**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»**

Лабораторная работа №4

«Создание страницы определенной записи»

по дисциплине:

«Web-программирование»

Выполнил:

Студент группы БФИ19

Бардюк Д.В.

Москва, 2021

**Задания:**

1. Сделайте так, чтобы при клике по названию статьи происходил переход на страницу указанной записи.
2. Создайте стили, соответствующие макету для страницы определенной записи. Не забудьте добавить ссылку «Все записи»;
3. Загрузите ваш проект на любой гит-репозиторий (GitHub, GitLab, Google Code, Bitbucket и т.п.).

**2. Ход работы**

Создаём новую директорию для лабораторной работы №4 и создаём новый проект blog, выполнив команды как показано на рисунке 1.

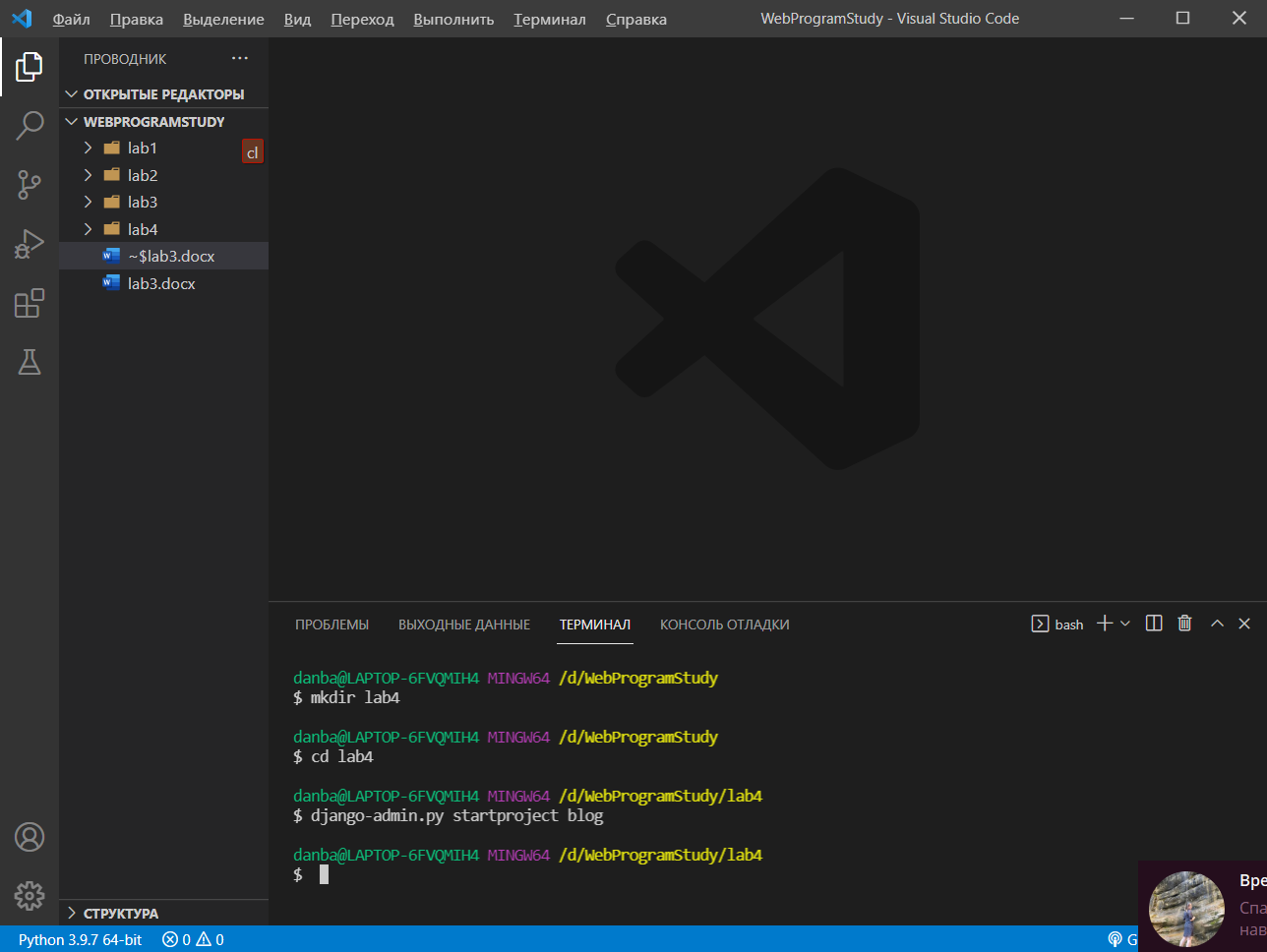


Рисунок 1 – Создание нового проекта

В начале работы над новым проектом, скопируем папку “articles” из предыдущей лабораторной работы, затем задаём базовые настройки: имя базы данных и создаём ее таблицы, выполнив команды как на рисунке 2.

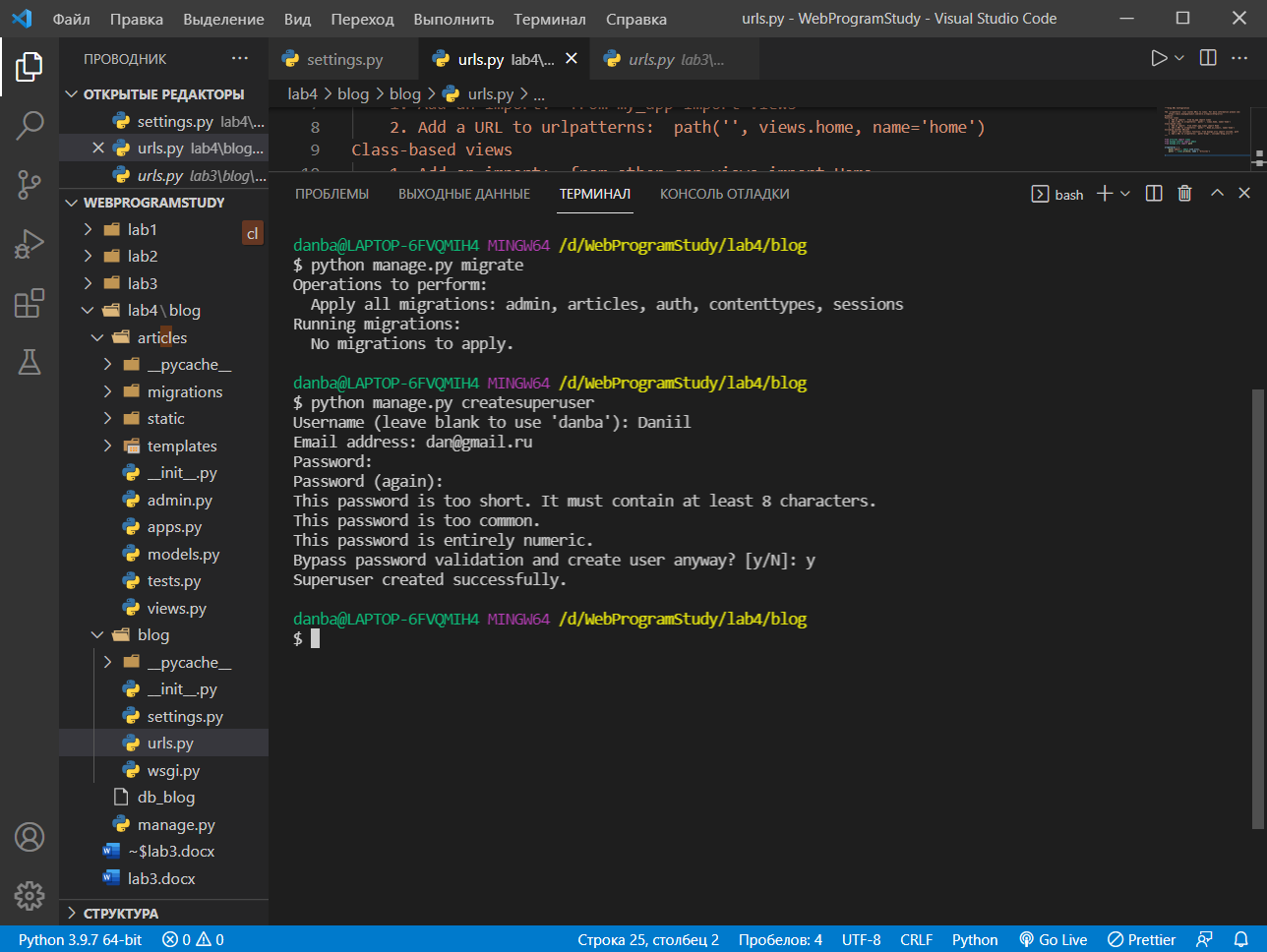


Рисунок 2 – Базовые настройки проекта

Для того, чтобы можно было бы прочитать значение идентификатора (id) отдельного поста, укажем в качестве адреса следующее регулярное выражение в файле urls.py как на рисунке 3.

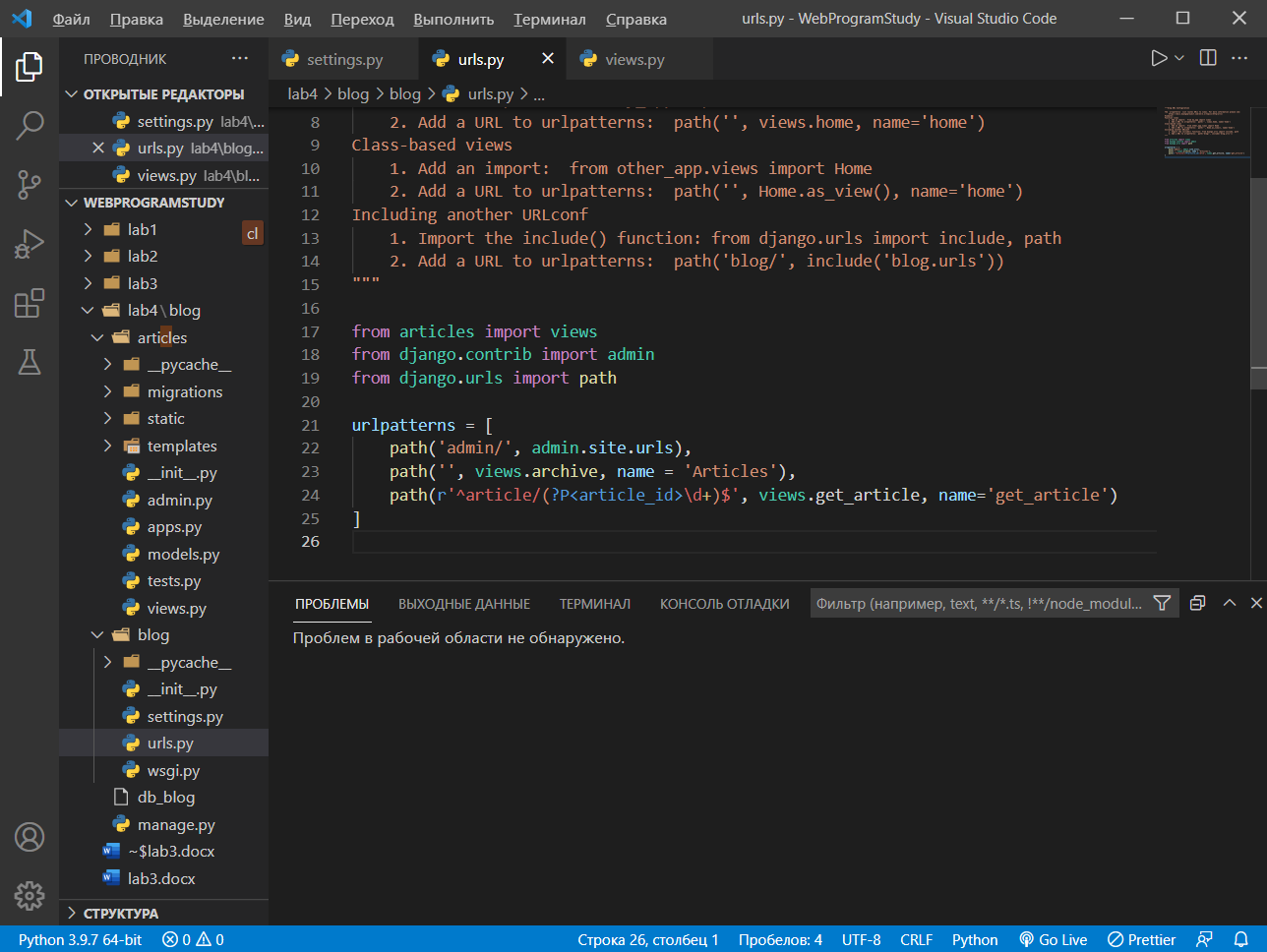


Рисунок 3 – Файл urls.py

После того, как в регулярном выражении мы объявили именованную группу, в представление можно добавлять именованный аргумент, куда и будет передано значение из адреса, по которому перешел пользователь. Создадим в файле views.py из директории articles помимо уже написанной в предыдущей работе функции archieve ещё и функцию get\_article, как показано на рисунке 4.

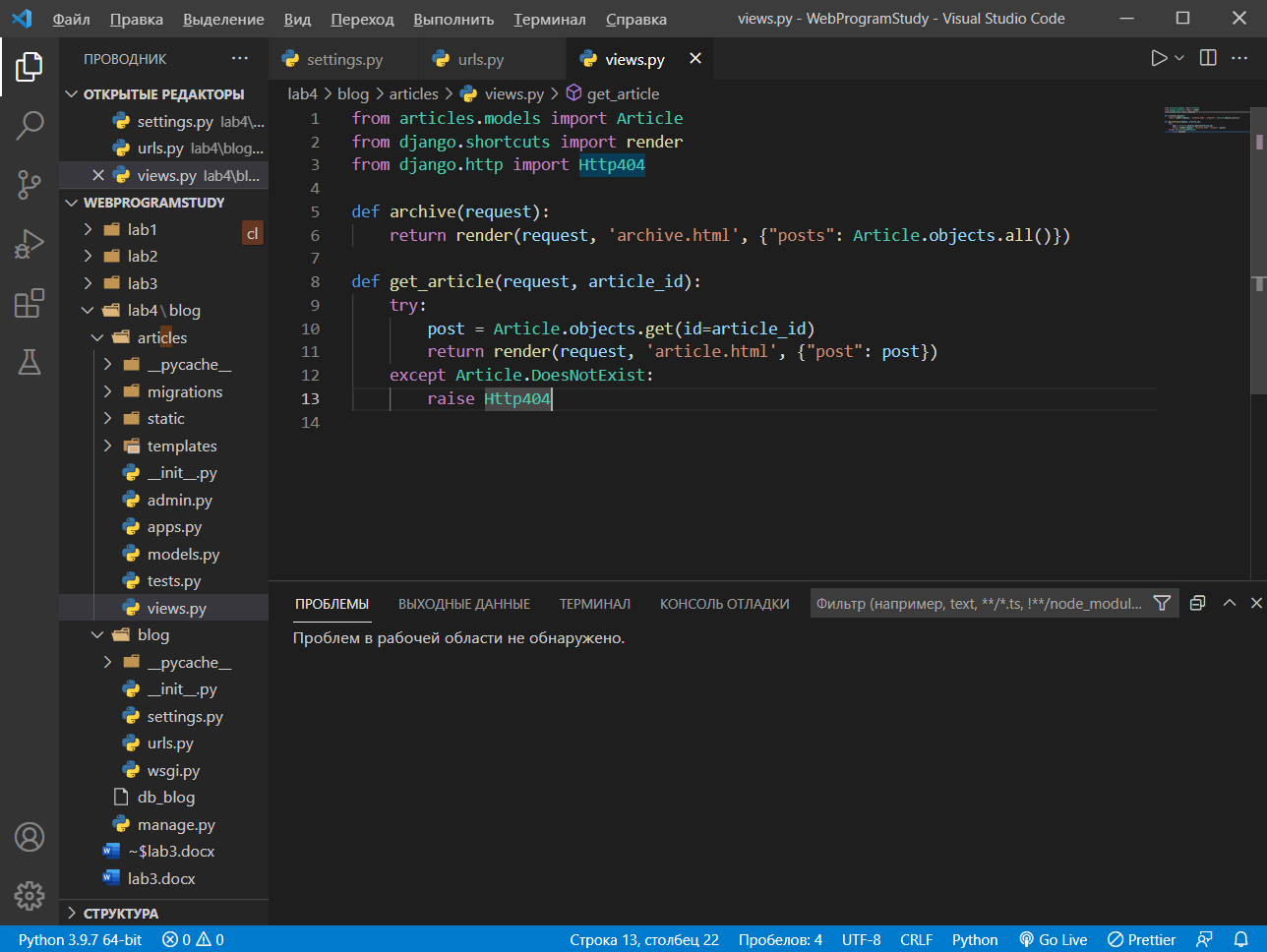


Рисунок 4 – Работа с файлом views.py

В папке articles/templates/ создадим файл article.html, как показано на рисунке 5.

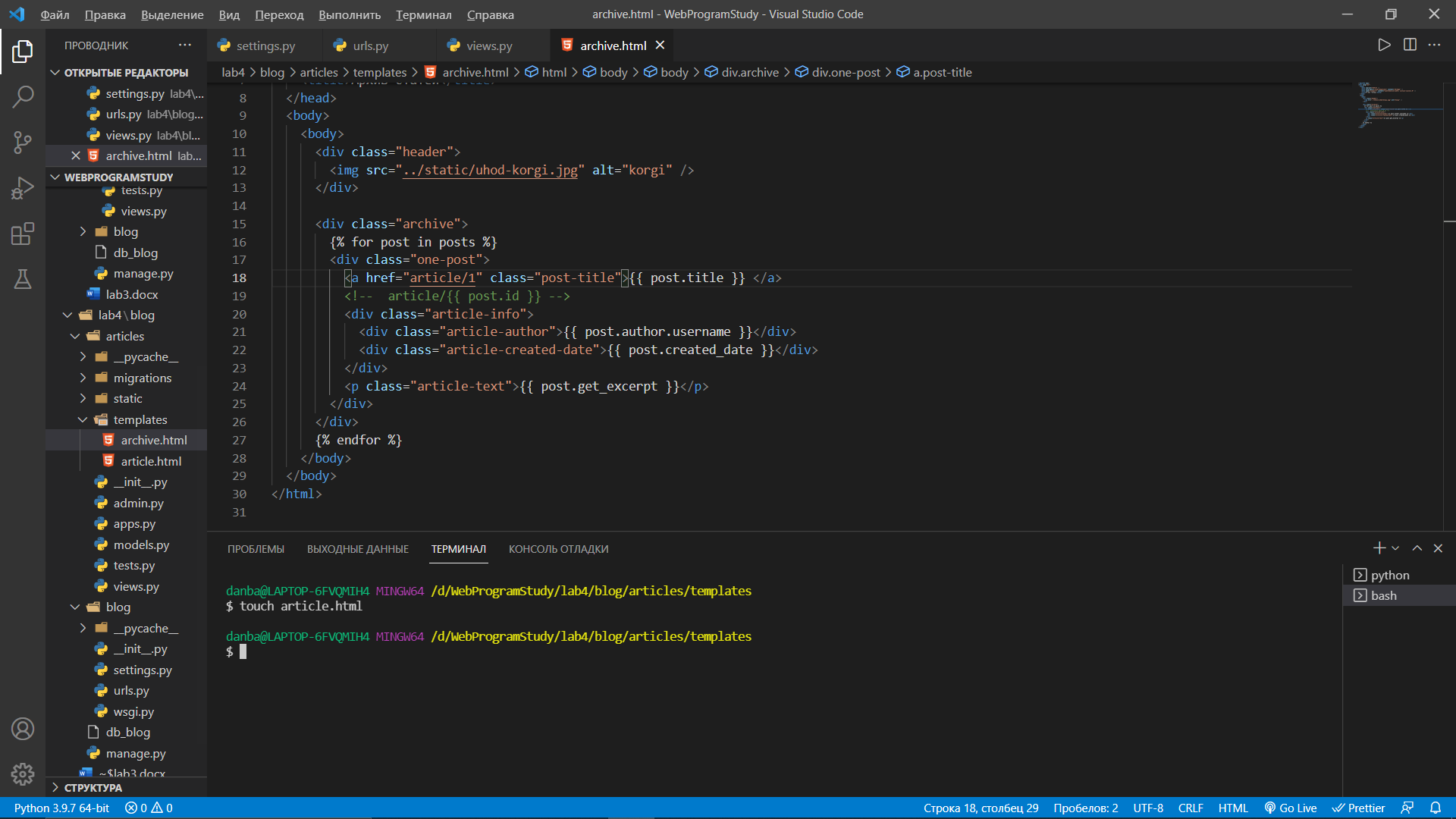


Рисунок 5 – создание файла article.html

По заданию необходимо создать 3 статьи и затем посмотреть их в файле базы данных с помощью программы управления базами данных. В нашем случае это программа SQLiteManager. Результаты представлены на рисунках 6-7.

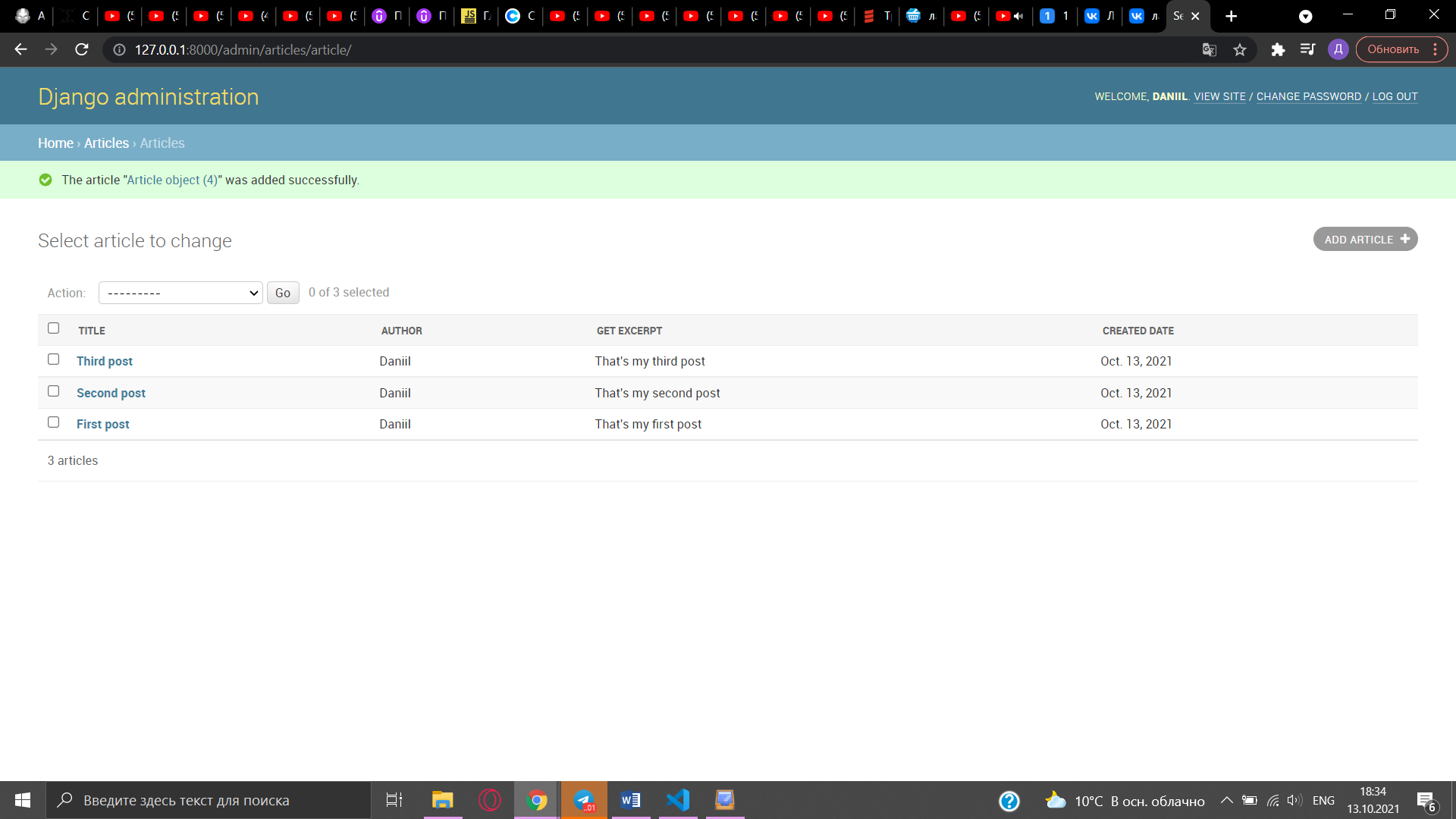


Рисунок 6 – Статьи на странице

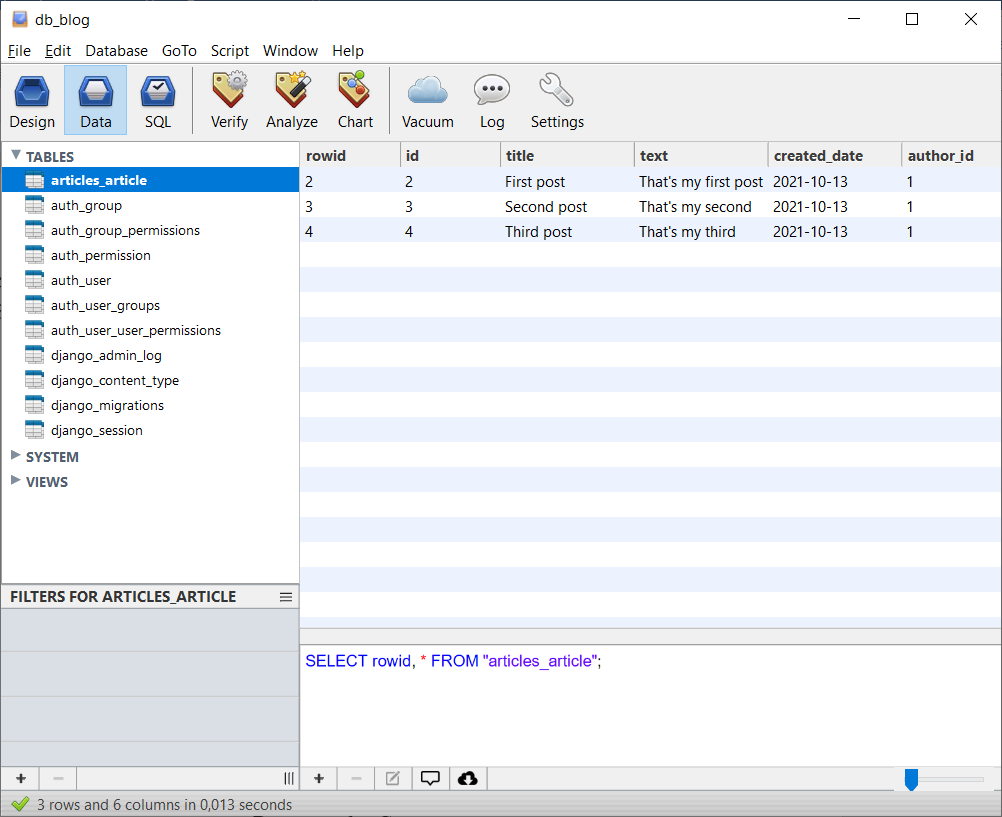


Рисунок 7 – Файл базы данных

Далее переходим в директорию articles, в ней создаём папку templates, внутри которой создаём файл archive.html со следующим кодом, представленным на рисунке 8.

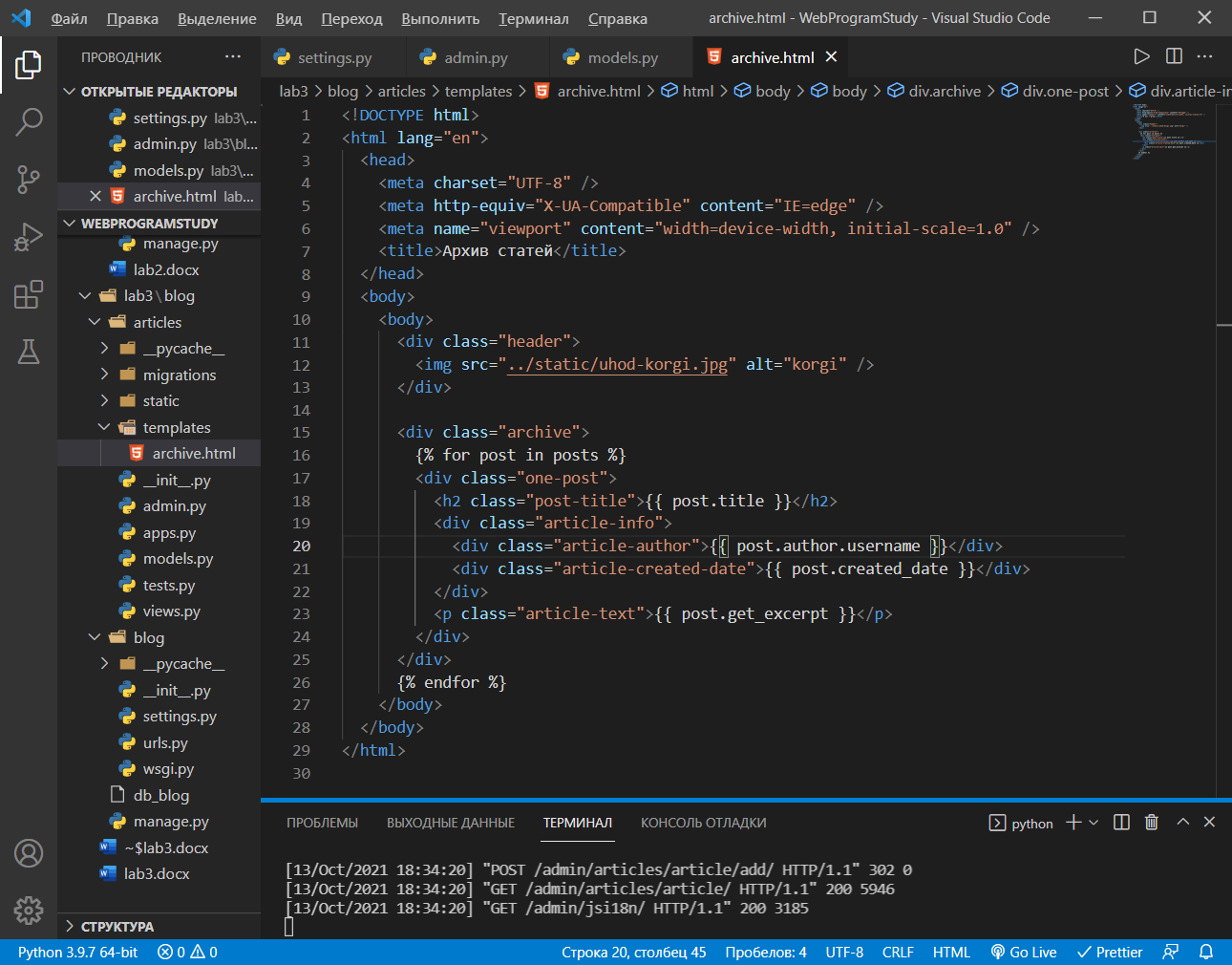


Рисунок 8 – файл archive.html

В файле views.py в директории articles создадим представление archive, которое будет возвращать html-страницу со всеми созданными постами в текущем проекте. Код файла представлен на рисунке 9.

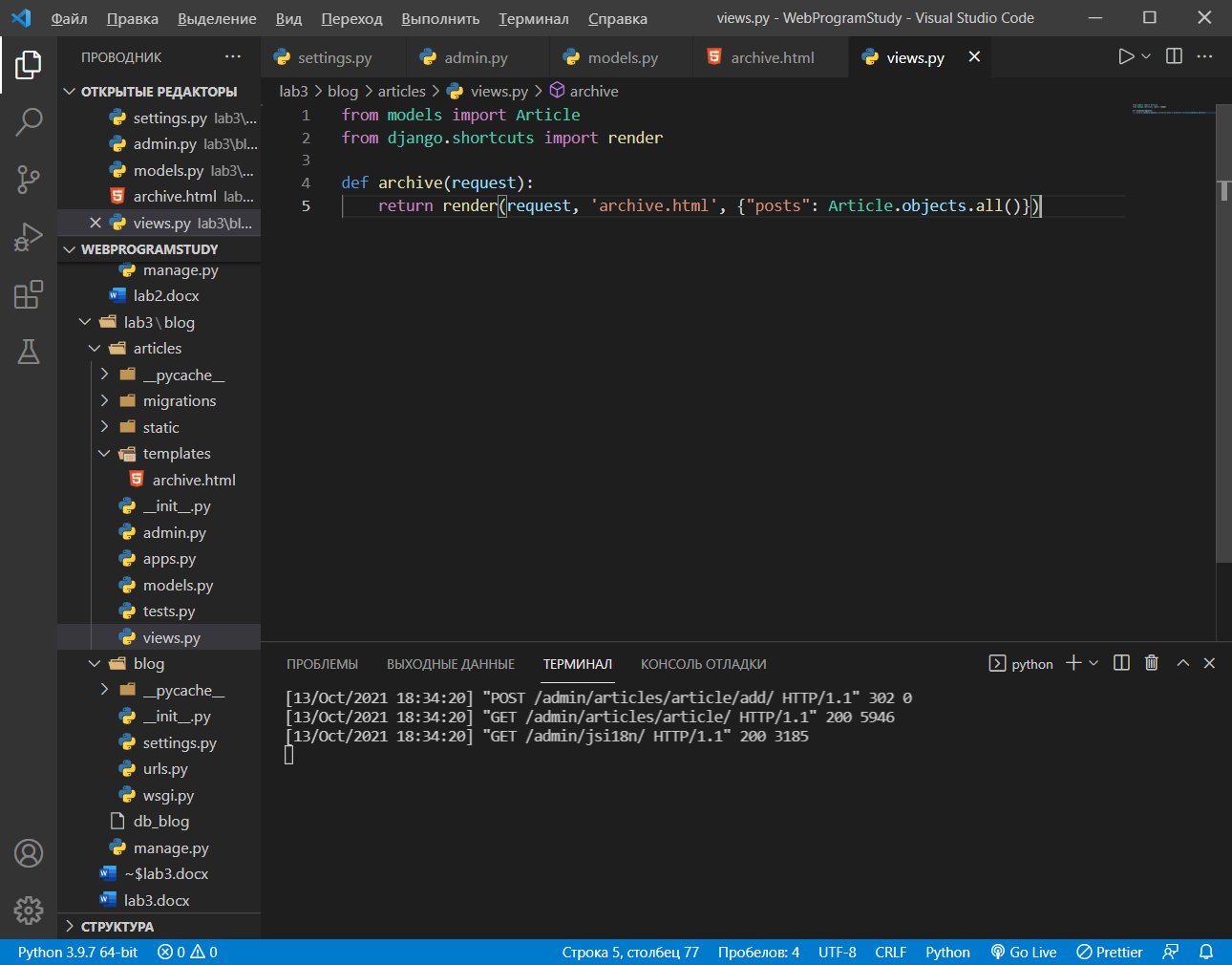


Рисунок 9 – файл views.py

Настраиваем наш url по которому будут отображаться все наши статьи(рисунок 10). По заданию это должна быть домашняя страница. Результат представлен на рисунке 11.

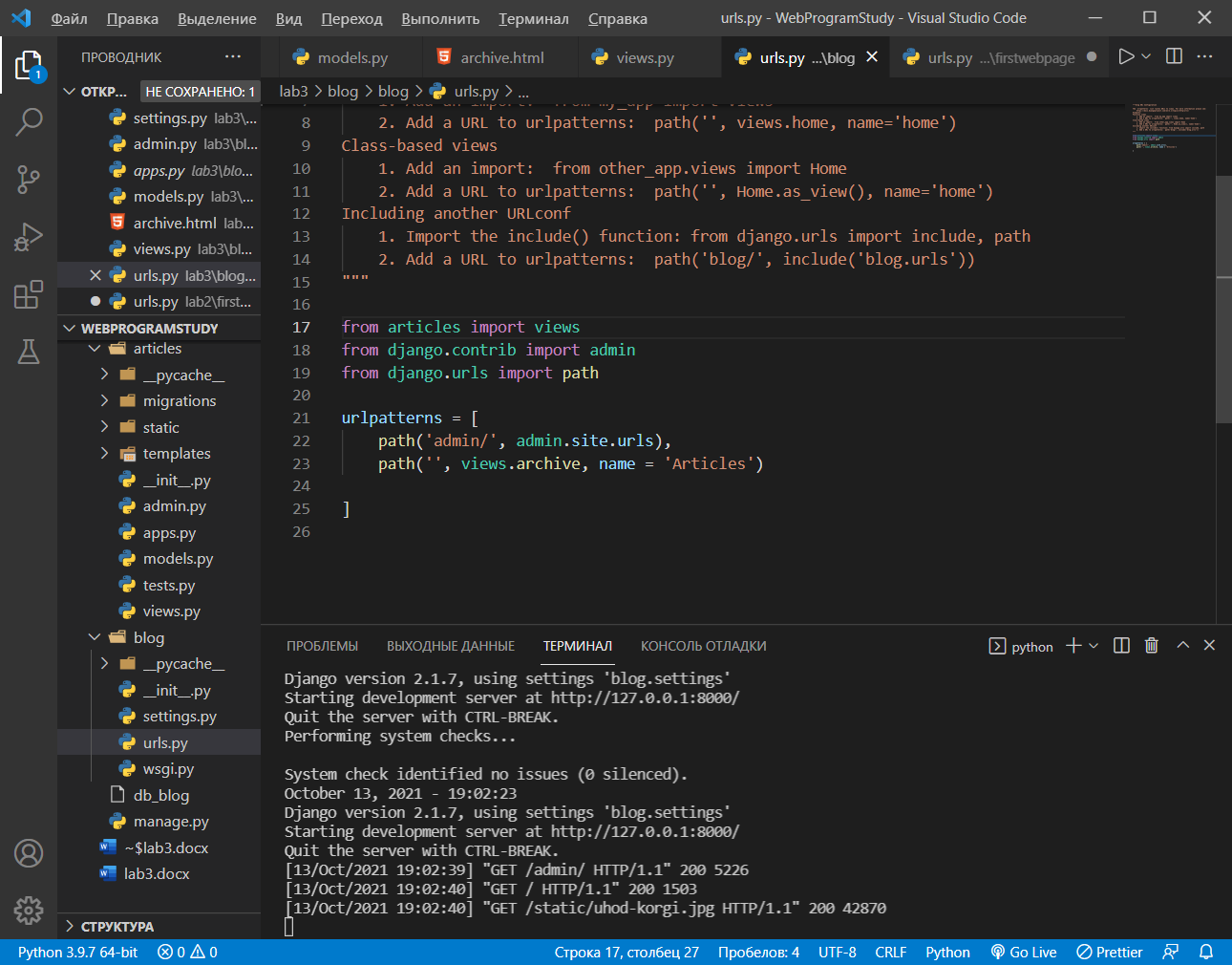


Рисунок 10 – Файл url.py

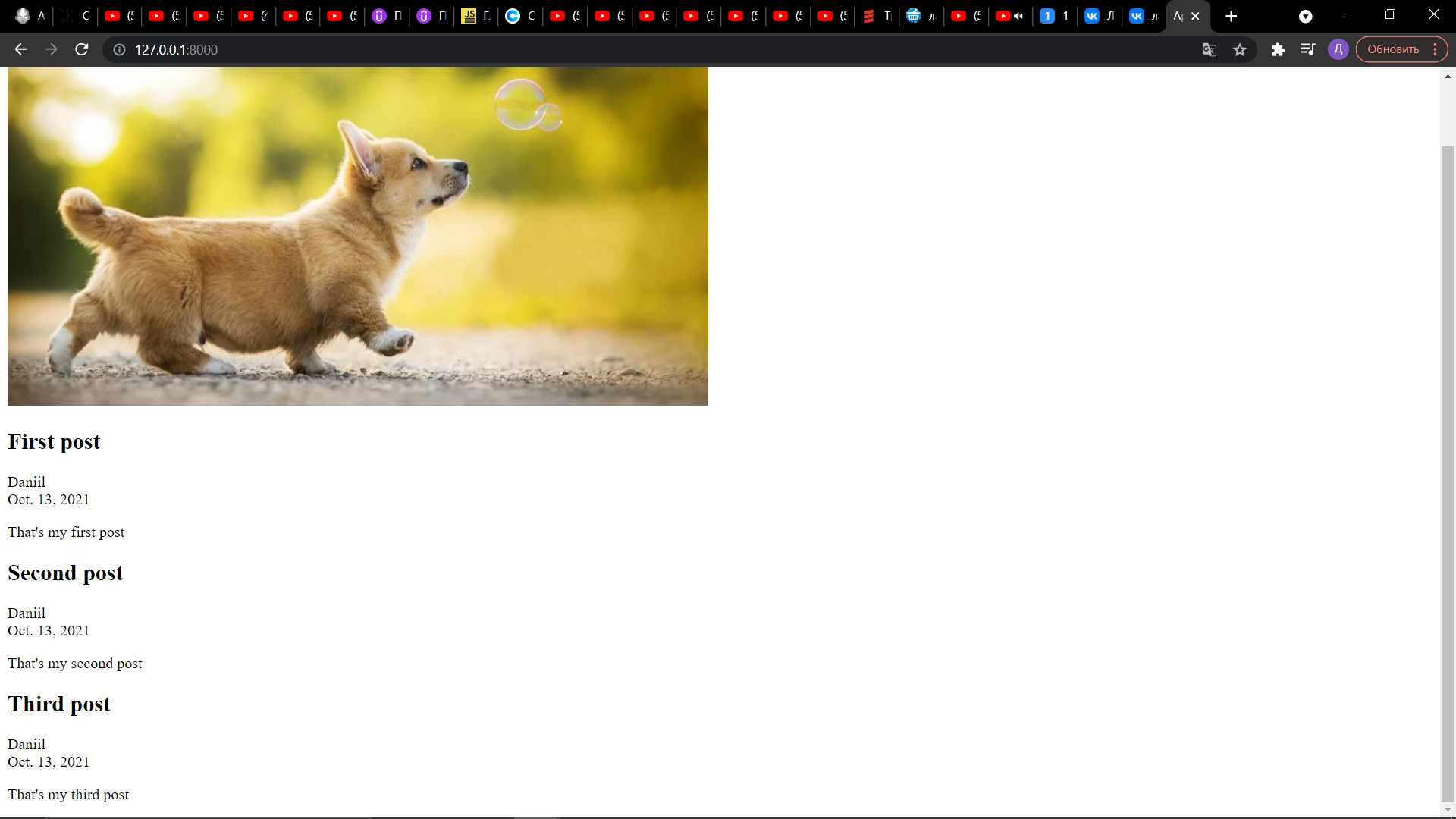


Рисунок 11 – Домашняя страница сайта

Далее необходимо открыть файл нашей базы данных с помощью программы SQLite Manager и добавить новую учётную запись. Результат представлен на рисунках 12-13.

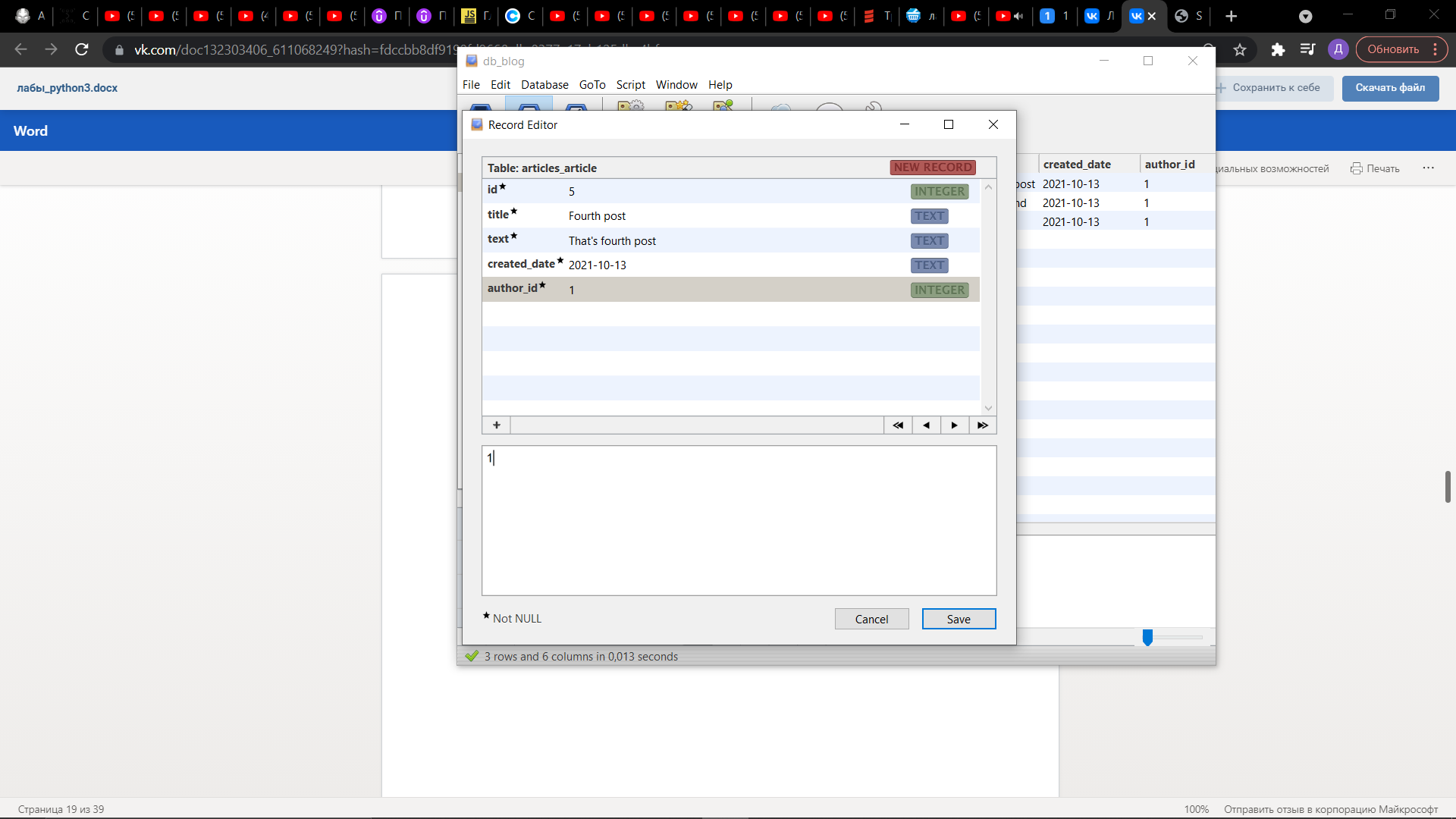


Рисунок 9 – Добавление новой статьи

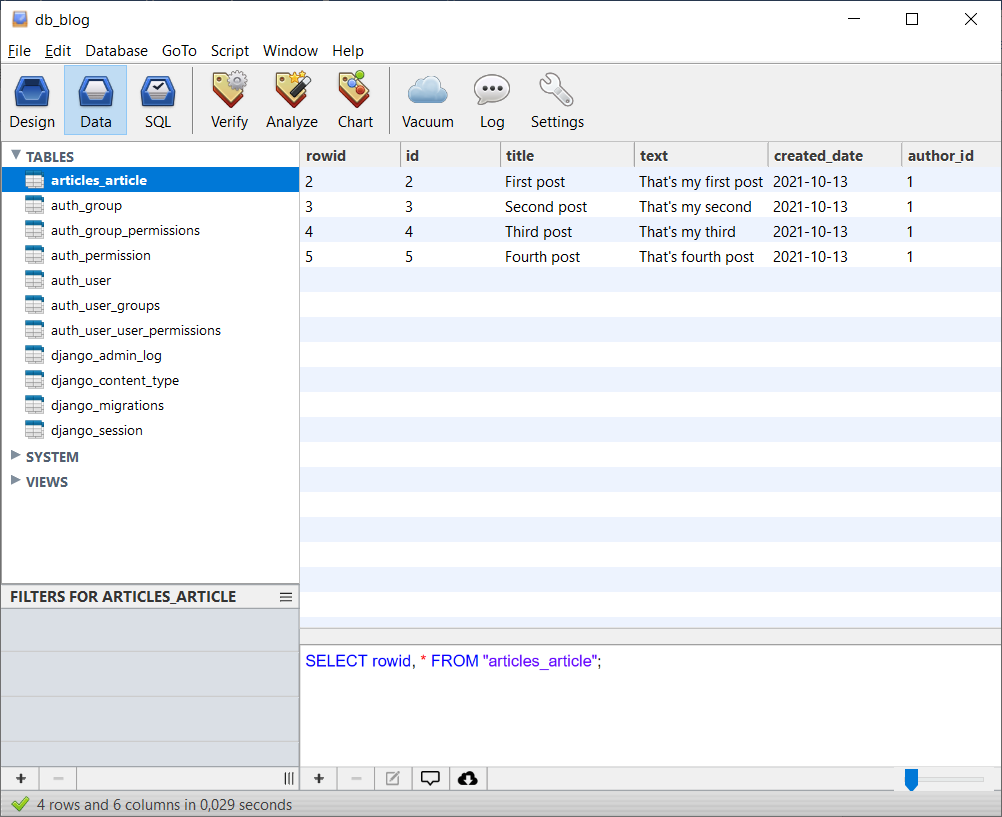


Рисунок 10 – Результат работы

**Вывод:** я установил язык программирования «Python» и фреймворк Django, ознакомился с синтаксисом и выполнил задания.