Deploy Django Project en Linux Ubuntu 22.0

Intall Ubuntu

Sudo apt update

Sudo apt upgrade

Instalar postgres (en caso de que se quiera usar esa bd)

sudo apt install postgresql

Empezar postgres

sudo systemctl start postgresql

**onsulta el estado de PostgreSQL** para comprobar si realmente se está ejecutando:

sudo systemctl status postgresql

Copy

**bash**

El servicio debería aparecer como “activo”. Utiliza la tecla [Q] para salir de la pantalla de estado.

Paso 3. Utilizar PostgreSQL[¶](https://www.ionos.es/digitalguide/servidores/configuracion/instalar-postgresql-en-ubuntu-2004/" \l "content-paso-3-utilizar-postgresql" \o "Permalink)

Después de la instalación, PostgreSQL dispone de una **cuenta por defecto** llamada “postgres”. Puedes activarla con el siguiente comando:

sudo -i -u postgres

Copy

**bash**

Ubuntu se comporta como si hubieras iniciado sesión con la cuenta de PostgreSQL, es a partir de entonces cuando puedes **acceder al prompt del SGBD**.

psql

Copy

**bash**

Una vez estés trabajando en el entorno de PostgreSQL, utiliza el siguiente comando para mostrar qué **conexión está activa actualmente**:

\conninfo

Copy

**bash**

Utiliza el siguiente comando para **abandonar el entorno de PostgreSQL**:

\q

Copy

**bash**

Paso 4. Crear bases de datos[¶](https://www.ionos.es/digitalguide/servidores/configuracion/instalar-postgresql-en-ubuntu-2004/#content-paso-4-crear-bases-de-datos)

El sistema ha creado automáticamente una base de datos correspondiente al usuario “postgres”. Sin embargo, tu servidor PostgreSQL puede trabajar con varias bases de datos al mismo tiempo. Crea una **nueva base de datos** con el siguiente comando, en el cual puedes cambiar “example” por el nombre que quieras:

CREATE DATABASE example;

Copy

**bash**

No olvides de poner el punto y coma al final del comando.

Utiliza el siguiente comando para mostrar **las bases de datos que has creado**:

\list

**Crear user**

CREATE USER nicolas WITH PASSWORD ‘password';

**Grant Privileges**

GRANT ALL PRIVILEGES ON DATABASE nombredatabase TO nombreuser

**Crear carpeta de proyecto**

Mkdir Project

**Ir a la carpeta e instalar dependencias pip, venv, Wheel, dev**

Sudo apt install pithon3-pip

Sudo apt install pithon3-venv

Sudo apt install pithon3-wheel

Sudo apt install pithon3-dev

**Clonar el repositorio con git**

Get clone dirección de repositorio aquí

Va a pedir usuario y contraseña

**Crear el entorno de trabajo venv**

Dentro de la carpeta donde está el proyecto (generalmente a la misma altura que manage.py)

Python –m venv env

**Instalar todas las dependencias.**

Pip install –r requirements.txt

**Migrar las tablas**

Python manage.py makemigrations

Python manage.py migrate

Cambiar dentro de settings.py el ALLOWED\_HOSTS = [‘\*’]

Probar runserver.

Python manage.py runserver 0.0.0.0:8000

Deberíamos poder ingresar al proyecto por la ip del server que iniciamos.