```java
public class BankAccount {

    private String accountNumber;

    private double balance;


    public BankAccount(String accountNumber,
    double initialBalance) {

        if (initialBalance < 0) {

            throw new IllegalArgumentException("Initial
    balance cannot be negative");

        }

        this.accountNumber = accountNumber;

        this.balance = initialBalance;

    }


    public String getAccountNumber() {

        return accountNumber;

    }


    public double getBalance() {

        return balance;

    }
```

```java
public void deposit(double amount) {

    if (amount <= 0) {

        throw new
IllegalArgumentException("Deposit amount must be
positive");

    }

    balance += amount;

}


public void withdraw(double amount) {

    if (amount <= 0) {

        throw new
IllegalArgumentException("Withdrawal amount
must be positive");

    }

    if (amount > balance) {

        throw new
IllegalArgumentException("Insufficient funds");

    }

    balance -= amount;

}
```

```java
    public void transfer(BankAccount targetAccount,
double amount) {

        if (targetAccount == null) {

            throw new IllegalArgumentException("Target
account cannot be null");

        }

        this.withdraw(amount);

        targetAccount.deposit(amount);

    }
}
```