

Financial Records

C++/OOP



```
6  #include <string>
7  using namespace std;
8
9  class Transaction {
10 private:
11     int trans_id;
12     double amount;
13     int no;
14     string date;
15     string desc;
16
17 public:
18
19     Transaction() : trans_id(0), amount(0), no(0), date(""), desc("") {}
20
21
22     void addTransaction() {
23         ifstream inFile("transaction.txt");
24         string line;
25         int temp = 0;
26
27
28         while (getline(inFile, line)) {
29             stringstream ss(line);
30             string _id;
31             getline(ss, _id, ',');
32             int num = stoi(_id);
33             if (num > temp) {
34                 temp = num;
35             }
36         }
```

Problem Statement

The program aims to manage transactions through a menu-driven interface. It allows users to add transactions, remove transactions by ID, view transaction history, and calculate the total transaction amount.

Solution Overview

Transaction Class:

1. Encapsulates transaction details and methods to add, remove, view transactions, and calculate total amounts.
2. Main Function: Provides a menu-driven interface using a while loop to continuously prompt the user for options until they choose to exit.

Challenges and Resolutions

File Handling:

Reading from and writing to files (transaction.txt) posed challenges, particularly in handling file operations safely (like removing and renaming files).

Date Handling:

Getting and formatting current date/time (chrono library) required careful handling to ensure accurate timestamping of transactions.

Input Validation:

Ensuring inputs like the number of items (no) are validated to prevent invalid or zero entries.

Instructions

To set up and use the application:

1. Compile: Compile the C++ code using a C++ compiler (e.g., g++).
2. Run: Execute the compiled binary.
3. Menu Options:
 - 1: Add a transaction.
 - 2: Remove a transaction by ID.

- 3: Calculate and display the total transaction amount.
- 4: View transaction history.
- 5: Exit the program.

Ensure transaction.txt exists or is created in the same directory as the executable, as it serves as the data storage for transactions