

Национальный исследовательский университет
информационных технологий, механики и оптики
Факультет программной инженерии и компьютерной техники

Лабораторная работа №7 по Программированию

Выполнил: Цыпандин Николай Петрович
Группа: Р3110
Вариант: 1087

Санкт-Петербург
2021

Задание:

Доработать программу из [лабораторной работы №6](#) следующим образом:

1. Организовать хранение коллекции в реляционной СУБД (PostgreSQL). Убрать хранение коллекции в файле.
2. Для генерации поля id использовать средства базы данных (sequence).
3. Обновлять состояние коллекции в памяти только при успешном добавлении объекта в БД
4. Все команды получения данных должны работать с коллекцией в памяти, а не в БД
5. Организовать возможность регистрации и авторизации пользователей. У пользователя есть возможность указать пароль.
6. Пароли при хранении хэшировать алгоритмом **SHA-384**
7. Запретить выполнение команд не авторизованным пользователям.
8. При хранении объектов сохранять информацию о пользователе, который создал этот объект.
9. Пользователи должны иметь возможность просмотра всех объектов коллекции, но модифицировать могут только принадлежащие им.
10. Для идентификации пользователя отправлять логин и пароль с каждым запросом.

Необходимо реализовать многопоточную обработку запросов.

1. Для многопоточного чтения запросов использовать **создание нового потока (java.lang.Thread)**
2. Для многопоточной обработки полученного запроса использовать **Cached thread pool**
3. Для многопоточной отправки ответа использовать **создание нового потока (java.lang.Thread)**
4. Для синхронизации доступа к коллекции использовать **синхронизацию чтения и записи с помощью java.util.concurrent.locks.ReentrantLock**

Порядок выполнения работы:

1. В качестве базы данных использовать PostgreSQL.
2. Для подключения к БД на кафедральном сервере использовать хост **pg**, имя базы данных - **studs**, имя пользователя/пароль совпадают с таковыми для подключения к серверу.

Отчёт по работе должен содержать:

1. Текст задания.
2. Диаграмма классов разработанной программы.
3. Исходный код программы.
4. Выводы по работе.

Вопросы к защите лабораторной работы:

1. Многопоточность. Класс **Thread**, интерфейс **Runnable**. Модификатор **synchronized**.
2. Методы **wait()**, **notify()** класса **Object**, интерфейсы **Lock** и **Condition**.
3. Классы-синхронизаторы из пакета **java.util.concurrent**.
4. Модификатор **volatile**. Атомарные типы данных и операции.
5. Коллекции из пакета **java.util.concurrent**.

6. Интерфейсы `Executor`, `ExecutorService`, `Callable`, `Future`
7. Пулы потоков
8. JDBC. Порядок взаимодействия с базой данных. Класс `DriverManager`. Интерфейс `Connection`
9. Интерфейсы `Statement`, `PreparedStatement`, `ResultSet`, `RowSet`
10. Шаблоны проектирования.

Код и диаграмма классов:

<https://github.com/kkkooolllyyyaaa/Laba7>

Вывод: в ходе выполнения лабораторной работы познакомился с многопоточностью, пулами потоков и использовал их в своем коде. Также, я узнал, как работать с JDBC и задействовал в своей программе реляционную базу данных PostgreSQL.