

python

kevinluo

# Contents

<b>1</b>	<b>install</b>	<b>1</b>
1.1	主要相关工具和包为:	1
1.2	目标是用 <b>PIP</b> 安装包。好处多, 如自动下载安装依赖包	1
1.3	怎么装上 <b>pip</b>	1
1.3.1	路径 1: <b>get-pip.py</b> 法把 <b>pip/easy_install</b> 一起装了	1
1.3.2	路径 2: 源包 <b>setup.py</b> <b>python setup.py install</b>	2
1.3.3	路径 3: <b>easy_install</b> 法	2
1.3.4	<b>pip</b> 文档链接	2
1.3.5	<b>pip</b> 命令用法	2
1.4	<b>virtualenv</b>	2
<b>2</b>	<b>web 资源</b>	<b>3</b>
2.1	website	3
<b>3</b>	<b>package</b>	<b>3</b>
	目录	

## 1 install

### 1.1 主要相关工具和包为:

- **setuptools**
- **pip**
- **virtualenv**

### 1.2 目标是用 **PIP** 安装包。好处多, 如自动下载安装依赖包

### 1.3 怎么装上 **pip**

#### 1.3.1 路径 1: **get-pip.py** 法把 **pip/easy\_install** 一起装了

download **get-pip.py**

```
curl https://bootstrap.pypa.io/get-pip.py -o get-pip.py
```

```
python get-pip.py
```

Download **get-pip.py**

```
python get-pip.py #install or upgrade pip. Additionally, it will install setuptools and wheel
```

```
python get-pip.py --prefix=/usr/local/ # 装到指定的目录
```

```
python -m pip install --upgrade pip setuptools wheel #up to date copies of the setuptools and wheel projects are useful
```

**pip** can automatically install dependency

**pip** can install from either Source Distributions (sdist) or Wheels,

**pip** will prefer a compatible wheel.

Wheels are a pre-built distribution format that provides faster installation compared to Source Distributions (sdist), especially when a project contains compiled extensions.

### 1.3.2 路径 2: 源包 `setup.py python setup.py install`

用源包先装 `setuptools`, 再装 `pip`  
[tutorial-help-install package](#)  
[setuptools 包下载](#)  
[PIP 包下载](#)  
解压进入目录执行,  
`python setup.py install`

### 1.3.3 路径 3: `easy_install` 法

[easy\\_install 下载地址](#)  
`python ez_setup.py`  
会在 `python` 的安装目录中生成 `scripts` 目录, 其中有 `easy_install.exe`  
然后用  
`easy_install pip`  
`easy_install` 是由 `PEAK`(Python Enterprise Application Kit) 开发的 `setuptools` 包里带的一个命令, 所以使用 `easy_install` 实际上是在调用 `setuptools` 来完成安装模块的工作。

### 1.3.4 `pip` 文档链接

[pip docs](#)  
[pip Reference Guide](#)  
[dependency management tutorial](#)

### 1.3.5 `pip` 命令用法

如果 `Python2` 和 `Python3` 同时有 `pip`, 则使用方法如下:  
`python3 -m pip install XXX`  
`pip --version`  
`pip --help`  
`pip install -U pip` # 升级 `pip`  
`python -m pip install -U pip`  
`pip install SomePackage` # 最新版本 `pip install SomePackage==1.0.4` # 指定版本  
`pip install 'SomePackage>=1.0.4'` # 最小版本  
`pip uninstall SomePackage`  
`pip freeze > requirements.txt` # 当前系统包系统  
`pip install -r requirements.txt`

## 1.4 `virtualenv`

`virtualenv`

`virtualenv <pathName>` #在`pathname`处建立环境, 可以 `-p` 指定母`python`路径  
`\path\to\env\Scripts\activate.bat`  
`deactivate.bat`

[virtualenv docs](#)  
[venv docs](#)  
[Pipenv](#)

## 2 web 资源

python - [pypi pypa](#)  
Perl - [CPAN](#)  
Ruby - [Gems](#)  
  
latex - [CTAN](#)  
sublime - [packagecontrol.io](#)

### 2.1 website

- main page:  
<https://www.python.org>

- package get:

PYPI/PYPA python package  
<https://www.pypa.io/>  
<https://pypi.org/>  
- tutorial: [教程](#) <https://readthedocs.org/projects/python/tutorial>  
<https://packaging.python.org/tutorials/>  
<https://packaging.python.org/tutorials/installing-packages/#>

## 3 package