

# Практическая часть

Задание:

Разработать REST сервис для работы с музыкальными композициями, данный сервис должен хранить информацию о композициях, альбомах и авторах в субд и предоставлять доступ на получение и изменение информации.

Техническая реализация:

В итоговой программе должны присутствовать следующие эндпоинты:

1. Работа с композициями:
  - a. Добавление композиции
  - b. Удаление композиции
  - c. Поиск композиций по названию
  - d. Поиск композиций по альбому
  - e. Поиск композиций по автору
  - f. Поиск композиции по ID
2. Работа с альбомами:
  - a. Добавление альбома
  - b. Удаление альбома
  - c. Поиск альбомов по названию
  - d. Поиск альбомов по автору
  - e. Поиск альбомов по композиции
  - f. Поиск альбома по ID
3. Работа с авторами:
  - a. Добавление альбома
  - b. Удаление альбома
  - c. Поиск авторов по псевдониму
4. Общие:
  - a. Удалить все данные

Минимальный набор свойств для каждого объекта:

Композиции:

- ID
- Название
- Продолжительность
- Автор
- Альбом
- Год создания

Альбомы:

- ID
- Название

- Дата выхода
- Сингл альбом (да/нет)
- Сборник (да/нет)
- Список композиций
- Жанр

Авторы:

- Псевдоним
- ФИО
- Дата рождения
- Страна рождения
- Жанр

Технологии:

Язык программирования - Java

Фреймворк - Spring Boot

Сборка проектов - Maven

Сериализация рест объектов - jackson/Gson

База данных - любая SQL база, самый простой вариант - использовать что то типа com.h2database

Тестирование - junit

Критерии оценки:

Сдача задания разделена на этапы со своими дедлайнами:

1. Архитектура приложения, создание объектов до 15.12.2018
2. Окончательная реализация в рамках зачетной недели

Каждый пропуск контрольной даты дает -1 к итоговой оценке.

Так есть следующие требования к выполненной работе:

1. Код должен соответствовать Java Code Conventions
2. В проекте должны использовать объектные модели с последующей (де)сериализацией
3. Для работы с базой данных должен быть создан DAO (варианты реализации могут быть любыми)
4. Код должен быть уникальным

Дополнительные задачи (если сроки не были соблюдены):

1. Реализовать юнит тесты для тестирования проекта во время сборки
2. Реализовать вывод ошибок
3. Реализовать удаление связанных объектов

Полезные ссылки:

Java Code Conventions:

<http://www.magnumblog.space/other/131-translating-java-code-conventions>

Spring boot (есть простой пример с описанием, а также ссылки на более подробную литературу и проекты на гитхабе):

<http://spring-projects.ru/projects/spring-boot/>