

# Python安装与环境

## Python介绍

- **Python 是一种解释型语言：**这意味着开发过程中没有了编译这个环节。

C++属于编译型语言，C在执行前需要使用诸如GCC等编译器编译成可执行文件才可以运行。而Python这类解释型语言可以直接运行代码文件。

- **Python 是交互式语言：**这意味着，您可以在一个Python提示符>>>后直接执行代码。
- **Python 是面向对象语言：**这意味着Python支持面向对象的风格或代码封装在对象的编程技术。

## 优点

- 简单、易学
- 免费、开源
- 面向对象
- 可扩展性 ----> 丰富的库
- 规范的代码

规范的代码主要指Python强制规范缩进，但这个东西有两面性

## 缺点

- 执行速度慢

如果是数据处理一类看重速度的算法与C++对于很容易看出来速度区别

这里给出一个对比：

Python

```
PS D:\PYTHON\teach> & C:/Users/ASUS/AppData/Local/Programs/Python/Python39/python.exe d:/PYTHON/teach/spend.py
一亿次加法赋值耗时3.3828680515289307秒
PS D:\PYTHON\teach> █
```

C++

```
PS D:\PYTHON\teach> .\spend.exe
Active code page: 65001
一亿次加法耗时0.048000秒
PS D:\PYTHON\teach> █
```

这里有个小插曲，我以为一百万次应该不少了。但是C++执行时间居然没超过一个tick，输出时间始终为0，让我怀疑了好久。

## 一点小提示

Python 2.x和Python 3.x在解释器方面是完全不同的，所以两者无法通用。在安装和后续程序编写中请注意

## Python安装

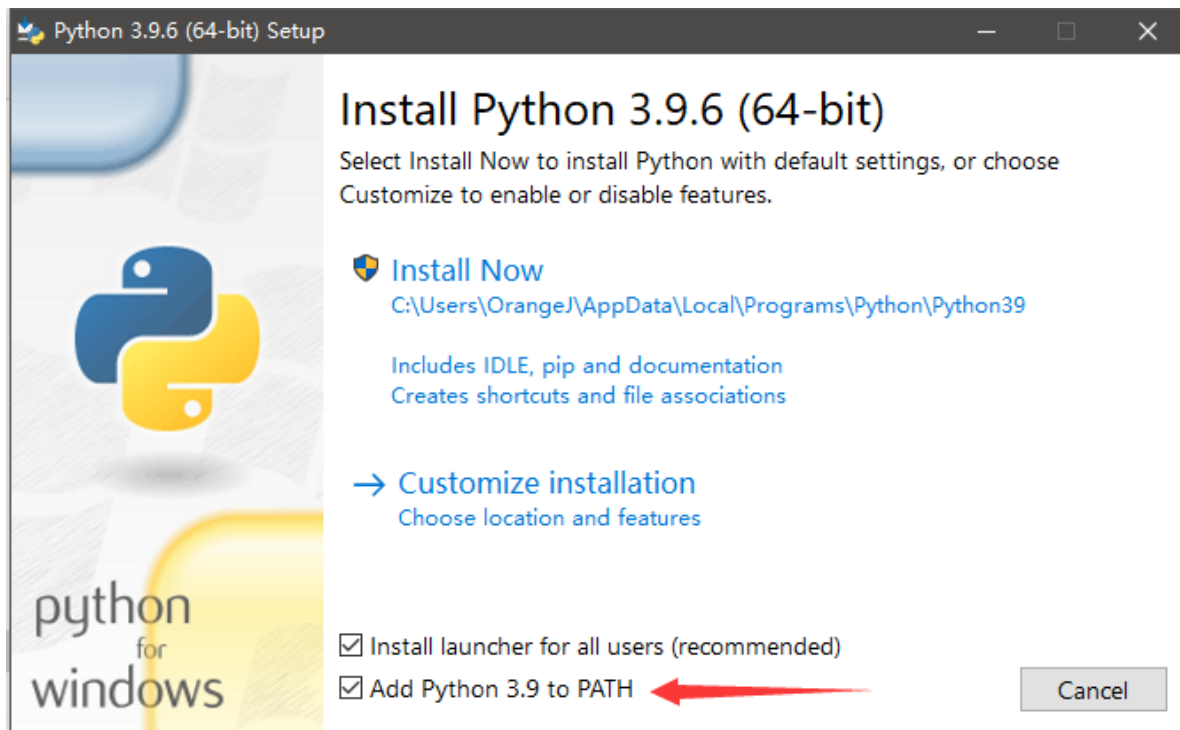
## windows

- 下载

建议去Python官网 ([www.python.org](http://www.python.org)) 下载。下载哪个版本不是很重要，用着顺手就行。

- 安装

唯一需要注意的是下图中红色箭头所指的选项。一定要勾选。可以省去一些之后麻烦的事情



- 验证安装

Ctrl+R输入cmd调出命令提示符。

输入 `python --version`，如果出现Python版本号则安装正确。

接下来输入 `pip --version`，同样如果出现pip版本号则安装正确

## Linux

你都Linux了你还来看这一段( '□' ) へ┐┐

你不是个合格的Linux选手

不论是CentOS还是Ubuntu都是自带Python 2的，部分较新的系统可能会带有python 3。不确定是否有的话可以先看下面的验证安装。

插一句，因为Linux中大部分时间是在命令行中操作所以在Linux中是需要明显区分Python2和Python3的指令的。

一般默认安装的情况下，Python2.x的调用指令为Python，其pip调用指令为pip；Python3.x的调用指令为Python3，其pip调用指令为pip3

**一定要不要卸载Linux中自带的Python环境，除非你有信心重建环境**

请先去官网 ([www.python.org/downloads](http://www.python.org/downloads)) 下载源码包（即Source Code，下面有个for Linux/Unix）

```
1 tar -zxvf Python-3.x.x.tgz
2 cd Python-3.x.x
3 ./configure
4 make && make install
5
6 // CentOS
7 yum -y install python3-pip
8 // Ubuntu
9 apt-get install python3-pip
```

- 验证安装

在命令行中输入 `python3 --version`，如果出现Python版本号则安装正确。

接下来输入 `pip3 --version`，同样如果出现pip版本号则安装正确

## 注意事项

后期使用的时候**不建议**盲目更新Python，这可能会导致一些稀奇古怪的错误。主要是依赖错误。为了避免出问题尽量停留在一个版本上。如果你有多台设备的的话我**建议**最好也安装相同版本的Python，可以极大减少换设备无法运行的情况。

## Pip使用与环境

### pip安装第三方库

```
1 // windows(Linux请使用pip3指令)
2 pip install PackageName
3 // 安装指定版本, (x.x.x)为版本号
4 pip install PackageName==x.x.x
5 // 从指定地址查找并安装库
6 pip install PackageName -i http://xxxxx.xxx
```

在一些情况下可以自己下载whl文件进行安装

包名中 `cp3x` 部分对应适配Python版本

如 `cp36` 适配Python 3.6.x，请尽量下载对应版本的，非对应版本安装时pip会报错。

如果是在找不到对应版本，可以下载最邻近版本，然后修改文件名中 `cp3x` 部分与自己的Python版本对应即可正常安装。

```
1 pip install xxx.whl
```

### pip更换镜像源

正常会从Python官方的pip源下载，速度会比较慢（可以加梯子）。

我们可以更换为国内的镜像源，我个人推荐使用清华源或者是阿里源

## Window

在 `C://users/你的用户名/` 下新建一个文件夹 `.pip`，在 `.pip` 内新建一个文件 `pip.ini`

```
1 // pip.ini文件内容，二选一即可
2
3 // 清华源
4 [global]
5 index-url = https://pypi.tuna.tsinghua.edu.cn/simple
6 [install]
7 trusted-host=pypi.tuna.tsinghua.edu.cn
8
9 // 阿里源
10 [global]
11 index-url = https://mirrors.aliyun.com/pypi/simple/
12 [install]
13 trusted-host=mirrors.aliyun.com
```

## Linux

```
1 // 进入用户文件夹
2 cd ~
3 mkdir .pip
4 cd .pip
5 vi pip.conf
```

```
1 // pip.conf内容，二选一即可
2
3 // 清华源
4 [global]
5 index-url = https://pypi.tuna.tsinghua.edu.cn/simple
6 [install]
7 trusted-host=pypi.tuna.tsinghua.edu.cn
8
9 // 阿里源
10 [global]
11 index-url = https://mirrors.aliyun.com/pypi/simple/
12 [install]
13 trusted-host=mirrors.aliyun.com
```