

```
# install tabula python package
!pip install tabula.py
```

```
Collecting tabula.py
  Downloading tabula_py-2.9.0-py3-none-any.whl (12.0 MB)
    12.0/12.0 MB 40.3 MB/s eta 0:00:00
Requirement already satisfied: pandas>=0.25.3 in /usr/local/lib/python3.10/dist-packages (from tabula.py) (1.5.3)
Requirement already satisfied: numpy in /usr/local/lib/python3.10/dist-packages (from tabula.py) (1.25.2)
Requirement already satisfied: distro in /usr/lib/python3/dist-packages (from tabula.py) (1.7.0)
Requirement already satisfied: python-dateutil>=2.8.1 in /usr/local/lib/python3.10/dist-packages (from pandas>=0.25.3->tabula.py) (2.8.1)
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.10/dist-packages (from pandas>=0.25.3->tabula.py) (2023.4)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.10/dist-packages (from python-dateutil>=2.8.1->pandas>=0.25.3->tabula.py) (1.16.0)
Installing collected packages: tabula.py
Successfully installed tabula.py-2.9.0
```

```
!pip install tabulate
```

```
Requirement already satisfied: tabulate in /usr/local/lib/python3.10/dist-packages (0.9.0)
```

```
# import the necessary libraries
from tabula import read_pdf
from tabulate import tabulate
```

```
import warnings
# ignore all warnings
warnings.filterwarnings("ignore")
```

```
# filename variable of the pdf file which needs to be updated into the folder/environment
pdf_file = 'FoodList.pdf'
# extracted data from page 1 of the pf file
page_number = 1
# returns the extracted tables as pandas dataframes
tables_df = read_pdf(pdf_file, pages=page_number)
# print the tables from page 1 of the pdf
print(tables_df)
# ignore any warnings
```

```
WARNING:tabula.backend>Error importing jpype dependencies. Fallback to subprocess.
WARNING:tabula.backend:No module named 'jpype'
WARNING:tabula.backend:Got stderr: Apr 02, 2024 1:25:42 AM org.apache.pdfbox.pdmodel.font.FileSystemFontProvider loadDiskCache
WARNING: New fonts found, font cache will be re-built
Apr 02, 2024 1:25:42 AM org.apache.pdfbox.pdmodel.font.FileSystemFontProvider <init>
WARNING: Building on-disk font cache, this may take a while
Apr 02, 2024 1:25:42 AM org.apache.pdfbox.pdmodel.font.FileSystemFontProvider <init>
WARNING: Finished building on-disk font cache, found 17 fonts
```

	BREADS & CEREALS	Portion size *	...	Unnamed: 0	energy	content
0	Bagel ( 1 average )	140 cals (45g)	...	NaN		Medium
1	Biscuit digestives	86 cals (per biscuit)	...	NaN		High
2	Jaffa cake	48 cals (per biscuit)	...	NaN		Med-High
3	Bread white (thick slice)	96 cals (1 slice 40g)	...	NaN		Medium
4	Bread wholemeal (thick)	88 cals (1 slice 40g)	...	NaN		Low-med
5	Chapatis	250 cals	...	NaN		Medium
6	Cornflakes	130 cals (35g)	...	NaN		Med-High
7	Crackerbread	17 cals per slice	...	NaN		Low Calorie
8	Cream crackers	35 cals (per cracker)	...	NaN		Low / portion
9	Crumpets	93 cals (per crumpet)	...	NaN		Low-Med
10	Flapjacks basic fruit mix	320 cals	...	NaN		High
11	Macaroni (boiled)	238 cals (250g)	...	NaN		Low calorie
12	Muesli	195 cals (50g)	...	NaN		Med-high
13	Naan bread (normal)	300 cals (small plate size)	...	NaN		Medium
14	Noodles (boiled)	175 cals (250g)	...	NaN		Low calorie
15	Pasta ( normal boiled )	330 cals (300g)	...	NaN		Low calorie
16	Pasta (wholemeal boiled)	315 cals (300g)	...	NaN		Low calorie
17	Porridge oats (with water)	193 cals (350g)	...	NaN		Low calorie
18	Potatoes** (boiled)	210 cals (300g)	...	NaN		Low calorie
19	Potatoes** (roast)	420 cals (300g)	...	NaN		Medium

```
[20 rows x 5 columns]]
```

```
# use list comprehension to create a new list, loop through each dataframe, drops any columns that contain NaN(missing) values
cleaned_tables = [table.dropna(axis='columns') for table in tables_df]
# loop through the table and print everything, should not have any NaN values
for idx, table in enumerate(cleaned_tables):
    print(f"Table {idx+1} after dropping NaN values:")
    print(table)
```

Table 1 after dropping NaN values:

	BREADS & CEREALS	Portion size * per 100 grams (3.5 oz)	energy content
0	Bagel ( 1 average )	140 cal (45g)	310 cal Medium
1	Biscuit digestives	86 cal (per biscuit)	480 cal High
2	Jaffa cake	48 cal (per biscuit)	370 cal Med-High
3	Bread white (thick slice)	96 cal (1 slice 40g)	240 cal Medium
4	Bread wholemeal (thick)	88 cal (1 slice 40g)	220 cal Low-med
5	Chapatis	250 cal	300 cal Medium
6	Cornflakes	130 cal (35g)	370 cal Med-High
7	Crackerbread	17 cal per slice	325 cal Low Calorie
8	Cream crackers	35 cal (per cracker)	440 cal Low / portion
9	Crumpets	93 cal (per crumpet)	198 cal Low-Med
10	Flapjacks basic fruit mix	320 cal	500 cal High
11	Macaroni (boiled)	238 cal (250g)	95 cal Low calorie
12	Muesli	195 cal (50g)	390 cal Med-high
13	Naan bread (normal)	300 cal (small plate size)	320 cal Medium
14	Noodles (boiled)	175 cal (250g)	70 cal Low calorie
15	Pasta ( normal boiled )	330 cal (300g)	110 cal Low calorie
16	Pasta (wholemeal boiled)	315 cal (300g)	105 cal Low calorie
17	Porridge oats (with water)	193 cal (350g)	55 cal Low calorie
18	Potatoes** (boiled)	210 cal (300g)	70 cal Low calorie
19	Potatoes** (roast)	420 cal (300g)	140 cal Medium

```
# extract data from page 1 of the pdf file
page_number = 3
# returns the extracted tables as pandas dataframes
tables_df = read_pdf(pdf_file, pages=page_number)
# print the tables from page 1 of the pdf
print(tables_df)
```

[	Fish cake	90 cal per cake	200 cal	Medium
0	Fish fingers	50 cal per piece	220 cal	Medium
1	Gammon	320 cal	280 cal	Med-High
2	Haddock fresh	200 cal	110 cal	Low calorie
3	Halibut fresh	220 cal	125 cal	Low calorie
4	NaN	NaN	NaN	NaN
5	Ham	6 cal	240 cal	Medium
6	Herring fresh grilled	300 cal	200 cal	Medium
7	Kidney	200 cal	160 cal	Medium
8	Kipper	200 cal	120 cal	Low calorie
9	NaN	NaN	NaN	NaN
10	Liver	200 cal	150 cal	Medium
11	Liver pate	150 cal	300 cal	Medium
12	Lamb (roast)	300 cal	300 cal	Med-High
13	Lobster boiled	200 cal	100 cal	Low calorie
14	NaN	NaN	NaN	NaN
15	Luncheon meat	300 cal	400 cal	High
16	Mackerel	320 cal	300 cal	Medium
17	Mussels	90 cal	90 cal	Low-Med
18	Pheasant roast	200 cal	200 cal	Medium
19	Pilchards (tinned)	140 cal	140 cal	Medium
20	Prawns	180 cal	100 cal	Low- Med
21	Pork	320 cal	290 cal	Med-High
22	Pork pie	320 cal	450 cal	High
23	Rabbit	200 cal	180 cal	Medium
24	Salmon fresh	220 cal	180 cal	Medium
25	Sardines tinned in oil	220 cal	220 cal	Medium
26	Sardines in tomato sauce	180 cal	180 cal	Medium
27	Sausage pork fried	250 cal	320 cal	High
28	Sausage pork grilled	220 cal	280 cal	Med-High
29	Sausage roll	290 cal	480 cal	High
30	Scampi fried in oil	400 cal	340 cal	High
31	Steak & kidney pie	400 cal	350 cal	High]

```
# use list comprehension to convert the dataframe into a JSON string
tables_json = [table.to_json() for table in tables_df]
# loop over each JSON string to print data from the table
for idx,table_json in enumerate(tables_json):
    print(f"Table {idx + 1}:")
    print(table_json)
# add a space/newline between tables
print()
```

Table 1:

```
{ "Fish cake":{ "0": "Fish fingers", "1": "Gammon", "2": "Haddock fresh", "3": "Halibut fresh", "4": null, "5": "Ham", "6": "Herring fresh grilled", "7":
```

```
# extracted tables from all pages
```

```
tables = read_pdf(pdf_file, pages='all', multiple_tables=True)
```

```
# print the tables extracted from each page
```

```
print(tables)
```

10	Cream fresh clotted	480 cals	...	NaN	High
11	Custard	210 cals	...	NaN	Medium
12	Eggs ( 1 average size)	90 cals	...	NaN	Medium
13	Eggs fried	120 cals	...	NaN	Med-High
14	Fromage frais	125 cals	...	NaN	Low calorie
15	Ice cream	200 cals	...	NaN	Medium
16	Milk whole	175 cals (250ml/half pint)	...	NaN	Med-High
17	Milk semi-skimmed	125 cals (250ml/half pint)	...	NaN	Medium
18	Milk skimmed	95 cals (250ml/half pint)	...	NaN	Low calorie
19	Milk Soya	90 cals	...	NaN	Low calorie
20	Mousse flavored	120 cals	...	NaN	Medium
21	Omelette with cheese	300 cals	...	NaN	Medium
22	Trifle with cream	290 cals	...	NaN	Medium
23	Yogurt natural	90 cals	...	NaN	Low calorie
24	Yogurt reduced fat	70 cals	...	NaN	Low calorie

[25 rows x 5 columns],		Fats & Sugars		Portion size *	... Unnamed: 0 energy content
0	PURE FAT	9 cals (1 gram)	...	NaN	High
1	Bombay mix	250 cals	...	NaN	High
2	Butter	112 cals	...	NaN	High
3	Chewing gum	8 cals per piece	...	NaN	Low calorie
4	Chocolate	200 cals	...	NaN	High
5	Cod liver oil	135 cals (1 tbspoon)	...	NaN	High
6	Corn snack	125 cals	...	NaN	High
7	Crisps (chips US) average	100 cals	...	NaN	High
8	Honey	42 cals	...	NaN	Medium
9	Jam	38 cals	...	NaN	Medium
10	Lard	225 cals	...	NaN	High
11	Low fat spread	50 cals	...	NaN	High
12	Margarine	50 cals	...	NaN	High
13	Mars bar	240 cals	...	NaN	Med-High
14	Mint sweets	10 cals per piece	...	NaN	High
15	Oils -corn, sunflower, olive	135 cals (1 Tbspoon)	...	NaN	High
16	Popcorn average	150 cals	...	NaN	High
17	Sugar white table sugar	20 cals (1 tspoon)	...	NaN	Medium
18	Sweets (boiled)	100 cals	...	NaN	Med-High
19	Syrup	15 cals	...	NaN	Medium
20	Toffee	100 cals	...	NaN	High

[21 rows x 5 columns],		Fruit Calories per piece		Carbs (grams)	Water Content
0	Apple (1 average)	44 calories	10.5	85 %	
1	Apple cooking	35 calories	9	88 %	
2	Apricot	30 calories	6.7	85 %	
3	Avocado	150 calories	2	60 %	
4	Banana	107 calories	26	75 %	
5	Blackberries each	1 calorie	0.2	85 %	
6	Blackcurrant each	1.1 calorie	0.25	77 %	
7	Blueberries (new) 100g	49 Cals ( 100g )	15 g	81 %	
8	Cherry each	2.4 calories	0.6	83 %	
9	Clementine	24 cals	5	66 %	
10	Currants	5 calories	1.4	16 %	
11	Damson	28 calories	7.2	70 %	
12	One average date 5g	5 cals	1.2	14 %	
13	Dates with inverted sugar 100g	250 calories	63	12 %	
14	Figs	10 calories	2.4	24 %	
15	Gooseberries	2.6 calories	0.65	80 %	
16	Grapes 100g Seedless	50 cals	15	82 %	

```
# set flag to process information page by page, peformance optimizer
```

```
stream_option = True
```

```
# extract contents from page 4
```

```
page_number = 4
```

```
# extract tables in a rectangular area defined by coordinates(top, left, bottom, right)
```

```
area = (270, 13, 790, 900)
```

```
# extract from the specified area using the stream option
```

```
tables_df = read_pdf(pdf_file, pages=page_number, stream=stream_option, area=area)
```

```
#loop over the table, print the information
```

```
for idx, table in enumerate(tables_df):
```

```
    print(f"Table {idx + 1}:")
```

```
    print(table)
```

Table 1:

	Fruits & Vegetables	Portion size *	oz)	energy content
0	Apple	44 cal	44 calories	Low calorie
1	Banana	107 cal	65 calories	Low calorie
2	Beans baked beans	170 cal	80 calories	Low calorie
3	Beans dried (boiled)	180 cal	130 calories	Low calorie
4	Blackberries	25 cal	25 calories	Low calorie
5	Blackcurrant	30 cal	30 calories	Low calorie
6	Broccoli	27 cal	32 cal	Very low
7	Cabbage (boiled)	15 calories	20 calories	Low calorie
8	Carrot (boiled)	16 calories	25 calories	Low calorie
9	Cauliflower (boiled)	20 calories	30 calories	Low calorie
10	Celery (boiled)	5 calories	10 calories	Low calorie
11	Cherry	35 calories	50 calories	Low calorie
12	Courgette	8 cal	20 cal	Very low cal
13	Cucumber	3 calories	10 calories	Low calorie
14	Dates	100 calories	235 calories	Med-High
15	Grapes	55 calories	62 calories	Low calorie
16	Grapefruit	32 calories	32 calories	Low calorie
17	Kiwi	40 calories	50 calories	Low calorie
18	Leek (boiled)	10 calories	20 calories	Low calorie