Lesson P2 – Pandas DataFrames

Rob Capra

INLS 570

Data Frame

- Tabular data structure, like a spreadsheet
  - Ordered collection of columns
  - Each column can be a diff data type
  - Row and column indexes

	A	B	C
0	v0	v1	v2
1	v3	v4	v5
2	v6	v7	v8
3	v9	v10	v11
4	v12	v13	v14

DataFrame

- Create a DataFrame from a dict of equal-length lists

	course	semester	enrollment
0	inls285	s13	31
1	inls285	s14	58
2	inls382	s13	26
3	inls382	s14	46
4	inls523	s13	19
5	inls523	s14	28

```
d = {'course': ['inls285', 'inls285', 'inls382', 'inls382', 'inls523', 'inls523'],
         'semester': ['s13', 's14', 's13', 's14', 's13', 's14'],
         'enrollment': [31, 58, 26, 46, 19, 28]}


df = DataFrame(d)
```

DataFrame

```
In [73]: d = {'course': ['inls285', 'inls285', 'inls382',
'inls382', 'inls523', 'inls523'], 'semester': ['s13',
's14', 's13', 's14', 's13', 's14'], 'enrollment': [31,
58, 26, 46, 19, 28]}

In [74]: df = DataFrame(d)

In [75]: df
Out[75]:
    course  enrollment semester
0  inls285          31      s13
1  inls285          58      s14
2  inls382          26      s13
3  inls382          46      s14
4  inls523          19      s13
5  inls523          28      s14

In [76]:
```

Retrieving Columns

- Retrieve columns by dict-like notation, or by attribute
- Columns are retrieved as a Series

```
In [76]: type(df)
Out[76]: pandas.core.frame.DataFrame

In [77]: df
Out[77]:
    course  enrollment semester
0  inls285          31      s13
1  inls285          58      s14
2  inls382          26      s13
3  inls382          46      s14
4  inls523          19      s13
5  inls523          28      s14

In [78]: s = df['course']

In [79]: type(s)
Out[79]: pandas.core.series.Series
```

```
In [80]: s
Out[80]:
0    inls285
1    inls285
2    inls382
3    inls382
4    inls523
5    inls523
Name: course, dtype: object

In [81]: s2 = df.course

In [82]: s2
Out[82]:
0    inls285
1    inls285
2    inls382
3    inls382
4    inls523
5    inls523
Name: course, dtype: object
```

Retrieving Columns

- Once you retrieved a column, you can use it like a collection

```
In [128]: df
Out[128]:
    course  enrollment semester
0  inls285          31      s13
1  inls285          58      s14
2  inls382          26      s13
3  inls382          46      s14
4  inls523          19      s13
5  inls523          28      s14

In [129]: df.enrollment
Out[129]:
0    31
1    58
2    26
3    46
4    19
5    28
Name: enrollment, dtype: int64

In [130]: type(df.enrollment)
Out[130]: pandas.core.series.Series
```

```
In [131]: for n in df.enrollment:
     ...:     print (n)
     ...:
31
58
26
46
19
28
```

DataFrame Index

- DataFrames can also have a customized index (like Series do)

```
In [83]: d = {'course': ['inls285', 'inls285', 'inls382', 'inls382', 'inls523',
'inls523'], 'semester': ['s13', 's14', 's13', 's14', 's13', 's14'], 'enrollment':
[31, 58, 26, 46, 19, 28]}

In [84]: d
Out[84]:
{'course': ['inls285', 'inls285', 'inls382', 'inls382', 'inls523', 'inls523'],
 'enrollment': [31, 58, 26, 46, 19, 28],
 'semester': ['s13', 's14', 's13', 's14', 's13', 's14']}

In [86]: df = DataFrame(d, index=['c1234', 'c2345', 'c8822', 'c7654', 'c5512',
'c4321'])

In [87]: df
Out[87]:
        course  enrollment semester
c1234  inls285          31      s13
c2345  inls285          58      s14
c8822  inls382          26      s13
c7654  inls382          46      s14
c5512  inls523          19      s13
c4321  inls523          28      s14
```

DataFrame Index

- Columns are retrieved as a Series w/ same index as DF

```
In [87]: df
Out[87]:
        course  enrollment semester
c1234  inls285          31      s13
c2345  inls285          58      s14
c8822  inls382          26      s13
c7654  inls382          46      s14
c5512  inls523          19      s13
c4321  inls523          28      s14

In [88]: df.course
Out[88]:
c1234    inls285
c2345    inls285
c8822    inls382
c7654    inls382
c5512    inls523
c4321    inls523
Name: course, dtype: object
```

Retrieve Rows using df.loc

```
In [89]: df
Out[89]:
        course  enrollment semester
c1234  inls285          31      s13
c2345  inls285          58      s14
c8822  inls382          26      s13
c7654  inls382          46      s14
c5512  inls523          19      s13
c4321  inls523          28      s14

In [90]: s = df.loc['c7654']

In [91]: type(s)
Out[91]: pandas.core.series.Series

In [92]: s
Out[92]:
course        inls382
enrollment         46
semester          s14
Name: c7654, dtype: object

In [93]: s.values
Out[93]: array(['inls382', 46, 's14'], dtype=object)

In [94]: s.index
Out[94]: Index(['course', 'enrollment', 'semester'], dtype='object')
```

Notice how the row is retrieved as a Series whose index is the columns of the DF.

Exercise P2.1 – DataFrame practice

- Create a DataFrame with the following play count data:

	Aug	Sept	Nov
David Bowie	571	623	409
The Beatles	725	518	822
New Order	274	492	368

- After creating the DF:
  1. Extract the Sept column and compute the total # of plays

DataFrame

- Create a new column and assign values to it using assignment

```
In [95]: df
Out[95]:
        course  enrollment semester
c1234  inls285          31      s13
c2345  inls285          58      s14
c8822  inls382          26      s13
c7654  inls382          46      s14
c5512  inls523          19      s13
c4321  inls523          28      s14

In [96]: df['tmp'] = [1, 3, 5, 7, 8, 9]

In [97]: df
Out[97]:
        course  enrollment semester  tmp
c1234  inls285          31      s13    1
c2345  inls285          58      s14    3
c8822  inls382          26      s13    5
c7654  inls382          46      s14    7
c5512  inls523          19      s13    8
c4321  inls523          28      s14    9
```

DataFrame

- A dict of dicts will create a DF with outer dict keys as the columns and inner dicts keys as row indices

```
In [98]: d = {'unc': {2012: 4.1, 2013: 4.3, 2014: 4.5}, 'duke': {2012: 3.8, 2013:
3.8, 2014: 4.1}}

In [99]: df = DataFrame(d)

In [100]: df
Out[100]:
      duke  unc
2012   3.8  4.1
2013   3.8  4.3
2014   4.1  4.5

In [101]: df.columns
Out[101]: Index(['duke', 'unc'], dtype='object')

In [102]: df.index
Out[102]: Int64Index([2012, 2013, 2014], dtype='int64')

In [103]: df.T
Out[103]:
      2012  2013  2014
duke   3.8   3.8   4.1
unc    4.1   4.3   4.5
```

Can transpose using T, like with numpy arrays

DataFrame

- DF columns can be extracted and operated on as either Series or numpy arrays

```
In [104]: df
Out[104]:
      duke  unc
2012   3.8  4.1
2013   3.8  4.3
2014   4.1  4.5

In [105]: s = df.unc

In [106]: type(s)
Out[106]: pandas.core.series.Series

In [107]: a = df.unc.values

In [108]: type(a)
Out[108]: numpy.ndarray

In [109]: s.sum()
Out[109]: 12.899999999999999

In [110]: a.sum()
Out[110]: 12.899999999999999
```

Exercise P2.2 – Sum rows and columns of a DataFrame

- Create a DataFrame with the following play count data:

	Aug	Sept	Nov
David Bowie	571	623	409
The Beatles	725	518	822
New Order	274	492	368

- After creating the DF:
  1. Compute the total # of plays for the Sept column
  2. Compute the total # of plays for the 'David Bowie' row

  (Hint: use df.loc())