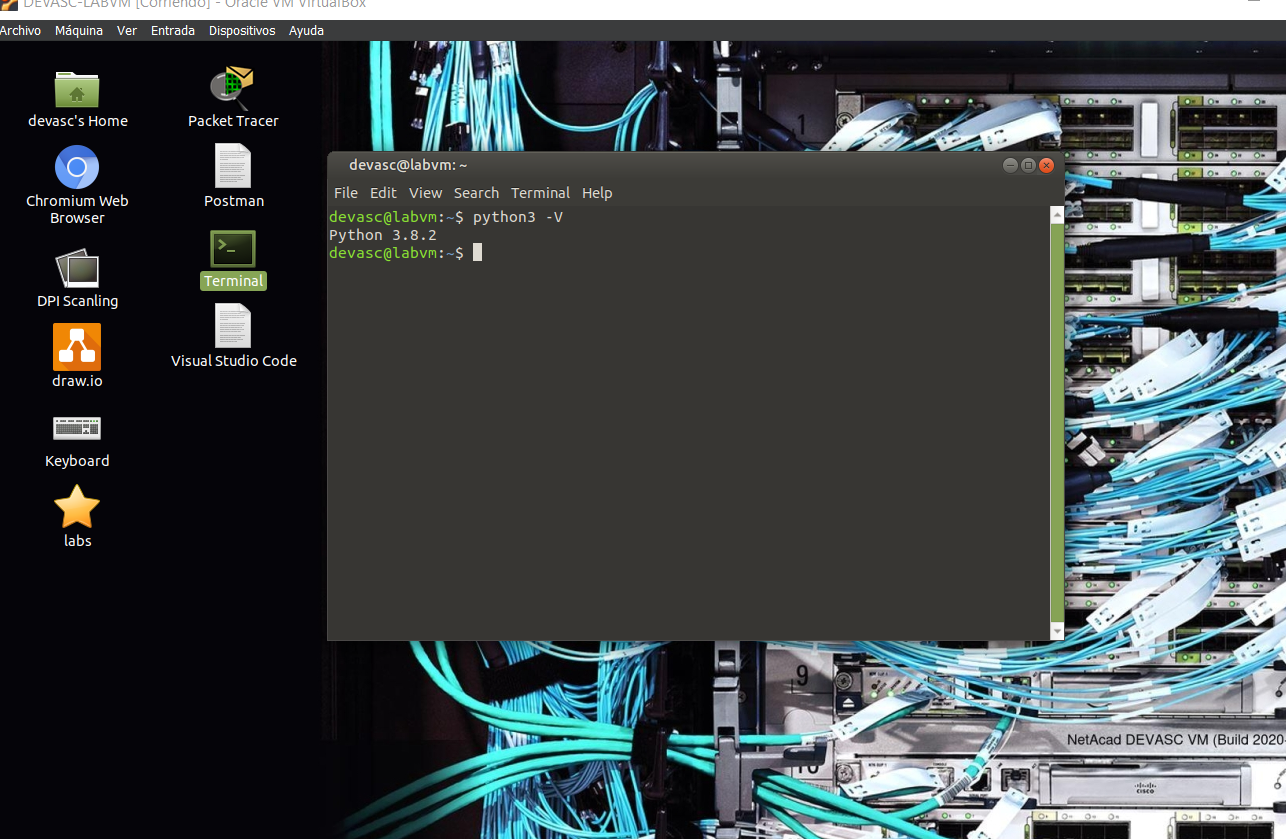
**Explorar las herramientas de desarrollo de Python**

**Paso 1: Revisión de la instalación de Python**

1. Iniciamos normalmente nuestra Máquina virtual DEVASC, y verificaremos si tenemos instalado Python en nuestra máquina virtual con el siguiente comando
2. Una captura de pantalla de una computadora

   Descripción generada automáticamentePara ver el directorio del entorno local de Python, utilice el comando

**Parte 2: Entornos virtuales PIP y Python**

**Paso 1: Cree un entorno virtual de Python 3**

1. Dentro de una máquina virtual de DEVASC, cambiaremos al directorio **labs/devnet-src/python/**
2. Introduzca el siguiente comando para usar la herramienta **venv** para crear un entorno virtual de Python 3 con el nombre **devfun.** El switch ­**-m** le dice a Python que ejecute el módulo **venv.**

Texto

Descripción generada automáticamente

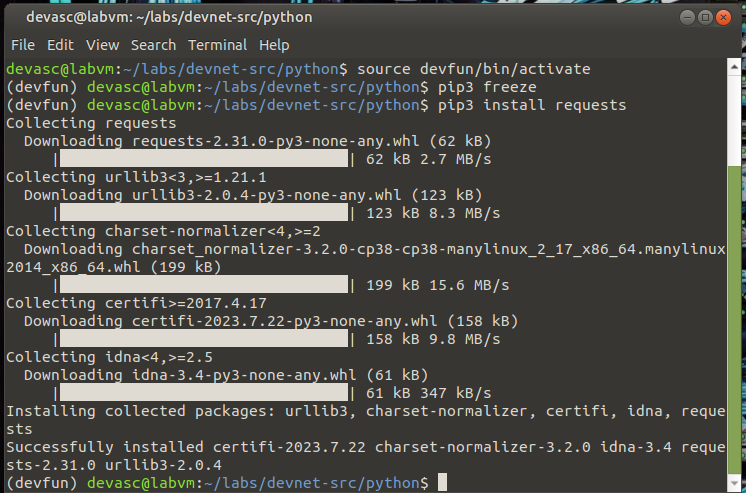
**Paso 2: Activar y probar el entorno virtual en Python 3**

* 1. Activar el entorno virtual. El "prompt" cambia para indicar el nombre del entorno en el que está trabajando actualmente, que es (**devfun)** en este ejemplo. Ahora, cuando se utiliza el comando **pip3 install** aquí, el sistema solo instalará paquetes para el entorno virtual activo.

Texto

Descripción generada automáticamente

* 1. Ejecute el comando **pip3 freeze** para verificar que no hay paquetes adicionales de Python instalados actualmente en el entorno **devfun.**
  2. Ahora instale el paquete **requests** de Python dentro del entorno **devfun**.



* 1. Vuelva a introducir el comando **pip3 freeze** para ver los paquetes ahora instalados en el entorno **devfun.**

Texto

Descripción generada automáticamente

* 1. Para desactivar el entorno virtual y volver al sistema, ingrese el comando **deactivate.**

Texto

Descripción generada automáticamente

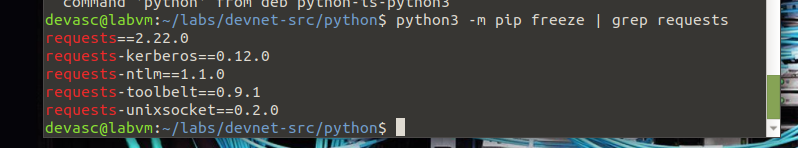
**Paso 3: Compruebe los paquetes actuales instalados en el entorno del sistema**

1. Introduzca el comando **python3 -m pip freeze** para ver qué paquetes están instalados en el entorno del sistema. Nota: Debido a que Python3 se invoca solo se usa pip en lugar de pip3.

**Texto

Descripción generada automáticamente**

1. Si desea encontrar rápidamente la versión de un paquete instalado, canalice la salida al comando **grep.** Introduzca lo siguiente para ver la versión del paquete de solicitudes actualmente instalado.



**Parte 3: Compartir su entorno virtual**

La salida del comando **pip3 freeze** está en un formato específico por una razón. Puede usar todas las dependencias enumeradas para que otras personas que quieran trabajar en el mismo proyecto que usted puedan obtener el mismo entorno que el suyo.

Crearemos un archivo de requisitos, requirements.txt, utilizando el comando **pip3 freeze > requirements.txt.** Entonces otro desarrollador puede desde otro entorno virtual activado, usar este comando **pip3 install -r requirements.txt** para instalar los paquetes requeridos por el proyecto.

1. Reactivar el entorno virtual **devfun**

**Texto

Descripción generada automáticamente**

1. Envíe la salida del comando **pip3 freeze** a aun archivo de texto llamado **requirements.txt.** Desactivaremos el entorno virtual **devfun.** Usaremos el comando **ls** para ver el archivo **requirements.txt** que se encuentra en el directorio **/python.**

**Texto

Descripción generada automáticamente**

1. Crearemos un nuevo entorno virtual de Python llamado **devnew** y lo activaremos. Luego utilizando el comando **pip3 install -r requirements.txt** instalaremos los mimos paquetes que están instalados en el entorno virtual **devfun.**

**Texto

Descripción generada automáticamente**

1. Confirmaremos con el comando **pip3 freeze** en el entorno **devnew,** las independecias.

**Texto

Descripción generada automáticamente**

1. Finalmente desactivamos el entorno virtual **devnew** con el comando **$ deactivate**