HW01

Project 'Producer-Queue-Consumer' Решение проекта реализовано в пакете producer\_queue\_consumer с использованием модели "один ко многим"

Task01 Реализовать поток Producer бесконечно считывающий целые числа, которые пользователь вводит в Standard Input.-Реализованно через цикл While (стр. 20 в классе Producer).

Task02 Поток Producer сохраняет целые числа в Queue – реализовано в стр.24 в классе Producer.

Task03 При вводе -1 поток Producer завершает свою работу. – реализовано через проверки if стейтмент в цикле while класса Producer

Task04 Когда пользователь вводит некорректные данные, то поток Producer бросает исключение UserInputException, но при этом продолжает работать. Реализованно с использованием класса UserInputException в пакете exception и отловом ошибки через метод handleException в классе Producer.

Task05 Реализовать поток Consumer, который забирает числа из Queue и засыпает на количество секунд равное этому числу. При этом засыпание потока Consumer происходит в том порядке, в котором значения целых чисел были введены пользователем. – Реализованно в классе Consumer в пакете subscriber, который имплементирует интерфейсы runnable(для создания потока) и ISleep (для отправки потока в сон)

Task06 После того как поток Consumer проснется, он выводит в File: timestamp−{thread-name} - I slept ${seconds} seconds.

Task 7 Когда у потока Consumer нет текущих задач, то он выводит в File: timestamp−{thread-name} - ... и засыпает на 1 секунду.

Реализация записи в фаил выполнена в классе FileCreator методом write и строками 20-24 в классе Consumer

Task08-09 Возможность добавления нескольких потоков Consumer и конфигурации их количества

Для удобства создания n-го количества потоков был реализован «фабричный метод» newFixedThreadPool

Демонстрация работы находится в тестовом классе Demo

Структура пакетов в проекте:

 