



## Práctica de Laboratorio 2b

### Explorar las herramientas de desarrollo de Python

#### Objetivos

Parte 1: Iniciar la Máquina Virtual (Virtual Machine) DEVASC

Parte 2: Revisar la instalación de Python

Parte 3: Entornos virtuales PIP y Python

Parte 4: Compartir su entorno virtual

#### Aspectos básicos/Situación.

Este curso tiene algunas expectativas básicas para los estudiantes, que tienen algunos antecedentes en Python y alguna experiencia práctica con la programación. La práctica de laboratorio con Python le da a sus dedos la "memoria muscular" para trabajar de manera más eficiente en Python. Sin embargo, es relevante que sepa cómo mantener un entorno de desarrollo de Python.

En este laboratorio, usted revisará la instalación de Python, PIP y entornos virtuales de Python.

#### Recursos necesarios

- Una computadora con el sistema operativo de su elección
- Virtual Box o VMWare
- Máquina virtual (Virtual Machine) DEVASC

#### Instrucciones

##### Parte 1: Ejecutar la Máquina virtual (Virtual Machine) DEVASC

Si no ha completado el **Laboratorio -Instalar el entorno de la máquina virtual**, hágalo ahora. Si ya ha completado el laboratorio, ejecute la **Máquina virtual (Virtual Machine) DEVASC**

##### Parte 2: Revisión de la instalación de Python

Estos comandos le proporcionan métodos básicos para aprender más sobre el entorno local de Python.

- a. En la máquina virtual DEVASC, puede verificar la versión de Python ya instalada con el comando **python3 -V**.

```
devasc@labvm:~$ python3 -V
Python 3.8.2
devasc@labvm:~$
```

- b. Para ver el directorio del entorno local de Python, utilice el comando **which python3**.

```
devasc @labvm: ~$ which python3
/usr/bin/python3
devasc@labvm:~$
```

### Parte 3: Entornos virtuales PIP y Python

PIP significa Pip Instala Paquetes (Pip Installs Packages). Muchas personas primero aprenden acerca de PIP y comienzan a usar los comandos **pip3 install** en la instalación de Python en todo el sistema. Cuando se ejecuta el comando **pip3 install** en su sistema, puede introducir dependencias competitivas en la instalación del sistema que puede o no desear para todos los proyectos de Python. Por lo tanto, la mejor práctica es habilitar un entorno virtual de Python. A continuación, instale solo los paquetes que son necesarios para el proyecto en el entorno virtual. De esta manera, usted sabe exactamente qué paquetes están instalados en una configuración determinada. Puede cambiar esas dependencias de paquetes fácilmente al cambiar a un nuevo entorno virtual, y no romper ni causar problemas debido a versiones de software competitivas.

Para instalar un entorno virtual de Python, utilice la herramienta **venv** en Python 3 y luego active el entorno virtual, como se muestra en los pasos siguientes.

#### Paso 1: Cree un entorno virtual de Python 3.

- Dentro de la máquina virtual de DEVASC, cambie el directorio **labs/devnet-src/python**.

```
devasc @labvm: ~$ cd labs/devnet-src/python/
devasc@labvm:~/labs/devnet-src/python$
```

- Introduzca el siguiente comando para usar la herramienta **venv** para crear un entorno virtual Python 3 con el nombre **devfun**. El switch **-m** le dice a Python que ejecute el módulo **venv**. El nombre es elegido por el programador.

```
devasc @labvm: ~/labs/devnet-src/python$ python3 -m venv devfun
devasc@labvm:~/labs/devnet-src/python$
```

#### Paso 2: Activar y probar el entorno virtual de Python 3.

- 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.

```
devasc @labvm: ~/labs/devnet-src/python$ fuente devfun/bin/activate
(devfun) devasc @labvm: ~/labs/devnet-src/python$
```

- Ejecute el comando **pip3 freeze** para verificar que no hay paquetes adicionales de Python instalados actualmente en el entorno **devfun**.

```
(devfun) devasc @labvm: ~/labs/devnet-src/python$ pip3 freeze
(devfun) devasc @labvm: ~/labs/devnet-src/python$
```

- Ahora instale el paquete **requests** de Python dentro del entorno **devfun**.

```
(devfun) devasc @labvm: ~/labs/devnet-src/python$ pip3 install requests
Recopilar solicitudes
  Descargando peticiones-2.23.0-py2.py3-none-any.whl (58 kB)
    |████████████████████████████████████████| 58 kB 290 Kb/s
;Recolectando urllib3!=1.25.0,!<1.26, >=1.25.1 ,=1.21.1
  Descargando urllib3-1.25.9-py2.py3-none-any.whl (126 kB)
    |████████████████████████████████████████| 126 kB 1.7 MB/s
<3, >Recogiendo idna=2.5
  Descargando idna-2.9-py2.py3-none-any.whl (58 kB)
```

```
| ████████████████████████████████████████████████████████████████ | 58 kB 18.3 MB/s  
Recopilación de certificación>=2017.4.17  
  Descargar certifi-2020.4.5.1-py2.py3-none-any.whl (157 kB)  
    | ████████████████████████████████████████████████████████████████ | 157 kB 19,8 MB/s  
<4, >Recolectando chardet=3.0.2  
  Descargando chardet-3.0.4-py2.py3-none-any.whl (133 kB)  
    | ████████████████████████████████████████████████████████████████ | 133 kB 59.2 MB/s  
Instalación de paquetes recopilados: urllib3, idna, certifi, chardet, solicitudes  
Se instaló correctamente certifi-2020.4.5.1 chardet-3.0.4 idna-2.9 requests-2.23.0  
urllib3-1.25.9  
  
(devfun) devasc @labvm: ~/labs/devnet-src/python$
```

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

**Nota:** Su lista de paquetes y números de versión puede ser ligeramente diferente.

```
(devfun) devasc @labvm: ~/labs/devnet-src/python$ pip3 freeze
certifi==2020.4.5.1
chardet==3.0.4
idna==2.10
requests==2.24.0
urllib3==1.25.9
(devfun) devasc @labvm: ~/labs/devnet-src/python$
```

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

```
(devfun) devasc @labvm: ~/labs/devnet-src/python$ deactivate
devasc@labvm:~/labs/devnet-src/python$
```

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

- a. Introduzca el comando **python3 -m pip freeze** para ver qué paquetes están instalados en el entorno del sistema.

Nota: Debido a que Python 3 se invoca con el siguiente comando, solo usa **pip** en lugar de **pip3**.

```
devasc @labvm: ~/labs/devnet-src/python$ python3 -m pip freeze
ansible==2.9.4
apache-libcloud==2.8.0
appdirs==1.4.3
argcomplete==1.8.1
astroid==2.3.3
bcrypt==3.1.7
blinker==1.4
certifi==2019.11.28
<output omitted>
xmltodict==0.12.0
zipp==1.0.0
devasc@labvm:~/labs/devnet-src/python$
```

- b. 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.

```
devasc @labvm: ~/labs/devnet-src/python$ python3 -m pip freeze | grep requests
requests==2.22.0
requests-kerberos==0.12.0
```

```
requests-ntlm==1.1.0
requests-toolbelt==0.9.1
requests-unixsocket==0.2.0
devasc@labvm: ~/labs/devnet-src/python$
```

### Parte 4: 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.

Un desarrollador puede crear un archivo de requisitos, como **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.

- a. Reactivar el entorno virtual **devfun**.

```
devasc @labvm: ~/labs/devnet-src/python$ fuelle devfun/bin/activate
(devfun) devasc @labvm:
```

- b. Envíe la salida del comando **pip3 freeze** a un archivo de texto llamado **requirements.txt**.

```
(devfun) devasc @labvm: ~/labs/devnet-src/python$ pip3 freeze > requirements.txt
```

- c. Desactive el entorno virtual **devfun**. Puede usar el comando **ls** para ver que el archivo **requirements.txt** que está en el directorio **/python**.

```
(devfun) devasc @labvm: ~/labs/devnet-src/python$ deactivate
devasc @labvm: ~/labs/devnet-src/python$ ls
devfun file-access-input.py if-acl.py requirements.txt
devices.txt file-access.py if-vlan.py while-loop.py
devnew hello-world.py person-info.py
```

- d. Cree y active un nuevo entorno virtual de Python llamado **devnew**.

```
devasc @labvm: ~/labs/devnet-src/python$ python3 -m venv devnew
devasc @labvm: ~/labs/devnet-src/python$ fuelle devnew/bin/activate
(devnew) devasc @labvm: ~/labs/devnet-src/python$
```

- e. Utilice el comando **pip3 install -r requirements.txt** para instalar los mismos paquetes que están instalados en el entorno virtual **devfun**.

```
(devnew) devasc @labvm: ~/labs/devnet-src/python$ pip3 install -r requirements.txt
Certificado de recolección==2020.4.5.1
  Usando certifi-2020.4.5.1-py2.py3-none-any.whl (157 kB)
Recolectando chardet==3.0.4
  Usando chardet-3.0.4-py2.py3-none-any.whl en caché (133 kB)
Recolectando IDNA ==2.9
  Usando idna-2.9-py2.py3-none-any.whl almacenado en caché (58 kB)
Requisito completado: pkg-resources==0.0.0 in ./devnew/lib/python3.8/site-packages (de
-r requirements.txt (línea 4)) (0.0.0)
Recopilación de solicitudes ==2.23.0
  Usando solicitudes almacenadas en caché 2.23.0-py2.py3-none-any.whl (58 kB)
Recolectando urllib3==1.25.9
  Usando urllib3-1.25.9-py2.py3-none-any.whl (126 kB)
```

```
Instalación de paquetes recopilados: certifi, chardet, idna, urllib3, requests  
Se instaló correctamente certifi-2020.4.5.1 chardet-3.0.4 idna-2.9 requests-2.23.0  
urllib3-1.25.9
```

```
(devnew) devasc @labvm: ~/labs/devnet-src/python$
```

- f. Al ingresar **pip3 freeze** en el entorno **devnew**, debería ver el siguiente resultado.

```
(devnew) devasc @labvm: ~/labs/devnet-src/python$ pip3 freeze  
certifi==2020.4.5.1  
chardet==3.0.4  
idna==2.9  
pkg-resources==0.0.0  
requests==2.23.0  
urllib3==1.25.9  
(devnew) devasc @labvm: ~/labs/devnet-src/python$
```

- g. Desactive el entorno virtual **devnew**.

```
(devnew) devasc @labvm: ~/labs/devnet-src/python$ deactivate  
devasc@labvm:~/labs/devnet-src/python$
```