1.Download financial report of any company. 10*7

2.Use camelot and tabula to extracts all the tables in the pdf.

3.Convert and tablelist to Dataframe and save them in CSV, EXCEL sheet TSV , Jason , HTML.

Read single and multiple tabels from pdf

```
#!pip install camelot-py[cv] tabula-py
#!pip install ghostscript
#!pip install ghostscript
#!pip install camelot-py[cv]
#!pip install excalibur-py
#!excalibur initdb

import tabula
pdf_path = "foo.pdf"

dfs = tabula.read_pdf(pdf_path, stream=True)
# read_pdf returns list of DataFrames
print(len(dfs))
dfs[0]
tabula.read_pdf(pdf_path, pages="all", stream=True)
```

```
'pages' argument isn't specified.Will extract only from page 1 by default.
1
[  Unnamed: 0 Unnamed: 1 Unnamed: 2 Unnamed: 3 Percent Fuel Savings Unnamed: 4
 0      Cycle         KI   Distance        NaN                  NaN        NaN
 1       Name     (1/km)       (mi)   Improved  Decreased Eliminate  Decreased
 2        NaN        NaN        NaN      Speed          Accel Stops       Idle
 3     2012_2       3.30        1.3       5.9%           9.5% 29.2%      17.4%
 4     2145_1       0.68       11.2       2.4%            0.1% 9.5%       2.7%
 5     4234_1       0.59       58.7       8.5%            1.3% 8.5%       3.3%
 6     2032_2       0.17       57.8      21.7%            0.3% 2.7%       1.2%
 7     4171_1       0.07      173.9      58.1%            1.6% 2.1%       0.5%]
```

Double-click (or enter) to edit

```
!pip install camelot-py[cv] tabula-py
!sudo apt install ghostscript
```

```
Collecting camelot-py[cv]
  Downloading camelot_py-0.10.1-py3-none-any.whl (40 kB)
     |████████████████████████████████| 40 kB 25 kB/s
Collecting tabula-py
  Downloading tabula_py-2.2.0-py3-none-any.whl (11.7 MB)
     |████████████████████████████████| 11.7 MB 6.9 MB/s
Collecting distro
  Downloading distro-1.6.0-py2.py3-none-any.whl (19 kB)
Requirement already satisfied: numpy in /usr/local/lib/python3.7/dist-packages (from tabula-py) (1.19.5)
Requirement already satisfied: pandas>=0.25.3 in /usr/local/lib/python3.7/dist-packages (from tabula-py) (1.1.5)
Requirement already satisfied: python-dateutil>=2.7.3 in /usr/local/lib/python3.7/dist-packages (from pandas>=0.25.3-
Requirement already satisfied: pytz>=2017.2 in /usr/local/lib/python3.7/dist-packages (from pandas>=0.25.3->tabula-py
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.7/dist-packages (from python-dateutil>=2.7.3->panda
Collecting PyPDF2>=1.26.0
  Downloading PyPDF2-1.26.0.tar.gz (77 kB)
     |████████████████████████████████| 77 kB 7.1 MB/s
Collecting pdfminer.six>=20200726
  Downloading pdfminer.six-20201018-py3-none-any.whl (5.6 MB)
     |████████████████████████████████| 5.6 MB 52.1 MB/s
Requirement already satisfied: openpyxl>=2.5.8 in /usr/local/lib/python3.7/dist-packages (from camelot-py[cv]) (2.5.9
Requirement already satisfied: tabulate>=0.8.9 in /usr/local/lib/python3.7/dist-packages (from camelot-py[cv]) (0.8.9
Requirement already satisfied: chardet>=3.0.4 in /usr/local/lib/python3.7/dist-packages (from camelot-py[cv]) (3.0.4)
Requirement already satisfied: click>=6.7 in /usr/local/lib/python3.7/dist-packages (from camelot-py[cv]) (7.1.2)
Requirement already satisfied: opencv-python>=3.4.2.17 in /usr/local/lib/python3.7/dist-packages (from camelot-py[cv]
Collecting pdftopng>=0.2.3
  Downloading pdftopng-0.2.3-cp37-cp37m-manylinux2010_x86_64.whl (11.7 MB)
     |████████████████████████████████| 11.7 MB 28.8 MB/s
Collecting ghostscript>=0.7
  Downloading ghostscript-0.7-py2.py3-none-any.whl (25 kB)
Requirement already satisfied: setuptools>=38.6.0 in /usr/local/lib/python3.7/dist-packages (from ghostscript>=0.7->c
Requirement already satisfied: jdcal in /usr/local/lib/python3.7/dist-packages (from openpyxl>=2.5.8->camelot-py[cv])
Requirement already satisfied: et-xmlfile in /usr/local/lib/python3.7/dist-packages (from openpyxl>=2.5.8->camelot-py
Collecting cryptography
  Downloading cryptography-3.4.7-cp36-abi3-manylinux2014_x86_64.whl (3.2 MB)
     |████████████████████████████████| 3.2 MB 34.0 MB/s
```

```
Requirement already satisfied: sortedcontainers in /usr/local/lib/python3.7/dist-packages (from pdfminer.six>=2020072
Requirement already satisfied: cffi>=1.12 in /usr/local/lib/python3.7/dist-packages (from cryptography->pdfminer.six>
Requirement already satisfied: pycparser in /usr/local/lib/python3.7/dist-packages (from cffi>=1.12->cryptography->pd
Building wheels for collected packages: PyPDF2
  Building wheel for PyPDF2 (setup.py) ... done
  Created wheel for PyPDF2: filename=PyPDF2-1.26.0-py3-none-any.whl size=61100 sha256=980bd2e547003f8a0d32403428016e8
  Stored in directory: /root/.cache/pip/wheels/80/1a/24/648467ade3a77ed20f35cfd2badd32134e96dd25ca811e64b3
Successfully built PyPDF2
Installing collected packages: cryptography, PyPDF2, pdfminer.six, pdftopng, ghostscript, distro, camelot-py, tabula-
Successfully installed PyPDF2-1.26.0 camelot-py-0.10.1 cryptography-3.4.7 distro-1.6.0 ghostscript-0.7 pdfminer.six-2
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  fonts-droid-fallback fonts-noto-mono gsfonts libcupsfilters1 libcupsimage2
  libgs9 libgs9-common libijs-0.35 libjbig2dec0 poppler-data
Suggested packages:
  fonts-noto ghostscript-x poppler-utils fonts-japanese-mincho
  | fonts-ipafont-mincho fonts-japanese-gothic | fonts-ipafont-gothic
  fonts-arphic-ukai fonts-arphic-uming fonts-nanum
The following NEW packages will be installed:
  fonts-droid-fallback fonts-noto-mono ghostscript gsfonts libcupsfilters1
  libcupsimage2 libgs9 libgs9-common libijs-0.35 libjbig2dec0 poppler-data
```

```
#import tabula
!pip install camelot-py[cv] tabula-py
#tabula.environment_info()
```

```
Collecting camelot-py[cv]
  Downloading camelot_py-0.10.1-py3-none-any.whl (40 kB)
     |████████████████████████████████| 40 kB 37 kB/s
Collecting tabula-py
  Downloading tabula_py-2.2.0-py3-none-any.whl (11.7 MB)
     |████████████████████████████████| 11.7 MB 244 kB/s
Requirement already satisfied: numpy in /usr/local/lib/python3.7/dist-packages (from tabula-py) (1.19.5)
Requirement already satisfied: pandas>=0.25.3 in /usr/local/lib/python3.7/dist-packages (from tabula-py) (1.1.5)
Collecting distro
  Downloading distro-1.6.0-py2.py3-none-any.whl (19 kB)
Requirement already satisfied: pytz>=2017.2 in /usr/local/lib/python3.7/dist-packages (from pandas>=0.25.3->tabula-py)
```

```
Requirement already satisfied: python-dateutil>=2.7.3 in /usr/local/lib/python3.7/dist-packages (from pandas>=0.25.3->t
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.7/dist-packages (from python-dateutil>=2.7.3->pandas>
Requirement already satisfied: tabulate>=0.8.9 in /usr/local/lib/python3.7/dist-packages (from camelot-py[cv]) (0.8.9)
Requirement already satisfied: chardet>=3.0.4 in /usr/local/lib/python3.7/dist-packages (from camelot-py[cv]) (3.0.4)
Requirement already satisfied: openpyxl>=2.5.8 in /usr/local/lib/python3.7/dist-packages (from camelot-py[cv]) (2.5.9)
Requirement already satisfied: click>=6.7 in /usr/local/lib/python3.7/dist-packages (from camelot-py[cv]) (7.1.2)
Collecting PyPDF2>=1.26.0
  Downloading PyPDF2-1.26.0.tar.gz (77 kB)
     |████████████████████████████████| 77 kB 8.9 MB/s
Collecting pdfminer.six>=20200726
  Downloading pdfminer.six-20201018-py3-none-any.whl (5.6 MB)
     |████████████████████████████████| 5.6 MB 27.1 MB/s
Collecting pdftopng>=0.2.3
  Downloading pdftopng-0.2.3-cp37-cp37m-manylinux2010_x86_64.whl (11.7 MB)
     |████████████████████████████████| 11.7 MB 29.7 MB/s
Collecting ghostscript>=0.7
  Downloading ghostscript-0.7-py2.py3-none-any.whl (25 kB)
Requirement already satisfied: opencv-python>=3.4.2.17 in /usr/local/lib/python3.7/dist-packages (from camelot-py[cv])
Requirement already satisfied: setuptools>=38.6.0 in /usr/local/lib/python3.7/dist-packages (from ghostscript>=0.7->cam
Requirement already satisfied: et-xmlfile in /usr/local/lib/python3.7/dist-packages (from openpyxl>=2.5.8->camelot-py[c
Requirement already satisfied: jdcal in /usr/local/lib/python3.7/dist-packages (from openpyxl>=2.5.8->camelot-py[cv]) (
Requirement already satisfied: sortedcontainers in /usr/local/lib/python3.7/dist-packages (from pdfminer.six>=20200726-
Collecting cryptography
  Downloading cryptography-3.4.7-cp36-abi3-manylinux2014_x86_64.whl (3.2 MB)
     |████████████████████████████████| 3.2 MB 44.5 MB/s
Requirement already satisfied: cffi>=1.12 in /usr/local/lib/python3.7/dist-packages (from cryptography->pdfminer.six>=2
Requirement already satisfied: pycparser in /usr/local/lib/python3.7/dist-packages (from cffi>=1.12->cryptography->pdfm
Building wheels for collected packages: PyPDF2
  Building wheel for PyPDF2 (setup.py) ... done
  Created wheel for PyPDF2: filename=PyPDF2-1.26.0-py3-none-any.whl size=61100 sha256=071151168fb88ed80eb7749e45963f7fc
  Stored in directory: /root/.cache/pip/wheels/80/1a/24/648467ade3a77ed20f35cfd2badd32134e96dd25ca811e64b3
Successfully built PyPDF2
Installing collected packages: cryptography, PyPDF2, pdfminer.six, pdftopng, ghostscript, distro, camelot-py, tabula-py
Successfully installed PyPDF2-1.26.0 camelot-py-0.10.1 cryptography-3.4.7 distro-1.6.0 ghostscript-0.7 pdfminer.six-202
```

```
import tabula
```

```
tabula.environment_info()
```

```
Python version:
    3.7.11 (default, Jul  3 2021, 18:01:19)
```

```
[GCC 7.5.0]
Java version:
    openjdk version "11.0.11" 2021-04-20
OpenJDK Runtime Environment (build 11.0.11+9-Ubuntu-0ubuntu2.18.04)
OpenJDK 64-Bit Server VM (build 11.0.11+9-Ubuntu-0ubuntu2.18.04, mixed mode, sharing)
tabula-py version: 2.2.0
platform: Linux-5.4.104+-x86_64-with-Ubuntu-18.04-bionic
uname:
    uname_result(system='Linux', node='f1eec458cb86', release='5.4.104+', version='#1 SMP Sat Jun 5 09:50:34 PDT 2021',
linux_distribution: ('Ubuntu', '18.04', 'bionic')
mac_ver: ('', ('', '', ''), '')
```

```
import tabula
pdf_path = "foo.pdf"

dfs = tabula.read_pdf(pdf_path, stream=True)
# read_pdf returns list of DataFrames
print(len(dfs))
dfs[0]
```

```
'pages' argument isn't specified.Will extract only from page 1 by default.
1
Error from tabula-java:
Exception in thread "main" java.lang.IndexOutOfBoundsException: Page number does not exist
        at technology.tabula.ObjectExtractor.extractPage(ObjectExtractor.java:19)
        at technology.tabula.PageIterator.next(PageIterator.java:29)
        at technology.tabula.CommandLineApp.extractFile(CommandLineApp.java:166)
        at technology.tabula.CommandLineApp.extractFileTables(CommandLineApp.java:129)
        at technology.tabula.CommandLineApp.extractTables(CommandLineApp.java:111)
        at technology.tabula.CommandLineApp.main(CommandLineApp.java:81)


---------------------------------------------------------------------------
CalledProcessError                        Traceback (most recent call last)
<ipython-input-7-5b6dc4cba1ee> in <module>()
      7 dfs[0]
      8
```

```
Error from tabula-java:
Exception in thread "main" java.lang.IndexOutOfBoundsException: Page number does not exist
        at technology.tabula.ObjectExtractor.extractPage(ObjectExtractor.java:19)
        at technology.tabula.PageIterator.next(PageIterator.java:29)
        at technology.tabula.CommandLineApp.extractFile(CommandLineApp.java:166)
        at technology.tabula.CommandLineApp.extractFileTables(CommandLineApp.java:129)
        at technology.tabula.CommandLineApp.extractTables(CommandLineApp.java:111)
        at technology.tabula.CommandLineApp.main(CommandLineApp.java:81)
```

```
import tabula
pdf_path = "foo.pdf"

#dfs = tabula.read_pdf(pdf_path, stream=True)
# read_pdf returns list of DataFrames
#print(len(dfs))
#dfs[0]

#tabula.read_pdf(pdf_path, pages="1-2,3", stream=True)
dfs = tabula.read_pdf(pdf_path, columns=[0,2], guess=False, pages=1)
df = dfs[0].drop(["Unnamed: 0"], axis=1)
df
```

	Unnamed: 1	2 Quantifying Fuel-Saving Opportunities from Specific Driving
0	NaN	Behavior Changes
1	NaN	2.1 Savings from Improving Individual Driving ...
2	NaN	2.1.1 Drive Profile Subsample from Real-World ...
3	NaN	The interim report (Gonder et al. 2010) includ...
4	NaN	selected from a large set of real-world global...
5	NaN	2006 as part of a study by the Texas Transport...
6	NaN	Transportation (Ojah and Pearson 2008). The cy...
7	NaN	intensity (KI) values. (KI represents a ratio ...
8	NaN	and has been shown to be a useful drive cycle ...
9	NaN	To determine the maximum possible cycle improv...
10	NaN	converted into equivalent “ideal” cycles using...
11	NaN	1. Calculate the trip distance of each sample ...
12	NaN	2. Eliminate stop-and-go and idling within eac...
13	NaN	3. Set the acceleration rate to 3 mph/s.
14	NaN	4. Set the cruising speed to 40 mph.
15	NaN	5. Continue cruising at 40 mph until the trip ...
16	NaN	To compare vehicle simulations over each real-...
17	NaN	midsize conventional vehicle model from a prev...
18	NaN	2010). The results indicated a fuel savings po...
19	NaN	either very high or very low KI and of 30%–40%...
20	NaN	Table 2-1 takes the analysis of these five cyc...
21	NaN	examining the impact of the optimization steps...
22	NaN	simulations from the interim report (Gonder et...

22	NaN	simulations from the interim report (Sensor Sim...
23	NaN	some small fuel savings, but avoiding accelera...
24	NaN	saves larger amounts of fuel. This suggests th...
25	NaN	reducing the number of stops in high KI cycles...
26	NaN	Table 2-1. Simulated fuel savings from isolate...
27	NaN	Percent Fuel Savings
28	NaN	Cycle KI Distance
29	NaN	Name (1/km) (mi) Improved Decreased Eliminate ...

```
import tabula
pdf_path = "foo.pdf"
tabula.read_pdf(pdf_path, output_format="json")
```

```
'pages' argument isn't specified.Will extract only from page 1 by default.
[{'bottom': 675.0,
  'data': [[{'height': 0.0, 'left': 0.0, 'text': '', 'top': 0.0, 'width': 0.0},
    {'height': 0.0, 'left': 0.0, 'text': '', 'top': 0.0, 'width': 0.0},
    {'height': 0.0, 'left': 0.0, 'text': '', 'top': 0.0, 'width': 0.0},
    {'height': 0.0, 'left': 0.0, 'text': '', 'top': 0.0, 'width': 0.0},
    {'height': 4.699999809265137,
     'left': 323.93,
     'text': 'Percent Fuel Savings',
     'top': 564.65,
     'width': 103.89000701904297},
    {'height': 0.0, 'left': 0.0, 'text': '', 'top': 0.0, 'width': 0.0}],
   [{'height': 4.699999809265137,
     'left': 129.23,
     'text': 'Cycle',
     'top': 570.64,
     'width': 29.420005798339844},
    {'height': 4.699999809265137,
     'left': 179.81,
     'text': 'KI',
     'top': 570.64,
     'width': 12.810004234313965},
    {'height': 4.699999809265137,
```

```
            'left': 210.41,
            'text': 'Distance',
            'top': 570.64,
            'width': 44.48999786376953},
           {'height': 0.0, 'left': 0.0, 'text': '', 'top': 0.0, 'width': 0.0},
           {'height': 0.0, 'left': 0.0, 'text': '', 'top': 0.0, 'width': 0.0},
           {'height': 0.0, 'left': 0.0, 'text': '', 'top': 0.0, 'width': 0.0}],
          [{'height': 4.699999809265137,
            'left': 128.93,
            'text': 'Name',
            'top': 582.16,
            'width': 30.030006408691406},
           {'height': 4.699999809265137,
            'left': 170.09,
            'text': '(1/km)',
            'top': 582.16,
            'width': 32.25000762939453},
           {'height': 4.699999809265137,
            'left': 222.11,
            'text': '(mi)',
            'top': 582.16,
            'width': 21.090003967285156},
           {'height': 4.699999809265137,
            'left': 262.92,
            'text': 'Improved',
            'top': 578.44,
            'width': 47.790000915527344},
           {'height': 4.699999809265137,
            'left': 318.78,
            'text': 'Decreased Eliminate',
            'top': 578.44,
            'width': 108.61998748779297},
           {'height': 4.699999809265137,
            'left': 435.42,
            'text': 'Decreased',
            'top': 578.44,
```


```
import tabula
pdf_path="foo.pdf"

tabula.convert_into(pdf_path, "test.json", output_format="json")
```

```
---------------------------------------------------------------------------
FileNotFoundError                         Traceback (most recent call last)
<ipython-input-12-7b717e0ae999> in <module>()
      2 pdf_path="foo.txt"
      3
----> 4 tabula.convert_into(pdf_path, "test.json", output_format="json")

/usr/local/lib/python3.7/dist-packages/tabula/io.py in convert_into(input_path,
output_path, output_format, java_options, **kwargs)
    552
    553     if not os.path.exists(path):
--> 554         raise FileNotFoundError(errno.ENOENT, os.strerror(errno.ENOENT), path)
    555
    556     if os.path.getsize(path) == 0:

FileNotFoundError: [Errno 2] No such file or directory: 'foo.txt'
```

SEARCH STACK OVERFLOW

```
#!pip install camelot-py[cv] tabula-py
#!sudo apt install ghostscript
import camelot
tables = camelot.read_pdf('foo.pdf')
tables.export('foo.csv', f='csv', compress=True) # json, excel, html, markdown, sqlite
#print(tables[0].parsing_report)
tables.export('foo.json', f='json', compress=True) # json, excel, html, markdown, sqlite
#tables[0].to_csv('foo.csv') # to_json, to_excel, to_html, to_markdown, to_sqlite
tables[0].df # get a pandas DataFrame!
```

	0	1	2	3	4	5	6
0	Cycle \nName	KI \n(1/km)	Distance \n(mi)	Percent Fuel Savings			
1				Improved \nSpeed	Decreased \nAccel	Eliminate \nStops	Decreased \nIdle

```
#!pip install camelot-py[cv] tabula-py
#!pip install ghostscript
#!pip install ghostscript
#!pip install camelot-py[cv]
#!pip install excalibur-py
#!excalibur initdb

import tabula
pdf_path = "foo.pdf"

dfs = tabula.read_pdf(pdf_path, stream=True)
# read_pdf returns list of DataFrames
print(len(dfs))
dfs[0]
tabula.read_pdf(pdf_path, pages="all", stream=True)
```

```
'pages' argument isn't specified.Will extract only from page 1 by default.
1
[  Unnamed: 0 Unnamed: 1 Unnamed: 2 Unnamed: 3 Percent Fuel Savings Unnamed: 4
 0      Cycle         KI   Distance        NaN                  NaN        NaN
 1       Name     (1/km)       (mi)   Improved  Decreased Eliminate  Decreased
 2        NaN        NaN        NaN      Speed          Accel Stops       Idle
 3     2012_2       3.30        1.3       5.9%           9.5% 29.2%      17.4%
 4     2145_1       0.68       11.2       2.4%            0.1% 9.5%       2.7%
 5     4234_1       0.59       58.7       8.5%            1.3% 8.5%       3.3%
 6     2032_2       0.17       57.8      21.7%            0.3% 2.7%       1.2%
 7     4171_1       0.07      173.9      58.1%            1.6% 2.1%       0.5%]
```

```
import tabula
pdf_path = "foo.pdf"
```

```
dfs = tabula.read_pdf(pdf_path, stream=True)
# read_pdf returns list of DataFrames
print(len(dfs))
dfs[0]
tabula.read_pdf(pdf_path, pages="all", stream=True)
```

```
#tabula.convert_into(pdf_path, "test.csv", output_format="csv", stream=True)
#tabula.convert_into(pdf_path, "test.json", output_format="json", stream=True)
dfs[0].to_html('Test.html')
print(dfs)
```

```
'pages' argument isn't specified.Will extract only from page 1 by default.
1
[  Unnamed: 0 Unnamed: 1 Unnamed: 2 Unnamed: 3 Percent Fuel Savings Unnamed: 4
0      Cycle         KI   Distance        NaN                  NaN        NaN
1       Name     (1/km)       (mi)   Improved  Decreased Eliminate  Decreased
2        NaN        NaN        NaN      Speed          Accel Stops       Idle
3     2012_2       3.30        1.3       5.9%           9.5% 29.2%      17.4%
4     2145_1       0.68       11.2       2.4%            0.1% 9.5%       2.7%
5     4234_1       0.59       58.7       8.5%            1.3% 8.5%       3.3%
6     2032_2       0.17       57.8      21.7%            0.3% 2.7%       1.2%
7     4171_1       0.07      173.9      58.1%            1.6% 2.1%       0.5%]
```

Double-click (or enter) to edit

```
import camelot
tables = camelot.read_pdf('foo.pdf')
tables.export('foo.csv', f='csv', compress=True) # json, excel, html, markdown, sqlite
#print(tables[0].parsing_report)
tables.export('foo.json', f='json', compress=True)
tables.export('foo.html', f='html', compress=True)

# json, excel, html, markdown, sqlite
```

```
# json, excel, html, markdown, sqlite
#tables[0].to_csv('foo.csv') # to_json, to_excel, to_html, to_markdown, to_sqlite
tables[0].df # get a pandas DataFrame!
```

	0	1	2	3	4	5	6
0	Cycle \nName	KI \n(1/km)	Distance \n(mi)	Percent Fuel Savings			
1				Improved \nSpeed	Decreased \nAccel	Eliminate \nStops	Decreased \nIdle
2	2012_2	3.30	1.3	5.9%	9.5%	29.2%	17.4%
3	2145_1	0.68	11.2	2.4%	0.1%	9.5%	2.7%
4	4234_1	0.59	58.7	8.5%	1.3%	8.5%	3.3%
5	2032_2	0.17	57.8	21.7%	0.3%	2.7%	1.2%

Does Camelot work with image-based PDFs? No, Camelot only works with text-based PDFs and not scanned documents. (As Tabula explains, “If you can click and drag to select text in your table in a PDF viewer, then your PDF is text-based”.)

```
import camelot
tables = camelot.read_pdf("leac204.pdf",pages="all")
#tables.export('foo.csv', f='csv', compress=True) # json, excel, html, markdown, sqlite
print(len((tables)))
#for i in (tables):
#  print(i.df)

#print(tables[0].parsing_report)
#tables.export('foo.json', f='json', compress=True)
#tables.export('foo.html', f='html', compress=True)

# json, excel, html, markdown, sqlite
#tables[0].to_csv('foo.csv') # to_json, to_excel, to_html, to_markdown, to_sqlite
tables[2].df # get a pandas DataFrame!
```

```
37
```

	0	1	2	3	
0	Particulars	2013-14	2014-15	Absolute\nIncrease (+) or\nDecrease (–)	Percentage\nIncrea (+)\nor Decrease (
1	I.\nRevenue from operations\nII. Add: Other in...	Rs.	Rs.	Rs.	
2		60,00,000\n1,50,000	75,00,000\n1,20,000	15,00,000\n(30,000)	25.00\n(20.0
3		44,00,000	61,50,000 76,20,000\n50,60,000	14,70,000\n6,60,000	23.90\n15.(
4		6,12,500	17,50,000 25,60,000\n10,24,000	8,10,000\n4,11,500	46.29\n67.

```
import tabula
pdf_path = "leac204.pdf"

dfs = tabula.read_pdf(pdf_path, stream=True,pages="all")
# read_pdf returns list of DataFrames
print(len(dfs))

#tabula.read_pdf(pdf_path, pages="all", stream=True)


#tabula.convert_into(pdf_path, "test.csv", output_format="csv", stream=True)
#tabula.convert_into(pdf_path, "test.json", output_format="json", stream=True)
#dfs[0].to_html('Test.html')
#print(dfs[0])
```

```
Got stderr: Aug 12, 2021 8:30:09 AM org.apache.pdfbox.pdmodel.font.PDSimpleFont toUnicode
WARNING: No Unicode mapping for .notdef (0) in font YGZXPE+EuclidSymbol
Aug 12, 2021 8:30:09 AM org.apache.pdfbox.rendering.Type1Glyph2D getPathForCharacterCode
WARNING: No glyph for code 0 (.notdef) in font YGZXPE+EuclidSymbol
```

```
41
```

```
import tabula
tab = tabula.read_pdf('foo.pdf', pages='all')
for t in tab:
    print(t, "\n=========================\n")
```

```
  Unnamed: 0 Unnamed: 1 Unnamed: 2 Unnamed: 3 Percent Fuel Savings Unnamed: 4
0      Cycle         KI   Distance        NaN                  NaN        NaN
1       Name     (1/km)       (mi)   Improved  Decreased Eliminate  Decreased
2        NaN        NaN        NaN      Speed          Accel Stops       Idle
3     2012_2       3.30        1.3       5.9%           9.5% 29.2%      17.4%
4     2145_1       0.68       11.2       2.4%            0.1% 9.5%       2.7%
5     4234_1       0.59       58.7       8.5%            1.3% 8.5%       3.3%
6     2032_2       0.17       57.8      21.7%            0.3% 2.7%       1.2%
7     4171_1       0.07      173.9      58.1%            1.6% 2.1%       0.5%
=========================
```

```
import camelot
tables = camelot.read_pdf('/content/01_MBAFT-Corporate-Finance-I.pdf', pages='all', split_text=True)
tables
for tabs in tables:
    print(tabs.df, "\n=================================\n")
```

```
                                                   0   1
0  Unit I: Finance Function and Finance Concepts ...  07
1  Unit II: Risk, Returns and Valuation of Securi...  07
2  Unit III: Investment Decision \nIntroduction t...  07
3  Unit IV: Capital Structure Decision  \n•  Cost...  06
4  Unit V: Dividend Decision \n•  Forms of divide...  03
=================================
```