Pip, Virtualenv, Pipenv Cheat Sheet

by DjackH via cheatography.com/44051/cs/13063/

Virtualenv Installation

`pip install virtualenv`	installer VirtualEnv
`pip install --upgrade virtualenv`	Mise à jour de Virtualenv

Virtualenv Créer et supprimer un environnement

`virtualenv -p python3 env`	Placez-vous dans votre dossier de travail et créer un environnement virtuel.
`(env) $ deactivate` `$ rm -rf env`	Commencez par le désactiver puis supprimez son répertoire

Virtualenv activer et désactiver un environnement

`source env/bin/activate`	Activer l'environnement
`deactivate`	Désctiver l'environnement

Désactiver le suivi Git

Il faut impérativement désactiver le suivi de modifications de votre environnement virtuel.
Pour cela, créez un fichier **.gitignore** et ajoutez à l'intérieur la ligne suivante :

```
1 env/
2 autre fichier à ignorer
3 ...
```

pip

pip (cont)

`pip install -r lib.txt`	...importer cette liste
`pip bundle <no m_d u_b und le>.py bundle -r lib.txt`	Crée un gr zip q conti toute les dépe ance
`pip install <no m_d u_b und le>.py bundle`	...installer les li

Pipenv

pipenv reprend les idées de pip, virtualenv, pew et même quelques trucs de npm, yarn, cargo, et essaye d'appliquer tout ça à Python. pipenv permet donc d'installer des packages Python, d'isoler cette installation et de la rendre reproductible. En effet, contrairement à la concurrence, pipenv permert :
La gestion du virtualenv est automatique et transparente
Les paquets installés sont sauvegardés dans des fichiers de config, encore une fois de manière automatique et transparente.
Les fichiers de config distinguent les dépendances de prod et de dev, et incluent les versions des sous-dépendances.

`python -m pip install pip --upgrade --user`	Mise à jour de pip, mais juste au niveau utilisateur pour pas casser le system
`python -m pip install pipenv --user`	Installation de pipenv

A moins d'être sous une Debian like type Ubuntu (qui demande un apt install de python-pip avant), tout le monde a pip installé avec une version moderne de Python. Voilà, vous devriez avoir la commande pipenv disponible, ou pour ceux qui ont un système mal configuré, python -m pipenv.

Pip est un système de gestion de paquets utilisé pour installer et gérer des librairies écrites en Python. Vous pouvez trouver une grande partie de ces librairies dans le Python Package Index (ou PyPI). Pip empêche les installations partielles en annonçant toutes les exigences avant l'installation.

`sudo apt-get install python 3-pip`	Installer pip python 3
`pip install django`	Pip permet d'installer une librarie aussi facilement que cela
`pip install django==1.7`	Choisir la version qui vous intéresse
`pip uninstall django`	Supprimer une lib
`pip install django --upgrade`	Mettre à jour une lib
`pip install django==1.6 --upgrade`	Downgrader une version
`pip search django`	Rechercher une nouvelle lib
`pip list --outdated`	Indique quels lib n'est plus à jour
`pip freeze`	Affiche toutes les lib installées et leur version
`pip freeze > lib.txt`	...exportez cette liste

By **DjackH**
cheatography.com/djackh/

Not published yet.
Last updated 9th October, 2017.

Sponsored by **Readable.com**
Measure your website readability!
https://readable.com