

TRƯỜNG ĐẠI HỌC BÁCH KHOA – ĐHQG THÀNH PHỐ HỒ CHÍ MINH



NGÔN NGỮ LẬP TRÌNH TIỀN TIẾN

BÁO CÁO KỸ THUẬT

Giảng viên hướng dẫn:

MSc. Nguyễn Cao Đạt

Sinh viên thực hiện:

Phuong Tấn Thành - 1811226

Quan Chấn Huy - 1810181

Nguyễn Quang Hưng - 1810967

MỤC LỤC

Chương I: GIỚI THIỆU	3
1.1. Mục tiêu của đề tài	3
1.2. Yêu cầu của ứng dụng cần xây dựng	3
1.3. Nội dung chính của mỗi chương	3
Chương II: PHÂN TÍCH VÀ THIẾT KẾ	4
2.1. Xây dựng ứng dụng không có giao diện đồ họa	4
2.1.1. Khối UI	5
2.1.2. Khối Controller	5
2.1.3. Khối Entity	6
2.2. Xây dựng ứng dụng có giao diện đồ họa	7
2.3. Xây dựng ứng dụng có quản lý cơ sở dữ liệu bằng MySQL	7
Chương III: HIỆN THỰC VÀ ĐÁNH GIÁ	7
3.1. Hiện thực chương trình	7
3.1.1. Các class trong khối Entity	7
3.1.2. Các class trong khối Controller	11
3.1.3. Các class trong khối UI	12
3.2. Chạy thử và đánh giá chương trình	15

Chương I: GIỚI THIỆU

1.1. Mục tiêu của đề tài

Từ thực tế các công việc liên quan đến tín dụng đòi hỏi khối lượng thông tin lớn cùng nhu cầu nhập liệu, chỉnh sửa phải nhanh chóng, thuận tiện. Tuy nhiên, việc tiến hành các thao tác này một cách thủ công đem lại nhiều bất tiện, khó cho việc quản lý các dữ liệu về người vay, thông tin vay. Đó là lý do đề tài về ứng dụng quản lý tín dụng được thực hiện, với mục tiêu giải quyết các vấn đề nêu trên.

Không chỉ đảm bảo tính thiết thực, đề tài còn cho thấy tính khả thi khi ứng dụng được xây dựng trên nền tảng ngôn ngữ Java – một ngôn ngữ hướng đối tượng cho phép giải quyết các vấn đề phức tạp, đồng thời có khả năng sửa chữa nâng cấp dễ dàng.

1.2. Yêu cầu của ứng dụng cần xây dựng

Ứng dụng quản lý tín dụng cần đảm bảo những yêu cầu khi hoàn thiện như sau:

- Hoạt động hiệu quả, không xảy ra lỗi trong quá trình sử dụng
- Thân thiện với người dùng, dễ dàng thao tác và sử dụng
- Có khả năng nâng cấp, phát triển về sau
- Thực hiện được những chức năng cơ bản như sau:
 - Quản lý (hiển thị danh sách, thêm, xóa, sửa) người vay
 - Quản lý (hiển thị, thêm, xóa, sửa) các khoản vay của mỗi người
 - Dự báo tháng tới sẽ thu được số tiền lãi và gốc từ tất cả các người vay

1.3. Nội dung chính của mỗi chương

Như vậy chương 1 đã nêu ra mục tiêu của đề tài cũng như đề ra những yêu cầu cho đầu ra của đề tài. Trong những chương tiếp theo những nội dung sẽ lần lượt trình bày như sau

Chương 2: trình bày các bước phân tích và thiết kế ứng dụng dựa trên các yêu cầu đặt ra ở chương 1

Chương 3: trình bày phần hiện thực ứng dụng, kiểm tra/đánh giá tính đúng đắn của ứng dụng đã xây dựng

Chương 4: trình bày những gì đã làm được và không làm được theo các yêu cầu đã đặt ra, ưu và nhược điểm của ứng dụng, hướng phát triển của ứng dụng

Chương II: PHÂN TÍCH VÀ THIẾT KẾ

Trong chương này, những phân tích và ý tưởng thiết kế chương trình sẽ được đưa ra nhằm thực hiện yêu cầu đã đề ra. Cụ thể, nhóm sẽ tiến hành phân tích và thiết kế theo phần riêng biệt: chương trình không có giao diện đồ họa mà thao tác diễn ra ngay trên màn hình console, chương trình có giao diện đồ họa, tiến hành lưu trữ và quản lý dữ liệu bằng MySQL.

2.1. Xây dựng ứng dụng không có giao diện đồ họa

Trong ứng dụng quản lý tín dụng sẽ có các trường chứa các thông tin về người vay như tên, địa chỉ cũng như các trường chứa thông tin về khoản vay. Ngoài ra, chương trình sẽ đảm nhận các chức năng trên các trường thông tin như tạo mới, xóa, sửa, dự báo lãi vay. Cuối cùng, chương trình này dành cho người dùng phổ thông sử dụng nên cần có một giao diện người dùng thân thiện và có thể sử dụng ngay. Vậy sau khi xác định được 3 khối công việc chính nhóm sẽ tiến hành tạo ra một hay một số class thực hiện riêng cho từng khối công việc, lần lượt là khối UI sẽ đảm nhận các vấn đề về giao diện, khối Controller đảm nhận thực hiện các chức năng của chương trình, khối Entity sẽ chứa các trường thông tin về người vay và khoản vay. Ở đây, nhóm nhận thấy để đảm bảo tính hiệu quả và tổng quát nhóm sẽ đi từ phân tích khối UI trước

2.1.1. Khối UI

Trong khối này, nhóm sẽ tiến hành thiết kế một giao diện người dùng theo dạng các menu hướng dẫn. Cụ thể, giao diện sẽ là một menu lựa chọn được đánh số từ 0 trở đi, mỗi số sẽ tương ứng với một tác vụ. Người dùng sẽ nhập số vào để đi đến tác vụ hoặc menu nhỏ hơn.

Điều thứ hai có trong khối UI này đó là việc đọc và ghi dữ liệu vào cơ sở dữ liệu, ở đây vì chưa tiến hành sử dụng MySQL để quản lý dữ liệu nên dữ liệu chỉ được quản lý ở dạng sơ khai là một file .txt. Cụ thể, khi bắt đầu chương trình giao diện console sẽ hiện ra với các menu như đã đề cập ở trên, đồng thời chương trình sẽ truy cập và đọc dữ liệu từ file .txt đã dẫn hướng. Khi kết thúc chương trình, những thay đổi về dữ liệu sẽ được ghi đè lên cơ sở dữ liệu.

Nói chi tiết hơn về các menu, giao diện ban đầu khi người dùng chạy chương trình sẽ là menu lớn với các lựa chọn:

- Mở menu người vay bao gồm các tác vụ liên quan đến người quay, nhập 0 để quay về menu chính
- Mở menu khoản vay bao gồm các tác vụ liên quan đến khoản quay, nhập 0 để quay về menu chính
- Dự báo tiền lãi cho vay
- Nhập 0 để kết thúc chương trình

Cuối cùng, cũng trong class thuộc khối UI sẽ là các method thực hiện các tác vụ có trong menu, những method này sẽ thực hiện yêu cầu nhập vào thông qua việc gọi các method có trong các class của khối Controller. Mục đích của việc này là giúp chương trình theo dõi và sửa chữa hơn khi có nhu cầu thay đổi nội dung của menu.

2.1.2. Khối Controller

Khối này như đã đề cập ở trên sẽ bao gồm một hoặc một số class trong đó chứa các method thực hiện các tác vụ trong menu. Cụ thể, các method sẽ được chia làm 3 loại:

các method thực hiện các tác vụ liên quan đến người vay (Client Tasks), các method thực hiện các tác vụ liên quan đến khoản vay (Loan Tasks), method thực hiện tác vụ dự báo lợi nhuận cho vay, các method hỗ trợ thực hiện các task trên.

Các method liên quan đến người vay bao gồm: thêm người vay mới, hiển thị thông tin tất cả người vay, thay đổi thông tin người vay, xóa người vay. Các method liên quan đến khoản vay bao gồm: thêm khoản vay mới, hiển thị thông tin tất cả khoản vay, thay đổi thông tin khoản vay, xóa khoản vay. Các method hỗ trợ có thể kể đến như đọc lệnh nhập vào từ bàn phím (ở đây chỉ số người dùng nhập vào), đọc ID người vay,...

Một số lưu ý cần làm rõ trong phần này đó là về ID người vay và tác vụ xóa người vay. Nói rõ hơn về ID người vay, vì thực tế sử dụng là mỗi người vay có thể có nhiều khoản vay do đó thay vì việc gọi method thêm người vay mỗi lần thêm một khoản vay cho người cũ, ta sẽ dùng ID người vay. Nghĩa là việc làm gọi method người vay khi người đó vay thêm một khoản vay khác sẽ tạo ra một đối tượng song song có thông tin giống với đối tượng người vay gốc. Còn về ID người vay, dù có nhiều đối tượng ID giống nhau tồn tại song song thì giá trị của nó vẫn là duy nhất và nó trỏ về một đối tượng người vay duy nhất. Thứ hai, về việc xóa người vay, việc xóa người vay sẽ xóa luôn cả các khoản vay của người đó. Điều này là hợp lý vì rõ ràng chỉ khi người người đó đã hoàn thành nghĩa vụ trả các khoản vay và có yêu cầu xóa người vay thì việc xóa này cũng sẽ bao gồm xóa các khoản vay của người đó.

2.1.3. Khối Entity

Trong khối này, các class thể hiện các trường dữ liệu sẽ được tạo ra với mục đích ghi và đọc các dữ liệu thuộc các trường này. Trong khối dự kiến sẽ có những class người (Person), người vay (Client), khoản vay (Loan), tên (Name), địa chỉ (Address).

Class Name sẽ bao gồm các thông tin về tên, họ, tên đệm. Class Address sẽ bao gồm số nhà, địa chỉ, quận/huyện, tỉnh/thành phố. Class Person có các thông tin tên, địa chỉ, ID liên hệ với class Name và class Address theo quan hệ kết hợp (Association) – quan hệ xảy ra khi một đối tượng có thuộc tính là các đối tượng thuộc lớp khác. Class Client sẽ

thừa kế class Person. Cuối cùng, class Loan sẽ mang các thông tin về ngày vay, thời gian vay, khoản vay, ID người vay và lãi suất vay.

2.2. Xây dựng ứng dụng có giao diện đồ họa

2.3. Xây dựng ứng dụng có quản lý cơ sở dữ liệu bằng MySQL

Chương III: HIỆN THỰC VÀ ĐÁNH GIÁ

Trong phần hiện thực và đánh giá này, nhóm sẽ trình bày chi tiết các việc hiện thực các khối làm việc ở chương II, thể hiện các class thông qua các bảng tóm tắt về thuộc tính (attribute), phương thức khởi tạo (constructor), các phương thức (method), cũng như làm rõ hướng giải quyết các vấn đề cho từng method. Chương trình đã viết sẽ đi kèm theo bài báo cáo. Sau cùng, nhóm tiến hành chạy một số test case cho chương trình đã viết, rút ra nhận xét đánh giá

3.1. Hiện thực chương trình

3.1.1. Các class trong khối Entity

Name	public
-firstName: String	-private
-middleName: String	-private
-lastName: String	-private
+Name(firstName: String, middleName: String, lastName: String)	+ public
+getFirstName(): String	+ public
+getMiddleName(): String	+ public
+getLastName(): String	+ public
+setFirstName(firstName: String): void	+ public
+setMiddleName(middleName: String) : void	+ public

+setLastName(lastName: String) : void	+ public
+getText(): String	+ public
+getForFile(): String	+ public

- Class Name có ba biến firstName, middleName và lastName chỉ tên, tên đệm và họ
- Name có cùng tên với class là một phương thức khai báo hay constructor
- getFirstName(), getMiddleName(), getLastName(), setFirstName(), setMiddleName(), setLastName() là các phương thức setter và getter để nhận vào và trả dữ liệu tương ứng với thuộc tính các đối tượng
- getText(), getForFile() đều trả về tên đầy đủ nhưng method thứ 2 phục vụ cho việc lưu trữ dữ liệu sau này
- Cột bên phải là phạm vi truy cập, sau dấu hai chấm sẽ là kiểu dữ liệu
- Những qui ước về mô tả class ở trên sẽ được áp dụng tương tự cho các bảng sau, nhóm sẽ chỉ chú thích những điểm quan trọng

Address	public
-number: int	-private
-street: String	-private
-district: String	-private
-city: String	-private
+Address(number: int, street: String, district: String, city: String)	+ public
+getNumber(): int	+ public
+getStreet(): String	+ public
+getDistrict(): String	+ public
+getCity(): String	+ public
+setNumber(number: int): void	+ public
+setStreet(street: String): void	+ public
+setDistrict(district: String) : void	+ public
+setCity(city: String) : void	+ public

+getText(): String	+ public
+getForFile(): String	+ public

- Các biến number, street, district, city chỉ số nhà, tên đường, quận/huyện, thành phố/tỉnh
- getText(), getForFile() đều trả về địa đầy đủ nhưng method thứ 2 phục vụ cho việc lưu trữ dữ liệu sau này

Person	public
-ID: int	-private
-name: Name	-private
-address: Address	-private
+Person()	+ public
+Person(name: Name, ID: int, address: Address)	+ public
+getID(): String	+ public
+setID(ID: int) : void	+ public
+getName(): Name	+ public
+getAddress(): Address	+ public

- Các biến name, ID, address chỉ thuộc tính tên, ID, địa chỉ của đối tượng. Ở đây ta thấy quan hệ kết hợp (Association) của class Person với class Name và Address, cụ thể đối tượng thuộc class Person có thuộc tính name và address là các đối tượng thuộc lớp Name và Address.

Client extends Person	public
-loanID: int	-private
+Client()	+ public
+Person(name: Name, ID: int, address: Address)	+ public
+getID(): String	+ public
+setID(ID: int) : void	+ public
+getName(): Name	+ public
+getAddress(): Address	+ public

- Biến loanID chỉ ID của khoản vay

- Ở đây class Client là class con của Person nên chỉ cần khai báo thêm biến loanID dùng riêng cho các này
- Trong class có tới 3 constructor nhằm các mục đích sử dụng khác nhau như tạo người vay mới, truy xuất đến một người vay hay thực hiện các method có trong class Client này

Loan	public
-annualInterestRate: double	-private
-numberOfYears: int	-private
-loanAmount: double	-private
-loanDate: final String	-private
-clientID: final int	-private
+Loan (annualInterestRate: double, numberOfYears: int, loanAmount: double, loanDate: String, clientID: int)	+ public
+ getClientID(): int	+ public
+ getLoanDate(): String	+ public
+ getAnnualInterestRate(): double	+ public
+ getNumberOfYears():int	+ public
+ getLoanAmount(): double	+ public
+ setAnnualInterestRate (annualInterestRate: double): void	+ public
+ setNumberOfYears(numberOfYears: int): void	+ public
+ setLoanAmount (loanAmount: double): void	+ public
+monthlyPayment(): double	+ public
+totalPayment(): double	+ public

- Các biến annualInterestRate, numberOfYears, loanAmount, loanDate và clientID lần lượt chỉ lãi suất vay hằng năm, số năm vay, số tiền vay, ngày vay, ID người vay
- Vì loanDate và clientID có kiểu final, có giá trị không đổi nên ta không có setter cho hai thuộc tính này
- Hai method monthlyPayment() và totalPayment() lần lượt cho ra số tiền mà người vay phải trả trong một tháng và một năm

3.1.2. Các class trong khối Controller

Trong khối này chỉ có một class là class Manager. Vì phần này chứa các method tương đối dài nên trong phần code, nhóm đã dùng phần comment ngăn các phần lớn với nhau.

Manager	public
-sc: final Scanner	-private
-listOfClients: final List<Client>	-private
-listOfLoans: final List<Loan>	-private
+ Manager()	+ public
+ readInt(min: int, max: int): int	+ private
+ addClient(): void	+ public
+ addClient(client: Client): void	+ public
+ removeClient(index: int): boolean	+ public
+ modifyClient(index: int): boolean	+ public
+ readID(): int	+ private
+ readAddress(): Address	+ private
+ readName(): Name	+ private
+ modifyID(clientModified: Client): void	+ private
+ modifyAddress(clientModified: Client): void	+ private
+ modifyName(clientModified: Client): void	+ private
+ clientCount(): int	+ public
+ getClient(index: int): Client	+ public
+ getClientByID(ID: int): Client	+ public
+ addLoan(): void	+ public
+ addLoan(loan: Loan): void	+ public
+ removeLoan(index: int): boolean	+ public
+ modifyLoan(index: int): boolean	+ public
+ loanCount(): int	+ public
+ getLoan(index: int): Loan	+ public
+ getLoanByID(ID: int): Loan	+ public

- Biến `sc` dùng để chứa dữ liệu nhập từ bàn phím, các biến `listOfClients` và `listOfLoans` có kiểu `List`, bên trong chứa các đối tượng có thuộc tính lần lượt là `Client` và `Loan` (quan hệ kết hợp)
- Phương thức `readInt(min: int, max: int)` trả về giá trị nhập từ bàn phím, giá trị này nằm trong khoảng `min` và `max`
- Phương thức `clientCount()` trả về kích thước của `listOfClients`, thực hiện chức năng tính số người vay, hỗ trợ cho các phương thức khác trong và ngoài class
- Phương thức `getClient()`, `getClientByID()` là các getter trả về người vay bằng index trong `listOfClients` hoặc bằng ID người vay
- Các phương thức `readAddress()`, `readName()` có kiểu là các đối tượng `Address` và `Name` thuộc 2 class tương ứng trong nhóm `Entity`
- Phương thức `addClient()` gọi các phương thức `readID()`, `readAddress()`, `readName()` để tạo một đối tượng `client` mới sau đó thêm đối tượng vào `listOfClients`
- Phương thức `removeClient()` xóa người vay khỏi `listOfClients`
- Phương thức `modifyClient()` chỉnh sửa thông tin người vay, khi thực hiện sẽ gọi các phương thức `modifyID()`, `modifyAddress()`, `modifyName()`.
- Các phương thức `addLoan()`, `removeLoan()`, `modifyLoan()`, `loanCount()`, `getLoan()`, `getLoanByID()` tương tự các phương thức liên quan đến `client` ở trên, chỉ khác là đối tượng thực hiện liên quan đến khoản vay
- Trong các phương thức nêu trên đều có lựa chọn giữ nguyên thông tin, nhằm tiết kiệm thời gian cho người sử dụng trong trường hợp người sử dụng chọn tác vụ nhưng lại đổi ý không muốn thực hiện tiếp.

3.1.3. Các class trong khối UI

Trong khối này chỉ có một class là class `Console` đảm nhận các chức năng như đã phân tích ở phần phân tích và thiết kế. Vì phần này chứa các method quan trọng của chương trình và tương đối dài nên trong phần code nhóm thực hiện, nhóm đã dùng phần comment để ngăn cách các phần lớn với nhau.

Console	public
-sc: final Scanner	-private
-m: final Manager	-private
+ Console()	+ public
+ Start(): void	+ public
+ menu(): int	+ private
+ readInt(min: int, max: int): int	+ private
+ clientTasks(): void	+ private
+ clientMenu(): int	+ private
+ loadClientData(): void	+ private
+ addClient(): void	+ private
+ showAllClient(): void	+ private
+ removeClient(): void	+ private
+ modifyClient(): void	+ private
+ writeClientData(): void	+ private
+ loanTasks(): void	+ private
+ loanMenu(): int	+ private
+ loadLoanData(): void	+ private
+ addLoan(): void	+ private
+ showAllLoan(): void	+ private
+ removeLoan(): void	+ private
+ modifyLoan(): void	+ private
+ writeLoanData(): void	+ private
+ PredictProfit(): void	+ private

- Biến sc dùng để chứa dữ liệu nhập từ bàn phím, còn biến m có kiểu Manager – một đối tượng của class Manager (quan hệ kết hợp), các thuộc tính này sẽ được gán cho đối tượng trong class Console ở phương thức khởi tạo Console()
- Phương thức readInt(min: int, max: int) trả về giá trị nhập từ bàn phím, giá trị này nằm trong khoảng min và max

- Phương thức `menu()` để hiển thị các danh sách chọn lựa và gọi phương thức `readInt(min: int, max: int)` với `min` và `max` là giá trị nhỏ nhất và lớn nhất của số thứ tự các lựa chọn trong menu
- Phương thức `Start()` đầu tiên sẽ gọi phương thức `menu()` và sau đó sử dụng `switch-case` để thực hiện các phương thức tương ứng với tác vụ ghi trong menu: mở menu người vay, mở menu khoản vay, dự đoán lợi nhuận và case 0 sẽ là thoát, case default sẽ là báo lỗi. Lưu ý, đầu tiên nhất trong phương thức, chương trình sẽ gọi `loadClientData()` và `loadLoanData()` để tải dữ liệu khoản vay và người vay từ cơ sở dữ liệu
- Phương thức `clientMenu()` để hiển thị các danh sách chọn lựa tác vụ đối với người vay và gọi phương thức `readInt(min: int, max: int)` với `min` và `max` là giá trị nhỏ nhất và lớn nhất của số thứ tự các lựa chọn trong menu
- Phương thức `clientTasks()` đầu tiên sẽ gọi phương thức `clientMenu()` và sau đó sử dụng `switch-case` để thực hiện các phương thức tương ứng với tác vụ ghi trong menu người vay: thêm, xóa, sửa người vay, hiển thị thông tin tất cả người vay và case 0 sẽ là thoát, case default sẽ là báo lỗi
- Phương thức `loadClientData()` sử dụng vòng lặp để ghi các thông tin người dùng (tên, địa chỉ, ID,...) vào bộ nhớ đang làm việc. Nghĩa là dữ liệu từ lần cuối chương trình được chạy sẽ được cập nhật lại trước khi bắt đầu một quá trình mới. Lưu ý: Đường dẫn đến file .txt khi tải về sẽ phải thay đổi tùy theo vị trí lưu file. Nhóm sẽ chú thích điều này trong file README.txt
- Các phương thức `addClient()`, `showAllClient()`, `modifyClient()`, `removeClient()` sẽ gọi các phương thức tương ứng trong class `Manager` để thực thi các tác vụ. Ở các phương thức này còn có nhưng thông báo cho người dùng báo lệnh gì đã được thực thi.
- Phương thức `writeClientData()` cũng truy xuất đến đúng đường dẫn của file .txt lưu trữ dữ liệu để cập nhật thông tin khi quá trình làm việc kết thúc, điều này đảm bảo các dữ liệu khi nhập vào sẽ được lưu lại cho những mục đích sử dụng sau này

- Các phương thức `loanTasks()`, `loanMenu()`, `loadLoanData()`, `addLoan()`, `showAllLoan()`, `removeLoan()`, `modifyLoan()`, `writeLoanData()` tương tự các phương thức liên quan đến client. Các phương thức liên quan đến loan sẽ thực hiện các chức năng trong menu khoản vay như thêm, sửa, xóa, hiển thị, đồng thời cũng có các phương thức cập nhật và tải cơ sở dữ liệu như client. Một số lưu ý: trong phương thức `addLoan()`, `removeLoan()` và `modifyLoan()` ngoài việc thêm/xóa/sửa khoản vay còn gọi thêm phương thức `showAllClient()` để người thực hiện có thể tham khảo trong trường hợp làm việc với người vay cũ.
- Phương thức `PredictProfit()` tính toán lợi nhuận thu được thông qua việc gọi các phương thức `getLoan()` và `loanCount()` trong class `Loan` thuộc khối `Entity`, kết quả sẽ được làm tròn 2 chữ số
- ❖ *Class `CreditManagement` chứa hàm `main`*: Sẽ có một class riêng chứa hàm `main` để chạy chương trình, class này sẽ tạo ra một đối tượng `Console` mới và chạy method `Start()`. Phần sử dụng sẽ được chú thích rõ trong file `README.txt`

3.2. Chạy thử và đánh giá chương trình

```

Successfully load Client data.

.------.
|===== CREDIT MANAGE MENU =====|
| 1. Client Manage Menu              |
| 2. Loan Manage Menu                |
| 3. Predict Profit                  |
| 0. Exit                            |
|-----|
Enter your choice: 1

.------.
|=====Client Menu=====|
| 1. Add New Client              |
| 2. Show All Client            |
| 3. Modify Client Info         |
| 4. Remove Client              |
| 0. Back to Main Menu          |
|-----|
Enter your choice: 2
===== All Client =====
No | Full Name | ID | Address
1 | Thanh Tan Phuong | 1811226 | 218 Pham Van Dong, Pleiku, Gia Lai
2 | Hung Quang Nguyen | 1810967 | 46 Le Duan, Quy Nhon, Binh Dinh
3 | Huy Chan Quan | 1810181 | 54 Le Loi, Thu Duc, Ho Chi Minh
4 | jsdkvj lsjdv vjsdlljd | 23 | 34 skldvkl, sdk, sdvkl
5 | Thu Anh Phuong | 1810100 | 218 pvd, pk, gl
=====

.------.
|=====Client Menu=====|
| 1. Add New Client              |
| 2. Show All Client            |
| 3. Modify Client Info         |
| 4. Remove Client              |
| 0. Back to Main Menu          |
|-----|
Enter your choice:

```

Xem dữ liệu được nạp sẵn


```
.-----.  
|=====Client Menu=====|  
| 1. Add New Client      |  
| 2. Show All Client     |  
| 3. Modify Client Info  |  
| 4. Remove Client       |  
| 0. Back to Main Menu   |  
|-----|  
Enter your choice: 1  
-----Enter Client Name-----  
-> First Name: T  
-> Last Name: P  
-> Middle Name: T  
-> Enter ID: 1811223  
-----Enter Client Address-----  
-> Number: 497  
-> Street: Hoa Hao  
-> District: 10  
-> City: HCMC
```

Nạp thông tin khách hàng

```

1 | Thanh Tan Phuong | 1811226 | 218 Pham Van Dong, Pleiku, Gia Lai
2 | Hung Quang Nguyen | 1810967 | 46 Le Duan, Quy Nhon, Binh Dinh
3 | Huy Chan Quan | 1810181 | 54 Le Loi, Thu Duc, Ho Chi Minh
4 | jsdkvj lsjdv vjsdlljd | 23 | 34 skldvkl, sdk, sdvkl
5 | Thu Anh Phuong | 1810100 | 218 pvd, pk, gl
6 | T T P | 1811223 | 497 Hoa Hao, 10, HCMC
=====
-> Enter the No of Client to modify: 4
Change the info. Type '0' to keep remain
-----Modify Client Name-----
-> First Name: A
-> Last Name: Van
-> Middle Name: Nguyen
-----Modify Client ID-----
-> Enter ID: 0
-----Modify Client Address-----
-> Number: 0
-> Street: 0
-> District: 0
-> City: 0
Client was modified!

'.-----'.
|====Client Menu====|
| 1. Add New Client   |
| 2. Show All Client  |
| 3. Modify Client Info |
| 4. Remove Client    |
| 0. Back to Main Menu |
|-----|
Enter your choice: 2
===== All Client =====
No | Full Name | ID | Address
1 | Thanh Tan Phuong | 1811226 | 218 Pham Van Dong, Pleiku, Gia Lai
2 | Hung Quang Nguyen | 1810967 | 46 Le Duan, Quy Nhon, Binh Dinh
3 | Huy Chan Quan | 1810181 | 54 Le Loi, Thu Duc, Ho Chi Minh
4 | A Nguyen Van | 23 | 34 skldvkl, sdk, sdvkl
5 | Thu Anh Phuong | 1810100 | 218 pvd, pk, gl
6 | T T P | 1811223 | 497 Hoa Hao, 10, HCMC
=====

```

Sửa thông tin khách hàng (sửa thông tin khách hàng số 4)

```

.------.
|=====Client Menu=====|
| 1. Add New Client      |
| 2. Show All Client     |
| 3. Modify Client Info  |
| 4. Remove Client       |
| 0. Back to Main Menu   |
|-----|
Enter your choice: 4
===== All Client =====
No | Full Name | ID | Address
1 | Thanh Tan Phuong | 1811226 | 218 Pham Van Dong, Pleiku, Gia Lai
2 | Hung Quang Nguyen | 1810967 | 46 Le Duan, Quy Nhon, Binh Dinh
3 | Huy Chan Quan | 1810181 | 54 Le Loi, Thu Duc, Ho Chi Minh
4 | jsdkvj lsjdv vjsdlljd | 23 | 34 skldvkl, sdk, sdvkl
5 | Thu Anh Phuong | 1810100 | 218 pvd, pk, gl
=====
-> Enter the No of Client to remove: 4
Client was removed!

.------.
|=====Client Menu=====|
| 1. Add New Client      |
| 2. Show All Client     |
| 3. Modify Client Info  |
| 4. Remove Client       |
| 0. Back to Main Menu   |
|-----|
Enter your choice: 2
===== All Client =====
No | Full Name | ID | Address
1 | Thanh Tan Phuong | 1811226 | 218 Pham Van Dong, Pleiku, Gia Lai
2 | Hung Quang Nguyen | 1810967 | 46 Le Duan, Quy Nhon, Binh Dinh
3 | Huy Chan Quan | 1810181 | 54 Le Loi, Thu Duc, Ho Chi Minh
4 | Thu Anh Phuong | 1810100 | 218 pvd, pk, gl
=====

```

Xóa thông tin khách hàng (xóa thông tin khách hàng số 4)

```

.------.
|==== CREDIT MANAGE MENU ====|
| 1. Client Manage Menu      |
| 2. Loan Manage Menu        |
| 3. Predict Profit          |
| 0. Exit                    |
|-----|
Enter your choice: 2

.------.
|=====Loan Menu=====|
| 1. Add New Loan          |
| 2. Show All Loan         |
| 3. Modify Loan           |
| 4. Remove Loan           |
| 0. Back to Main Menu     |
|-----|
Enter your choice: 2

===== ALL LOANS =====
No | Client's Name | Loan Amount | AIR | NoY | Monthly Payment | Total Payment | Loan Date
1 | Thanh Tan Phuong | 500.6 | 12.0 | 2 | 23.56 | 565.56 | 12/10/2019
2 | Hung Quang Nguyen | 400.8 | 24.0 | 1 | 37.9 | 454.79 | 5/11/2021
3 | Huy Chan Quan | 1000.0 | 15.5 | 4 | 28.08 | 1348.07 | 12/4/2020
4 | Thu Anh Phuong | 500.0 | 12.0 | 5 | 11.12 | 667.33 | 19/10/2021
* AIR = Annual Interest Rate, NoY = Number of Year
Result was rounded in 2 digits
=====

```

Hiện thị toàn bộ các khoản vay

```

.------.
|=====Loan Menu=====|
| 1. Add New Loan      |
| 2. Show All Loan     |
| 3. Modify Loan       |
| 4. Remove Loan       |
| 0. Back to Main Menu |
|-----|
Enter your choice: 1
===== All Client =====
No | Full Name | ID | Address
1 | Thanh Tan Phuong | 1811226 | 218 Pham Van Dong, Pleiku, Gia Lai
2 | Hung Quang Nguyen | 1810967 | 46 Le Duan, Quy Nhon, Binh Dinh
3 | Huy Chan Quan | 1810181 | 54 Le Loi, Thu Duc, Ho Chi Minh
4 | Thu Anh Phuong | 1810100 | 218 pvd, pk, gl
=====
-> Enter No of Client (If Client doesn't exist, enter '0' to add new client): 1
-> Enter annual interest rate: 1
-> Enter number of year: 1
-> Enter loan amount: 100
-> Enter loan date: 29/12/2021
-> Enter loan ID: 12

.------.
|=====Loan Menu=====|
| 1. Add New Loan      |
| 2. Show All Loan     |
| 3. Modify Loan       |
| 4. Remove Loan       |
| 0. Back to Main Menu |
|-----|
Enter your choice: 2
===== ALL LOANS =====
No | Client's Name | Loan Amount | AIR | NoY | Monthly Payment | Total Payment | Loan Date
1 | Thanh Tan Phuong | 500.6 | 12.0 | 2 | 23.56 | 565.56 | 12/10/2019
2 | Hung Quang Nguyen | 400.8 | 24.0 | 1 | 37.9 | 454.79 | 5/11/2021
3 | Huy Chan Quan | 1000.0 | 15.5 | 4 | 28.08 | 1348.07 | 12/4/2020
4 | Thu Anh Phuong | 500.0 | 12.0 | 5 | 11.12 | 667.33 | 19/10/2021
5 | Thanh Tan Phuong | 100.0 | 1.0 | 1 | 8.38 | 100.54 | 29/12/2021
* AIR = Annual Interest Rate NoY = Number of Year

```

Thêm khoản vay mới (khoản số 5)

```

===== ALL LOANS =====
No | Client's Name | Loan Amount | AIR | NoY | Monthly Payment | Total Payment | Loan Date
1 | Thanh Tan Phuong | 500.6 | 12.0 | 2 | 23.56 | 565.56 | 12/10/2019
2 | Hung Quang Nguyen | 400.8 | 24.0 | 1 | 37.9 | 454.79 | 5/11/2021
3 | Huy Chan Quan | 1000.0 | 15.5 | 4 | 28.08 | 1348.07 | 12/4/2020
4 | Thu Anh Phuong | 500.0 | 12.0 | 5 | 11.12 | 667.33 | 19/10/2021
5 | Thanh Tan Phuong | 100.0 | 1.0 | 1 | 8.38 | 100.54 | 29/12/2021
* AIR = Annual Interest Rate, NoY = Number of Year
Result was rounded in 2 digits
=====
-> Enter the No of Loan to modify: 5
Change the info. Type '0' to keep remain
-> Annual Interest Rate:
2
-> Number of Year:
2
-> Loan Amount:
0
Loan was modified!
.-----
|=====Loan Menu=====|
| 1. Add New Loan      |
| 2. Show All Loan     |
| 3. Modify Loan       |
| 4. Remove Loan       |
| 0. Back to Main Menu |
|-----|
Enter your choice: 2
===== ALL LOANS =====
No | Client's Name | Loan Amount | AIR | NoY | Monthly Payment | Total Payment | Loan Date
1 | Thanh Tan Phuong | 500.6 | 12.0 | 2 | 23.56 | 565.56 | 12/10/2019
2 | Hung Quang Nguyen | 400.8 | 24.0 | 1 | 37.9 | 454.79 | 5/11/2021
3 | Huy Chan Quan | 1000.0 | 15.5 | 4 | 28.08 | 1348.07 | 12/4/2020
4 | Thu Anh Phuong | 500.0 | 12.0 | 5 | 11.12 | 667.33 | 19/10/2021
5 | Thanh Tan Phuong | 100.0 | 2.0 | 2 | 4.25 | 102.1 | 29/12/2021
* AIR = Annual Interest Rate, NoY = Number of Year
Result was rounded in 2 digits
=====

```

Sửa thông tin khoản vay (sửa khoản số 5)


```

===== ALL LOANS =====
No | Client's Name | Loan Amount | AIR | NoY | Monthly Payment | Total Payment | Loan Date
1 | Thanh Tan Phuong | 500.6 | 12.0 | 2 | 23.56 | 565.56 | 12/10/2019
2 | Hung Quang Nguyen | 400.8 | 24.0 | 1 | 37.9 | 454.79 | 5/11/2021
3 | Huy Chan Quan | 1000.0 | 15.5 | 4 | 28.08 | 1348.07 | 12/4/2020
4 | Thu Anh Phuong | 500.0 | 12.0 | 5 | 11.12 | 667.33 | 19/10/2021
* AIR = Annual Interest Rate, NoY = Number of Year
Result was rounded in 2 digits
=====
-> Enter the No of Loan to remove: 3
Loan was removed!

.-----
|=====Loan Menu=====|
| 1. Add New Loan      |
| 2. Show All Loan     |
| 3. Modify Loan       |
| 4. Remove Loan       |
| 0. Back to Main Menu |
|-----|
Enter your choice: 2
===== ALL LOANS =====
No | Client's Name | Loan Amount | AIR | NoY | Monthly Payment | Total Payment | Loan Date
1 | Thanh Tan Phuong | 500.6 | 12.0 | 2 | 23.56 | 565.56 | 12/10/2019
2 | Hung Quang Nguyen | 400.8 | 24.0 | 1 | 37.9 | 454.79 | 5/11/2021
3 | Thu Anh Phuong | 500.0 | 12.0 | 5 | 11.12 | 667.33 | 19/10/2021
* AIR = Annual Interest Rate, NoY = Number of Year
Result was rounded in 2 digits

```

Xóa khoản vay (xóa khoản vay số 3)

```

Successfully load Client data.

.------.
|==== CREDIT MANAGE MENU ====|
| 1. Client Manage Menu      |
| 2. Loan Manage Menu        |
| 3. Predict Profit           |
| 0. Exit                     |
|-----|
Enter your choice: 3

===== ALL LOANS =====
No | Loan Amount | Monthly Payment | Loan Date | NoY
1 | 500.6 | 23.56 | 12/10/2019 | 2
2 | 400.8 | 37.9 | 5/11/2021 | 1
3 | 1000.0 | 28.08 | 12/4/2020 | 4
4 | 500.0 | 11.12 | 19/10/2021 | 5
TOTAL PROFIT THIS MONTH: 100.67
Result was rounded in 2 digits
=====

.------.
|==== CREDIT MANAGE MENU ====|
| 1. Client Manage Menu      |
| 2. Loan Manage Menu        |
| 3. Predict Profit           |
| 0. Exit                     |
|-----|
Enter your choice:

```

Dự đoán lợi nhuận