**README ::**

**1. BankAccount Class (Handles individual accounts)**

* Each account has:
  + accountHolder: Name of the account holder.
  + balance: Account balance.
* Methods:
  + deposit(amount): Adds money to the balance.
  + withdraw(amount): Deducts money if sufficient balance is available.

**2. Bank Class (Manages multiple accounts)**

* Stores all accounts in a HashMap.
* Methods:
  + createAccount(name, deposit): Creates a new account.
  + getAccount(name): Returns account details.

**3. BankGUI Class (Main GUI Interface)**

* **Frame** (JFrame): The main application window.
* **Text Fields** (JTextField): For entering account holder name and amount.
* **Text Area** (JTextArea): Displays transaction messages.
* **Buttons** (JButton):
  + Create Account: Creates a new bank account.
  + Deposit: Adds money to the account.
  + Withdraw: Deducts money from the account.
  + Check Balance: Displays the current balance.
* **Event Listeners** (ActionListener): Detects button clicks and performs corresponding actions.

**Program Output (GUI Interface)**

When the program runs, a window appears with input fields and buttons:

1️⃣ **Creating an Account**

* Enter **Name**: "John Doe"
* Enter **Initial Deposit**: "1000"
* Click Create Account
* Output: Account created successfully for John Doe

2️⃣ **Depositing Money**

* Enter **Name**: "John Doe"
* Enter **Amount**: "500"
* Click Deposit
* Output: Deposited: $500

3️⃣ **Withdrawing Money**

* Enter **Name**: "John Doe"
* Enter **Amount**: "300"
* Click Withdraw
* Output: Withdrawn: $300

4️⃣ **Checking Balance**

* Enter **Name**: "John Doe"
* Click Check Balance
* Output: Balance for John Doe: $1200

**Key Features:**

✅ Simple **GUI-based Banking System**  
✅ Uses **Swing** components for user interaction  
✅ **Stores multiple accounts** using HashMap  
✅ Handles **Deposits, Withdrawals, and Balance Checking**

This is a **basic version** of a banking system. You can further improve it by:

* Adding **error handling** for invalid inputs.
* Using **database storage** instead of HashMap.
* Implementing **transaction history**.

CREATED BY : CHETAN DHAWAS

PRIYADARSHANI JL COLLAGE OF ENGINEERING, NAGPUR

[Chetandhawas9@gmail.com](mailto:Chetandhawas9@gmail.com)