

pandas

October 9, 2019

1 Dataframes

1.1 Pandas

A Data frame is a two-dimensional data structure, i.e., data is aligned in a tabular fashion in rows and columns.

Features of DataFrame * Potentially columns are of different types * Size – Mutable * Labeled axes (rows and columns) * Can Perform Arithmetic operations on rows and columns

For exploration purposes, we will use the CMBX constituent data from Markit1: <https://www.markit.com/NewsInformation/GetNews/CMBX>

Import the pandas library to use dataframes. The "as pd" is a useful convention for having the library in the namespace.

```
[3]: import pandas as pd
```

Creating an empty dataframe:

```
[6]: df = pd.DataFrame()  
print(df)
```

```
Empty DataFrame  
Columns: []  
Index: []
```

Creating an example dataframe from list of lists:

```
[11]: data1 = [['Alex',10],['Bob',12],['Charlie',13]]  
df1 = pd.DataFrame(data1,columns=['Name','Age'])  
print(df1)
```

```
      Name  Age  
0    Alex   10  
1     Bob   12  
2  Charlie   13
```

Creating an example dataframe from a dictionary:

```
[12]: data2 = {'Name': ['Tom', 'Jack', 'Steve', 'Ricky'], 'Age': [28, 34, 29, 42]}
      df2 = pd.DataFrame(data2)
      print(df2)
```

```
      Name  Age
0     Tom   28
1    Jack   34
2   Steve   29
3   Ricky   42
```

Column selection:

```
[13]: df2['Name']
```

```
[13]: 0     Tom
      1    Jack
      2   Steve
      3   Ricky
      Name: Name, dtype: object
```

```
[21]: df2[['Name']]
```

```
[21]:      Name
0     Tom
1    Jack
2   Steve
3   Ricky
```

Adding a new column:

```
[18]: df2['Country'] = pd.Series(['USA', 'UK', 'Australia', 'USA'])
      print(df2)
```

```
      Name  Age  Country
0     Tom   28     USA
1    Jack   34      UK
2   Steve   29  Australia
3   Ricky   42     USA
```

Selecting multiple columns, list of columns, as a dataframe:

```
[20]: df2[['Country', 'Age']]
```

```
[20]:      Country  Age
0         USA    28
1         UK     34
2  Australia    29
3         USA    42
```

Selecting rows:

```
[24]: df2.loc[0]
```

```
[24]: Name      Tom
      Age       28
      Country   USA
      Name: 0, dtype: object
```

```
[28]: df2['Country']=='USA'
```

```
[28]: 0      True
      1     False
      2     False
      3      True
      Name: Country, dtype: bool
```

```
[26]: df2[df2['Country']=='USA']
```

```
[26]:   Name  Age Country
      0   Tom   28     USA
      3  Ricky  42     USA
```

```
[27]: df2[df2['Age']<30]
```

```
[27]:   Name  Age  Country
      0   Tom   28     USA
      2  Steve  29  Australia
```

```
[34]: df2[(df2['Age']<30) & (df2['Country']=='USA')]
```

```
[34]:   Name  Age Country
      0   Tom   28     USA
```

```
[35]: df2[(df2['Age']<30) | (df2['Country']=='USA')]
```

```
[35]:   Name  Age  Country
      0   Tom   28     USA
      2  Steve  29  Australia
      3  Ricky  42     USA
```

Examples using the CMBX data: Loading a csv (comma separated values) file requires no further package loads. Reading an excel file will require the install of xlrd.

```
[42]: df = pd.read_csv('CMBX12.csv')
      df
```

```
[42]:   Deal Name Deal Type  Closing Date  Vintage  Cutoff Balance \
      0  ms18bn11   Conduit    20180426    2018    6.882007e+08
      1  wf18bn10   Conduit    20180213    2018    1.287149e+09
```

2	jpm18c8	Conduit	20180615	2018	7.131377e+08
3	gs18gs9	Conduit	20180329	2018	8.871306e+08
4	ubs18c15	Conduit	20181228	2018	6.464773e+08
5	com18co3	Conduit	20180522	2018	1.006083e+09
6	wf18bn13	Conduit	20180802	2018	9.441900e+08
7	ubs18c11	Conduit	20180710	2018	8.038161e+08
8	ba18bn12	Conduit	20180530	2018	9.011700e+08
9	ms18h4	Conduit	20181227	2018	7.968060e+08
10	cs18cx11	Conduit	20180418	2018	9.528688e+08
11	bmar18b5	Conduit	20180821	2018	1.039132e+09
12	bmar18b4	Conduit	20180713	2018	1.158480e+09
13	ms18h3	Conduit	20180712	2018	1.024238e+09
14	ubs18c8	Conduit	20180227	2018	1.045168e+09
15	gs18gs10	Conduit	20180730	2018	8.738348e+08
16	bmar18b8	Conduit	20181227	2018	1.049017e+09
17	bb18c2	Conduit	20181220	2018	8.918536e+08
18	cs18c14	Conduit	20181129	2018	7.702287e+08
19	ubs18c14	Conduit	20181212	2018	6.508850e+08
20	db18c1	Conduit	20181030	2018	1.100682e+09
21	bmar18b6	Conduit	20181009	2018	1.147030e+09
22	cgc18c5	Conduit	20180621	2018	6.682384e+08
23	cgc18c6	Conduit	20181211	2018	7.363992e+08
24	bmar18b7	Conduit	20181128	2018	1.167915e+09

		Issuer	Series \
0		Morgan Stanley Capital I	2018-BN11
1		Wells Fargo Commercial Mortgage Trust	2018-BNK10
2		JP Morgan Commercial Mortgage Finance Corp.	2018-C8
3		GS Mortgage Securities Corp. II	2018-GS9
4		UBS Commercial Mortgage Trust	2018-C15
5		COMM	2018-COR3
6		Wells Fargo Commercial Mortgage Trust	2018-BNK13
7		UBS Commercial Mortgage Trust	2018-C11
8		Banc of America Commercial Mortgage	2018-BNK12
9		Morgan Stanley Capital I	2018-H4
10		Credit Suisse	2018-CX11
11		Deutsche Mortgage and Asset Receiving Corp.	2018-B5
12		Deutsche Mortgage and Asset Receiving Corp.	2018-B4
13		Morgan Stanley Capital I	2018-H3
14		UBS Commercial Mortgage Trust	2018-C8
15		GS Mortgage Securities Corp. II	2018-GS10
16		JP Morgan Chase Commercial Mortgage Securities...	2018-B8
17		BBCMS	2018-C2
18		CSAIL	2018-C14
19		UBS Commercial Mortgage Trust	2018-C14
20		Deutsche Mortgage and Asset Receiving Corp.	2018-C1
21		Citigroup Commercial Mortgage Trust	2018-B6

22	Citigroup Commercial Mortgage Trust	2018-C5
23	Citigroup Commercial Mortgage Trust	2018-C6
24	BMARK	2018-B7

	Bloomberg Name	AAA	Amt Outstanding	# of mortgages	Risk Retention Type
0	BANK 2018-BN11		\$209,853,000	42	Vertical
1	BANK 2018-BN10		\$602,750,000	68	Vertical
2	JPMDB 2018-C8		\$183,513,650	41	L-Shape
3	GSMS 2018-GS9		\$361,127,000	37	L-shape
4	UBSCM 2018-C15		\$165,779,000	41	Horizontal
5	COMM 2018-COR3		\$372,251,000	41	Horizontal
6	BANK 2018-BN13		\$244,803,000	62	Vertical
7	UBSCM 2018-C11		\$210,569,000	48	L-shape
8	BANK 2018-BN12		\$243,028,000	63	Vertical
9	MSC 2018-H4		\$353,164,000	51	Horizontal
10	CSAIL 2018-CX11		\$230,612,000	44	Horizontal
11	BMARK 2018-B5		\$280,117,000	55	Horizontal
12	BMARK 2018-B4		\$339,983,000	44	Horizontal
13	MSC 2018-H3		\$278,376,000	66	Horizontal
14	UBSCM 2018-C8		\$320,645,000	67	Horizontal
15	GSMS 2018-GS10		\$199,668,000	33	L-shape
16	BMARK 2018-B8		\$478,261,000	41	Horizontal
17	BBCMS 2018-C2		\$376,521,028	44	L-Shape
18	CSAIL 2018-C14		\$230,612,000	44	Vertical
19	UBSCM 2018-C14		\$166,162,000	45	Vertical
20	DBGS 2018-C1		\$504,969,000	31	L-shape
21	BMARK 2018-B6		\$305,239,000	55	L-shape
22	CGCMT 2018-C5		\$208,766,000	40	Horizontal
23	CGCMT 2018-C6		\$205,979,000	35	Horizontal
24	BMARK 2018-B7		\$320,279,000	51	L-Shape

```
[45]: pd.read_excel('CMBX12.xlsx')
```

```
[45]: Deal Name Deal Type Closing Date Vintage Cutoff Balance \
0 ms18bn11 Conduit 20180426 2018 6.882007e+08
1 wf18bn10 Conduit 20180213 2018 1.287149e+09
2 jpm18c8 Conduit 20180615 2018 7.131377e+08
3 gs18gs9 Conduit 20180329 2018 8.871306e+08
4 ub18c15 Conduit 20181228 2018 6.464773e+08
5 com18co3 Conduit 20180522 2018 1.006083e+09
6 wf18bn13 Conduit 20180802 2018 9.441900e+08
7 ub18c11 Conduit 20180710 2018 8.038161e+08
8 ba18bn12 Conduit 20180530 2018 9.011700e+08
9 ms18h4 Conduit 20181227 2018 7.968060e+08
10 cs18cx11 Conduit 20180418 2018 9.528688e+08
11 bmar18b5 Conduit 20180821 2018 1.039132e+09
12 bmar18b4 Conduit 20180713 2018 1.158480e+09
```

13	ms18h3	Conduit	20180712	2018	1.024238e+09
14	ubs18c8	Conduit	20180227	2018	1.045168e+09
15	gs18gs10	Conduit	20180730	2018	8.738348e+08
16	bmar18b8	Conduit	20181227	2018	1.049017e+09
17	bb18c2	Conduit	20181220	2018	8.918536e+08
18	cs18c14	Conduit	20181129	2018	7.702287e+08
19	ubs18c14	Conduit	20181212	2018	6.508850e+08
20	db18c1	Conduit	20181030	2018	1.100682e+09
21	bmar18b6	Conduit	20181009	2018	1.147030e+09
22	cgc18c5	Conduit	20180621	2018	6.682384e+08
23	cgc18c6	Conduit	20181211	2018	7.363992e+08
24	bmar18b7	Conduit	20181128	2018	1.167915e+09

		Issuer	Series \
0		Morgan Stanley Capital I	2018-BN11
1		Wells Fargo Commercial Mortgage Trust	2018-BNK10
2		JP Morgan Commercial Mortgage Finance Corp.	2018-C8
3		GS Mortgage Securities Corp. II	2018-GS9
4		UBS Commercial Mortgage Trust	2018-C15
5		COMM	2018-COR3
6		Wells Fargo Commercial Mortgage Trust	2018-BNK13
7		UBS Commercial Mortgage Trust	2018-C11
8		Banc of America Commercial Mortgage	2018-BNK12
9		Morgan Stanley Capital I	2018-H4
10		Credit Suisse	2018-CX11
11		Deutsche Mortgage and Asset Receiving Corp.	2018-B5
12		Deutsche Mortgage and Asset Receiving Corp.	2018-B4
13		Morgan Stanley Capital I	2018-H3
14		UBS Commercial Mortgage Trust	2018-C8
15		GS Mortgage Securities Corp. II	2018-GS10
16		JP Morgan Chase Commercial Mortgage Securities...	2018-B8
17		BBCMS	2018-C2
18		CSAIL	2018-C14
19		UBS Commercial Mortgage Trust	2018-C14
20		Deutsche Mortgage and Asset Receiving Corp.	2018-C1
21		Citigroup Commercial Mortgage Trust	2018-B6
22		Citigroup Commercial Mortgage Trust	2018-C5
23		Citigroup Commercial Mortgage Trust	2018-C6
24		BMARK	2018-B7

	Bloomberg Name	AAA Amt Outstanding	# of mortgages	Risk Retention Type
0	BANK 2018-BN11	209853000	42	Vertical
1	BANK 2018-BN10	602750000	68	Vertical
2	JPMDB 2018-C8	183513650	41	L-Shape
3	GSMS 2018-GS9	361127000	37	L-shape
4	UBSCM 2018-C15	165779000	41	Horizontal
5	COMM 2018-COR3	372251000	41	Horizontal

6	BANK 2018-BN13	244803000	62	Vertical
7	UBSCM 2018-C11	210569000	48	L-shape
8	BANK 2018-BN12	243028000	63	Vertical
9	MSC 2018-H4	353164000	51	Horizontal
10	CSAIL 2018-CX11	230612000	44	Horizontal
11	BMARK 2018-B5	280117000	55	Horizontal
12	BMARK 2018-B4	339983000	44	Horizontal
13	MSC 2018-H3	278376000	66	Horizontal
14	UBSCM 2018-C8	320645000	67	Horizontal
15	GSMS 2018-GS10	199668000	33	L-shape
16	BMARK 2018-B8	478261000	41	Horizontal
17	BBCMS 2018-C2	376521028	44	L-Shape
18	CSAIL 2018-C14	230612000	44	Vertical
19	UBSCM 2018-C14	166162000	45	Vertical
20	DBGS 2018-C1	504969000	31	L-shape
21	BMARK 2018-B6	305239000	55	L-shape
22	CGCMT 2018-C5	208766000	40	Horizontal
23	CGCMT 2018-C6	205979000	35	Horizontal
24	BMARK 2018-B7	320279000	51	L-Shape

```
[67]: df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 25 entries, 0 to 24
Data columns (total 11 columns):
Deal Name          25 non-null object
Deal Type          25 non-null object
Closing Date       25 non-null int64
Vintage            25 non-null int64
Cutoff Balance     25 non-null float64
Issuer             25 non-null object
Series             25 non-null object
Bloomberg Name     25 non-null object
AAA Amt Outstanding 25 non-null object
# of mortgages     25 non-null int64
Risk Retention Type 25 non-null object
dtypes: float64(1), int64(3), object(7)
memory usage: 2.3+ KB
```

```
[68]: df.size
```

```
[68]: 275
```

```
[71]: len(df)
```

```
[71]: 25
```

```
[72]: df.dtypes
```

```
[72]: Deal Name      object
      Deal Type     object
      Closing Date  int64
      Vintage       int64
      Cutoff Balance float64
      Issuer        object
      Series        object
      Bloomberg Name object
      AAA Amt Outstanding object
      # of mortgages int64
      Risk Retention Type object
      dtype: object
```

```
[73]: df.shape
```

```
[73]: (25, 11)
```

Find the row with the maximum cutoff balance:

```
[48]: df[df['Cutoff Balance'] == max(df['Cutoff Balance'])]
```

```
[48]: Deal Name Deal Type Closing Date Vintage Cutoff Balance \
1  wf18bn10  Conduit      20180213      2018      1.287149e+09

      Issuer      Series Bloomberg Name \
1  Wells Fargo Commercial Mortgage Trust  2018-BNK10  BANK 2018-BN10

      AAA Amt Outstanding # of mortgages Risk Retention Type
1          $602,750,000          68          Vertical
```

Find the deals with Morgan Stanley as Issuer:

```
[50]: df[df['Issuer'] == 'Morgan Stanley Capital I']
```

```
[50]: Deal Name Deal Type Closing Date Vintage Cutoff Balance \
0  ms18bn11  Conduit      20180426      2018      6.882007e+08
9  ms18h4    Conduit      20181227      2018      7.968060e+08
13 ms18h3    Conduit      20180712      2018      1.024238e+09

      Issuer      Series Bloomberg Name AAA Amt Outstanding \
0  Morgan Stanley Capital I  2018-BN11  BANK 2018-BN11      $209,853,000
9  Morgan Stanley Capital I  2018-H4    MSC 2018-H4      $353,164,000
13 Morgan Stanley Capital I  2018-H3    MSC 2018-H3      $278,376,000

      # of mortgages Risk Retention Type
0          42          Vertical
```


9	51	Horizontal
13	66	Horizontal

A large number of methods collectively compute descriptive statistics and other related operations on DataFrame. Most of these are aggregations like `sum()`, `mean()`.

```
[60]: df['Issuer'].unique()
```

```
[60]: array(['Morgan Stanley Capital I',
        'Wells Fargo Commercial Mortgage Trust',
        'JP Morgan Commercial Mortgage Finance Corp.',
        'GS Mortgage Securities Corp. II', 'UBS Commercial Mortgage Trust',
        'COMM', 'Banc of America Commercial Mortgage', 'Credit Suisse',
        'Deutsche Mortgage and Asset Receiving Corp.',
        'JP Morgan Chase Commercial Mortgage Securities Corp.', 'BBCMS',
        'CSAIL', 'Citigroup Commercial Mortgage Trust', 'BMARK'],
        dtype=object)
```

```
[65]: df['Cutoff Balance'].sum()
```

```
[65]: 22950131242.02
```

- Which deal contains the maximum and minimum number of mortgage underliers?
- Which deal has the earliest and latest Closing Date?
- What kind of risk retention types are there?
- Update the table with a new column, Average Original Mortgage Balance from the Cutoff Balance and # of mortgages.
- What is the average number of mortgages?