Федеральное государственное автономное образовательное учреждение

высшего образования «Национальный исследовательский университет

ИТМО»

*Факультет программной инженерии и компьютерной техники*

*Направление подготовки: 09.03.04 – Программная инженерия, Системное и прикладное программное обеспечение.*

*Дисциплина «Программирование»*

**Отчет**

**По лабораторной работе №6**

**Программа на языке Java**

**Вариант №2711113**

Выполнил:

Молчанов Фёдор Денисович

Группа: Р3113

Преподаватель: Письмак Алексей Евгеньевич

Г. Санкт-Петербург, 2024 г.

# Задание

Разделить программу из лабораторной работы №5 на клиентский и серверный модули. Серверный модуль должен осуществлять выполнение команд по управлению коллекцией. Клиентский модуль должен в интерактивном режиме считывать команды, передавать их для выполнения на сервер и выводить результаты выполнения.

Необходимо выполнить следующие требования:

    Операции обработки объектов коллекции должны быть реализованы с помощью Stream API с использованием лямбда-выражений.

    Объекты между клиентом и сервером должны передаваться в сериализованном виде.

    Объекты в коллекции, передаваемой клиенту, должны быть отсортированы по размеру

    Клиент должен корректно обрабатывать временную недоступность сервера.

    Обмен данными между клиентом и сервером должен осуществляться по протоколу TCP

    Для обмена данными на сервере необходимо использовать потоки ввода-вывода

    Для обмена данными на клиенте необходимо использовать сетевой канал

    Сетевые каналы должны использоваться в неблокирующем режиме.

Обязанности серверного приложения:

    Работа с файлом, хранящим коллекцию.

    Управление коллекцией объектов.

    Назначение автоматически генерируемых полей объектов в коллекции.

    Ожидание подключений и запросов от клиента.

    Обработка полученных запросов (команд).

    Сохранение коллекции в файл при завершении работы приложения.

    Сохранение коллекции в файл при исполнении специальной команды, доступной только серверу (клиент такую команду отправить не может).

Серверное приложение должно состоять из следующих модулей (реализованных в виде одного или нескольких классов):

    Модуль приёма подключений.

    Модуль чтения запроса.

    Модуль обработки полученных команд.

    Модуль отправки ответов клиенту.

Сервер должен работать в однопоточном режиме.

Обязанности клиентского приложения:

    Чтение команд из консоли.

    Валидация вводимых данных.

    Сериализация введённой команды и её аргументов.

    Отправка полученной команды и её аргументов на сервер.

    Обработка ответа от сервера (вывод результата исполнения команды в консоль).

    Команду save из клиентского приложения необходимо убрать.

    Команда exit завершает работу клиентского приложения.

Важно! Команды и их аргументы должны представлять из себя объекты классов. Недопустим обмен "простыми" строками. Так, для команды add или её аналога необходимо сформировать объект, содержащий тип команды и объект, который должен храниться в вашей коллекции.

Дополнительное задание:

Реализовать логирование различных этапов работы сервера (начало работы, получение нового подключения, получение нового запроса, отправка ответа и т.п.) с помощью Log4J2

Отчёт по работе должен содержать:

    Текст задания.

    Диаграмма классов разработанной программы (как клиентского, так и серверного приложения).

    Исходный код программы.

    Выводы по работе.

Вопросы к защите лабораторной работы:

    Сетевое взаимодействие - клиент-серверная архитектура, основные протоколы, их сходства и отличия.

    Протокол TCP. Классы Socket и ServerSocket.

    Протокол UDP. Классы DatagramSocket и DatagramPacket.

    Отличия блокирующего и неблокирующего ввода-вывода, их преимущества и недостатки. Работа с сетевыми каналами.

    Классы SocketChannel и DatagramChannel.

    Передача данных по сети. Сериализация объектов.

    Интерфейс Serializable. Объектный граф, сериализация и десериализация полей и методов.

    Java Stream API. Создание конвейеров. Промежуточные и терминальные операции.

    Шаблоны проектирования: Decorator, Iterator, Factory method, Command, Flyweight, Interpreter, Singleton, Strategy, Adapter, Facade, Proxy.

# Выполнение задания

Исходный код доступен по ссылке: <https://github.com/fefumo/itmo/tree/main/Prog/lab6>

# Выводы

Во время выполнения данной лабораторной работы я научился работать tcp клиентом и севрером.