Programming Fundamentals

Fall 2020

Project Details

Project	
Project Title	**Online Shopping and Warehouse Management System**
Registration Number	**2020-CS-144**
Summary	*There will be three main/basic users for the software: An Admin, an employee, and a buyer. Admin will just be able to edit employees, their ID's and passwords and view a balance sheet of the company. The employee will be able to manage all incoming and outgoing goods. An employee can add a list of items and their buying cost and price, view all the buyers/users with their orders that they placed. The buyer will register him/herself first where they will be prompted to add their username and password, address, and a telephone number. All these inputs will be checked with validation functions. Buyers can see all the goods, their availability that whether that good is in stock or not, and the price of the product. Buyer can also add listed products in cart and checkout at the end where a buyer can see the total price for the products, he/she is going to buy.*
Features	*At the starting of the software, it will be prompted whether it is being accessed by the admin or buyer. buyers and register themselves and then proceed to their shopping. Admin can view total purchase and sale in a form of the balance sheet which will be printed in a form of an excel file where total profit will be also displayed. All buyer data will also be available in a text file. Details of goods in the warehouse, availability of every item in the inventory will be accessible to employees which are also stored in an excel sheet. the program also sorts which items have a higher demand. buyers have an option whether to view items by category, price, and most selling items. This program also alerts employees that an item is about to finish, or it is out of stock, so you need to order more.*
Implementation Details	
Write down the names of all datatypes used in the code?	Int, char, string, double, short int.
How many times for, while and	For loop: 23 times

do-while loop is used?	While loop: 8 times Do While: 4 times
Do you have used switch statements, if yes then enter the purpose of each occurrence.	5 Switch Statements were used in menu-based options. 1. To switch between **company menu** and **user menu.** 2. Used in company menu to switch between **admin menu** and **employee menu**. 3. Then 3 Switch Statements were used in each sub menus i.e., Admin, User, and Employee to select option number.
Do you have any compile time errors in your code?	No.
Do you have any runtime errors in your code?	No.
Enter the names of major modules in your system.	It is totally menu based so all are major modules.
Write down all parallel 1D arrays and their purpose	**For Login:** string username, password string usersList[MAX_USERS], passwordList[MAX_USERS] string employeeCode[MAX_EMPLOYEE], employeePass[MAX_EMPLOYEE] **For Employee:** string itemList[MAX_ITEMS] double costList[MAX_ITEMS], priceList[MAX_ITEMS] int barcodeList[MAX_ITEMS], quantityList[MAX_ITEMS] **For User:** string userItemsList[MAX_ITEMS] int userQuantityList[MAX_ITEMS] double userPriceList[MAX_ITEMS]
Write down all 2D arrays and their purpose.	No 2D array used in this program.
What is the code length? Size in KBs and lines of code	CS142F20R144.cpp (42KB) **1520** lines of code. Employee.txt (1KB) itemList.txt (1KB) profit.txt (1KB)

Details of functions	
Function Name 1	**Welcome**
Function Prototype	void welcome()
Description	This functions displays Welcome Screen of the program
Function Return type and purpose	This function has no return type.
Parameter names and purpose of each parameter	There is not any parameter in this function.
Function Name 2	**Admin Menu**
Function Prototype	void adminMenu()
Description	This functions displays Admin Menu Screen
Function Return type and purpose	This function has no return type.
Parameter names and purpose of each parameter	There is not any parameter in this function.
Function Name 3	**Company Menu**
Function Prototype	void companyMenu()
Description	This functions displays Company Menu Screen
Function Return type and purpose	This function has no return type.

Parameter names and purpose of each parameter	There is not any parameter in this function.
Function Name 4	**Employee Menu**
Function Prototype	void employeeMenu()
Description	This functions displays Employee Menu Screen
Function Return type and purpose	This function has no return type.
Parameter names and purpose of each parameter	There is not any parameter in this function.
Function Name 5	**User Menu**
Function Prototype	void userMenu()
Description	This functions displays User Menu Screen
Function Return type and purpose	This function has no return type.
Parameter names and purpose of each parameter	There is not any parameter in this function.
Function Name 6	**Invalid Option**
Function Prototype	void invalid()

Description	This functions displays Invalid Halt Screen when user enters invalid option number
Function Return type and purpose	This function has no return type.
Parameter names and purpose of each parameter	There is not any parameter in this function.
Function Name 7	**Program Exit Screen**
Function Prototype	void exit()
Description	Exits the program with return value 0.
Function Return type and purpose	This function has no return type.
Parameter names and purpose of each parameter	There is not any parameter in this function.
Function Name 8	**Halt Screen**
Function Prototype	void hold()
Description	This function displays halt screen and gets a character input from user to continue
Function Return type and purpose	This function has no return type.
Parameter names and purpose of each parameter	There is not any parameter in this function.

Function Name 9	**Login Screen**
Function Prototype	void login(string &username, string &password)
Description	Asks for username and password from user
Function Return type and purpose	No return types.
Parameter names and purpose of each parameter	1. **&username** Gets username as an input from user 2. **&password** Gets password as an input from user
Function Name 10	**Logout Screen**
Function Prototype	void logout()
Description	This function logouts current user
Function Return type and purpose	This function has no return type.
Parameter names and purpose of each parameter	There is not any parameter in this function.
Function Name 11	**Login Failure**
Function Prototype	void loginfail()
Description	This function is called whenever user inputs wrong credentials

Function Return type and purpose	This function has no return type.
Parameter names and purpose of each parameter	There is not any parameter in this function.
Function Name 12	**User Existence in Array**
Function Prototype	bool userExist(string user[], string pass[], string users, string passs)
Description	Checks if username and password input from user is present in array or not.
Function Return type and purpose	Boolean return type. Returns true if user is present in array
Parameter names and purpose of each parameter	1. **user[]** whole username array is checked with user's input in user's variable. 2. **pass[]** whole passwordLists array is checked with user's input in passs variable. 3. **Users** Brings input from user in the function. 4. **Passs** Brings input from user in the function.
Function Name 13	**Employee User Existence in Array**
Function Prototype	bool E_userExist(string user[], string pass[], string users, string pass s)
Description	Checks if username and password input from user is present in array or not.

Function Return type and purpose	Boolean return type. Return true if user is present in array
Parameter names and purpose of each parameter	1. **user[]** whole E_usersList array is checked with user's input in user's variable. 2. **pass[]** whole E_passwordLists array is checked with user's input in pass variable. 3. **Users** Brings input from user in the function. 4. **Passs** Brings input from user in the function.
Function Name 14	**Load Users**
Function Prototype	bool loadUser(string usersList[], string passwordList[])
Description	Initializes all users' credentials from Users text file.
Function Return type and purpose	Boolean return type. Returns true if users are loaded into arrays
Parameter names and purpose of each parameter	1. **usersList[]** usernames are stored into array from input file 2. **passwordList[]** passwords are stored into array from input file

Function Name 15	**Load Employee Credentials**
Function Prototype	bool loadEmployeeUser(string employeeCode[], string employeePass[])
Description	Initializes all employee credentials from Employee text file.
Function Return type and purpose	Boolean return type. Returns true if users are loaded into arrays
Parameter names and purpose of each parameter	1. **employeeCode[]** usernames are stored into array from input file 2. **EmployeePass[]** passwords are stored into array from input file
Function Name 16	**Save Employee Credentials**
Function Prototype	bool saveEmployeeUser(string employeeCode[], string employeePass[])
Description	Saves all employee credentials after editing in program
Function Return type and purpose	Boolean return type. Returns true if all employee credentials are saved successfully.
Parameter names and purpose of each parameter	1. **employeeCode[]** usernames are stored into input file from array 2. **EmployeePass[]** passwords are stored into input file from array

Function Name 17	**Save User's shopping List**
Function Prototype	bool userSaveShopping(string userItemsList[], int userQuantityList[], double userPriceList[], double userTotal, string phone, string address)
Description	Saves all user shopping list into a text file.
Function Return type and purpose	Boolean return type. Returns true if all data is saved in text file successfully.
Parameter names and purpose of each parameter	1. **userItemsList[]** Whole array is passed in this function to save data in the text file 2. **userQuantityList[]** Whole array is passed in this function to save data in the text file 3. **userPriceList[]** Whole array is passed in this function to save data in the text file 4. **userTotal** Whole array is passed in this function to save data in the text file 5. **phone** Brings input from the user to be saved in text file 6. **address** Brings input from the user to be saved in text file
Function Name 18	**Add Employee Credentials**
Function Prototype	bool addEmployee(string employeeCode[], string employeePass[], string empCode, string empPass)
Description	This function is used to add employee credentials i.e., employee code and password

Function Return type and purpose	Boolean return type. Returns true if employee credentials are added successfully.
Parameter names and purpose of each parameter	1. **employeeCode[]** whole array is passed in this function and particular employee code is saved in empty index 2. **employeePass[]** whole array is passed in this function and particular employee password is saved in empty index 3. **empCode** this variable brings user input from main to this function and its value is assigned to an empty index in employeeCode array 4. **empPass** this variable brings user input from main to this function and its value is assigned to an empty index in employeePass array
Function Name 19	**Update Employee Credentials**
Function Prototype	bool updateEmployee(string employeeCode[], string employeePass[], string empCode, string empPass)
Description	This function is used to update any existing employee credentials.
Function Return type and purpose	Boolean return type. Returns true if employee credentials are updated successfully.
Parameter names and purpose of each parameter	1. **employeeCode[]** whole array is passed in this function and particular employee code is saved in empty index 2. **employeePass[]** whole array is passed in this function and particular employee password is saved in empty index 3. **empCode** this variable brings user input from main to this function and its value is assigned to an empty index in employeeCode array

	4. **empPass** this variable brings user input from main to this function and its value is assigned to an empty index in employeePass array
Function Name 20	**Delete Employee Credentials**
Function Prototype	bool deleteEmp(string employeeCode[], string employeePass[], string empCode, string empPass)
Description	This function is used to delete any existing employee credentials.
Function Return type and purpose	Boolean return type. Returns true if employee credentials are deleted successfully.
Parameter names and purpose of each parameter	1. **employeeCode[]** whole array is passed in this function and particular employee code is deleted from the desired index 2. **employeePass[]** whole array is passed in this function and particular employee password is deleted from the desired index 3. **empCode** this variable brings user input from main to this function and its value is assigned to an empty index in employeeCode array 4. **empPass** No purpose of this parameter. It is just useless.
Function Name 21	**View Employee List**
Function Prototype	void viewEmployee(string employeeCode[], string employeePass[])
Description	This function is used to view all employees and their credentials

Function Return type and purpose	No return types.
Parameter names and purpose of each parameter	1. **employeeCode[]** whole array is passed in this function and all employee codes are displayed which are present in this array 2. **employeePass[]** whole array is passed in this function and all employee password are displayed which are present in this array
Function Name 22	**Print Total Profit in text file**
Function Prototype	void print(double sale, double purchase)
Description	This function is called when admin desires to view all sales and purchases by their company. This prints all data in a text file.
Function Return type and purpose	No return types.
Parameter names and purpose of each parameter	1. **sale** This parameter contains sales admin made from selling item to users (total of selling prices) 2. **purchase** This parameter contains purchases made my employee when items into the list (total of cost prices)
Function Name 23	**Add Item Listing**
Function Prototype	bool addItem(string itemList[], int barcodeList[], int quantityList[], double costList[], double priceList[], string itemName, int barcode, int quantity, double cost, double price)
Description	This function used to add item in the inventory.

Function Return type and purpose	Boolean return type. Returns true if item is added successfully
Parameter names and purpose of each parameter	1. **itemList[]** Whole array is passed in this function and value is assigned to a particular index. 2. **barcodeList[]** Whole array is passed in this function and value is assigned to a particular index. 3. **quantityList[]** Whole array is passed in this function and value is assigned to a particular index. 4. **costList[]** Whole array is passed in this function and value is assigned to a particular index. 5. **priceList[]** Whole array is passed in this function and value is assigned to a particular index. 6. **itemName** Whole array is passed in this function and value is assigned to a particular index. 7. **barcode** this variable brings user input from main to this function and its value is assigned to an empty index in barcodeList array 8. **quantity** this variable brings user input from main to this function and its value is assigned to an empty index in quantityList array 9. **cost** this variable brings user input from main to this function and its value is assigned to an empty index in costList array 10. **price** this variable brings user input from main to this function and its value is assigned to an empty index in priceList array

Function Name 24	**Update Item Listing**
Function Prototype	bool updateItem(string itemList[], int barcodeList[], int quantityList[], double costList[], double priceList[], string itemName, int barcode, int quantity, double cost, double price)
Description	This function used to update existing item in the inventory.
Function Return type and purpose	Boolean return type. Returns true if item is updated successfully
Parameter names and purpose of each parameter	1. **itemList[]** Whole array is passed in this function and value is assigned to a particular index. 2. **barcodeList[]** Whole array is passed in this function and value is assigned to a particular index 3. **quantityList[]** Whole array is passed in this function and value is assigned to a particular index 4. **costList[]** Whole array is passed in this function and value is assigned to a particular index. 5. **priceList[]** Whole array is passed in this function and value is assigned to a particular index. 6. **itemName** this variable brings user input from main to this function and its value is assigned to an empty index in nameList array 7. **barcode** this variable brings user input from main to this function and its value is assigned to an empty index in barcodeList array 8. **quantity** this variable brings user input from main to this function and its value is assigned to an empty index in quantityList array 9. **cost** this variable brings user input from main to this function and its value is assigned to an empty index in costList array 10. **price** this variable brings user input from main to this function and its value is assigned to an empty index in priceList array

Function Name 25	**Delete Item Listing**
Function Prototype	bool deleteItem(string itemList[], int barcodeList[], int quantityList[], double costList[], double priceList[], string itemName, int barcode, int quantity, double cost, double price)
Description	This function used to delete existing item from the inventory.
Function Return type and purpose	Boolean return type. Returns true if item is deleted successfully
Parameter names and purpose of each parameter	1. **itemList[]** Whole array is passed in this function and value is assigned to a particular index. 2. **barcodeList[]** Whole array is passed in this function and value is assigned to a particular index. 3. **quantityList[]** Whole array is passed in this function and value is assigned to a particular index. 4. **costList[]** Whole array is passed in this function and value is assigned to a particular index. 5. **priceList[]** Whole array is passed in this function and value is assigned to a particular index. 6. **itemName** this variable brings user input from main to this function and corresponding data in parallel array is deleted or assigned their initial values 7. **Barcode** Useless 😊 8. **Quantity** Useless 😊 9. **Cost** Useless 😊 10. **Price** Useless 😊

Function Name 26	**View Item Listing**
Function Prototype	void viewItemsWithStock(string itemList[], int barcodeList[], int quantityList[], double costList[], double priceList[])
Description	This function displays all the item list and their data.
Function Return type and purpose	No return types.
Parameter names and purpose of each parameter	1. **itemList[]** displays all the data present in this array except initial values 2. **barcodeList[]** displays all the data present in this array except initial values 3. **quantityList[]** displays all the data present in this array except initial values 4. **costList[]** displays all the data present in this array except initial values 5. **priceList[]** displays all the data present in this array except initial values
Function Name 27	**View Item Listing with Barcode**
Function Prototype	void viewItemsWithBarcode(string itemList[], int barcodeList[], int quantityList[], double costList[], double priceList[])
Description	This function displays all item with their barcodes
Function Return type and purpose	No return types.
Parameter names and purpose of each parameter	1. **itemList[]** displays all the data present in this array except initial values 2. **barcodeList[]** displays all the data present in this array except initial values

	3. **quantityList[]** displays all the data present in this array except initial values 4. **costList[]** displays all the data present in this array except initial values 5. **priceList[]** displays all the data present in this array except initial values
Function Name 28	**Search Item from List**
Function Prototype	void searchItem(string itemList[], int barcodeList[], int quantityList[], string itemName)
Description	This function displays item and its data according to user's input
Function Return type and purpose	No return Types.
Parameter names and purpose of each parameter	1. **itemList[]** displays all the data present in this array except initial values 2. **barcodeList[]** displays all the data present in this array except initial values 3. **quantityList[]** displays all the data present in this array except initial values 4. **costList[]** displays all the data present in this array except initial values 5. **priceList[]** displays all the data present in this array except initial values 6. **itemName** this variable brings value from main to this function and item is searched in parallel arrays according to this

Function Name 29	**Initialize Items into Arrays**
Function Prototype	void initializeItems(string itemList[], int barcodeList[], int quantityList[], double costList[], double priceList[])
Description	This function stores a particular value to all indexes in the array.
Function Return type and purpose	No return types.
Parameter names and purpose of each parameter	1. **itemList[]** whole array is passed in this function and a particular initial value is assigned to it from itemList.txt file 2. **barcodeList[]** whole array is passed in this function and a particular initial value is assigned to it from itemList.txt file 3. **quantityList[]** whole array is passed in this function and a particular initial value is assigned to it from itemList.txt file 4. **costList[]** whole array is passed in this function and a particular initial value is assigned to it from itemList.txt file 5. **priceList[]** whole array is passed in this function and a particular initial value is assigned to it from itemList.txt file
Function Name 30	**Save Items**
Function Prototype	bool saveItems(string itemList[], int barcodeList[], int quantityList[], double costList[], double priceList[])
Description	This function saves data from arrays into itemList.txt file
Function Return type and purpose	Boolean return type. Returns true if items are saved in text file successfully

Parameter names and purpose of each parameter	1. **itemList[]** whole array is passed in this function and all data present in this array is stored in text file 2. **barcodeList[]** whole array is passed in this function and all data present in this array is stored in text file 3. **quantityList[]** whole array is passed in this function and all data present in this array is stored in text file 4. **costList[]** whole array is passed in this function and all data present in this array is stored in text file 5. **priceList[]** whole array is passed in this function and all data present in this array is stored in text file
Function Name 31	**Load Items**
Function Prototype	bool loadItems(string itemList[], int barcodeList[], int quantityList[], double costList[], double priceList[])
Description	This function load items from itemList.txt file into parallel arrays
Function Return type and purpose	Boolean return type. Returns true if items are loaded from text file successfully
Parameter names and purpose of each parameter	1. **itemList[]** whole array is passed in this function and all item names present in itemList.txt file is stored in this array 2. **barcodeList[]** whole array is passed in this function and all barcodes present in itemList.txt file is stored in this array 3. **quantityList[]**

	whole array is passed in this function and all item quantity present in itemList.txt file is stored in this array 4. **costList[]** whole array is passed in this function and all item costs present in itemList.txt file is stored in this array 5. **priceList[]** whole array is passed in this function and all item prices present in itemList.txt file is stored in this array
Function Name 32	**Initialize Users Purchased Items**
Function Prototype	void initializeUserItems(string userItemsList[], int userQuantityList[], double userPriceList[], double userTotal)
Description	This function assigns a particular initial value to user's arrays
Function Return type and purpose	No return types.
Parameter names and purpose of each parameter	1. **userItemsList[]** whole array is passed in this function and a particular initial value is assigned to it. 2. **userQuantityList[]** whole array is passed in this function and a particular initial value is assigned to it. 3. **userPriceList[]** whole array is passed in this function and a particular initial value is assigned to it. 4. **userTotal** This parameter is passed in this function and a zero initial value is assigned to it.

Function Name 33	**Users' Registration**
Function Prototype	bool registerUser(char phone[], char address[])
Description	This function is used to register new user and a text file is generated having phone number as its name. it contains user's phone number and home address.
Function Return type and purpose	Boolean return type. Returns true if user is registered successfully
Parameter names and purpose of each parameter	1. **phone[]** this variable contains user's input which is then stored in the new text file later. 2. **address[]** this variable contains user's input which is then stored in the new text file later.
Function Name 34	**Search Item**
Function Prototype	void userSearchItem(string itemList[], int quantityList[], double costList[], double priceList[], string itemName)
Description	This function takes item name as an input from user and displays item's data according to item name.
Function Return type and purpose	No return types.
Parameter names and purpose of each parameter	1. **itemList[]** displays all the data present in this array except initial values 2. **quantityList[]** displays all the data present in this array except initial values 3. **costList[]**

	displays all the data present in this array except initial values 4. **priceList[]** displays all the data present in this array except initial values 5. **itemName** contains user's input and item is searched according to its value.
Function Name 35	**Buy from Available Items List**
Function Prototype	void userItemList(string itemList[], int barcodeList[], int quantityList[], double costList[], double priceList[], string itemName, double userTotal, string userItemsList[], int userQuantityList[], double userPriceList[])
Description	This function is called when user want to buy any item from the inventory.
Function Return type and purpose	No return types.
Parameter names and purpose of each parameter	1. **itemList[]** displays all the data present in this array except initial values 2. **barcodeList[]** displays all the data present in this array except initial values 3. **quantityList[]** displays all the data present in this array except initial values 4. **costList[]** Useless. 5. **priceList[]** displays all the data present in this array except initial values 6. **itemName** takes user's input and assign its value to userItemList array 7. **userTotal** calculates bill of the user

	8. **userItemsList[]** an array which stores all the items purchased by the user 9. **userQuantityList[]** It contains all the corresponding quantities of item purchased by the user. 10. **userPriceList[]** It contains all the corresponding prices of item purchased by the user.
Function Name 36	**View All Items List**
Function Prototype	void viewAllItems(string itemList[], int quantityList[], double costList[], double priceList[], string itemName, double userTotal)
Description	Displays all items with their prices and quantity available.
Function Return type and purpose	No return types.
Parameter names and purpose of each parameter	1. **itemList[]** displays all the data present in this array except initial values 2. **barcodeList[]** displays all the data present in this array except initial values 3. **quantityList[]** displays all the data present in this array except initial values 4. **costList[]** Useless. 5. **priceList[]** displays all the data present in this array except initial values 6. **itemName** takes user's input and assign its value to userItemList array 7. **userTotal** calculates bill of the user

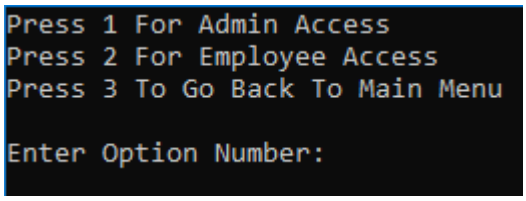
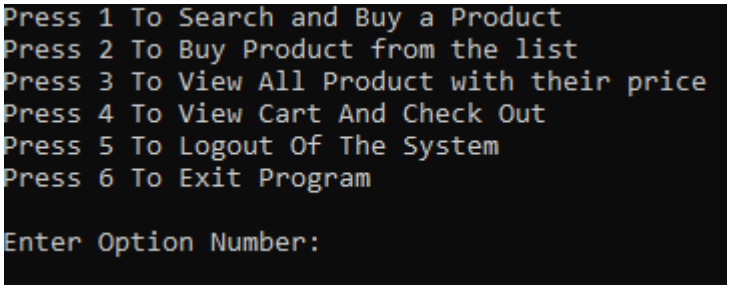
Function Name 37	**Users' Purchased Items List**
Function Prototype	void viewCart(string userItemsList[], int userQuantityList[], double userPriceList[], double userTotal)
Description	Displays all items, their quantity and total bill of the user.
Function Return type and purpose	No return types.
Parameter names and purpose of each parameter	1. **userItemsList[]** displays all data present in this array except initial values 2. **userQuantityList[]** displays all data present in this array except initial values 3. **userPriceList[]** displays all data present in this array except initial values 4. **userTotal** this variable stores total bill of the user.
Function Name 38	**Phone Number Validator**
Function Prototype	bool isValidPhoneNumber(char phone[])
Description	This function validates user's phone number input
Function Return type and purpose	Boolean return type. Returns true if phone number is valid
Parameter names and purpose of each parameter	1. **phone[]** contains input from the user which is going to be validated

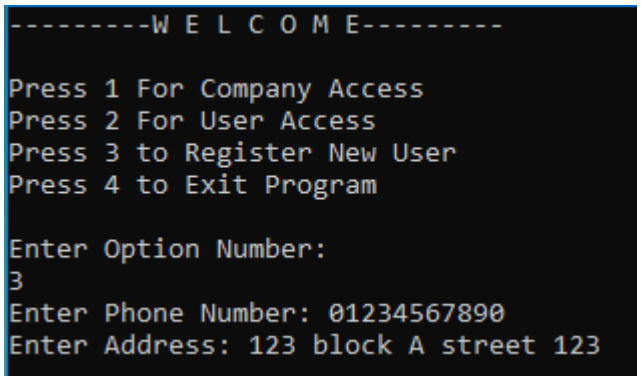
Function Name 39	**Employee Credentials Validator**
Function Prototype	bool isValidEmpCode(string empCode)
Description	This function validates employee code when admin tries to add employee data in program
Function Return type and purpose	Boolean return type. Returns true if employee code is valid
Parameter names and purpose of each parameter	1. **empCode** contains input from the user which is going to be validated
Function Name 40	**Item Name Validator**
Function Prototype	bool isValidItemName(string itemName)
Description	This function validates item name when employee tries to add item data in program
Function Return type and purpose	Boolean return type. Returns true if item name is valid
Parameter names and purpose of each parameter	1. **itemName** contains input from the user which is going to be validated
Function Name 41	**Barcode Validator**
Function Prototype	bool isValidBarcode(int barcode)
Description	This function validates item barcode number when employee tries to add item data in program
Function Return type and purpose	Boolean return type. Returns true if barcode number is valid
Parameter names and purpose of each parameter	1. **barcode** contains input from the user which is going to be validated

Function Name 42	**Quantity Validator**
Function Prototype	bool isValidQuantity(int quantity)
Description	This function validates item quantity number when employee tries to add item data in program
Function Return type and purpose	Boolean return type. Returns true if quantity input is valid.
Parameter names and purpose of each parameter	1. **quantity** contains input from the user which is going to be validated
Function Name 43	**Cost and Price Validator**
Function Prototype	bool isValidPrice(double cost, double price)
Description	This function validates item price and cost when employee tries to add item data in program
Function Return type and purpose	Boolean return type. Returns true if
Parameter names and purpose of each parameter	1. **price** contains input from the user which is going to be validated 2. **cost** contains input from the user which is going to be validated
Function Name 44	**Search Employee Credentials**
Function Prototype	int searchEmployeeCode(string employeeCode[], string empCode)
Description	This function is to search an existing employee and its credentials

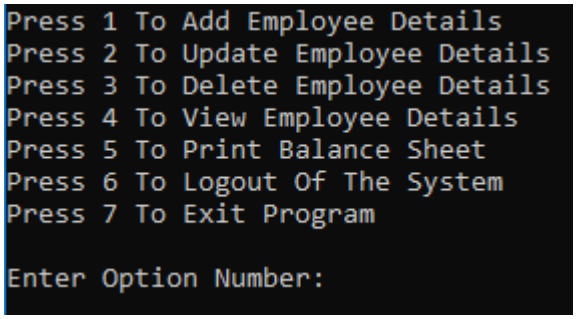
Function Return type and purpose	Integer return type. Return a particular index of the array when employee code is found.
Parameter names and purpose of each parameter	1. **employeeCode[]** whole array is passed into this function to search employee code according to user input in empCode variable 2. **empCode** Brings input from the user into this function to match value with array
Function Name 45	**Search Item Name**
Function Prototype	int searchItemName(string itemList[], string itemName)
Description	This function is used to search item and its data according to user input.
Function Return type and purpose	Integer return type. Return a particular index of the array when item is found
Parameter names and purpose of each parameter	1. **itemList[]** whole array is passed into this function to search item name according to user input in itemName variable 2. **itemName** Brings input from the user into this function to match value with array
Format of files	
File Name 1	**CS142F20R144**
File Type	Source Code
File Format	C++ File (.cpp)

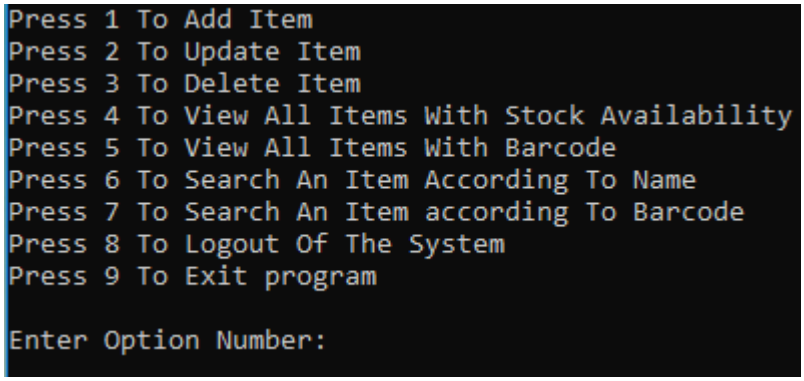
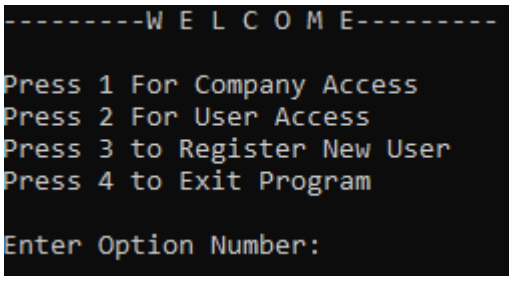
File Name 2	**Employee**
File Type	Input File
File Format	Text File (.txt)
File Name 3	**itemList**
File Type	Input File
File Format	Text File (.txt)
File Name 4	**profit**
File Type	Output File
File Format	Text File (.txt)
Details of Interfaces	
Main Menu Details (void welcome())	
Option 1	***Company Access***
Purpose	*To view company menu.*
Input/ Output	*Integer input.*
Validation	*Only options from 1 to 4 can be selected.*
Test cases with sample inputs and outputs	*No test cases*
Which array is used on this option for data storage and retrieval?	*No array is used in this option.*

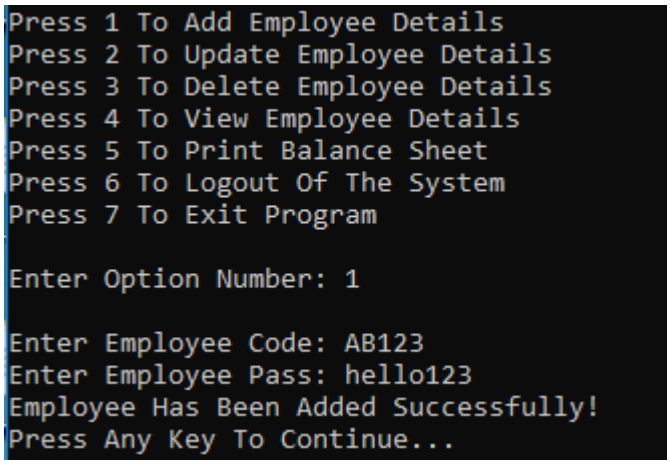
Which function is used to call for this option?	*void companyMenu()*
Screenshot	
Option 2	***User Access***
Purpose	*To view user menu.*
Input/ Output	*Integer input.*
Validation	*Only options from 1 to 9 can be selected.*
Test cases with sample inputs and outputs	*No test cases*
Which array is used on this option for data storage and retrieval?	*No array is used in this option.*
Which function is used to call for this option?	*void userMenu()*
Screenshot	

Option 3	***Register New User***
Purpose	*To register a new user who can access the program*
Input/ Output	*Phone Number and Home Address* *Output is in form of a text file.*
Validation	*Phone Number and Home Address are validated according to standards.* *Phone:01234567890 (11 digits)* *Address: Any input in for of string.*
Test cases with sample inputs and outputs	*Phone: 03331234567* *Address: 123 ABC road, Lahore* *Output: Text file generated with phone number as its name.*
Which array is used on this option for data storage and retrieval?	*char phone[11]* *char address[200]*
Which function is used to call for this option?	*bool registerUser(char phone[], char address[])*
Screenshot	
Option 4	***Exit Program***
Purpose	*To save and exit program with return value 0*

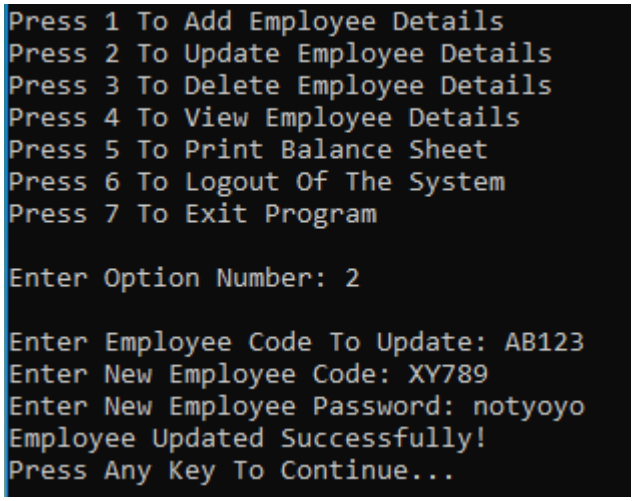
Input/ Output	*No input required.*
Validation	*No validation required*
Test cases with sample inputs and outputs	*No test cases*
Which array is used on this option for data storage and retrieval?	string usersList[MAX_USERS], passwordList[MAX_USERS] string employeeCode[MAX_EMPLOYEE], employeePass[MAX_EMPLOYEE] string userItemsList[MAX_ITEMS] int userQuantityList[MAX_ITEMS] double userPriceList[MAX_ITEMS]
Which function is used to call for this option?	*Void Exit()*
Screenshot	
Company Menu Details	
Option 1	***Admin Access***
Purpose	*To view admin menu.*
Input/ Output	*Integer input.*

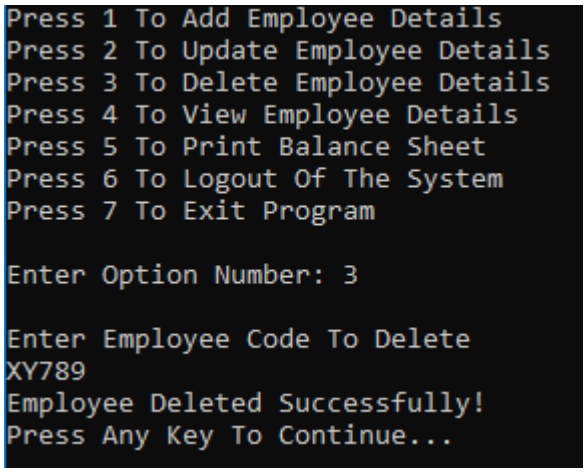
Validation	*Only options from 1 to 7 can be selected.*
Test cases with sample inputs and outputs	*No test cases*
Which array is used on this option for data storage and retrieval?	*No array is used in this option.*
Which function is used to call for this option?	*void adminMenu()*
Screenshot	
Option 2	***Employee Access***
Purpose	*To view Employee menu.*
Input/ Output	*Integer input from 1 to 9.*
Validation	*No validation*
Test cases with sample inputs and outputs	*No test cases*
Which array is used on this option for data storage and retrieval?	*No array is used in this option.*
Which function is used to call for this option?	*void employeeMenu()*

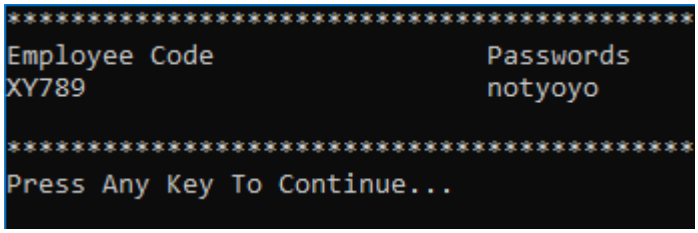
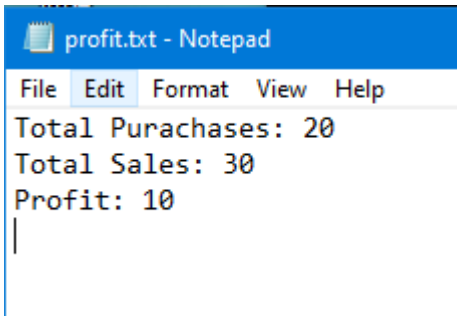
Screenshot	
Option 3	***Go Back to Main Menu***
Purpose	*To go back to previous menu(welcome screen).*
Input/ Output	*Input=3*
Validation	*No validation*
Test cases with sample inputs and outputs	*No test cases.*
Which array is used on this option for data storage and retrieval?	*No array used in this option.*
Which function is used to call for this option?	*void welcome()*
Screenshot	

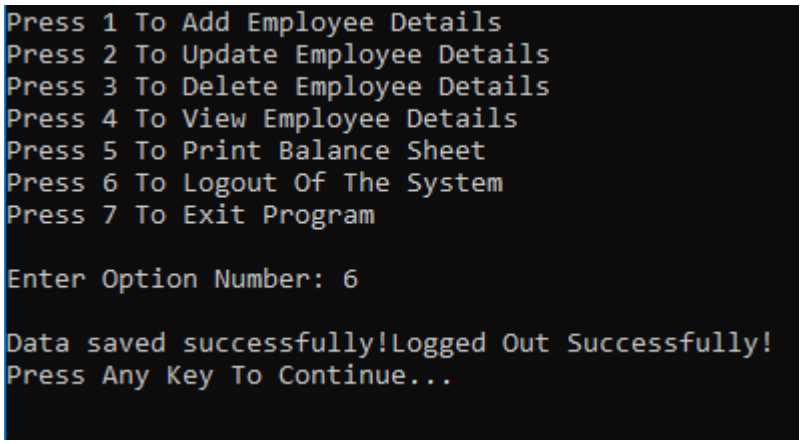
Admin Menu Details(Submenu of Company Menu)	
Option 1	***Add Employee Details***
Purpose	*To add new employee*
Input/ Output	*Employee code and password*
Validation	*Employee Code: AB123* *Employee Password: any input.*
Test cases with sample inputs and outputs	*Enter Employee Code: AB123* *Enter Employee Pass: hello123* *Employee Has Been Added Successfully!* *Enter Employee Code: 123AB* *Enter Employee Pass: hello123* *Error! Invalid employee code.*
Which array is used on this option for data storage and retrieval?	*string employeeCode[MAX_EMPLOYEE]* *string employeePass[MAX_EMPLOYEE]*
Which function is used to call for this option?	*bool addEmployee(string employeeCode[], string employeePass[], string empCode, string empPass)*
Screenshot	

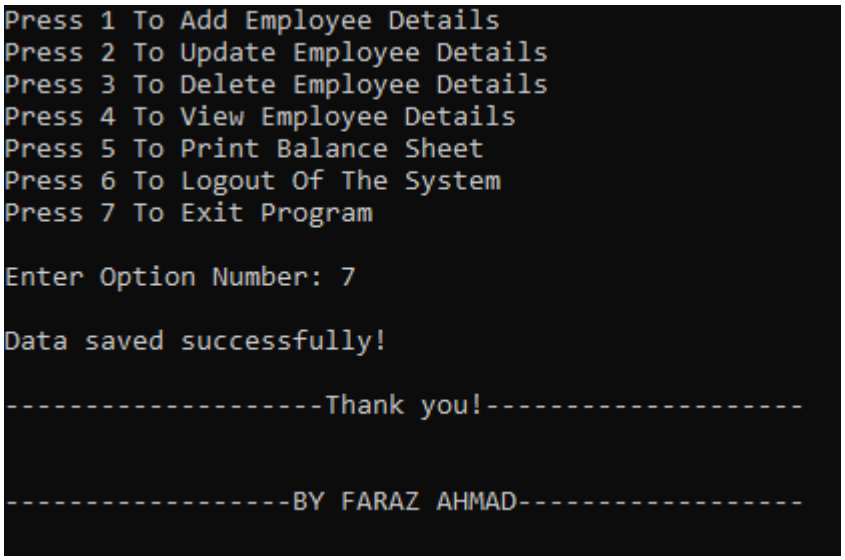
Option 2	***Update Employee Details***
Purpose	*To update credentials of any existing employee*
Input/ Output	*Employee code to update, new employee code and new password*
Validation	*Employee Code: AB123* *Employee Password: any input.*
Test cases with sample inputs and outputs	*Enter Employee code to update: XY789* *Enter new Employee Code: AB123* *Enter New Employee Pass: hello123* *Employee Updated Successfully!* *Enter Employee code to update: XY785* *Error! No employee found with this code.* *Enter Employee Code: 123AB* *Enter Employee Pass: hello123* *Error! Invalid employee code.*
Which array is used on this option for data storage and retrieval?	*string employeeCode[MAX_EMPLOYEE]* *string employeePass[MAX_EMPLOYEE]*
Which function is used to call for this option?	*bool updateEmployee(string employeeCode[], string employeePass[], string empCode, string empPass)*

Screenshot	
Option 3	***Delete Employee Details***
Purpose	*To delete credentials of any existing employee*
Input/ Output	*Employee code to delete.* *Employee Deleted Successfully!*
Validation	*Employee Code: AB123* *Employee Password: any input.*
Test cases with sample inputs and outputs	*Enter Employee code to update: XY789* *Employee Deleted Successfully!* *Enter Employee code to update: XY785* *Error! No employee found with this code.* *Enter Employee Code: 123AB* *Error! Invalid employee code.*
Which array is used on this option for data storage and retrieval?	*string employeeCode[MAX_EMPLOYEE]* *string employeePass[MAX_EMPLOYEE]*

Which function is used to call for this option?	*bool deleteEmp(string employeeCode[], string employeePass[], string empCode, string empPass)*
Screenshot	
Option 4	***View Employee Details***
Purpose	*To view all employees with their credentials*
Input/ Output	*No input required.* *Output: List of all employees*
Validation	*No validation required*
Test cases with sample inputs and outputs	*Employee Code* *Password* *AB123* *hello123*
Which array is used on this option for data storage and retrieval?	*string employeeCode[MAX_EMPLOYEE]* *string employeePass[MAX_EMPLOYEE]*
Which function is used to call for this option?	*void viewEmployee(string employeeCode[], string employeePass[])*

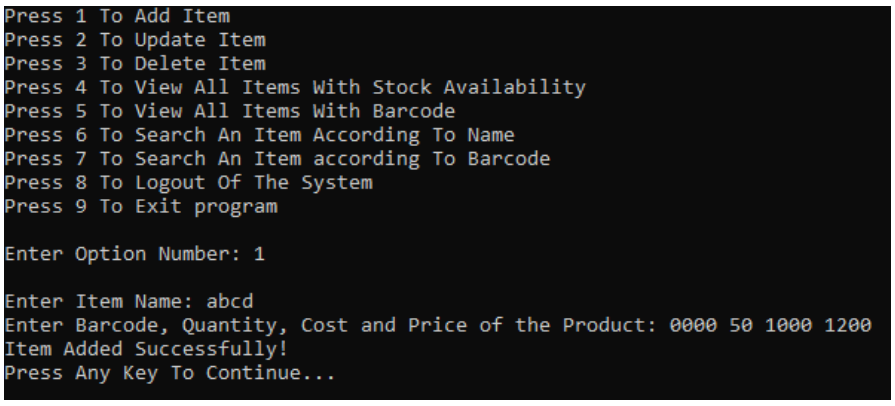
Screenshot	
Option 5	***Print Balance Sheet***
Purpose	*To print all sales and purchases in a text file*
Input/ Output	*No input required* *Output: a text file*
Validation	*No validation*
Test cases with sample inputs and outputs	*No test case required.*
Which array is used on this option for data storage and retrieval?	double userPriceList[MAX_ITEMS] int userQuantityList[MAX_ITEMS] *double costList[MAX_ITEMS]* *int quantityList[MAX_ITEMS]*
Which function is used to call for this option?	*void print(double sale, double purchase)*
Screenshot	

Option 6	***Logout from The System***
Purpose	*To save and exit program with return value 0*
Input/ Output	*No input required.*
Validation	*No validation required*
Test cases with sample inputs and outputs	*No test cases*
Which array is used on this option for data storage and retrieval?	string usersList[MAX_USERS], passwordList[MAX_USERS] string employeeCode[MAX_EMPLOYEE], employeePass[MAX_EMPLOYEE] string userItemsList[MAX_ITEMS] int userQuantityList[MAX_ITEMS] double userPriceList[MAX_ITEMS]
Which function is used to call for this option?	*void logout()*
Screenshot	
Option 7	***Exit Program***
Purpose	*To save and exit program with return value 0*
Input/ Output	*No input required.*

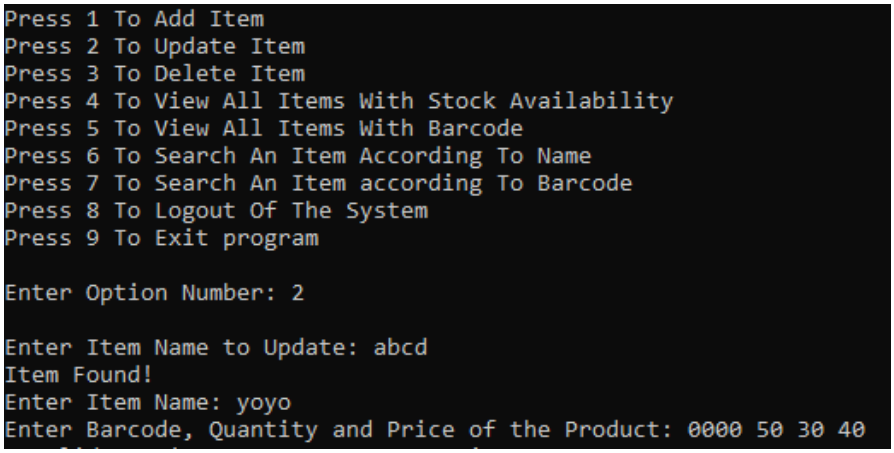
Validation	*No validation required*
Test cases with sample inputs and outputs	*No test cases*
Which array is used on this option for data storage and retrieval?	string usersList[MAX_USERS], passwordList[MAX_USERS] string employeeCode[MAX_EMPLOYEE], employeePass[MAX_EMPLOYEE] string userItemsList[MAX_ITEMS] int userQuantityList[MAX_ITEMS] double userPriceList[MAX_ITEMS]
Which function is used to call for this option?	*Void Exit()*
Screenshot	 Press 1 To Add Employee Details Press 2 To Update Employee Details Press 3 To Delete Employee Details Press 4 To View Employee Details Press 5 To Print Balance Sheet Press 6 To Logout Of The System Press 7 To Exit Program Enter Option Number: 7 Data saved successfully! ----------Thank you!---------- ----------BY FARAZ AHMAD----------

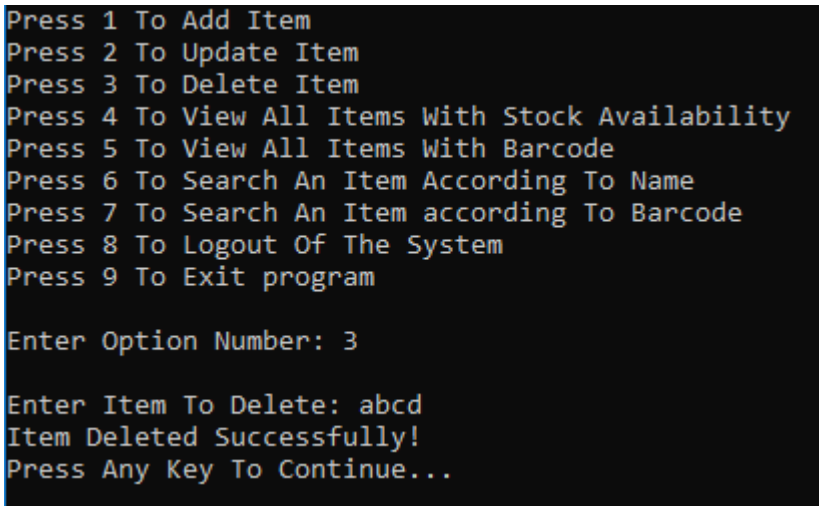
Employee Menu Details(Submenu of Company Menu)

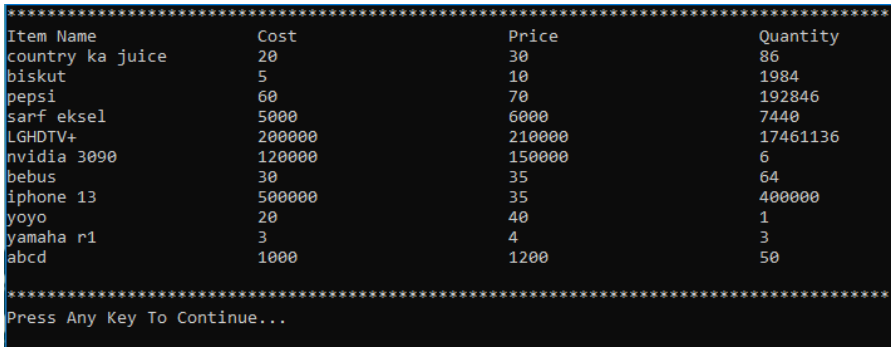
Option 1	***Add Item***
Purpose	*To add new item in the listing*
Input/ Output	*Input: Item name, Quantity, Barcode, Cost, Price* *Output: Item added successfully! OR Error adding item data!*

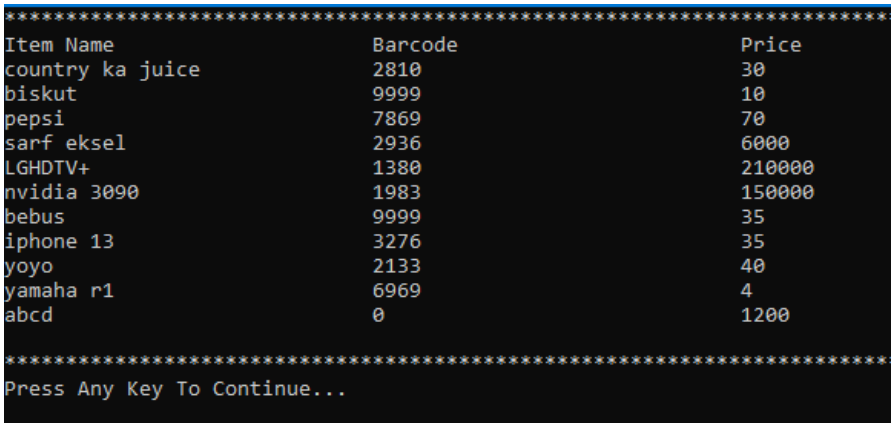
Validation	*Item name can only consist of letters.* *Quantity is in integer* *Price and cost are double* *Barcode can only consist of 4-digit number.*
Test cases with sample inputs and outputs	*Enter item name: abcd* *Enter Barcode, Quantity, Cost and Price of the Product:* *0000 50 1000 1200* *Item added successfully!* *Enter item name: abcd3* *Enter Barcode, Quantity, Cost and Price of the Product:* *12345 50 1000 1200* *Error adding item data!*
Which array is used on this option for data storage and retrieval?	*string itemList[MAX_ITEMS]* *double priceList[MAX_ITEMS]* *int quantityList[MAX_ITEMS]* *int barcodeList[MAX_ITEMS]* *double costList[MAX_ITEMS]*
Which function is used to call for this option?	*bool addItem(string itemList[], int barcodeList[], int quantityList[], double costList[], double priceList[], string itemName, int barcode, int quantity, double cost, double price)*
Screenshot	

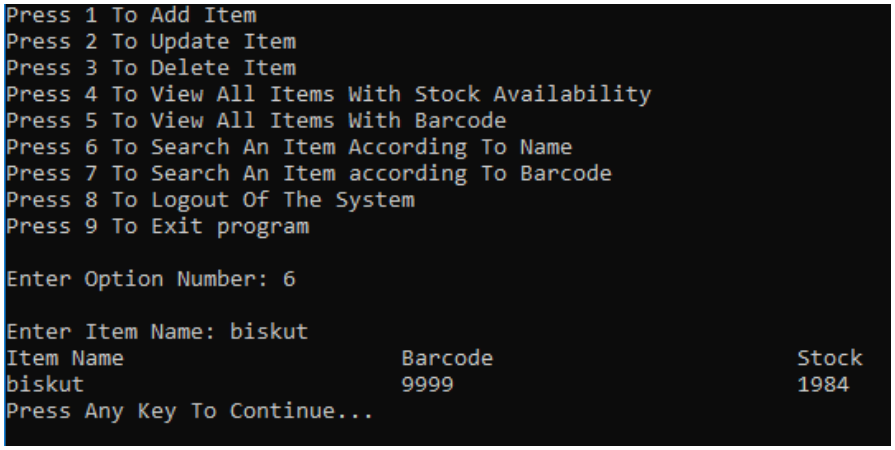
Option 2	***Update Item***
Purpose	*To update existing item in the listing*
Input/ Output	*Input: Item name, Quantity, Barcode, Cost, Price* *Output: Item updated successfully! OR Error updating item data!*
Validation	*Item name can only consist of letters.* *Quantity is in integer* *Price and cost are double* *Barcode can only consist of 4-digit number.*
Test cases with sample inputs and outputs	*Enter item name to update qwerty* *Enter new item name: abcd* *Enter new Barcode, Quantity, Cost and Price of the Product:* *0000 50 1000 1200* *Item added successfully!* *Enter new item name: abcd3* *Enter new Barcode, Quantity, Cost and Price of the Product:* *12345 50 1000 1200* *Error adding item data!*
Which array is used on this option for data storage and retrieval?	*string itemList[MAX_ITEMS]* *double priceList[MAX_ITEMS]* *int quantityList[MAX_ITEMS]* *int barcodeList[MAX_ITEMS]* *double costList[MAX_ITEMS]*
Which function is used to call for this option?	*bool updateItem(string itemList[], int barcodeList[], int quantityList[], double costList[], double priceList[], string itemName, int barcode, int quantity, double cost, double price)*

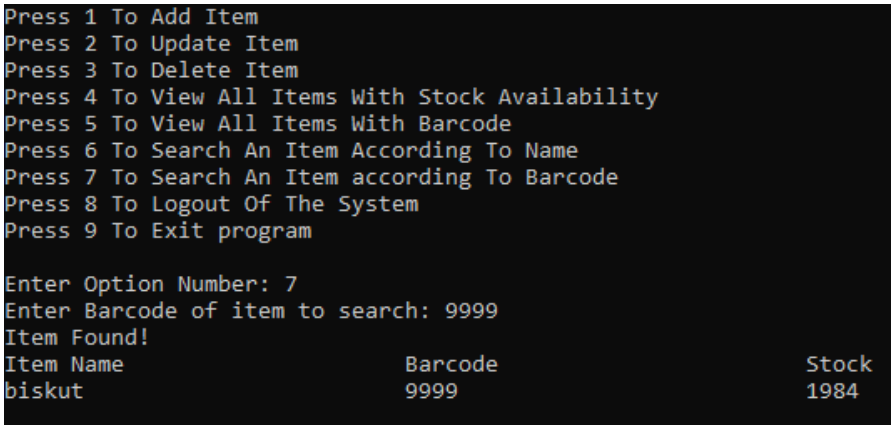
Screenshot	
Option 3	***Delete Item***
Purpose	*To delete item for the listing*
Input/ Output	*Input: Item name* *Output: item deleted successfully! OR Item does not exist!*
Validation	*Item name can only consist of letters.*
Test cases with sample inputs and outputs	*Enter item name to delete: abcd* *item deleted successfully!* *Enter item name to delete: wxyz* *Item does not exist!*
Which array is used on this option for data storage and retrieval?	*string itemList[MAX_ITEMS]* *double priceList[MAX_ITEMS]* *int quantityList[MAX_ITEMS]* *int barcodeList[MAX_ITEMS]* *double costList[MAX_ITEMS]*
Which function is used to call for this option?	*bool deleteItem(string itemList[], int barcodeList[], int quantityList[], double costList[], double priceList[], string*

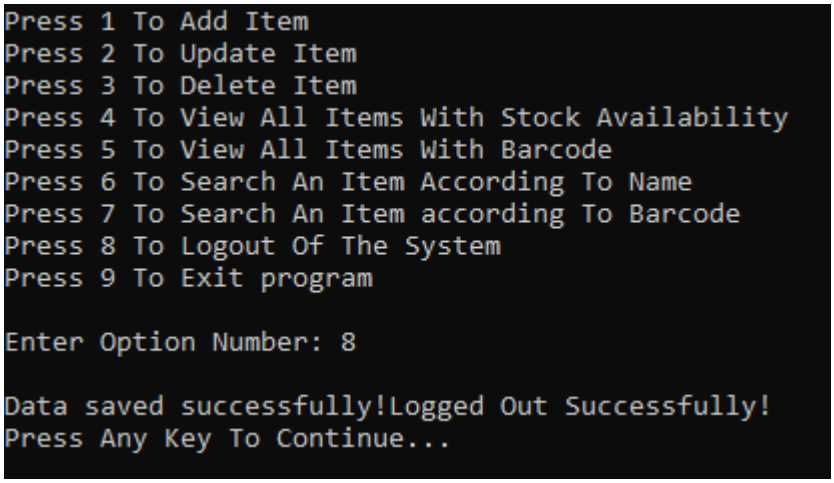
	itemName, int barcode, int quantity, double cost, double price)
Screenshot	
Option 4	***View All Items with Stock Availability***
Purpose	*To View All Items with Stock Availability*
Input/ Output	*No input* *Output: List of items with stock availability*
Validation	*No validation*
Test cases with sample inputs and outputs	*Item name cost price quantity* *Biskut 5 10 1984*
Which array is used on this option for data storage and retrieval?	*string itemList[MAX_ITEMS]* *int quantityList[MAX_ITEMS]* *int barcodeList[MAX_ITEMS]*
Which function is used to call for this option?	*void viewItemsWithStock(string itemList[], int barcodeList[], int quantityList[], double costList[], double priceList[])*

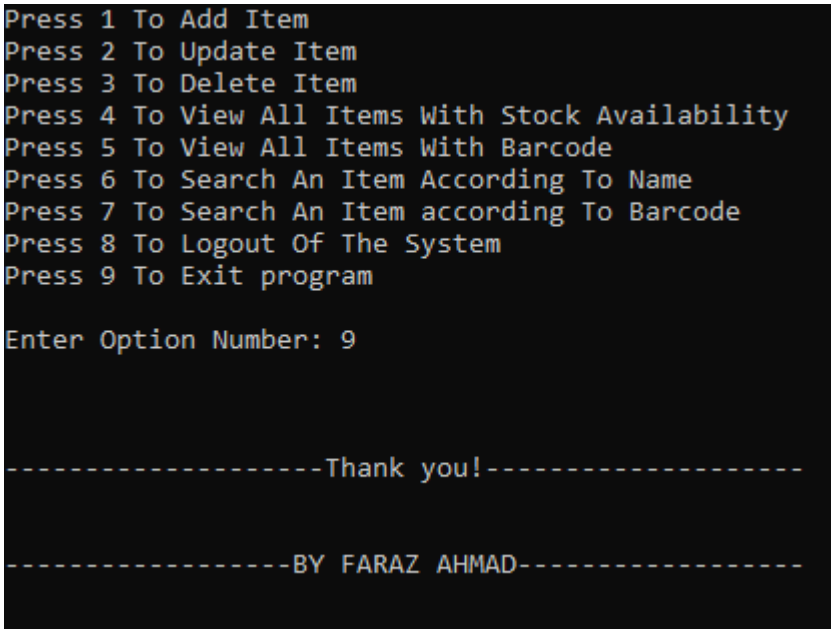
Screenshot	
Option 5	***View All Items with Barcode***
Purpose	*To View All Items with their Barcode only*
Input/ Output	*No input* *Output: List of items with their barcodes*
Validation	*No validation*
Test cases with sample inputs and outputs	*Item name* *barcode* *price* *Biskut* *9999* *10*
Which array is used on this option for data storage and retrieval?	*string itemList[MAX_ITEMS]* *double priceList[MAX_ITEMS]* *int quantityList[MAX_ITEMS]* *int barcodeList[MAX_ITEMS]* *double costList[MAX_ITEMS]*
Which function is used to call for this option?	*void viewItemsWithBarcode(string itemList[], int barcodeList[], int quantityList[], double costList[], double priceList[])*

Screenshot	
Option 6	***Search an Item According to Name***
Purpose	*To Search an Item According to Name*
Input/ Output	*No input* *Output: item data*
Validation	*No validation*
Test cases with sample inputs and outputs	*Enter item name to search: Abcd* *Item Name barcode quantity cost price* *Abcd 0000 30 1000 1200*
Which array is used on this option for data storage and retrieval?	*string itemList[MAX_ITEMS]* *double priceList[MAX_ITEMS]* *int quantityList[MAX_ITEMS]* *int barcodeList[MAX_ITEMS]* *double costList[MAX_ITEMS]*
Which function is used to call for this option?	*void searchItem(string itemList[], int barcodeList[], int quantityList[], string itemName)*

Screenshot	
Option 7	***Search an Item according To Barcode***
Purpose	*To Search an Item according To Barcode*
Input/ Output	*No input* *Output: item data*
Validation	*No validation*
Test cases with sample inputs and outputs	*Enter item barcode to search: 0000* *Item Name barcode stock* *Abcd 0000 30*
Which array is used on this option for data storage and retrieval?	*string itemList[MAX_ITEMS]* *double priceList[MAX_ITEMS]* *int quantityList[MAX_ITEMS]* *int barcodeList[MAX_ITEMS]* *double costList[MAX_ITEMS]*
Which function is used to call for this option?	*void searchItem(string itemList[], int barcodeList[], int quantityList[], string itemName)*

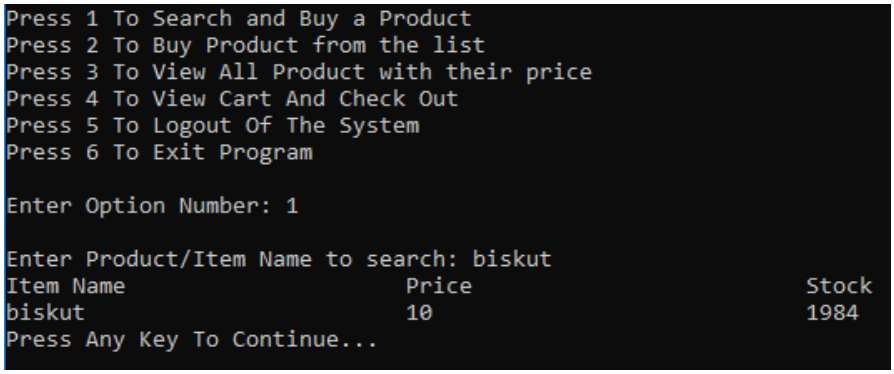
Screenshot	
Option 8	***Logout from The System***
Purpose	*To save and exit program with return value 0*
Input/ Output	*No input required.*
Validation	*No validation required*
Test cases with sample inputs and outputs	*No test cases*
Which array is used on this option for data storage and retrieval?	string usersList[MAX_USERS], passwordList[MAX_USERS] string employeeCode[MAX_EMPLOYEE], employeePass[MAX_EMPLOYEE] string userItemsList[MAX_ITEMS] int userQuantityList[MAX_ITEMS] double userPriceList[MAX_ITEMS]
Which function is used to call for this option?	*void logout()*

Screenshot	
Option 9	***Exit Program***
Purpose	*To save and exit program with return value 0*
Input/ Output	*No input required.*
Validation	*No validation required*
Test cases with sample inputs and outputs	*No test cases*
Which array is used on this option for data storage and retrieval?	string usersList[MAX_USERS], passwordList[MAX_USERS] string employeeCode[MAX_EMPLOYEE], employeePass[MAX_EMPLOYEE] string userItemsList[MAX_ITEMS] int userQuantityList[MAX_ITEMS] double userPriceList[MAX_ITEMS]
Which function is used to call for this option?	*Void Exit()*

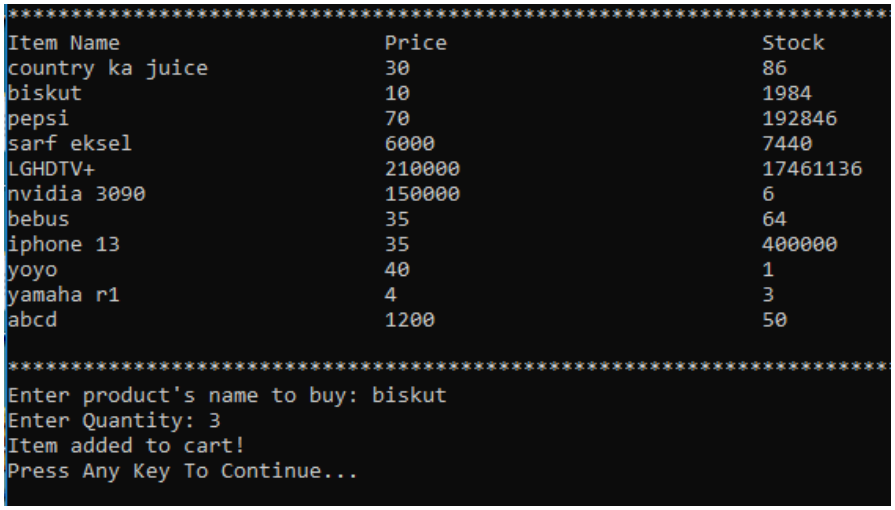
Screenshot	

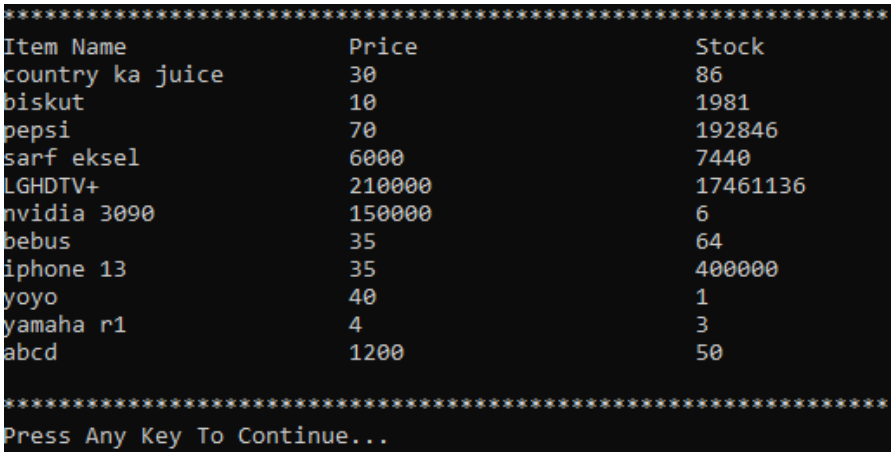
User Menu Details

Option 1	***Search and Buy a Product***
Purpose	*User can purchase item from the list*
Input/ Output	*Input: item name and its quantity.*
Validation	*It is checked that whether the item is present or not and the quantity user wish to buy is also less than the quantity present in the inventory.*
Test cases with sample inputs and outputs	*Enter item name to buy: abcd* *Enter quantity you wish to buy: 123* *Item added to cart successfully!* *Enter item name to buy: abcd* *Enter quantity you wish to buy: 12* *Error adding item to cart OR item was not found.*

Which array is used on this option for data storage and retrieval?	*string itemList[MAX_ITEMS]* *double priceList[MAX_ITEMS]* *int quantityList[MAX_ITEMS]*
Which function is used to call for this option?	*void userSearchItem(string itemList[], int quantityList[], double costList[], double priceList[], string itemName)*
Screenshot	

Option 2	***Buy Product from the list***
Purpose	*Whole list of items is displayed, and user can choose any item.*
Input/ Output	*Input: item name and its quantity.*
Validation	*It is checked that whether the item is present or not and the quantity user wish to buy is also less than the quantity present in the inventory.*
Test cases with sample inputs and outputs	*Enter item name to buy: abcd* *Enter quantity you wish to buy: 123* *Item added to cart successfully!* *Enter item name to buy: abcd* *Enter quantity you wish to buy: 12* *Error adding item to cart OR item was not found.*

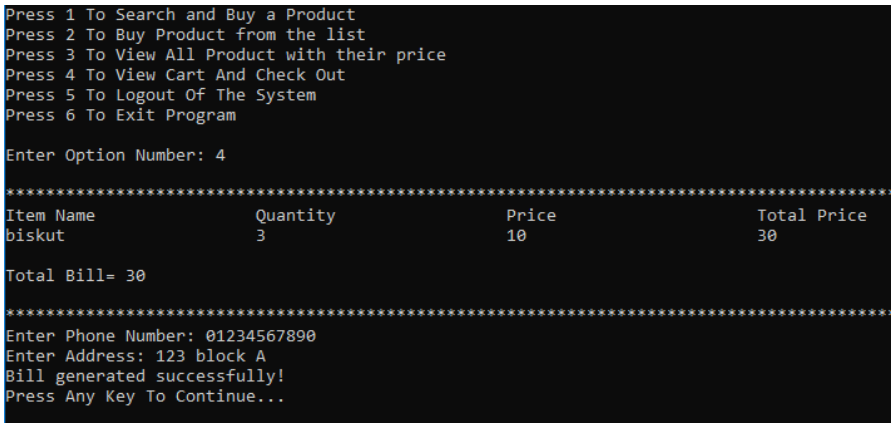
Which array is used on this option for data storage and retrieval?	*string itemList[MAX_ITEMS]* *double priceList[MAX_ITEMS]* *int quantityList[MAX_ITEMS]* *string userItemsList[MAX_ITEMS]* *int userQuantityList[MAX_ITEMS]*
Which function is used to call for this option?	*void userItemList(string itemList[], int barcodeList[], int quantityList[], double costList[], double priceList[], string itemName, double userTotal, string userItemsList[], int userQuantityList[], double userPriceList[])*
Screenshot	
Option 3	***View All Product with their price***
Purpose	*Displays all item with their prices and quantities. Just for viewing purpose.*
Input/ Output	*No input required.*
Validation	*No validation required.*
Test cases with sample inputs and outputs	*Item name price quantity*

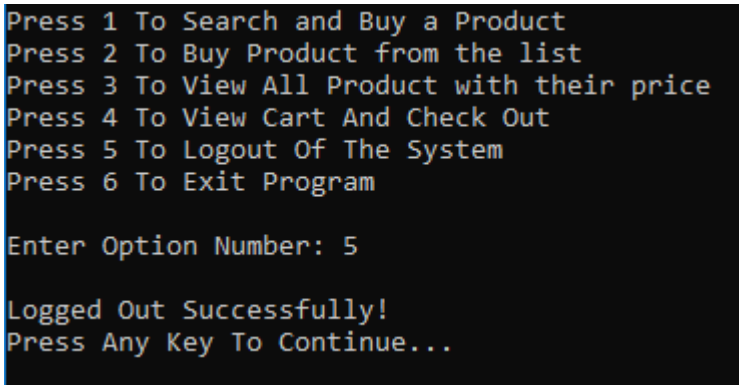
Which array is used on this option for data storage and retrieval?	*string itemList[MAX_ITEMS]* *double priceList[MAX_ITEMS]* *int quantityList[MAX_ITEMS]*
Which function is used to call for this option?	*void viewAllItems(string itemList[], int quantityList[], double costList[], double priceList[], string itemName, double userTotal)*
Screenshot	

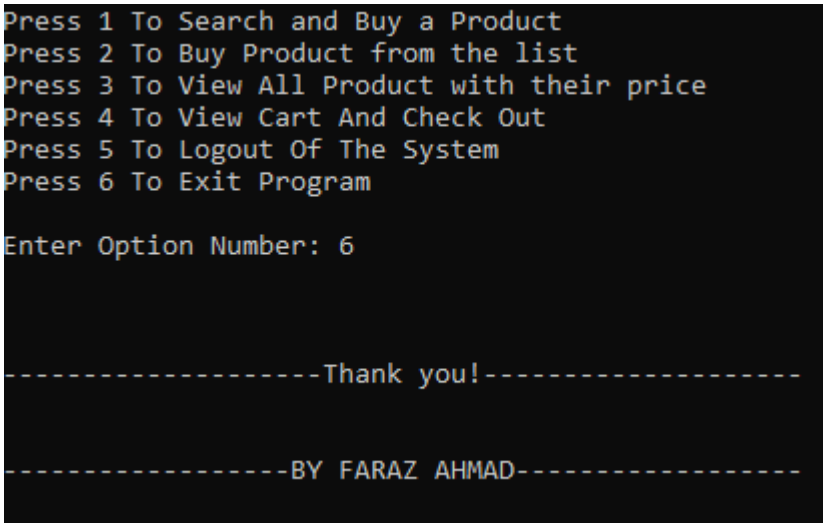
```
********************************************************************************
Item Name                 Price                     Stock
country ka juice          30                        86
biskut                    10                        1981
pepsi                     70                        192846
sarf eksel                6000                      7440
LGHDTV+                   210000                    17461136
nvidia 3090               150000                    6
bebus                     35                        64
iphone 13                 35                        400000
yoyo                      40                        1
yamaha r1                 4                         3
abcd                      1200                      50

********************************************************************************
Press Any Key To Continue...
```

Option 4	***View Cart and Check Out***
Purpose	*User bill is generated according to their purchases in a form of a text file.*
Input/ Output	*Input: Phone number and address* *Output: text file containing their bill*
Validation	*Phone number and home address is validated.*
Test cases with sample inputs and outputs	*Item name quantity price total price* *Biskut 3 10 30*
Which array is used on this option for data storage and retrieval?	*string userItemsList[MAX_ITEMS]* *int userQuantityList[MAX_ITEMS]* *double userPriceList[MAX_ITEMS]*

	char phone[11 *char address[100]*
Which function is used to call for this option?	*void viewCart(string userItemsList[], int userQuantityList[], double userPriceList[], double userTotal)*
Screenshot	
Option 5	***Logout from The System***
Purpose	*To save and exit program with return value 0*
Input/ Output	*No input required.*
Validation	*No validation required*
Test cases with sample inputs and outputs	*No test cases*
Which array is used on this option for data storage and retrieval?	string usersList[MAX_USERS], passwordList[MAX_USERS] string employeeCode[MAX_EMPLOYEE], employeePass[MAX_EMPLOYEE] string userItemsList[MAX_ITEMS] int userQuantityList[MAX_ITEMS] double userPriceList[MAX_ITEMS]
Which function is used to call for this option?	*void logout()*

Screenshot	
Option 6	***Exit Program***
Purpose	*To save and exit program with return value 0*
Input/ Output	*No input required.*
Validation	*No validation required*
Test cases with sample inputs and outputs	*No test cases*
Which array is used on this option for data storage and retrieval?	string usersList[MAX_USERS], passwordList[MAX_USERS] string employeeCode[MAX_EMPLOYEE], employeePass[MAX_EMPLOYEE] string userItemsList[MAX_ITEMS] int userQuantityList[MAX_ITEMS] double userPriceList[MAX_ITEMS]
Which function is used to call for this option?	*Void Exit()*

Screenshot	

Note:

- Add the details of all options of the main menu in the same format described above.
- In case of sub, menus, add the menu in the same format as the main menu.
- In case of the login screen etc., explain it after all menus, in the same format as the menu option.