Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования «Ульяновский государственный технический университет»

Кафедра «Вычислительная техника»

**Отчет по лабораторным работам №7 и №8**

Дисциплина: «Разработка профессиональных приложений»

Связи в БД. CRUD. Регистрация и авторизация

Вариант № 12

Выполнил:

студент группы ИВТАСбд-21

Клишин К. С.

Проверил:

преподаватель кафедры

«Вычислительная техника»

Исхаков И.И.

Ульяновск, 2023

**Задание по варианту**

Для предложенного варианта заданий необходимо реализовать веб-сайт в среде Django, удовлетворяющий следующим требованиям:

1. Количество таблиц в БД - не менее 5.

2. Должны быть реализованы 3 типа связей: one-to-one, one-to-many, man-to-many.

3. Таблицы создаются с использованием моделей в Django и миграций.

4. Как минимум, для 1 сущности должны быть реализованы CRUD операции (Create, read, update, delete) + представление списка сущностей.

5. Остальные таблицы должны быть доступны для редактирования в административной части сайта.

6. Необходимые пять таблиц должны быть разработаны для моделирования предметной области, заданной вариантом. Необходимо придумать эти таблицы, не забывая про здравый смысл.

7. Добавить минимум 2 роли, соответствующих реальным ролям в выбранной предметной области. Добавить пользователей, обладающих этими ролями.

8. Реализовать средства регистрации пользователей, аутентификации.

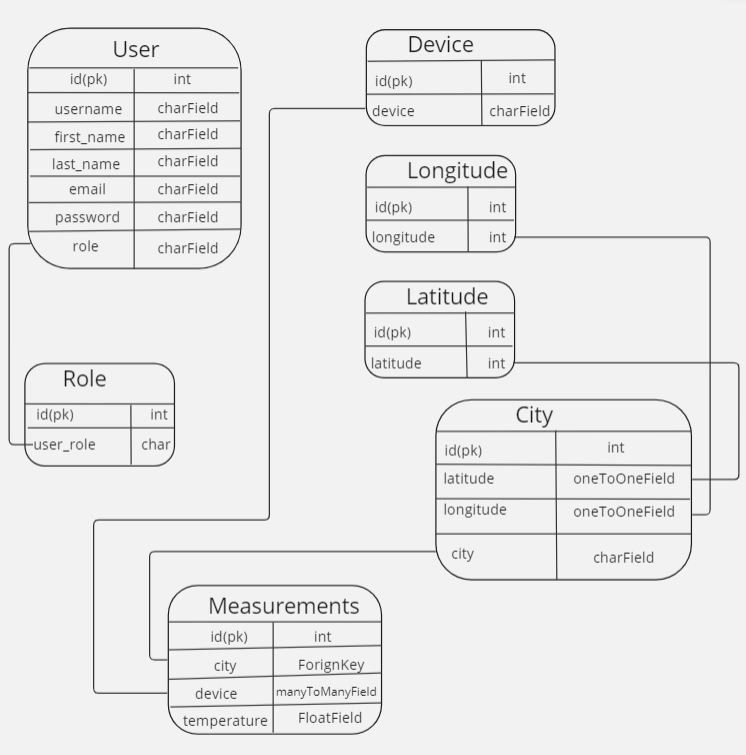
9. Добавить средства авторизации и разграничения прав доступа разным ролям к разным сущностям.

**Описание проекта**

1. Пользователь может зарегистрировать аккаунт на платформе
2. Пользователю необходимо авторизоваться, после авторизации, если он админ – он может добавлять товары, заказы, заказчиков и т.д.
3. На странице таблица учёта, юзер может просмотреть все таблицы, главную таблицу со 3 видами связей и имеет возможность удалить или добавить записи
4. Также пользователь может сменить свои пароль и никнейм
5. Юзер может просмотреть данные об аккаунте, посмотреть дату регистрации и последнего изменения профиля

**Описание реализации**

Для того чтобы начать писать код приложения, была разработана структура базы данных



Дизайн web-приложения был придуман самостоятельно во всех лабораторных работах на фрейме Django (7 – 9 лабы).

В файле models.py были созданы модели таблиц бд.



Рис. 1. Таблицы заказ, тэг, продукт

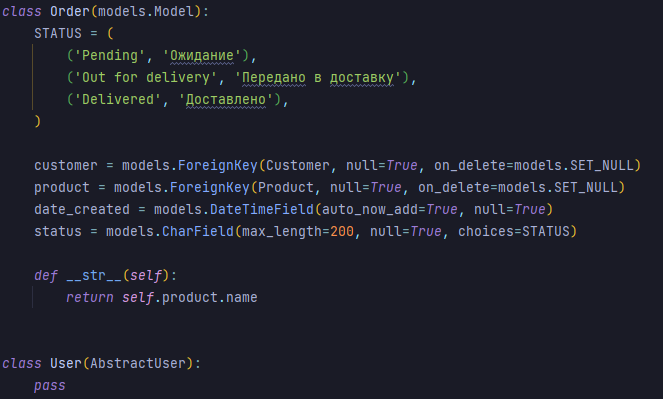


Рис.2. Таблица заказ

В данном файле описываются поля наших таблиц.

В файле forms.py мы описываем формы для ввода данных (Регистрация, авторизация, обновление информации пользователя, создание записи для 7 Лабы)

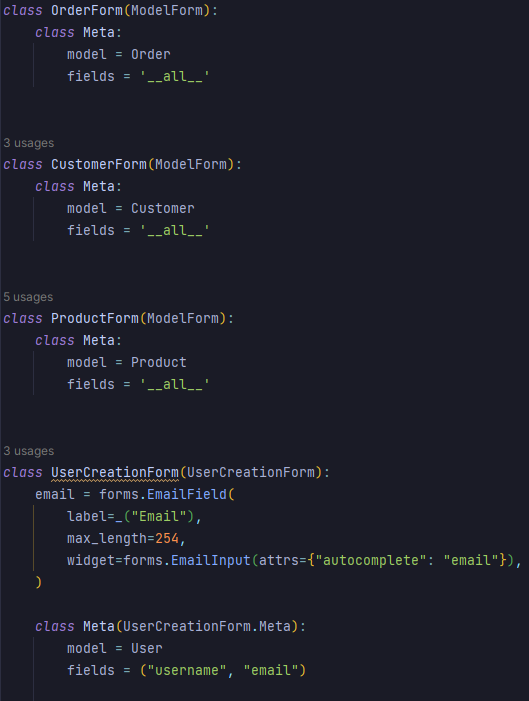


Рис.3 Файл forms.py

В файле urls.py мы указываем по каким ссылкам, какие функции будет срабатывать

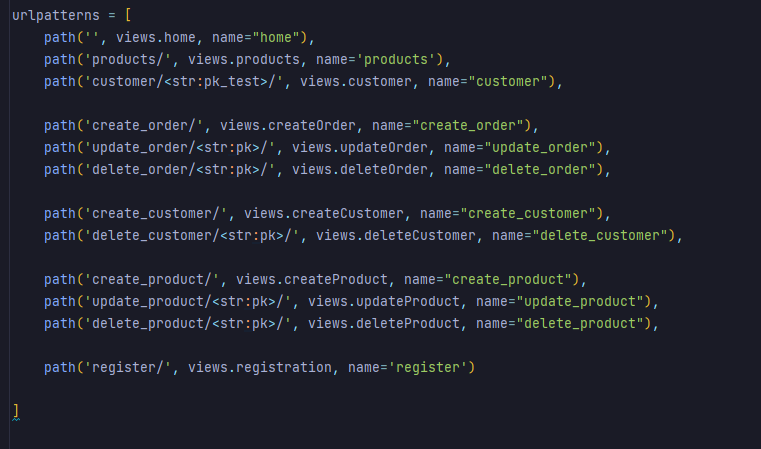


Рис.4 Файл urls

Функции описываются в файле views.py

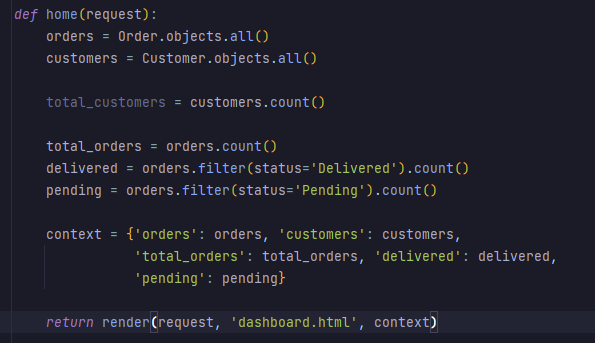


Рис.5 Функция отображения панели учёта

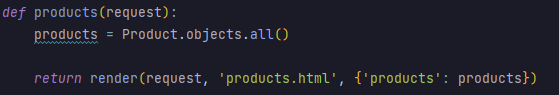


Рис.6. Функция отображения списка продуктов

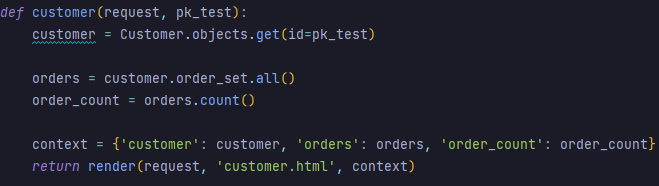


Рис.7. Функция отображения таблицы с заказчиками

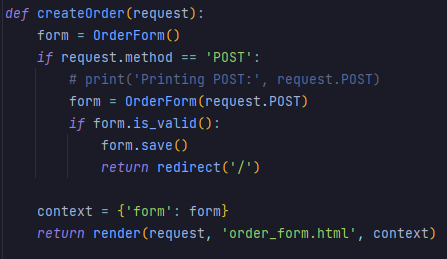


Рис.8. Функция формы создания заказа

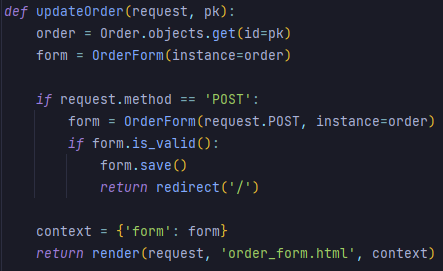


Рис.9. Функция обновления заказа

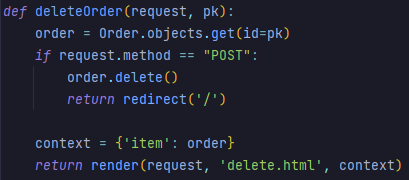


Рис.10. Функция удаления заказа



Рис.11. Функция создания и удаления заказчика

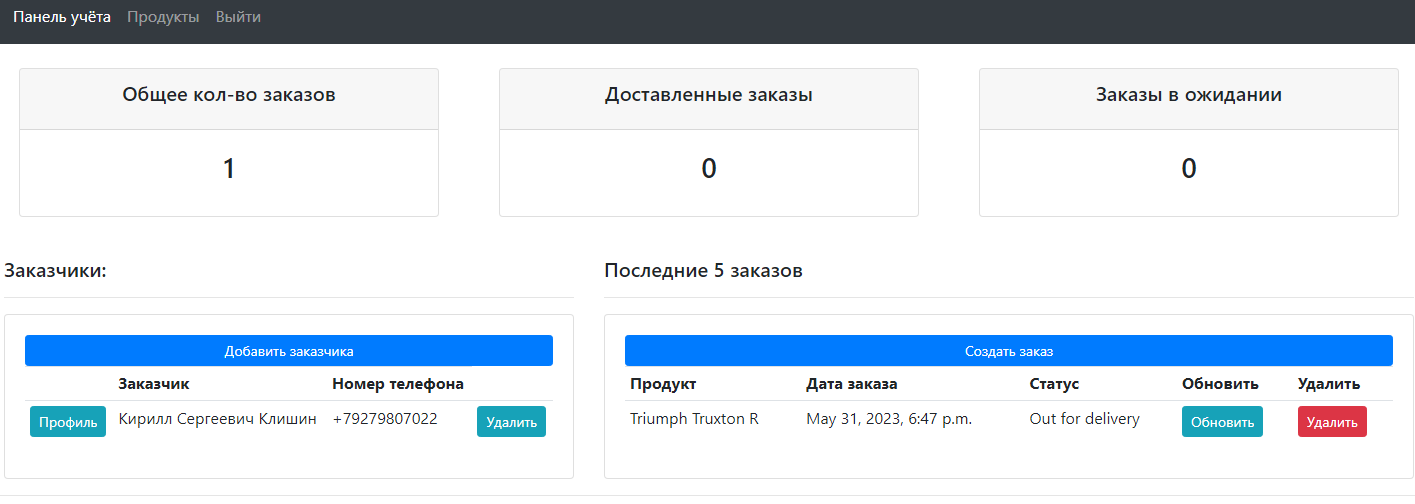


Рис.12. Функция создания и обновления продукта



Рис.13. Функции для удаления продукта и регистрации пользователя

**Тестирование web приложения**

  
Рис.14. Страница с таблицами из 7 лабы

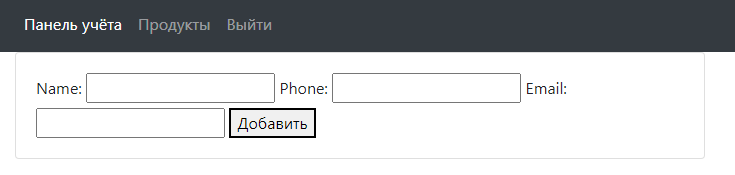


Рис.13. Страница с добавлением записи в таблицу заказчики

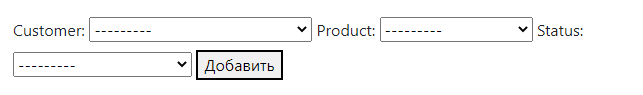


Рис.14. Страница с добавлением записи в таблицу заказы

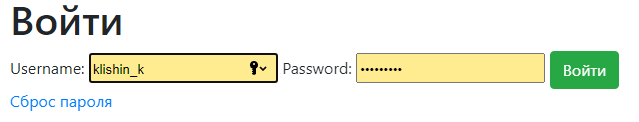


Рис.15. Страница с авторизацией

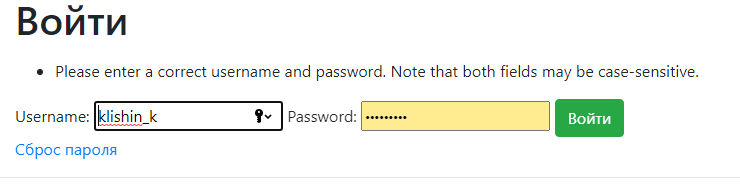


Рис.16. Вывод ошибки



Рис.17. Сайт после авторизации

**Памятка для разработчика:**

1. Скачать проект на официальном **Github** разработчиков по ссылке ниже
2. Запустить проект в PyCharm или любой другой удобной IDE
3. Открыть терминал
4. Установить Django командой «pip install Django»
5. Перейти в директорию «app» командой «cd .\app\»
6. Запустить проект командой «py manage.py runserver»
7. После запуска откройте **url** сайта <http://127.0.0.1:8000/>