
타생적 데이터 분석을 통한 지역사회 문제 해결

창의융합 A반

이이화 강사

1. 공공데이터 수집

1.1 국가통계포털 소개

국가통계포털은 국가승인 통계를 국민에게 서비스하기 위하여 통계정보를 한 곳에서 검색, 분석·활용할 수 있도록 대한민국 통계청에서 구축하여 운영하고 있는 사이트로, 국내 및 국제, 북한의 주요통계를 한 곳에 모아 이용자가 원하는 통계를 한 번에 찾을 수 있도록 통계청이 제공하는 One-Stop통계 서비스입니다.

국가통계포털 주소

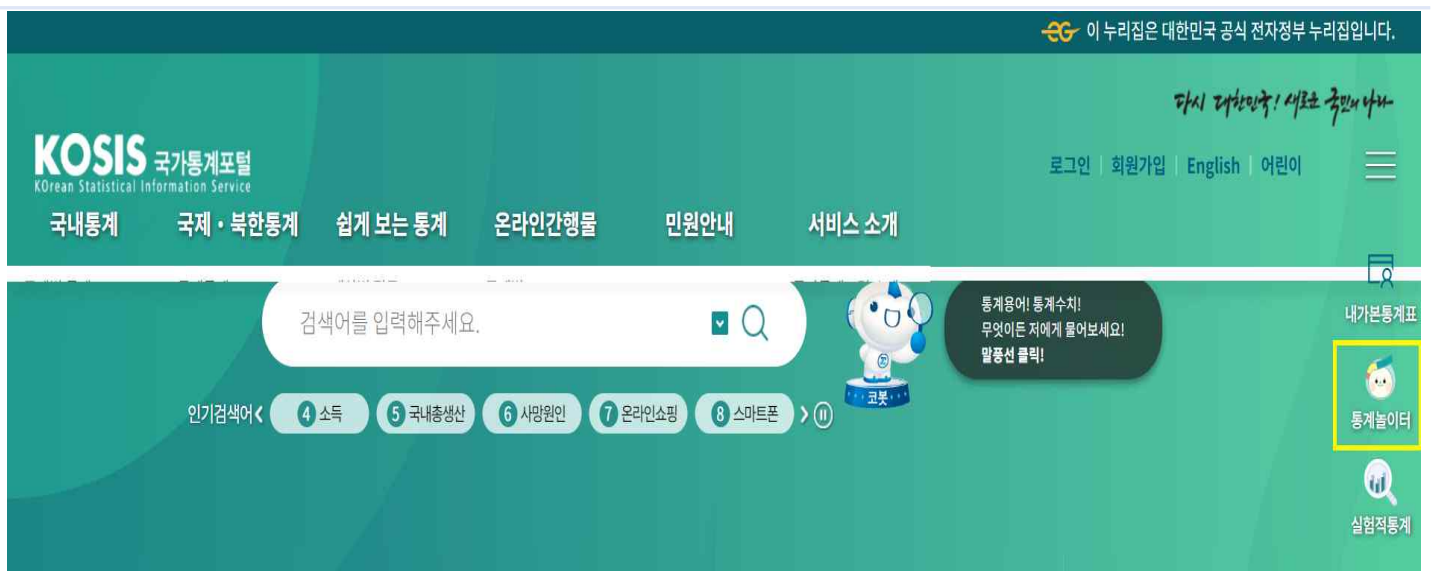
<https://kosis.kr/index/index.do>

1.2 공공데이터 수집방법

- ① 포털에서 “국가통계포털” 검색합니다.



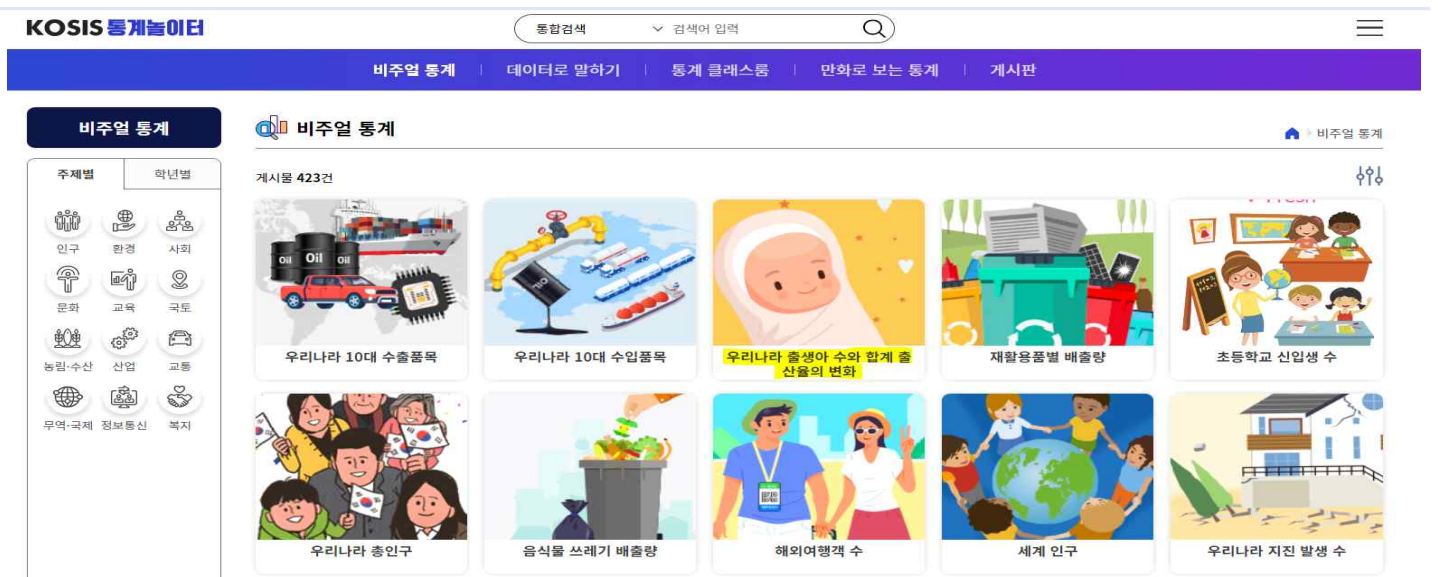
- ② KOSIS 국가통계포털 - 통계놀이터를 클릭합니다.



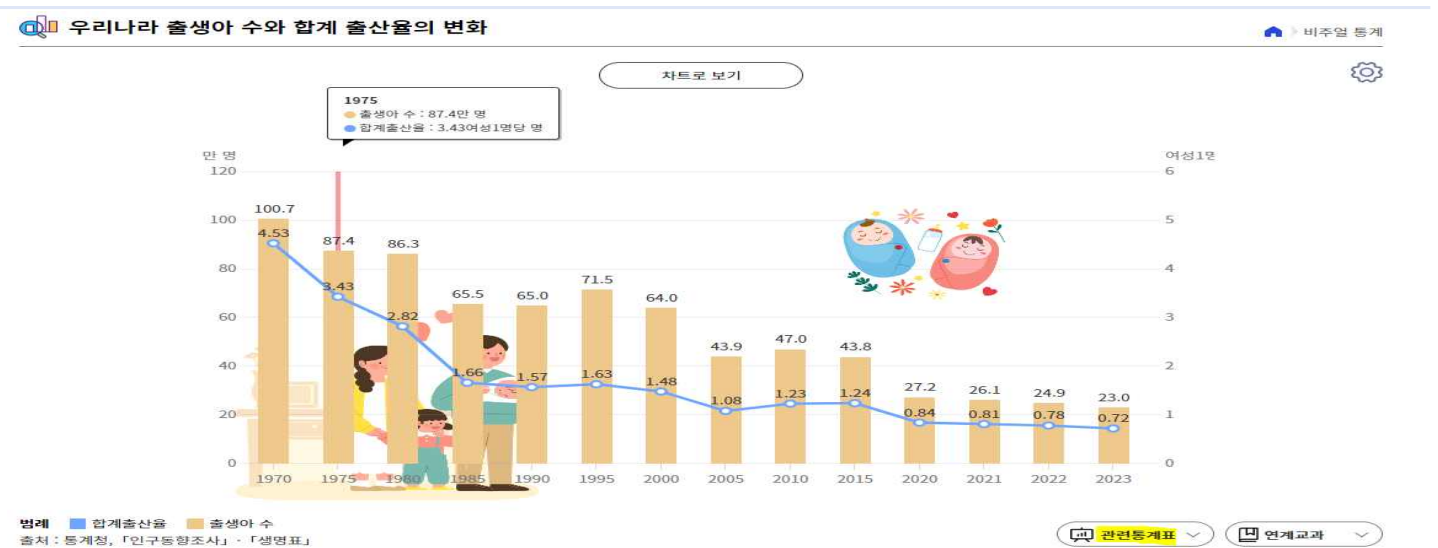
③ KOSIS통계놀이터 - 비주얼통계 - ⌚ 를 클릭합니다.



④ 비주얼통계 - “우리나라 출생아 수와 합계출산율의 변화”를 클릭합니다.



⑤ “우리나라 출생아 수와 합계출산율의 변화” - 관련통계표를 클릭합니다.



⑥ 관련통계표 - “인구동태건수 및 동태을추이(출생,사망,혼인,이혼)” 클릭합니다.

관련통계표

인구동태건수 및 동태을 추이(출생,사망,혼인,이혼)

보관함

다운로드

연계교과

과이미점 : 초등 3학년 2학기 수학 23p

- 지학사 : 중등 사회1 230p
- 지학사 : 초등 4학년 2학기 사회 105p
- 천재교과서 : 중등 사회1 229p
- 천재교과서 : 중등 사회과부도 97p
- 천재교과서 : 초등 4학년 2학기 사회 112p
- 천재교과서 : 초등 4학년 2학기 수학 109p
- 천재교과서 : 초등 4학년 2학기 수학익힘 75p
- 천재교육 : 고등 사회문화 181p

⑦ KOSIS “인구동태건수 및 동태을추이(출생,사망,혼인,이혼)” - 조회설정을 클릭합니다.

KOSIS

통계목록 : 주제별 통계

스크랩

내가 본 통계표

통계목록

단어검색

검색

맨위로

오동자순

인구동태건수 및 동태을 추이(출생,사망,혼인,이혼)

시도/인구동태건수 및 동태을 추이(출생,사망,혼인,이혼)

시군구/인구동태건수 및 동태을 추이(출생,사망,혼인,이혼)

읍면동/성별/인구동태건수(출생,사망,혼인,이혼)

출생

사망

혼인

이혼

다문화

장래인구추계

국내인구이동통계

출입국자및체류외국인통계

지방자치단체외국인주민현황

장래가구추계

국제인구이동통계

고령친화용품제조업실태조사

사회일반

범죄·안전

노동

1. 인구동태건수 및 동태을 추이(출생,사망,혼인,이혼)

모두 닫기

인구동태건수 및 동태을 추이(출생,사망,혼인,이혼)

「인구동향조사」 통계청 (자료문의처: 042-481-2258)

통계설명자료

온라인간행물

보도자료

수색기간 : 년 1970 ~ 2023 / 자료경신일: 2024-03-19 / 추적정보

시점

증감/증감률

행렬전환

열고정해제

새 열 열기

화면복사

주소/출처

스크랩

OPENAPI

인쇄

다운로드

조회설정

기본항목별	2021	2022	2023 p)
출생아수(명)	260,562	249,186	230,000
사망자수(명)	317,680	372,939	352,700
자연증가건수(명)	-57,118	-123,753	-122,800
조출생률(천명당)	5.1	4.9	4.5
조사망률(천명당)	6.2	7.3	6.9
자연증가율(천명당)	-1.1	-2.4	-2.4
합계출산율(명)	0.808	0.778	0.720
출생성비(명)	105.1	104.7	105.1
영아사망률(출생아 천명당)	2.4	2.3	-
혼인건수(건)	192,507	191,690	193,657
조혼인율(천명당)	3.8	3.7	3.8
이혼건수(건)	101,673	93,232	92,394
조이혼율(천명당)	2.0	1.8	1.8
기대수명(출생시 기대여명)	83.6	82.7	-
기대수명(출생시 기대여명) - 남	80.6	79.9	-
기대수명(출생시 기대여명) - 여	86.6	85.6	-

⑧ 조회조건 - 항목 - “인구동태건수 및 동태을 추이”에 체크합니다.

조회조건

부가기능

조회

항목

☒ 인구동태건수 및 동태을 추이

기본항목별

시점

⑨ 조회조건 - 기본항목별 - “출생아수, 사망자수, 합계출산율”에 체크합니다.

조회조건 부가기능

Q 조회

▶ 항목

▼ 기본항목별

검색하기 검색

☐ 전체선택

- ☐ 계
- ☒ 출생아수(명)
- ☒ 사망자수(명)
- ☐ 자연증가건수(명)
- ☐ 조출생률(천명당)
- ☐ 조사망률(천명당)
- ☐ 자연증가율(천명당)
- ☒ 합계출산율(명)

⑩ 조회조건 - 시점(년)- “2015 ~ 2023”에 체크한 다음 “조회”를 클릭합니다.

조회조건 부가기능

Q 조회

▶ 항목

▶ 기본항목별

▼ 시점

✓ 년

2015 ~ 2023

☒ 전체선택

- ☒ 2023
- ☒ 2022
- ☒ 2021
- ☒ 2020
- ☒ 2019
- ☒ 2018
- ☒ 2017
- ☒ 2016
- ☒ 2015

⑪ 조회된 데이터를 확인 후 상단의 “다운로드”를 클릭합니다.

1. 인구통계건수 및 통계를 추... X 모두 닫기

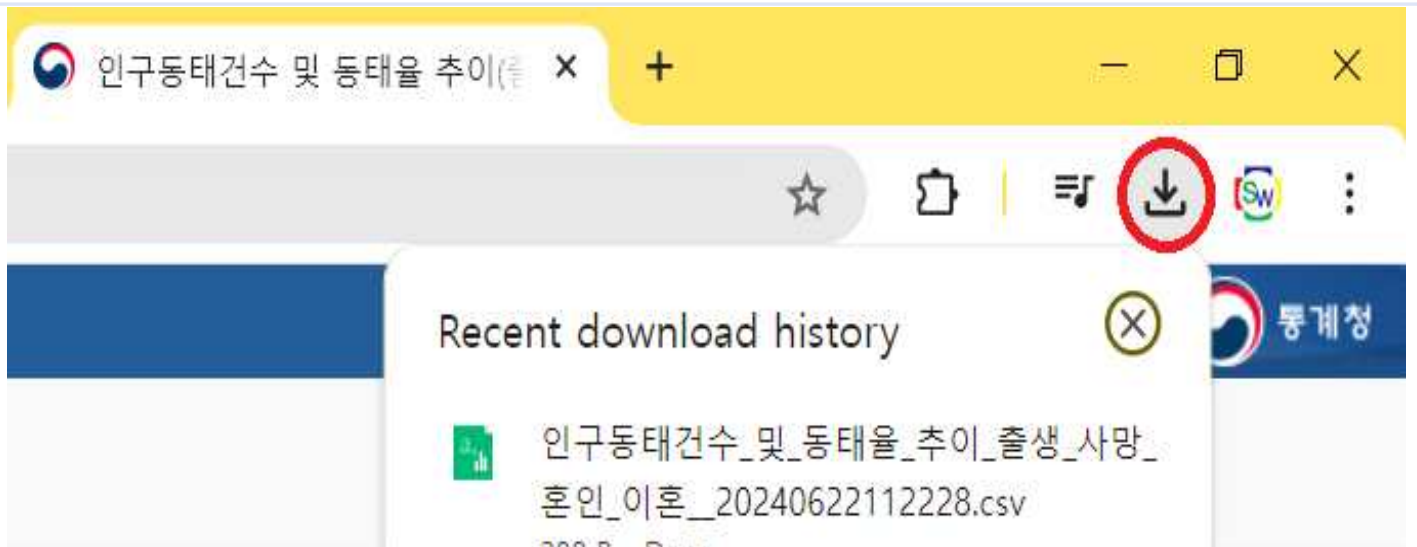
수록기간: 년 1970 ~ 2023 / 자료갱신일: 2024-03-19 / 주석정보

시점 증감/증감률 행렬전환 열고정해제 새 탭 열기 화면복사 주소/출처 스크랩 OPENAPI 인쇄 **다운로드** 조회설정

기본항목별	2015	2016	2017	2018	2019	2020	2021	2022	2023
출생아수(명)	438,420	406,243	357,771	326,822	302,676	272,337	260,562	249,186	
사망자수(명)	275,895	280,827	285,534	298,820	295,110	304,948	317,680	372,939	
합계출산율(명)	1.239	1.172	1.052	0.977	0.918	0.837	0.808	0.778	

⑫ 파일형태 - CSV(인코딩 : ANSI)로 설정한 다음 다운로드를 클릭합니다.

⑬ 상단의 ☰ 에서 다운로드되는지 확인합니다.



⑭ 아래와 같이 다운로드된 CSV파일을 확인합니다.

	A	B	C	D	E	F	G	H	I	J
	기본항목별	2015	2016	2017	2018	2019	2020	2021	2022	2023 p)
1	출생아수(명)	438420	406243	357771	326822	302676	272337	260562	249186	230000
2	사망자수(명)	275895	280827	285534	298820	295110	304948	317680	372939	352700
3	합계출산율(명)	1.239	1.172	1.052	0.977	0.918	0.837	0.808	0.778	0.72

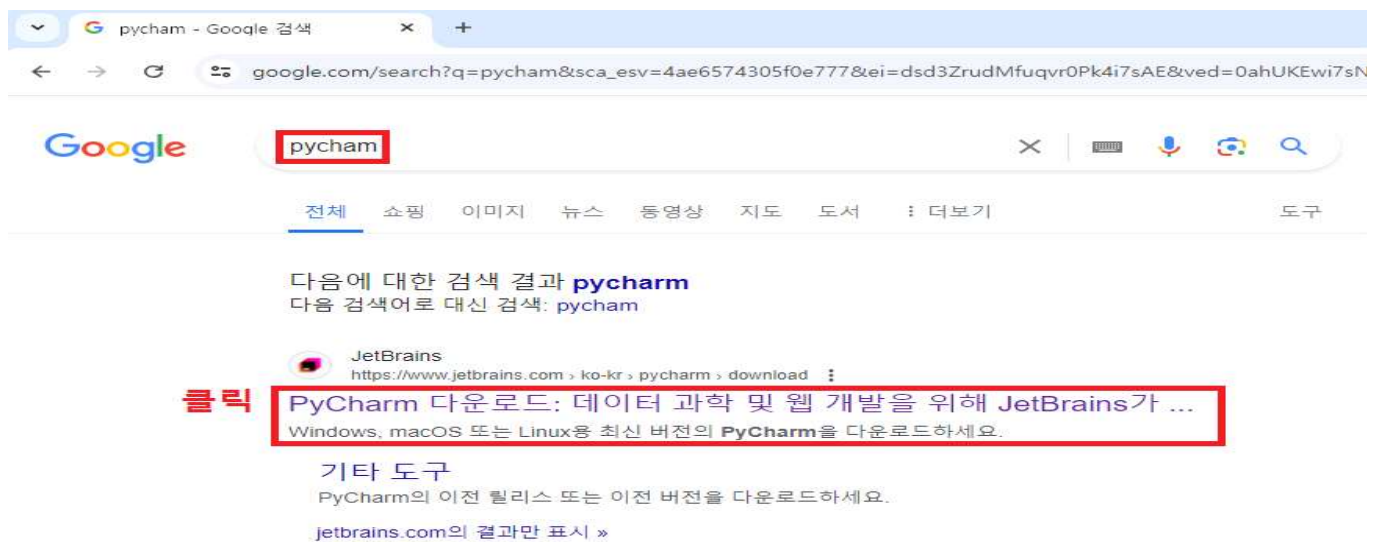
2. 파이참(Pycharm) 활용

2.1 파이참(Pycharm) 소개

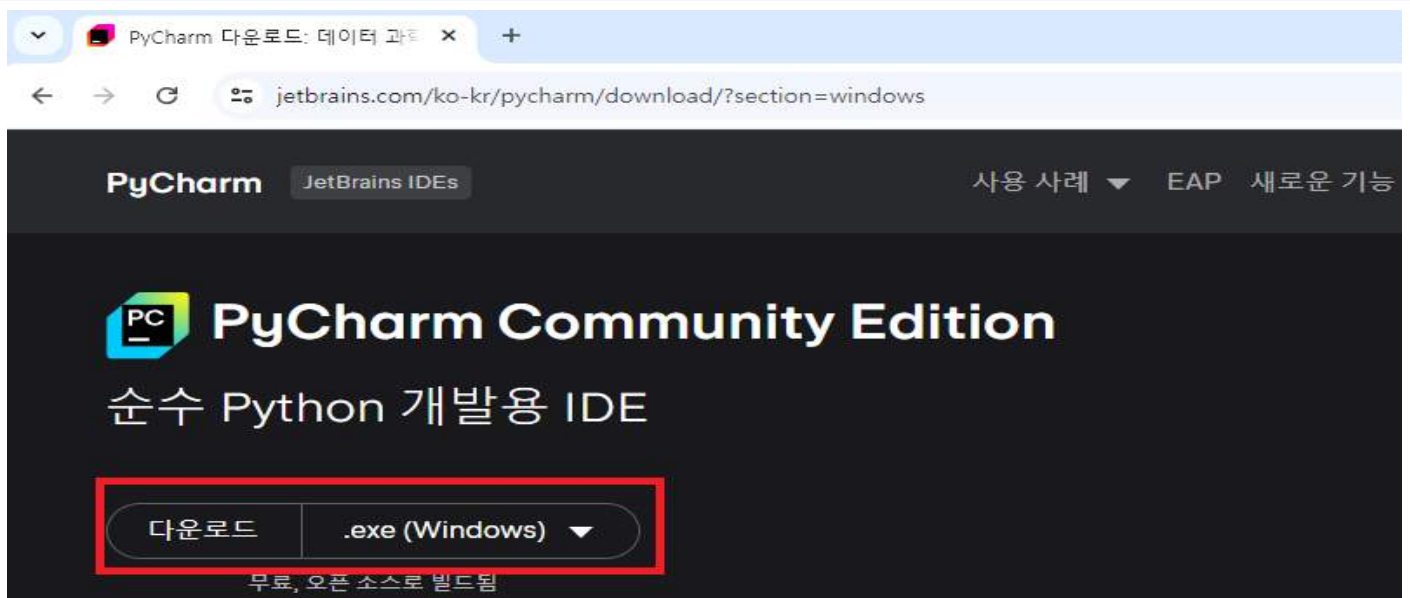
Pycharm은 특히 파이썬 프로그래밍 언어에 특화된 컴퓨터 프로그래밍에 사용되는 통합개발환경으로, 코드 분석, 그래픽얼 디버거, 통합단위 시험기, 버전 관리 시스템과의 연동을 제공하고 장고, 그리고 아나콘다가 있는 데이터 사이언스를 지원합니다.

2.2 파이참(Pycharm) 설치방법

- ① 크롬에서 pycharm을 검색해서 Pycharm 다운로드 사이트를 클릭합니다.



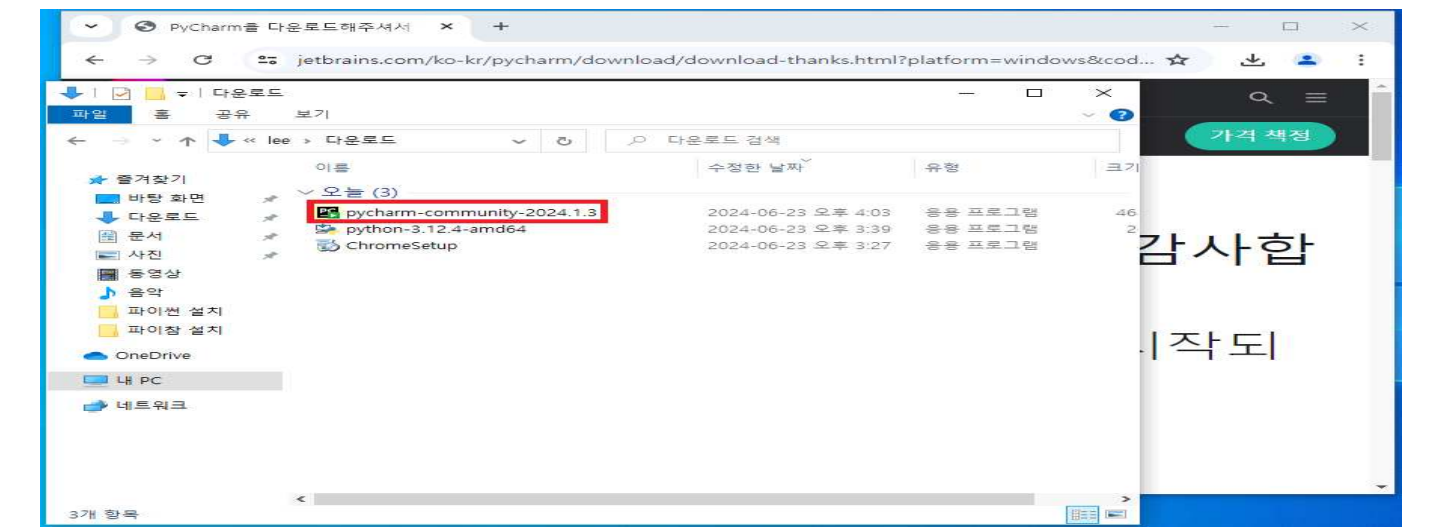
- ② PyCharm Community Edition을 다운로드 받습니다.



③ 폴더 열기를 해서 다운로드 받은 폴더로 이동합니다.



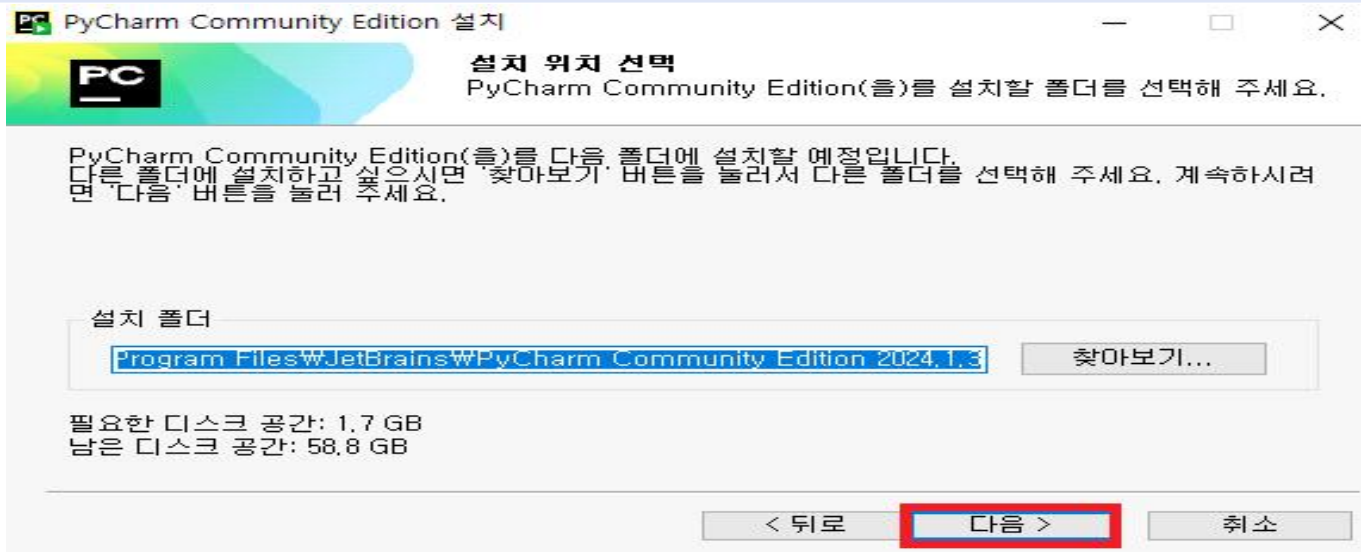
④ 다운로드 받은 설치 파일을 더블 클릭합니다.



⑤ 다음을 클릭합니다.



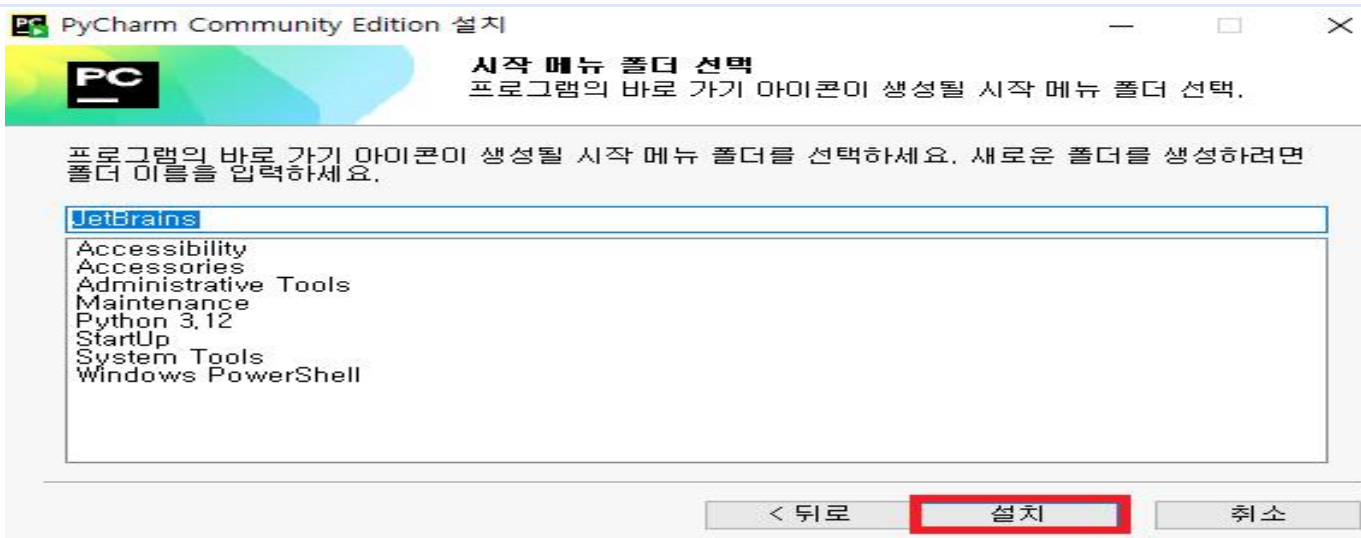
⑥ 다음을 클릭합니다.



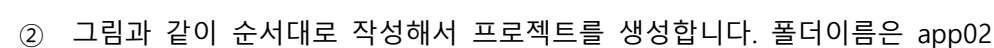
⑦ 체크박스를 모두 체크하고 다음을 클릭합니다.



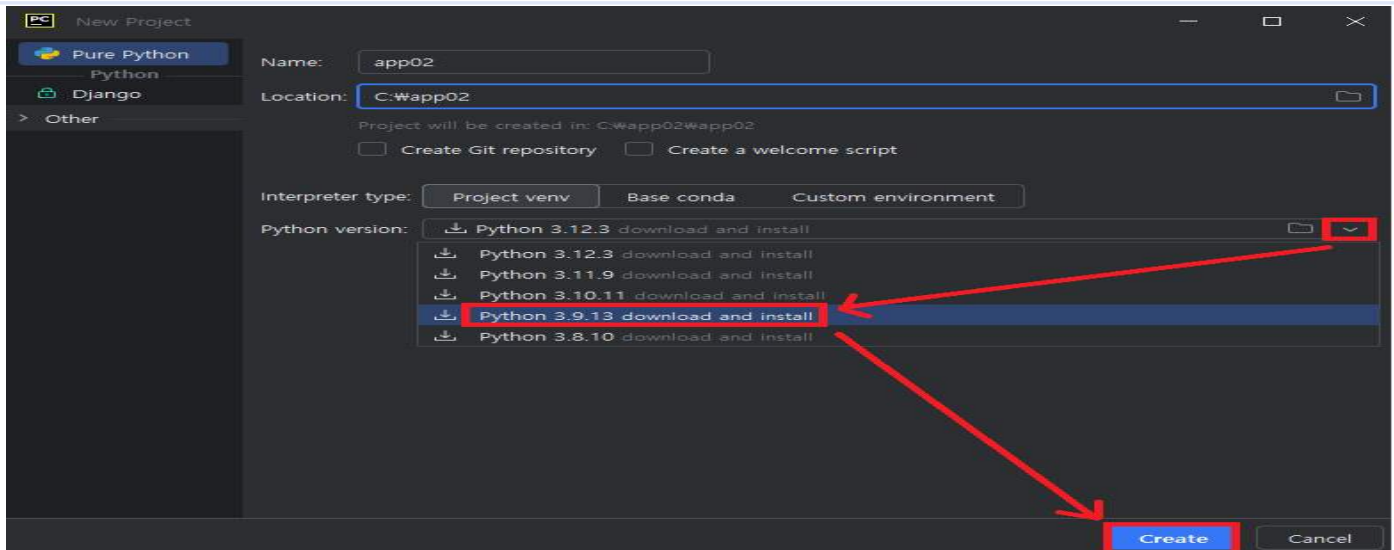
⑧ 설치를 클릭합니다.



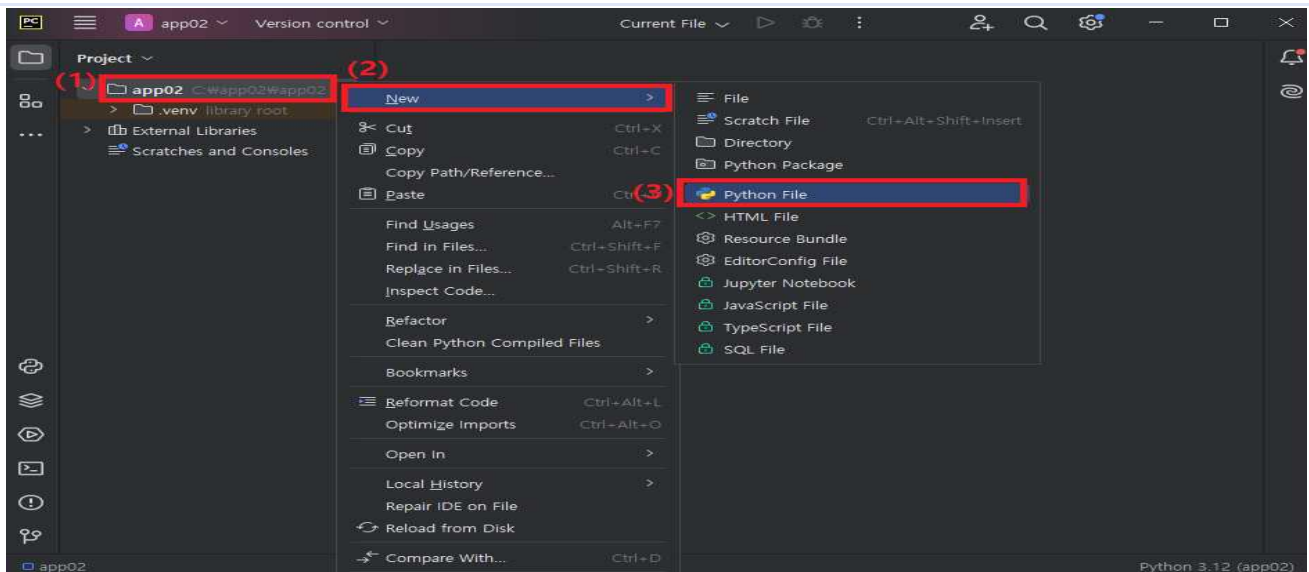
① New Project를 클릭합니다.



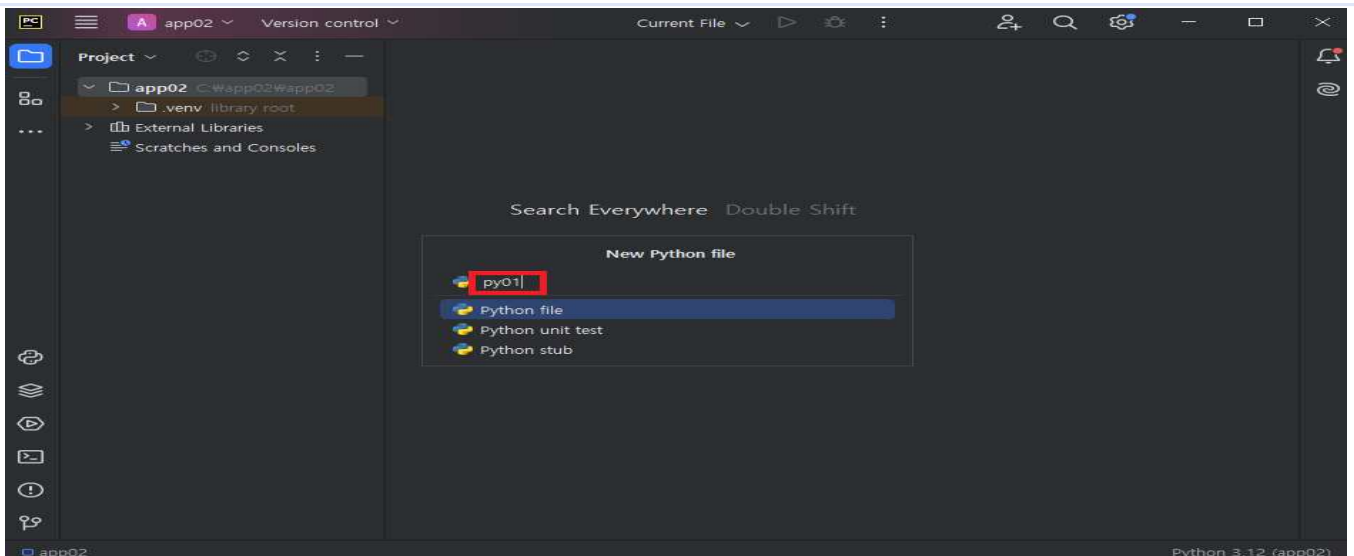
③ Python version을 3.9.13으로 선택합니다. Create를 클릭합니다.



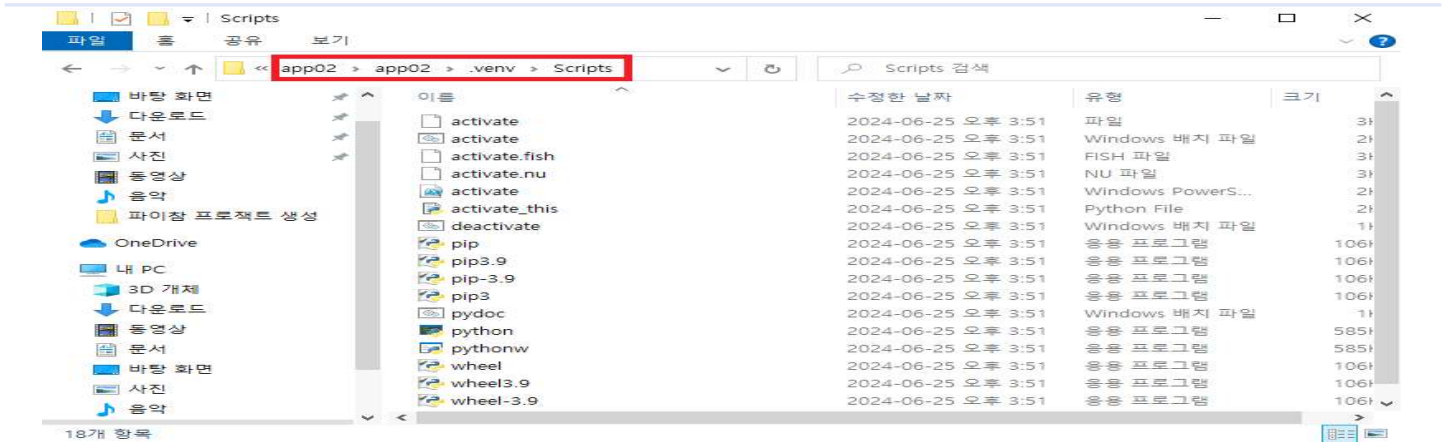
④ app02 폴더에 마우스 오른쪽 버튼을 클릭합니다. New에서 Python File을 선택합니다.



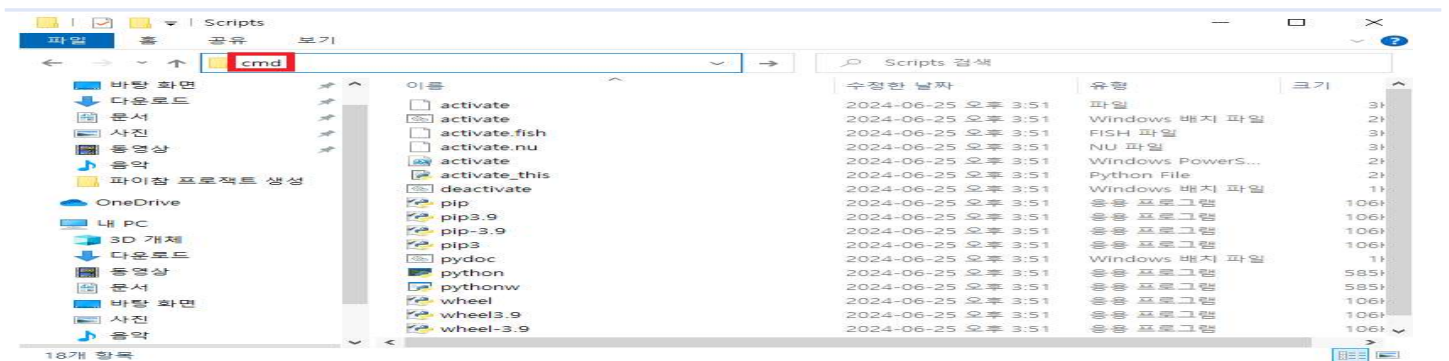
⑤ 파일 이름을 py01로 입력합니다. 파이썬 파일이 생성됩니다.



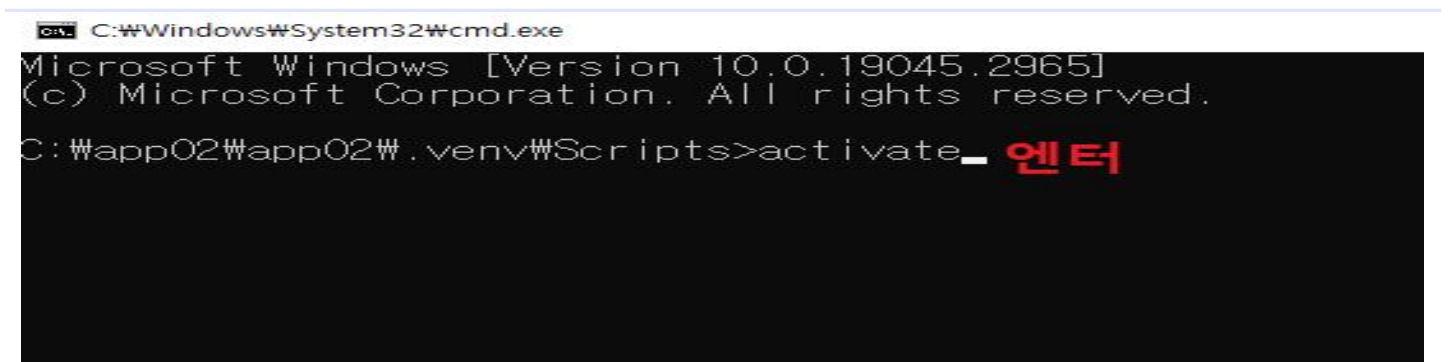
⑥ 파일 탐색기에서 프로젝트가 있는 위치로 이동합니다. C:\app02\app02\.venv\Scripts



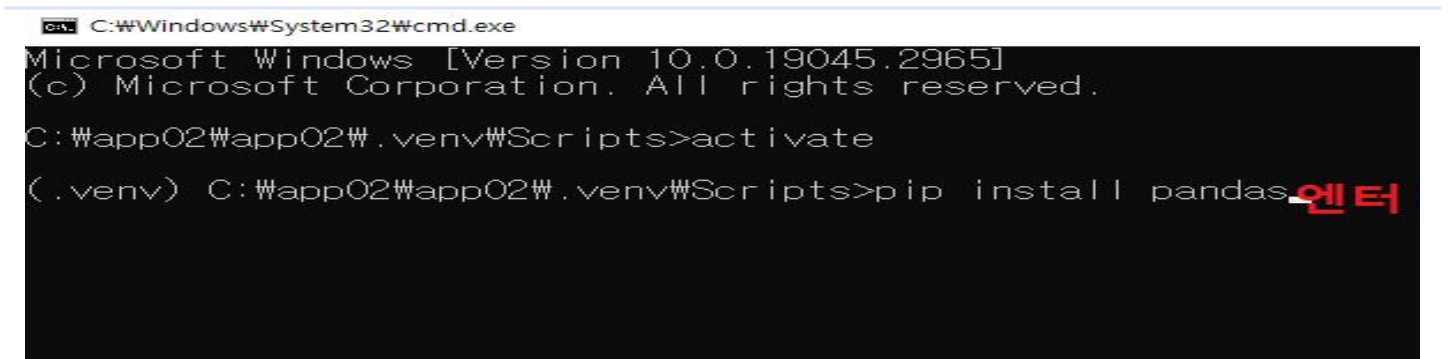
⑦ 그림과 같이 cmd를 입력하고 엔터를 누릅니다.



⑧ activate를 입력하고 엔터를 누릅니다. 가상환경이 실행됩니다.



⑨ 판다스 라이브러리를 설치합니다. pip install pandas

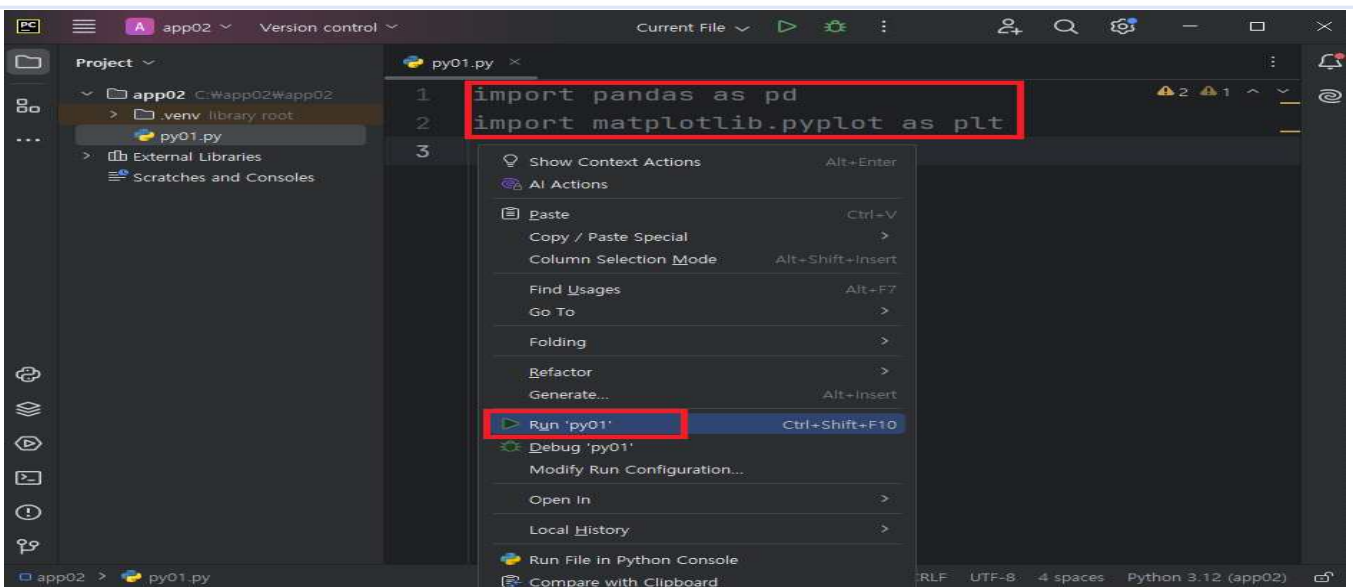


⑩ matplotlib 라이브러리를 설치합니다. `pip install matplotlib`

C:\Windows\System32\cmd.exe

```
(.venv) C:\app02\app02\venv\Scripts>pip install matplotlib_엔터
```

⑪ 실행. 마우스 오른쪽 버튼 클릭 → Run 클릭.



실행되지 않을 경우 파워셸에서 다음과 같이 진행합니다.

⑫ `pip install msvc-runtime`

관리자: Windows PowerShell

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

새로운 크로스 플랫폼 PowerShell 사용 https://aka.ms/pscore6

PS C:\Windows\system32> pip install msvc-runtime
```

3. 맵플로우립(Matplotlib)기본 사용법

3.1 맵플로우립(Matplotlib) 소개

Matplotlib는 Python 프로그래밍 언어 및 수학적 확장 NumPy 라이브러리를 활용한 플로팅 라이브러리로, Tkinter , wxPython , Qt 또는 GTK 와 같은 범용 GUI 킷을 사용하여 애플리케이션에 플롯을 포함하기 위한 객체 지향 API 를 제공합니다.

3.2 맵플로우립(Matplotlib) 설정

① 한글 폰트 설정 - 'matplotlib.rcParams['font.family'] = 'Malgun Gothic' '

```
import matplotlib.pyplot as plt
import matplotlib
```

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

② 선 그래프를 그리는 함수 - 'ax.plot'

```
import matplotlib.pyplot as plt
import matplotlib
```

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

모든 그래프 구성요소(축, 그래픽, 텍스트, 레이블 등)를 포함하는 최상위 객체 fig 생성.

(그래프를 그리기 전에 액자와 같은 모양을 **Figure**라고 하겠습니다.)

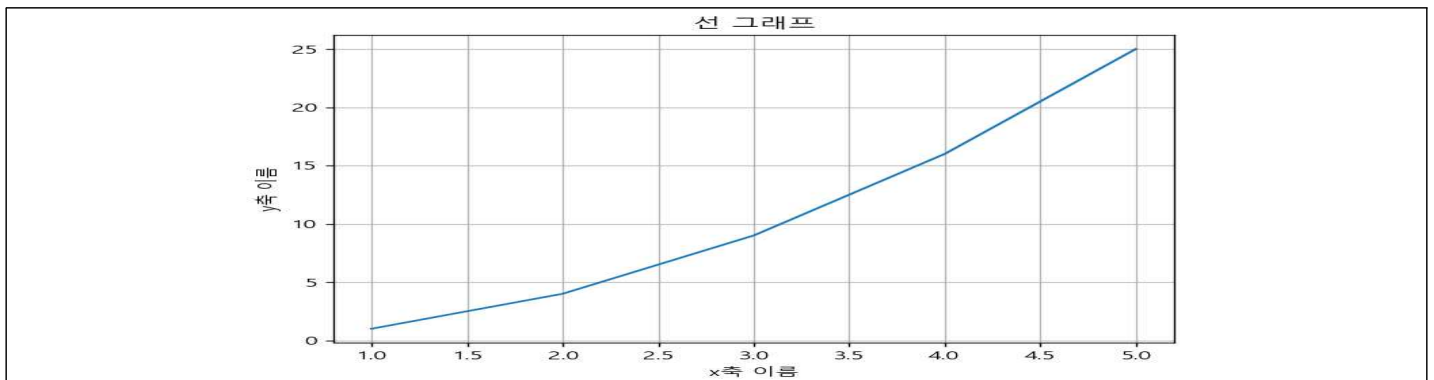
눈금과 레이블이 있는 **테두리박스(그래프)** 생성. **서브플롯**입니다.

```
fig, ax = plt.subplots()
```

```
x = [1, 2, 3, 4, 5]
```

```
y = [1, 4, 9, 16, 25]
```

```
ax.plot(x, y)          # 리스트 x, y를 넣어서 그래프 그리기. 리스트 x는 x축의 값, 리스트 y는 y축의 값.
ax.set_xlabel('x축 이름') # x축 레이블 설정
ax.set_ylabel('y축 이름') # y축 레이블 설정
plt.grid(True)         # 격자(grid) 설정
plt.title('선 그래프')  # 제목 설정
plt.show()             # 그래프를 화면에 나타낸다.
```



③ 산점도를 그리는 함수 - 'ax.scatter'

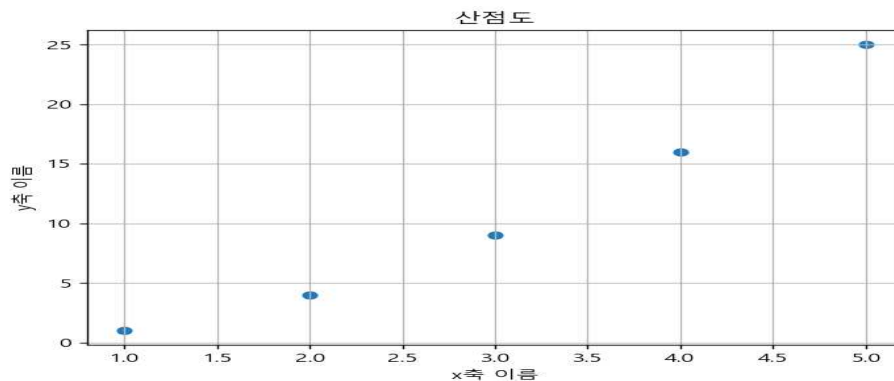
```
import matplotlib.pyplot as plt
import matplotlib
```

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

```
fig, ax = plt.subplots()
```

```
x = [1, 2, 3, 4, 5]
y = [1, 4, 9, 16, 25]
```

```
ax.scatter(x, y)
ax.set_xlabel('x축 이름')
ax.set_ylabel('y축 이름')
plt.grid(True)
plt.title('산점도 그래프')
plt.show()
```



④ 가로 막대 그래프를 그리는 함수 - 'ax.barh'

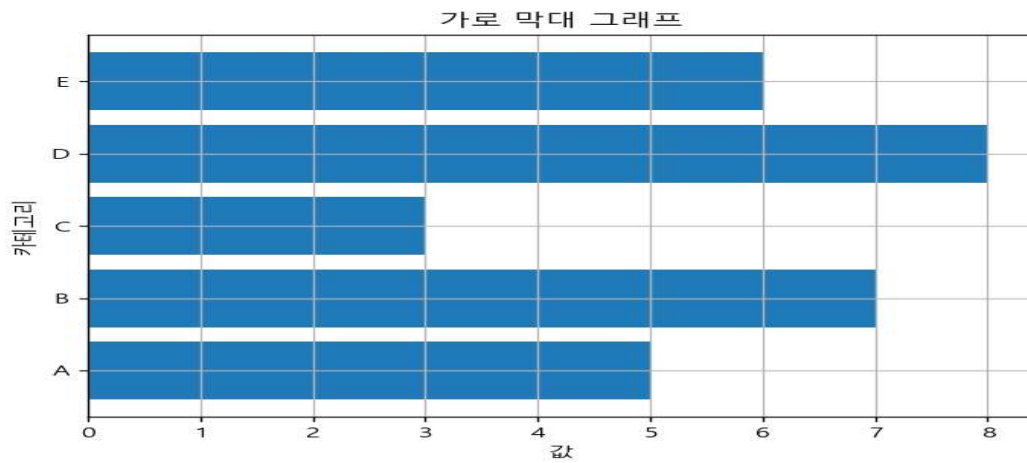
```
import matplotlib.pyplot as plt
import matplotlib
```

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

```
fig, ax = plt.subplots()
```

```
categories = ['A', 'B', 'C', 'D', 'E']
values = [5, 7, 3, 8, 6]
```

```
ax.barh(categories, values) # 첫 번째 인자는 Y축. 두 번째 인자는 X축.
ax.set_xlabel('값')
ax.set_ylabel('카테고리')
plt.grid(True)
plt.title('가로막대 그래프')
plt.show()
```



⑤ 세로 막대 그래프를 그리는 함수 - 'ax.bar'

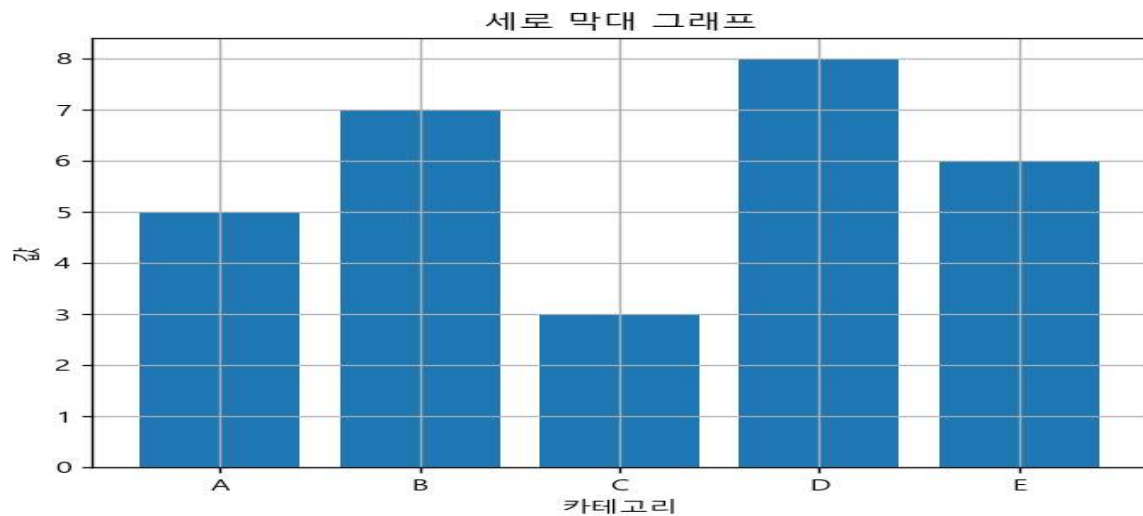
```
import matplotlib.pyplot as plt
import matplotlib
```

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

```
fig, ax = plt.subplots()
```

```
categories = ['A', 'B', 'C', 'D', 'E']
values = [5, 7, 3, 8, 6]
```

```
ax.bar(categories, values) # 첫 번째 인자는 X축. 두 번째 인자는 Y축.
ax.set_xlabel('카테고리')
ax.set_ylabel('값')
plt.grid(True)
plt.title('세로막대 그래프')
plt.show()
```



⑥ 그래프의 제목 설정 - 'plt.title'

```
plt.title('그래프 제목')          # 기본
plt.title('그래프 제목', loc='left') # 제목을 넣을 위치 (left, center, right)
plt.title('그래프 제목', pad=20)    # pad : 제목과 그래프의 간격 (숫자)

# fontdict : 제목의 글꼴. 딕셔너리로 설정
#   fontsize - 제목 크기 (숫자)
#   fontweight - normal, bold, heavy, light
#   color - 제목 색깔
plt.title('그래프 제목', fontdict={'fontsize': 18, 'fontweight':'bold', 'color':'blue'})
```

⑦ 그래프 화면에 보여주기 - 'plt.show'

```
plt.show()
```

⑧ x축 이름 설정 - 'ax.set_xlabel'

```
# labelpad: 레이블 여백 (숫자)
# loc: 레이블 위치.
#   ax.set_xlabel(loc='center') - 'left', 'center', 'right'
#   ax.set_ylabel(loc='top')    - 'bottom', 'center', 'top'
# color: 레이블 색깔.
ax.set_xlabel('x축 이름', labelpad=10, loc='center', color='blue')
```

⑨ 범례 설정 - 'ax.legend'

```
# loc: 위치.
#   best, upper right, upper left, upper center
#   lower right, lower left, lower center
#   center right, center left, center
# fontsize: 글씨 크기 (숫자)
# shadow : 그림자 (True, False)
# fontsize: 글씨 크기 (숫자)
# shadow : 그림자 (True, False)
plt.plot(x, y, label='선 그래프')
ax.legend(loc='upper right', fontsize=18, shadow=True)

# x축, y축 범위는 0~1 값의 범위
# x축 기준으로 0.5 y축 기준으로 0.5에 그린다. 가운데에 그려진다.
ax.legend(loc=(0.5, 0.5))
```

⑩ 격자 설정 - 'plt.grid'

```
# True로 설정하면 x, y 축 둘 다 표시
# plt.grid(True, axis='y') 가로 방향
# plt.grid(True, axis='x') 세로 방향
# 격자(그리드) y는 가로 방향. 색깔 회색, 투명도 0.2, 선 스타일 실선, 선 두께 1
plt.grid(True, axis='y', color='gray', alpha=0.2, linestyle='-', linewidth=1)
```

4. 차트 그리기 실습

① 꺾은 선 차트 그리기 예제

```
import matplotlib.pyplot as plt
import matplotlib
```

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

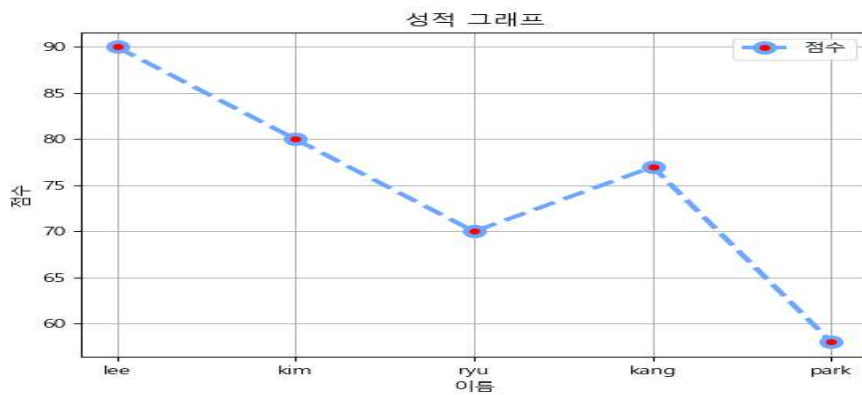
```
# 꺾은 선 그래프 그리기
# =====
# color : 선 색깔
# marker: 마커 스타일
# markersize: 마커 크기
# markeredgecolor: 마커 테두리 색깔
# markerfacecolor: 마커 색깔
# markeredgewidth: 마커 테두리 너비
# linewidth: 선 두께
# linestyle: 선 스타일
# https://matplotlib.org/stable/api/\_as\_gen/matplotlib.pyplot.plot.html
# https://matplotlib.org/stable/api/markers\_api.html
```

```
fig, ax = plt.subplots()
```

```
score=[90, 80, 70, 77, 58]
x=['lee','kim','ryu','kang','park']
```

```
ax.plot(x, score,
        linestyle='--',
        color='#6BA1FF',
        marker='o',
        markersize=8,
        markeredgecolor='#6BA1FF',
        markerfacecolor='red',
        markeredgewidth=3,
        linewidth=3,
        label='점수')
```

```
ax.set_xlabel('이름')
ax.set_ylabel('점수')
ax.legend(loc='upper right')
plt.grid(True)
plt.title('성적 그래프')
plt.show()
```



② 세로막대 차트 그리기 예제

```
import matplotlib.pyplot as plt
import matplotlib
```

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

```
fig, ax = plt.subplots()
```

```
data = [78, 80, 90, 77, 100]
```

```
x = ['김철수', '이영희', '홍길동', '김민수', '김민우']
```

```
# 막대그래프
```

```
# color: 막대 색깔
```

```
# width: 너비. 0.2/0.6/0.8/1.0 디폴트는 0.8입니다.
```

```
# align: center, edge
```

```
# edgecolor: 테두리 색깔
```

```
# linewidth: 테두리 굵기
```

```
colors = ['red', 'green', 'blue', '#20FF1F', '#FF6C6D']
```

```
ax.bar(x, data, color=colors, width=0.6, align='center', edgecolor='lightgray', linewidth=5)
```

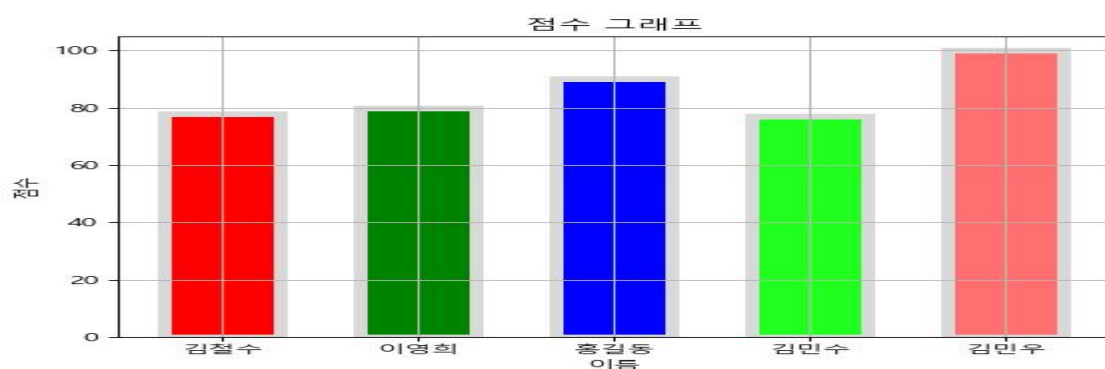
```
ax.set_xlabel('이름')
```

```
ax.set_ylabel('점수')
```

```
plt.grid(True)
```

```
plt.title('점수 그래프')
```

```
plt.show()
```



③ 혼합 차트 그리기 예제

```
import matplotlib.pyplot as plt
import matplotlib
```

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

```
fig, ax = plt.subplots()
```

```
kor = [78, 80, 90, 77, 100]
```

```
eng = [70, 80, 90, 50, 60]
```

```
x=['김철수', '이영희', '홍길동', '김민수', '김민우']
```

```
ax.bar(x, kor, width=0.3, color='red', label='국어')
```

```
ax.bar(x, eng, width=0.3, color='blue', label='영어', bottom=kor)
```

```
ax.set_xlabel('이름')
```

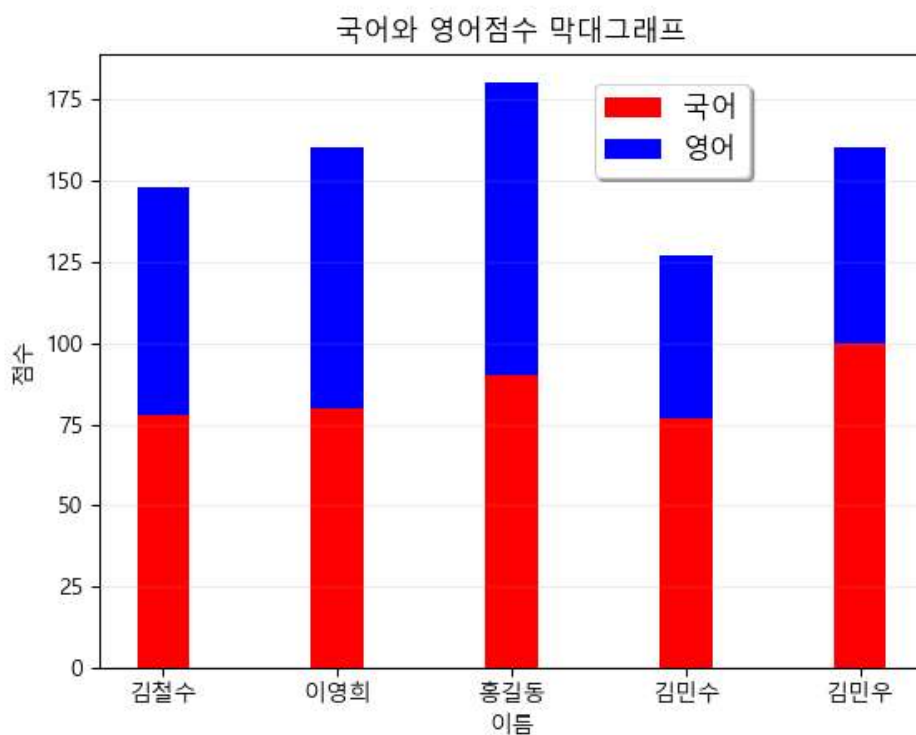
```
ax.set_ylabel('점수')
```

```
ax.legend(loc=(0.6, 0.8), fontsize=12, shadow=True)
```

```
plt.grid(True, axis='y', color='gray', alpha=0.2, linestyle='-', linewidth=1)
```

```
plt.title('국어와 영어점수 막대그래프')
```

```
plt.show()
```



④ 파이 차트 그리기 예제

```
import matplotlib.pyplot as plt
import matplotlib
```

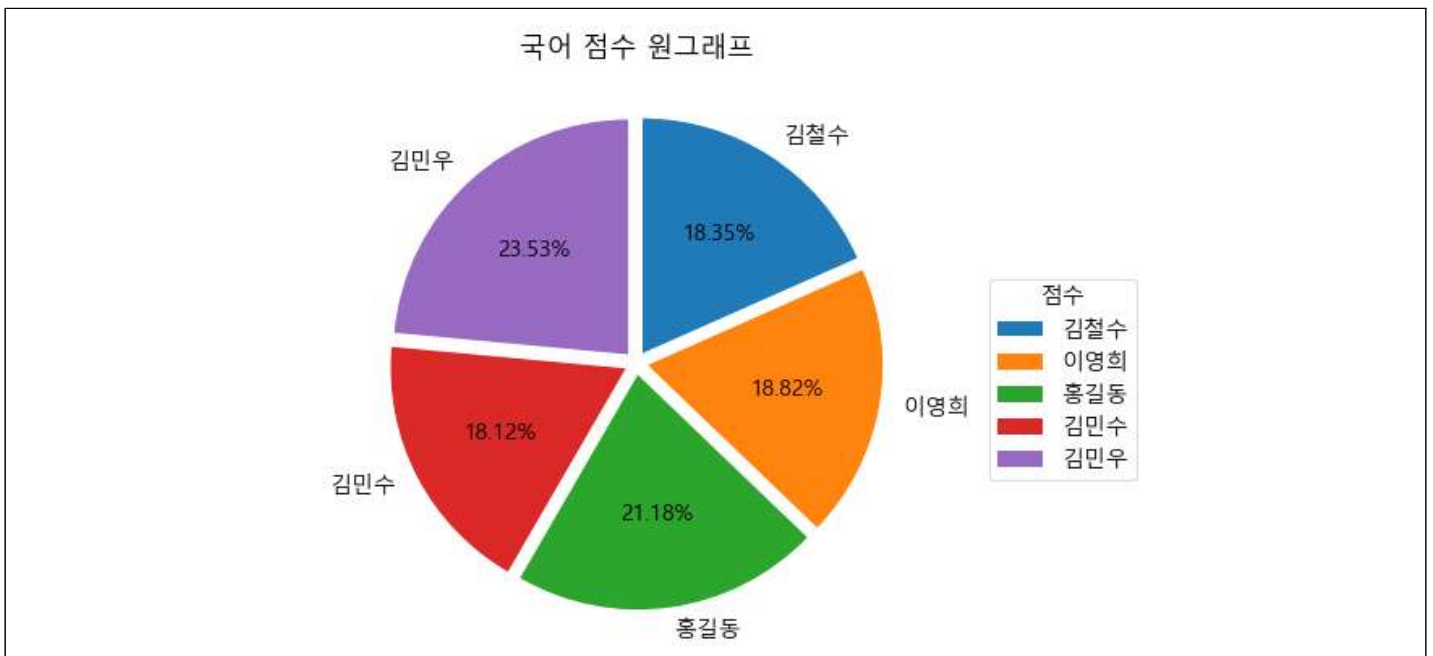
```
matplotlib.rcParams['font.family'] = 'Malgun Gothic'
matplotlib.rcParams['axes.unicode_minus'] = False
```

```
fig, ax = plt.subplots()
```

```
kor = [78, 80, 90, 77, 100]
names = ['김철수', '이영희', '홍길동', '김민수', '김민우']
explode = [0.05, 0.05, 0.05, 0.05, 0.05]
```

```
# startangle=90 시작 위치를 90도로 설정.
# counterclock=False 시계방향으로 그리기.
# explode: 간격 띄우기
ax.pie(kor, labels=names, autopct='%0.2f%%', startangle=90, counterclock=False, explode=explode)
ax.legend(loc=(1.1, 0.3), title='점수')
```

```
plt.grid(True, axis='y', color='gray', alpha=0.2, linestyle='-', linewidth=1)
plt.title('국어 점수 원그래프')
plt.show()
```



⑤ 파이 차트 그리기 예제

```
import matplotlib.pyplot as plt
import matplotlib
```

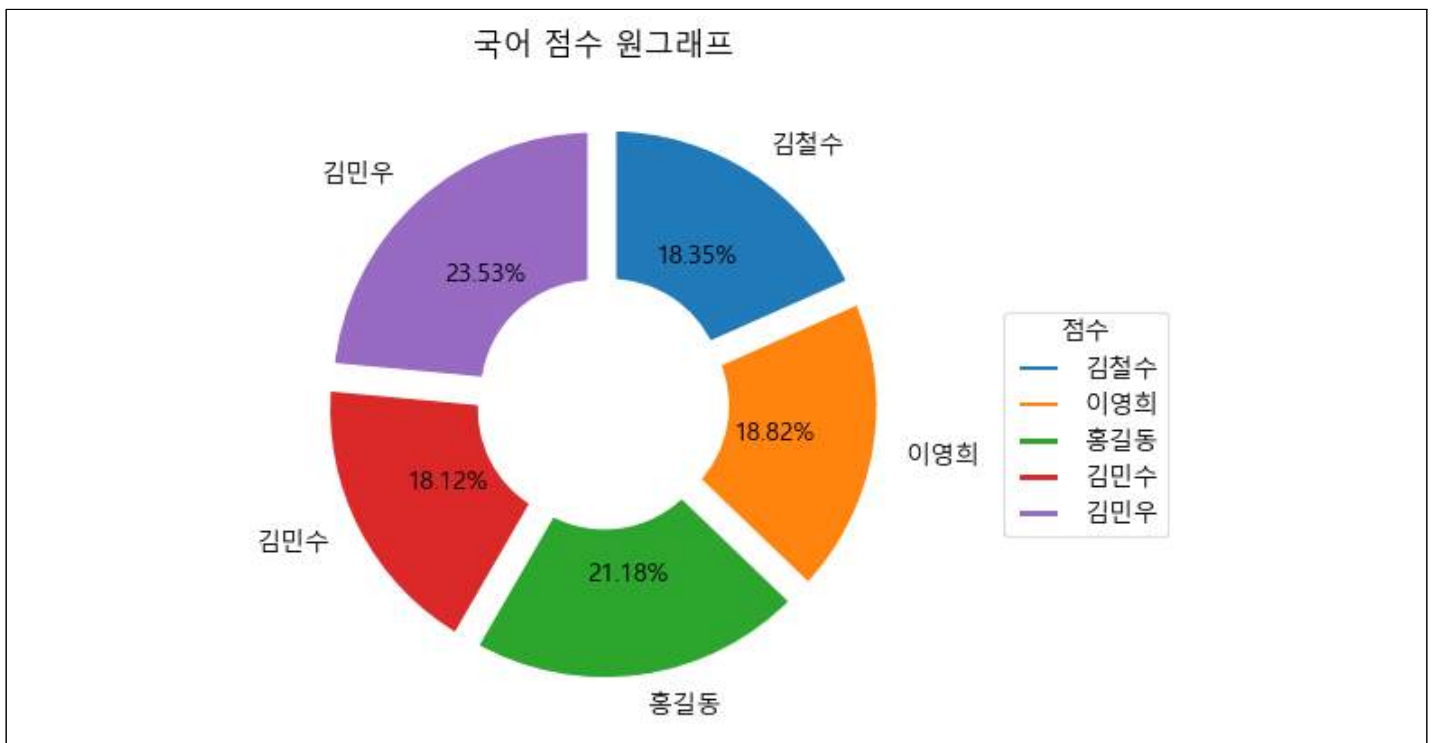
```
matplotlib.rcParams['font.family'] = 'Malgun Gothic'
matplotlib.rcParams['axes.unicode_minus'] = False
```

```
fig, ax = plt.subplots()
```

```
kor = [78, 80, 90, 77, 100]
names = ['김철수', '이영희', '홍길동', '김민수', '김민우']
explode = [0.05, 0.05, 0.05, 0.05, 0.05]
colors = ['#ffadad', '#ffd6a5', '#fdffb6', '#caffbf', '#9bf6ff']
wedgeprops = {'width': 0.6, 'edgecolor': 'w', 'linewidth': 5}
```

```
# 도넛 모양으로 하기위해서 딕셔너리를 만든다
# 테두리 색은 흰색, 두께 5
ax.pie(kor, labels=names, autopct='%0.2f%%', startangle=90,
       counterclock=False, explode=explode, wedgeprops=wedgeprops)
ax.legend(loc=(1.1, 0.3), title='점수')
```

```
plt.grid(True, axis='y', color='gray', alpha=0.2, linestyle='-', linewidth=1)
plt.title('국어 점수 원그래프')
plt.show()
```



5. 판다스

5.1 판다스(Pandas)데이터 분석 처리

다양한 분야에서 데이터 분석 처리에 편리하게 사용할 수 있는 판다스에 대해 알아보겠습니다. 판다스(pandas)는 파이썬의 기본적인 코딩을 익힌 후에 공부하는 것을 추천드리며, 파이썬 IDLE과 PyCharm 어떤 것을 사용해도 무관합니다.

5.2 판다스(Pandas) 시작하기

판다스는 파이썬 설치 시 기본 포함되어 있지 않기 때문에 설치가 필요합니다. 파이썬 IDLE을 사용한다면 cmd 커맨드 창을 열고 아래 명령어를 입력하여 설치합니다.

① 판다스 모듈 설치하기

```
pip install pandas
```



파이썬(PyCharm)에서도 동일하게 터미널에서 위 명령어를 입력해 판다스를 설치합니다. 설치 후, 판다스 모듈을 불러옵니다. 'as' 키워드를 사용하여 별칭을 지정할 수 있습니다.

② 판다스 모듈 불러오기

```
import pandas as pd
data = {
    'Name': ['홍길동', '이순신', '강감찬'],
    'Age': [23, 45, 35]
}
print(data)
```

```
{'Name': ['홍길동', '이순신', '강감찬'], 'Age': [23, 45, 35]}
```

딕셔너리 자료형을 이용해 객체를 생성하고 출력할 수 있습니다. 딕셔너리는 키를 기반으로 값을 저장하며, 각 키에 여러 개의 값을 리스트 형태로 저장하고 1차원 배열구조인 시리즈 형태로 출력합니다.

③ 데이터프레임 생성하기

```
import pandas as pd
data = {
    'Name': ['홍길동', '이순신', '강감찬'],
    'Age': [23, 45, 35]
}
df = pd.DataFrame(data) # 데이터프레임 만들기
print(df)
```

	Name	Age
0	홍길동	23
1	이순신	45
2	강감찬	35

④ 인덱스 추가하기

```
import pandas as pd
data = {
    'Name': ['홍길동', '이순신', '강감찬'],
    'Age': [23, 45, 35]
}
df = pd.DataFrame(data)
print(df)
df.index=['A','B','C'] # 인덱스 추가
print(df)
```

	Name	Age
A	홍길동	23
B	이순신	45
C	강감찬	35

⑤ columns 추가하기

```
import pandas as pd
data = {
    'Name': ['홍길동', '이순신', '강감찬'],
    'Age': [23, 45, 35]
}
df = pd.DataFrame(data)
print(df)
df.index=['A','B','C']
print(df)
df.columns=['이름', '나이'] # 컬럼 추가
print(df)
```

	이름	나이
A	홍길동	23
B	이순신	45
C	강감찬	35

⑥ 원하는 항목 뽑아오기

```
import pandas as pd
data = {
    'Name': ['홍길동', '이순신', '강감찬'],
    'Age': [23, 45, 35]
}
df = pd.DataFrame(data)
print(df)
df.index = ['A', 'B', 'C']
print(df)
df.columns = ['이름', '나이']
print(df)
print(df.loc['B'])
```

이름	이순신
나이	45
Name: B, dtype: object	

5.3 판다스(Pandas) 데이터프레임으로 차트 그리기

① 판다스 데이터프레임 출력하기

```
import pandas as pd
data = {
    '이름': ['김철수', '이영희', '김민수'],
    '국어': [90, 70, 80],
    '영어': [99, 77, 88],
    '수학': [75, 85, 90]
}
df = pd.DataFrame(data)
print(df)
```

	이름	국어	영어	수학
0	김철수	90	99	75
1	이영희	70	77	85
2	김민수	80	88	90

② 판다스 인덱스 추가하기

```
import pandas as pd
data = {
    '이름': ['김철수', '이영희', '김민수'],
    '국어': [90, 70, 80],
    '영어': [99, 77, 88],
    '수학': [75, 85, 90]
}
df = pd.DataFrame(data)
print(df)
df = df.set_index('이름')
print(df)
```

	국어	영어	수학
이름			
김철수	90	99	75
이영희	70	77	85
김민수	80	88	90

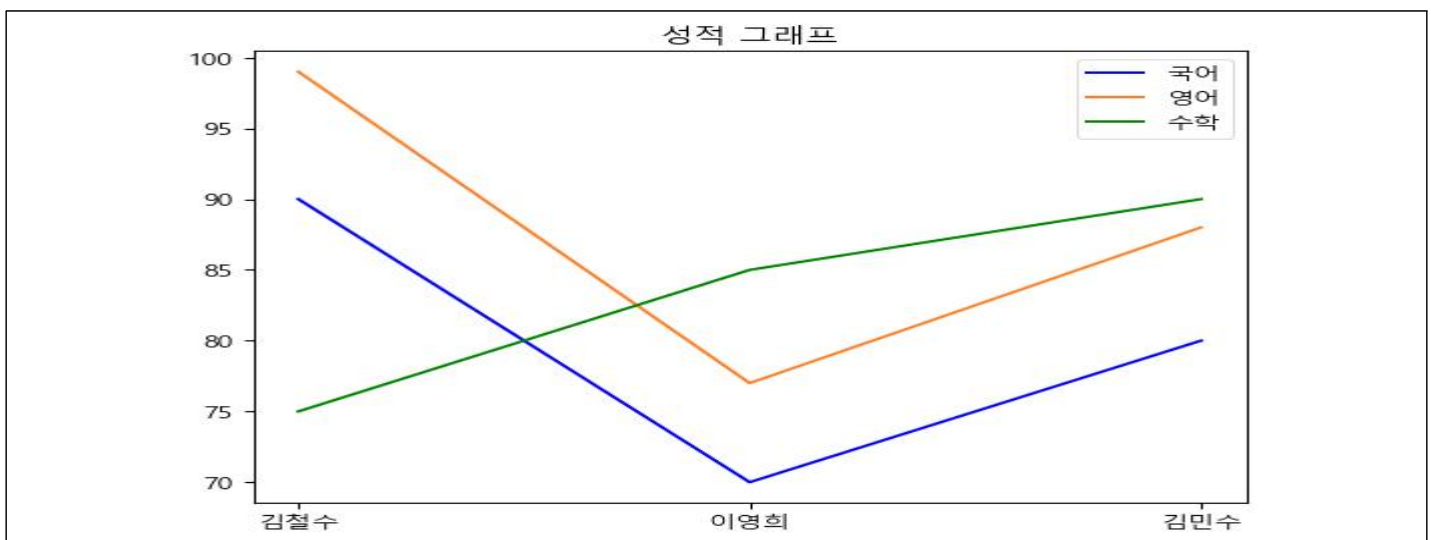
③ 꺾은선 그래프 그리기

```
import matplotlib.pyplot as plt
import matplotlib
import pandas as pd
```

```
matplotlib.rcParams['font.family'] = 'Malgun Gothic'
```

```
data = {
    '이름':['김철수', '이영희', '김민수'],
    '국어':[90, 70, 80],
    '영어':[99, 77, 88],
    '수학':[75, 85, 90]
}
df = pd.DataFrame(data)

fig, ax = plt.subplots()
ax.plot(df['이름'], df['국어'], color='blue', label='국어')
ax.plot(df['이름'], df['영어'], color='#FE7D27', label='영어')
ax.plot(df['이름'], df['수학'], color='green', label='수학')
ax.legend(loc='upper right')
ax.tick_params(rotation=0)
plt.title('성적 그래프')
plt.show()
```



④ 막대 그래프 그리기

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

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

data = {
    '이름':['김철수', '이영희', '김민수'],
    '국어':[90, 70, 80],
    '영어':[99, 77, 88],
    '수학':[75, 85, 90]
}

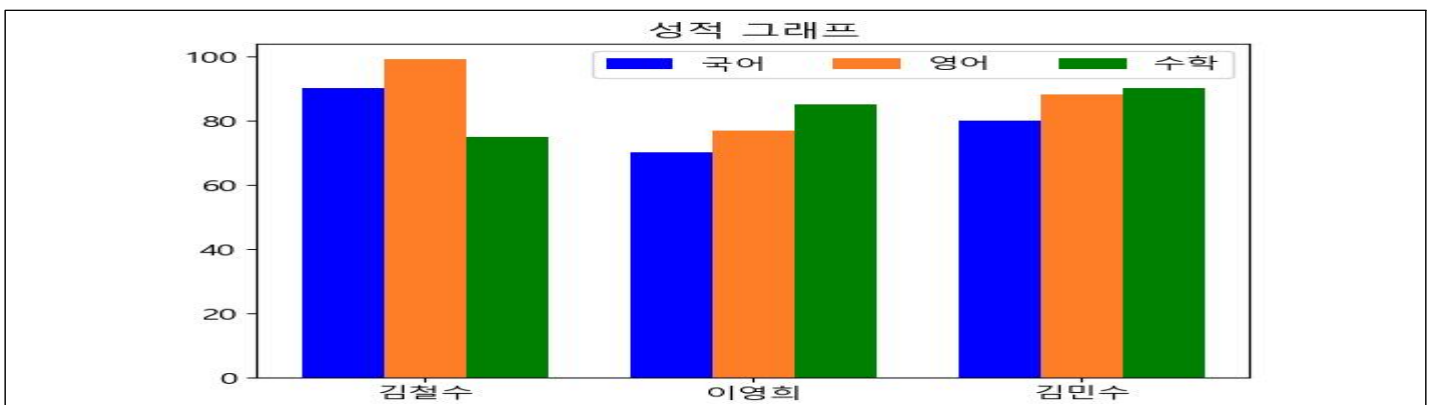
df = pd.DataFrame(data)
# print(df)
# print(df.shape)      # (3, 4)
# print(df.shape[0])   # 3

N = df.shape[0]
index = np.arange(N) # [0 1 2]

fig, ax = plt.subplots()

w = 0.25
ax.bar(df.index - w, df['국어'], color='blue', label='국어', width=0.25)
ax.bar(df.index, df['영어'], color='#FE7D27', label='영어', width=0.25)
ax.bar(df.index + w, df['수학'], color='green', label='수학', width=0.25)
ax.set_xticks(index, df['이름'])
ax.tick_params(rotation=0)
ax.legend(loc='upper right', ncol=3)

plt.title('성적 그래프')
plt.show()
```



⑤ 원 그래프 그리기

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

matplotlib.rcParams['font.family'] = 'Malgun Gothic'

data = {
    '이름':['김철수', '이영희', '김민수'],
    '국어':[90, 70, 80],
    '영어':[99, 77, 88],
    '수학':[75, 85, 90] }

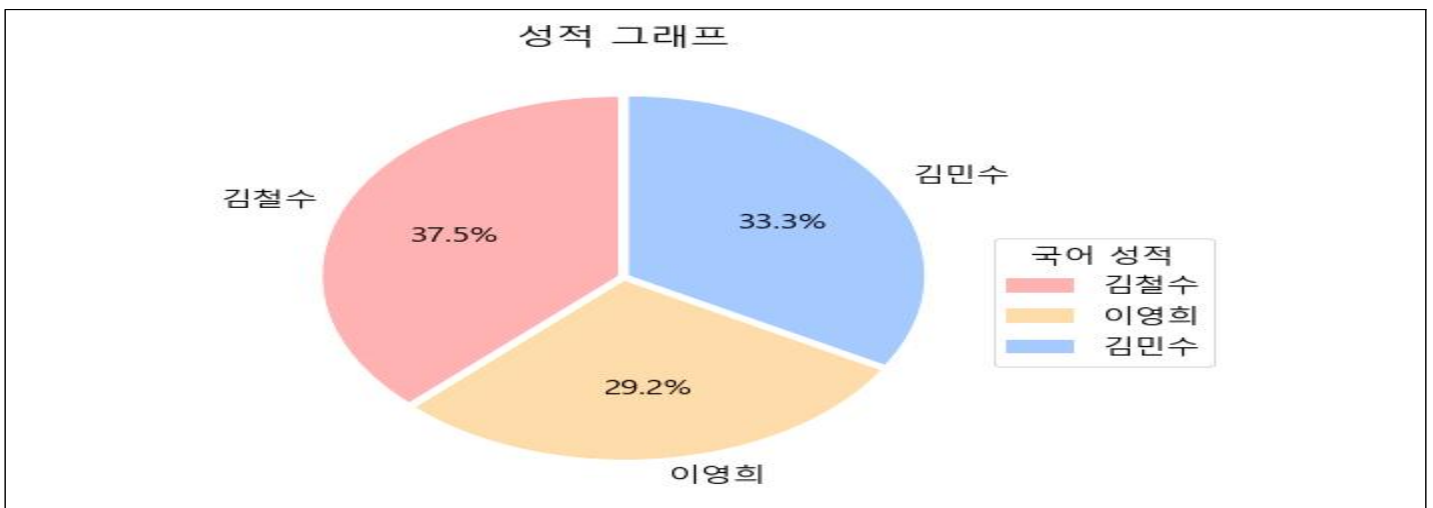
df = pd.DataFrame(data)
df = df.set_index('이름')
# print(df['국어'])
# print(df['국어'].iloc[0])

fig, ax = plt.subplots()

values = [df['국어'].iloc[0], df['국어'].iloc[1], df['국어'].iloc[2]]
labels = [df.index[0], df.index[1], df.index[2]]
explode = [0.02] * 3
colors = ['#ffadad', '#ffd6a5', '#a0c4ff']

ax.pie(values, labels=labels, autopct='%1.1f%%', startangle=90, explode=explode, colors=colors)
# wedgeprops = {'width': 0.6, 'edgecolor':'white', 'linewidth':5}
# ax.pie(values, labels=labels, autopct='%1.1f%%', startangle=90, explode=explode, colors=colors,
# wedgeprops=wedgeprops)

ax.legend(loc=(1.0, 0.3), title='국어 성적')
plt.title('성적 그래프')
plt.show()
```



⑥ 과목별 원 그래프 그리기

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

```
matplotlib.rcParams['font.family'] = 'Malgun Gothic'
```

```
data = {
    '이름':['김철수', '이영희', '김민수'],
    '국어':[90, 70, 80],
    '영어':[99, 77, 88],
    '수학':[75, 85, 90] }

df = pd.DataFrame(data)
df = df.set_index('이름')
```

```
# print(df['국어'])
# print(df['국어'].iloc[0])
```

```
fig, axes = plt.subplots(1, 3, figsize=(18, 6))
```

```
labels = [df.index[0], df.index[1], df.index[2]]
```

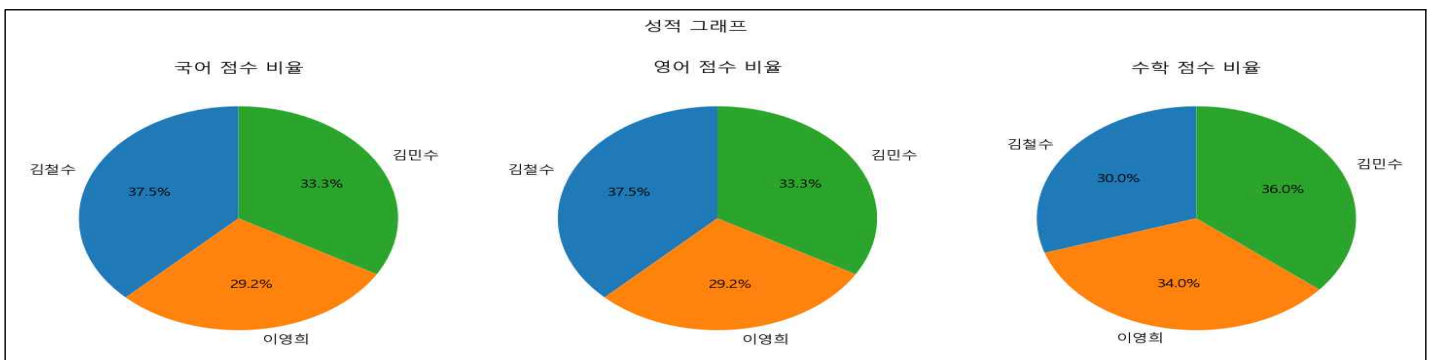
```
values1 = [df['국어'].iloc[0], df['국어'].iloc[1], df['국어'].iloc[2]]
values2 = [df['영어'].iloc[0], df['영어'].iloc[1], df['영어'].iloc[2]]
values3 = [df['수학'].iloc[0], df['수학'].iloc[1], df['수학'].iloc[2]]
```

```
axes[0].pie(values1, labels=labels, autopct='%1.1f%%', startangle=90)
axes[0].set_title('국어 점수 비율')
```

```
axes[1].pie(values2, labels=labels, autopct='%1.1f%%', startangle=90)
axes[1].set_title('영어 점수 비율')
```

```
axes[2].pie(values3, labels=labels, autopct='%1.1f%%', startangle=90)
axes[2].set_title('수학 점수 비율')
```

```
plt.suptitle('성적 그래프')
plt.show()
```



⑦ 과목별 막대그래프 그리기

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

matplotlib.rcParams['font.family'] = 'Malgun Gothic'

data = {
    '이름':['김철수', '이영희', '김민수'],
    '국어':[90, 70, 80],
    '영어':[99, 77, 88],
    '수학':[75, 85, 90]}

df = pd.DataFrame(data)
df = df.set_index('이름')

# print(df['국어'])
# print(df['국어'].iloc[0])

fig, axes = plt.subplots(1, 3, figsize=(18, 6))

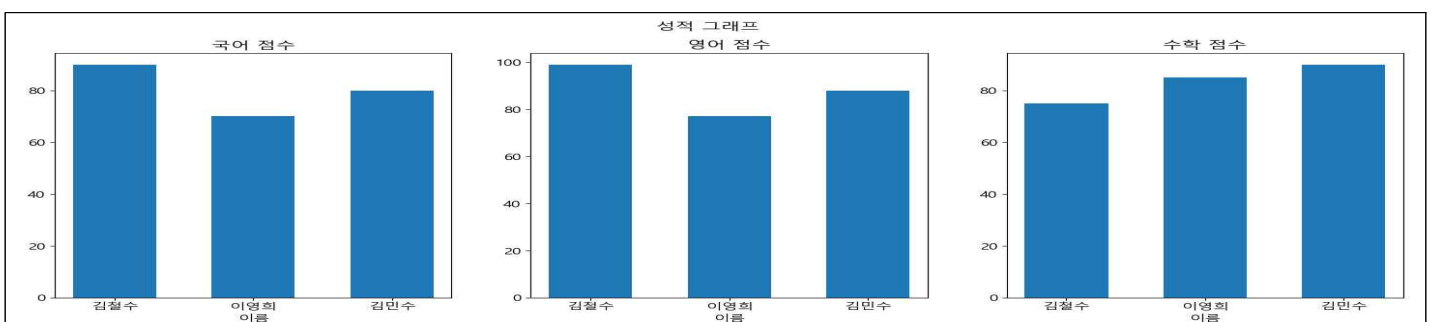
names = [df.index[0], df.index[1], df.index[2]]
values1 = [df['국어'].iloc[0], df['국어'].iloc[1], df['국어'].iloc[2]]
values2 = [df['영어'].iloc[0], df['영어'].iloc[1], df['영어'].iloc[2]]
values3 = [df['수학'].iloc[0], df['수학'].iloc[1], df['수학'].iloc[2]]

axes[0].bar(names, values1, width=0.6)
axes[0].set_xlabel('이름')
axes[0].set_title('국어 점수')

axes[1].bar(names, values2, width=0.6)
axes[1].set_xlabel('이름')
axes[1].set_title('영어 점수')

axes[2].bar(names, values3, width=0.6)
axes[2].set_xlabel('이름')
axes[2].set_title('수학 점수')

plt.suptitle('성적 그래프')
plt.show()
```



⑧ 성적 그래프 그리기 (중간고사 성적) - p01.py

```
import pandas as pd
import numpy as np

# 난수 생성기를 초기화합니다. 777은 시드 값입니다.
# 시드 값을 설정하면, 이후에 생성되는 난수들이 항상 동일한 순서로 생성됩니다.
# 이는 코드 실행 시마다 같은 난수를 얻기 위해 사용됩니다.
np.random.seed(777)
# print(np.random.rand(5)) # 항상 동일한 값이 출력

# 5행 4열의 2차원 배열을 생성하고, 각 요소는 0과 100 사이의 무작위 수(난수)로 채웁니다.
data = np.random.randint(0, 101, size=(5, 4))
# print(data)

col = ['국어', '영어', '수학', '과학']
ind = ['홍길동','김철수','이영희','김민수','김민우']

df = pd.DataFrame(
    data=data,
    columns=col,
    index=ind
)

print(f'*****중간고사 성적*****')
print(df)
```

*****중간고사 성적*****

	국어	영어	수학	과학
홍길동	47	59	38	87
김철수	49	71	29	39
이영희	89	46	24	39
김민수	18	13	14	32
김민우	91	65	82	84

⑨ 성적 그래프 그리기 (기말고사 성적) - p02.py

```
import pandas as pd
import numpy as np

np.random.seed(888)

data = np.random.randint(0, 101, size=(5, 4))
# print(data)

col = ['국어', '영어', '수학', '과학']
ind = ['홍길동','김철수','이영희','김민수','김민우']

df = pd.DataFrame(
    data=data,
    columns=col,
    index=ind
)

print(f'*****기말고사 성적*****')
print(df)
```

	*****기말고사 성적*****				
	국어	영어	수학	과학	
홍길동	26	22	46	60	
김철수	17	16	93	12	
이영희	96	3	80	26	
김민수	19	26	82	46	
김민우	94	52	1	2	

⑩ 종합성적 처리하기 - p03.py

```
import p01, p02

# print(p01.df)
# print(p02.df)

df = (p01.df + p02.df)/2
print(f'*****종합성적*****')
print(df)
```



	국어	영어	수학	과학
홍길동	36.5	40.5	42.0	73.5
김철수	33.0	43.5	61.0	25.5
이영희	92.5	24.5	52.0	32.5
김민수	18.5	19.5	48.0	39.0
김민우	92.5	58.5	41.5	43.0

⑪ 종합성적 막대그래프 그리기 - p04.py

```
import p03
import matplotlib.pyplot as plt
import matplotlib
import pandas as pd
import numpy as np

matplotlib.rcParams['font.family'] = 'Malgun Gothic'

df = p03.df
# print(df)

N = df.shape[0]
index = np.arange(N)    # [0 1 2 3 4]

# print(index)

names = [df.index[0], df.index[1], df.index[2], df.index[3], df.index[4]]
korean_scores = df['국어'].tolist()
english_scores = df['영어'].tolist()
math_scores = df['수학'].tolist()
science_scores = df['과학'].tolist()
```

⑫ 종합성적 막대그래프 그리기 (계속) - p04.py

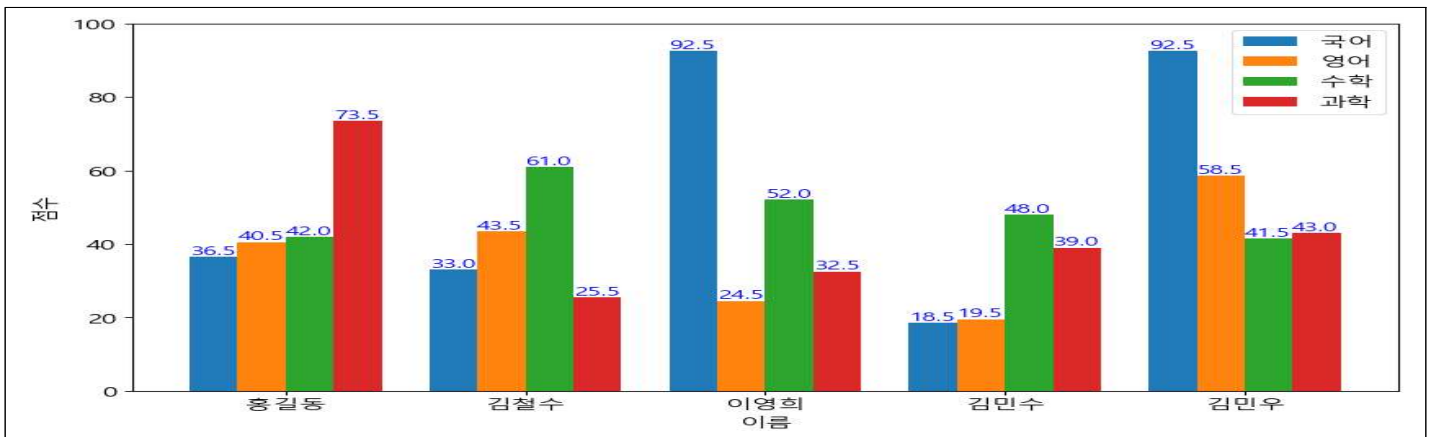
```
fig, ax = plt.subplots(figsize=(10,6))
```

```
ax.bar(index-0.3, korean_scores, width=0.2, label='국어')
ax.bar(index-0.1, english_scores, width=0.2, label='영어')
ax.bar(index+0.1, math_scores, width=0.2, label='수학')
ax.bar(index+0.3, science_scores, width=0.2, label='과학')
```

```
for i in range(0, len(index), 1):
    ax.text(i-0.4, korean_scores[i]+1.5, korean_scores[i], va='center', fontsize=11, color='blue')
    ax.text(i-0.2, english_scores[i]+1.5, english_scores[i], va='center', fontsize=11, color='blue')
    ax.text(i, math_scores[i]+1.5, math_scores[i], va='center', fontsize=11, color='blue')
    ax.text(i+0.2, science_scores[i]+1.5, science_scores[i], va='center', fontsize=11, color='blue')
```

```
ax.set_xlabel('이름')
ax.set_ylabel('점수')
ax.set_xticks(index, names)
ax.set_yticks(np.arange(0, 120, 20))
ax.legend(loc='upper right', ncol=1)
```

```
plt.show()
```



⑬ 종합성적 과목별 막대그래프 그리기

```
import p03
```

```
import matplotlib.pyplot as plt
```

```
import matplotlib
```

```
import pandas as pd
```

```
import numpy as np
```

```
matplotlib.rcParams['font.family'] = 'Malgun Gothic'
```

```
df = p03.df
```

```
N = df.shape[0]
```

```
index = np.arange(N) # [0 1 2 3 4]
```

```
names = [df.index[0], df.index[1], df.index[2], df.index[3], df.index[4]]
```

```
korean_scores = df['국어'].tolist()
```

⑭ 종합성적 과목별 막대그래프 그리기 (계속)

```
english_scores = df['영어'].tolist()
math_scores = df['수학'].tolist()
science_scores = df['과학'].tolist()
```

```
fig, ax = plt.subplots(2, 2, figsize=(12, 8))
```

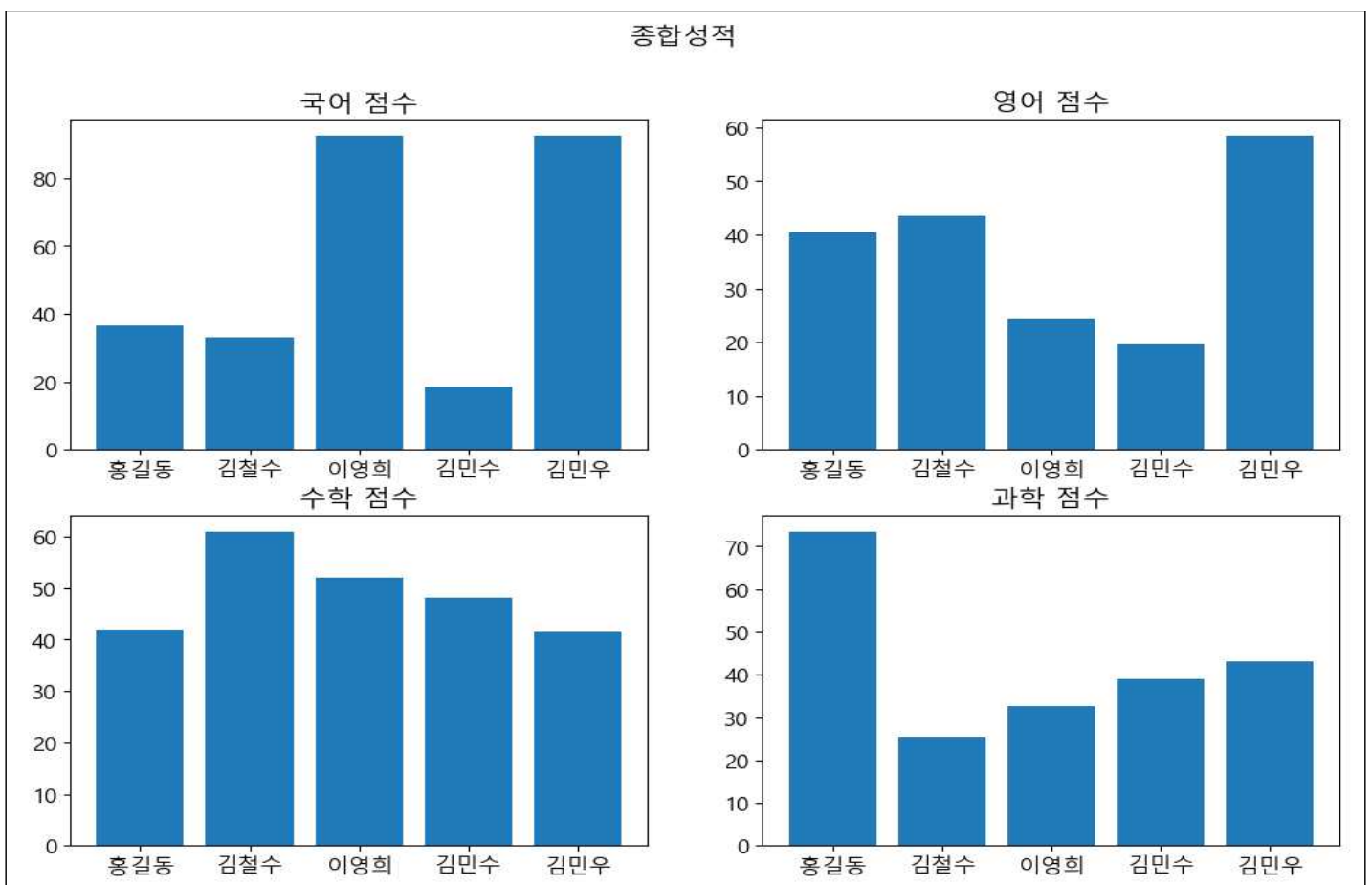
```
ax[0][0].bar(names, korean_scores, label='국어')
# ax[0][0].bar(names, df['국어'], label='국어')
ax[0][0].set_title('국어 점수')
```

```
ax[0][1].bar(names, english_scores, label='영어')
ax[0][1].set_title('영어 점수')
```

```
ax[1][0].bar(names, math_scores, label='수학')
ax[1][0].set_title('수학 점수')
```

```
ax[1][1].bar(names, science_scores, label='과학')
ax[1][1].set_title('과학 점수')
```

```
plt.suptitle('종합성적')
plt.show()
```



⑮ 종합성적 과목별 원그래프 그리기

```
import p03
import matplotlib.pyplot as plt
import matplotlib
import pandas as pd

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

df = p03.df
N = df.shape[0]
index = np.arange(N) # [0 1 2 3 4]
names = df.index.values # ['홍길동' '김철수' '이영희' '김민수' '김민우']
subject_names = df.columns.values # ['국어' '영어' '수학' '과학']
# print('index: ', index)
# print('학생 이름:', names)
# print('과목 이름:', subject_names)
korean_scores = df['국어'].tolist()
english_scores = df['영어'].tolist()
math_scores = df['수학'].tolist()
science_scores = df['과학'].tolist()

scores_list = [korean_scores, english_scores, math_scores, science_scores]
# print(scores_list)
'''
이것도 가능
scores_list = []
for sub_name in subject_names:
    scores_list.append(df[sub_name].tolist())
'''

fig, ax = plt.subplots(2, 2, figsize=(12, 8))

n = 0

for i in range(0, 2, 1): # 0 1
    for j in range(0, 2, 1): # 0 1
        ax[i][j].pie(scores_list[n], labels=names, autopct='%1.1f%%', startangle=90)
        ax[i][j].set_title(subject_names[n]) # 0 0, 0 1, 1 0, 1 1
        n=n+1

plt.legend(loc=(1.0, 0.8))
plt.suptitle('종합성적')
plt.show()
```

종합성적



6. 연도별 출산율 데이터 분석

6.1 활용데이터 찾기

- ① 활용데이터 찾기 : 국가통계포털 - 통계놀이터 - "출산율" 검색 - 지역별(시도)합계 출산율을 클릭합니다.

KOSIS 통계놀이터

비주얼 통계 | 데이터로 말하기 | 통계 클래스룸 | 만화로 보는 통계 | 게시판

통합검색

통합검색

"출산율"에 대한 검색 결과는 4건입니다.

비주얼 통계 : 4건

지역별(시도) 합계 출산율

국가별 합계출산율

우리나라 합계 출산율

우리나라 출생아 수와 합계 출산율의 변화

- ② 관련통계표 - KOSIS 통계목록 - 시군구/출생아수,합계출생아수를 클릭합니다.

비주얼 통계 | KOSIS 통계놀이터

kosis.kr/edu/visualStats/detail.do

무역·국제 정보통신 복지

0.79명 0.70명 0.81명 0.78명 0.80명 0.66명 0.71명 0.97명 0.83명

법률 자료없음 ~ 0.74 0.75 ~ 0.99 1.00 ~ 1.24 1.25 ~ 1.49 1.50 ~

이전시점 2000 2023 이후시점

출처 : 「인구동향조사」 통계청

관련통계표

시군구/출생아수, 합계출산율

다운로드

통계표

- ③ 전체선택 [출생아 수, 합계출산율], 시군구별[구미시], 시점[2011 ~ 2023]

비주얼 통계 | KOSIS 통계놀이터

kosis.kr/edu/visualStats/detail.do

무역·국제 정보통신 복지

0.79명 0.70명 0.81명 0.78명 0.80명 0.66명 0.71명 0.97명 0.83명

법률 자료없음 ~ 0.74 0.75 ~ 0.99 1.00 ~ 1.24 1.25 ~ 1.49 1.50 ~

이전시점 2000 2023 이후시점

출처 : 「인구동향조사」 통계청

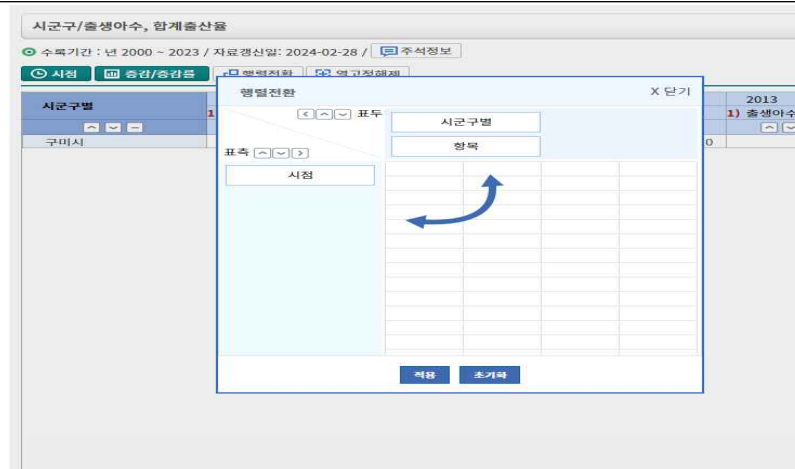
관련통계표

시군구/출생아수, 합계출산율

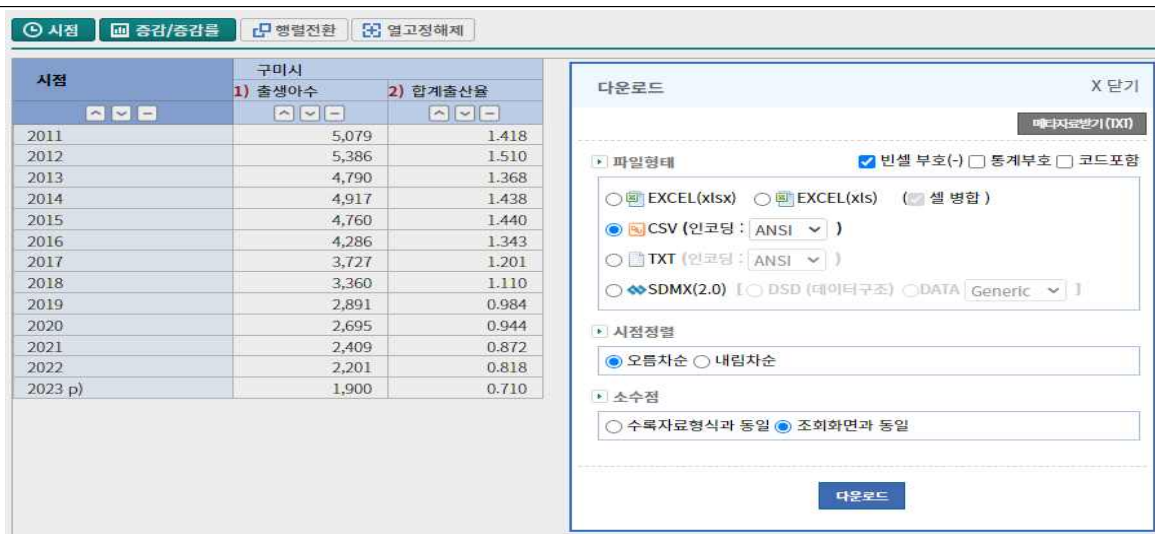
다운로드

통계표

④ 행렬전환해서 적용 합니다.



⑤ 다운로드 - CSV(인코딩 : ANSI - 다운로드 클릭합니다.



⑥ 다운로드한 활용데이터를 파이썬 디렉토리에 넣고 파이썬 파일을 만들어 코딩합니다.

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

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

```
df = pd.read_csv('시군구_출생아수__합계출산율_20240627002952.csv', encoding='cp949')
# print(df)
```

```
year = df['시점'][1:]
df.iloc[13, 0] = df.iloc[13, 0][:4]
# print(df)
```

⑦ 계속

```
new_df = pd.DataFrame(
    {
        '년도': df['시점'][1:].astype(int),
        '출생아수': df['구미시'][1:].astype(int),
        '합계출산율': df['구미시.1'][1:].astype(float),
    }
)

print(new_df)

fig, ax = plt.subplots(figsize=(10, 6))

ax.bar(new_df['년도'], new_df['출생아수'], color='#E8C288', label='출생아 수')

for i in range(0, len(new_df.index), 1):
    ax.text(
        new_df.iloc[i]['년도'], int(new_df.iloc[i]['출생아수']+95), int(new_df.iloc[i]['출생아수']),
        fontsize=10,                # 글씨 크기
        color='royalblue',          # 글씨 색깔
        fontweight='bold',          # 굵게
        backgroundcolor='lightgray', # 밝은 회색 배경 색
        alpha=0.5,                  # 0.5 투명도
        va='center',                # 세로축 정렬: top, bottom, center
        ha='center',                # 가로축 정렬: left, right, center
        rotation='horizontal',      # 회전: vertical, horizontal
        bbox=dict(boxstyle='round', facecolor='white', alpha=0.4, edgecolor="blue", pad=0.1)
    )

ax.legend(loc=(0.81, 0.88), fontsize=12, shadow=True)
ax.set_xlabel('년도', labelpad=10, loc='center', color='blue')
ax.set_ylabel('출생아 수(명)', labelpad=20, loc='top', color='blue')

# x축을 공유하는 axis를 만든다
ax2 = ax.twinx()

ax2.plot(new_df['년도'], new_df['합계출산율'],
        color='#6BA1FF',
        marker='o',
        markersize=8,
        markeredgecolor='#6BA1FF',
        markerfacecolor='white',
        markeredgewidth='3',
        linewidth=3,
        label='합계출산율')
```

⑧ 계속

```
for i in range(0, len(new_df.index), 1):
```

```
    ax2.text(new_df.iloc[i]['년도'], new_df.iloc[i]['합계출산율']-0.04, new_df.iloc[i]['합계출산율'],
             fontsize=10,
             color='green',
             fontweight='bold',
             va='center',
             ha='center'
            )
```

```
ax2.set_ylabel('합계출산율', labelpad=10, loc='center', color='#6BA1FF')
```

```
ax2.legend(loc=(0.81, 0.8), fontsize=12, shadow=True)
```

```
ax2.tick_params(axis='y', direction='inout', length=5, pad=10, labelsize=12, labelcolor='blue', width=2,
               color='red')
```

```
plt.grid(True, axis='y', color='gray', alpha=0.2, linestyle='-', linewidth=1)
```

```
plt.title('우리고장 출생아 수 및 합계출산율', loc='center', pad=20, fontdict={'fontsize': 18, 'fontweight':'bold',
'color':'blue'})
```

```
plt.show()
```

