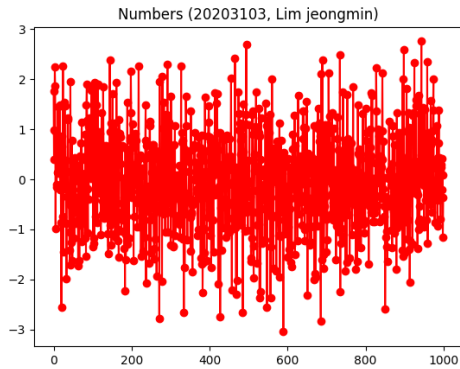


[11장 과제]

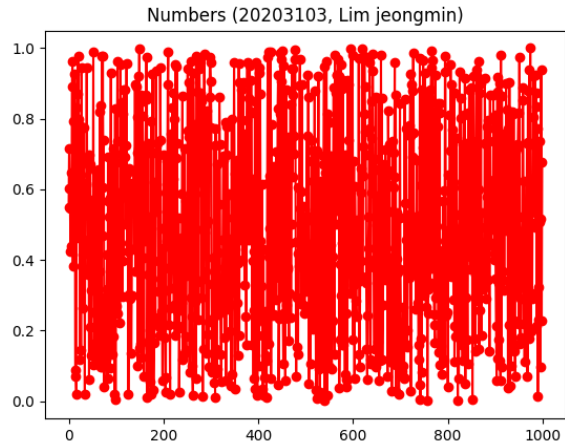
맑은 고딕 12 ☞ 학번 : 20203103 이름 : 임정민

실습 11-1 (p. 286 도전문제 11-1) 넘파이의 난수 생성을 이용하여 1,000개의 난수를 생성하고, 생성된 순서대로 화면에 [결과 화면]과 같이 그리는 프로그램을 작성하시오.

[결과 화면]



[randn 함수]



[rand 함수]

[프로그램 소스]

```
print('\n11-1. 20203103 임정민 \n')
import matplotlib.pyplot as plt
import numpy as np

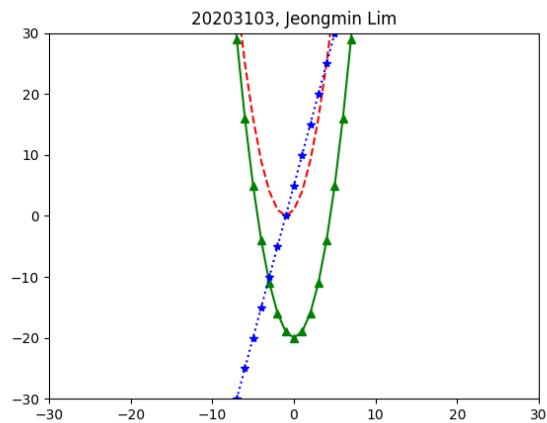
np.random.seed(0)
x = [x for x in range(1000)]
y = np.random.rand(1000)
plt.title("Numbers (20203103, Lim jeongmin)")
plt.plot(x, y, 'r-', marker='o')

plt.show()
```

실습 11-2. (p.287 본문 참조) 다음에 주어진 3개의 함수를 다음의 [결과 화면]과 같이 출력하시오.

$y_1 = x^2 + 2x + 1$, $y_2 = x^2 - 20$, $y = 5x + 5$

[결과 화면]

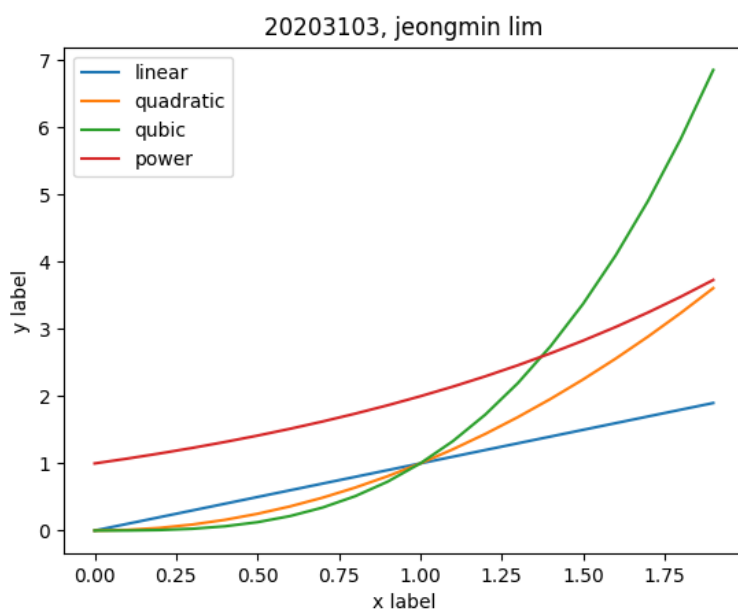


[프로그램 소스]

```
print('\n11-2. 20203103, 임정민 \n')
import matplotlib.pyplot as plt
plt.title('20203103, Jeongmin Lim')
x = [x for x in range(-20, 20)]
y1 = [t**2+2*t+1 for t in x]
y2 = [t**2 -20 for t in x]
y3 = [5*t+5 for t in x]
plt.plot(x, y1, 'r--', x, y2, 'g^--', x, y3, 'b*:')
plt.axis([-30, 30, -30, 30])
plt.show()
```

실습 11-3. (p.288 도전문제 11.2) 도전문제 11.2의 내용을 수행하여, [결과 화면]과 같은 내용을 출력하시오.

[결과 화면]



[프로그램 소스]

```
#ex11-3.py
```

```
print('11-3 20203103 임정민')
import matplotlib.pyplot as plt
x = [x / 10 for x in range(20)]
y = [(x / 10)**2 for x in range(20)]
z = [(x / 10)**3 for x in range(20)]
i = [2**(x / 10) for x in range(20)]

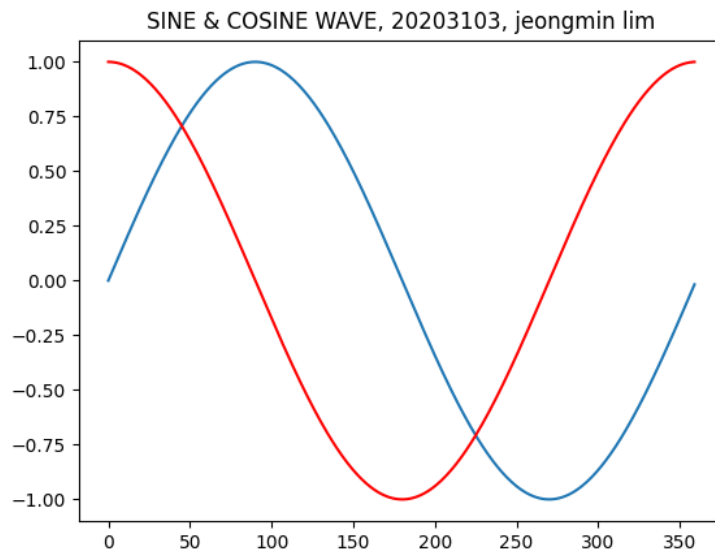
plt.title('20203103, jeongmin lim')
plt.plot(x, x, label='linear')
plt.plot(x, y, label='quadratic')
plt.plot(x, z, label='qubic')
plt.plot(x, i, label='power')

plt.xlabel('x label')
plt.ylabel('y label')

plt.legend()
plt.show()
```

실습 11-4. (p.289 도전문제 11.3) 도전문제 11.3의 내용을 수행하여, [결과 화면]과 같은 내용을 출력하시오.

[결과 화면]



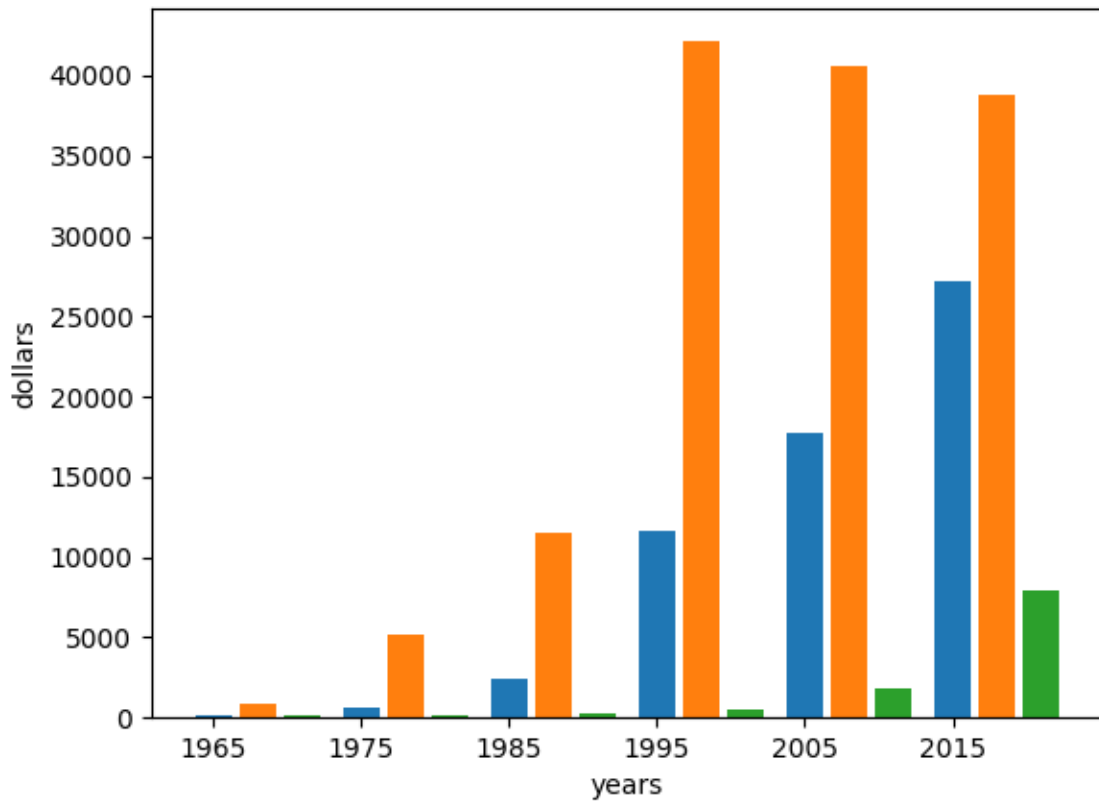
[프로그램 소스]

```
#ex11-4.py
print('11-4 20203103 임정민')
import math
import matplotlib.pyplot as plt
x = []
y = []
z = []
for angle in range(360):
    x.append(angle)
    z.append(math.cos(math.radians(angle)))
    y.append(math.sin(math.radians(angle)))
plt.plot(x, y)
plt.plot(x, z, 'r-')
plt.title("SINE & COSINE WAVE, 20203103, jeongmin lim")
plt.show()
```

실습 11-5. (p.291 본문 내용 참조) p.291의 본문을 참조하여 [결과 화면]과 같은 내용을 출력하시오.

[결과 화면]

GDP per capita, 20203103,jeongmin lim



[프로그램 소스]

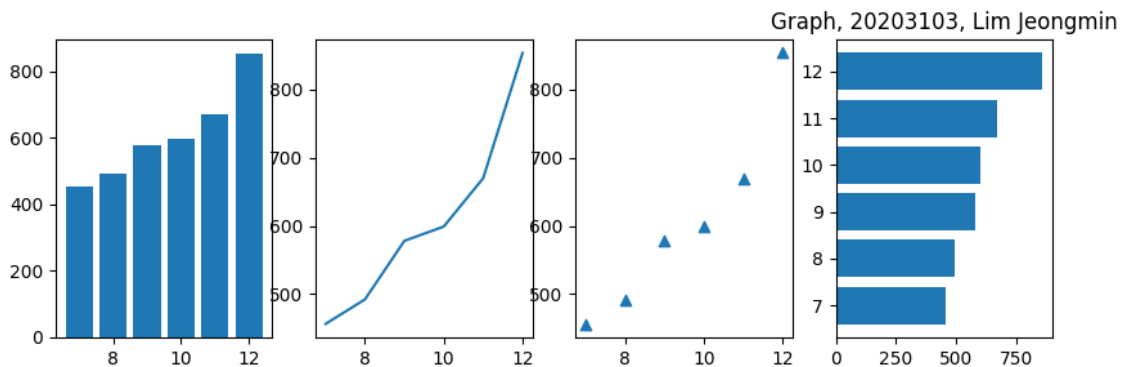
```
#ex11-5.py
print('11-5 20203103 임정민')
import numpy as np
from matplotlib import pyplot as plt

years=[1965, 1975, 1985, 1995, 2005, 2015]
ko=[130, 650, 2450, 11600, 17790, 27250]
jp=[890, 5120, 11500, 42130, 40560, 38780]
ch=[100, 200, 290, 540, 1760, 7940]

x_range=np.arange(len(years))
plt.title("GDP per capita, 20203103,jeongmin lim")
plt.ylabel("dollars")
plt.xlabel("years")
plt.bar(x_range + 0.0,ko, width=0.25)
plt.bar(x_range + 0.3,jp, width=0.25)
plt.bar(x_range +0.6,ch, width=0.25)
plt.xticks(range(len(years)), years)
plt.show()
```

실습 11-6. (p.302 심화문제 11.1 (2)) subplot() 함수를 이용하여 [결과 화면]과 같이 막대 그래프, 직선 그래프, 산포도 그래프, 수평막대 그래프로 표현하여라.

[결과 화면]



[프로그램 소스]

```
#ex11-6.py
import matplotlib.pyplot as plt
fig, ax = plt.subplots(1, 4, figsize=(10, 3))
plt.title("Graph, 20203103, Lim Jeongmin")
x = [x for x in range(7, 13)]
y = [456, 492, 578, 599, 670, 854]

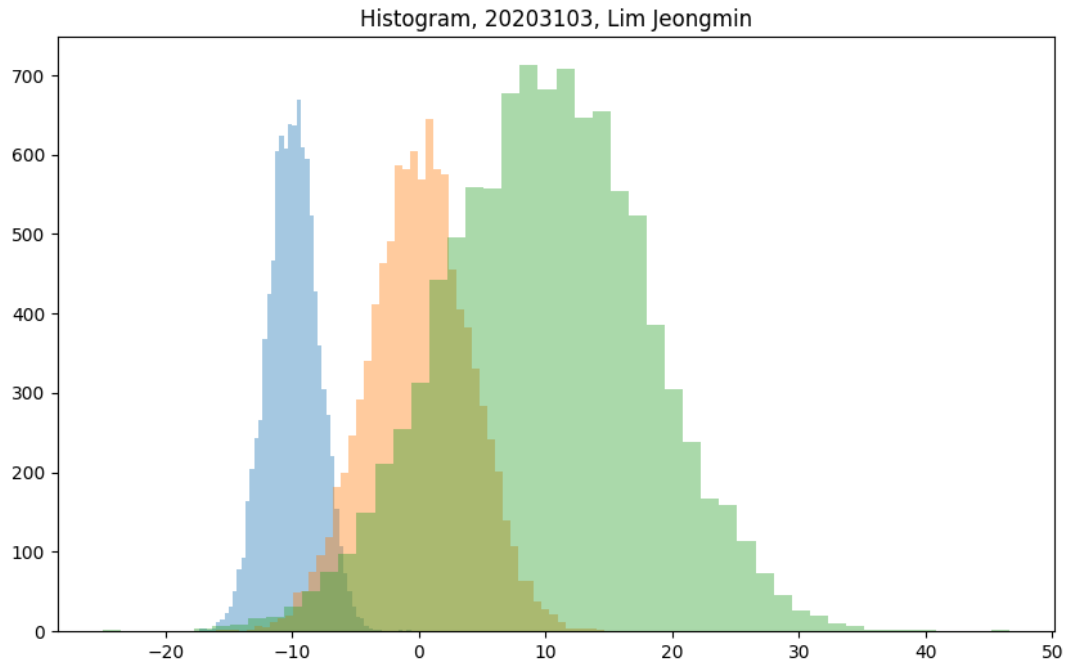
ax[0].bar(x, y)
ax[1].plot(x, y)
ax[2].scatter(x, y, marker='^')
ax[3].barh(x, y)
```

```
plt.show()
```

실습 11-7. (p.296 LAB11-3 응용) 아래 (a), (b), (c)에 대해 정규 분포로 난수를 생성하고, 이 수들이 어떤 방식으로 분산되어 있는지 확인해 보자.

(a) 평균 -10, 표준편차 2 (b) 평균 0, 표준편차 4 (c) 평균 10, 표준편차 8

[결과 화면]



[프로그램 소스]

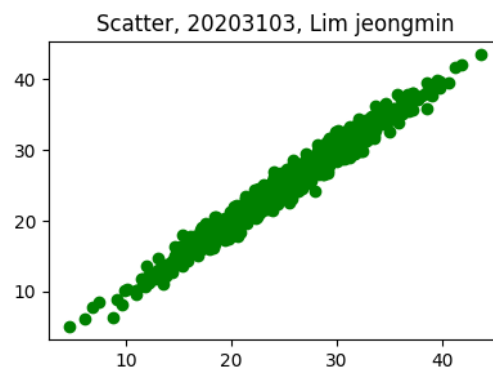
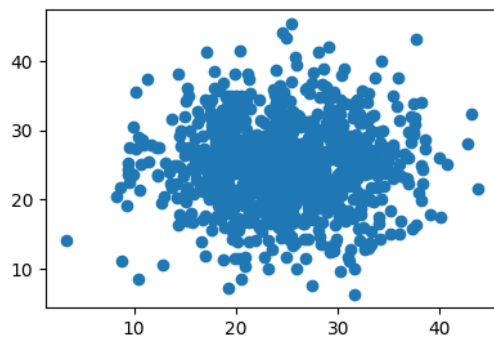
```
#ex11-7.py
import numpy as np
import matplotlib.pyplot as plt
mu1, sigma1 = -10, 2
mu2, sigma2 = 0, 4
mu3, sigma3 = 10, 8
Gauss1 = mu1 + sigma1 * np.random.randn(10000)
Gauss2 = mu2 + sigma2 * np.random.randn(10000)
Gauss3 = mu3 + sigma3 * np.random.randn(10000)
plt.figure(figsize=(10,6))
plt.hist(Gauss1, bins=50, alpha=0.4)
plt.hist(Gauss2, bins=50, alpha=0.4)
plt.hist(Gauss3, bins=50, alpha=0.4)
plt.title("Histogram, 20203103, Lim Jeongmin")
plt.show()
```

실습 11-8. (p.303 심화 11.4) numpy의 난수 생성기를 이용하여 각각 1000개의 난수를 가지는 2가지 종류의 (x, y) 분포를 생성하고 matplotlib의 산포도 그래프를 나타내시오.

(a) x, y 값의 평균이 25, 표준편차 6

(b) x 값의 평균 25, 표준편차 6, y 값은 x값의 표준 정규분포 난수를 더한 값

[결과 화면]



[프로그램 소스]

```
#ex11-8.py
import matplotlib.pyplot as plt
import numpy as np

fig, ax = plt.subplots(1, 2, figsize=(10, 3))

xData1 = 6 * np.random.randn(1000) + 25
yData1 = 6 * np.random.randn(1000) + 25

xData2 = 6 * np.random.randn(1000) + 25
yData2 = []
for value in xData2:
    yData2.append(value + np.random.randn())

ax[0].scatter(xData1, yData1)
ax[1].scatter(xData2, yData2, c='green')
plt.title("Scatter, 20203103, Lim jeongmin")

plt.show()
```