Logo, company name

Description automatically generated

**AACS1084**

**Programming Concepts & Design II**

**Assignment**

**2022/2023**

| Programme | : | Diploma in Computer Science |
| --- | --- | --- |
| Tutorial Group | : | 8 |
| Date Submitted to Tutor | : | 30/4/2023 |

**Team Members:**

| **No** | **Student Name** | **Student ID** |
| --- | --- | --- |
| 1. | Gregory Chia Ming Feng | 2208692 |
| 2. | Aloysius Khoo | 2208707 |
| 3. | Nicholas Lim | 2208641 |
| 4. | Khoo Li Xuan | 2208619 |

| No. | Team Member | Task(s) Allocated | Overall Contribution (%) |
| --- | --- | --- | --- |
| 1. | Gregory Chia Ming Feng | > Stock Management System - Binary File  > Assignment Report | 25% |
| 2. | Aloysius Khoo | > Staff Management System - Binary File  > Assignment Report | 25% |
| 3. | Nicholas Lim Sze Whye | > Sales Management System - Text File  > Assignment Report | 25% |
| 4. | Khoo Li Xuan | > Member Management System - Text File  >Assignment Report | 25% |

**Coursework Declaration**

We confirm that we have read and shall comply with all the terms and conditions of TAR University College’s plagiarism policy.

We declare that this assignment is free from all forms of plagiarism and for all intents and purposes is our own properly derived work.

| Signature | : |  |  |  | Khoo |
| --- | --- | --- | --- | --- | --- |
| Name | : | Gregory Chia Ming Feng | Aloysius  Khoo | Nicholas Lim Sze Whye | Khoo Li Xuan |
| Photo | : |  |  |  |  |
| Date | : | 26/4/2023 | 26/4/2023 | 26/4/2023 | 26/4/2023 |

# Assignment Evaluation Form

| **Student Name** | |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Module handled** | |  |  |  |  |  |
| ***Assessment Criteria*** | ***Marks Awarded*** | | | | | |
| ***Application of Programming Knowledge*** | |  | | | | |
| Structures   * Basic + extra fields, * Use of nesting & arrays * Appropriate usage | **12** |  |  |  |  |  |
| File Processing   * successful processing * Validation of data * Appropriate usage | **12** |  |  |  |  |  |
| Functions   * 4 Basic file fns, * Extra file fns, useful/correct * Approp. menu design & process | **12** |  |  |  |  |  |
| Others   * Use of parameters; correct usage * Subsidiary working functions * Minimized global variables | **12** |  |  |  |  |  |
| Teamwork   * Main menu, logo etc * System/module integration * Report integration/content * Sharing of files, functions * Communication, cooperation | **12** |  |  |  |  |  |
| Program originality, efficiency and readability   * Innovation, uniqueness * Approp. logic structures * Code Readability | **10** |  |  |  |  |  |
| Report (indiv. section)   * Clarity of description * Narrated Screens for all processes (basic + extra) | **20** |  |  |  |  |  |
| Penalty - Late Submission   * 1-3 days late 🡪 -10 * 4-7 days late 🡪 -20   >7 days late => **Total mark = 0** |  |  |  |  |  |  |
| **Subtotal (i)** |  |  |  |  |  |  |
| Presentation   * Organization, explanation * Enthusiasm, preparedness, attitude, understanding * Q&A handling **Subtotal (ii)** | **10** |  |  |  |  |  |
| **TOTAL MARK** |  |  |  |  |  |  |

Note: In addition to the late penalty, marks will also be deducted under the different categories for other reasons, eg. bad programming practices, uncooperativeness, etc.

# 

# Table Of Contents

Cover Page ——————————————————————————— 1

Declaration of Originality ————————————————————— 3

Assignment Evaluation Form ———————————————————— 4

Table of Contents ———————————————————————— 5

Introduction ——————————————————————————— 6

Module Structure Chart —————————————————————— 7

Home/Splash Screen ——————————————————————— 8

System Modules ————————————————————————— 9

# 1. Introduction

The assignment system is a system for a coffee shop, specialised in selling coffee to customers. The system has 4 main modules: sales module, staff module, member module and stock module.

## 

## Staff Module:

Staff module manages the staff info for the coffee shop. The staff info contains functions that it can add a newly hired staff member’s Info, The staff module will add information about the newly hired staff for example like their name, gender, job position, contact number and their salary. And picking a job position will automatically set a salary value for them unless you want to modify it directly using the modify function in the staff module.

**Member Module:**

Member module manages the membership information of all the members that registered. The member module contains functions that allows for adding, deleting and modifying the details of members which includes their member id, membership tier, name, contact number, the date when they joined, the membership expiry date, whether it is still active or expired, the discount percent and the points they accumulated. It also allows user to search for existing members, display all members and generate reports based on filters.

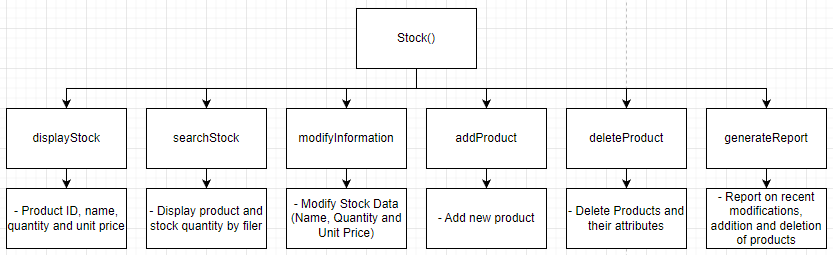
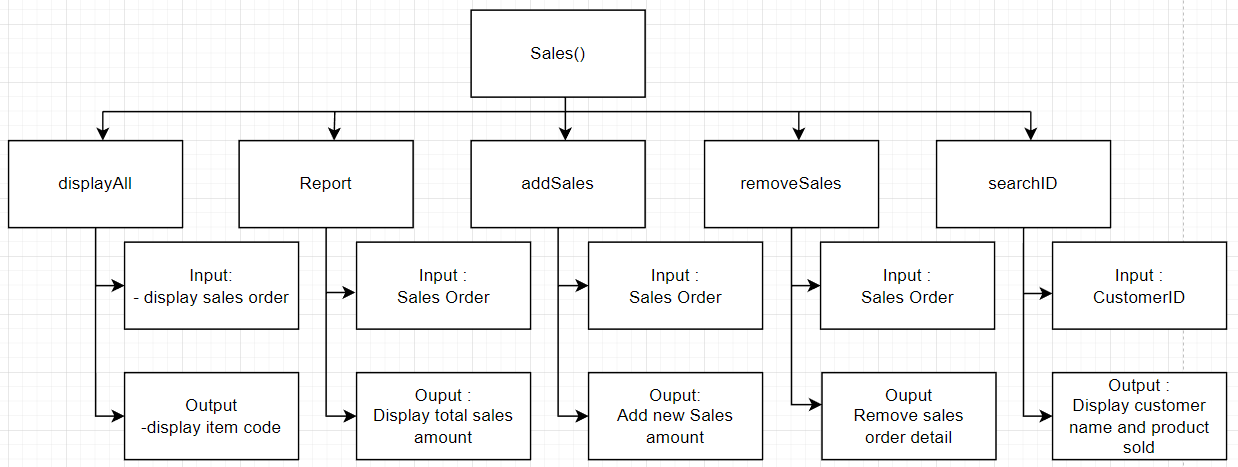
## Sales Module:

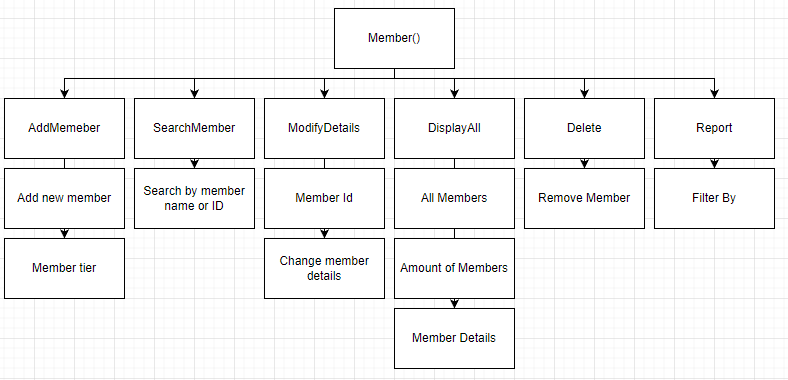
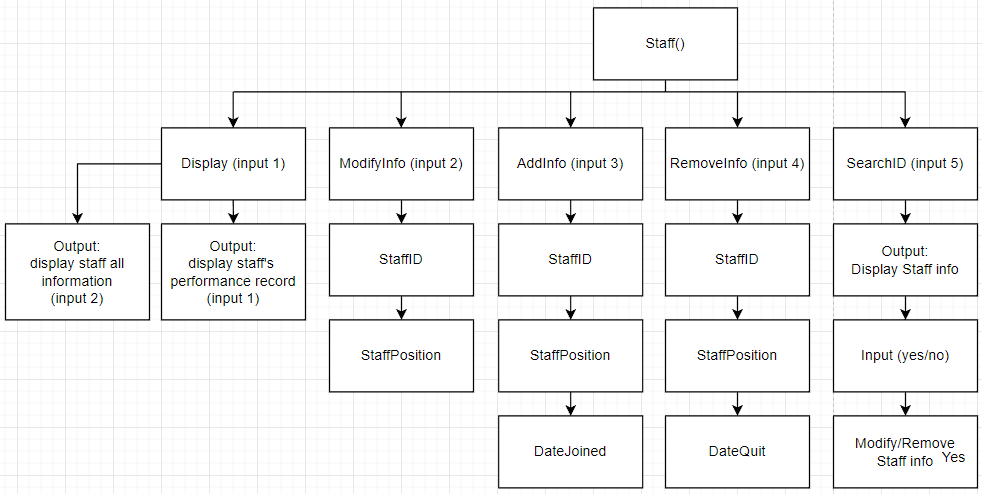
Sales module manages the sales info of the coffee shop. The sales module contains functions that allow us to track the sales data such as the price and the quantity sold. It is also able to add and remove data from the sales, and not only that it can be modified too. The module also generates a report of all the edits to the sales data.

## Stock Module:

Stock module manages the stock of the coffee shop. The stock module contains functions that allow modification to products in stock manually. Example: name of products, quantity of products and unit price of the product. Users may add new products to the stock file as well as delete them. Additionally, users can also display all products currently in stock or search using filters by product ID.

# 2. Overall System / Module Structure Chart





# 3. Home / Splash Screen

# 4. System Modules

## 4.1 Stock Management Module - by Gregory Chia Ming Feng

### 4.1.1 Brief Description

The stock management module manages the stock belonging to the coffee shop. Easy to understand, the module allows users to easily input new products introduced to the store as well as delete them. The system also displays all stock and can even search by filter. Management can modify individual product types in the stock such as its ID, current stock quantity as well as unit price. All additions, deletions or modifications of products are then recorded in a report file which can be generated with a generate report function.

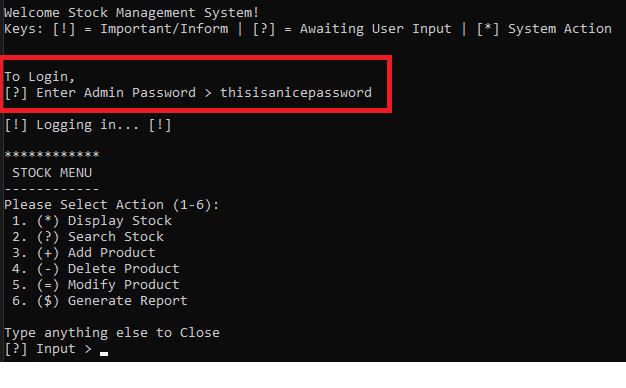
### 4.1.2 Stock Module Home Page

Only authorised personnel are able to use the stock module as it is important that all data remain managed only by professionals. Thus, a password key is required.

Easy-to-understand keys are also given to users, indicating the weight of messages prompted in the stock system.

#### 

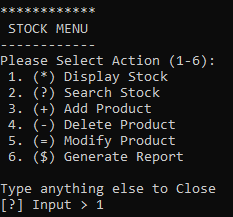
After keying the password, users can input their desired function by choice from numbers 1-6.



### 4.1.3 Existing Data

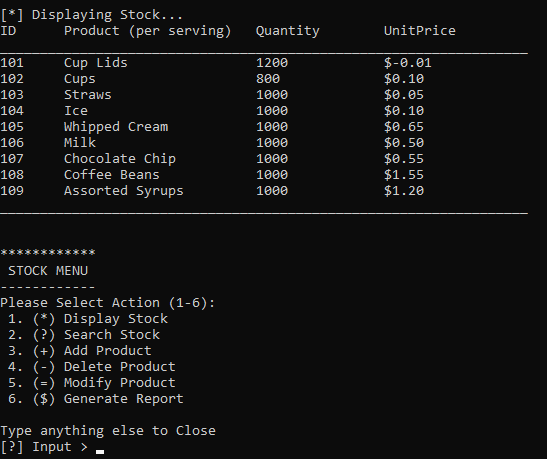
#### 4.1.3.1 Display Stock Function

At the stock menu, users can input ‘1’ to select the **displayStock** function.



Then, a table displaying all product ID, name, quantity and unit price per serving will appear.

Example of displaying stock:

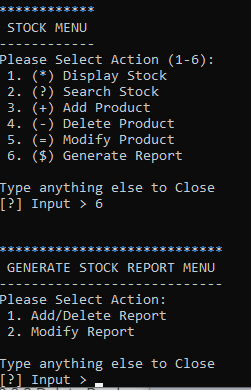


The table clearly displays the product as well as its information to the user.

The display function also loops back to the stock menu.

#### 4.1.3.2 Generate Report Function

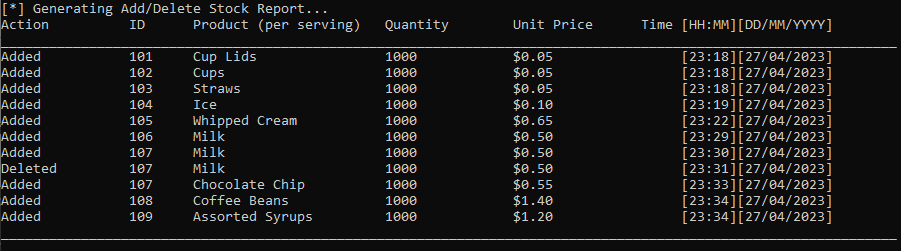
At the stock menu, users can input ‘6’ to open the generate report menu.



Here, users can choose between 2 separate report files.

1. Add/Delete Report
2. Modify Report

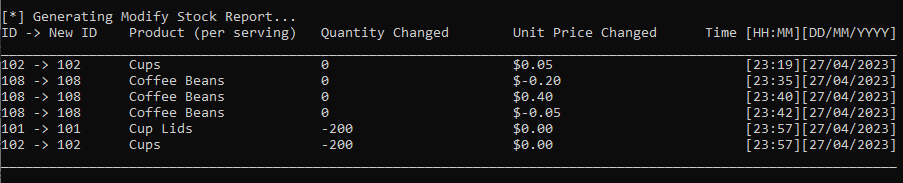
Users can input ‘1’ to call the **generateReportStock** function and display the **add/delete** stock report:



This table clearly displays the action performed by all users as well as the time of the action.

The report displays the products added or deleted as well as the product’s information.

Users can input ‘2’ to to call the **generateReportModify** function anddisplay the **modify** stock report:

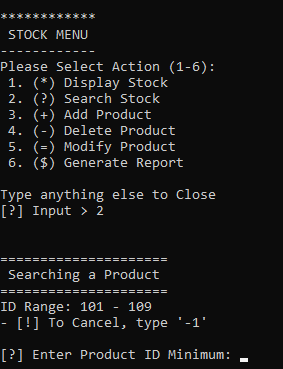


This table clearly displays the modifications of the products modified by the **modifyProduct** function. That includes, the change in product ID, increment/decrease in quantity of the product, increment/decrease in unit price and the time of the modification.

Both report generating functions loop back to the stock menu.

#### 4.1.3.3 Search Product Function

At the stock menu, users can input ‘2’ to call the **searchProduct** function.

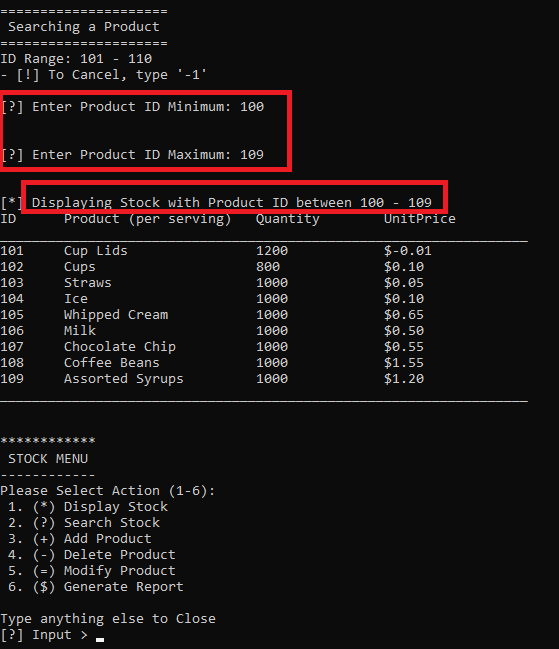


Where users will be asked to enter the product ID of the product they are specifically looking for.

Let’s say in the future of the coffee shop, the manager has to search through hundreds of different ingredients. Thus, a search function will be suitable to find products.

Example, all coffee beans can be distinguished by ID (400 - 499) are all IDs for coffee beans or all different types of syrups can be categorised by IDs (300 - 399).

Example of Search Function:



The function loops then back to the stock main.

### 4.1.4 Performing Transactions

### The planned transactions of the stock system consists of:

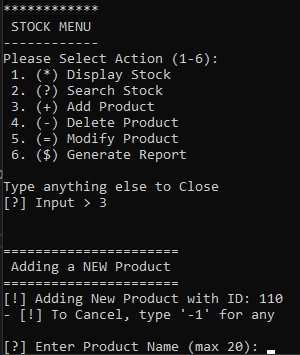
* adding/removing products from the system when necessary
* modifying the product quantity for every purchase of coffee
* updating the report file for the stated actions above
* displaying current stock and information

#### 4.1.4.1 Add Product Function

The product add function allows users to add new products to the stock.bin file. Example, if a new recipe for a new drink is introduced, new ingredients/products are required to be entered into the system to be managed by the user.

**Adding Process:**

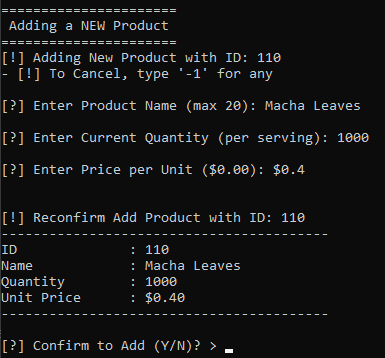
At stock menu, input value 3 to select **addProduct** function. Then, a “adding new product” menu will appear.



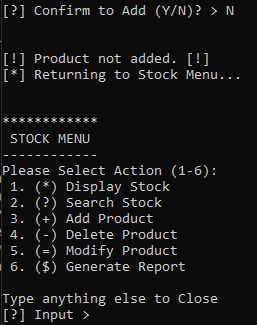
Product ID is automatically generated depending on the ID with the highest value currently in the stock.bin file.

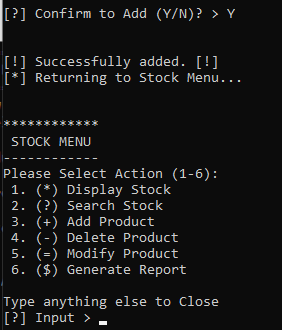
Then, users are required to enter the product's name, current quantity as well as unit price to keep track.

Example:



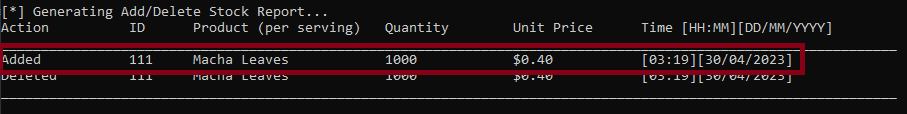
Once a product’s details have been entered, users will be prompted with a reconfirm text as well as a review of the product they are adding. Where they can answer Y for Yes or N for No.

Outcome of Y: Outcome of N (Or any other input):

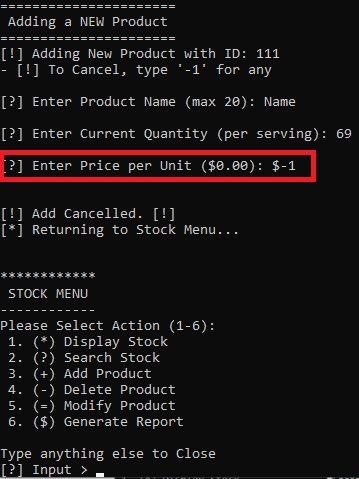
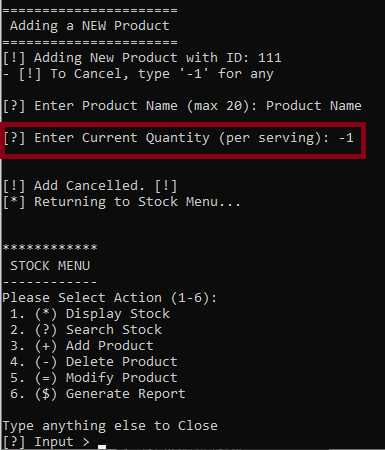


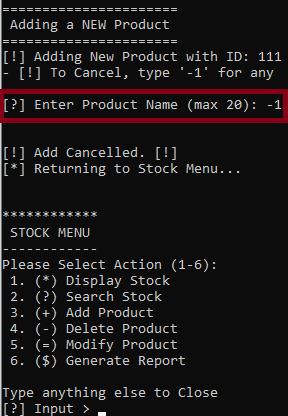
Once a product is successfully added. A report is recorded which can be displayed by the report generation function. Then, it loops back to the stock menu.

Example of report generated by adding new products:



Users may also cancel adding a new product if they changed their minds. By entering ‘-1’ at any of the prompts for entering new product information.

Example:



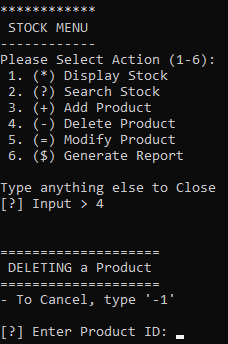
The function loops back to the stock menu if the user decides to cancel.

#### 4.1.4.2 Delete Product Function

The product delete function allows users to delete products from the stock.bin file. In case of the removal of specific recipes from the coffee shop, user can clear the stock from the system.

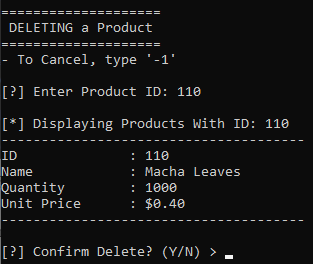
**Delete Process:**

At the stock menu, users can input ‘4’ to select the **deleteProduct** function. Then, a delete product menu will appear.

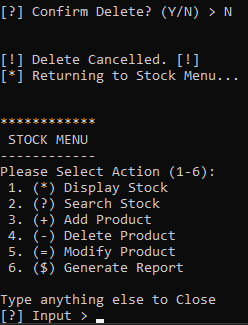


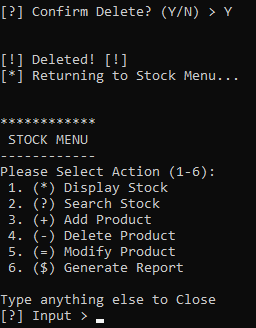
Users are required to enter the ID of the product they desire to permanently remove from the stock system.

Example, Let’s remove the product we added before:



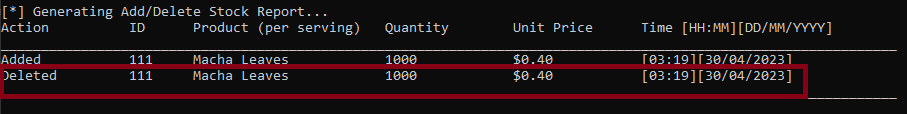
After entering the product ID, users can double check and reconfirm whether or not they actually want to delete the product they’ve selected by reviewing the product information displayed and the prompt with the choice between Y for Yes or N for No.

Outcome of Y: Outcome of N (Or any other input):



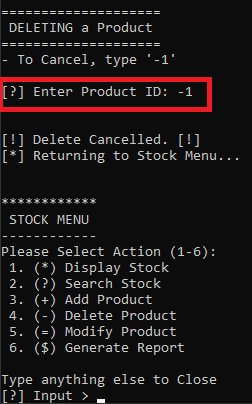
Once a product is successfully deleted. A report is recorded which can be displayed by the report generation function. Then, it loops back to the stock menu.

Example of report generated by deleting products:



Users may also cancel deleting a product if they changed their minds. By entering ‘-1’ at the prompt requesting for product ID.

Example:



Cancelling a product deletion will loop back to the stock menu as well.

#### 

#### 4.1.4.3 Modify Product

The product modification function allows users to change the ID, quantity as well as unit price of the selected product. This is so that stock managers can update the quantity manually.

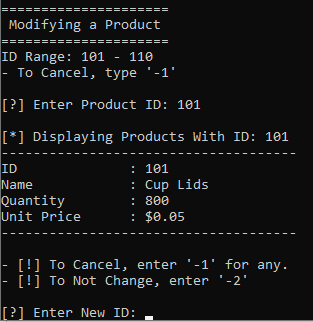
**Modify Process:**

At the stock menu, users can input ‘5’ to select the **deleteProduct** function. Then, a delete product menu will appear.

****

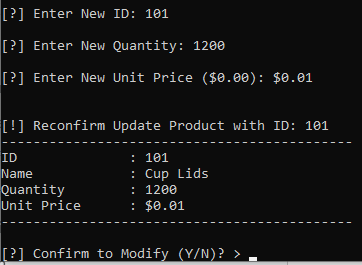
Users are then required to input the product ID of the product they desire to modify.

Example, Let us modify product with ID 101:



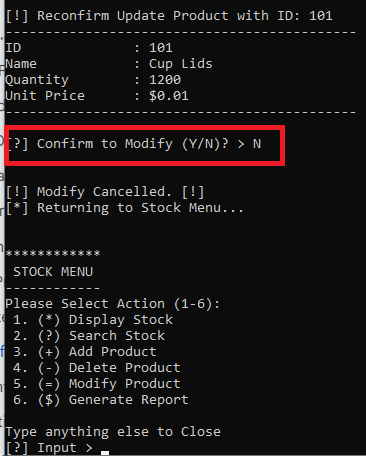
After entering the product ID to be modified, users may enter information such as new ID, new Quantity and new Unit Price.

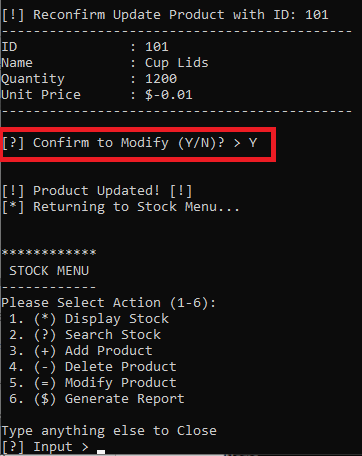
Example, Let’s update the quantity to 1200 and unit price to $0.01:



Once a product’s desired update information has been keyed in, the function will ask the user to reconfirm the product with its updated information.

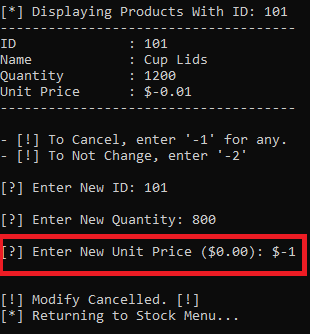
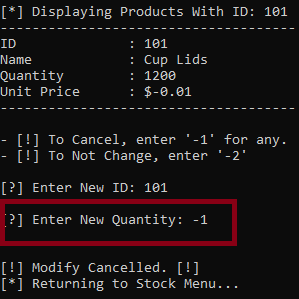
Here, users can input Y for Yes or N for No to confirm the modification.

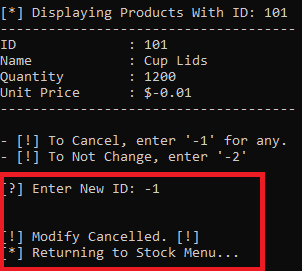
Outcome of Y: Outcome of X (Or any other input)



After the choice is made, the function will loop back to the stock menu.

Users can also input ‘-1’ on any of the prompts to cancel the modification of a product.

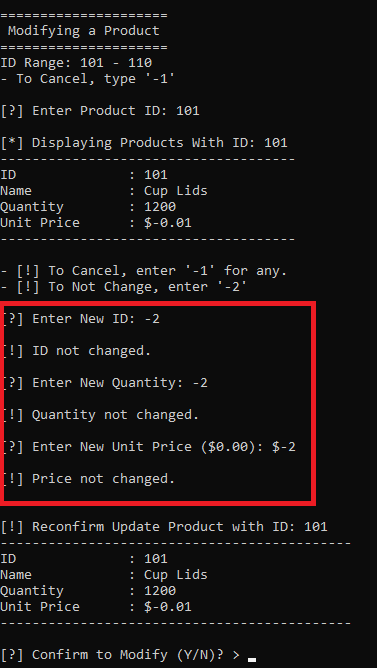
Example:



The function loops back to stock menu after being cancelled.

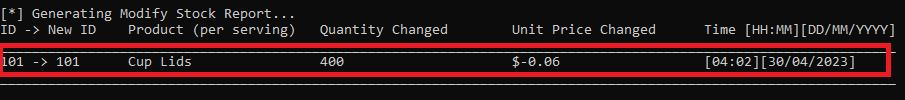
Users can also input ‘-2’ on any of the prompts to leave the current product information the modification of a product **unchanged**

Example:



Once a product is successfully modified, a report is recorded which can be displayed by the report generation function. Then, it loops back to the stock menu.

Example of report generated by modifying products:



This is the report of the product we modified above.

## 

## 

## 

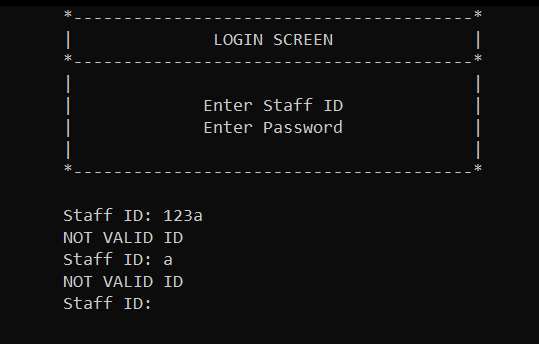
## 4.2 Staff Management Module (ALOYSIUS KHOO)

### 4.2.1 Brief Description for staff module

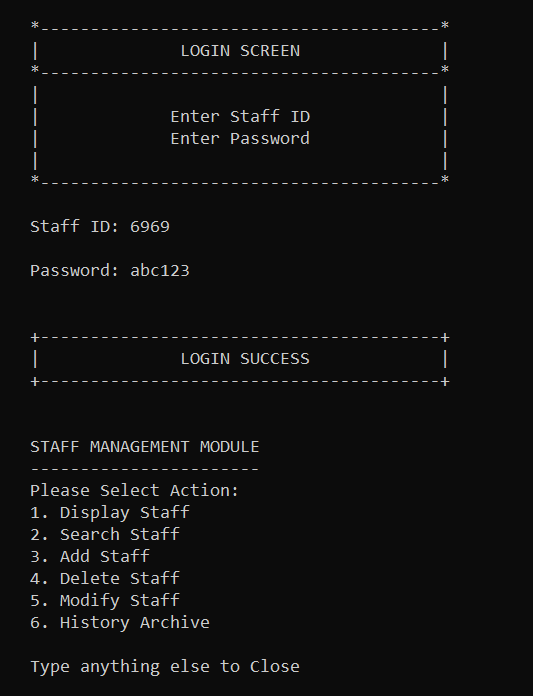
Staff module contains 6 main functions, they are display, search, add, delete, modify staff Info, and finally history archive, This module can only be access by using an admin ID and password to make sure that no other staff member can easily modify the staff info or it will disrupt the work flow or giving out wrong staff info. And for admin this function is pretty intuitive to use as it has simple input and the UI can send out error messages that help guide the operator through the staff module.

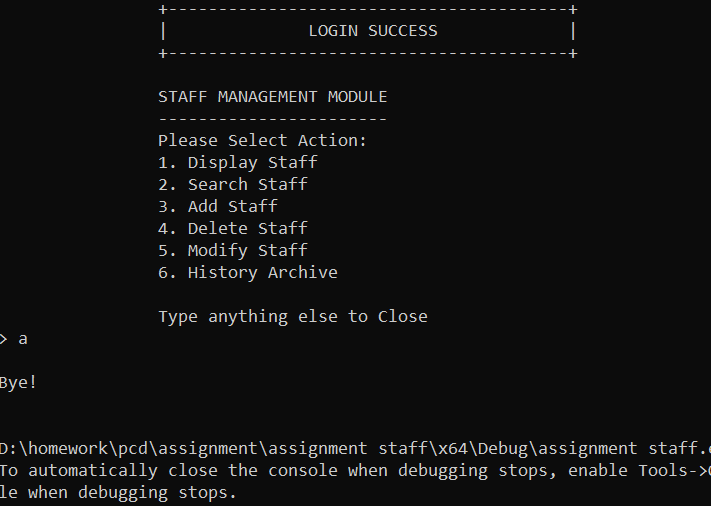
### 4.2.2 Login Screen

First we have the login function, it is an added security to make sure no one other than admin can access the module as if others modify the info haphazardly will affect the shop.

  
As you can see if the ID is invalid the login function won't allow entry.

Now we enter the correct ID and password to proceed to the select screen. The select screen over here can let us choose which function do you want to enter. Pressing the number from 1 to 6 accordingly will allow the user to use them. And pressing anything other than 1 to 6 will result in exit



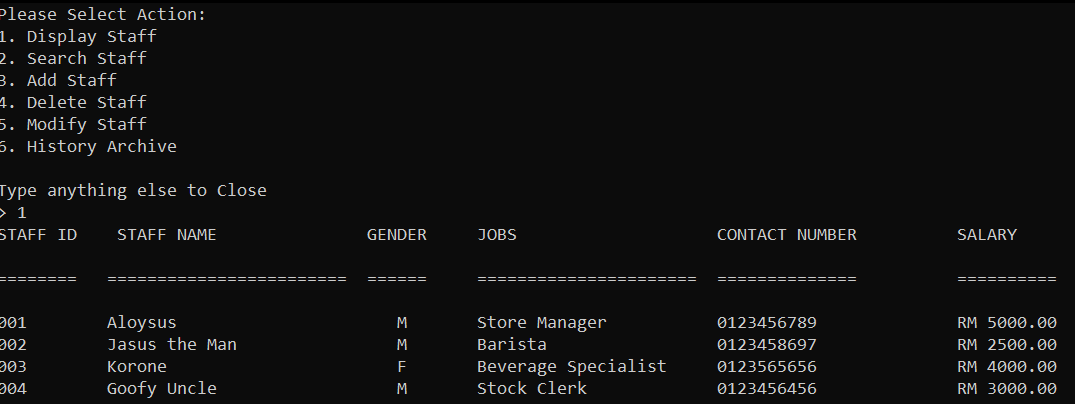


### 

### 

### 4.2.3 Display Screen

When you enter value 1 into the select screen, it will automatically display all the staff and their info, afterward it will loop back to the select screen again.

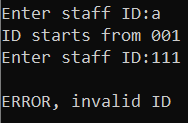


### 4.2.4 Search Staff

Entering value 2 in the select screen will enter the search function, type the id number that you want to search the info on.

### 

If you enter anything that is not digit it will show an error message, and typing an ID number that doesn't exist will also result in an error message.



### 

### 4.2.5 Add Staff

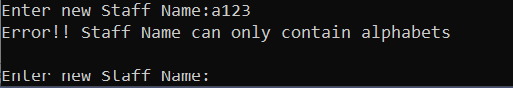
First adding staff we have to add their name and gender

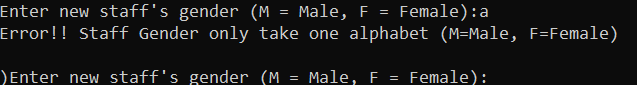


After that we can choose from 8 jobs which one the staff should assign with, and each job has it’s own set salary unless you want to modify it in the modify function and after that enter the staff's phone number and an ID will be automatically generated for the staff.

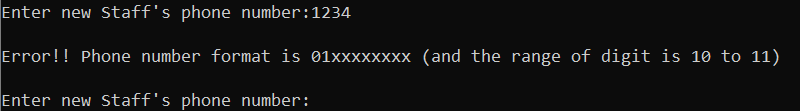


These are the validation from the user input and it’s error message



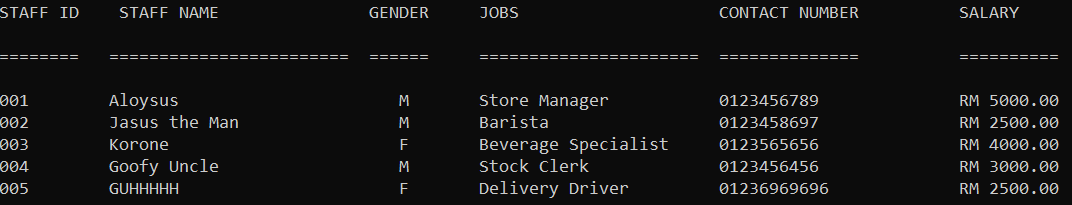




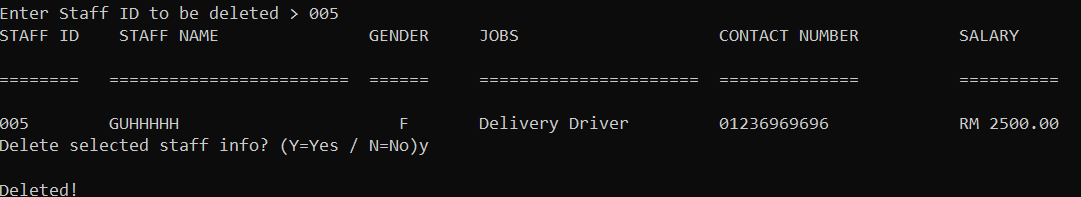


### 4.2.6 Delete Staff

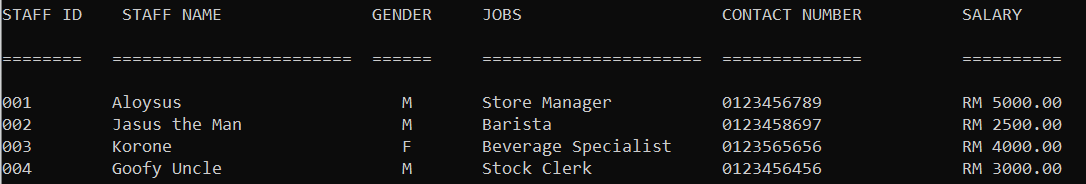
Enter value 4 in the select screen will enter the delete function, enter the ID number of the staff info that you want to delete, for example ID 005 will be the one user wants to delete.



Entering the staff ID 005 and press y to confirm delete

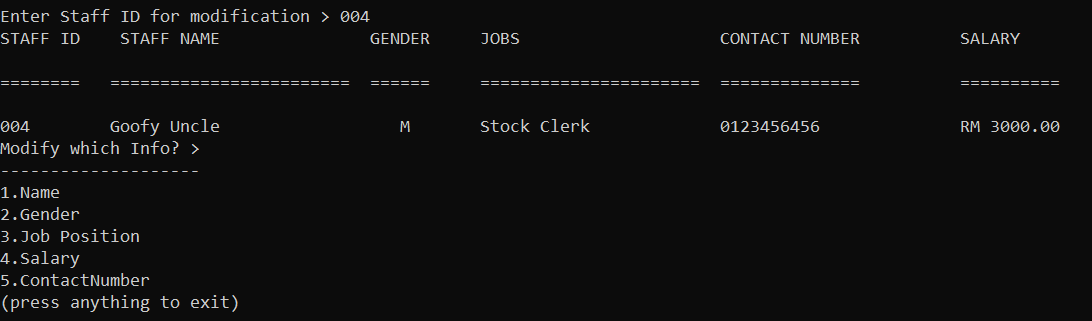


When you display again the staff info should be deleted.



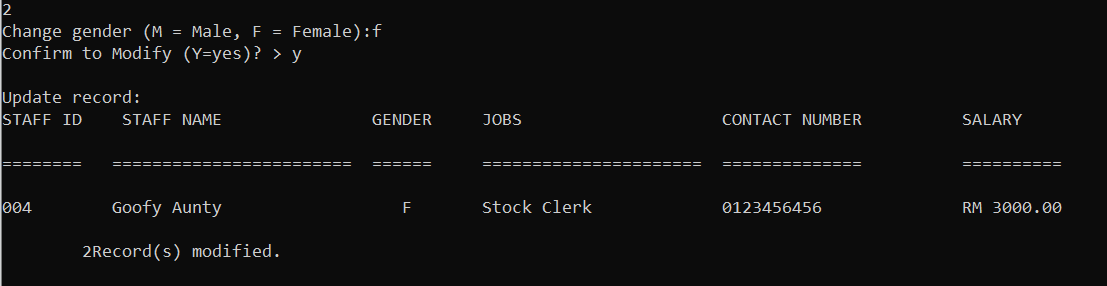
### 4.2.7 Modify Staff

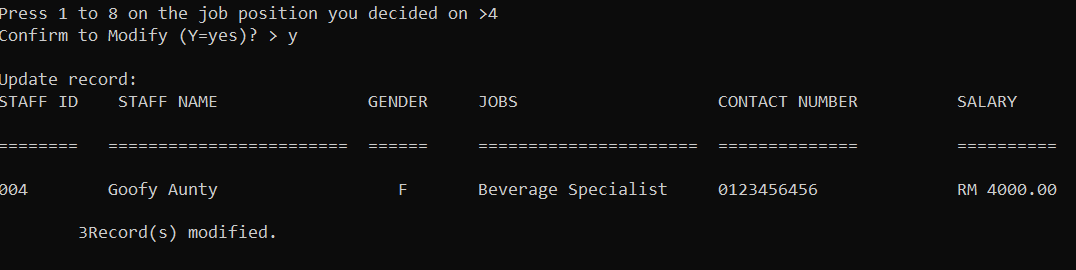
Entering value 5 in the select screen will enter the modify function, for example if a user wants to modify staff info for ID 004, they first have to search it



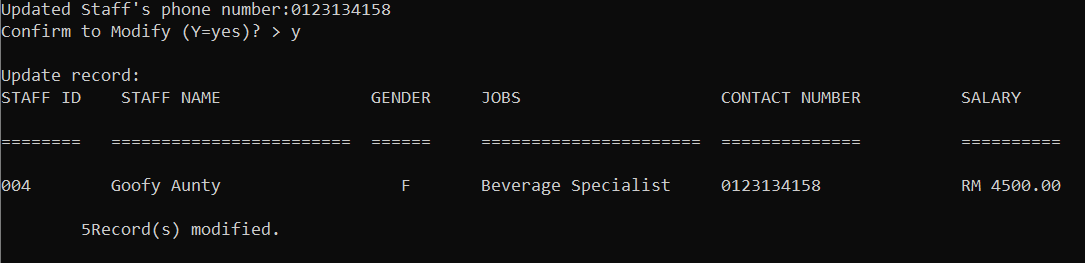
And now you have the option to modify which aspect by typing value from 1 to 5 as the option lay out, the next couple screenshot will be showing me modifying each aspect of the staff info.



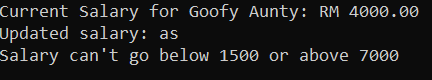






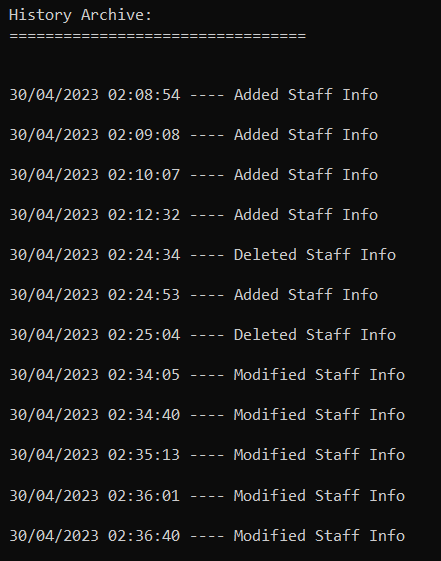


Most of the error message are the same as Add Staff function, this is the only new one



### 4.2.8 History Archive

History Archive will record all the date and time when an activity occur in the staff management module



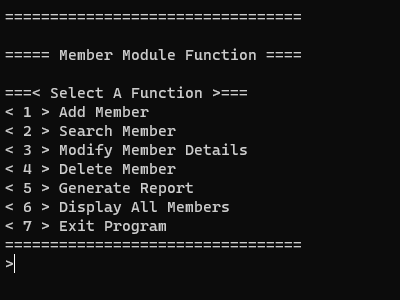
This is the recent activity that has been recorded into this function

## 4.3 Members module

#### 4.3.1 Brief Description

There are 6 functions to the member module, which includes adding, searching, modifying, deleting, generating reports and displaying all members.

#### 4.3.2 Menu Screen

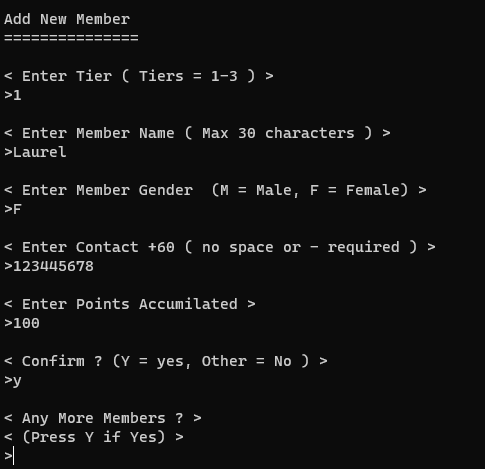


From the menu screen, 7 options could be made, with each option on 1-6 corresponding to their respective functions and option number 7 closing the program altogether. Selecting any outside of the 7 options will result in an error message to be displayed instead and being returned to the member menu screen instead.

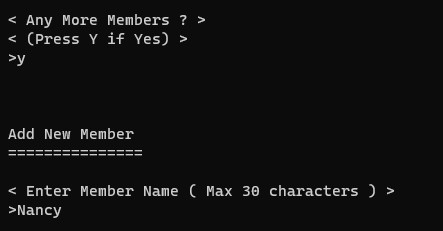
#### 4.3.3 Add Member

When Option 1 is selected, the user will be brought to the add member function.

The user will then be prompted to insert the member’s data such as membership tier, their name, gender, contact number and the points they accumulated. The system will automatically generate their member id and the day they joined depending on the current day and depending on the membership tier, determine the member’s expiry date and the percentage of discount they get to enjoy. At the end, there will be prompts to confirm adding the member and to ask if there are more members to add.



Should the user press “y”, the program will loop and allow another member to be added.



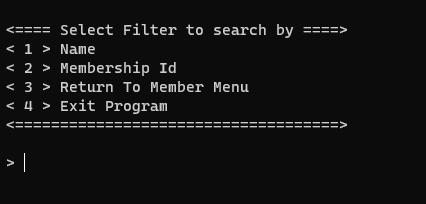
should there be no more to add, it will display how many members were added and the program will return to the menu.



#### 4.3.4 Search Function

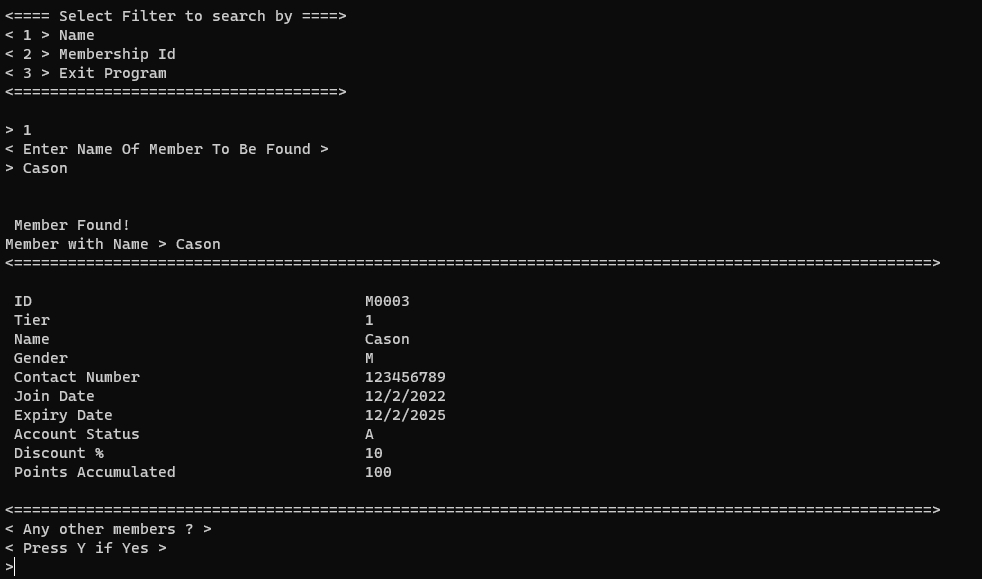
When Option 2 is selected, the user will be brought to the search member function.

The user is then prompted with 4 options, 2 ways to search for the members, by name or by member id, to return to the member menu or to exit the program.

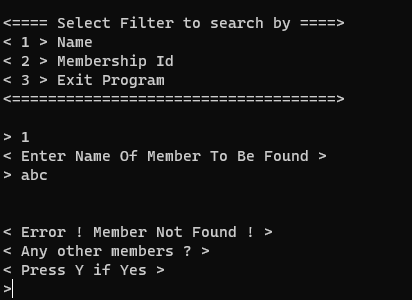


#### 4.3.4.1 Search By Name

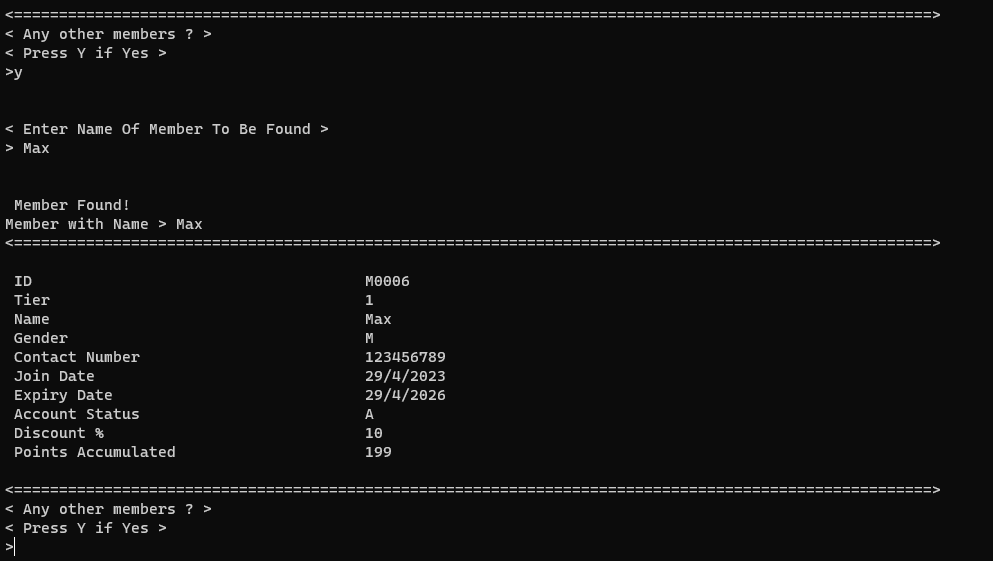
The user is prompted to enter the member’s name that they are searching for



Should the member not be found, a error message will be displayed instead

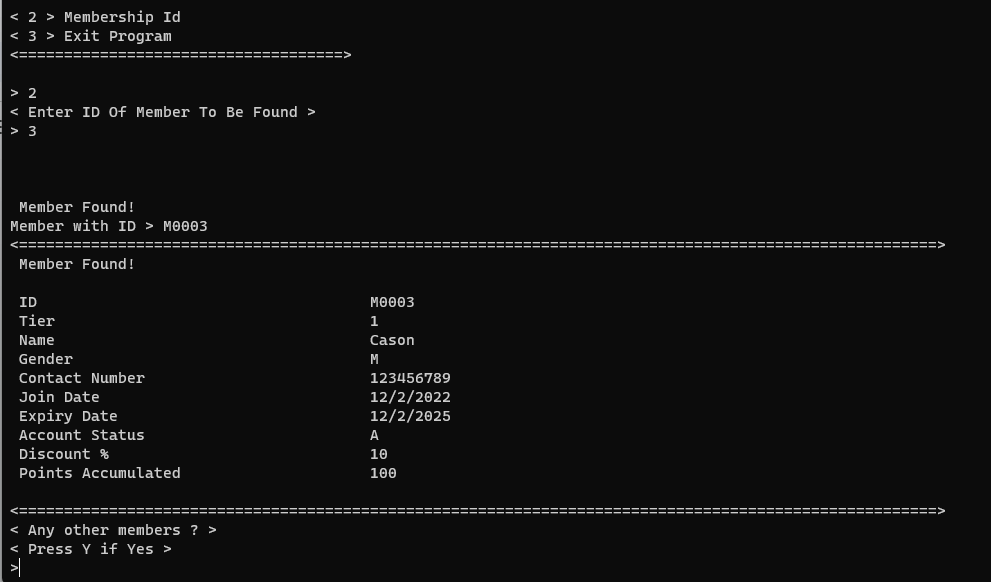


The user will be prompted if they wish to continue searching for other members by pressing ‘Y’, the user will be prompted to enter the members name again.

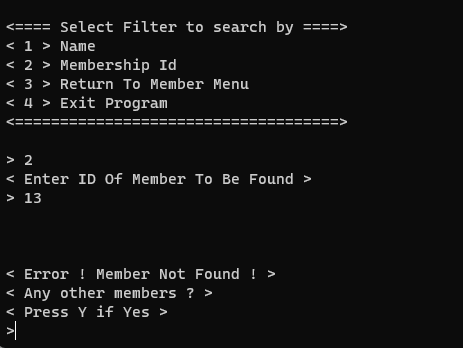


#### 4.3.4.2 Search By Member ID

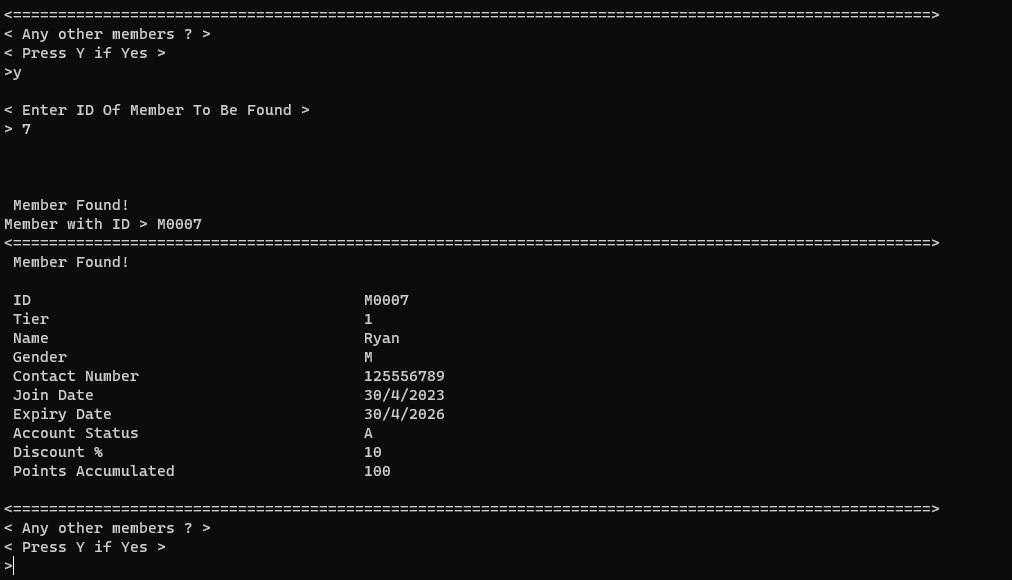
The user is prompted to enter the member’s ID that they are searching for



An error message will be displayed should the member not exist or should the user input any alphabets



The user will be prompted if they wish to continue searching for other members by pressing ‘Y’, the user will be prompted to enter the members name again.

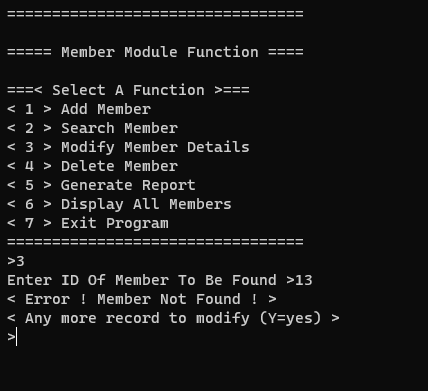


#### 4.3.5 Member Detail Modifications Function

When Option 3 is selected, the user will be brought to the modify member details function.

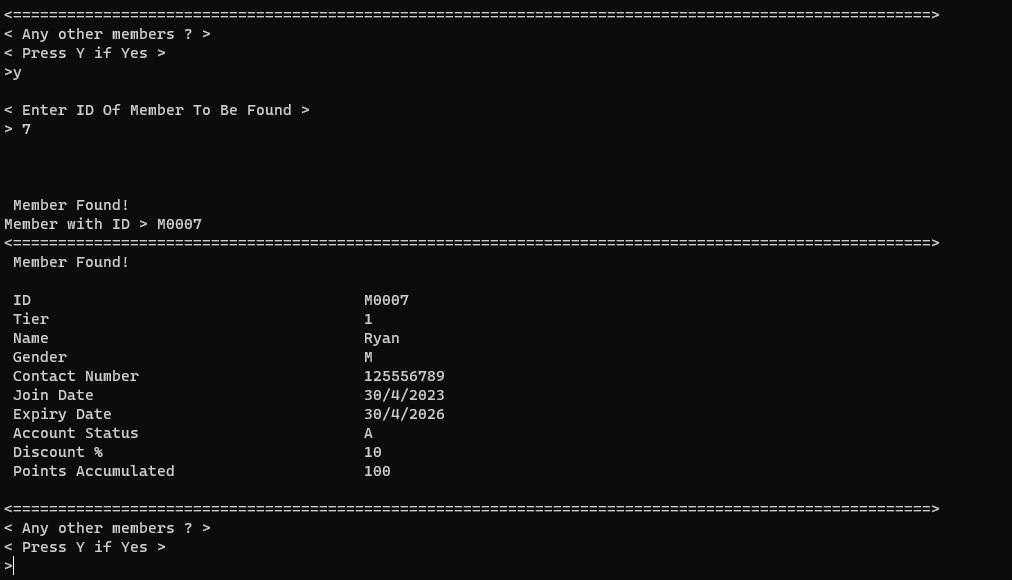
The user is first prompted to search for the member whose details are to be modified by searching with their id should the member not be found, an error message will be displayed.

The user will also be asked if they wish to proceed with the modification, and if there are any more to be modified. At the end There will be a counter showing how many were modified.



The user will be then prompted if they wish to modify any more records, ‘Y’ if yes.

The user will then be prompted to enter the id for the member’s record they wish to modify again.



Should the member be found, the user will be prompted to enter the member’s updated tier, name, gender, contact number, and points accumulated.

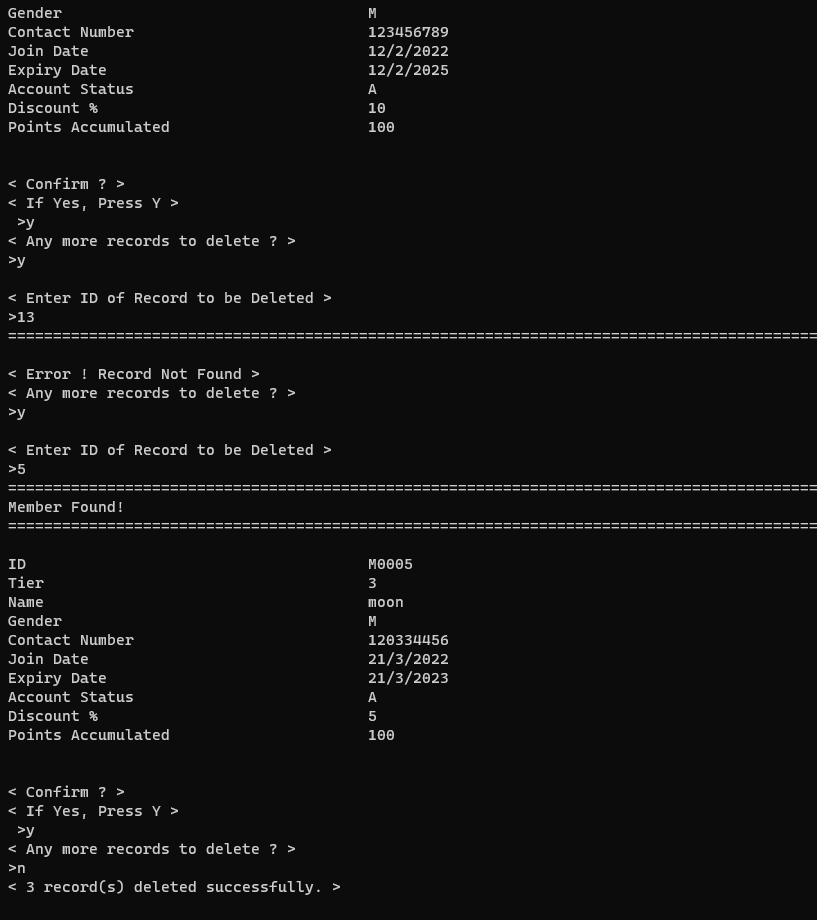
#### 4.3.6 Member Delete Function

When option 4 is selected, the user is brought to the delete member function.

The user can then choose between a single selected member by member ID or to delete all members.

#### 4.3.6.1 Delete a selected member

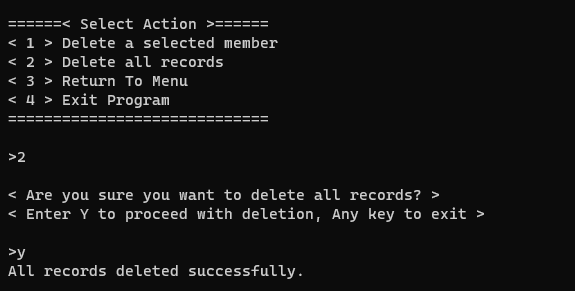
The user is prompted to select to delete by first finding the member by their ID, should the member not be found, a message is displayed instead. If the member is found, the user will be asked if they wish to proceed with the deletion. The user will then be asked if there are any other records to delete, should there be any. When the user wishes to stop, it will display how many member records did the user delete.



At the end, the program will display how many members were deleted.

4.4.2 Delete all members

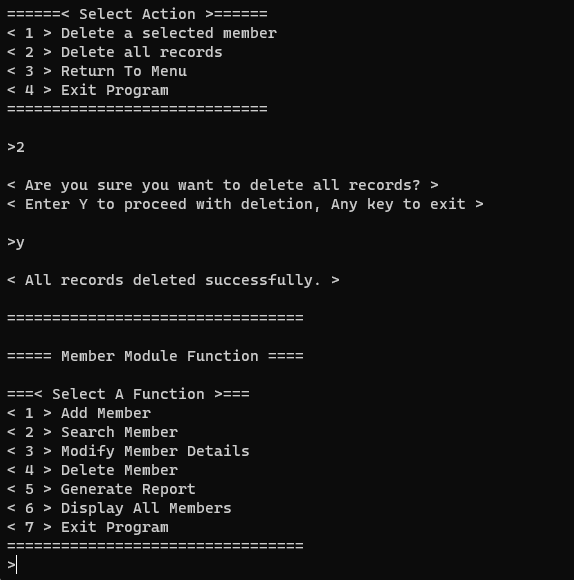
The user will be prompted if they wish to continue. Should they, the whole file will be wiped clean.



4.5 Generate Report

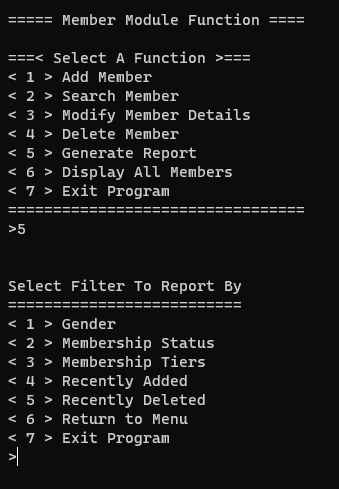
#### 4.3.6.2 Delete All Members

When this function is selected. The user will be prompted if they wish to delete all records, should they proceed, the program will then delete all records from the member file, clearing all data. The program will then tell the user that all records have been deleted.



#### 4.3.7 Generate Report

When option 5 is selected, the user will be brought to the generate report function menu.

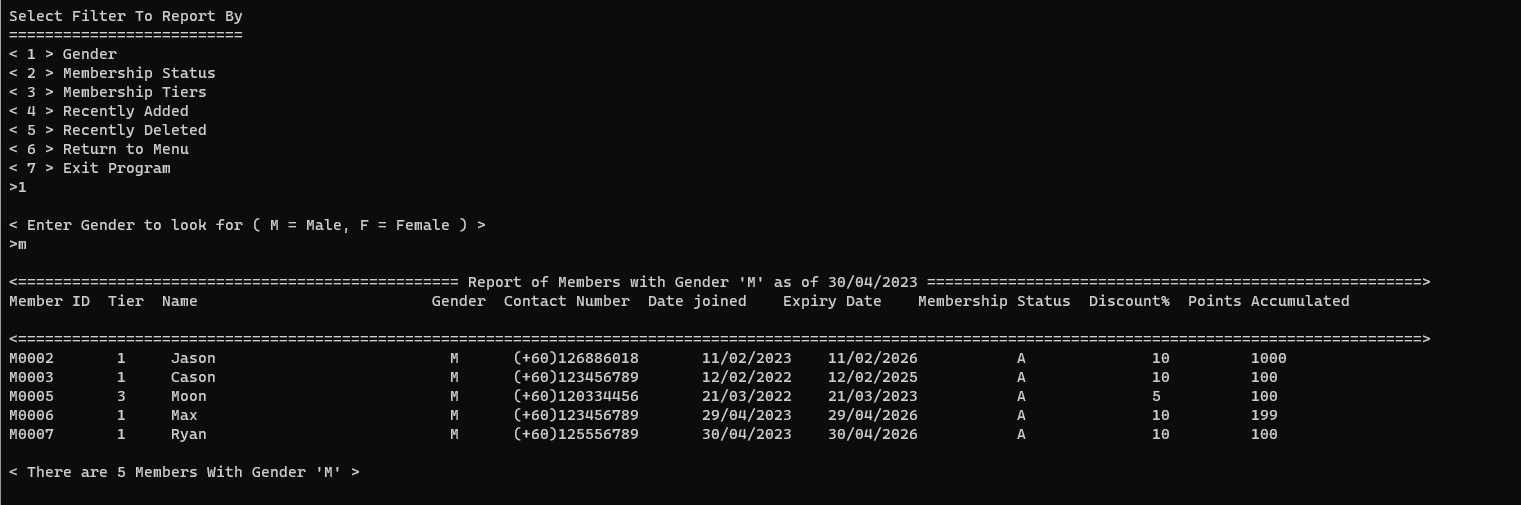


The user will then be prompted to generate a report based on the filter provided as shown above or the user can leave and return to the member menu or to leave the program altogether.

#### 4.3.7.1 Generate report by gender

When selected, the user will then be prompted to enter the gender they wish to filter the report by.

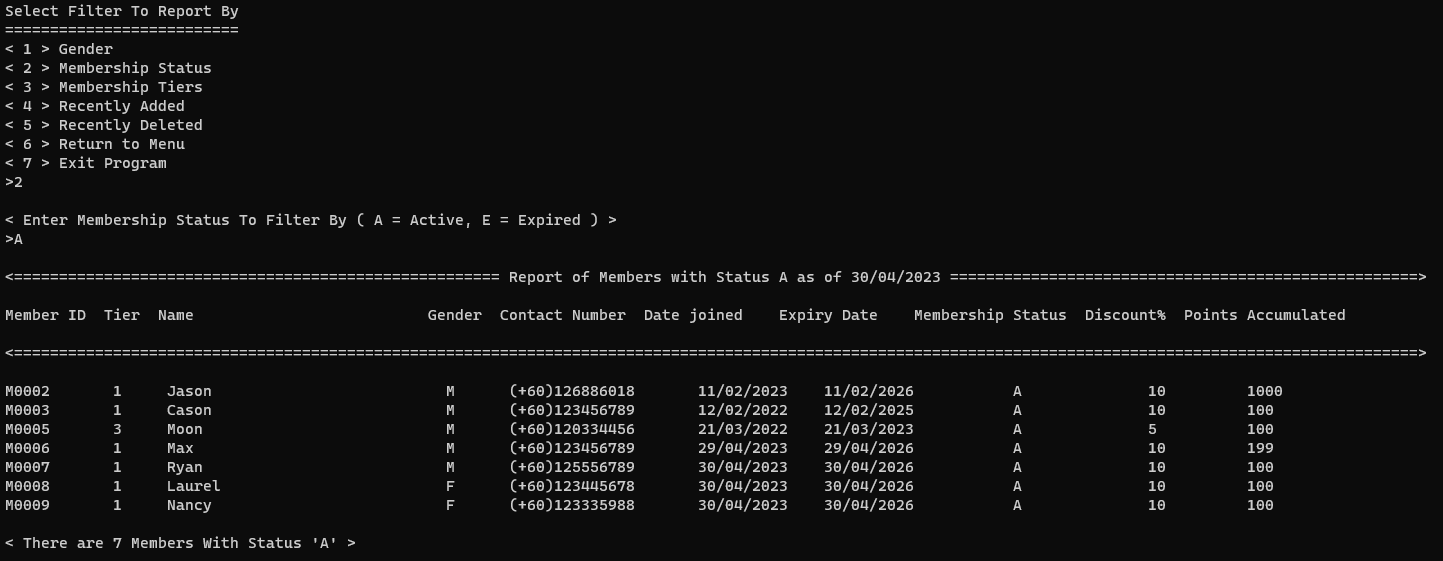
The program will then generate the report based on the selected gender and display all with the gender the user entered. At the end of the report, the program will show how many users were found using the filter. An example will be provided by the picture below with the filter ‘M’ for Male.



#### 4.3.7.2 Generate report by Membership Status

When selected, the user will then be prompted to enter the membership status they wish to filter the report by.

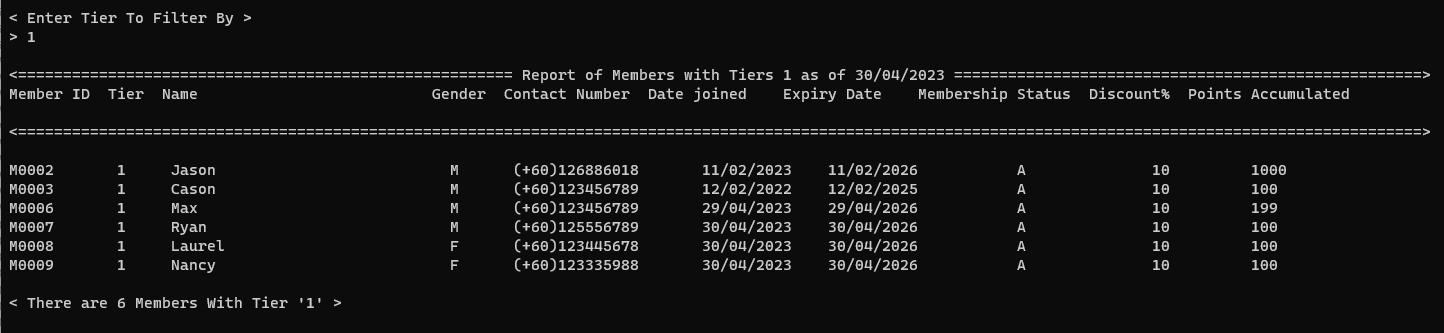
The program will then generate the report based on the selected status and display all with the membership status the user entered. At the end of the report, the program will show how many users were found using the filter. An example will be provided by the picture below with the filter ‘A’ for Active.



#### 4.3.7.3 Generate report by Membership Tier

When selected, the user will then be prompted to enter the membership tier they wish to filter the report by.

The program will then generate the report based on the selected status and display all with the membership tier the user entered. At the end of the report, the program will show how many users were found using the filter. An example will be provided by the picture below with the tier filter 1.



#### 4.3.7.3 Generate report by Recently Added

When selected, the program will the show all recently added members

At the end of the report, the program will show how many users were recently added.

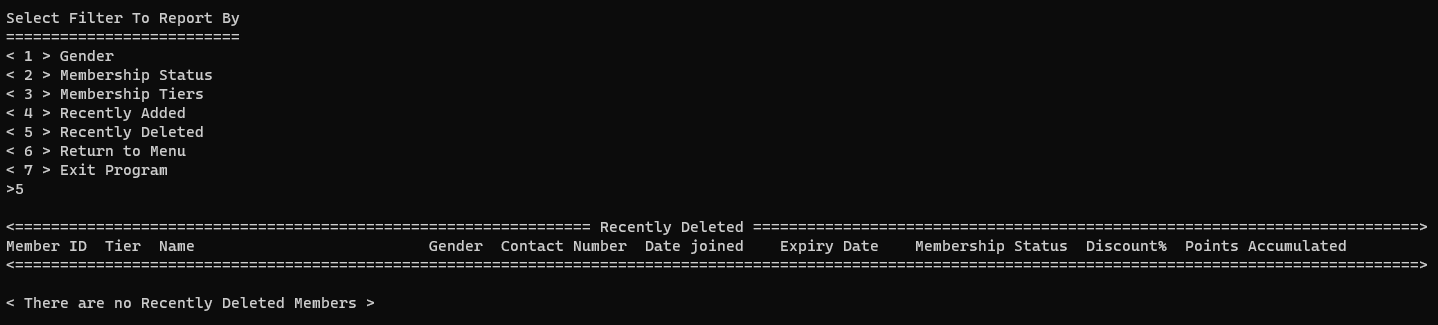


#### 4.3.7.4 Generate report by Recently Deleted

When selected, the program will the show all recently deleted members

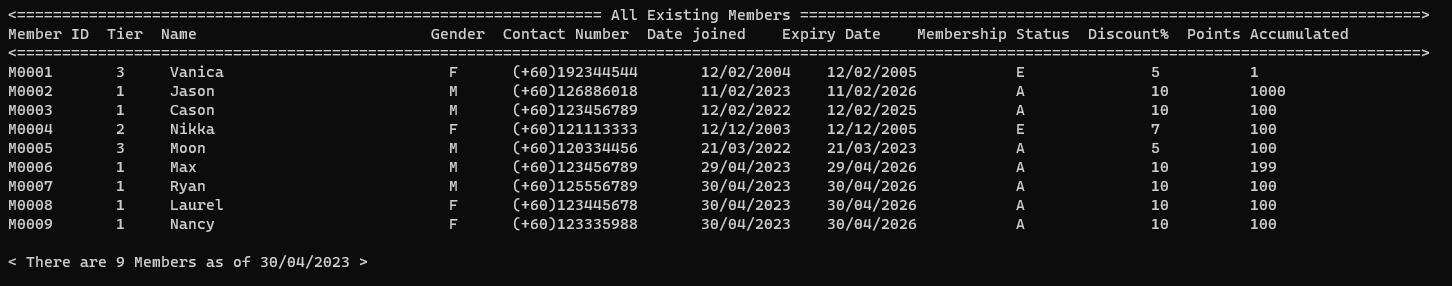
At the end of the report, the program will show how many users were recently deleted.

Should there be none, the program would then prompt, telling the user that there are no recently deleted members.



#### 4.3.8 Display All Members Existing Members

When option 6 is selected, the program will then display all existing members in the file.



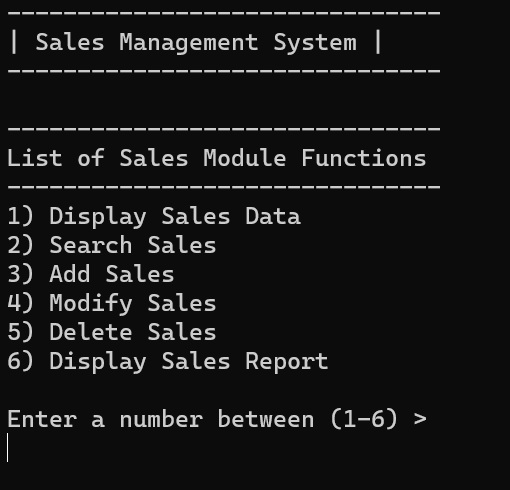
## 4.4 Sales Management Module - by Nicholas Lim Sze Whye

### 4.4.1 Brief Description for sales module

The sales module includes 6 functions which are displaying, searching, add, modify, delete and generating sales reports. It helps keep track of the sales which provide increased efficiency to the business and also accurate data.

#### 4.4.2 Sales Main Menu

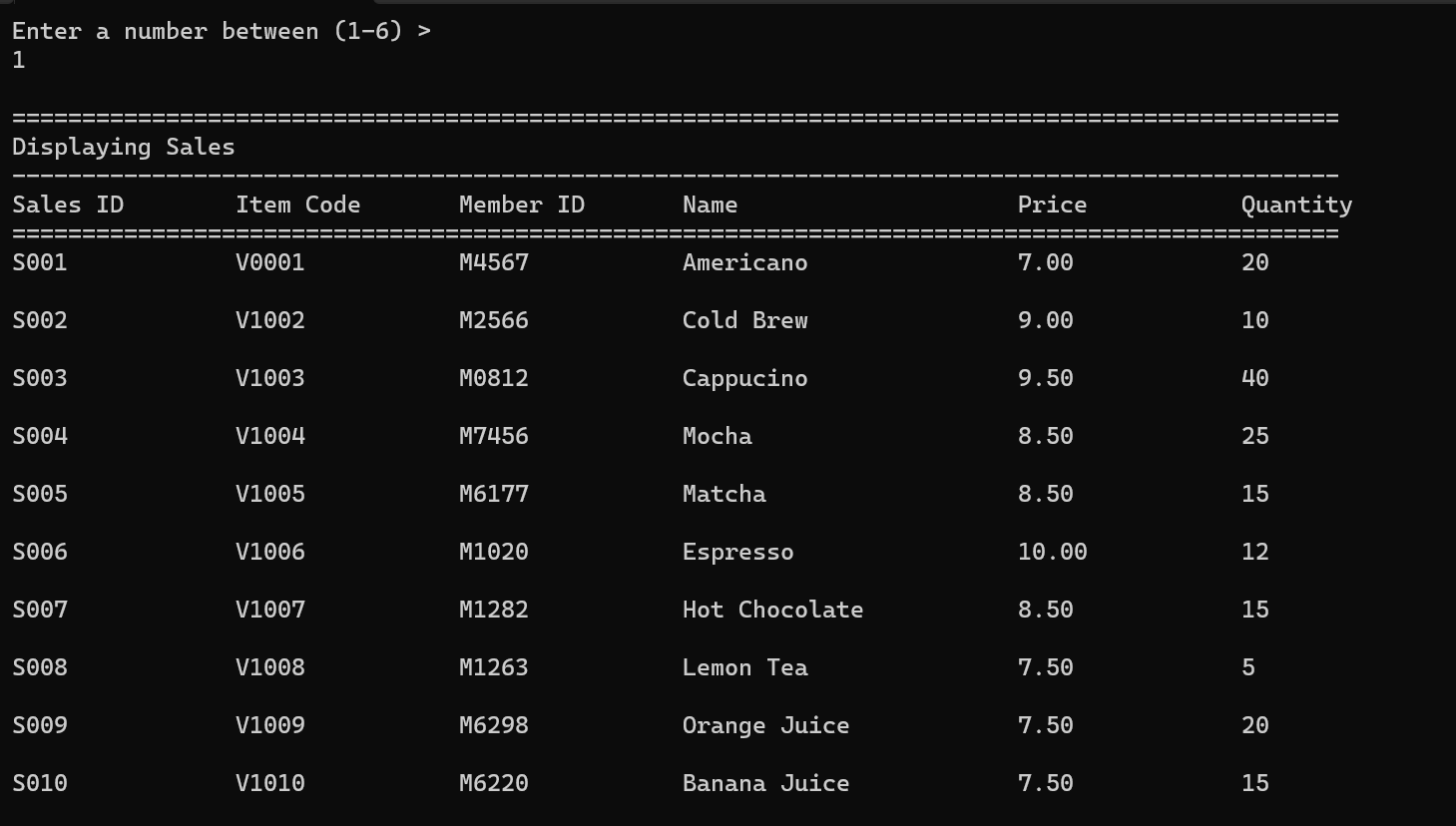
The main menu displayed 6 types of functions where you can select from.



#### 4.4.2 Display Sales Data

When users enter the value 1, it will display the sales data in a table form neatly. The table has displayed 6 data fields which are sales ID, item code, member ID, name, price, and quantity.

Example of displaying sales data:

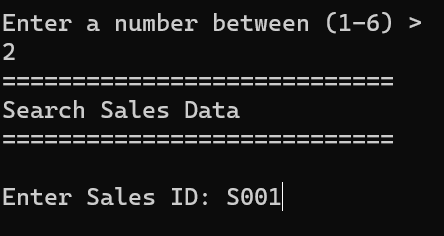


The table above displays the sales information to the user.

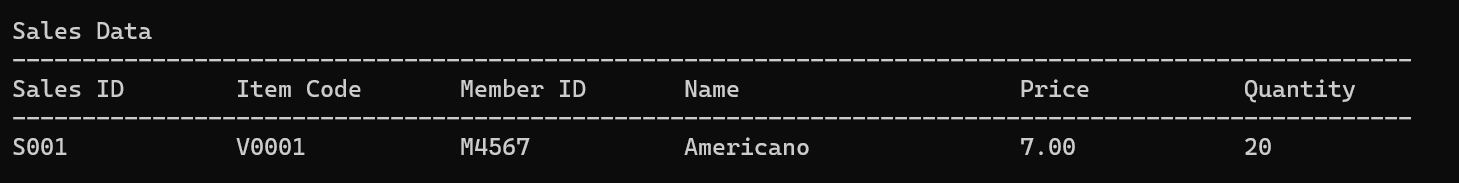
#### 4.4.3 Search Sales Function

When the user enters the value ‘2’ in the main menu, it will display the search function. Here in this function users are allowed to search sales data.

In order to do so, the user must first enter the sales ID.

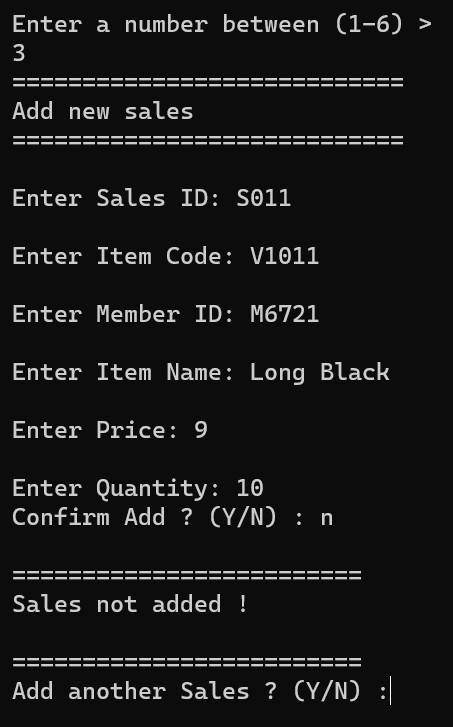


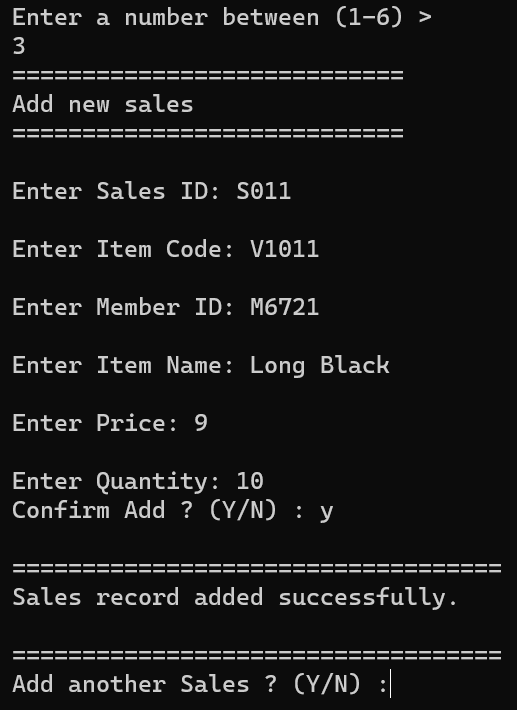
After entering the sales ID, the sales data will be displayed neatly as shown below.



#### 4.4.4 Add Sales Function

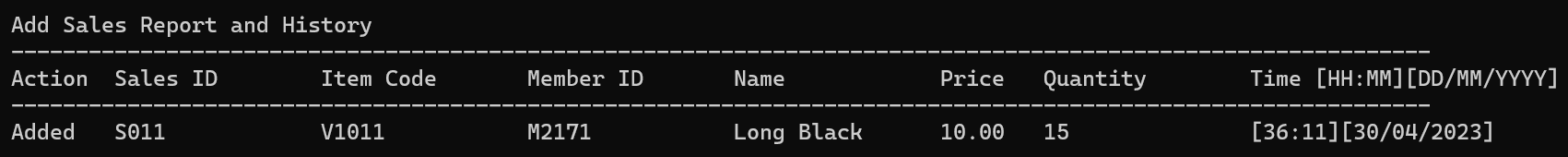
When the user enters the value 3, it will display the add function. Here users can add in new sales by displaying the requirements given. After filling in the data, the users are to confirm their data in order to save it.

When the user confirms their data and enters ‘Y’, it will save the sales record into the file, while if the user enters ‘N’, the sales record won’t be saved into the file. An option to add another sales is provided.



Once the sales data is successfully added. A report is recorded which can be displayed by the sales report function.

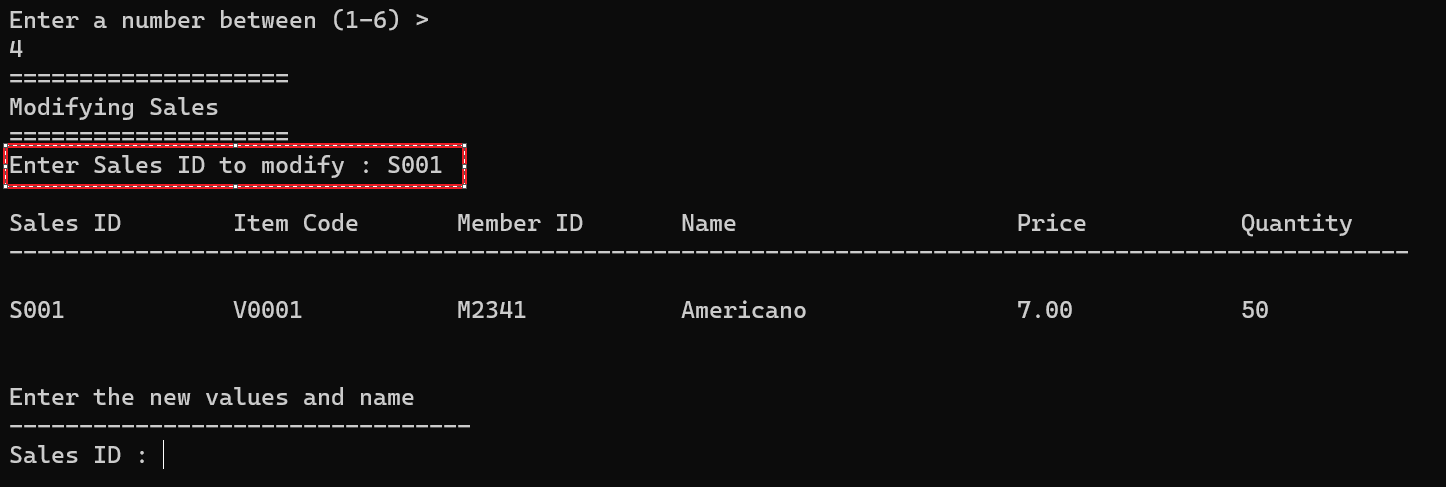
Here is an example of the sales data added and saved into the sales report file with the time and date displayed too.



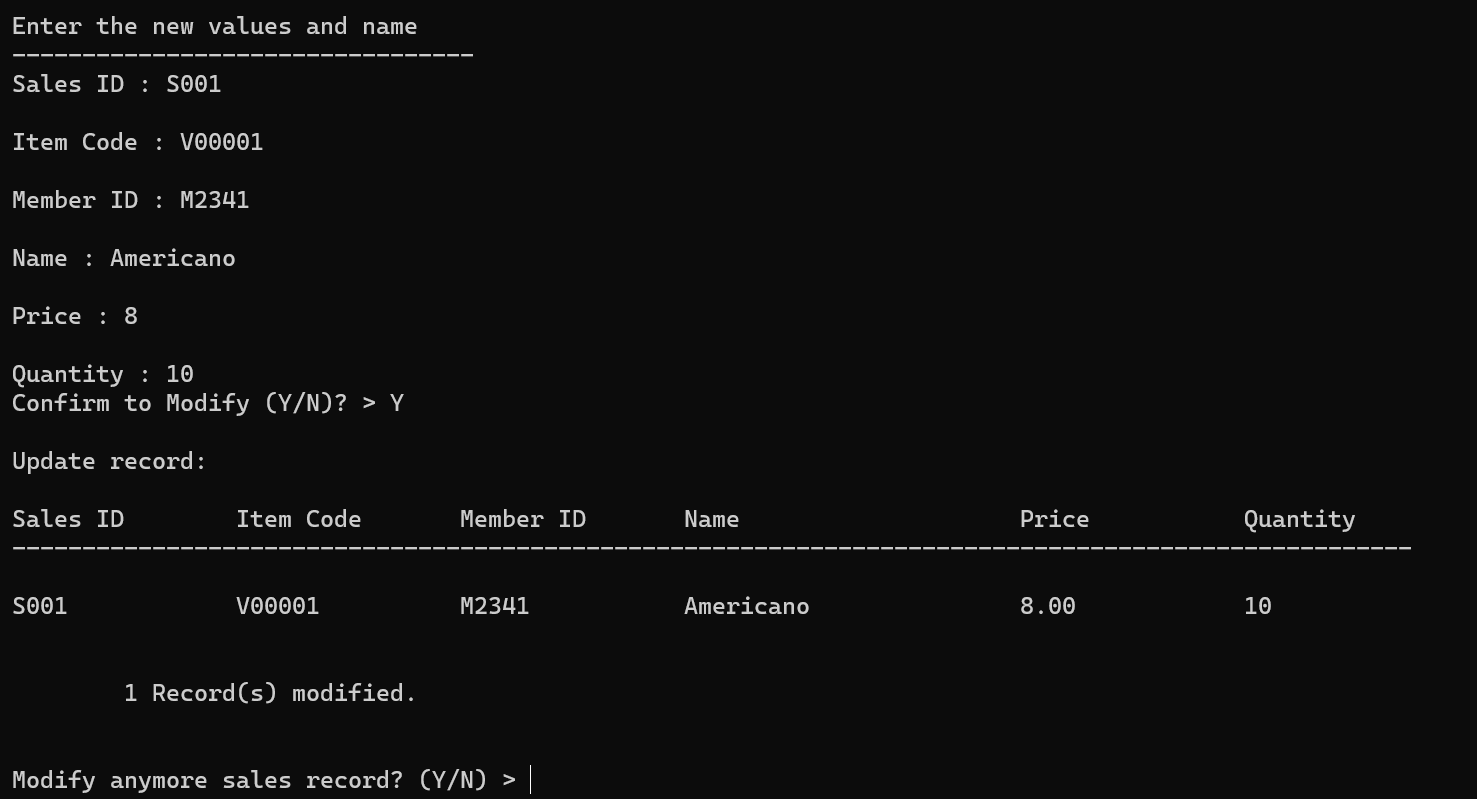
#### 4.4.5 Modify Sales Function

When the user enters the value ‘4’ in the main menu, it will display the modify function. Here in this function users are allowed to modify the data fields.

In order to use the function, the user must first enter the sales ID.



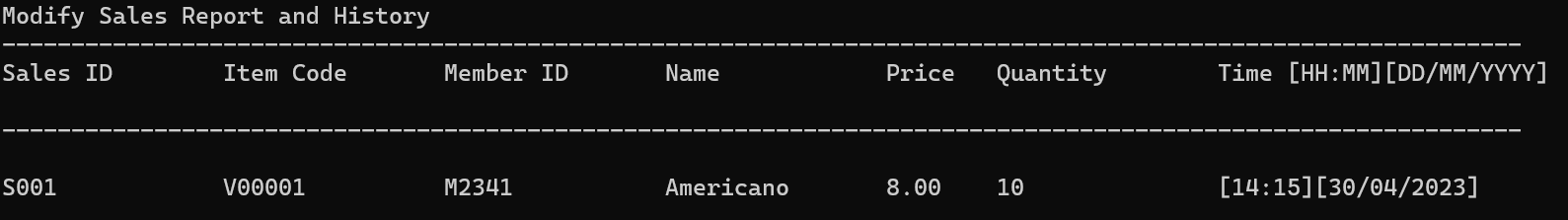
Once entered, it will display the data of the sales ID before asking the user to enter the new values of the data field.



We can modify every data field, but for now we will just modify the PRICE and QUANTITY. When the user confirmed it, the record would be modified and saved in the file. Users are then able to modify again or exit the function.

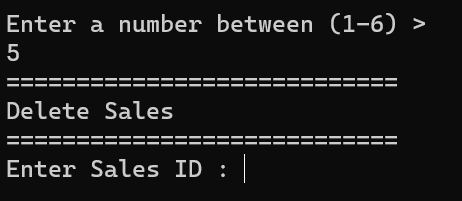
Once the sales data is successfully modified. A report is recorded which can be displayed by the sales report function.

Here is an example of the sales data modified and saved into the sales report file with the time and date displayed too.

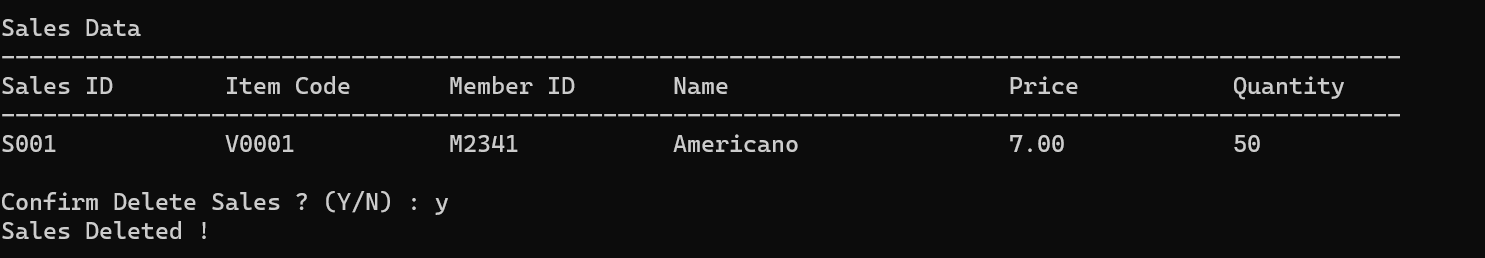


#### 4.4.6 Delete Sales Function

When the user enters the value ‘5’ in the main menu, it will display the report function. Here in this function, users can enter a sales ID they want to delete.



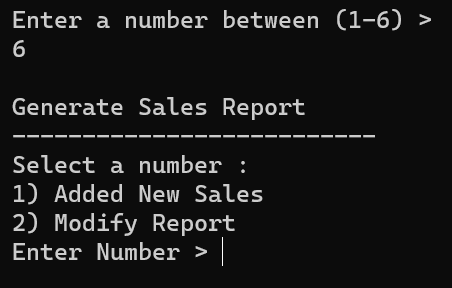
When the sales ID is entered, it will display the data field to ensure it is the one that you want to remove. By entering Y, the sales data will be deleted from the file and it will loop back to the Sales main menu.



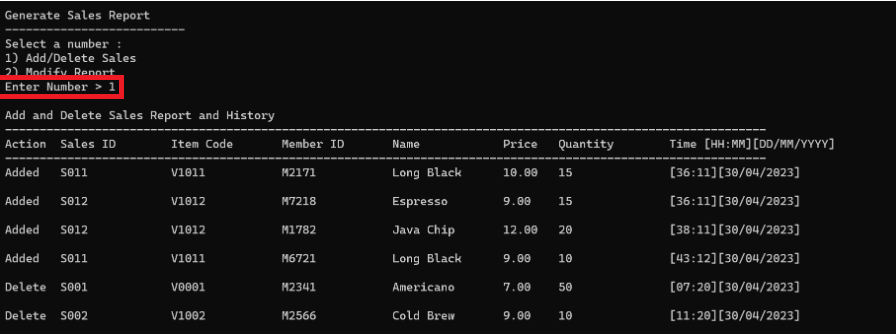
#### 4.4.7 Display Report Sales Function

When the user enters the value ‘6’ in the main menu, it will display the report function. Here in this function users can view the reports on add/delete sales and the modified sales.

In order to use the function, the user must first enter the sales ID.



When a user enters the value 1 it will display the added and deleted sales in the report. It will also display the time that it got updated.



Meanwhile, when users enter the value 2, it will display the modify report with the date and time that it got updated.

