

Program Write-Up

Project Overview

The project involves the creation of a Store Management System that efficiently manages products, transactions, and credit card information for customers. The system includes functionalities to add and retrieve products and transactions, process credit card purchases and payments, and display detailed information about the store's inventory and transaction history.

See the UML diagram and the program write-up below for more information on what components the program needs to have.

Project Title: Store Management System

Objective: Develop a system to manage products, transactions, and customer information for an online store.

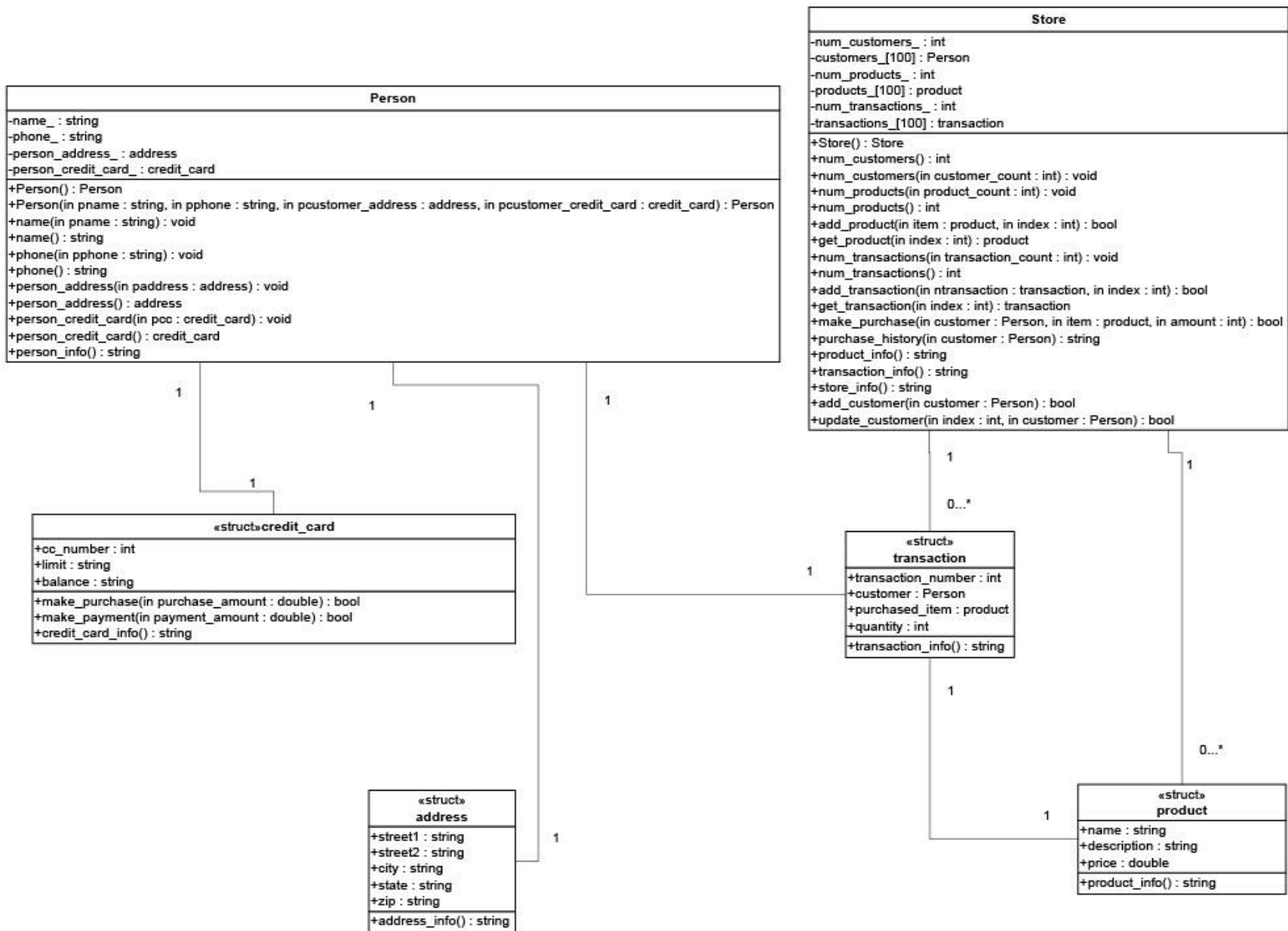
Requirements:

1. Implement all of the classes and structs outlined below.
 1. Use appropriate C++ classes and struct to model real-world entities.
 2. Implement error handling for edge cases (e.g., attempting to make a purchase over the credit limit).
 3. Successful purchases should be reflected in updated credit card balance.
2. Write a main.cpp file that will
 1. scan information in from a text file (provided in github), create appropriate objects, and execute the scenarios described in the test cases provided below.
 2. Show appropriate error messages when operations fail (e.g., exceeding credit limit).
3. Your output on the Jenkins server must be similar to the sample output I've provided.

Deliverables:

- All your code should be on Github.
- See the sample output below. You must call your executable from Jenkins and the output should be similar to the sample output below.

UML Diagram



Class and Struct Definitions

1. Credit Card Management:

- **Struct:** `credit_card`
- **Attributes:**
 - `cc_number`: An integer representing the credit card number.
 - `limit`: A double representing the credit limit of the credit card.
 - `current_balance`: A double representing the current balance on the credit card.
- **Methods:**
 - `make_purchase(double purchase_amount)`: Attempts to make a purchase against the credit card limit. Returns true if successful and false otherwise.
 - `make_payment(double payment_amount)`: Attempts to make a payment towards the credit card balance. Returns true if successful and false otherwise.
 - `credit_card_info()`: Returns a string containing the credit card details.

2. Product Management:

- **Struct:** `product`
- **Attributes:**
 - `name`: A string representing the product name.
 - `description`: A string representing the product description.
 - `price`: A double representing the product price.
- **Methods:**
 - `product_info()`: This method returns a string containing information about the product, including its name, description, and price.

3. Transaction Management:

- **Struct:** `transaction`
- **Attributes:**
 - `transaction_number`: An integer representing the transaction number.
 - `customer`: A Person object representing the customer involved in the transaction.
 - `purchased_item`: A product object representing the item purchased in the transaction.
 - `quantity`: An integer representing the quantity of the purchased item.

- **Methods:**

- `transaction_info()`: Returns a string containing information about the transaction.

4. Store Management:

- **Class:** `Store`

- **Attributes:**

- `num_customers_`: An integer representing the number of products in the store.
- `customers_`: An array of product objects representing the products in the store.
- `num_products_`: An integer representing the number of products in the store.
- `products_`: An array of product objects representing the products in the store.
- `num_transactions_`: An integer representing the number of transactions in the store.
- `transactions_`: An array of transaction objects representing the transactions in the store.

- **Methods:**

- `Store()`: Default constructor that initializes all values to empty.
- `num_products(int product_count)`: Setter method for `num_products_`.
- `int num_products()`: Getter method for `num_products_`.
- `num_customers(int customer_count)`: Setter method for `num_customers_`.
- `int num_customers()`: Getter method for `num_customers_`.
- `num_transactions(int transaction_count)`: Setter method for `num_transactions_`.
- `int num_transactions()`: Getter method for `num_transactions_`.
- `bool add_product(product item)`: Adds a single product to the `products_` array. Returns false if the array is full and true otherwise.
- `bool add_customer(Person customer)`: Adds a single customer to the `customers_` array. Returns false if the array is full and true otherwise.
- `bool update_customer(int index, Person customer)`: Replaces a single customer in the `customers_` array with another. Returns false if the array is full or customer is not present in the array and true otherwise.
- `product get_product(int index)`: Returns the product at the specified index in the `products_` array.
- `bool add_transaction(transaction ntransaction)`: Adds a single transaction to the `transactions_` array. Returns false if the array is full and true otherwise.

- `transaction get_transaction(int index)`: Returns the transaction at the specified index in the `transactions_` array.
- `bool make_purchase(Person customer, product item, int amount)`: Attempts to make a purchase using a Person's credit card. Returns true if successful, false otherwise.
- `string product_info()`: Returns a string listing all products associated with the store.
- `string transaction_info()`: Returns a string listing all transactions associated with the store.
- `string store_info()`: Returns a string listing all information about the store, including transactions and products.
- `string purchase_history(Person customer)`: Searches through the `transactions_` array and returns a string representing all transactions that belong to a specific customer.

5. Person Management:

- **Class:** `Person`
- **Attributes:**
 - `name_`: A string representing the person's name.
 - `phone_`: A string representing the person's phone number.
 - `person_address_`: An address struct representing the person's address.
 - `person_credit_card_`: A `credit_card` struct representing the person's credit card information.
- **Methods:**
 - `Person(string name, string phone, address person_address, credit_card person_credit_card)`: Constructor for a Person object with parameters for name, phone, address, and credit card.
 - `Person()`: Default constructor that initializes all data members to basic defaults.
 - `void name(string)`: Setter method for `name_`.
 - `string name()`: Getter method for `name_`.
 - `void phone(string)`: Setter method for `phone_`.
 - `string phone()`: Getter method for `phone_`.
 - `void person_address(address)`: Setter method for `person_address_`.
 - `address person_address()`: Getter method for `person_address_`.

- `void person_credit_card(credit_card):` Setter method for `person_credit_card_`.
- `credit_card person_credit_card():` Getter method for `person_credit_card_`.
- `string person_info():` Returns a string representation of a person, including their name, phone number, address, and credit card information.

6. Address Management:

- **Struct:** `address`
- **Attributes:**
 - `address_line1:` A string representing the first line of the address.
 - `address_line2:` A string representing the second line of the address.
 - `city:` A string representing the city.
 - `state:` A string representing the state.
 - `zip:` A string representing the zip code.
- **Methods:**
 - `address_info():` This method returns a string containing the address information, including the address lines, city, state, and zip code.

Test Cases / Rubric

For this assignment, each store is limited to an inventory of 100 products MAX and 100 separate transactions MAX and 100 customers MAX.

- 1) **(10 pts)** Create arrays of appropriate sizes and use them to save the information from the text files provided. You may add more content to the text file, if you wish. Just please keep the formatting the same.
- 2) **(15 pts)** Create a store object and add all products and customers to the store (in the text file).
- 3) **(20 pts)** After initially populating your People and Product arrays and assigning your products to the store, print that information (as shown in the sample output below). It is not enough that “something” is printed. The information printed MUST match the information in the text files.
- 4) **(15 pts)** Using `rand`, perform a random amount of transactions where $10 < \text{transactions} < 100$. So, at least 10 transactions, but no more than 100. The transaction should consist of one customer purchasing 1 or more items of a single product, as long as they have sufficient funds on their credit card.
- 5) **(20 pts)** While performing transactions, you should be printing messages for successful and unsuccessful transactions as well as the transaction detail itself.
- 6) **(20 pts)** After all transactions are done, reprint all customer information and the store information.

NOTE: For any printed output (3, 5, 6), it is NOT ENOUGH that “something” gets printed. The information printed **MUST** be correct to get ANY credit for that item.

NOTE: The information **MUST** be printed in Jenkins to get any points at all. If your entire project works correctly, but there is not output in your Jenkins job, your grade for the assignment will be a 0.

Sample Output

Store info:

Customers:

Name: John Wick

Phone: 123-456-7890

Address:

111 Hollywood Lane

Hollywood

CA

99941

Credit Card:

Credit card number: 11111111

Credit Card Limit: \$100000.00

Current Balance: \$0.00

Name: Fresh Prince

Phone: 111-222-3333

Address:

451 Hollywood Blvd

Bel Aire

CA

13456

Credit Card:

Credit card number: 23456789

Credit Card Limit: \$10000.00

Current Balance: \$2500.00

Name: John Smith

Phone: 221-098-2818

Address:

123 Nowhere Rd

Some City

Some State

12345

Credit Card:

Credit card number: 11330094

Credit Card Limit: \$500.00

Current Balance: \$150.00

Name: Jane Doe

Phone: 129-992-5533

Address:

998 Somewhere Rd

Another City

Another State

54321

Credit Card:

Credit card number: 45678901

Credit Card Limit: \$6000.00

Current Balance: \$1200.00

Name: Albert Einstein

Phone: 334-222-1098

Address:

1000 Scientific Lane

EMC²

Wurttemberg

Germany

11145

Credit Card:

Credit card number: 12345678

Credit Card Limit: \$5000.00

Current Balance: \$1500.00

Name: Alex Johnson

Phone: 892-001-2234

Address:

123 Maple Street

Springfield

IL

62701

Credit Card:

Credit card number: 91042834

Credit Card Limit: \$9000.00

Current Balance: \$500.00

Name: Maria Gonzalez

Phone: 128-009-2198

Address:

456 Oak Avenue

Rivertown

TX

75001

Credit Card:

Credit card number: 39104912

Credit Card Limit: \$13000.00

Current Balance: \$10000.00

Name: David Smith

Phone: 899-335-5812

Address:

789 Pine Road

Lakeside

CA

92040

Credit Card:

Credit card number: 39201123

Credit Card Limit: \$14000.00

Current Balance: \$3500.00

Name: Emily Brown

Phone: 992-110-9922

Address:

101 Elm Boulevard

Greenfield

NY

12110

Credit Card:

Credit card number: 12890223

Credit Card Limit: \$12000.00

Current Balance: \$2000.00

Name: Michael Lee

Phone: 664-992-2211

Address:

202 Birch Lane

Meadowville

FL

32003

Credit Card:

Credit card number: 34567890

Credit Card Limit: \$7500.00

Current Balance: \$3000.00

Products:

Name: Laptop

Description: A high-performance laptop with 16GB RAM and 512GB SSD.

Price: \$3999.99

Name: Smartphone

Description: A latest model smartphone with a 6.5-inch, foldable display and 128GB storage.

Price: \$1899.99

Name: Headphones

Description: Noise-cancelling over-ear headphones with Bluetooth connectivity.

Price: \$199.99

Name: Smartwatch

Description: A smartwatch with fitness tracking and heart rate monitoring.

Price: \$249.99

Name: Tablet

Description: A 10-inch tablet with 64GB storage and a high-resolution display.

Price: \$329.99

Name: Camera

Description: A digital camera with 24MP resolution and 4K video recording.

Price: \$949.99

Name: Printer

Description: A wireless all-in-one printer with scanning and copying functions.

Price: \$129.99

Name: Monitor

Description: A 27-inch 4K UHD monitor with HDR support.

Price: \$399.99

Name: Keyboard

Description: A mechanical keyboard with RGB backlighting.

Price: \$159.99

Name: Mouse

Description: A wireless ergonomic mouse with adjustable DPI.

Price: \$49.99

All Transactions:

Successfull transaction!

Transaction Details:

Transaction #: 1

Customer:

Name: Albert Einstein

Phone: 334-222-1098

Address:

1000 Scientific Lane

EMC^2

Wurttemberg

Germany

11145

Credit Card:

Credit card number: 12345678

Credit Card Limit: \$5000.00

Current Balance: \$3149.95

Product:

Name: Tablet

Description: A 10-inch tablet with 64GB storage and a high-resolution display.

Price: \$329.99

Quantity: 5

Successfull transaction!

Transaction Details:

Transaction #: 2

Customer:

Name: Fresh Prince

Phone: 111-222-3333

Address:

451 Hollywood Blvd

Bel Aire

CA

13456

Credit Card:

Credit card number: 23456789

Credit Card Limit: \$10000.00

Current Balance: \$7249.95

Product:

Name: Camera

Description: A digital camera with 24MP resolution and 4K video recording.

Price: \$949.99

Quantity: 5

Unsuccessfull transaction!

Transaction Details:

Person:

Name: Fresh Prince

Phone: 111-222-3333

Address:

451 Hollywood Blvd

Bel Aire

CA

13456

Credit Card:

Credit card number: 23456789

Credit Card Limit: \$10000.00

Current Balance: \$7249.95

Product:

Name: Tablet

Description: A 10-inch tablet with 64GB storage and a high-resolution display.

Price: \$329.99

Purchase Quantity: 9

Successfull transaction!

Transaction Details:

Transaction #: 3

Customer:

Name: Albert Einstein

Phone: 334-222-1098

Address:

1000 Scientific Lane

EMC^2

Wurttemberg

Germany

11145

Credit Card:

Credit card number: 12345678

Credit Card Limit: \$5000.00

Current Balance: \$4589.86

Product:

Name: Keyboard

Description: A mechanical keyboard with RGB backlighting.

Price: \$159.99

Quantity: 9

Store info:

Customers:

Name: John Wick

Phone: 123-456-7890

Address:

111 Hollywood Lane

Hollywood

CA

99941

Credit Card:

Credit card number: 11111111

Credit Card Limit: \$100000.00

Current Balance: \$0.00

Name: Fresh Prince

Phone: 111-222-3333

Address:

451 Hollywood Blvd

Bel Aire

CA

13456

Credit Card:

Credit card number: 23456789

Credit Card Limit: \$10000.00

Current Balance: \$7249.95

Name: John Smith

Phone: 221-098-2818

Address:

123 Nowhere Rd

Some City

Some State

12345

Credit Card:

Credit card number: 11330094

Credit Card Limit: \$500.00

Current Balance: \$150.00

Name: Jane Doe

Phone: 129-992-5533

Address:

998 Somewhere Rd

Another City

Another State

54321

Credit Card:

Credit card number: 45678901

Credit Card Limit: \$6000.00

Current Balance: \$1200.00

Name: Albert Einstein

Phone: 334-222-1098

Address:

1000 Scientific Lane

EMC²

Wurttemberg

Germany

11145

Credit Card:

Credit card number: 12345678

Credit Card Limit: \$5000.00

Current Balance: \$4589.86

Name: Alex Johnson

Phone: 892-001-2234

Address:

123 Maple Street

Springfield

IL

62701

Credit Card:

Credit card number: 91042834

Credit Card Limit: \$9000.00

Current Balance: \$500.00

Name: Maria Gonzalez

Phone: 128-009-2198

Address:

456 Oak Avenue

Rivertown

TX

75001

Credit Card:

Credit card number: 39104912

Credit Card Limit: \$13000.00

Current Balance: \$10000.00

Name: David Smith

Phone: 899-335-5812

Address:

789 Pine Road

Lakeside

CA

92040

Credit Card:

Credit card number: 39201123

Credit Card Limit: \$14000.00

Current Balance: \$3500.00

Name: Emily Brown

Phone: 992-110-9922

Address:

101 Elm Boulevard

Greenfield

NY

12110

Credit Card:

Credit card number: 12890223

Credit Card Limit: \$12000.00

Current Balance: \$2000.00

Name: Michael Lee

Phone: 664-992-2211

Address:

202 Birch Lane

Meadowville

FL

32003

Credit Card:

Credit card number: 34567890

Credit Card Limit: \$7500.00

Current Balance: \$3000.00

Products:

Name: Laptop

Description: A high-performance laptop with 16GB RAM and 512GB SSD.

Price: \$3999.99

Name: Smartphone

Description: A latest model smartphone with a 6.5-inch, foldable display and 128GB storage.

Price: \$1899.99

Name: Headphones

Description: Noise-cancelling over-ear headphones with Bluetooth connectivity.

Price: \$199.99

Name: Smartwatch

Description: A smartwatch with fitness tracking and heart rate monitoring.

Price: \$249.99

Name: Tablet

Description: A 10-inch tablet with 64GB storage and a high-resolution display.

Price: \$329.99

Name: Camera

Description: A digital camera with 24MP resolution and 4K video recording.

Price: \$949.99

Name: Printer

Description: A wireless all-in-one printer with scanning and copying functions.

Price: \$129.99

Name: Monitor

Description: A 27-inch 4K UHD monitor with HDR support.

Price: \$399.99

Name: Keyboard

Description: A mechanical keyboard with RGB backlighting.

Price: \$159.99

Name: Mouse

Description: A wireless ergonomic mouse with adjustable DPI.

Price: \$49.99

All Transactions:

Transaction #: 1

Customer:

Name: Albert Einstein

Phone: 334-222-1098

Address:

1000 Scientific Lane

EMC^2

Wurttemberg

Germany

11145

Credit Card:

Credit card number: 12345678

Credit Card Limit: \$5000.00

Current Balance: \$3149.95

Product:

Name: Tablet

Description: A 10-inch tablet with 64GB storage and a high-resolution display.

Price: \$329.99

Quantity: 5

Transaction #: 2

Customer:

Name: Fresh Prince

Phone: 111-222-3333

Address:

451 Hollywood Blvd

Bel Aire

CA

13456

Credit Card:

Credit card number: 23456789

Credit Card Limit: \$10000.00

Current Balance: \$7249.95

Product:

Name: Camera

Description: A digital camera with 24MP resolution and 4K video recording.

Price: \$949.99

Quantity: 5

Transaction #: 3

Customer:

Name: Albert Einstein

Phone: 334-222-1098

Address:

1000 Scientific Lane

EMC²

Wurttemberg

Germany

11145

Credit Card:

Credit card number: 12345678

Credit Card Limit: \$5000.00

Current Balance: \$4589.86

Product:

Name: Keyboard

Description: A mechanical keyboard with RGB backlighting.

Price: \$159.99

Quantity: 9