python -m venv venv – создание виртуального окружения

django-admin startproject name – создание нового проекта. Пишется в терминале. В папке проекта с виртуальным окружением создается папка с проектом.

Запуск проекта:

1 Перейти в папку с проектом.

2 python manage.py runserver – запуск проекта

'django.contrib.admin',

Здесь хранятся данные о там как будет происходить администрирование нашего проекта.

Python manage.py migrate – (миграция) команда, которая запускает процесс налаживания взаимодействий приложений с нашей базой данных (db.sqlite3). Если в окне терминале мы не увидели указания взаимодействия приложения с БД, то это значит, что данное приложение ничего не записывает в БД.

Для быстрого примера рассмотрим 'django.contrib.admin'. Пока для понимания будем держать в голове информацию, что есть такая папка *admin*, в которой лежат скрипты *Python*, описывающие настройку и логику работы данного приложения. Исходя из названия также примерно понятно, что данное приложение в нашем проекте будет отвечать за административную часть, а точнее за возможность вносить изменения и следить за ними через специальную панель, с которой мы вскоре познакомимся подробнее.

[AUTHENTICATION AND AUTHORIZATION](http://127.0.0.1:8000/admin/auth/) (за неё отвечает приложение 'django.contrib.auth')

python manage.py createsuperuser — создание логина и пароля суперпользователя

'django.contrib.sites',  
'django.contrib.flatpages'

Здесь мы добавляем приложения, необходимые для работы со статической страницей

Добавляем строку ниже:

SITE\_ID = 1

А в MIDDLEWARE = [...] добавляем:

'django.contrib.flatpages.middleware.FlatpageFallbackMiddleware'

Теперь, когда мы добавили новые приложенияБ нам нужно указать ссылки по которым они будут доступны. Для этого перейдем в файл urls.py и добавим к списку urlpatterns следующее:

from django.urls import path, include

path('pages/', include('django.contrib.flatpages.urls'))

«pages/» это название ссылки, а вторым аргументом идет include, который подгружает url-паттерн.