탐색적 데이터 분석

데이터를 요약하고 시각화하는 것을 통해 프로젝트와 데이터에 대한 가치있는 통찰과 이해를 얻게 됨.
가설 수립, 상관관계 분석, 트랜드 분석 등을 목표로 탐색적 데이터 분석을 수행할 수 있음

In [1]:

```
import matplotlib
from matplotlib import font_manager,rc
#한글 폰트 등록
font_location = "c:/Windows/fonts/malgun.ttf"
font_name = font_manager.FontProperties(fname=font_location).get_name()
matplotlib.rc('font', family=font_name)

import matplotlib.pyplot as plt
%matplotlib inline
```

In [2]:

```
import pandas as pd

students = pd.read_csv("data/students.csv")
students
```

Out[2]:

	english	math	class
0	100	999	1
1	90	90	1
2	80	80	1
3	70	20	1
4	20	90	2
5	90	100	2
6	80	80	2
7	90	99	A

위치 추정

데이터를 살펴보는 가장 기초적인 단계는 각 피쳐(변수)의 '대푯값(typical value)'을 구하는 것
'대푯값(typical value)' - 대부분의 값이 어디쯤에 위치하는지 (중심경향성)을 나타내는 추정 값
평균, 중앙값 등

In [3]:

```
print(students.english.mean())        # "english" 열 값의 평균
print(students.math.mean())           # "math" 열 값의 평균

import numpy as np
print(   np.median(students.english) ) # "english" 열 값의중앙값
print(   np.median(students.math) )    # "math" 열 값의중앙값
```

```
77.5
194.75
85.0
90.0
```

변이 추정

변이는 데이터 값이 얼마나 밀집해 있는지 혹은 퍼져 있는 지를 나타내는 산포도(dispersion)
가장 유명한 방법은 분산과 표준 편차. 그 외에 평균절대편차, 범위, 백분위수, 사분위수 등

표준 편차와 관련 추정 값들

- 평균 절대 편차

$$AD = \frac{\sum|X_i - \bar{X}|}{n}$$

- 중위 절대 편차

$$MAD = \frac{\sum|X_i - 중간값|}{n}$$

- 분산

$$s^2 = \frac{\sum(X_i - \bar{X})^2}{n-1}$$

- 표준 편차

$$S = \sqrt{S^2} = \sqrt{\frac{\sum(X_i - \bar{X})^2}{n-1}}$$

In [4]:

```
print(students.english.std())        # "english" 열 값의 표준편차
print(students.math.std())           # "math" 열 값의 표준편차
```

```
24.9284690951645
325.96702287194637
```

In [5]:

```
students.describe()
```

Out[5]:

	english	math
count	8.000000	8.000000
mean	77.500000	194.750000
std	24.928469	325.967023
min	20.000000	20.000000
25%	77.500000	80.000000
50%	85.000000	90.000000
75%	90.000000	99.250000
max	100.000000	999.000000

분포 탐색

In [6]:

```
## 사분위수

print(np.percentile(students.english, 0 )) # 최소값
print(np.percentile(students.english, 25 )) # 1/4
print(np.percentile(students.english, 50 )) # 2/4
print(np.percentile(students.english, 75 )) # 3/4
print(np.percentile(students.english, 100 )) # 최대값
```

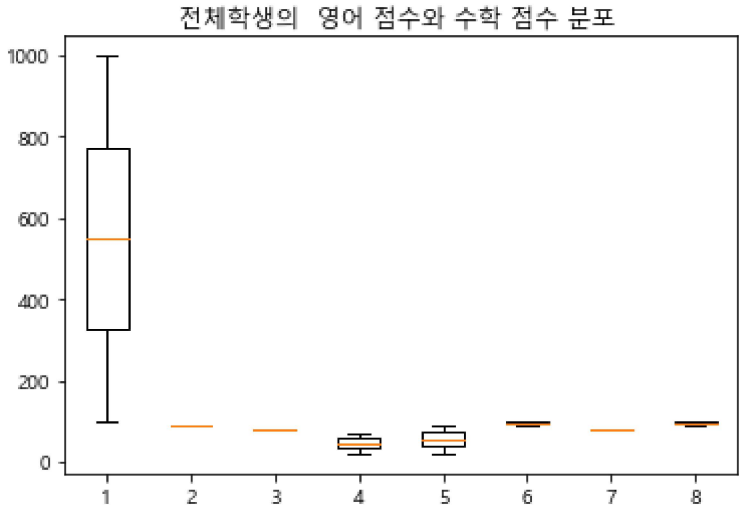
```
20.0
77.5
85.0
90.0
100.0
```

상자그래프

(연속형 변수 분포)

In [7]:

```
plt.boxplot(students.loc[  : , ('english','math') ].T)
plt.title("전체학생의  영어 점수와 수학 점수 분포 ")
plt.show()
```

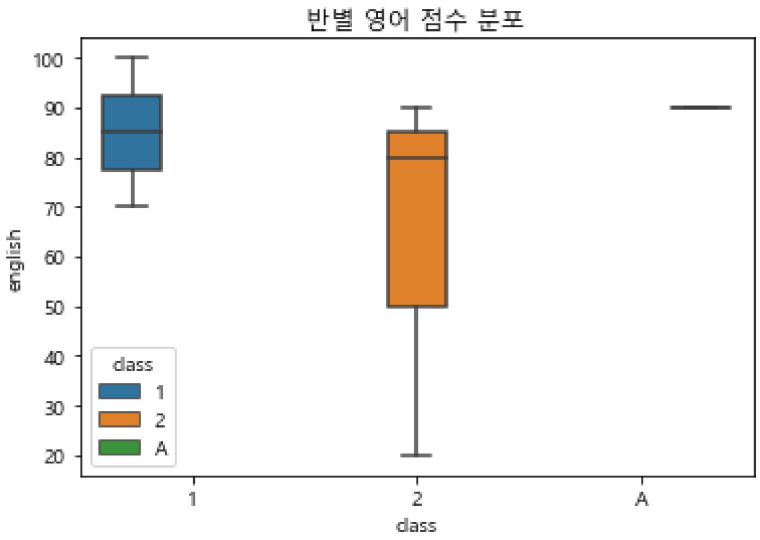


In [8]:

```
import seaborn as sns
sns.boxplot(x="class", y='english'  ,data=students  , hue="class" )
plt.title(" 반별 영어 점수 분포 ")
```

Out[8]:

Text(0.5, 1.0, ' 반별 영어 점수 분포 ')

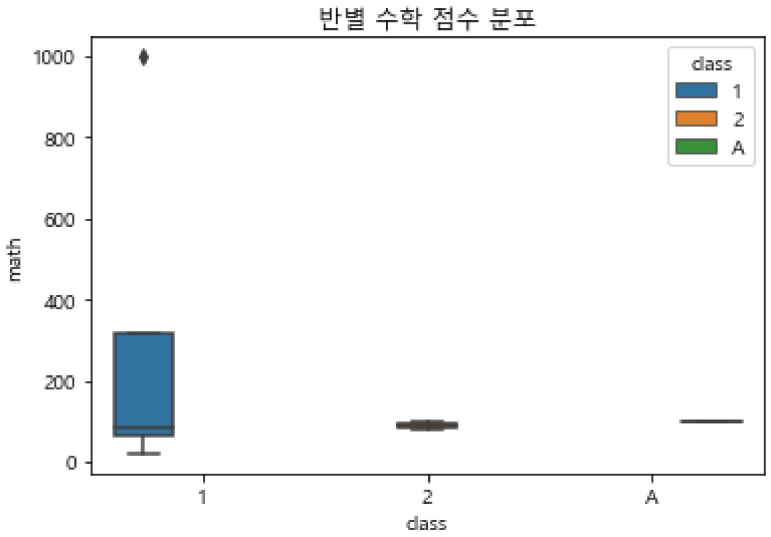


In [9]:

```
import seaborn as sns
sns.boxplot(x="class", y='math'  ,data=students  , hue="class" )
plt.title(" 반별 수학 점수 분포 ")
```

Out[9]:

Text(0.5, 1.0, ' 반별 수학 점수 분포 ')



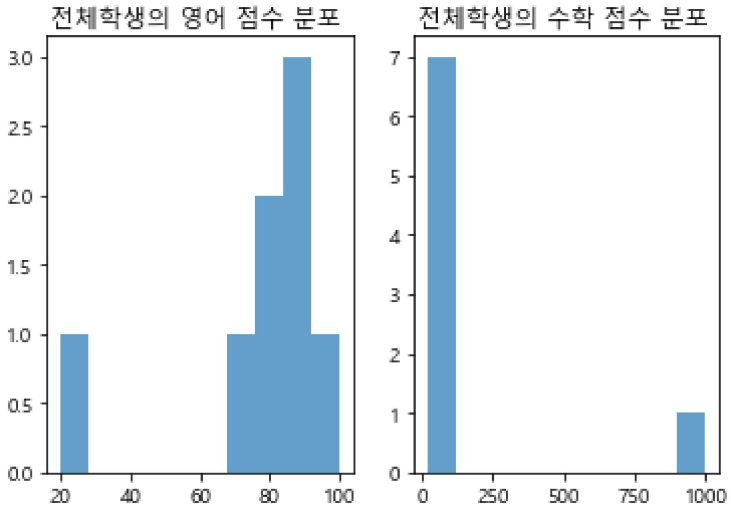
히스토그램

(연속형 변수 분포)

In [10]:

```
plt.figure()
plt.subplot(1,2,1)  # 1행 2열 그래프의 첫번째 그래프
plt.hist(students["english"], bins=10,  alpha=0.7, histtype='stepfilled')
plt.title ("전체학생의 영어 점수 분포 ")


plt.subplot(1,2,2) # 1행 2열 그래프의 두번째 그래프
plt.hist(students["math"], bins=10,   alpha=0.7, histtype='stepfilled')
plt.title ("전체학생의 수학 점수 분포 ")
plt.show()
```



범주형 데이터 탐색

파이 그래프

파이 차트 (Pie chart, 원 그래프)는 범주별 구성 비율을 원형으로 표현한 그래프입니다.

In [11]:

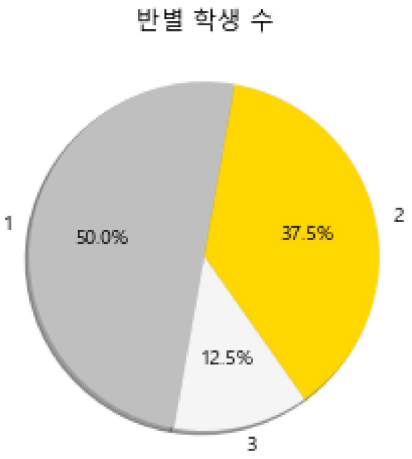
```
import matplotlib.pyplot as plt
%matplotlib inline

colors = ['silver', 'gold', 'whitesmoke']

data = students["english"].groupby(students['class'])
print( data.mean() )  # 반별 영어 점수 평균
print( data.size() )  # 반별 학생수

plt.pie(data.size(), labels=[1,2,3], autopct='%.1f%%', startangle=260, counterclock=False,  shadow=
plt.title("반별 학생 수")
plt.show()
```

```
class
1    85.000000
2    63.333333
A    90.000000
Name: english, dtype: float64
class
1    4
2    3
A    1
Name: english, dtype: int64
```



막대 그래프

(범주형변수, 이산형 변수 분포)

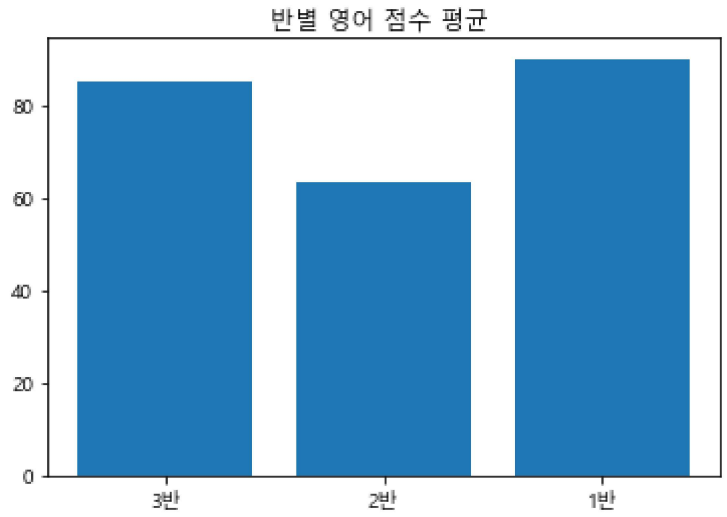
In [12]:

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

data = students["english"].groupby(students['class'])
print( data.mean() )  # 반별 영어 점수 평균

years = ['3반', '2반', '1반']
plt.title("반별 영어 점수 평균 ")
plt.bar(years, data.mean())
plt.show()
```

```
class
1    85.000000
2    63.333333
A    90.000000
Name: english, dtype: float64
```

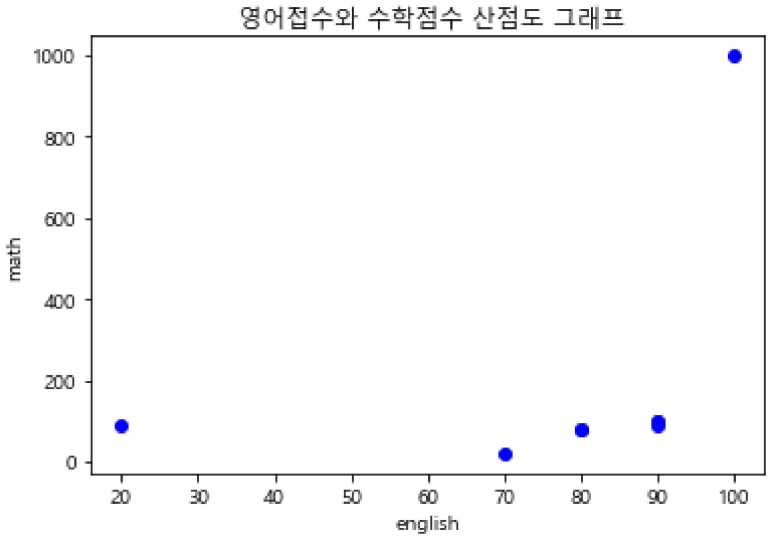


관계시각화

산점도 그래프

In [13]:

```
plt.plot( students["english"], students["math"], 'bo')
plt.xlabel('english')
plt.ylabel('math')
plt.title(" 영어접수와 수학점수 산점도 그래프")
plt.show()
```



In [14]:

```
import seaborn as sns

# tips = sns.load_dataset("iris")
iris = sns.load_dataset("iris")# seaborn 팩키지의 샘플 데이터
iris.head()
```

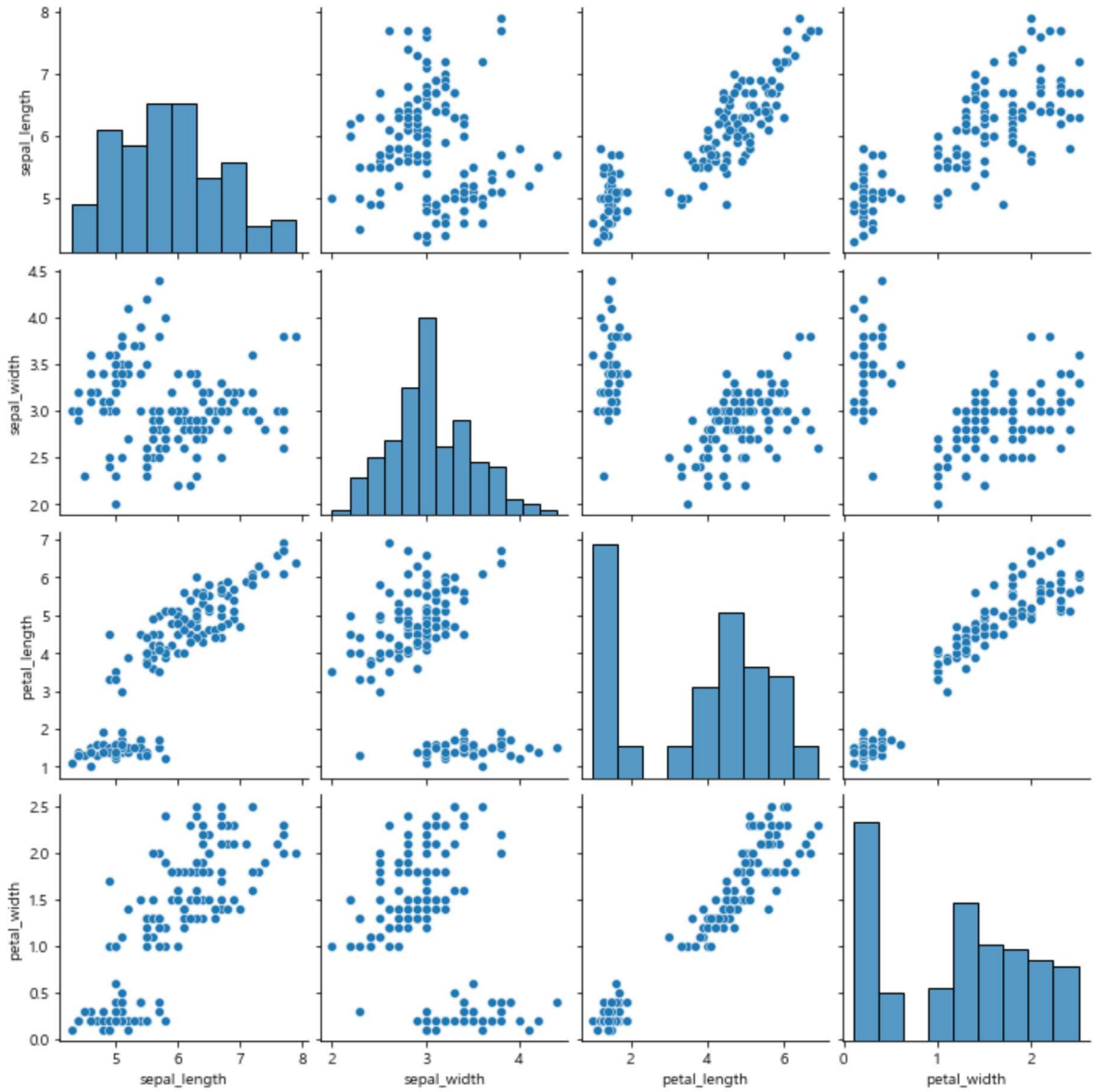
Out[14]:

	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	setosa
1	4.9	3.0	1.4	0.2	setosa
2	4.7	3.2	1.3	0.2	setosa
3	4.6	3.1	1.5	0.2	setosa
4	5.0	3.6	1.4	0.2	setosa

In [15]:

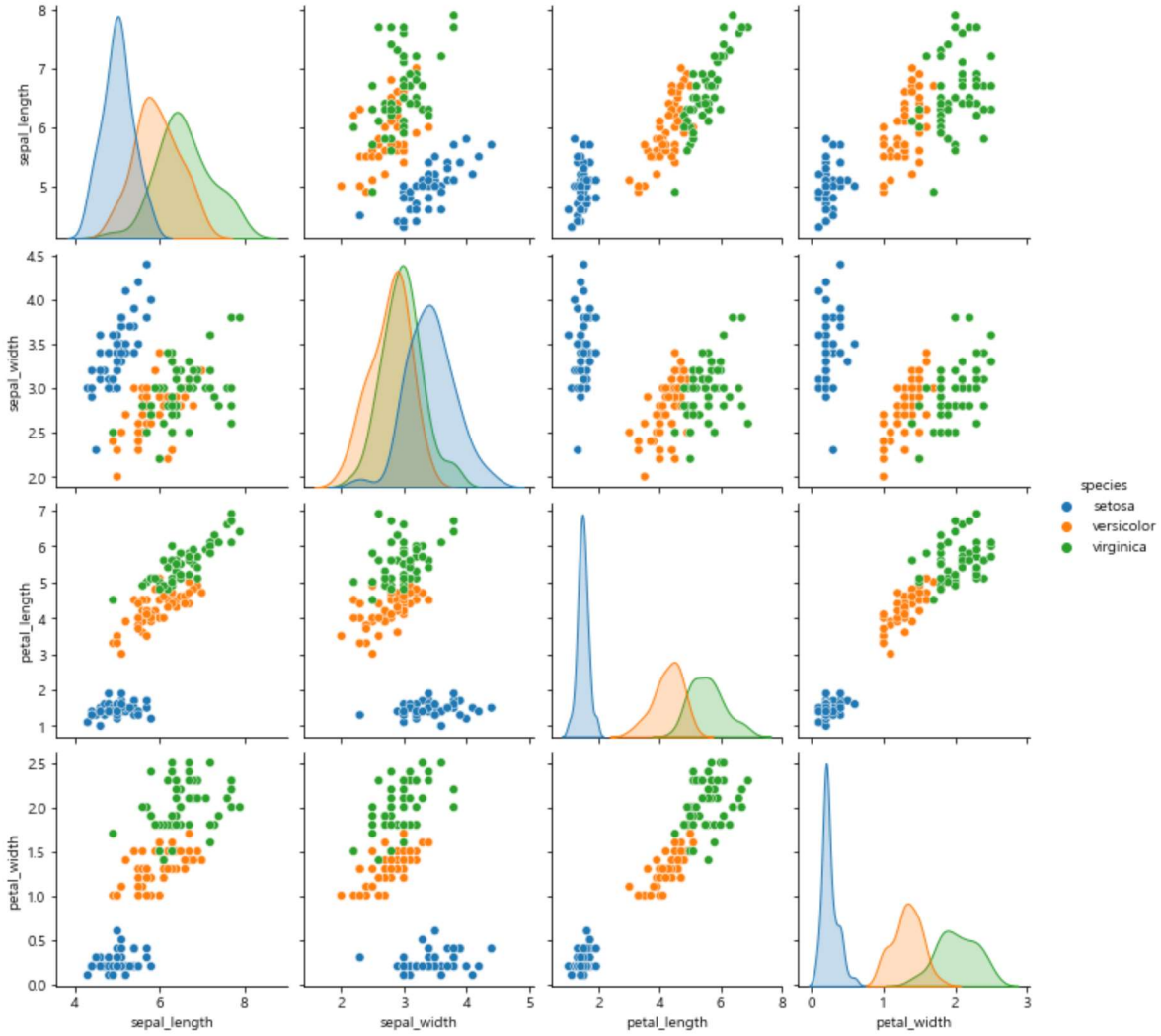
```
import matplotlib.pyplot as plt
%matplotlib inline

sns.pairplot(iris)
plt.show()
```



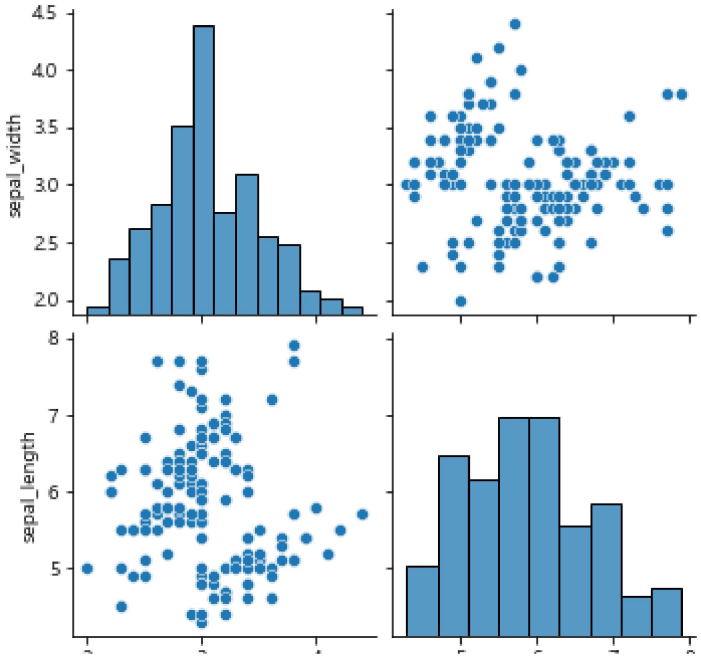
In [16]:

```
sns.pairplot(iris, hue="species")
plt.show()
```



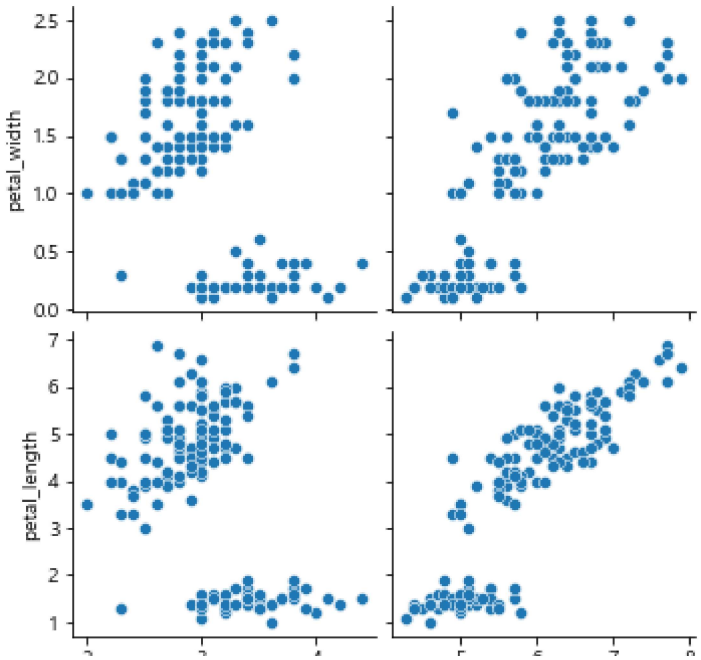
In [17]:

```
sns.pairplot(iris, vars=["sepal_width", "sepal_length"])
plt.show()
```



In [18]:

```
sns.pairplot(iris, x_vars=["sepal_width", "sepal_length"],
             y_vars=["petal_width", "petal_length"])
plt.show()
```



피어슨 상관계수

- **상관관계**
  - X가 큰 값을 가지면, Y도 큰 값을 갖고, X가 작은 값을 가지면, Y도 작은 값을 갖는 경우, 변수 X와 Y는 서로 **양의 상관 관계**를 갖는다고 말함.
  - 반대로, X가 큰 값을 가지면, Y은 작은 값을 갖고, X가 작은 값을 가지면, Y는 큰 값을 갖는 경우, 변수 X와 Y는 서로 **음의 상관 관계**를 갖는다고 말함.
- **상관계수**
  - 수치적으로 변수 간의 어떤 관계가 있는지 나타내기 위해 사용되는 측정량
  - 피어슨 상관 계수
    - 선형적인 상관 관계를 -1 ~1 사이의 값으로 표현

In [19]:

```
students.loc[:, ("english" ,"math" ) ].corr(method='pearson')
```

Out[19]:

	english	math
english	1.000000	0.372093
math	0.372093	1.000000

In [20]:

```
%matplotlib inline
import matplotlib.pyplot as plt
import seaborn as sns
```

In [21]:

```
sns.heatmap(data = students.corr(), annot=True,  fmt = '.2f', linewidths=.5, cmap='Blues')


# 내부 속성중 annot 은 annotation. 각 셀의 값을 표시할지 결정하는 것이고,
# fmt 는  annot=True 인 경우에, 숫자 표시
# (.2f 는  소수 두번째자리까지 표시 )
# cmap 은 색상
```

Out[21]:

<AxesSubplot:>