

Задание 6 - Многопоточное программирование

Банк

Структура программы:

- Observer.java
- Logger.java
- Client.java
- Transaction.java
- DepositTransaction.java
- WithdrawTransaction.java
- ExchangeTransaction.java
- TransferTransaction.java
- Cashier.java
- Bank.java
- BankSimulation.java

Описание классов:

1. Классы Observer и Logger реализуют паттерн наблюдатель для логирования событий.
2. Класс Client создает клиента банка с соответствующими полями: идентификатор, баланс, валюта и механизмом синхронизации ReentrantLock. Каждый клиент имеет свою собственную блокировку.
3. Класс Cashier является наследником класса Thread. Касса в бесконечном цикле берет транзакции из очереди в основном методе run(). Каждая касса работает в отдельном потоке.
4. Абстрактный класс Transaction представляет банковскую операцию как объект, с основным методом process()

Содержит:

- type - тип операции (DEPOSIT, WITHDRAW, EXCHANGE, TRANSFER)
 - clientId - идентификатор клиента
 - amount - сумма операции
 - абстрактный метод process(Bank bank)
5. Основные операции и им соответствующие классы: DepositTransaction - Пополнение счета, WithdrawTransaction - Снятие средств, ExchangeTransaction - Обмен валют, TransferTransaction - Перевод между клиентами
 6. Основной класс Bank, функционал: Хранит данные всех клиентов и курсов валют, Управляет потоками-кассами, Обработывает очереди транзакций.
 7. BankExample - реализует пример работы банка.

