BỘ GIÁO DỤC VÀ ĐÀO TẠO

**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT THÀNH PHỐ HỒ CHÍ MINH**

🙠🙟🕮🙝🙢

**Logo

Description automatically generated**

**NHÓM 04:**

KIẾN ĐỨC TRỌNG: 21110332

NGUYỄN TRẦN VĂN TRUNG: 21110335

Đề tài:

**CHƯƠNG TRÌNH**

**QUẢN LÝ QUÁN CÀ PHÊ**

**Môn: Lập trình hướng đối tượng**

**GIÁO VIÊN HƯỚNG DẪN:**

LÊ VĂN VINH

Thành phố Hồ Chí Minh, tháng 12, 2022

**MỤC LỤC**

[**LỜI NÓI ĐẦU** 1](#_Toc122260634)

[**PHẦN NỘI DUNG** 2](#_Toc122260635)

[**1. Các class được tạo ra trong chương trình** 2](#_Toc122260636)

[**1.1 Date** 2](#_Toc122260637)

[**1.2 Time** 2](#_Toc122260638)

[**1.3 Account** 3](#_Toc122260639)

[**1.4 Service** 3](#_Toc122260640)

[**1.5 Order** 4](#_Toc122260641)

[**1.6 Table** 4](#_Toc122260642)

[**1.7 Bill** 4](#_Toc122260643)

[**1.8 Staff** 4](#_Toc122260644)

[**1.9 Admin** 5](#_Toc122260645)

[**1.10 StaffService** 5](#_Toc122260646)

[**1.11 Cafe** 5](#_Toc122260647)

[**2. Các chức năng được thực hiện trong đồ án** 6](#_Toc122260648)

[**2.1 Màn hình login() - đăng nhập** 6](#_Toc122260649)

[**2.2 Phân quyền đăng nhập Admin** 6](#_Toc122260650)

[**2.2.1 Staff Manager** 7](#_Toc122260651)

[**2.2.2 Table Manager** 11](#_Toc122260652)

[**2.2.3 Service Manager** 19](#_Toc122260653)

[**2.2.4 Income Manager** 22](#_Toc122260654)

[**2.3 Phân quyền đăng nhập StaffService** 23](#_Toc122260655)

[**2.3.1 Information** 24](#_Toc122260656)

[**2.3.2 Table Manager** 25](#_Toc122260657)

[**BẢNG PHÂN CÔNG CÔNG VIỆC** 26](#_Toc122260658)

# **LỜI NÓI ĐẦU**

Có thể nói trong thời đại hiện nay, mỗi người chúng ta đều tiếp xúc rất nhiều với khái niệm “cà phê”, nó dường như là thứ thức uống phổ biến với người dân Việt Nam nói riêng cũng như thế giới, việc đó thúc đẩy nhu cầu phát triển của những dịch vụ liên quan ngày càng lớn. Và những quán cà phê mở ra phục vụ nhu cầu đó cũng phát sinh càng nhiều, lúc đó chính là lúc cần đến sự quản lý quán cà phê. Vì vậy, chúng tôi đã lựa chọn đề tài cho đồ án cuối kỳ lần này: “Quản lý quán cà phê”.

Project có tổng cộng là 11 class gồm: Date, Time, Account, Service, Order, Table, Bill, Staff, Admin, StaffService và Cafe. Trong đó có hai class được sử dụng lại từ bài tập ứng dụng operator overloading là Date và Time. Bài đồ án có áp dụng các tính chất hướng đối tượng cơ bản như tính đóng gói, tính kế thừa và đa hình. Ngoài ra còn có các phương thức ảo, interface, operator so sánh, ép kiểu dữ liệu và các hàm nhập – xuất được tinh chỉnh hoàn thiện, dễ nhìn và dễ quản lý, tránh trường hợp xảy ra các bug khi nhập sai dữ liệu…

Các chức năng của chương trình có thể được sơ lược như sau. Có hai loại người dùng trong chương trình là chủ (Admin) và nhân viên (StaffService), thông tin và tài khoản của admin được ghi sẵn trong file text. Khi chạy chương trình cho phép người dùng nhập vào tài khoản và mật khẩu. Nếu là Admin, người dùng sẽ có các chức năng như: thêm, xóa, sửa, sắp xếp và tìm kiếm thông tin của các danh sách được quản lý, cụ thể là quản lý nhân viên (Staff Manager), quản lý dịch vụ (Service Manager), quản lý bàn (Table Manager), đặt dịch vụ và thanh toán. Cuối cùng là quản lý thống kê hóa đơn theo ngày – tháng – năm của quán (Income Manager), có thể xem và sắp xếp nhưng không cho phép xóa và sửa. Còn nếu là nhân viên, người dùng chỉ có thể xem thông tin cá nhân và quản lý bàn, order dịch vụ và thanh toán, có thể tìm kiếm, sắp xếp nhưng không thể xóa và sửa.

# **PHẦN NỘI DUNG**

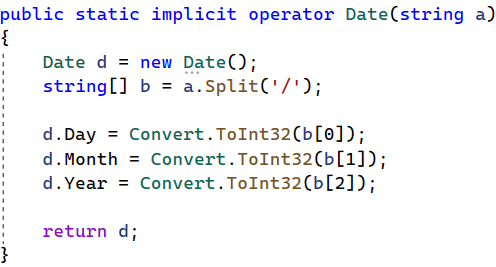
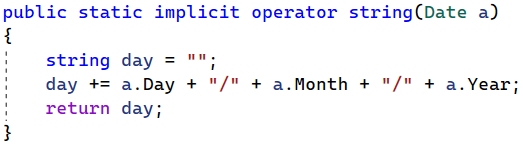
# **1. Các class được tạo ra trong chương trình**

Như đã được biết, một class trong lập trình hướng đối tượng bao gồm các thuộc tính (fields), properties, constructor, destructor, input, output và các phương thức. Áp dụng **tính đóng gói** để đảm bảo dữ liệu được bảo vệ, các thuộc tính trong class đều được khai báo là private, hoặc protected (lớp cha chia kế thừa). Vì vậy mà để lấy dữ liệu thuộc tính xử lý trong lập trình, chúng ta cần các phương thức get, set của properties.

Các class tạo ra trong bài đồ án “Quản lý quán cà phê” lần này bao gồm: Date, Time, Account, Service, Order, Table, Bill, Staff, Admin, StaffService, Cafe.

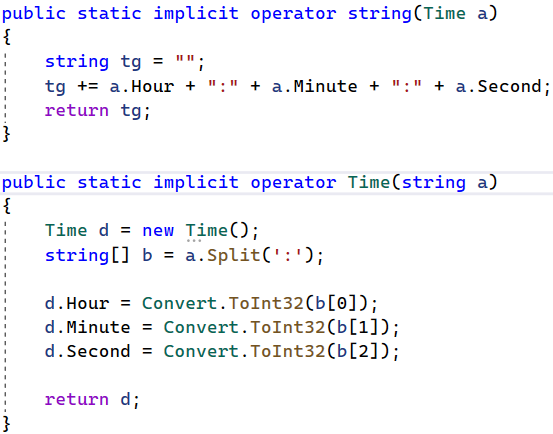
## **1.1 Date**

Class Date có các thuộc tính là iDay, iMonth, iYear, dùng để lưu biến dữ liệu như ngày sinh, ngày xuất hóa đơn.Trong class này, có sự ứng dụng **Operator Overloading** như là so sánh (>, <, =, …) dùng để thực hiện kỹ thuật sắp xếp theo dữ liệu theo ngày. Ngoài ra, ở class này còn có các hàm ép kiểu như ép kiểu Date về string và ép kiểu string về Date.



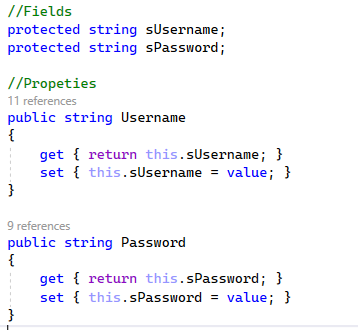
## **1.2 Time**

Ở class này gồm các thuộc tính là iHour, iMinute, iSecond dùng để lưu biến dữ liệu thời gian là giờ. Tương tự như Date, class cũng có những hàm ứng dụng operator overloading so sánh dùng để sắp xếp dữ liệu theo thứ tự ưu tiên, sắp xếp theo ngày tăng dần nếu trùng ngày sẽ sắp xếp theo thời gian. Và class cũng có các hàm ép kiểu string sang Time và ngược lại.



## **1.3 Account**

1 Account gồm Username và Password. Hình dưới đây thể hiện tính **đóng gói** dữ liệu và tương tự cho các class bên dưới. Account dùng để xử lý dữ liệu tài khoản người dùng (**Admin** và **Staff**), phân quyền đăng nhập thông qua class **Staff**.



## **1.4 Service**

Một Service (dịch vụ) gồm ID, Name (tên dịch vụ), Type (loại dịch vụ: đồ uống hay thức ăn), Amount (số lượng tồn kho), Price (giá tiền). Service lưu dữ liệu của dịch vụ, xử lý và quản lý các dịch vụ từ list **Service** của class **Cafe**.

## **1.5 Order**

Một Order (đơn hàng) gồm ServiceName(tên dịch vụ), Amount (số lượng), Price (giá) và Cost (thành tiền) ( = số lượng \* giá). Order dùng để thao tác đặt đơn hàng, lựa chọn các dịch vụ từ list Service (thuộc class Cafe) và lưu vào list **Order** của mỗi bàn.

## **1.6 Table**

Một Table (bàn) gồm ID (mã bàn), Status (Empty - Served) và một danh sách các hóa đơn phục vụ tính tổng tiền cho bàn (list **Order**). Table lưu trữ thông tin của từng bàn, được lựa chọn để đặt đơn hàng cho thực khách và còn có thể thanh toán tổng cộng list **Order** và lưu **Bill** và list **Bill** của **Cafe**.

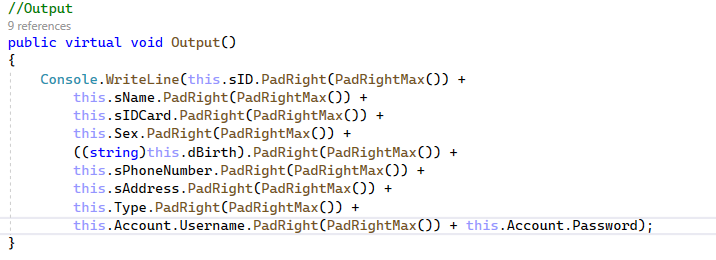
## **1.7 Bill**

Một **Bill** (hóa đơn) gồm mã bill, tên nhân viên, ngày tháng làm hóa đơn, và tổng giá trị của hóa đơn đó. **Bill** dùng để lưu trữ thông tin và thành tiền của mỗi bàn sau thanh toán, mỗi **Bill** sau khi xuất sẽ được lưu vào list **Bill** của class **Cafe** để quản lý.

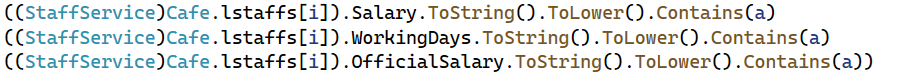
## **1.8 Staff**

Class Staff lưu ID, Name, IDCard, Sex, DateOfBirth, Phone, Address, Type (vai trò), và một Account. Là class cha được kế thừa từ các class con là Admin và StaffService.





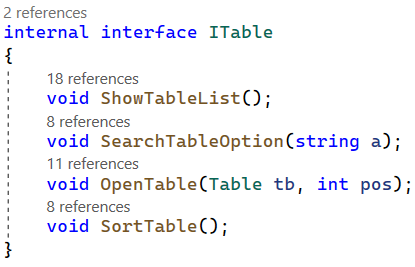
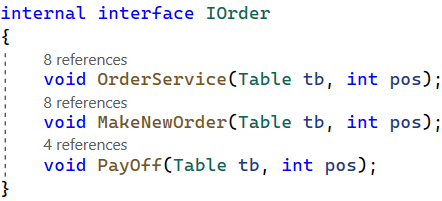
Ở class này chúng ta có đặc biệt ở hàm ảo **virtual** cho phép khi ghi đè (override) ở các class con, ở đây ta áp dụng **tính đa hình.**



Như ở một class con kế thừa **Staff** là class **StaffService**, khi thực hiện ép kiểu ta hoàn toàn có thể sử dụng được một đối tượng **StaffService** mới có toàn bộ các thuộc tính và chức năng của **StaffService** thể hiện rõ **tính đa hình.**

## **1.9 Admin**

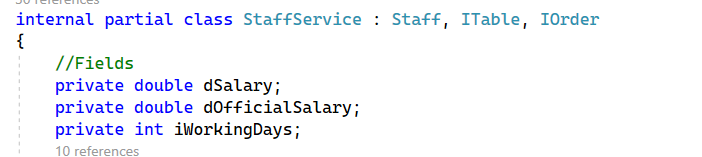
Admin là class kế thừa từ class Staff, áp dụng **tính kế thừa** nhận tất cả thông tin có thể kế thừa từ class cha là Staff và có những hàm riêng lẽ chỉ dành cho riêng nó như các hàm thao tác Login, ShowStaffList, ShowTableList, ShowServiceList, ShowBillList,... Admin là class người dùng, có thể truy cập và xử lý các kỹ thuật quản lý (thêm, xóa, sửa, sắp xếp, tìm kiếm…). Ngoài ra, một số hàm của nó implement **Interface** **Table** và **Order** (Dùng chung giữa StaffService và Admin).





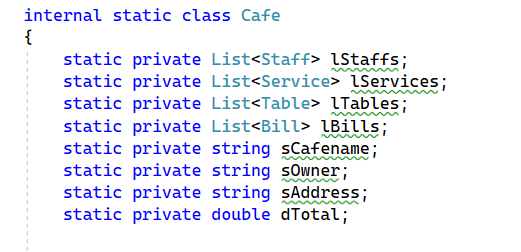
## **1.10 StaffService**

StaffService là class kế thừa từ class Staff, áp dụng **tính kế thừa** nhận tất cả các thuộc tính và chức năng có thể kế thừa từ class cha là Staff và có các hàm thao tác riêng như Login, ShowTableList, hiển thị thông tin của chính nhân viên đó. Quản lý Table list và có các tính năng như tìm kiếm, sắp xếp, và lựa chọn đặt đơn hàng phục vụ, thanh toán các đơn hàng cho từng Table.



Thêm vào đó là các thuộc tính cơ bản riêng của nó là lương cơ bản, lương chính thức và ngày làm việc.

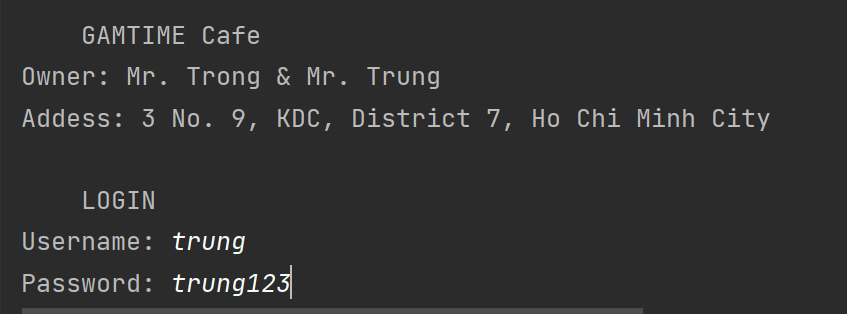
## **1.11 Cafe**



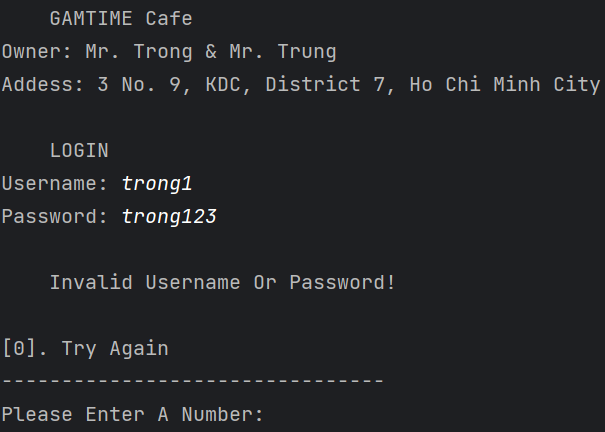
Cafe là **class static**, dùng để lưu giữ danh sách nhân viên, dịch vụ, bàn, hóa đơn phục vụ cho việc truy cập dữ liệu chung của cả project, áp dụng tính đóng gói và truy cập dữ liệu thông qua public của propeties.

# **2. Các chức năng được thực hiện trong đồ án**

## **2.1 Màn hình login() - đăng nhập**

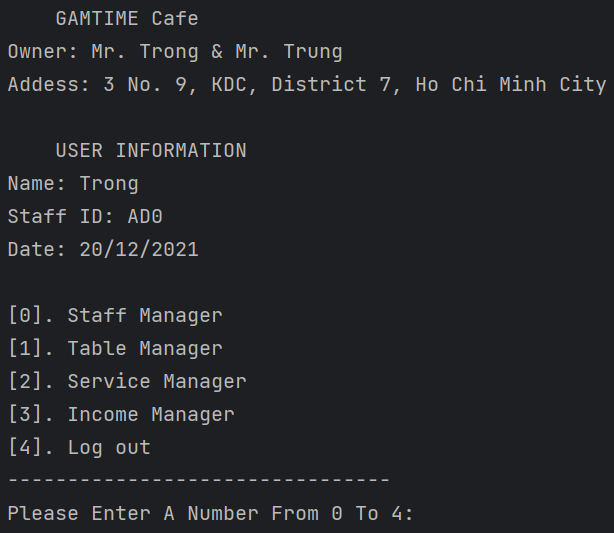
****

Nếu đăng nhập sai tài khoản hoặc mật khẩu, màn hình sẽ thông báo nhập lại.



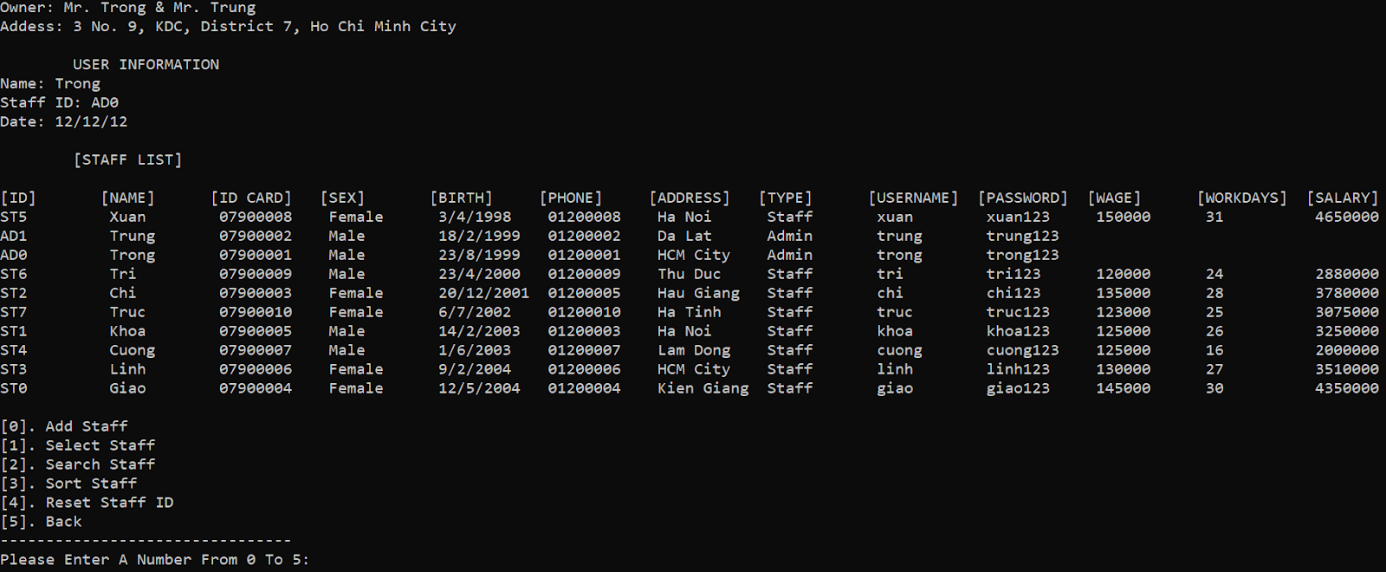
## **2.2 Phân quyền đăng nhập Admin**

Sau khi đăng nhập vào tài khoản của admin, người dùng sẽ có các tính năng quản lý như Staff Manager (Quản lý nhân viên), Table Manager (Quản lý bàn), Service Manager (Quản lý các dịch vụ), Income Manager và cuối cùng là Log out (Đăng xuất).

****

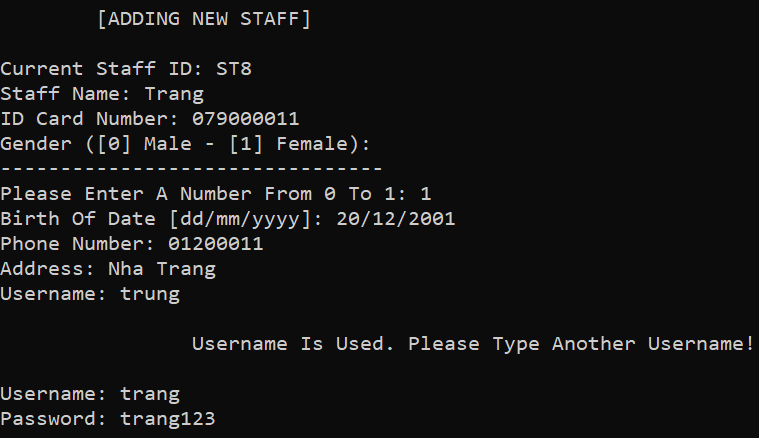
Khi lựa chọn option 4 (Log out), console sẽ chiếu lại màn hình đăng nhập ban đầu.

### **2.2.1 Staff Manager**

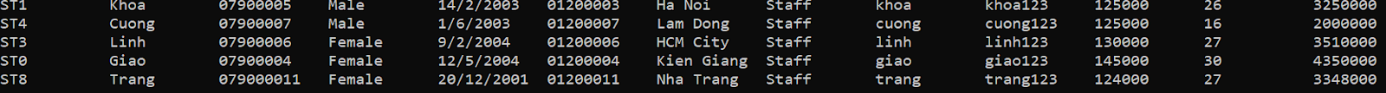


Khi lựa chọn option này, Màn hình sẽ xuát ra danh sách thông tin các nhân viên bao gồm cả Admin và Staff. (Admin không có WAGE, WORKING DAYS và SALARY).

* **Add Staff: Thêm một nhân viên mới**



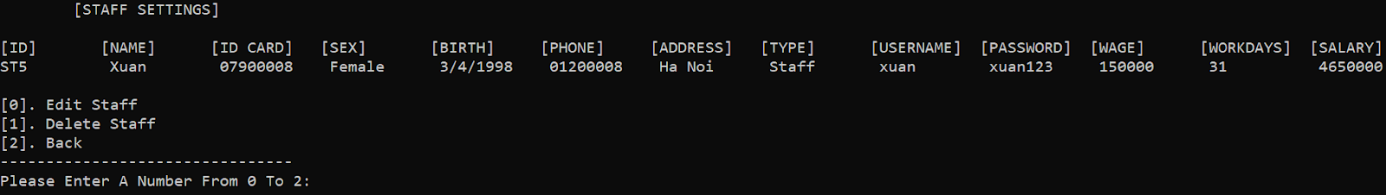
Khi lựa chọn option này, người dùng nhập vào từng thông tin cụ thể của nhân viên mới (Nhân viên mới mặc định là Staff, không thể thêm Admin). Đến phần nhập tên tài khoản (username), nếu nhập lại tài khoản đã có sẵn trong danh sách rồi thì console thông báo nhập lại. Sau khi nhập xong hết thì nhân viên sẽ được thêm vào bảng danh sách các nhân viên.

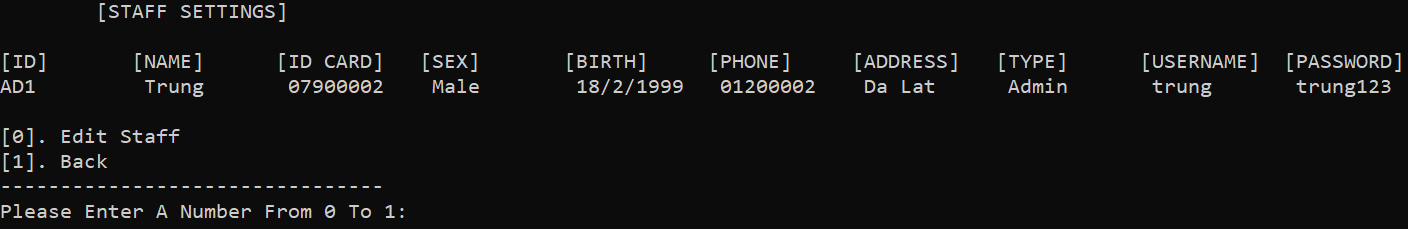


* **Select Staff: Lựa chọn một nhân viên cụ thể**

****

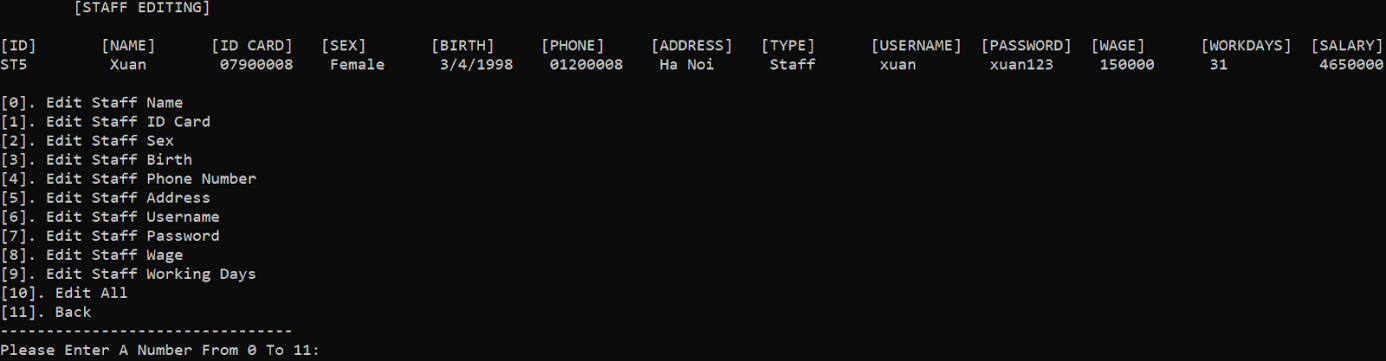
Nhập vào ID của nhân viên, giao diện chuyển tới màn hình quản lý đối với một nhân viên.





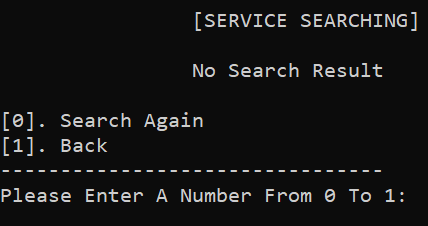
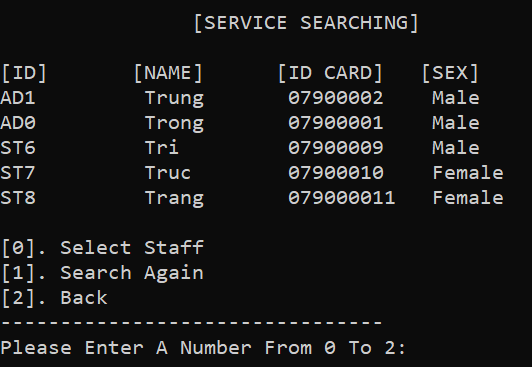
Nếu là nhân viên đó là Staff thì có thêm option Delete Staff, khi thực hiện sẽ xóa nhân viên đó khỏi danh sách Staff, còn nếu nhân viên được chọn là Admin thì không thể xóa.

Khi thực hiện option 0, màn hình chuyển đến tab Staff Editting, cho phép chỉnh sửa dữ liệu của nhân viên (trừ mã ID). Option Edit All là chỉnh sửa lại tất cả thông tin.



* **Search Staff: Tìm kiếm thông tin nhân viên nào có thông tin chứa từ khóa được nhập**

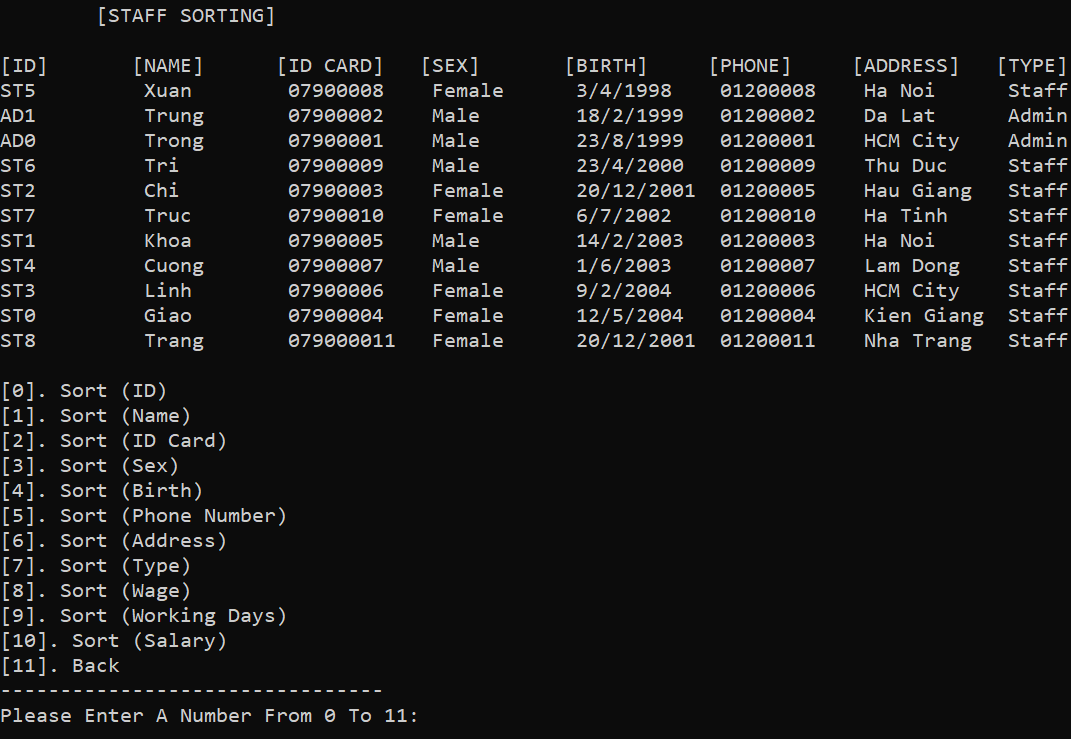
****

****

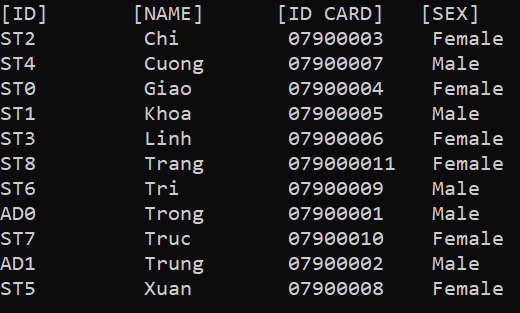
Khi chọn tính năng này, cho phép người dùng nhập vào từ khóa và sẽ tìm kiếm xuất ra danh sách các nhân viên có thông tin chứa từ khóa đó, bất kể ở thuộc tính nào đi nữa. Nếu không tìm thấy sẽ xuất ra thông báo không có. Ta có thể tìm kiếm lại hoặc lựa chọn Back để quay về.

* **Sort Staff**

Màn hình sẽ chuyển đến tab Sort, đưa ra các lựa chọn sắp xếp theo từng thuộc tính.

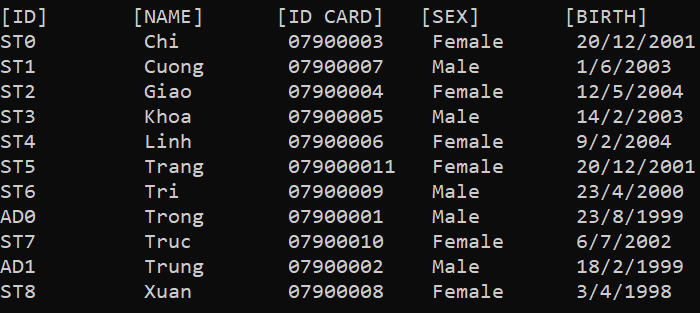


Ví dụ, ta chọn Sort Name (Sắp xếp theo tên nhân viên), màn hình sẽ chiếu ra danh sách các nhân viên được sắp xếp theo tên.

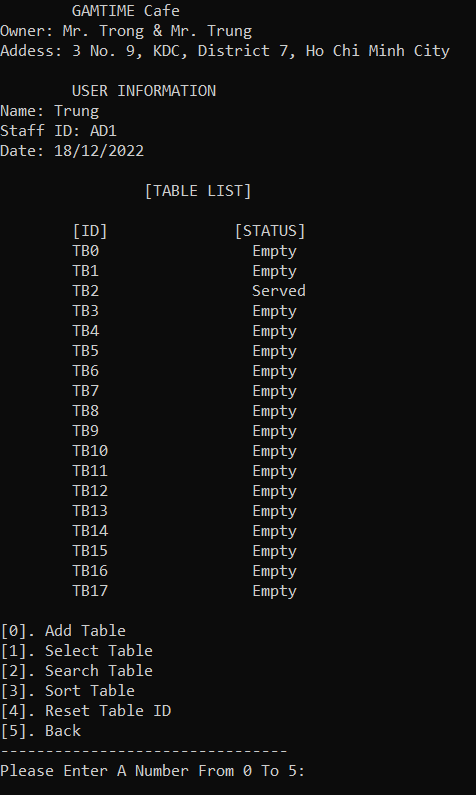


* **Reset Staff ID**

Chỉnh sửa lại tất cả các ID của Staff theo thứ tự tăng dần (trừ Admin). Dùng trong trường hợp khi xóa một nhân viên nhưng chúng ta muốn đẩy thứ tự ID của các nhân viên giảm về một đơn vị.

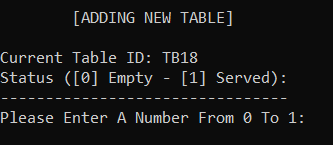


### **2.2.2 Table Manager**



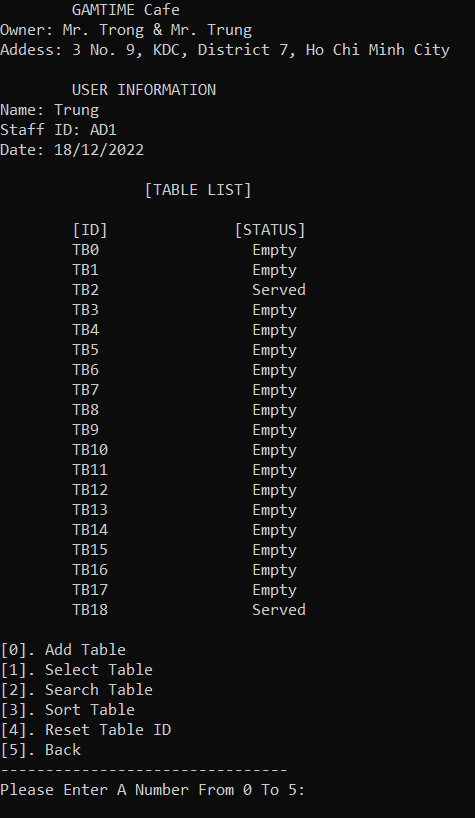
Ở mục này chúng ta có thể thực hiện các chức năng

* **Add Table: cho phép thêm 1 table mới**

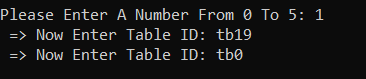


ID sẽ chạy theo thứ tự, và ta chỉ cần nhập trạng thái của bàn để tạo bàn mới.

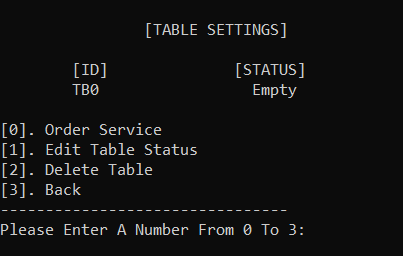
Nhập số 1 (Chọn Served) và bàn mới đã được thêm **TB18**



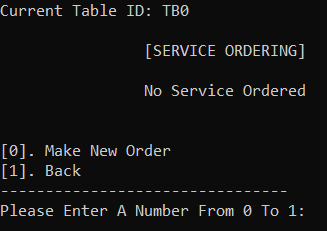
* **Select Table**

****

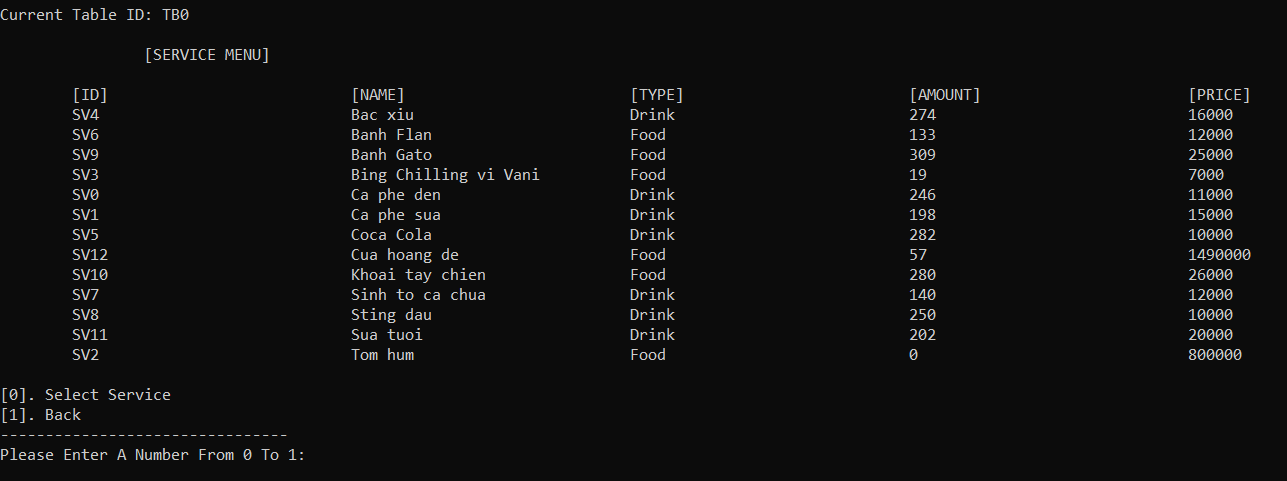
Ở đây đều thiết kế khả năng nhập sai của người nhập chỉ khi nào nhập đúng cấu trúc ID thì vào giao diện Select



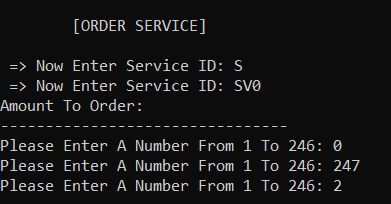
* Order Service



Chọn 0 ta sẽ ra giao diện của Service

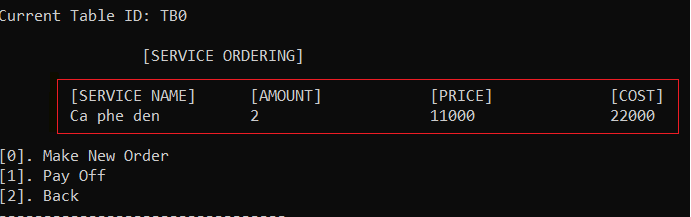


Chọn 0 ta có giao diện:



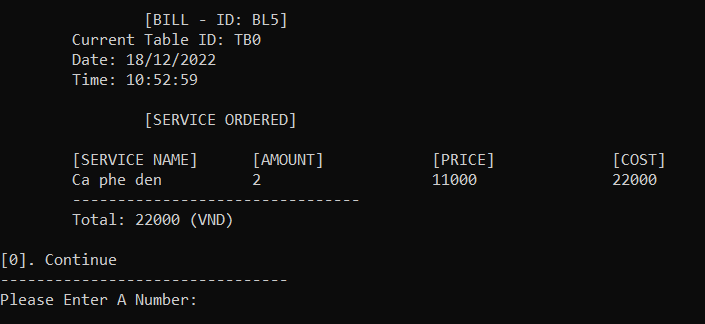
Chọn cà phê đen với số lượng 2, nếu nhập số lượng 0 hoặc lớn hơn số lượng tồn kho => **NHẬP LẠI**

**Sau khi xong thì back lại Order Service**

****

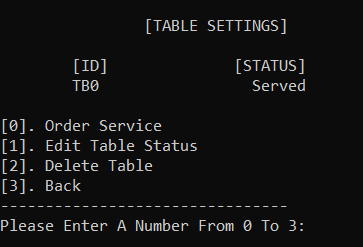
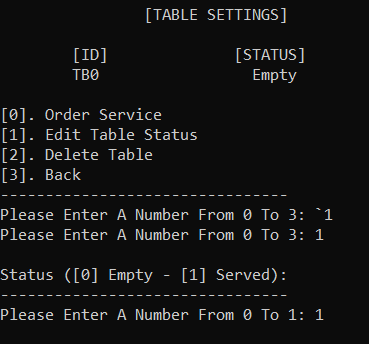
Đơn hàng vừa rồi đã được đưa vào danh sách đơn hàng của bàn. Và ở đây khi đã có Order thì xuất hiện thêm chức năng **Pay Off**

****

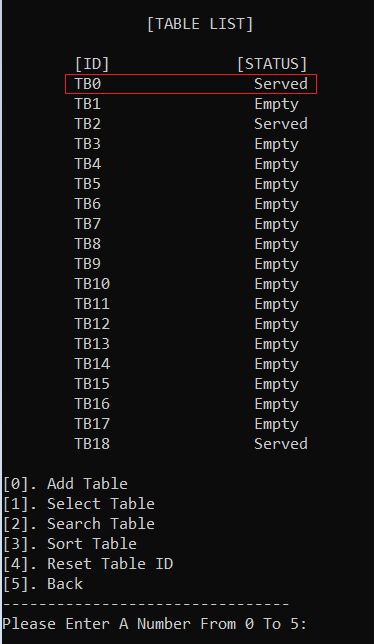


**Thực hiện xong việc xuất hóa đơn**

* Edit Table Status

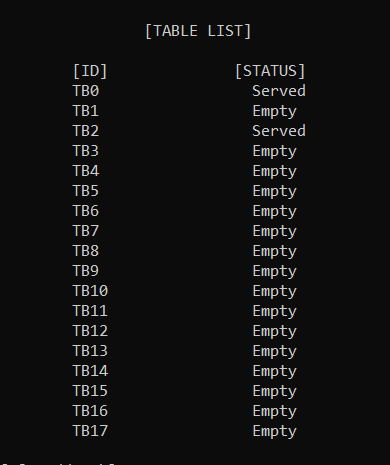
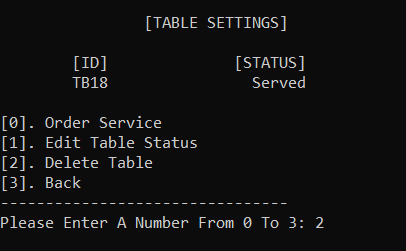


Sau khi edit liền cập nhật ở Table List



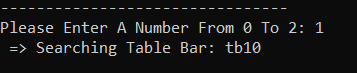
* **Delete Table**

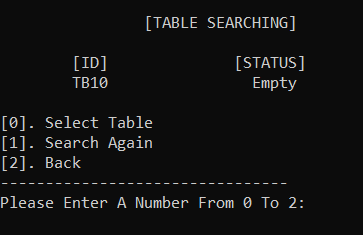
Thực hiện xóa **TB18**



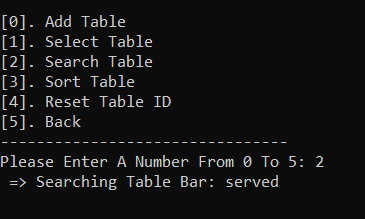
TB18 liền được xóa khỏi Table List

* Back: quay lại giao diện trước đó (Table Manager)
* **Search Table**

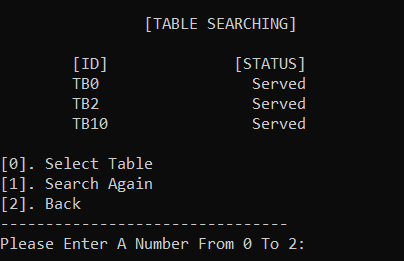
****

****

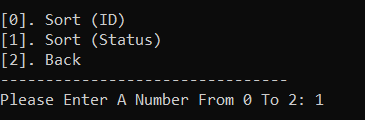
Cho phép search theo kí tự, số, nói chung là chuỗi con của các dữ liệu ở ID và Status



Và đây là kết quả trả ra



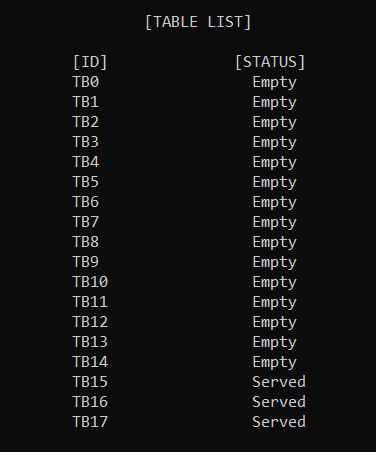
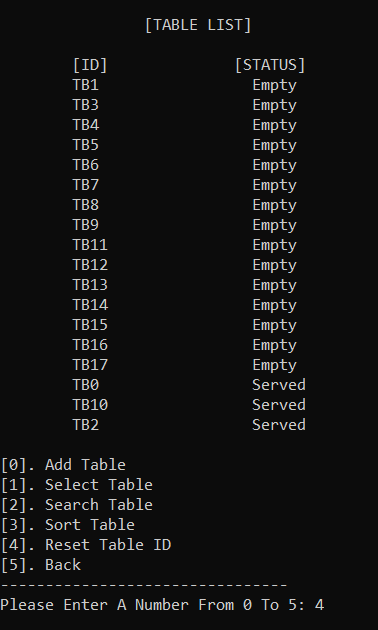
* **Sort Table**

****

VD: Sort theo status, Sort (ID) tương tự

****

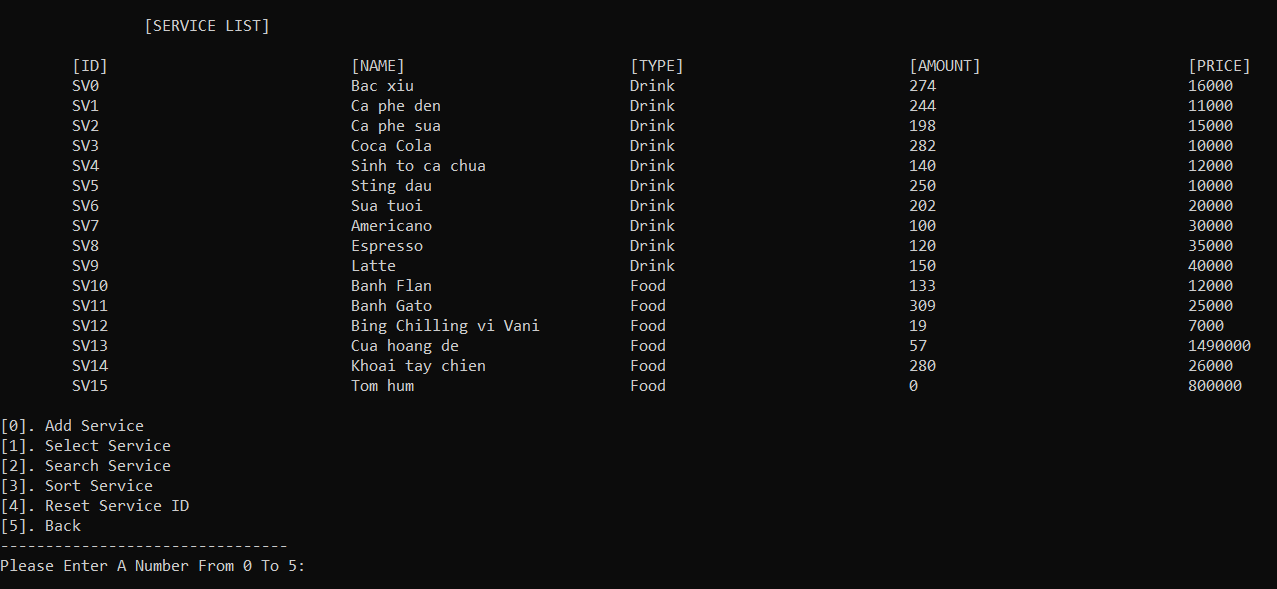
* **Reset Table ID**

****

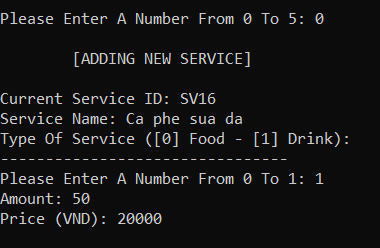
Dùng khi chúng ta muốn reset lại số thứ tự của các bàn, chủ yếu là do người quản lý quyết định

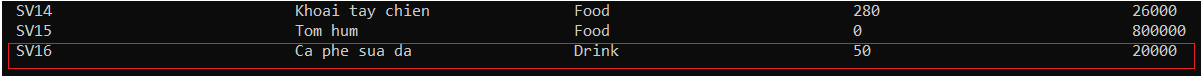
* **Back (quay lại giao diện của login)**

### **2.2.3 Service Manager**

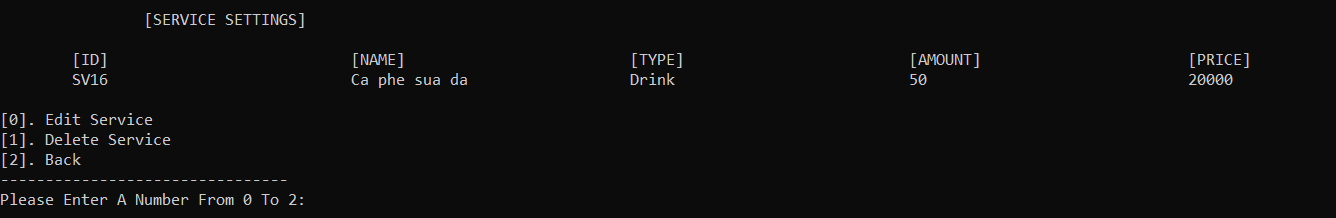


* **Add Service**

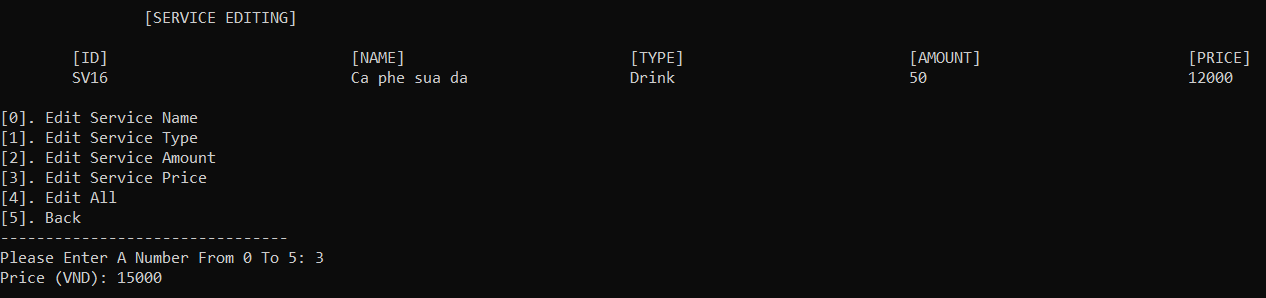
****

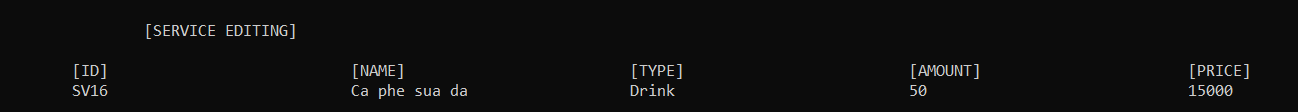
****

* **Select Service**

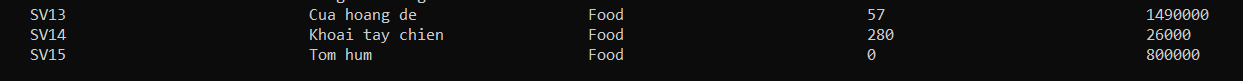
****

Thực hiện Edit Price, tương tự cho các Edit khác

****

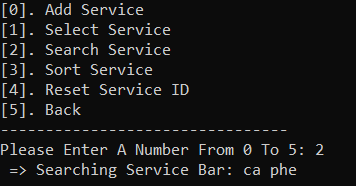
****

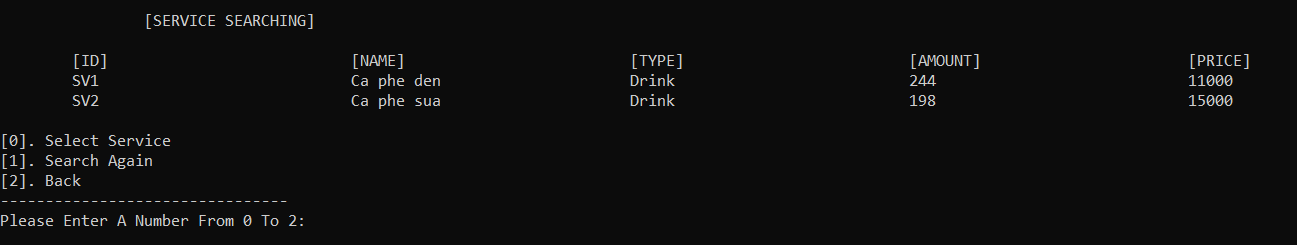
Xóa Service chọn 1



Sv16 vừa thêm đã có thể xóa

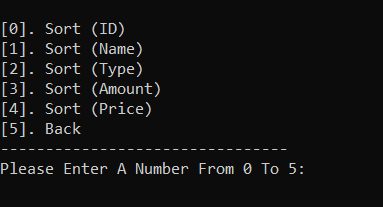
* **Search Service**

****

****

**Search tương tự như cách search ở Table**

* **Sort Service**

****

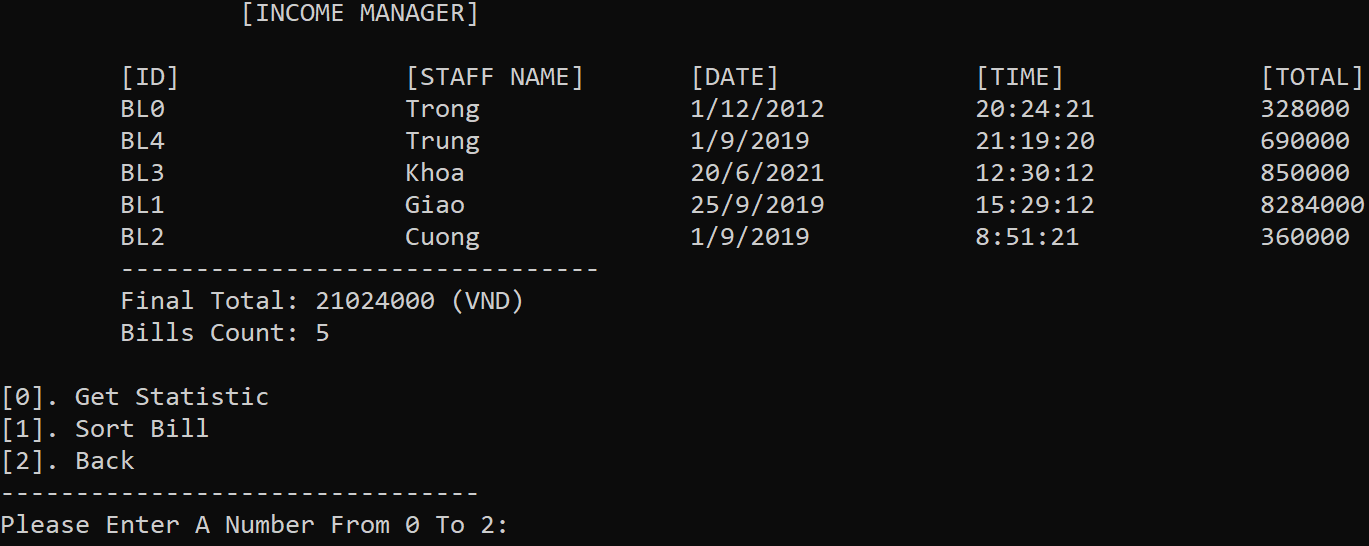
**VD: Sort theo Amount ta có Amount tăng dần:**

****

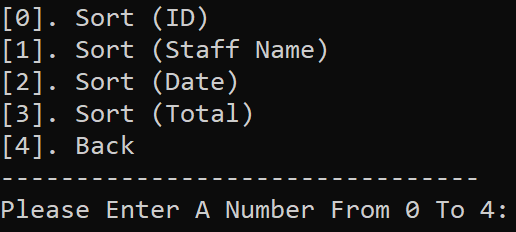
* **Reset Service ID**

Tương tự như cách reset Table ID.

### **2.2.4 Income Manager**

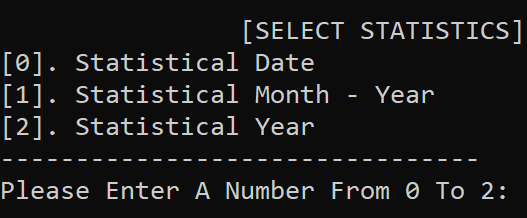
Màn hình chiếu ra danh sách thông tin các Bill đã được thanh toán trước đó. 

* **Sort Bill**

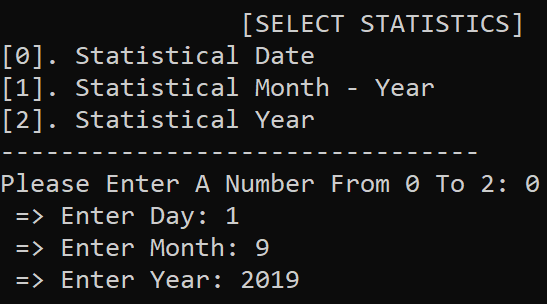
Tương tự như những tính năng sort trên, khi lựa chọn option này sẽ đưa người dùng đến tab Sorting và cho phép chọn sắp xếp theo từng thuộc tính cụ thể.

* **Get Statistic**

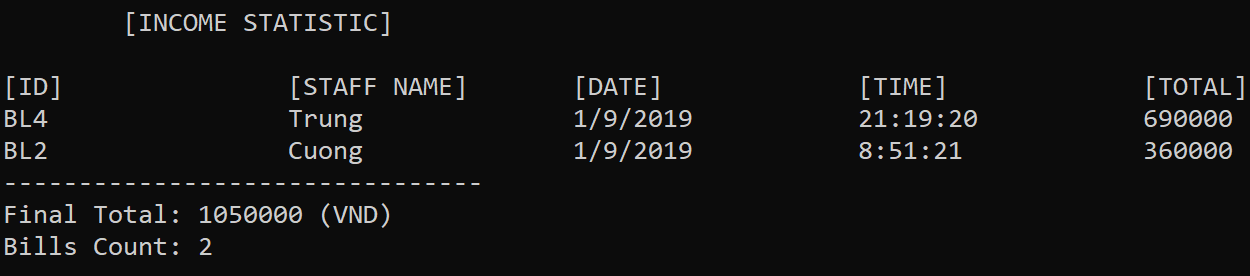
Màn hình hiện ra ba option thống kê theo một ngày cụ thể, thống kê theo tháng và cuối cùng là thống kê theo năm.



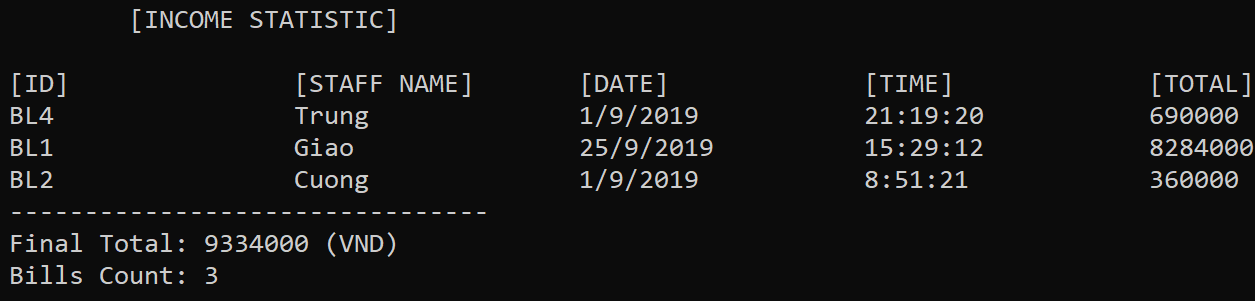
Ví dụ ta chọn thống kê theo ngày cụ thể là ngày 1/9/2019.



Kết quả trả ra là danh sách các Bill được thanh toán trong ngày đó.

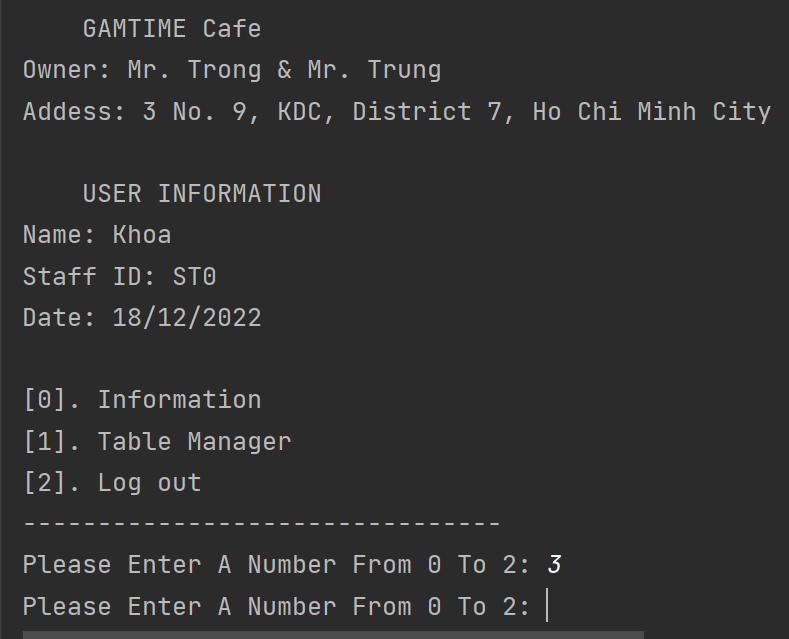


Tương tự, thực hiện với tháng 9/2019 hoặc trong năm 2019.

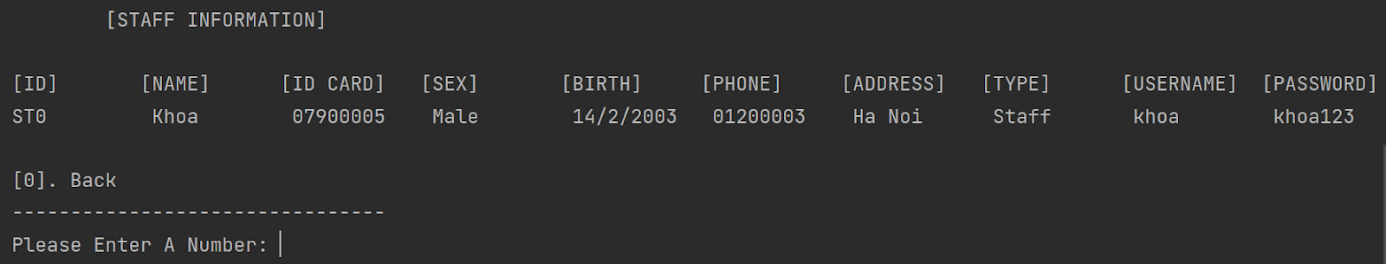


## **2.3 Phân quyền đăng nhập StaffService**

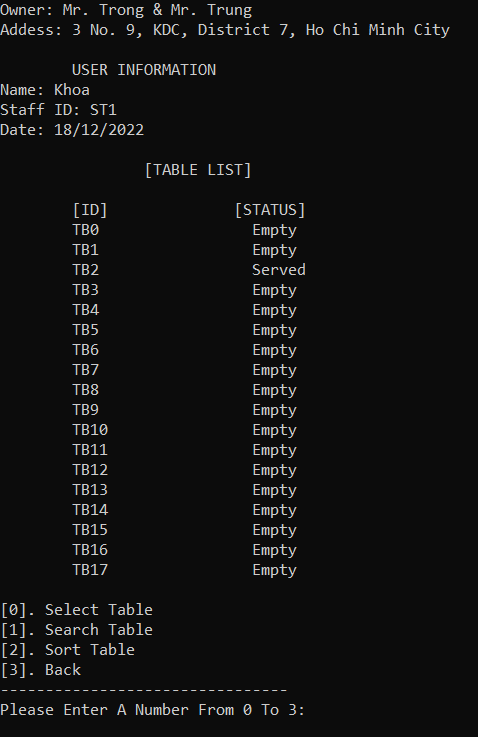
Một nhân viên chỉ có thể thực hiện 3 chức năng đó là xem thông tin cá nhân của chính mình, thực hiện thao tác với các bàn và log out. Nếu người dùng nhập sai, lập tức hiển thị cho người dùng chọn lại chức năng. **Và không có các chức năng như Staff Manager, Service Manager hay Income Manager như ở Admin**

****

### **2.3.1 Information**

****

### **2.3.2 Table Manager**

****

Trong mục này có thể thực hiện các thao tác **chọn bàn** (để xóa hay thay đổi tình trạng), **tìm kiếm bàn** (tìm những bàn còn trống), **sắp xếp** các bàn theo ID hay theo STATUS theo các phím chức năng [0], [1], [2], [3] (back)

**Các chức năng này này tương tự trên class Admin.**

**Và không có các chức năng như ADD Table, Reset Table ID như ở Admin**

# **BẢNG PHÂN CÔNG CÔNG VIỆC**

|  |  |  |
| --- | --- | --- |
| **PHIÊN BẢN** | **CÔNG VIỆC** | **NGƯỜI THỰC HIỆN** |
| C# | Class Date | Trọng |
| Class Time | Trọng |
| Class Account | Trọng |
| Class Service | Trọng |
| Class Staff | Trung |
| Class StaffService | Trung |
| Class Table | Trung |
| Class Admin | Trọng, Trung |
| Class Cafe | Trọng, Trung |
| Class Order | Trọng, Trung |
| Class Bill | Trọng, Trung |
| Main | Trọng, Trung |
| Java | Class Date | Trọng |
| Class Time | Trọng |
| Class Account | Trung |
| Class Service | Trung |
| Class Staff | Trọng |
| Class StaffService | Trọng |
| Class Table | Trung |
| Class Admin | Trọng, Trung |
| Class Cafe | Trọng, Trung |
| Class Order | Trọng, Trung |
| Class Bill | Trọng, Trung |
| Main | Trọng, Trung |
| C++ | Class Date | Trọng |
| Class Time | Trọng |
| Class Account | Trọng |
| Class Service | Trung |
| Class Staff | Trọng, Trung |
| Class Table | Trung |
| Class Cafe | Trọng, Trung |
| Class Order | Trung |
| Class Bill | Trọng |
| Main | Trọng, Trung |