**ATM Interface Project using Java**

This project simulates a basic Automated Teller Machine (ATM) system. It is developed in Java and allows users to perform essential banking operations such as checking balance, depositing money, withdrawing money, and user authentication.

**Features**

* **User Authentication:** Verifies user identity using a user ID and PIN.
* **Check Balance:** Allows users to view their current account balance.
* **Deposit Money:** Enables users to deposit money into their account.
* **Withdraw Money:** Allows users to withdraw money from their account, provided they have sufficient balance.
* **Session Termination:** Users can exit the ATM session.

**Code Structure**

The project consists of the following classes:

1. **ATM:** Contains the main logic for the ATM interface, including user authentication and transaction processing.
2. **ATMMain:** Contains the main method to start the ATM interface.
3. **User:** Represents a user of the ATM system, with attributes for user ID, PIN, and account balance.

**Getting Started**

These instructions will get you a copy of the project up and running on your local machine for development and testing purposes.

**Prerequisites**

To run this project, you need to have Java installed on your machine.

* Java Development Kit (JDK) 8 or later.

**Running the Application**

1. Open a terminal and navigate to the project directory.
2. Compile the Java files using **javac** command:

javac com/atm/\*.java

1. Run the application:

java com.atm.ATMMain

**Usage**

1. **User Authentication:**

Enter your user ID and PIN when prompted for authentication.

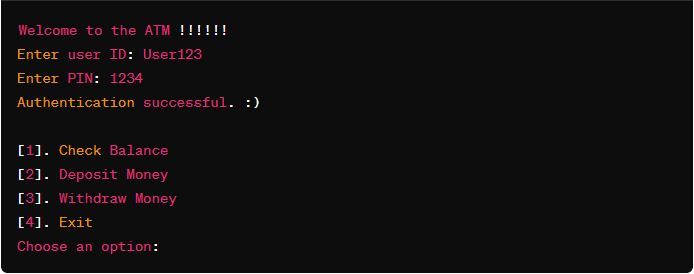
1. **Main Menu:**

Choose an operation you wish to perform (Check Balance, Deposit Money, Withdraw Money, Exit).

1. **Transaction Processing:**

Follow the on-screen prompts to complete your transaction.

**Example**this is the output example after running the code of “ATM interface”



**Comments in the Code**

* Comments have been added to explain the purpose of each class, method, and significant block of code.
* Comments also provide guidance on how to interact with the ATM system and what each part of the code does.

**Conclusion**

This project demonstrates a simple ATM system implemented in Java. It can be further expanded and improved to add more features and functionality.

This is all about this project. Enjoy using the ATM interface!