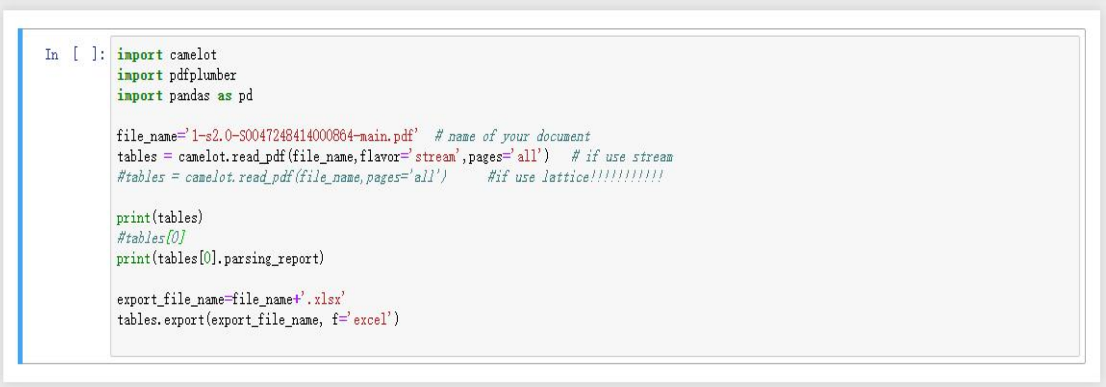
Project_python handbook

1. Extract data from the tables in pdf format document

Code in Jupyter Notebook (screenshot):



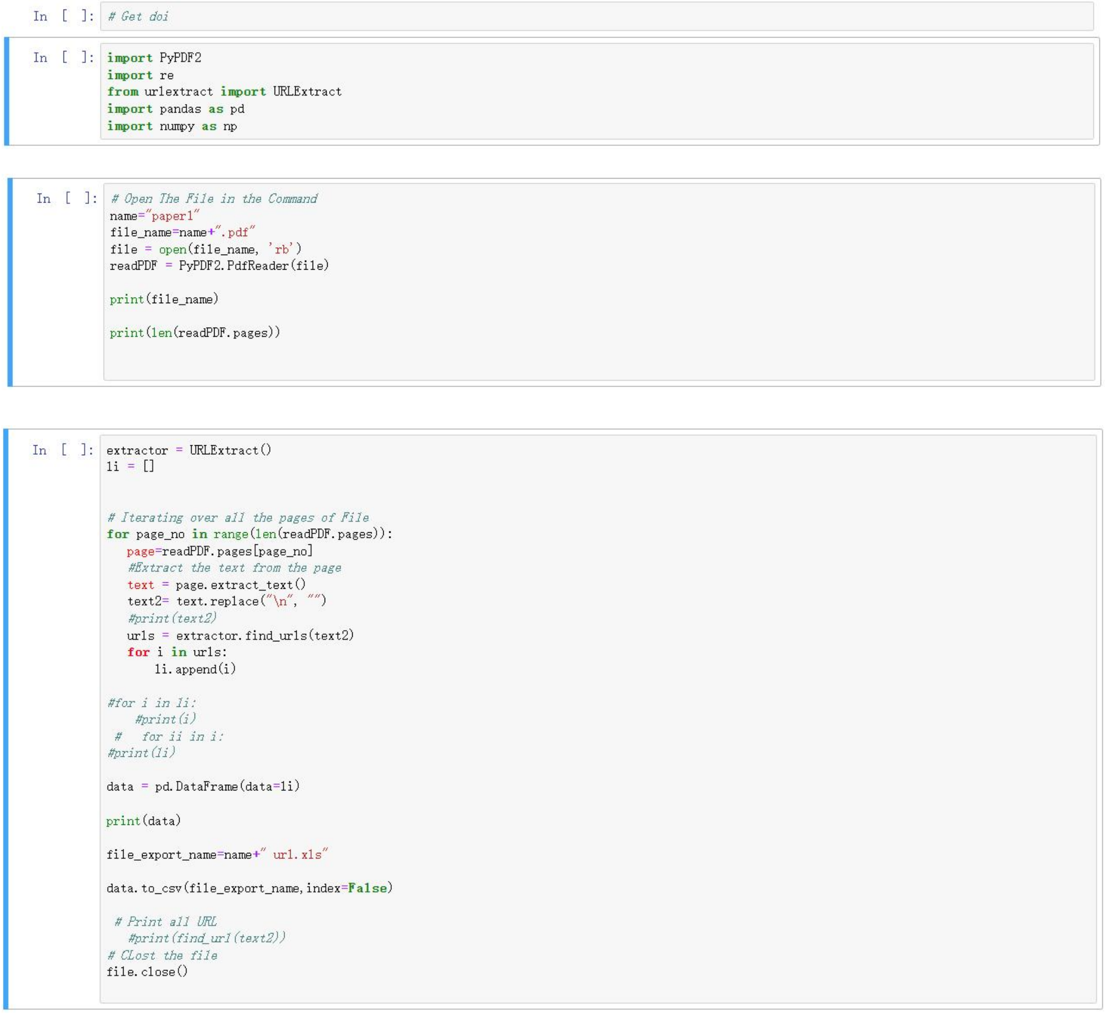
```
import camelot
import pdfplumber
import pandas as pd

file_name='1-s2.0-S0047248414000864-main.pdf'  # name of your document
tables = camelot.read_pdf(file_name,flavor='stream',pages='all')   # if use stream
#tables = camelot.read_pdf(file_name,pages='all')      #if use lattice!!!!!!!!!!!

print(tables)
#tables[0]
print(tables[0].parsing_report)

export_file_name=file_name+'.xlsx'
tables.export(export_file_name, f='excel')
```

Code in text:

```
import camelot
import pdfplumber
import pandas as pd

file_name='1-s2.0-S0047248414000864-main.pdf'   # name of your document
tables = camelot.read_pdf(file_name,flavor='stream',pages='all')    # if use stream
#tables = camelot.read_pdf(file_name,pages='all')        #if use lattice!!!!!!!!!!!

print(tables)
#tables[0]
print(tables[0].parsing_report)

export_file_name=file_name+'.xlsx'
tables.export(export_file_name, f='excel')
```

2. **Extract DOI from paper**

Code in Jupyter Notebook (screenshot):



```
# Get doi
```

```
import PyPDF2
import re
from urlextract import URLExtract
import pandas as pd
import numpy as np
```

```
# Open The File in the Command
name="paper1"
file_name=name+".pdf"
file = open(file_name, 'rb')
readPDF = PyPDF2.PdfReader(file)

print(file_name)

print(len(readPDF.pages))
```

```
extractor = URLExtract()
li = []


# Iterating over all the pages of File
for page_no in range(len(readPDF.pages)):
   page=readPDF.pages[page_no]
   #Extract the text from the page
   text = page.extract_text()
   text2= text.replace("\n", "")
   #print(text2)
   urls = extractor.find_urls(text2)
   for i in urls:
       li.append(i)

#for i in li:
    #print(i)
 #   for ii in i:
#print(li)

data = pd.DataFrame(data=li)

print(data)

file_export_name=name+" url.xls"

data.to_csv(file_export_name,index=False)

 # Print all URL
   #print(find_url(text2))
# CLost the file
file.close()
```

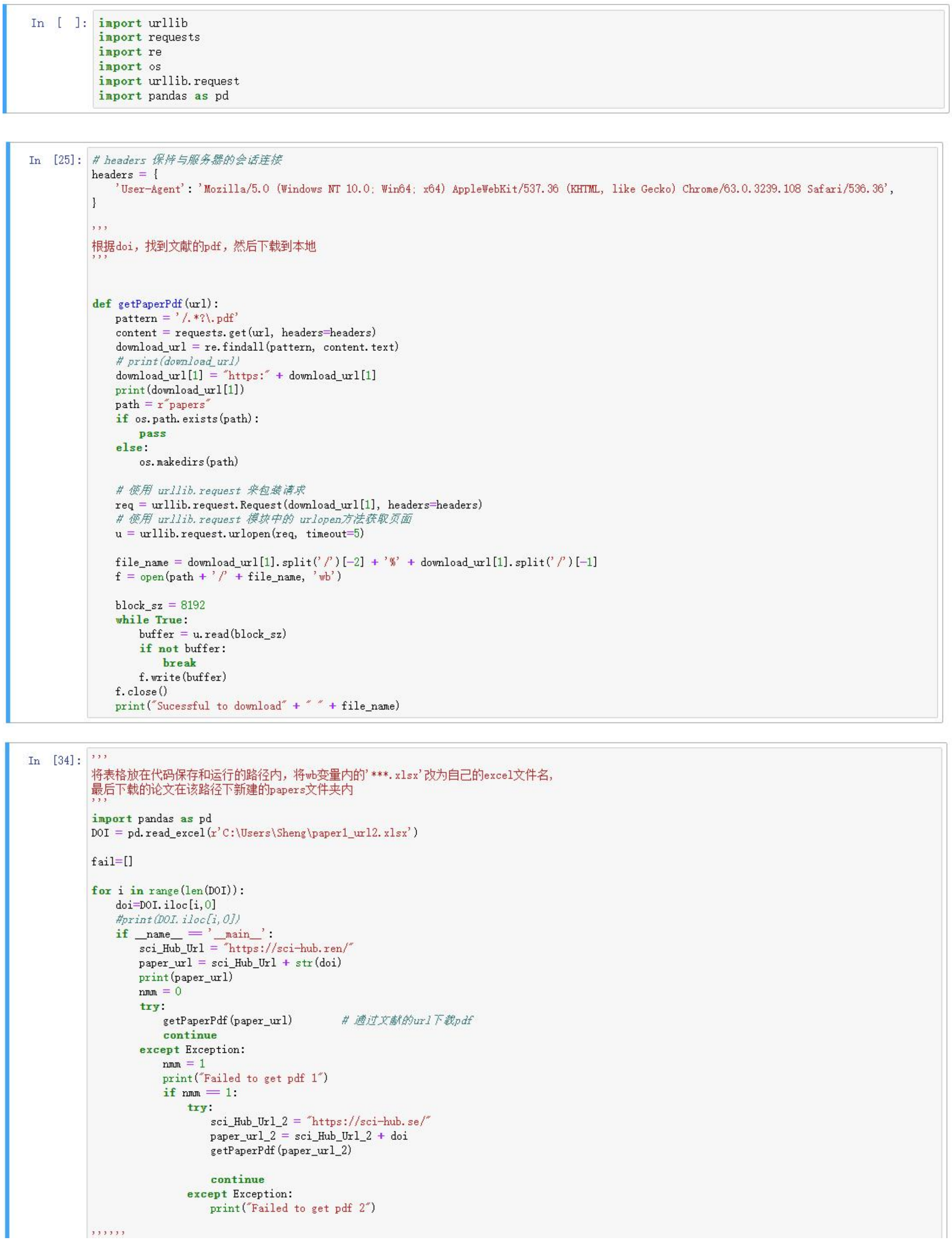
Code in text:

```
import PyPDF2
import re
from urlextract import URLExtract
import pandas as pd
import numpy as np

# Open The File in the Command
name="paper1"
file_name=name+".pdf"
file = open(file_name, 'rb')
readPDF = PyPDF2.PdfReader(file)
```

```
print(file_name)

print(len(readPDF.pages))

extractor = URLExtract()
li = []


# Iterating over all the pages of File
for page_no in range(len(readPDF.pages)):
    page=readPDF.pages[page_no]
    #Extract the text from the page
    text = page.extract_text()
    text2= text.replace("\n", "")
    #print(text2)
    urls = extractor.find_urls(text2)
    for i in urls:
         li.append(i)

#for i in li:
     #print(i)
 #     for ii in i:
#print(li)

data = pd.DataFrame(data=li)

print(data)

file_export_name=name+" url.xls"

data.to_csv(file_export_name,index=False)

 # Print all URL
    #print(find_url(text2))

# Clost the file
file.close()
```

3. **Paper download**

Code in Jupyter Notebook (screenshot):



```
In [ ]: import urllib
        import requests
        import re
        import os
        import urllib.request
        import pandas as pd
```

```
In [25]: # headers 保持与服务器的会话连接
         headers = {
             'User-Agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/63.0.3239.108 Safari/536.36',
         }

         '''
         根据doi，找到文献的pdf，然后下载到本地
         '''


         def getPaperPdf(url):
             pattern = '/.*?\.pdf'
             content = requests.get(url, headers=headers)
             download_url = re.findall(pattern, content.text)
             # print(download_url)
             download_url[1] = "https:" + download_url[1]
             print(download_url[1])
             path = r"papers"
             if os.path.exists(path):
                 pass
             else:
                 os.makedirs(path)

             # 使用 urllib.request 来包装请求
             req = urllib.request.Request(download_url[1], headers=headers)
             # 使用 urllib.request 模块中的 urlopen方法获取页面
             u = urllib.request.urlopen(req, timeout=5)

             file_name = download_url[1].split('/')[-2] + '%' + download_url[1].split('/')[-1]
             f = open(path + '/' + file_name, 'wb')

             block_sz = 8192
             while True:
                 buffer = u.read(block_sz)
                 if not buffer:
                     break
                 f.write(buffer)
             f.close()
             print("Sucessful to download" + " " + file_name)
```

```
In [34]: '''
         将表格放在代码保存和运行的路径内，将wb变量内的'***.xlsx'改为自己的excel文件名，
         最后下载的论文在该路径下新建的papers文件夹内
         '''
         import pandas as pd
         DOI = pd.read_excel(r'C:\Users\Sheng\paper1_url2.xlsx')

         fail=[]

         for i in range(len(DOI)):
             doi=DOI.iloc[i,0]
             #print(DOI.iloc[i,0])
             if __name__ == '__main__':
                 sci_Hub_Url = "https://sci-hub.ren/"
                 paper_url = sci_Hub_Url + str(doi)
                 print(paper_url)
                 nmm = 0
                 try:
                     getPaperPdf(paper_url)        # 通过文献的url下载pdf
                     continue
                 except Exception:
                     nmm = 1
                     print("Failed to get pdf 1")
                     if nmm == 1:
                         try:
                             sci_Hub_Url_2 = "https://sci-hub.se/"
                             paper_url_2 = sci_Hub_Url_2 + doi
                             getPaperPdf(paper_url_2)

                             continue
                         except Exception:
                             print("Failed to get pdf 2")

         ''''''
```

Code in text:

----

import urllib

```
import requests
import re
import os
import urllib.request
import pandas as pd

headers = {
    'User-Agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/63.0.3239.108 Safari/536.36',
}



def getPaperPdf(url):
    pattern = '/.*?\.pdf'
    content = requests.get(url, headers=headers)
    download_url = re.findall(pattern, content.text)
    # print(download_url)
    download_url[1] = "https:" + download_url[1]
    print(download_url[1])
    path = r"papers"
    if os.path.exists(path):
        pass
    else:
        os.makedirs(path)

    req = urllib.request.Request(download_url[1], headers=headers)

    u = urllib.request.urlopen(req, timeout=5)

    file_name = download_url[1].split('/')[-2] + '%' + download_url[1].split('/')[-1]
    f = open(path + '/' + file_name, 'wb')

    block_sz = 8192
    while True:
        buffer = u.read(block_sz)
        if not buffer:
            break
        f.write(buffer)
    f.close()
    print("Sucessful to download" + " " + file_name)

    import pandas as pd
    DOI = pd.read_excel(r'C:\Users\Sheng\paper1_url2.xlsx')
```

```
fail=[]

for i in range(len(DOI)):
    doi=DOI.iloc[i,0]
    #print(DOI.iloc[i,0])
    if __name__ == '__main__':
        sci_Hub_Url = "https://sci-hub.ren/"
        paper_url = sci_Hub_Url + str(doi)
        print(paper_url)
        nmm = 0
        try:
            getPaperPdf(paper_url)
            continue
        except Exception:
            nmm = 1
            print("Failed to get pdf 1")
            if nmm == 1:
                try:
                    sci_Hub_Url_2 = "https://sci-hub.se/"
                    paper_url_2 = sci_Hub_Url_2 + doi
                    getPaperPdf(paper_url_2)

                    continue
                except Exception:
                    print("Failed to get pdf 2")
```