|  |  |  |
| --- | --- | --- |
| **目录树字符** | **├──** | **└──** |

**查找glob：**

|  |  |
| --- | --- |
| **glob(pathname, recursive)** | **返回满足表达式的路径** |
| **\*：多个任意字符** | **?：单个字符** |

**路径pathlib (>= 3.4)：**

|  |  |  |
| --- | --- | --- |
| **Path(str)** | | **实例化路径实例** |
| **类方法** | **cwd()** | **当前工作目录** |
| **home()** | **用户主目录** |
| **实例属性** | **drive** | **盘符** |
| **parent** | **前级路径** |
| **name** | **有后缀文件名** |
| **stem** | **无后缀文件名** |
| **suffix** | **后缀名称** |
| **实例方法** | **stat()** | **文件状态** |
| **absolute() / resolve()** | **绝对路径** |
| **relative\_to(other)** | **相对路径** |
| **with\_name(name)** | **改变名称** |
| **with\_stem(stem)** | **改变文件名** |
| **with\_suffix(suffix)** | **改变后缀** |
| **rename(target)** | **重命名** |
| **目录操作** | **is\_dir()** | **判断** |
| **mkdir(exist\_ok)** | **创建** |
| **rmdir()** | **删除** |
| **match(pattern)** | **匹配 (自右向左)** |
| **iterdir()** | **目录子项** |
| **glob(pattern) / rglob(pattern)** | **子树搜索** |
| **文件操作** | **is\_file()** | **判断** |
| **unlink(missing\_ok)** | **删除** |
| **open(mode, encoding)** | **打开** |
| **read\_text(encoding)** | **读取文本** |
| **read\_bytes()** | **读取字节** |
| **write\_text(data, encoding)** | **写入文本** |
| **write\_bytes(data)** | **写入字节** |

**高级shutil：**

|  |  |  |  |
| --- | --- | --- | --- |
| **move(tree, path)** | **剪切文件夹到指定目录** | | **返回最**  **终文件**  **的路径** |
| **copy(file, path)** |  | **拷贝文件**  **到指定目录** |
| **copy2(file, path)** | **保留原有信息** |
| **copyfile(file1, file2)** | **文件均需存在** | **写入拷贝的内容** |
| **copyfileobj(file1, file2)** | **文件均需open()** |  |
| **copystat(file1, file2)** | **文件均需存在** | **写入拷贝的状态信息** | |
| **copymode(file1, file2)** | **写入拷贝的文件权限** | |
| **copytree(path1, path2)** | **拷贝文件夹到新建目录** | | |
| **rmtree(path)** | **删除非空文件夹** | | |
| **which(command)** | **返回cmd命令文件路径** | | |
| **disk\_usage("C:")** | **返回磁盘使用情况** | | |

数据sqlite3/pymysql：

数据库url：mysql://<user>:<pwd>@<host>:<port>/<database>

|  |  |
| --- | --- |
| install\_as\_MySQLdb() | 充当MySQLdb模块 |

|  |  |  |
| --- | --- | --- |
| 游标： | | |
| connect(host, user, password, database) | | 连接数据库 |
| 实例方法 | cursor() | 创建游标 |
| commit() | 提交更改 |
| close() | 关闭 |
| 游标方法 | execute(query) | 返回行数 |
| fetchall() | 返回结果 |

**临时tempfile：**

|  |  |  |
| --- | --- | --- |
| **TemporaryDirectory()** | | **临时目录管理器** |
| **实例方法** | **cleanup()** | **清空目录** |

**格式文件：**

**bib：**

|  |  |  |  |
| --- | --- | --- | --- |
| bibtexparser： | | | |
| **dump(data, file\_obj)** | | **将参考文献写入到bib文件** | |
| **dumps(data)** | | **返回数据转换的bib文本流** | |
| **load(file\_obj)** | | **返回解析结果** | **bib文件** |
| **loads(str)** | | **bib文本流** |
| **实例属性** | **entries** | **返回解析结果的字典** | |

**bin：**

|  |  |
| --- | --- |
| **pickle (built-in)：** | |
| **PickleError** | **pickle异常基类** |
| **dump(data, file\_obj)** | **数据 -> bytes文件** |
| **dumps(data)** | **数据 -> bytes** |
| **load(file\_obj)** | **bytes文件 -> 数据** |
| **loads(byte)** | **bytes -> 数据** |

**csv：**

|  |  |  |
| --- | --- | --- |
| **csv (built-in)：** | | |
| **reader / DictReader(textIO, delimiter)** | | **行迭代器** |
| **writer(textIO) / DictWriter(textIO, fieldnames)** | | **写入器** |
| **Dict\*属性** | **fieldnames** | **字段** |
| **\*writer方法** | **writerow / writerows(list\_or\_dict)** | **写入行** |
| **DictWriter方法** | **writeheader()** | **写入字段** |

**docx：**

|  |  |  |
| --- | --- | --- |
| **文档：** | | |
| **Document(file)** | | **实例化文档** |
| **实例属性** | **paragraphs** | **段落实例列表** |
| **tables** | **表格实例列表** |
| **实例方法** | **add\_paragraph(str)** | **写入正文** |
| **add\_heading(str, level=1)** | **写入标题** |
| **add\_page\_break()** | **写入分页符** |
| **save(file)** | **保存** |

|  |  |  |
| --- | --- | --- |
| **段落：** | | |
| **实例属性** | **text** | **内容实例** |
| **runs** | **元素实例列表** |
| **style** | **样式** |
| **实例方法** | **insert\_paragraph\_before(str)** | **插入段落** |
| **insert\_paragraph\_after(str)** |

|  |  |  |
| --- | --- | --- |
| **表格：** | | |
| **实例属性** | **rows** | **行实例列表** |
| **columns** | **列实例列表** |
| **Row / Col**  **实例属性** | **cells** | **返回行/列单元格实例列表** |

**dot：**

|  |  |  |
| --- | --- | --- |
| **pydotplus：** | | |
| **graph\_from\_dot\_data(str)** | | **文本流 -> dot文件实例** |
| **实例方法** | **write\_jpg(file)** | **写入jpg文件** |
| **write\_pdf(file)** | **写入pdf文件** |

**json：**

|  |  |  |
| --- | --- | --- |
| **json (built-in)：** | | |
| **dump(data, file\_obj, indent)** | **将数据写入文件** | |
| **dumps(data, indent)** | **返回数据的文本流** | |
| **load(file\_obj)** | **返回数据** | **解析文件** |
| **loads(str)** | **解析文本流** |

**pdf：**

**from pdfminer.pdfparser import PDFParser**

**from pdfminer.pdfdocument import PDFDocument**

**from pdfminer.pdfinterp import PDFPageInterpreter, PDFResourceManager**

**from pdfminer.converter import PDFPageAggregator**

**from pdfminer.pdfpage import PDFPage, PDFTextExtractionNotAllowed**

**from pdfminer.layout import \***

**# rsrcmgr：pdf资源管理器，储存共享资源**

**rsrcmgr = PDFResourceManager(caching = False)**

**# device：pdf解析器**

**# laparams = LAParams() 创建pdf页面聚合器**

**device = PDFPageAggregator(rsrcmgr, laparams=LAParams())**

**# interpreter：pdf页面处理器**

**interpreter = PDFPageInterpreter(rsrcmgr, device)**

**# 返回"rb"模式下PDFPage迭代器**

**page = PDFPage.get\_pages(pdf.read\_bytes(), maxpages=3)[0]**

**# 加载PDFPage的LTPage实例**

**interpreter.process\_page(page)**

**# 获取LTPage实例**

**layout = device.get\_result()**

|  |  |  |
| --- | --- | --- |
| **实例layout：** | | |
| **LTPage** | **页面，解析结果迭代器** | |
| **LTTextBox** | **几何分析创建的文本框，LTTextLine迭代器** | **LTTEXT.get\_text() 返回文本内容** |
| **LTTextLine** | **文本行，LTChar、LTAnno混合迭代器** |
| **LTFigure** | **表单实例使用的区域，可以通过在页面中嵌入另一个PDF文档来显示数字或图片** | |
| **LTChar** | **属性：size/x0/y0/x1/y1/fontname** | **LTCHAR.get\_text() 返回文本内容** |
| **LTAnno** | **Unicode字符串** | |
| **LTImage** | **图像** | |
| **LTLine** | **直线，分离文本或附图** | |
| **LTRect** | **矩形，用于框架的图片或数字** | |
| **LTCurve** | **Bezier曲线** | |

|  |  |  |
| --- | --- | --- |
| **图像fitz：** | | |
| **open(file)** | | **返回pdf页面迭代器** |
| **open('pdf', ImgBytes)** | | **向pdf写入图片** |
| **Page**  **实例方法** | **get\_pixmap(matrix=fitz.Matrix(3,3))** | **返回提高像素倍数的图像** |
| **Pixmap**  **实例方法** | **save(file)** | **将图像导出到file** |

|  |  |  |
| --- | --- | --- |
| **解密pikepdf：** | | |
| **open(file)** | | **返回已解密的pdf文件** |
| **实例方法** | **save(file)** | **保存pdf文件** |

|  |  |  |
| --- | --- | --- |
| **文档pdfplumber：** | | |
| **open(file)** | | **返回打开的pdf文件实例** |
| **实例属性** | **metadata** | **基础信息** |
| **pages** | **页面列表** |

|  |  |  |
| --- | --- | --- |
| **页面pdfplumber：** | | |
| **实例属性** | **page\_number** | **页码** |
| **width** | **页宽** |
| **height** | **页高** |
| **实例方法** | **extract\_text()** | **文本** |
| **extract\_tables(table\_settings)** | **表格列表** |

**ply：**

**from plyfile import PlyData, PlyElement**

|  |  |  |
| --- | --- | --- |
| **结构体：** | | |
| **PlyData(elements)** | | **结构体 (可迭代)** |
| **实例属性** | **["element"]** | **元素** |
| **实例方法** | **read / write(file)** | **读写** |

|  |  |  |
| --- | --- | --- |
| **元素：** | | |
| **PlyElement** | | **元素 (可迭代)** |
| **实例属性** | **name** | **名称** |
| **count** | **长度** |
| **data** | **memmap数组** |
| **["attr"]** | **属性数组** |
| **类方法** | **describe(struct\_array, name)** | **构造** |

**xml (html)：**

**<?xml version="1.0"?>**

**<root>**

**<!--注释-->**

**<element1>内容</element1>**

**<element2>**

**<leaf attr="属性" />**

**<leaf attr="属性" />**

**</element2>**

**</root>**

|  |  |
| --- | --- |
| **解析lxml.etree：** | |
| **XML(text)** | **xml文本 -> 元素树** |
| **HTML(text)** | **html文本 -> 元素树** |

**import xml.etree.ElementTree # 允许修改结点**

**import lxml.etree # 不允许修改结点，支持xpath**

|  |  |  |
| --- | --- | --- |
| **元素树：** | | |
| **ElementTree(element, file)** | | **element：实例化该根节点的树** |
| **file：解析文件返回树实例** |
| **实例方法** | **getroot()** | **返回根结点 (可迭代)** |
| **write(file, encoding,**  **xml\_declaration = True)** | **写入文件** |

|  |  |  |
| --- | --- | --- |
| **树结点：** | | |
| **tostring(ELE, encoding)** | | **返回结点转化的字符串** |
| **Element(tag, attrib)** | | **实例化根结点** |
| **SubElement(ELE, tag, attrib)** | | **实例化子结点** |
| **实例属性** | **tag** | **标签** |
| **text** | **内容 (str)** |
| **attrib** | **属性 (dict)** |
| **tail** | **注释** |
| **实例方法** | **xpath(exp)** | **所有结点** |
| **find(str)** | **第一个对应结点** |
| **findall(str)** | **所有对应结点** |
| **iter(tag)** |

|  |  |  |
| --- | --- | --- |
| **文档xml.dom.minidom：** | | |
| **parseString(str)** | | **实例化字符串对应的DOC** |
| **实例方法** | **toprettyxml()** | **返回DOC的xml文本** |

**yaml：**

|  |  |
| --- | --- |
| **yaml：** | |
| **dump(data)** | **数据 -> 文本流** |
| **load(str, Loader=Loader)** | **文本流 -> 数据** |

**key1: &<addr> <value>**

**key2: \*<addr>**

**zip/rar：**

**zipfile源码：'cp437' -> 'gbk'**

|  |  |  |  |
| --- | --- | --- | --- |
| zipfile (built-in) / r**arfile** | | | |
| **zipfile.ZipFile(file, mode)** | | **创建并返回 (类似open)** | **zip文件实例** |
| **rarfile.RarFile(file, mode)** | | **rar文件实例** |
| **zipfile.is\_zipfile(file)** | | **判断是否为** | **zip压缩文件** |
| **rarfile.is\_rarfile(file)** | | **rar压缩文件** |
| **实例属性** | **infolist()** | **文件信息列表** | |
| **namelist()** | **文件名称列表** | |
| **实例方法** | **close()** | **关闭文件** | |
| **extractall(path, pwd=b'pwd')** | **输入解压密码解压到指定目录** | |
| **write(file)** | **将文件写入压缩文件列表** | |