Bìa ở đây

Mục lục

[1. Định nghĩa bài toán 4](#_Toc156052943)

## Lời mở đầu

Trong thời kỳ công nghệ ngày càng phát triển, việc lập trình hướng đối tượng trở thành một phần quan trọng và không thể thiếu trong quá trình phát triển ứng dụng. Bài tập lớn không chỉ là cơ hội để em hiểu sâu về lập trình hướng đối tượng mà còn là dịp để áp dụng kiến thức đã học vào thực tế thông qua việc xây dựng một ứng dụng quản lý thông tin.

Mục tiêu của đề bài là không chỉ kiểm tra khả năng triển khai lập trình hướng đối tượng của sinh viên mà còn đề xuất họ áp dụng kiến thức về giao diện đồ họa, quản lý dữ liệu và xử lý file nhị phân. Điều này giúp sinh viên hiểu rõ hơn về quy trình phát triển ứng dụng và cách tích hợp các khái niệm lập trình hướng đối tượng vào dự án thực tế.

Em mong rằng, qua đề thi này, em sẽ có cơ hội tự tin hơn trong việc áp dụng kiến thức đã học vào thực tế và đồng thời hiểu rõ hơn về quá trình phát triển phần mềm, từ việc thiết kế giao diện đến quản lý dữ liệu và xử lý file.

## Định nghĩa bài toán

* 1. Giới thiệu

Hãy tưởng tượng bạn sở hữu một chuỗi kho hang lớn nhất Việt Nam. Hiện tại tất cả các thông tin của công ty đều quản lý bằng hình thức lưu trữ giấy tờ .Do cách quản lý thủ công khiến cho mất rất nhiều chi phí thuê nhân công đồng thời dữ liệu không được đồng nhất, tìm kiếm thông tin tốn rất nhiều thời gian .Từ các vấn đề đó đặt ra cho bạn cần một ứng dụng quản lý kho hàng thông minh và tiện lợi hơn giúp bạn tối ưu chi phí cũng như thời gian.

* 1. Xác định các đối tượng cần quản lý

Từ các vấn đề nêu trên, đặt ra 3 đối tượng cần quản chính

* + 1. Hàng hóa
* Mã hàng hóa (int) không trùng lặp không âm để xác định các đơn hàng riêng biệt
* Tên hàng hóa (string) ít nhất 1 ký tự và nhiều nhất 10 ký tự
* Thời gian vào kho(string)
* Giá trị hàng hóa (float ) không âm
* Id khách hàng (int) khách hàng
  + 1. Nhân viên
* Mã nhân viên (int) không trùng lặp không âm để xác định các nhân viên riêng biệt
* Tên nhân viên(string) ít nhất 1 ký tự và nhiều nhất 10 ký tự
* Thời gian vào làm (string)
* Lương (float ) không âm
  + 1. Khách hàng
* Mã khách hàng (int) không trùng lặp không âm để xác định cách khách hàng riêng biệt
* Tên khách hàng (string) ít nhất 1 ký tự và nhiều nhất 10 ký tự
* Địa chỉ (string)
* Số điện thoại (string)

1.3 Cách lưu trữ dữ liệu  
 Lưu trữ và quản lý dữ liệu trong ứng dụng là một phần quan trọng, và việc sử dụng các Collection như ArrayList, Map, và LinkedList có thể giúp hiệu quả hóa quá trình này. Trong bài tập lớn lần này, em sử dụng loại map Collection để lưu trữ dữ liệu ánh xạ khóa và giá trị việc thêm, xóa dữ liệu hiệu quả.

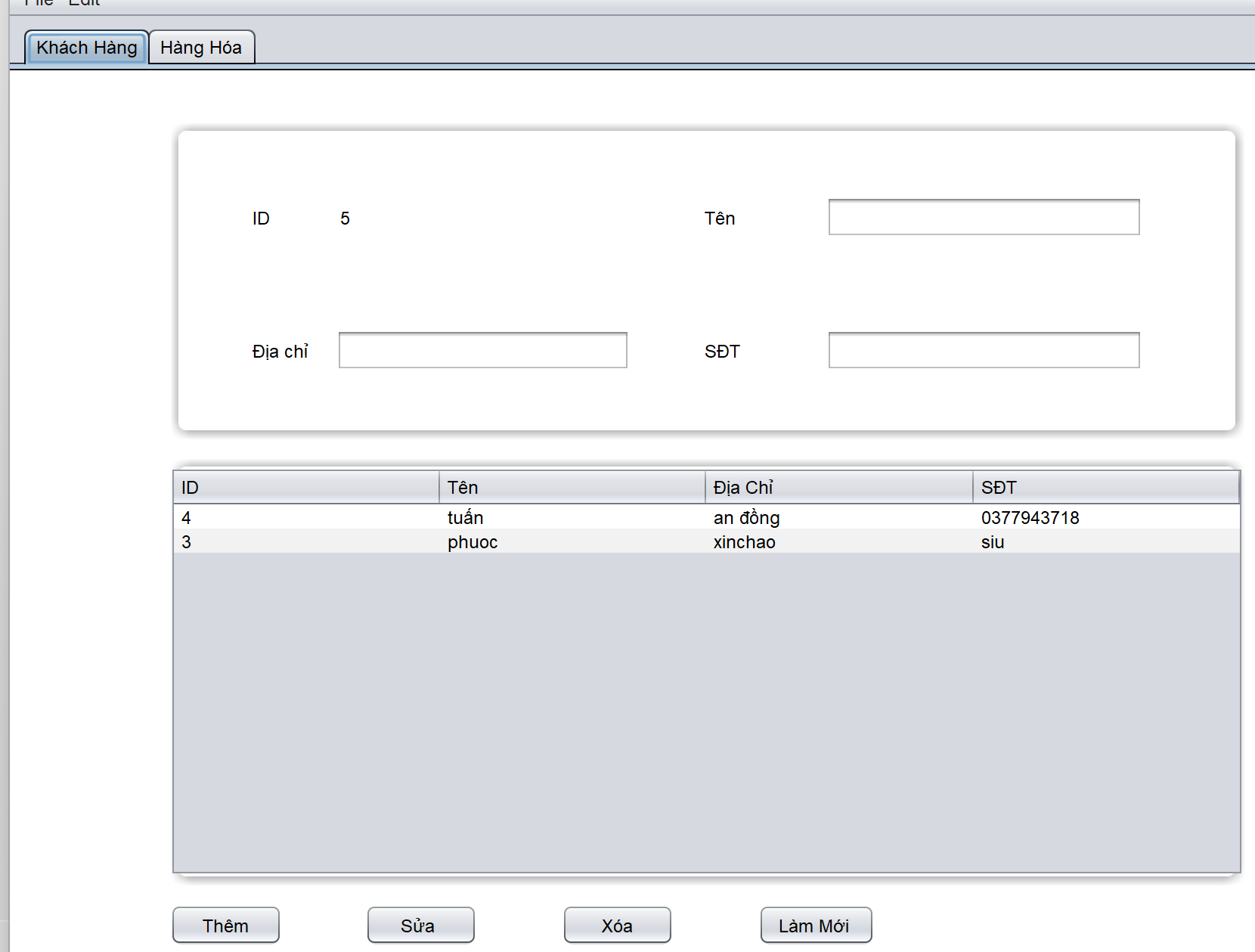
Dữ liệu được ghi xuống file nhị phân. Khi chương trình bắt đầu, chúng ta kiểm tra xem file đã tồn tại hay chưa. Nếu đã tồn tại, chúng ta thực hiện việc đọc dữ liệu từ file và tái tạo các Collection cần thiết. Ngược lại, nếu file chưa tồn tại, chúng ta khởi tạo các Collection mới.

Khi người dùng nhập dữ liệu, chúng ta sử dụng hàm validateInput() để kiểm tra tính hợp lệ của dữ liệu trước khi thêm vào Collection tương ứng. Điều này giúp đảm bảo rằng chỉ những dữ liệu hợp lệ mới được chấp nhận và lưu trữ.

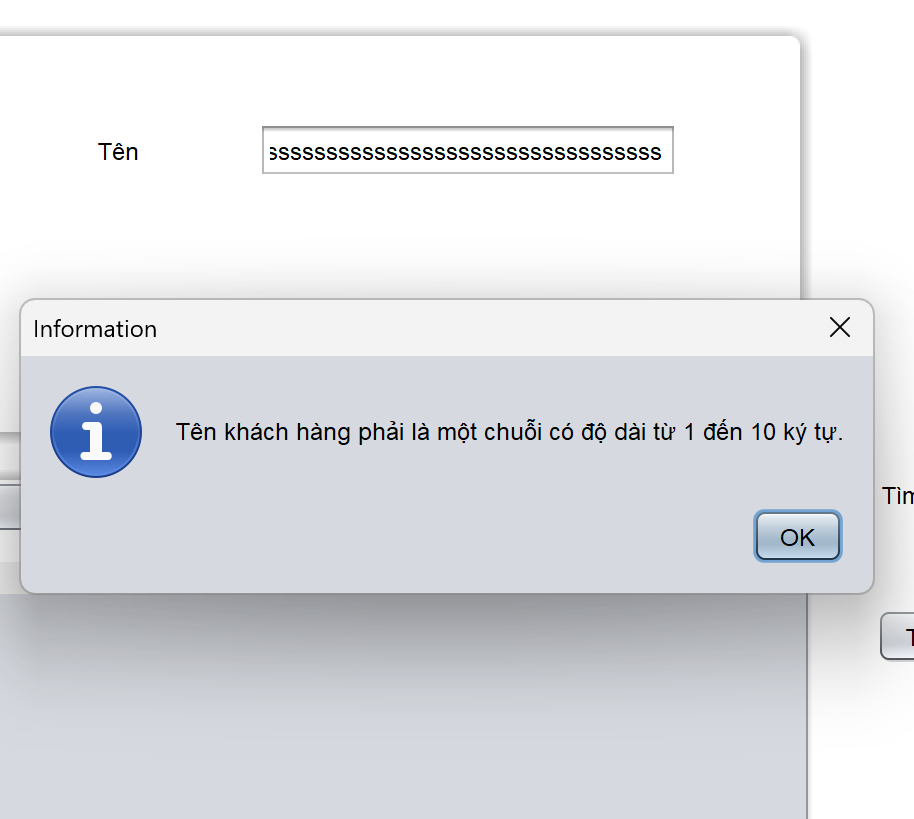
Cuối cùng, sau khi người dùng nhập dữ liệu và chúng ta thực hiện các thay đổi cần thiết trong các Collection, chúng ta lưu trữ lại dữ liệu xuống file để đảm bảo rằng thông tin mới nhất được duy trì giữa các lần chạy ứng dụng.

Tổ chức này giúp chúng ta duy trì một cơ sở dữ liệu nhẹ, linh hoạt và tự động cập nhật trong ứng dụng của chúng ta, đồng thời đảm bảo tính toàn vẹn và tính nhất quán của dữ liệu.

## Các chức năng chính

* 1. Thêm
     1. Thêm Khách Hàng

Id khách sẽ được tự động sinh ra theo id khách lớn nhất + 1.Bên cạnh đó sẽ người dùng sẽ cần nhập vào các trường tên , địa chỉ số điện thoại.Trước khi thêm vào cơ sở dữ liệu ta cần check xem người dùng nhập có thỏa mãn các điều kiện của một khách hàng hay không .Nếu không sẽ hiển thị lên một cửa sổ lỗi



* + 1. Thêm hàng hóa

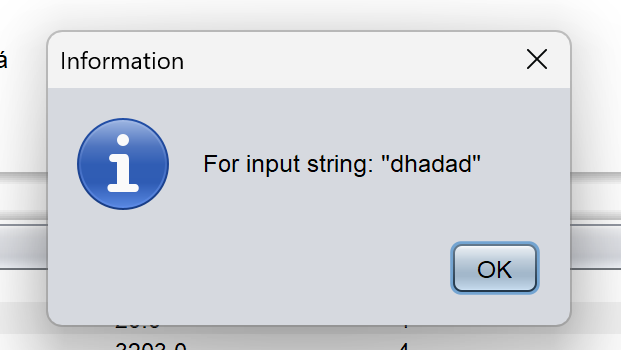
A screenshot of a computer

Description automatically generated

­­­­ Id hàng hóa sẽ được tự động sinh ra theo id hàng lớn nhất + 1.Bên cạnh đó sẽ người dùng sẽ cần nhập vào các trường tên và giá .Thời gian sẽ được lấy thời gian lúc nhập vào .Thời gian và id không thể sửa trong quá trình lưu trữ.

Vì mỗi hàng hóa sẽ thuộc về một khách hàng nên ta chỉ cho khách hàng chọn ID những khách hàng dã tồn tại .

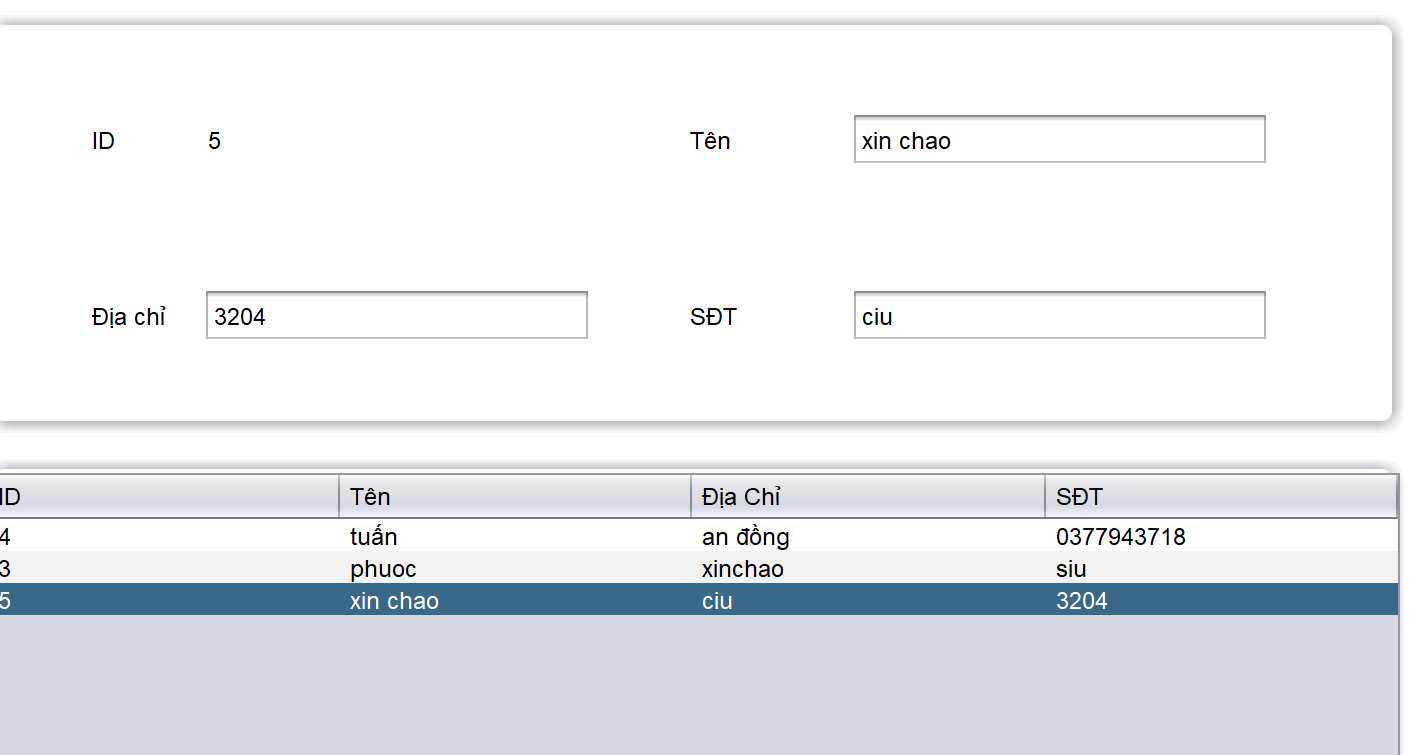
Trước khi thêm vào cơ sở dữ liệu ta cần check xem người dùng nhập có thỏa mãn các điều kiện của một hàng hóa hay không .Nếu không sẽ hiển thị lên một cửa sổ lỗi



* + 1. Thêm nhân viên
  1. Sửa

Trong quá nhập dữ liệu sai số là không thể tránh khỏi . Vậy ta cần chức năng sửa để khắc phục điều đó

* + 1. Sửa khách hàng

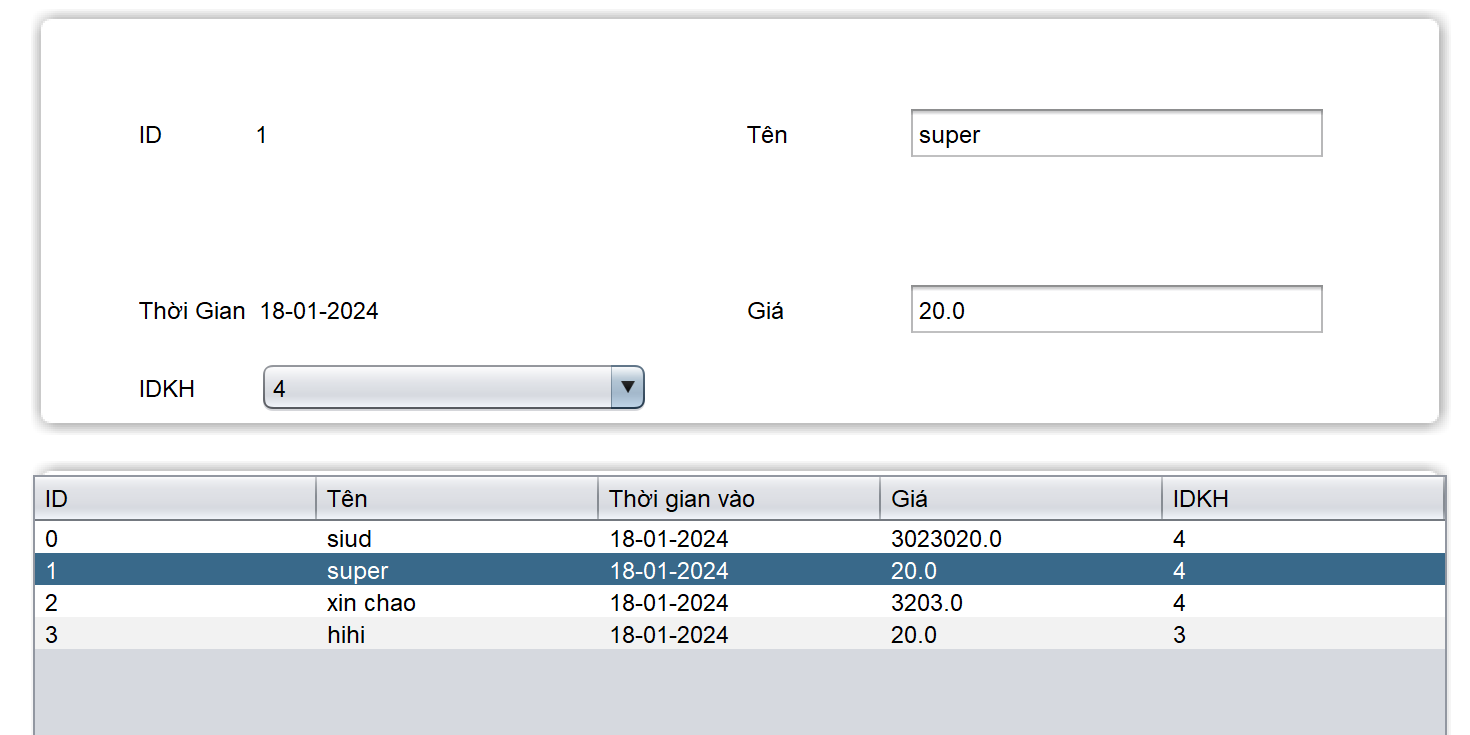
Trước khi sửa khách hàng, cần chọn chọn dòng thay đổi và ấn nút sửa.Tất cả thông tin sẽ hiện lên như sau

Vì ID khách hàng là duy nhất để xác định khách hàng đang sửa nên ID này không thể thay thế

Trước khi thêm vào cơ sở dữ liệu ta cần check xem người dùng nhập có thỏa mãn các điều kiện của một khách hàng hay không .Nếu không sẽ hiển thị lên một cửa sổ lỗi

* + 1. Sửa hàng hóa

Trước khi sửa hàng hóa, cần chọn chọn dòng thay đổi và ấn nút sửa.Tất cả thông tin sẽ hiện lên như sau:



Vì ID khách hàng là duy nhất để xác định khách hàng đang sửa nên ID này không thể thay thế . Thời gian là do máy tự nhập nên không thể có sai số nên cũng không thể thay thế.

Trước khi thêm vào cơ sở dữ liệu ta cần check xem người dùng nhập có thỏa mãn các điều kiện của một khách hàng hay không .Nếu không sẽ hiển thị lên một cửa sổ lỗi

* + 1. Sửa nhân viên
  1. Xóa
     1. Xóa khách hàng
     2. Xóa hàng hóa
     3. Xóa nhân viên
  2. Tìm kiếm
     1. Tìm kiếm khách hàng
     2. Tìm kiếm nhân viên
     3. Tìm kiếm hàng hóa
  3. Thống kê
     1. Thống kê trung bình lượng hàng hóa trên một khách hàng
     2. Thống kê tổng số lượng khách
     3. Thống kê tổng số lượng hàng hóa
     4. Thống kê trung bình giá hàng hóa
     5. Thống kê tổng số lượng nhân viên