1

**ONEBOT**

Python编程用户手册

目录

[一、 简介 - 1 -](#_Toc5079)

[二、 开发环境搭建 - 2 -](#_Toc20067)

[ 装备工作 - 2 -](#_Toc16438)

[ 安装Python解释器 - 2 -](#_Toc2117)

[ 安装AiDragonfly包，适配ONEBOT主控 - 7 -](#_Toc4526)

[三、 AiDragonfly编程说明 - 9 -](#_Toc25122)

[ AiDragonfly编程提供主要模块如下 - 9 -](#_Toc13930)

[ AiDragonfly的三种引用方式 - 9 -](#_Toc1436)

[ AiDragonfly编程中大量常量的使用 - 10 -](#_Toc7451)

[ 常见报错及解决方法 - 11 -](#_Toc19401)

[四 实验一·制作门铃 - 12 -](#_Toc4841)

[ 准备 - 12 -](#_Toc27646)

[ 材料清单 - 12 -](#_Toc7569)

[ 示例代码： - 12 -](#_Toc14800)

[ 运行py程序 - 13 -](#_Toc23519)

[ 实验现象 - 13 -](#_Toc13373)

[ 实验拓展 - 13 -](#_Toc18094)

[五 实验二 乐器 - 14 -](#_Toc30504)

[ 准备 - 14 -](#_Toc16133)

[ 材料清单 - 14 -](#_Toc27006)

[ 示例代码： - 14 -](#_Toc31251)

[ 运行py程序 - 15 -](#_Toc5572)

[ 实验现象 - 15 -](#_Toc8726)

[ 实验拓展 - 15 -](#_Toc18400)

[六 实验三·电机 - 17 -](#_Toc31602)

[ 介绍 - 17 -](#_Toc8792)

[ 材料清单 - 17 -](#_Toc6019)

[ 硬件连接 - 17 -](#_Toc11871)

[ 示例代码： - 18 -](#_Toc10984)

[ 运行py程序 - 18 -](#_Toc17761)

[ 实验现象 - 19 -](#_Toc2919)

[ 实验拓展 - 19 -](#_Toc26615)

[七实验四·传感器 - 20 -](#_Toc14734)

[ 介绍 - 20 -](#_Toc30053)

[ 材料清单 - 20 -](#_Toc15549)

[ 硬件连接 - 20 -](#_Toc4108)

[ 示例代码： - 20 -](#_Toc3147)

[ 运行py程序 - 21 -](#_Toc7804)

[ 实验现象 - 21 -](#_Toc22225)

[ 实验拓展 - 21 -](#_Toc5641)

[附录一 Python API接口说明 - 22 -](#_Toc16620)

[ 通用Python API接口 - 22 -](#_Toc25359)

[ 电机、舵机 Python API接口 - 25 -](#_Toc16116)

[ motor相关方法如下 - 25 -](#_Toc9288)

[ digitalServo 相关方法如下 - 29 -](#_Toc2422)

[ 传感器Python API接口 - 31 -](#_Toc23008)

[ 音效模块Python API接口 - 40 -](#_Toc27252)

[ 乐器模块Python API接口 - 40 -](#_Toc28210)

[附录二 Python集成开发环境的安装 - 43 -](#_Toc20922)

[ 安装Python集成开发环境PyCharm Community Edition - 43 -](#_Toc27978)

[ 小试牛刀 开始 Hello World - 47 -](#_Toc20683)

2. 简介

ONEBOT教育机器人 是小米生态链企业爱其科技 推出的一款面向 教育、创客 的编程机器人。同时提供模块化图形编程 和 Python编程。模块化图形编程 面向中小学生可视化编程，Python面向高中生、大学生、创客教育等。

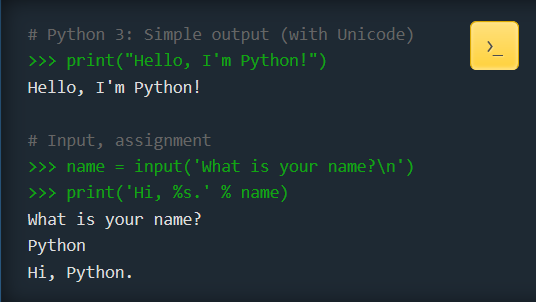
Python是一种解释型、面向对象、动态数据类型的高级程序设计语言，具有卓越的通用性、高效性、平台移植性和安全性。近几年 Python 在数据挖掘、人工智能等领域较为流行。

AiDragonfly 是爱其科技推出的python编程包，支持Python 3.0 及以上。

和ONEBOT主控、智能积木主控深度融合，和手机端拖块编程 Dragonfly 一脉相承。

推荐采用流行的集成开发环境PyCharm Community Edition进行python程序开发。

基于集成开发环境PyCharm Community Edition 的安装、使用 参见附录二 Python集成开发环境的安装。



1. 开发环境搭建

* 装备工作

ONEBOT Python的开发是在PC端进行编程运行的，所以需要安装python解释器，可以选择以下两种方法：

1. Python官网下载：

网址：<https://www.python.org/>

请根据网站提示及所用平台进行选择。

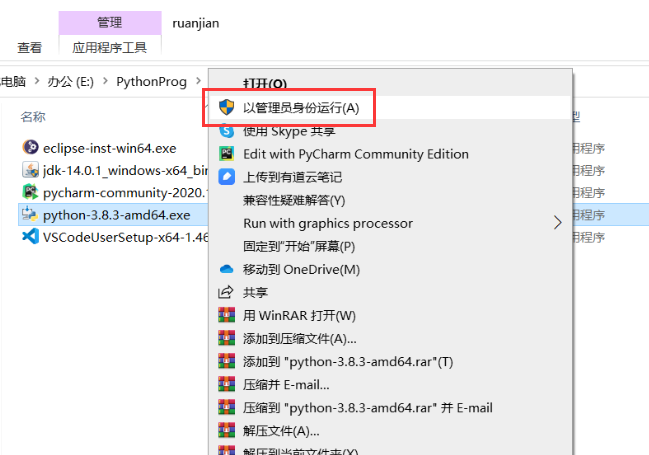
1. 开源代码管理平台下载：

网址：<https://gitee.com/beijing-aiqi/ai-dragonfly>

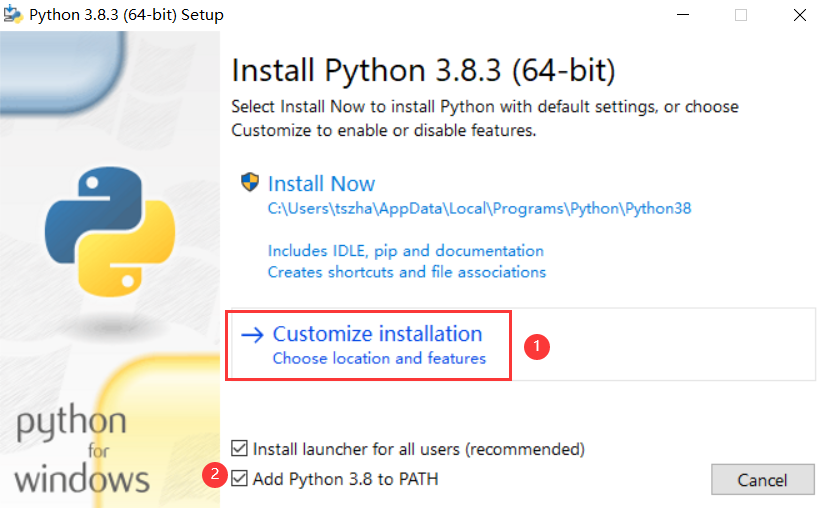
请在平台software目录下找到“python-3.8.3-amd64.exe”进行下载安装。

* 安装Python解释器

1. 右键点击python-3.8.3-amd64.exe，选择 以管理员身份运行如下图

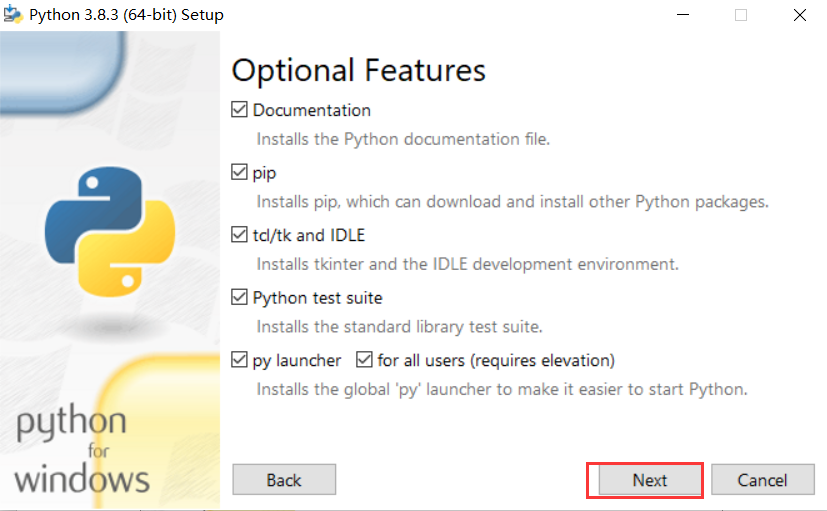


1. 进入以下页面，勾选Add Python 3.8 to PATH（如数字2位置所示），添加路径，然后 选择Customize installation自定义安装（如数字1 所示）



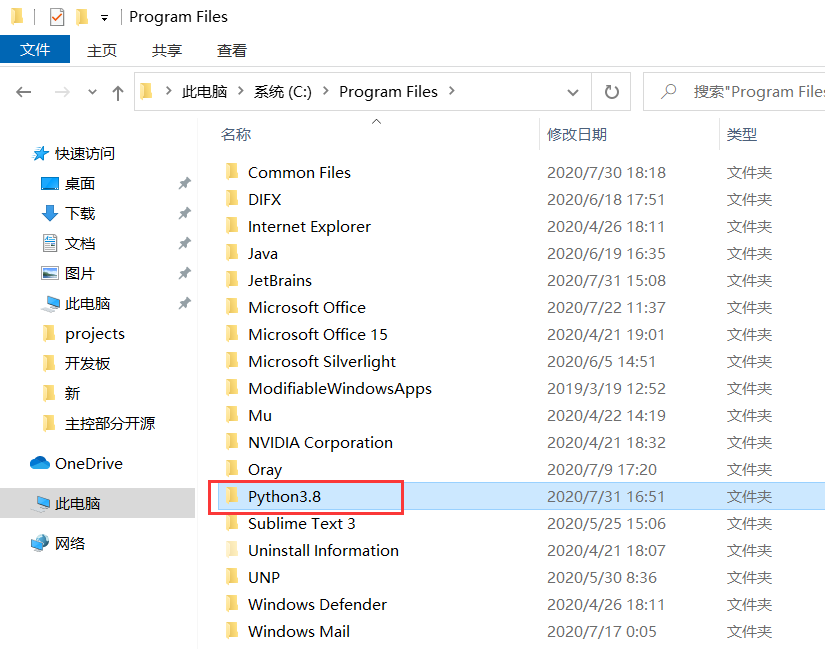
**提示：**注意勾选 Add Python 3.8 to PATH 添加路径。否则安装完后需要自己配置环境变量

1. 直接点击Next

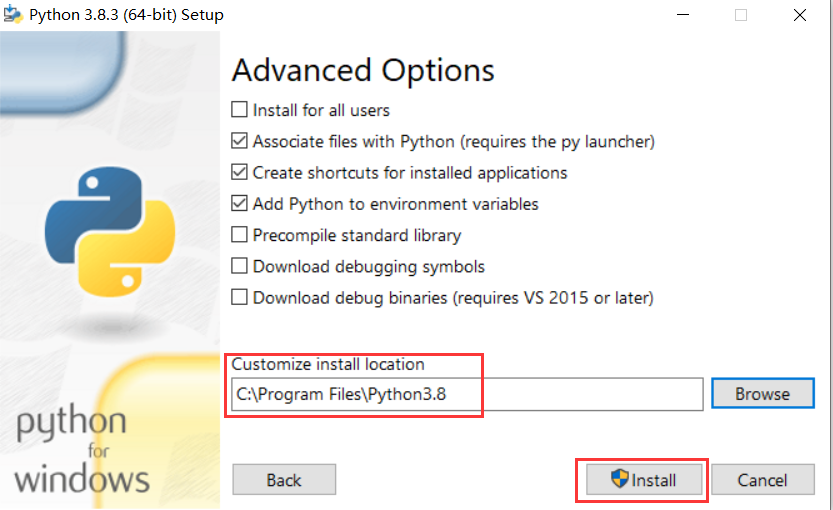


1. 选择安装路径并安装

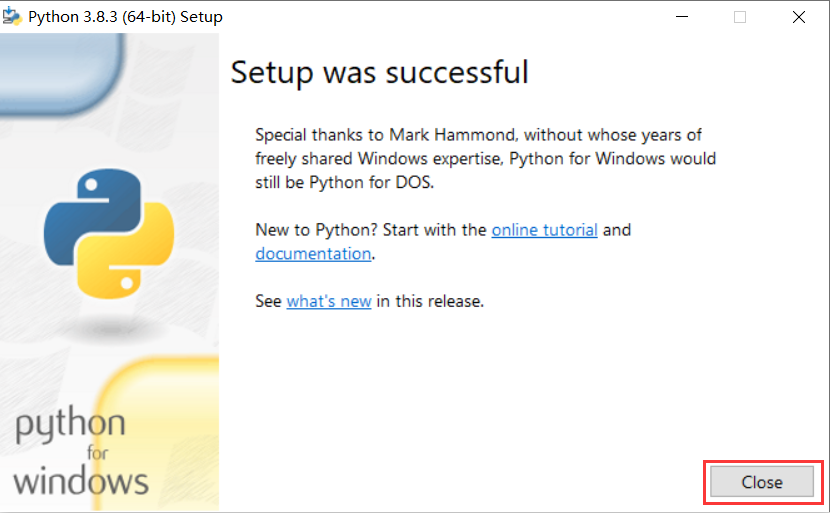
在需要安装的位置创建一个新的文件夹 如我要安装到C盘的Program Files文件夹下 ，就在此文件夹下 新建了一个python3.8文件夹 如下图



点击Browse 后 选择刚才的文件夹，如下图 然后点击 Install

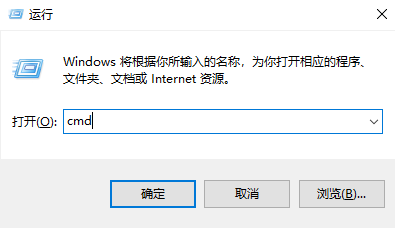


1. 等待安装完成 点击Close

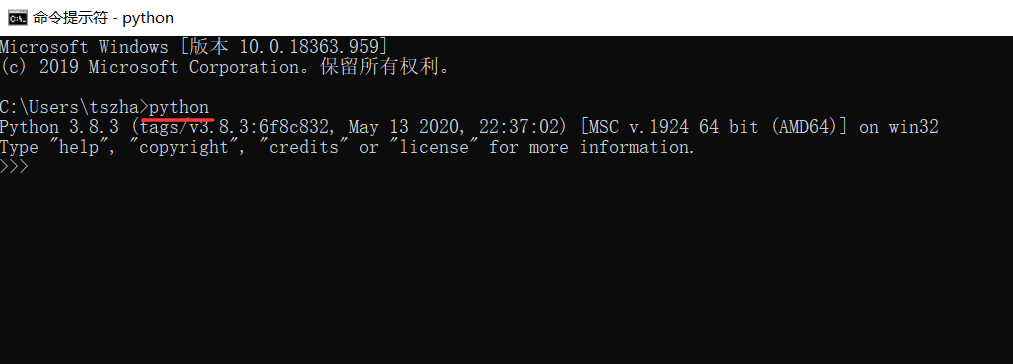


1. 检测是否安装成功

同时按住键盘上的“Win+R”打开运行窗口，输入cmd打开命令行。

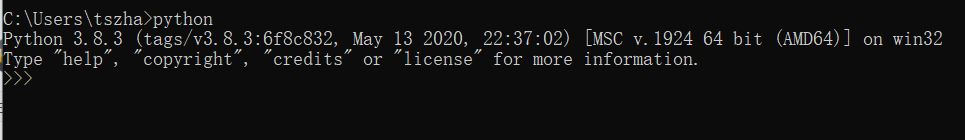


输入python命令出现python版本号提示，说明python安装成功



1. 进入和退出python命令行方式

在安装了python解释器后，打开cmd命令行，直接输入python即可进入。



在python命令行下>>>，有三种方式退出，如下：

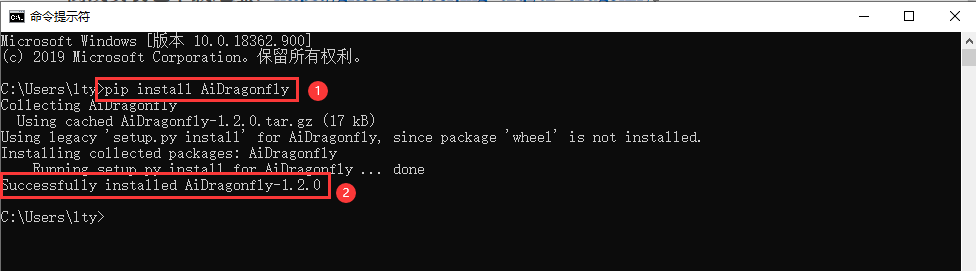
1. 输入exit()，回车
2. 输入quit()，回车
3. 同时按一下键盘上的ctrl和z按钮，然后回车

* 安装AiDragonfly包，适配ONEBOT主控

AiDragonfly为ONEBOT教育机器人必备支持包，其可使用python pip进行在线安装或使用AiDragonfly.tar.gz安装包离线安装。

1. 在线安装方法：

打开cmd命令行，输入pip install AiDragonfly进行自动安装（如下图中数字1所示），提示“Successfully installed”（如下图数字2所示）表示安装成功。（注意：版本号以当前最新版本号为准）

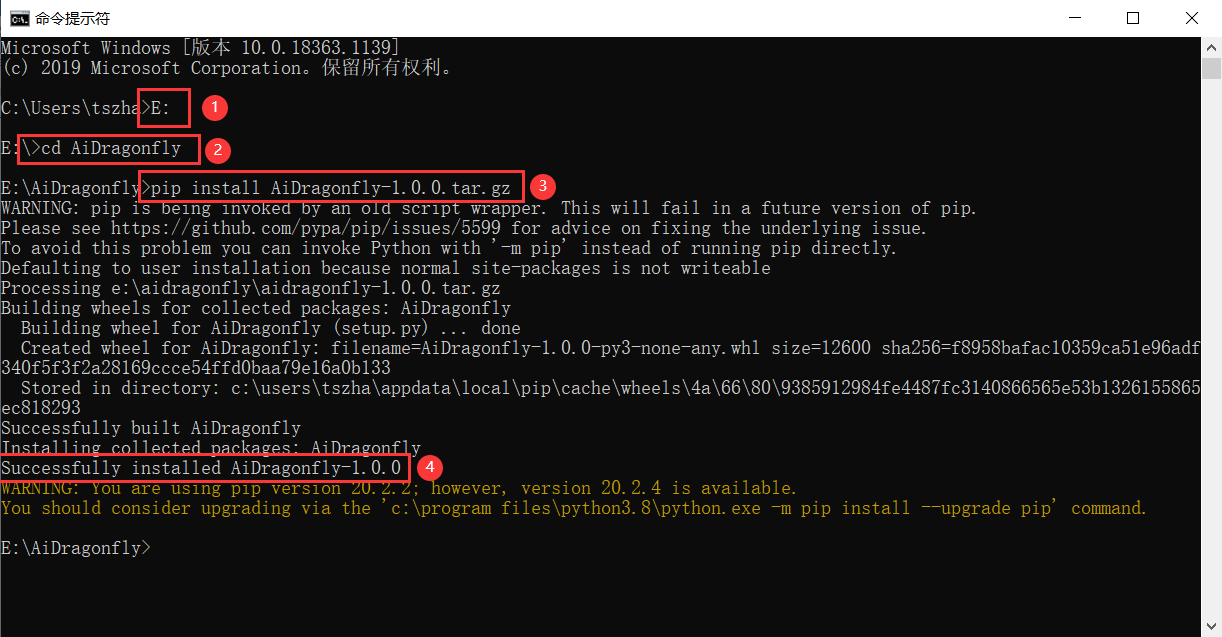


1. 离线安装方法：

离线安装包下载地址：<https://gitee.com/beijing-aiqi/ai-dragonfly>

请在平台software目录下找到“AiDragonfly-x.x.x.tar.gz”进行下载安装。

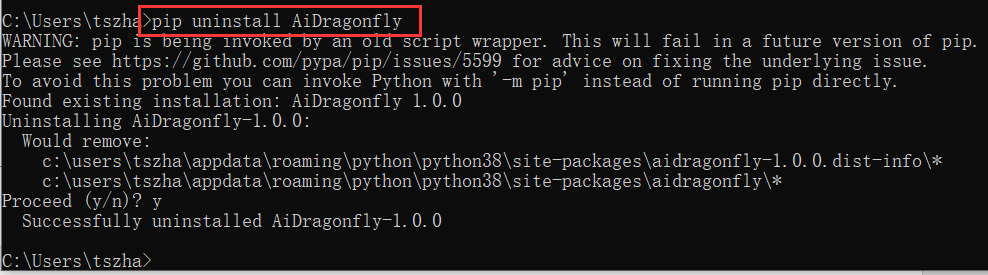
打开cmd命令行，使用cd指令进入到AiDragonfly安装包所在目录如上图中数字1、2指令所示（示例中AiDragonfly安装包在E盘下AiDragonfly文件夹中）输入pip install AiDragonfly-1.0.0.tar.gz回车安装（如数字3指令所示）。提示“Successfully installed”（如数字4所示）表示安装成功。（注意：版本号以当前最新版本号为准）



1. 卸载方法：

若不再使用AiDragonfly包或需要重新安装，可使用以下方法卸载AiDragonfly包

打开cmd命令行，输入pip uninstall AiDragonfly (如下图所示)。



1. AiDragonfly编程说明

* AiDragonfly编程提供主要模块如下

motor、digitalServo：伺服电机、数字舵机的角度、速度控制和获取

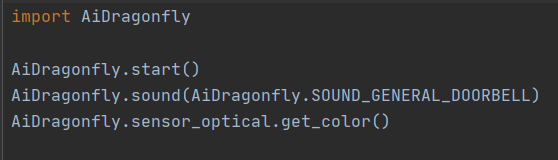
Sensor：传感器控制 、数据获取

sound、midi：主控支持的音效、乐器

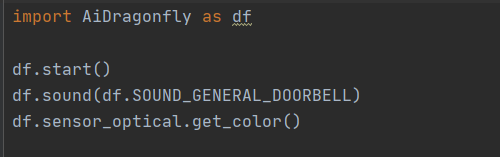
AiDragonfly提供的具体API接口及详细说明[参见：附录一 Python API接口说明](#附录)

* AiDragonfly的三种引用方式

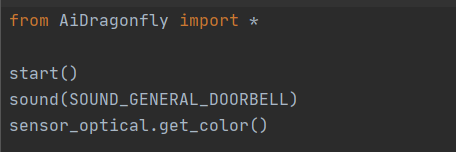
1. 第一种引用方式 如下



1. 第二种引用方式 如下



1. 第三种引用方式 如下



AiDragonfly 的引用方式 以上三种均可，编程时用户根据自己的编程习惯进行引用

AiDragonfly的对象、函数、常量等详细使用说明参见：[附录一 Python API接口说明](#附录)

* AiDragonfly编程中大量常量的使用

由于一些函数需要输入参数进行控制或获取信息，使用常量来表示。一方面方便了程序编写，一方面大大提高程序可读性。

注意：

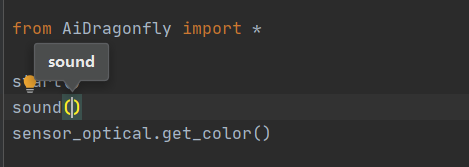
1.常量均使用大写字母，使用下划线‘\_’分割不同含义

2.调用方式:

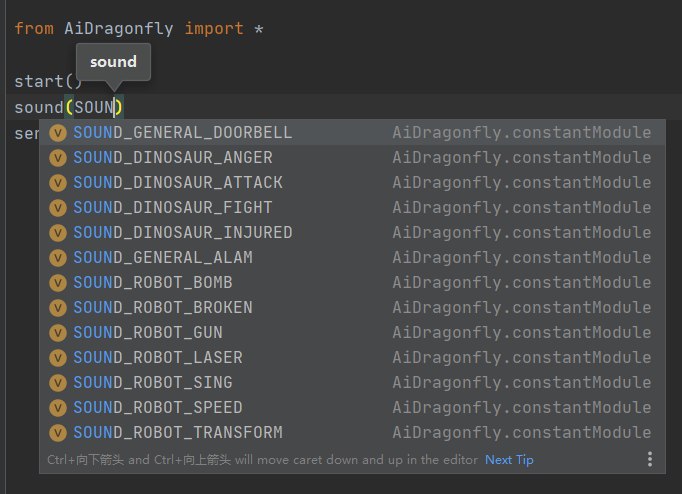
引入AiDragonfly包方式的不同，常量参数的调用方式也有区别：

2.1采用from import引入AiDragonfly的方式

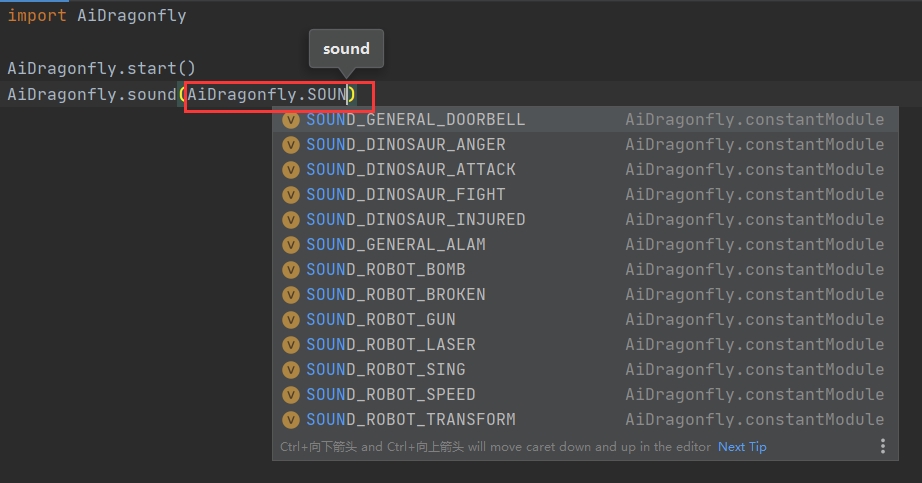
如sound方法中的形参sound（如下图所示）



根据形参sound提示，输入前几个字母的大写即可（如下图所示）



2.2 采用import引入AiDragonfly的方式



需要先输入“AiDragonfly.”然后根据形参sound提示，输入前几个字母的大写即可。

* 常见报错及解决方法

1. 蓝牙连接失败

AiDragonfly需要使用电脑的蓝牙功能连接ONEBOT教育机器人。

若所使用的电脑不支持蓝牙功能（例如：大部分桌面端电脑、系统缺失驱动或系统版本不支持等情况）注意配合外接“USB蓝牙适配器”使用，并根据适配器说明安装蓝牙驱动程序。

1. 缺少支持组件

若在运行程序时，如果出现以下错误



则需要安装pythonnet支持组件。在cmd命令行中输入 pip install pythonnet 如下图所示



四 实验一·制作门铃

* 准备

按照以上步骤安装开发环境，了解 AiDragonfly编程说明 及附录一中的python API接口说明。

* 材料清单

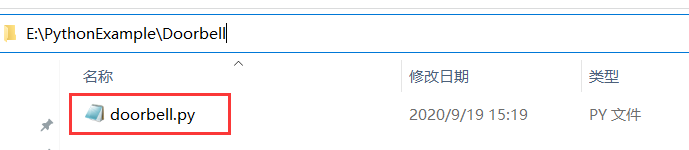
|  |  |  |
| --- | --- | --- |
| 所需硬件 | 数量 | 备注 |
| ONEBOT 主控 | 1 |  |

* 示例代码：

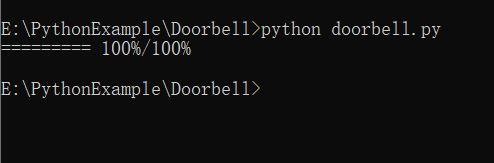
|  |
| --- |
| **import** AiDragonfly  **import** time  AiDragonfly**.**start**()** #连接主控  AiDragonfly**.**sound**(**AiDragonfly**.**SOUND\_GENERAL\_DOORBELL**)** #播放门铃  time**.**sleep**(**1**)** |

* 运行py程序

在记事本中编写如上代码，且修改文件格式为py。（如下图所示）



打开cmd命令行，进入doorbell.py所在目录，输入python doorbell.py。（如下图所示）



* 实验现象

运行程序，连接主控后，会听到主控模拟 门铃 叮咚的声音。

* 实验拓展

扩展1：通过修改程序，可以让主控模拟不同的声音，如机器人的、恐龙的等。

扩展2：主控的音效可以和其它逻辑联动，如检测到有人开门，门铃响起。

说明：主控支持的音效可以参考：附录一 Python API接口说明。

五 实验二 乐器

* 准备

该例程为ONEBOT教育机器人主控独立编程使用实验（以主控喇叭播放不同音效功能为例）。请先按照以上步骤安装开发环境，了解AiDragonfly编程说明及附录一中的python API接口说明之后进行该实验。

* 材料清单

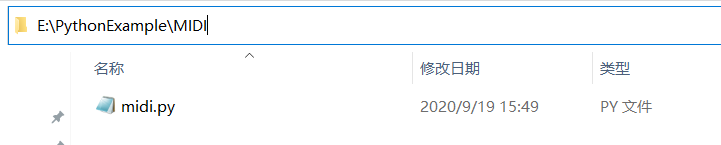
|  |  |  |
| --- | --- | --- |
| 所需硬件 | 数量 | 备注 |
| ONEBOT 主控 | 1 |  |

* 示例代码：

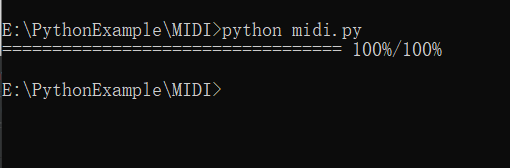
|  |
| --- |
| **import** AiDragonfly  **import** time  AiDragonfly**.**start**()**  time**.**sleep**(**1**)**  AiDragonfly**.**midi**.**piano**(**AiDragonfly**.**MIDI\_DO**,**10**)**  time**.**sleep**(**1**)** #音效之间需要添加延时  AiDragonfly**.**midi**.**piano**(**AiDragonfly**.**MIDI\_RE**,**10**)**  time**.**sleep**(**1**)** #音效之间需要添加延时  AiDragonfly**.**midi**.**piano**(**AiDragonfly**.**MIDI\_MI**,**10**)**  time**.**sleep**(**1**)**  AiDragonfly**.**midi**.**piano**(**AiDragonfly**.**MIDI\_FA**,**10**)**  time**.**sleep**(**1**)**  AiDragonfly**.**midi**.**piano**(**AiDragonfly**.**MIDI\_SO**,**10**)**  time**.**sleep**(**1**)**  AiDragonfly**.**midi**.**piano**(**AiDragonfly**.**MIDI\_LA**,**10**)**  time**.**sleep**(**1**)**  AiDragonfly**.**midi**.**piano**(**AiDragonfly**.**MIDI\_SI**,**10**)**  time**.**sleep**(**1**)** |

* 运行py程序

在记事本中编写如上代码，且修改文件格式为py。（如下图所示）



打开cmd命令行，进入midi.py所在目录，输入python midi.py如下图所示



* 实验现象

运行程序，连接主控后，可以听到主控播放钢琴的基本音节do、re、mi、fa、so、la、si

* 实验拓展

扩展1 通过修改程序，可以进行其它音乐创作。

扩展2 通过调用不同乐器，进行音乐创作。

说明：主控支持的乐器种类非常丰富，参考：附录一 Python API接口说明。

六 实验三·电机

* 介绍

该例程为ONEBOT教育机器人电机编程使用实验（以伺服电机正反向旋转功能为例）。请先按照以上步骤安装开发环境，了解AiDragonfly编程说明及附录一中的python API接口说明之后进行该实验。

* 材料清单

|  |  |  |
| --- | --- | --- |
| 所需硬件 | 数量 | 备注 |
| ONEBOT主控 | 1 |  |
| 伺服电机 | 2 | 执行器 |

* 硬件连接

将两个伺服电机插到主控的A、B端口，并打开ONEBOT主控电源开关。

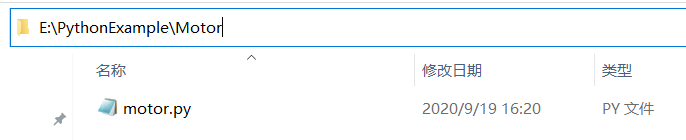


* 示例代码：

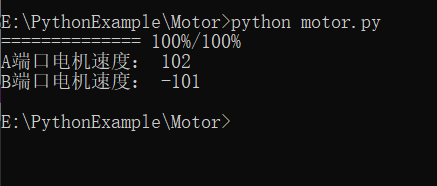
|  |
| --- |
| **import** AiDragonfly  **import** time  AiDragonfly**.**start**()**  AiDragonfly**.**motor**.**set\_speed\_portA**(**100**)** #设置A端口伺服电机正向转动  time**.**sleep**(**1**)**  speedA **=** AiDragonfly**.**motor**.**get\_speed**(**AiDragonfly**.**MOTOR\_PORTA**)**  **print(**'A端口电机速度：'**,**speedA**)**  AiDragonfly**.**motor**.**set\_speed\_portB**(-**100**)** #设置B端口伺服电机反向转动  time**.**sleep**(**1**)**  speedB **=** AiDragonfly**.**motor**.**get\_speed**(**AiDragonfly**.**MOTOR\_PORTB**)**  **print(**'B端口电机速度：'**,**speedB**)**  AiDragonfly**.**motor**.**set\_speed\_portA**(**0**)** #停止A端口伺服电机转动  AiDragonfly**.**motor**.**set\_speed\_portB**(**0**)** |

* 运行py程序

在记事本中编写如上代码，且修改文件格式为py。（如下图所示）



打开cmd命令行，进入motor.py所在目录，输入python motor.py 如下图所示



* 实验现象

点击运行，等待主控连接，可以看到A、B端口伺服电机，一个正转之后一个反转。同时可获得伺服电机速度。

* 实验拓展

扩展1：可以在不同端口控制伺服电机以不同的速度正转或反转。

扩展2：根据场景需求，合理的使用电机速度和角度控制，也可获取电机转动速度、转动角度进行更多扩展。

扩展3：支持数字舵机的角度控制、功率控制、角度获取，根据场景需求选择

七实验四·传感器

* 介绍

该例程为ONEBOT教育机器人传感器编程使用实验（以多功能光学传感器颜色识别功能为例）。请先按照以上步骤安装开发环境，了解AiDragonfly编程说明及附录一中的python API接口说明之后进行该实验。

* 材料清单

|  |  |  |
| --- | --- | --- |
| 所需硬件 | 数量 | 备注 |
| ONEBOT主控 | 1 |  |
| 多功能光学传感器 | 1 | 支持8种颜色识别，灰度识别，光强度识别等功能 |

* 硬件连接

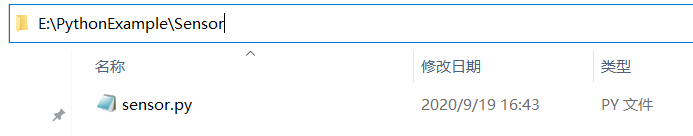
将多功能光学传感器插到主控任一端口，并打开ONEBOT主控电源开关。

* 示例代码：

|  |
| --- |
| **import** AiDragonfly  AiDragonfly**.**start**()**  color **=** AiDragonfly**.**sensor\_optical**.**get\_color**()**  **print(**'颜色：'**,**color**)** |

* 运行py程序

在记事本中编写如上代码，且修改文件格式为py。（如下图所示）

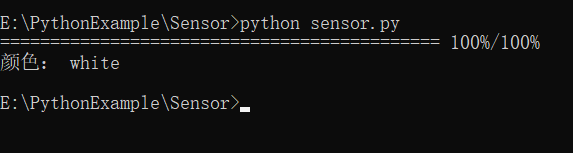


注意：多功能光学传感器颜色识别方法

手持多功能光学传感器，使其正面（有三个小圆灯的一面，type-c接口的对面）对着主控正面的白色表面，距离主控正面大概1cm左右即可。类似下图所示



打开cmd命令行，进入sensor.py所在目录，输入python sensor.py 如下图所示



* 实验现象

点击运行，等待主控连接，可以看到多功能光学传感器识别到了主控正面的白色如上图所示。

* 实验拓展

扩展1：参考如上示例 根据实际需求 获取颜色 进行小车循迹、边缘检测等。

扩展2：可以利用多功能光学传感器实现颜色识别、灰度识别、环境光识别等。

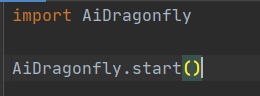
注：其它传感器使用方式参考如上示例。不同传感器的功能及Python API接口说明详见：附录一Python API接口说明。

附录一 Python API接口说明

**注：**AiDragonfly导入有三种方式，以下接口示例以import AiDragonfly方式导入进行说明。

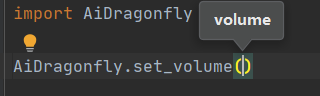
* 通用Python API接口

1. 连接主控



需要在程序开始时调用，函数功能：连接主控。没有参数

1. 设置 主控 音量大小

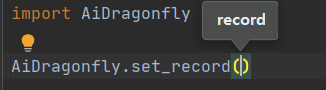


参数：

Volume：主控音量

【0 ~ 12】

1. 主控录音放音功能



参数：

Record：录音选项（最多录制80s时长）

【RECORD\_START、RECORD\_END、RECORD\_PLAY】

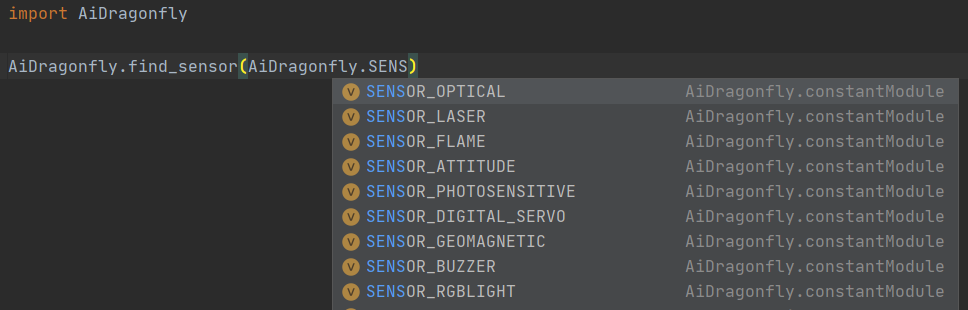
1. 查找传感器



参数:

Sensor：要查找的传感器（可选设备名如下图所示）

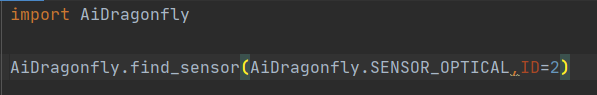
ID：传感器编号（用以区分多个同样的传感器，根据查询的具体情况设定，默认值为1）



|  |
| --- |
| SENSOR\_DIGITAL\_SERVO #数字舵机  SENSOR\_OPTICAL #多功能光学传感器  SENSOR\_BUZZER #无源蜂鸣器  SENSOR\_RGBLIGHT #RGB真彩灯  SENSOR\_LASER #激光传感器  SENSOR\_GEOMAGNETIC #地磁传感器  SENSOR\_ATTITUDE #姿态传感器  SENSOR\_PHOTOSENSITIVE #光敏传感器 |

示例：

主控上插入两个同样的多功能光学传感器，调用此函数可以查找传感器，并确认传感器ID编号。



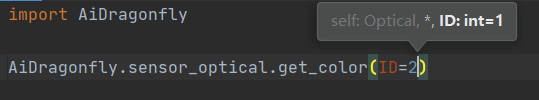
调用此函数后，会列出注册在主控上的所有传感器，同时ID为2（第二个注册的）的多功能光学传感器的按键指示灯会闪烁。

1. 关于相同类型多个传感器同时插入主控的操作说明

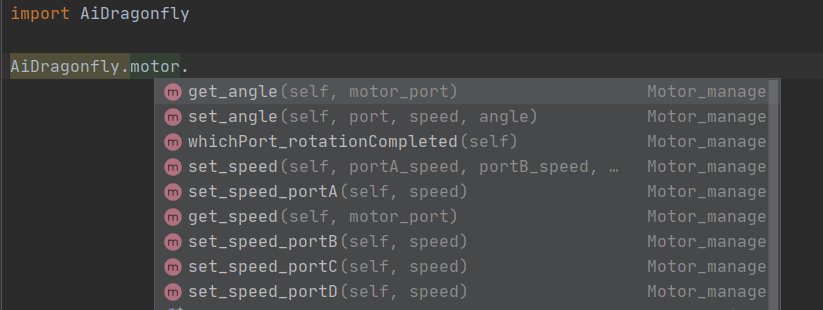
在传感器功能中，ID含义同上“4） 查找传感器->参数2：ID”。

当插入了两个或两个以上的同类型传感器时，需要指定控制或获取数据的传感器的ID编号，若不进行指定，默认为ID=1。

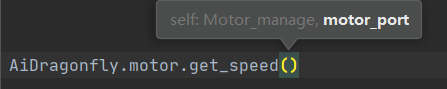
例如：获取第二个多功能光学传感器识别的颜色值



* 电机、舵机 Python API接口
* 伺服电机（motor）相关方法如下



1. 获取伺服电机速度



参数：

motor\_port：伺服电机所在的主控端口

【MOTOR\_PORTA、MOTOR\_PORTB、

MOTOR\_PORTC、MOTOR\_PORTD】

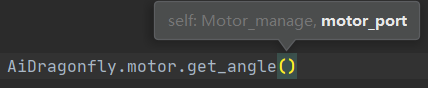
返回值：转动速度

【0 ~ 256转/分】

举例：获取端口A上伺服电机的速度



1. 获取伺服电机角度



参数：

motor\_port：伺服电机所在的主控端口

【MOTOR\_PORTA、MOTOR\_PORTB、

MOTOR\_PORTC、MOTOR\_PORTD】

返回值：伺服电机角度

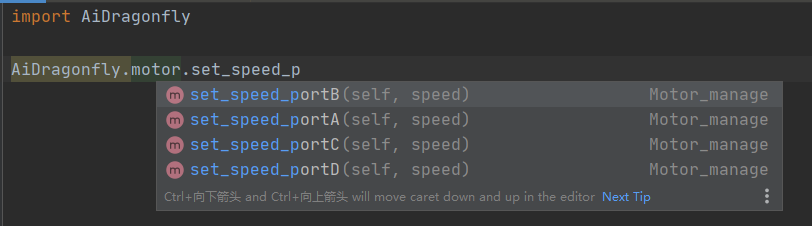
【-32768 ~ 32767】（0~32767 表示正转，0 ~ -32768 表示反转）

举例：获取端口A上伺服电机的角度



1. 设置伺服电机速度

3.1：设置某个端口伺服电机速度



参数：

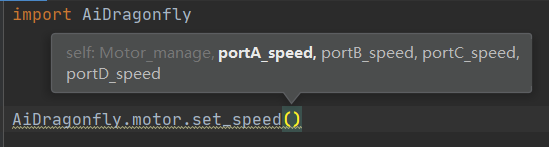
Speed：转动速度

【-127 ~ 127】（0 表示停止，1~127 表示正转，-1 ~ -127 表示反转）

举例：设置A端口伺服电机正向速度100



3.2：设置多端口伺服电机速度



参数：

portA\_speed：A端口伺服电机速度

【-127 ~ 127】（0 表示停止，1~127 表示正转，-1 ~ -127 表示反转）

portB\_speed：B端口伺服电机速度

【-127 ~ 127】（0 表示停止，1~127 表示正转，-1 ~ -127 表示反转）

portC\_speed：C端口伺服电机速度

【-127 ~ 127】（0 表示停止，1~127 表示正转，-1 ~ -127 表示反转）

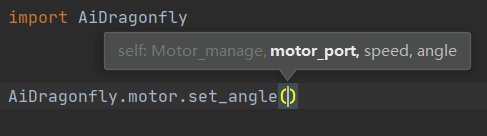
portD\_speed：D端口伺服电机速度

【-127 ~ 127】（0 表示停止，1~127 表示正转，-1 ~ -127 表示反转）

举例：同时设置端口A、C端口伺服电机速度为100和-100，B、D端口伺服电机速度为0



1. 设置伺服电机角度



参数：

motor\_port：伺服电机所在的主控端口

【MOTOR\_PORTA、MOTOR\_PORTB、

MOTOR\_PORTC、MOTOR\_PORTD】

speed：转动速度

【-127 ~ 127】（0 表示停止，1~127 表示正转，-1 ~ -127 表示反转）

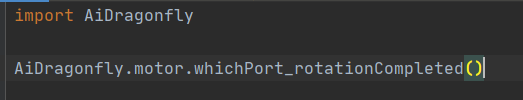
angle：转动角度

【0 ~ 65535】

举例：设置端口A电机 以100的速度转动200°



1. 查询电机是否转动完成

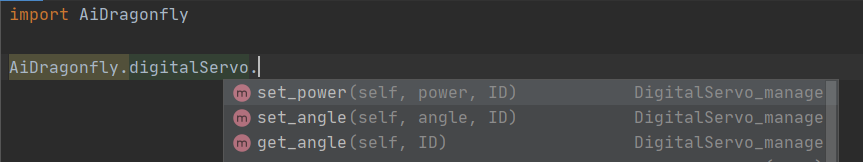


返回值：端口电机转动完成

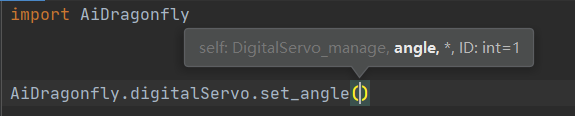
【portA\_completed、portB\_completed、

portC\_completed、portD\_completed】

* 数字舵机（digitalServo）相关方法如下



1. 设置数字舵机角度



参数：

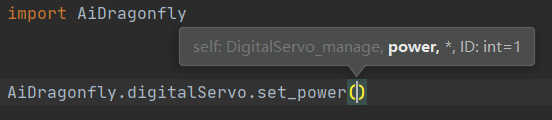
angle：旋转角度

【-160 ~ 160】（0~160表示舵机正向转动的角度，0 ~ -160表示舵机反向转动的角度）

举例：设置舵机正向转动100°



1. 设置数字舵机功率



参数：

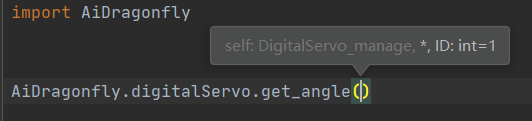
Power：转动速度

【-100 ~ 100】（ 0~100 表示舵机正向转动，0 ~ -100 表示舵机反向转动）

举例：设置舵机旋转速度为50



1. 获取数字舵机角度



返回值：转动角度

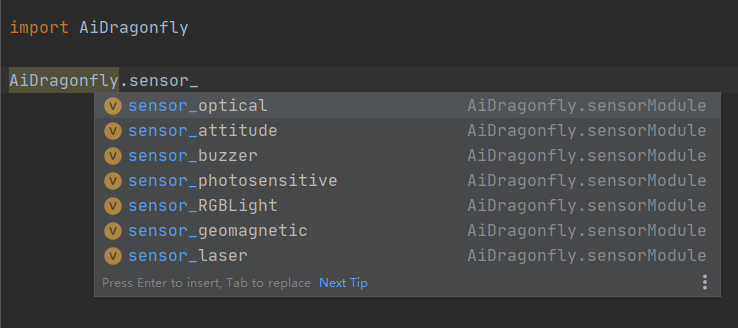
【-160 ~ 160】（0~160表示舵机正向转动的角度，0 ~ -160表示舵机反向转动的角度）

举例：获取舵机转动的角度



* 传感器Python API接口

传感器按照不同功能分类，如下所示：

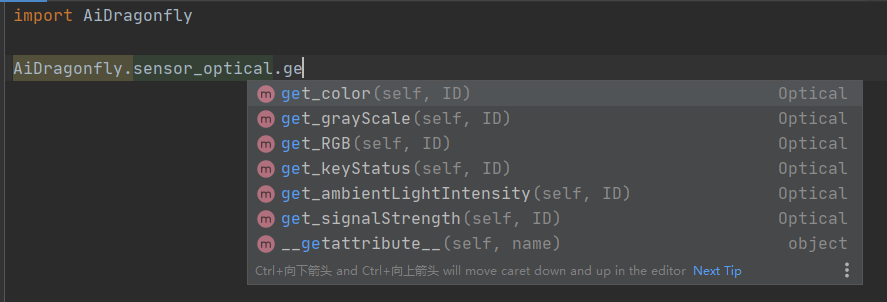


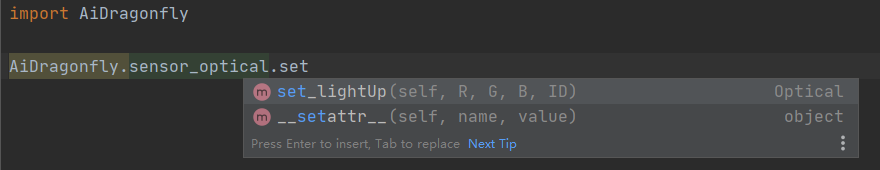
|  |
| --- |
| sensor\_opticval #多功能光学传感器  sensor\_attitude #姿态传感器  sensor\_buzzer #无源蜂鸣器  sensor\_photosensitive #光敏传感器  sensor\_RGBLight #RGB真彩灯  sensor\_geomagnetic #地磁传感器  sensor\_laser #激光传感器 |

传感器功能分为两类：

一类是获取数据（get），例如获取多功能光学传感器识别的颜色“get\_color”。

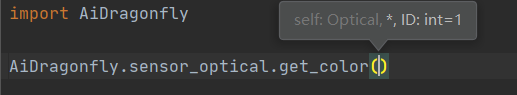
一类是设置数据（set），例如设置多功能光学传感器显示颜色“set\_lightUP”





1. 多功能光学传感器

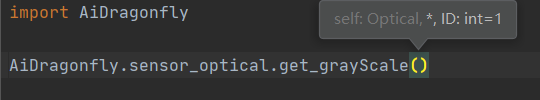
1.1获取 多功能光学传感器识别的颜色值



返回值：识别的颜色值

【blue、red、yellow、white、green、black、grey、purple】

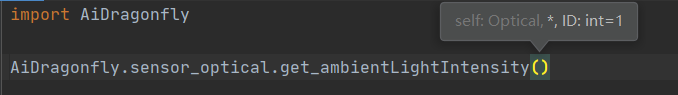
1.2 获取 多功能光学传感器识别的灰度



返回值：灰度值

【0 ~ 100】

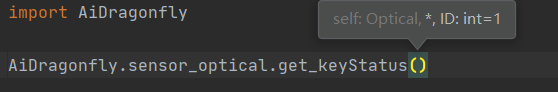
1.3 获取 多功能光学传感器识别的环境光强度



返回值：环境光强度

【0 ~ 1000】

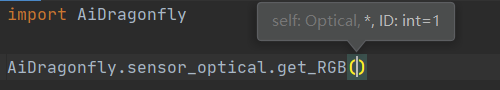
1.4 获取 多功能光学传感器识别的按键状态



返回值：按键状态

【press、release】

1.5 获取 多功能光学传感器识别的R、G、B的值



返回值：

R：红色颜色值

【0 ~ 255】

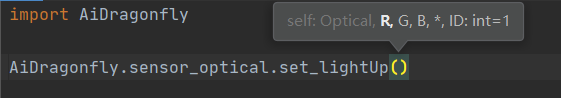
G：绿色颜色值

【0 ~ 255】

B：蓝色颜色值

【0 ~ 255】

1.6 设置 多功能光学传感器显示的R、G、B的值



参数：

R：红色颜色值

【0 ~ 255】

G：绿色颜色值

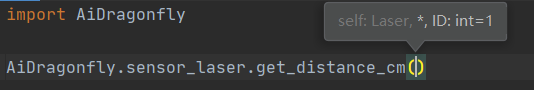
【0 ~ 255】

B：蓝色颜色值

【0 ~ 255】

1. 激光传感器

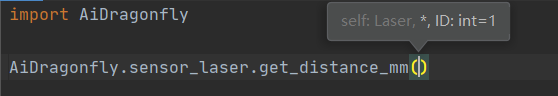
2.1 获取 激光传感器的



返回值：距离（单位cm）

【0 ~ 100】

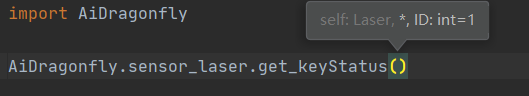
2.2 获取 激光传感器的距离



返回值：距离（单位mm）

【0 ~ 1000】

2.3 获取 激光传感器的按键状态

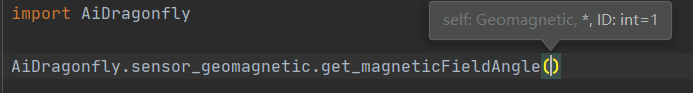


返回值：按键状态

【press、release】

1. 地磁传感器

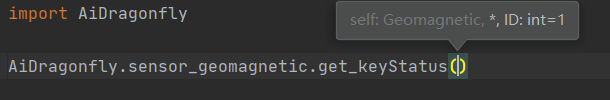
3.1 获取 地磁传感器的磁场角度



返回值：磁场角度（单位 °）

【0 ~ 359°】

3.2 获取 地磁传感器的按键状态

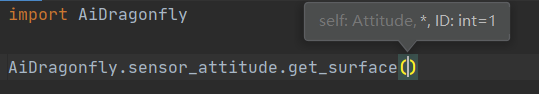


返回值：按键状态

【press、release】

1. 姿态传感器

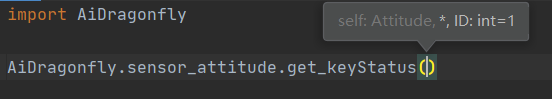
4.1 获取 姿态传感器的面



返回值：传感器朝上的面编号

【1 ~ 6】

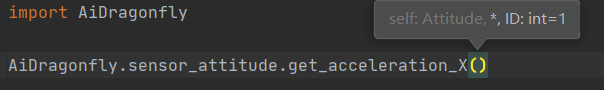
4.2 获取 姿态传感器的按键状态



返回值：按键状态

【press、release】

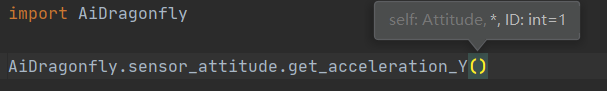
4.3 获取 姿态传感器的X轴加速度



返回值：X轴加速度（单位 cm/s^2）

【-2000.00cm/s\*s ~ 2000.00cm/s\*s】

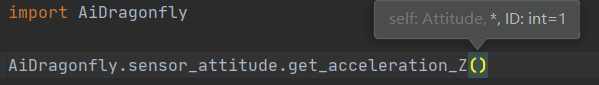
4.4 获取 姿态传感器的Y轴加速度



返回值：Y轴加速度（单位 cm/s^2）

【-2000.00cm/s\*s ~ 2000.00cm/s\*s】

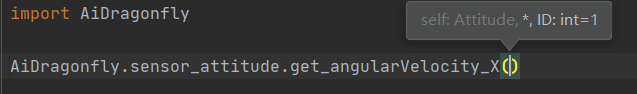
4.5 获取 姿态传感器的Z轴加速度



返回值：Z轴加速度（单位 cm/s^2）

【-2000.00cm/s\*s ~ 2000.00cm/s\*s】

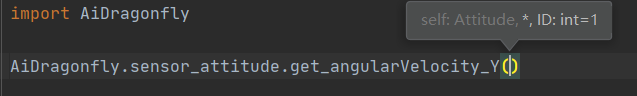
4.6 获取 姿态传感器的X轴角速度



返回值：X轴角速度（单位 cm/s^2）

【-2000.00cm/s\*s ~ 2000.00cm/s\*s】

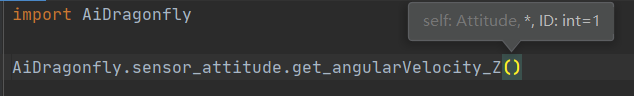
4.7 获取 姿态传感器的Y轴角速度



返回值：Y轴角速度（单位 cm/s^2）

【-2000.00cm/s\*s ~ 2000.00cm/s\*s】

4.8 获取 姿态传感器的Z轴角速度

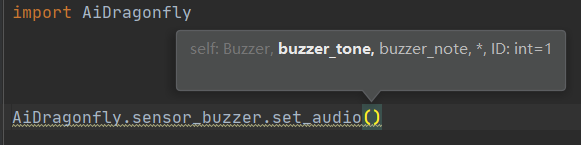


返回值：Z轴角速度（单位 cm/s^2）

【-2000.00cm/s\*s ~ 2000.00cm/s\*s】

1. 无源蜂鸣器

5.1 设置蜂鸣器音频



参数：

buzzer\_tone：蜂鸣器的音调高中低

【BUZZER\_TONE\_LOW、BUZZER\_TONE\_MIDDLE、BUZZER\_TONE\_HIGH】

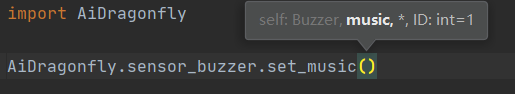
buzzer\_note：蜂鸣器的音符

【BUZZER\_NOTE\_DO、BUZZER\_NOTE\_RE

BUZZER\_NOTE\_MI、BUZZER\_NOTE\_FA

BUZZER\_NOTE\_SO、BUZZER\_NOTE\_LA、BUZZER\_NOTE\_SI】

5.2 设置蜂鸣器播放音乐



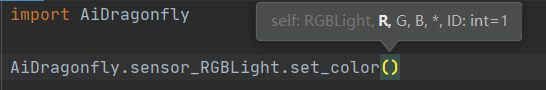
参数：

Music：音乐编号（需要和开发板程序对应）

【0 ~ 255】

1. RGB真彩灯

6.1 设置RGB真彩灯颜色值



参数：

R：红色颜色值

【0 ~ 255】

G：绿色颜色值

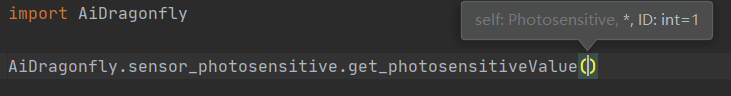
【0 ~ 255】

B：蓝色颜色值

【0 ~ 255】

1. 光敏传感器

7.1 获取 光敏传感器的光照强度



返回值：光照强度

【0 ~ 1024】

**特别说明：**

无源蜂鸣器、RGB真彩灯、光敏传感器 这三个传感器需要分别下载对应的程序才能使用。关于这三个传感器的具体使用方式请参考OneDuino的使用方式，资料链接如下：

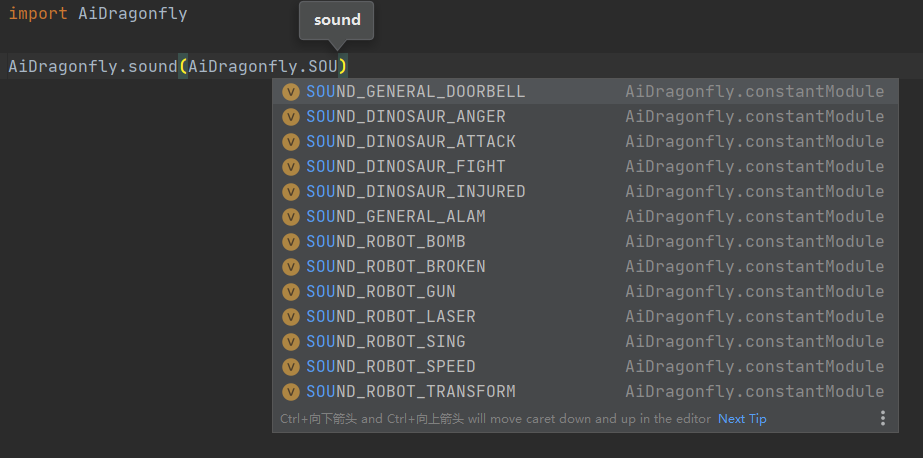
<https://gitee.com/AIQI-Developer/Development-Board>

* 音效模块Python API接口

主控音效的控制参数是通过常量来控制的。音效分为常规、机器人、恐龙、钢琴四大类

参数：

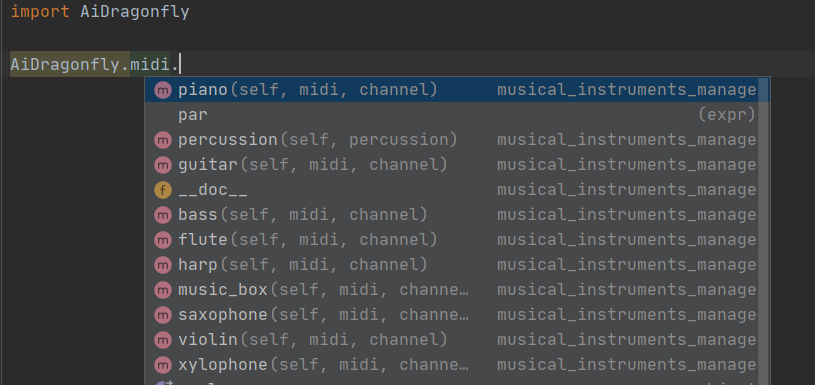
Sound：主控支持的音效。音效列表如下所示。



* 乐器模块Python API接口

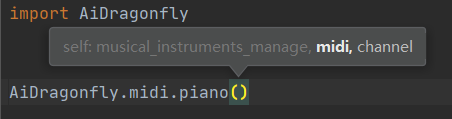
主控midi乐器的控制参数是通过常量来控制的。

midi乐器类型有钢琴、八音盒、贝斯、吉他、萨克斯、笛子、木琴、小提琴、竖琴、打击乐十种乐器。如下图所示



1. midi乐器：钢琴

除了打击乐之外，每种乐器都有两个参数，以钢琴为例进行介绍



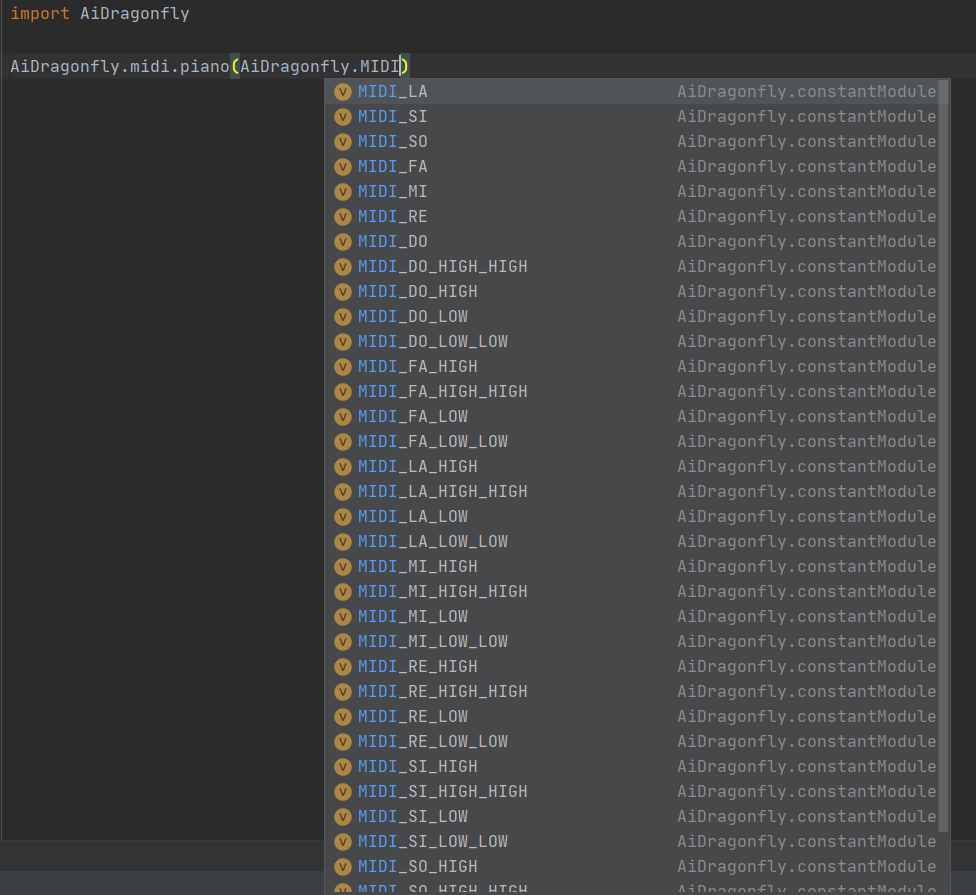
参数：

midi：乐器的音阶，七个基本音节的高高音、高音、中音、低音、低低音

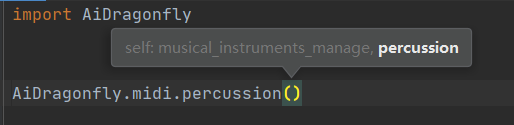
常量值如下图所示

channel：midi的通道值

【0 ~ 15】



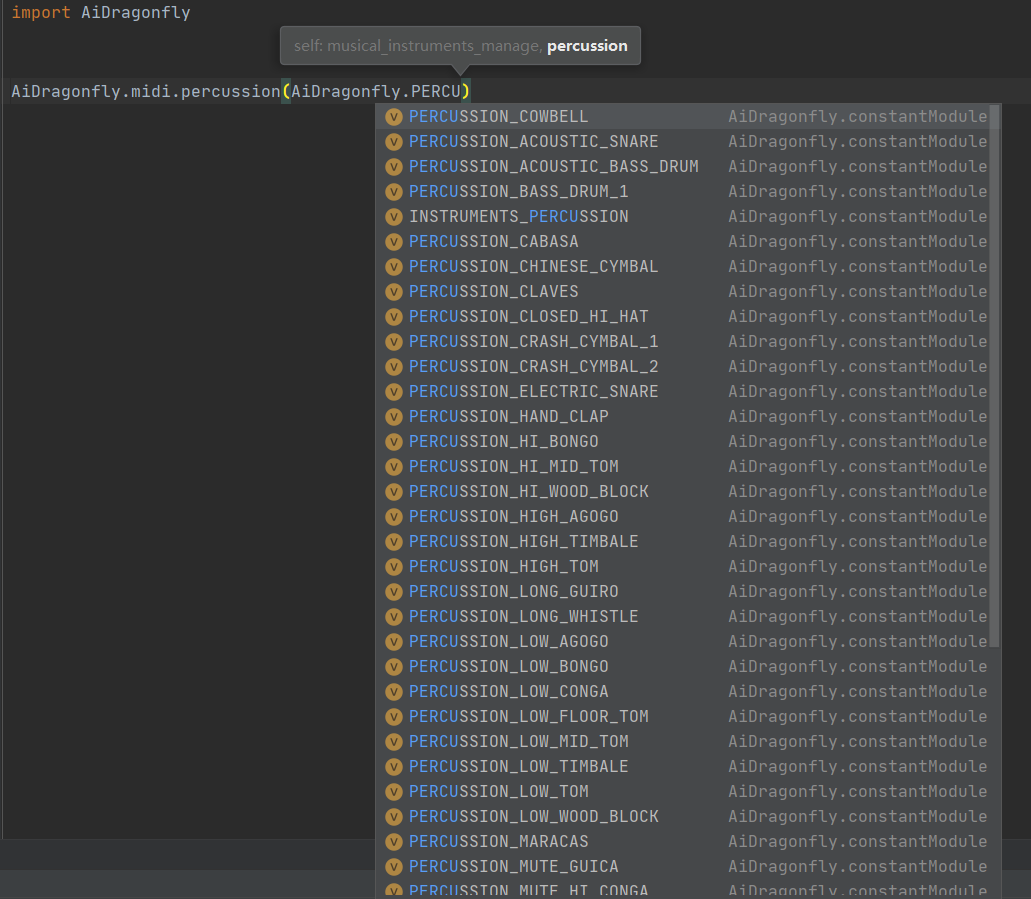
1. midi乐器：打击乐



参数：

Percussion：打击乐种类

参数如下图所示



附录二 Python集成开发环境的安装

* 安装Python集成开发环境PyCharm Community Edition

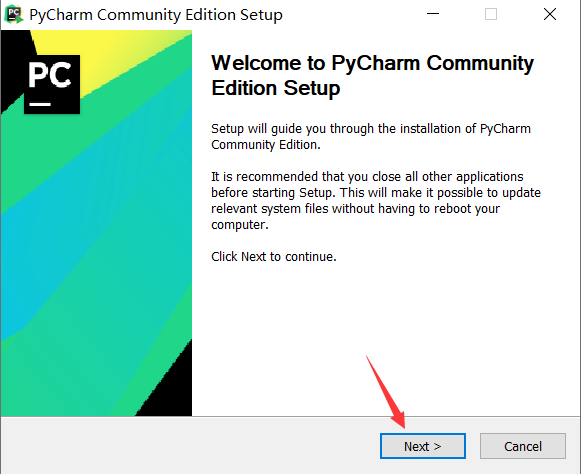
1. 准备工作

Python的集成开发环境PyCharm Community Edition请到官方网站下载：

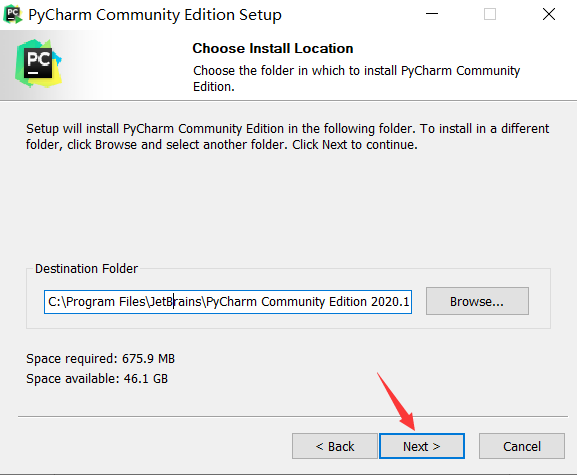
<https://www.jetbrains.com/pycharm/download/#section=windows>

（提示：首先根据所使用的不同平台进行选择，推荐下载 开源免费的community版本）

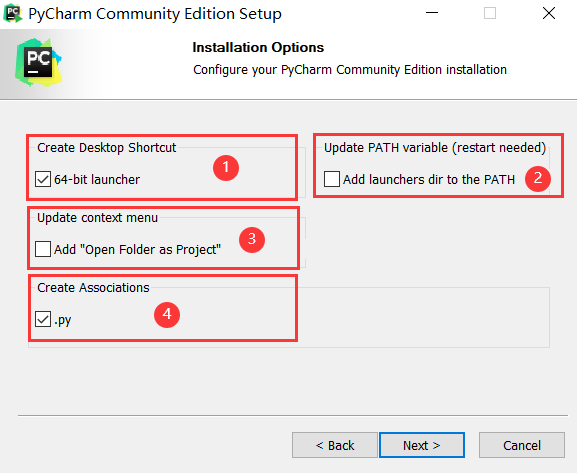
1. 双击安装包，出现如下界面，点击 Next 进行安装



1. 此界面为选择软件安装路径（可以选择默认），点击Next



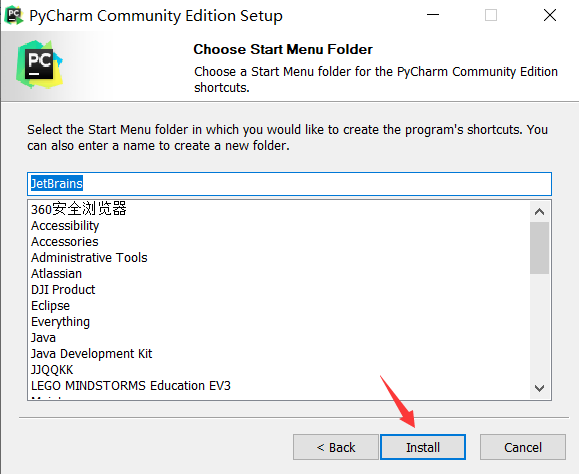
1. 点击Next 进入如下界面



数字1 创建桌面快捷方式；数字2 更新路径变量（需要重新启动）；数字3 更新上下文菜单 ；数字 4 创建关联 关联.py文件。

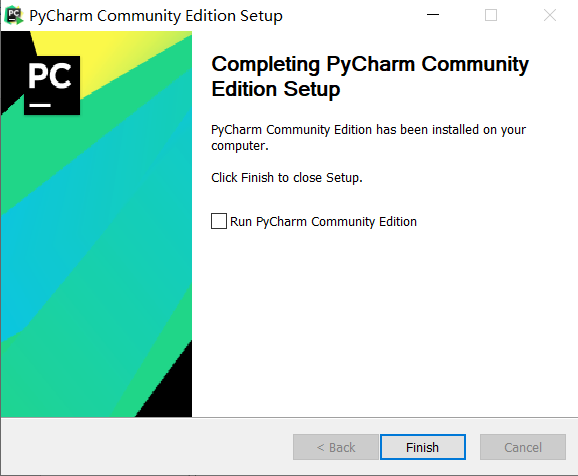
勾选数字1 和 数字4 就好，数字2 和 数字3 看自己需求。

1. 勾选数字1和4 后，点击Next，进入如下界面



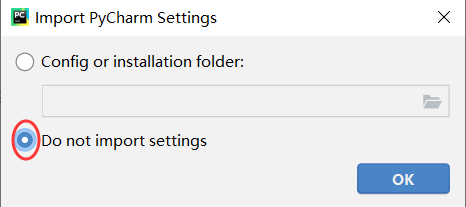
默认安装即可，直接点击Install即可。

1. 等待一下，安装完成，如下



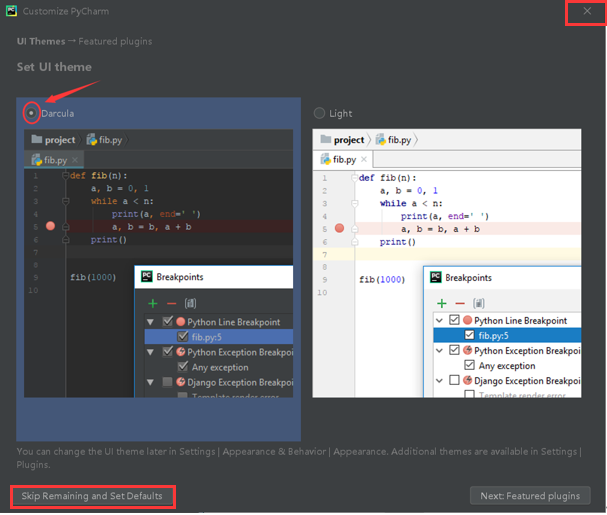
点击Finish安装完成。

1. 安装完成后，需要对Pycharm进行配置，双击桌面Pycharm图标，进入如下界面



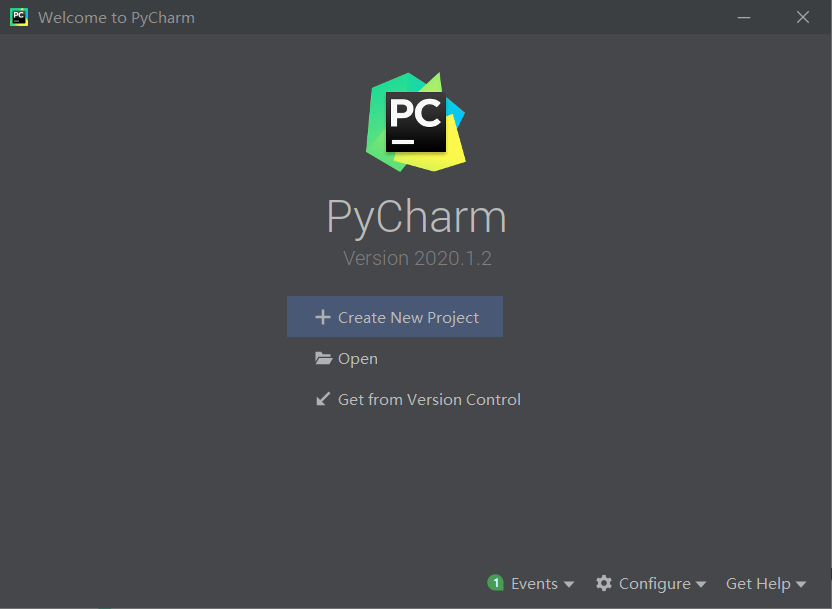
此界面 选择从哪里导入pycharm设置，选择Do not import settings，然后点击OK 进入下一步。

1. 点击OK 进入如下界面



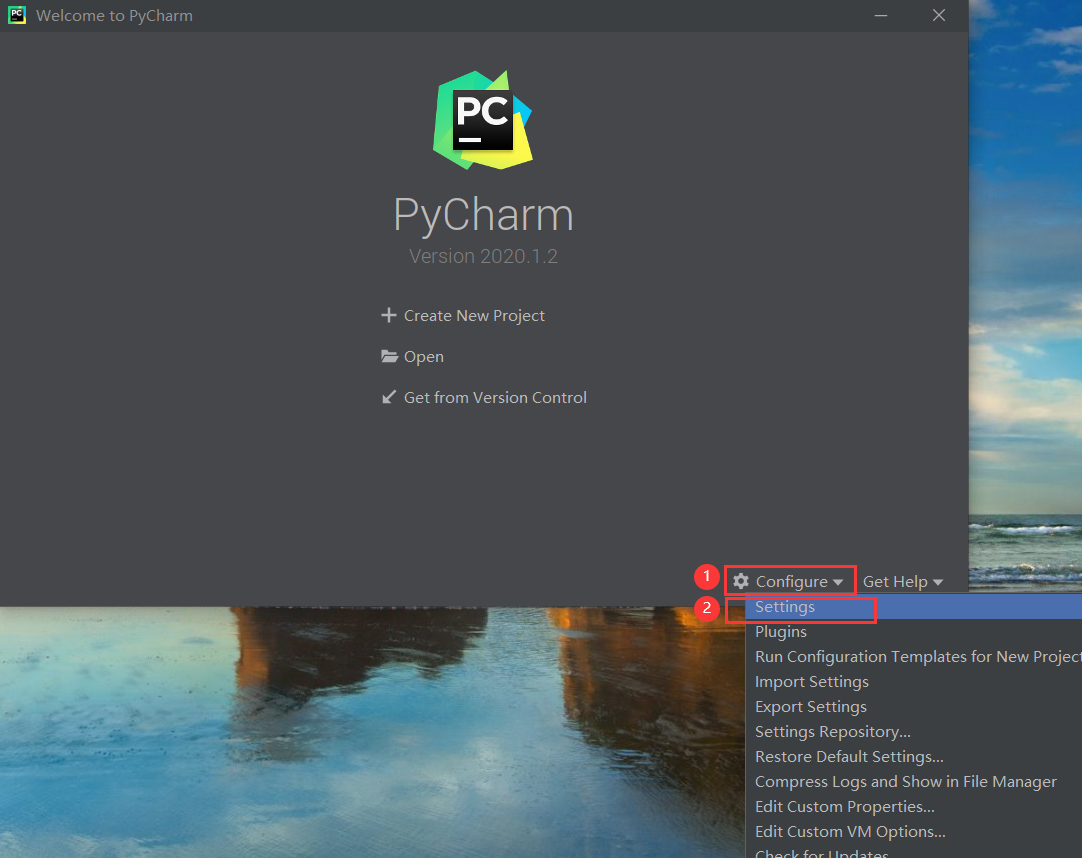
皮肤选择，根据自己喜好选择皮肤。然后点击右上角关闭按钮 或者左下角（跳过其余和设置默认值）。

1. 进入pycharm工作界面，安装完成



* 小试牛刀 开始 Hello World

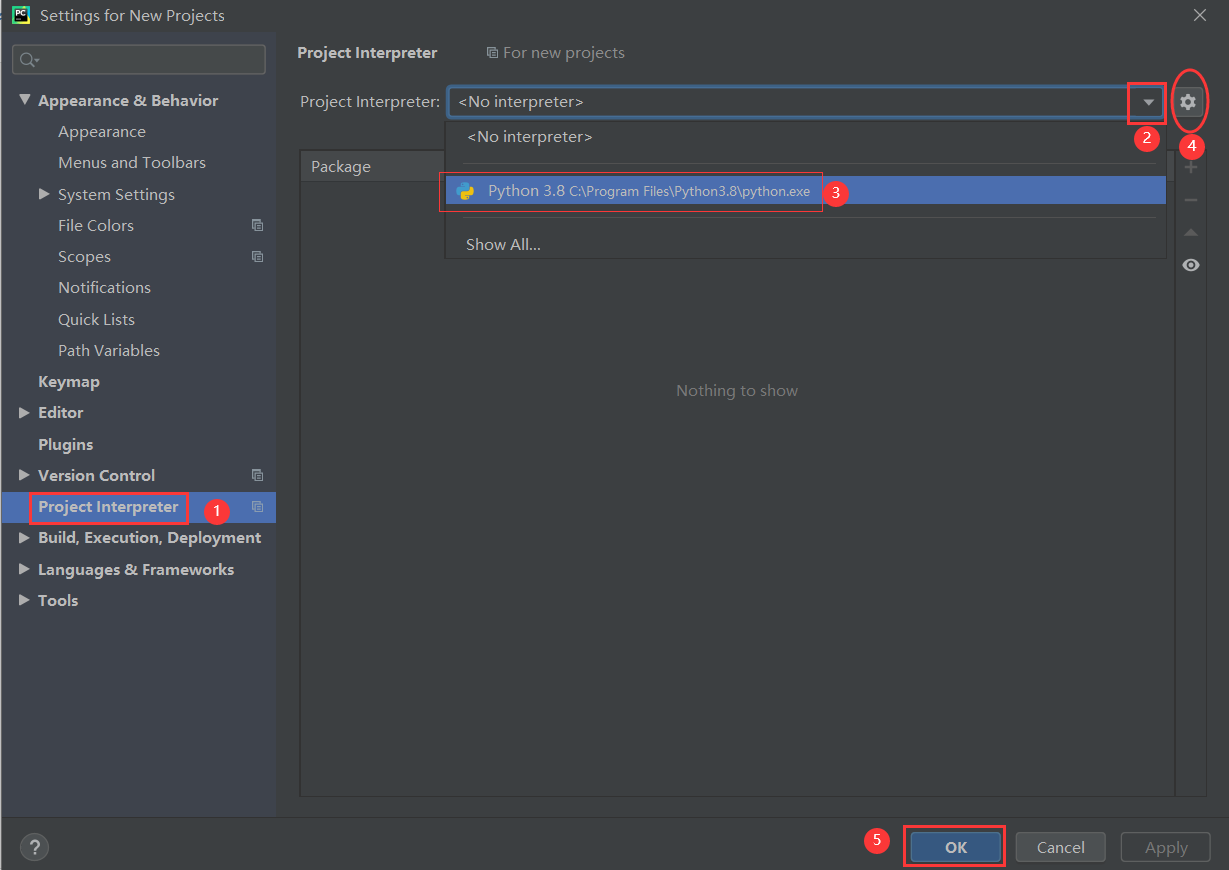
1. 双击桌面pycharm图标，进入欢迎界面



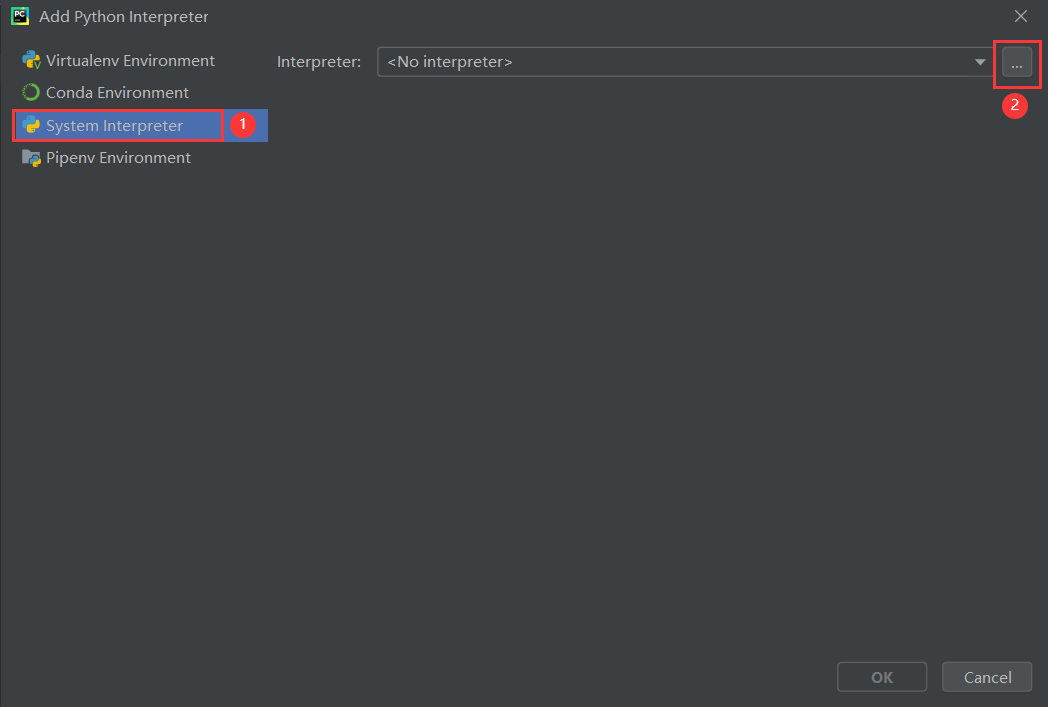
点击Configure（如数字1所示），选择Settings（如数字2）。

1. 配置工程解释器

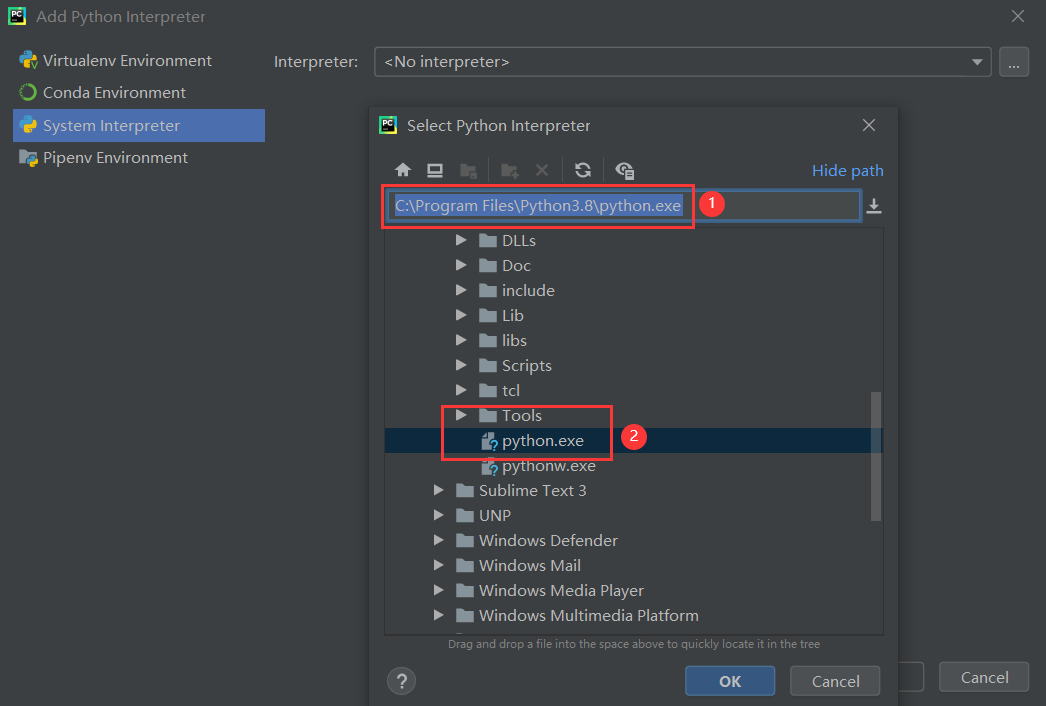
界面如下，点击Project Interpreter（如数字1所示），然后点击右上角的下拉小三角（如数字2所示）。如果安装了Python 3.8，一般会显示出python3.8解释器（如数字3所示）。选择即可。最后点击 OK (如数字5所示)。



如果没有出现解释器（类似数字3），则点击右上角的配置按钮（如数字4所示），出现如下界面。点击System Interpreter（如数字1所示），然后点击右上角的选择按钮（如数字2所示）。

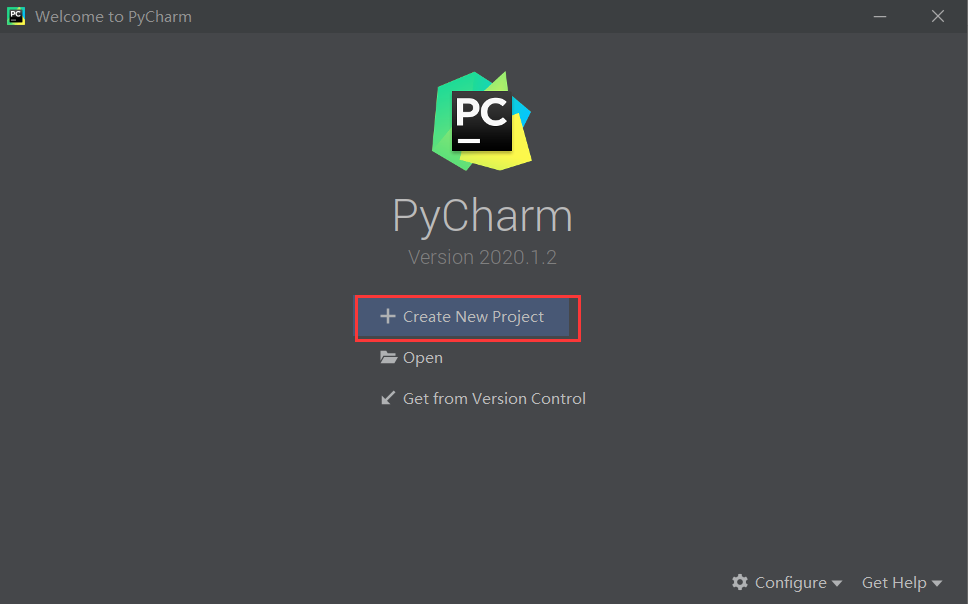


进入解释器选择界面（示例的python3.8装在了C盘），用户需要根据自己安装python3.8 时的路径进行选择，找到python.exe后，选择即可。



一路点击OK，再次回到欢迎界面。至此解释器配置完成。

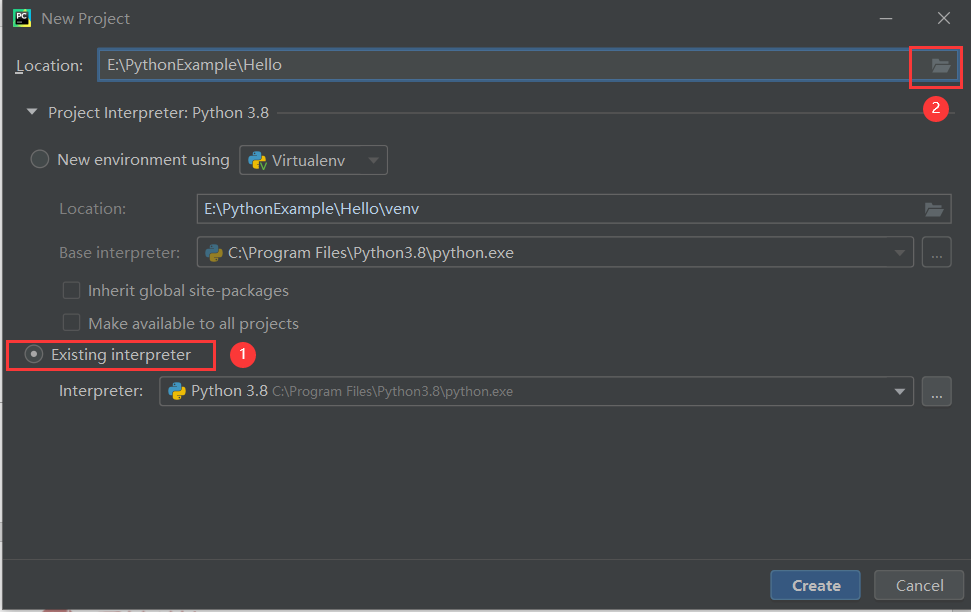
1. 创建工程 点击Create New Project



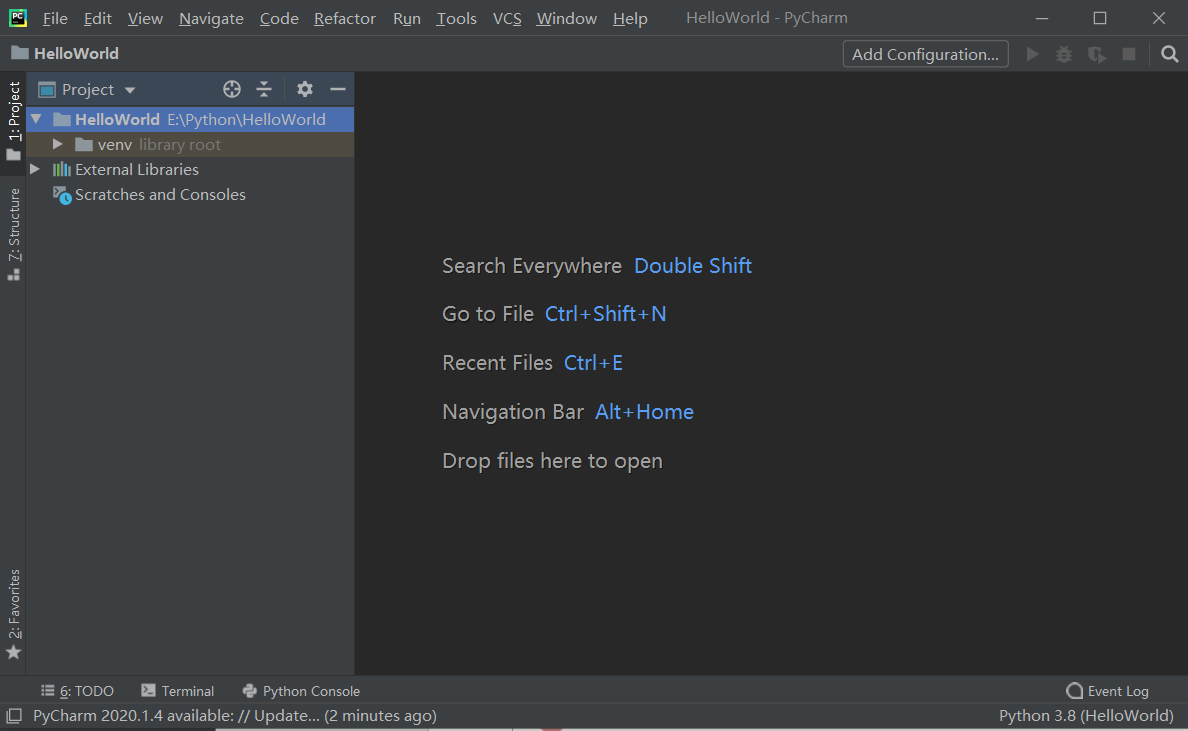
1. 选择工程路径及解释器

选择 Existing interpreter （如数字1所示），点击右上角的文件夹图标（如数字2所示），选择创建的python工程路径。（示例路径在E:\PythonExample\Hello）。

然后点击右下角的Create按钮，创建工程。

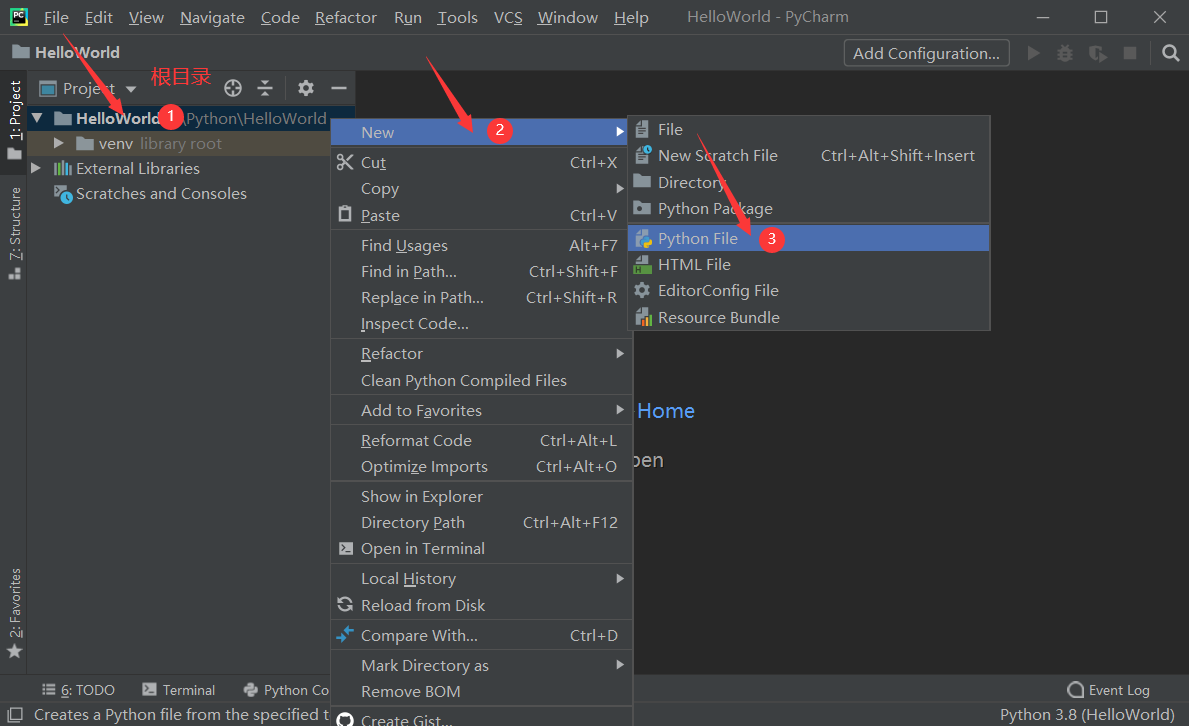


1. 项目创建完成



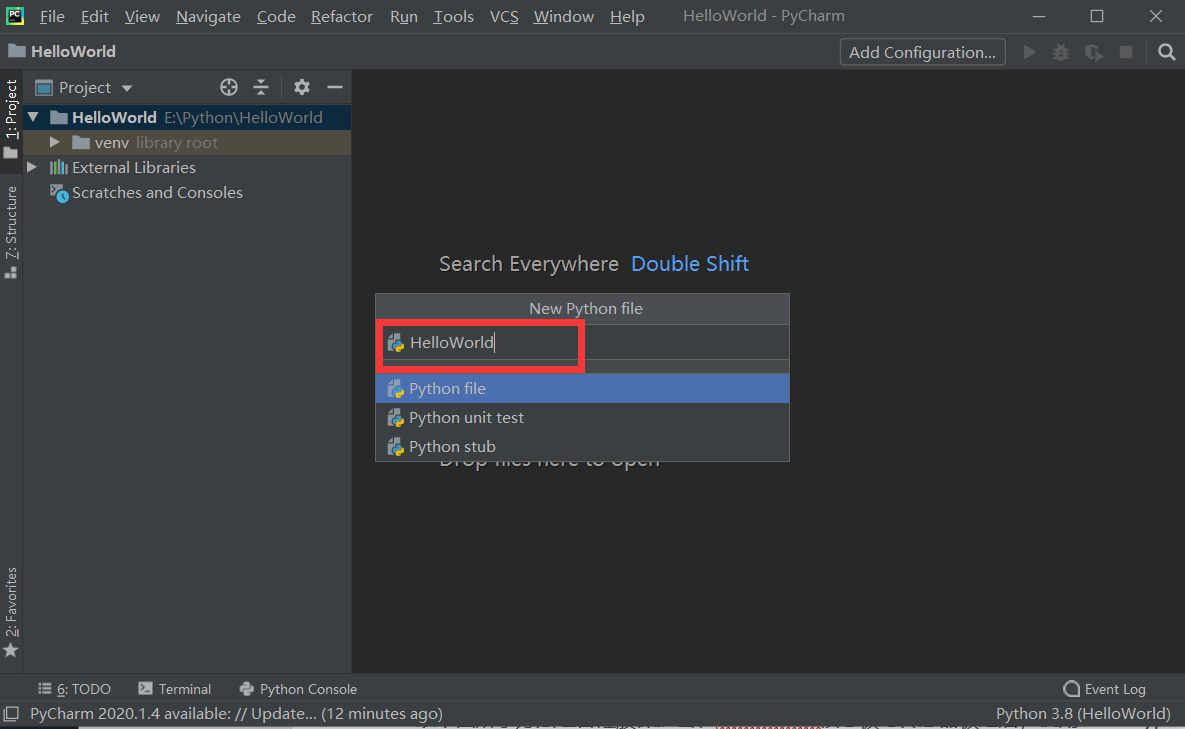
1. 创建python文件

在数字1根目录右键，然后依次点击New->Python File



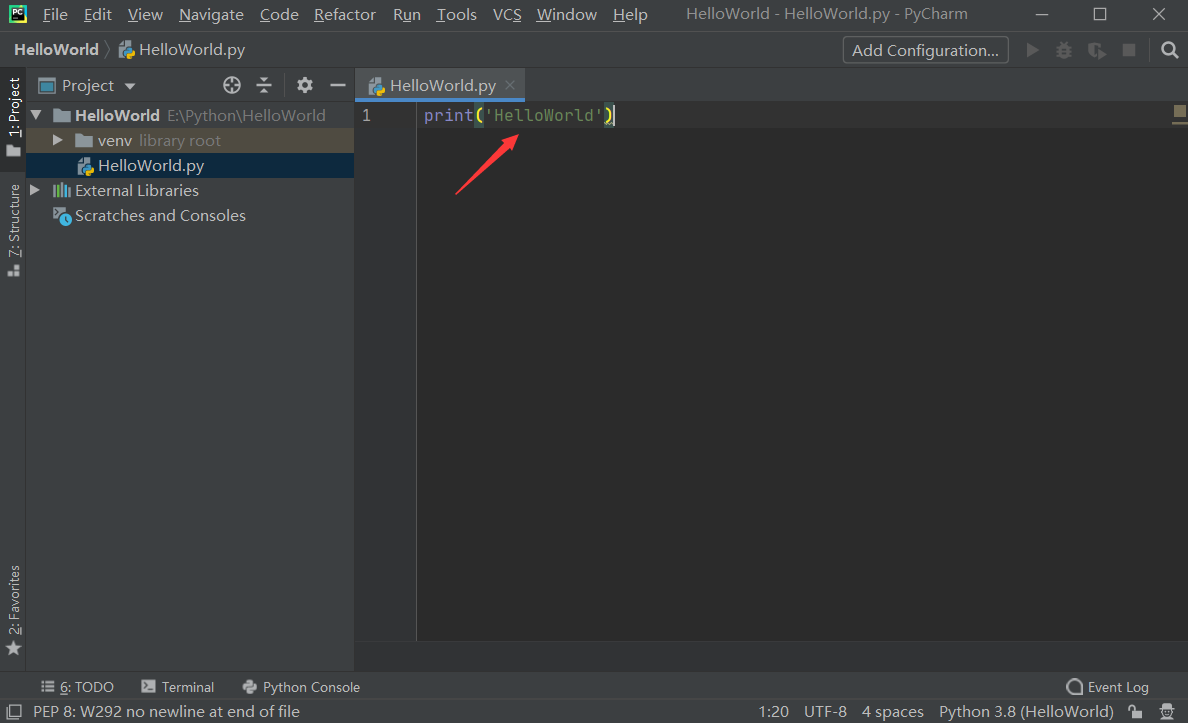
1. python文件命名

在对话框中输入文件名称如HelloWord，回车，至此python文件创建完成。

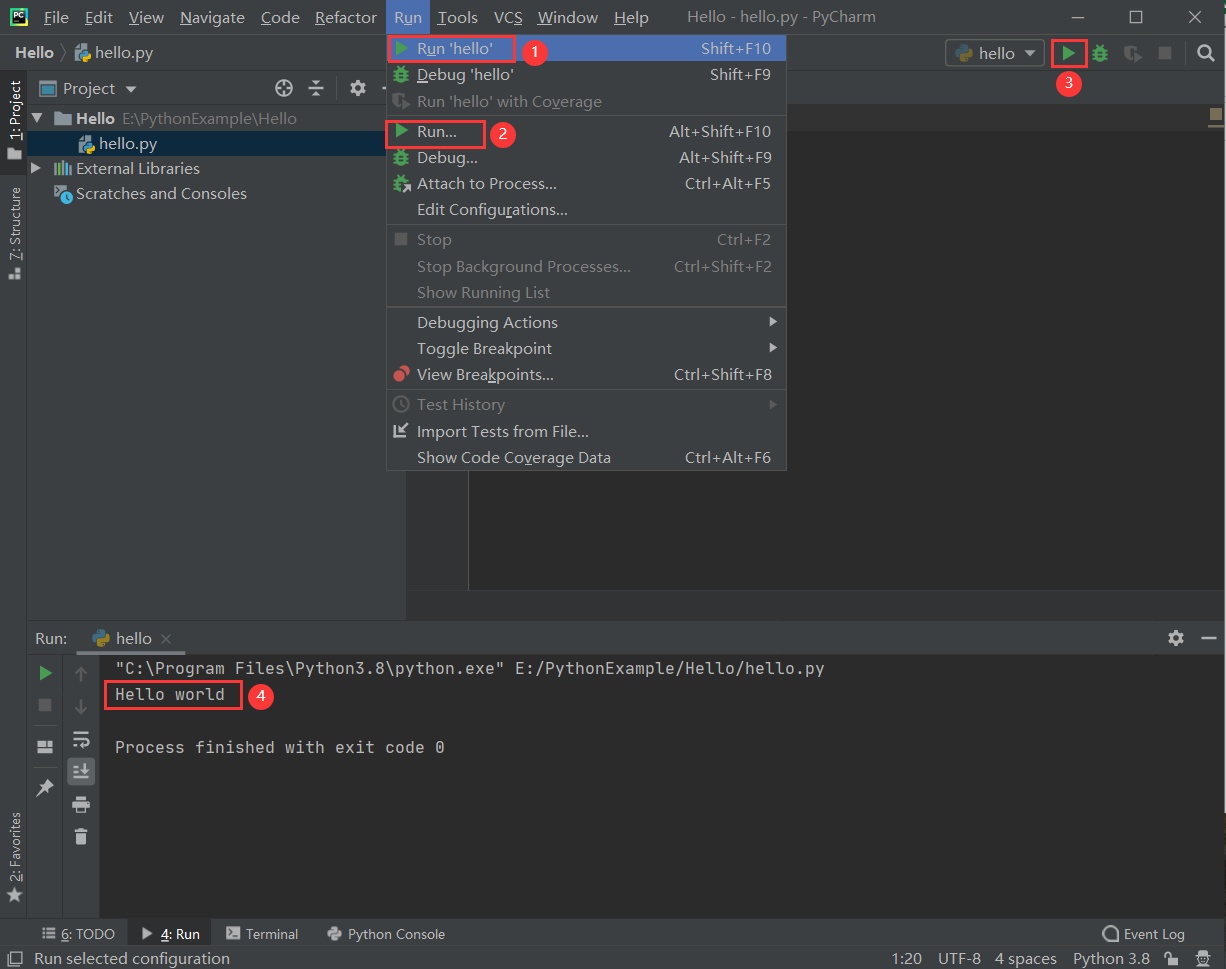


1. 体验python编程

在代码编辑区输入print(‘HelloWorld’)文件编辑完成。



运行python程序点击Run（如图中数字1 数字2 数字3 均可 运行），结果如数字4所示。



至此，python工程的创建、文件的编辑、运行就大功告成啦。

北京爱其科技有限公司 服务热线：400-0126-281

地址：北京海淀区上地三街9号D座1204

感谢您选择爱其科技

