기술인재를 위한 무한한 기대!

AI 프로그래밍 12

융합학과 권오영
oykwon@koreatech.ac.kr

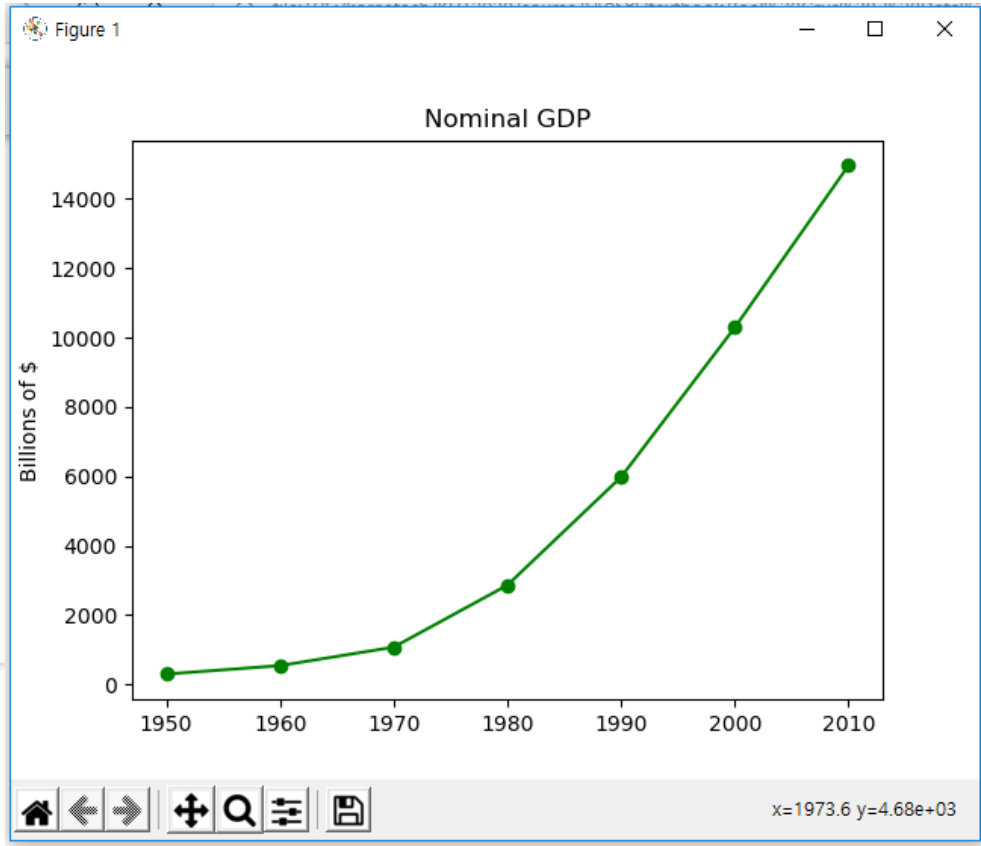
KOREATECH
한국기술교육대학교

시각화

matplotlib

❖ 가장 널리 사용되는 가시화 라이브러리

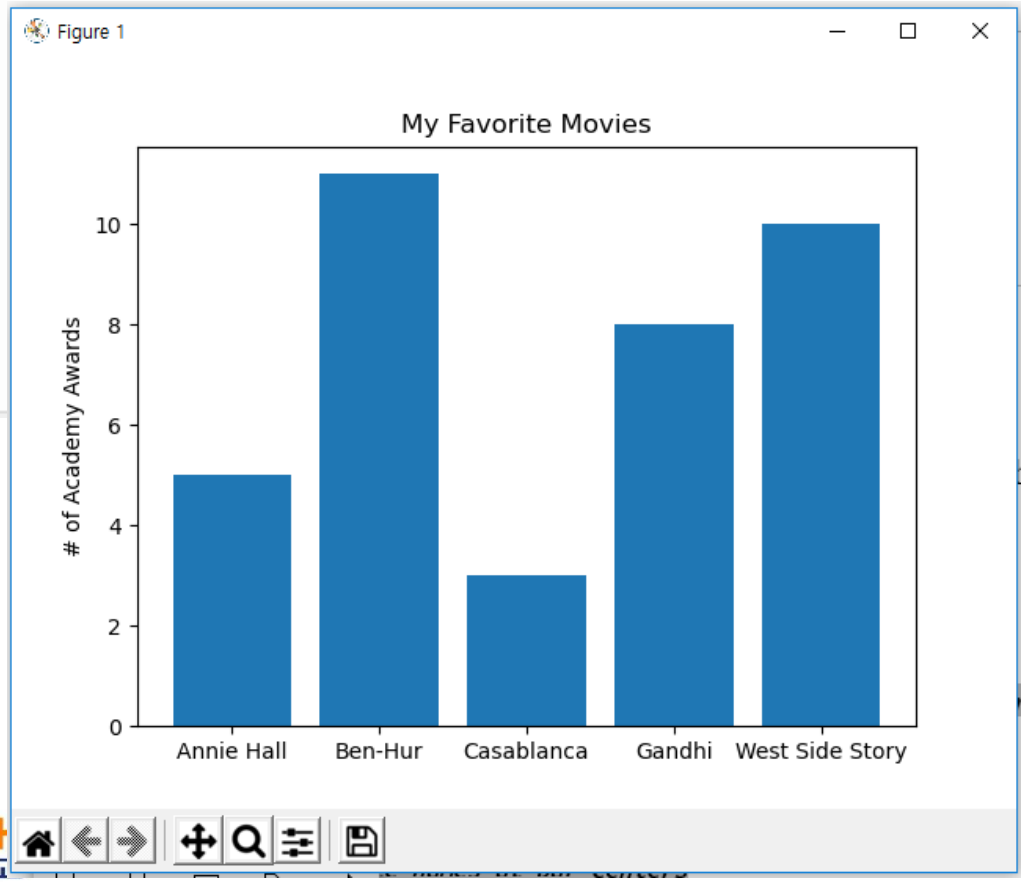
```python
from matplotlib import pyplot as plt
#import matplotlib.pyplot as plt

years = [1950, 1960, 1970, 1980, 1990, 2000, 2010]
gdp = [300.2, 543.3, 1075.9, 2862.5, 5979.6, 10289.7, 14958.3]

# create a line chart, years on x-axis, gdp on y-axis
plt.plot(years, gdp, color='green', marker='o', linestyle='solid')

# add a title
plt.title("Nominal GDP")

# add a label to the y-axis
plt.ylabel("Billions of $")
plt.show()
```



KOREATECH 한국기술교육대학교

Bar Charts

```python
import matplotlib.pyplot as plt

movies = ["Annie Hall", "Ben-Hur", "Casablanca", "Gandhi", "West Side Story"]
num_oscars = [5, 11, 3, 8, 10]

# plot bars with left x-coordinates [0, 1, 2, 3, 4], heights [num_oscars]
plt.bar(range(len(movies)), num_oscars)

plt.title("My Favorite Movies")     # add a title
plt.ylabel("# of Academy Awards")   # label the y-axis

# label x-axis with movie names at bar centers
plt.xticks(range(len(movies)), movies)

plt.show()
```



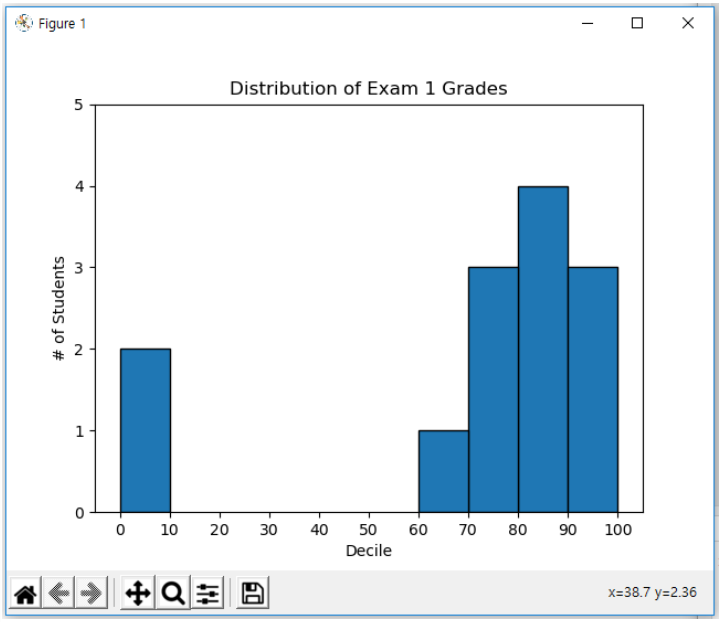
KOREA TECH
한국기술교육대학교

Bar chart (histogram)

```python
from collections import Counter
import matplotlib.pyplot as plt

grades = [83, 95, 91, 87, 70, 0, 85, 82, 100, 67, 73, 77, 0]

# Bucket grades by decile, but put 100 in with the 90s
# 점수구간의 빈도수구함 (Counter 이용)
histogram = Counter(min(grade // 10 * 10, 90) for grade in grades)

print(histogram)

plt.bar([x + 5 for x in histogram.keys()],  # Shift bars right by 5
        histogram.values(),                 # Give each bar its correct height
        10,                                 # Give each bar a width of 10
        edgecolor=(0, 0, 0))                # Black edges for each bar

plt.axis([-5, 105, 0, 5])                   # x-axis from -5 to 105,
                                            # y-axis from 0 to 5

plt.xticks([10 * i for i in range(11)])     # x-axis labels at 0, 10, ..., 100
plt.xlabel("Decile")
plt.ylabel("# of Students")
plt.title("Distribution of Exam 1 Grades")
plt.show()
```

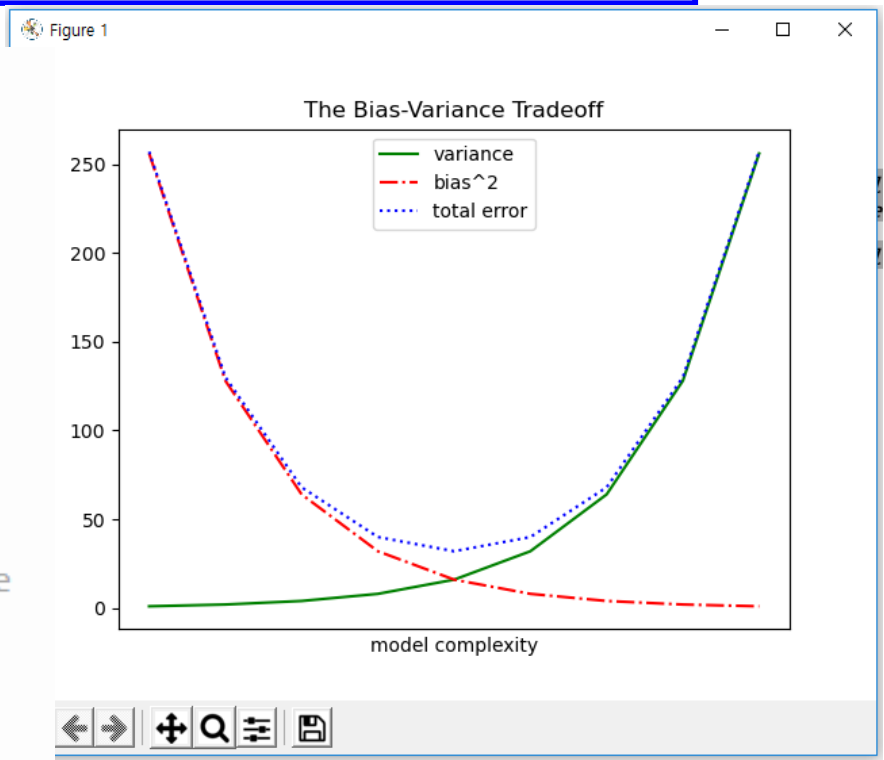
Shell

```
>>> %Run vis-3.py
  Counter({80: 4, 90: 3, 70: 3, 0: 2, 60: 1})
```



Line Charts

```python
import matplotlib.pyplot as plt

variance     = [1, 2, 4, 8, 16, 32, 64, 128, 256]
bias_squared = [256, 128, 64, 32, 16, 8, 4, 2, 1]

total_error  = [x + y for x, y in zip(variance, bias_squared)]
xs = [i for i, _ in enumerate(variance)]

# We can make multiple calls to plt.plot
# to show multiple series on the same chart
plt.plot(xs, variance,     'g-',  label='variance')    # green solid line
plt.plot(xs, bias_squared, 'r-.', label='bias^2')      # red dot-dashed line
plt.plot(xs, total_error,  'b:',  label='total error') # blue dotted line

# Because we've assigned labels to each series,
# we can get a legend for free (loc=9 means "top center")
plt.legend(loc=9)
plt.xlabel("model complexity")
plt.xticks([])
plt.title("The Bias-Variance Tradeoff")
plt.show()
```

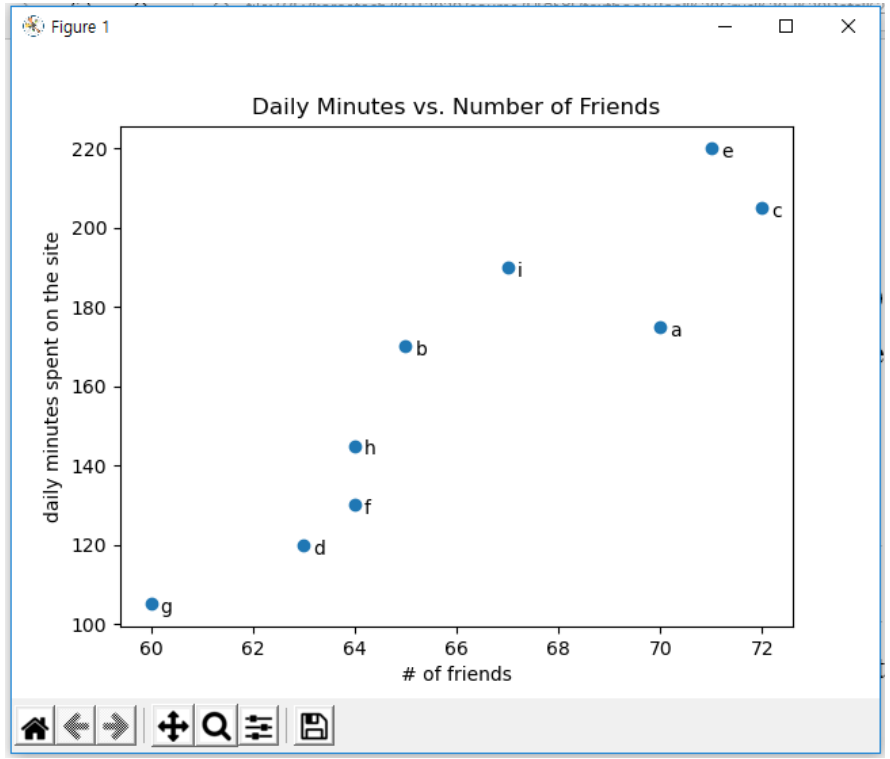


KOREATECH
한국기술교육대학교

Scatterplots

❖ 쌍으로 이루어진 데이터의 관계를 보여주는데 적합함

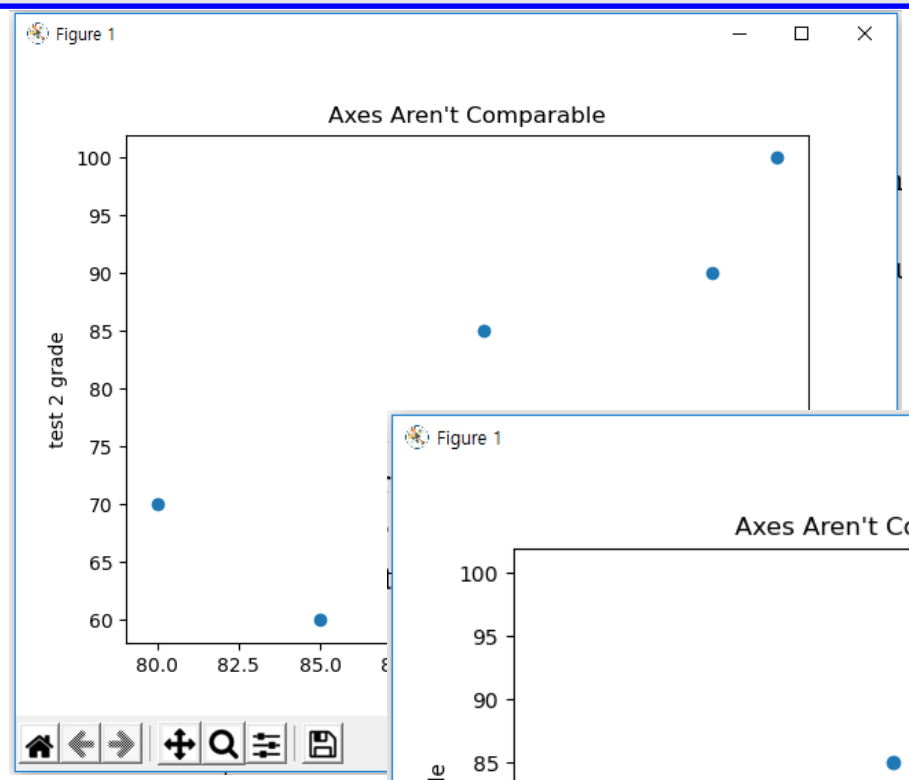
```python
import matplotlib.pyplot as plt

friends = [ 70,  65,  72,  63,  71,  64,  60,  64,  67]
minutes = [175, 170, 205, 120, 220, 130, 105, 145, 190]
labels =  ['a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i']

plt.scatter(friends, minutes)

# label each point
for label, friend_count, minute_count in zip(labels, friends, minutes):
    plt.annotate(label,
        xy=(friend_count, minute_count), # Put the label with its point
        xytext=(5, -5),                  # but slightly offset
        textcoords='offset points')

plt.title("Daily Minutes vs. Number of Friends")
plt.xlabel("# of friends")
plt.ylabel("daily minutes spent on the site")
plt.show()
```



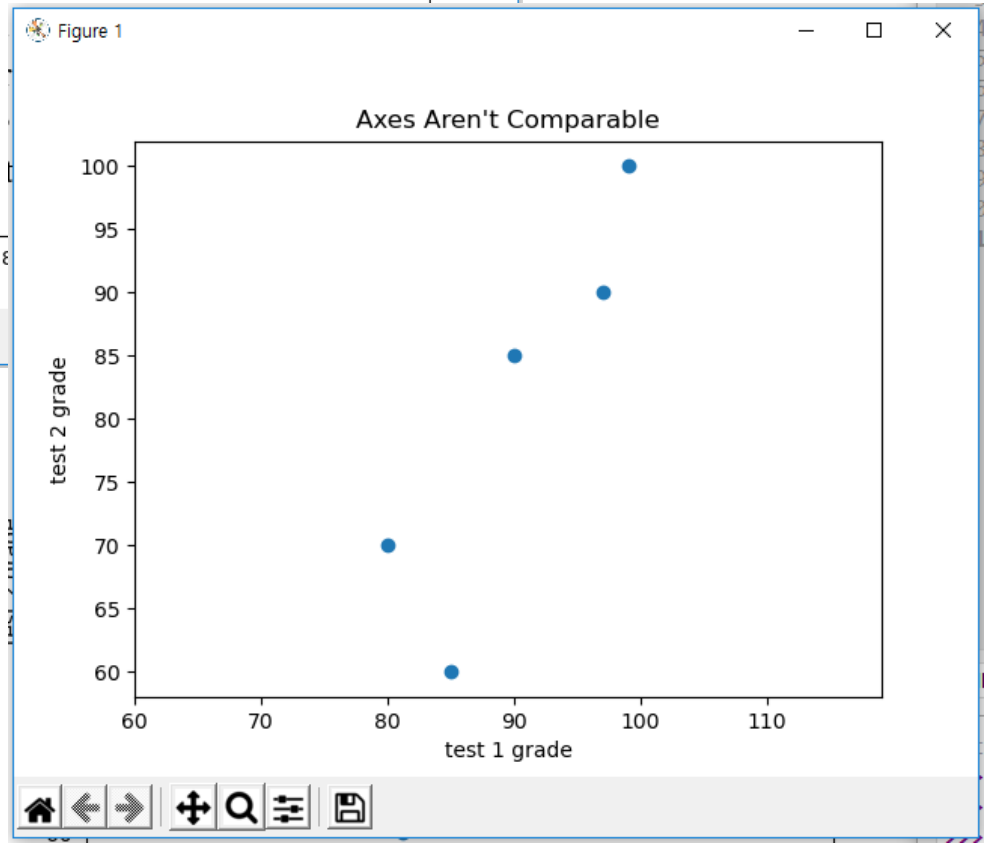
KOREATECH 한국기술교육대학교

Scatterplot

```python
import matplotlib.pyplot as plt

test_1_grades = [ 99, 90, 85, 97, 80]
test_2_grades = [100, 85, 60, 90, 70]

plt.scatter(test_1_grades, test_2_grades)
plt.title("Axes Aren't Comparable")
plt.xlabel("test 1 grade")
plt.ylabel("test 2 grade")
plt.show()
```




```python
import matplotlib.pyplot as plt

test_1_grades = [ 99, 90, 85, 97, 80]
test_2_grades = [100, 85, 60, 90, 70]

plt.scatter(test_1_grades, test_2_grades)
plt.title("Axes Aren't Comparable")
plt.xlabel("test 1 grade")
plt.ylabel("test 2 grade")
plt.axis("equal")
plt.show()
```



KOREATECH 한국기술교육대학교

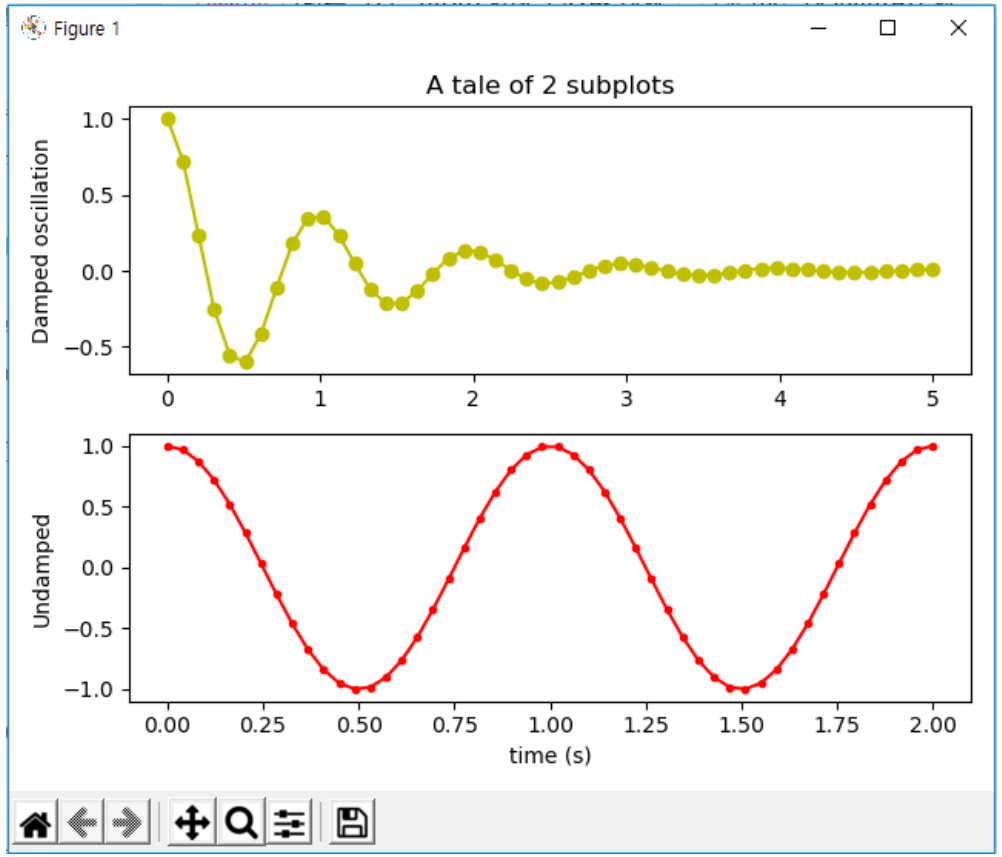
가시화

- matplotlib galley (https://matplotlib.org/gallery.html) : 참고
- seaborn (https://seaborn.pydata.org/)
- Altair (https://altair-viz.github.io/)
- Bokeh (https://docs.bokeh.org/en/latest/)

https://datascienceschool.net/intro.html

Subplot 사용

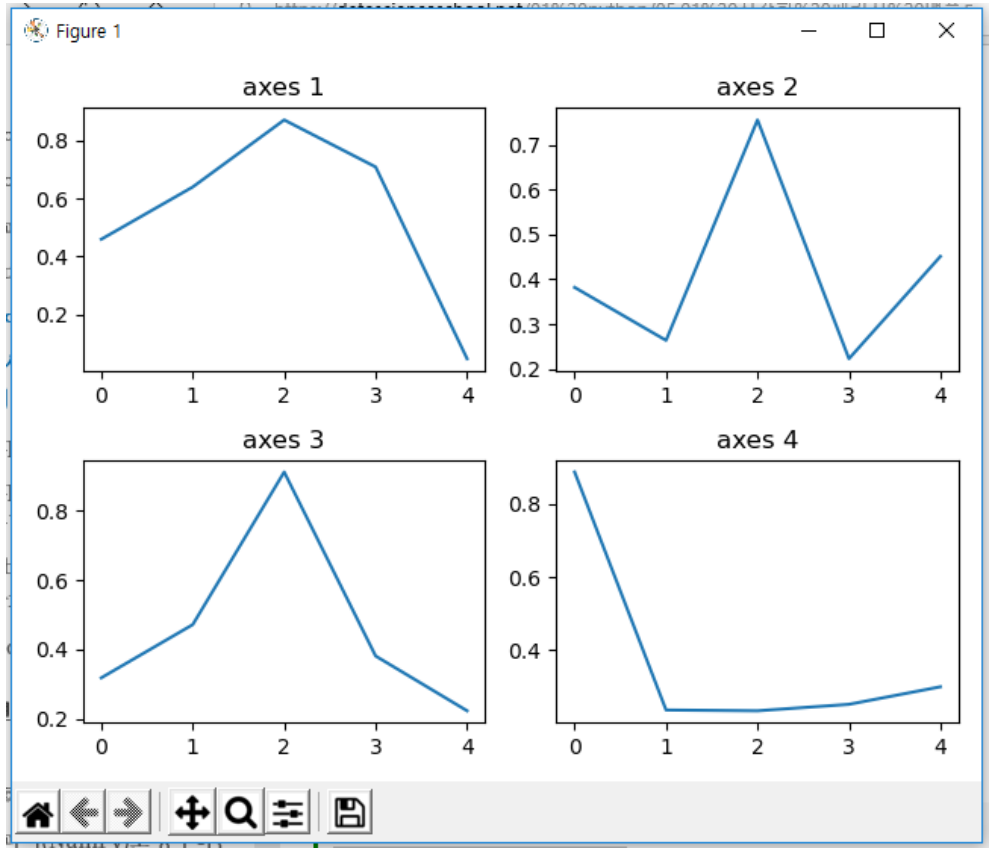
```python
import matplotlib.pyplot as plt
import numpy as np

x1 = np.linspace(0.0, 5.0)
x2 = np.linspace(0.0, 2.0)
y1 = np.cos(2 * np.pi * x1) * np.exp(-x1)
y2 = np.cos(2 * np.pi * x2)

ax1 = plt.subplot(2, 1, 1)  # 2x1 1 (위)
plt.plot(x1, y1, 'yo-')
plt.title('A tale of 2 subplots')
plt.ylabel('Damped oscillation')

ax2 = plt.subplot(2, 1, 2)  # 2x1 2 (아래)
plt.plot(x2, y2, 'r.-')
plt.xlabel('time (s)')
plt.ylabel('Undamped')

plt.tight_layout()
plt.show()
```



KOREATECH 한국기술교육대학교

subplots

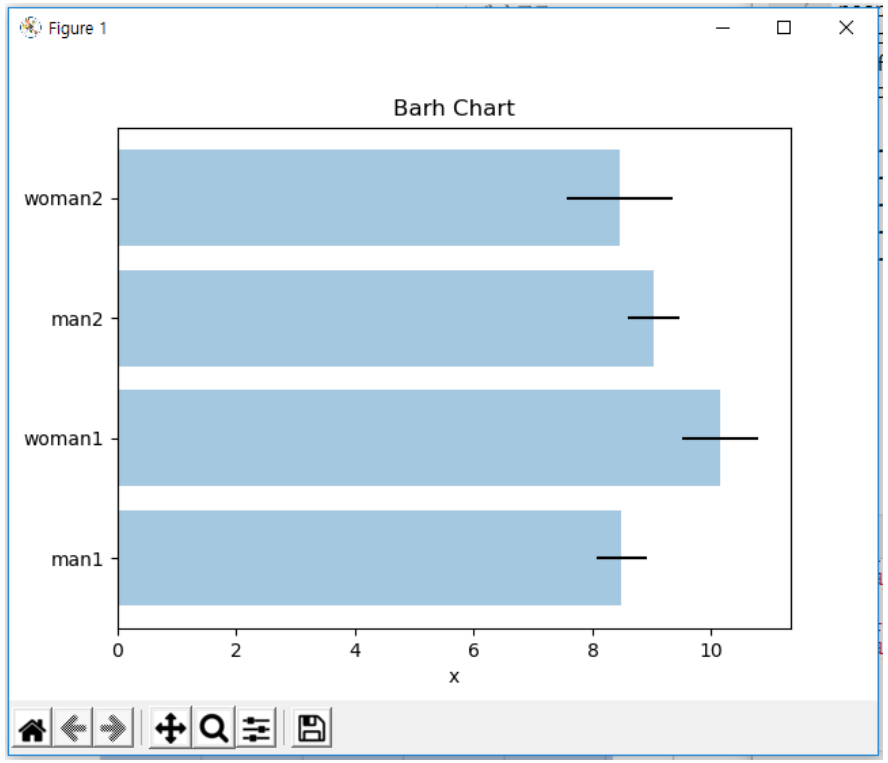
```python
import matplotlib.pyplot as plt
import numpy as np

plt.subplot(221)
plt.plot(np.random.rand(5))
plt.title("axes 1")

plt.subplot(222)
plt.plot(np.random.rand(5))
plt.title("axes 2")

plt.subplot(223)
plt.plot(np.random.rand(5))
plt.title("axes 3")

plt.subplot(224)
plt.plot(np.random.rand(5))
plt.title("axes 4")

plt.tight_layout()
plt.show()
```



KOREATECH
한국기술교육대학교

Barh 차트

```python
import matplotlib.pyplot as plt
import numpy as np

np.random.seed(0)

people = ['man1', 'woman1', 'man2', 'woman2']
y_pos = np.arange(len(people))
performance = 3 + 10 * np.random.rand(len(people))
error = np.random.rand(len(people))

plt.title("Barh Chart")
plt.barh(y_pos, performance, xerr=error, alpha=0.4)
plt.yticks(y_pos, people)
plt.xlabel('x')
plt.show()
```

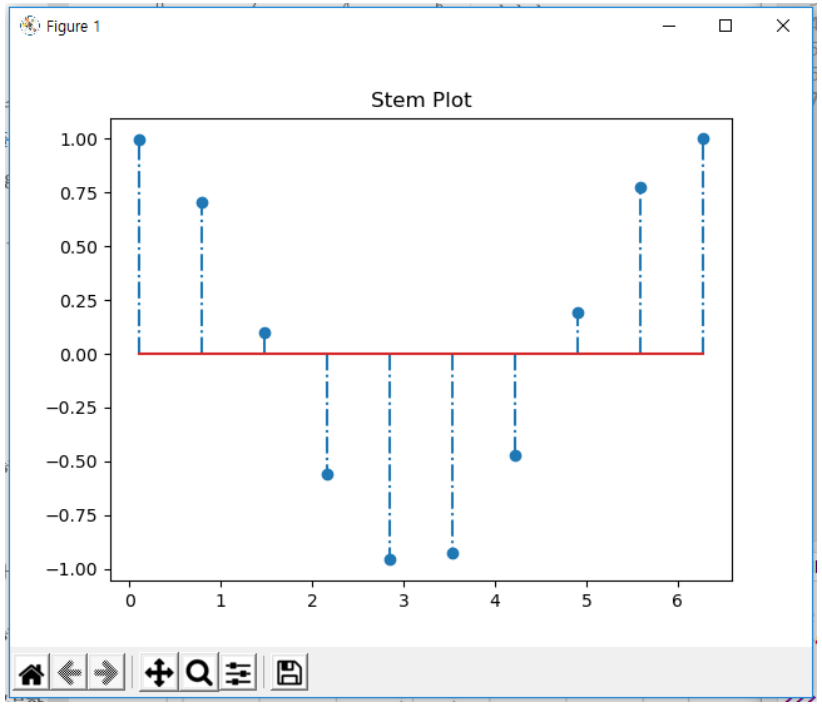


KOREATECH
한국기술교육대학교

스템플롯

❖ 폭이 없는 바차트

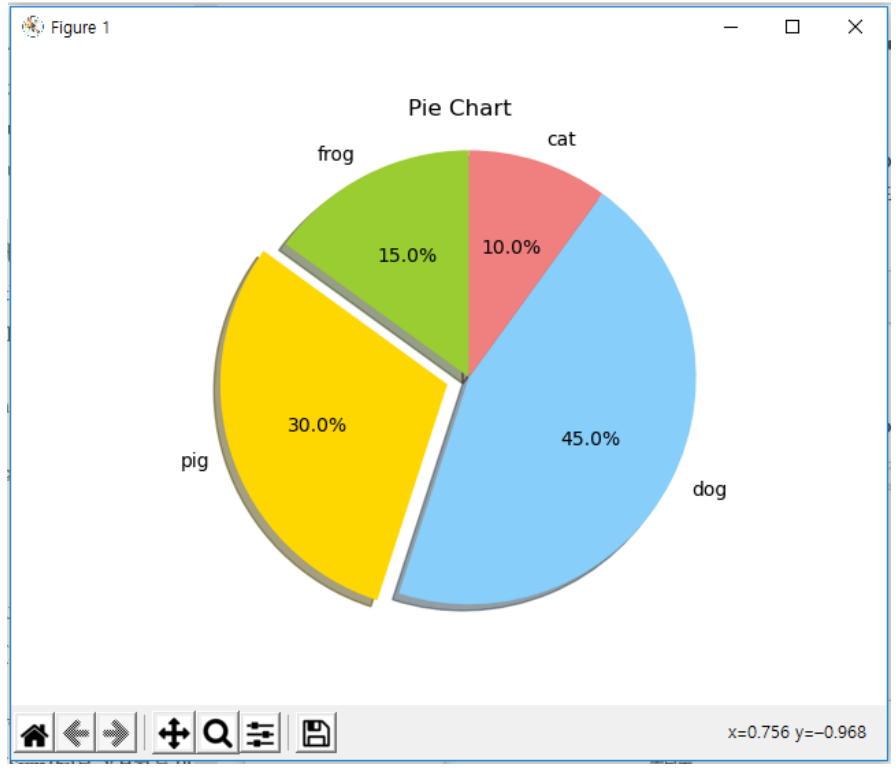
```
1  import matplotlib.pyplot as plt
2  import numpy as np
3
4  x = np.linspace(0.1, 2 * np.pi, 10)
5  plt.title("Stem Plot")
6  plt.stem(x, np.cos(x), '-.')
7  plt.show()
```



KOREATECH 한국기술교육대학교

파이차트

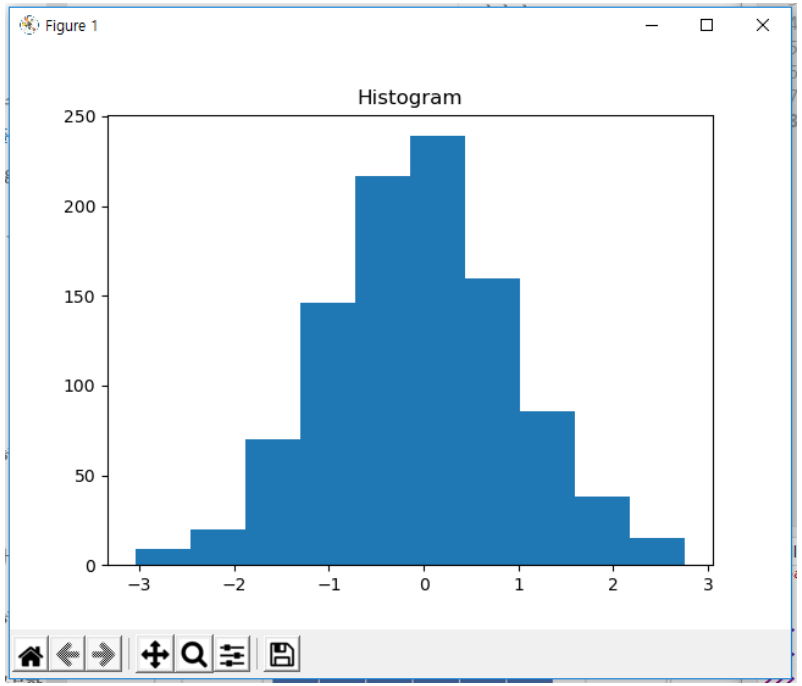
```python
import matplotlib.pyplot as plt
import numpy as np

labels = ['frog', 'pig', 'dog', 'cat']
sizes = [15, 30, 45, 10]
colors = ['yellowgreen', 'gold', 'lightskyblue', 'lightcoral']
explode = (0, 0.1, 0, 0)
plt.title("Pie Chart")
plt.pie(sizes, explode=explode, labels=labels, colors=colors,
        autopct='%1.1f%%', shadow=True, startangle=90)
plt.axis('equal')
plt.show()
```



KOREATECH 한국기술교육대학교

Hist (histogram)

```python
import matplotlib.pyplot as plt
import numpy as np

np.random.seed(0)
x = np.random.randn(1000)
plt.title("Histogram")
arrays, bins, patches = plt.hist(x, bins=10)
plt.show()
```



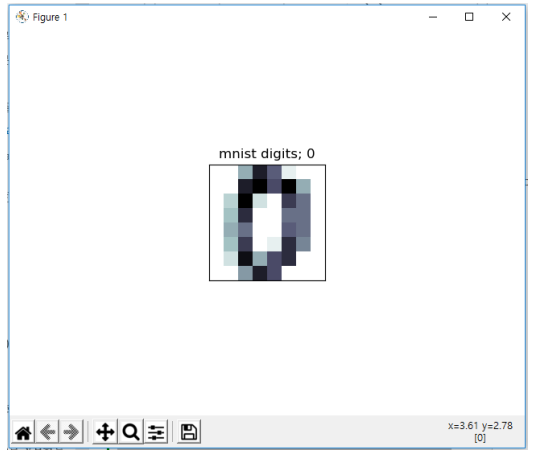
KOREATECH
한국기술교육대학교

이미지 출력(비트맵)

```python
import matplotlib.pyplot as plt
import numpy as np

from sklearn.datasets import load_digits

digits = load_digits()
X = digits.images[0]
print(X)

plt.title("mnist digits; 0")
plt.imshow(X, interpolation='nearest', cmap=plt.cm.bone_r)
plt.xticks([])
plt.yticks([])
plt.grid(False)
plt.subplots_adjust(left=0.35, right=0.65, bottom=0.35, top=0.65)
plt.show()
```

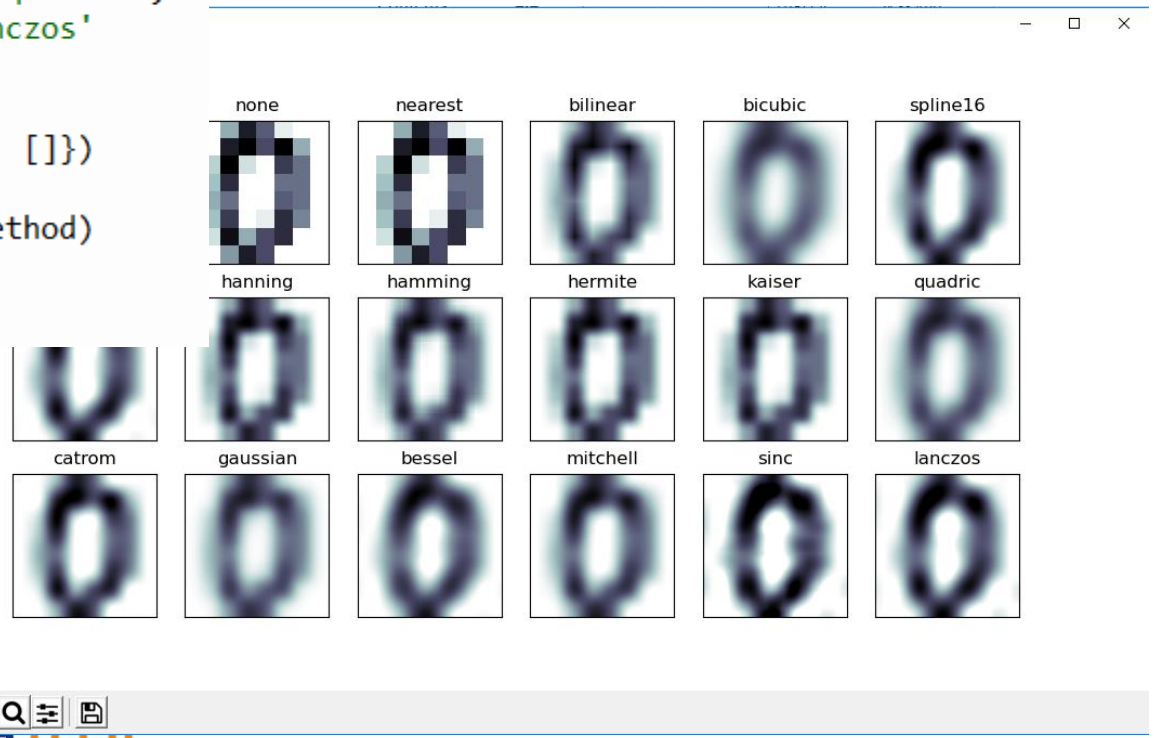
Shell

```
>>> %Run vis-14.py
  [[ 0.  0.  5. 13.  9.  1.  0.  0.]
   [ 0.  0. 13. 15. 10. 15.  5.  0.]
   [ 0.  3. 15.  2.  0. 11.  8.  0.]
   [ 0.  4. 12.  0.  0.  8.  8.  0.]
   [ 0.  5.  8.  0.  0.  9.  8.  0.]
   [ 0.  4. 11.  0.  1. 12.  7.  0.]
   [ 0.  2. 14.  5. 10. 12.  0.  0.]
   [ 0.  0.  6. 13. 10.  0.  0.  0.]]
```



여러 이미지 출력

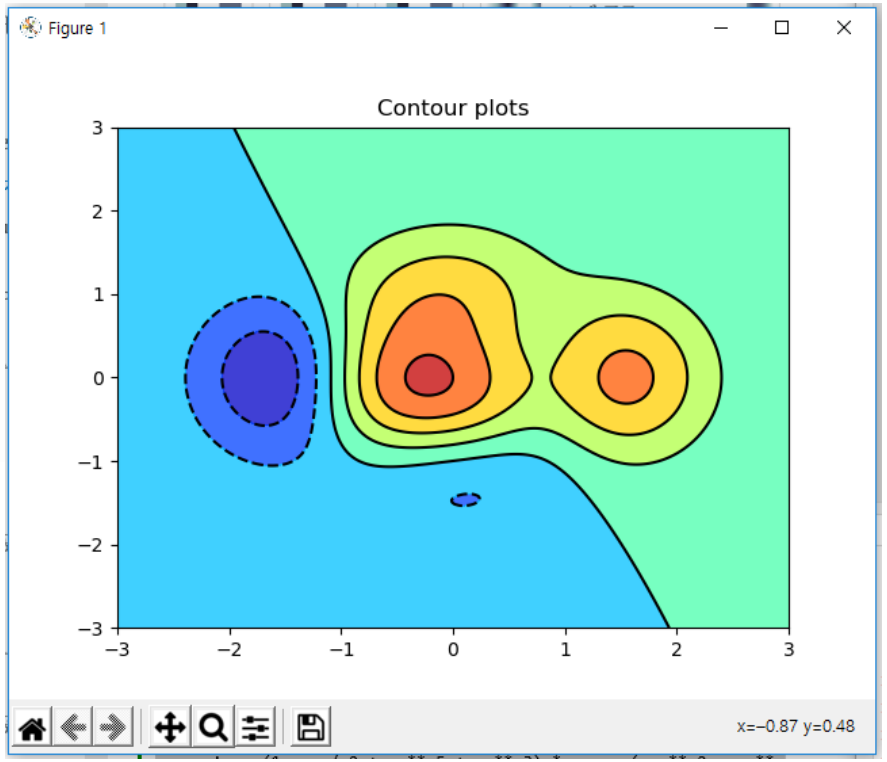
```python
import matplotlib.pyplot as plt
import numpy as np

from sklearn.datasets import load_digits

digits = load_digits()
X = digits.images[0]

methods = [
    None, 'none', 'nearest', 'bilinear', 'bicubic', 'spline16',
    'spline36', 'hanning', 'hamming', 'hermite', 'kaiser', 'quadric',
    'catrom', 'gaussian', 'bessel', 'mitchell', 'sinc', 'lanczos'
]
fig, axes = plt.subplots(3, 6, figsize=(12, 6),
                         subplot_kw={'xticks': [], 'yticks': []})
for ax, interp_method in zip(axes.flat, methods):
    ax.imshow(X, cmap=plt.cm.bone_r, interpolation=interp_method)
    ax.set_title(interp_method)
plt.show()
```



한국기술교육대학교

등고선 출력

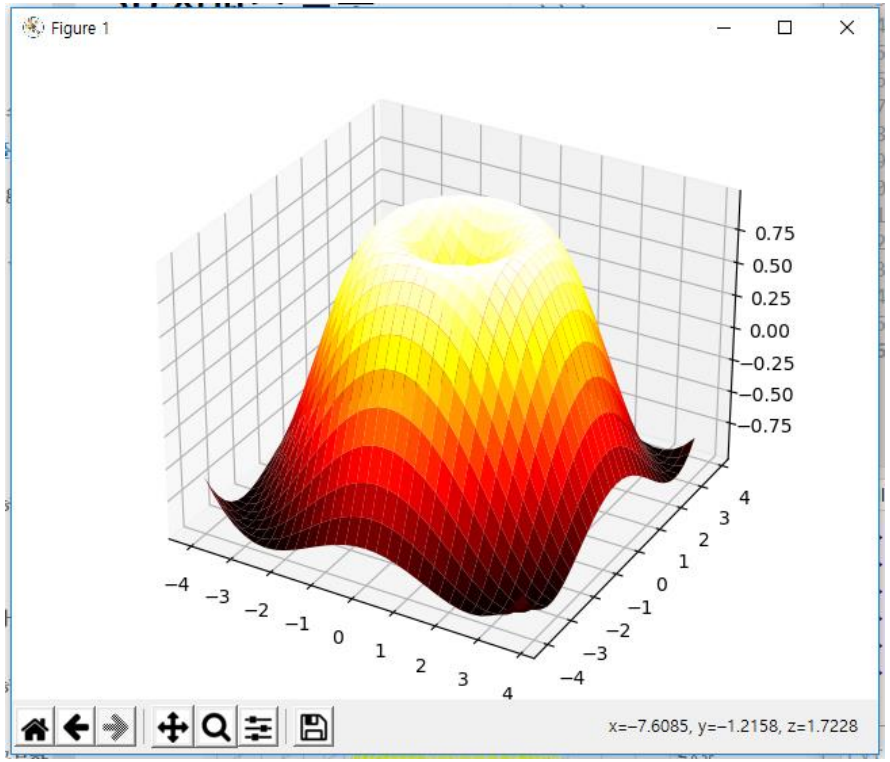
```python
import matplotlib.pyplot as plt
import numpy as np

def f(x, y):
    return (1 - x / 2 + x ** 5 + y ** 3) * np.exp(-x ** 2 - y ** 2)

n = 256
x = np.linspace(-3, 3, n)
y = np.linspace(-3, 3, n)
XX, YY = np.meshgrid(x, y)
ZZ = f(XX, YY)

plt.title("Contour plots")
plt.contourf(XX, YY, ZZ, alpha=.75, cmap='jet') # 색깔표시
plt.contour(XX, YY, ZZ, colors='black')         # 등고선만 표시
plt.show()
```



KOREATECH 한국기술교육대학교

3차원 서비스 플롯

```python
import matplotlib.pyplot as plt
import numpy as np

from mpl_toolkits.mplot3d import Axes3D

X = np.arange(-4, 4, 0.25)
Y = np.arange(-4, 4, 0.25)
XX, YY = np.meshgrid(X, Y)
RR = np.sqrt(XX**2 + YY**2)
ZZ = np.sin(RR)

fig = plt.figure()
ax = Axes3D(fig)
ax.set_title("3D Surface Plot")
ax.plot_surface(XX, YY, ZZ, rstride=1, cstride=1, cmap='hot')
plt.show()
```



KOREATECH 한국기술교육대학교

가시화라이브러리 seaborn

❖ 라이브러리 seaborn 이고 약칭으로 sns을 사용한다.

```
import seaborn as sns
```

❖ 일부 데이터를 포함

- 타이타닉
- 붓꽃데이터

sepallength (꽃받침 길이)	spealWidth (꽃받침 너비)	petalLength (꽃잎 길이)	petalWidth (꽃잎 너비)	species (꽃 종류)
5.1	3.5	1.4	0.2	Iris-setosa
4.9	3	1.4	0.2	Iris-setosa
4.7	3.2	1.3	0.2	Iris-setosa
4.6	3.1	1.5	0.2	Iris-setosa
5	3.6	1.4	0.2	Iris-setosa
5.4	3.9	1.7	0.4	Iris-setosa
4.6	3.4	1.4	0.3	Iris-setosa

예제

```python
import matplotlib.pyplot as plt
import seaborn as sns

iris = sns.load_dataset('iris')

sns.jointplot(x="sepal_length", y="sepal_width", data=iris)
plt.suptitle("Joint Plot of sepal length and width")
plt.show()

sns.jointplot(x="sepal_length", y="sepal_width", data=iris, kind="kde")
plt.suptitle("Joint Plot and Kernel Density Plot of sepal length and width")
plt.show()
```

```python
import matplotlib.pyplot as plt
import seaborn as sns

iris = sns.load_dataset('iris')

sns.pairplot(iris)
plt.title("Pair Plot of Iris Data")
plt.show()

sns.pairplot(iris, hue="species", markers=["o", "s", "D"])
plt.title("Iris Pair Plot")
plt.show()
```

지도그리기

- 지도를 지원하는 folium 라이브러리 확인
- 맵을 html 파일로 저장해야하는 불편함이 있음

```
1  import folium
2
3  seoul_map = folium.Map(location=[37.55,126.98], zoom_start=12)
4
5  seoul_map.save('./seoul.html')
6
7  cheonan_map = folium.Map(location=[36.815,127.113], zoom_start=12)
8
9  cheonan_map.save('./cheonan.html')
```

지도그리기

❖ 스타일 변경: tiles 옵션사용
'Stamen Terrain': 지형을 강조
'Stamen Toner': 도로망을 강조

```
1  import folium
2
3  cheonan_map1 = folium.Map(location=[36.815,127.113],
4                            tiles='Stamen Terrain', zoom_start=12)
5
6  cheonan_map2 = folium.Map(location=[36.815,127.113],
7                            tiles='Stamen Toner', zoom_start=12)
8
9  cheonan_map1.save('./cheonan1.html')
10 cheonan_map2.save('./cheonan2.html')
```

위치표시하기

```python
import pandas as pd
import folium

data = [['Koreatech', 36.763, 127.281],
        ['Baekseok',  36.84,  127.185],
        ['Dankook',   36.837, 127.168],
        ['Hoseo',     36.828, 127.183],
        ['Sunmoon',   36.801, 127.077],
        ['Soonchunhyang', 36.802, 127.134]]

df = pd.DataFrame(data, columns=['univ', 'lat', 'lng'])
print(df)

cheonan_map_univ = folium.Map(location=[36.815,127.113],
                              tiles='Stamen Terrain', zoom_start=12)

for name, lat, lng in zip(df.univ, df.lat, df.lng):
    folium.Marker([lat, lng], popup=name).add_to(cheonan_map_univ)

cheonan_map_univ.save('./cheonanU.html')
```

Shell

```
>>> %Run vis-map3.py
            univ     lat      lng
0      Koreatech  36.763  127.281
1       Baekseok  36.840  127.185
2        Dankook  36.837  127.168
3          Hoseo  36.828  127.183
4        Sunmoon  36.801  127.077
5  Soonchunhyang  36.802  127.134
```