```java
public class Bank {

    private Customer[] customers;  // Array to store customer accounts

    public Bank() {
        // Initialize the customer array with a fixed size initially
        customers = new Customer[10];
    }

    public void createAccount(String name, double balance) {
        // Find an empty slot in the customer array
        int index = findEmptySlot();
        if (index != -1) {
            customers[index] = new Customer(name, balance);
            System.out.println("Account created successfully for " + name);
        } else {
            System.out.println("Customer list is full. Account creation failed.");
        }
    }

    public int findEmptySlot() {
        for (int i = 0; i < customers.length; i++) {
            if (customers[i] == null) {
                return i;
            }
        }
        return -1; // All slots are full
    }

    public void deposit(String name, double amount) {
        Customer customer = findCustomer(name);
        if (customer != null) {
            customer.deposit(amount);
            System.out.println("Deposit successful for " + name + ". New balance: $" + customer.get
        } else {
            System.out.println("Account not found for " + name);
        }
    }

    public void withdraw(String name, double amount) {
        Customer customer = findCustomer(name);
        if (customer != null) {
            if (customer.withdraw(amount)) {
                System.out.println("Withdrawal successful for " + name + ". New balance: $" + custo
            } else {
                System.out.println("Insufficient funds for withdrawal.");
            }
        } else {
            System.out.println("Account not found for " + name);
        }
    }

    public void transfer(String fromName, String toName, double amount) {
        Customer fromCustomer = findCustomer(fromName);
        Customer toCustomer = findCustomer(toName);

        if (fromCustomer != null && toCustomer != null) {
            if (fromCustomer.withdraw(amount)) {
                toCustomer.deposit(amount);
                System.out.println("Transfer successful from " + fromName + " to " + toName + ".");
            } else {
                System.out.println("Insufficient funds in " + fromName + "'s account.");
            }
        } else {
            if (fromCustomer == null) {
```

```
                System.out.println("Account not found for " + fromName);
            }
            if (toCustomer == null) {
                System.out.println("Account not found for " + toName);
            }
        }
    }

    private Customer findCustomer(String name) {
        for (Customer customer : customers) {
            if (customer != null && customer.getName().equals(name)) {
                return customer;
            }
        }
        return null;
    }

    public static void main(String[] args) {
        // Create a Bank object
        Bank bank = new Bank();

        // Sample usage: Create accounts and perform transactions
        bank.createAccount("Alice", 100);
        bank.createAccount("Bob", 50);

        bank.deposit("Alice", 20);
        bank.withdraw("Bob", 30);
        bank.transfer("Alice", "Bob", 10);
    }
}

class Customer {

    private String name;
    private double balance;

    public Customer(String name, double balance) {
        this.name = name;
        this.balance = balance;
    }

    public String getName() {
        return name;
    }

    public double getBalance() {
        return balance;
    }

    public void deposit(double amount) {
        balance += amount;
    }

    public boolean withdraw(double amount) {
        if (balance >= amount) {
            balance -= amount;
            return true;
        } else {
            return false;
        }
    }
}
```

External Libraries Upload Files Input Output JDroid

```
Account created successfully for Alice
Account created successfully for Bob
Deposit successful for Alice. New balance: $120.0
Withdrawal successful for Bob. New balance: $20.0
Transfer successful from Alice to Bob.
```

Compiled and & executed in **1.393 sec(s)**

Generated Files

Share this awesome tool with your peers

Facebook Twitter Email LinkedIn Copy Link

Like coding with JDoodle? Share a review!

LEAVE A REVIEW ON Product Hunt