Travaux pratiques - Explorer les outils de développement Python

# Objectifs

Partie 1 : Lancer la DEVASC VM

Partie 2 : Révision de l'installation Python

Partie 3 : Environnements virtuels PIP et Python

Partie 4 : Partage de votre environnement virtuel

# Contexte/scénario

Ce cours a quelques attentes de base pour les étudiants, qu'ils ont une certaine expérience en Python et une certaine expérience pratique de la programmation. La pratique en laboratoire avec Python donne à vos doigts la "mémoire musculaire" pour travailler plus efficacement en Python. Cependant, au départ, il est important pour vous de savoir comment maintenir un environnement de développement Python.

Dans ce TP, vous passez en revue l'installation Python, PIP et les environnements virtuels Python.

# Ressources requises

* 1 PC avec système d'exploitation de votre choix
* Boîte virtuelle ou VMWare
* Machine virtuelle DEVASC

# Instructions

## Lancez la machine virtuelle DEVASC

Si vous n'avez pas encore terminé **TP - Installer l'environnement de laboratoire de machines virtuelles**, faites-le maintenant. Si vous avez déjà terminé ce laboratoire, lancez la **machine virtuelle DEVASC**.

## Révision de l'installation Python

Ces commandes vous donnent des méthodes de base pour en savoir plus sur l'environnement Python local.

* + - 1. Dans la machine virtuelle DEVASC, vous pouvez vérifier la version de Python déjà installée avec la commande **python3 -V**.

devasc@labvm:~$ **python3 -V**

Python 3.8.2

devasc@labvm:~$

* + - 1. Pour afficher le répertoire de l'environnement Python local, utilisez la commande **dont python3**.

devasc@labvm:~$ **which python3**

/usr/bin/python3

devasc@labvm:~$

## Environnements virtuels PIP et Python

PIP est l'acronyme de Pip Installs Packages. Beaucoup de gens apprennent d'abord à propos de PIP et commencent à utiliser les commandes **pip3 install** sur l'installation Python à l'échelle du système. Lorsque vous exécutez la commande **pip3 install** sur votre système, vous pouvez introduire des dépendances concurrentes dans votre installation système que vous pouvez ou non souhaiter pour tous les projets Python. Par conséquent, la meilleure pratique consiste à activer un environnement virtuel Python. Installez ensuite uniquement les packages nécessaires au projet dans cet environnement virtuel. De cette façon, vous savez exactement quels paquets sont installés dans un paramètre donné. Vous pouvez facilement changer ces dépendances de paquets lorsque vous passez à un nouvel environnement virtuel, et ne pas casser ou causer de problèmes en raison de versions concurrentes de logiciels.

Pour installer un environnement virtuel Python, utilisez l'outil **venv** dans Python 3, puis activez l'environnement virtuel, comme illustré dans les étapes suivantes.

### Créez un environnement virtuel Python 3.

* + - 1. À l'intérieur de la machine virtuelle DEVASC, modifiez le répertoire **labs/devnet-src/python**.

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

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

* + - 1. Entrez la commande suivante pour utiliser l'outil **venv** pour créer un environnement virtuel Python 3 avec le nom **devfun**. Le commutateur **-m** indique à Python d'exécuter le module **venv**. Le nom est choisi par le programmeur.

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

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

### Activez et testez l'environnement virtuel Python 3.

* + - 1. Activez l'environnement virtuel. L'invite change pour indiquer le nom de l'environnement dans lequel vous travaillez actuellement, qui est **devfun** dans cet exemple. Maintenant, lorsque vous utilisez le formulaire de commande **pip3 install** ici, le système installe uniquement les paquets pour l'environnement virtuel actif.

devasc@labvm:~/labs/devnet-src/python$ **source devfun/bin/activate**

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

* + - 1. Exécutez la commande **pip3 freeze** pour vérifier qu'aucun package Python supplémentaire n'est actuellement installé dans l'environnement **devfun**.

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

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

* + - 1. Installez maintenant le paquet de **requests** Python dans l'environnement **devfun**.

(devfun) devasc@labvm:~/labs/devnet-src/python$ **pip3 install requests**

Collecting requests

Downloading requests-2.23.0-py2.py3-none-any.whl (58 kB)

|████████████████████████████████| 58 kB 290 kB/s

Collecting urllib3!=1.25.0,!=1.25.1,<1.26,>=1.21.1

Downloading urllib3-1.25.9-py2.py3-none-any.whl (126 kB)

|████████████████████████████████| 126 kB 1.7 MB/s

Collecting idna<3,>=2.5

Downloading idna-2.9-py2.py3-none-any.whl (58 kB)

|████████████████████████████████| 58 kB 18.3 MB/s

Collecting certifi>=2017.4.17

Downloading certifi-2020.4.5.1-py2.py3-none-any.whl (157 kB)

|████████████████████████████████| 157 kB 19.8 MB/s

Collecting chardet<4,>=3.0.2

Downloading chardet-3.0.4-py2.py3-none-any.whl (133 kB)

|████████████████████████████████| 133 kB 59.2 MB/s

Installing collected packages: urllib3, idna, certifi, chardet, requests

Successfully installed 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$

* + - 1. Entrez de nouveau la commande **pip3 freeze** pour voir les paquets installés dans l'environnement **devfun**.

**Remarque** : Votre liste de paquets et de numéros de version peut être légèrement différente.

(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$

* + - 1. Pour désactiver l'environnement virtuel et revenir à votre système, entrez la commande **deactivate**.

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

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

### Vérifiez les packages actuels installés dans l'environnement système.

* + - 1. Entrez la commande system wide **python3 -m pip freeze** pour voir quels paquets sont installés dans l'environnement système.

Remarque : Parce que Python 3 est appelé avec la commande suivante, vous n'utilisez que **pip** au lieu 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$

* + - 1. Si vous voulez trouver rapidement la version d'un paquet installé, dirigez la sortie vers la commande **grep**. Entrez ce qui suit pour voir la version du package de requêtes actuellement installé.

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$

## Partage de votre environnement virtuel

La sortie de la commande **pip3 freeze** est dans un format spécifique pour une raison. Vous pouvez utiliser toutes les dépendances répertoriées afin que les autres personnes qui veulent travailler sur le même projet que vous puissent obtenir le même environnement que le vôtre.

Un développeur peut créer un fichier d'exigences, tel que **requirements.txt**, à l'aide de la commande **pip3 freeze > requirements.txt** . Ensuite, un autre développeur peut, à partir d'un autre environnement virtuel activé, utiliser cette commande **pip3 install -r requirements.txt** pour installer les paquets requis par le projet.

* + - 1. Réactivez l'environnement virtuel **devfun**.

devasc@labvm:~/labs/devnet-src/python$ **source devfun/bin/activate**

(devfun) devasc@labvm:

* + - 1. Envoyez la sortie de la commande **pip3 freeze** dans un fichier texte appelé **requirements.txt**.

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

* + - 1. Désactivez l'environnement virtuel **devfun**. Vous pouvez utiliser la commande **ls** pour voir que le fichier **requirements.txt** se trouve dans le répertoire /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

* + - 1. Créez et activez un nouvel environnement virtuel Python appelé **devnew**.

devasc@labvm:~/labs/devnet-src/python$ **python3 -m venv devnew**

devasc@labvm:~/labs/devnet-src/python$ **source devnew/bin/activate**

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

* + - 1. Utilisez la commande **pip3 install -r requirements.txt** pour installer les mêmes paquets que ceux installés dans l'environnement virtuel **devfun**.

(devnew) devasc@labvm:~/labs/devnet-src/python$ **pip3 install -r requirements.txt**

Collecting certifi==2020.4.5.1

Using cached certifi-2020.4.5.1-py2.py3-none-any.whl (157 kB)

ChardetCollecting chardet==3.0.4te==3.0.4

Using cached chardet-3.0.4-py2.py3-none-any.whl (133 kB)

Collecting idna==2.9

Using cached idna-2.9-py2.py3-none-any.whl (58 kB)

Requirement already satisfied: pkg-resources==0.0.0 in ./devnew/lib/python3.8/site-packages (from -r requirements.txt (line 4)) (0.0.0)

Collecting requests==2.23.0

Using cached requests-2.23.0-py2.py3-none-any.whl (58 kB)

Collecting urllib3==1.25.9

Using cached urllib3-1.25.9-py2.py3-none-any.whl (126 kB)

Installing collected packages: certifi, chardet, idna, urllib3, requests

Successfully installed 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$

* + - 1. Lorsque vous entrez **pip3 freeze** dans l'environnement **devnew**, vous devriez voir la sortie suivante.

(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$

* + - 1. Désactivez l'environnement virtuel **devnew**.

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

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

Fin du document