#include <iostream>

#include <fstream>

#include <vector>

#include <iomanip>

using namespace std;

class BankAccount {

private:

string username;

string password;

double balance;

vector<string> transactions;

public:

BankAccount() : balance(0.0) {}

void registerAccount() {

cout << "Enter username: ";

cin >> username;

cout << "Enter password: ";

cin >> password;

balance = 0.0;

saveToFile();

cout << "Account registered successfully!\n";

}

bool login() {

cout << "Enter username: ";

cin >> username;

cout << "Enter password: ";

cin >> password;

return loadFromFile();

}

void deposit() {

double amount;

cout << "Enter deposit amount: ";

cin >> amount;

if (amount > 0) {

balance += amount;

transactions.push\_back("Deposit: " + to\_string(amount));

saveToFile();

cout << "Deposit successful!\n";

} else {

cout << "Invalid amount.\n";

}

}

void withdraw() {

double amount;

cout << "Enter withdrawal amount: ";

cin >> amount;

if (amount > 0 && amount <= balance) {

balance -= amount;

transactions.push\_back("Withdrawal: " + to\_string(amount));

saveToFile();

cout << "Withdrawal successful!\n";

} else {

cout << "Invalid or insufficient funds.\n";

}

}

void checkBalance() {

cout << "Current balance: " << fixed << setprecision(2) << balance << "\n";

}

void viewTransactionHistory() {

if (transactions.empty()) {

cout << "No transaction history available.\n";

return;

}

cout << "Transaction History:\n";

for (const string& t : transactions) {

cout << t << "\n";

}

}

private:

void saveToFile() {

ofstream file(username + ".txt");

file << password << "\n" << balance << "\n";

for (const string& t : transactions) file << t << "\n";

file.close();

}

bool loadFromFile() {

ifstream file(username + ".txt");

if (!file) {

cout << "Account not found.\n";

return false;

}

string storedPass;

file >> storedPass >> balance;

file.ignore();

if (storedPass != password) {

cout << "Incorrect password.\n";

return false;

}

transactions.clear();

string transaction;

while (getline(file, transaction)) transactions.push\_back(transaction);

file.close();

cout << "Login successful!\n";

return true;

}

};

int main() {

BankAccount account;

int choice;

bool loggedIn = false;

while (true) {

cout << "\nBanking System\n";

cout << "1. Register\n2. Login\n3. Deposit\n4. Withdraw\n5. Check Balance\n6. View Transaction History\n7. Exit\n";

cout << "Choose an option: ";

cin >> choice;

switch (choice) {

case 1:

account.registerAccount();

break;

case 2:

loggedIn = account.login();

break;

case 3:

if (loggedIn) account.deposit();

else cout << "Please log in first.\n";

break;

case 4:

if (loggedIn) account.withdraw();

else cout << "Please log in first.\n";

break;

case 5:

if (loggedIn) account.checkBalance();

else cout << "Please log in first.\n";

break;

case 6:

if (loggedIn) account.viewTransactionHistory();

else cout << "Please log in first.\n";

break;

case 7:

cout << "Exiting system.\n";

return 0;

default:

cout << "Invalid choice, please try again.\n";

}

}

return 0;

}