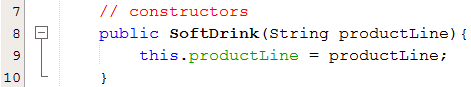


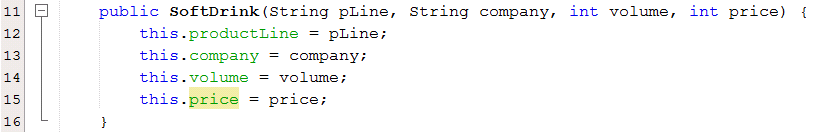
**Source code: Class SoftDrink**

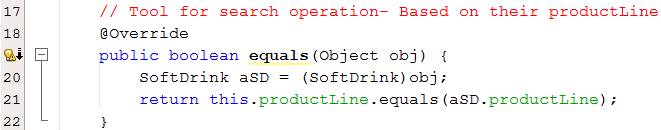
Vì cần cơ chế sắp xếp nên lớp này có implements interface **java.lang.Comparable**

Để có thể tìm một soft drink dựa trên productLine, lớp này cũng ovverride hành vi **equals(…)**

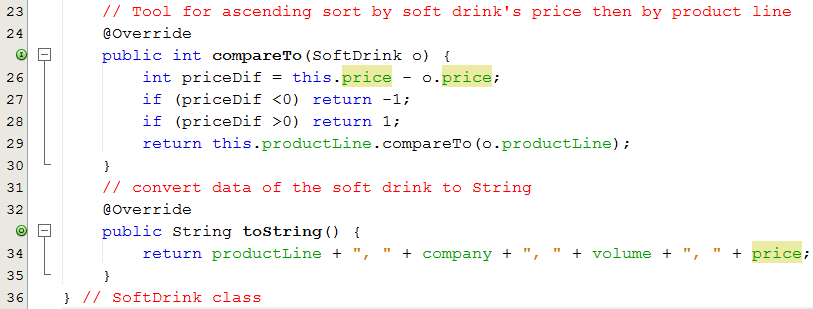




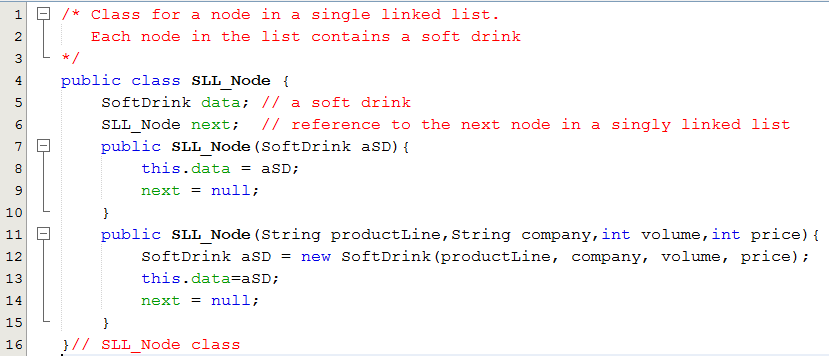




**Hành vi giúp so sánh 2 soft drink để sắp xếp tăng dần về giá, cùng giá thì theo productLine tăng dần**

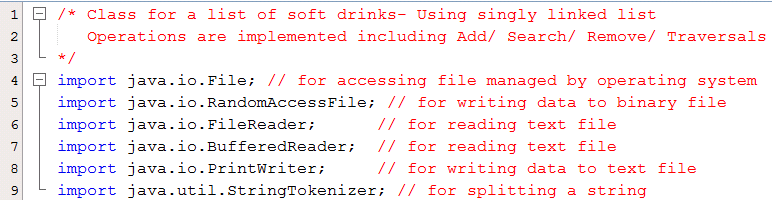


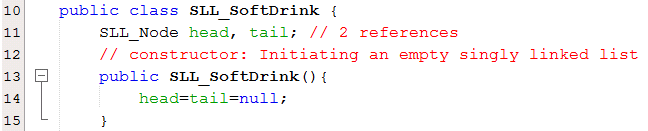
**Source code: Class SLL\_Node** mô tả cho 1 node trong SLL chứa 1 soft drink

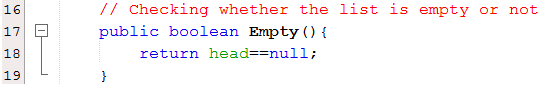


**Source code: Class SLL\_SoftDrink** mô tả cho 1 SLL

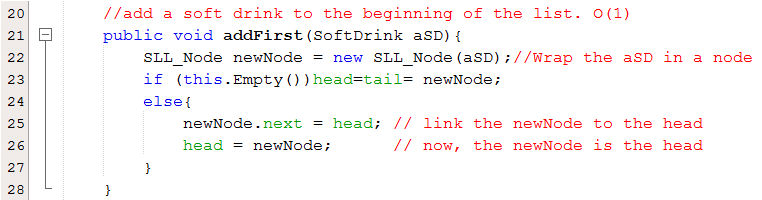
Trong lớp này, việc ghi file dùng cả 2 lớp thư viện, RandomAccessFile và PrintWriter. Việc này là cố tình để sinh viên chú ý khi dùng RandomAccessFile mà data nhị phân lại là mã ASCII thìu kết quả vẫn là file văn bản.

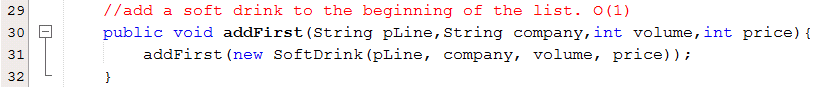




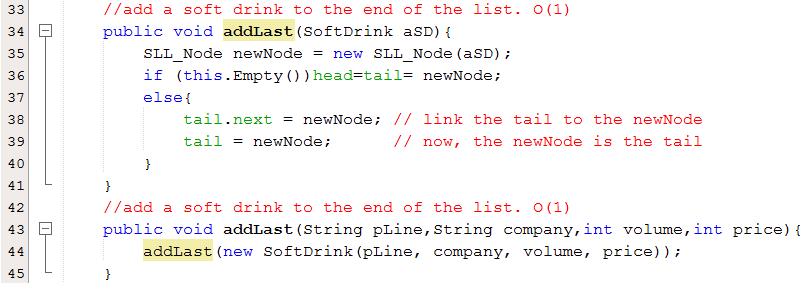


**Hành vi thêm một soft drink vào đầu danh sách**

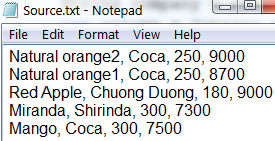




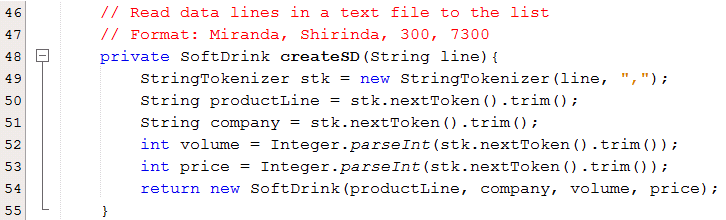
**Hành vi thêm một soft drink vào cuối danh sách**



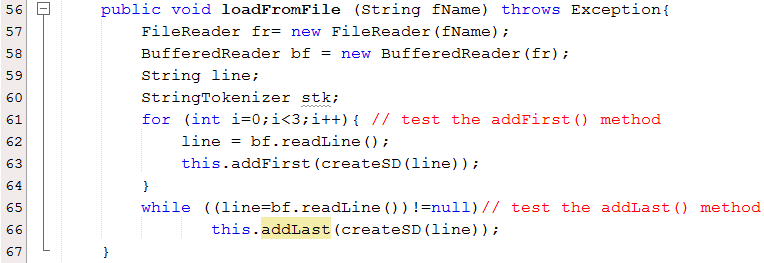
**Các hành vi giúp truy xuất dữ liệu từ file**



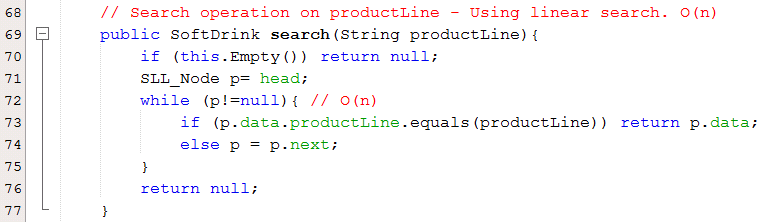
**Hành vi cắt 1 chuỗi dữ liệu – do đọc từ file ra- để tạo một đối tượng SoftDrink**



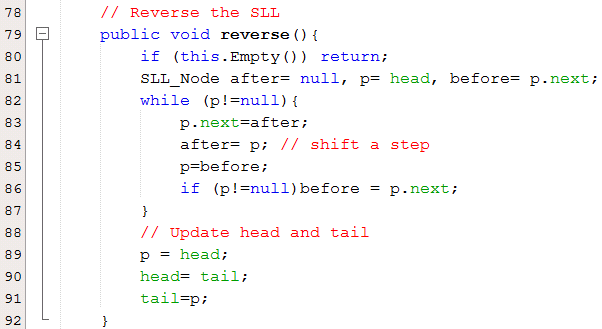
**Hành vi nạp các SoftDrink trong file để đưa vào danh sách. Hành vi này giúp test luôn các hành vi addFirst() và addLast() đã được hiện thực ở trên.**



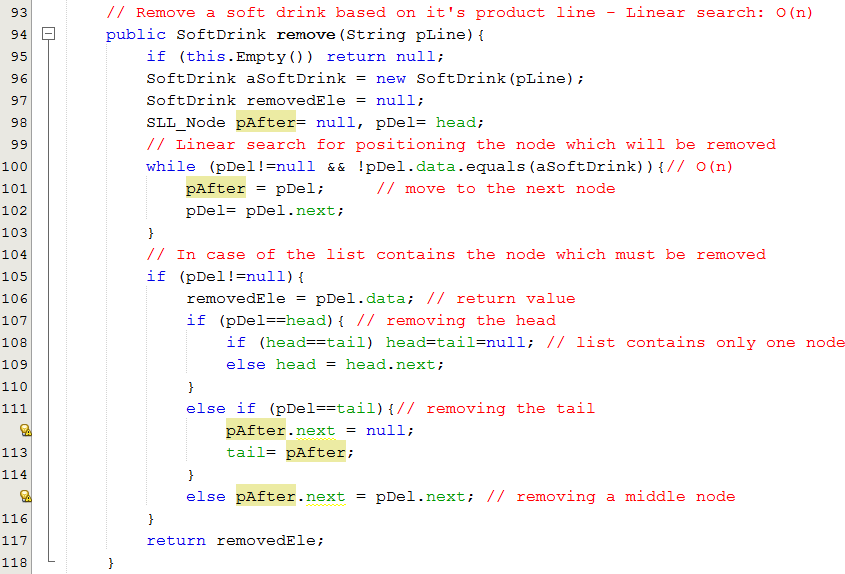
**Hành vi giúp tìm kiếm một SoftDrink dựa trên productLine, Hành vi này dùng hành vi equals(..) để đọ trong quá trình tìm tuyến tính 🡪 O(n).**



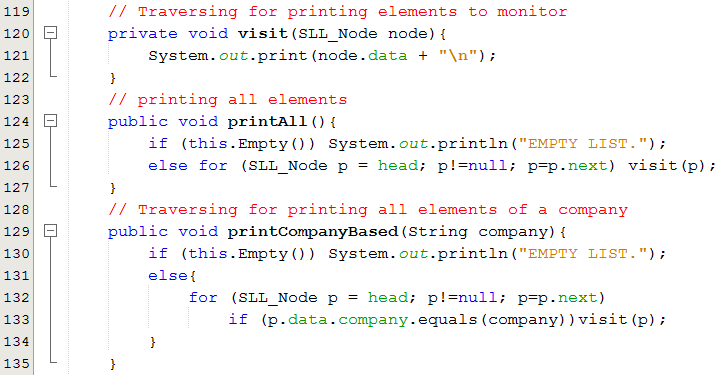
**Hành vi giúp đảo ngược 1 danh sách**



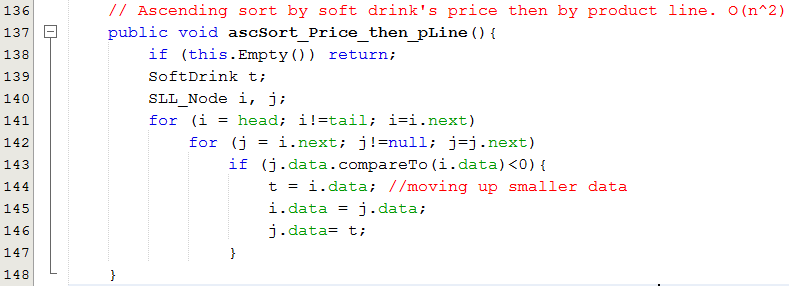
Hành vi giúp xóa một soft drink dựa trên productLine



**Các hành vi giúp duyệt, xuất dữ liệu ra màn hình**



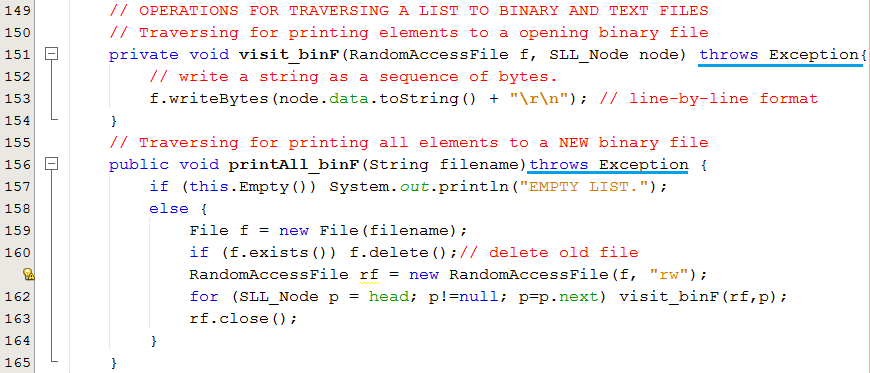
**Hành vi sắp xếp các soft drink theo productline tăng dần**



**Hành vi giúp duyệt xuất dữ liệu lên file sử dụng lớp thư viện RandomAccessFile.**

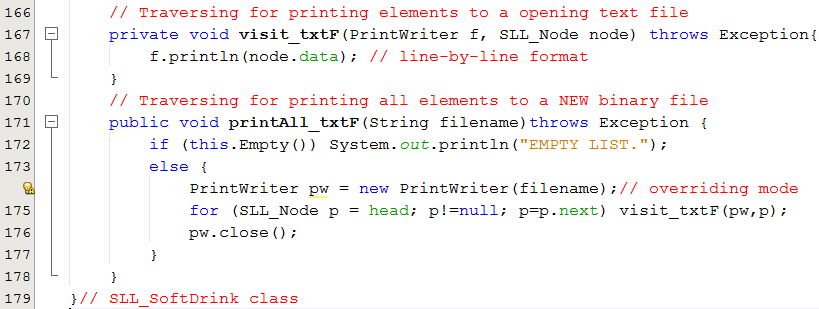
Hành vi writeBytes(String) sẽ chuyển đổi một string thành mảng các byte (mã ASCII) rồi ghi file.

Hành vi này không tự động điền mã ký tự xuống dòng nên muốn ghi có xuống dòng thì cần thêm mã “\r\n”



**Hành vi giúp duyệt xuất dữ liệu lên file sử dụng lớp thư viện PrintWriter.**

Hành vi println(String) sẽ chuyển đổi một string thành mảng các byte (mã ASCII) rồi ghi file có tự động thêm mã New line.



**Source code – main class**

