

Написать REST API для создания, настройки и запуска рекламных кампаний.

Модель кампании:

campaigns	
id	int
name	varchar
status	integer
start_date	timestamp
end_date	timestamp

Статусы для кампаний и рекламных объявлений:

0 - planned (кампании и рекламные объявления должны создаваться с этим статусом)

1 - active

2 - paused

3 - finished

Каждая кампания должна содержать хотя бы одно рекламное объявление:

ads	
id	int
name	varchar
status	int
platforms	int[]
asset_url	varchar

Платформы для рекламных объявлений:

0 - web

1 - android

2 - ios

Рекламное объявление должно содержать хотя бы одну платформу.

Эндпоинты:

1. **GET /summaries/**

Возвращает в формате JSON сокращенную информацию по кампаниям: ид, имя, статус, кол-во рекламных объявлений.

Должен поддерживать пагинацию, фильтры по имени и статусу, сортировки по имени, статусу, кол-ву рекламных объявлений.

2. **GET /campaign/:id**

Возвращает в формате JSON модель кампании со списком ее рекламных объявлений

3. **POST /campaign/**

Создание кампании. Возвращает в формате JSON модель новой кампании.

4. **PUT /campaign/:id**

Изменение кампании. Возвращает в формате JSON обновленную модель кампании.

5. **DELETE /campaign/:id**

Удаление кампании

6. **GET /ad/:id**

Возвращает в формате JSON модель рекламного объявления.

7. **POST /ad/**

Создание рекламного объявления. Возвращает в формате JSON модель нового рекламного объявления. При создании рекламное объявление должно быть связано с существующей кампанией.

8. **PUT /ad/:id**

Изменение рекламного объявления. Возвращает в формате JSON обновленную модель рекламного объявления.

9. **DELETE /ad/:id**

Удаление рекламного объявления

**Требования к заданию:**

1. Разместить код на github, bitbucket, etc.
2. Написать readme с инструкцией по запуску приложения.
3. Не использовать hibernate (или другие ORM фреймворки).
4. Использовать in memory базу данных (или docker compose с imag'ами для БД и API).
5. Все request / response body должны быть валидным JSON.
6. Использовать стандартные http коды ошибок