Для работы с фреймворком Django на Python нужно:

1. Создать папку с проектом
2. Открыл терминал в папке проекта
3. Необходимо создать виртуальное окружение, внутри которого будут установлены все библиотеки и будет работать сам проект. Ввести **python -m venv ll\_env** для создания виртуального окружения. Внутри папке каталога создается папка ll\_env
4. Затем виртуальное окружение нужно активировать (Windows): **ll\_env/Scripts/activate**
5. После появления в начале строки следующего: **(ll\_env)** можно приступать к работе с проектом
6. Установить библиотеку: **pip install Django**
7. Теперь можно создать проект в Джанго: **“django-admin starproject *название\_проекта* .”**
8. После предыдущей команды создается папка с проектом, ее содержимое можно посмотреть: **ls *название\_проекта***
9. Теперь нужно создать БД для проекта: **python manage.py migrate**
10. Каждое изменение базы называется миграцией. Самая первая миграция – это создание БД проекта, остальные – ее изменение
11. Теперь можно убедиться, что проект успешно создан, запустив локальный сервер: **python manage.py runserver *номер\_порта***
12. По умолчанию сервер запускается на порте 8000, его можно открыть в браузере: <http://localhost:8000/> или <http://127.0.0.1:8000/>
13. Поскольку весь проект на Django состоит из миниприложений, то создать такое приложение можно при помощи команды: **python manage.py startapp *название\_приложения***
14. Модель в Django сообщает как работать с данными, которые будут храниться в приложении.
15. В файле models.py приложения можно создавать модели – функциональности
16. Нужно также дописать в кортеж INSTALLED\_APPS файла settings.py название нашего приложения чтобы активировать модели приложения
17. После активации модели необходимо изменить БД, чтобы она смогла хранить данные нашей модели: **python manage.py makemigrations learning\_logs**
18. После изменения БД нужно применить миграцию для автоматического изменения базы данных: **python manage.py migrate**
19. Т.е. изменение модели всегда должно сопровождать командами **makemigrations** (определяет, как изменить БД для хранения информации, связанными с новыми моделями) и **migrate** (выполнить миграцию проекта), чтобы подтянуть БД