

# pyenv多版本管理

官网 <https://github.com/pyenv/pyenv>

快捷安装 <https://github.com/pyenv/pyenv#the-automatic-installer>

```
1 # yum install git curl
2
3 python编译依赖如下
4 # yum install gcc make patch gdbm-devel openssl-devel
5 # yum sqlite-devel readline-devel zlib-devel bzip2-devel
6
7 创建普通用户
8 # useradd python
9 # su - python
10
11
12 在python用户下安装
13 $ curl -L https://github.com/pyenv/pyenv-installer/raw/master/bin/pyenv-
    installer | bash
14
15 由于某些原因，浏览器可以访问，大家自行下载保存为shell脚本执行
16 https://raw.githubusercontent.com/pyenv/pyenv-installer/master/bin/pyenv-
    installer
17 或者使用项目源码文件
18 https://github.com/pyenv/pyenv-installer/blob/master/bin/pyenv-installer
19
20 以后更新pyenv使用
21 $ pyenv update
```

```
1 安装完，按照提示处理，把下面的脚本放到当前用户的.bashrc文件末尾
2 # the following to ~/.bashrc:
3
4 export PATH="/home/python/.pyenv/bin:$PATH"
5 eval "$(pyenv init -)"
6 eval "$(pyenv virtualenv-init -)"
```

## 缓存文件

由于连接国外网站速度太慢，可以先下载好将要安装的Python版本的安装包，直接放置在~/.pyenv/cache目录下，再进行安装。此cache目录不存在，请自行创建。

## 安装python多版本

```
1 $ pyenv install -l
2 $ pyenv install 3.6.9 -vvv
3 $ pyenv install 3.7.7 -vvv
4
5 $ pyenv version 当前python版本，system表示当前操作系统使用的版本
6 $ pyenv versions 查看所有已经安装版本，*星号表示当前版本
```

pyenv是源代码编译安装Python解释器，所以一定要安装gcc等依赖。

## 创建虚拟环境

由于不同项目，或使用不同Python版本，或使用同版本Python但使用不同版本库，很难将它们部署在一起，所以，应该使用不同的虚拟环境隔离部署。

使用pyenv local 命令建立一个目录和一个Python版本或虚拟版本之间的关系，该目录的子孙目录默认也继承这个版本。

```
1 $ mkdir -p projects/test369
2 $ cd projects/test369/
3
4 [python@nodex test369]$ pyenv virtualenv 3.6.9 py369
5 [python@nodex test369]$ pyenv local py369
6 (py369) [python@nodex test369]$ python -V
7 Python 3.6.9
8 (py369) [python@nodex test369]$ pyenv version
9 py369 (set by /home/python/projects/test369/.python-version)
```