МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ, СВЯЗИ И МАССОВЫХ КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ

Ордена Трудового Красного Знамени

федеральное государственное бюджетное образовательное учреждение

высшего образования

«Московский технический университет связи и информатики»

Изображение выглядит как эмблема, логотип, символ, Торговая марка

Автоматически созданное описание

**ОТЧЕТ ПО ЛАБОРАТОРНЫМ РАБОТАМ**

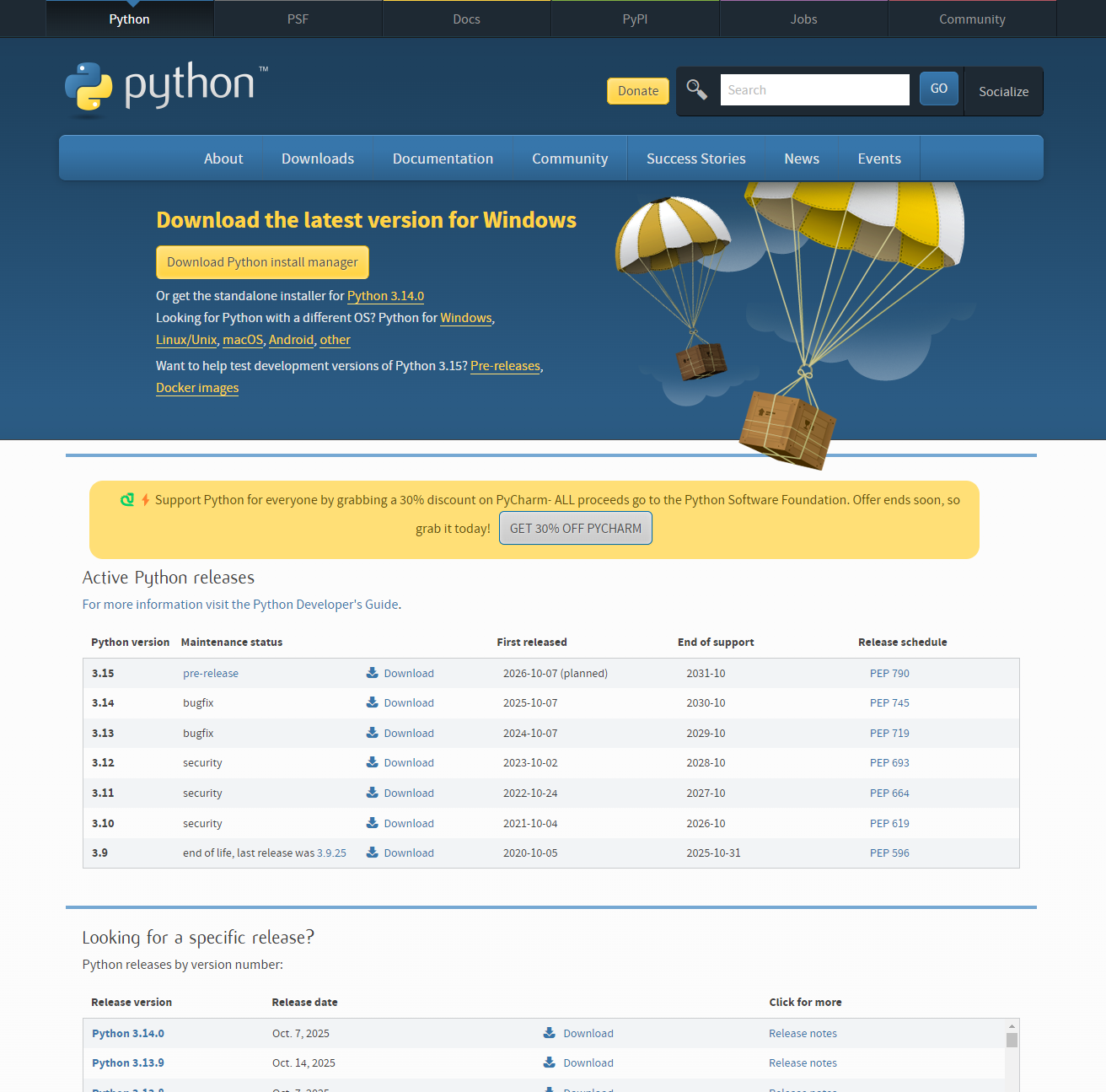
|  |
| --- |
| Выполнил:  Студент гр. № 2301: Лукинов Д.Д.  Проверил: |
| Старший преподаватель: Сергеев А.Е. |

Оценка: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Дата\_\_\_\_\_\_\_

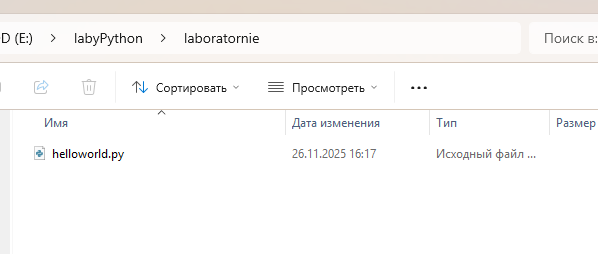
**Москва 2025 г.**

**Лабораторная работа № 1**

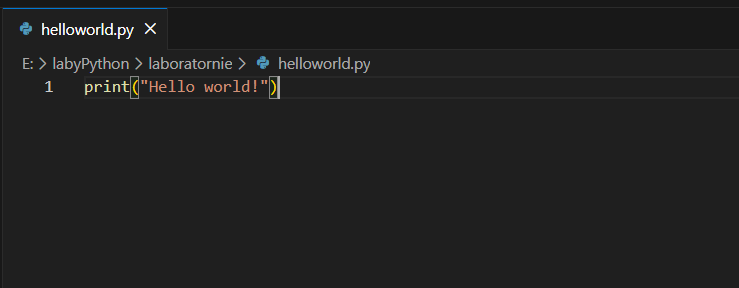
1. Установим последнюю версию интерпретатора Python. Ссылка: https://www.python.org/downloads/

****

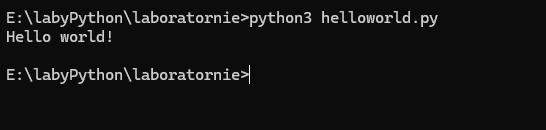
2. Создадим директорию, а также файл и назовём его helloworld.py

****

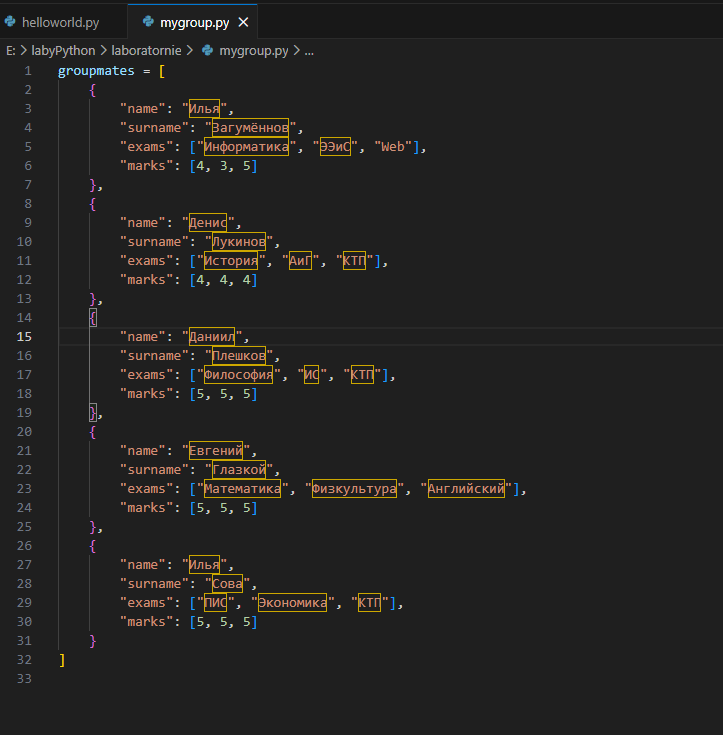
3. Вставим следующий код



4. Запустим скрипт

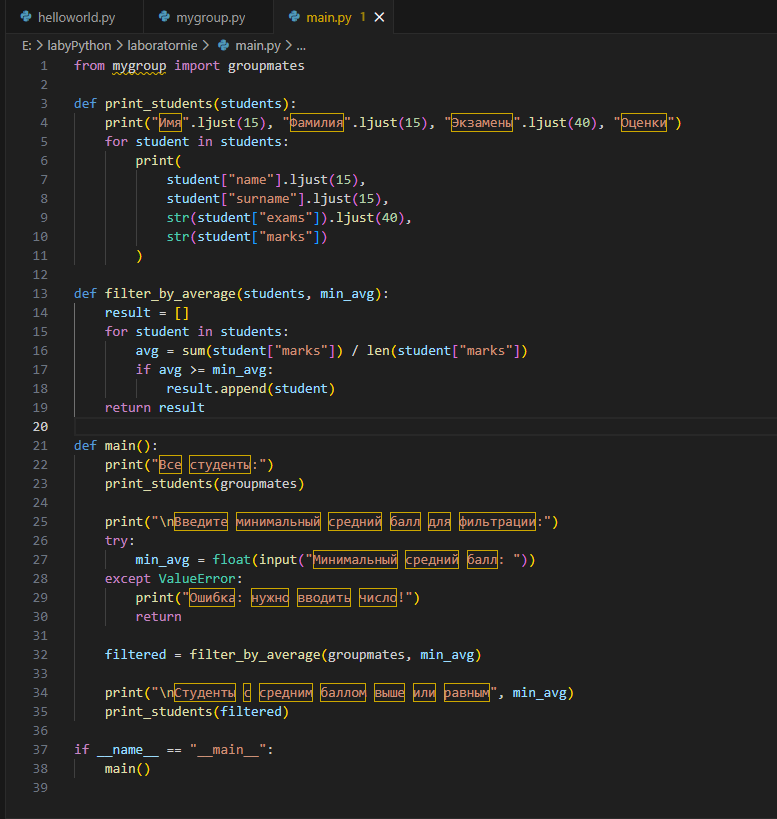


5. Добавим еще один скрипт и добавим туда данные о студентах

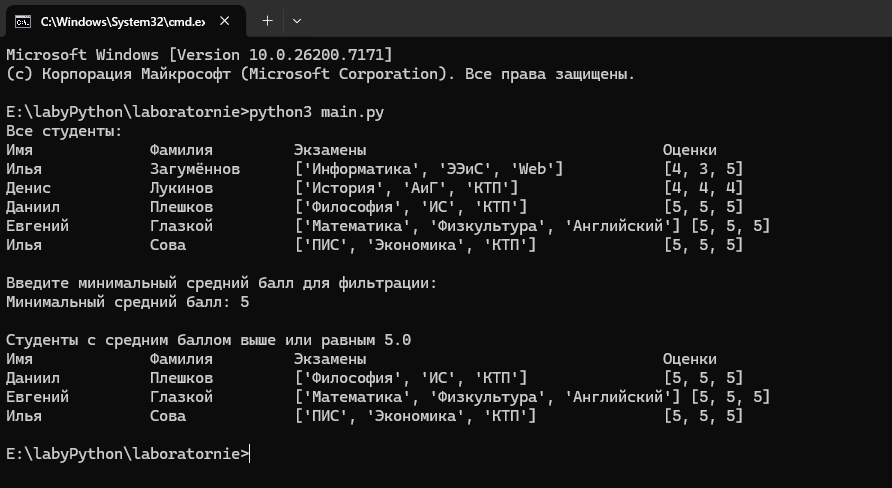


6. Реализуем функцию фильтрации студентов по средней оценке.

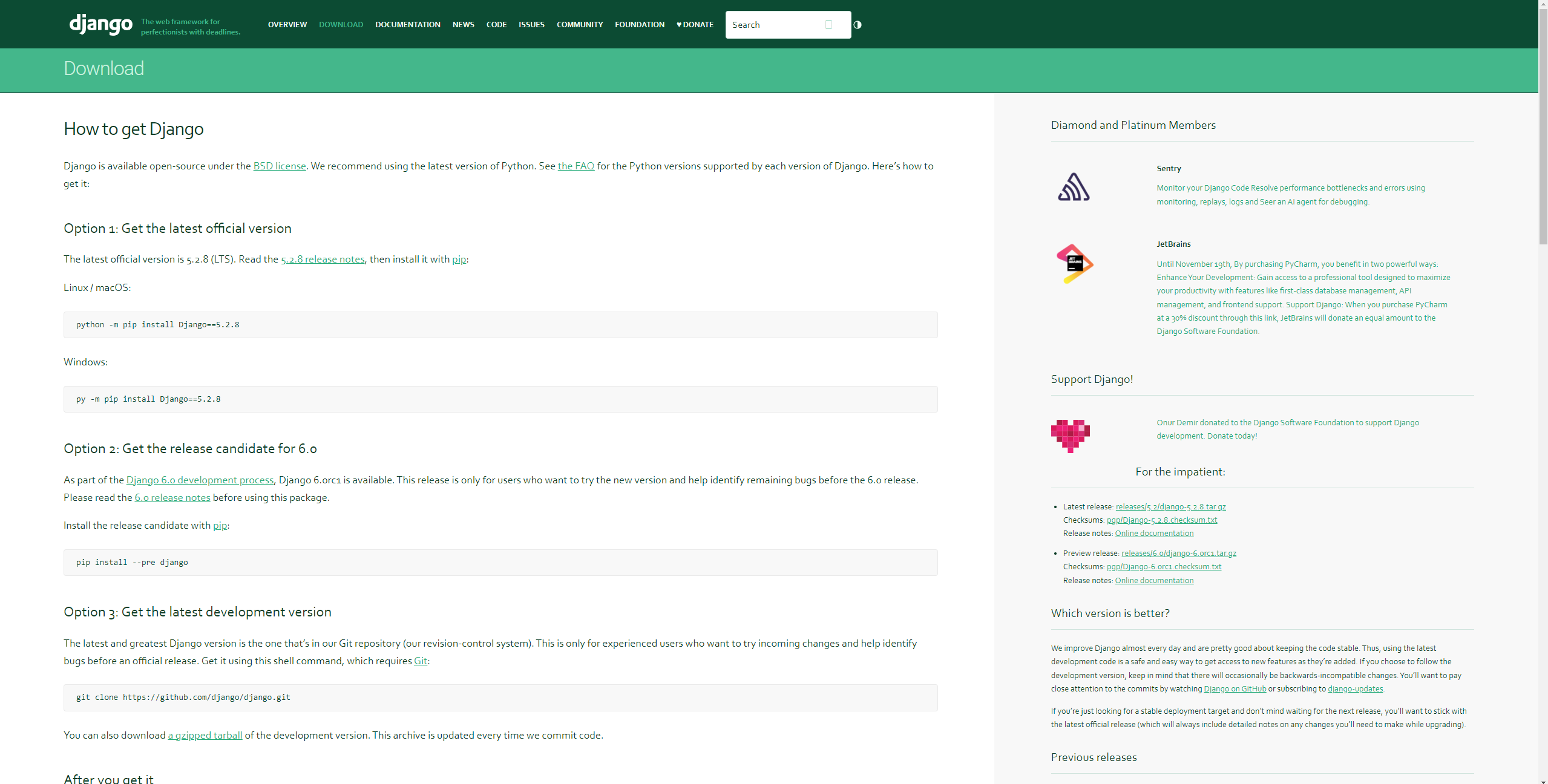
Для этого создадим еще один скрипт main.py.



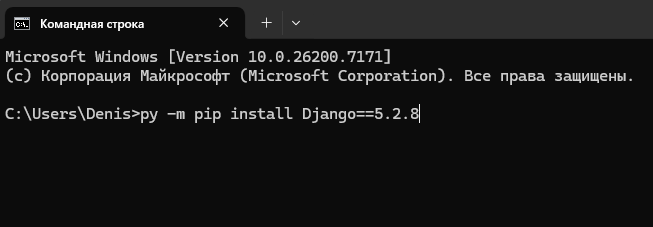
7. Запустим и посмотрим результат



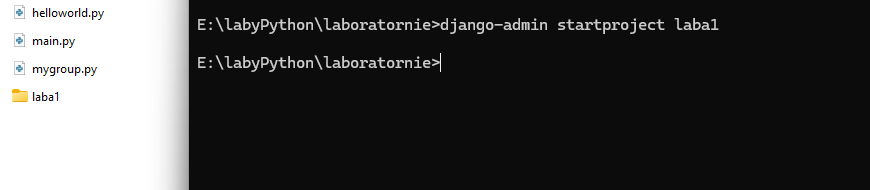
8. Установим Django. Перейдём на сайт: <https://www.djangoproject.com/download/>



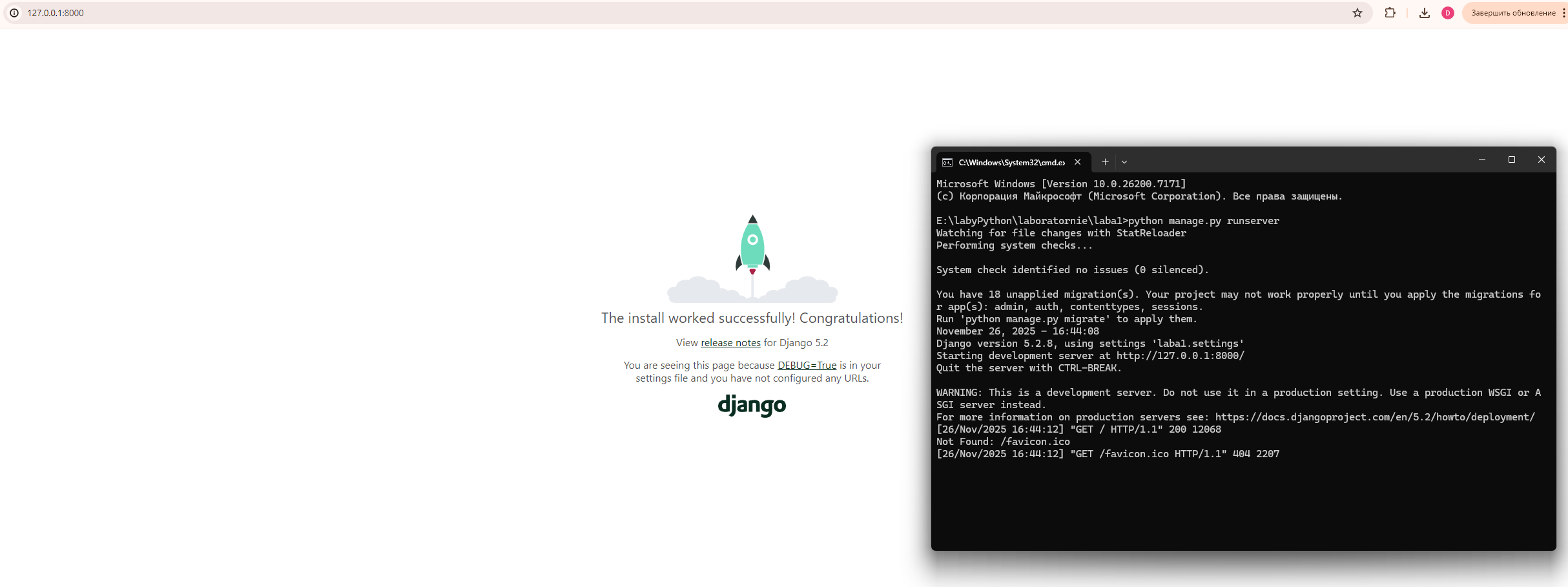
9. Установим последнюю версию



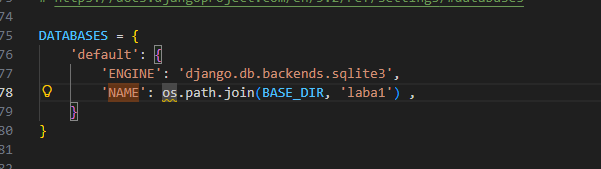
10. Создадим проект Django



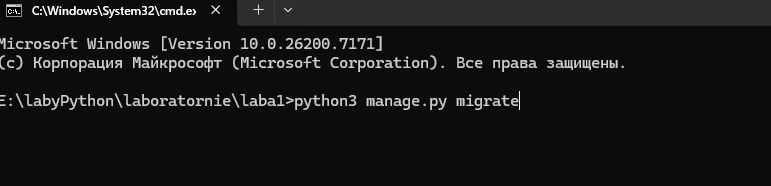
11. Запустим проект, чтобы убедиться в правильности установки



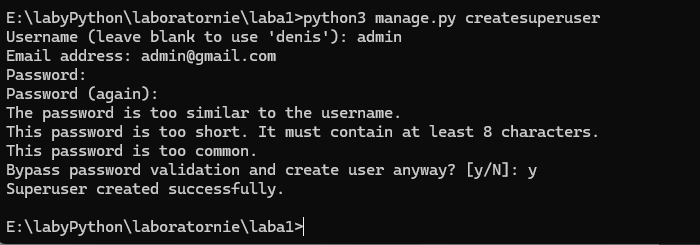
12. Изменим параметр NAME в файле settings.py



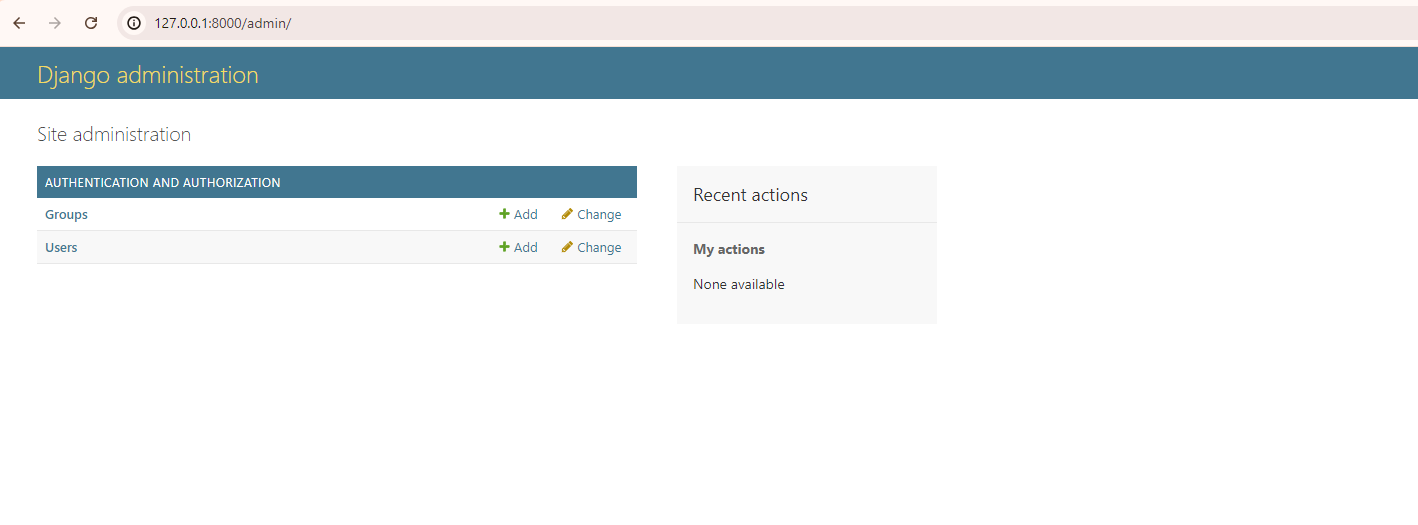
13. Создадим базу данных



14. Создадим суперпользователя

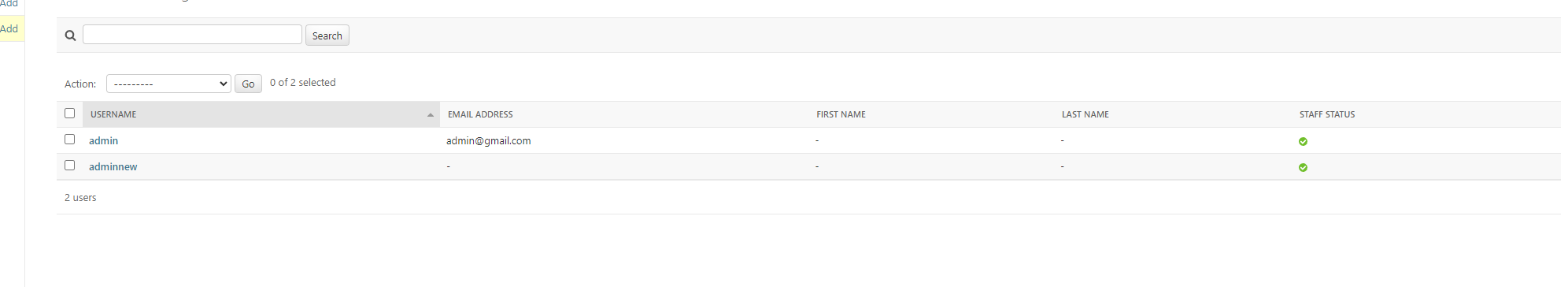


15. Запустим локальный сервер и перейдём по адресу: http://127.0.0.1:8000/admin/

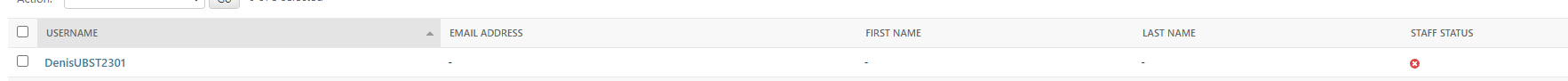


16. Создадим суперпользователя через админку

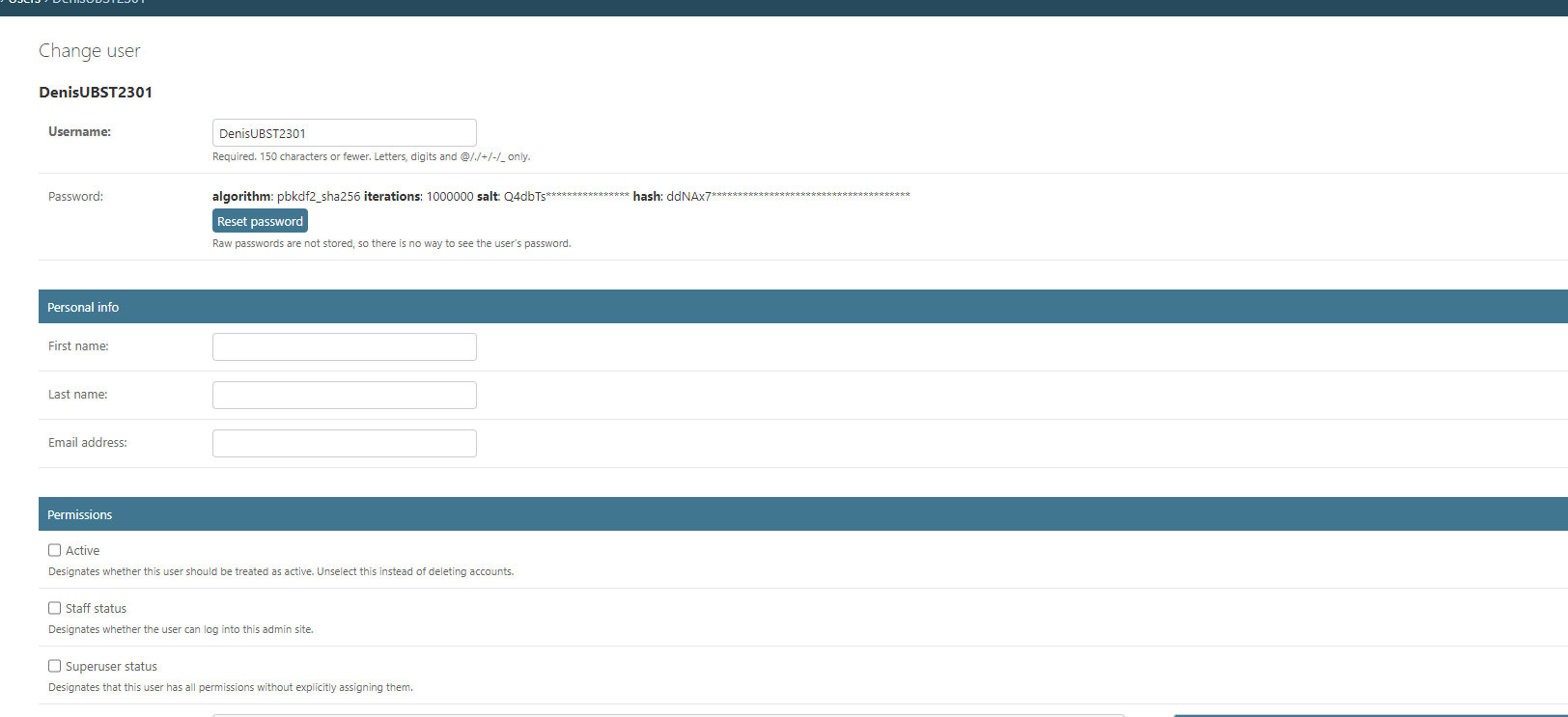




17. Создадим обычного пользователя



18. Забаним пользователя через админку

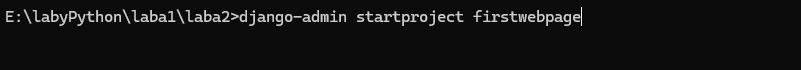


**Лабораторная работа № 2**

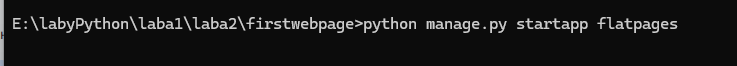
1. Создадим директорию laba2



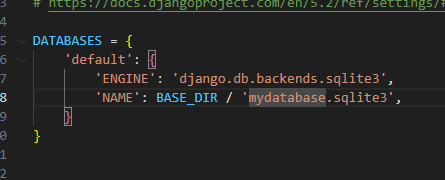
2. Создадим новый проект Django



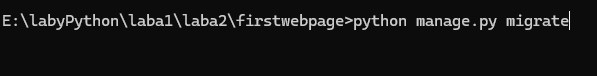
3. Создадим новое приложение



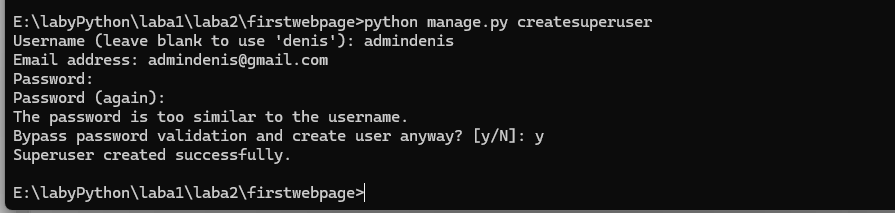
4. Изменим название БД



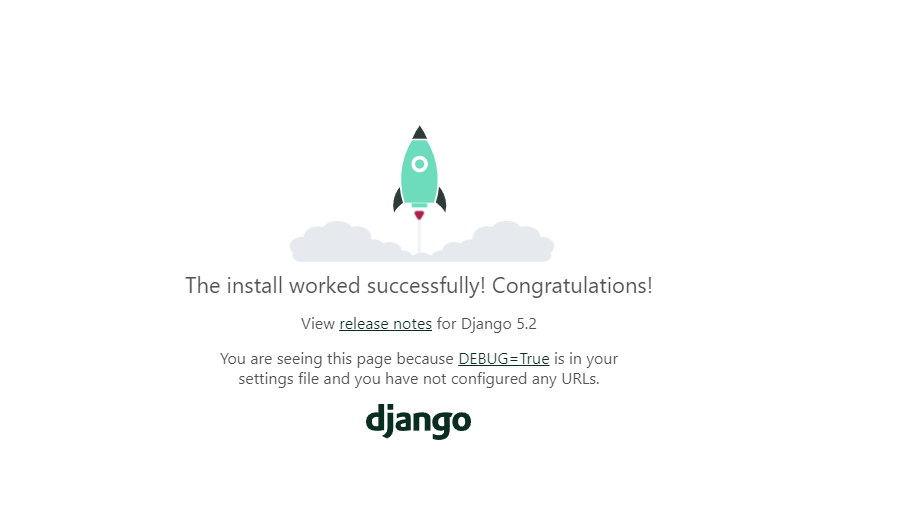
5. Создадим таблицы



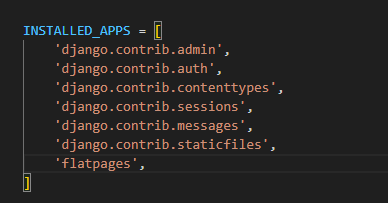
6. Создадим суперюзера



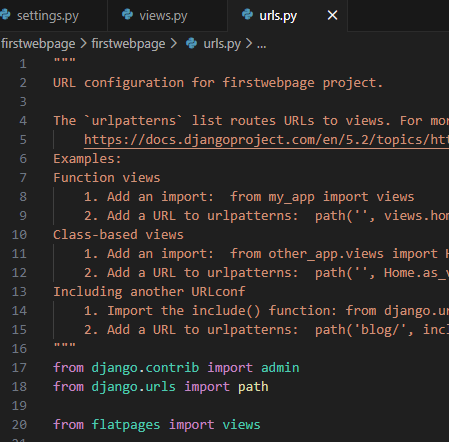
7. Запустим проект для проверки



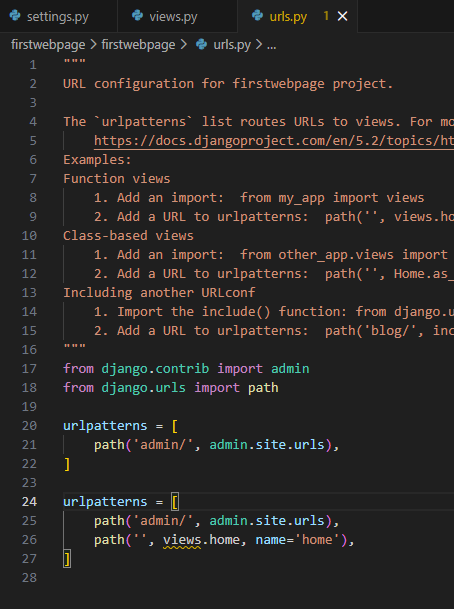
8. Добавим flatpages в installed\_apps



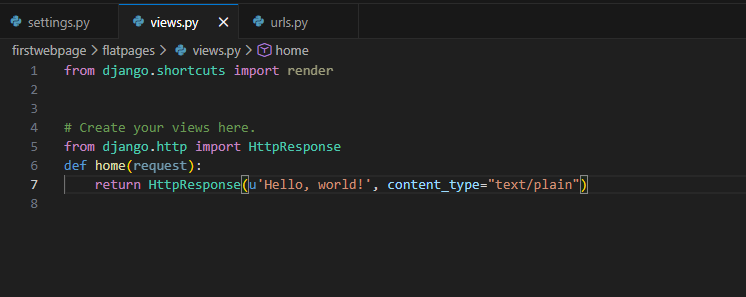
9. Импортируем views



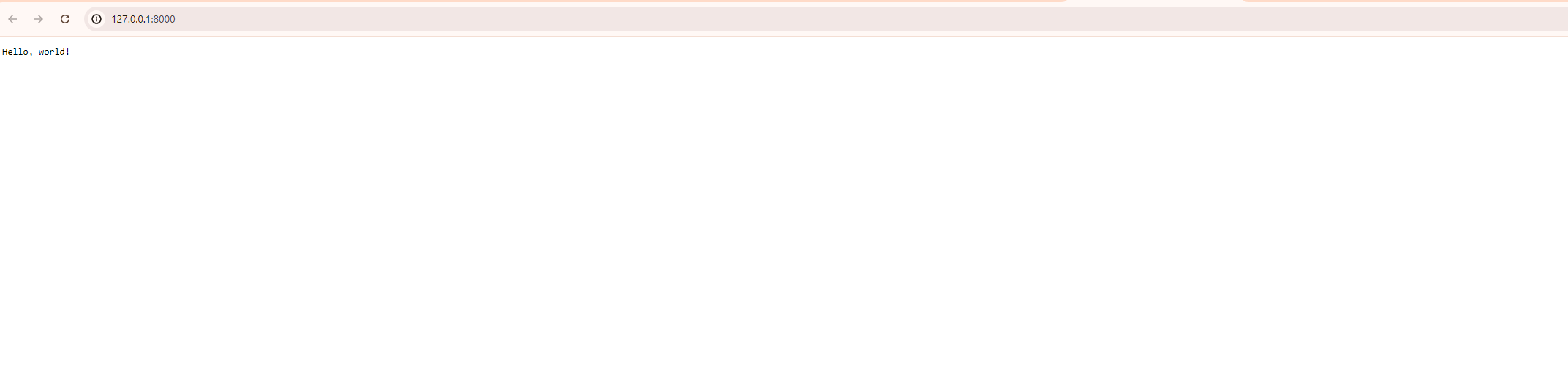
10. Создадим новый адрес в urls.py



11. Создадим функцию в файле views.py

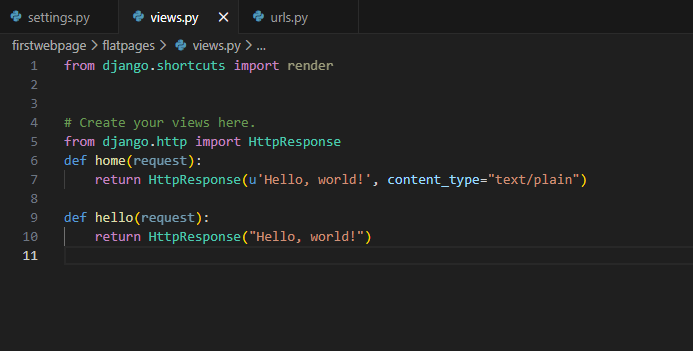


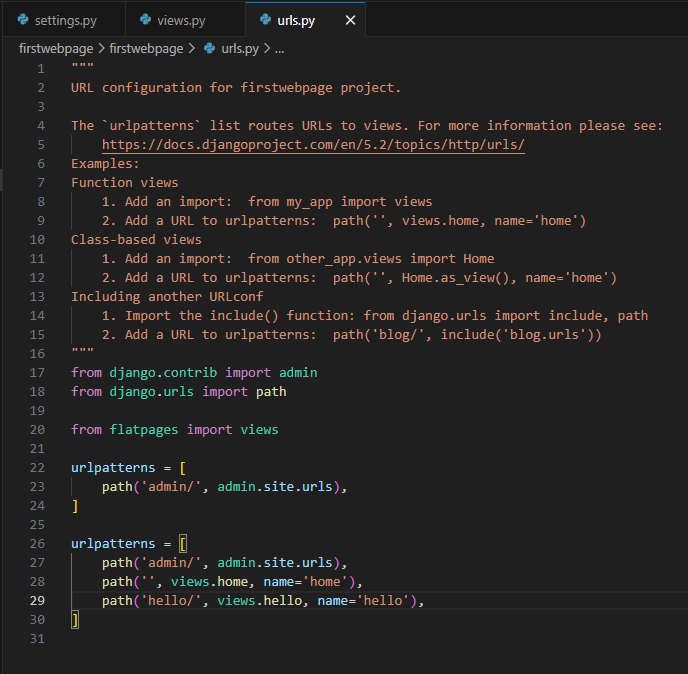
12. Проверим результат



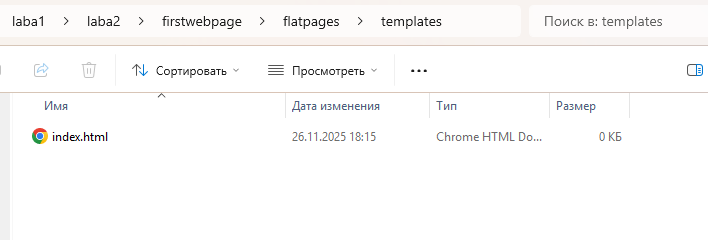
13. Сделаем, чтобы по адресу /hello возвращался тот же текст.

Если тип явно не указывать, HttpResponse сам выставляет Content-Type: text/html.

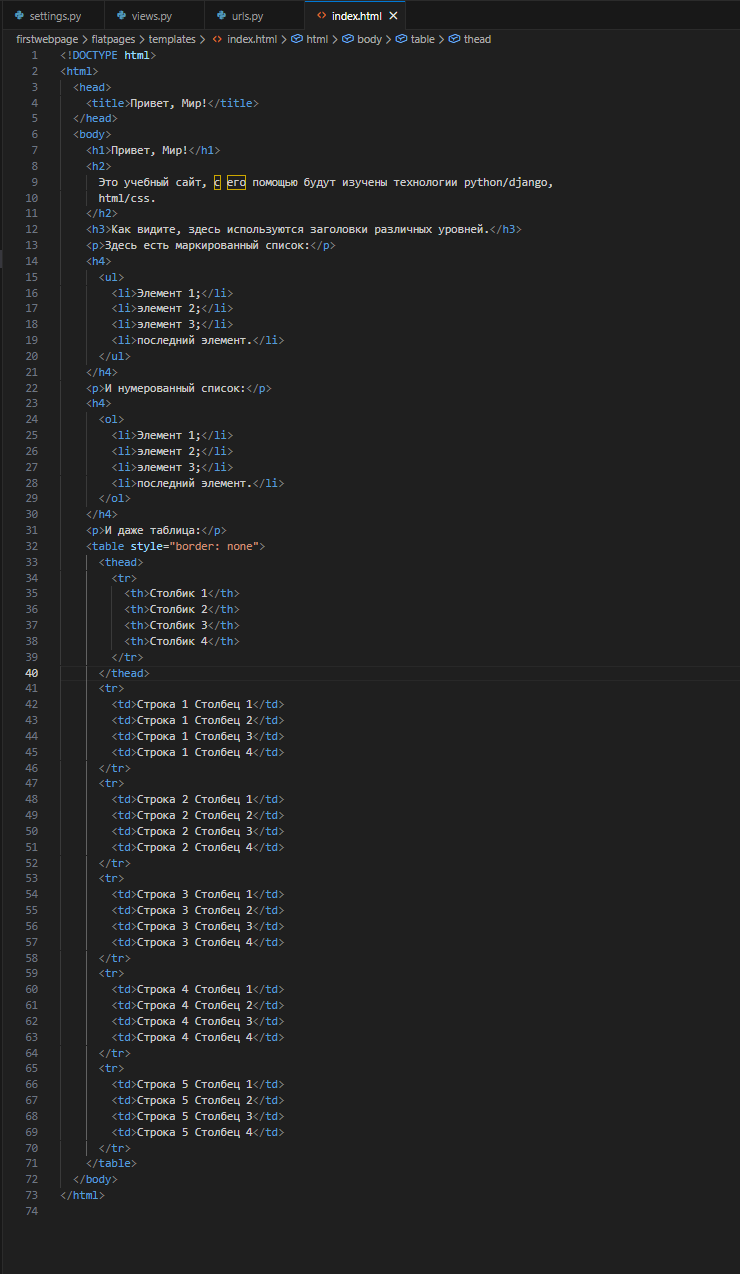




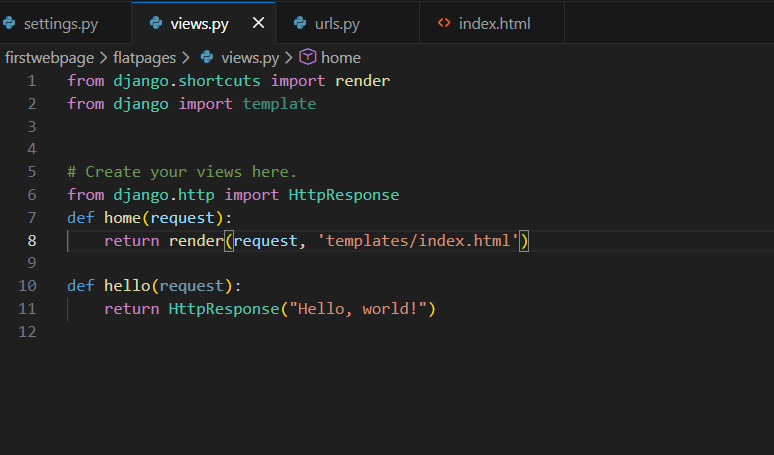
14. Создадим директорию **templates** и в ней создадим файл **index.html.**



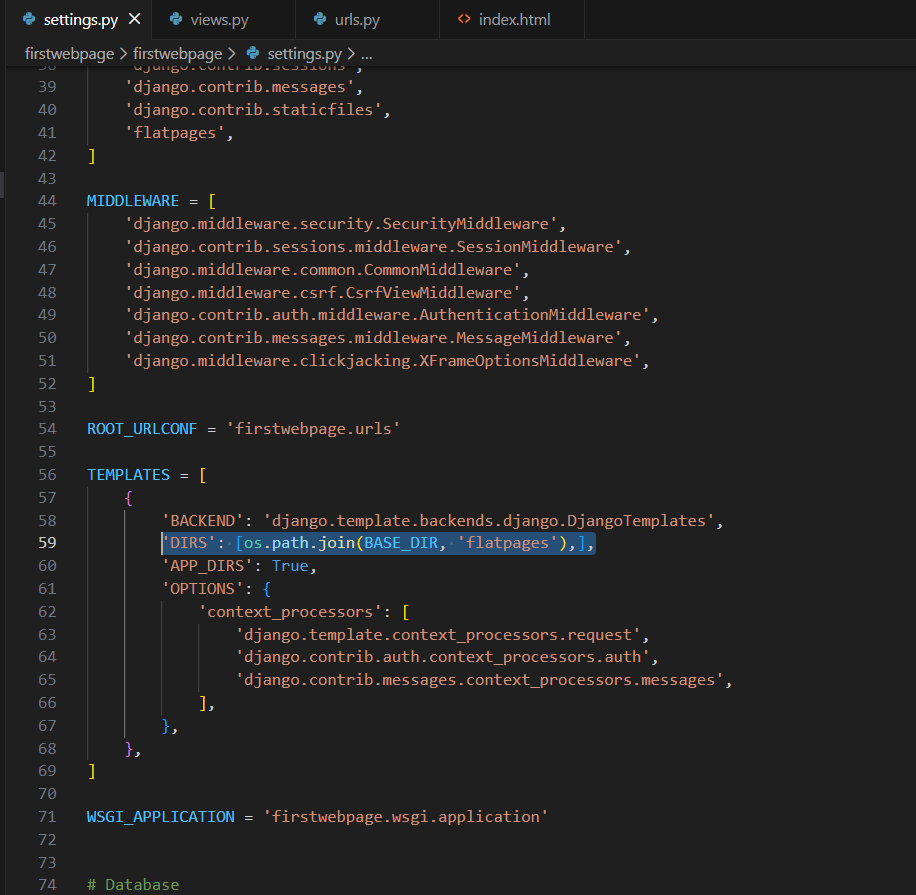
15. Добавим в этот файл следующий код



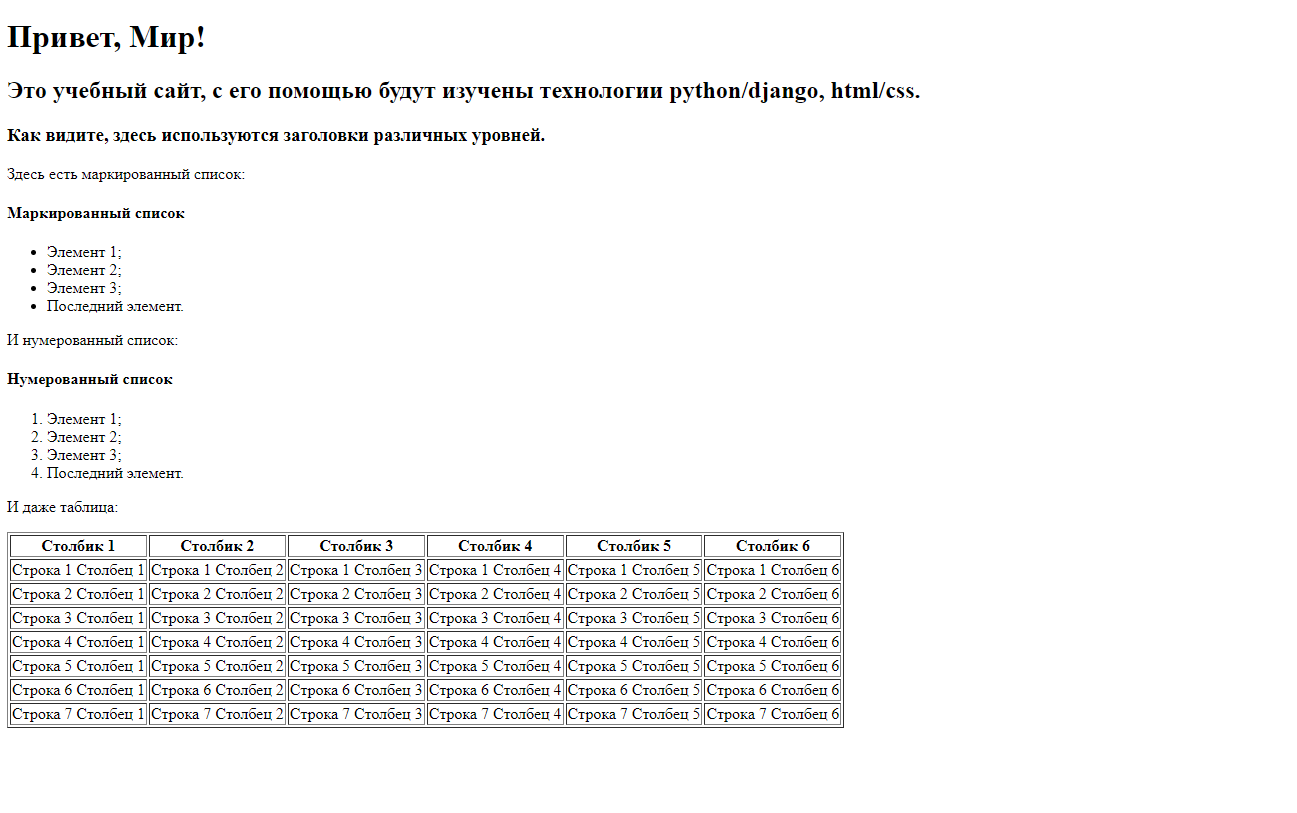
16. Изменим файл views.py



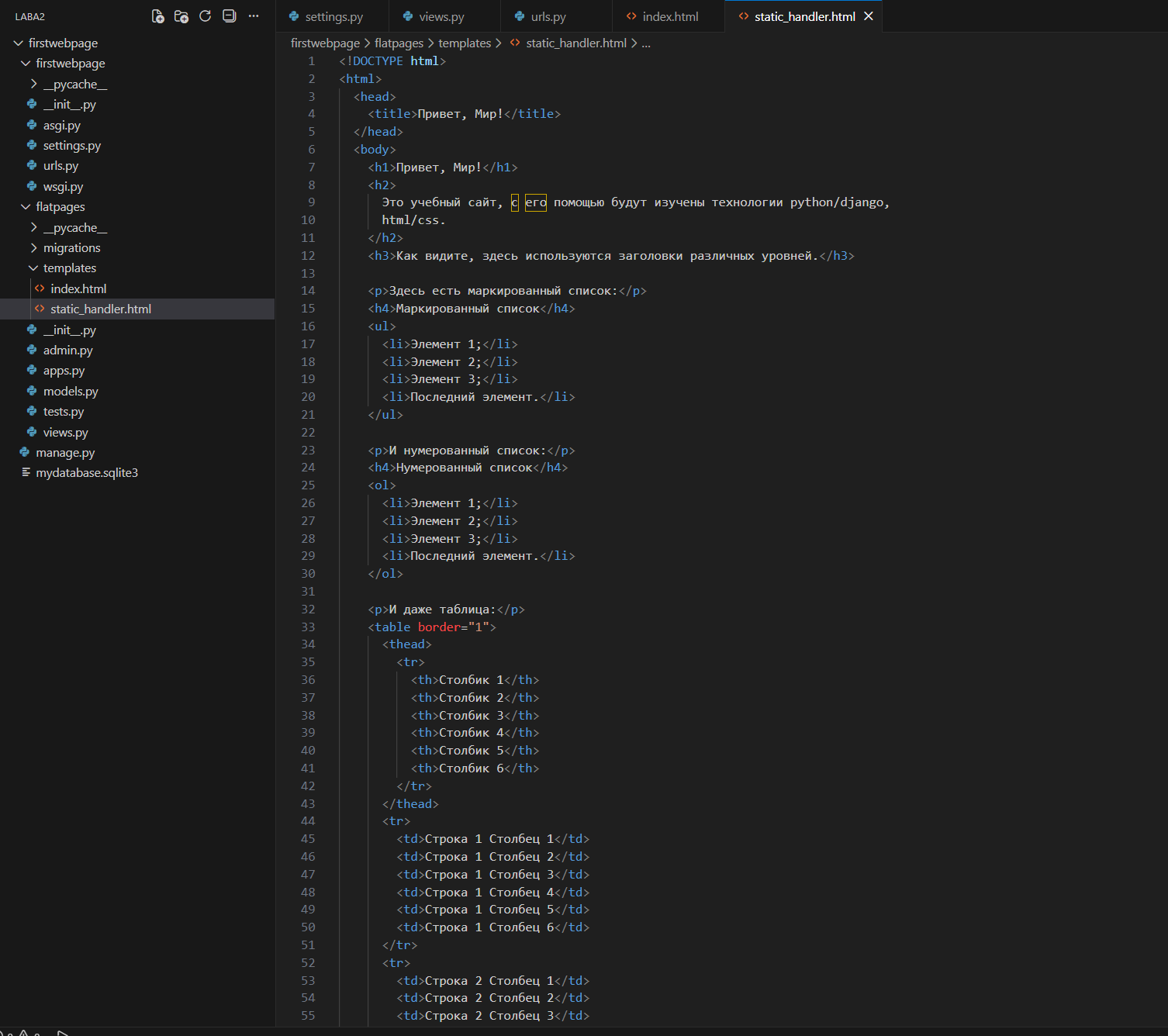
17. В файл settings.py добавим изменения



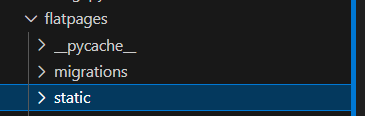
18. Добавим 2 строки, 2 столбца, границы таблицы, изменим подзаголовки.



19. Создадим шаблон static\_handler.html



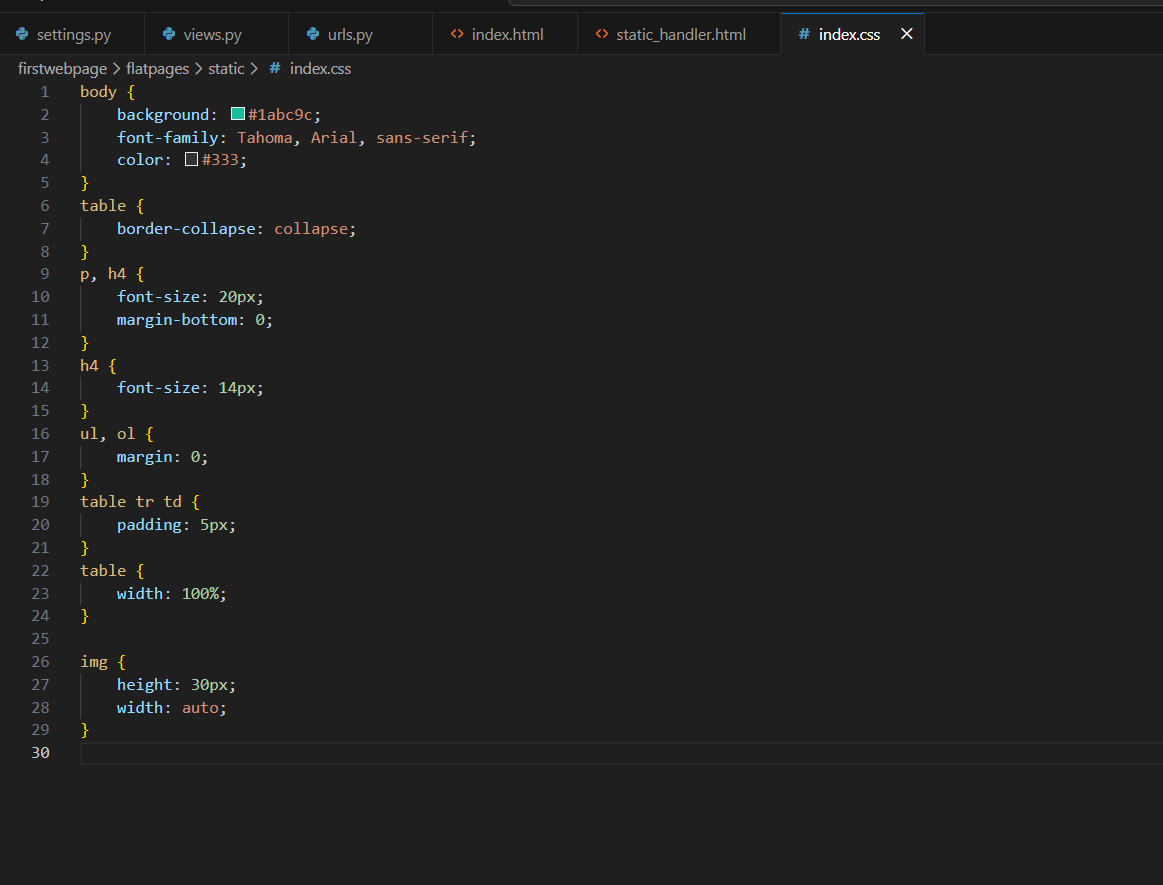
20. Создадим папку static



21. Создадим файл



22. Добавим в файл код



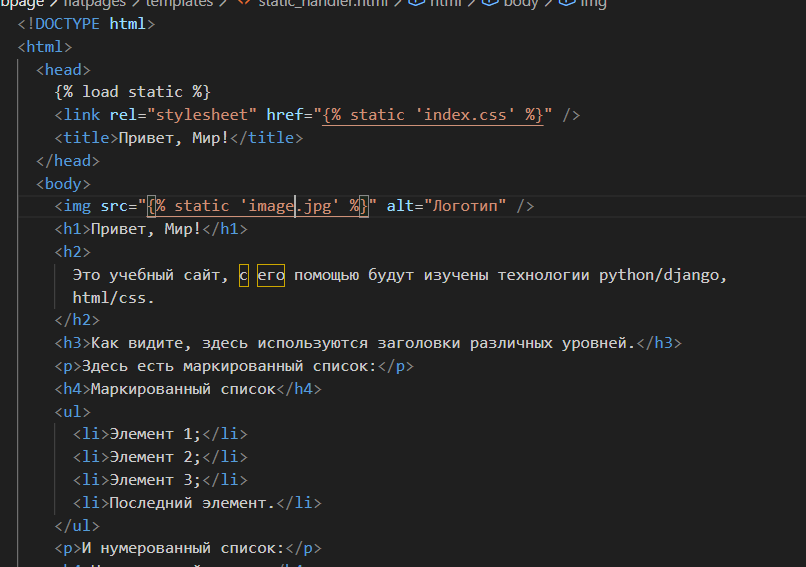
23. Добавим строку в файл

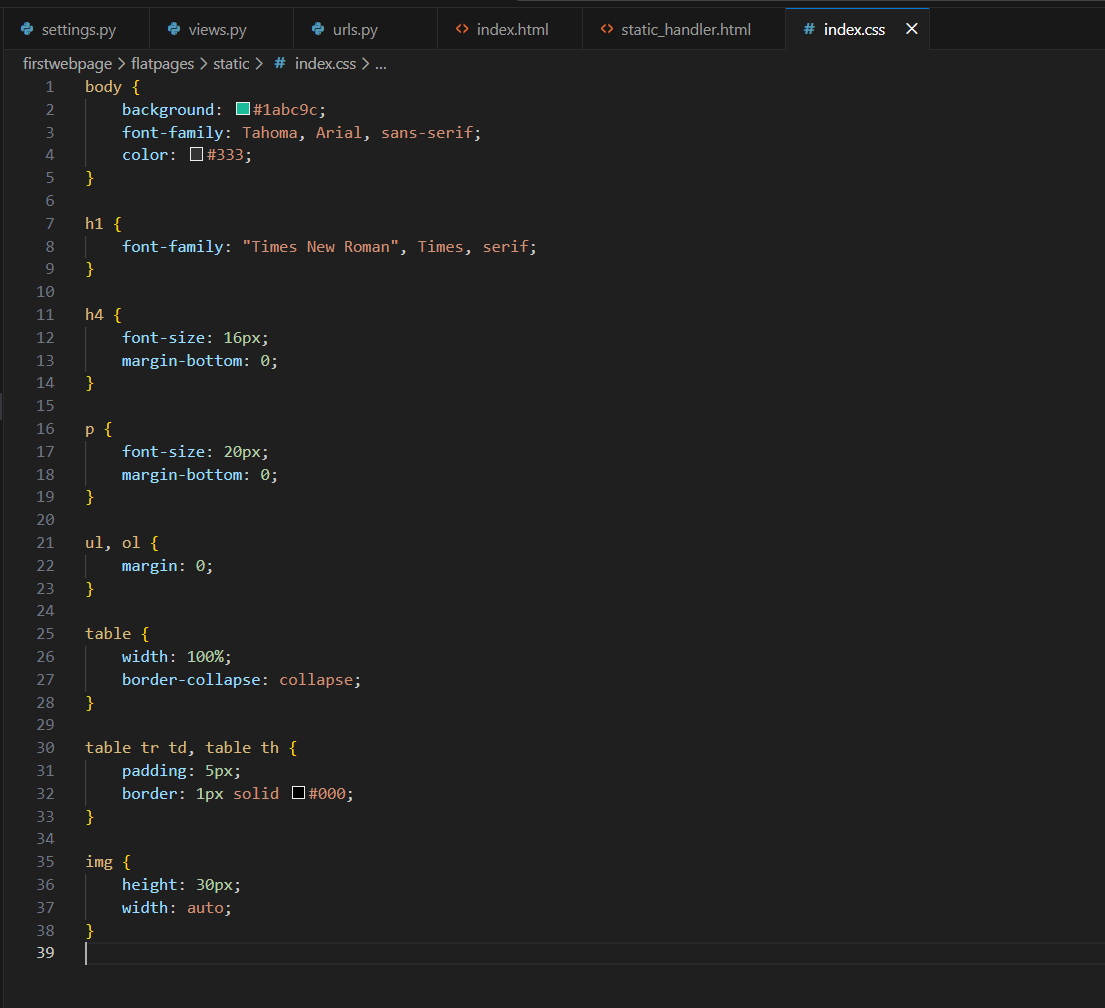


24. Результат вывода



25.Установим для заголовка уровня шрифт с засечками, добавим картинку и сделаем ее высотой 30 px, изменим размер шрифта для подзаголовка 4-ого уровня, сделаем ширину таблицы на 100% экрана.





26. Результат

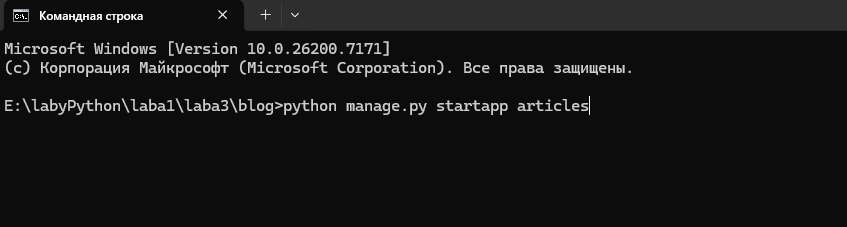


**Лабораторная работа № 3**

1. Создадим новый проект

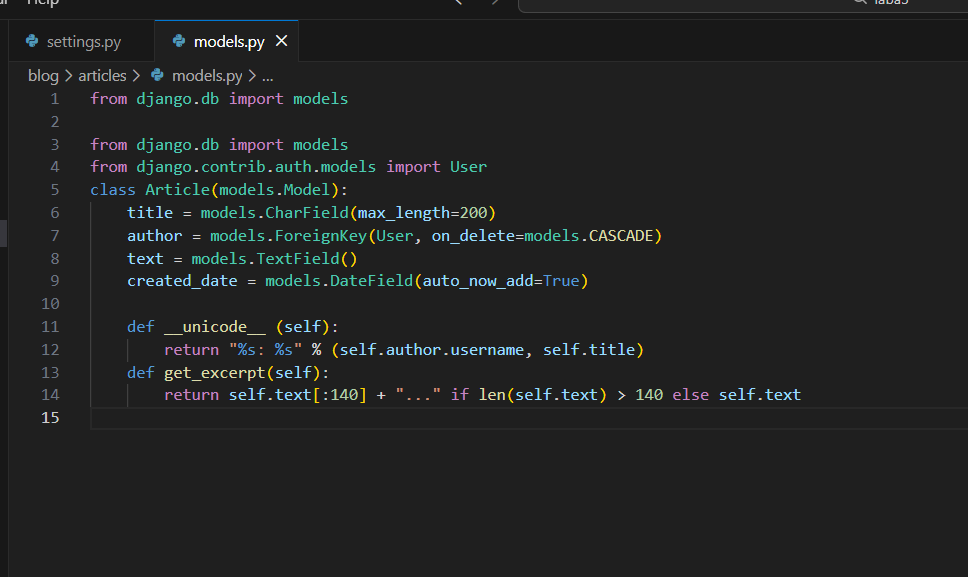


2. Перейдём в папку blog и выполним команду

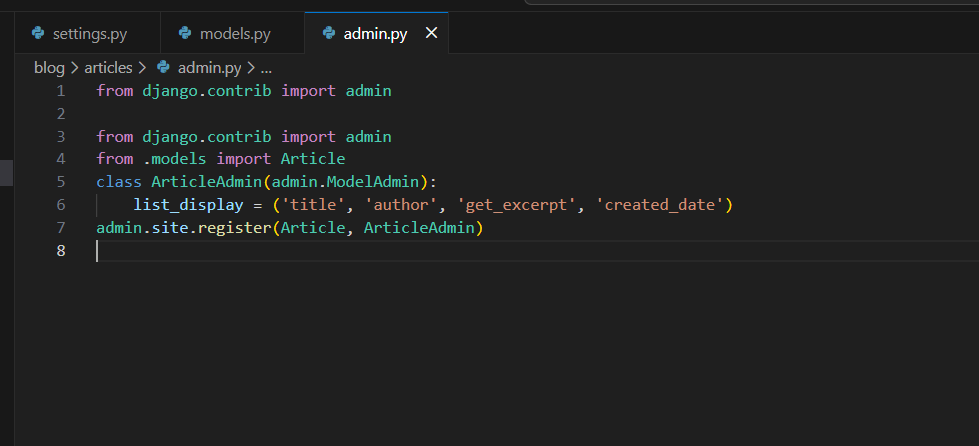


3. Выполним настройку проекта как в предыдущих работах

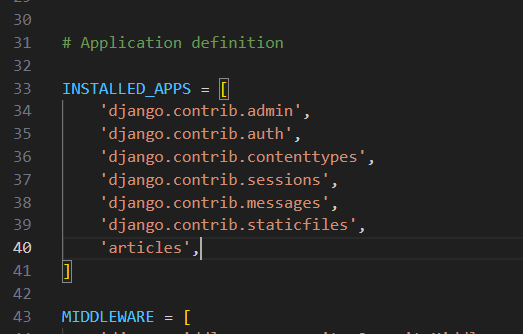
4. Зайдём в models.py и впишем код



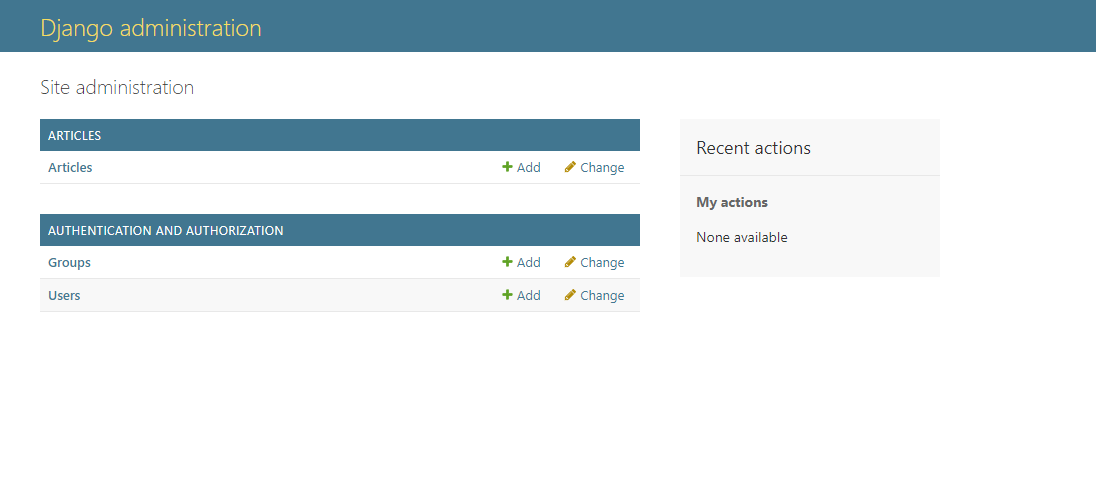
5. Откроем admin,py и впишем следующий код



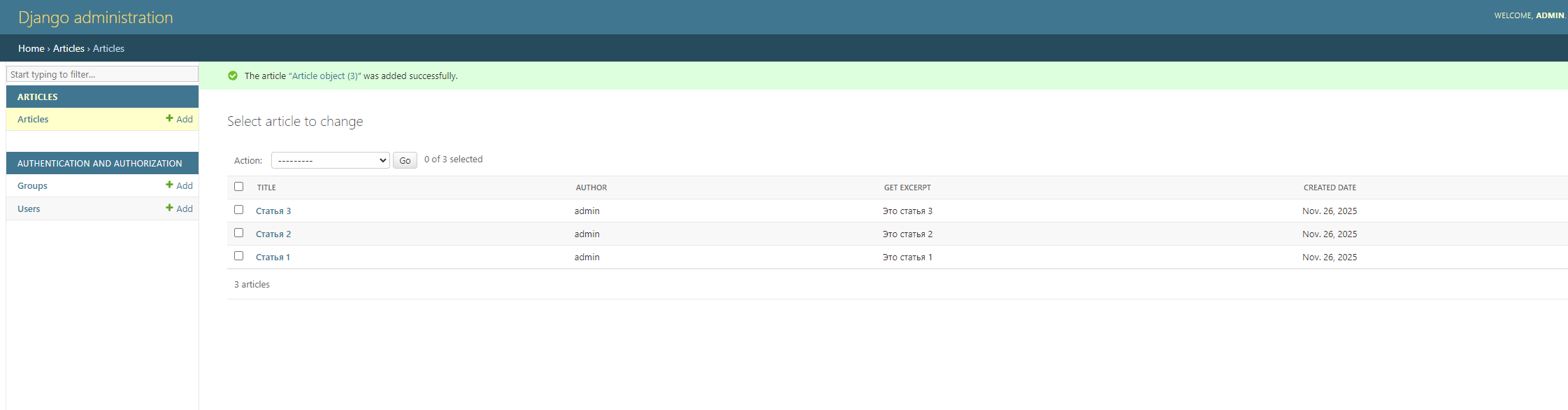
6. Добавим в settings.py приложение



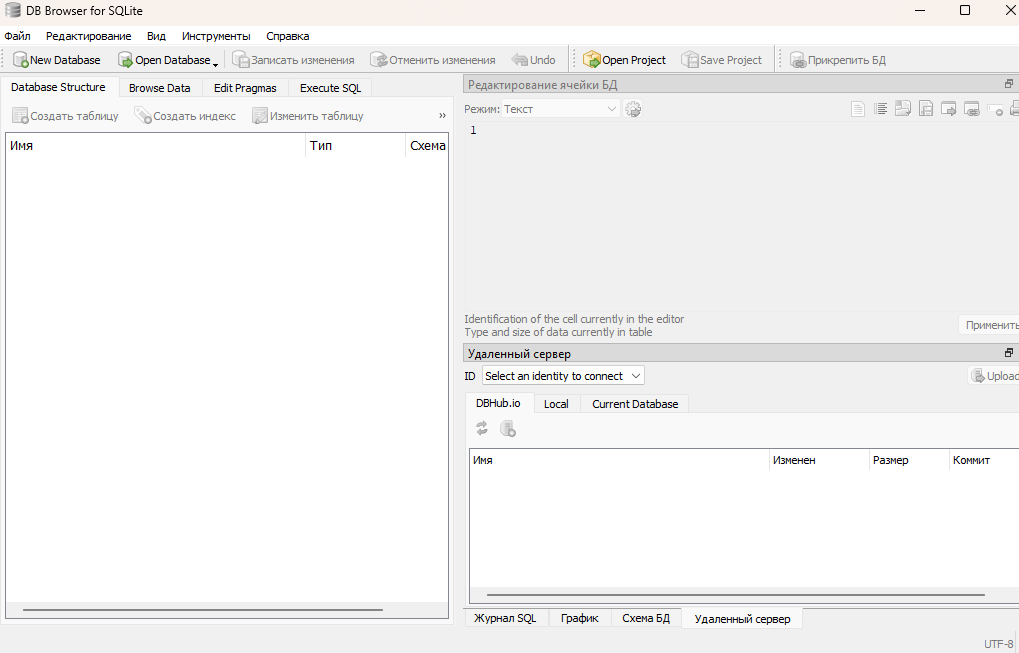
7. Результат



8. Создадим 3 статьи



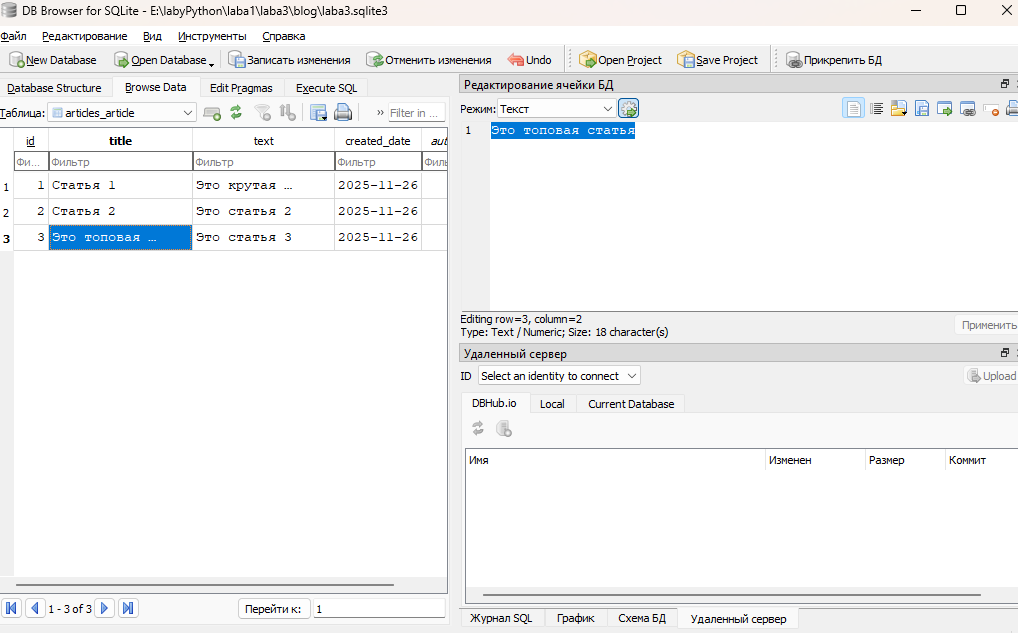
9. Скачаем программу



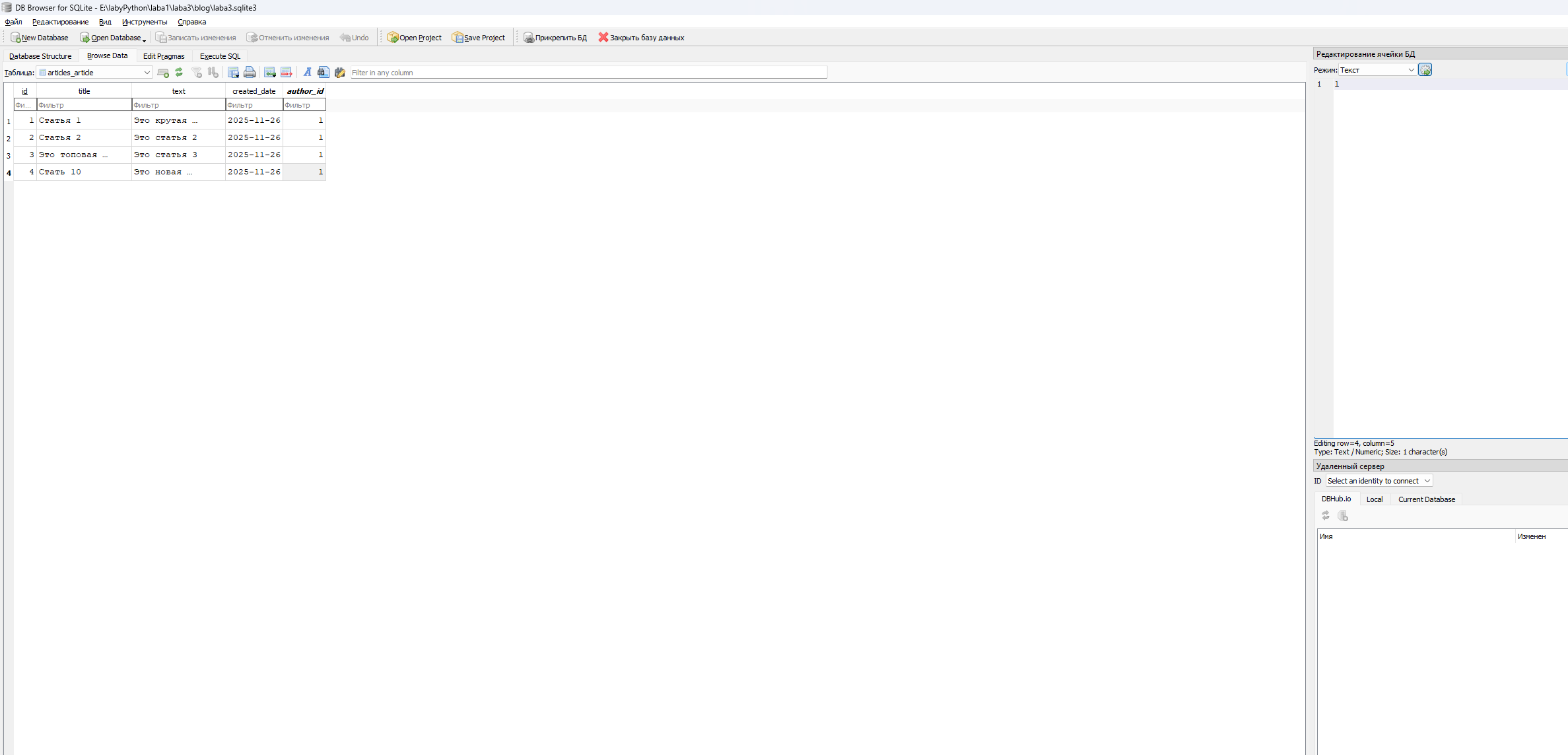
10. Откроем базу и увидим записи



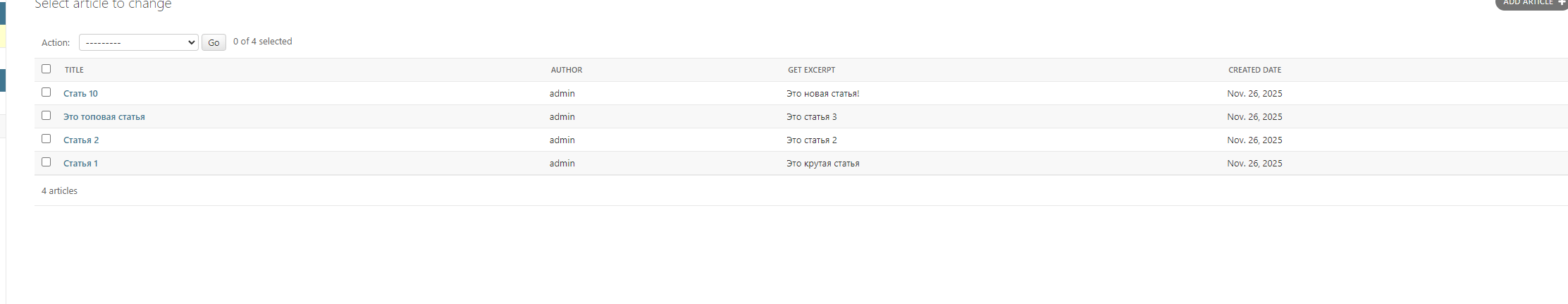
11. Изменим данные статей



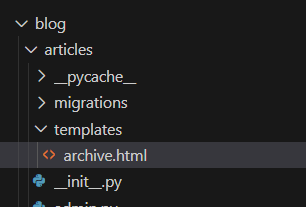
12. Добавим новую статью



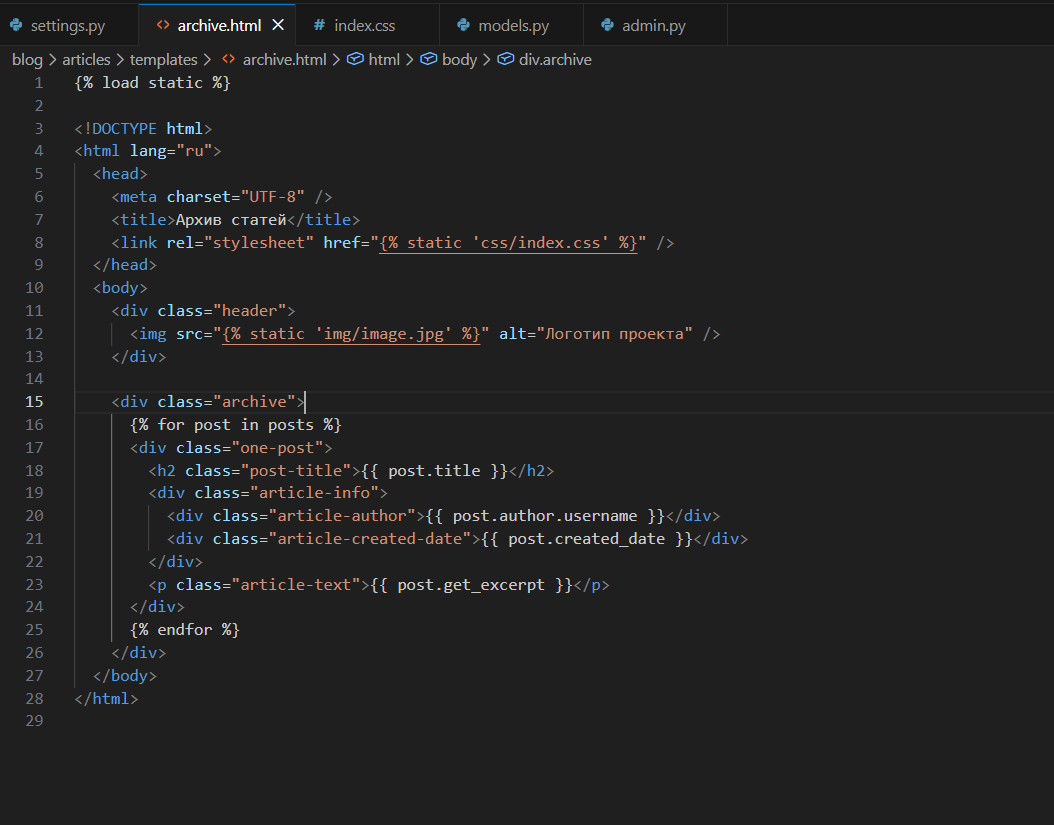
13. Результат



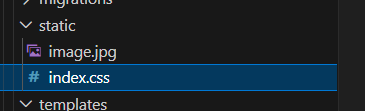
14. Создадим новую папку и файл



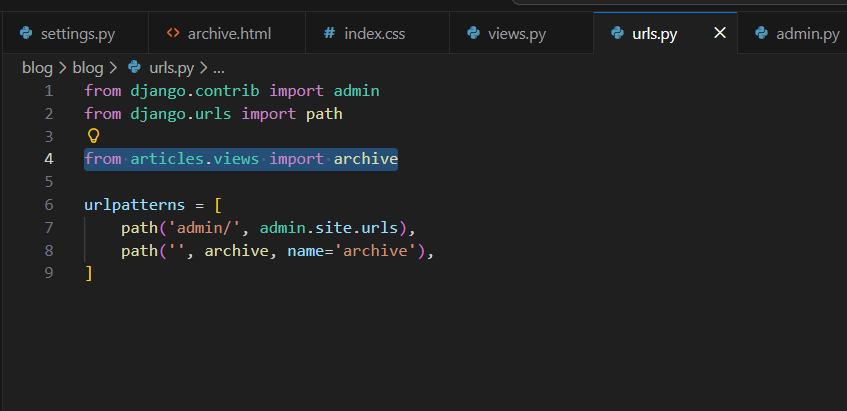
15. Впишем код



16. Добавим папку static



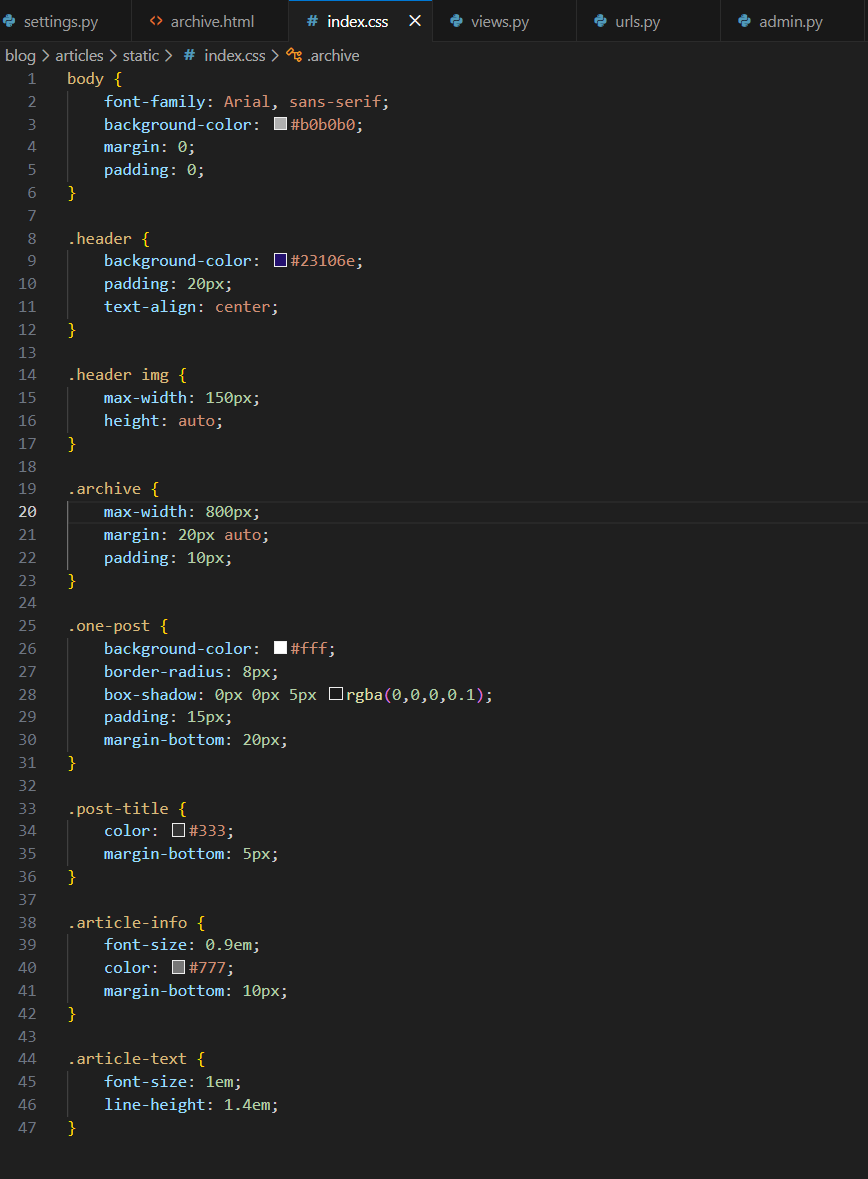
17. Настроим urls.py



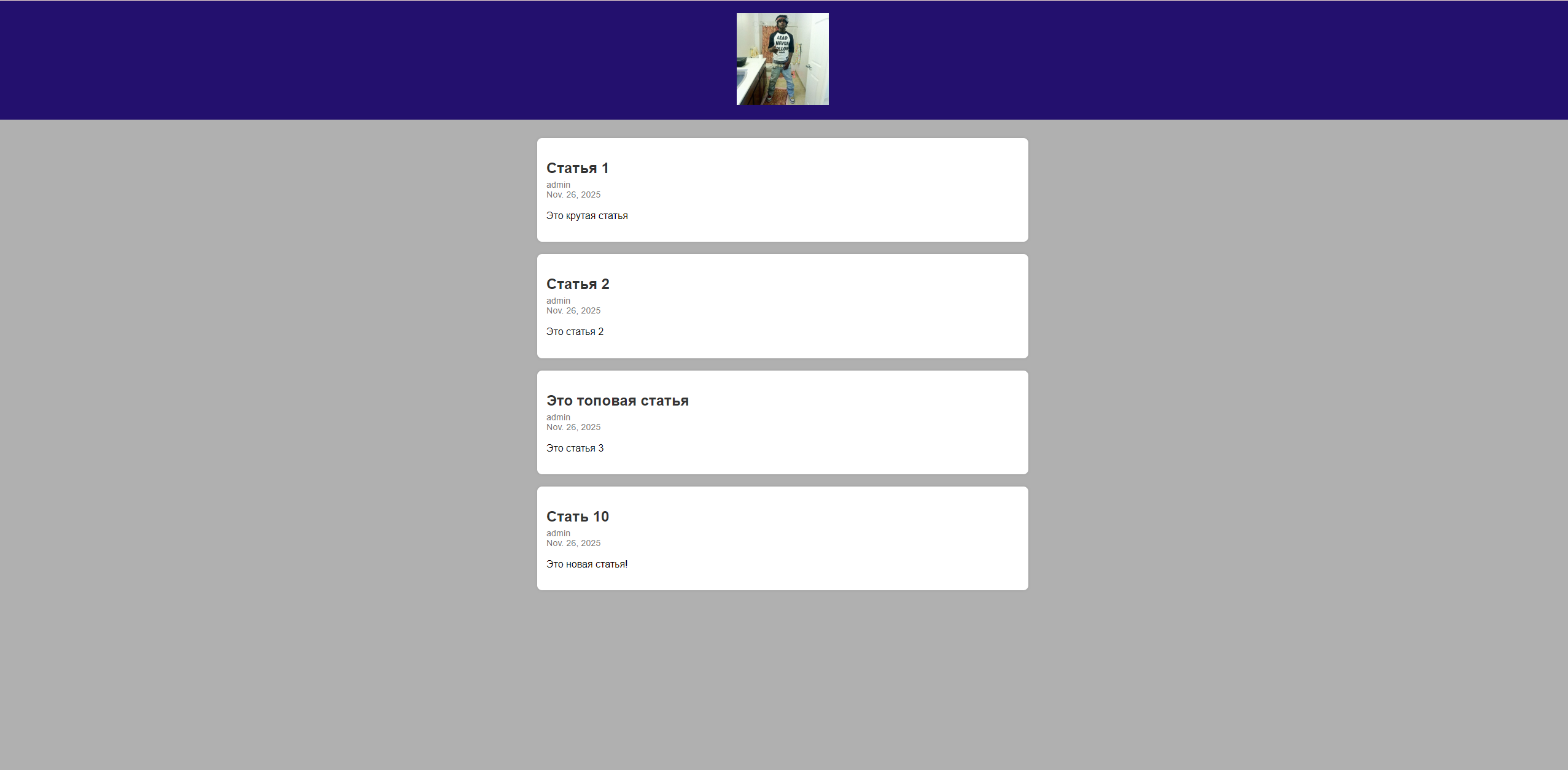
18. Настроим views.py



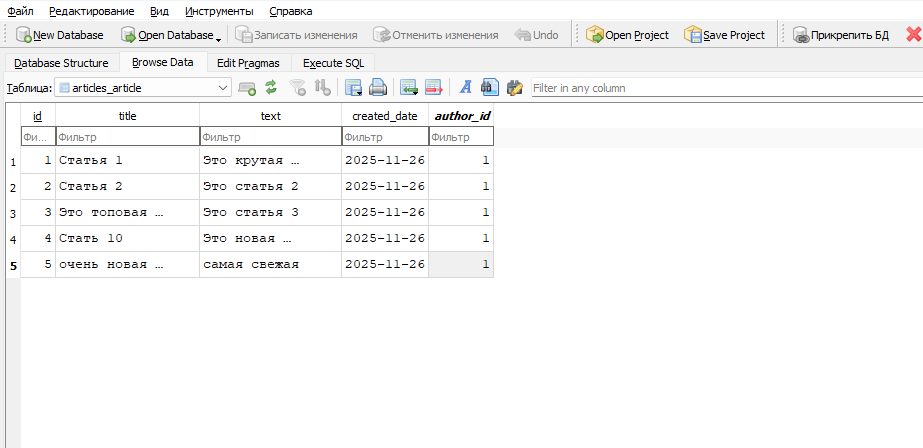
19. Заполним файл css



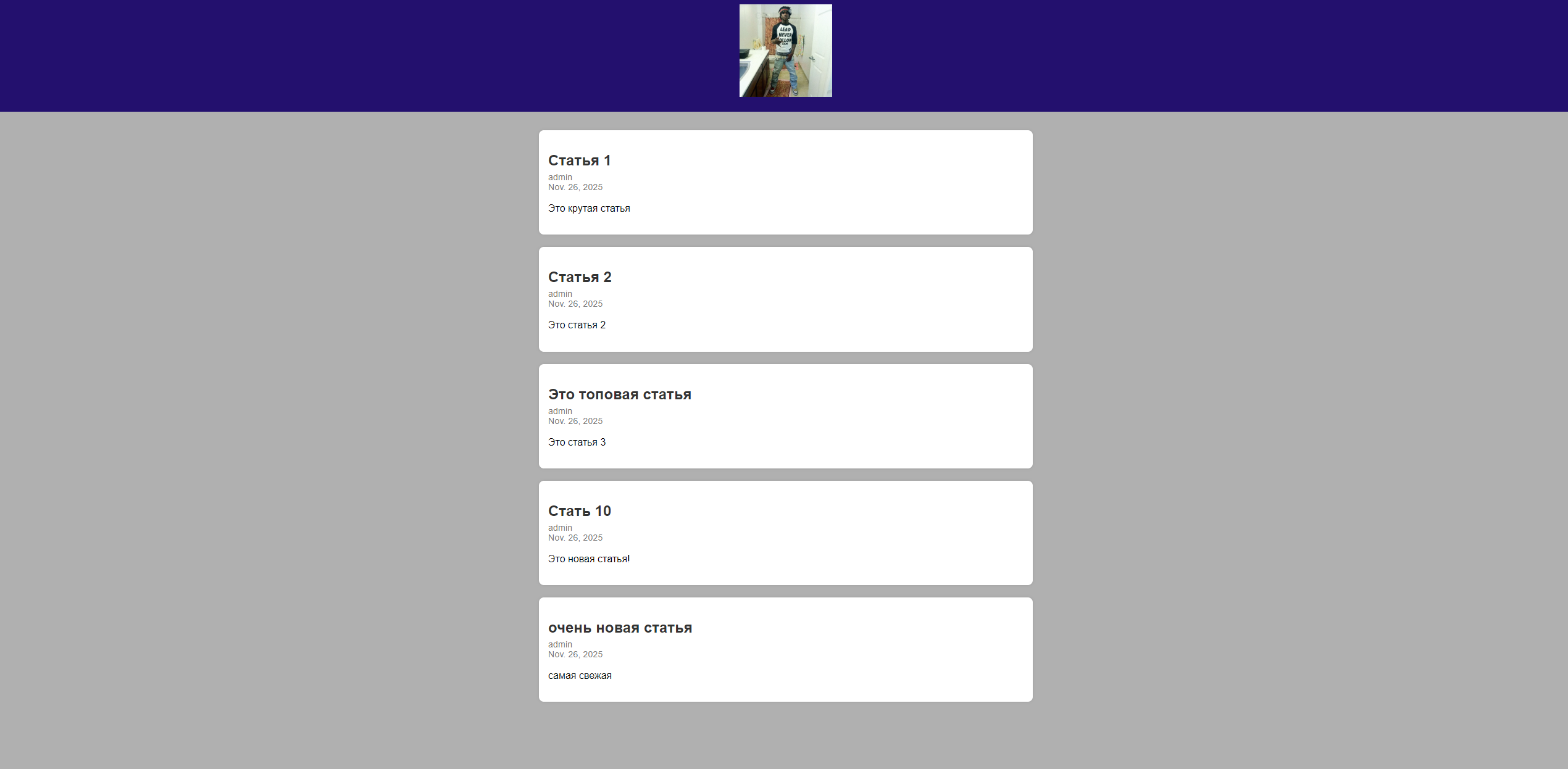
20. Результат



21. Добавим снова статью

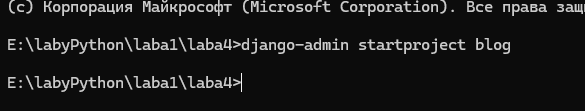


22. Результат работы



**Лабораторная работа № 4**

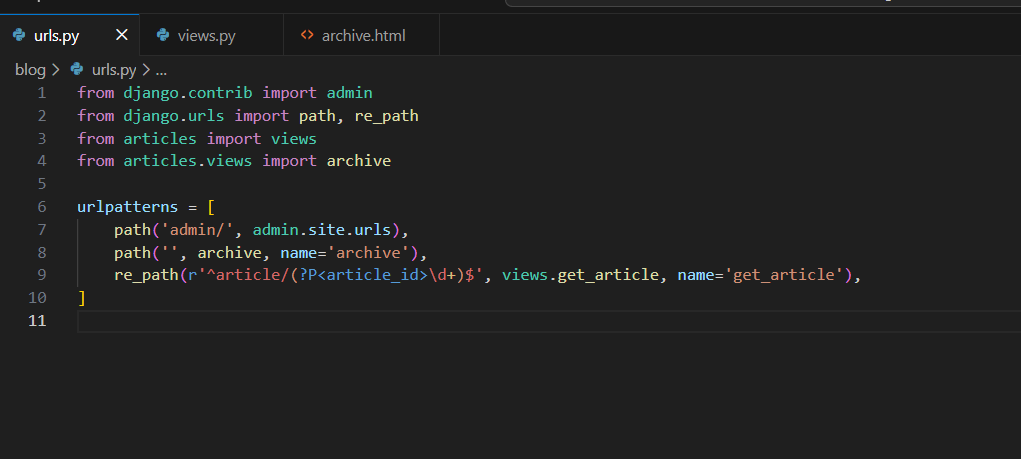
1. Создадим новый проект



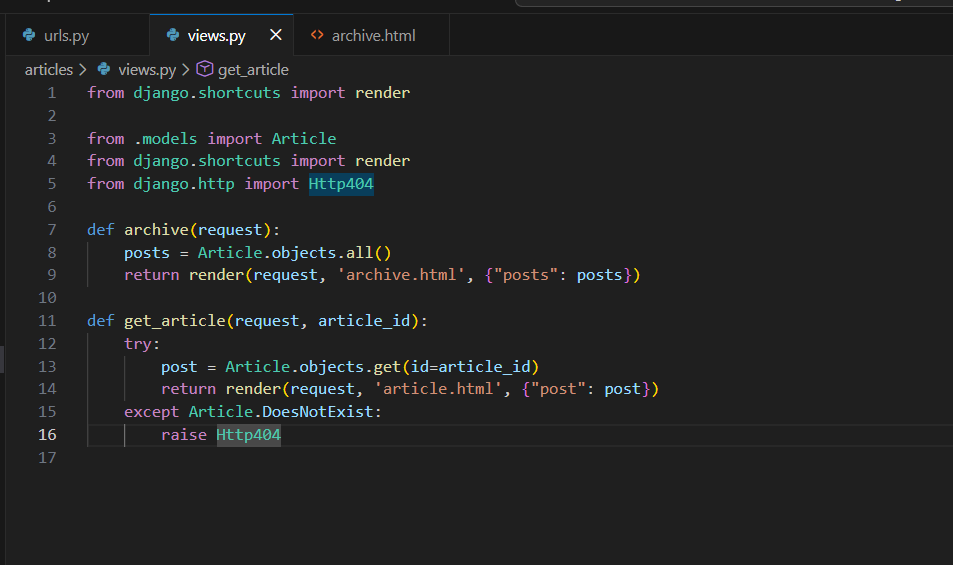
2. Произведем настройку

3. Перенесем приложение из предыдущего проекта, настроим данные в INSTALLED\_APP, а также настроем файл **urls.py**

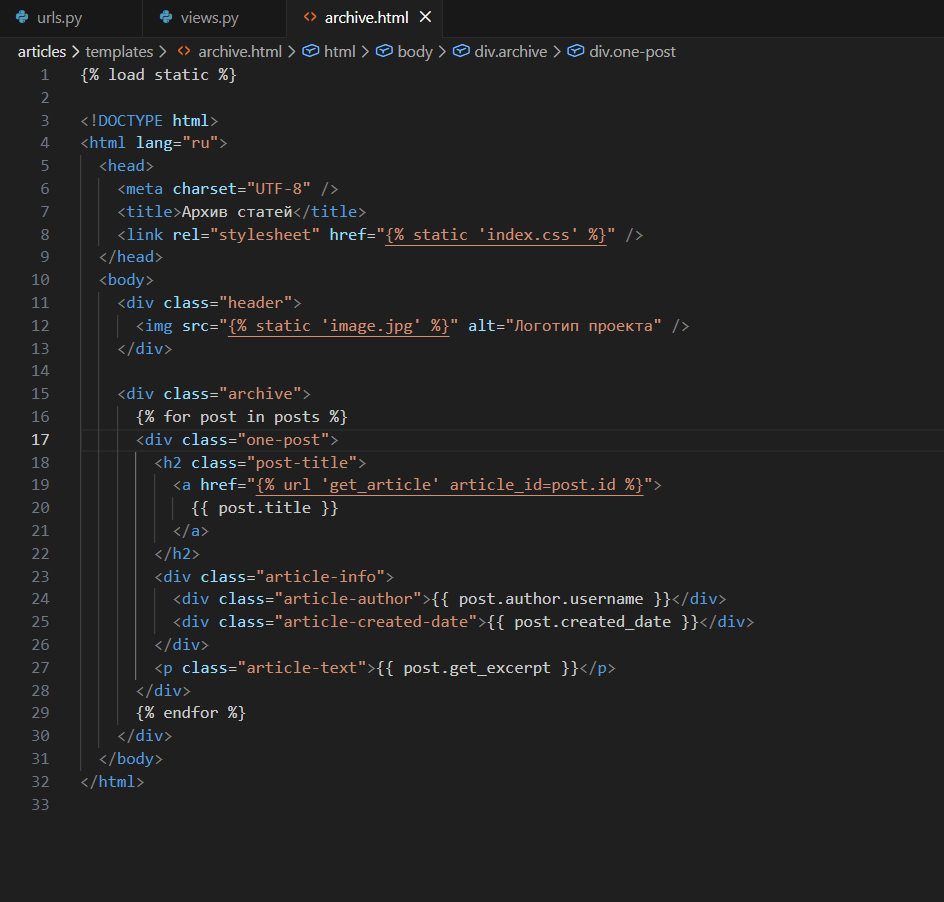
4. Изменим urls.py



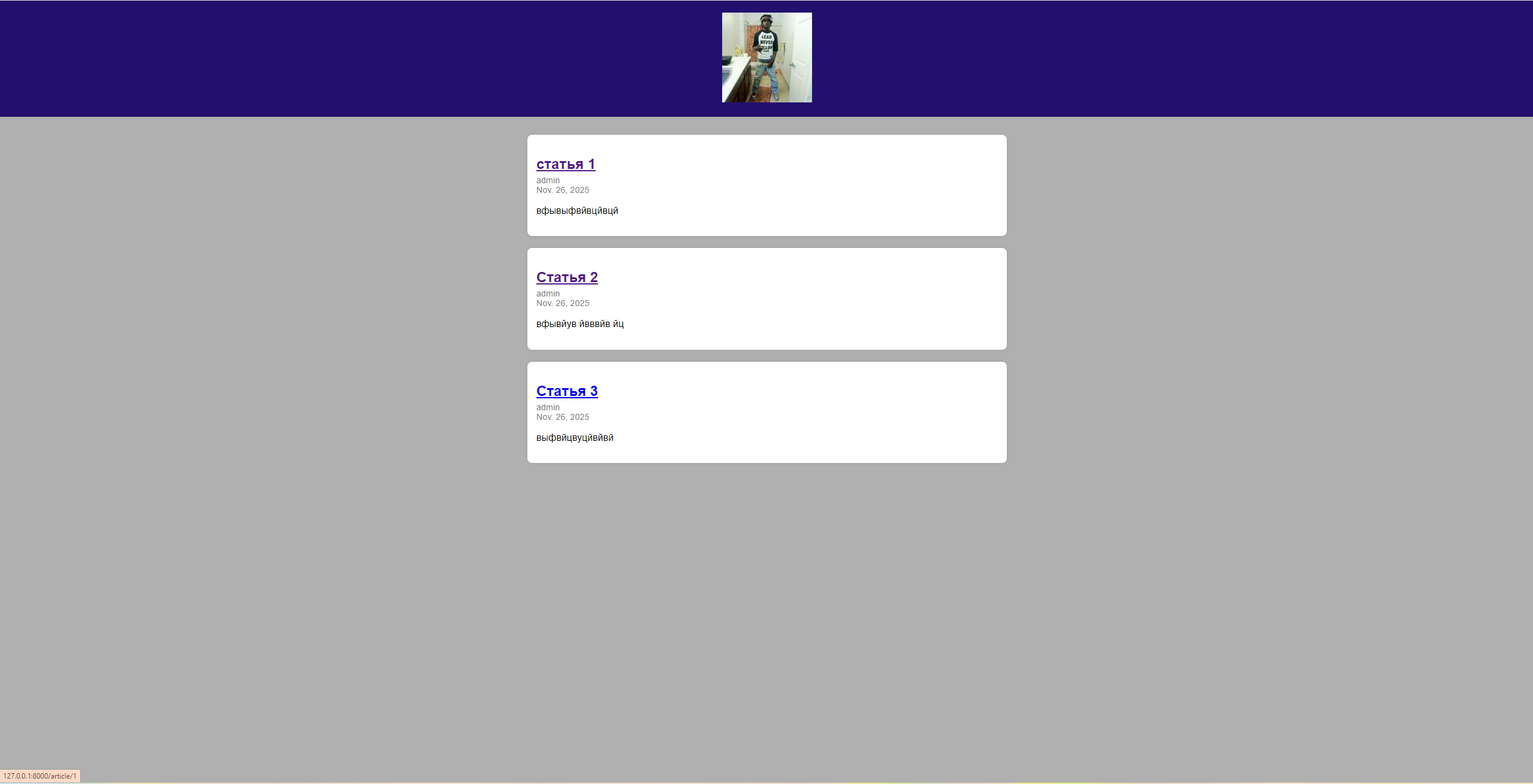
5. Затем изменим views.py



6. Изменим archive.html

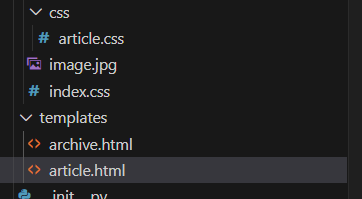


7. Результат. Снизу виден путь, если навестись на статью

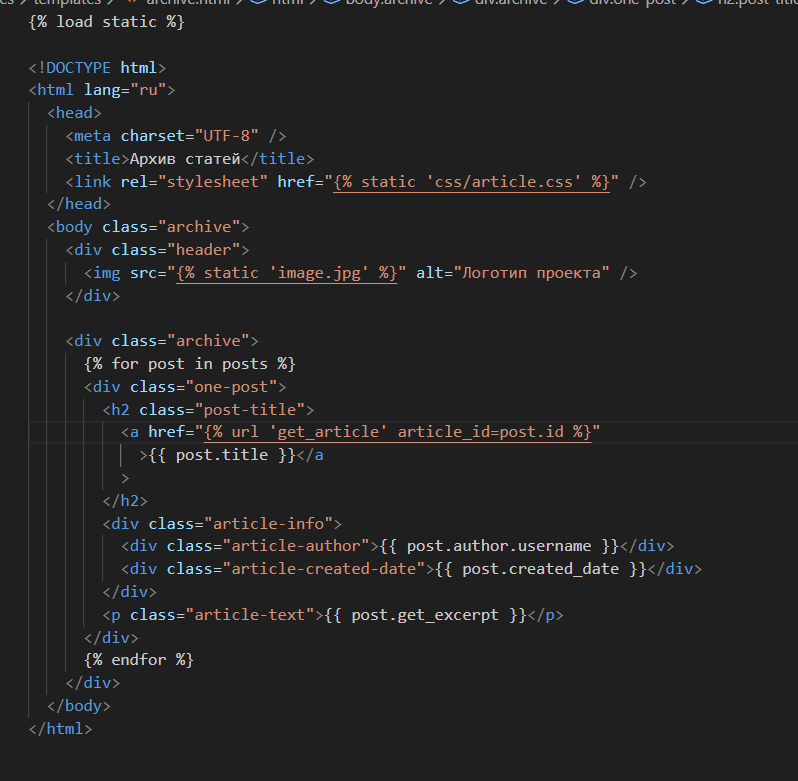


8. Придадим странице стили, сделаем страницу для каждой новости.

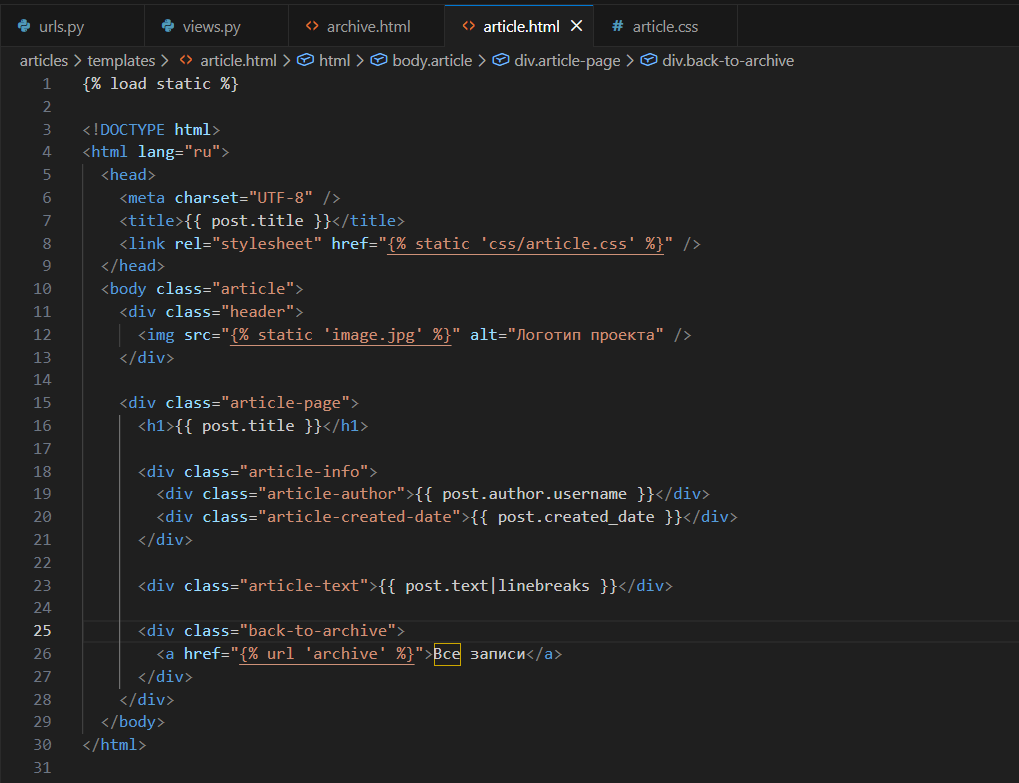
Добавим недостающие папки и файлы в проект



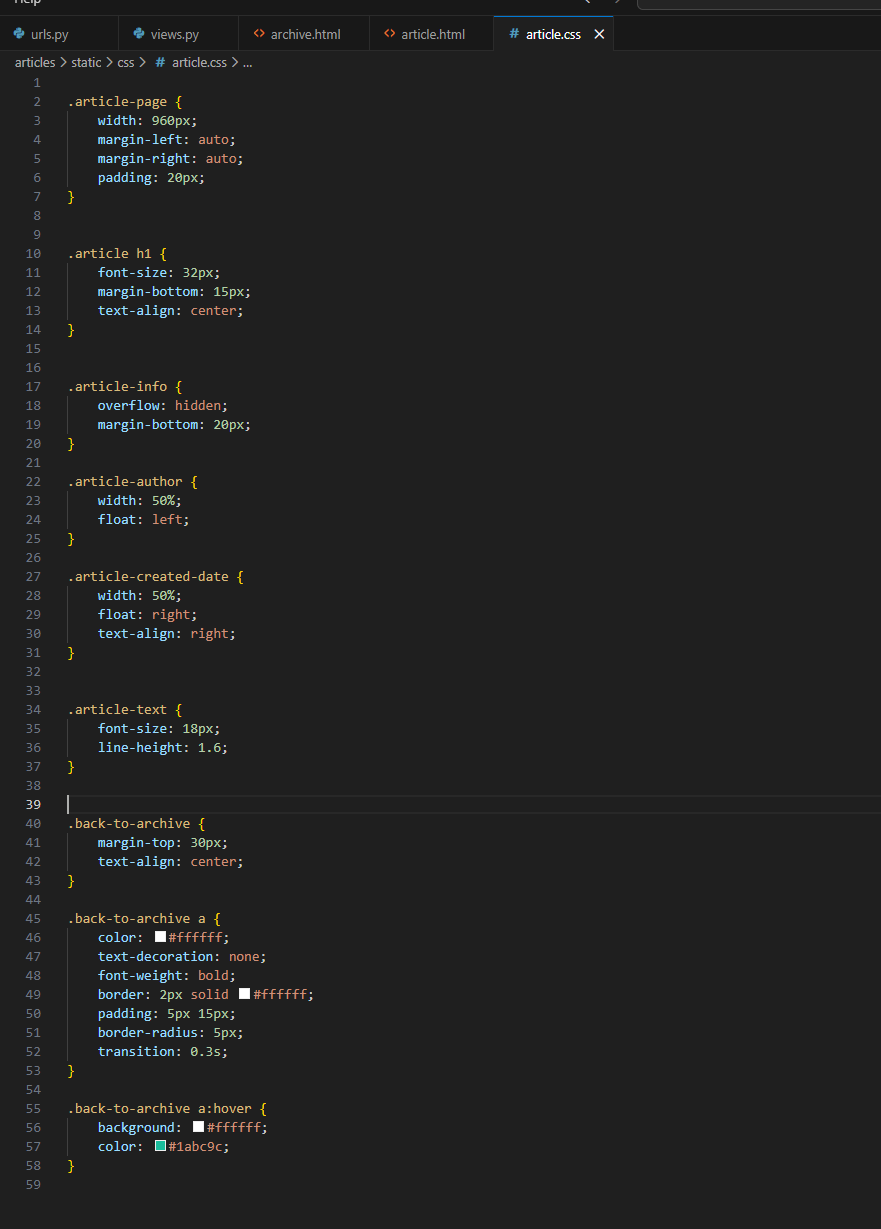
Изменим archive.html



Изменим article.html



Изменим article.css



9. Результат

