

시계열 데이터

- datetime 오브젝트
- 사례별 시계열 데이터 계산하기

■ datetime 오브젝트

● datetime 으로 현재 시각 출력하기

- datetime 라이브러리 가져오기

```
from datetime import datetime
```

- 현재 시각 출력

```
now1 = datetime.now()  
print(now1)  
now2 = datetime.today()  
print(now2)
```

```
2021-02-12 15:10:51.689885
```

```
2021-02-12 15:10:51.690888
```

■ datetime 오브젝트

● datetime 으로 현재 시각 출력하기

- 시간을 직접 입력하여 datetime 오브젝트 생성

```
t1 = datetime.now()  
t2 = datetime(1970, 1, 1)  
t3 = datetime(1970, 12, 12, 13, 24, 34)  
  
print(t1)  
print(t2)  
print(t3)
```

2021-02-12 15:12:02.133773

1970-01-01 00:00:00

1970-12-12 13:24:34

- datetime 오브젝트 연산

```
diff1 = t1 - t2  
  
print(diff1)
```

18670 days, 15:12:02.133773

■ datetime 오브젝트

● datetime 오브젝트로 변환하기 : to_datetime()

- ebola 데이터 집합 가져오기

```
import pandas as pd
ebola = pd.read_csv('data/country_timeseries.csv')
print(ebola.info())
print(ebola.head(2))
```

```
<class 'pandas.core.frame.DataFrame'>
```

```
RangeIndex: 122 entries, 0 to 121
```

```
Data columns (total 18 columns):
```

#	Column	Non-Null Count	Dtype
0	Date	122 non-null	object
1	Day	122 non-null	int64
2	Cases_Guinea	93 non-null	float64
3	Cases_Liberia	83 non-null	float64

	Date	Day	Cases_Guinea	Cases_Liberia
0	1/5/2015	289	2776.0	NaN
1	1/4/2015	288	2775.0	NaN

■ datetime 오브젝트

● datetime 오브젝트로 변환하기 : to_datetime()

- Date 열의 자료형을 datetime 오브젝트로 변환

```
ebola['date_dt'] = pd.to_datetime(ebola['Date'])  
print(ebola.info())  
print(ebola.head(2))
```

```
15 Deaths_UnitedStates    18 non-null    float64  
16 Deaths_Spain           16 non-null    float64  
17 Deaths_Mali            12 non-null    float64  
18 date_dt                 122 non-null   datetime64[ns]  
dtypes: datetime64[ns](1), float64(16), int64(1), object(1)  
memory usage: 18.2+ KB
```

	Deaths_UnitedStates	Deaths_Spain	Deaths_Mali	date dt
0	NaN	NaN	NaN	2015-01-05
1	NaN	NaN	NaN	2015-01-04

■ datetime 오브젝트

● datetime 오브젝트로 변환하기 : to_datetime()

- 시간 형식을 지정하여 datetime 오브젝트 생성

```
test_df1 = pd.DataFrame({'order_day': ['01/01/15', '02/01/15', '03/01/15']})

test_df1['date_dt1'] = pd.to_datetime(test_df1['order_day'], format='%d/%m/%y')
test_df1['date_dt2'] = pd.to_datetime(test_df1['order_day'], format='%m/%d/%y')
test_df1['date_dt3'] = pd.to_datetime(test_df1['order_day'], format='%y/%m/%d')

print(test_df1)
```

	order_day	date_dt1	date_dt2	date_dt3
0	01/01/15	2015-01-01	2015-01-01	2001-01-15
1	02/01/15	2015-01-02	2015-02-01	2002-01-15
2	03/01/15	2015-01-03	2015-03-01	2003-01-15

■ datetime 오브젝트

● datetime 오브젝트로 변환하기

- 시간 형식 지정자

지정자	설명	결과	지정자	설명	결과
%Y	년(4자리)	2002	%y	년(2자리)	02
%m	월	01-12	%B, %b	월(영어)	January, Jan
%d	일	01-31			
%H	시(24시간)	00-23	%I	시(12시간)	01-12
%M	분	00-59			
%S	초	00-59	%u	요일	1-7(월-일)
%w	요일	0-6(일-토)	%A, %a	요일(영어)	Sunday, Sun
%p	오전, 오후	AM, PM	%f	마이크로초	000000-999999
%z	UTC 차이	UTC+0900	%Z	기준 지역명	UTC, EST, ...
%j	올해 지난 일	001-366	%U	올해 지난 주	00-53
%c, %x	날짜와 시간				

■ datetime 오브젝트

● datetime 오브젝트로 변환하기 : to_datetime()

- 시계열 데이터 분리 : strftime()

```
now = datetime.now()
nowDate = now.strftime('%Y-%m-%d')
nowTime = now.strftime('%H:%M:%S')
nowWeek = now.strftime('%w요일')
```

```
print(now)
print(nowDate)
print(nowTime)
print(nowWeek)
```

2021-02-12 16:49:16.533203

2021-02-12

16:49:16

5요일

■ datetime 오브젝트

- datetime 오브젝트로 변환하기 : `read_csv(parse_dates=['열 이름'])`
 - 에볼라 데이터 집합 가져오기

```
ebola1 = pd.read_csv('data/country_timeseries.csv', parse_dates=['Date'])
ebola1.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 122 entries, 0 to 121
Data columns (total 18 columns):
#   Column                Non-Null Count  Dtype
---  -
0   Date                  122 non-null   datetime64[ns]
1   Day                   122 non-null   int64
2   Cases_Guinea          93 non-null    float64
3   Cases_Liberia         83 non-null    float64
```

- year / month / day 속성 사용

```
print(ebola1['Date'][0].year)
print(ebola1['Date'][0].month)
print(ebola1['Date'][0].day)
```

```
2015
1
5
```

■ datetime 오브젝트

● dt 접근자 사용하기

- 인덱스를 사용하지 않고 dt 접근자로 년/월/일 추출하여 저장

```
ebola = pd.read_csv('data/country_timeseries.csv')
ebola['date_dt'] = pd.to_datetime(ebola['Date'])
ebola['year'] = ebola['date_dt'].dt.year
ebola['month'] = ebola['date_dt'].dt.month
ebola['day'] = ebola['date_dt'].dt.day

ebola[['Date', 'date_dt', 'year', 'month', 'day']].head()
```

	Date	date_dt	year	month	day
0	1/5/2015	2015-01-05	2015	1	5
1	1/4/2015	2015-01-04	2015	1	4
2	1/3/2015	2015-01-03	2015	1	3
3	1/2/2015	2015-01-02	2015	1	2
4	12/31/2014	2014-12-31	2014	12	31

■ 사례별 시계열 데이터 계산하기

● 에볼라 최초 발병일 계산하기

- 최초 발병일

```
print(ebola['date_dt'].min())  
print(ebola.loc[ebola['date_dt'] == ebola['date_dt'].min()])
```

2014-03-22 00:00:00

	Date	Day	Cases_Guinea	Cases_Liberia	Cases_SierraLeone
121	3/22/2014	0	49.0	NaN	NaN

1 rows x 22 columns

- 에볼라 진행 정도 확인

```
ebola['outbreak_d'] = ebola['date_dt'] - ebola['date_dt'].min()  
print(ebola[['Date', 'Day', 'outbreak_d']].head())
```

	Date	Day	outbreak_d
0	1/5/2015	289	289 days
1	1/4/2015	288	288 days
2	1/3/2015	287	287 days
3	1/2/2015	286	286 days
4	12/31/2014	284	284 days

■ 사례별 시계열 데이터 계산하기

● 파산한 은행의 수 계산

- 은행 데이터 확인

```
banks = pd.read_csv('data/banklist.csv')
print(banks.info())
print(banks.head())
```

```
#   Column      Non-Null Count  Dtype
---  -
0   Bank Name    555 non-null      object
1   City          555 non-null      object
2   ST            555 non-null      object
3   CERT          555 non-null      int64
4   Acquiring Institution 555 non-null      object
5   Closing Date  555 non-null      object
6   Updated Date  555 non-null      object
```

dtypes: int64(1), object(6)

	Bank Name	City	ST	CERT	Acquiring Institution	Closing Date	Updated Date
0	Washington Federal Bank for Savings	Chicago	IL	30570	Royal Savings Bank	15-Dec-17	20-Dec-17
1	The Farmers and Merchants State Bank of Argonia	Argonia	KS	17719	Conway Bank	13-Oct-17	20-Oct-17
2	Fayette County Bank	Saint Elmo	IL	1802	United Fidelity Bank, fsb	26-May-17	26-Jul-17
3	Guaranty Bank, (d/b/a BestBank in Georgia & Mi...	Milwaukee	WI	30003	First-Citizens Bank & Trust Company	5-May-17	26-Jul-17
4	First NBC Bank	New Orleans	LA	58302	Whitney Bank	28-Apr-17	5-Dec-17

■ 사례별 시계열 데이터 계산하기

● 파산한 은행의 수 계산

- 5번, 6번 열을 datetime 오브젝트로 변환

```
banks = pd.read_csv('data/banklist.csv', parse_dates=[5, 6])  
print(banks.info())
```

#	Column	Non-Null Count	Dtype
0	Bank Name	555 non-null	object
1	City	555 non-null	object
2	ST	555 non-null	object
3	CERT	555 non-null	int64
4	Acquiring Institution	555 non-null	object
5	Closing Date	555 non-null	datetime64[ns]
6	Updated Date	555 non-null	datetime64[ns]

dtypes: datetime64[ns](2), int64(1), object(4)

■ 사례별 시계열 데이터 계산하기

● 파산한 은행의 수 계산

- dt 접근자와 quarter(분기), year(연도) 사용

```
banks['closing_quarter'] = banks['Closing Date'].dt.quarter  
banks['closing_year'] = banks['Closing Date'].dt.year
```

```
banks.head()
```

	Bank Name	City	ST	CERT	Acquiring Institution	Closing Date	Updated Date	closing_quarter	closing_year
0	Washington Federal Bank for Savings	Chicago	IL	30570	Royal Savings Bank	2017-12-15	2017-12-20	4	2017
1	The Farmers and Merchants State Bank of Argonia	Argonia	KS	17719	Conway Bank	2017-10-13	2017-10-20	4	2017
2	Fayette County Bank	Saint Elmo	IL	1802	United Fidelity Bank, fsb	2017-05-26	2017-07-26	2	2017
3	Guaranty Bank, (d/b/a BestBank in Georgia & Mi...	Milwaukee	WI	30003	First-Citizens Bank & Trust Company	2017-05-05	2017-07-26	2	2017
4	First NBC Bank	New Orleans	LA	58302	Whitney Bank	2017-04-28	2017-12-05	2	2017

■ 사례별 시계열 데이터 계산하기

● 파산한 은행의 수 계산

- 연도별 파산 현황 확인 - 1

```
banks['closing_year'].value_counts().head()
```

```
2010      157
2009      140
2011       92
2012       51
2008       25
Name: closing_year, dtype: int64
```

- 연도별 파산 현황 확인 - 2

```
banks.groupby(['closing_year']).size().head()
```

```
closing_year
2000         2
2001         4
2002        11
2003         3
2004         4
dtype: int64
```

■ 사례별 시계열 데이터 계산하기

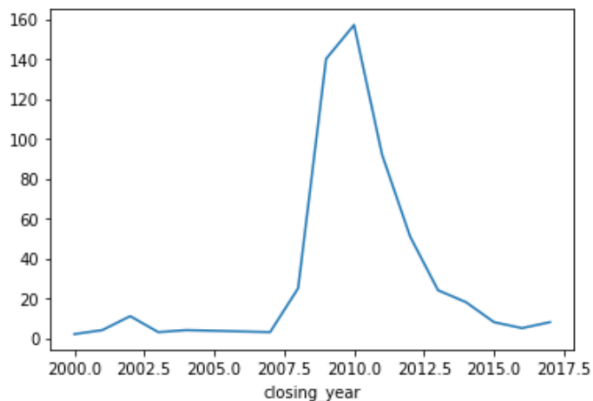
● 파산한 은행의 수 계산

- 연도별 / 분기별 파산 현황 확인

```
closing_year_q = banks.groupby(  
    ['closing_year', 'closing_quarter']).size()  
closing_year_q.head()
```

```
closing_year  closing_quarter  
2000          4                2  
2001          1                1  
              2                1  
              3                2  
2002          1                6  
dtype: int64
```

```
import matplotlib.pyplot as plt  
closing_year.plot()
```



```
import matplotlib.pyplot as plt  
closing_year_q.plot()
```

