



文件和文件夹处理

北京理工大学计算机学院 高玉金

2019年3月



处理文件和文件夹

- Python的open(“test.txt”)函数中，可以指定具体的文件路径，但并不灵活
- 通过Python的标准os库提供通用的、基本的操作系统交互操作
- 通过glob库进行文件查找
- 通过shutil库对文件进行增强操作



使用内置os库

- 获取当前目录（相对）——`os.getcwd()`
- 获取当前目录（绝对）——`os.path.abspath()`
- 创建子目录——`os.mkdir()` 和 `os.makedirs("path")`
- 更改当前目录——`os.chdir(name)`
- 改名——`os.rename(old, new)`
- 将路径分解为目录名和文件名——`os.path.split(fullname)`
- 将路径合并——`os.path.join(path, file)`
- 分解扩展名——`fname, fext = os.path.splitext("分解路径")`

```
>>> a, b=os.path.split("c:\\dir1\\dir2\\file.txt")
>>> print(a)
c:\\dir1\\dir2
>>> print(b)
file.txt
```



使用内置os库

- 判断一个路径（目录或文件）是否存在——
`os.path.exists(path)`
- 判断一个路径是否是文件——`os.path.isfile("文件")`
- 判断一个路径是否存在——`os.path.isdir("路径")`
- 获取目录中的文件及子目录的列表——`os.listdir("路径")`
- 删除子目录——`os.rmdir("path")`，只能删除空目录
- 目录树遍历——`root, dirs, files = os.walk(top)`
- `os.walk()` 方法是一个简单易用的文件、目录遍历器，可以帮助我们高效的处理文件、目录方面的事情



os.walk详解

- **top**是要遍历的目录地址， 返回一个三元组 (root,dirs,files):
 - root ——当前正在遍历的这个文件夹的本身的地址
 - dirs ——一个 list ， 包含该文件夹中所有目录的名字(不包括子目录)

```
import os
from os.path import join, getsize
for root, dirs, files in os.walk('python/Lib/email'):
    print(root, "consumes", end="")
    print(sum([getsize(join(root, name)) for name in files]), end="")
    print("bytes in", len(files), "non-directory files")
    if 'CVS' in dirs:
        dirs.remove('CVS') # don't visit CVS directories
```




使用glob 库进行文件查找

- glob模块是最简单的模块之一
- 用它可以查找符合特定规则的文件路径名
- 只用到三个匹配符：“*”，“?”，“[]”。“*”匹配0个或多个字符；“?”匹配单个字符；“[]”匹配指定范围内的字符，如：
[0-9]匹配数字
- 返回所有匹配的文件路径列表
- 应用：批量修改符合规则的文件名

```
#获取指定目录下的所有图片
print glob.glob(r"E:/Picture/*/*.jpg")

#获取上级目录的所有.py文件
print glob.glob(r'../*.py') #相对路径
```

```
1. >>> import glob
2. >>> glob.glob('./[0-9].*')
3. ['./1.gif', './2.txt']
4. >>> glob.glob('*.gif')
5. ['1.gif', 'card.gif']
6. >>> glob.glob('?.gif')
7. ['1.gif']
```



文件处理增强SHUTIL

- os模块提供了对目录或者文件的新建/删除/查看文件属性，以及对文件以及目录的路径操作
- Shutil是高级文件操作模块，对文件移动，复制，打包，压缩和解压等
- `shutil.copyfile(src,dst)` #copy文件内容
- `shutil.move(src, dst)` #递归的移动文件
- `shutil.make_archive()` #压缩打包

```
1  shutil.copy("test.file", "test.copy.file")
2
3  print("test.file" in os.listdir(os.curdir))
4  print("test.copy.file" in os.listdir(os.curdir))
```



综合示例

- 假设你是一名学院春季运动会的摄影志愿者，需要把拍摄的高清照片发给学院编辑选择使用。为了减少传输的数据量，请你先把这些图片生成缩略图，打包发给编辑，待她选择之后再选中的原图发给她。
- 1、高清图片存在当前目录下的多个文件夹内
 - 2、把所有的**高清**图片生成的缩略图保存到一个新文件夹内
 - 3、保持生成的图片名称为：缩略图_原名
 - 4、把缩略图压缩打包为一个zip文件（邮件发送*）



综合示例

```
path="c:\\diegoTemp"
os.chdir(path)

if not os.path.exists(path+"\\newJpgs") :
    os.mkdir("newJpgs")

for d,subd,files in os.walk(path+"\\运动会照片") :
    jpgs = glob.glob(d+"\\*.jpg")
    for jpg in jpgs:
        im = Image.open(jpg)
        im.thumbnail((312,312))
        p,n = os.path.split(jpg)
        im.save(path+"\\newJpgs\\缩略图_{}".format(n))
    else:
        shutil.make_archive("imgs","zip",root_dir = path+"\\newJpgs")
```

```
import os
from PIL import Image
import shutil
import glob
```

```
for jpg in jpgs:
    if os.path.getsize(jpg)>1024*1024:
        im = Image.open(jpg)|
        im.thumbnail((312,312))
```

```
#p,n = os.path.split(jpg)
im.save(path+"\\newJpgs\\缩略图_{1}".format(*os.path.split(jpg)))
```