Министерство науки и высшего образования Российской Федерации

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

Кафедра «Вычислительная техника»

**Отчет по лабораторной работе №1**

Дисциплина: «Разработка профессиональных приложений»

**Web-разработка**

Вариант №4

Выполнил:

студент группы ИВТАСбд-22

Данилин Н.Г.

Проверил:

преподаватель кафедры

«Вычислительная техника»

Исхаков И.И.

Ульяновск, 2023

**Постановка задачи**

Необходимо реализовать web-интерфейс для просмотра всех объектов, хранящихся в базе данных. Предметная область - из лабораторной работы 3 (4). Для доступа к данным желательно использовать ORM peewee (возможны альтернативные библиотеки). Для реализации web-части - cerrypy. Ajax использовать необязательно, можно сделать все на статических объектах.

Добавление / изменение необходимо реализовать минимум для 1 таблицы БД с внешними ключами.

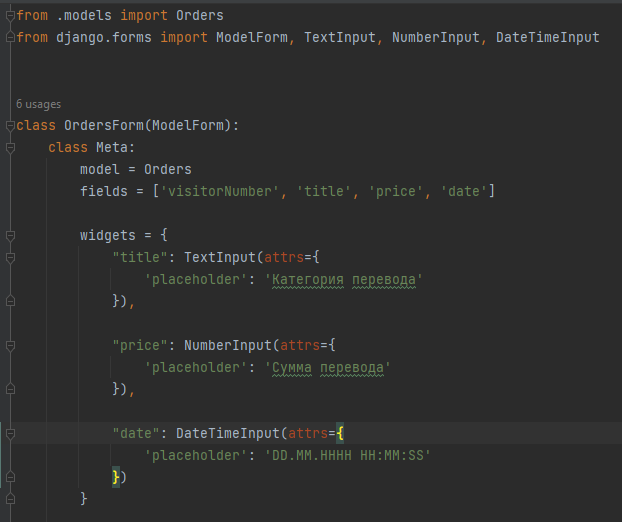
**Описание реализации**

Для реализации Wed-интерфейса

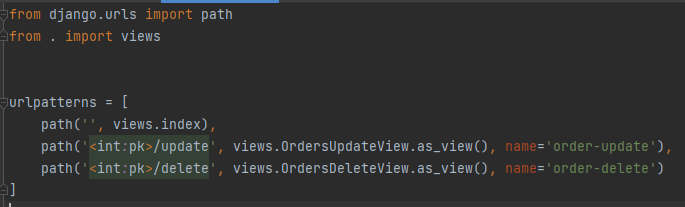
Web-интерфейс для просмотра всех объектов предметной области из лабораторной работы №3, хранящихся в базе данных, был реализован с помощью фреймворка Django. Он использовался как для реализации web-части, так и для доступа к данным. Также были использованы статические объекты. Для начала был создан проект python, на него был установлен фреймворк Django с помощью менеджера пакетов pip, командой pip install Django в терминале PyCharm. После этого командой django-admin startproject lab6, был создан проект Django. Далее командой python manage.py startapp main, было создано, непосредственно, само приложение с которым будет производиться дальнейшая работа. Потом добавляем созданное приложение в настройки проекте, в файл settings.



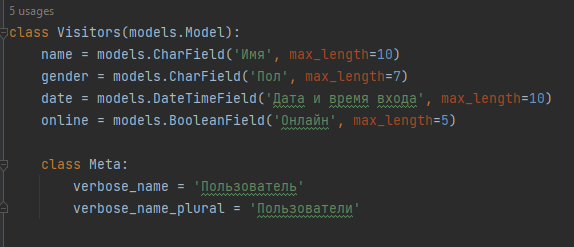
Предварительно в приложение было добавлено два файла forms и urls. В первом файле прописывается форма для ввода данных в нашу базу данных.



Второй файл нужен для того, чтобы отслеживать переходы по страницам приложения



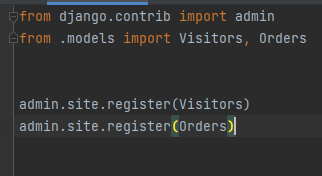
В файле models создаются таблицы для базы данных, задается их структура, каждому полю таблицы задается тип данных.



Для создания внешнего ключа используется метод ForeignKey, в который в качестве параметров передается таблица на который будет ссылаться ключ и вторым параметром передается поведение записи таблицы при удалении записи из исходной таблицы.



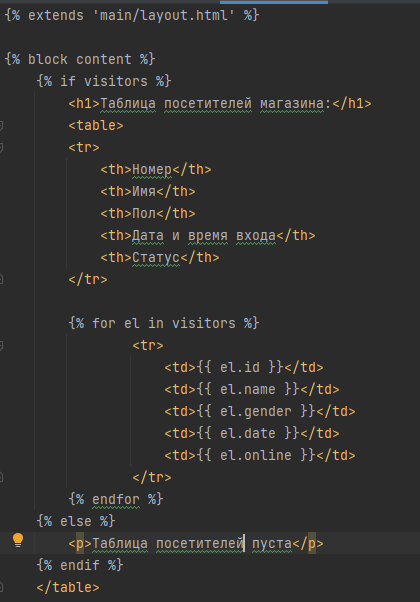
Созданные таблицы нужно зарегистрировать в базе данных, в файле admin.py, чтобы регистрация прошла успешно нужно совершить миграции в терминале с помощью команд: python manage.py makemigrations, python manage.py migrate.



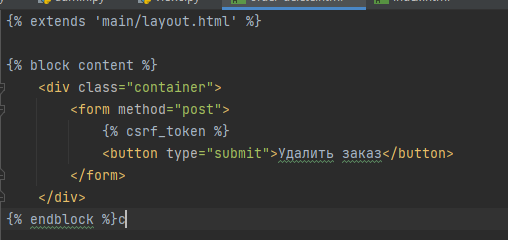
После заполнения формы для добавление записи в БД и нажатия на кнопку добавить на сервер отправляется пост запрос и вызывается метод index из файла views.py, соответственно, в нем данные из формы считываются и при помощи умной формы из файла forms.py сохраняются в БД и отображаются на странице



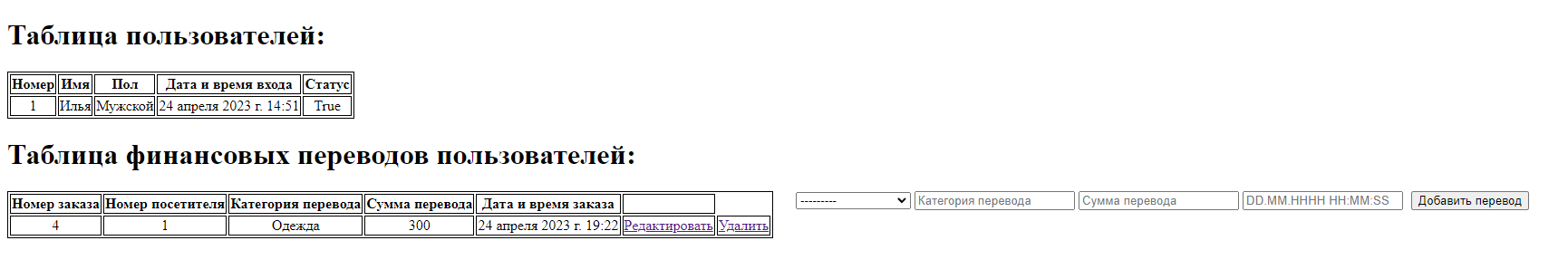
Для отображения данных из базы создавались html-файлы в которых при использовании шаблонизатора Jinja они интегрировались в html разметку.



Редактирование и удаление таблицы производилось путем передачи идентификатора записи из таблицы в url строку, после этого на новой страницы при помощи токена crsf производилось редактирование или удаление



Отображение на странице:



**Описание возникших затруднений**

Web-разработка на python с использованием фреймворка Django, для меня была в новинку, поэтому пришлось изучать с нуля работу с Django.

**Вывод**

В ходе выполнения данной лабораторной работы, был получен опыт web-разработки на языке программирования python с использованием фреймворка Django. Реализованное приложение выполняет все поставленные задачи, поэтому можно считать работу выполненной.