Федеральное агентство связи

Ордена Трудового Красного Знамени федеральное государственное бюджетное образовательное учреждение высшего образования

«Московский технический университет связи и информатики»

Кафедра математической кибернетики и информационных технологий

Лабораторная работа №5

по дисциплине

«Кроссплатформенные технологии программирования»

Выполнил студент группы

БСТ1702

Панкратов В.И.

Вариант №14

Москва

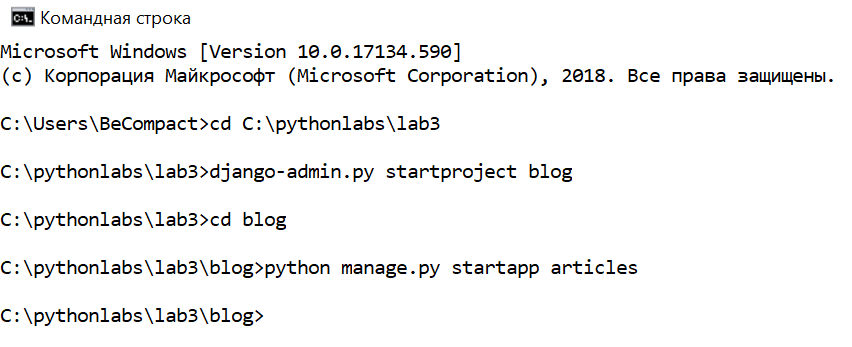
2019

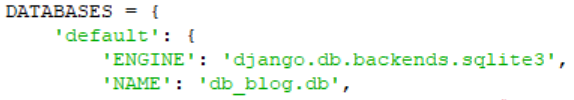
**Создание первой модели данных и её регистрация в административном приложении django.**

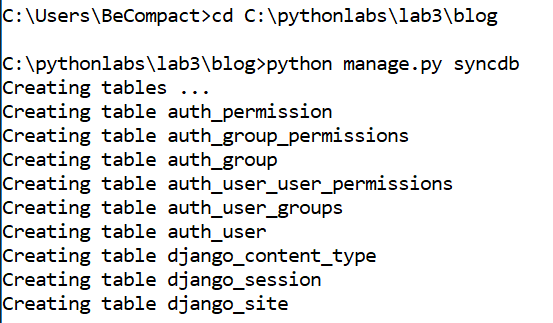
С помощью команды django-admin.py startproject blog создаём в папке lab3 проект под названием blog. Теперь в папке blog выполняем команду

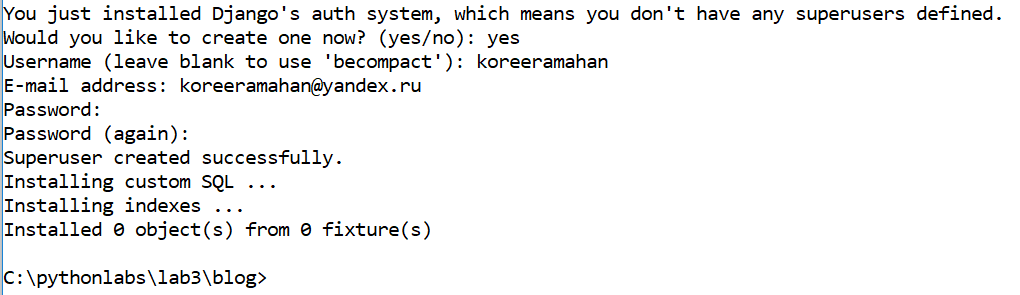
python manage.py startapp articles (создает новое приложение).

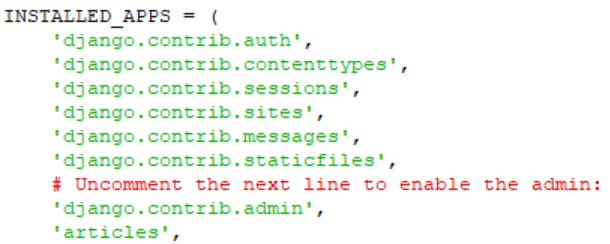
Выполняем базовую настройку проекта.

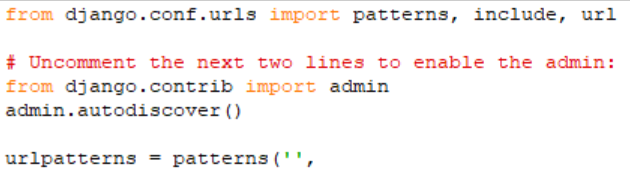








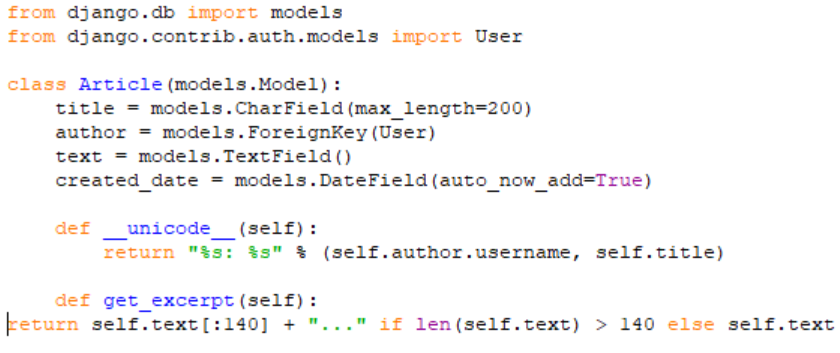




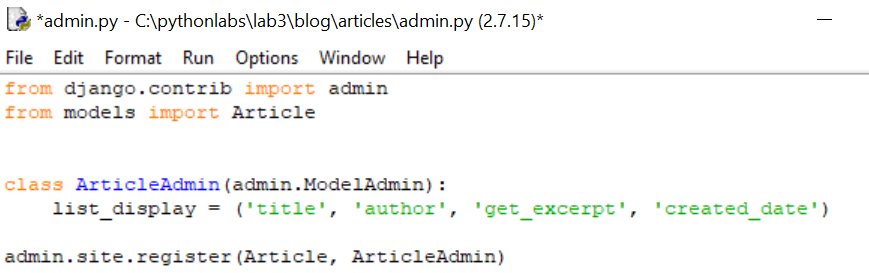


Открываем файл articles/models.pу, в котором будут храниться будущие

модели статей. Пишем в этом файле:



В articles создаем файл admin.py и прописываем следующее:



**Задание:**

* Когда модель зарегистрирована, можно создавать новые записи,

достаточно перейти в административную панель, находящуюся по адресу

http://127.0.0.1:8000/admin/. Перейдите по вкладке Articles и кликните

по кнопке “Add article” справа наверху. Заполните поля Title, Author и Text

значениями по своему усмотрению. Создайте ещё две статьи подобным

образом.

* С помощью программы управления базами данных sqlite3 (например,

программы SQLiteManager) откройте файл базы данных текущего

проекта, который хранится в папке проекта с именем, объявленным в

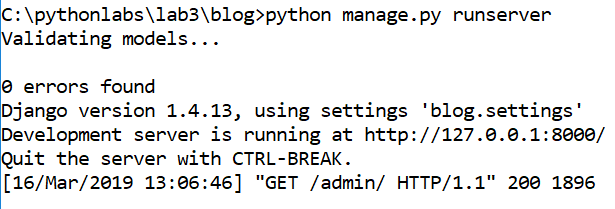
настройках проектах в переменной «DATABASES.NAME». Найдите

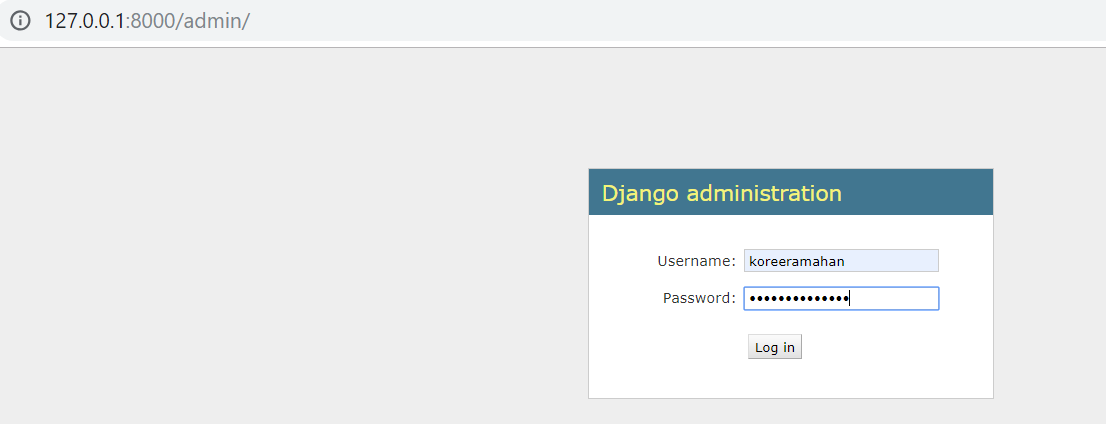
созданные в предыдущем пункте задания экземпляры записей. Измените

текст одной записи и название статьи для другой.

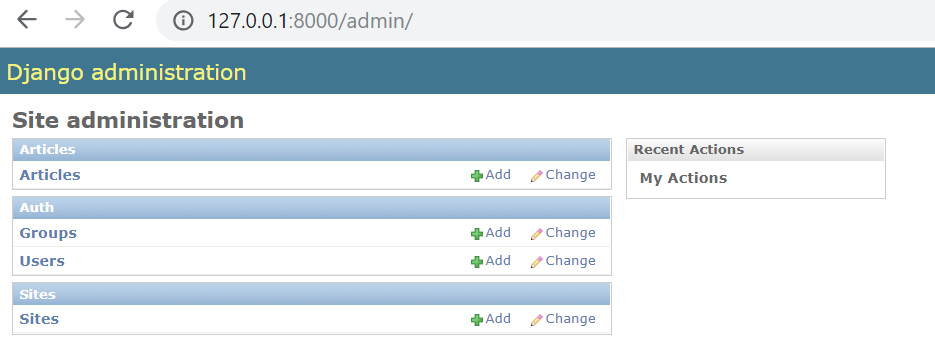
**Выполнение:**

Запускаем сервер и переходим по ссылке <http://127.0.0.1:8000/admin/>.

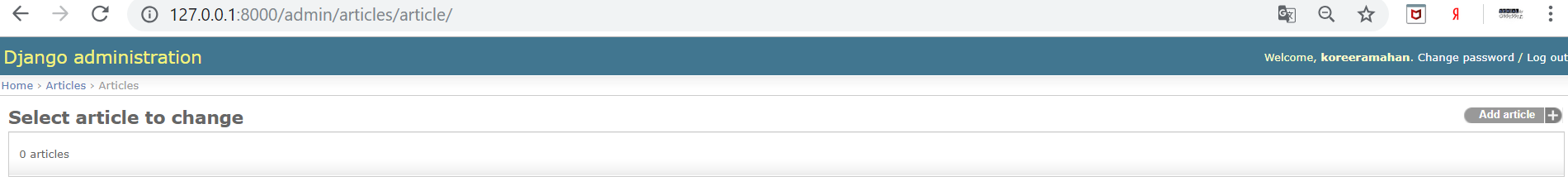




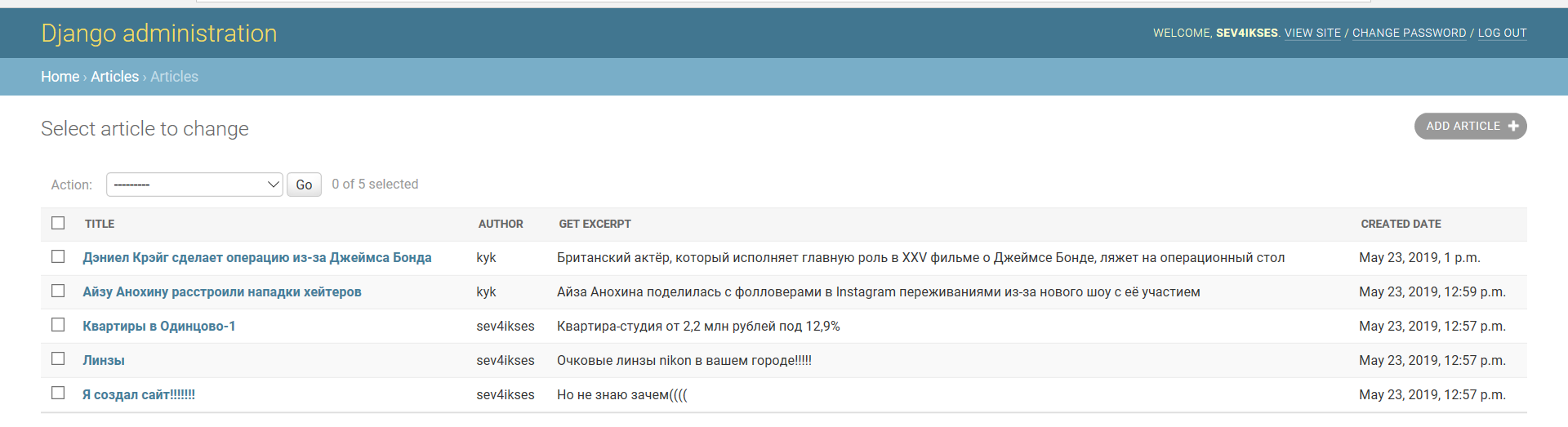
Внешний вид сайта:



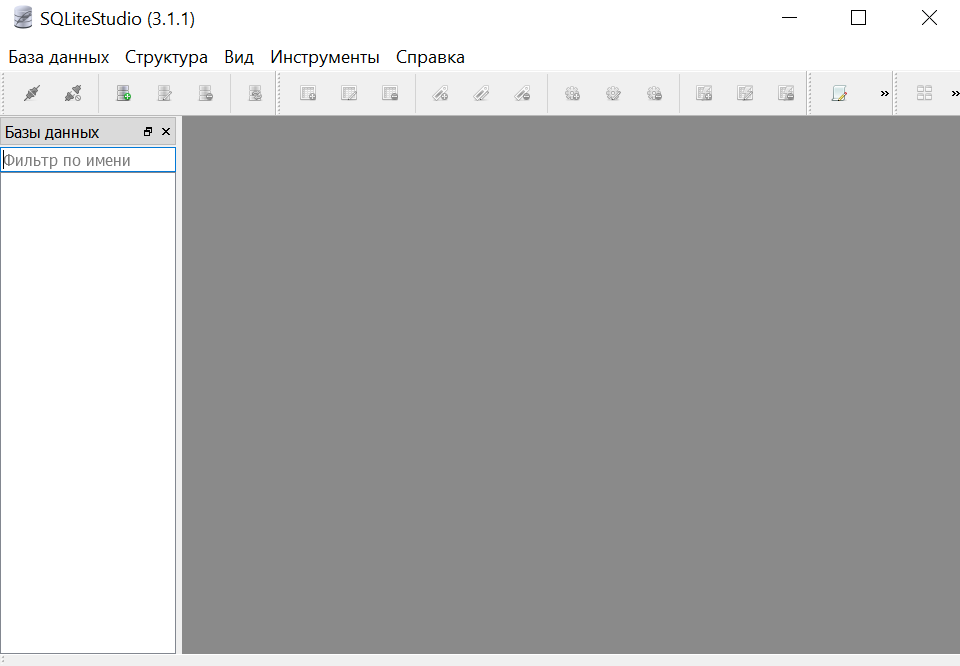
Переходим по вкладке Articles:



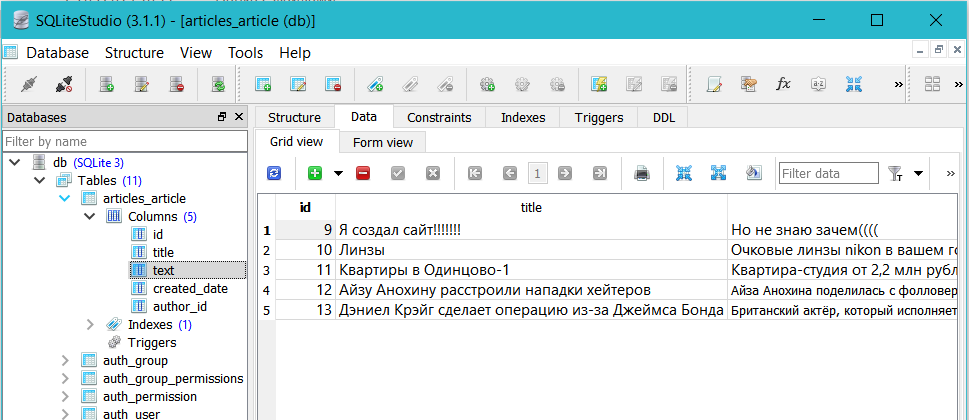
Добавляем несколько статей:



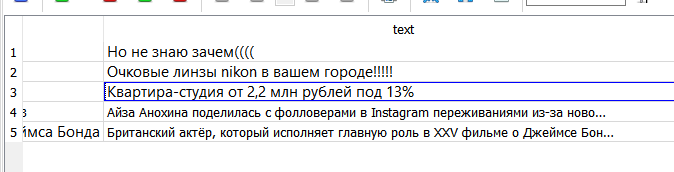
Скачиваем SQLiteStudio:



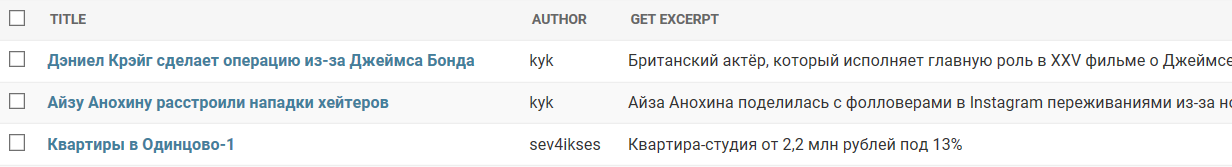
Открываем базу данных db\_blog:



Изменяем текст одной записи и название статьи другой:



Проверяем изменения на сайте:

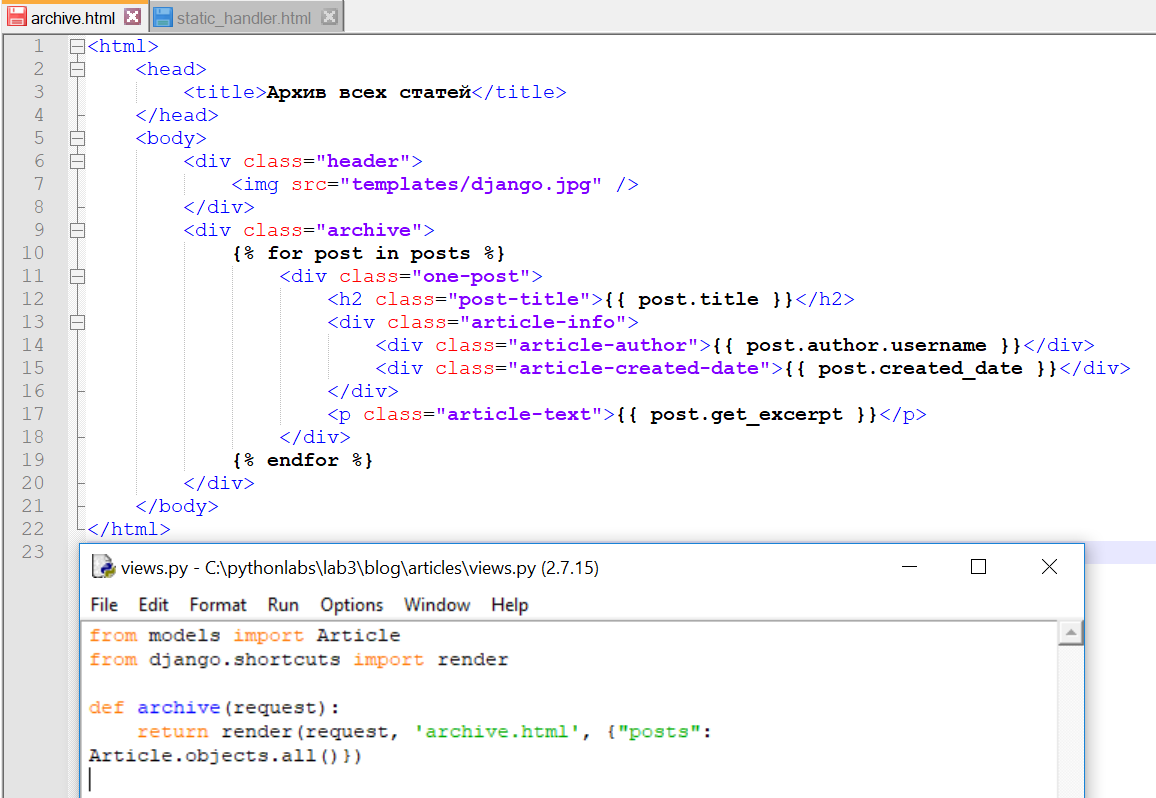


**Динамическое генерирование шаблона для вывода всех экземпляров**

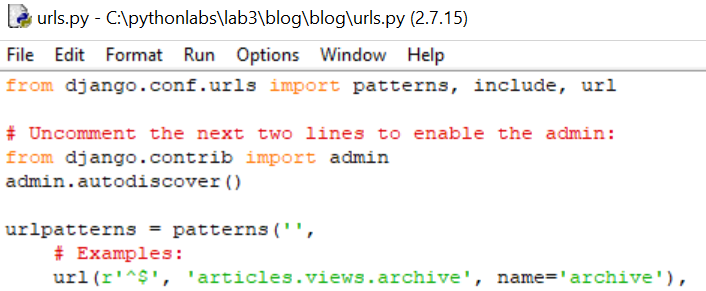
**этой модели.**

Создаем archive.html в папке templates в articles.

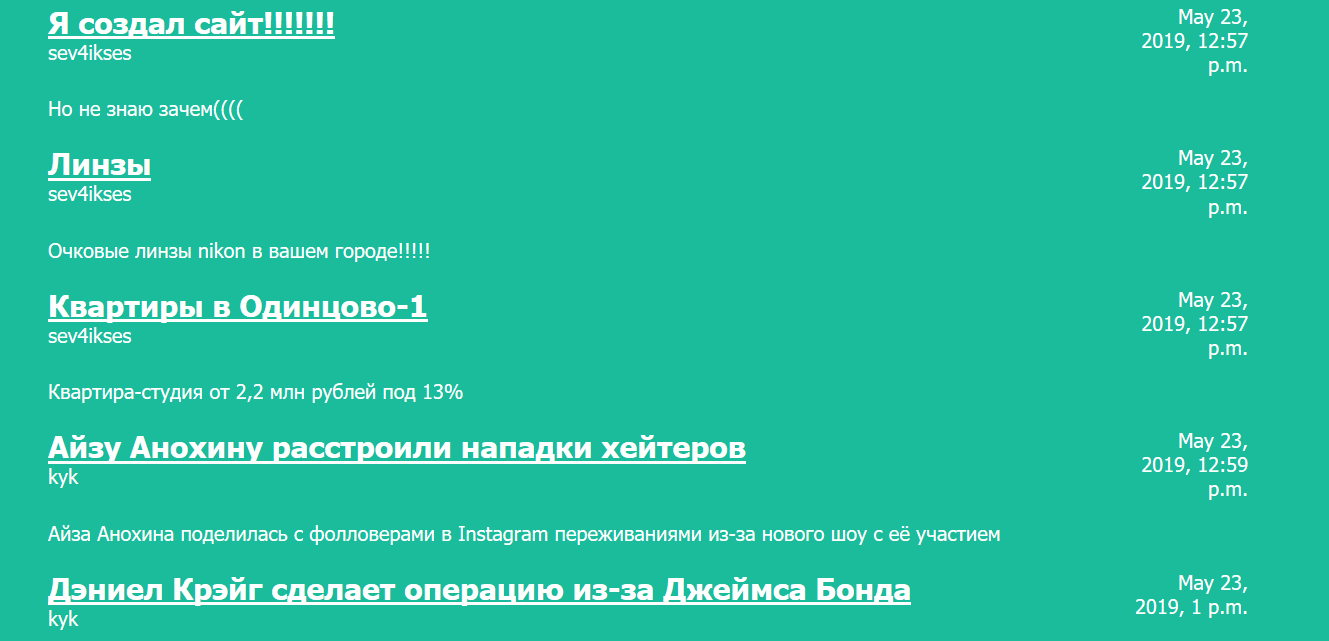
Прописываем код в этом файле и в файле views.py.



Нужно указать Django, что при заходе на главную страницу сайта будут отображаться все статьи сайта.



Тогда, переходя по ссылке <http://127.0.0.1:8000/>, будет отображаться:



**Задание:**

* Вновь откройте файл базы данных sqlite3, где хранятся экземпляры

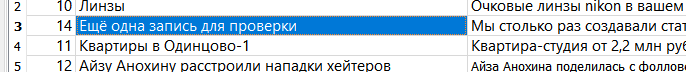
статей к текущему проекту. Добавьте новую запись в блог

непосредственно через менеджер базы данных (не забудьте зафиксировать

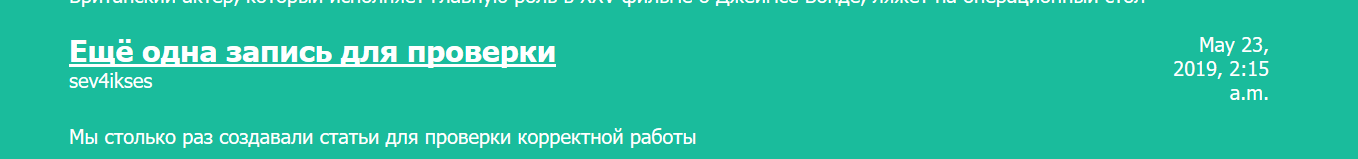
транзацкию, иными словами, сохранить все внесенные изменения).

**Выполнение:**

Заходим в SQLite Studio в нашу базу данных и нажимаем на плюс (вставить строку). Заполняем строку данными и нажимаем подтвердить.



Заходим на <http://127.0.0.1:8000/>:



Заходим на <http://127.0.0.1:8000/admin/> в раздел статьи:

