**py 1 os、shutil--路径、文件夹、文件**

**列出文件夹内的所有文件**

*#方法1：使用os.listdir*

**import** os

**for** filename **in** os.listdir(r'c:\windows'):

**print** filename

*#方法2：使用glob模块，可以设置文件过滤*

**import** glob

**for** filename **in** glob.glob(r'c:\windows\\*.exe'):

**print** filename

*#方法3：通过os.path.walk递归遍历，可以访问子文件夹*

**import** os.path

**def** **processDirectory** ( args, dirname, filenames ):

**print** 'Directory',dirname

**for** filename **in** filenames:

**print** ' File',filename

os.path.walk(r'c:\windows', processDirectory, **None** )

*#方法4：非递归*

*#!/bin/python*

*#*

*#site:www.jbxue.com*

**import** os

**for** dirpath, dirnames, filenames **in** os.walk('c:\\winnt'):

print 'Directory', dirpath

**for** filename **in** filenames:

print ' File', filename

**一、路径**

import os

**# 当前工作目录**

os.getcwd()

**# 改变工作目录到dirname**

os.chdir('C:\Users\Python\_Folder')

**# 返回当前工作目录**

os.curdir

os.\_\_file\_\_

# D:\envs\py27\lib\os.pyc

# 获取路径名

os.path.dirname(os.\_\_file\_\_)

# 获取文件名

os.path.basename(os.\_\_file\_\_)

# 获得绝对路径

os.path.abspath(os.\_\_file\_\_)

# 获取相对路径

os.path.realpath(os.\_\_file\_\_)

# 路径是否真地存在

os.path.exists('D:\envs\py27\lib')

# 路径是否是一个文件

os.path.isfile('D:\envs\py27\lib\os.pyc')

# 路径是否是一个目录

os.path.isdir('D:\envs\py27\lib')

# 判断是否是绝对路径

os.path.isabs('D:\envs\py27\lib')

**二、文件夹**

import shutil

**1 新建与删除**

*# 创建单个目录*

**if** **not** os.path.exists("python27"):

os.mkdir("python27")

*# 创建多级目录*

**if** **not** os.path.exists("python26"):

os.makedirs(r'python26\test')

*# 创建多级目录*

**if** **not** os.path.exists("python25"):

os.makedirs(r'python25\nofile')

*# 只能删除空目录*

**if** os.path.exists("python27"):

os.rmdir("python27")

*# 删除多个目录*

**if** os.path.exists("python26"):

os.removedirs(r'python26\test')

*# 空目录、有内容的目录都可以删 。*

*# 递归删除一个目录以及目录内的所有内容*

**if** os.path.exists("python25"):

shutil.rmtree("python25")

**2 改名、复制、移动**

**# 重命名文件（目录）：**

**# 文件或目录都是使用这条命令**

if os.path.exists("python26"):

os.rename("python26","python21")

# os.rename("oldname","newname")

**# 把olddir拷贝一份newdir：olddir和newdir都只能是目录，且newdir必须不存在**

**# 递归的去拷贝文件夹。相当于在原目录下，将文件夹复制之后重命名**

**# 如果第3个参数是True，则复制目录时将保持文件夹下的符号连接，**

**# 如果第3个参数是False，则将在复制的目录下生成物理副本来替代符号连接**

if os.path.exists("python21"):

shutil.copytree("python21","python22")

# shutil.copytree('olddir', 'newdir', ignore=shutil.ignore\_patterns('\*.pyc', 'tmp\*'))

**# 相当于在原目录下，将文件夹直接重命名**

if os.path.exists("python21"):

shutil.move("python21","python20")

# shutil.move("oldpos","newpos")

**三、文件**

**1 修改、改名、复制、移动、删除**

**# 获取文件属性**

os.stat(fil)

**# 更新文件时间戳**

os.utime()

**# 创建并打开（w+b）一个新的临时文件**

os.tmpfile()

**# 获取文件大小**

os.path.getsize(filename)

**# 访问权限**

os.access('C:\\Users\Desktop\ex3.txt',1)

**# True 检验用户是否对文件有访问权限**

**# 修改文件权限与时间戳**

os.chmod()

**# 修改属主，属主**

os.chown()

**# 设置默认权限模式**

os.umask()

**# 创建设备**

os.makedev()

**# 创建设备文件**

os.mknod()

**# 复制文件：oldfile 和 newfile 都只能是文件**

shutil.copyfile("oldfile","newfile")

**# 将文件内容拷贝到另一个文件中**

shutil.copyfileobj(open('old.xml','r'),\

open('new.xml', 'w'))

**# 拷贝文件和权限**

shutil.copy('f1.log', 'f2.log')

**# 拷贝文件和状态信息**

shutil.copy2('f1.log', 'f2.log')

**# 拷贝文件**

shutil.copyfile('f1.log', 'f2.log')

**# 仅拷贝权限。内容、组、用户均不变**

shutil.copymode('f1.log', 'f2.log')

**# 仅拷贝状态的信息，包括：**

**# mode bits, atime, mtime, flags**

shutil.copystat('f1.log', 'f2.log')

**# 删除一个文件**

os.remove()

**# 递归的去删除文件**

shutil.rmtree('folder1')