# 打开pydoc

(1)Cmd:

Python -m pydoc -p 12345

(2)浏览器中输入：<http://localhost:12345/>

# Openpyxl 操作EXCEL

## 打开excel wb= load\_workbook(filename, read\_only=False, keep\_vba=KEEP\_VBA, data\_only=False, keep\_links=True)

返回rtype : class: openpyxl.workbook.Workbook

例子:wb=openpyxl.load\_workbook(r‘D:\test.txt’)

## 取得excel所有sheet名字 wb.get\_sheet\_names()

返回：一个包含所有sheet名字的List

例子：for sheetName in wb.get\_sheet\_names():

## 返回某个sheet wb[sheetName]

返回：一个sheet对象

例子：ws=wb[‘sheet1’]

## 删除某个sheet wb.remove（sheet对象）

例子：wb.remove(wb[‘sheet1’])

## 获取一个sheet的最大行 sheet.max\_row

## 获取sheet中的一个cell sheet.**cell**(self, row, column, value=None)

返回：一个cell对象

例子：sheet.cell(row=i+1,column=13).value='haha'

## 取得或者设置一个单元格的值 cell.**value**

例子：sheet.cell(row=i+1,column=13).value='haha'

## 修改完成后保存到文件 wb.save(filename)

# Os 操作文件和目录

## 取得一个目录下所有文件和目录名字 os.listdir(rootPath)

## 判断是否是目录 os.path.isdir(filename)

# 文件读写

## **open(filename，mode=’r’)** 读取一个文件

参数mode：'r' open for reading (default)  
'w' open for writing, truncating the file first  
'x' create a new file and open it for writing  
'a' open for writing, appending to the end of the file if it exists  
'b' binary mode  
't' text mode (default)  
'+' open a disk file for updating (reading and writing)  
'U' universal newline mode (deprecated)

返回：一个file对象

**例子：helloFile = open('C:\\Users\\*your\_home\_folder*\\hello.txt')**

## read() 将文件中的内容读入内存

例子：**helloFile = open('C:\\Users\\*your\_home\_folder*\\hello.txt')**

**StrRead=helloFile.read()**

## readlines() 一行一行读取

返回：所有行组成的List.

## write() 写入文件

# 日期和时间

## time

每个时间戳都以自从1970年1月1日午夜（历元）经过了多长时间来表示。

Python 的 time 模块下有很多函数可以转换常见日期格式。如函数time.time()用于获取当前时间戳,

### time.time()

返回：当前时间的时间戳(秒)

例子ticks = time.time()

print "当前时间戳为:", ticks

返回：当前时间戳为: 1459994552.51

### time.localtime(second)

参数second: 时间戳，秒。不带参数表示当前时间

返回：一个struct\_time元组

|  |  |  |
| --- | --- | --- |
| **序号** | **属性** | **值** |
| 0 | tm\_year | 2008 |
| 1 | tm\_mon | 1 到 12 |
| 2 | tm\_mday | 1 到 31 |
| 3 | tm\_hour | 0 到 23 |
| 4 | tm\_min | 0 到 59 |
| 5 | tm\_sec | 0 到 61 (60或61 是闰秒) |
| 6 | tm\_wday | 0到6 (0是周一) |
| 7 | tm\_yday | 1 到 366(儒略历) |
| 8 | tm\_isdst | -1, 0, 1, -1是决定是否为夏令时的旗帜 |

例子：

import time

localtime = time.localtime(time.time())

print "本地时间为 :", localtime

以上实例输出结果：

本地时间为 : time.struct\_time(tm\_year=2016, tm\_mon=4, tm\_mday=7, tm\_hour=10, tm\_min=3, tm\_sec=27, tm\_wday=3, tm\_yday=98, tm\_isdst=0)

### time.strftime(format[, t])

格式化日期

例子：

# 格式化成2016-03-20 11:45:39形式

print time.strftime("%Y-%m-%d %H:%M:%S", time.localtime())

# 格式化成Sat Mar 28 22:24:24 2016形式

print time.strftime("%a %b %d %H:%M:%S %Y", time.localtime())

# 将格式字符串转换为时间戳

a = "Sat Mar 28 22:24:24 2016" print time.mktime(time.strptime(a,"%a %b %d %H:%M:%S %Y"))

# requests, bs4

## res=requests.get(net,timeout=35) 获取网站

例子：res= requests.get(‘http://www.baidu.com’,timeout=35)

Print(res.text) #打印出网站内容

## soup=bs4.BeautifulSoup(res.text) 根据网站内容生成解析对象

## tablist=soup.select('css语言') 根据css过滤出所有标签的集合

返回：所有标签内容的集合。

## str=tab.get(‘href’) 取得该标签名为href的属性

返回：标签名为href的属性的内容。为字符串

例子：str=soup.select('table#archiveResult')[0].get(‘href’)

## str=tab.getText() 取得该标签的内容

返回：标签的内容。为字符串。