Minhaj University Lahore

Semester Project: 1st		Mid ○	[✓ Final ●]
Course Title & Code	Introduction to Programing Fundamental		
Submitted By:	Atif Nasir		
Registration No:	2023f-mulbscs-059		
Semester/Class/ Section:	1st ‖ Bscs ‖ "B"		
Submitted To:	Miss:Uniza Rehman		
Due Date	Online: yes	Hard copy: yes	
Student's Signature:			

Note: *Please avoid cutting/ overwriting in any of the above fields.
*Complete the task on standard A4 size papers/assignment pages.

***For Instructor's use only.**

Total Marks:	
Obtained Marks:	
Signatures:	

Faculty of CS & IT,Minhaj University Lahore

Project Title:

S Mart: Supermarket Billing System

---

1. Introduction

The "S Mart: Supermarket Billing System" is a simple C++ application designed as a first-semester project to illustrate the concepts of arrays, loops, and conditional statements. The system simulates a supermarket environment where users can view available products, make purchases, and get a running total of the bill. This project serves as an introduction to basic programming structures and enhances understanding of user interaction and control flow in C++.

---

2. Objectives

The main objectives of the project are:

- **To demonstrate array handling:** Storing and retrieving product information such as names, prices, and stock levels.
- **practice control structures:** Implementing loops and conditional statements for program flow control.
- **To simulate real-life applications:** Applying programming fundamentals in a scenario that mimics a supermarket's billing process
- **User interaction:** Engaging the user through a text-based interface where they can select options to display items, purchase products, and view the total bill

---

3. System Design and Architecture:

3.1. Program Flow

The overall flow of the application is based on a menu-driven interface that repeatedly prompts the user until they choose to exit. The primary steps include:

1. **Displaying the Menu:** The user is presented with several options: show items, buy an item, display the total bill, or exit.
2. **User Choice Handling:** The program uses a loop (do-while) to continuously ask for the user's choice and execute the corresponding functionality.

3. **Product Management:** Arrays are used to maintain product names, prices, and stock levels.
4. **Transaction Process:** When a user opts to buy an item, the program checks for valid inputs and sufficient stock, updates the stock accordingly, and computes the cost to add to the overall total bill.
5. **Bill Display:** At any point, the user can view the total amount for their purchases.

3.2. Components

Arrays:

- items[8]: Stores the names of the products.
- prices[8]: Stores the price of each product in PKR.
- stock[8]: Maintains the available stock for each product.

Control Flow:

- do-while loop: Ensures that the menu is displayed until the user opts to exit.
- if-else statements: Handle the logic for each menu option.

User Input & Output: Uses standard input/output (cin and cout) for interaction.

---

4. Code Walkthrough :

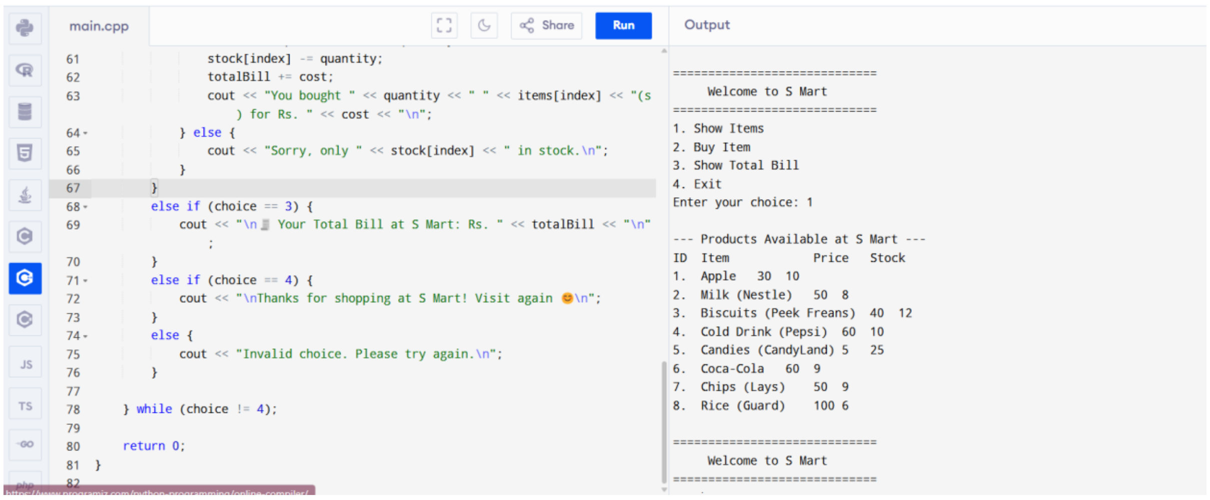
Below is the annotated version of the code:

```
#include <iostream> // Include the iostream library for input/output operations.using
namespace std;

int main() {

  // Product names

  string items[8] = {

    "Apple",              // Index 0

    "Milk (Nestle)",       // Index 1

    "Biscuits (Peek Freans)", // Index 2

    "Cold Drink (Pepsi)",  // Index 3

    "Candies (CandyLand)", // Index 4

    "Coca-Cola",          // Index 5
    "Chips (Lays)",        // Index 6
    "Rice (Guard)"         // Index 7
  };

  // Prices in PKR for each corresponding product.
  int prices[8] = {30, 50, 40, 60, 5, 60, 50, 100};

  // Stock available for each product.
  int stock[8] = {10, 8, 12, 10, 25, 9, 9, 6};

  // Variable to keep track of the total bill.
  int totalBill = 0;
  int choice;

  // The main loop that continues until the user opts to exit.
  do {
    // Display the menu header.
    cout << "\n==============================\n";
    cout << "     Welcome to S Mart\n";
    cout << "==============================\n";
    cout << "1. Show Items\n";
    cout << "2. Buy Item\n";
    cout << "3. Show Total Bill\n";
    cout << "4. Exit\n";
    cout << "Enter your choice: ";
    cin >> choice;

    // Option 1: Display available products.
    if (choice == 1) {
        cout << "\n--- Products Available at S Mart ---\n";
```

```
            cout << "ID\tItem\t\t\tPrice\tStock\n";
            for (int i = 0; i < 8; i++) {
                // List each product with its corresponding details.
                cout << i + 1 << ".\t" << items[i] << "\t" << prices[i] << "\t" << stock[i] << "\n";
            }
        }
        // Option 2: Process the purchase of an item.
        else if (choice == 2) {
            int itemID, quantity;
            cout << "Enter item ID (1 to 8): ";
            cin >> itemID;

            // Validate the item ID.
            if (itemID < 1 || itemID > 8) {
                cout << "Invalid item ID!\n";
                continue;
            }

            cout << "Enter quantity: ";
            cin >> quantity;

            int index = itemID - 1;
            // Check whether the requested quantity is available in stock.
            if (quantity <= stock[index]) {
                int cost = prices[index] * quantity;
                stock[index] -= quantity;   // Update the stock.
                totalBill += cost;          // Update the total bill.
                cout << "You bought " << quantity << " " << items[index] << "(s) for Rs. " << cost
    << "\n";
            } else {
                cout << "Sorry, only " << stock[index] << " in stock.\n";
            }
        }
        // Option 3: Display the current total bill.
        else if (choice == 3) {
            cout << "\n□ Your Total Bill at S Mart: Rs. " << totalBill << "\n";
        }
        // Option 4: Exit the application.
        else if (choice == 4) {
            cout << "\nThanks for shopping at S Mart! Visit again □\n";
        }
        // Handle any invalid menu choices.
        else {
            cout << "Invalid choice. Please try again.\n";
        }

    } while (choice != 4); // Continue the loop until the user selects option 4.

    return 0; // End of the program.
}
```

Program Compile Link:
https://www.programiz.com/online-compiler/9DMlCjlwB05Nu

Screenshot:



```cpp
                stock[index] -= quantity;
                totalBill += cost;
                cout << "You bought " << quantity << " " << items[index] << "(s
                    ) for Rs. " << cost << "\n";
            } else {
                cout << "Sorry, only " << stock[index] << " in stock.\n";
            }
        }
        else if (choice == 3) {
            cout << "\n▒ Your Total Bill at S Mart: Rs. " << totalBill << "\n"
                ;
        }
        else if (choice == 4) {
            cout << "\nThanks for shopping at S Mart! Visit again 😊\n";
        }
        else {
            cout << "Invalid choice. Please try again.\n";
        }

    } while (choice != 4);

    return 0;
}
```

```
=============================
     Welcome to S Mart
=============================
1. Show Items
2. Buy Item
3. Show Total Bill
4. Exit
Enter your choice: 1

--- Products Available at S Mart ---
ID  Item            Price   Stock
1.  Apple   30  10
2.  Milk (Nestle)   50  8
3.  Biscuits (Peek Freans)  40  12
4.  Cold Drink (Pepsi)  60  10
5.  Candies (CandyLand) 5   25
6.  Coca-Cola   60  9
7.  Chips (Lays)    50  9
8.  Rice (Guard)    100 6

=============================
     Welcome to S Mart
=============================
```

4.1 Key Concepts Illustrated :

- **Arrays:**
  The three arrays (items, prices, and stock) store related data using corresponding indices. This practice reinforces how to maintain and reference related data in C++.
- **Conditional Statements:**
  The if-else blocks ensure that the program responds appropriately to user input, validating inputs such as item IDs and ensuring that purchases do not exceed available stock.
- **Loops:**
  The do-while loop continuously presents the user with the menu options until the exit option is selected, demonstrating control flow for interactive applications.
- **Input/Output Operations:**
  The use of cin and cout illustrates handling basic console I/O operations, an important aspect of beginner-level programming projects.

5. Testing and Results :

The application was tested by simulating various user interactions:

- **Displaying Items:**
  Confirmed that all items, along with their price and stock, are displayed correctly.
- **Buying Items:**
  Verified that the application correctly handles valid inputs and cases where the requested quantity exceeds available stock.
- **Total Bill Calculation:**
  Checked that the total bill accurately reflects the cumulative cost of purchased items.

- **Edge Cases:**
  Ensured that invalid inputs (e.g., incorrect item ID or invalid menu selection) are handled gracefully.

The testing confirms that the application meets the intended functionalities for a basic supermarket billing system.

---

6. Conclusion:

The S Mart supermarket billing system is a foundational project for exploring C++ programming concepts. It integrates array handling, control structures, and basic I/O operations in a user-friendly menu-driven environment. The project not only helps reinforce theoretical concepts from class but also serves as a building block for more complex applications.

Future Enhancements

- **Data Persistence:**
  Introduce file handling to save transaction details and stock updates.
- **GUI Implementation:**
  Develop a graphical user interface for an enhanced user experience.
- **Additional Features:**
  Incorporate functionalities such as discount management, multiple counters, or customer management to simulate a real-world retail system.

References

1. C++ Programming Language
Documentation -
https://cplusplus.com/doc/tutorial/
2. "Programming Fundamentals" Lecture Notes by Miss Unza Rehman
3. Book: "Object-Oriented Programming in C++" by Robert Lafore
4. YouTube Tutorials - CodeWithHarry, The Cherno (C++ Basics and Projects)
5. Stack Overflow Community Discussions - https://stackoverflow.com/
6. GeeksforGeeks C++ Articles - https://www.geeksforgeeks.org/c-plus-plus/
7. Microsoft Learn: Introduction to C++ - https://learn.microsoft.com/en- us/cpp/
8. W3Schools C++ Tutorial - https://www.w3schools.com/cpp/
9. Sololearn C++ Course - https://www.sololearn.com/