**1介绍**

glob模块用来查找**文件目录和文件**，并将搜索的到的结果返回到一个列表中，

常见的两个方法有glob.glob()和glob.iglob()，可以和常用的find功能进行类比，glob支持\*?[]这三种通配符。

**2.三种通配符**

* \*代表0个或多个字符
* ?代表一个字符
* []匹配指定范围内的字符，如[0-9]匹配数字

**3、如需使用，使用前先导入glob，这里有两种方法导入glob**

1、import glob #导入整个glob模块

2、from glob import golb #从glob模块导入glob函数

**4示例**

#1、定义一个函数，搜索C盘下的所有文件和目录，并返回一个列表(这里使用import glob)

import glob

def search():

f = glob.glob(r'c:\\*')#glob.glob表示glob模块下的glob函数

print(f)

if \_\_name\_\_ == "\_\_main\_\_"：

search()

############################################

#2、定义一个函数查询E盘中以.exe结尾的文件有哪些（这里使用from glob import glob）：

from glob import glob#导入模块的同名函数

def search():

f = glob(r'e:\\*.exe')#直接调用函数

for i in f:

print(i)

if \_\_name\_\_ == "\_\_main\_\_":

search()

**5、glob.iglob() 函数**

获取一个可遍历对象，使用它可以逐个获取匹配的文件路径名。

与glob.glob()的区别是：glob.glob()可同时获取所有的匹配路径，而glob.iglob()一次只能获取一个匹配路径。

import glob

# 父目录中的.py文件

f = glob.iglob(r'../\*.py')

print(f) # generator object iglob as 0x00B9FF80

for py in f:

print(py)