Programming Fundamentals

|  |  |
| --- | --- |
| **Student’s Name:** | Raja Shahmir Muhammad |
| **Id. No:** | 717 |
| **Github Repository Name:** | Programming-Skills-Portfolio |
| **Github Repository Link:** | https://github.com/shahmir-raja/Programming-Skills-Portfolio |

**Repository Screen Shot:**

A screenshot of a computer

Description automatically generated

ASSESSMENT 1&2

| Contribution towards overall module mark | 100% (40%+60%) |
| --- | --- |
| Date set | September 5, 2023, |
| Marked work returned by | Within 3 weeks of submission |
| DEADLINES | Deadline : Jan 10, 2024– 23:59 |

**Brief**

I took a course called Programming Fundamentals in the first semester of my BSc in Creative Computing. We had to write a vending machine program for the module's programming component. The program's needs and goals are described in full in this paper. You may find the whole program needed to run it in the repository (Python 8.7.0 https://github.com/shahmir-raja/Programming-Skills-Portfolio/tree/main

**Specification**

Products:

|  |  |  |
| --- | --- | --- |
| ID | Name | Price |
| 0 | Pepsi | 2.50 Dhs |
| 1 | Lays | 5.50 Dhs |
| 2 | Water | 1 Dhs |
| 3 | Chocolate | 1.75 Dhs |
| 4 | Protein Bar | 5 Dhs |
| 5 | Gum | 0.75 Dhs |

Explore the variety of items available in the vending machine. Each product is represented by a unique ID that corresponds to the selection button. Among the offerings are Pepsi, Lays, Water, Chocolate, Protein Bar. The indicated price signifies the cost of each item. To make a purchase, simply input the ID of your desired product into the program.

**System Flowchart**

Customers begin by inputting a code for their desired food or drink item. The vending machine retrieves the item's cost and manages coin insertion and calculation. If the inserted coins match the item's price, the product is dispensed. If the payment exceeds the item's cost, the machine returns change. Customers are prompted to choose a receipt. If selected, they are asked for payment before concluding the transaction.

**A diagram of a company

Description automatically generated**

**Technical Description**

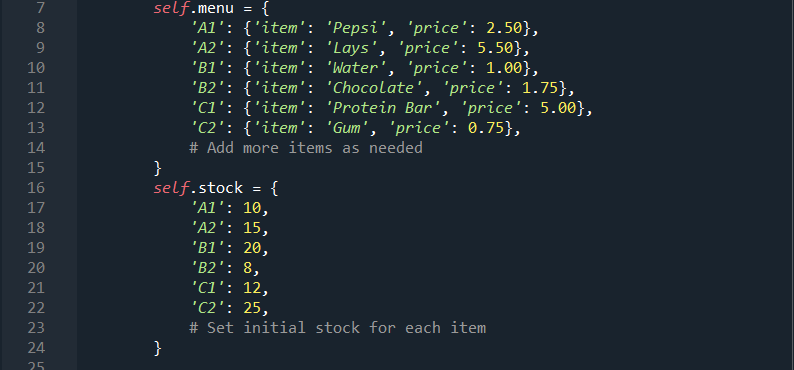
The vending machine program is implemented using Python and encapsulated within a class named VendingMachine. It consists of methods and data structures to facilitate the vending process.

**A screen shot of a computer

Description automatically generated**

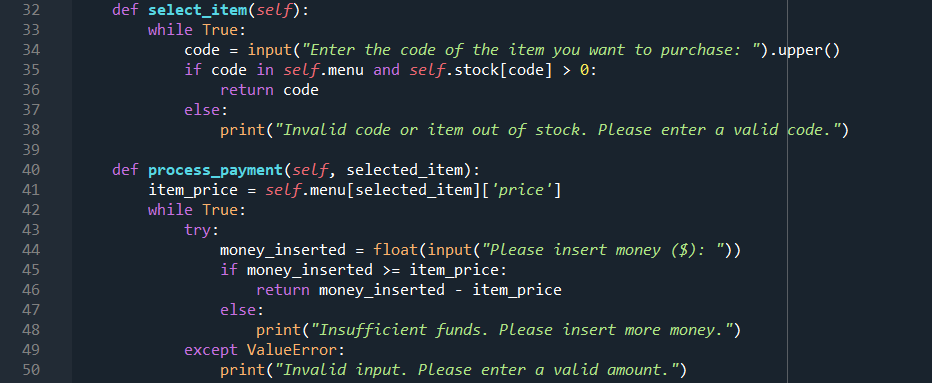
Initialization: The class initializes with two dictionaries: menu and stock. menu stores the items available for purchase, each associated with a unique code, item name, and price. stock maintains the available quantity for each item.

Displaying Menu: The display\_menu() method presents the available items, their codes, names, and respective prices to the user through the console.



Item Selection: The select\_item() method allows the customer to input a code corresponding to their desired item. It verifies the code's validity and checks if the selected item is in stock.

Payment Processing: Upon item selection, the process\_payment() method prompts the customer to input the payment amount in dollars. It compares the inserted amount with the selected item's price and handles cases of insufficient funds or valid payment.



Dispensing Items: The dispense\_item() method is responsible for dispensing the chosen product if the payment is adequate. It updates the stock count for the dispensed item and prints a "Thank you!" message to acknowledge the successful transaction.

A screen shot of a computer program

Description automatically generated

Running the Vending Machine: The run() method orchestrates the vending process. It repeatedly displays the menu, handles item selection, payment processing, item dispensing, and change calculation until the user ends the program.

A blue background with white text

Description automatically generated

This vending machine program effectively simulates a simplified vending process, managing item selection, payment, and item dispensing, providing a concise and functional interaction for the user.

**Appendix**

