

- Thinking: behind the theory, original from the real problem
- Action: solve problems by tools, present the results

>> 今天的学习目标

Excel、Word自动化

- 什么是办公自动化
- xlrd, xlwt使用
- 处理Excel表格信息
- Word自动化
- Email自动化
- Project A: Excel报表自动高亮&自动发邮件件
- Thinking: 哪些办公场景可以进行自动化

PDF、PPT自动化

- 提取PDF文档中的文本
- 提取PDF中的表格信息
- PDF转化HTML
- PDF的旋转，合并，拆分
- 使用Python创建PPT（slide，标题，正文，文本框，图片）
- Project B: 给一本图书制作PPT

1/2 使用Python进行Excel, Word自动化

办公自动化：

- Excel自动化
- PDF自动化
- Word自动化
- PPT自动化
- 邮件自动化

为什么需要办公自动化：

- 价值化，事情的价值是第一要素
- 无人化，人 VS Python自动化
- 智能化，人的智能 VS Python数据分析的智能



Excel自动化可以帮我们完成日常数据的记录、汇总、统计，方便我们更好的进行日常管理

To Do 统计每天楼宇进入的人员：

- 姓名
- 体温
- 进入时间

门禁考勤机的Excel统计模块，该Excel自动记录模块，未来可以和人脸识别、非接触式远距离红外测温模块进行结合

产品一 经济型测温门禁考勤一体机（无需平台，一人一档）



壁挂式测温一体机



立式测温一体机

产品优势

- 热成像测温：支持2米内检测测温，测温精度 $\pm 0.5^{\circ}\text{C}$ ；
- 高容量，比对速度快：支持>5000张人脸名单，比对速度30人/分钟；
- 工作模式可设置：“身份验证+体温检测”/“快速测温（仅测温）”；
- 支持口罩检测：可配置提醒戴口罩模式、强制戴口罩模式。

产品二 经济型测温人员通道组件（利旧通道，一人一档）



7寸人员通道
测温组件



10寸人员通道
测温组件

产品优势

- 支持已建设人员通道改造建设
- 热成像测温：支持2米内检测测温，测温精度 $\pm 0.5^{\circ}\text{C}$ ；
- 高容量，比对速度快：支持>5000张人脸名单，比对速度30人/分钟；
- 工作模式可设置：“身份验证+体温检测”/“快速测温（仅测温）”；
- 支持口罩测：可配置提醒戴口罩模式、强制戴口罩模式。

使用Pytho操作Excel:

- xlrd工具, excel reader, 读excel,
- xlwt工具, excel writer, 写excel

Excel的工作簿 (book) 由sheets组成, 获取指定sheet:

- 通过索引顺序获取

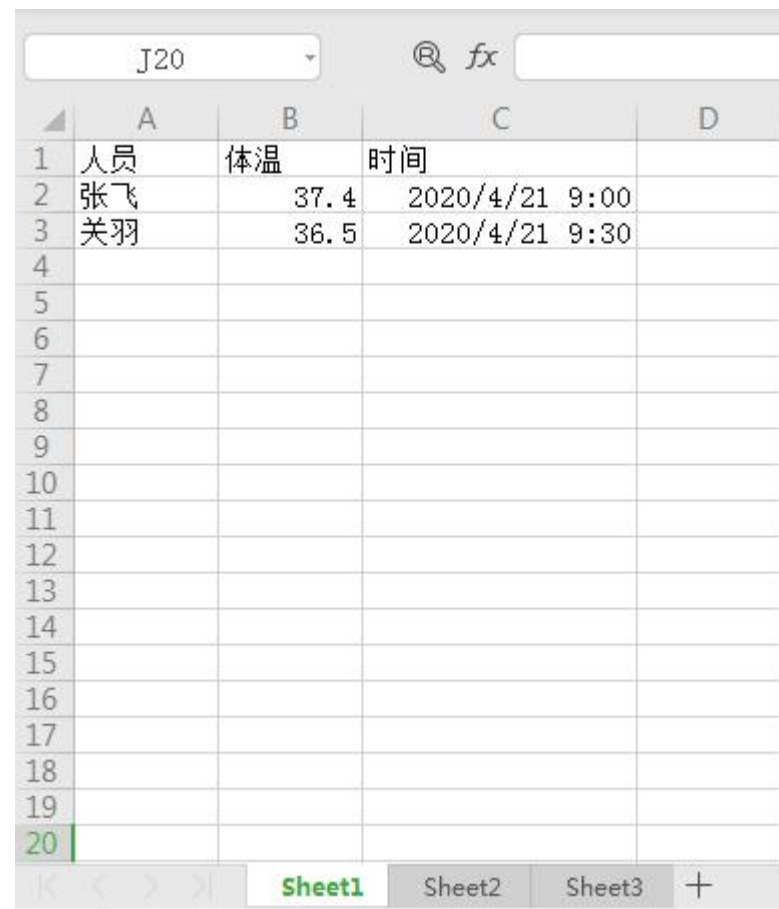
```
table = data.sheets()[0]
```

- 通过索引顺序获取

```
table = data.sheet_by_index(0)
```

- 通过名称获取

```
table = data.sheet_by_name('Sheet1')
```



	A	B	C	D
1	人员	体温	时间	
2	张飞	37.4	2020/4/21 9:00	
3	关羽	36.5	2020/4/21 9:30	
4				
5				
6				
7				
8				
9				
10				
11				
12				
13				
14				
15				
16				
17				
18				
19				
20				

#返回book中所有工作表的名字

```
names = data.sheet_names()
```

```
print(names)
```

""" 工作表中行/列的操作 """

#获取该sheet中的有效行数

```
nrows = table.nrows
```

```
print(nrows)
```

```
row_index, col_index = 0, 0
```

获取某行信息

```
print(table.row(row_index))
```

```
print(table.row_slice(row_index))
```

获取某列信息

```
print(table.col(col_index))
```

```
print(table.col_slice(col_index))
```

输出结果:

```
['Sheet1', 'Sheet2', 'Sheet3']
```

```
3
```

```
[text:'人员', text:'体温', text:'时间']
```

```
[text:'人员', text:'体温', text:'时间']
```

```
[text:'人员', text:'张飞', text:'关羽']
```

```
[text:'人员', text:'张飞', text:'关羽']
```

#返回某行的 数据类型列表

```
print(table.row_types(row_index, start_colx=0, end_colx=None))
```

```
print(table.row_types(1, start_colx=0, end_colx=None))
```

输出结果:

['人员', '体温', '时间']

3

返回某行的数值

```
print(table.row_values(row_index, start_colx=0, end_colx=None))
```

返回某行的有效单元格长度

```
print(table.row_len(row_index))
```



```
""" 单元格操作 """
```

```
#返回单元格对象
```

```
print(table.cell(row_index, col_index))
```

```
#返回单元格中的数据类型
```

```
print(table.cell_type(row_index, col_index))
```

```
#返回单元格中的数据
```

```
print(table.cell_value(row_index, col_index))
```

输出结果:

text:'人员'

1

人员

- 使用xlwt对Excel进行写入

创建一个workbook 设置编码

```
workbook = xlwt.Workbook(encoding = 'utf-8')
```

创建一个worksheet

```
worksheet = workbook.add_sheet('国贸写字楼')
```

写入excel, 参数对应 行, 列, 值

```
worksheet.write(0, 0, label = '人员')
```

保存文件

```
workbook.save('./楼宇安防2.xls')
```

""" 设置字体样式 """

初始化样式

```
style = xlwt.XFStyle()
```

为样式创建字体

```
font = xlwt.Font()
```

```
font.name = 'Times New Roman'
```

```
font.bold = True # 加粗
```

```
font.underline = True # 下划线
```

```
font.italic = True # 斜体字
```

设定样式

```
style.font = font
```

```
worksheet.write(0, 1, '体温', style) # 带样式的写入
```

```
workbook.save('楼宇安防2.xls')
```

""" 设置单元格宽度 """

```
#worksheet = workbook.add_sheet('世贸天阶')
```

```
worksheet.write(0, 2, '时间')
```

设置单元格宽度

```
worksheet.col(0).width = 3333
```

```
workbook.save('./楼宇安防2.xls')
```

""" 添加日期到单元格 """

```
import datetime
```

获取样式

```
style = xlwt.XFStyle()
```

Other options: D-MMM-YY, D-MMM, MMM-YY, h:mm, h:mm:ss, h:mm, h:mm:ss, M/D/YY h:mm, mm:ss, [h]:mm:ss, mm:ss.0

```
style.num_format_str = 'M/D/YY'
```

```
worksheet.write(1, 2, datetime.datetime.now(), style)
```

```
workbook.save('楼宇安防2.xls')
```

""" 向单元格添加一个公式 """

```
worksheet.write(1, 1, 37.4)
```

```
worksheet.write(2, 1, 36.5)
```

求两个单元格的平均值

```
worksheet.write(3, 1, xlwt.Formula('AVERAGE(B2,B3)'))
```

```
workbook.save('楼宇安防2.xls')
```

""" 向单元格添加一个链接 """

```
worksheet.write(1, 3,  
xlwt.Formula('HYPERLINK("http://www.baidu.com";"baidu")'))
```

```
workbook.save('楼宇安防2.xls')
```

```
""" 合并列和行 """
```

```
# write_merge(行开始, 行结束, 列开始, 列结束, '数据内容')
```

```
# 创建字体
```

```
font = xlwt.Font()
```

```
# 加粗
```

```
font.bold = True
```

```
style = xlwt.XFStyle()
```

```
style.font = font
```

```
worksheet.write_merge(4, 5, 0, 2, '备注: 需要检查是否佩戴口罩',  
style)
```

```
workbook.save('楼宇安防2.xls')
```

```
""" 给单元格添加边框和背景色 """
```

```
# 边框样式
```

```
borders = xlwt.Borders()
```

```
borders.left = xlwt.Borders.DASHED
```

```
borders.right = xlwt.Borders.DASHED
```

```
borders.top = xlwt.Borders.DASHED
```

```
borders.bottom = xlwt.Borders.DASHED
```

```
borders.left_colour = 0x40
```

```
borders.right_colour = 0x40
```

```
borders.top_colour = 0x40
```

```
borders.bottom_colour = 0x40
```

```
# 创建样式
```

```
style = xlwt.XFStyle()
```

```
style.borders = borders
```

```
worksheet.write(1, 0, '张飞', style)
```

```
worksheet.write(2, 0, '关羽', style)
```

```
workbook.save('楼宇安防2.xls')
```

使用Python操作Word:

- `pip install python-docx`

`from docx import Document`

- 文档创建与打开

空白文档, `document = Document()`

打开原有文档, `document = Document('./test.docx')`

- 添加段落

`paragraph = document.add_paragraph('基于Python的办公自动化处理')`

`paragraph = document.add_paragraph('段落1:Word自动化')`

- 添加章节, 并添加段落

`document.add_section()` #添加新章节

`paragraph = document.add_paragraph('章节2-1')`

`document.add_section()` #添加新章节

`paragraph = document.add_paragraph('章节3-1')`

`sections = document.sections` #返回所有章节引用的对象

`print('章节数=', len(sections))` #返回章节总数

Thinking: 章节数 = ?

- 操作指定section

```
section = sections[0] #返回指定章节的对象
```

```
section = document.sections[-1] # 返回文档最后一个章节
```

```
# 得到section的信息，1英寸=914400像素
```

```
new_height= section.page_height
```

```
new_width=section.page_width
```

```
print('Section页面高={}, Section页面宽={}'.format(new_height,  
new_width))
```

```
print('左边距={}, 右边距={}, 上边距={}, 下边距  
= {}'.format(section.left_margin, section.right_margin,  
section.top_margin, section.bottom_margin))
```

```
print('页眉距离={}, 页脚距离={}'.format(section.header_distance,  
section.footer_distance))
```

输出结果:

Section页面高=10058400, Section页面宽=7772400

左边距=1143000, 右边距=1143000, 上边距=914400, 下边距
=914400

页眉距离=457200, 页脚距离=457200

- 给word添加表格和图片

添加标题

```
document.add_heading('标题：楼宇进出人员名单', level=0)
```

添加表格

```
table = document.add_table(rows=3, cols=3)
```

返回行对象

```
row = table.rows[0]
```

给行对象的第n个单元格赋值

```
row.cells[0].text = '姓名'
```

```
row.cells[1].text = '温度'
```

```
row.cells[2].text = '时间'
```

#返回表格的单元格对象，并赋值

```
cell = table.cell(1, 0)
```

```
cell.text = '张飞'
```

```
cell = table.cell(2, 0)
```

```
cell.text = '关羽'
```

添加图片--添加的图像以原始大小显示

```
document.add_picture('pic1.jpg')
```


Email自动化

- Email自动化的工具:

Python有两个内置库: `smtplib`和`email`, 能够实现邮件功能

`smtplib`工具负责发送邮件

`email`工具负责构造邮件格式和内容

负责构造文本

```
from email.mime.text import MIMEText
```

负责构造图片

```
from email.mime.image import MIMEImage
```

负责将多个对象集合起来

```
from email.mime.multipart import MIMEMultipart
```

```
from email.header import Header
```



Email自动化

- 获取QQ邮箱授权码

`https://service.mail.qq.com/cgi-bin/help?subtype=1&&id=28&&no=1001256`

- 设置邮箱域名、发件人邮箱、邮箱授权码、收件人邮箱

发信服务器

```
smtp_server = 'smtp.qq.com'
```

发信方的信息：发信邮箱，QQ邮箱授权码

```
from_addr = 'XXXX@qq.com' #你的QQ邮箱
```

```
password = 'XXXX' # 你拿到的授权码
```

收信方邮箱

```
to_addr = 'chenyang@wucai.com'
```

验证密保

短信验证

请先用密保手机 158*****86 发短信，然后点“我已发送”按钮

发短信：配置邮件客户端

到号码：1069 0700 69

短信费用

短信用不了？

验不了,试试其他

我已发送

生成授权码

在第三方客户端登录时，密码框请输入以下授权码：

ujjo gnhk

取消

QQ

下一步

名称	Mail team
电子邮件	mailteam@qq.com
密码
描述	Mail Team

提示：你可拥有多个授权码，所以无需记住该授权码，也不要告诉其他人。[了解更多](#)

确定

- MIMEText构建邮箱正文内容

```
msg = MIMEText('send by python class demo','plain','utf-8')
```

第一个参数为内容，第二个参数为格式(plain 为纯文本)，第三个参数为编码

- 设置邮件头信息

```
msg['From'] = Header(from_addr)
```

```
msg['To'] = Header(to_addr)
```

```
msg['Subject'] = Header('python automation test')
```

开启发信服务，这里使用的是加密传输

```
server = smtplib.SMTP_SSL(host='smtp.qq.com')
```

连接服务器

```
server.connect(host='smtp.qq.com', port=465)
```

登录发信邮箱

```
server.login(from_addr, password)
```

发送邮件

```
server.sendmail(from_addr, to_addr, msg.as_string())
```

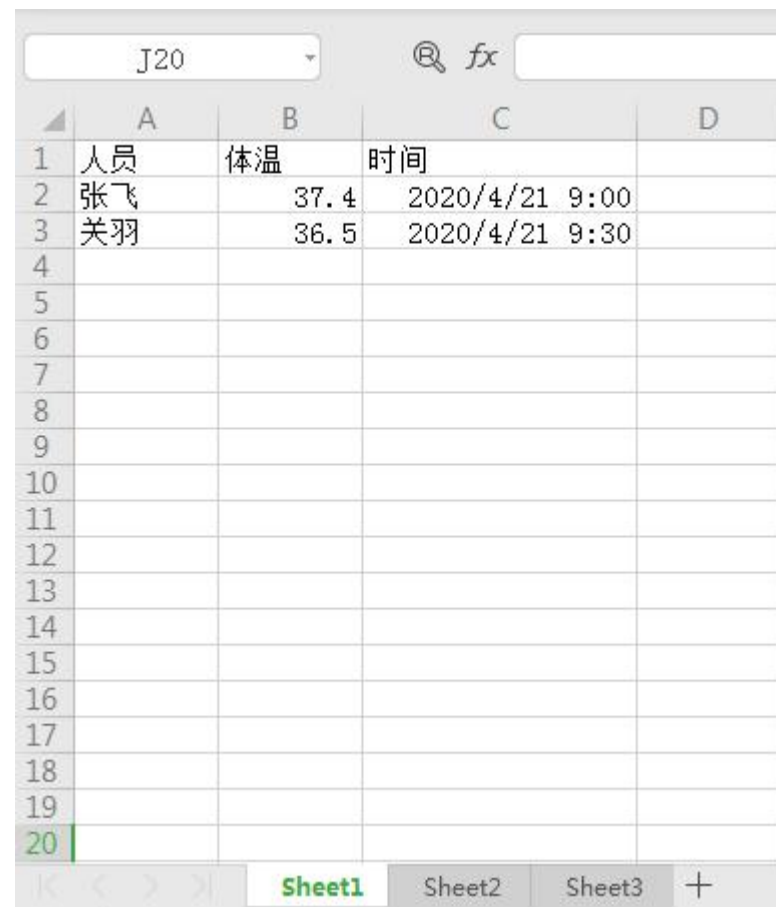
关闭服务器

```
server.quit()
```

Project A: Excel报表自动高亮&自动发邮件

针对每天进出楼宇的报表Excel进行自动处理:

- Step1, 高亮体温 > 37.4度的用户
- Step2, 统计每天进出人员, 平均体温
- Step3, 每日汇总数据自动发邮件



The image shows a screenshot of an Excel spreadsheet. The active cell is J20. The spreadsheet has four columns: A (人员), B (体温), C (时间), and D. The data is as follows:

	A	B	C	D
1	人员	体温	时间	
2	张飞	37.4	2020/4/21 9:00	
3	关羽	36.5	2020/4/21 9:30	
4				
5				
6				
7				
8				
9				
10				
11				
12				
13				
14				
15				
16				
17				
18				
19				
20				

Project A: Excel报表自动高亮&自动发邮件

- 数据加载的同时，保留原excel格式

```
data = xlrd.open_workbook('./楼宇安防.xls', formatting_info=True)
```

- Copy XLS

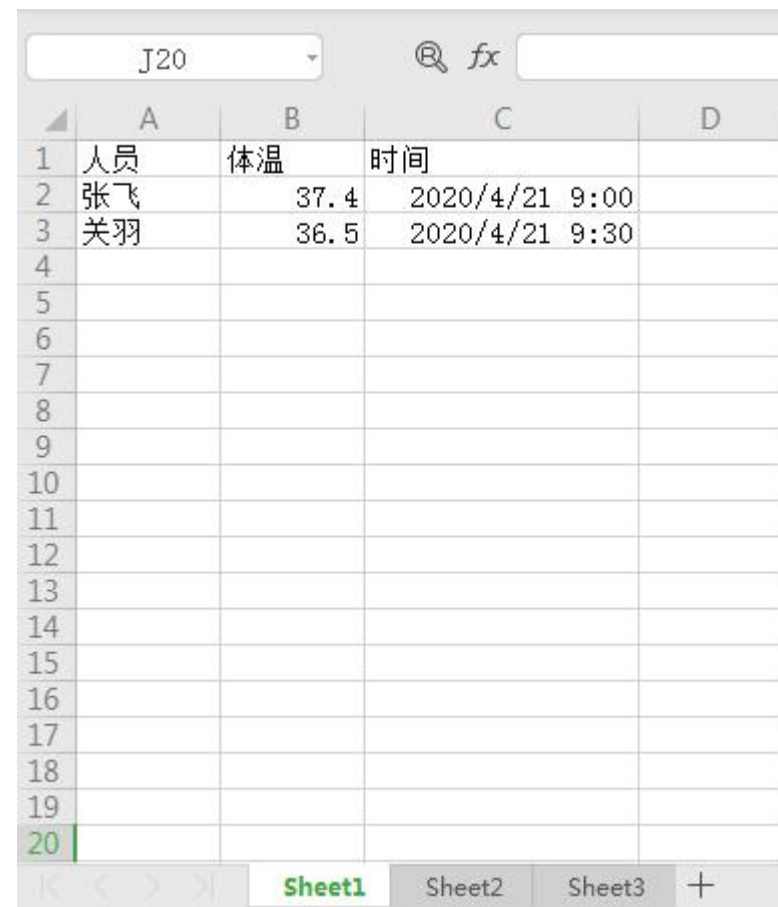
```
from xlutils.copy import copy
```

```
wb = copy(data)
```

- 自定义格式（时间格式）

在num_format_str数据性中设置EXCEL自定义格式

```
style.num_format_str = 'm/dd/yy h:mm:ss'
```



	A	B	C	D
1	人员	体温	时间	
2	张飞	37.4	2020/4/21 9:00	
3	关羽	36.5	2020/4/21 9:30	
4				
5				
6				
7				
8				
9				
10				
11				
12				
13				
14				
15				
16				
17				
18				
19				
20				

Project A: Excel报表自动高亮&自动发邮件

- 创建高亮样式:

```
style = xlwt.XFStyle()
```

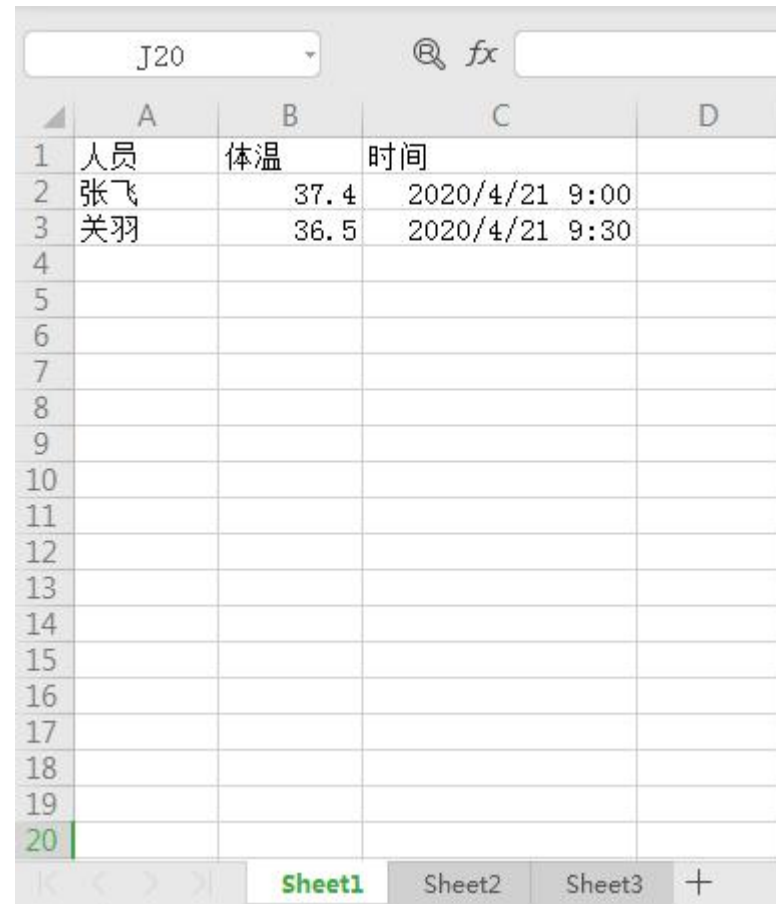
```
pattern = xlwt.Pattern()
```

```
pattern.pattern = xlwt.Pattern.SOLID_PATTERN
```

设置单元格背景色为黄色

```
pattern.pattern_fore_colour = xlwt.Style.colour_map['yellow']
```

```
style.pattern = pattern
```



	A	B	C	D
1	人员	体温	时间	
2	张飞	37.4	2020/4/21 9:00	
3	关羽	36.5	2020/4/21 9:30	
4				
5				
6				
7				
8				
9				
10				
11				
12				
13				
14				
15				
16				
17				
18				
19				
20				

Project A: Excel报表自动高亮&自动发邮件

- 邮件发送中创建一个带附件的实例:

```
from email.mime.multipart import MIMEMultipart
```

```
msg = MIMEMultipart()
```

```
msg.attach(MIMEText('这里写你的邮件正文', 'plain', 'utf-8'))
```

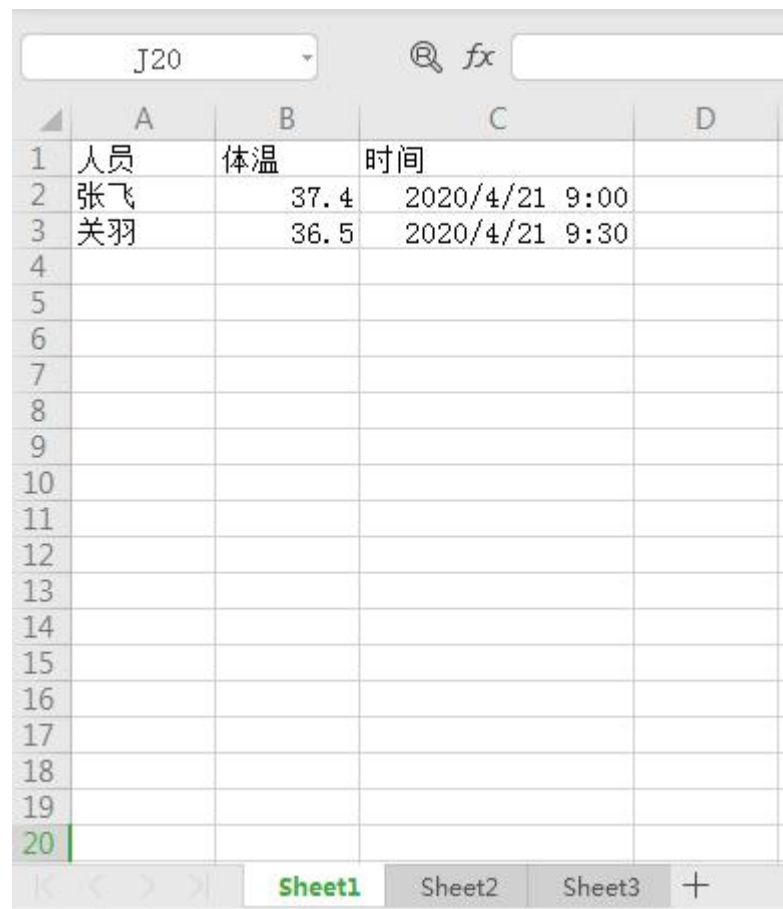
- 读取本地文件的内容

```
att2 = MIMEText(open('本地文件.xls', 'rb').read(), 'based64', 'utf-8')
```

```
att2["Content-Type"]='application/octet-stream'
```

```
att2["Content-Disposition"] = 'attachment;
```

```
filename="daily_report.xls"
```



	A	B	C	D
1	人员	体温	时间	
2	张飞	37.4	2020/4/21 9:00	
3	关羽	36.5	2020/4/21 9:30	
4				
5				
6				
7				
8				
9				
10				
11				
12				
13				
14				
15				
16				
17				
18				
19				
20				

办公自动化:

- 针对Excel, Word的重度用户
- 重复工作多 => 方便写成规则

Thinking: 如果是paper work, 能用办公自动化么?

Thinking: 你的老板让你做个智能考勤机（带红外体温），每天汇报考勤和人员温度给他，你的技术方案是怎样的



Chinese startup Rokid pitches COVID-19 detection glasses in U.S.

Jake Bright, Rita Liao

/ 3:05 pm CST • April 16, 2020



Read our latest COVID-19 coverage, and find out more about our events strategy.

$$0.0576 + 0.0084 + 0.1836$$

2/2 使用Python进行PDF, PPT自动化

使用Python操作PDF:

- pdfminer, 解析PDF是一件耗时和耗内存的工作, PDFMiner库使用了lazy parsing的策略, 也就是只在需要的时候才去解析 (减少时间和内存的使用)

文档: <http://euske.github.io/pdfminer/programming.html>

- pdf2htmlEX, 将pdf格式转换成html格式, 然后再提取信息
- pdfplumber, 用于提取完整表格
- pypdf2, 切割合并PDF

PDF提取

- pdfminer 库

Windows环境, pip install pdfminer3k

Linux环境, pip install pdfminer

pdfminer3k 是 pdfminer 的 python3 版本, 主要用于读取 pdf 中的文本

- 解析PDF所需的类:

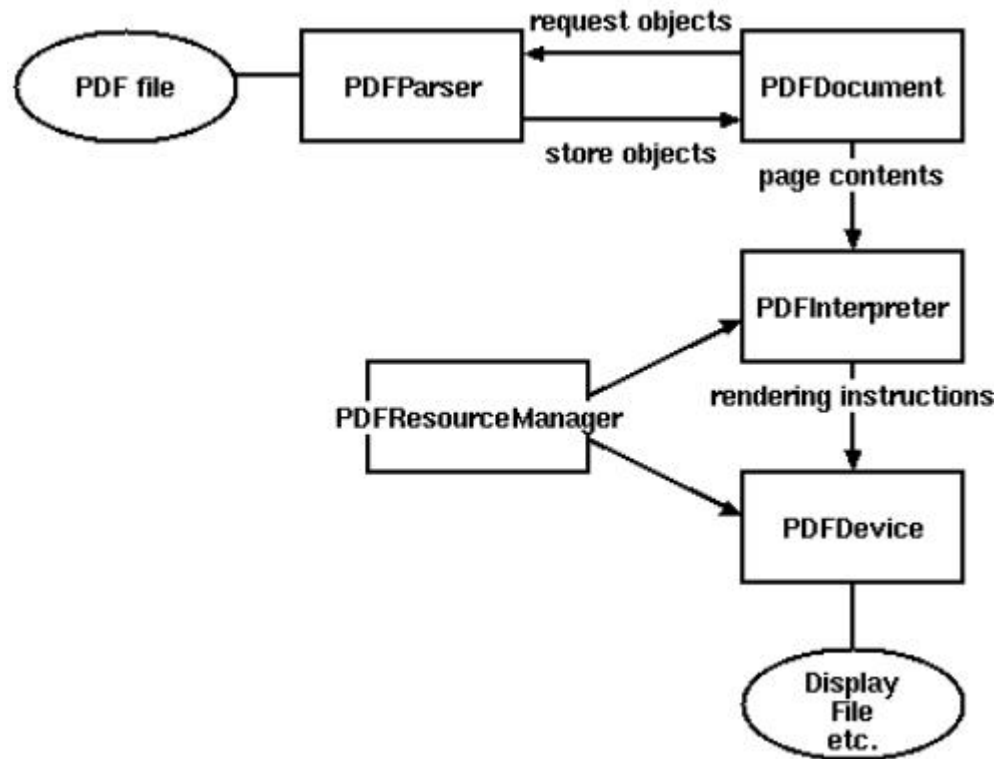
PDFParser, 从文件中提取数据

PDFDocument, 保存获取的数据, 和PDFParser是相互关联的

PDFPageInterpreter, 处理页面内容

PDFDevice, 将其转换为你需要的格式

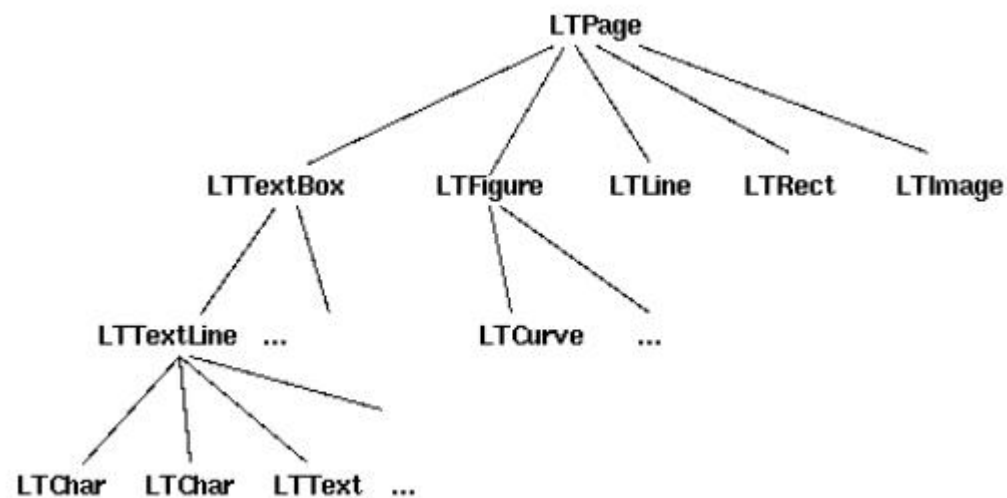
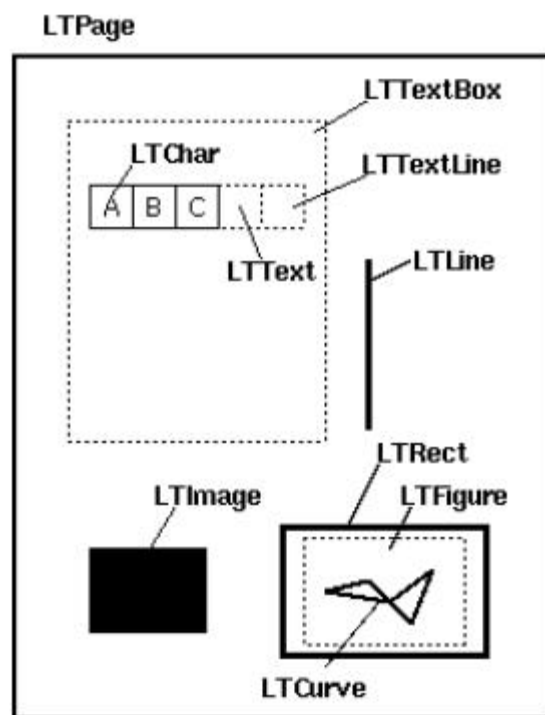
PDFResourceManager, 用于存储共享资源, 如字体或图像



pdfminer使用:

布局分析返回PDF文档中的每个页面LTPage对象

LTPage对象和页内包含的子对象，形成一个树结构



PDF提取（表格提取）

表格提取工具（pdfplumber）：

- pdfminer 对于表格的处理非常的不友好，能提取出文字，但是没有格式
- pdfplumber，能提取完整表格，提取结果较为规范
- To DO：如何将“2018高新认证名单.pdf”转化为表格形式（csv文件）

附件：

北京市2018年第一批拟认定高新技术企业名单

1	北京路桥通国际工程咨询有限公司
2	北京勤思勤智科技有限公司
3	北京多联科技有限公司
4	北京东展科博科技股份有限公司
5	中知知识互联（北京）科技有限公司
6	北京弘图科技有限公司
7	北京星辰万有科技有限公司
8	北京欧瑞科斯科技有限公司
9	北京国信博飞科技发展有限公司
10	北京市进联国玉制动泵有限责任公司
11	北京合立星源光电科技有限公司
12	北京迪科创新科技有限公司
13	北京汇达城数科技发展有限公司
14	北京中益信联科技有限公司
15	北京深育技术服务有限公司
16	北京羽胜科技有限公司

pdfplumber使用:

- .open() 打开指定的pdf文件
- .extract_text() 用来提页面中的文本，将页面的所有字符对象整理为字符串
- .extract_words() 返回所有的单词及其相关信息
- .extract_tables() 提取页面的表格
- .to_image() 用于可视化调试时，返回PageImage类的一个实例

附件:

北京市2018年第一批拟认定高新技术企业名单

1	北京路桥通国际工程咨询有限公司
2	北京勤思勤智科技有限公司
3	北京多联科技有限公司
4	北京东展科博科技股份有限公司
5	中知知识互联（北京）科技有限公司
6	北京弘图科技有限公司
7	北京星辰万有科技有限公司
8	北京欧瑞科斯科技有限公司
9	北京国信博飞科技发展有限公司
10	北京市进联国玉制动泵有限责任公司
11	北京合立星源光电科技有限公司
12	北京迪科创新科技有限公司
13	北京汇达城数科技发展有限公司
14	北京中益信联科技有限公司
15	北京深育技术服务有限公司
16	北京羽胜科技有限公司

pdf2htmlEX作用:

- 在HTML文件中精确显示原生文本

保持PDF中的字体, 颜色, 位置。同时可以选择, 复制, 搜索

- 生成的文件大小与解压缩后的PDF文件相当, 如果放到支持HTTP压缩的服务器上, 网络消耗和PDF相当

- HTML文件输出

使用场景 (PDF在线发布):

- 在个人主页发布个人简历, 在其中嵌入流量统计代码
- 内容发布商 (报纸, 杂志, 电子书) 提供在线阅读服务, 可以加入权限管理, 广告, 个性化内容
- 网盘服务, PDF在线预览 (MediaFire使用的pdf2htmlEX)

Course Description

As the course title suggests, this class is meant to acquaint you with the literary and rhetorical tradition of the essay, a genre which has been described by one scholar as "the meeting ground between art and philosophy," and by another as "the place where the self finds a pattern in the world, and the world finds a pattern in the self". Though the essay is part of a tradition of prose which stretches back to antiquity, it is also a thoroughly modern and popular form of writing, found in print media and on the web. This class will emphasize a specific range of the essay's functions:

- The essay as commentary on the writing of others and on textual/numerical data.
- The essay as the discovery of meaning in personal experience through the narration of experience.
- The essay as a vehicle conveying specialized knowledge to a general audience.

By the end of the semester, you will be familiar with each of these modes in the class reading and in your own writing.

Course Requirements

- Assigned reading (see schedule below)
- 5 essays (5-7 pages each), rough and final drafts
- 2 oral reports (5-10 minutes long)
- Participation in class discussion
- In-class peer review (see below)
- Peer-review-related photocopying
- In-class and out-of-class writing exercises (1-2 pp)
- Maintenance of a portfolio (see below)
- Regular attendance (see below)

PDF文件

pdf2htmlEX使用:

- <https://github.com/coolwanglu/pdf2htmlEX/wiki>

<http://soft.rubypdf.com/software/pdf2htmlex-windows-version>

用法: pdf2htmlEX [options] <input.pdf> [<output.html>]

- | | |
|-----------------------|----------------------------|
| -f,--first-page <int> | 需要转换的起始页 (默认: 1) |
| -l,--last-page <int> | 需要转换的最后一页 (默认: 2147483647) |
| --zoom <fp> | 缩放比例 |
| --fit-width <fp> | 适合宽度 <fp> 像素 |
| --fit-height <fp> | 适合高度 <fp> 像素 |
| --use-cropbox <int> | 使用剪切框 (default: 1) |

Course Description

As the course title suggests, this class is meant to acquaint you with the literary and rhetorical tradition of the essay, a genre which has been described by one scholar as "the meeting ground between art and philosophy," and by another as "the place where the self finds a pattern in the world, and the world finds a pattern in the self". Though the essay is part of a tradition of prose which stretches back to antiquity, it is also a thoroughly modern and popular form of writing, found in print media and on the web. This class will emphasize a specific range of the essay's functions:

- The essay as commentary on the writing of others and on textual/numerical data.
 - The essay as the discovery of meaning in personal experience through the narration of experience.
 - The essay as a vehicle conveying specialized knowledge to a general audience.
- By the end of the semester, you will be familiar with each of these modes in the class reading and in your own writing.

Course Requirements

- Assigned reading (see schedule below)
- 5 essays (5-7 pages each), rough *and*

final drafts

- 2 oral reports (5-10 minutes long)
- Participation in class discussion
- In-class peer review (see below)
- Peer-review-related photocopying
- In-class and out-of-class writing exercises (1-2 pp)
- Maintenance of a portfolio (see below)
- Regular attendance (see below)

NOTE: It is not possible to pass the course if you have failed one of the essays, if you have failed to participate in peer review, or if you have more than 5 absences. Failure to participate in class discussion will result in a deduction of one-half or one grade point from the final grade, depending on the severity of the problem.

转换后的HTML

pdfpy2使用:

- 从PDF中提取文字
- 旋转pdf页
- 合并pdf
- 分割pdf
- 文档 <https://pythonhosted.org/PyPDF2/>

```
# 对PDF文档进行旋转

def PDFrotate(origFileName,newFileName,rotation):

    pdfFile = open(origFileName,'rb')

    pdfReader = PyPDF2.PdfFileReader(pdfFile)

    pdfWriter = PyPDF2.PdfFileWriter()

    # 对每一页进行旋转

    for page in range(pdfReader.numPages):

        pageObj = pdfReader.getPage(page)

        pageObj.rotateClockwise(rotation)

        pdfWriter.addPage(pageObj)

    # 保存文件

    newFile = open(newFileName,'wb')

    pdfWriter.write(newFile)

    pdfFile.close()

    newFile.close()

PDFrotate('course.pdf', 'course_rotate.pdf', 270)
```

Course Description

As the course title suggests, this class is meant to acquaint you with the literary and rhetorical tradition of the essay, a genre which has been described by one scholar as "the meeting ground between art and philosophy," and by another as "the place where the self finds a pattern in the world, and the world finds a pattern in the self". Though the essay is part of a tradition of prose which stretches back to antiquity, it is also a thoroughly modern and popular form of writing, found in print media and on the web. This class will emphasize a specific range of the essay's functions:

- The essay as commentary on the writing of others and on textual/numerical data.
- The essay as the discovery of meaning in personal experience through the narration of experience.
- The essay as a vehicle conveying specialized knowledge to a general audience.

By the end of the semester, you will be familiar with each of these modes in the class reading and in your own writing.

Course Requirements

- Assigned reading (see schedule below)
- 5 essays (5-7 pages each), rough and final drafts
- 2 oral reports (5-10 minutes long)
- Participation in class discussion
- In-class peer review (see below)
- Peer-review-related photocopying
- In-class and out-of-class writing exercises (1-2 pp)
- Maintenance of a portfolio (see below)
- Regular attendance (see below)

NOTE: It is not possible to pass the course if you have failed one of the essays, if you have failed to participate in peer review, or if you have more than 5 absences. Failure to participate in class discussion will result in a deduction of one-half or one grade point from the final grade, depending on the severity of the problem.

```
def merge_pdfs(paths, output):  
    pdf_writer = PdfFileWriter()  
    for path in paths:  
        pdf_reader = PdfFileReader(path)  
        for page in range(pdf_reader.getNumPages()):  
            # 将每页添加到writer对象  
            pdf_writer.addPage(pdf_reader.getPage(page))  
    # 写入合并的pdf  
    with open(output, 'wb') as out:  
        pdf_writer.write(out)  
paths = ['course.pdf', '2018高新认证名单.pdf']  
merge_pdfs(paths, output='merged.pdf')
```

```
def split(filename, name_of_split):  
    pdf = PdfFileReader(filename)  
    # 拆分，即读取PDF中的每页，单独进行保存  
    for page in range(pdf.getNumPages()):  
        pdf_writer = PdfFileWriter()  
        pdf_writer.addPage(pdf.getPage(page))  
  
        output = f'{name_of_split}{page}.pdf'  
        with open(output, 'wb') as output_pdf:  
            pdf_writer.write(output_pdf)  
  
filename = 'course.pdf'  
split(filename, 'course_page')
```

- pptx工具:

```
pip install python-pptx
```

```
from pptx import Presentation
```

- 创建pptx文档类并插入一页幻灯片

```
# 实例化Presentation
```

```
ppt = Presentation()
```

```
# 使用ppt自带的模板, ppt自带了1-48中模板
```

```
title_slide_layout = ppt.slide_layouts[0]
```

```
# 新建一页幻灯片
```

```
slide = ppt.slides.add_slide(title_slide_layout)
```

- 设置slide的标题和正文内容

```
# 设置标题
```

```
title = slide.shapes.title
```

```
# 使用placeholders索引获取一页幻灯片中的元素
```

```
subtitle = slide.placeholders[1]
```

```
# 设置text
```

```
title.text = "Hello, World!"
```

```
subtitle.text = "PPT自动化 from python-pptx!"
```

```
# 保存文件
```

```
ppt.save('demo.pptx')
```

- 给slide添加新文本框

```
from pptx.util import Inches
```

```
# 预设位置及大小
```

```
left, top = Inches(0.5), Inches(1.6)
```

```
width, height = Inches(4), Inches(1)
```

```
# left, top为相对位置, width, height为文本框大小
```

```
textbox = slide.shapes.add_textbox(left, top, width, height)
```

```
# 文本框中文字
```

```
textbox.text = 'this is a new textbox'
```

```
# 在新文本框中添加段落
```

```
new_para = textbox.text_frame.add_paragraph()
```

```
# 段落文字
```

```
new_para.text = 'this is second paragraph in textbox'
```

- 给slide添加图片

文件路径

```
img_path = 'pic1.jpg'
```

预设位置及大小

```
left, top, width, height = Inches(0.5), Inches(2.5), Inches(8), Inches(5)
```

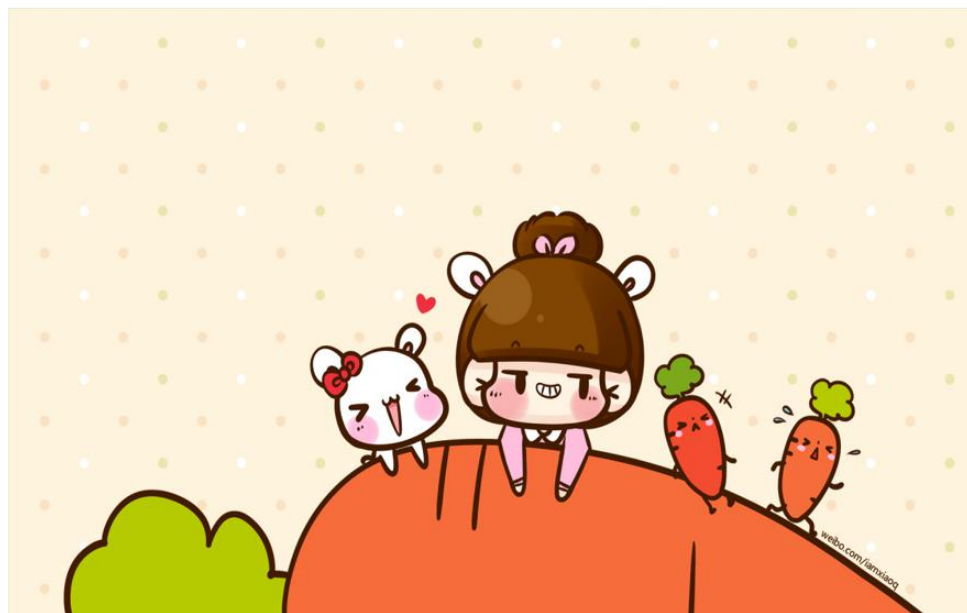
在指定位置按预设值添加图片

```
pic = slide.shapes.add_picture(img_path, left, top, width, height)
```

```
ppt.save('demo.pptx')
```

单击此处添加标题

this is a new textbox
this is second paragraph in textbox



To Do: 使用Python编写一页PPT

Step1, 创建PPT, 添加slide (使用ppt自带版本1)

Step2, 标题为 Office Automation by Python

Step3, 正文内容

Excel Operation by Python

- PPT Operation by Python
 - Use pptx tools to control PPT files
 - add_paragraph() function usage

Step4, 保存文件 pptx_usage.pptx

Office Automation by Python

- Excel Operation by Python
- PPT Operation by Python
 - Use pptx tools to control PPT files
 - add_paragraph() function usage

Tips:

获取文本框

```
body = slide.placeholders[1]
```

```
tf = body.text_frame
```

Office Automation by Python

- Excel Operation by Python
- PPT Operation by Python
 - Use pptx tools to control PPT files
 - add_paragraph() function usage

Project B: 给一本图书制作PPT

To Do: 给一本图书制作PPT

Step1, 分析PDF => 文本和图片

使用pdfminer, 解析LTTextBox中的文本 => text目录

解析LTFigure中的LTImage => images目录

Step2, 每页PDF转成PPT格式

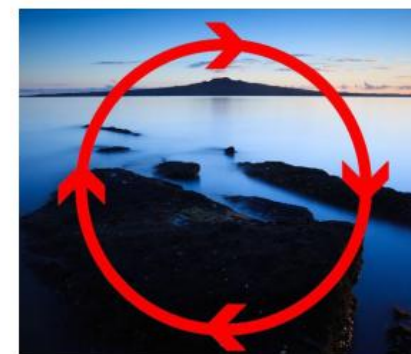
读取text目录中的文本, images中的图片, 自动生成
PPT

1、从方画幅学到的6个构图技巧

使用方画幅格式构图与其他的构图很不一样。方画幅构图很有魅力！即使正式的拍摄中不打算使用这种格式，短期地尝试使用方画幅构图仍然会为摄影师带来好处，因为这可能是一种全新的观点，帮助你以不同角度观察世界。我们来看看里面隐藏着怎么样的构图技巧！

注意平衡

正方形是个完美的平衡形状，上下左右中心各种对称，因此方画幅构图在引导观众视线方面，不似横幅从左至右、中间至两边，也不似纵幅从上至下。方画幅构图并没有过于引导观众视线的走向，于是决定权就全在画面内容安排上，框架的形状再不是一个主要因素。下图我用3:2的画幅拍摄了一张风景照，水平走向的线条引导视线沿箭头方向滑动。而另一张方画幅构图的风光照中，画面内容充实，观众视线会四处游走，全面观察，没有固定的方向。



Project B: 给一本图书制作PPT

- 保存图片

```
def save_image(lt_image):
```

```
    result = None
```

```
    if lt_image.stream:
```

```
        file_stream = lt_image.stream.get_rawdata()
```

```
        if file_stream:
```

```
            file_ext = determine_image_type(file_stream[0:4])
```

```
            if file_ext:
```

```
                if write_file(images_folder, file_name, file_stream, flags='wb'):
```

```
                    result = file_name
```

```
    return result
```

- 写文件

```
def write_file (folder, filename, filedata, flags='w'):
```

```
    result = False
```

```
    if os.path.isdir(folder):
```

```
        try:
```

```
            file_obj = open(os.path.join(folder, filename), flags)
```

```
            file_obj.write(filedata)
```

```
            file_obj.close()
```

```
            result = True
```

```
        except IOError:
```

```
            pass
```

```
    return result
```

Project B: 给一本图书制作PPT

- 判断图像的类型

```
def determine_image_type (stream_first_4_bytes):  
    file_type = None  
    bytes_as_hex = b2a_hex(stream_first_4_bytes).decode(encoding='utf-8')  
    if bytes_as_hex.startswith('ffd8'):  
        file_type = '.jpeg'  
    elif bytes_as_hex == '89504e47':  
        file_type = '.png'  
    elif bytes_as_hex == '47494638':  
        file_type = '.gif'  
    elif bytes_as_hex.startswith('424d'):  
        file_type = '.bmp'  
    return file_type
```

Python自动化办公：

- repeated working，越是重复的事情，越有价值
比如日报，周报，提醒，按照某个规则整理文档
(excel, word, pdf, ppt)

通过自动化工具，可以进行自动化运营

- 自动发微博
- 自动加关注，私信，评论

自动化的价值：

- 通过规律持续性创造价值，数据采集，自动化运营

技术点：

- 流程设计，规定每一步做什么
- XPath，通过元素的XPath路径获取元素信息
- puppet工具，微信、微博Robot

自动化运营的步骤：

- 多个手机号

尽管早期注册只需要邮箱就可以，但现在账号注册都需要绑定手机号，所以手机号是必备的

- 多个IP

很多社交网站都会有反垃圾的措施。共用同一个IP，一定会被封号

- 模拟操作

因为需求是个性化的，所以在这一步，可以封装出一些基本的操作，比如关注、发布动态、转发、阅读文章等



自动化运营

多个手机号：

- 虚拟手机号：被歧视的号码段
- 阿里小号：一个看似可行的解决方案
- 国外号码，贵但价值明显

自动切换IP：

- IP代理。
- 飞行模式。
- 小区宽带。



- Thinking1: Python自动化的价值是什么，你的日常工作有需要自动化的部分么
- Thinking2: 你的老板让你做个智能考勤机（带红外体温），每天汇报考勤和人员温度给他，你的技术方案是怎样的



Chinese startup Rokid pitches COVID-19 detection glasses in U.S.



Jake Bright, Rita Liao

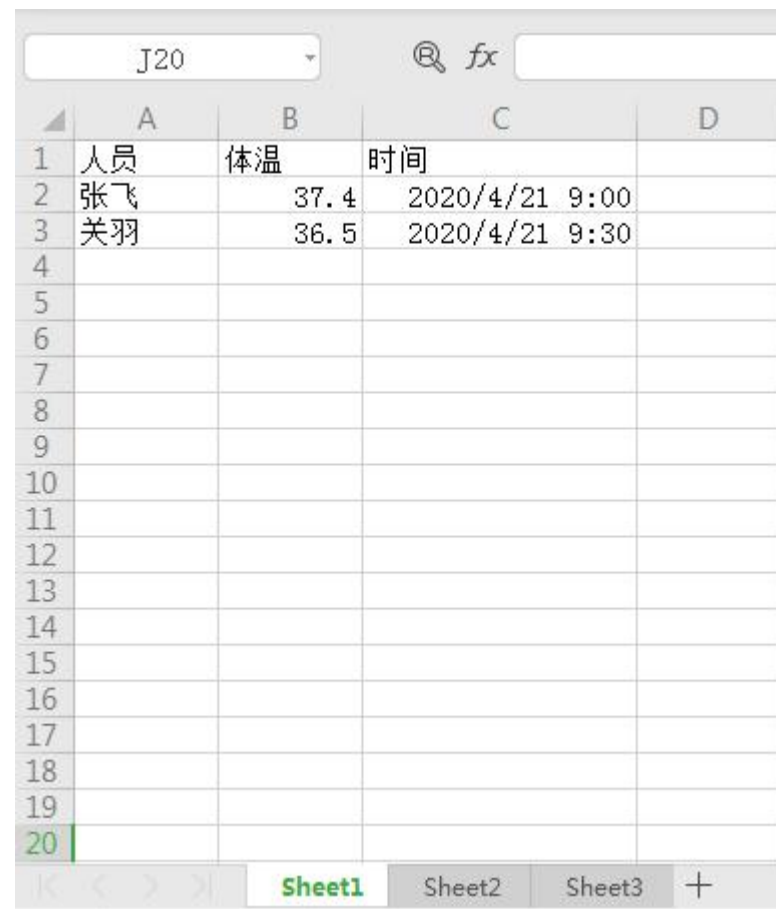
/ 3:05 pm CST • April 16, 2020



Read our latest COVID-19 coverage, and find out more about our events strategy.

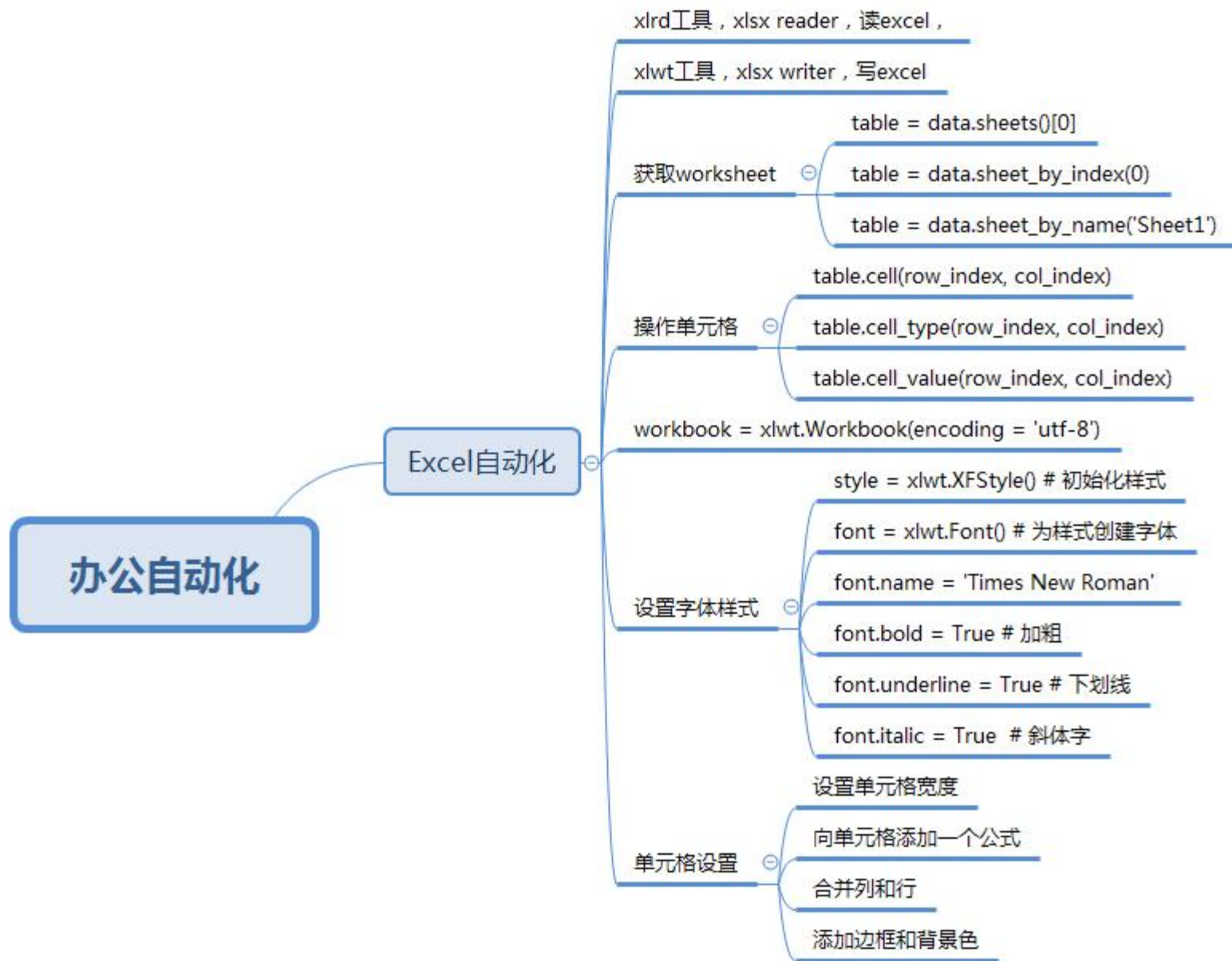
Action1: 针对每天进出楼宇的报表Excel进行自动处理:

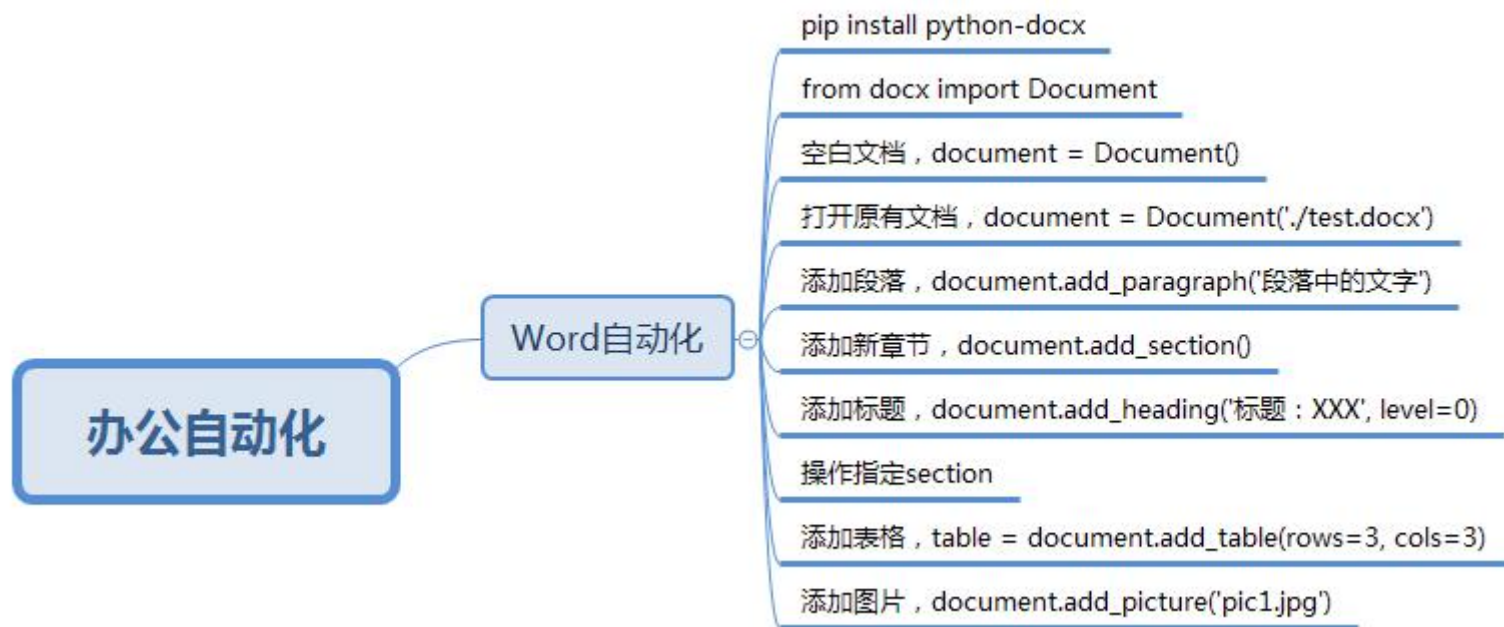
- Step1, 高亮体温 > 37.4度的用户
- Step2, 统计每天进出人员, 平均体温
- Step3, 每日汇总数据自动发邮件



	A	B	C	D
1	人员	体温	时间	
2	张飞	37.4	2020/4/21 9:00	
3	关羽	36.5	2020/4/21 9:30	
4				
5				
6				
7				
8				
9				
10				
11				
12				
13				
14				
15				
16				
17				
18				
19				
20				

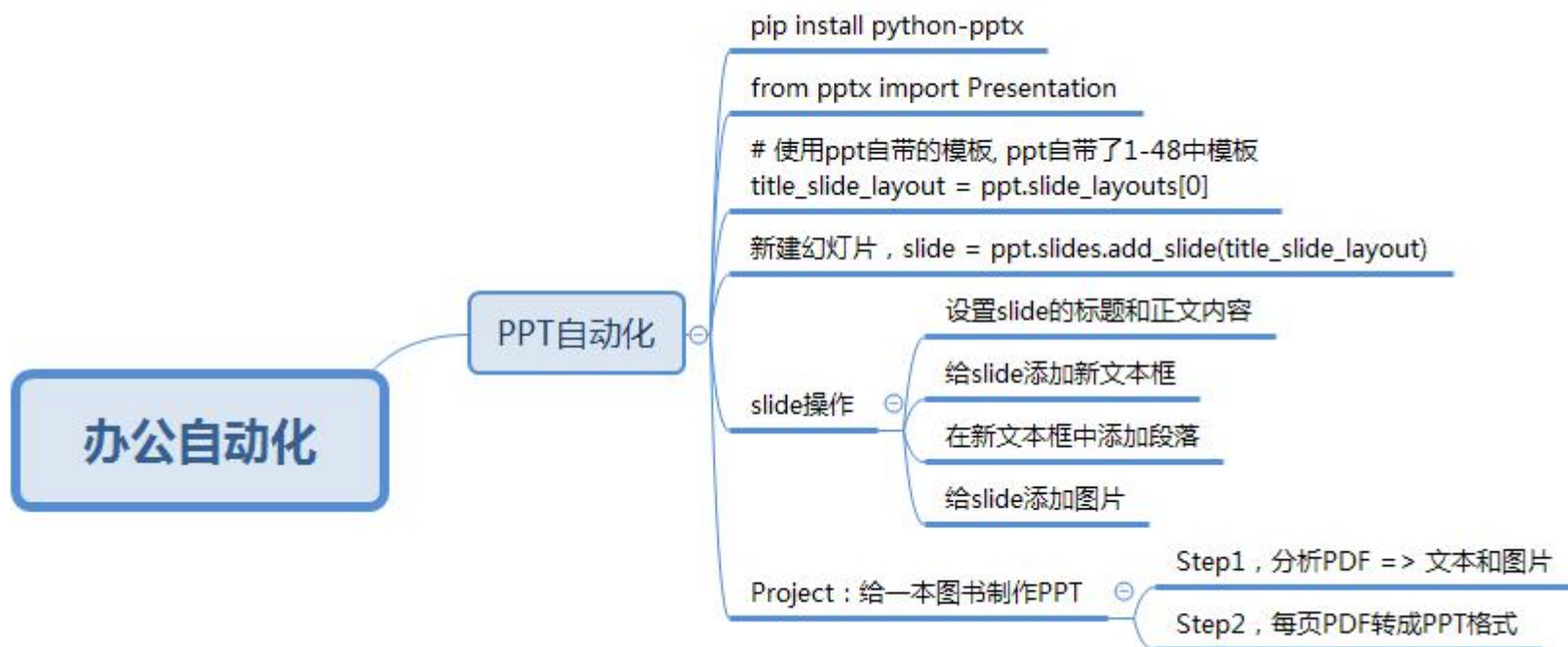
Summary













Thank You
Using data to solve problems

