

Simple ATM Simulator

This is a simple C++ program that simulates an ATM with basic functionalities like checking balance, withdrawing money, and depositing money.

- Class : The program revolves around the ATM class, which encapsulates the properties and functionalities of an ATM.
- Balance : A private data member that stores the user's current balance. It is private to ensure encapsulation, meaning it can only be modified or accessed through member functions.
- Constructor (ATM) : This constructor initializes the balance when an ATM object is created.
- Checkbalance function : This function simply displays the current balance.
- DepositMoney function : This function allows the user to deposit money into their account.
 - I. The user is prompted to enter an amount to deposit.
 - II. If the amount is valid (greater than 0), it is added to the current balance
 - III. After deposit, the new balance is displayed using the checkBalance() function.
 - IV. If the amount is invalid (e.g., negative or zero), an error message is displayed.
- WithdrawMoney Function: This function allows the user to withdraw money, ensuring the withdrawal amount does not exceed the available balance.
 - I. The user is prompted to enter an amount to withdraw.
 - II. If the amount is valid and does not exceed the available balance, the amount is deducted from the balance, and the new balance is displayed.

- III. If the amount exceeds the available balance, an error message ("Insufficient balance!") is shown.
- IV. For invalid amounts (e.g., negative or zero), another error message is displayed.

- **Displaymenu Function** : This function displays the ATM menu and allows the user to choose different options (check balance, deposit, withdraw, or exit).
 - I. The menu repeatedly displays a list of options (check balance, deposit, withdraw, and exit) until the user chooses to exit.
 - II. A do-while loop ensures the menu is shown repeatedly after each transaction.
 - III. The switch statement handles each choice:
 - 1) Calls checkbalance()
 - 2) Calls depositmoney()
 - 3) Calls Withdrawmoney()
 - 4) Exits the loop
- **Main Function** : The main() function creates an ATM object and calls the displayMenu() function to start the program. The program will continue showing the menu and handling user inputs until the user chooses to exit.

Source Code :

```

1  #include <iostream>
2  using namespace std;
3
4  class ATM
5  {
6  private:
7      double balance;
8
9  public:
10     // Constructor to initialize the balance
11     ATM()
12     {
13         balance = 1000.0; // Default balance
14     }
15
16     // Function to check balance
17     void checkBalance()
18     {
19         cout << "Your current balance is: $" << balance << endl;
20     }
21
22     // Function to deposit money
23     void depositMoney()
24     {
25         double amount;
26         cout << "Enter the amount to deposit: $";
27         cin >> amount;
28
29         if (amount > 0)
30         {
31             balance += amount;
32             cout << "$" << amount << " has been deposited successfully." << endl;
33             checkBalance();
34         }
35         else
36         {
37             cout << "Invalid deposit amount!" << endl;

```

```

38         }
39     }
40
41     // Function to withdraw money
42     void withdrawMoney()
43     {
44         double amount;
45         cout << "Enter the amount to withdraw: $";
46         cin >> amount;
47
48         if (amount > 0 && amount <= balance)
49         {
50             balance -= amount;
51             cout << "$" << amount << " has been withdrawn successfully." << endl;
52             checkBalance();
53         }
54         else if (amount > balance)
55         {
56             cout << "Insufficient balance!" << endl;
57         }
58         else
59         {
60             cout << "Invalid withdrawal amount!" << endl;
61         }
62     }
63
64     // Function to display the main menu
65     void displayMenu()
66     {
67         int choice;
68         do {
69             cout << "\nATM Main Menu:\n";
70             cout << "1. Check Balance\n";
71             cout << "2. Deposit Money\n";
72             cout << "3. Withdraw Money\n";
73             cout << "4. Exit\n";
74             int choice;
75             cin >> choice;
76             switch (choice) {
77                 case 1: checkBalance(); break;
78                 case 2: depositMoney(); break;
79                 case 3: withdrawMoney(); break;
80                 case 4: exit(0); break;
81                 default: cout << "Invalid choice!" << endl;

```

```

        cout << "4. Exit\n";
        cout << "Enter your choice (1-4): ";
        cin >> choice;

        switch (choice) {
            case 1:
                checkBalance();
                break;
            case 2:
                depositMoney();
                break;
            case 3:
                withdrawMoney();
                break;
            case 4:
                cout << "Thank you for using the ATM. Goodbye!" << endl;
                break;
            default:
                cout << "Invalid choice! Please select a valid option." << endl;
        }
    } while (choice != 4);
};

int main()
{
    ATM atm; // Create an ATM object
    atm.displayMenu(); // Display the ATM menu
    return 0;
}

```

Expected Output :

```

C:\Users\ISHAN\Docume × + v

ATM Main Menu:
1. Check Balance
2. Deposit Money
3. Withdraw Money
4. Exit
Enter your choice (1-4): |

```

```

C:\Users\ISHAN\Docume × + v

ATM Main Menu:
1. Check Balance
2. Deposit Money
3. Withdraw Money
4. Exit
Enter your choice (1-4): 1
Your current balance is: $1000

ATM Main Menu:
1. Check Balance
2. Deposit Money
3. Withdraw Money
4. Exit
Enter your choice (1-4): |

```

```

ATM Main Menu:
1. Check Balance
2. Deposit Money
3. Withdraw Money
4. Exit
Enter your choice (1-4): 2
Enter the amount to deposit: $200
$200 has been deposited successfully.
Your current balance is: $1200

ATM Main Menu:
1. Check Balance
2. Deposit Money
3. Withdraw Money
4. Exit
Enter your choice (1-4): |

```

```

ATM Main Menu:
1. Check Balance
2. Deposit Money
3. Withdraw Money
4. Exit
Enter your choice (1-4): 3
Enter the amount to withdraw: $500
$500 has been withdrawn successfully.
Your current balance is: $700

ATM Main Menu:
1. Check Balance
2. Deposit Money
3. Withdraw Money
4. Exit
Enter your choice (1-4): |

```

Few Summary :

- 1) The program uses basic OOP concepts, including encapsulation, to simulate an ATM.
- 2) It ensures proper validation for deposits and withdrawals.
- 3) The use of a loop allows the program to remain interactive until the user chooses to exit.

Conclusion :

The ATM Simulator program demonstrates basic C++ concepts like classes, functions, and control structures. It allows users to check their balance, deposit, and withdraw money, with input validation and a simple menu interface. The program uses loops and conditional statements for continuous interaction, and highlights object-oriented programming principles, such as encapsulation. It serves as a foundational project for further expansion, showcasing how basic C++ can be applied to real-world scenarios.