**Шаг 1 – Обновление системы:**

Перед установкой любого нового программного обеспечения всегда полезно обновить индекс пакетов вашего сервера. Это гарантирует наличие последних обновлений и исправлений безопасности. Для этого откройте терминал и выполните следующую команду:

 apt-**get** **update**

Эта команда обновляет индекс пакетов, гарантируя, что ваша система знает обо всех последних доступных версиях программного обеспечения.

**Проверка на наличие Python 3:**

Ubuntu 22.04 обычно поставляется с предустановленным Python 3. Чтобы проверить, установлен ли Python 3 в вашей системе, вы можете набрать:

 python3 --version



Если Python 3 установлен, эта команда выведет номер версии. Если он не установлен, или вы хотите установить более новую версию, перейдите к следующему шагу.

**Установка Python 3**:

Если Python 3 не установлен в вашей системе, вы можете легко установить его с помощью менеджера пакетов. Выполните следующую команду:

 apt-**get** install python3



Эта команда указывает менеджеру пакетов Ubuntu' установить Python 3 на вашу систему если не установлен.

**Установка Pip и Venv:**

После установки Python вам необходимо установить pip и venv. pip - это программа установки пакетов Python, которая позволяет устанавливать и управлять дополнительными библиотеками и зависимостями, не включенными в стандартную библиотеку Python. venv - это инструмент для создания изолированных окружений Python, который необходим для бесконфликтного управления зависимостями ваших проектов Python. Чтобы установить эти инструменты, выполните следующие действия:

 sudo apt install python3-pip python3-venv



Эта команда устанавливает pip и venv для Python 3, настраивая вашу среду для разработки на Python.

**Шаг 2: Установка Django и настройка проекта Django**

**Создание новой директории проекта и переход к ней:**

Начните с создания новой директории для вашего проекта Django. Это поможет организовать ваши проекты и файлы. Например, чтобы создать проект с именем 'myDjangoProject', вы должны использовать следующие команды в терминале:

 mkdir ~/myDjangoProject cd ~/myDjangoProject

Это создаст каталог с именем **myDjangoProject** в вашей домашней директории, а затем изменит текущий рабочий каталог на эту новую папку.

**Установка виртуального окружения**:

В каталоге проекта рекомендуется создать виртуальное окружение. Это окружение представляет собой автономный каталог, содержащий установку Python для определенной версии Python, а также ряд дополнительных пакетов. Чтобы создать виртуальное окружение с именем my\_env, выполните следующие действия:

 python3 -m venv my\_env

Эта команда создает новый каталог с именем my\_env в каталоге проекта, который содержит полное, автономное окружение Python.

**Активация виртуальной среды:**

Чтобы начать использовать эту среду, ее необходимо активировать. Выполните:

 source my\_env/bin/activate



После активации вы'увидите имя вашей виртуальной среды (**my\_env**) в приглашении оболочки. Это означает, что вы теперь работаете в виртуальной среде. Чтобы выйти из среды, вы можете выполнить команду deactivate.

**Установка Django:**

Когда виртуальная среда активна, установите Django с помощью **pip**, менеджера пакетов Python':

 pip install django



Эта команда загружает и устанавливает последнюю версию Django в ваше виртуальное окружение.

 django-admin --version



**Запуск нового проекта Django:**

Теперь создайте новый проект Django в текущей директории:

 django-admin startproject djangoproject .

Точка . в конце команды указывает, что проект должен быть создан в текущем каталоге.

**Миграция базы данных**:

Перед первым запуском сервера необходимо перенести базу данных:

 python manage.py migrate



Эта команда устанавливает таблицы базы данных, которые нужны Django для работы.

**Шаг 3: Создание суперпользователя для приложения Django**

**Создание учетной записи суперпользователя**:

Для административного доступа к вашему проекту Django создайте суперпользователя:

 python3 manage.py createsuperuser



Следуйте подсказкам, чтобы задать имя пользователя, электронную почту и пароль для суперпользователя.

**Шаг 4: Запуск приложения Django**

**Конфигурирование разрешенных хостов**:

Перед запуском приложения необходимо настроить Django на разрешение доступа с вашего сервера. Отредактируйте файл **settings.py** вашего проекта Django:

 vim ~/myDjangoProject/djangoproject/settings.py

В этом файле найдите параметр **ALLOWED\_HOSTS** и добавьте IP-адрес вашего сервера:

 ALLOWED\_HOSTS = ['your\_server\_IP']



**Запуск сервера Django:**

Наконец, запустите сервер разработки Django:

 python manage.py runserver your\_server\_ip:8000



Эта команда запускает сервер разработки Django на указанном вами IP и порту 8000. Убедитесь, что этот порт открыт в настройках брандмауэра вашего сервера, чтобы разрешить внешний доступ.



**Заключение**

Вы успешно установили приложение Django на свой сервер. Теперь вы можете получить доступ к интерфейсу администратора Django, добавив **/admin/**к вашему URL.



Это руководство должно было дать исчерпывающую информацию о настройке Django на Ubuntu 22.04, чтобы вы могли начать разработку веб-приложений.