**Практическая работа №4**

В ходе выполнения практической работы было создано клиент-серверное приложение с использованием протокола RSocket.

Был создан класс Hat – основной класс объекта данных. Он представлен на рисунке 1.

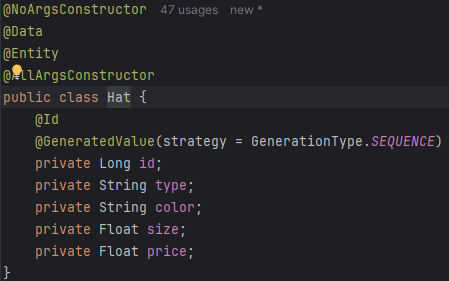


Рисунок 1 – Класс Hat

Был создан репозиторий, он представлен на рисунке 2.

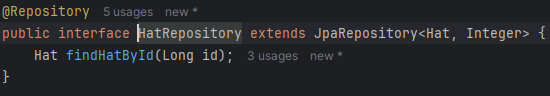


Рисунок 2 – Репозиторий HatRepository

Серверный контроллер представлен на рисунке 3.



Рисунок 3 – Серверный контроллер

Конфигурация клиента для того, чтобы клиент смог связаться с

сервером через RSocket представлена на рисунке 4.

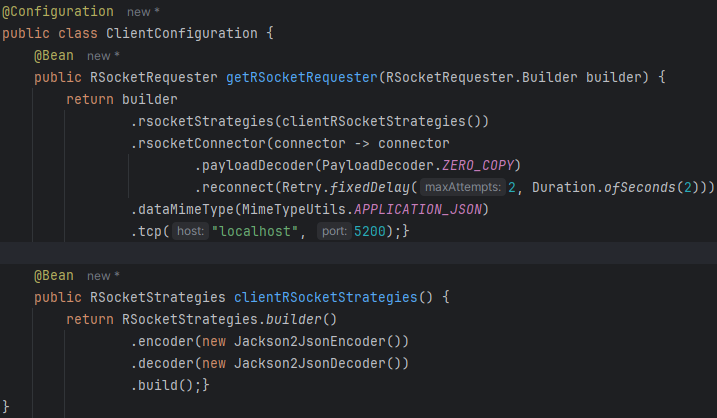


Рисунок 4 – Конфигурация клиента

Request-Response контроллер представлен на рисунке 5.

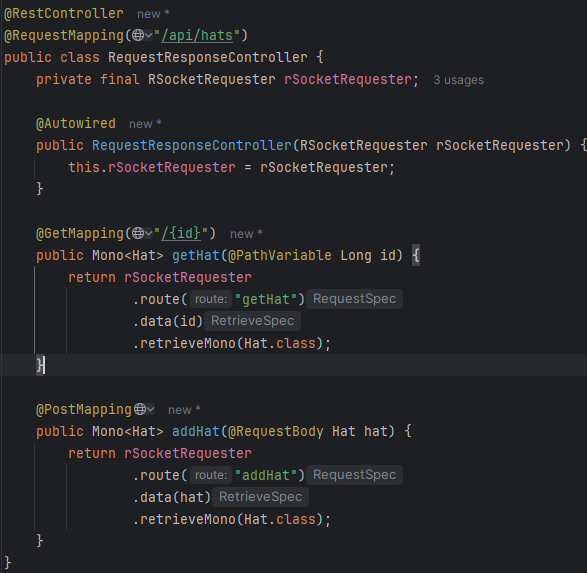


Рисунок 5 – Request-Response контроллер

Request-Stream контроллер представлен на рисунке 6.

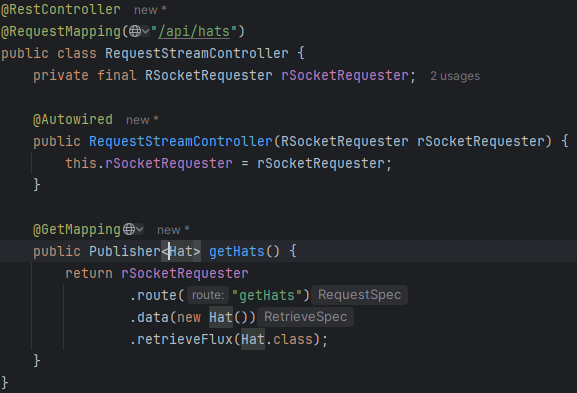


Рисунок 6 – Request-Stream контроллер

FireAndForget контроллер представлен на рисунке 7.



Рисунок 7 – FireAndForget контроллер

Channel контроллер представлен на рисунке 8.



Рисунок 8 – Channel контроллер

Серверная часть была покрыта тестами. Unit тесты представлены на рисунках 9 и 10.



Рисунок 9 – Unit тесты



Рисунок 10 – Unit тесты

Пример работы с сервисом преставлен на рисунках 11 – 15.

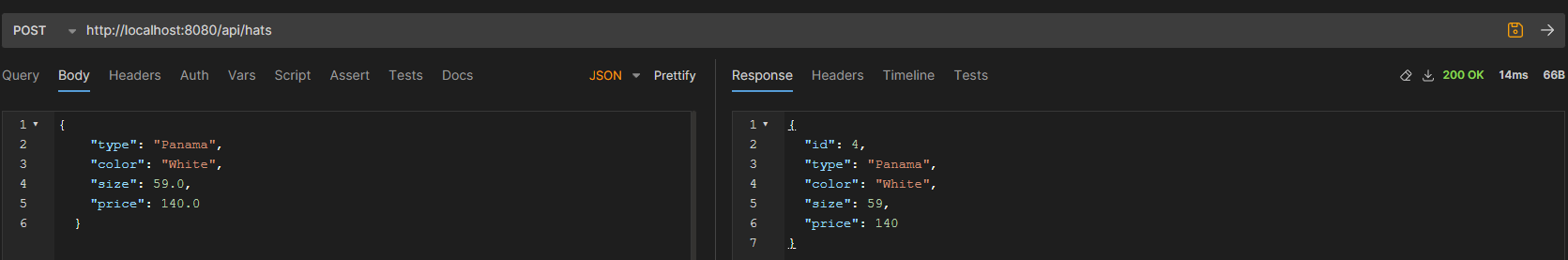
****

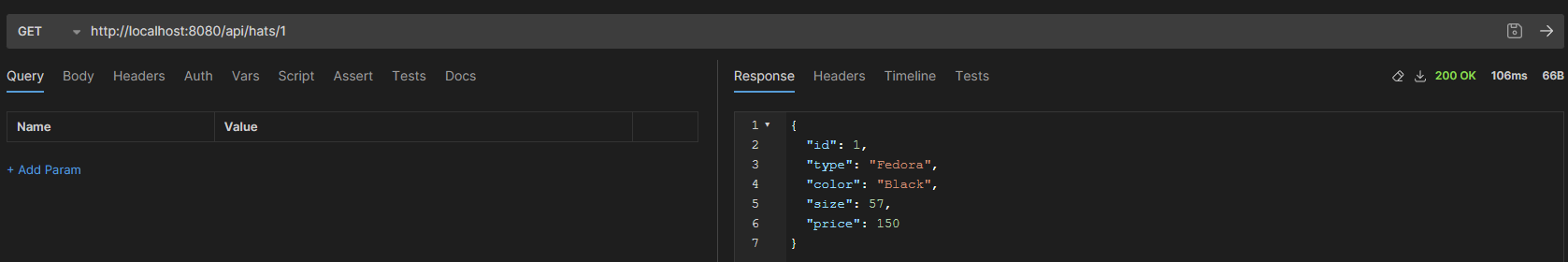
Рисунок 11 – Пример создания шляпы ****

Рисунок 12 – Пример получения шляпы

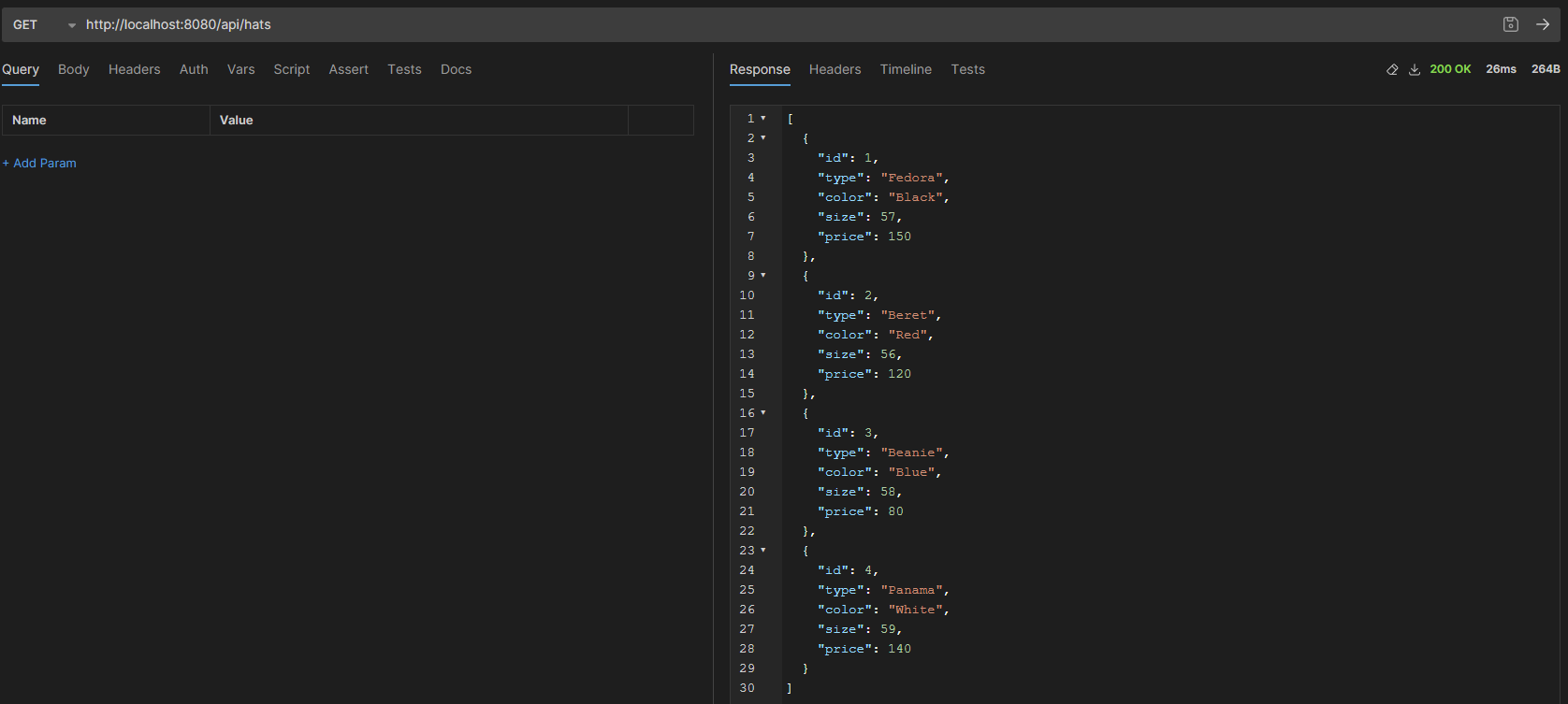
****

Рисунок 13 – Пример получения шляп

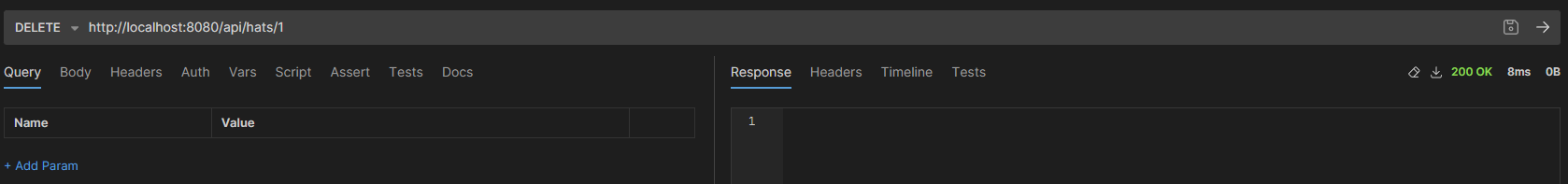
****

Рисунок 14 – Пример удаления шляпы

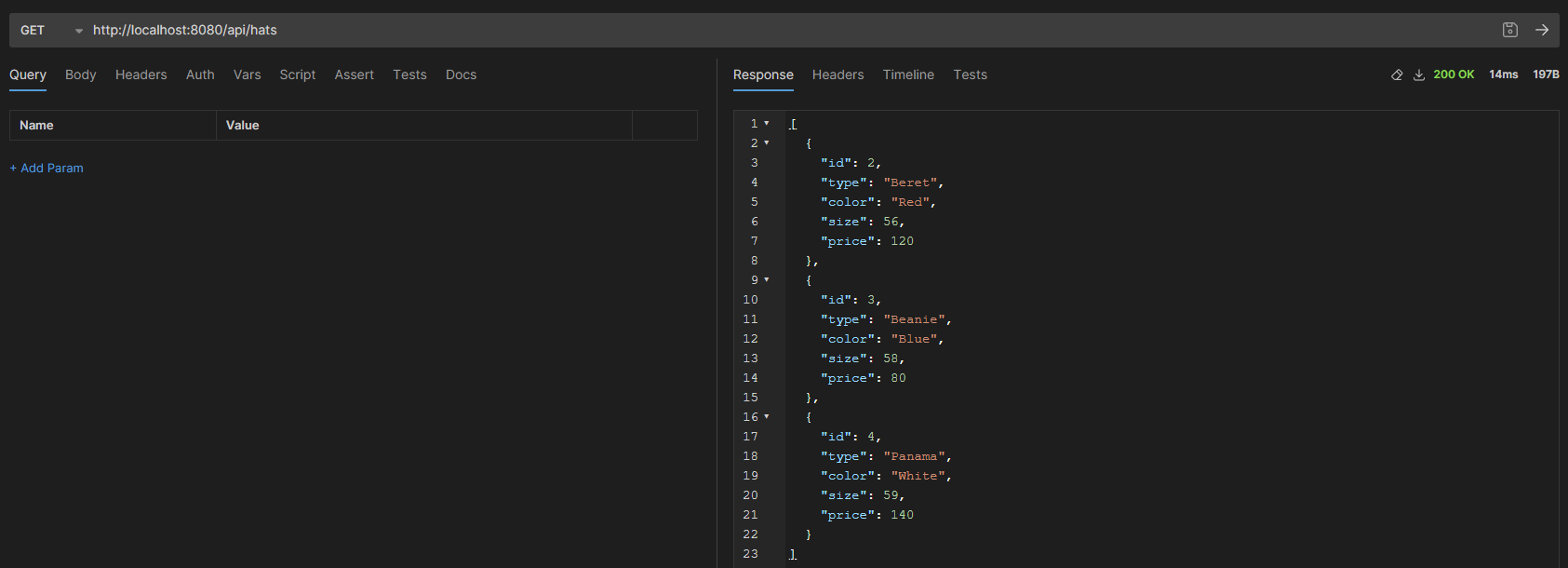
****

Рисунок 15 – Результат удаления шляпы

Взаимодействие с бд на уровне приложения представлено на рисунке 16.

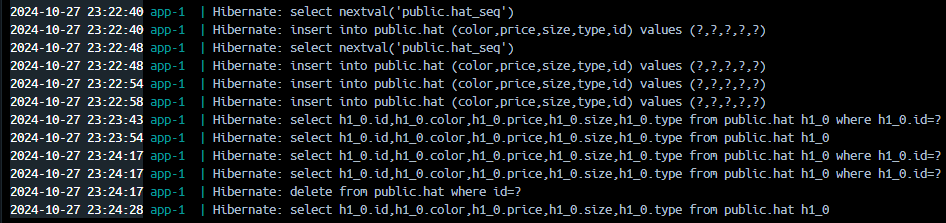
****

Рисунок 16 – Взаимодействие с бд