Diploma in

Common Engineering Programme (C42)

Programming Report

TRIPLE INS’ GROCERY

Mock Ying Jing (224416M)

Ngo Ke Xin (221411J)

# 

# 

# 

# 

# **Contribution Lists**

|  |  |  |
| --- | --- | --- |
| **Members** | **Contributions** | **Strengths** |
| Ying Jing | **Code**  - SignUp/Login  - View Menu  - Place Orders  - View Orders Placed  - Remove Existing Items  - Feedback  **Others**  - Flowchart (SignUp/Login, View Menu, Place Orders, Feedback/Review)  - Powerpoint Slides & Report | - Using Functions, Lists, Conditional Statements, global and local variables  - Debugging Errors  - Make the program more user-friendly  - Organizing Flowchart, Report, and Powerpoint Slides |
| Ke Xin | **Code**  View Menu  Payment/Checkout  **Others**  - Flowchart (View Orders, Remove Items,Payment/CheckOut, Receipt)  - Powerpoint Slides & Report | - Calculations for payment/checkout  - Detailed payment/checkout process  - Debugging errors  - Using lists, Conditional Statements, global and local variables |

**Table of Contents**

Page

**Acknowledgment** 0

**Objectives**

* The Purpose of Program 1

- What The Program Does 1-2

**Scope**

- Key Specifications and Features 3

* Major Deliverables 3

**Implementation**

- Flowchart 4

- Key Technical Issues 5

- Solutions to Solve Them 5-6

**Special Features**

- Features Not Included in Requirements 7

- Reason to Add Special Features in Program 7

**Conclusion**

* Reflections and Outcomes 8
* Future Enhancement 8

**Appendixes** 9

# 

# 

# **Acknowledgment**

# Firstly, we would like to express our special gratitude to our Programming Module Leader of Nanyang Polytechnic, who gave us the golden opportunity to work as a team on this online shopping mini project. We can think creatively and apply our programming knowledge to develop a user-friendly online ordering and payment system.

# 

# Secondly, we would like to convey our heartier thanks to our Programming Lecturer, Mr. Tay Huck Sun, who heartily welcomed, guided, and encouraged us throughout these past 17 weeks. Mr. Tay does not hesitate to help us whenever we are in doubt, especially while we are doing the mini project. His explanation of our specific doubt is clear and easy to understand. He would also take his time to hear and guide us toward the right path.

# 

# Finally, we would like to use this opportunity to express our deepest thanks to all our Module Lecturers. Their guidance throughout these past 17 weeks has taught us new skills and knowledge, benefiting, and allowing us to apply them in the future.

# 

# We couldn't thank them enough. Our module lecturers are the ones who nurtured and accompanied us. If we're not for them, we are not who we are today.

# 

**Objectives**

**The Purpose of the Program**

Online shopping is one of the most popular choices among people in the world. Many people prefer online shopping rather than shopping physically at the shop. This is because online shopping is convenient and efficient, especially for busy workers who may have insufficient time to drop by the physical shop to purchase their needs.

Due to the rapid increase in COVID'19 cases, many people were unable to leave their homes to purchase their needs. People who constantly cook for their families at home would need to restock their groceries once their ingredients ran out. These may put their health in danger, especially those elders with poor health.

With these factors, we have developed a user-friendly and responsive online ordering and payment system for everyone and future generations. These will help more people to lower the risk of being infected by the COVID'19 virus. Busy workers do not need to worry they have insufficient time to purchase their needs.

As Triple Ins' Grocery is a newly established company in July 2022, there are not many customers interacting with the physical shop. Creating brand awareness with the program development is another purpose to attract a wider audience to use our online ordering and payment system.

**What The Program Does**

This program allows people to choose and buy what they want more conveniently. When the user first enters the program, it will show different options such as signup or login for an account, view the menu, place orders, remove items and check out. The program will require the user to sign up for an account and login for them to place their order and for checkout.

After logging in to the system, the user can then proceed to view menu options. They can view different orders based on their choice. The program will then sort out and display the user’s choice, either by alphabetical, ascending prices, or by category. The user can choose and decide on what they want to buy.

When placing the order for items, the program will first ask the user which category they want, followed by the items from the category, and lastly the quantity of the item. After choosing their desired items from a specific category, the program will then ask the user to confirm if they want to add their items to the cart. After the current items was added, their items will be displayed, and ask the user if they want to continue to add another item. If the user continued to add items, their current placed items will be displayed below their previous placed items.

The users can choose to view what are the items in their cart. The program will then display all the items, quantity, and subtotal of the user's choice.

The user can also choose to remove an item from the cart. If the user has a change of mind, they can select which item to remove from the cart based on the ID of the item. The program will then sort out and display the selected item to remove and ask the user to confirm if that selected item is the one the user wants to remove. If the user confirms, the program will then extract the selected item from the cart. After removing one item, the user will be asked if they want to continue removing another item.

Lastly, the user will check out once they are done selecting all the items they want to buy. The program will display all the items, quantity, and subtotal. Followed by the calculation of all items' prices including GST, discount, and the final total price. Every user who used this program will have a 10% discount as they had all signed up for an account. The program will then ask the user which payment method they want to pay by.

After selecting the payment method, the program will require the user to enter their details to confirm the purchase. After paying, the program will ask if the user wants a receipt and if they would want to leave feedback.

**Scope**

**Key Specifications and Features**

**View menu**

When a user selects to view the menu, the program should ask how the user wants to view the menu. Be it in Alphabetical, Prices from lowest to highest, or Category. The program should then display all the items based on the user’s choice.

**Adding of items**

The user is allowed to select multiple groceries items from various categories and add them to the cart. The program will display the selected category, and the user will need to choose which item and the quantity they want to buy. The user will be asked if they want to add another item to the cart.

**Viewing of all items in the cart**

When the user chooses to view what are the items in the cart, the program should display all the added items, quantity, and the subtotal.

**Removing of items**

The user is allowed to remove an existing item if needed. If the user chooses to remove an item, the selected item will be removed, and the program will display the updated list of items in the cart. The user will be asked if they want to remove another item from the cart.

**Check-out**

The program will display all the items, quantity, and subtotal in the cart. The program will also display the total price including GST, discount, and the final total. The user will then confirm if they want to check out.

**Major Deliverables**

**Progress Timeline:**

**Week 12:** Understanding what the project requirements are, planning who does which part.

**Week 13:** Storing all items and their prices in a dictionary. Sorting of all items in alphabetical, ascending price, and category order. Creating a loop for users to use the program.

**Week 14:** Using functions to group all categories by alphabetical, ascending price order. Working on code for adding items into the cart and viewing all items in the cart.

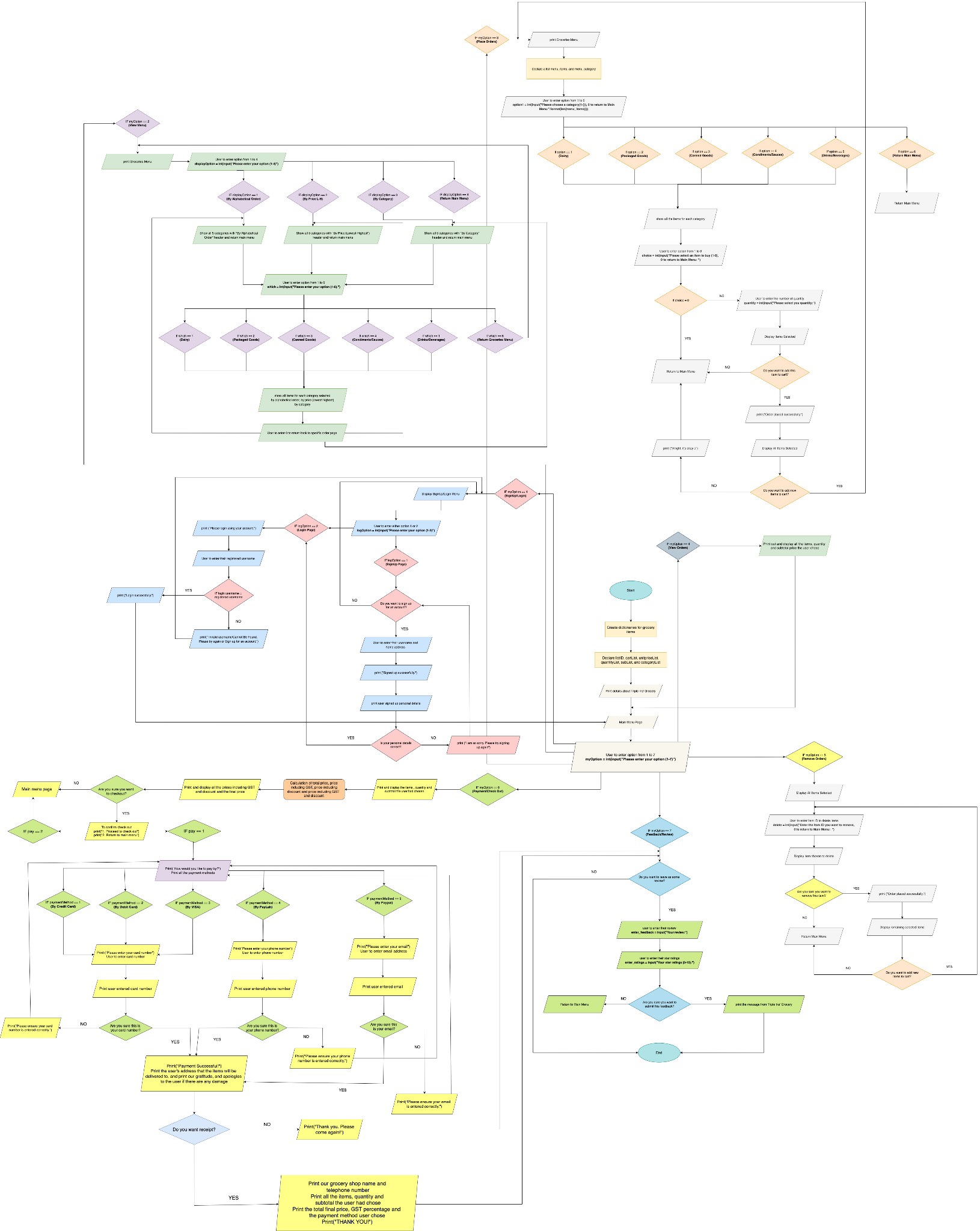
**Week 15:** Working on code for removing items in cart and checkout. Adding extra features, code for sign-up, payment method, and feedback. Flowchart.

**Week 16:** Final touch-up for all the functions. Rerun the whole program and debugging errors.

**Week 17:** Presentation and Report writing.

**Implementation**

**Flowchart**

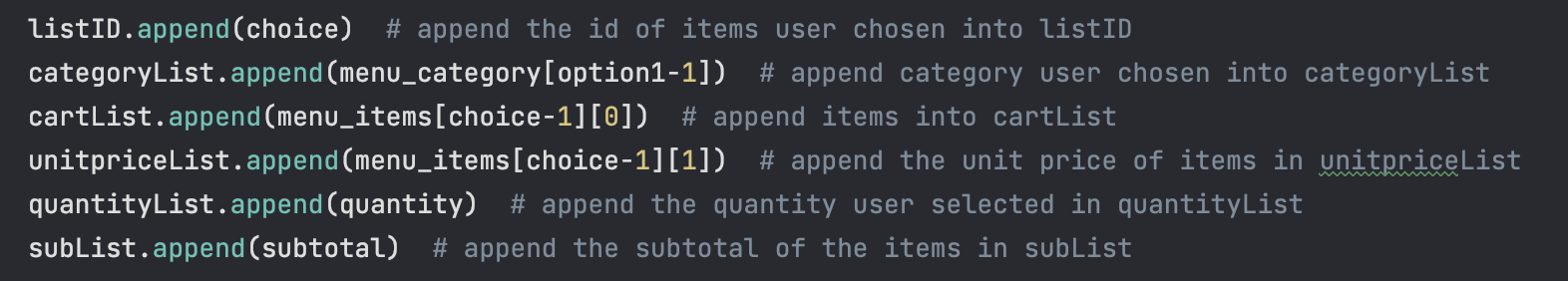
****

**Key Technical Issues**

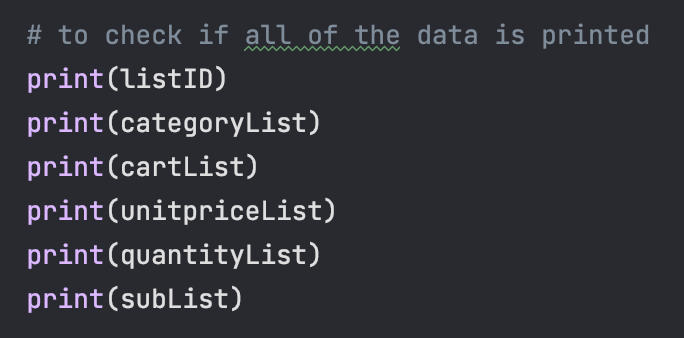
The key technical issues we faced are:

1. Place Order (Unable to add multiple items to cart)
2. View Order (Item’s ID does not change)
3. Remove Existing Order (Does not print the unremoved items)
4. Payment/Checkout (Unable to print out the Receipt)

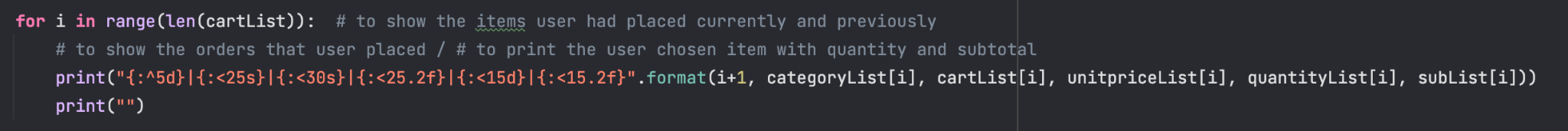
**Solutions to Solve Them**

**Place Order (Unable to add multiple items to cart)**

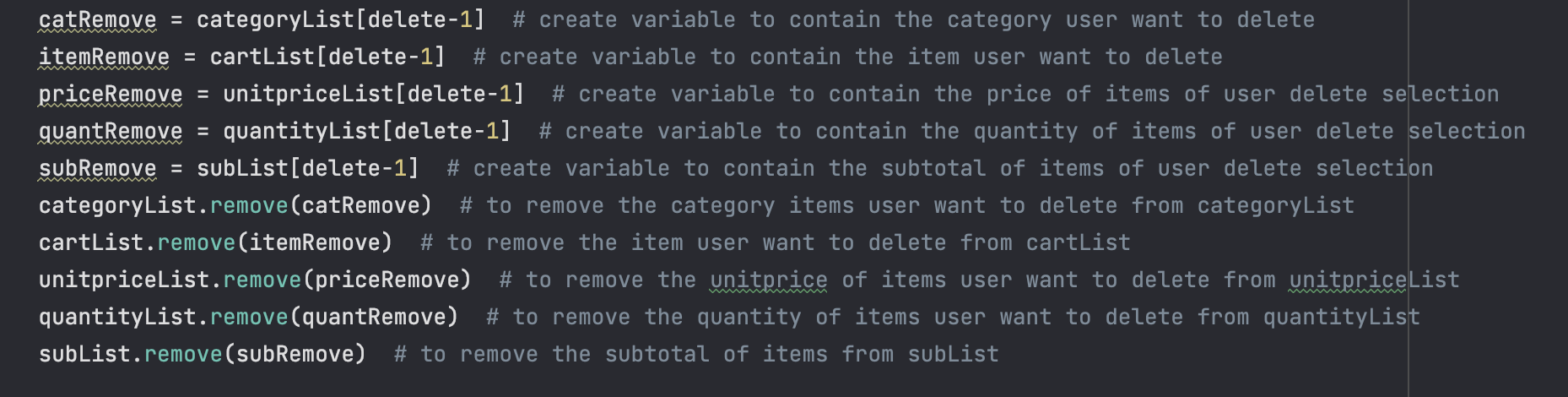
We used the append method in the place order functions to add the items into the cart each time user placed an order.



We used the print method after the appending method to check if all the data is printed out correctly as expected.

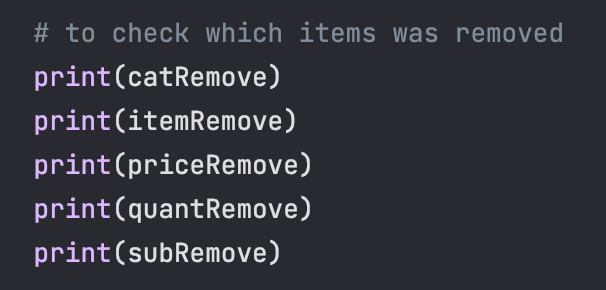
**View Order (Item’s ID does not change)**

We used the [i] beside each list to print out all the items placed by the user currently and previously. The i+1 is used to print out the items in order starting from 1, 2, 3, and so on.

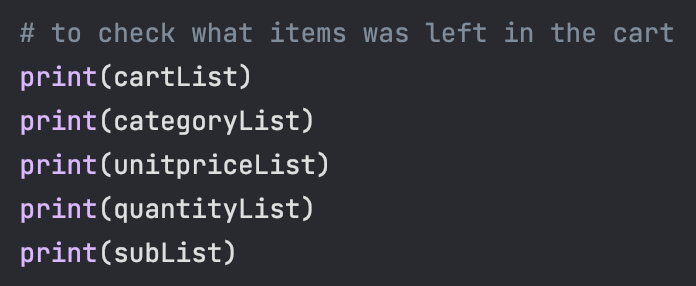
**Remove Existing Orders (Does not print the unremoved items)**

We created each variable to contain all the item data (such as category, items, price, quantity, and subtotal) that the user wants to delete. The [delete-1] is to delete the items from the list user specified.

To remove the items that the user has specified, we used the created lists (such as categoryList, cartList, unitpriceList, quantityList, and subList) to remove the added items from the specific lists.



**Image A**



**Image B**

We used the print method after the removing method to check if all the data is printed out correctly as expected. For example, the print method in Image A is to check which items were removed from the items list created. Whereas in Image B, the print method is to check the remaining items left in the cart.

**Payment/Checkout (Unable to print out the Receipt)**

We need to look through the entire code to identify the error based on the written error displayed on the console output. We realized that it was syntax and logic errors. We also need to find out what is the formatting code for the receipt.

**Special Features**

**Features Not Included in Requirements**

Extra Features we added are:

1. Sign Up
2. Login
3. Payment Methods
4. Receipt
5. Feedback

**Reason to Add Special Features in Program**

Online shopping has become a widely popular choice for shoppers. As reported in the news, many online shoppers lost their money due to online transaction fraud. To secure the safety of users, we added special features to our program so that they can rest assured in buying their groceries online.

The Signup feature allows the users to register their username and home address. The Login feature allows the users to log in using their registered username after they sign up for their account. These are mandatory for users to access the full features of the program. Their registered username and home address will be submitted and stored inside the database. An invalid username logged into the system will prompt the users with an error message and direct them back to the signup and login menu page.

In payment methods, we added credit cards because credit cards are a good option for purchasing goods online since most credit card companies have lower liability for fraud. If the users use a credit card, they have more time to dispute fraudulent charges, and the bank can withhold payment while investigating the incident. As such, we implemented this option for the user to pay for their items safely. On the other hand, if the user does not want to pay by credit card, they are given different payment methods to choose from to pay for their items.

After each payment method, a confirmation email will be sent to the user through their email for verification. Their confirmation email will include their groceries orders, personal details, and types of payment methods used for checkout. These allow the users to know who had used their email and phone numbers when they check their email for confirmation. The registered home address is the critical source of who used their personal information for transactions. Thus, they can make reports easily with the given evidence in the email.

Adding the receipt features allows users to get a refund or replacement if there are any damages or frauds to the groceries they bought. If there is no evidence of receipt, refunds or replacements will be unable to be made.

Implementing the feedback features allows the user to provide us some feedback and ratings on how well our program performs. These enable us to collate all feedback and brainstorm a few strategies on what we should do to improve the program for future users.

**Conclusion**

**Reflections and Outcomes**

In this mini project, we put in our blood, sweat, and tears to develop a user-friendly online ordering and payment system. When we started to code the program, we understood that it looks simple on the outside, but it is not easy as it requires time and effort to complete the whole program.

When coding the functions for the main menu page, we experienced a lot of errors in the code, causing the entire program to stop working. We would need to find out the cause by looking through the whole code to debug them. We felt a sense of relief when we found out that it is just logic and syntax errors.

Using lists and dictionaries is challenging as we need to understand how to use them efficiently, logically, and correctly. With Mr. Tay's help in explaining to us again, we understand better how we should use them.

Brainstorming ideas on how we should implement a user-friendly online ordering and payment system allows us to develop our own creative and critical thinking skills. With the new skills and knowledge gained from this mini-project experience, we are confident that we could apply them in the future that is waiting for us.

**Future Enhancement**

**1. Communication**

We can spend more time discussing with each other which part of the project we would want to do. As such, this would not cause one to overtake each other’s part and it ensures each of us does the tasks assigned evenly. We should also find a time when both parties are free to do the project together. This helps us to communicate better and complete the project on time.

**2. Procrastination**

We should not procrastinate on the tasks assigned to each of us. Procrastinating only causes us to get things undone and unable to meet the deadline requirements. If our tasks assigned were not completed, we would not have much time to revise for our upcoming examinations and debug for errors that may occur during the running of the program. We may also have insufficient time to add new features and meet the expected program requirements.

**3. Adding of ‘Offer’ feature**

Adding an offer feature would attract more customers to our online ordering and payment system. This is because when there is an offer on items, more customers will feel the urge to buy as it is much cheaper than the original price. This will also help to improve their experience when they are using our program.

**Appendixes**

Our Flowchart

<https://app.diagrams.net/#G1bYEo-oFETJoSQ0YJEyRtReqEG-CIlvAJ>

Flowchart Software

<https://app.diagrams.net/>

How credit cards secure the safety of online shoppers

<https://www.csa.gov.sg/gosafeonline/go-safe-for-me/homeinternetusers/protect-yourself-with-these-tips-when-shopping-online>