|  |  |  |
| --- | --- | --- |
|  |  |  |
| МИНОБРНАУКИ РОССИИ | | |
| Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** | | |

Институт Информационных технологий

Кафедра Инструментального и прикладного программного обеспечения

**ПРАКТИЧЕСКАЯ РАБОТА №7**

по дисциплине «Разработка клиент-серверных приложений»

**Студент группы** ИКБО-20-19 Анваржонов Ж. Т

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись студента)

**Руководитель практической работы** преподаватель Волков М.Ю.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись руководителя)

Работа представлена «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_ 2022 г.

Допущен к работе «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_ 2022 г.

Москва 2022

**Создание реактивного приложения реального времени с использованием Reactor**

**Задание:** Реализовать сервис предоставляющий реактивный REST API с использованием Spring WebFlux. Предметная область выбирается студентом самостоятельно, но должна быть уникальна в рамках учебной группы.

**Требования к сервису:**

- Представлено как минимум 5 endpoint’ов;

- Представлены endpoint’в возвращающие как Mono<>, так и Flux<>;

- Продемонстрирована работа с базами данных;

- Продемонстрировано применение операторов преобразования потоков;

- Продемонстрирована работа в формате backpressure;

- Продемонстрирована обработка ошибок в потоках;

- Сервис покрыт unit-тестами.

**Выполнение задания**

В качестве предметной области было выбрано “Сейфы”

Продемонстрируем 5 endploint’ов, возвращающие как Mono, так и Flux

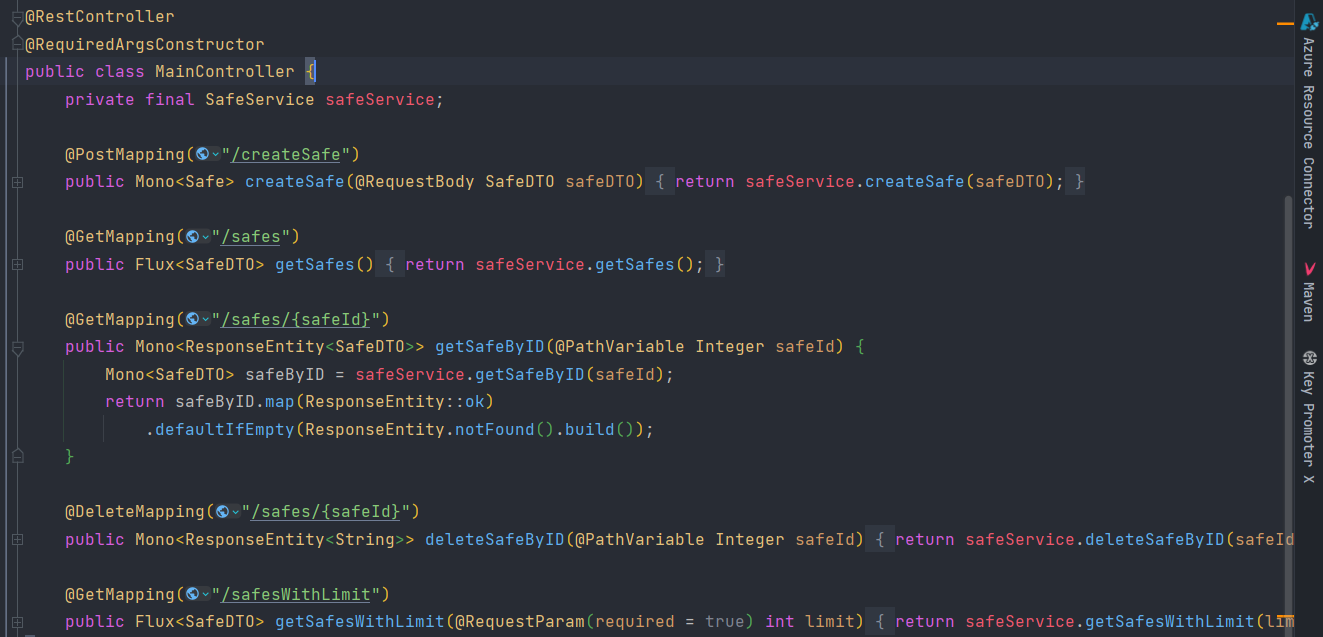


Рисунок 1 – демонстрация 5 endpoint’ов

Продемонстрируем работу с базами данными:

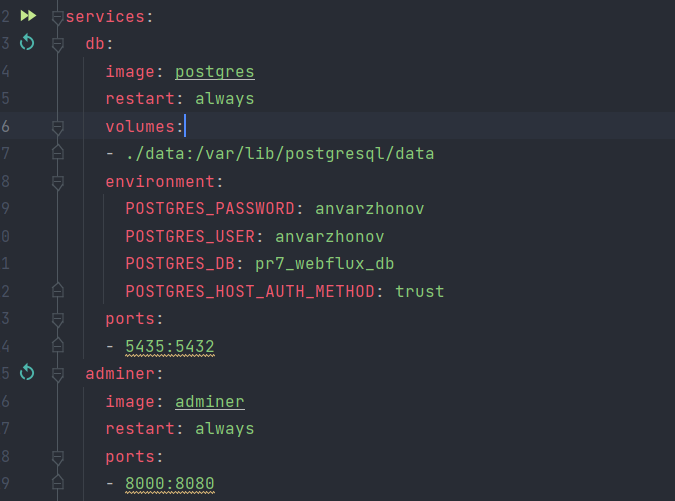


Рисунок 7.2 – docker-compose с postgresql и adminer

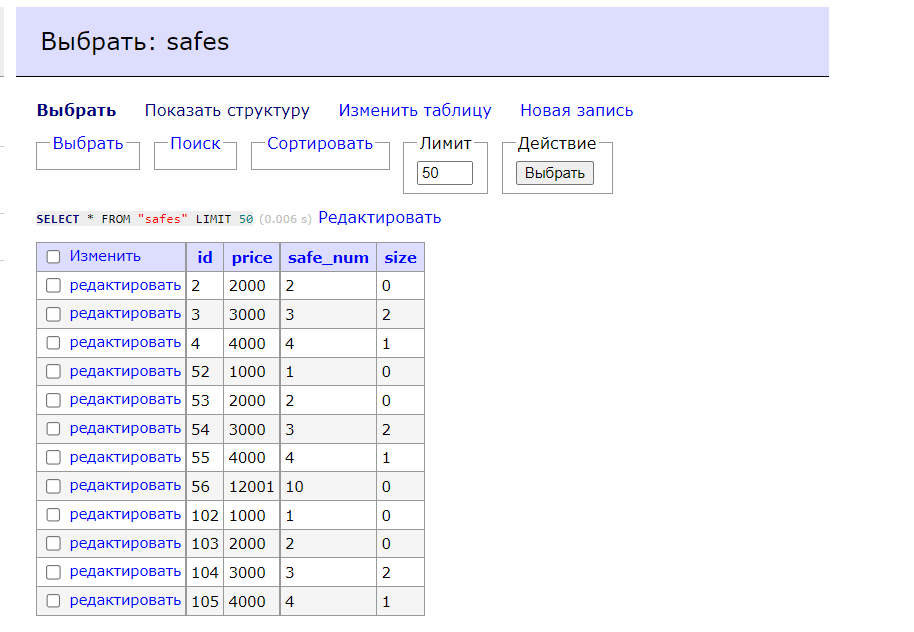
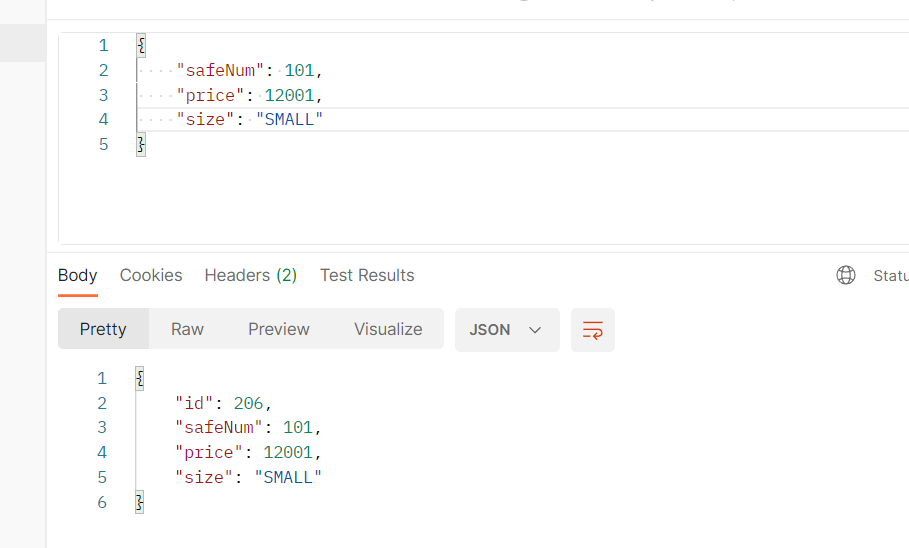


Рисунок 7.3 – таблица safes в бд

Покажем работоспособность с базой данных

1. Создадим новый сейф:



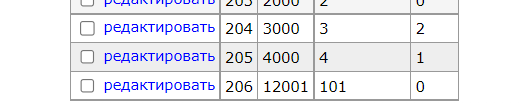


Рисунок 7.4 – сохранение нового сейфа

1. Получим список всех сейфов

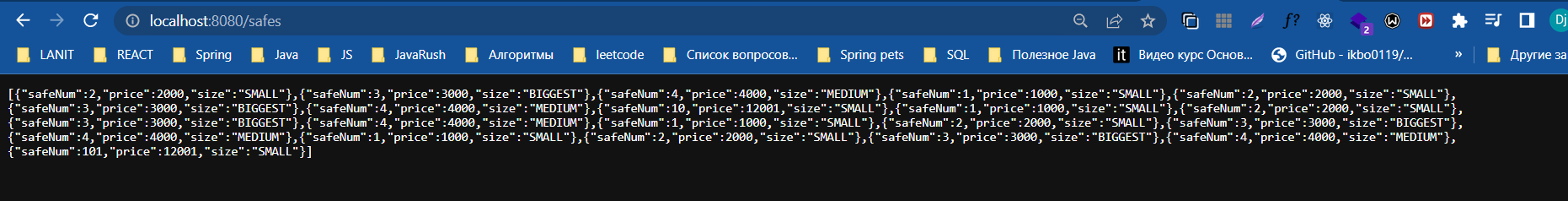


Рисунок 7.5 – список всех сейфов

1. Получение сейфа по конкретному ID:

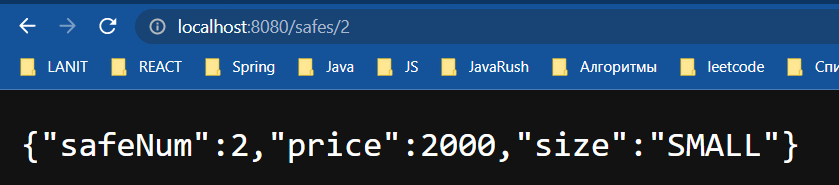


Рисунок 7.6 – получение сейфа по Id

1. Удаление сейфа по ID:

Fkasfkkfa

1. Получение списка сейфов, с использованием backpressure. Здесь с клиентской стороны задаем сколько сейфов хотим видеть

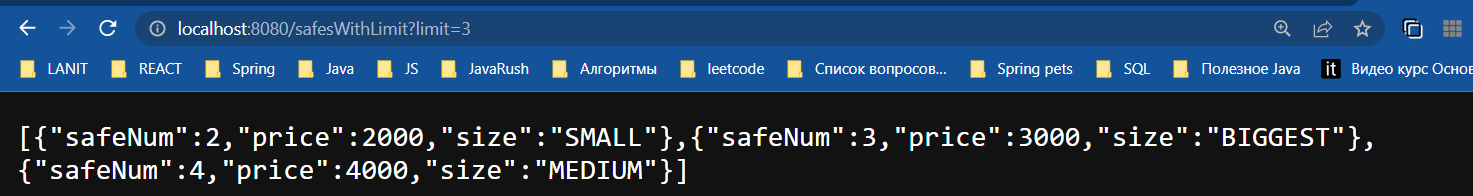


Рисунок 7.7 - демонстрация работы в формате backpressure

Продемонстрируем применение операторов преобразования потоков

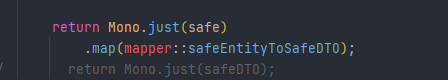


Рисунок 7.8 – преобразование потоков с помощью map