**Sử Dụng Danh Sách Liên Kết**

**Nội dung**:

* Khi nào dùng danh sách liên kết?
* Nên dùng loại danh sách nào?
* Dùng danh sách liên kết như thế nào trong Java
* Minh hoạ
* Bài tập

1. **Khi nào dùng danh sách liên kết**

* Số phần tử của danh sách không biết trước
* Chức năng thêm/xóa thường được dùng
* Ít/ không có yêu cầu sử dụng chức năng sắp xếp.

1. **Nên dùng loại danh sách nào?**

Có 3 loại danh sách liên kết : DSLK đơn (Singly Linked List), DSLK đôi (Doubly Linked List) và DSLK vòng (Circular List). Dùng loại nào cũng được. Mỗi loại có khác nhau một chút về cách xử lý.

|  |  |  |  |
| --- | --- | --- | --- |
| **Chức năng** | **SLL** | **DLL** | **CL** |
| Thêm vào đầu | Tương đương nhau: O(1) | | |
| Thêm vào cuối | Tương đương nhau: O(1) | | |
| Chèn vào trước 1 phần tử đã biết địa chỉ | O(n) vì phải xác định vị trí đứng trước | O(1) | O(n) vì phải xác định vị trí đứng trước |
| Chèn vào trước 1 phần tử đã biết địa chỉ | Tương đương nhau: O(1) | | |
| Tìm kiếm | Tương đương nhau: O(n) vì tìm tuyến tính | | |
| Xoá phần tử tại địa chỉ đã biết | O(n) vì phải xác định phần tử đứng ngay trước | O(1) vì trong phần tử có chứa tham khảo đến phần tử trước nó. | O(n) vì phải xác định phần tử đứng ngay trước |

Tuỳ thuộc vào các chức năng cần hỗ trợ chúng ta chọn được loại danh sách phù hợp.

**Chú ý về chức năng insert một phần tử vào DSLK đơn theo một tiêu chuẩn**

1. Dựa vào tiêu chuẩn định trước, duyệt DSLK để xác định phần tử đứng trước (before) và phần tử đứng sau phần tử mới sẽ được chèn vào.
2. Chèn phần tử mới vào DSLK

info

info

next

next

**after**

**before**

**newNode.next= after;**

**before.next= newNode;**

info

next

**newNode**

1. **Dùng danh sách liên kết như thế nào trong Java?**

**Bước 1**: Xây dựng class mô tả cho một nút trong danh sách ở mức tổng quát.

**Bước 2**: Xây dựng class mô tả cho một DSLK chứa các nút có dữ liệu trong danh sách ở mức tổng quát cùng các tác vụ cơ bản trên DSLK này ( tham khảo giải thuật cho từng loại DSLK trong tài liệu giáo khoa).

**Bước 3**: Xây dựng class cho dữ liệu của bài toán

**Bước 4**: Xây dựng class mô tả cho DSLK cụ thể của bài toán. Class này là class con của class DSLK tổng quát của bước 2. Ở class này, chúng ta hiện thực thêm các hành vi phù hợp với các chức năng của bài toán.

**Bước 5**: Xây dựng lớo có hàm main để dùng các lớp đã có để thành chương trình.

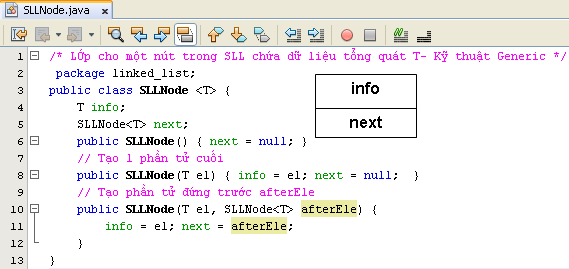
1. **Minh hoạ**

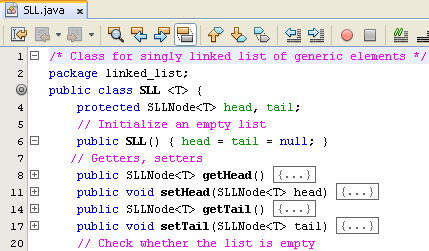
Phát triển một ứng dụng loại Java console dùng DSLK đơn để quản lý một danh sách nhân viên < code, name, salary> các chức năng phải hỗ trợ thông qua một menu đơn giản gồm các chức năng: Thêm nhân viên mới, xoá nhân viên với mã được nhập, tăng lương cho nhân viên với mã được nhập, in danh sách.

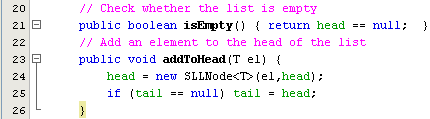
**Kiến trúc các lớp**

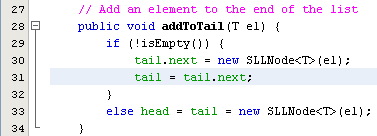
|  |  |
| --- | --- |
|  | Lớp Mô tả cho  **SLLNode** một nút tổng quát trong DS  **SLL**  DSLK tồng quát  **Menu**  menu dạng ứng dụng console  **Employee**  Nhân viên  **LL\_EmpList** DSLK cụ thể, con của SLL  **LL\_Employee\_Program** : chương trình |

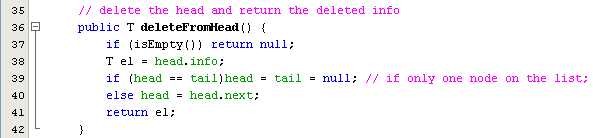
**Code cụ thể:**

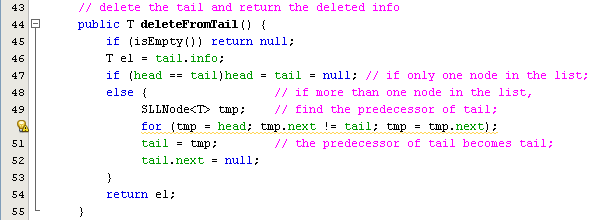


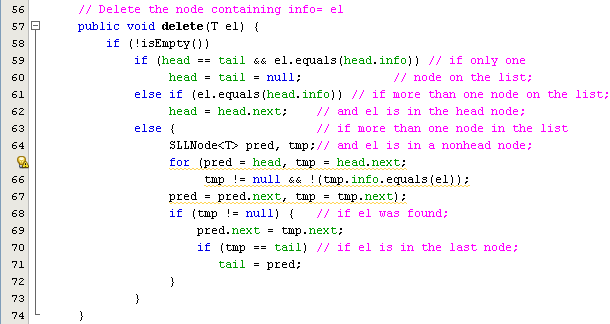


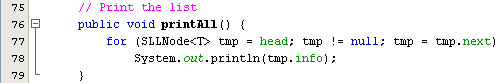


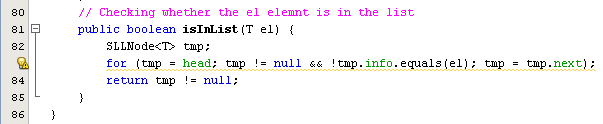


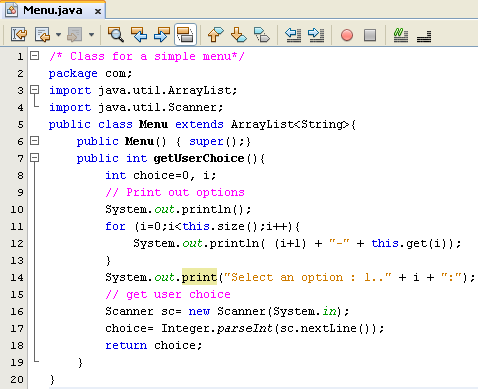


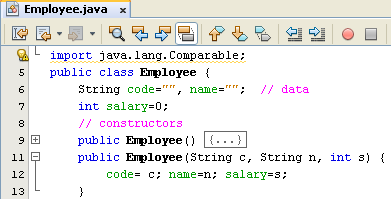


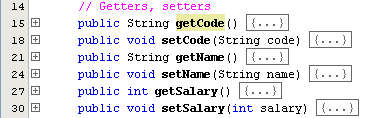


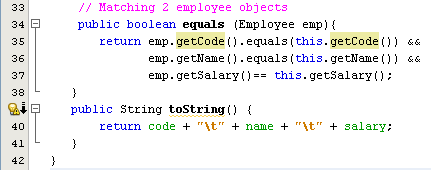




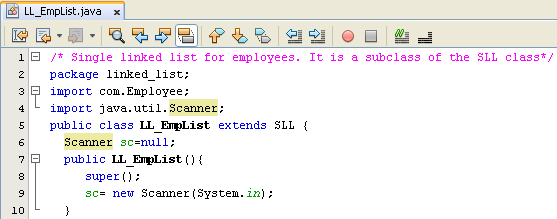


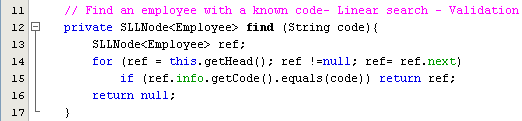


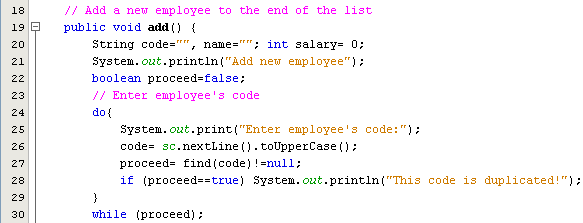


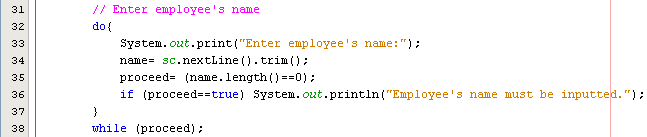


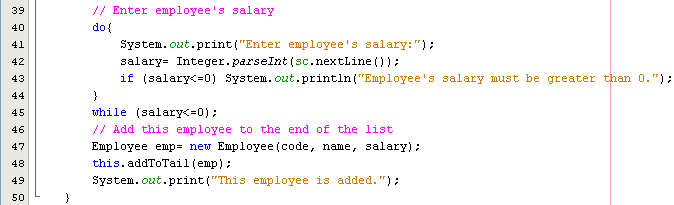
**Class cho DSLK của bài toán**

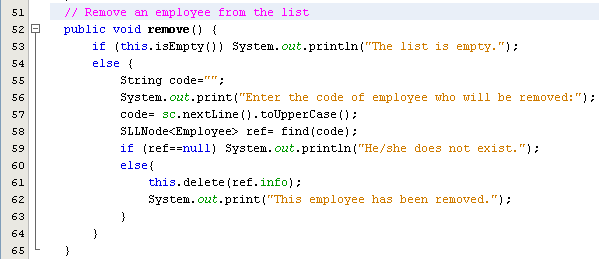


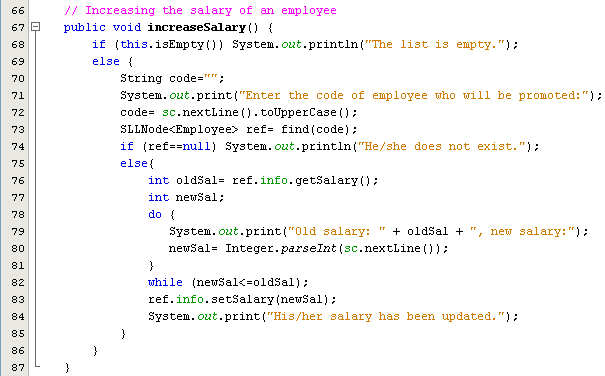


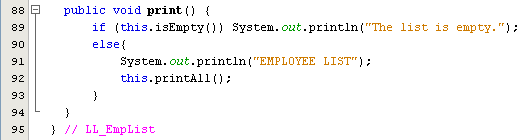




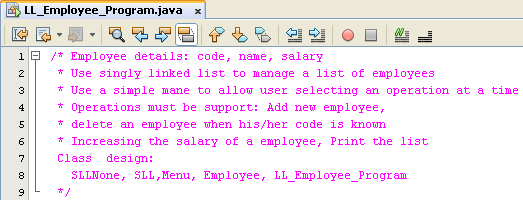


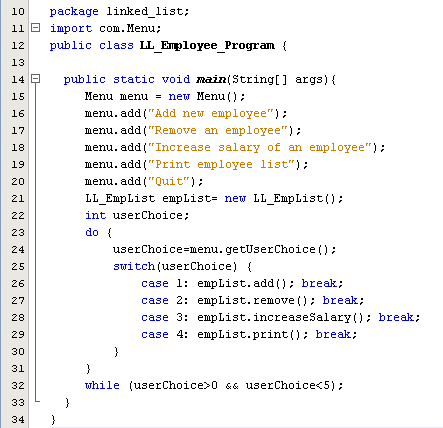




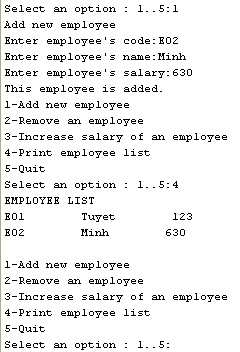
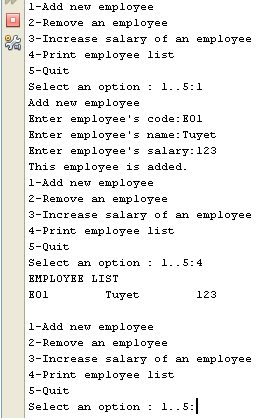


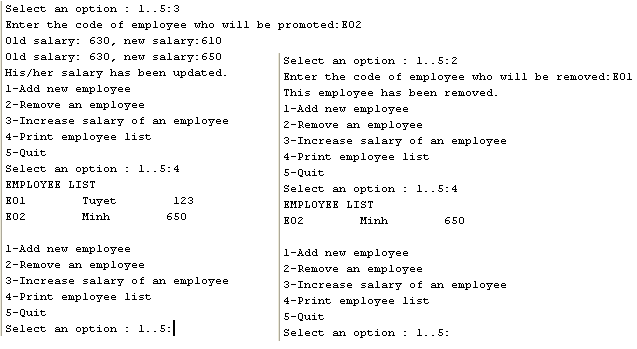
**Class for the problem**





**Một Kết Quả**





1. **Bài tập**
2. Hiện thực lại bài toán này nhưng dùng danh sách liên kết đôi hoặc vòng
3. Tương tự bài mẫu, xây dựng ứng dụng thêm/xoá/sửa giá (cho phép tăng/giảm)/ sửa số tháng bảo hành cho danh sách mặt hàng <code, name, price, guaranty>.