

```
#include <iostream>

#include <vector>

using namespace std;

class Account {
public:
    int accountNumber;
    string accountHolderName;
    double balance;
    vector<string> transactionHistory;

    Account(int accountNumber, string accountHolderName, double initialBalance) {
        this->accountNumber = accountNumber;
        this->accountHolderName = accountHolderName;
        this->balance = initialBalance;
    }

    void deposit(double amount) {
        balance += amount;
        transactionHistory.push_back("Deposited: $" + to_string(amount));
    }

    void withdraw(double amount) {
        if (amount <= balance) {
            balance -= amount;
            transactionHistory.push_back("Withdrew: $" + to_string(amount));
        } else {
            cout << "Insufficient balance." << endl;
        }
    }
}
```

```

void checkBalance() {
    cout << "Account Balance: $" << balance << endl;
}

void printTransactionHistory() {
    cout << "Transaction History:" << endl;
    for (const string& transaction : transactionHistory) {
        cout << "- " << transaction << endl;
    }
}

};

int main() {
    int accountNumber;
    string accountHolderName;
    double initialBalance;

    cout << "Enter account number: ";
    cin >> accountNumber;

    cout << "Enter account holder name: ";
    cin.ignore();
    getline(cin, accountHolderName);

    cout << "Enter initial balance: ";
    cin >> initialBalance;

    Account account(accountNumber, accountHolderName, initialBalance);

    int choice;

```

```
do {  
    cout << "\n1. Deposit\n2. Withdraw\n3. Check Balance\n4. Print Transaction History\n5.  
Exit\n";  
    cout << "Enter your choice: ";  
    cin >> choice;  
  
    switch (choice) {  
        case 1: {  
            double amount;  
            cout << "Enter amount to deposit: ";  
            cin >> amount;  
            account.deposit(amount);  
            break;  
        }  
        case 2: {  
            double amount;  
            cout << "Enter amount to withdraw: ";  
            cin >> amount;  
            account.withdraw(amount);  
            break;  
        }  
        case 3:  
            account.checkBalance();  
            break;  
        case 4:  
            account.printTransactionHistory();  
            break;  
        case 5:  
            cout << "Exiting..." << endl;  
            break;  
        default:
```

```
        cout << "Invalid choice. Please try again." << endl;
    }
} while (choice != 5);

return 0;
}
```