

FUNCTIONS SPECIFICATIONS

Function Name	Description	Input Parameter	Return Data
TEXT FILES			
1. saveUsers	Use to save the data information of the users to the text file	1. (UserInfo) *aUserData = array list of users 2. (int) nCountIndex = index of array list of users	No return value
2. openUsers	Use to open the text file of Users.txt	1. (UserInfo) *aUserData = array list of users 2. (int) *nCountIndex = index of array list of users	1. (UserInfo) *aUserData = returns the user information to the array of users 2. (int) *CountIndex = index will update after reading the text file
3. saveItems	Use to save the item information of the seller items to the text file	1. (ItemInfo) *aItemData = array list of items 2. (int) nSIndex = index of array list of items	No return value
4. openItems	Use to open the items information of the seller items from the text file Items.txt	1. (ItemInfo) *aItemData = array list of items 2. (int) *nSIndex = index of array list of items	1. (ItemInfo) *aItemData = returns the item information to the array of items 2. (int) *nSIndex = index will update after reading the text file
5. saveCart	Use to save the cart information of the buyer to “id number”.bag in binary file	1. (CartInfo) *aCartData = array list of items from the cart 2. (int) nCartIndex = index of array list of items from the cart 3. (int) ID = id of the buyer / user	No return value
6. openCart	Use to open the item information of the cart from the binary file “id number”.bag	1. (CartInfo) *aCartData = array list of items from the cart 2. (int) *nCartIndex = index of array list of items from the cart 3. (int) ID = id of the buyer / user	1. (CartInfo) *aCartData = returns the item information to the array of cart of the certain user 2. (int) *nCartIndex = index will update after reading the binary file
7. saveTransac	Use to save the transaction item information of the list of transactions to binary file Transactions.dat	1. (TransacInfo) TransacData = a single structure of the transaction information	No return value
8. openTransac	Use to open the transaction item information of the list of transactions from the binary file Transaction.dat. This also stores the items in the array whose date falls between the start and end date.	1. (TransacInfo) *aTransacData = array list of transaced items 2. (int) *nTransacIndex = index of array list of transaced items	1. (TransacInfo) *aTransacData = stores the item information that is between the start and end date 2. (int) *nTransacIndex = index will update after reading the binary file
MENU / OPTIONS			
9. MainMenu	Use for main menu selection	No input parameter	(char)cChoice = choice entered by the user from the list options
10. UserMenu	Use for user menu selection	No input parameter	(char)cChoice = choice entered by the user from the list options
11. SellMenu	Use for sell menu selection	No input parameter	(char)cChoice = choice entered by the user from the list options
12. editStock	Use for edit stock selection	No input parameter	(char)cChoice = choice entered by the user from the list options

13. BuyMenu	Use for buy menu selection	No input parameter	(char)cChoice = choice entered by the user from the list options
14. showTransacMenu	Use for Transaction menu selection (new feature)	No input parameter	(char)cChoice = choice entered by the user from the list options
15. editCart	Use for edit cart selection	No input parameter	(char)cChoice = choice entered by the user from the list options
16. checkOut	Use for checkout selection	No input parameter	(char)cChoice = choice entered by the user from the list options
17. AdminMenu	Use for admin menu selection	No input parameter	(char)cChoice = choice entered by the user from the list options
18. Confirm	Use for Option to confirm, re-enter, or cancel selection	No input parameter	(char)cChoice = choice entered by the user from the list options
CHECKING			
19. checkUserID	Use to check the user id from the list of users	1. (UserInfo) *aUserData = array list of users 2. (int) nCountIndex = index of array list of users 3. (int) checkID = user id	(Int) nCheck = 1 if user is found and -1 if not
20. checkProductID	Use to check the product id from the list of items	1. (ItemInfo) *aItemData = array list of items 2. (int) nSIndex = index of array list of items 3. (int) checkID = product id	(int) nCheck = 1 if user is found and -1 if not
21. checkCSellerID	Use to check the seller id from the cart	1. (CartInfo) *aCartData = array list of items from the cart 2. (int) nCartIndex = index of array list of items from the cart 3. (int) checkID = seller id from the cart	(int) nCheck = 1 if seller is found and 0 if not
22. checkCProductID	Use to check the product id from the cart	1. (CartInfo) *aCartData = array list of items from the cart 2. (int) nCartIndex = index of array list of items from the cart 3. (int) checkID = product id from the cart	(int) nCheck = 1 if seller is found and 0 if not
23. checkContact	Use to check the array of the contact number if there exists a letter	1. (char) *Contact = is the array used the contact number	(int) nCheck = 1 if user is found and -1 if not
USER ACTION			
24. Register	Use to register a new user and input the following information	1. (UserInfo) *aUserData = array list of users 2. (int) *nCountIndex = index of array list of users	1. (UserInfo) User = returns the entered information to the array of users 2. (int) *nCountIndex = will increment after the user confirmed his information
25. displayUser	Use to display the entered information of the user from the Register function	1. (UserInfo) aUserData = information of the user entered information using the structure of UserInfo	No return value
26. sortID	Use to sort the user ID in increasing order	1. (UserInfo) *aUserData = array list of users 2. (Int) nCountIndex = index of array list of users	(UserInfo) *aUserData = returns the sorted user id in increasing order
27. Log_In	Function for user log in	1. (UserInfo) *aUserData = array list of users 2. (int) nCountIndex = index of array list of users 3. (Int) *uID = is the id of the user login in	(int) nCheck = returns 1 if user is found and gets the *uID(user id) and -1 if not

28. addItem	Use to add a new item and input the following information	<ol style="list-style-type: none"> 1. (ItemInfo) *aItemData = array list of items 2. (int) *nSIndex = index of array list of items 3. (Int) sID 	<ol style="list-style-type: none"> 1. (ItemInfo) *aItemData = returns the entered information to the array of items 2. (int) *nSIndex = will increment after the user confirmed the item information
29. displayProduct	Use to display the information of the product entered in the add item	<ol style="list-style-type: none"> 1. (ItemInfo) ItemData = information of the user entered information for the items using the structure ItemInfo 	No return value
30. sellBag20	Use to get the sellers' items from the list of items	<ol style="list-style-type: none"> 1. (ItemInfo) *aItemData = array list of items 2. (int) nSIndex = index of array list of items 3. (ItemInfo) *aItem20 = array of the users' bag to be sold / list of items that the user will sell 4. (int) ID = user id of the current user 	<ol style="list-style-type: none"> 1. (ItemInfo) *aItem20 = returns all the the items of the current user 2. (int) nCount = returns the index of the users' items to be sold.
31. showProducts	Use to show the all products or show one product	<ol style="list-style-type: none"> 1. (ItemInfo) *aItemData = array list of items 2. (int) nSIndex = index of array list of items 3. (int) nCheck = checks which action to do if -1 will show all products and 1 will show one product. 4. (int) pID = product id to be shown in show one product 	No return value
32. sortProducts	Use to sort the product ID in increasing order	<ol style="list-style-type: none"> 1. (ItemInfo) *aItemData = array list of items 2. (Int) nSIndex = index of array list of items 	(ItemInfo) *aItemData = returns the sorted items in increasing order
33. Replenish_Reduce	use to add and reduce the quantity of a specific item that the user wishes to	<ol style="list-style-type: none"> 1. (ItemInfo) *aItem20 = array of the users' bag to be sold 2. (int) nIndex20 = index of array list users' bag to be sold 3. (ItemInfo) *aItemData = array list of items 4. (int) nSIndex = index of array list of items 5. (int) pID = entered product id 6. (int) nCheck = either 1 or -1; If 1 = replenish, -1 = deduct 	Returns the replenished / reduce quantity of the item to the structure of *aItemData & *aItem20
34. changePrice	Use to change the price of a specific item that the user wishes to change	<ol style="list-style-type: none"> 1. (ItemInfo) *aItem20 = array of the users' bag to be sold 2. (int) nIndex20 3. (ItemInfo) *aItemData = array list of items 4. (int) nSIndex = index of array list of items 5. (int) pID = entered product id 	Returns the new entered price of the item to the structure of *aItemData & *aItem20
35. changeName	Use to change the name of a specific item that the user wishes to change	<ol style="list-style-type: none"> 1. (ItemInfo) *aItem20 = array of the users' bag to be sold 2. (Int) nIndex20 3. (ItemInfo) *aItemData = array list of items 4. (int) nSIndex = index of array list of items 5. (int) pID = entered product id 	Returns the newly entered name of the item to the structure of *aItemData & *aItem20
36. changeCategory	Use to change the category of a specific item that the user wishes to change	<ol style="list-style-type: none"> 1. (ItemInfo) *aItem20 = array of the users' bag to be sold 2. (int) nIndex20 3. (ItemInfo) *aItemData = array list of items 4. (int) nSIndex = index of array list of items 5. (int) pID = entered product id 	Returns the newly entered category of the item to the structure of *aItemData & *aItem20

37. changeDescription	Use to change the description of a specific item that the user wishes to change	1. (ItemInfo) *aItem20 = array of the users' bag to be sold 2. (int) nIndex20 = index of array list of items from the users' bag 3. (ItemInfo) *aItemData = array list of items 4. (int) nIndex = index of array list of items 5. (int) pID = entered product id	Returns the newly entered description of the item to the structure of *aItemData & *aItem20
38. showLowProducts	Use to show the products whose quantity falls below 5	1. (ItemInfo) *aItemData = array list of items 2. (int) nIndex = index of array list of items	No return value
39. addDiscount	Use to add discount to all or selected items	1. ItemInfo *aItemData = array list of items 2. int nIndex = index of array list of items 3. int ID = ID of the user	No return value
40. removeDiscount	Use to remove discount to all or selected items	1. ItemInfo *aItemData = array list of items 2. int nIndex = index of array list of items 3. int ID = ID of the user	No return value
41. openDiscount	Use to open the discount of the item in binary file	1. int pID = product id 2. int sID = seller id	(float) fDiscount = returns the discount of the item
42. enterPID	Use to enter the product id	1. (ItemInfo) *aItemData = array list of items 2. (int) nIndex = index of array list of items	(int) ID = if the user is found and -1 if not
43. allProducts	Use to show all the products of the users	1. (ItemInfo) *aItemData = array list of items 2. (int) nIndex = index of array list of items 3. (int) uID = user id of the current user 4. (UserInfo) *aUserData = array list of users 5. (int) nCountIndex = index of array list of users	No return value
44. viewDiscount	Use to display the products with discounts	1. (ItemInfo) *aItemData = array list of items 2. (int) nIndex = index of array list of items 3. (int) uID = user id of the current user 4. (UserInfo) *aUserData = array list of users 5. (int) nCountIndex = index of array list of users	No return value
45. specificSeller	Use to show the products of the specific seller	1. (ItemInfo) *aItemData = array list of items 2. (int) nIndex = index of array list of items 3. (int) uID = user id of the current user	No return value
46. searchCategory	Use to search the category of the items	1. (ItemInfo) *aItemData = array list of items 2. (int) nIndex = index of array list of items 3. (int) uID = user id of the current user	No return value
47. searchName	Use to search the name of the products	1. (ItemInfo) *aItemData = array list of items 2. int nIndex = index of array list of items 3. (int) uID = user id of the current user	No return value
48. displayCart	Use to display the cart of the user	1. (CartInfo) *aCartData = array list of items from the users' cart 2. (int) nCartIndex = index of the array of items from the users' cart	No return value

49. addCart	Use to add an item and enter the information that the user wanted to add from the existing items and quantity	1. (CartInfo) *aCartData = array list of items from the users' cart 2. (int) *nCartIndex = index of the array list of items from the users' cart 3. (ItemInfo) *aItemData = array list of items 4. (int) nSIndex = index of array list of items 5. (int) bID = current users' id	1. (CartInfo) Cart = returns the entered information to the array of users' cart 2. (int) *nCartIndex = will increment the index if the user confirmed the item information he wanted to add to the cart
50. compareItem	Use to compare the item if there were changes made after the user added the item to the cart	1. (CartInfo) *aCartData = array list of items from the users' cart 2. (int) nCartIndex = index of the array list of items from the users' cart 3. (ItemInfo) *aItemData = array list of items 4. (int) nSIndex = index of array list of items	No return value
51. roundUp	Use to round up the (float) price	1. float fNum = price	1. (float) fRound = returns the round-up of the price
52. removeSeller	Use to remove the seller from the cart	1. (CartInfo) *aCartData = array list of items from the users' cart 2. (int) *nCartIndex = index of the array list of items from the users' cart	1. (CartInfo) *aCartData = updates the list of items of the user's cart 2. (int) *nCartIndex = will decrement after removing the seller
53. removeItem	Use to remove an item from the cart	1. (CartInfo) *aCartData = array list of items from the users' cart 2. (int) *nCartIndex = index of the array list of items from the users' cart 3. (int) pID = product id; if -1 the user will enter the product id else it will skip the enter product id	1. (CartInfo) *aCartData = array list of items from the users' cart 2. (int) *nCartIndex = will decrement after removing the item
54. editQty	Use to edit the quantity of the item from the cart	1. (CartInfo) *aCartData = array list of items from the users' cart 2. (Int) *nCartIndex = index of the array list of items from the users' cart 3. (ItemInfo) *aItemData = array list of items 4. (int) nSIndex = index of array list of items	1. (CartInfo) *aCartData = updates the list of items of the users' cart if the user wishes to remove the product which has a quantity of 0 2. (int) *nCartIndex = will decrement after the user confirmed to remove the item with 0 quantity
TRANSACTION			
55. getDate	Use to ask the user to enter the month, date and year	1. (int) *nMonth = pointer month 2. (int) *nDay = pointer month 3. (int) *nYear = pointer year	Returns the entered month, day, year
56. confirmDate	Use to ask the user whether to confirm, re-enter or cancel the entered date	1. (TransacInfo) TransacData = a single structure of the transaction information	(TransacInfo) TransacData = returns the date information needed for the transaction
57. sortData	Use to sort the date of the transactions in increasing order	1. (TransacInfo) *aTransacData = array list of transactions 2. (int) nTransacIndex = index of array list of transactions	(TransacInfo) TransacData = returns the sorted array by date in increasing order
58. transacSeller	Use to transac from all items and specific seller	1. (CartInfo) *aCartTemp = array list of items from the users' cart 2. (int) nCartTemp = index of array list of items from the users' cart 3. (ItemInfo) *aItemData = array list of items	(TransacInfo) tempTransac = returns the transaction information

		4. (int) nSIndex = index of array list of items 5. (TransacInfo) Transac = a single structure of the transaction information 6. (Int) sID = seller id; if -1 the user will enter the seller id else it will skip the enter seller id	
59. transacItem	Use to transac an item	1. (CartItem) *aCartData = array list of items from the users' cart 2. (int) nCartItemIndex = index of the array list of items from the users' cart 3. (ItemInfo) *aItemData = array list of items 4. (int) nSIndex = index of array list of items 5. (TransacInfo) TransacData = a single structure of the transaction information (int) nID = product id; if -1 the user will enter the seller id else it will skip the enter product id	(TransacInfo) tempTransac = returns the transaction information
60. completeInfo	Use to copy the need information which is the seller and buyer information to the transaction information	1. (UserInfo) *UserData = array list of users 2. (int) nCountIndex = index of array list of users 3. (TransacInfo) Transac = a single structure of the transaction information 4. (int) nID = user id of the current user	(TransacInfo) Transac = returns the needed information for the transaction
61. displayReceipt	Use to display the receipt	1. (TransacInfo) Transac = a single structure of the transaction information	No return value
ADMIN			
62. adminUsers	Use to display all the users	1. (UserInfo) *aUserData = array list of users 2. (Int) nCountIndex = index of array list of users	No return value
63. adminSellers	Use to display all the sellers	1. (ItemInfo) *aItemData = array list of items 2. (int) nSIndex = index of array list of items 3. (UserInfo) *aUserData = array list of users 4. (int) nCountIndex = index of array list of users	No return value
64. adminTotalSales	Use to get the total amount of the transaction from the entered date	No input parameter	No return value
65. TransacSID	Use to get the index of the seller	1. (UserInfo) *aUserData = array list of users 2. (TransacInfo) *TransacData = array of transacted items 3. (int) nTransacIndex = index of the array of transacted items	(int) nIndex = returns the index of the seller
66. adminSellerSales	Use to display the total sales for each seller	1. (int) nID = id of the user 2. (string20) UserName = name of the user	No return value
67. adminShopaholics	Use to display the total amount for each buyer in table format and total amount bought in the duration	1. (UserInfo) *aUserData = array list of users 2. (int) nCountIndex = index of array list of users	No return value
68. userReceipt	Use to display the receipt or the transacted items of the user	1. (int) bID = the id of the user	No return value