

## Запрещается

1. Использование Django;
2. Использовать только функциональный стиль программирования;
3. Писать весь код только в 1 файле.

## Рекомендуется

1. `requirements.txt` - список зависимостей Python;
2. `README.md` - описание запуска приложения, зависимостей;
3. `Dockerfile` - если вы владеете технологией контейнеризации Docker и хотите продемонстрировать свои знания в этом направлении;
4. Шаблоны конфигурации.

## Задание

Необходимо разработать backend WEB-приложения, управляющего файловым хранилищем. В качестве файлового хранилища может выступать файловая система компьютера. Приложение должно взаимодействовать с базой данных, в которой хранится информация обо всех файлах хранилища (название, расширение, путь хранения и др). Одна запись из базы данных должна соответствовать 1 файлу из хранилища.

## Состав записи базы данных информации о файле

1. Имя (*изменяемое*) - произвольная не пустая строка без расширения;
2. Расширение (*не изменяемое*) - строка с обозначением расширения файла (пример: `.txt`);
3. Размер (*не изменяемое*) - целочисленное значение размера файла в байтах;
4. Полный путь расположения (*изменяемое*) - определяет иерархию родительских директорий расположения файла (пример: `/root-folder/my-folder/`);
5. Дата создания (*не изменяемое*) - строка даты и времени в ISO-формате;
6. Дата изменения (*изменяемое*) - опциональная строка даты и времени в ISO-формате;
7. Комментарий к файлу (*изменяемое*) - произвольная опциональная строка с комментарием;

## Обязательные функции для реализации

1. Получение списка информации о всех файлах. В этом случае должна производиться выборка всех записей из базы данных.
2. Получение информации по конкретному файлу.
3. Загрузка нового файла в файловое хранилище и создание записи в базе данных, соответствующей этому файлу.

4. Удаление записи о файле из базы данных и удаление самого файла из файлового хранилища.

Ответы от WEB-приложения должны быть в формате JSON или XML (на выбор).

Пример ответа на запрос информации по конкретному объекту:

```
{
  "name": "my-file",
  "extension": ".txt",
  "size": 10325,
  "path": "/root-folder/my-storage/",
  "created_at": "2020-05-12T13:48:10.034677",
  "updated_at": null,
  "comment": "Мой первый текстовый файл"
}
```

## Необязательные функции для реализации

1. Поиск всех файлов, которые расположены по определенной части пути. Эта функция должна выдавать информацию обо всех файлах, в пути которых есть передаваемый путь для поиска.
2. Скачивание существующего файла из хранилища. Данная функция позволяет скачать (получить) из WEB-приложения файл на локальный компьютер.
3. Изменение информации о файле в базе данных. Изменять возможно часть информации о файле, которая доступна для редактирования (название, путь расположения, комментарий). При изменении названия или пути расположения соответствующие изменения должны быть отражены и в хранилище. Любое изменение должно актуализировать дату изменения записи в базе данных.
4. Синхронизация файлового хранилища и базы данных информации о файлах. В этом случае должна быть запущена процедура актуализации состояния базы данных на основе состояния хранилища, т.е. если файл был удален из хранилища, то из базы данных должна быть удалена соответствующая запись; если файл был создан (добавлен) вручную в хранилище, в базе данных должна появиться соответствующая запись о нем.