## Os模块

### 环境变量函数

1. os.name 如果是windows操作系统返回’nt’，如果是其他系统则返回 ‘posix’
2. os.environ 返回系统的环境变量，以dict形式显示

### 文件操作函数

Python os 模块包含普遍的操作系统功能。

常用方法

Os.getcwd() 返回当前工作目录

Os.chdir(path) 改变工作目录

Os.listdir(path=’.’) 列举指定目录中的文件名(‘.’表示当前目录，’..’代表上一级目录)

Os.mkdir(path) 创建单层目录，如该目录已存在抛出异常

Os.makedirs(path) 递归创建多层目录，如果该目录已经存在则抛出异常。

Os.remove(path) 删除文件

Os.rmdir(path) 删除单层目录，如改目录非空则抛出异常

Os.removedirs(path) 递归删除目录，从子目录到父目录逐层尝试删除，遇到目录非空则抛出异常。

Os.rename(old,new) 将文件old 重命名为new，文件和目录都使用这条命令

Os.path 模块中关于路径常用的函数使用方法：

Os.path.basename(path) 去掉目录路径，单独返回文件名

Os.path.dirname(path) 去掉文件名，单独返回目录路径

Os.path.join(path1,path2) 将path1,path2各部分组合成一个路径名

Os.path.split(path) 分割文件名和路径，返回（f\_path,f\_name）元组。如果完全使用目录，它也会将最后一个目录作为文件名分离，且不会判断文件或者目录是否存在。

Os.path.splitext(path)分离文件名与扩展名，返回（f\_name,f\_extension）元组

Os.path.getsize(file)返回指定文件的尺寸，单位是字节

Os.path.getatime(file) 返回指定文件最近的访问时间（浮点型秒数，可用time模块的gmtime()或localtime()函数换算）

Os.path.getctime(file) 返回指定文件创建时间（浮点型秒数，可用time模块的gmtime()或localtime()函数换算）

Os.path.getmtime(file) 返回指定文件最新的修改时间（浮点型秒数，可用time模块的gmtime()或localtime()函数换算）

返回True或False的函数

Os.path.exists(path) 判断指定路径（目录或文件）是否存在

Os.path.isabs(path) 判断指定路径是否为绝对路径

Os.path.isdir(path) 判断指定路径是否存在且是一个目录

Os.path.isfile(path) 判断指定路径是否存在且是一个文件

## shutil模块

复制文件

shutil.copyfile(‘oldfile’,’newfile’) oldfile和newfile都只能是文件

shutil.copy(‘oldfile’,’newfile’) oldfile只能是文件，newfile可以是文件也可以是目标目录

复制文件夹

shutil.copytree(‘olddir’,’newdir’),olddir和newdir都只能是目录，且newdir必须不存在

移动文件（目录）

shutil.move(‘oldname’,’newname’)

删除目录

os.rmdir(‘dir’)只能删除空目录

shutil.rmtree(‘dir’)空目录，有内容的目录都可以删

## Sys模块

Sys.argv

用来向python解释器传递参数，名曰“命令行参数”

Sys.exit()

退出当前程序

Sys.stdout

与Python中的函数功能对照，sys.stdin获得输入，等价于python2中的raw\_inpurt()，python3中的input(),sys.stdout负责输出

Sys.path

返回python目录下所有.pth路径文件下的内容集系统默认设置。

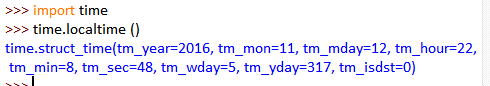
可以通过列表的操作对其进行修改，不过这种更改只对当前的程序起作用。

## Time模块

时间获取和转换

时间元组（time.struct\_time）

Gettime(),localtime()和striptime()以时间元组（struct\_time）的形式返回



返回的元组的内容

|  |  |  |
| --- | --- | --- |
| 索引值(index) | 属性(attribute) | 值(value) |
| 0 | tm\_year(年) | 例如：2016 |
| 1 | tm\_mon(月) | 1~12 |
| 2 | tm\_mday(日) | 1~31 |
| 3 | tm\_hour(时) | 0~23 |
| 4 | tm\_min(分) | 0~59 |
| 5 | tm\_sec(秒) | 0~61 |
| 6 | tm\_wday(星期几) | 0~6(0代表星期一) |
| 7 | tm\_yday(一年中的第几天) | 1~366 |
| 8 | tm\_isdst(是否为夏令时) | 0，1，-1(-1代表夏令时) |

time.asctime([t])

接收时间元组并返回一个可读的形式。



time.clock()

用以浮点数计算的秒数返回当前的cpu时间,用来衡量不同程序的耗时

time.time()

返回当前时间的时间戳，可以用来计算程序的耗时

时间戳(timestamp)表示的是从1970年1月1日00：00：00开始按秒计算的偏移量（time.gmtime(0)）

time.sleep(secs)

推迟调用线程的运行，secs的单位是秒