

```
1 1. Matplotlib
2 1)주로 2차원의 data를 시각화를 하기 위한 third-party package이다.
3 2)동작하는 OS를 가리지 않는다는 점, 자세한 그리기 설정이 가능한 점, 다양한 출력 형식에 대응하고
  있는 점 등 대표적인 시각화 툴로 널리 사용되고 있다.
4 3)2003년 version 1.0이 발표된 이후로 15년 이상의 역사를 가진 tool이다.
5 4)사용자가 많은 이유는 산업계, 교육계에서 널리 사용되고 있는 수치 해석 S/W로, MATLAB과 같은
  그리기를 Python에서 사용할 수 있는 것이다.
6 5)Matplotlib History :
  http://jakevdp.github.io/blog/2013/03/23/matplotlib-and-the-future-of-visualization-in-python/
7 6)Matplotlib에서는 graph의 종류나 축, 눈금선 graph 이름 등 다양한 그림의 요소에 대해 상세한
  서식(색이나 선 종류 등)을 설정할 수 있다.
8 7)다양한 출력 형식(PNG, SVG, JPG 등)에 대응하고 있다.
9
10 2. Information
11 1)Version : 3.0.2
12 2)Site : https://matplotlib.org
13 3)Repository : https://github.com/matplotlib/matplotlib
14 4)PyPI : https://pypi.python.org/pypi/matplotlib/
15
16 3. Graph 그리기 기초
17 1)Graph 그리기 준비하기
18   -matplotlib.pyplot module을 불러온다.
19
20   import matplotlib.pyplot as plt
21
22   -matplotlib graph를 출력할 때는 show()를 이용한다.
23
24   plt.show()
25
26 4. pandas를 사용한 data의 시각화
27 1)pandas의 Series 또는 DataFrame의 plot()를 사용하여 쉽게 시각화할 수 있다.
28 2)plot()은 내부에서 Matplotlib를 사용하고 있다.
29 3)Notebook에 graph 표시하기
30   -Notebook에 graph를 표시하기 위해서는 pyplot.show()를 사용한다.
31
32   import pandas as pd
33   import matplotlib.pyplot as plt
34
35   ax = pd.Series([1,2,3]).plot()
36   ax.set_title('Line Chart')
37   plt.show()
38   #graph를 그릴 때는 cell을 바꾸기 않고 하나의 cell안에 모두 coding해야 한다.
39
40   column_names = ['Hakbun', 'Name', 'Kor', 'Eng', 'Mat', 'Edp']
41   df = pd.read_csv('pandas_data/sungjuk_utf8.csv', names = column_names)
42   df['Total'] = df['Kor'] + df['Eng'] + df['Mat'] + df['Edp']
43   df['Average'] = df['Total'] / 4
44   grade_list = []
45   for avg in df['Average'] :
46       if 90 <= avg <= 100 : grade_list.append('A')
47       elif 80 <= avg < 90 : grade_list.append('B')
```

```

48         elif 70 <= avg < 80 : grade_list.append('C')
49         elif 60 <= avg < 70 : grade_list.append('D')
50         else : grade_list.append('F')
51     df['Grade'] = grade_list
52     df

```

```

53
54     ax = df['Kor'].plot().set_title('Line Chart')
55     plt.show()
56

```

-Graph의 style을 변경하는 경우에는 pyplot.style.use()의 인수에 style명을 넘긴다.

```

58
59     plt.style.use('ggplot')    #기본은 'default'
60

```

#### 4)DataFrame에서 plot하기

-DataFrame에서 plot()을 호출할 경우 기본적으로 Series와 같은 동작을 수행하는데, 열 수에 상응하는 요소가 그려진다.

-Index가 X값, 이름 열의 값이 Y값이 된다.

```

64
65     df = pd.DataFrame({'a':[1,2,3], 'b':[3,2,1]})
66     ax = df.plot().set_title('Line Chart')
67     plt.show()
68

```

#### 5)Y축 범위가 다른 경우

-Keyword 인수 secondary\_y에 두번째 축이 되는 열 이름을 list형으로 지정한다.

-Y축의 label은 set\_ylabel(), right\_ax.set\_ylabel()의 인수에 각각의 이름을 넘겨준다.

```

72
73     ser1 = df['Hakbun']
74     ser2 = df['Kor']
75     df2 = pd.DataFrame(ser1, columns=['Hakbun'])
76     df2['Kor'] = ser2
77
78     ax = df2.plot(secondary_y=['Kor'])
79     ax.set_title('Two Line Chart')
80     ax.set_ylabel('Hakbun')
81     ax.right_ax.set_ylabel('Kor')
82     plt.show()
83

```

#### 6)산포도 graph 그리기

-plot.scatter()를 사용한다.

-Keyword 인수 x에 X값이 되는 열 이름, keyword 인수 y에 Y 값이 되는 열 이름을 지정한다.

```

87
88     ax = df2.plot.scatter(x='Hakbun', y='Kor')
89     ax.set_title('Scatter')
90     plt.show()
91

```

#### 7)막대 Graph 작성하기

-plot.bar()를 사용한다.

```

94
95     import cx_Oracle
96     conn = cx_Oracle.connect('scott', 'tiger', 'localhost:1521/XE')
97     cursor = conn.cursor()
98     sql = "SELECT empno, ename, job, TO_CHAR(hiredate, 'YYYY'), mgr, sal,

```

```

comm, deptno FROM emp"
99 cursor.execute(sql)
100 emp_list = []
101 for empno, ename, job, hiredate, mgr, sal, comm, deptno in cursor:
102     emp_list.append([empno, ename, job, hiredate, mgr, sal, comm, deptno])
103 columns = ['empno', 'ename', 'job', 'hiredate', 'mgr', 'sal', 'comm', 'deptno']
104 df = pd.DataFrame(emp_list, columns = columns)
105 df
106
107 group_deptno = df.groupby('deptno')
108 group_deptno.groups
109 -----
110 {10: Int64Index([6, 8, 13], dtype='int64'),
111  20: Int64Index([0, 3, 7, 10, 12], dtype='int64'),
112  30: Int64Index([1, 2, 4, 5, 9, 11], dtype='int64')}
113
114 for name, group in group_deptno:
115     print(str(name) + ": " + str(len(group)))
116     print(group)
117     print()
118     -----
119 --
120 10: 3
121      empno  ename  job      hiredate  mgr  sal      comm  deptno
122 6   7782   CLARK  MANAGER    1981      7839.0  2450.0   NaN    10
123 8   7839   KING   PRESIDENT  1981      NaN  5000.0   NaN    10
124 13  7934   MILLER CLERK     1982      7782.0  1300.0   NaN    10
125
126 20: 5
127      empno  ename  job      hiredate  mgr  sal      comm  deptno
128 0   7369   SMITH  CLERK     1980      7902.0   800.0 NaN    20
129 3   7566   JONES  MANAGER    1981      7839.0  2975.0 NaN    20
130 7   7788   SCOTT  ANALYST    1987      7566.0  3000.0 NaN    20
131 10  7876   ADAMS  CLERK     1987      7788.0  1100.0 NaN    20
132 12  7902   FORD   ANALYST    1981      7566.0  3000.0 NaN
133 20
134
135 30: 6
136      empno  ename  job      hiredate  mgr  sal      comm  deptno
137 1   7499   ALLEN  SALESMAN    1981      7698.0  1600.0  300.0
138 30
139 2   7521   WARD   SALESMAN    1981      7698.0  1250.0  500.0
140 30
141 4   7654   MARTIN SALESMAN    1981      7698.0  1250.0 1400.0
142 30
143 5   7698   BLAKE   MANAGER    1981      7839.0  2850.0   NaN
144 30
145 9   7844   TURNER SALESMAN    1981      7698.0  1500.0   0.0
146 30
147 11  7900   JAMES   CLERK     1981      7698.0   950.0   NaN
148 30
149
150

```

```
142     names = []
143     values = []
144     for name, group in group_deptno:
145         names.append(name)
146         values.append(len(group))
147
148     plt.figure(1, figsize=(9, 3))
149     plt.subplot(131)
150     plt.bar(names, values)
151     plt.subplot(132)
152     plt.scatter(names, values)
153     plt.subplot(133)
154     plt.plot(names, values)
155     plt.suptitle('Categorical Plotting')
156     plt.xlabel('Department Number')
157     plt.show()
158
159
160 8)다양한 Graph 그리기
161
162     fig, axs = plt.subplots(2,2, figsize=(5,5))
163     axs[0,0].hist(df['sal'])
164     axs[1,0].scatter(df['empno'], df['sal'])
165     axs[0,1].plot(df['empno'], df['sal'])
166     axs[1,1].hist2d(df['empno'], df['sal'])
```