데이터 크롤링과 정제

7장. 문서 읽기

빅데이터 분석가 양성과정

목차

- 문서 인코딩
- PDF

7.2 텍스트

- 일반 텍스트 파일 읽기

```
from urllib.request import urlopen

import requests.exceptions

try:
    text_page =
       urlopen('http://www.pythonscraping.com/pages/warandpeace/chapter1.txt')
    print(text_page.read())
except  requests.exceptions.RequestException:
    print('해당 페이지가 존재하지 않습니다.')
```

```
b'CHAPTER I\n\n"Well, Prince, so Genoa and Lucca are now just family estates of
theBuonapartes. But I warn you, if you don\'t tell me that this means war,if you still try to
defend the infamies and horrors perpetrated bythat Antichrist- I really believe he is
Antichrist- I will havenothing more to do with you and you are no longer my friend, no
longermy \'faithful slave,\' as you call yourself! But how do you do? I seeI have frightened
you- sit down and tell me all the news."\n\nIt was in July, 1805, and the speaker was the
well-known AnnaPavlovna Scherer, maid of honor and favorite of the Empress MaryaFedorovna.
With these words she greeted Prince Vasili Kuragin, a manof high rank and importance, who was
the first to arrive at herreception.

. . .
```

7.2.1 텍스트 인코딩

- ASCII 인코딩
  - 7 비트만 사용: 저장 공간이 매우 비쌌음 (1960년대)
  - 128개의 문자를 인코딩

- Unicode 인코딩
  - 1990년대 초반, 영어 이외의 다른 언어 표현
  - 알파벳, 키릴문자, 중국어, 논리 및 수학 기호, 이모티콘 등 포함

  - UTF-8: Unicode를 인코딩하는 방식
    - 가변 인코딩 방식(1바이트~최대 4바이트)

Binary format of bytes in sequence

1st Byte	2nd Byte	3rd Byte	4th Byte	Number of Free Bits
0xxxxxxx				7
110xxxxx	10xxxxxx			(5+6)=11
1110xxxx	10xxxxxx	10xxxxxx		(4+6+6)=16
11110xxx	10xxxxxx	10xxxxxx	10xxxxxx	(3+6+6+6)=21

  - UTF-16, UTF-24, UTF-32등

http://daplus.net/unicode-utf-8과-유니-코드의-차이점은-무엇입니까/

7.2.1 텍스트 인코딩

- 러시아어와 프랑스어로 쓰인 ‘전쟁과 평화’ 원문

```
# 러시아어와 프랑스어 원문
from urllib.request import urlopen

textPage = urlopen(
              'http://www.pythonscraping.com/pages/warandpeace/chapter1-ru.txt’)

print(textPage.read())
```

```
b"\xd0\xa7\xd0\x90\xd0\xa1\xd0\xa2\xd0\xac
\xd0\x9f\xd0\x95\xd0\xa0\xd0\x92\xd0\x90\xd0\xaf\n\nI\n\n\xe2\x80\x94 Eh bien,

. . .
```

- python에서 해당 문서를 ASCII 문서로 읽으려고 시도
- 명확히 UTF-8로 encoding 하도록 명시해야 됨

7.2.1 텍스트 인코딩

- str 클래스 생성자

```
class str(object=b'', encoding='utf-8', errors='strict')
```

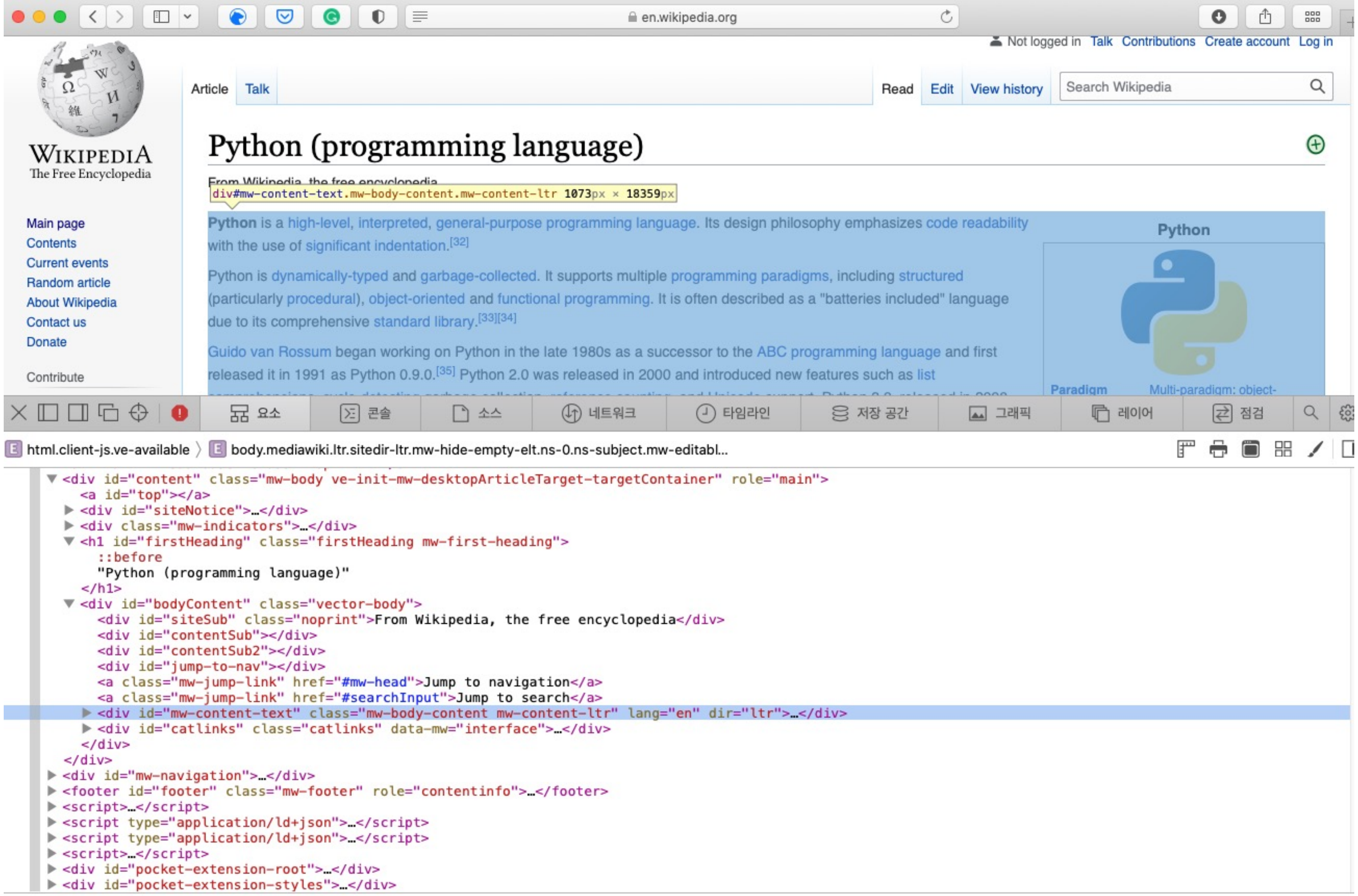
- str 클래스를 이용하여 UTF-8로 인코딩

```
from urllib.request import urlopen
textPage = urlopen(
              'http://www.pythonscraping.com/pages/warandpeace/chapter1-ru.txt')
print(str(textPage.read(), 'utf-8'))
```

```
ЧАСТЬ ПЕРВАЯ

I

— Eh bien, mon prince. Gênes et Lucques ne sont plus que des apanages, des поместья, de la
famille Buonaparte. Non, je vous préviens que si vous ne me dites pas que nous avons la
guerre, si vous vous permettez encore de pallier toutes les infamies, toutes les atrocités
de cet Antichrist (ma parole, j'y crois) — je ne vous connais plus, vous n'êtes plus mon ami,
vous n'êtes plus мой верный раб, comme vous dites. Ну, здравствуйте, здравствуйте. Je vois
que je vous fais peur, садитесь и рассказывайте.
Так говорила в июле 1805 года известная Анна Павловна Шерер, фрейлина и приближенная
императрицы
```

7.2.1 텍스트 인코딩

- BeautifulSoup에 인코딩 적용
  - URL: https://en.wikipedia.org/wiki/Python_(programming_language)



Python (programming language)

From Wikipedia, the free encyclopedia

div#mw-content-text.mw-body-content.mw-content-ltr 1073px × 18359px

Python is a high-level, interpreted, general-purpose programming language. Its design philosophy emphasizes code readability with the use of significant indentation.[32]

Python is dynamically-typed and garbage-collected. It supports multiple programming paradigms, including structured (particularly procedural), object-oriented and functional programming. It is often described as a "batteries included" language due to its comprehensive standard library.[33][34]

Guido van Rossum began working on Python in the late 1980s as a successor to the ABC programming language and first released it in 1991 as Python 0.9.0.[35] Python 2.0 was released in 2000 and introduced new features such as list

html.client-js.ve-available 〉 body.mediawiki.ltr.sitedir-ltr.mw-hide-empty-elt.ns-0.ns-subject.mw-editabl...

```
▼ <div id="content" class="mw-body ve-init-mw-desktopArticleTarget-targetContainer" role="main">
    <a id="top"></a>
  ▶ <div id="siteNotice">…</div>
  ▶ <div class="mw-indicators">…</div>
  ▼ <h1 id="firstHeading" class="firstHeading mw-first-heading">
      ::before
      "Python (programming language)"
    </h1>
  ▼ <div id="bodyContent" class="vector-body">
      <div id="siteSub" class="noprint">From Wikipedia, the free encyclopedia</div>
      <div id="contentSub"></div>
      <div id="contentSub2"></div>
      <div id="jump-to-nav"></div>
      <a class="mw-jump-link" href="#mw-head">Jump to navigation</a>
      <a class="mw-jump-link" href="#searchInput">Jump to search</a>
    ▶ <div id="mw-content-text" class="mw-body-content mw-content-ltr" lang="en" dir="ltr">…</div>
    ▶ <div id="catlinks" class="catlinks" data-mw="interface">…</div>
    </div>
  </div>
▶ <div id="mw-navigation">…</div>
▶ <footer id="footer" class="mw-footer" role="contentinfo">…</footer>
▶ <script>…</script>
▶ <script type="application/ld+json">…</script>
▶ <script type="application/ld+json">…</script>
▶ <script>…</script>
▶ <div id="pocket-extension-root">…</div>
▶ <div id="pocket-extension-styles">…</div>
```

7.2.1 텍스트 인코딩

- BeautifulSoup 적용 예제
  - bytes(source, encoding='utf-8') 함수
    - 문자열을 bytes형태로 encoding

```
from urllib.request import urlopen
from bs4 import BeautifulSoup

html = urlopen("http://en.wikipedia.org/wiki/Python_(programming_language)")
bs = BeautifulSoup(html, "html.parser")
content = bs.find("div", {"id":"mw-content-text"}).get_text()

# 문자열(str)을 bytes 형태로 encoding
content = bytes(content, "utf-8")
# bytes 형태를 UTF-8 문자열로 decoding
content = content.decode(”utf-8")
print(content)
```

```
General-purpose programming language


PythonParadigmMulti-paradigm: object-oriented,[1] procedural (imperative), functional,
structured, reflectiveDesigned byGuido van RossumDeveloperPython Software
FoundationFirst appeared20 February 1991; 31 years ago (1991-02-20)[2]Stable
release3.10.5[3]
   / 6 June 2022; 27 days ago (6 June 2022)Preview release3.11.0b3[4]
   / 1 June 2022; 32 days ago (1 June 2022)
. . .
```

7.3 CSV 파일 읽기: StringIO

- StringIO 클래스
  - 문자 스트림 데이터를 파일 객체처럼 사용할 수 있는 기능 제공
  - 데이터가 파일로 저장되는 대신, 메모리의 버퍼에 기록
  - 특정 URL에 존재하는 csv파일을 문자 스트림으로 읽고 이를 파일처럼 처리 (read(), write() 등)
  - close() 함수: 메모리의 버퍼가 삭제됨

```python
from urllib.request import urlopen
from io import StringIO
import csv

data = urlopen('https://pythonscraping.com/files/MontyPythonAlbums.csv')
data = data.read() # bytes 타입

# errors=‘ignore’, UnicodeEncodeError 무시
data = data.decode(encoding='ascii', errors='ignore')
dataFile = StringIO(data)
csvReader = csv.reader(dataFile)

for row in csvReader:
    print(row)

dataFile.close()
```

```
['Name', 'Year']
["Monty Python's Flying Circus", '1970']
['Another Monty Python Record', '1971']
["Monty Python's Previous Record", '1972']
['The Monty Python Matching Tie and Handkerchief', '1973']
['Monty Python Live at Drury Lane', '1974']
```

7.3 CSV 파일 읽기: csv.DictReader

- csv.DictReader
  - csv 파일의 각 행을 읽어서 딕셔너리 객체를 리턴
  - 딕셔너리의 필드 이름: fieldnames에 저장

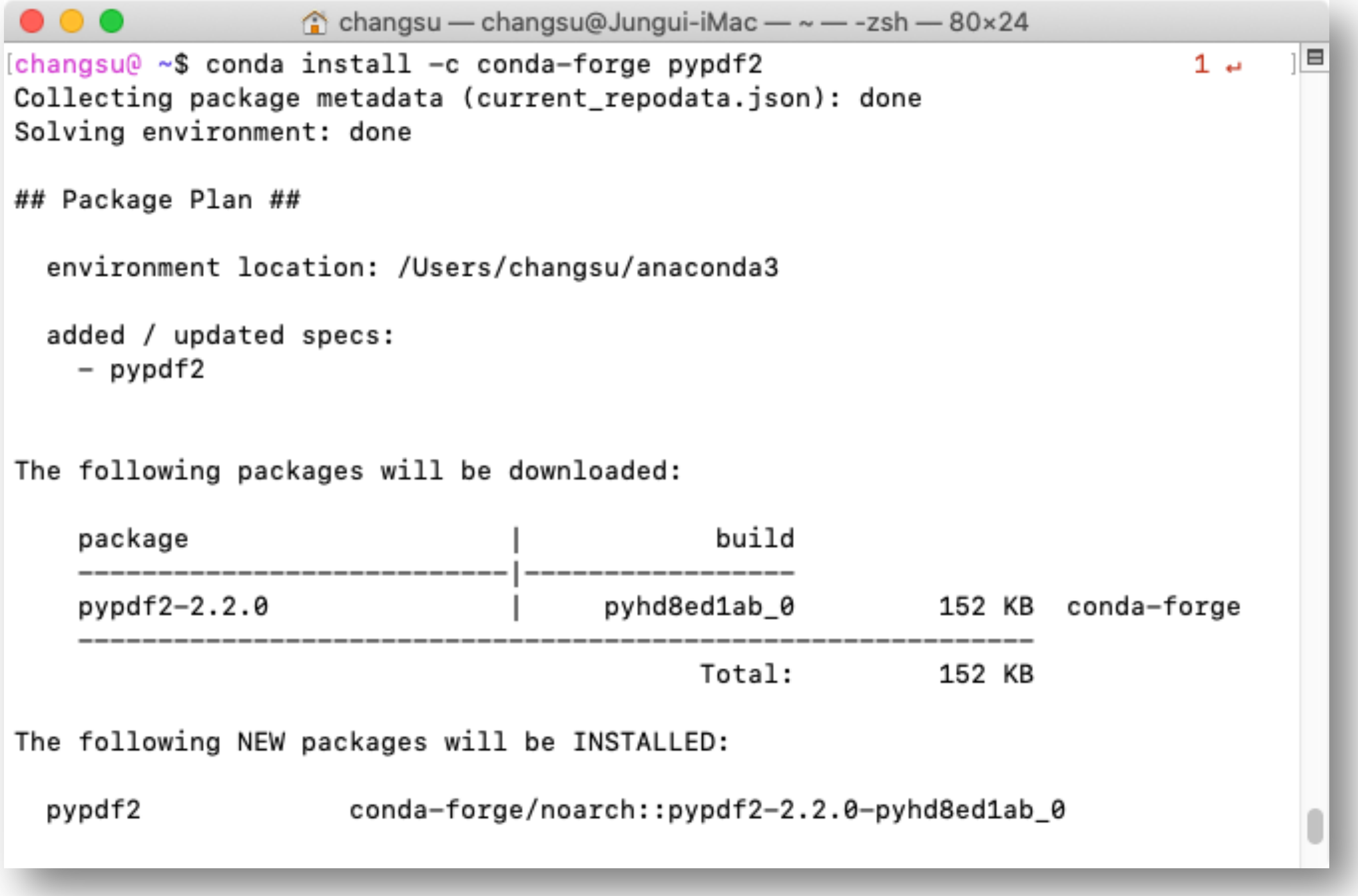
```
from urllib.request import urlopen
from io import StringIO
import csv

data = urlopen('https://pythonscraping.com/files/MontyPythonAlbums.csv')
data = data.read() # bytes 타입

# UnicodeEncodeError 무시
data = data.decode(encoding='ascii', errors='ignore')

dataFile = StringIO(data)
dictReader = csv.DictReader(dataFile)

print(dictReader.fieldnames) # 딕셔너리의 필드 이름 출력

for row in dictReader:
    print(row)
    #print(row['Name'], row['Year'])
```

```
['Name', 'Year']
OrderedDict([('Name', "Monty Python's Flying Circus"), ('Year', '1970')])
OrderedDict([('Name', 'Another Monty Python Record'), ('Year', '1971')])
OrderedDict([('Name', "Monty Python's Previous Record"), ('Year', '1972')])
```

7.4 PDF 파일 읽기: PyPDF2 설치

- PyPDF2 라이브러리 설치

```
$ conda install -c conda-forge pypdf2
```



```
[changsu@ ~$ conda install -c conda-forge pypdf2                          1 ↵   ]
Collecting package metadata (current_repodata.json): done
Solving environment: done

## Package Plan ##

  environment location: /Users/changsu/anaconda3

  added / updated specs:
    - pypdf2


The following packages will be downloaded:

    package                    |            build
    ---------------------------|-----------------
    pypdf2-2.2.0               |     pyhd8ed1ab_0         152 KB  conda-forge
    ------------------------------------------------------------
                                           Total:         152 KB

The following NEW packages will be INSTALLED:

  pypdf2             conda-forge/noarch::pypdf2-2.2.0-pyhd8ed1ab_0
```

https://wikidocs.net/153818

7.4 PDF 파일 읽기: PyPDF2 사용

```python
from urllib.request import urlopen
from PyPDF2 import PdfFileReader
from io import BytesIO

url = 'http://pythonscraping.com/pages/warandpeace/chapter1.pdf'

remote_file = urlopen(url).read()
memory_file = BytesIO(remote_file)

pdf_file = PdfFileReader(memory_file)
page_num = pdf_file.getNumPages()

for i in range(page_num):
    page = pdf_file.getPage(i).extract_text()
    print('Page {0}:'.format(i+1))
    print(page)

print('End of file')
```

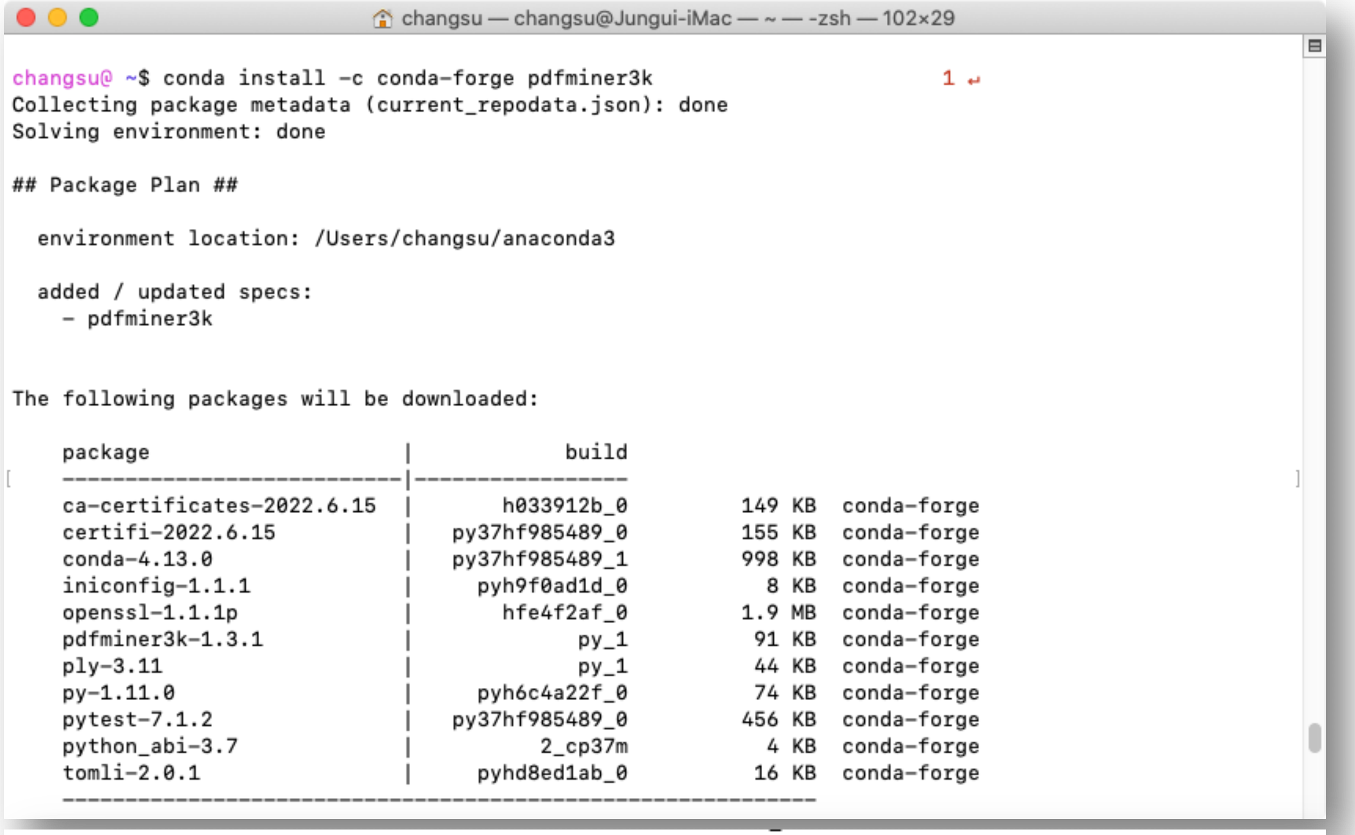
```
Page 1:
CHAPTER I"Well, Prince, so Genoa and Lucca are now just family estates of theBuonapartes. But I warn you, if you
don't tell me that this means war,if you still try to defend the infamies and horrors perpetrated bythat Antichrist- I
really believe he is Antichrist- I will havenothing more to do with you and you are no longer my friend, no longermy
'faithful slave,' as you call yourself! But how do you do? I seel have frightened you- sit down and tell me all the
. . .
Page 7:
. . .
End of file
```

참고 자료

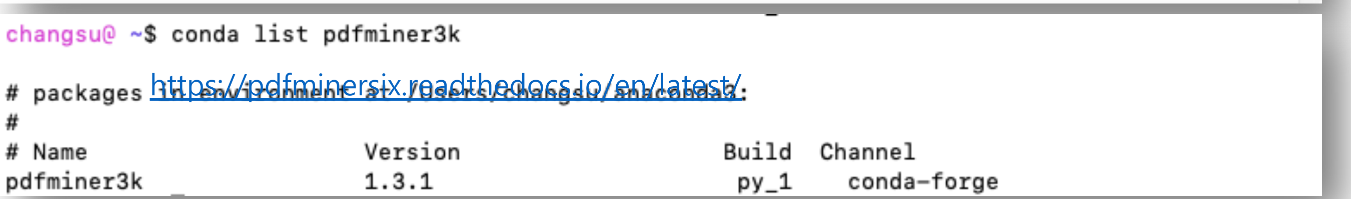
7.4 PDF 파일 읽기

- PDF 라이브러리 설치(PDFMiner3K)

```
$ conda install -c conda-forge pdfminer3k
```



```
changsu@ ~$ conda install -c conda-forge pdfminer3k                    1 ↵
Collecting package metadata (current_repodata.json): done
Solving environment: done

## Package Plan ##

  environment location: /Users/changsu/anaconda3

  added / updated specs:
    - pdfminer3k


The following packages will be downloaded:

    package                    |            build
    ---------------------------|-----------------
    ca-certificates-2022.6.15  |       h033912b_0         149 KB  conda-forge
    certifi-2022.6.15          |   py37hf985489_0         155 KB  conda-forge
    conda-4.13.0               |   py37hf985489_1         998 KB  conda-forge
    iniconfig-1.1.1            |     pyh9f0ad1d_0           8 KB  conda-forge
    openssl-1.1.1p             |       hfe4f2af_0         1.9 MB  conda-forge
    pdfminer3k-1.3.1           |             py_1          91 KB  conda-forge
    ply-3.11                   |             py_1          44 KB  conda-forge
    py-1.11.0                  |     pyh6c4a22f_0          74 KB  conda-forge
    pytest-7.1.2               |   py37hf985489_0         456 KB  conda-forge
    python_abi-3.7             |          2_cp37m           4 KB  conda-forge
    tomli-2.0.1                |     pyhd8ed1ab_0          16 KB  conda-forge
    ------------------------------------------------------------
```



```
changsu@ ~$ conda list pdfminer3k

# packages in environment at /Users/changsu/anaconda3:
#
# Name                    Version                   Build  Channel
pdfminer3k                1.3.1                      py_1    conda-forge
```

https://pdfminersix.readthedocs.io/en/latest/

Questions?