

Project 3 (Day 3): Bank Class for Account Management & Transfer

Description

Develop a `Bank` class that manages a collection of various `Account` objects. This class will allow the addition of accounts and provide a method to transfer funds between any two accounts (demonstrating polymorphism as it handles multiple account types).

Attributes

- **accounts** (Collection of `Account` objects): A list or array holding all account instances managed by the bank.

Methods

1. `addAccount(account)`

- **Purpose:** Add an `Account` (or any of its subclasses) to the bank's collection.
- **Return:** `Void`.

2. `findAccount(accountNumber)`

- **Purpose:** Search for an account using its `accountNumber`.
- **Return:** The found `Account` object or `null` if not found.

3. `transfer(fromAccountNumber, toAccountNumber, amount)`

- **Purpose:** Facilitate a transfer by withdrawing the amount from the source account and depositing it into the target account.
- **Return:** Boolean – `true` if the transfer is successful, `false` otherwise.
- **Behavior:**
 - Use `findAccount` to get both accounts.
 - If the withdrawal from the source account succeeds, deposit the amount into the

target account.

Manual Test Cases (Using if/else Logic)

- **Test Case 1: Add and Find Account**

- **Action:** Create two accounts (e.g., Account A with balance 1000 and Account B with balance 500) and add them to the Bank.
- **If/Else Check:**
 - **If** `findAccount("A_Number")` returns Account A, **then** print "Account A found successfully."
 - **Else** print "Error: Account A not found."

- **Test Case 2: Successful Transfer**

- **Action:** Transfer 300 from Account A (balance 1000) to Account B (balance 500).
- **If/Else Check:**
 - **If** `transfer("A_Number", "B_Number", 300)` returns `true`, **then** check that Account A's balance is now 700 and Account B's is 800; print "Transfer successful."
 - **Else** print "Error: Transfer failed."

- **Test Case 3: Failed Transfer Due to Insufficient Funds**

- **Action:** Attempt to transfer 1200 from Account A (balance 700 after the previous transfer) to Account B.
- **If/Else Check:**
 - **If** `transfer("A_Number", "B_Number", 1200)` returns `false`, **then** print "Transfer rejected due to insufficient funds."
 - **Else** print "Error: Transfer should have failed but succeeded."