

병아리 데이터 chick001 chick002로 그래프 그려보기

1. chick001.csv 데이터 읽어 와서 기본통계량 점검하기

```
import pandas as pd
```

```
import seaborn as sns
```

```
hat = pd.read_csv('data/chick001.csv')
```

```
print(hat)
```

```
print(" ")
```

```
#-----
```

	<u>hatchery</u>	<u>chick</u>
0	A	30
1	B	30
2	C	29
3	D	26
4	E	24
5	F	28
6	G	27

--> 부화장 7개의 닭들 자료이다.

2. 기술 통계량

```
print(hat['chick'].sum())
```

```
print(hat['chick'].mean())
```

```
print(hat['chick'].std())
```

```
print(hat['chick'].median())
```

```
print(hat['chick'].min())
```

```
print(hat['chick'].max())
```

```
print(" ")
```

194

27.714285714285715

2.2146697055682827

28.0

24

30

3. 데이터 내용

```
print(hat.describe())
```

```
      chick
count  7.000000
mean   27.714286
std    2.214670
min    24.000000
25%    26.500000
50%    28.000000
75%    29.500000
max    30.000000
```

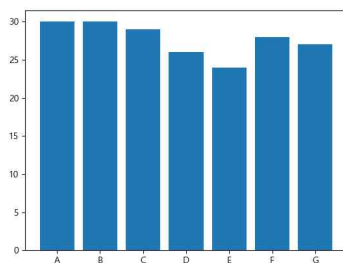
4. 내림차순으로 정렬

```
hat1 = hat.sort_values(by=['chick'], ascending=True)
hat1 = hat.sort_values(by=['chick'], ascending=False)
print(hat1)
```

```
   hatchery  chick
0         A     30
1         B     30
2         C     29
5         F     28
6         G     27
3         D     26
4         E     24
```

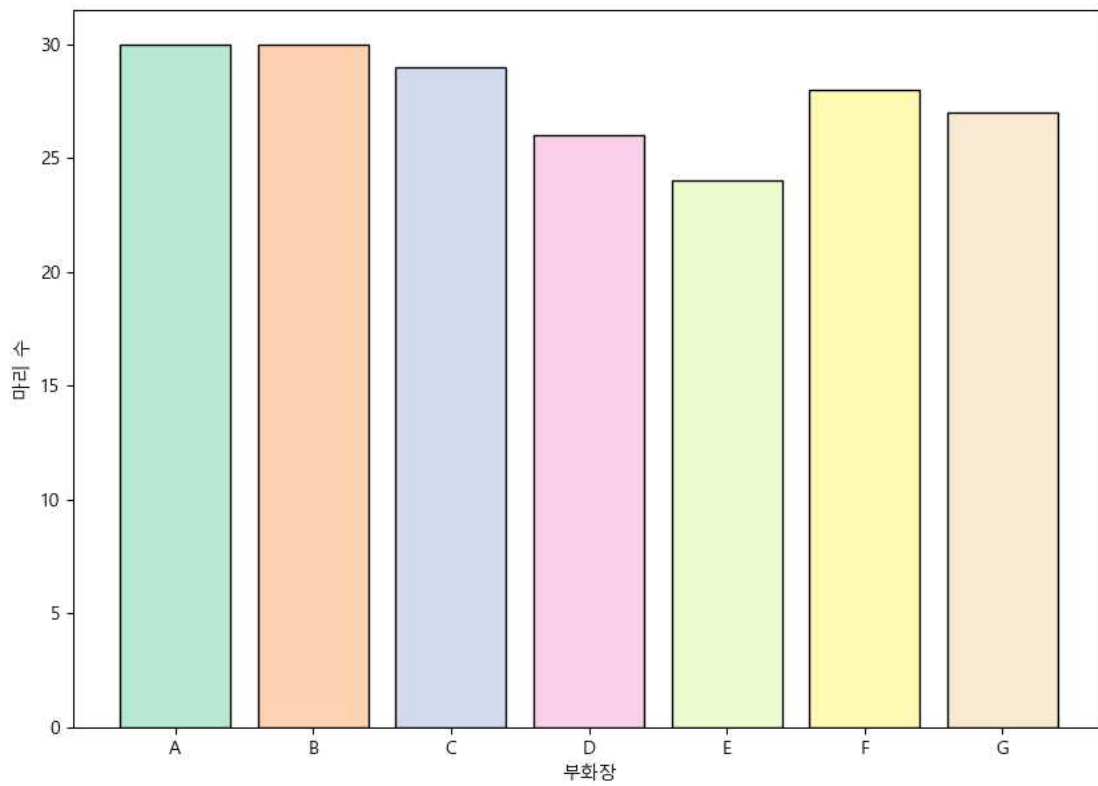
5. 막대그래프 그리기

```
import matplotlib.pyplot as plt
print(" ")
plt.rc('font', family='Malgun Gothic')
plt.rcParams['axes.unicode_minus'] = False # 축의 음수 기호 표시
plt.bar(hat['hatchery'], hat['chick'])
plt.show()
print("-----")
```



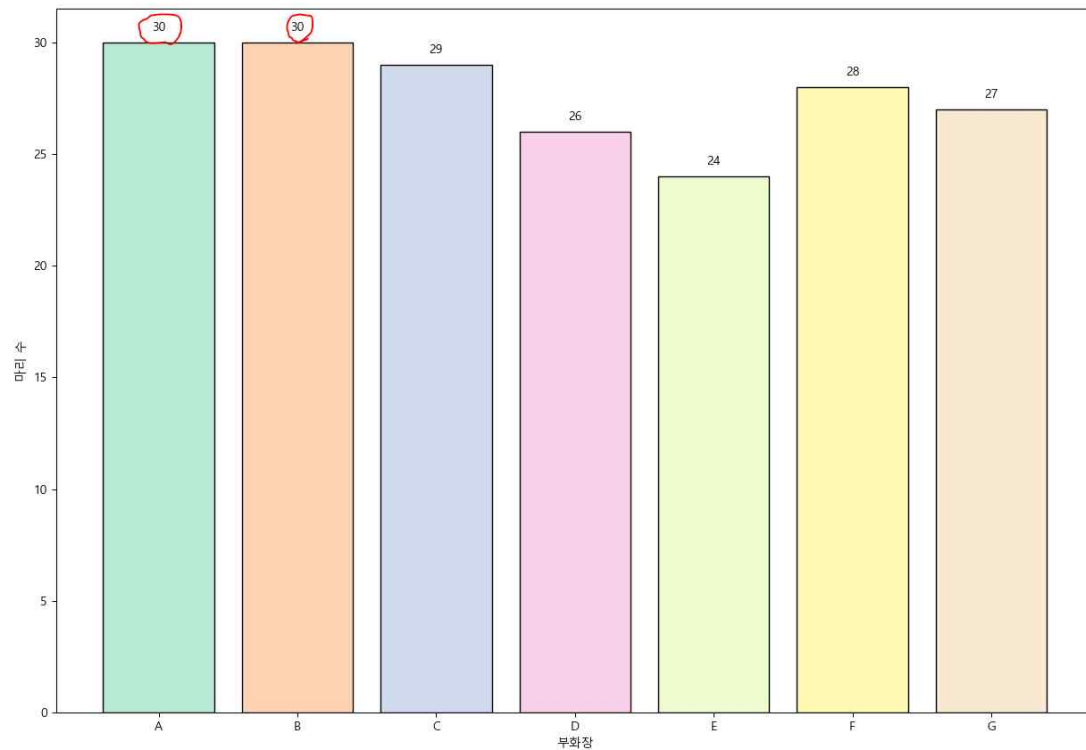
6. seaborn으로 색을 넣어서 바차트 그리기

```
import seaborn as sns
col7 = sns.color_palette('Pastel2', 7)
plt.figure(figsize=(10, 7))
plt.bar(hat['hatchery'], hat['chick'], color=col7, edgecolor='black')
plt.xlabel('부화장')
plt.ylabel('마리 수')
plt.show()
print("-----")
```



7. 바차트에 숫자 넣기

```
def addtext(x, y):
    for i in range(len(x)):
        plt.text(i, y[i] + 0.5, y[i], ha='center')
col7 = sns.color_palette('Pastel2', 7)
plt.figure(figsize=(15, 12))
plt.bar(hat['hatchery'], hat['chick'], color=col7, edgecolor='black')
addtext(hat['hatchery'], hat['chick'])
plt.xlabel('부화장')
plt.ylabel('마리 수')
plt.show()
print("-----")
```



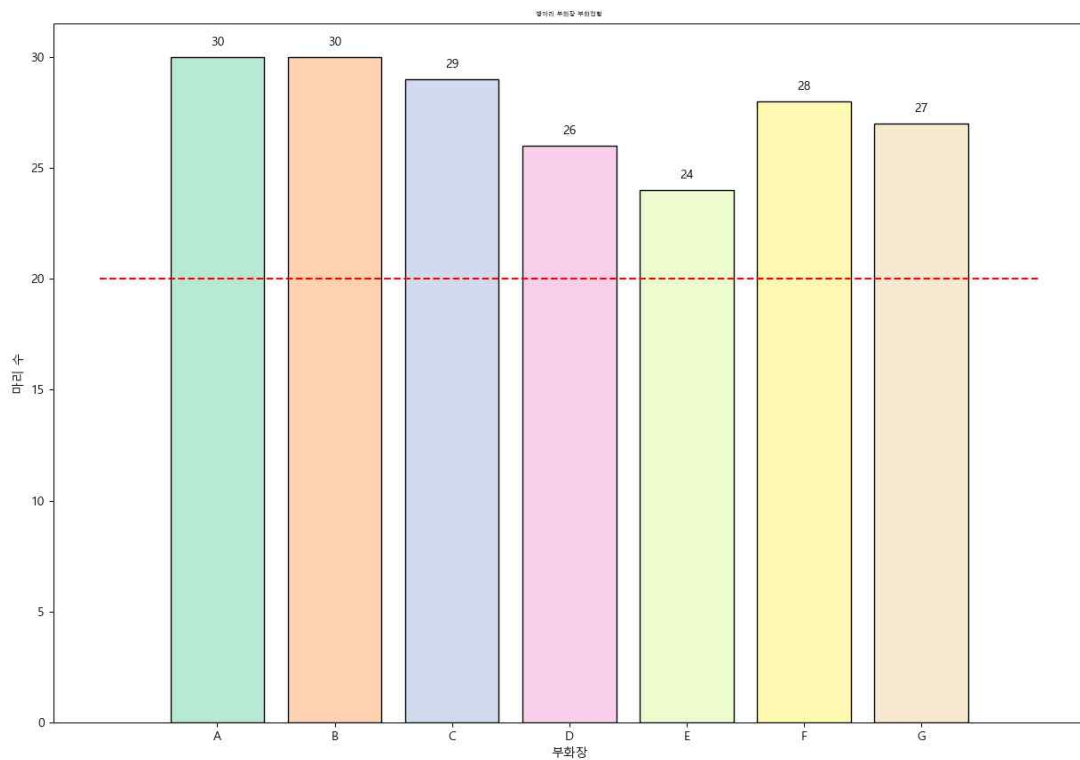
- > $y[i] + 0.5$ 를 변경하면 숫자의 위치가 달라진다.
- > $y[i] - 0.5$ 기둥안으로 들어가고
- > $y[i]/2$ 로 하면 기둥 중간에 숫자가 들어간다.

8. 기둥그래프에 가로선 그어주기

```
def addtext(x, y):
    for i in range(len(x)):
        plt.text(i, y[i] + 0.5, y[i], ha='center')

col7 = sns.color_palette('Pastel2', 7)
plt.figure(figsize=(15, 12))
plt.bar(hat['hatchery'], hat['chick'], color=col7, edgecolor='black')
addtext(hat['hatchery'], hat['chick'])

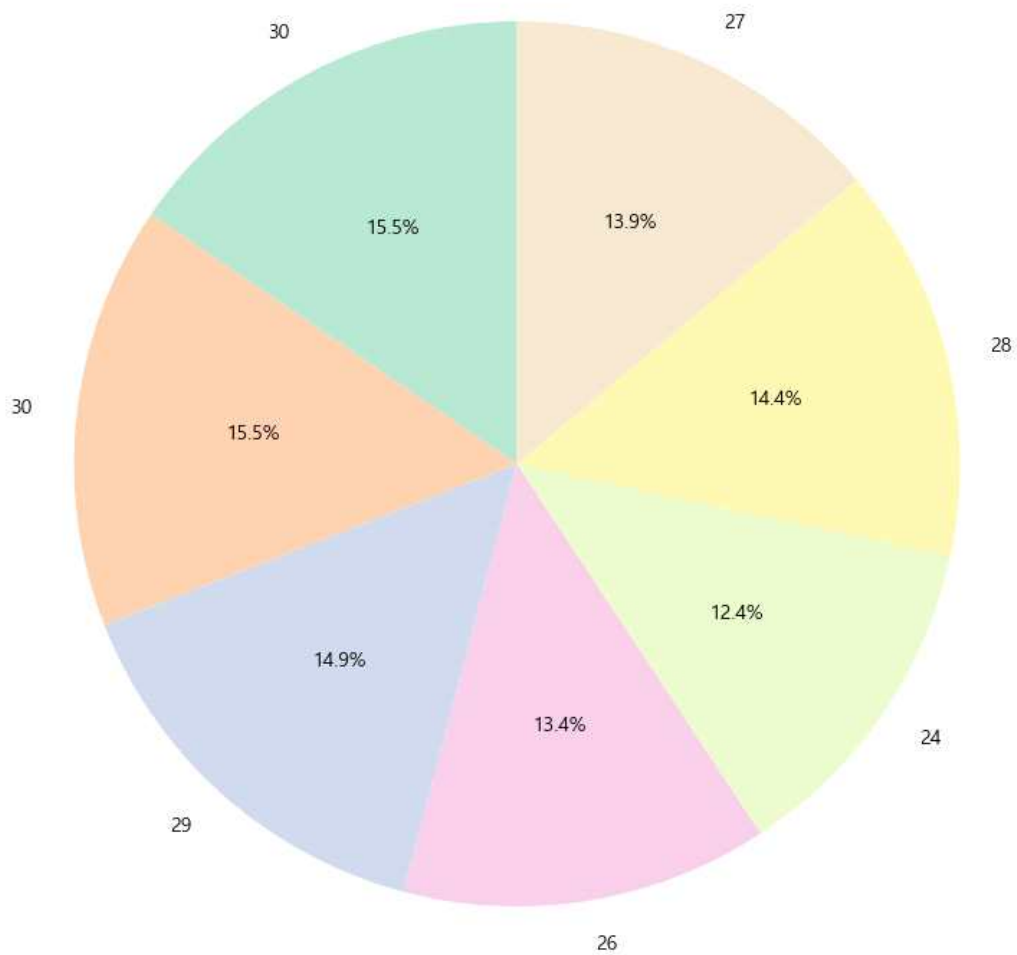
plt.hlines(20, -1, 7, colors='red', linestyle='dashed')
plt.title('병아리 부화장 부화현황', fontsize=5)
plt.xlabel('부화장')
plt.ylabel('마리 수')
plt.show()
print("-----")
```



--> plt.hlines(20, -1, 7을 변경하면 선의 위치 (20->30), 시작점(0->1) 끝점(7->6)

9. 파이차트 그리기

```
col7 = sns.color_palette('Pastel2', 7)
plt.figure(figsize=(10, 10))
plt.pie(hat['chick'], labels=hat['chick'], colors=col7, autopct='%0.1f%%', startangle=90)
plt.show()
print("-----")
```



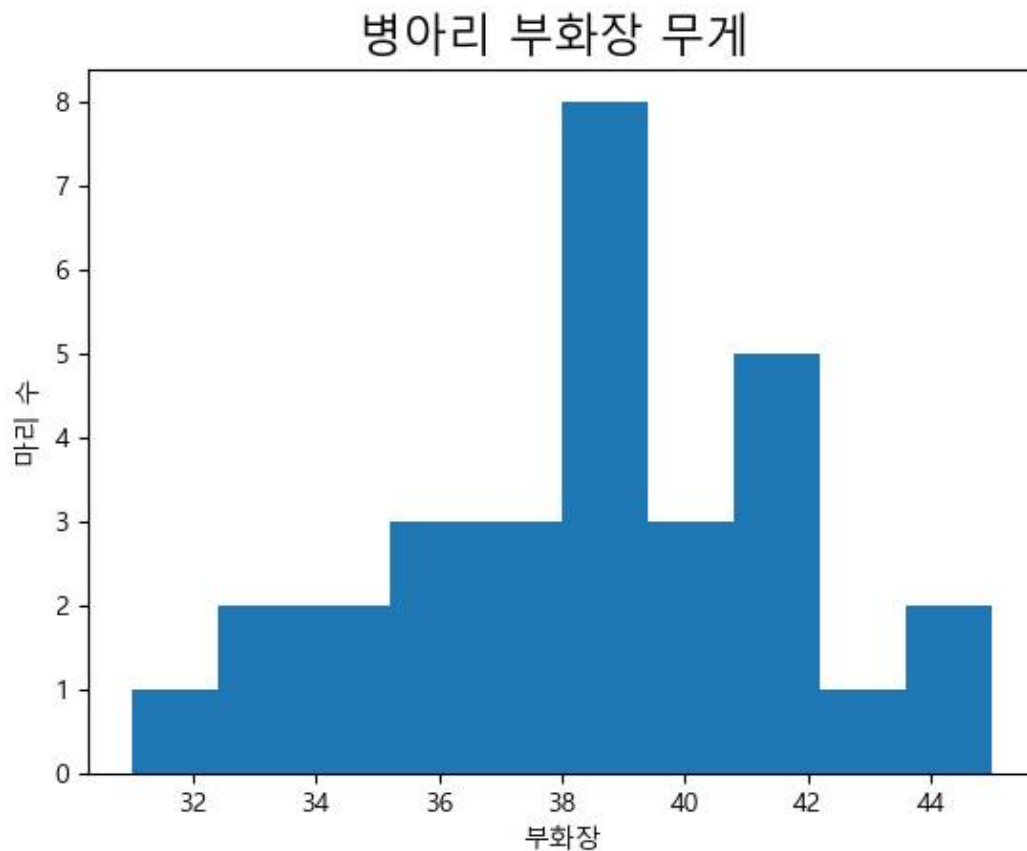
`autopct='%0.1f%%'`, 차트내의 숫자 소숫점 자리 수
`startangle=90` --> 글자의 각도를 틀어준다.

10. chick002.csv 히스토그램 그리기

	chick_nm	weight
0	b01	37
1	b02	39
2	b03	41
3	b04	45
4	b05	37

```
b = pd.read_csv('data/chick002.csv') # b 변수에 데이터셋 입력
print(b)
print(b.describe())
```

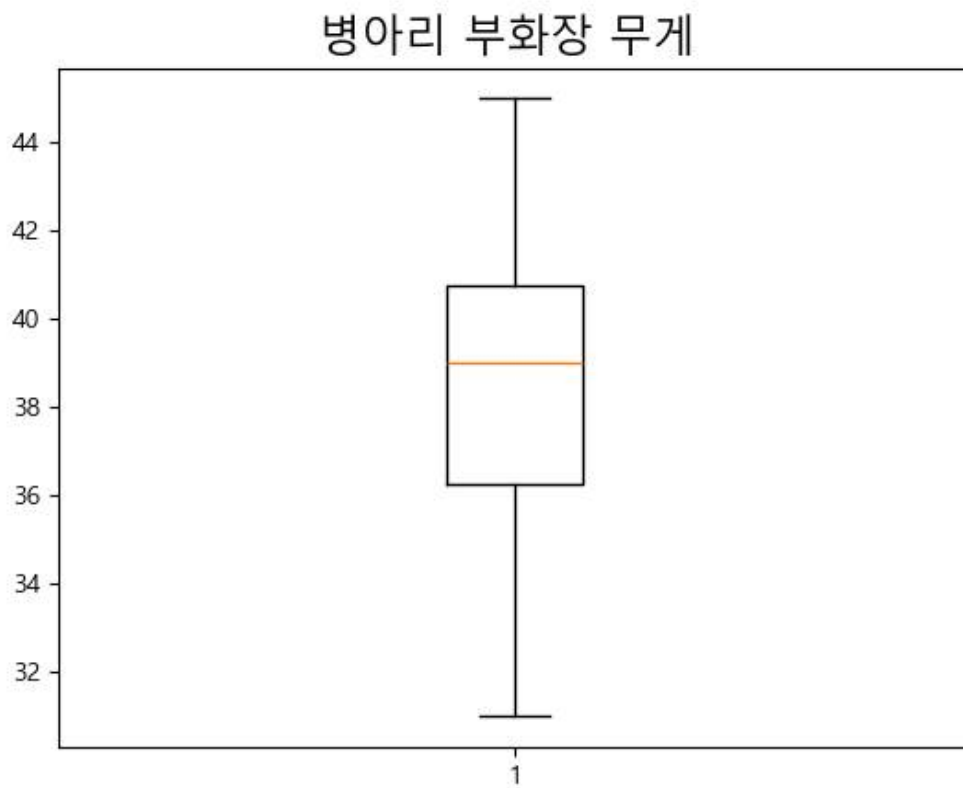
```
plt.hist(b.weight)
plt.title('병아리 부화장 무게 ', fontsize=17)
plt.xlabel('부화장')
plt.ylabel('마리 수')
plt.show()
print("-----")
```



--> 정규분포의 윤곽을 알 수 있다.

11. 상자 그림 chick004.csv

```
plt.boxplot(b.weight)
plt.title('병아리 부화장 무게 ', fontsize=17)
plt.show()
print("-----")
```

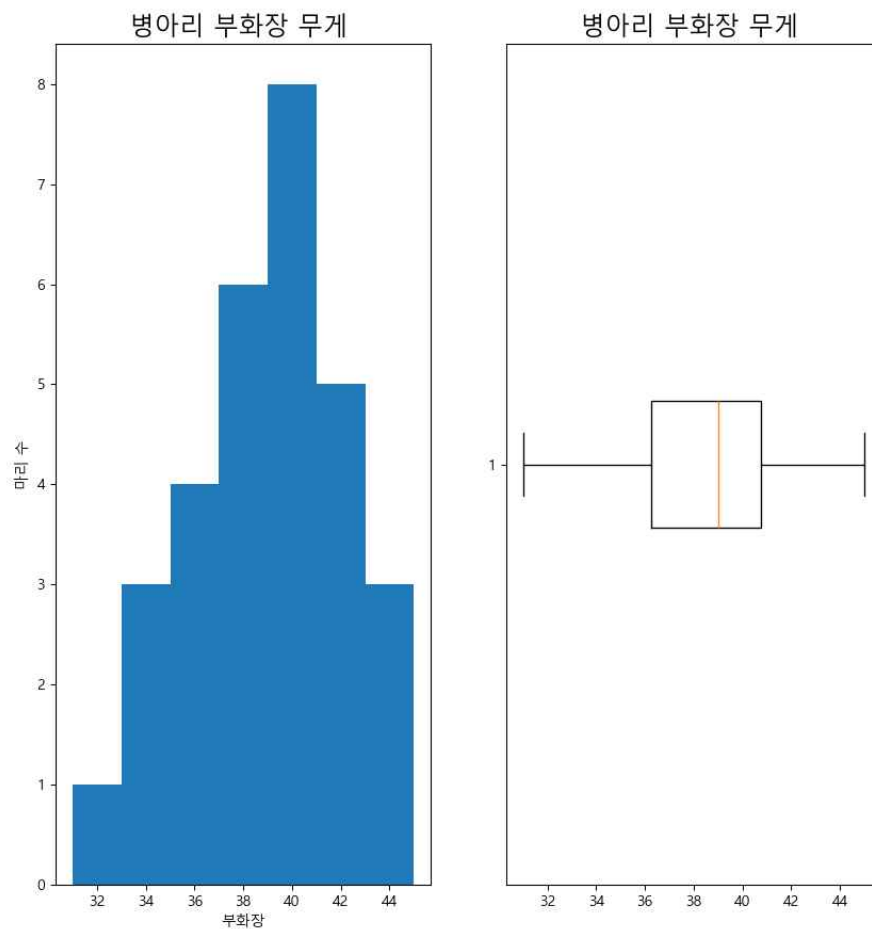


- > 주황색선이 중간치 (median)
- > 사각형 밑이 1사분위 위가 3사분위 그 차이가 IQR
- > 아랫수염= $Q1 - 1.5 * IQR$ 윗수염= $Q3 + 1.5 * IQR$
- > 수염 위아래로 이상치 : 없음

12. 히스토그램과 상자그림을 같은 화면에 그리기

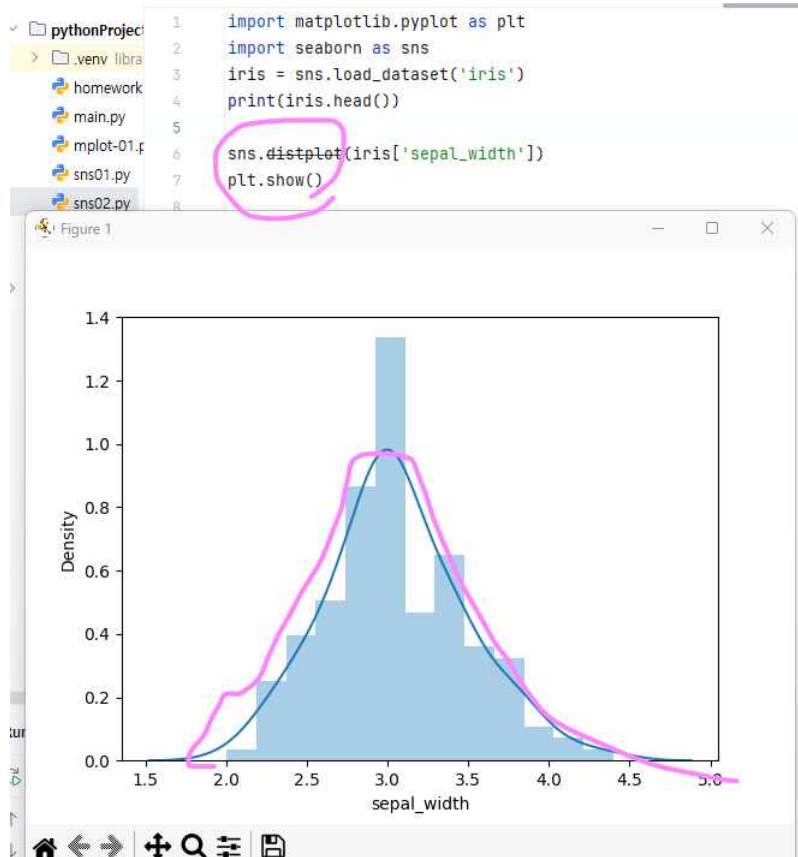
```
plt.figure(figsize=(10, 12))
plt.subplot(1, 2, 1)
plt.hist(b.weight, bins=7)
plt.title('병아리 부화장 무게 ', fontsize=17)
plt.xlabel('부화장')
plt.ylabel('마리 수')

plt.subplot(1, 2, 2)
plt.boxplot(b.weight, vert=False) # 가로로 그리기
plt.title('병아리 부화장 무게 ', fontsize=17)
plt.show()
```

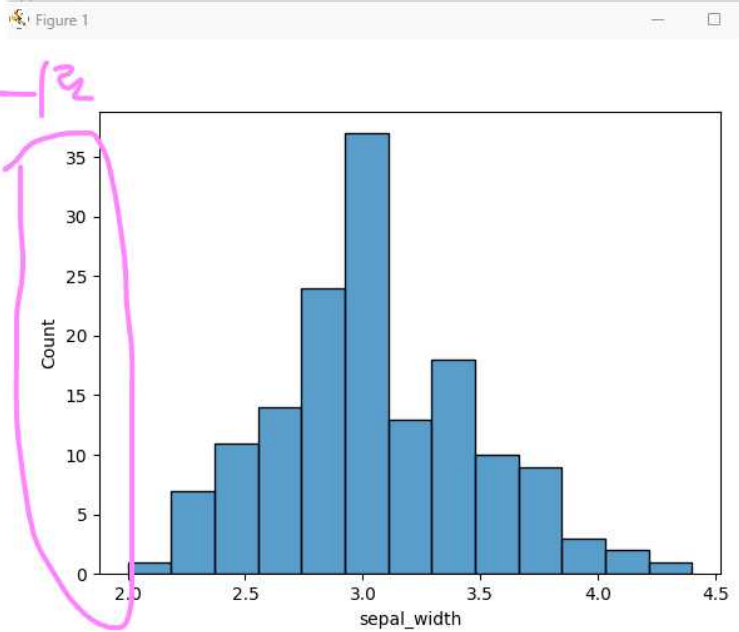


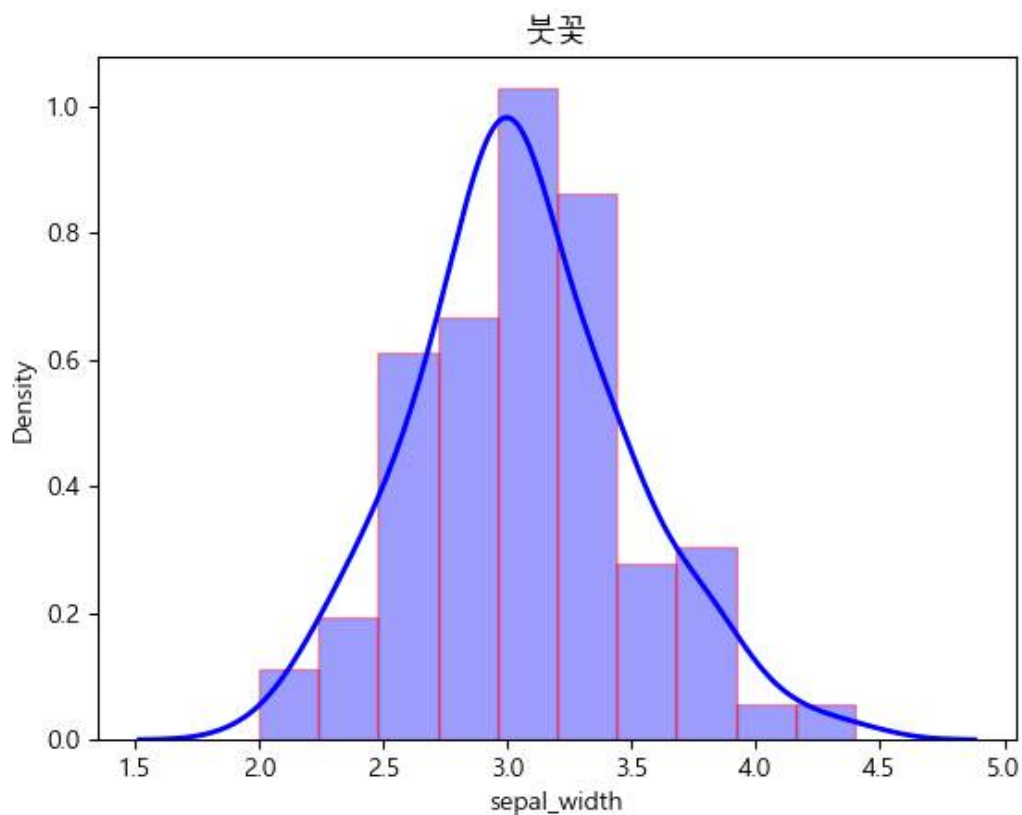
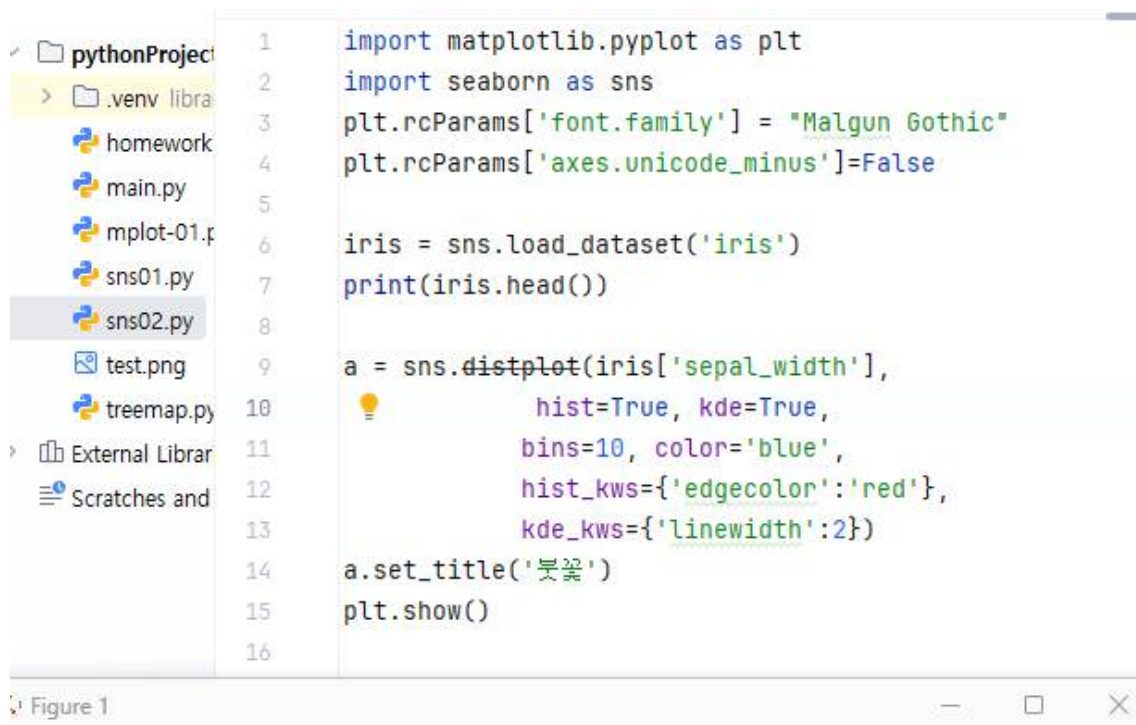
참고 자료

matplotlib 2024-10-24



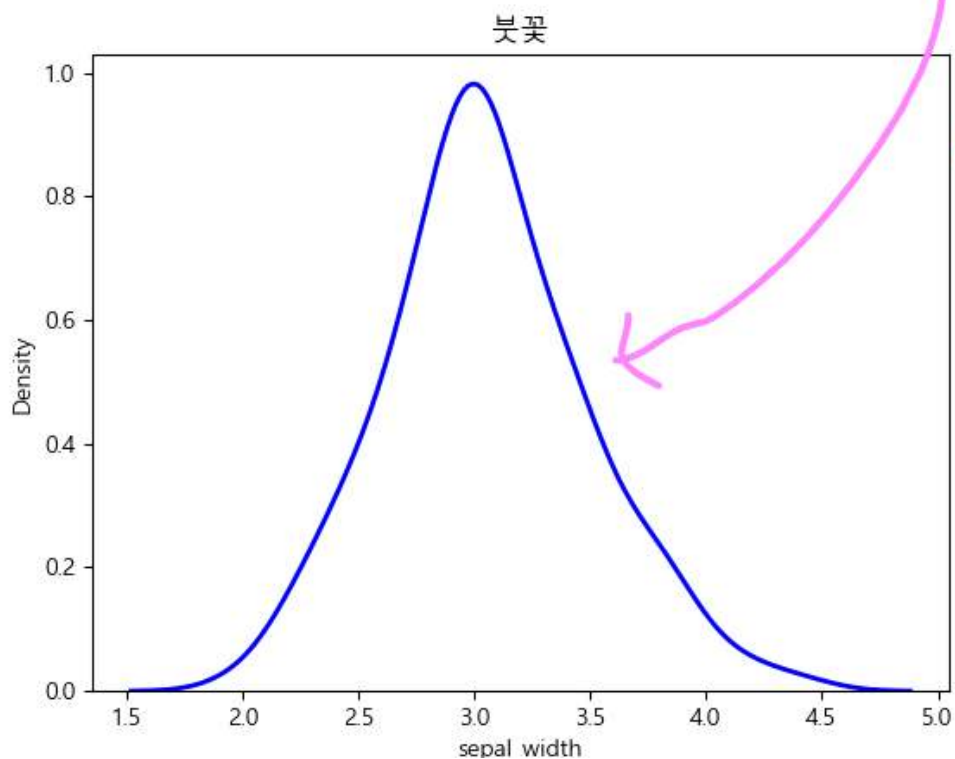
```
9 sns.histplot(iris['sepal_width'])
10 plt.show()
```





```
pythonProject
├── .venv
├── lib
├── homework
├── main.py
├── mplot-01.py
├── sns01.py
├── sns02.py
├── test.png
├── treemap.py
├── External Libraries
├── Scratches and
1 import matplotlib.pyplot as plt
2 import seaborn as sns
3 plt.rcParams['font.family'] = "Malgun Gothic"
4 plt.rcParams['axes.unicode_minus']=False
5
6 iris = sns.load_dataset('iris')
7 print(iris.head())
8
9 a = sns.distplot(iris['sepal_width'],
10                 hist=False, kde=True,
11                 bins=10, color='blue',
12                 hist_kws={'edgecolor':'red'},
13                 kde_kws={'linewidth':2})
14 a.set_title('붓꽃')
15 plt.show()
16
```

Figure 1



붓꽃

Seaborn 히스토그램 그리기

distplot에는 설정시에 다양한 속성들이 존재함

hist는 히스토그램 표시 여부, kde는 추세선(kernel density curve) 표시 여부

bins는 막대의 개수, color는 그래프의 색상을 의미

hist_kws는 히스토그램의 옵션을 설정하며, 예제에서는 히스토그램의 테두리를 빨간색으로 설정

kde_kws는 추세선의 옵션을 설정하며, 예제에서는 선 두께를 2로 설정

set_title은 그래프 제목 설정, set_xlabel, set_ylabel은 x축과 y축의 제목을 설정

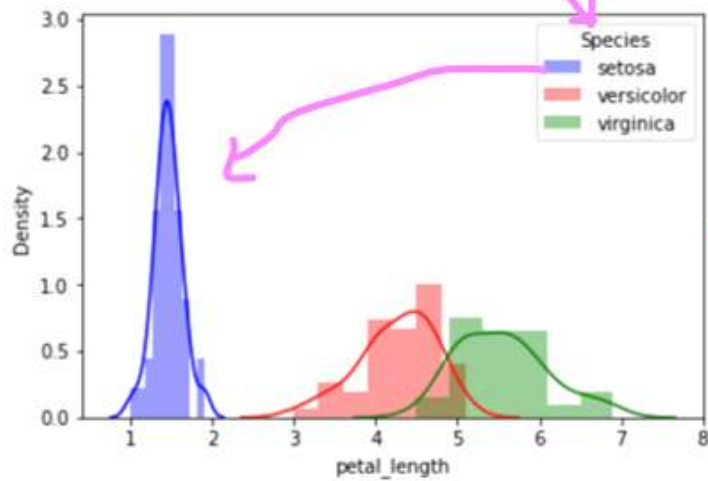
```
a = sns.distplot(iris['sepal_width'],
                  hist=True, kde=True,
                  bins=10, color='blue',
                  hist_kws={'edgecolor': 'red'},
                  kde_kws={'linewidth': 2})

a.set_title('sepal width histogram')
a.set_xlabel('sepal width')
a.set_ylabel('density')
plt.show()
```

<https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751>

seaborn 매뉴얼 참고하면 좋다.

```
sns.distplot(iris[iris.species=='setosa']['petal_length'],
             color='blue', label='setosa')
sns.distplot(iris[iris.species=='versicolor']['petal_length'],
             color='red', label='versicolor')
sns.distplot(iris[iris.species=='virginica']['petal_length'],
             color='green', label='virginica')
plt.legend(title='Species')
plt.show()
```



이해하기 쉽도록

```
tips = sns.load_dataset('tips')
print(tips.shape)
print(tips.head())
print(tips.tail())
print(tips.describe())
```

(244, 7)

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2
1	10.34	1.66	Male	No	Sun	Dinner	3
2	21.01	3.50	Male	No	Sun	Dinner	3
3	23.68	3.31	Male	No	Sun	Dinner	2
4	24.59	3.61	Female	No	Sun	Dinner	4

	total_bill	tip	sex	smoker	day	time	size
239	29.03	5.92	Male	No	Sat	Dinner	3
240	27.18	2.00	Female	Yes	Sat	Dinner	2
241	22.67	2.00	Male	Yes	Sat	Dinner	2
242	17.82	1.75	Male	No	Sat	Dinner	2
243	18.78	3.00	Female	No	Thur	Dinner	2

	total_bill	tip	size
count	244.000000	244.000000	244.000000
mean	19.785943	2.998279	2.569672
std	8.902412	1.383638	0.951100
min	3.070000	1.000000	1.000000
25%	13.347500	2.000000	2.000000
50%	17.795000	2.900000	2.000000
75%	24.127500	3.562500	3.000000
max	50.810000	10.000000	6.000000


```
tips = sns.load_dataset('tips')
print(tips.shape)
print(tips.head())
print(tips.tail())
print(tips.describe())
```

(244, 7)

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2
1	10.34	1.66	Male	No	Sun	Dinner	3
2	21.01	3.50	Male	No	Sun	Dinner	3
3	23.68	3.31	Male	No	Sun	Dinner	2
4	24.59	3.61	Female	No	Sun	Dinner	4

	total_bill	tip	sex	smoker	day	time	size
239	29.03	5.92	Male	No	Sat	Dinner	3
240	27.18	2.00	Female	Yes	Sat	Dinner	2
241	22.67	2.00	Male	Yes	Sat	Dinner	2
242	17.82	1.75	Male	No	Sat	Dinner	2
243	18.78	3.00	Female	No	Thur	Dinner	2

	total_bill	tip	size
count	244.000000	244.000000	244.000000
mean	19.785943	2.998279	2.569672
std	8.902412	1.383638	0.951100
min	3.000000	1.000000	1.000000
25%	13.347500	2.000000	2.000000
50%	17.795000	2.900000	2.000000
75%	24.127500	3.562500	3.000000
max	50.810000	10.000000	6.000000

```
tips_day = tips.groupby('day').tip.sum()
tips_day
```

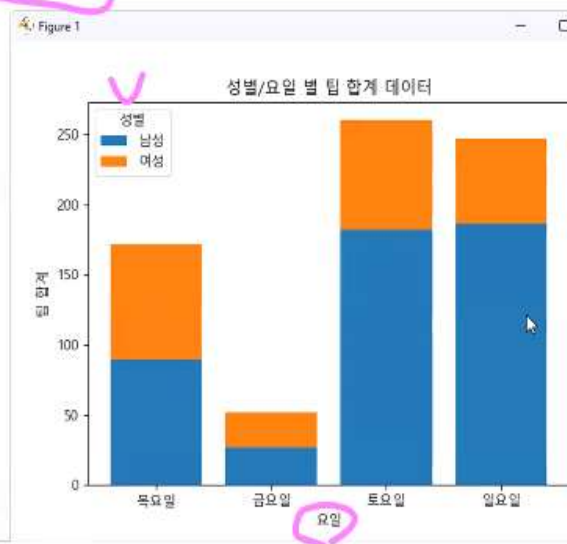
```
day
Thur    171.83
Fri      51.96
Sat     260.40
Sun     247.39
Name: tip, dtype: float64
```



```

16 xtick = ["목요일", "금요일", "토요일", "일요일"]
17 xIndex = [1, 2, 3, 4]
18
19 tips_day = tips.groupby("day").tip.sum()
20
21 male_tip = tips[tips["sex"] == "Male"].groupby("day").tip.sum()
22 # 요일별 남성의 팁 합계
23 female_tip = tips[tips["sex"] == "Female"].groupby("day").tip.sum()
24 # 요일별 여성의 팁 합계
25
26 p1 = plt.bar(xIndex, male_tip, label="남성")
27 p2 = plt.bar(xIndex, female_tip, bottom=male_tip, label="여성")
28
29 plt.title("성별/요일 별 팁 합계 데이터")
30 plt.ylabel("팁 합계")
31 plt.xlabel("요일")
32 plt.xticks(xIndex, xtick)
33 plt.legend(title="성별")
34
35 plt.show()
36

```

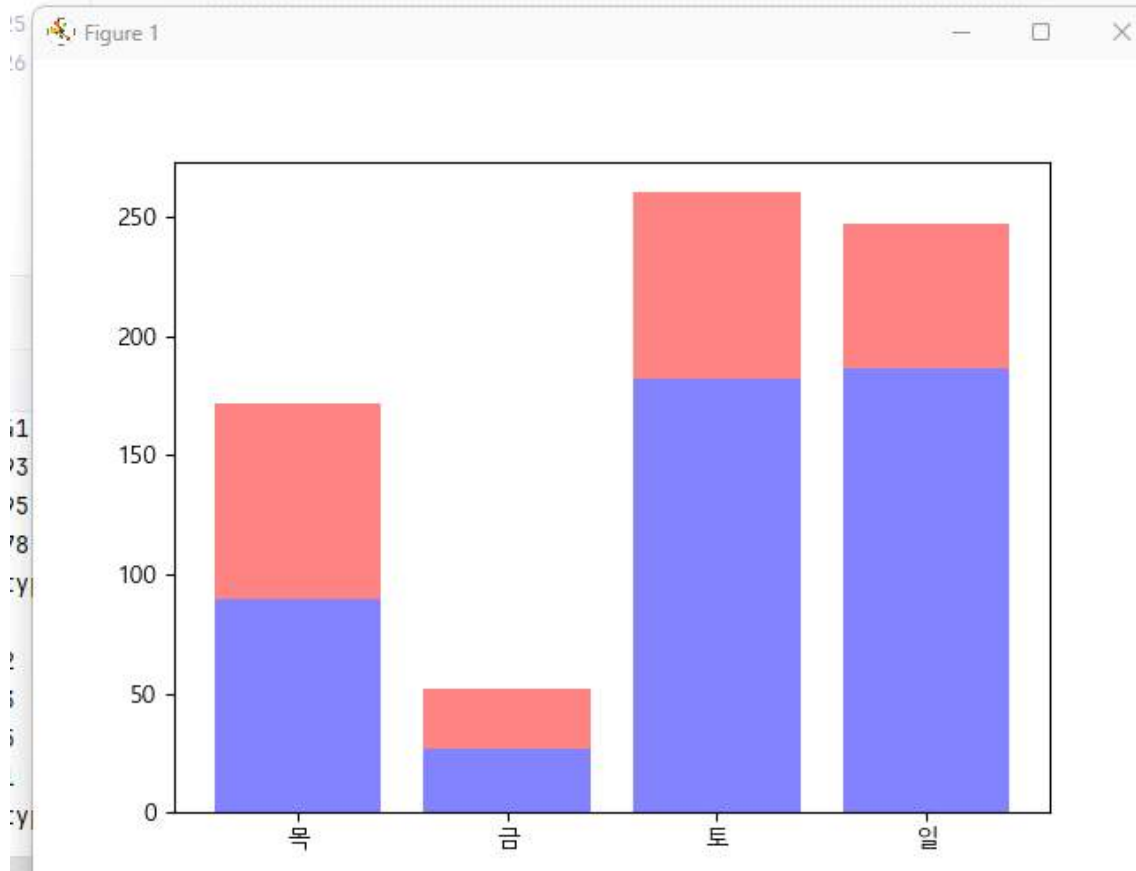


bottom 아래쪽 그림이다.

```

8 tips = sns.load_dataset('tips')
9 tips_day = tips.groupby('day').tip.sum()
0 print(tips_day)
1
2 male_tip = tips[tips["sex"]=="Male"].groupby("day").tip.sum()
3 print(male_tip)
4
5 female_tip = tips[tips["sex"]=="Female"].groupby("day").tip.sum()
6 print(female_tip)
7
8 p1 = plt.bar(xindex, male_tip, color='blue', alpha=0.5)
9 p2 = plt.bar(xindex, female_tip, color='red', alpha=0.5, bottom=male_tip)
10
11 plt.xticks(xindex, xticks)
12
13 plt.show()

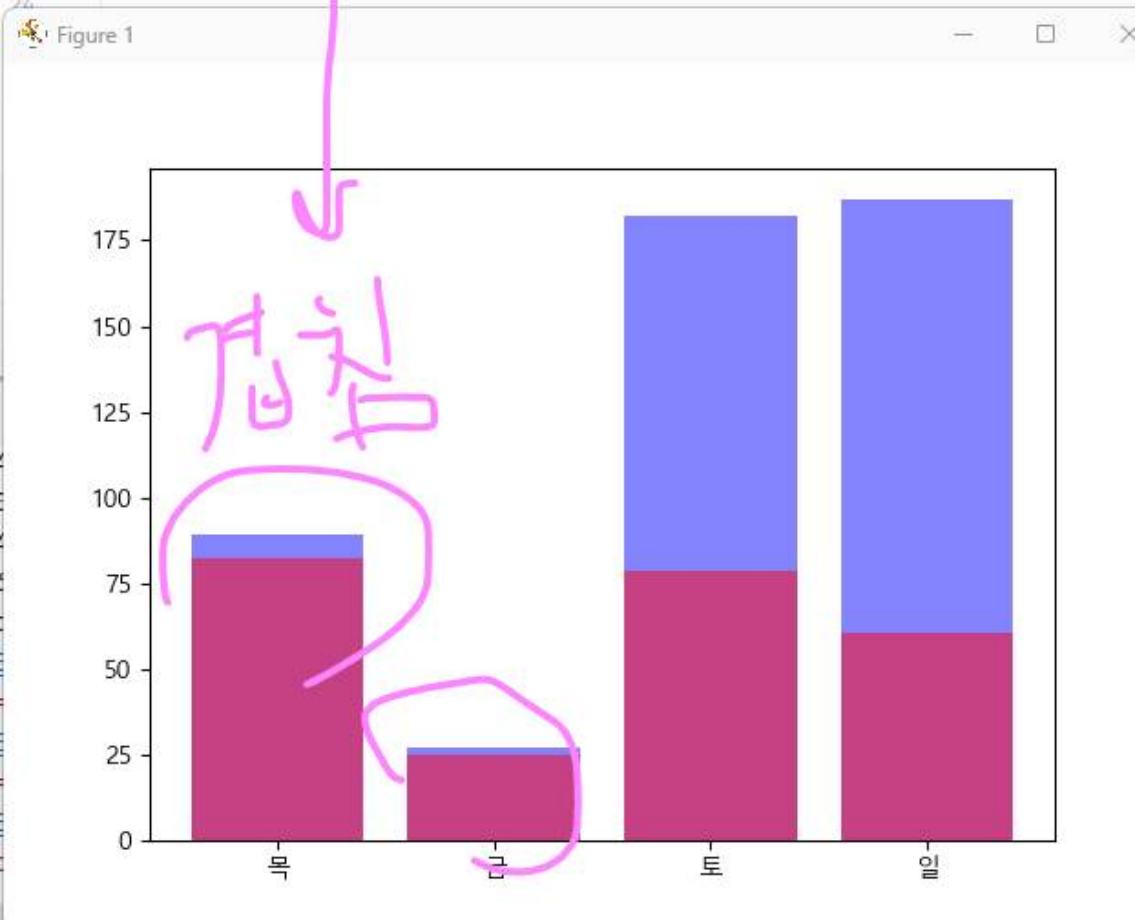
```

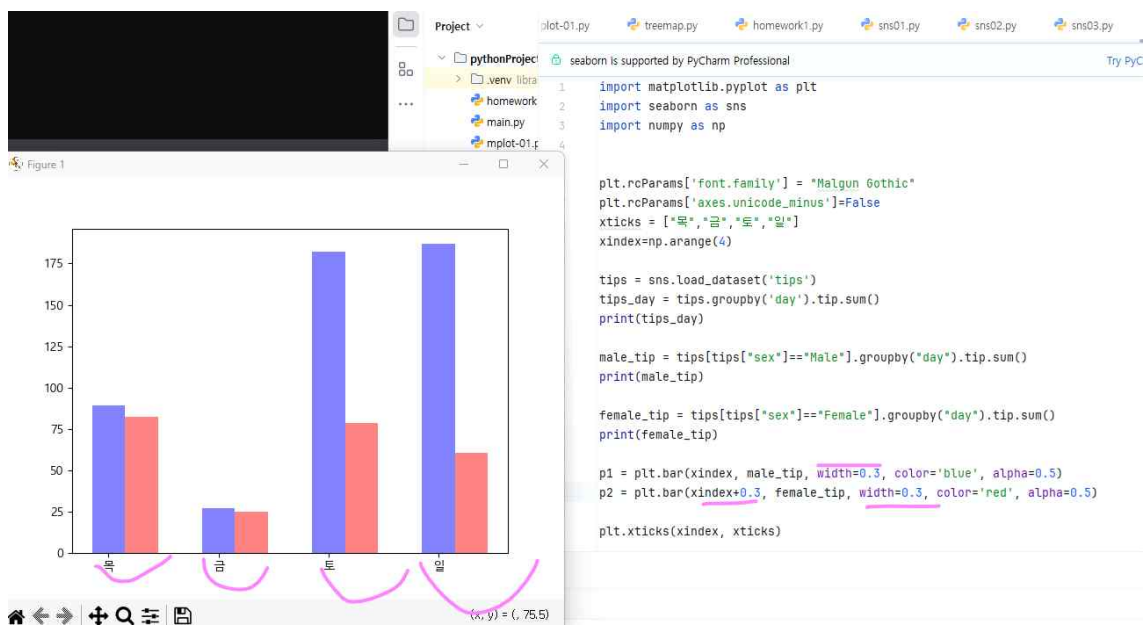
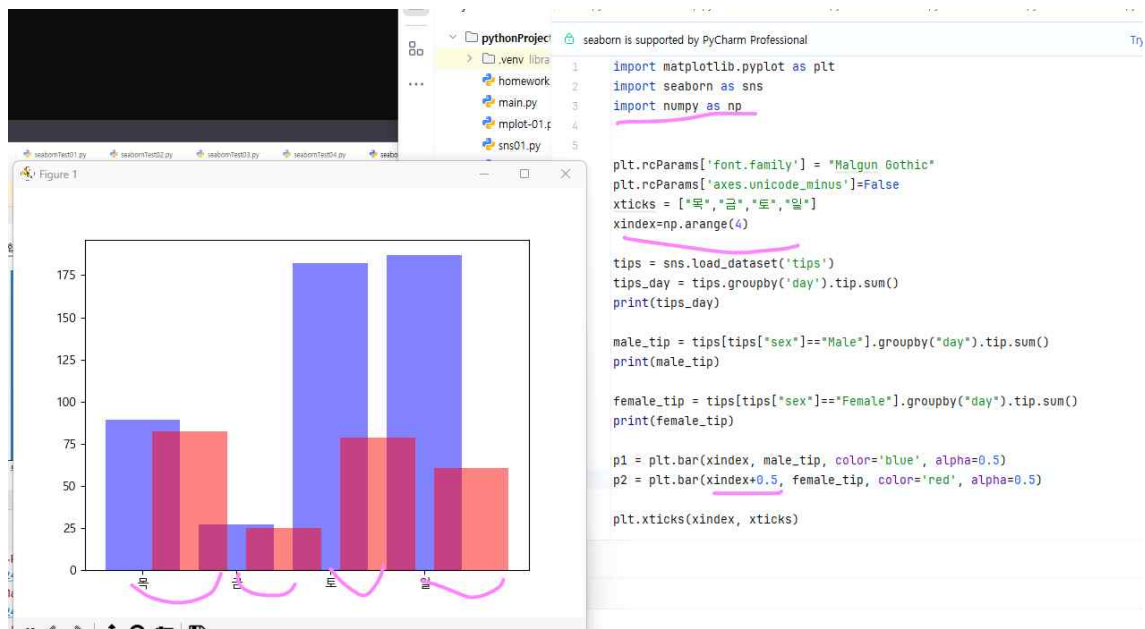


```

8 tips = sns.load_dataset('tips')
9 tips_day = tips.groupby('day').tip.sum()
10 print(tips_day)
11
12 male_tip = tips[tips["sex"]=="Male"].groupby("day").tip.sum()
13 print(male_tip)
14
15 female_tip = tips[tips["sex"]=="Female"].groupby("day").tip.sum()
16 print(female_tip)
17
18 p1 = plt.bar(xindex, male_tip, color='blue', alpha=0.5)
19 p2 = plt.bar(xindex, female_tip, color='red', alpha=0.5)
20
21 plt.xticks(xindex, xticks)
22
23 plt.show()
24

```



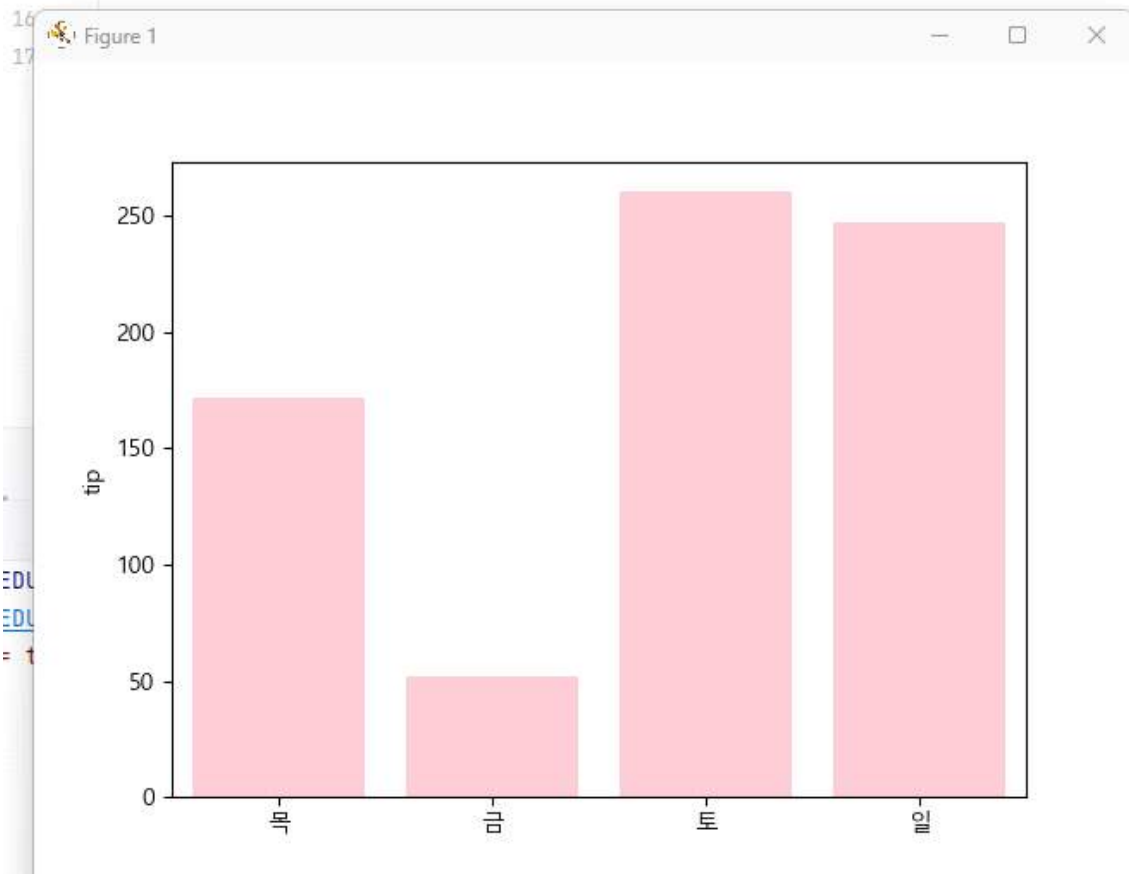


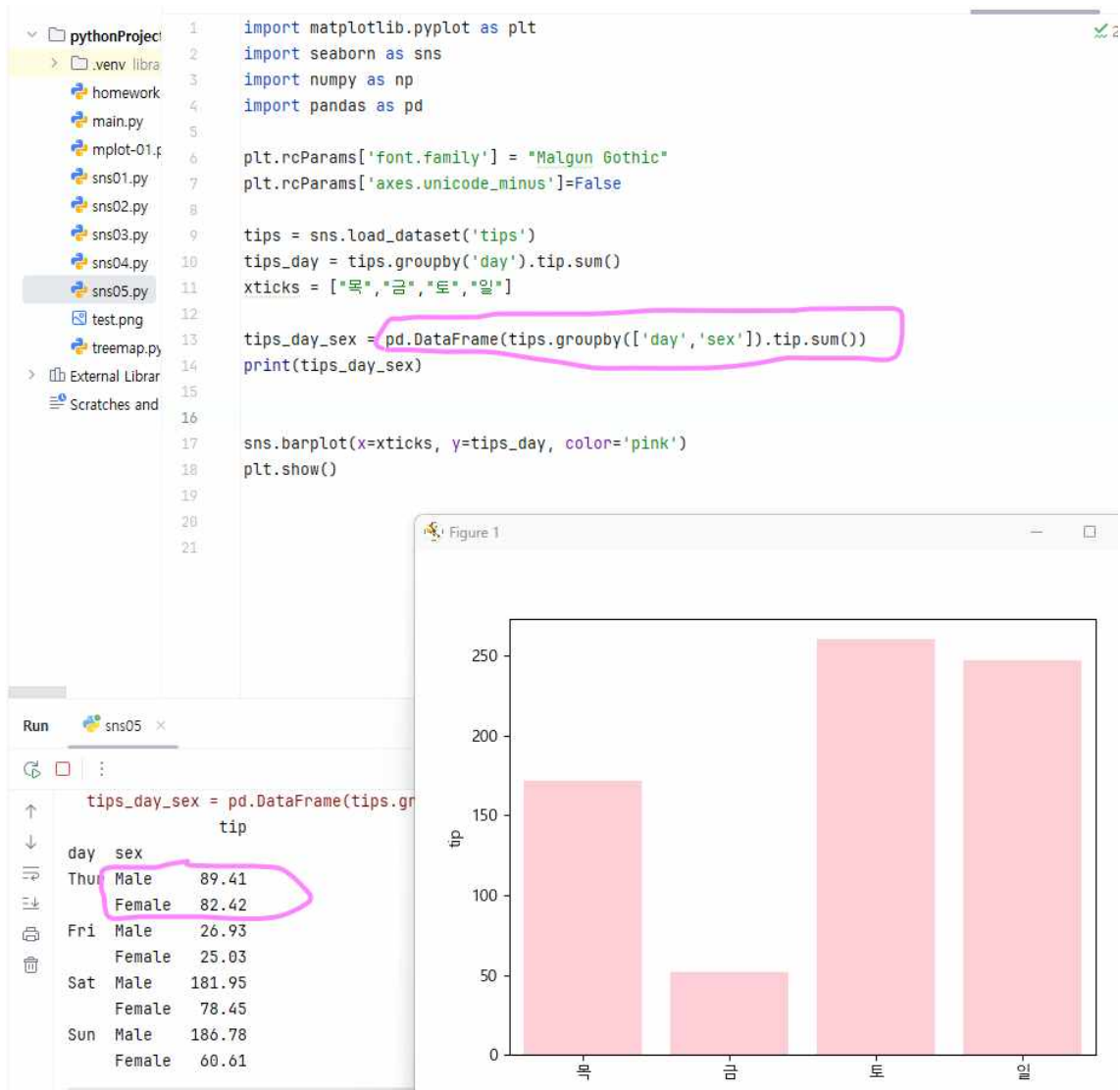
위는 matplotlib로 그린 것이다
이것을 seaborn으로 그려보자.

```

1 import matplotlib.pyplot as plt
2 import seaborn as sns
3 import numpy as np
4
5
6 plt.rcParams['font.family'] = "Malgun Gothic"
7 plt.rcParams['axes.unicode_minus']=False
8
9 tips = sns.load_dataset('tips')
10 tips_day = tips.groupby('day').tip.sum()
11 xticks = ["목", "금", "토", "일"]
12
13 sns.barplot(x=xticks, y=tips_day, color='pink')
14 plt.show()
15

```





문제가 생긴다.

pd.DataFrame(tips.groupby(["day", "sex"]).tip.sum())

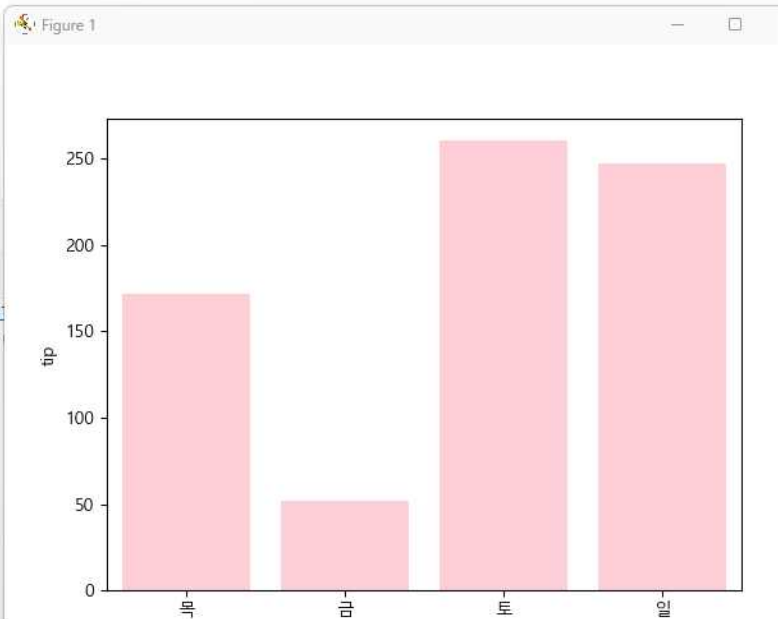
```
pythonProject
├── .venv
├── lib
├── homework
├── main.py
├── mplot-01.f
├── sns01.py
├── sns02.py
├── sns03.py
├── sns04.py
├── sns05.py
├── test.png
├── treemap.py
├── External Librar
├── Scratches and
└──
```

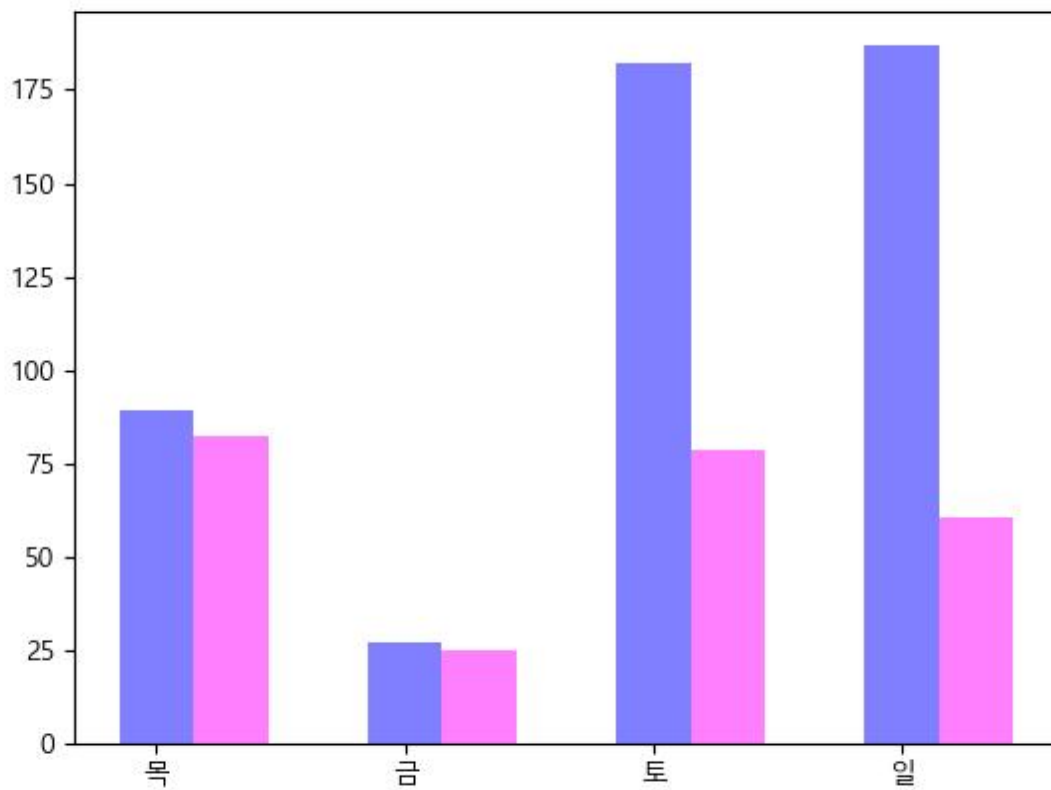
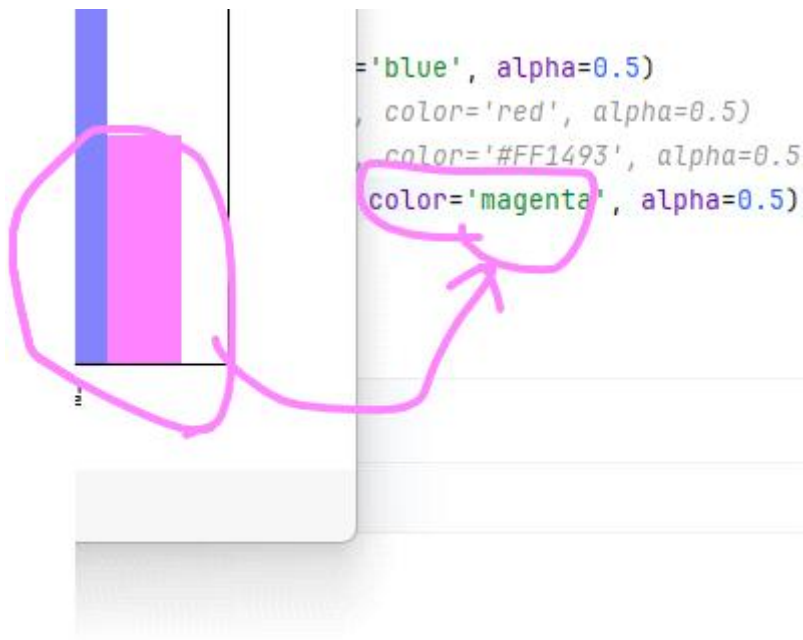
```
1 import matplotlib.pyplot as plt
2 import seaborn as sns
3 import numpy as np
4 import pandas as pd
5
6 plt.rcParams['font.family'] = "Malgun Gothic"
7 plt.rcParams['axes.unicode_minus']=False
8
9 tips = sns.load_dataset('tips')
10 tips_day = tips.groupby('day').tip.sum()
11 xticks = ["목", "금", "토", "일"]
12
13 tips_day_sex = pd.DataFrame(tips.groupby(['day', 'sex']).tip.sum())
14 tips_day_sex = tips_day_sex.reset_index()
15
16
17
18 sns.barplot(x=xticks, y=tips_day, color='pink')
19 plt.show()
20
21
22
```

sns05

C:\Users\JBEDU30\pythonProject

	day	sex	tip
0	Thur	Male	89.41
1	Thur	Female	82.42
2	Fri	Male	26.93
3	Fri	Female	25.03
4	Sat	Male	181.95
5	Sat	Female	78.45
6	Sun	Male	186.78
7	Sun	Female	60.61

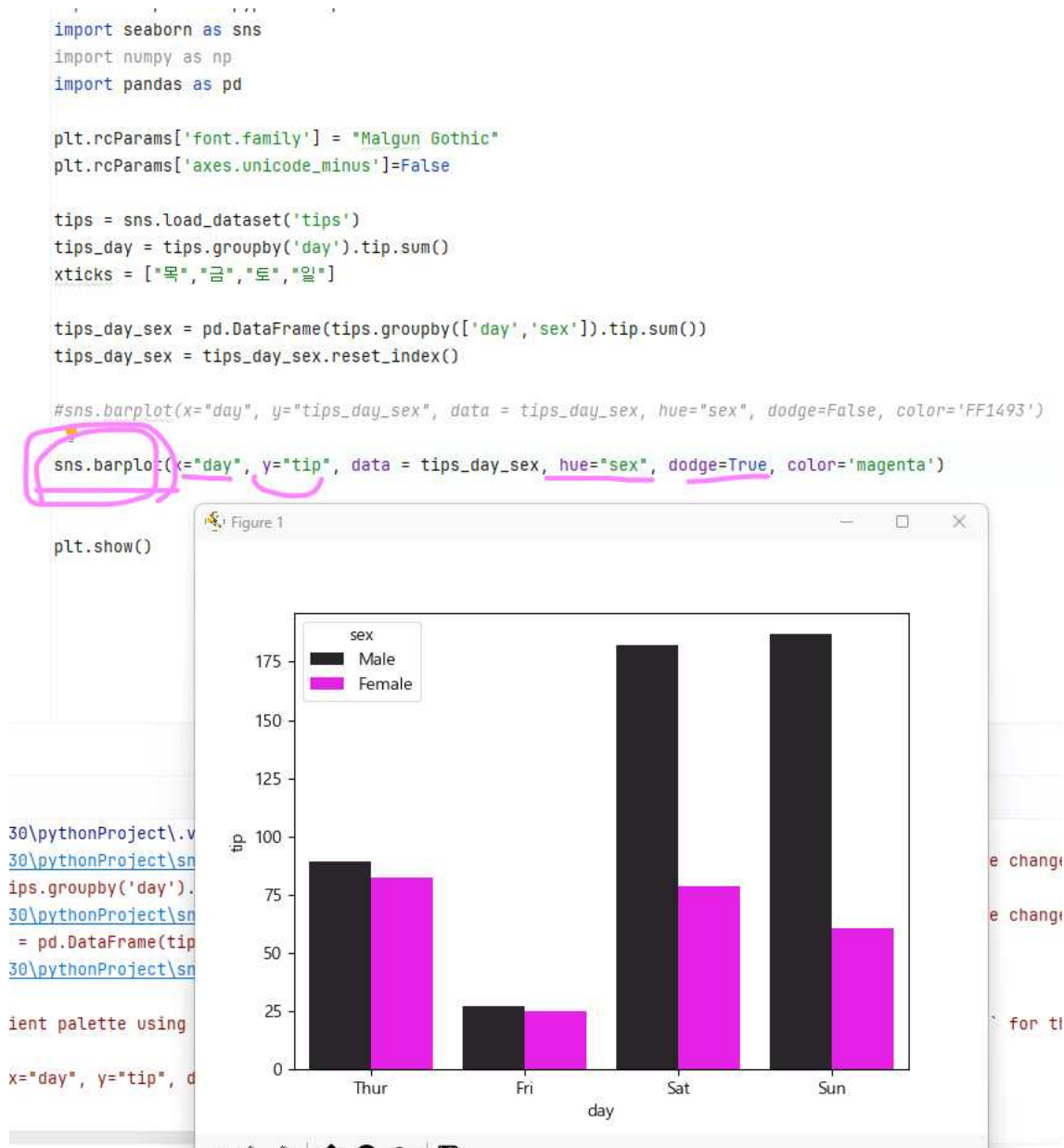




```

plt.rcParams['font.family'] = 'Malgun Gothic'
plt.rcParams['axes.unicode_minus'] = False

```

hue 범례

넣어야만 그래프가 나온다

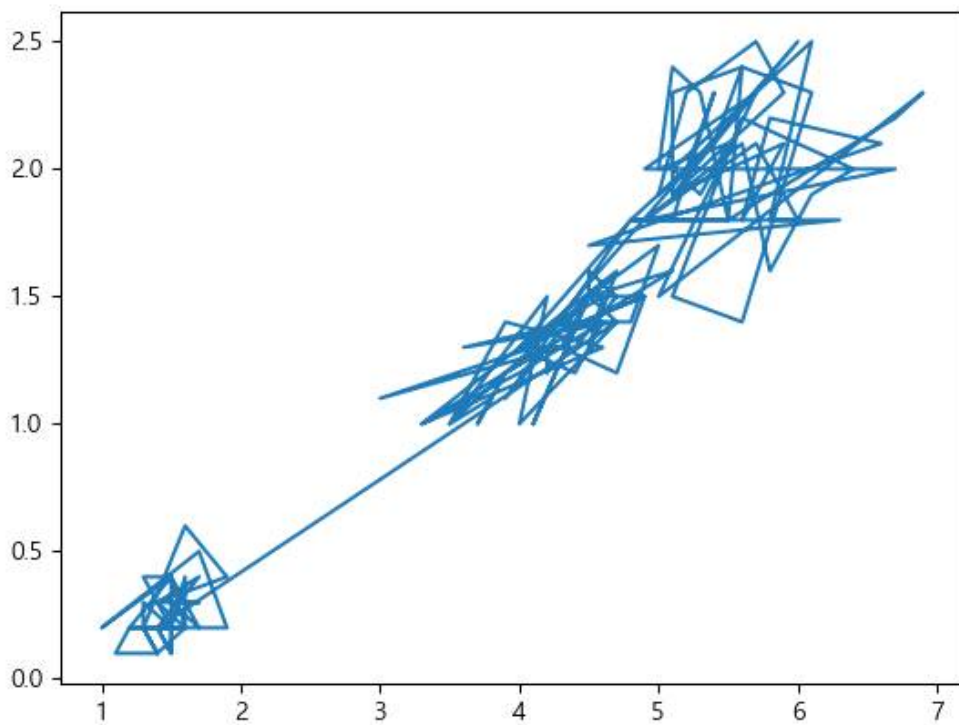
seaborn은 복잡하구나

산점도는 좋단다.

seaborn으로 산점도그리기

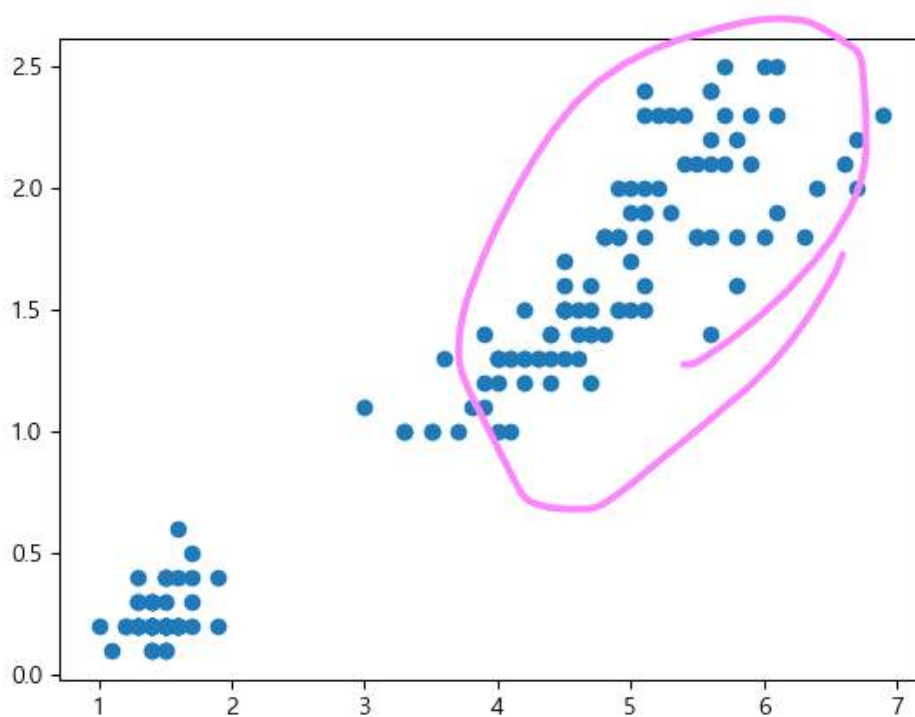
```
1 import matplotlib.pyplot as plt
2 import seaborn as sns
3 import numpy as np
4 import pandas as pd
5
6 plt.rcParams['font.family'] = "Malgun Gothic"
7 plt.rcParams['axes.unicode_minus']=False
8
9
10 iris = sns.load_dataset('iris')
11 plt.plot(*args: 'petal_length', 'petal_width', data=iris)
12 plt.show()
```

Figure 1



```
1 import matplotlib.pyplot as plt
2 import seaborn as sns
3 import numpy as np
4 import pandas as pd
5
6 plt.rcParams['font.family'] = "Malgun Gothic"
7 plt.rcParams['axes.unicode_minus']=False
8
9
10 iris = sns.load_dataset('iris')
11 plt.plot(*args: 'petal_length', 'petal_width', data=iris,
12         linestyle='none', marker='o')
13 plt.show()
```

Figure 1

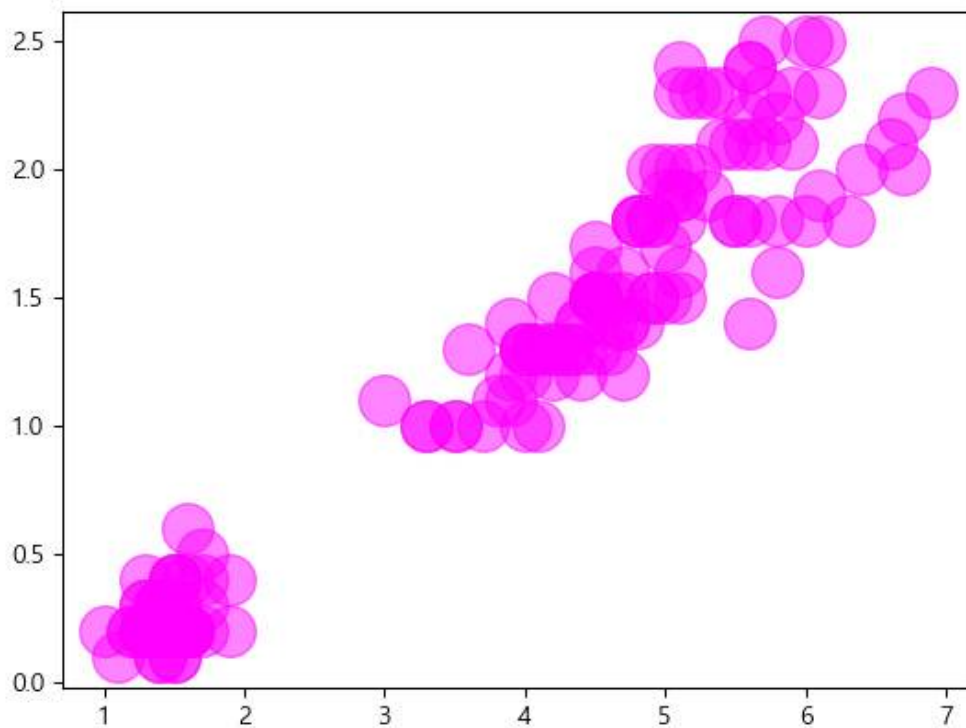


```

1 import matplotlib.pyplot as plt
2 import seaborn as sns
3 import numpy as np
4 import pandas as pd
5
6 plt.rcParams['font.family'] = "Malgun Gothic"
7 plt.rcParams['axes.unicode_minus']=False
8
9
10 iris = sns.load_dataset('iris')
11 plt.plot(*args: 'petal_length', 'petal_width', data=iris,
12          linestyle='none', marker='o',
13          markersize=20, color="magenta", alpha=0.5)
14 plt.show()

```

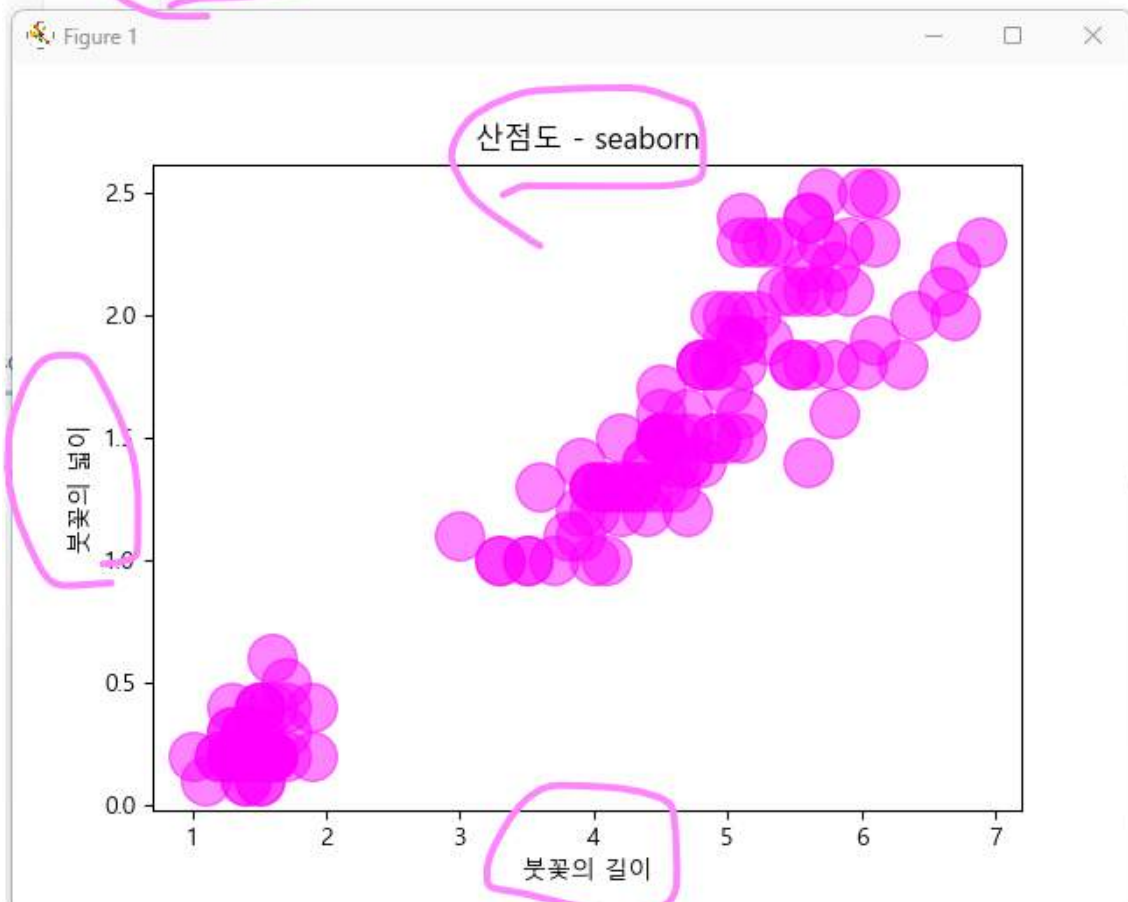
Figure 1



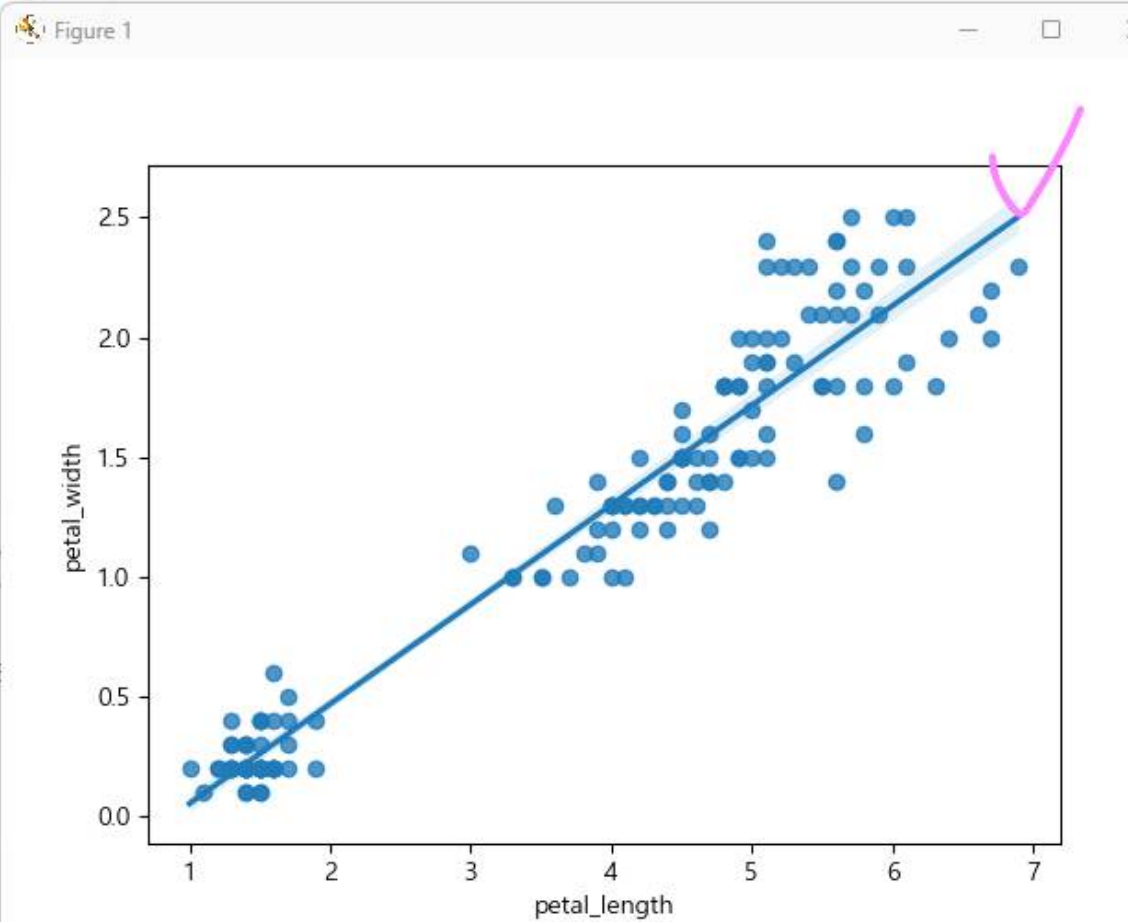
```

1 import matplotlib.pyplot as plt
2 import seaborn as sns
3 import numpy as np
4 import pandas as pd
5
6 plt.rcParams['font.family'] = "Malgun Gothic"
7 plt.rcParams['axes.unicode_minus']=False
8
9
10 iris = sns.load_dataset('iris')
11 plt.plot(*args: 'petal_length', 'petal_width', data=iris,
12         linestyle='none', marker='o',
13         markersize=20, color="magenta", alpha=0.5)
14
15 plt.title('산점도 - seaborn')
16 plt.xlabel('붓꽃의 길이')
17 plt.ylabel('붓꽃의 넓이')
18 plt.show()

```

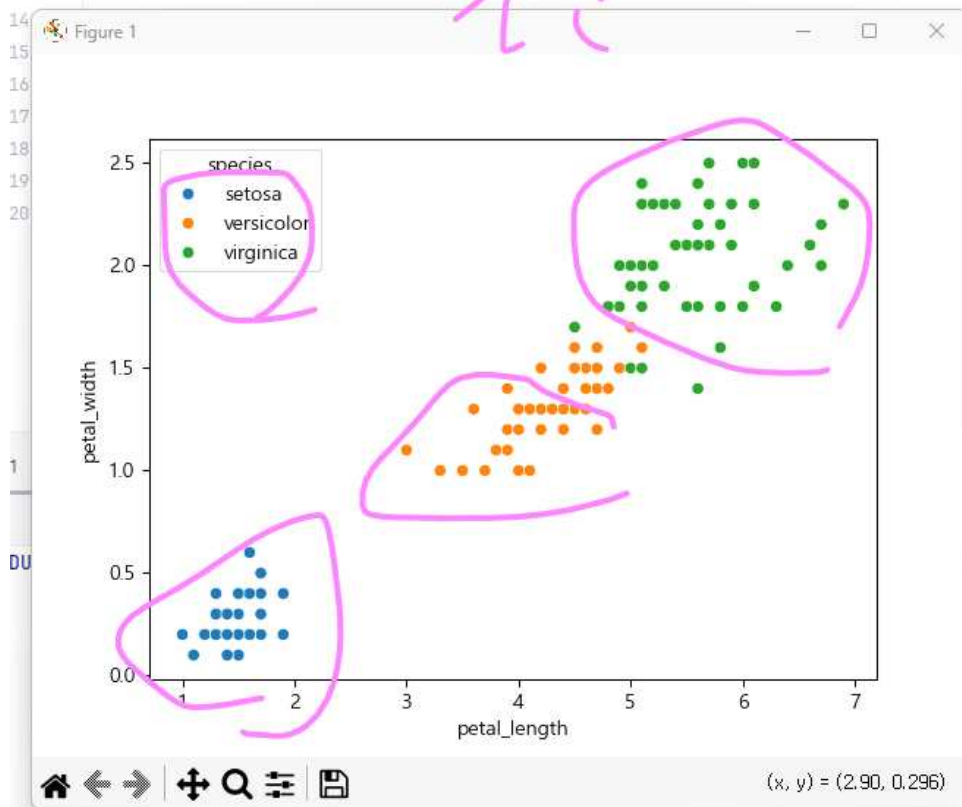


```
1 import matplotlib.pyplot as plt
2 import seaborn as sns
3 import numpy as np
4 import pandas as pd
5
6 plt.rcParams['font.family'] = "Malgun Gothic"
7 plt.rcParams['axes.unicode_minus']=False
8
9
10 iris = sns.load_dataset('iris')
11 sns.regplot(x=iris['petal_length'],y=iris['petal_width'])
12 plt.show()
13
14
```



```
10 iris = sns.load_dataset('iris')
11 sns.scatterplot(x=iris['petal_length'], y=iris['petal_width'], data=iris, hue="species")
12 plt.show()
```

간단

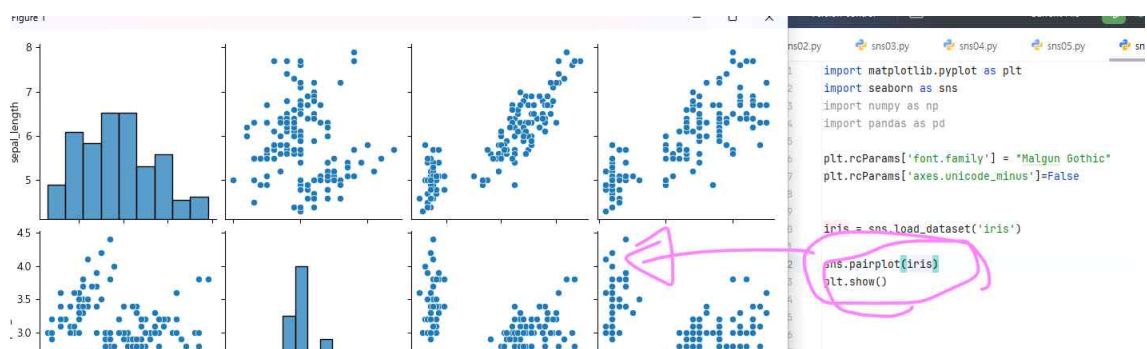
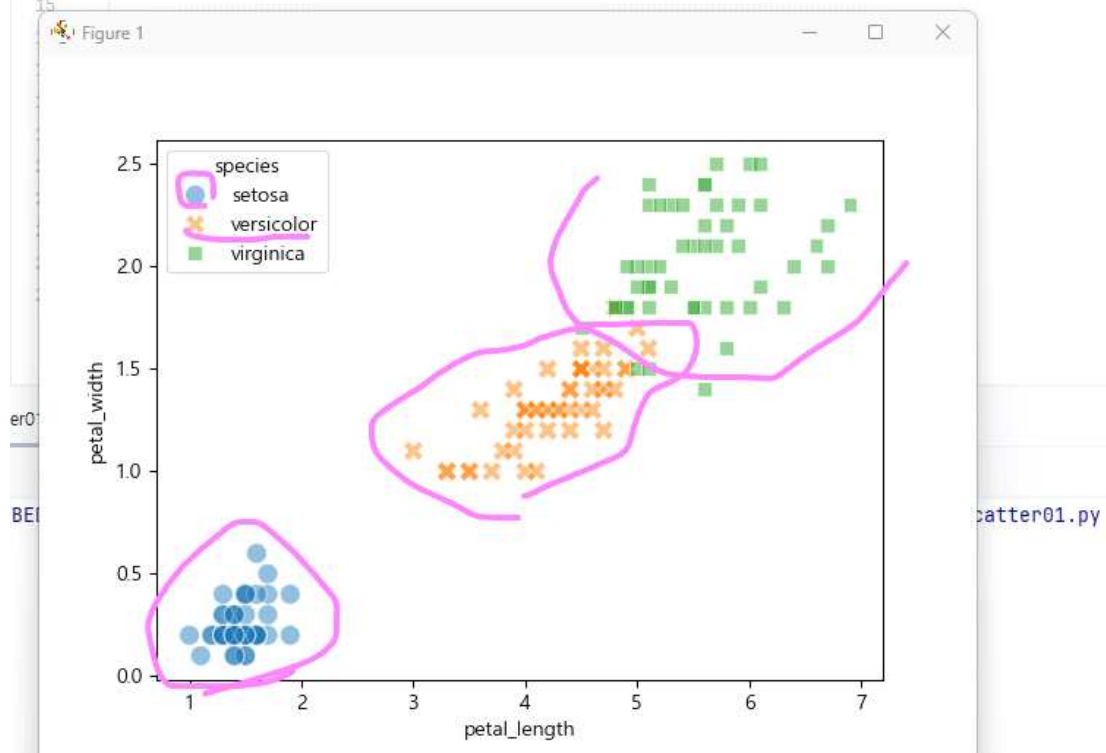


tter01.py


```

9
10 iris = sns.load_dataset('iris')
11 sns.scatterplot(x=iris['petal_length'], y=iris['petal_width'], data=iris, hue="species",
12                style="species", s=100, alpha=0.5)
13 plt.show()
14
15

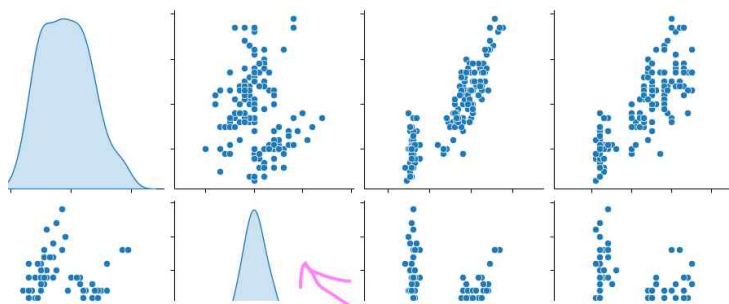
```



```

sns.pairplot(iris)
plt.show()

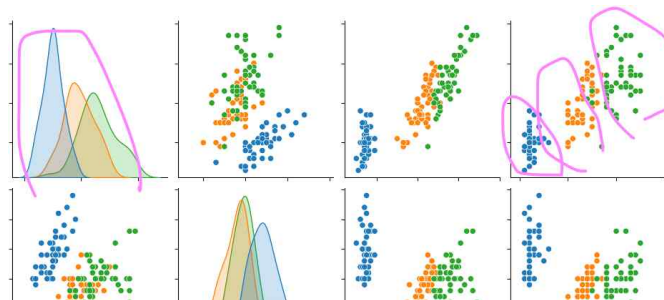
```

```
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np
import pandas as pd

plt.rcParams['font.family'] = "Malgun Gothic"
plt.rcParams['axes.unicode_minus']=False

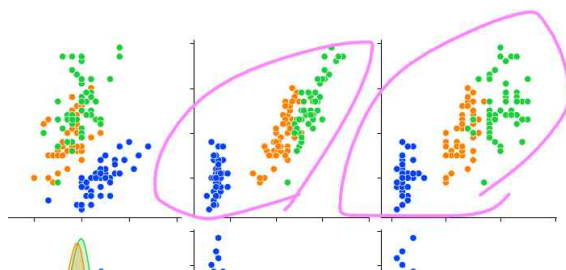
iris = sns.load_dataset('iris')
sns.pairplot(iris, diag_kind='kde')
plt.show()
```



```
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np
import pandas as pd

plt.rcParams['font.family'] = "Malgun Gothic"
plt.rcParams['axes.unicode_minus']=False

iris = sns.load_dataset('iris')
sns.pairplot(iris, diag_kind='kde', hue='species')
plt.show()
```



```
import seaborn as sns
import numpy as np
import pandas as pd

plt.rcParams['font.family'] = "Malgun Gothic"
plt.rcParams['axes.unicode_minus']=False

iris = sns.load_dataset('iris')
sns.pairplot(iris, diag_kind='kde', hue='species', palette='bright')
plt.show()
```

<https://www.geeksforgeeks.org/how-to-make-a-mosaic-plot-in-matplotlib/>

```
print(sns.get_dataset_names())
```

seaborn에 들어있는 데이터셋의 종류

```
['anagrams', 'anscombe', 'attention', 'brain_networks', 'car_crashes', 'diamonds',  
'dots', 'dowjones', 'exercise', 'flights', 'fmri', 'geyser', 'glue', 'healthexp', 'iris',  
'mpg', 'penguins', 'planets', 'seaice', 'taxis', 'tips', 'titanic']
```

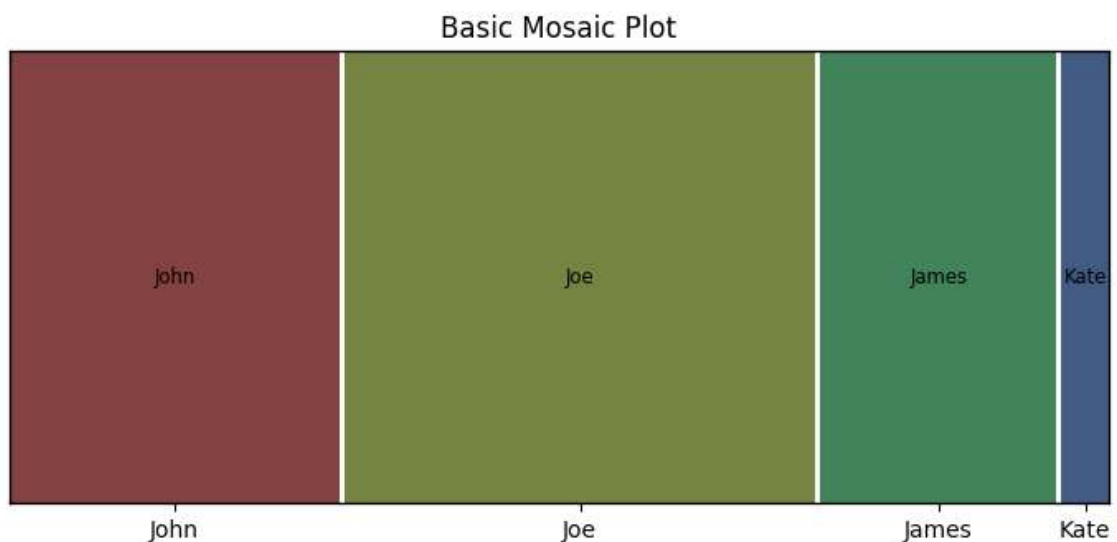
```
pip install matplotlib  
pip install statsmodels
```

1) Example 1: Basic Mosaic Plot

Python

```
import matplotlib.pyplot as plt  
from statsmodels.graphics.mosaicplot import mosaic  
  
plt.rcParams["figure.figsize"] = [7.00, 3.50]  
plt.rcParams["figure.autolayout"] = True  
  
# Dictionary for mosaic plot data  
data = {'John': 7, 'Joe': 10, 'James': 5, 'Kate': 1}  
  
# Create mosaic plot  
mosaic(data, title='Basic Mosaic Plot')  
plt.show()
```

Output:

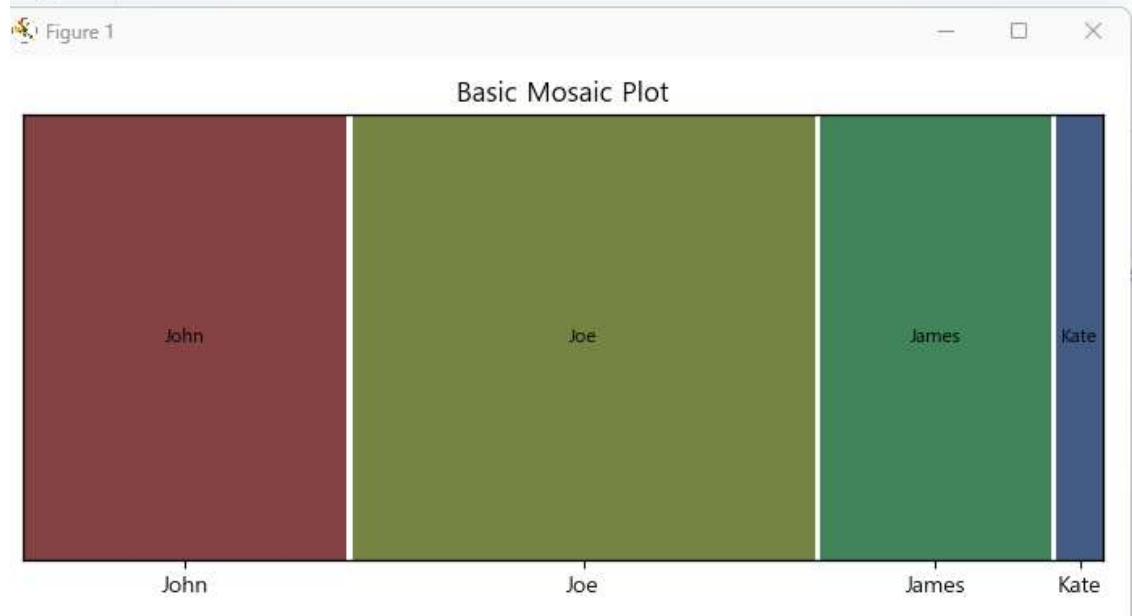


Basic Mosaic Plot

In this example, we create a simple mosaic plot from a dictionary of data. The mosaic function from the

statsmodels.graphics.mosaicplot module is used to create the plot, and plt.show() is used to display it.

```
1  import matplotlib.pyplot as plt
2  import seaborn as sns
3  import numpy as np
4  import pandas as pd
5
6  plt.rcParams['font.family'] = "Malgun Gothic"
7  plt.rcParams['axes.unicode_minus']=False
8
9  import matplotlib.pyplot as plt
10 from statsmodels.graphics.mosaicplot import mosaic
11
12 plt.rcParams["figure.figsize"] = [7.00, 3.50]
13 plt.rcParams["figure.autolayout"] = True
14
15 # Dictionary for mosaic plot data
16 data = {'John': 7, 'Joe': 10, 'James': 5, 'Kate': 1}
17
18 # Create mosaic plot
19 mosaic(data, title='Basic Mosaic Plot')
20 plt.show()
21
```

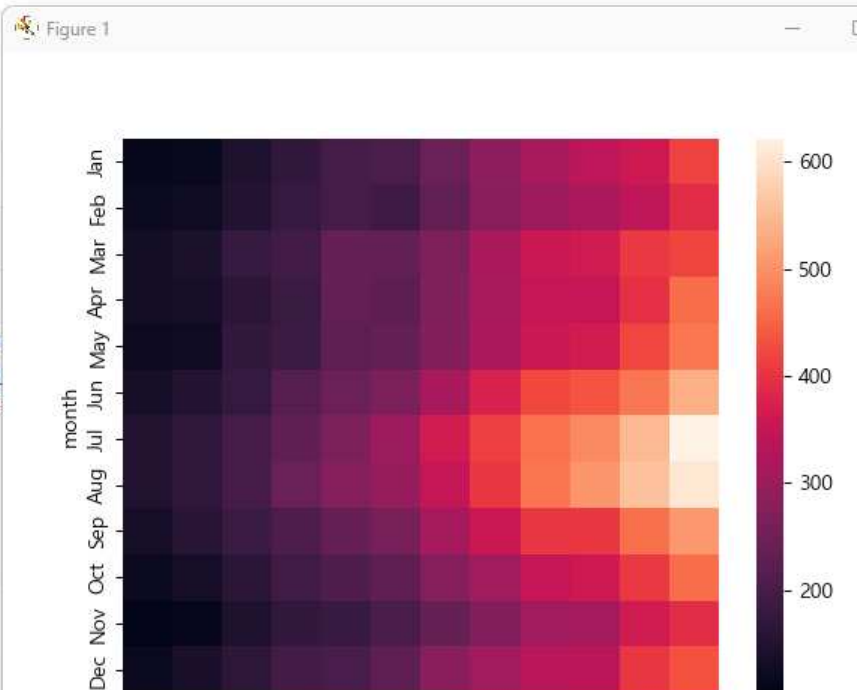


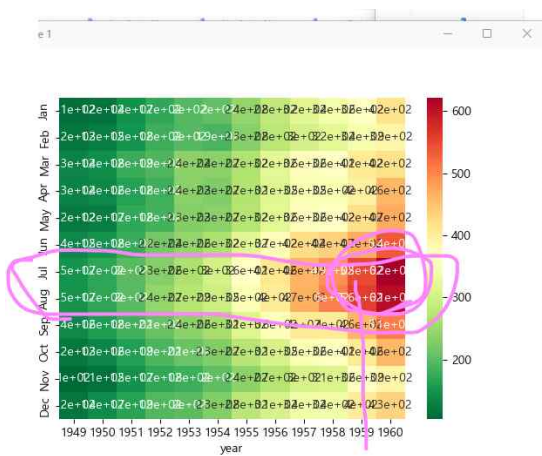
```

1 import matplotlib.pyplot as plt
2 import seaborn as sns
3 import numpy as np
4 import pandas as pd
5
6 import matplotlib.pyplot as plt
7 from statsmodels.graphics.mosaicplot import mosaic
8
9 plt.rcParams['font.family'] = "Malgun Gothic"
10 plt.rcParams['axes.unicode_minus']=False
11
12 flights = sns.load_dataset("flights")
13 #print(flights.shape)
14 #print(flights.head())
15 #print(flights.tail())
16 #print(flights.describe())
17
18 #df = flights.pivot("month", "year", "passenger")
19 df = flights.pivot_table(index="month", columns="year", values="passengers")
20 #print(df)
21
22 sns.heatmap(df)
23 plt.show()

```

U30\pythonProject\.venv
 U30\pythonProject\sns-h
 s.pivot_table(index="mo





```
import pandas as pd

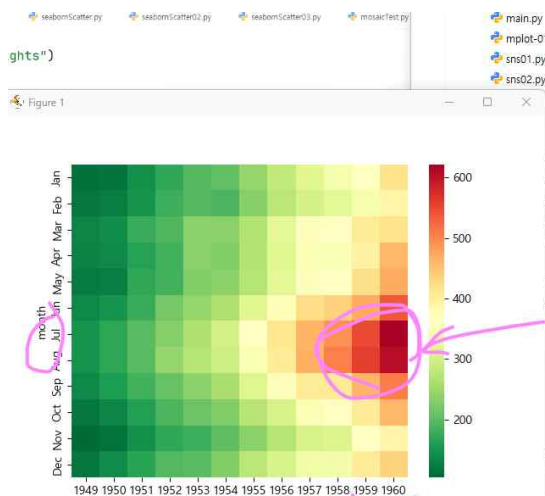
import matplotlib.pyplot as plt
from statsmodels.graphics.mosaicplot import mosaic

plt.rcParams['font.family'] = 'Malgun Gothic'
plt.rcParams['axes.unicode_minus']=False

flights = sns.load_dataset("flights")
#print(flights.shape)
#print(flights.head())
#print(flights.tail())
#print(flights.describe())

#df = flights.pivot("month","year","passenger")
df = flights.pivot_table(index="month", columns="year", values="passengers")
#print(df)

sns.heatmap(df, annot=True, cmap="RdYlGn_r")
plt.show()
```



```
5
6 import matplotlib.pyplot as plt
7 from statsmodels.graphics.mosaicplot import mosaic
8
9 plt.rcParams['font.family'] = 'Malgun Gothic'
10 plt.rcParams['axes.unicode_minus']=False
11
12 flights = sns.load_dataset("flights")
13 #print(flights.shape)
14 #print(flights.head())
15 #print(flights.tail())
16 #print(flights.describe())
17
18 #df = flights.pivot("month","year","passenger")
19 df = flights.pivot_table(index="month", columns="year", values="passengers")
20 #print(df)
21
22 sns.heatmap(df, cmap="RdYlGn_r")
23 plt.show()
24
25
26
```