BIG DATA FARM

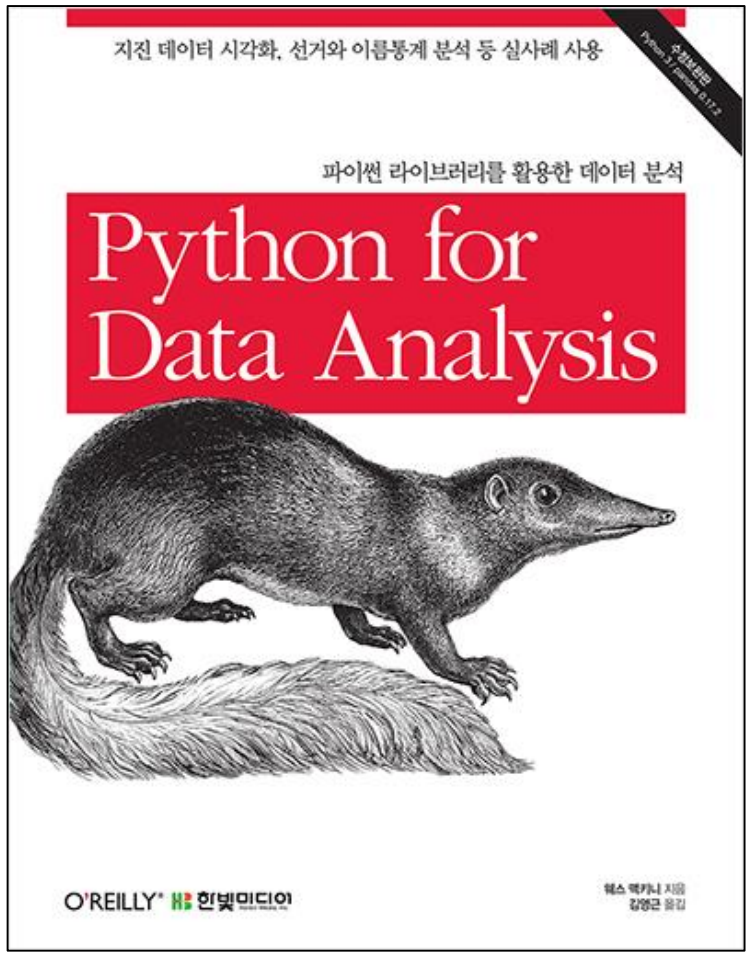
Data Science in DGU
Universe of Farm

SESSION #3

PANDAS 이해하기

By Jaehoon Kim

© 2018. BigData Farm all rights reserved



Chapter 01 Pandas는 무엇인가?

1.1 Pandas의 기본 구조

Chapter 02 Pandas는 어떻게 쓸까?

2.1 Pandas로 데이터프레임 다루기
- 출력 / 변경 / 선택 / 합치기 / 불러오기 / 저장하기

Chapter 03 실습

3.1 데이터프레임 다루기
3.2 데이터프레임 생성하기

© 2018. BigData Farm all rights reserved

1.0 Pandas는 무엇인가?

Pandas 란?

- Pandas는 데이터 분석을 하기 위해서 필요한 자료구조나 데이터 분석 도구를 제공하는 Python 라이브러리이다.

- 엑셀 시트와 같이 데이터프레임을 활용하여 데이터를 쉽게 핸들링할 수 있다.

© 2018. BigData Farm all rights reserved

1.1 Pandas의 기본 구조

시리즈(Series)

- Pandas의 시리즈(Series)는 1차원 리스트, 혹은 Numpy의 1차원 배열과 유사하다.
- 다만 Numpy와는 달리 서로 다른 자료형도 시리즈 내에 담을 수 있다.

© 2018. BigData Farm all rights reserved

1.1 Pandas의 기본 구조

시리즈(Series) 생성하기

- 일반적으로 파이썬 리스트를 시리즈로 변환하여 생성한다.

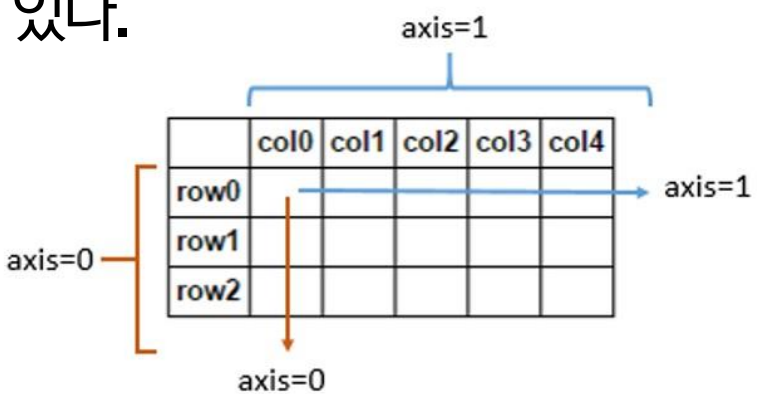
```
In [2]: a1 = pd.Series([1,3,5,7,9,'dongguk'])
        a1

Out[2]: 0         1
        1         3
        2         5
        3         7
        4         9
        5    dongguk
        dtype: object

In [3]: a2 = pd.Series([1,3,5, np.nan, 9, 11])
        a2

Out[3]: 0     1.0
        1     3.0
        2     5.0
        3     NaN
        4     9.0
        5    11.0
        dtype: float64
```

© 2018. BigData Farm all rights reserved

1.1 Pandas의 기본 구조

데이터프레임(Dataframe)

- Pandas의 데이터프레임(Dataframe)은 2차원 리스트와 유사한 자료구조이다.
- 행렬(matrix)과 같이 행(row)과 열(column)로 이루어져 있으며, 이를 인덱스를 이용해 접근할 수 있다.
- 두 개의 축(axis)가 있어 행과 열을 관리할 수 있다.
- 데이터프레임에는 시리즈와 같이 서로 다른 자료형을 담을 수 있다.



*이미지 출처: https://stackoverflow.com/questions/25773245/ambiguity-in-pandas-dataframe-numpy-array-axis-definition

© 2018. BigData Farm all rights reserved

1.1 Pandas의 기본 구조

데이터프레임(Dataframe) 생성하기

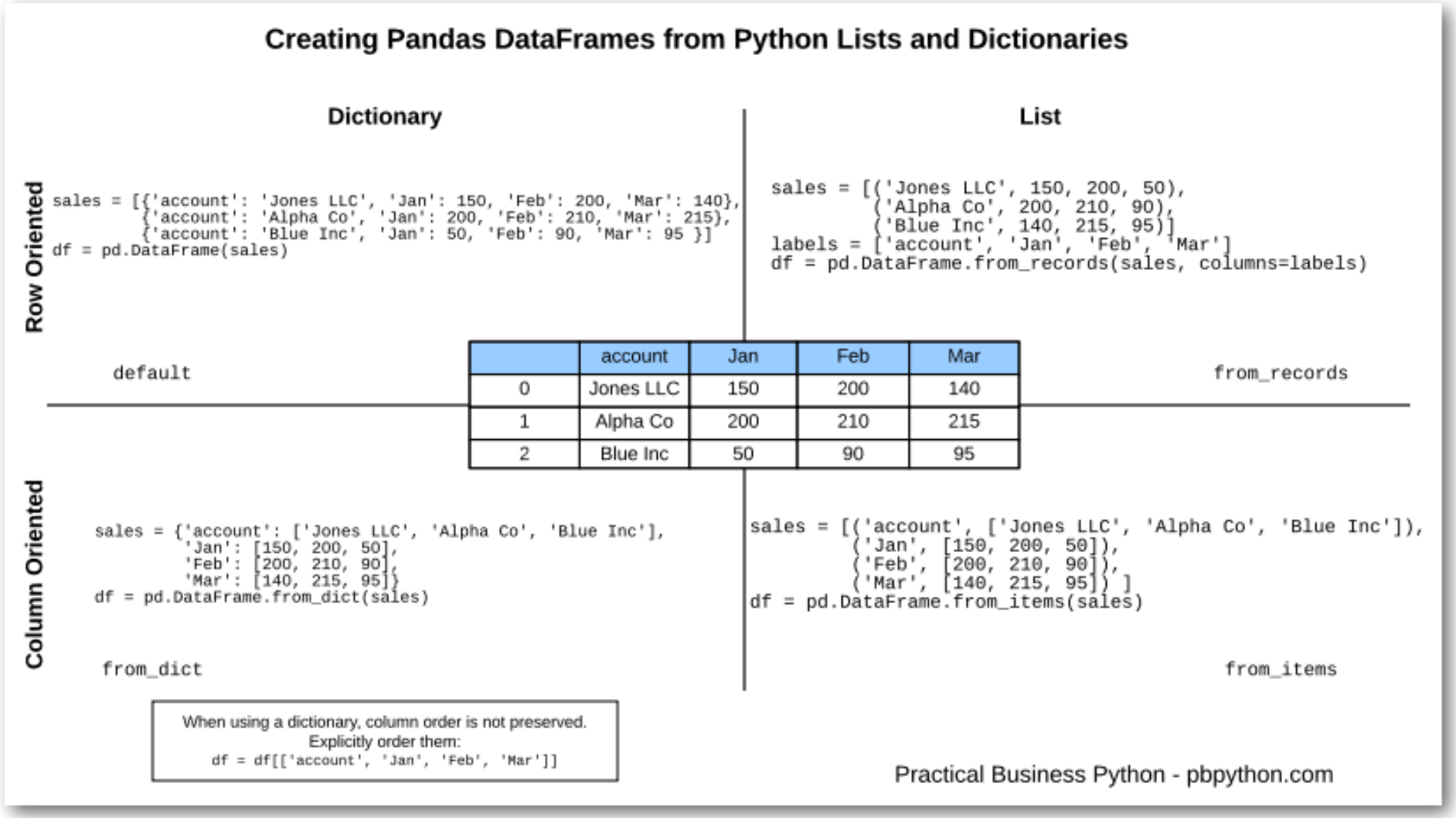
- 데이터프레임은 리스트, 배열 혹은 딕셔너리로 생성할 수 있다.

→ 다음 슬라이드 참고

© 2018. BigData Farm all rights reserved

1.1 Pandas의 기본 구조

*이미지 출처: http://pbpython.com/images/pandas-dataframe-shadow.png



© 2018. BigData Farm all rights reserved

1.1 Pandas의 기본 구조

데이터프레임(Dataframe) 생성하기 – 배열 & 리스트 사용

In [4]:
```
df1 = pd.DataFrame(
    [
    [1,2,3],
    [4,5,6]
    ] )
df1
```

Out[4]:

	0	1	2
0	1	2	3
1	4	5	6

In [5]:
```
df2 = pd.DataFrame(np.eye(3))
df2
```

Out[5]:

	0	1	2
0	1.0	0.0	0.0
1	0.0	1.0	0.0
2	0.0	0.0	1.0

© 2018. BigData Farm all rights reserved

1.1 Pandas의 기본 구조

데이터프레임(Dataframe) 생성하기 – 딕셔너리 사용

In [7]:
```
places = ['1st','2nd','3rd']
scores = [100, 50, 20]
```

In [8]:
```
df3 = pd.DataFrame(
    {'score':scores,
    'place':places}
    )
df3
```

Out[8]:

	place	score
0	1st	100
1	2nd	50
2	3rd	20

© 2018. BigData Farm all rights reserved

2.1 Pandas로 데이터프레임 다루기

칼럼명 및 인덱스 설정하기

```
In [11]: df4 = pd.DataFrame(np.random.randn(10, 3), index = range(1, 11), columns = ['one','two','three'])
         df4
```

Out[11]:

	one	two	three
1	0.649247	0.302397	-0.514178
2	1.409297	0.459228	-2.321946
3	-0.106654	-0.822974	0.976719
4	0.829096	0.299768	-1.368047
5	1.273300	1.244874	-0.648139
6	-0.772035	2.146269	0.383709
7	0.302706	-0.545409	-2.091717
8	0.216951	-0.475571	0.612606
9	-1.042621	-2.048551	0.986646
10	1.024734	0.269994	-2.391017

따로 설정하지 않을 경우 **default값**으로 설정된다.

 # 이 경우 default 값은 0부터 시작하는 정수이다.

© 2018. BigData Farm all rights reserved

2.1 Pandas로 데이터프레임 다루기

데이터 출력하기

- **head( )와 tail( ) 함수로 데이터의 일부를 출력할 수 있다.**

 # 따로 인자 값을 설정하지 않으면 상위 혹은 하위 5개의 데이터를 출력한다.

```
In [12]: df4.head()
```

Out[12]:

	one	two	three
1	0.649247	0.302397	-0.514178
2	1.409297	0.459228	-2.321946
3	-0.106654	-0.822974	0.976719
4	0.829096	0.299768	-1.368047
5	1.273300	1.244874	-0.648139

```
In [13]: df4.tail(1)
```

Out[13]:

	one	two	three
10	1.024734	0.269994	-2.391017

© 2018. BigData Farm all rights reserved

2.1 Pandas로 데이터프레임 다루기

데이터 출력하기

- 인덱스와 칼럼명만 출력하기

함수가 아니기 때문에 뒤에 소괄호를 붙이지 않는다.

```
In [14]: df4.index
Out[14]: RangeIndex(start=1, stop=11, step=1)

In [15]: df4.columns
Out[15]: Index(['one', 'two', 'three'], dtype='object')
```

© 2018. BigData Farm all rights reserved

2.1 Pandas로 데이터프레임 다루기

데이터 출력하기

- **describe( ) 함수로 데이터의 요약(기초 통계량)을 출력할 수 있다.**

 # 단, 기초 통계량은 수치형 자료의 경우에만.

```
In [16]: df4.describe()
```

Out[16]:

	one	two	three
count	10.000000	10.000000	10.000000
mean	0.378402	0.083002	-0.637536
std	0.827829	1.152286	1.350931
min	-1.042621	-2.048551	-2.391017
25%	-0.025753	-0.527950	-1.910799
50%	0.475977	0.284881	-0.581158
75%	0.975824	0.420021	0.555381
max	1.409297	2.146269	0.986646

© 2018. BigData Farm all rights reserved

2.1 Pandas로 데이터프레임 다루기

데이터 변경하기

- T 구문을 통해서 데이터프레임의 열과 행을 바꿀 수 있다. (전치 **transpose**)

 # 전치 명령어는 함수가 아니기 때문에 뒤에 소괄호가 붙지 않는다.

```
In [17]: df4
```

Out[17]:

	one	two	three
1	0.649247	0.302397	-0.514178
2	1.409297	0.459228	-2.321946
3	-0.106654	-0.822974	0.976719
4	0.829096	0.299768	-1.368047
5	1.273300	1.244874	-0.648139
6	-0.772035	2.146269	0.383709
7	0.302706	-0.545409	-2.091717
8	0.216951	-0.475571	0.612606
9	-1.042621	-2.048551	0.986646
10	1.024734	0.269994	-2.391017

```
In [18]: df4.T
```

Out[18]:

	1	2	3	4	5	6	7	8	9	10
one	0.649247	1.409297	-0.106654	0.829096	1.273300	-0.772035	0.302706	0.216951	-1.042621	1.024734
two	0.302397	0.459228	-0.822974	0.299768	1.244874	2.146269	-0.545409	-0.475571	-2.048551	0.269994
three	-0.514178	-2.321946	0.976719	-1.368047	-0.648139	0.383709	-2.091717	0.612606	0.986646	-2.391017

© 2018. BigData Farm all rights reserved

2.1 Pandas로 데이터프레임 다루기

데이터 변경하기

- **sort_values( ) 함수를 통해서 특정 컬럼의 값을 기준으로 정렬할 수 있다.**

 # 특정 컬럼을 지정할 때에는 by 인자를 사용한다. 아래 예시를 참고할 것.

```
In [19]: df4.sort_values(by = 'three')
```

Out[19]:

	one	two	three
10	1.024734	0.269994	-2.391017
2	1.409297	0.459228	-2.321946
7	0.302706	-0.545409	-2.091717
4	0.829096	0.299768	-1.368047
5	1.273300	1.244874	-0.648139
1	0.649247	0.302397	-0.514178
6	-0.772035	2.146269	0.383709
8	0.216951	-0.475571	0.612606
3	-0.106654	-0.822974	0.976719
9	-1.042621	-2.048551	0.986646

© 2018. BigData Farm all rights reserved

2.1 Pandas로 데이터프레임 다루기

데이터 변경하기

- **DataFrame**에 새로운 칼럼을 삽입할 수 있다.

 # 칼럼을 추가할 때에는 데이터프레임 변수명 옆에 [칼럼명] 을 붙이고 진행한다.

```
In [21]: df4['four'] = np.random.random(10)
         df4
```

Out[21]:

	one	two	three	four
1	0.649247	0.302397	-0.514178	0.447244
2	1.409297	0.459228	-2.321946	0.654479
3	-0.106654	-0.822974	0.976719	0.258383
4	0.829096	0.299768	-1.368047	0.051405
5	1.273300	1.244874	-0.648139	0.407364
6	-0.772035	2.146269	0.383709	0.714396
7	0.302706	-0.545409	-2.091717	0.407108
8	0.216951	-0.475571	0.612606	0.859004
9	-1.042621	-2.048551	0.986646	0.504898
10	1.024734	0.269994	-2.391017	0.103385

df4['d'] = np.random.random(10)

© 2018. BigData Farm all rights reserved

2.1 Pandas로 데이터프레임 다루기

데이터 선택하기

- 칼럼명을 통해서 데이터프레임에서 **Series**를 가져올 수 있다.
  # Series로 가져올 때에는 대괄호 한 쌍([##])을 사용,
  DataFrame으로 가져올 때에는 대괄호 두 쌍([[##]])을 사용한다.

```
In [22]: one = df4['one']
         one
```

```
Out[22]: 1     0.649247
         2     1.409297
         3    -0.106654
         4     0.829096
         5     1.273300
         6    -0.772035
         7     0.302706
         8     0.216951
         9    -1.042621
         10    1.024734
         Name: one, dtype: float64
```

© 2018. BigData Farm all rights reserved

2.1 Pandas로 데이터프레임 다루기

데이터 합치기

- **concat()** : 따로 축을 설정하지 않을 경우 데이터를 칼럼 기준으로 병합한다.

```
In [23]: df1
Out[23]:
```

	0	1	2
0	1	2	3
1	4	5	6

```
In [24]: df2
Out[24]:
```

	0	1	2
0	1.0	0.0	0.0
1	0.0	1.0	0.0
2	0.0	0.0	1.0

```
In [26]: pd.concat([df1, df2])
Out[26]:
```

	0	1	2
0	1.0	2.0	3.0
1	4.0	5.0	6.0
0	1.0	0.0	0.0
1	0.0	1.0	0.0
2	0.0	0.0	1.0

© 2018. BigData Farm all rights reserved

2.1 Pandas로 데이터프레임 다루기

데이터 합치기

- **concat() : 축을 1로 설정할 경우 행을 기준으로 병합한다.**

```
In [27]: pd.concat([df1, df2], axis = 1)
```

Out[27]:

	0	1	2	0	1	2
0	1.0	2.0	3.0	1.0	0.0	0.0
1	4.0	5.0	6.0	0.0	1.0	0.0
2	NaN	NaN	NaN	0.0	0.0	1.0

© 2018. BigData Farm all rights reserved

2.1 Pandas로 데이터프레임 다루기

데이터 합치기

- **concat()** 외의 다른 다양한 병합 함수는 다음의 링크를 참고한다.

 # Link : https://pandas.pydata.org/pandas-docs/stable/merging.html#merging

© 2018. BigData Farm all rights reserved

2.1 Pandas로 데이터프레임 다루기

데이터 불러오기

- **Pandas에서 csv 데이터를 불러오는 구문은 다음과 같다.**

```
In [29]: df = pd.read_csv('2018_03_18_04.csv', encoding = 'cp949')
         df.head()
```

Out[29]:

	artist	artist_code	title	title_code	rank	date	hour	site
0	BIGBANG	198094	꽃 길	30948698	1	2018-03-18	4	melon
1	마마무	750053	별이 빛나는 밤	30937275	2	2018-03-18	4	melon
2	iKON	895741	사랑을 했다 (LOVE SCENARIO)	30859584	3	2018-03-18	4	melon
3	헤이즈 (Heize)	751611	Jenga (Feat. Gaeko)	30939452	4	2018-03-18	4	melon
4	Wanna One (워너원)	1865973	약속해요 (I.P.U.)	30930312	5	2018-03-18	4	melon

© 2018. BigData Farm all rights reserved

2.1 Pandas로 데이터프레임 다루기

데이터 저장하기

- DataFrame을 csv 데이터로 저장하는 구문은 다음과 같다.

```
〉〉〉 데이터프레임명.to_csv('파일명.csv')
```

© 2018. BigData Farm all rights reserved

실습

데이터프레임 다루기

- ‘2018_03_18_04.csv’의 데이터를 사용해서 아래 예시와 같도록 만드시오.

Out[32]:

	rank	title	artist
0	1	꽃 길	BIGBANG
1	2	별이 빛나는 밤	마마무
2	3	사랑을 했다 (LOVE SCENARIO)	iKON
3	4	Jenga (Feat. Gaeko)	헤이즈 (Heize)
4	5	약속해요 (I.P.U.)	Wanna One (워너원)
5	6	그날처럼	장덕철
6	7	그때 헤어지면 돼	로이킴
7	8	Beautiful	Wanna One (워너원)
8	9	뿜뿜	모모랜드 (MOMOLAND)
9	10	에너제틱 (Energetic)	Wanna One (워너원)

© 2018. BigData Farm all rights reserved

실습

데이터프레임 생성하기

- 다음 예시와 같도록 데이터프레임을 생성하시오.

Out[34]:

	Major	Student_Number	Professor_Number	Building
0	Computer Science	100	10	신공학관
1	Engineering	150	20	원흥관
2	Business Administration	300	40	경영관
3	English	50	8	명진관

© 2018. BigData Farm all rights reserved