

Домашнее задание будет включать следующие шаги

1. Установить postgres на ПК
2. В сервисе настроить конфигурацию для подключения к БД через стартер spring-data-jpa
3. На старте приложение должно автоматически создавать схему с именем animals. Способ создания выбрать самостоятельно.
4. В liquibase сделать несколько xml с добавлением следующих таблиц:
  - a. таблица animal
  - b. таблица animal\_typeКонфигурацию (yml / xml) можно выбрать по желанию. Учесть первичные ключи и связи.
5. Заполнение первичных данных нужно пролить через liquibase. Рандомные данные можно исключить. Желающие оставить рандомную генерацию могут воспользоваться [этим](#)
6. Для указанных выше таблиц создать entity. Теперь сущности Dog/Cat/Shark и прочие удаляются. Все типы животных контролируются через сущность animal\_type. В описании entity использовать аннотации ManyToOne / OneToMany между animal и animal\_type
7. При создании репозитория использовать наследование от CrudRepository или JpaRepository. Класс AnimalsRepositoryImpl не должен хранить состояния в Map, а должен обращаться в БД
8. Бонусом будет написание интеграционных тестов с использованием testcontainers.
9. Супер бонусом будет, если ранее использованные методы
  - a. findLeapYearNames
  - b. findOlderAnimal
  - c. findDuplicate

.....

и остальные будут реализованы. Место реализации (сервисный слой или методы / query у репозитория) выбрать самостоятельно. Если на этот пункт времени/сил/желания не хватает, то сервисный слой должен содержать методы:

- a. List<Animal> getAnimalsByName
- b. List<Animal> getAllAnimals
- c. void deleteAnimal
- d. void addAnimal

По желанию можно добавить различные вариации CRUD с byId, byName или любой другой комбинацией полей.