

Факультет Программной Инженерии и Компьютерной Техники

Программирование

Лабораторная работа №5

Вариант № 820

Преподаватель: Харитонова Анастасия Евегеньевна

Выполнил: Бахтаиров Илья Андреевич

Группа: R3181

Санкт-Петербург

2020

**Текст задания**

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 умолчанию
* 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 **сетевой канал**
* Для обмена данными на клиенте необходимо использовать **потоки ввода-вывода**
* 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.

**Исходный код**

Доступен по ссылке:

<https://github.com/FJPO/ifmo_lab6>

**Диаграмма классов**

См Приложение.

**Вывод**

В ходе выполнения данной лабораторной работы я научился использовать пакет java.net, предоставляющий возможность сетевого взаимодействия. Помимо этого, я познакомился с достаточно элегантным способом обработки массива или коллекции – стримами и их Stream API. Также интерес для меня представляют селекторы из java.nio, позволяющие обрабатывать сигналы от множества клиентов, не привлекая зачастую затратную многопоточность.

**Приложение**

Пакет Lab6 (Он один и для сервера, и для клиента, при этом за работу сервера отвечает класс ServerSide, а за работу клиента – Client Side):

