


2043. Simple Bank System

Solved 

Medium

 Topics

 Companies

 Hint

You have been tasked with writing a program for a popular bank that will automate all its incoming transactions (transfer, deposit, and withdraw). The bank has n accounts numbered from 1 to n . The initial balance of each account is stored in a **0-indexed** integer array `balance`, with the $(i + 1)^{\text{th}}$ account having an initial balance of `balance[i]`.

Execute all the **valid** transactions. A transaction is **valid** if:

- The given account number(s) are between 1 and n , and
- The amount of money withdrawn or transferred from is **less than or equal** to the balance of the account.

Implement the `Bank` class:

- `Bank(long[] balance)` Initializes the object with the **0-indexed** integer array `balance`.
- `boolean transfer(int account1, int account2, long money)` Transfers `money` dollars from the account numbered `account1` to the account numbered `account2`. Return `true` if the transaction was successful, `false` otherwise.
- `boolean deposit(int account, long money)` Deposit `money` dollars into the account numbered `account`. Return `true` if the transaction was successful, `false` otherwise.
- `boolean withdraw(int account, long money)` Withdraw `money` dollars from the account numbered `account`. Return `true` if the transaction was successful, `false` otherwise.

Example 1:

Input

```
["Bank", "withdraw", "transfer", "deposit", "transfer", "withdraw"]  
[[[10, 100, 20, 50, 30]], [3, 10], [5, 1, 20], [5, 20], [3, 4, 15], [10, 50]]
```

Output

```
[null, true, true, true, false, false]
```

```
1 class Bank:
2
3     def __init__(self, balance: List[int]):
4         self.balance = balance
5         self.n = len(balance)
6
7     def helper_valid(self, acc):
8         return 0 <= acc <= self.n
9
10    def transfer(self, account1: int, account2: int, money: int) -> bool:
11        if not self.helper_valid(account1) or not self.helper_valid(account2) or self.balance[account1-1] < money:
12            return False
13
14        self.balance[account1-1] -= money
15        self.balance[account2-1] += money
16        return True
17
18    def deposit(self, account: int, money: int) -> bool:
19        if not self.helper_valid(account):
20            return False
21
22        self.balance[account-1] += money
23        return True
24
25    def withdraw(self, account: int, money: int) -> bool:
26        if not self.helper_valid(account) or self.balance[account-1] < money:
27            return False
28
29        self.balance[account - 1] -= money
30        return True
```