**Задание для Python developer - IT Bootcamp Test Task Python**

**Компактный веб-сервис, позволяющий работать с авторами и книгами.**

**Для запуска проекта необходимо:**

- Установить на машине Python 3.11

- Установить и запустить на локальной машине БД PostgreSQL – 16 с параметрами:

'NAME': 'author\_book',

'USER': 'postgres',

'PASSWORD': 'root',

'HOST': 'localhost',

'PORT': '5432'

- Перенести проект на локальную машину.

- Запускаем терминал Пайтон

- Убеждаемся, что мы не находимся в виртуальном окружении другого проекта.

- При необходимости выходим из него: .venv\Scripts\deactivate.bat

- pip list – убеждаемся, что poetry установлен для всего Пайтон.

- Если poetry не установлен, устанавливаем его: pip install poetry.

- настраиваем Poetry так, чтобы он создавал виртуальное окружение virtualenvs не в cache-директории C:\Users\<username>\AppData\Local\pypoetry\Cache\virtualenvs, а в корневом каталоге проекта. Для этого установим переменной virtualenvs.in-project значение true:

poetry config virtualenvs.in-project true

- переходим в папку проекта на локальной машине

- python -m poetry shell - создаем виртуальное окружение проекта

- poetry install – устанавливаем зависимости из pyproject.toml

- python manage.py makemigrations – подготавливаем файл миграций

- python manage.py migrate – выполняем миграцию БД.

- python manage.py runserver – запускаем сервер

- <http://127.0.0.1:8000/> - переходим на стартовую страницу проекта

**1. Схема базы данных проекта**

**Модель «Author»**

Поля:

- id SERIAL PRIMARY KEY (целочисленный, первичный ключ, автоинкрементируемый)

- name VARCHAR (35) NOT NULL UNIQUE (строковый, не пустой, уникальный (35))

**Модель «Book»**

- id SERIAL PRIMARY KEY (целочисленный, первичный ключ, автоинкрементируемый)

- title VARCHAR (35) NOT NULL UNIQUE (строковый, не пустой, уникальный (35))

Связи:

- fk\_book\_to\_author многие к одному с моделью «Author»

**2. Описание функционала проекта**

1. Добавление нового автора

2. Редактирование имени автора

3. Поиск всех книг автора

4. Просмотр списка всех авторов

5. Просмотр данных автора по id

6. Удаление автора

7. Удаление всех авторов

8. Добавление книги

9. Редактирование данных книги

10. Просмотр списка всех книг

11. Просмотр книги по id

12. Поиск книги по названию

13. Поиск книг по имени автора

14. Удаление книги

15. Удаление всех книг

**3. Описание эндпоинтов**

1. author/{author\_id}

- method= GET / name='AuthorView'

- Описание: Просмотр данных автора

- Входные данные: ID Автора

- Работа с данными: получение данных из таблицы Author

- Запись в таблицу: Нет

2. author/create

- method=POST / name='author\_create'

- Описание: Добавление нового автора

- Входные данные: Нет.

- Работа с данными: создание новой записи в таблице Author с введенными данными

- Запись таблицы: Author

3. author/edit/{author\_id}

- method=POST / name=’author\_update’

- Описание: Редактирование имени автора

- Входные данные: ID автора

- Работа с данными: изменение записи в таблице Author с введенными данными

- Запись таблицы: Author

4. author/del/{author\_id}

- method=POST / name='author\_del'

- Описание: Удаление автора

- Входные данные: ID автора

- Работа с данными: удаление записи из таблицы Author с веденными данными

- Запись таблицы: Author

5. author/del/all

- method=POST / name='author\_all\_del'

- Описание: Удаление всех авторов

- Входные данные: Нет

- Работа с данными: удаление всех записей из таблицы Author

- Запись таблицы: Author

6. author/find/{author\_id}

- method=GET / name='author\_find\_book'

- Описание: Поиск всех книг автора

- Входные данные: ID автора

- Работа с данными: получение записей из таблицы Book

- Запись таблицы: Нет

7. author/

- method=GET / name='AuthorAllView’

- Описание: Вывод списка всех авторов

- Входные данные: Нет

- Работа с данными: получение записей из таблицы Author

- Запись таблицы: Нет.

8. book/{book\_id}

- method= GET / name='BookView'

- Описание: Просмотр данных книги

- Входные данные: ID Книги

- Работа с данными: получение данных из таблицы Book и Author

- Запись в таблицу: Нет

9. book/create

- method=POST / name='book\_create'

- Описание: Добавление новой книги

- Входные данные: Нет.

- Работа с данными: создание новой записи в таблице Book с введенными данными

- Запись таблицы: Book

10. book/edit/{book\_id}

- method=POST / name=’book\_update’

- Описание: Редактирование книги

- Входные данные: ID книги

- Работа с данными: изменение записи в таблице Book с введенными данными

- Запись таблицы: Book

11. book/del/{book\_id}

- method=POST / name='book\_del'

- Описание: Удаление книги

- Входные данные: ID книги

- Работа с данными: удаление записи из таблицы Book с веденными данными

- Запись таблицы: Book

12. book/del/all

- method=POST / name='book\_all\_del'

- Описание: Удаление всех книг

- Входные данные: Нет

- Работа с данными: удаление всех записей из таблицы Book

- Запись таблицы: Book

13. book/

- method=GET / name='BookAllView’

- Описание: Вывод списка всех книг

- Входные данные: Нет

- Работа с данными: получение записей из таблицы Book

- Запись таблицы: Нет.

14.1 book/find\_author/

- method=GET / name='book\_find\_author'

- Описание: Поиск всех книг по имени автора

- Входные данные: В веб интерфейсе после ввода имени автора, формируется строка запроса: books/find\_author/?find\_text={имя автора}

- Работа с данными: получение записей из таблицы Book, Author

- Запись таблицы: Нет

14.2 book/find\_author/

- method=GET / name='book\_find\_book'

- Описание: Поиск книги по названию

- Входные данные: В веб интерфейсе после ввода названия книги формируется строка запроса: books/find\_book/?find\_text={название книги}

- Работа с данными: получение записей из таблицы Book, Author

- Запись таблицы: Нет