**pyenv install for CentOS 6.5 2.6.32 x86\_64**

**1、更换阿里云yum源**

[**python**@localhost ~]# mv /etc/yum.repos.d/CentOS-Base.repo /etc/yum.repos.d/CentOS-Base.repo.backup

[**python**@localhost ~]# wget -O /etc/yum.repos.d/CentOS-Base.repo http:*//mirrors.aliyun.com/repo/Centos-6.repo*

**2、安装相关依赖包**

[python@localhost ~]**# yum -y install git**

[python@localhost ~]**# yum -y install gcc make patch gdbm-devel openssl-devel sqlite-devel readline-devel zlib-devel bzip2-devel nss**

**3、添加python用户并设置密码**

[**python**@localhost ~]# useradd **python**

[**python**@localhost ~]# passwd **python**

**4、pyenv**

# 进入 python用户

[**python**@localhost ~]# su - **python**

**4.1、 pyenv安装**

# 在线安装

[python@localhost ~]$ curl -L https://github.com/pyenv/pyenv-installer/raw/master/bin/pyenv-installer | bash

# 离线安装

[python@localhost ~]$ git clone https://github.com/pyenv/pyenv.git ~/.pyenv

[python@localhost ~]$ checkout https://github.com/pyenv/pyenv-doctor.git ~/.pyenv/plugins/pyenv-doctor

[python@localhost ~]$ checkout https://github.com/pyenv/pyenv-installer.git ~/.pyenv/plugins/pyenv-installer

[python@localhost ~]$ checkout https://github.com/pyenv/pyenv-update.git ~/.pyenv/plugins/pyenv-update

[python@localhost ~]$ checkout https://github.com/pyenv/pyenv-virtualenv.git ~/.pyenv/plugins/pyenv-virtualenv

[python@localhost ~]$ checkout https://github.com/pyenv/pyenv-which-ext.git ~/.pyenv/plugins/pyenv-which-ext

# 添加环境变量

[python@localhost ~]$ vi ~/.bashrc

export PATH="/home/python/.pyenv/bin:$PATH"

eval "$(pyenv init -)"

eval "$(pyenv virtualenv-init -)"

# 使环境变量生效

[**python**@localhost ~]$ **source** ~/.bashrc

**4.2、 pyenv安装python**

# 使用pyenv安装python # 在线安装 python 3.6.3版本

[python@localhost ~]$ pyenv install 3.6.3

# 离线安装python3.5.3版本

[python@localhost .pyenv]$ cd ~/.pyenv/ *# 进入用户家目录的.pyenv目录*

[python@localhost .pyenv]$ mkdir cache *# 创建cache目录，用于放python源码包*

[python@localhost .pyenv]$ cd cache/ *# 进入cache目录并下载python源码包*

[python@localhost cache]$ wget https://www.python.org/ftp/python/3.5.3/Python-3.5.3.tar.xz

[python@localhost cache]$ wget https://www.python.org/ftp/python/3.5.3/Python-3.5.3.tgz

[python@localhost cache]$ wget https://www.python.org/ftp/python/3.6.3/Python-3.5.3.tar.gz

[python@localhost cache]$ pyenv install 3.5.3 *# 进行离线安装*

**4.3、pyenv控制python版本**

# global 全局设置  
global设置全局Python版本，该设置永久生效。该操作不影响使用local设置的本地目录的python版本。

[python@localhost ~]$ python -V

Python 2.6.6

[python@localhost ~]$ pyenv version

system (set by /home/python/.pyenv/version)

[python@localhost ~]$ pyenv versions

\* system (set by /home/python/.pyenv/version)

3.5.3

3.6.3

[python@localhost ~]$ pyenv global 3.6.3

[python@localhost ~]$ python -V

Python 3.6.3

[python@localhost ~]$ pyenv version

3.6.3 (set by /home/python/.pyenv/version)

[python@localhost ~]$ pyenv versions

system

3.5.3

\* 3.6.3 (set by /home/python/.pyenv/version)

# shell 设置当前会话的python版本 shell设置当前会话的python版本，退出当前会话后设置失效。

[**python**@localhost ~]$ **python** -V

Python 3.6.3

[**python**@localhost ~]$ pyenv shell 3.5.3

[**python**@localhost ~]$ **python** -V

Python 3.5.3

[**python**@localhost ~]$ exit

logout

[root@localhost ~]# su - **python**

[**python**@localhost ~]$ **python** -V

Python 3.6.3

# local 设置本地程序目录python版本 该设置永久生效，不会因为shell和global设置的python版本而改变。 使用local设置的程序目录，其子目录会继承该目录的python版本。

[python@localhost ~]$ python -V

Python 2.6.6

[python@localhost ~]$ pyenv version

system (set by /home/python/.pyenv/version)

[python@localhost ~]$ pyenv versions

\* system (set by /home/python/.pyenv/version)

3.5.3

3.6.3

[python@localhost ~]$ mkdir ~/www/wap/lib -p

[python@localhost ~]$ cd ~/www/wap/

[python@localhost wap]$ pyenv local 3.6.3

[python@localhost wap]$ python -V

Python 3.6.3

[python@localhost wap]$ cd **lib**/

[python@localhost **lib**]$ **python** -**V**

Python 3.6.3

[python@localhost **lib**]$ **cd** ../../

[python@localhost www]$ python -V

Python 2.6.6

**4.4、Virtualenv虚拟环境**

virtualenv虚拟环境中的配置和安装的库，只能在当前virtualenv环境使用，和其他环境不重复不冲突。 库的路径：

${HOME}/.pyenv/versions/${PYENV\_PYTHON\_VERSION}/envs/%{VIRTUALENV\_NAME}/**lib**/${**PYTHON\_VERSION**}/**site**-**packages**/

# 语法

Usage: pyenv virtualenv [-f|--force] [VIRTUALENV\_OPTIONS] [version] <virtualenv-name>

[python@localhost js]$ pyenv virtualenv 3.5.3 andy353 *# 创建一个andy353的Virtualenv名称并绑定在3.5.3这个版本上。*

Requirement already satisfied: setuptools in /home/python/.pyenv/versions/3.5.3/envs/andy353/**lib**/**python3**.5/**site**-**packages**

Requirement already satisfied: pip in /home/python/.pyenv/versions/3.5.3/envs/andy353/**lib**/**python3**.5/**site**-**packages**

[python@localhost js]$ pyenv versions

system

\* 3.5.3 (set by /home/python/andy/web/html/.python-version)

3.5.3/envs/andy353

3.6.3

andy353

[python@localhost js]$ mkdir ~/test

[python@localhost js]$ cd ~/test/

[python@localhost test]$ pyenv local andy353 # 使用local指令为指定的目录设置virtualenv版本

(andy353) [python@localhost test]$ cd ..

[python@localhost ~]$ cd test/

(andy353) [python@localhost test]$

**5、安装jupyter**

Jupyter Notebook（此前被称为 IPython notebook）是一个交互式笔记本，支持运行 40 多种编程语言。 # 配置阿里云pip源

[python@localhost ~]$ mkdir .pip

[python@localhost ~]$ **cd** .pip/

[python@localhost .pip]$ **vi** pip.**conf**

[**global**]

index-url=https://mirrors.aliyun.**com**/pypi/simple/

trusted-host=mirrors.aliyun.**com**

# 安装jupyter

[python@localhost test]$ pip instlal jupyter

# 设置jupyter notebook 密码，等会web登陆时使用

[python@localhost test]$ jupyter notebook password

# 启动jupyter

[python@localhost **test**]$ nohup jupyter notebook --ip=0.0**.0.0** --no-browser &

# 设置开机自启

[**python**@localhost ~]# vi /etc/rc.local

su - **python** -c "cd test;nohup jupyter notebook --ip=0.0.0.0 --no-browser &"