Chapter 3 Data acquisition

CSS 341 Introduction to Data Science

Chukiat Worasucheep

Important Notice

การเรียนการสอนหัวข้อนี้ ผ่านทางสื่อออนไลน์ (Online meeting)

และมีการบันทึกภาพและเสียงเพื่อประโยชน์ทางการศึกษาต่อไปในอนาคต.

หากท่านไม่ยินยอมให้มีการเผยแพร่การบันทึกดังกล่าว ขอให้แจ้งให้ผู้สอนทราบภายใน 36 ชั่วโมง.

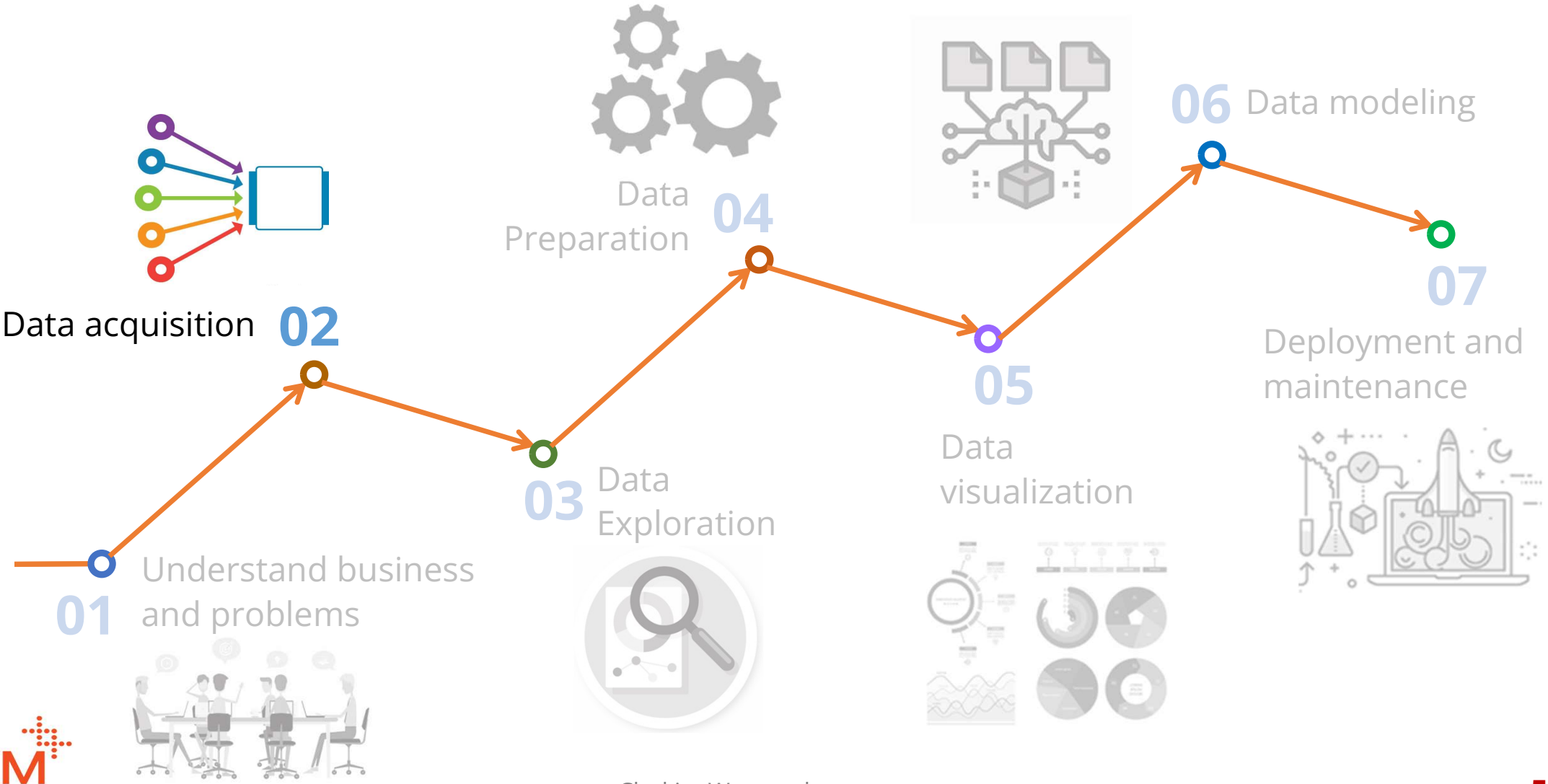
Learning objectives

- Understand and realize importance of data acquisition.
- Describe different common data sources and input methods.
- Practice different ways of read and write csv files and Excel files.
- Understand and practice basic methods of parsing web pages.

Contents

- What is data acquisition?
- Handle csv files
- Handle Excel files and tables in HTML files
- Extract web page with BeautifulSoup

Data science process



What is data acquisition?

- *Data acquisition* is all about obtaining the input data from a variety of sources, which follows by extracting the useful information and converting it into representations suitable for further processing.

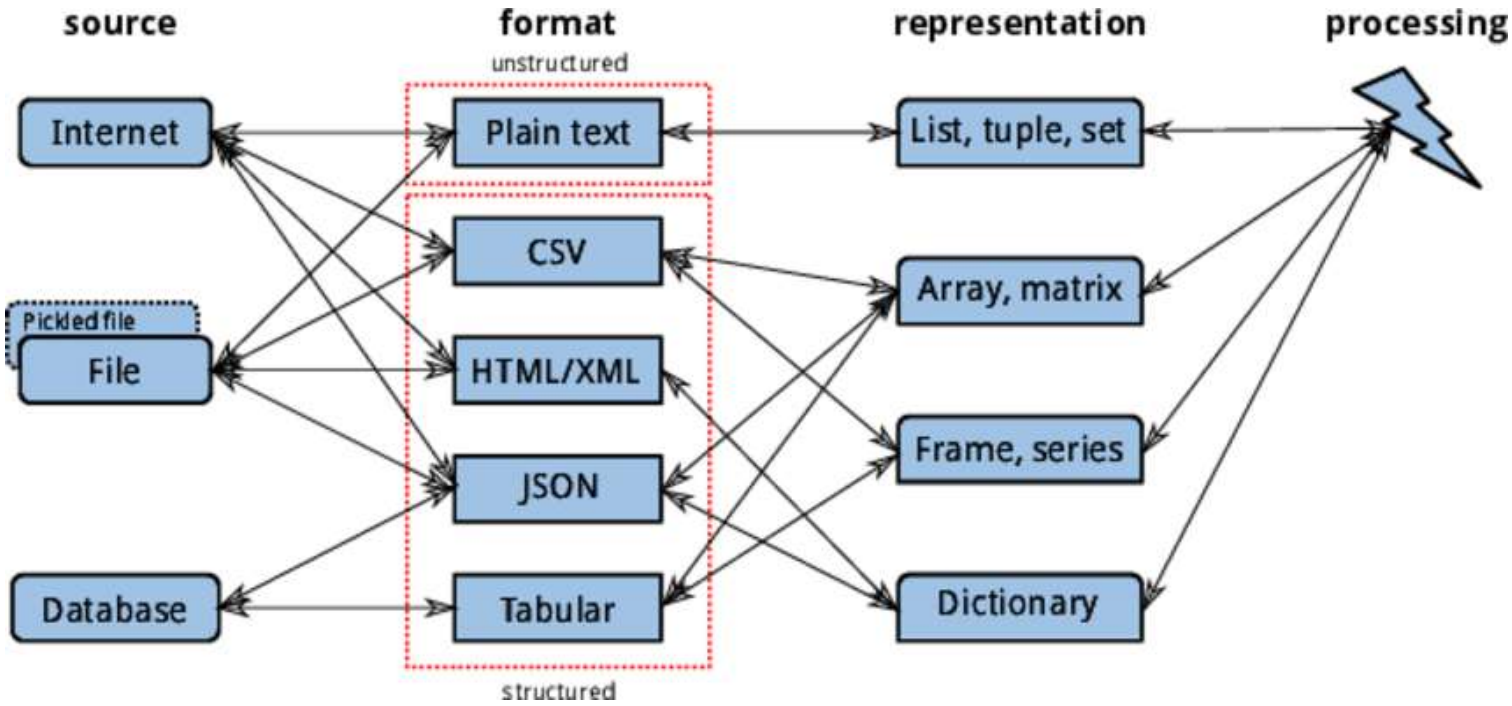


Image source: https://medium.com/pragmatic-programmers/data-acquisition-pipeline-8d58d4ec1944

Contents

- What is data acquisition?
- Handle csv files
- Handle Excel files and tables in HTML files
- Extract web page with BeautifulSoup

Comma-Separated Values (CSV) files

$ pip install pandas

```
import pandas as pd
df = pd.read_csv('titanic.csv')
df
```

```
In [2]: import pandas as pd

In [3]: df = pd.read_csv('titanic.csv')
        df
```

Out[3]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S
...	...	...	...	...	...	...	...	...	...	...	...	...
886	887	0	2	Montvila, Rev. Juozas	male	27.0	0	0	211536	13.0000	NaN	S
887	888	1	1	Graham, Miss. Margaret Edith	female	19.0	0	0	112053	30.0000	B42	S
888	889	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	NaN	1	2	W./C. 6607	23.4500	NaN	S
889	890	1	1	Behr, Mr. Karl Howell	male	26.0	0	0	111369	30.0000	C148	C
890	891	0	3	Dooley, Mr. Patrick	male	32.0	0	0	370376	7.7500	NaN	Q

891 rows × 12 columns

pandas library

pandas

Getting started User Guide API reference Development Release notes

Input/output

- pandas.read_pickle
- pandas.DataFrame.to_pickle
- pandas.read_table
- pandas.read_csv
- pandas.DataFrame.to_csv
- pandas.read_fwf
- pandas.read_clipboard
- pandas.DataFrame.to_clipboard
- pandas.read_excel
- pandas.DataFrame.to_excel
- pandas.ExcelFile.parse
- pandas.io.formats.style.Styler.to_excel
- pandas.ExcelWriter
- pandas.read_json
- pandas.json_normalize
- pandas.DataFrame.to_json
- pandas.io.json.build_table_schema

Reference: https://pandas.pydata.org/docs/reference/api/pandas.read_csv.html

pandas.read_csv

```
pandas.read_csv(filepath_or_buffer, sep=NoDefault.no_default, delimiter=None,
header='infer', names=NoDefault.no_default, index_col=None, usecols=None, squeeze=None,
prefix=NoDefault.no_default, mangle_dupe_cols=True, dtype=None, engine=None,
converters=None, true_values=None, false_values=None, skipinitialspace=False,
skiprows=None, skipfooter=0, nrows=None, na_values=None, keep_default_na=True,
na_filter=True, verbose=False, skip_blank_lines=True, parse_dates=None,
infer_datetime_format=False, keep_date_col=False, date_parser=None, dayfirst=False,
cache_dates=True, iterator=False, chunksize=None, compression='infer', thousands=None,
decimal='.', lineterminator=None, quotechar='"', quoting=0, doublequote=True,
escapechar=None, comment=None, encoding=None, encoding_errors='strict', dialect=None,
error_bad_lines=None, warn_bad_lines=None, on_bad_lines=None, delim_whitespace=False,
low_memory=True, memory_map=False, float_precision=None, storage_options=None)
```
[source]

Read a comma-separated values (csv) file into DataFrame.

Also supports optionally iterating or breaking of the file into chunks.

Additional help can be found in the online docs for IO Tools.

Most commonly-used parameters of read_csv()

- sep ,
- header 0
- usecols None (ทุกคอลัมน์)
- nrows None (ทั้งหมด)
- index_col None (ไม่ใช้)
- encoding None (more details here)

Some read_csv()'s options

In [3]:
```
df = pd.read_csv('titanic.csv', usecols=[0, 1, 2, 4, 5])
df
```

Out[3]:

	PassengerId	Survived	Pclass	Sex	Age
0	1	0	3	male	22.0
1	2	1	1	female	38.0
2	3	1	3	female	26.0
3	4	1	1	female	35.0
4	5	0	3	male	35.0
...	...	...	...	...	...
886	887				
887	888				
888	889				
889	890				
890	891				

891 rows × 5 columns

In [4]:
```
df = pd.read_csv('titanic.csv',
                 usecols=[0, 1, 2, 4, 5],
                 header=2, nrows=4)
df
```

Out[4]:

	2	1	1.1	female	38
0	3	1	3	female	26.0
1	4	1	1	female	35.0
2	5	0	3	male	35.0
3	6	0	3	male	NaN

All columns

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S
...	...	...	...	...	...	...	...	...	...	...	...	...

read_csv()'s index_col option

In [7]:
```
df = pd.read_csv('titanic.csv', index_col=0)
df
```

Out[7]:

PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Tick
1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 2117
2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 1759
3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 310128
4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	1138(
5	0	3	Allen, Mr. William Henry	male	35.0	0	0	37345
...	...	...	...	...	...	...	...	
887	0	2	Montvila, Rev. Juozas	male	27.0	0	0	21153
888	1	1	Graham, Miss. Margaret Edith	female	19.0	0	0	11205
889	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	NaN	1	2	W./C. 66(
890	1	1	Behr, Mr. Karl Howell	male	26.0	0	0	11136
891	0	3	Dooley, Mr. Patrick	male	32.0	0	0	37037

891 rows × 11 columns

head() and tail()

In [9]:
```
df.head()
```

Out[9]:

PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85	C
3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S
5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S

In [10]:
```
df.tail(3)
```

Out[10]:

PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
889	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	NaN	1	2	W./C. 6607	23.45	NaN	S
890	1	1	Behr, Mr. Karl Howell	male	26.0	0	0	111369	30.00	C148	C
891	0	3	Dooley, Mr. Patrick	male	32.0	0	0	370376	7.75	NaN	Q

pandas's to_csv() function

pandas.DataFrame.to_csv

```
DataFrame.to_csv(path_or_buf=None, sep=',', na_rep='', float_format=None, columns=None,
header=True, index=True, index_label=None, mode='w', encoding=None, compression='infer',
quoting=None, quotechar='"', line_terminator=None, chunksize=None, date_format=None,
doublequote=True, escapechar=None, decimal='.', errors='strict', storage_options=None)
```

Write object to a comma-separated values (csv) file. [source]

```
df.to_csv('new.csv')
```

File Edit Selection Find View Goto Tools Project Preferences Help

new.csv

```
PassengerId,Survived,Pclass,Name,Sex,Age,SibSp,Parch,Ticket,Fare,Cabin,Embarked
1,0,3,"Braund, Mr. Owen Harris",male,22.0,1,0,A/5 21171,7.25,,S
2,1,1,"Cumings, Mrs. John Bradley (Florence Briggs Thayer)",female,38.0,1,0,PC 1759
3,1,3,"Heikkinen, Miss. Laina",female,26.0,0,0,STON/O2. 3101282,7.925,,S
4,1,1,"Futrelle, Mrs. Jacques Heath (Lily May Peel)",female,35.0,1,0,113803,53.1,C1
5,0,3,"Allen, Mr. William Henry",male,35.0,0,0,
6,0,3,"Moran, Mr. James",male,,0,0,330877,8.458
7,0,1,"McCarthy, Mr. Timothy J",male,54.0,0,0,1
8,0,3,"Palsson, Master. Gosta Leonard",male,2.0
9,1,3,"Johnson, Mrs. Oscar W (Elisabeth Vilhelm
10,1,2,"Nasser, Mrs. Nicholas (Adele Achem)",fe
11,1,3,"Sandstrom, Miss. Marguerite Rut",female
12,1,1,"Bonnell, Miss. Elizabeth",female,58.0,0
13,0,3,"Saundercock, Mr. William Henry",male,20
14,0,3,"Andersson, Mr. Anders Johan",male,39.0,
15,0,3,"Vestrom, Miss. Hulda Amanda Adolfina",female,14,0,0,350406,7.8542,,S
```

	A	B	C	D	E	F	G	H	I	J	K	L
1	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
2	1	0	3	Braund, Mr. Owen Harri	male	22	1	0	A/5 21171	7.25		S
3	2	1	1	Cumings, Mrs. John Bra	female	38	1	0	PC 17599	71.2833	C85	C
4	3	1	3	Heikkinen, Miss. Laina	female	26	0	0	STON/O2.	7.925		S
5	4	1	1	Futrelle, Mrs. Jacques H	female	35	1	0	113803	53.1	C123	S
6	5	0	3	Allen, Mr. William Henr	male	35	0	0	373450	8.05		S
7	6	0	3	Moran, Mr. James	male		0	0	330877	8.4583		Q
8	7	0	1	McCarthy, Mr. Timothy	male	54	0	0	17463	51.8625	E46	S

Some parameters of to_csv()

```
df.to_csv('new.csv',
  columns=['Survived', 'Age', 'Sex'])
```

	A	B	C	D	E
1	PassengerId	Survived	Age	Sex	
2	1	0	22	male	
3	2	1	38	female	
4	3	1	26	female	
5	4	1	35	female	
6	5	0	35	male	
7	6	0		male	
8	7	0	54	male	
9	8	0	2	male	
10	9	1	27	female	
11	10	1	14	female	
12	11	1	4	female	
13	12	1	58	female	

```
df.to_csv('new.csv', sep=';',
  columns=['Survived', 'Age', 'Sex'])
```

	A	B	C
1	PassengerId;Survived;Age;Sex		
2	1;0;22.0;male		
3	2;1;38.0;female		
4	3;1;26.0;female		
5	4;1;35.0;female		
6	5;0;35.0;male		
7	6;0;;male		
8	7;0;54.0;male		
9	8;0;2.0;male		
10	9;1;27.0;female		
11	10;1;14.0;female		
12	11;1;4.0;female		

Contents

- What is data acquisition?
- Handle csv files
- Handle Excel files and tables in HTML files
- Extract web page with BeautifulSoup

Pandas IO tools (ref: here)

pandas

Getting started | User Guide | API reference | Development | Release notes

Search the docs ...

IO tools (text, CSV, HDF5, ...)

The pandas I/O API is a set of top level `reader` functions accessed like `pandas.read_csv()` that generally return a pandas object. The corresponding `writer` functions are object methods that are accessed like `DataFrame.to_csv()`. Below is a table containing available `readers` and `writers`.

Format Type	Data Description	Reader	Writer
text	CSV	read_csv	to_csv
text	Fixed-Width Text File	read_fwf	
text	JSON	read_json	to_json
text	HTML	read_html	to_html
text	LaTeX		Styler.to_latex
text	XML	read_xml	to_xml
text	Local clipboard	read_clipboard	to_clipboard
binary	MS Excel	read_excel	to_excel
binary	OpenDocument	read_excel	
binary	HDF5 Format	read_hdf	to_hdf
binary	Feather Format	read_feather	to_feather

Pandas with Excel files

pandas.read_excel

```
pandas.read_excel(io, sheet_name=0, header=0, names=None, index_col=None, usecols=None,
squeeze=None, dtype=None, engine=None, converters=None, true_values=None,
false_values=None, skiprows=None, nrows=None, na_values=None, keep_default_na=True,
na_filter=True, verbose=False, parse_dates=False, date_parser=None, thousands=None,
decimal='.', comment=None, skipfooter=0, convert_float=None, mangle_dupe_cols=True,
storage_options=None)
```
[source]

Read an Excel file into a pandas DataFrame.

Supports *xls*, *xlsx*, *xlsm*, *xlsb*, *odf*, *ods* and *odt* file extensions read from a local filesystem or URL. Supports an option to read a single sheet or a list of sheets.

pandas.ExcelWriter

```
class pandas.ExcelWriter(path, engine=None, date_format=None, datetime_format=None,
mode='w', storage_options=None, if_sheet_exists=None, engine_kwargs=None, **kwargs)
```
[source]

Class for writing DataFrame objects into excel sheets.

Default is to use : * xlwt for xls * xlsxwriter for xlsx if xlsxwriter is installed otherwise openpyxl * odf for ods. See DataFrame.to_excel for typical usage.

The writer should be used as a context manager. Otherwise, call *close()* to save and close any opened file handles.

```
import pandas as pd

# Read excel file with sheet names,
# returns Dict of DataFrame
dict_df = pd.read_excel('d:/data/schedule.xlsx',
                        sheet_name=['Tech', 'Social'])
```

Example of read_excel

Example of ExcelWriter

Pandas's read_html()

Read *tables* from a HTML link (URL) into a list of DataFrame objects.

pandas.read_html

```
pandas.read_html(io, match='.+', flavor=None, header=None, index_col=None, skiprows=None,
attrs=None, parse_dates=False, thousands=',', encoding=None, decimal='.', converters=None,
na_values=None, keep_default_na=True, displayed_only=True)                    [source]
```

Read HTML tables into a `list` of `DataFrame` objects.

```
import pandas as pd
df = pd.read_html('https://en.wikipedia.org/wiki/Wonders_of_the_World')
type(df), len(df)
for i in range(len(df)):
    print('\n***************', i, '***************')
    print(df[i].head(10))
```

en.wikipedia.org/wiki/Wonders_of_the_World

Recent lists

Following in the tradition of the classical list, modern people and organisations have made their own lists of wonderful things, both ancient and modern, natural and artificial. Some of the most notable lists are presented below.

American Society of Civil Engineers

In 1994, the American Society of Civil Engineers compiled a list of Seven Wonders of the Modern World, paying tribute to the "greatest civil engineering achievements of the 20th century".[12][13]

American Society of Civil Engineers Wonders

Wonder	Date started	Date finished	Location	Significance
Channel Tunnel	December 1, 1987	May 6, 1994	Strait of Dover, in the English Channel between the United Kingdom and France	Longest undersea portion of any tunnel in the world
CN Tower	February 6, 1973	June 26, 1976	Toronto, Ontario, Canada	Tallest freestanding structure in the world from 1976 to 2007
Empire State Building	March 17, 1930	April 11, 1931	New York City, New York, United States	Tallest structure in the world from 1931 to 1954; tallest freestanding structure in the world from 1931 to 1967; tallest building in the world from 1931 to 1970; first building with 100+ stories
Golden Gate Bridge	January 5, 1933	May 27, 1937	Golden Gate Strait, north of San Francisco, California, United States	Longest main span of any suspension bridge in the world from 1937 to 1964
Itaipú Dam	January 1970	May 5, 1984	Paraná River, on the border between Brazil and Paraguay	Largest operating hydroelectric facility in the world in terms of annual energy generation[14]

USA Today's New Seven Wonders

In November 2006, the American national newspaper *USA Today* and the American television show *Good Morning Am* revealed a list of the "New Seven Wonders", both natural and man-made, as chosen by six judges.[15] The Grand Can was added as an eighth wonder on November 24, 2006, in response to viewer feedback.[16]

***USA Today*'s New Seven Wonders**

Wonder	Location
Potala Palace	Lhasa, Tibet
Old City of Jerusalem	Israel [n 1]
Polar ice caps	Earth's polar regions (Arctic and Antarctic)
Papahānaumokuākea Marine National Monument	Hawaii, United States
Internet	
Mayan ruins	Yucatán Peninsula, México
Great Migration of Serengeti and Masai Mara	Tanzania and Kenya
Grand Canyon (viewer-chosen eighth wonder)	Arizona, United States

pandas.read_html(Seven Wonders of the World)

```
In [11]: URL = 'https://en.wikipedia.org/wiki/Wonders_of_the_World'
         dfs = pd.read_html(URL)
         type(dfs), len(dfs)

Out[11]: (list, 4)

In [12]: for i in range(len(dfs)):
             print('\n***************', i, '***************')
             print(dfs[i].head(10))

*************** 0 ***************
                      Wonder      Date started    Date finished  \
0             Channel Tunnel  December 1, 1987      May 6, 1994
1                  CN Tower  February 6, 1973    June 26, 1976
2      Empire State Building    March 17, 1930   April 11, 1931
3         Golden Gate Bridge   January 5, 1933     May 27, 1937
4                 Itaipú Dam      January 1970      May 5, 1984
5  Delta and Zuiderzee Works              1920     May 10, 1997
6              Panama Canal   January 1, 1880  January 7, 1914

                                          Location  \
0  Strait of Dover, in the English Channel betwee...
1                          Toronto, Ontario, Canada
2            New York City, New York, United States
3  Golden Gate Strait, north of San Francisco, Ca...
4  Paraná River, on the border between Brazil and...
5  Zeeland, South Holland, North Holland, Friesla...
6                                Isthmus of Panama

                                     Significance
0  Longest undersea portion of any tunnel in the ...
1  Tallest freestanding structure in the world fr...
```

```
                                     Significance
0  Longest undersea portion of any tunnel in the ...
1  Tallest freestanding structure in the world fr...
2  Tallest structure in the world from 1931 to 19...
3  Longest main span of any suspension bridge in ...
4  Largest operating hydroelectric facility in th...
5  Largest hydraulic engineering project undertak...
6  Allows passage of oceangoing vessels between t...

*************** 1 ***************
                                      Wonder  \
0                               Potala Palace
1                        Old City of Jerusalem
2                                Polar ice caps
3  Papahānaumokuākea Marine National Monument
4                                    Internet
5                                 Mayan ruins
6  Great Migration of Serengeti and Masai Mara
7   Grand Canyon (viewer-chosen eighth wonder)

                                     Location
0                                Lhasa, Tibet
1                                Israel [n 1]
2  Earth's polar regions (Arctic and Antarctic)
3                        Hawaii, United States
4                                          NaN
5                    Yucatán Peninsula, México
6                           Tanzania and Kenya
7                        Arizona, United States

*************** 2 ***************
                              Wonder                  Date of construction  \
0                  Great Wall of China                Since 7th century BC[21]
1                                Petra                                c. 100 BC
2                  Christ the Redeemer  opened to the public October 12, 1931
3                          Machu Picchu                                c. AD 1450
```

Pros and cons of pandas.read_html()

- Pro
  - Easy and straightforward
- Con
  - Only tables in a HTML file.

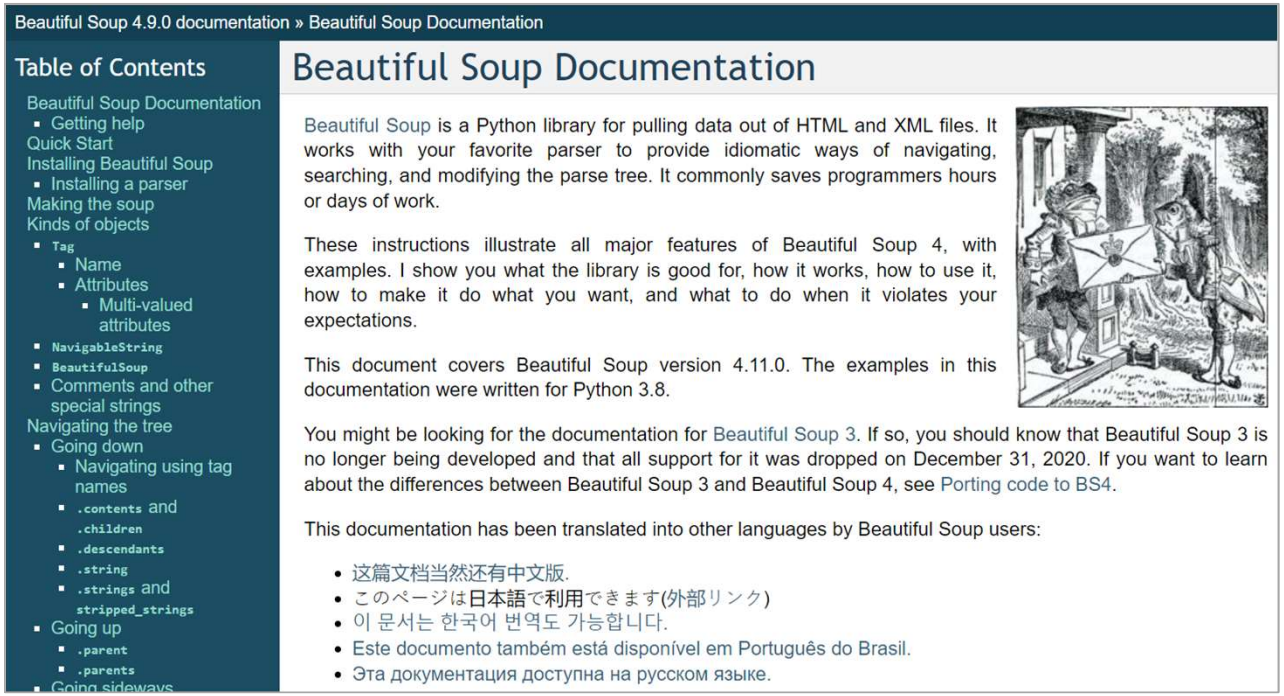
Contents

- What is data acquisition?
- Handle csv files
- Handle Excel files and tables in HTML files
- Extract web page with BeautifulSoup

BeautifulSoup

- BeautifulSoup is a Python library for pulling data out of HTML and XML files.
- Helps: https://www.crummy.com/software/BeautifulSoup/bs4/doc/

- Install: pip install bs4, request
- Benefits:
  - Easy to use
  - Best for beginners
  - Good for quick projects.



Beautiful Soup 4.9.0 documentation » Beautiful Soup Documentation

Table of Contents

Beautiful Soup Documentation

Beautiful Soup is a Python library for pulling data out of HTML and XML files. It works with your favorite parser to provide idiomatic ways of navigating, searching, and modifying the parse tree. It commonly saves programmers hours or days of work.

These instructions illustrate all major features of Beautiful Soup 4, with examples. I show you what the library is good for, how it works, how to use it, how to make it do what you want, and what to do when it violates your expectations.

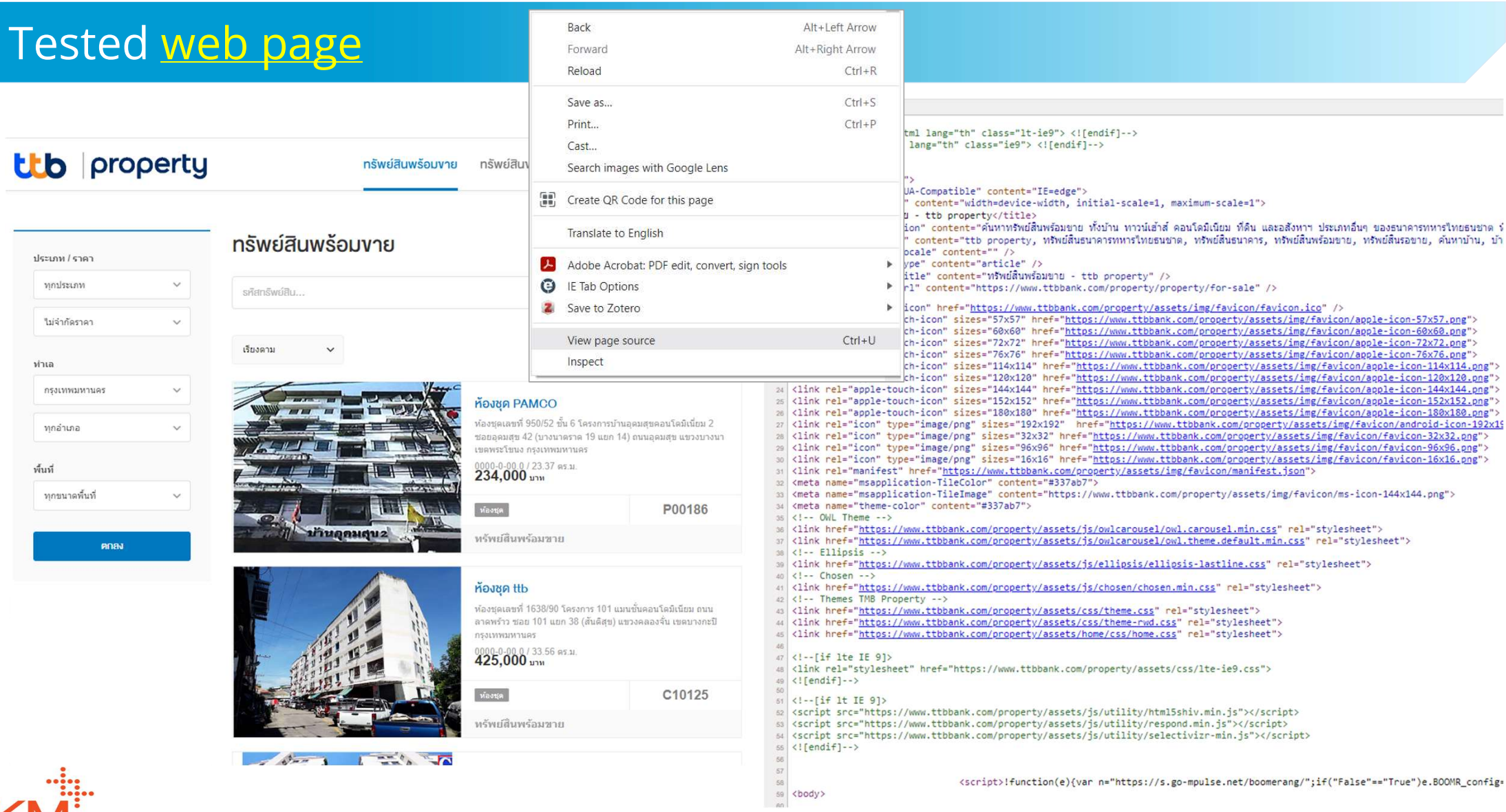
This document covers Beautiful Soup version 4.11.0. The examples in this documentation were written for Python 3.8.

You might be looking for the documentation for Beautiful Soup 3. If so, you should know that Beautiful Soup 3 is no longer being developed and that all support for it was dropped on December 31, 2020. If you want to learn about the differences between Beautiful Soup 3 and Beautiful Soup 4, see Porting code to BS4.

This documentation has been translated into other languages by Beautiful Soup users:

- 这篇文档当然还有中文版.
- このページは日本語で利用できます(外部リンク)
- 이 문서는 한국어 번역도 가능합니다.
- Este documento também está disponível em Português do Brasil.
- Эта документация доступна на русском языке.

Tested web page



Division landind-item

```
137
138 <div class="clearall"></div>
139
140                         <div class="landind-items">
141                             <div class="col-list-items">
142         <div class="thumb-items list-items ">
143         <div class="entry-img">
144             <a href="https://www.ttbbank.com/property/property/detail/P00186">
145                 <img src="https://media.tmbbank.com/uploads/npa/product/img/3746_20220512114823.jpg" alt="">
146                                                                                                    </a>
147         </div>
148         <div class="box-inr">
149             <div class="entry-caption">
150                 <a href="https://www.ttbbank.com/property/property/detail/P00186"><h3 data-ellipsis-lastline>ห้องชุด  PAMCO</h3></a>
151                 <div class="box-group">
152                     <p class="entry-desc">ห้องชุดเลขที่ 950/52 ชั้น 6 โครงการบ้านอุดมสุขคอนโดมิเนียม 2 ซอยอุดมสุข 42 (บางนาตราด 19 แยก 14) ถนนอุดมสุข แขวงบางนา
153                     <p class="entry-area">0000-0-00.0 / 23.37 ตร.ม.</p>
154                 </div>
155                 <p class="entry-price">234,000<span>บาท</span></p>
156             </div>
157             <div class="box-bottom">
158                 <div class="entry-2col">
159                     <div class="col-inline">
```


```
om/property/property/for-sale?type=3" class="entry-tags">ห้องชุด</a>
                <div class="col-inline entry-code">P00186</div>

roperty/property/for-sale" class="entry-category">ทรัพย์สินพร้อมขาย </a>

tems">

/property/detail/C10125">
oloads/npa/product/img/3013_20210621171407.jpg" alt="">
                                                                                                    </a>
```

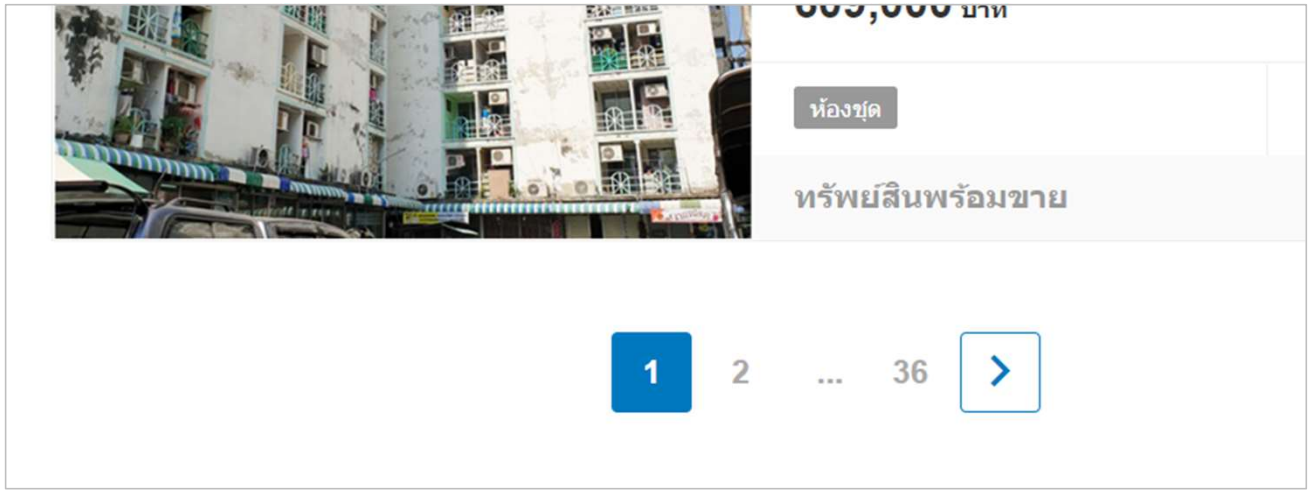
```
179         <div class="box-inr">
180             <div class="entry-caption">
```

Components of each land item

```
                <div class="landind-items">
                    <div class="col-list-items">
    <div class="thumb-items list-items ">
    <div class="entry-img">
        <a href="https://www.ttbbank.com/property/property/detail/P00186">
            <img src="https://media.tmbbank.com/uploads/npa/product/img/3746_20220512114823.jpg" alt="">
                                                                                                    </a>
    </div>
    <div class="box-inr">
        <div class="entry-caption">
            <a href="https://www.ttbbank.com/property/property/detail/P00186"><h3 data-ellipsis-lastline>ห้องชุด  PAMCO</h3></a>
            <div class="box-group">
                <p class="entry-desc">ห้องชุดเลขที่ 950/52 ชั้น 6 โครงการบ้านอุดมสุขคอนโดมิเนี่ยม 2 ซอยอุดมสุข 42 (บางนาตราด 19 แยก 14) ถนนอุดมสุข แขวงบางนา เขตพระโขนง
                <p class="entry-area">0000-0-00.0 / 23.37 ตร.ม.</p>
            </div>
            <p class="entry-price">234,000<span>บาท</span></p>
        </div>
        <div class="box-bottom">
            <div class="entry-2col">
                <div class="col-inline">
                    <a href="https://www.ttbbank.com/property/property/for-sale?type=3" class="entry-tags">ห้องชุด</a>
                </div>
                                                <div class="col-inline entry-code">P00186</div>

            </div>
            <div class="entry-bottom">
                <a href="https://www.ttbbank.com/property/property/for-sale" class="entry-category">ทรัพย์สินพร้อมขาย </a>
            </div>
        </div>
    </div>
</div>
</div>
                        <div class="col-list-items">
    <div class="thumb-items list-items ">
    <div class="entry-img">
        <a href="https://www.ttbbank.com/property/property/detail/C10125">
            <img src="https://media.tmbbank.com/uploads/npa/product/img/3013_20210621171407.jpg" alt="">
                                                                                                    </a>
```

Pagination - Find links to next/previous pages




```
                                    </div>
                                </div>
                            </div>
                        </div>
                        </div>
                                                    </div>                                                    <div class="pagination-container">
                    <ul class="pagination"><li class="disabled"><span class="disabled"><i class="ic ic-arrow-left"></i></span></li><li class="active"><
                                        </div>
            </div>
            <div id="filter-sidebar" class="sidebar">
                <div class="modal-sidebar">
                    <div class="sidebar-content sidebar-filter">
                        <form action="https://www.ttbbank.com/property/property/for-sale" method="get" class="fieldset-group">
```

Scrapy vs BeautifulSoup

- Scrapy
  - An open-source framework
  - Somewhat faster
  - Better for larger projects
  - More complex, good for web scraping
- BeautifulSoup
  - A Python library
  - Somewhat slower
  - Best for smaller projects
  - Good for beginners, for parsing a web page

Important Notices

- *Before* web scraping, you should *always check* your target website's *acceptable use policy* to see if accessing the website with automated tools is a violation of its terms of use.
- Legally, web scraping against the wishes of a website is very much a gray area.
- It may be illegal when used on websites that prohibit web scraping.

Contents

- What is data acquisition?
- Handle csv files
- Handle Excel files and tables in HTML files
- Extract web page with BeautifulSoup