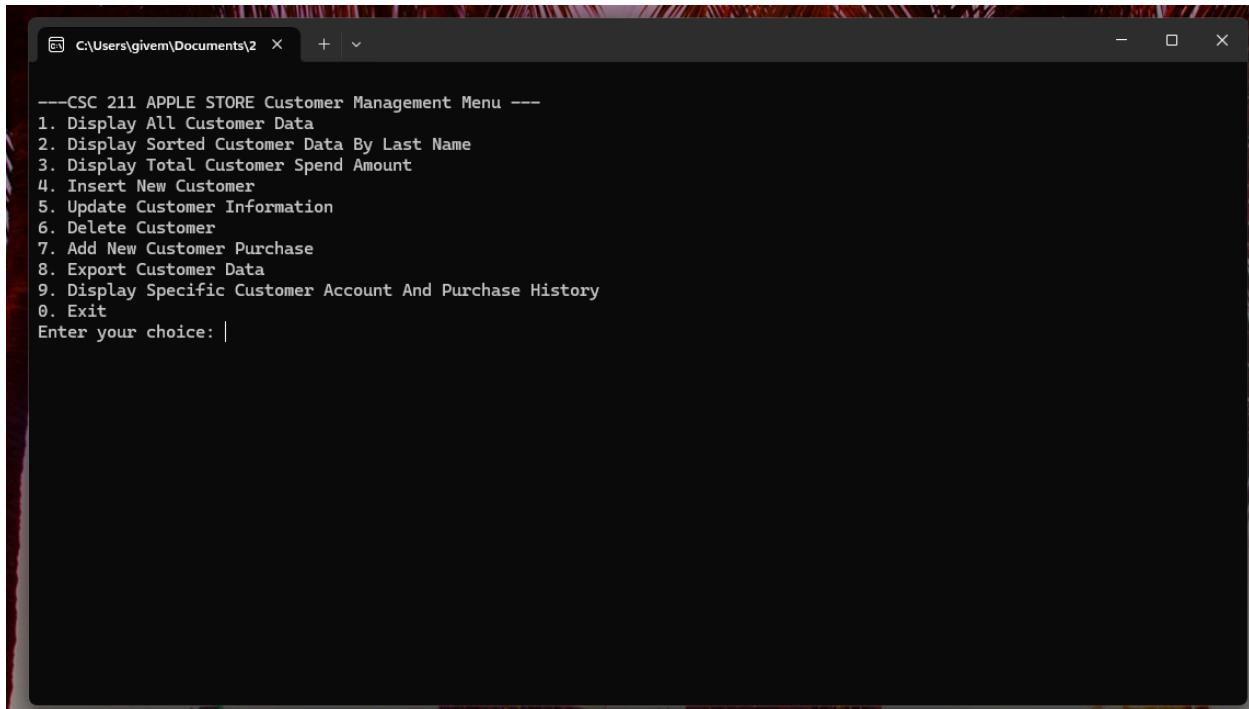


CSC-211 Final Project Mockup

Florin Dorleans

- Menu driven program that loops after each selection until user chooses to exit



ManageAllCustomers.cpp line 711

```
• int ManageAllCustomers::customerManageMenuApp()
• {
•     bool customerDataLoaded =
•         loadCustomerDataFromFile("customerAccountData.csv");
•     bool purchaseDataLoaded =
•         loadCustomerPurchaseDataFromFile("customerPurchases.csv");
•
•     if (!customerDataLoaded && !purchaseDataLoaded) //If
•         default files do not load correctly give user option to
•         enter file paths
•     {
```

```
•         std::cout << "Could Not load default customer or  
• purchase data files." << std::endl;  
•             std::cout << "Enter customer data file path? (leave  
blank to close program): ";  
•                 std::string customerDataFilePath;  
•                 std::getline(std::cin, customerDataFilePath);  
•                     if (customerDataFilePath.empty()){  
•                         return 0; //Exit program if no file path  
provided  
•                         }  
•                         std::cout << "Enter purchase data file path? (leave  
blank to close program): ";  
•                             std::string purchaseDataFilePath;  
•                             std::getline(std::cin, purchaseDataFilePath);  
•                                 if (purchaseDataFilePath.empty()){  
•                                     return 0; //Exit program if no file path  
provided  
•                                     }  
•                                     customerDataLoaded =  
loadCustomerDataFromFile(customerDataFilePath);  
•                                         purchaseDataLoaded =  
loadCustomerPurchaseDataFromFile(purchaseDataFilePath);  
•                                         }  
•  
•     int choice;  
•     while(true)  
•     {  
•         std::cout << "\n---" << getStoreName() << " Customer  
Management Menu ---\n"  
•                         << "1. Display All Customer Data\n"  
•                         << "2. Display Sorted Customer Data By  
Last Name\n"  
•                         << "3. Display Total Customer Spend  
Amount\n"  
•                         << "4. Insert New Customer\n"  
•                         << "5. Update Customer Information\n"
```

```
•             << "6. Delete Customer\n"
•             << "7. Add New Customer Purchase\n"
•             << "8. Export Customer Data\n"
•             << "9. Display Specific Customer Account
And Purchase History\n"
•                 << "0. Exit\n"
•                 << "Enter your choice: ";
choice = readMenuChoice();

•
•         switch(choice)
{
    case 1:
        displayAllCustomerData();
        break;
    case 2:
        displaySortedCustomerData();
        break;
    case 3:
        displayTotalCustomerSpendAmt();
        break;
    case 4:
        insertNewCustomer();
        break;
    case 5:
        updateCustomerInformation();
        break;
    case 6:
        deleteCustomer();
        break;
    case 7:
        addNewCustomerPurchase();
        break;
    case 8:
        exportCustomerData();
        break;
    case 9:
```

```

•     displaySpecificCustomerAccountAndPurchaseHistory();
•         break;
•     case 0:
•         std::cout << "Exiting program.\n";
•         return 0;
•     default:
•         std::cout << "Invalid choice. Please try
again.\n";
•     }
• }
• }
```

◦

- **Load all Customer account Info from a file into a vector of the class**

- ManageAllCustomers.cpp line 81&130: Customer & Purchases

```

• bool ManageAllCustomers::loadCustomerDataFromFile(const
std::string& filename) //Load Customer Data from CSV file
into Database
• {
•     //Checking if file opens correctly
•     std::ifstream inputFile;
•     inputFile.open(filename);
•     if (!inputFile.is_open()){
•         std::cerr << "Error opening file: " << filename <<
std::endl;
•         return false;
•     }
•     std::string line = ""; //Temp string to hold each line
of file
•
•     getline(inputFile, line); //Skip header line
•     line = ""; //Clear line variable
• }
```

```
•     while(getline(inputFile, line)) //While there are lines
      to read in file
•     {
•         std::string tempString; //Temp string to hold each
      data value
•
•         std::string firstName, lastName,
      streetAddress, city, state, zipCode, phoneNumber;
      int accountNumber; //Temp variables to hold data
      from file
•
•         std::stringstream inputString(line); //Create string
      stream from line
      //Read in each data value from line
      //Convert account number from string to int
      getline(inputString, tempString, ',');
      accountNumber = std::stoi(tempString);
      tempString = ""; //Clear temp string
•
•         getline(inputString,firstName, ',');
      getline(inputString,lastName, ',');
      getline(inputString,streetAddress, ',');
      getline(inputString,city, ',');
      getline(inputString,state, ',');
      getline(inputString,zipCode, ',');
      getline(inputString,phoneNumber, ',');
•
•
•         //Create new Allcustomers object with data from file
      AllCustomers newCustomer(firstName, lastName,
      accountNumber,
      streetAddress, city, state, zipCode, phoneNumber);
•
•         //Add new customer to database
      defaultCustomerDataBase.push_back(newCustomer);
```

```
•     }
•     return true; //File loaded successfully data loaded into
  database
• }
```

- **Use a vector within the class to store purchase information**

- AllCustomers.h Line 23 Class Definition

```
• class AllCustomers {
•     std::string m(firstName;
•     std::string m(lastName;
•     std::string m(streetAddress;
•     std::string m(city;
•     std::string m(state;
•     std::string m(zipCode;
•     std::string m(phoneNumber;
•     std::vector<AllPurchases> customerPurchaseHistory;
```

- **All functions should be a public member function of your classes**

- Besides the use of some helper/inline functions this is completed

- **Menu Options:**

- Print out all list of all customers

The screenshot shows a terminal window titled 'C:\Users\givem\Documents\2'. The window displays a menu with options 8, 9, and 0. Option 1 is selected, prompting for a choice. The data is presented in a table format with columns: Account #, First Name, Last Name, Street Address, City, State, Zip Code, and Phone Number. The data is as follows:

Account #	First Name	Last Name	Street Address	City	State	Zip Code	Phone Number
1001	Tim	Cook	1 Infinite Loop	Cupertino	CA	95014	408-555-1010
1002	Eddy	Cue	25 Apple Park Way	Cupertino	CA	95014	408-555-2020
1003	Craig	Federighi	45 De Anza Blvd	Cupertino	CA	95015	408-555-3030
1004	Jony	Ive	99 Design Dr	San Jose	CA	95110	408-555-4040
1005	Angela	Ahrendts	77 Retail St	Palo Alto	CA	94301	650-555-1111
1006	Philip	Schiller	188 Market St	San Francisco	CA	94103	415-555-2222
1007	Jeff	Williams	133 Supply Ln	San Jose	CA	95112	408-555-5050
1008	Luca	Maestri	200 Finance Blvd	Sunnyvale	CA	94085	408-555-6060
1009	Katherine	Adams	12 Legal Way	Cupertino	CA	95014	408-555-7070
1010	Dan	Riccio	500 Hardware Dr	Mountain View	CA	94040	650-555-3333

---CSC 211 APPLE STORE Customer Management Menu ---

1. Display All Customer Data
2. Display Sorted Customer Data By Last Name
3. Display Total Customer Spend Amount

ManageAllCustomers.cpp line 197

```
void ManageAllCustomers::displayAllCustomerData(){ //Display
Each Customer's Data in Database

    printCustomerHeader(); //Print Header
    for (int i = 0; i < defaultCustomerDataBase.size(); ++i){
        defaultCustomerDataBase[i].printCustomerData();
    }
}

void ManageAllCustomers::displayAllCustomerData(const
std::vector<AllCustomers>& customers){

    printCustomerHeader();
    for (int i = 0; i < customers.size(); ++i){
```

```

        customers[i].printCustomerData();
    }
}

```

Option to sort and print Customer list in descending or ascending order (by name)

- ManageAllCustomers.cpp line 214

1:Sort By Last Name Ascending 2: Sort By Last Name Descending Enter Option: 2							
Account #	First Name	Last Name	Street Address	City	State	Zip Code	Phone Number
1007	Jeff	Williams	133 Supply Ln	San Jose	CA	95112	408-555-5050
1006	Philip	Schiller	188 Market St	San Francisco	CA	94103	415-555-2222
1010	Dan	Riccio	500 Hardware Dr	Mountain View	CA	94040	650-555-3333
1008	Luca	Maestri	200 Finance Blvd	Sunnyvale	CA	94085	408-555-6060
1004	Jony	Ive	99 Design Dr	San Jose	CA	95110	408-555-4040
1003	Craig	Federighi	45 De Anza Blvd	Cupertino	CA	95015	408-555-3030
1002	Eddy	Cue	25 Apple Park Way	Cupertino	CA	95014	408-555-2020
1001	Tim	Cook	1 Infinite Loop	Cupertino	CA	95014	408-555-1010
1005	Angela	Ahrendts	77 Retail St	Palo Alto	CA	94301	650-555-1111
1009	Katherine	Adams	12 Legal Way	Cupertino	CA	95014	408-555-7070

Print out a specific customer's account information with all purchases

Line 686 ManageAllCustomer.cpp

Select Customer from list by index number: 0							
Customer Information:							
Account #	First Name	Last Name	Street Address	City	State	Zip Code	Phone Number
1002	Eddy	Cue	25 Apple Park Way	Cupertino	CA	95014	408-555-2020
Purchase History:							
Account #	Item	Date	Cost				
1001	iPhone 16 Pro	2025-02-10	\$1299.00				
1001	AirPods Pro 2	2025-02-12	\$249.00				
1001	Apple Watch Series 9	2025-02-12	\$399.00				
1002	Iphone 29000	2003-11-19	\$600.00				

Print out the total spend for all a customer's purchases

Line 250 ManageAllCustomer.cpp

Select Customer from list by index number: 0			
Account #	Item	Date	Cost
1001	iPhone 16 Pro	2025-02-10	\$1299.00
1001	AirPods Pro 2	2025-02-12	\$249.00
1001	Apple Watch Series 9	2025-02-12	\$399.00

Total Spend Amount for Customer Account #1001 is: \$1947.00

Add a new customer

Line 277 ManageAllCustomer.cpp

```
0. EXIT
Enter your choice: 4
Customer insertion Tool: Type -1 at any prompt to cancel insertion and return to main menu.
Enter First Name:
Flo
Enter Last Name:
test
Enter Street Address:
211 CSI AVE
Enter City:
Staten Island
Enter State (2-letter abbreviation: E.g.NY NJ):
new york
Invalid state abbreviation. Please enter a valid state name:
ny
Enter Zip Code:
103011
Invalid zip code. Please enter a valid 5-digit zip code:
10301
Enter Phone Number (10 digits, no spaces or dashes):
3339991111
Account Number will be Generated Automatically.
Customer added successfully.
Add another customer? (1 for Yes, -1 to Cancel): -1
Returning to main menu.

---CSC 211 APPLE STORE Customer Management Menu ---
1. Display All Customer Data
2. Display Sorted Customer Data By Last Name
3. Display Total Customer Spend Amount
4. Insert New Customer
5. Update Customer Information
6. Delete Customer
7. Add New Customer Purchase
8. Export Customer Data
9. Display Specific Customer Account And Purchase History
0. Exit
Enter your choice: |
```

Add multiple customers (Use Recursive Function)

Line 416(recursive Call): ManageAllCustomers.cpp

```
0. Exit
Enter your choice: 4
Customer insertion Tool: Type -1 at any prompt to cancel insertion and return to main menu.
Enter First Name:
Flo
Enter Last Name:
test
Enter Street Address:
211 CSI AVE
Enter City:
Staten Island
Enter State (2-letter abbreviation: E.g.NY NJ):
new york
Invalid state abbreviation. Please enter a valid state name:
ny
Enter Zip Code:
103011
Invalid zip code. Please enter a valid 5-digit zip code:
10301
Enter Phone Number (10 digits, no spaces or dashes):
3339991111
Account Number will be Generated Automatically.
Customer added successfully.
Add another customer? (1 for Yes, -1 to Cancel): -1
Returning to main menu.

---CSC 211 APPLE STORE Customer Management Menu ---
1. Display All Customer Data
2. Display Sorted Customer Data By Last Name
3. Display Total Customer Spend Amount
4. Insert New Customer
5. Update Customer Information
6. Delete Customer
7. Add New Customer Purchase
8. Export Customer Data
9. Display Specific Customer Account And Purchase History
0. Exit
Enter your choice: |
```

Update a customer's information

Line 422 ManageAllCustomers.cpp

```

1011 |Flo      |test      |211 CSI AVE      |Staten Island|NY    |10301    |333-999-1111 |10
Select Customer from list by index number: 0
Customer Found.
Account # |First Name     |Last Name     |Street Address     |City       |State    |Zip Code   |Phone Number
-----|-----|-----|-----|-----|-----|-----|-----
1001    |Tim          |Cook         |1 Infinite Loop   |Cupertino |CA      |95014    |408-555-1010
Select Field to Update:
1: First Name
2: Last Name
3: Street Address
4: City
5: State
6: Zip Code
7: Phone Number
8: Exit Update Menu
Enter Option: 1
Enter New First Name: Timothy
Updated Customer Information:
Account # |First Name     |Last Name     |Street Address     |City       |State    |Zip Code   |Phone Number
-----|-----|-----|-----|-----|-----|-----|-----
1001    |Timothy       |Cook         |1 Infinite Loop   |Cupertino |CA      |95014    |408-555-1010
Select Field to Update:
1: First Name
2: Last Name
3: Street Address
4: City
5: State
6: Zip Code
7: Phone Number
8: Exit Update Menu
Enter Option: |

```

Delete a customer's information

Line 541: ManageAllCustomers.cpp

```

1011 |Flo      |test      |211 CSI AVE      |Staten Island|NY
Select Customer from list by index number: 0
Are you sure you want to delete this customer? Type 1 for Yes, 0 for No: 1
Customer Found. Deleting Customer Account #: 1001
Customer Deleted Successfully. Returning to Main Menu.

---CSC 211 APPLE STORE Customer Management Menu ---
1. Display All Customer Data
2. Display Sorted Customer Data By Last Name
3. Display Total Customer Spend Amount
4. Insert New Customer
5. Update Customer Information
6. Delete Customer
7. Add New Customer Purchase
8. Export Customer Data
9. Display Specific Customer Account And Purchase History
0. Exit
Enter your choice: |

```

Add a new customer's purchase

Line 570: ManageAllCustomers.cpp

```

1001 | Eddy | Cue | 25 Apple Park Way | Cupertino | CA | 95014 | 408-555-2020
Select Customer from list by index number: 0
Are you adding a purchase for Account Number: 1002 ? (1 for Yes, 0 for No): 1
Enter Item Name: Iphone 29000
Enter Date of Purchase (YYYY MM DD):Year (YYYY):
2003
Month (MM):
11
Day (DD):
19
Enter Cost of Purchase: 600
Purchase added successfully.
Add another purchase? (1 for Yes, 0 for No): 0
Returning to main menu.

```

Customer Information:							
Account #	First Name	Last Name	Street Address	City	State	Zip Code	Phone Number
1002	Eddy	Cue	25 Apple Park Way	Cupertino	CA	95014	408-555-2020

Purchase History:			
Account #	Item	Date	Cost
1001	iPhone 16 Pro	2025-02-10	\$1299.00
1001	AirPods Pro 2	2025-02-12	\$249.00
1001	Apple Watch Series 9	2025-02-12	\$399.00
1002	Iphone 29000	2003-11-19	\$600.00

Add multiple new customer purchases (Use Recursive Function)

Line 637(Recursive Call): ManageAllCustomers.cpp

```
1001      11 00          test          1211 ESI AVE          Staten Island  
Select Customer from list by index number: 0  
Are you adding a purchase for Account Number: 1002 ? (1 for Yes, 0 for No): 1  
Enter Item Name: Iphone 29000  
Enter Date of Purchase (YYYY MM DD):Year (YYYY):  
2003  
Month (MM):  
11  
Day (DD):  
19  
Enter Cost of Purchase: 600  
Purchase added successfully.  
Add another purchase? (1 for Yes, 0 for No): 0  
Returning to main menu.
```

Let the user save over the existing import files or copy to a new file

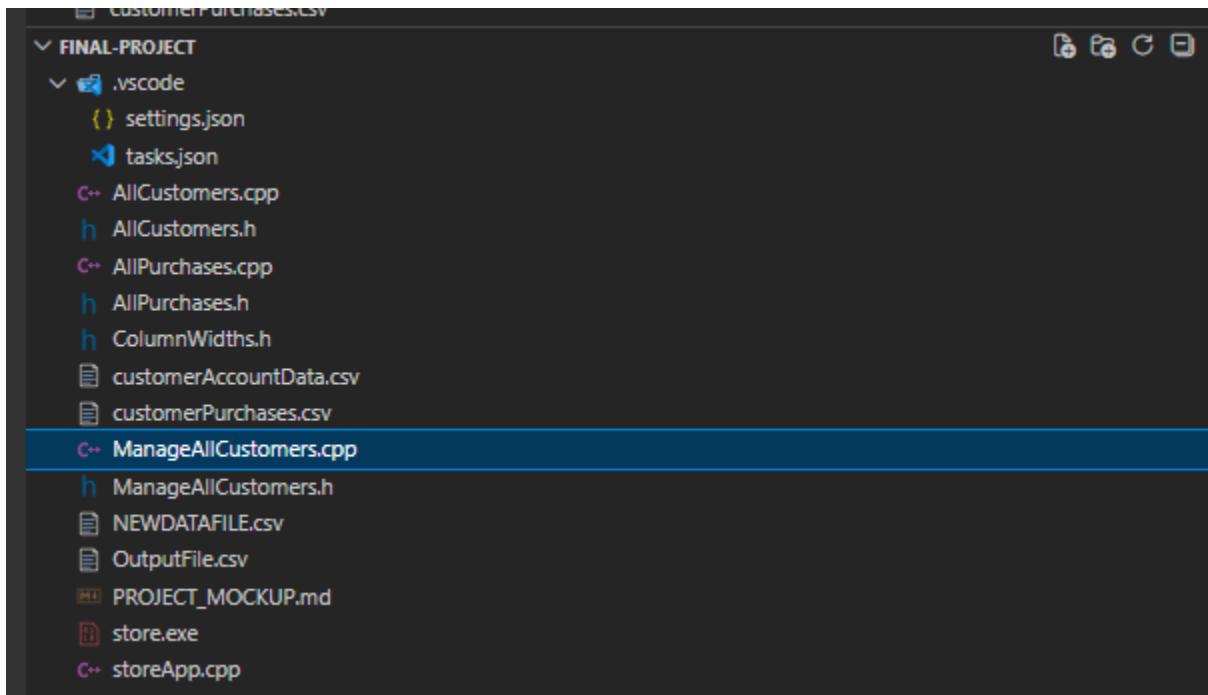
Line 647: ManageAllCustomers.cpp

```
0. Exit  
Enter your choice: 8  
Would you like to (1) Overwrite existing file or (2) Export to new file? Enter 1 or 2: 2  
Enter new filename (with .csv extension): outputFile.csv  
Customer data exported successfully to outputFile.csv
```

All work and procedures to be completed in Member Functions

```
1 #include "ManageAllCustomers.h"
2
3 > void printCustomerHeader(){ ...
40
41 > void printCustomerHeaderWithIndex() ...
80
81 > bool ManageAllCustomers::loadCustomerDataFromFile(const std::string& filename) //Load Customer Data from CSV file into Database...
129
130 > bool ManageAllCustomers::loadCustomerPurchaseDataFromFile(const std::string& filename) ...
193
194
195
196
197 > void ManageAllCustomers::displayAllCustomerData(){ ///Display Each Customer's Data in Database...
204
205 > void ManageAllCustomers::displayAllCustomerData(const std::vector<AllCustomers>& customers){ ...
212
213
214 > void ManageAllCustomers::displaySortedCustomerData() { ...
246
247
248
249
250 > void ManageAllCustomers::displayTotalCustomerSpendAmt() { ...
275
276
277 > void ManageAllCustomers::insertNewCustomer() { //Gives User ability to insert new customer data into database...
421
422 > void ManageAllCustomers::updateCustomerInformation() { //Gives User ability to update existing customer data in database...
539
540
541 > void ManageAllCustomers::deleteCustomer() { //Gives User ability to delete existing customer data in database...
569
570 > void ManageAllCustomers::addNewCustomerPurchase() { ...
646
647 > void ManageAllCustomers::exportCustomerData() { ...
685
686 > void ManageAllCustomers::displaySpecificCustomerAccountAndPurchaseHistory() ...
709
710 |
711 > int ManageAllCustomers::customerManageMenuApp() //Main Menu for Customer Management System...
787
788
789 > int ManageAllCustomers::userCustomerSelectIndex() ...
```

Code should be split across minimum of 5 Files (Main.CPP, Class1.H, Class1.CPP, Class2.H, Class2.CPP)



- **Comment all code and functions**
 - <https://github.com/InvisibleDoctor/Final-Project>
- **Use advanced course material where necessary (Constructors, Overloaded Assignment Operators, Destructors, Pointers, Recursion, Sorting and Searching, Function or Class Templates, Range Based Loops)**
 - <https://github.com/InvisibleDoctor/Final-Project>