

说明：由于Go语言没有提供进程、线程的简单易用的库，为了让大家明白并发原理，使用语法易懂的Python语言演示。大家把学习重点放到原理的理解，不要纠结于Python语法。

Python开发环境

Python是动态语言，其代码需要在虚拟机PVM上运行。

CPython

Python下载地址：<https://www.python.org/downloads/>

从官网下载Python3.7以上版本，当下3.7、3.8、3.9为稳定版本。本次使用3.9版。

Version	Operating System	Description	MD5 Sum	File Size
Gzipped source tarball	Source release		6dbe644dd1a520d9853cf6648084c346	26071329
XZ compressed source tarball	Source release		7bf85df71bbe7f95e5370b983e6ae684	19627028
macOS 64-bit universal2 installer	macOS	for macOS 10.9 and later	892634724ab799569b512082c8f48c83	41005648
Windows embeddable package (32-bit)	Windows		a681a7f9b242fe35b4d96d79e15e57d6	7663448
Windows embeddable package (64-bit)	Windows		f38a9e7e02a992daa62569b758d0a388	8625602
Windows help file	Windows		448f8401ade49a7e2156d02512f2f9bf	9391521
Windows installer (32-bit)	Windows		a81b81687bc2575c05a30f4b31d6ea00	27859200
Windows installer (64-bit)	Windows	Recommended	9735797853cba809b13c8396c91354a0	29010904

以Windows为例，讲下安装官方解释器。

Windows下载Windows Installer(64-bit)版本。



勾选增加到PATH环境变量，下面就是按照提示点击，安装完成即可。

安装完成后，打开命令行，运行下面命令

```
1 | $ python -V
2 | Python 3.9.13
```

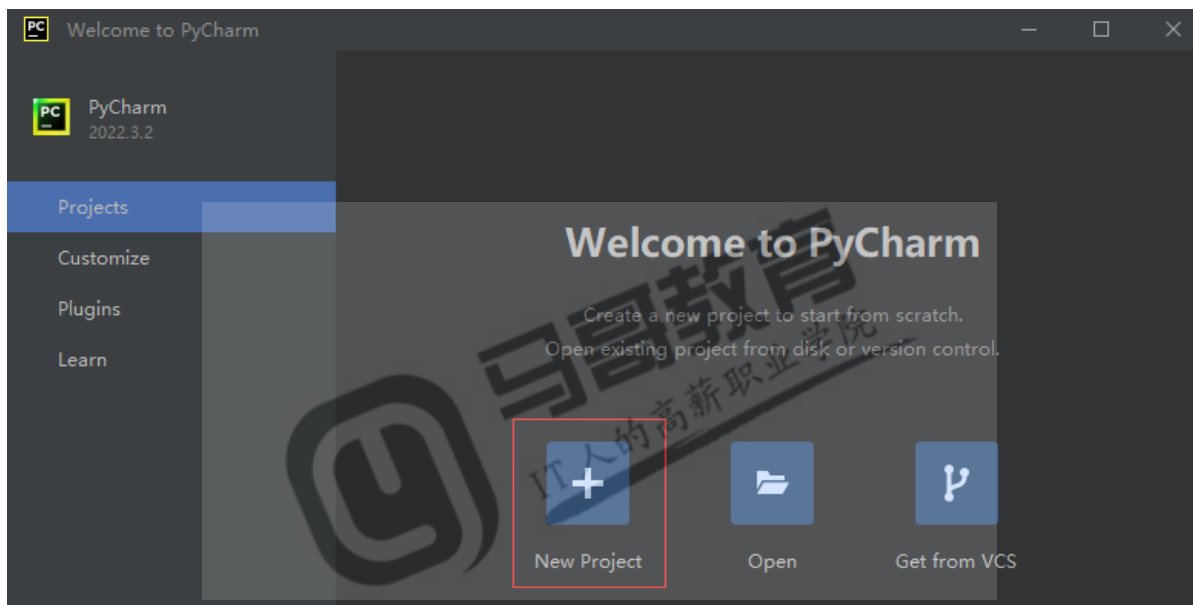
Pycharm

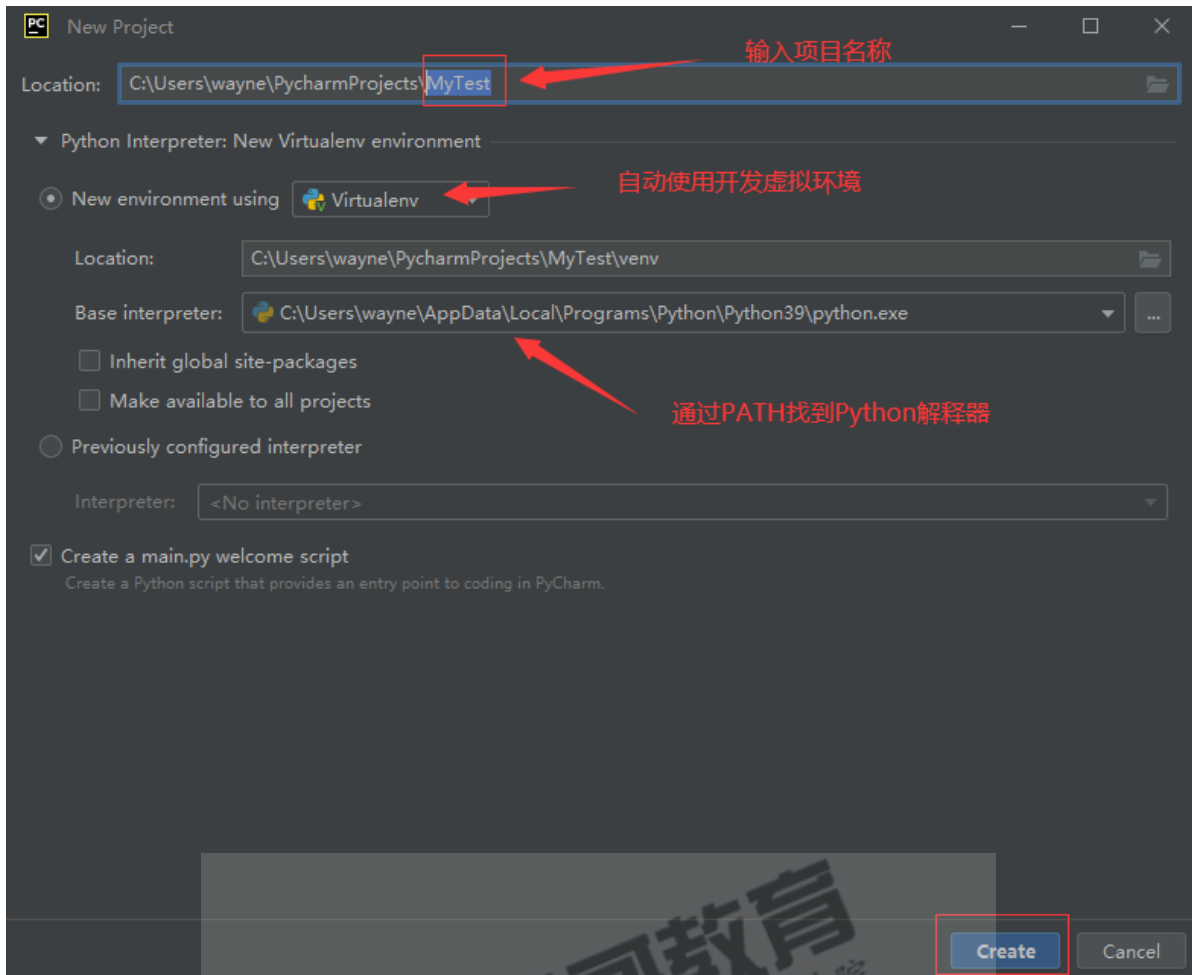
<https://www.jetbrains.com/pycharm/download/#section=windows>

从官网下载对应操作系统的Pycharm社区版，日常项目开发足够用了。只需按照安装提示选择进行安装即可。

Pycharm会搜索PATH环境变量中的Python解释器目录，以找到Python解释器来编译代码。

创建项目





至此就可以使用Python标准库开发了。

Python语法

Python是动态语言，不需要指定标识符类型，但是任何一个合法值都有类型。

注释使用#开头。

内建函数，可以全局使用，常用有print、int、str等。

Python没有常量，只有变量，标识符不需要指定类型，赋值给标识符，标识符就确定了类型。

Python没有字符类型，只有字符串类型，使用双引号"、单引号'、三双引号"""、三单引号'''来表示字符串字面量。

布尔型两个值是True、False。

bytes是字节序列，可以简单认为是在字符串前面加上b前缀，例如b"abc"、b'\x61\x62\x63'。

小括号括起来定义元组，元素是不可改变元素和长度的顺序表。例如()、(1,)、(1,2)。

中括号括起来定义列表，元素和长度可变的顺序表。

函数定义

- 可以形参
 - 仅位置形参
 - 普通形参
 - 仅关键字形参
 - 可变参数

- 可变关键字参数
- 前3种可以有缺省值
- 不需要指定返回值

```
1 def test(a, b=10): # 没有大括号，使用冒号加缩进方式表示语句块
2     return a + b
3
4 print(test(4, 5))
```

注意，下面是函数注解，它们不是Go语言中的类型定义，只是一种非强制的提醒

```
1 def test(a:int, b:int) -> int:
2     return a + b
3
4 print(test(4, 5))
5 print(test("a", 'b'))
```

