

Лабораторна Робота 11 – Візуалізація даних в Matplotlib.

Мета роботи – вивчити та засвоїти навички роботи з базовим функціоналом модуля Matplotlib. Ознайомитися з принципами та особливостями візуалізації даних шляхом побудови графіків та засвоїти практичні навички їх використання.

1.1 Введення в Matplotlib

Matplotlib - бібліотека на мові програмування Python для візуалізації даних двовимірної (2D) графікою (3D графіка також підтримується). Одержувані зображення можуть бути використані в якості ілюстрацій в публікаціях.

Matplotlib написаний і підтримувався в основному Джоном Хантером (англ. John Hunter) і поширюється на умовах BSD-подібної ліцензії. Генеруються в різних форматах зображення можуть бути використані в інтерактивній графіці, в наукових публікаціях, графічному інтерфейсі користувача, веб-додатках, де потрібна побудова діаграм. Для установки використовуйте пакетний Matplotlib менеджер:

```
conda install matplotlib
або
pip install matplotlib
```

Імпорт модуля:

```
[5]: import matplotlib.pyplot as plt
```

Для відображення графіків в notebook введіть:

```
[6]: %matplotlib inline
```

Завдяки цьому всі графіки будуть виводитися в векторному форматі svg:

```
[7]: %config InlineBackend.figure_format = 'svg'
```

Для відображення графіків у відриві від jupyter notebook використовуйте `plt.show ()` після всіх команд для виведення графіків.

1.2 API matplotlib в функціональному стилі

Matplotlib може будувати графіки зі звичайних списків або масивів numpy.

Створимо масив numpy:

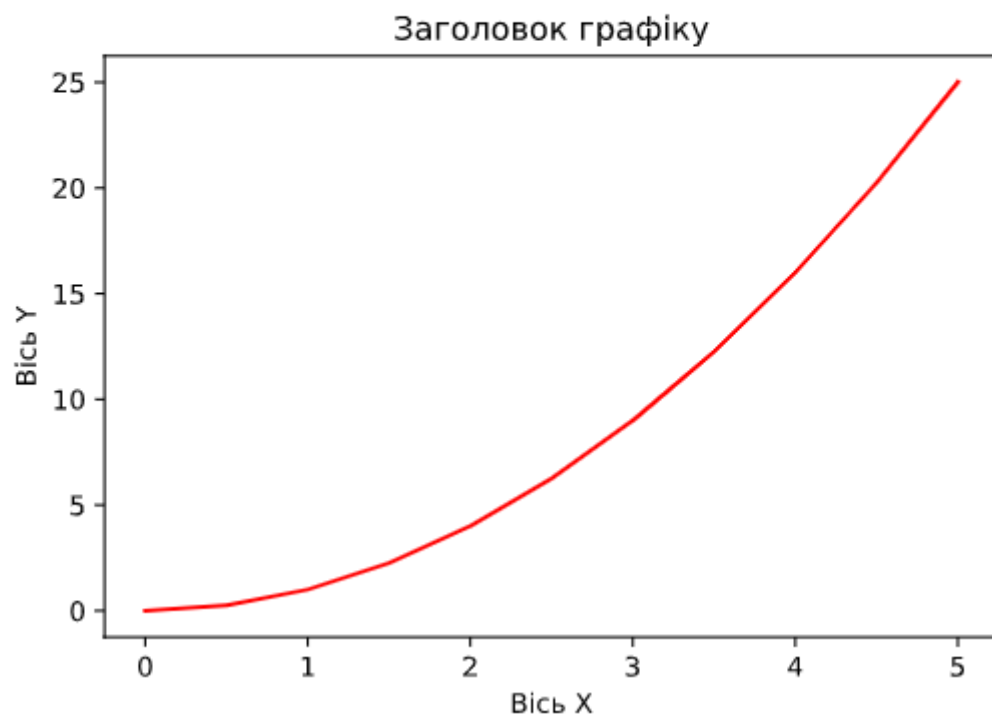
```
[10]: import numpy as np
x = np.linspace(0, 5, 11)
y = x**2
print(x,y,sep="\n\n")

[0.  0.5 1.  1.5 2.  2.5 3.  3.5 4.  4.5 5. ]

[ 0.    0.25  1.    2.25  4.    6.25  9.    12.25 16.    20.25 25.   ]
```

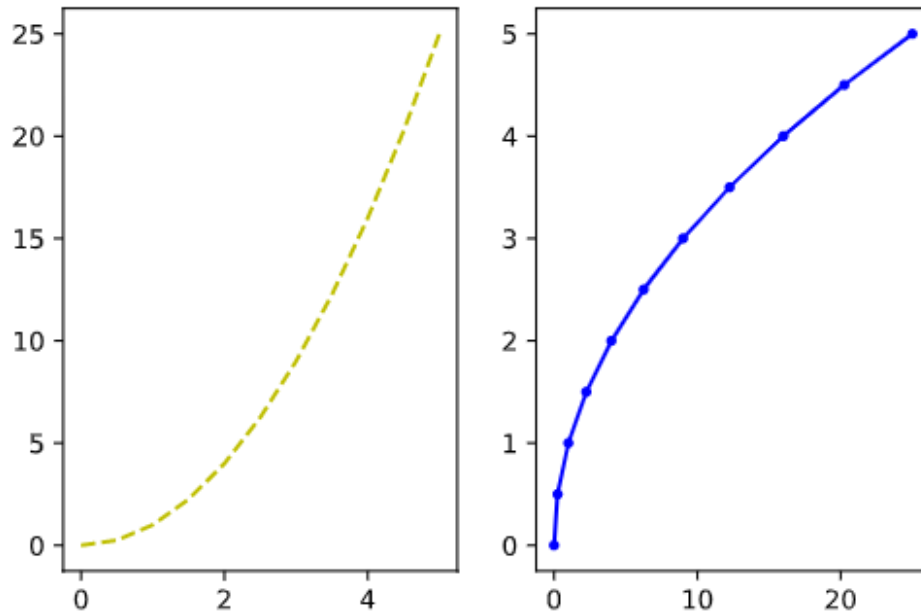
Базові команди matplotlib:

```
[11]: plt.plot(x, y, 'r') # r - red
plt.xlabel('Вісь X')
plt.ylabel('Вісь Y')
plt.title('Заголовок графіку')
plt.show();
```



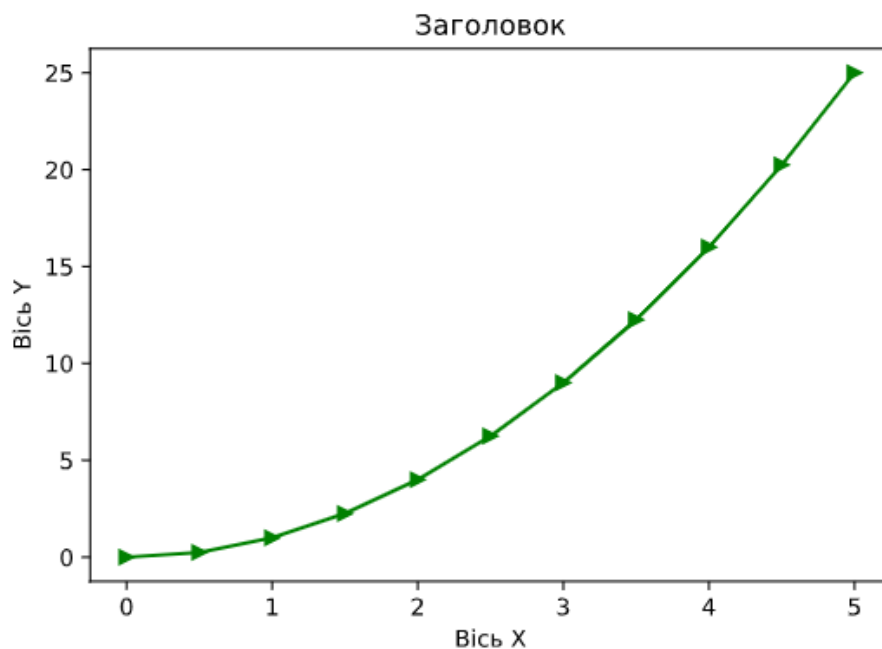
Кілька графіків на одному полотні:

```
# plt.subplot(nrows, ncols, plot_number)
plt.subplot(1, 2, 1)
plt.plot(x, y, 'y--') # "--" - вид лінії
plt.subplot(1, 2, 2)
plt.plot(y, x, 'b.-');# "-." - вид лінії
```



Об'єктно-орієнтований стиль створення графіків:

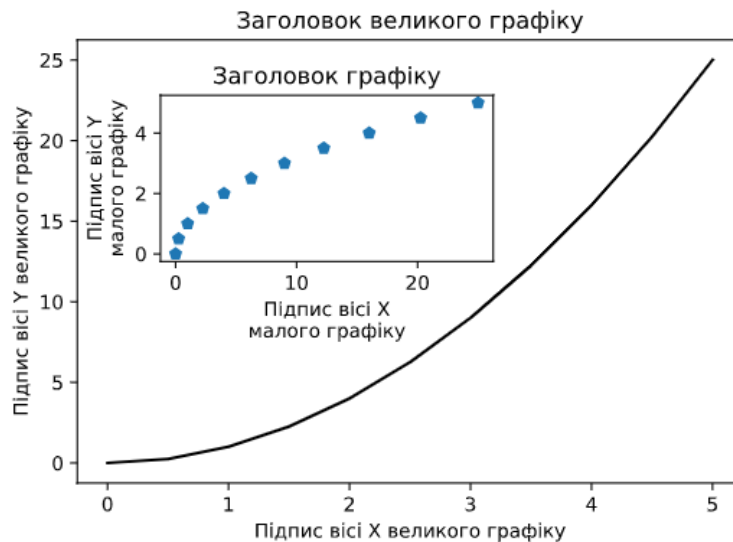
```
[18]: # Create Figure (empty canvas)
fig = plt.figure()
# Add set of axes to figure
axes = fig.add_axes([0.1, 0.1, 0.8, 0.8]) # left, bottom, width, height (range 0 to 1)
# Plot on that set of axes
axes.plot(x, y, 'g->')
axes.set_xlabel('Вісь X') # Notice the use of set_ to begin methods
axes.set_ylabel('Вісь Y')
axes.set_title('Заголовок');
```



Тепер ми маємо повний контроль над вісями графіка:

```
[39]: # Створимо пусте полотно
fig = plt.figure()
# Вісі задаються через список из 4-х чисел: координати лівого нижнього угла, ширину та высоту.
axes1 = fig.add_axes([0.1, 0.1, 0.8, 0.8])
axes2 = fig.add_axes([0.2, 0.5, 0.4, 0.3])
axes1.plot(x, y, 'k')
axes1.set_xlabel('Підпис вісі X великого графіку')
axes1.set_ylabel('Підпис вісі Y великого графіку')
axes1.set_title('Заголовок великого графіку')

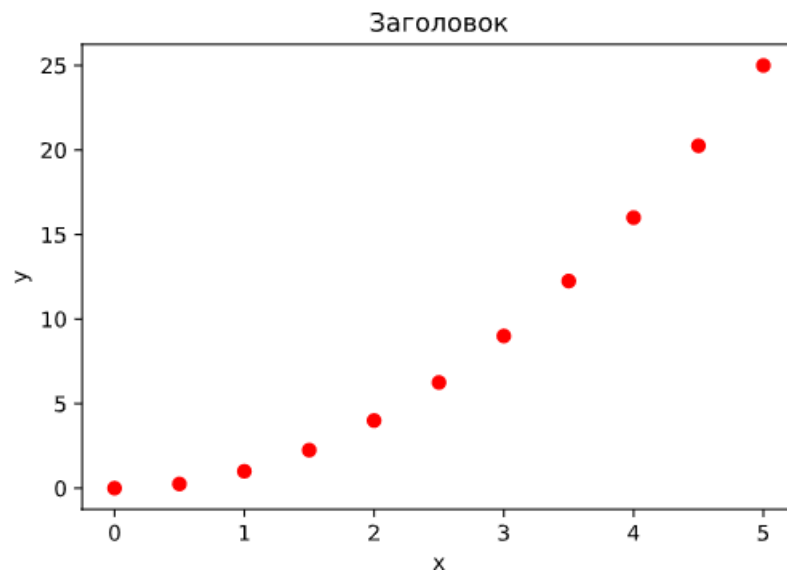
axes2.plot(y, x, 'p')
axes2.set_xlabel('Підпис вісі X\нмалого графіку')
axes2.set_ylabel('Підпис вісі Y\нмалого графіку')
axes2.set_title('Заголовок графіку');
```



subplots() - те ж, що і plt.figure(), тільки повертає відразу кортеж з фігурою:

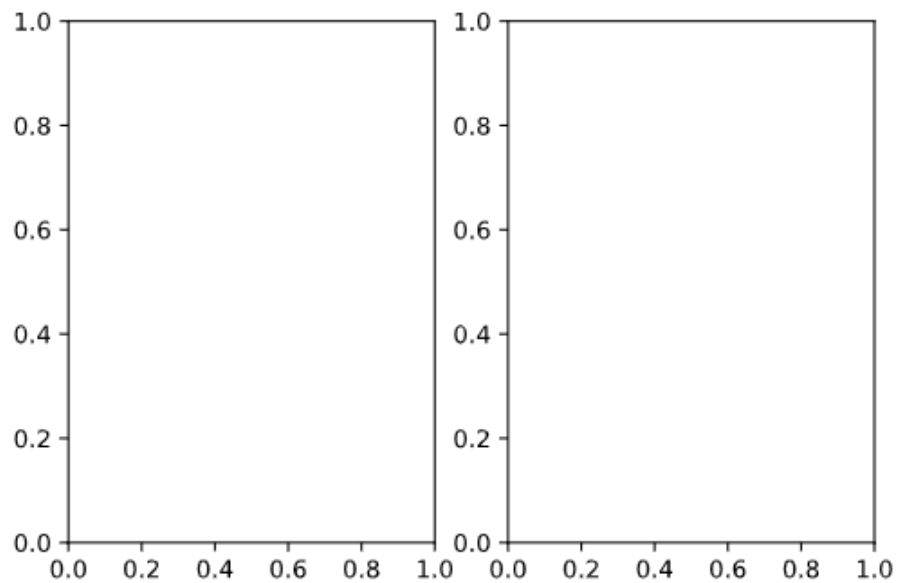
```
[41]: fig, axes = plt.subplots()
axes.plot(x, y, 'ro')
axes.set_xlabel('x')
axes.set_ylabel('y')
axes.set_title('Заголовок')

[41]: Text(0.5, 1.0, 'Заголовок')
```



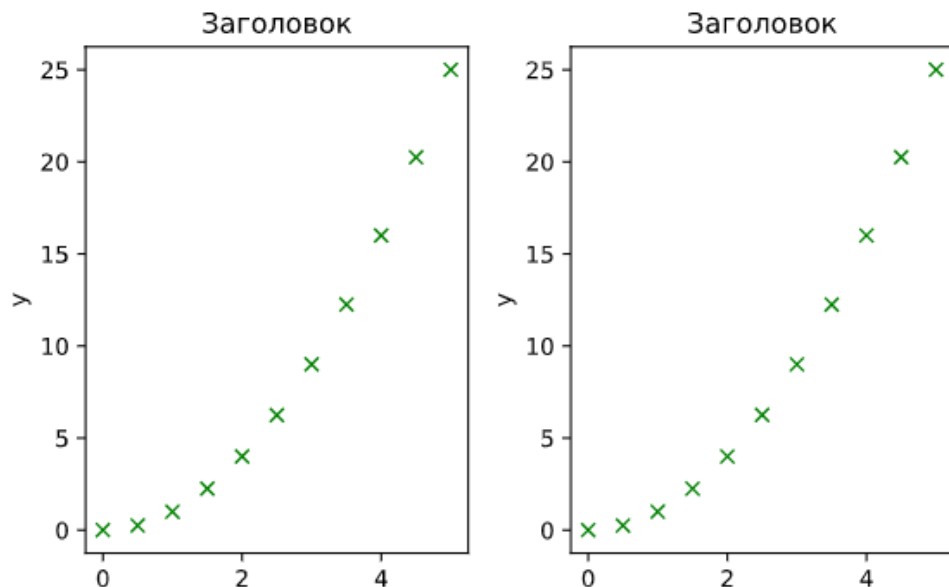
Якщо вказати кількість рядків і стовпців, можна створити кілька графіків.

```
[42]: # два порожніх графіка  
fig, axes = plt.subplots(nrows=1, ncols=2)
```



Axes - це просто список осей « axes array ([,], dtype = object) » При побудові кількох графіків осі часто накладаються одна на одну. Для усунення цієї проблеми використовуйте методи `fig.tight_layout()` або `plt.tight_layout()`:

```
[43]: fig, axes = plt.subplots(nrows=1, ncols=2)  
for ax in axes:  
    ax.plot(x, y, 'gx')  
    ax.set_xlabel('x')  
    ax.set_ylabel('y')  
    ax.set_title('Заголовок')  
fig  
plt.tight_layout()
```



Matplotlib дозволяє задавати розмір сторін графіка і DPI при створенні фігури. Для цього використовуйте наступні іменовані аргументи:

- **figsize** приймає кортеж і двох чисел - ширини і висоти графіка в дюймах.
- **dpi** - dots-per-inch.

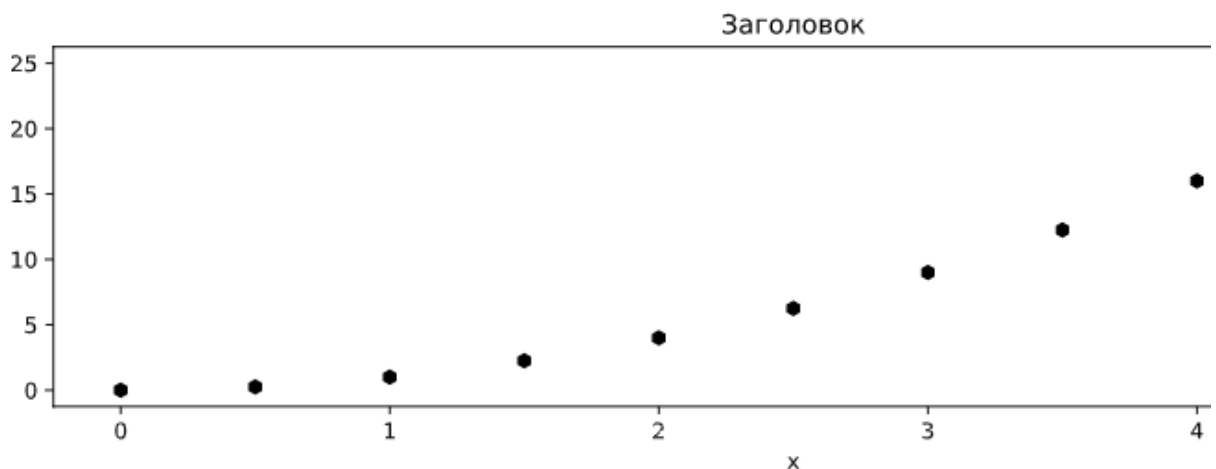
наприклад:

```
[44]: fig = plt.figure(figsize=(8,4), dpi=100)
      <Figure size 800x400 with 0 Axes>
```

Те ж саме можна передати в subplots.

```
[46]: fig, axes = plt.subplots(figsize=(12,3), dpi=300)
      axes.plot(x, y, 'kh')
      axes.set_xlabel('x')
      axes.set_ylabel('y')
      axes.set_title('Заголовок')
```

```
[46]: Text(0.5, 1.0, 'Заголовок')
```



Можна зберігати графіки в форматах PNG, JPG, EPS, SVG, PGF і PDF. Для збереження графіка треба використовувати метод `savefig` класу `Figure`:

```
[ ]: fig.savefig("filename.png", dpi=200)
```

1.3 Легенди, підписи осей, заголовки

Заголовки графіків:

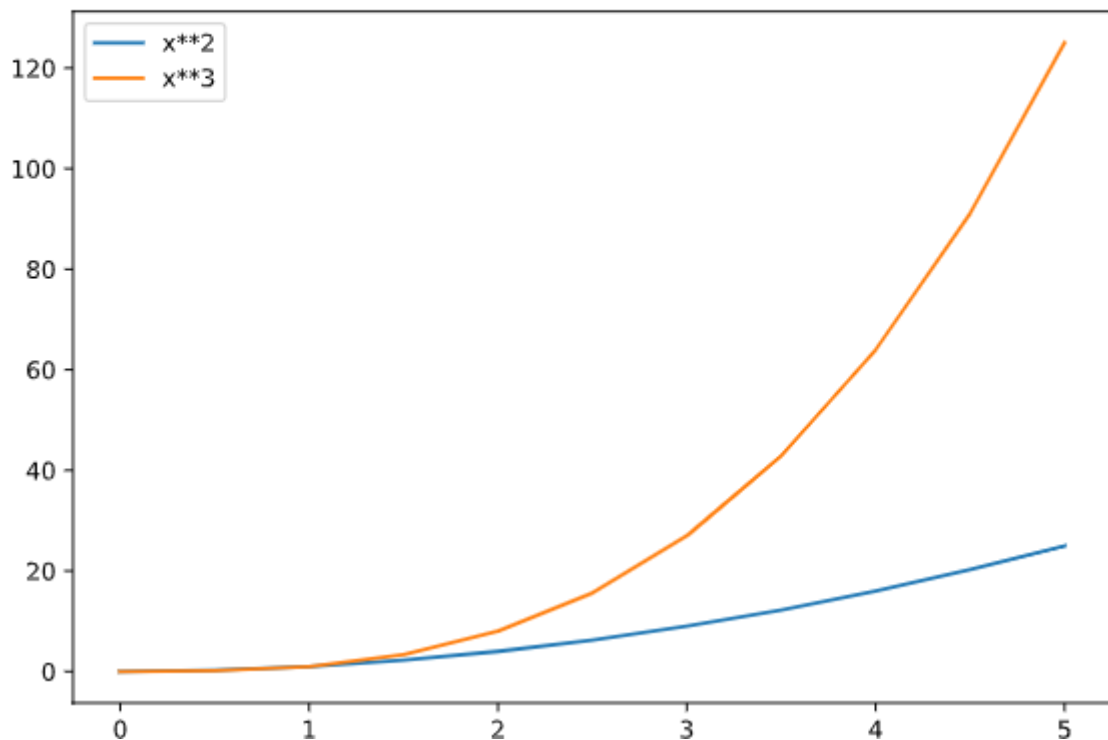
```
[ ]: ax.set_title("Заголовок");
```

Підписи осей:

```
[48]: ax.set_xlabel("x")
      ax.set_ylabel("y");
```

Легенди. При побудові графіків можна вказувати аргумент `label = "label text"`, і зазначені в ньому підписи будуть відображатися як підписи легенд і використанні методу `legend`:

```
fig = plt.figure()
ax = fig.add_axes([0,0,1,1])
ax.plot(x, x**2, label="x**2")
ax.plot(x, x**3, label="x**3")
ax.legend();
```



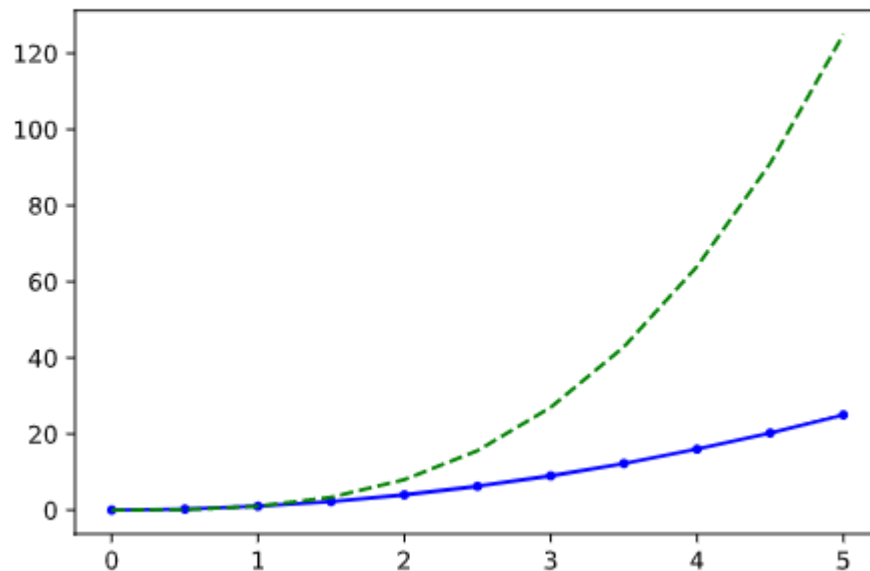
Легенду можна розташовувати в різних місцях графіків:

```
[ ]: ax.legend(loc=1) # upper right
      ax.legend(loc=2) # upper left
      ax.legend(loc=3) # lower left
      ax.legend(loc=4) # lower right
      ax.legend(loc=0) # оптимальне розташування
```

1.4 Установка кольору, товщини і типу ліній

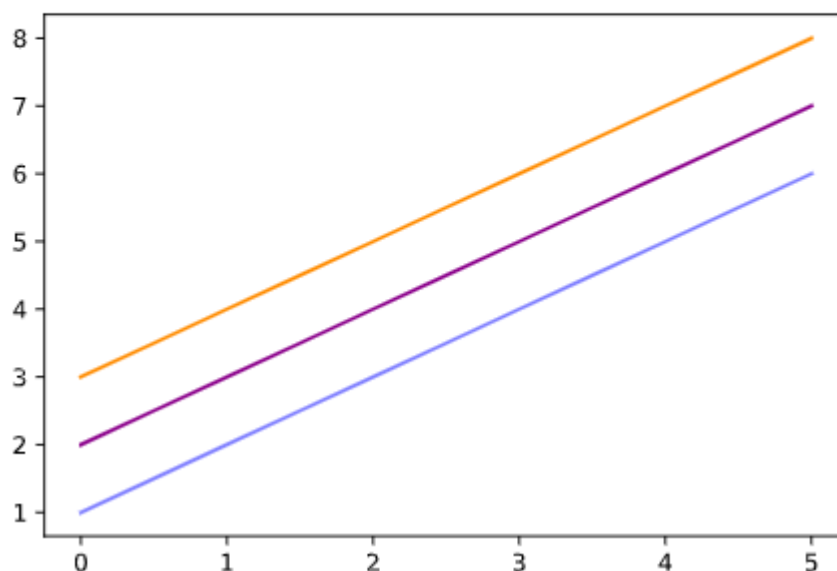
Стиль MatLab:

```
[53]: fig, ax = plt.subplots()
      ax.plot(x, x**2, 'b.-') # Синя лінія, точка пунктир
      ax.plot(x, x**3, 'g--'); # Зелена пунктирна лінія
```



Зазначення кольорів в параметрі color = і alpha:

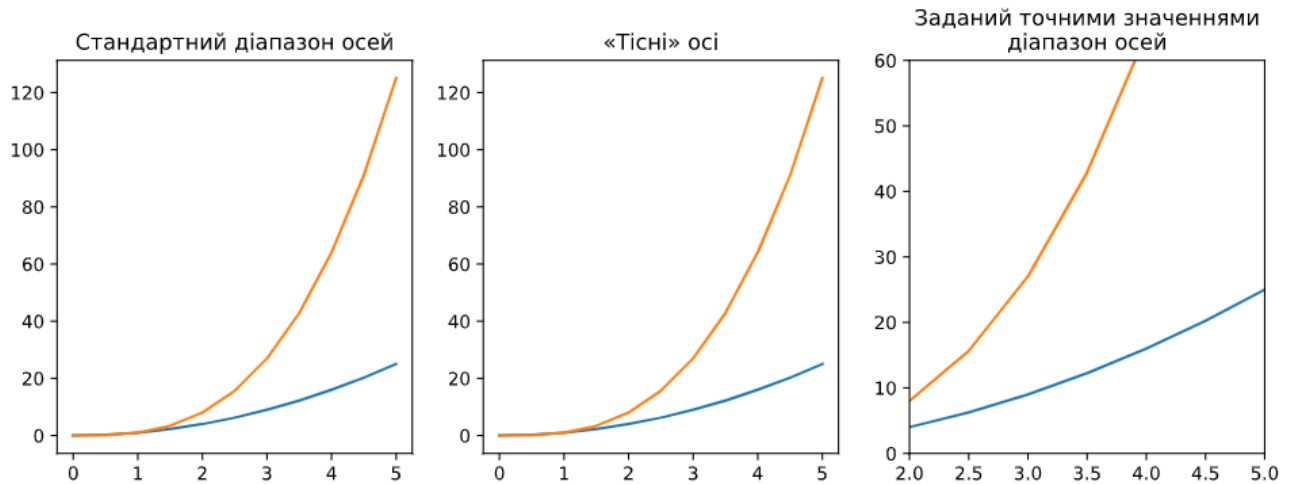
```
fig, ax = plt.subplots()
ax.plot(x, x+1, color="blue", alpha=0.5) # напівпрозора лінія
ax.plot(x, x+2, color="#8B008B")       # RGB
ax.plot(x, x+3, color="#FF8C00");      # RGB
```



1.5 Діапазон значень графіка

Діапазон можна змінювати за допомогою методів `set_ylim`, `set_xlim` і `axis('tight')`:

```
fig, axes = plt.subplots(1, 3, figsize=(12, 4))
axes[0].plot(x, x**2, x, x**3)
axes[0].set_title("Стандартний діапазон осей")
axes[1].plot(x, x**2, x, x**3)
axes[1].axis('tight')
axes[1].set_title("«Тісні» осі")
axes[2].plot(x, x**2, x, x**3)
axes[2].set_ylim([0, 60])
axes[2].set_xlim([2, 5])
axes[2].set_title("Заданий точними значеннями\ndіапазон осей");
```

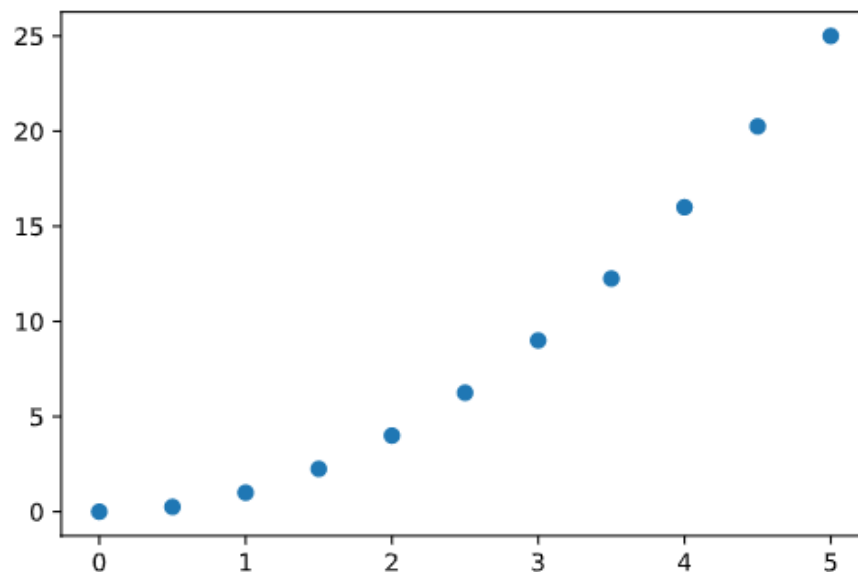


1.6 Інші типи графіків

За допомогою `matplotlib` можна створювати барплоти, гістограми, точкові діаграми і багато ще чого:

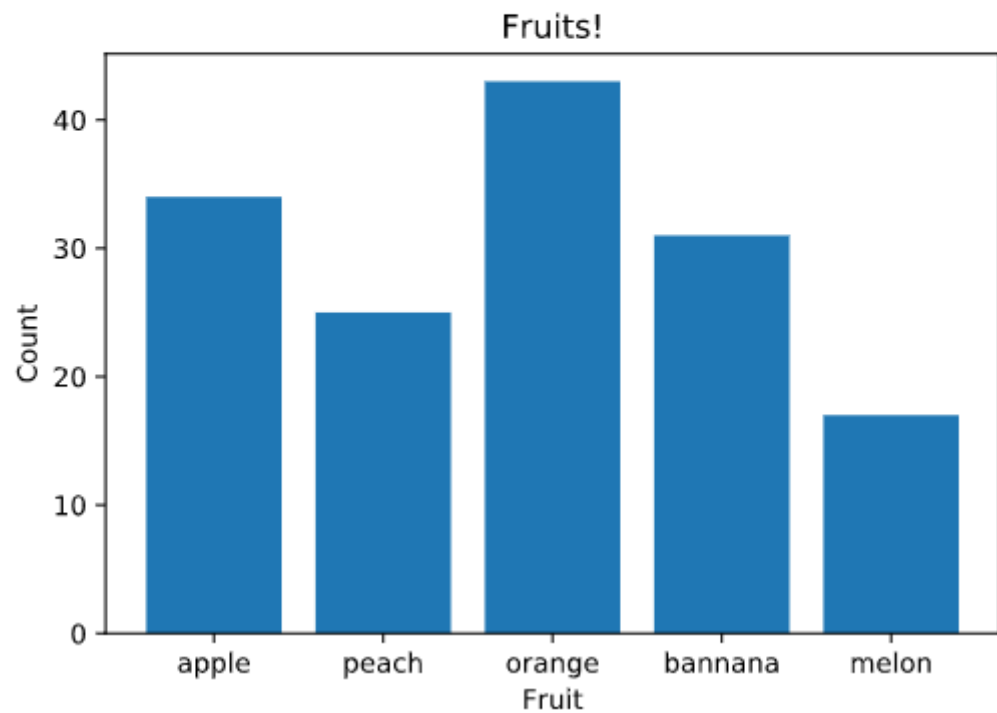
```
[57]: plt.scatter(x,y)
```

```
[57]: <matplotlib.collections.PathCollection at 0x1a107a50f48>
```



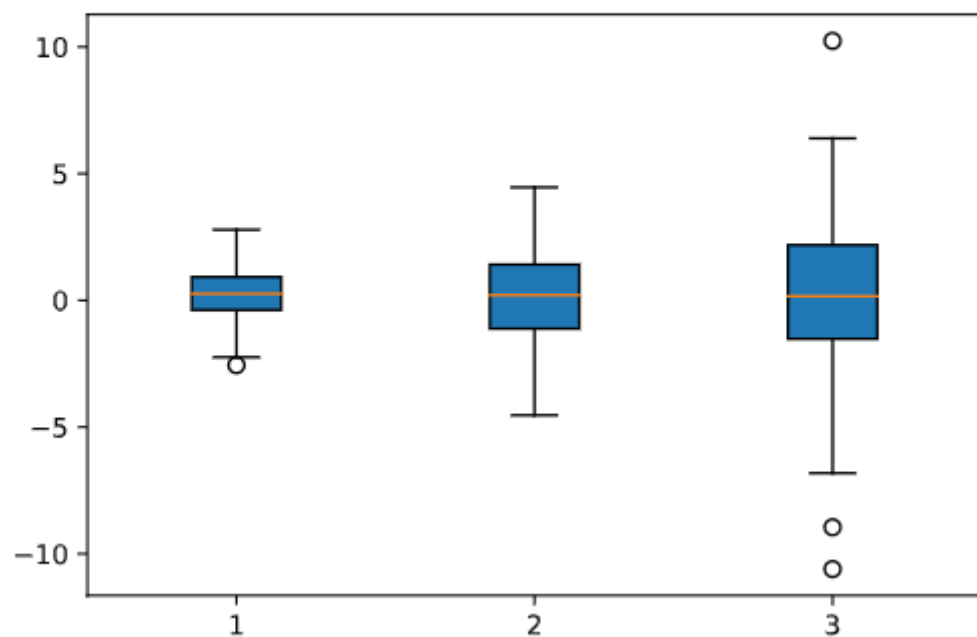
```
[60]: fruits = ["apple", "peach", "orange", "bannana", "melon"]
      counts = [34, 25, 43, 31, 17]
      plt.bar(fruits, counts)
      plt.title("Fruits!")
      plt.xlabel("Fruit")
      plt.ylabel("Count")
```

```
[60]: Text(0, 0.5, 'Count')
```



```
[59]: data = [np.random.normal(0, std, 100) for std in range(1, 4)]

      # rectangular box plot
      plt.boxplot(data,vert=True,patch_artist=True);
```



```
[63]: import matplotlib as mpl
import matplotlib.pyplot as plt
import matplotlib.dates as mdates
import datetime as dt
import csv
data_names = ['Київ', 'Харків', 'Суми', 'Донецьк',
              'Одеса', 'Чернівці', 'Полтава', 'Херсон',
              'Кривий ріг', 'Львів']
data_values = [1076, 779, 222, 189, 1137, 134, 124, 124, 91, 789]
dpi = 80
fig = plt.figure(dpi = dpi, figsize = (512 / dpi, 384 / dpi) )
mpl.rcParams.update({'font.size': 9})
plt.title('Розподіл кафе по городах України (%)')
xs = range(len(data_names))
plt.pie(
    data_values, autopct='%.1f', radius = 1.1,
    explode = [0.15] + [0 for _ in range(len(data_names) - 1)] )
plt.legend(
    bbox_to_anchor = (-0.16, 0.45, 0.25, 0.25),
    loc = 'lower left', labels = data_names )
```

[63]: <matplotlib.legend.Legend at 0x1a10901abc8>



2. Варіанти завдання

Використовуючи засоби matplotlib побудувати графіки відповідно до таблиці варіантів (таб. 10.1).

Таблиця 10.1 – Варіанти завдань

| № | Завдання |
|----|---|
| 1. | Побудувати графік функції $y = 5x^2 - 2x + 8$ тип лінії – “-”, розташування легенди (ліво низ) діапазон значень – y (-10:50) x (-25:50), колір лінії - #80AC55 , прозорість лінії – 69% |
| | Побудувати кругову діаграму (не менше 10 сегментів) "Продажі автомобілів в Україні за 2019 рік", дані можна взяти довільні або знайти в мережі Інтернет. У діаграмі передбачити назви сегментів в поле легенд. |
| 2. | Побудувати графік функції $y = 5x^3 - 4$ тип лінії – “.”, розташування легенди (ліво верх) діапазон значень – y (5:80) x (-40:60), колір лінії - #225533 , прозорість лінії – 76% |
| | Побудувати стовпчикову діаграму (не менше 10 стовпчиків) "Рівень продажу гречки за останні 10 місяців", дані можна взяти довільні або знайти в мережі Інтернет. У діаграмі передбачити назви стовпчиків. Колір використовувати як для попереднього графіку. |
| 3. | Побудувати графік функції $y = 6x^{1/2} + 8$ тип лінії – “--”, розташування легенди (право низ) діапазон значень – y (10:70) x (-15:55), колір лінії - #8EC0BC , прозорість лінії – 41% |
| | Побудувати кругову діаграму (не менше 10 сегментів) "Популяція різновидів їжаків", дані можна взяти довільні або знайти в мережі Інтернет. У діаграмі передбачити назви сегментів в поле легенд. |
| 4. | Побудувати графік функції $y = x/2 + 7$ тип лінії – “-.”, розташування легенди (право верх) діапазон значень – y (-5:35) x (-60:60), колір лінії - #FF0B0B , прозорість лінії – 36% |
| | Побудувати стовпчикову діаграму (не менше 10 стовпчиків) "Температура води Азовського моря за останні 10 місяців", дані можна взяти довільні або знайти в мережі Інтернет. У діаграмі передбачити назви стовпчиків. Колір використовувати як для попереднього графіку |

| № | Завдання |
|----|---|
| 5. | Побудувати графік функції $y = 4e^x - 9$ тип лінії – “.”, розташування легенди (ліво низ) діапазон значень – y (-10:100) x (-35:40), колір лінії - #F0F0F0 , прозорість лінії – 91% |
| | Побудувати кругову діаграму (не менше 10 сегментів) "Кількість постраждалих від пандемії по місяцях", дані можна взяти довільні або знайти в мережі Інтернет. У діаграмі передбачити назви сегментів в поле легенд. |
| 6. | Побудувати графік функції $y = \ln x + 4$ тип лінії – “-.”, розташування легенди (ліво верх) діапазон значень – y (-15:100) x (-45:60), колір лінії - #0F0F0F , прозорість лінії – 88% |
| | Побудувати стовпчикову діаграму (не менше 10 стовпчиків) "Рівень атмосферного тиску за останні 10 днів", дані можна взяти довільні або знайти в мережі Інтернет. У діаграмі передбачити назви стовпчиків. Колір використовувати як для попереднього графіку |
| 7. | Побудувати графік функції $\log x + 6$ тип лінії – “-”, розташування легенди (право низ) діапазон значень – y (-5:100) x (-25:55), колір лінії - #E8E0BD , прозорість лінії – 76% |
| | Побудувати кругову діаграму (не менше 10 сегментів) "Кількість проданого туалетного паперу фірмами за 1й квартал", дані можна взяти довільні або знайти в мережі Інтернет. У діаграмі передбачити назви сегментів в поле легенд. |
| 8. | Побудувати графік функції $y = 2\cos x - 2$ тип лінії – “--”, розташування легенди (право верх) діапазон значень – y (0:90) x (-40:40), колір лінії - #80F0AB , прозорість лінії – 54% |
| | Побудувати стовпчикову діаграму (не менше 10 стовпчиків) "Кількість пожеж за 2019 рік", дані можна взяти довільні або знайти в мережі Інтернет. У діаграмі передбачити назви стовпчиків. Колір використовувати як для попереднього графіку |
| 9. | Побудувати графік функції $y = 3\sin x + 2$ тип лінії – “.”, розташування легенди (ліво низ) діапазон значень – y (5:75) x (-30:30), колір лінії - #00CFFC , прозорість лінії – 68% |
| | Побудувати кругову діаграму (не менше 10 сегментів) "Кількість проданих смартфонів компаніями за 2019", дані можна взяти довільні |

| № | Завдання |
|-----|---|
| | або знайти в мережі Інтернет. У діаграмі передбачити назви сегментів в поле легенд. |
| 10. | <p>Побудувати графік функції $y = \operatorname{tg} x - 4$ тип лінії – “-.”, розташування легенди (ліво верх) діапазон значень – y (10:80) x (-60:35), колір лінії - #55AABB , прозорість лінії – 70%</p> <p>Побудувати стовпчикову діаграму (не менше 10 стовпчиків) "Рівень привабливості котів різних порід згідно голосування", дані можна взяти довільні або знайти в мережі Інтернет. У діаграмі передбачити назви стовпчиків. Колір використовувати як для попереднього графіку</p> |
| 11. | <p>Побудувати графік функції $y = \frac{1}{2} \operatorname{ctg} x + 7$ тип лінії – “--”, розташування легенди (право низ) діапазон значень – y (15:65) x (-50:50), колір лінії - #8F0FBF , прозорість лінії – 90%</p> <p>Побудувати кругову діаграму (не менше 10 сегментів) "загальний обіг різновидів печива магазину «Вацак» за поточний квартал", дані можна взяти довільні або знайти в мережі Інтернет. У діаграмі передбачити назви сегментів в поле легенд.</p> |
| 12. | <p>Побудувати графік функції $y = 5e^x + 8$ тип лінії – “-”, розташування легенди (право верх) діапазон значень – y (-20:50) x (-55:45), колір лінії - #F8F0FB , прозорість лінії – 80%</p> <p>Побудувати стовпчикову діаграму (не менше 10 стовпчиків) "Популяція горобців по регіонах проживання", дані можна взяти довільні або знайти в мережі Інтернет. У діаграмі передбачити назви стовпчиків. Колір використовувати як для попереднього графіку</p> |
| 13. | <p>Побудувати графік функції $y = \ln x - 8$ тип лінії – “.”, розташування легенди (право верх) діапазон значень – y (20:55) x (-30:55), колір лінії - #55FFBB , прозорість лінії – 75%</p> <p>Побудувати кругову діаграму (не менше 10 сегментів) "Зареєстровані порушення ПДД у 2019", дані можна взяти довільні або знайти в мережі Інтернет. У діаграмі передбачити назви сегментів в поле легенд.</p> |
| 14. | <p>Побудувати графік функції $y = x/5 + 17$ тип лінії – “-.”, розташування легенди (право низ) діапазон значень – y (-10:60) x (-35:25), колір лінії - #FFBBFF , прозорість лінії – 100%</p> <p>Побудувати стовпчикову діаграму (не менше 10 стовпчиків) "Середній вік жителів України за останні 10 років", дані можна взяти довільні або</p> |

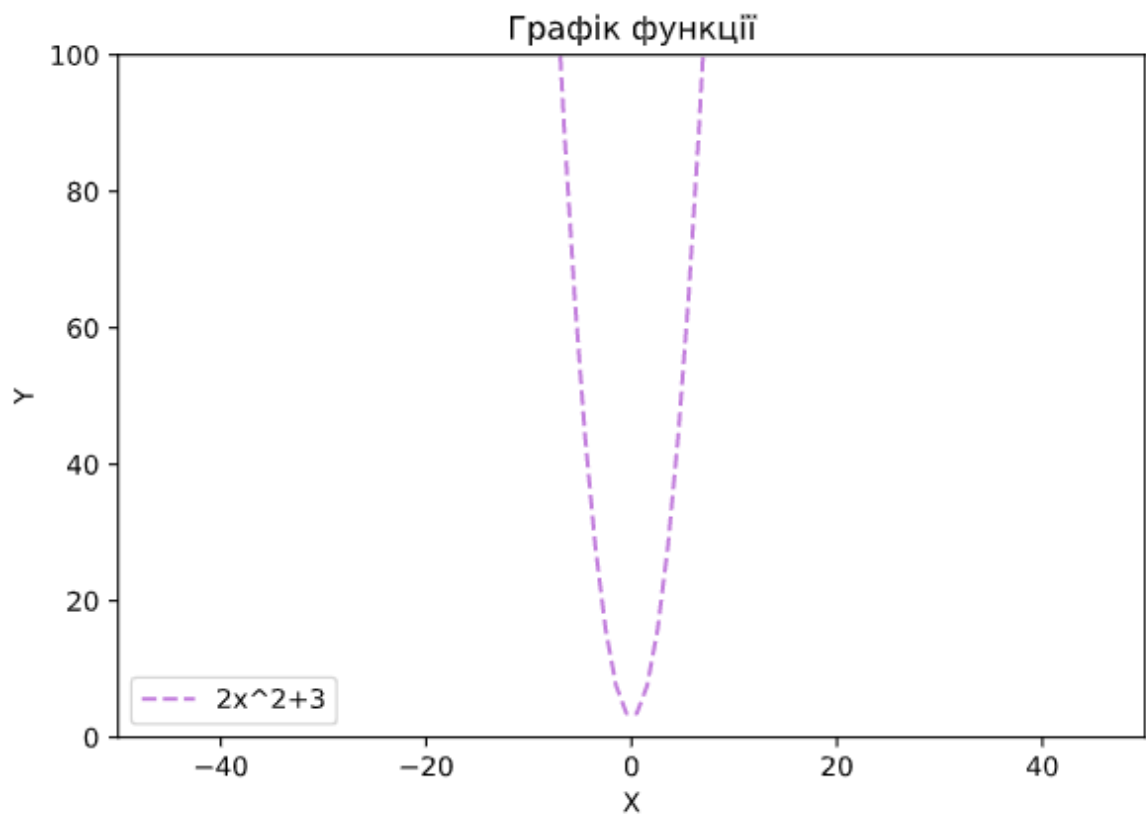
| № | Завдання |
|-----|--|
| | знайти в мережі Інтернет. У діаграмі передбачити назви стовпчиків. Колір використовувати як для попереднього графіку |
| 15. | Побудувати графік функції $y = 5x^{1/3} - 3$ тип лінії – “--”, розташування легенди (ліво верх) діапазон значень – у (-15:70) х (-40:40), колір лінії - #00BBFF , прозорість лінії – 25% |
| | Побудувати кругову діаграму (не менше 10 сегментів) "врожай картоплі за 2019 по областях", дані можна взяти довільні або знайти в мережі Інтернет. У діаграмі передбачити назви сегментів в поле легенд. |
| 16. | Побудувати графік функції $y = 2x^2 + 3$ тип лінії – “-”, розташування легенди (ліво низ) діапазон значень – у (0:100) х (-50:50), колір лінії - #8800BB , прозорість лінії – 50% |
| | Побудувати стовпчикову діаграму (не менше 10 стовпчиків) "Курс долара до гривні за останні 10 місяців", дані можна взяти довільні або знайти в мережі Інтернет. У діаграмі передбачити назви стовпчиків. Колір використовувати як для попереднього графіку |

3. Приклад

Варіант -16.

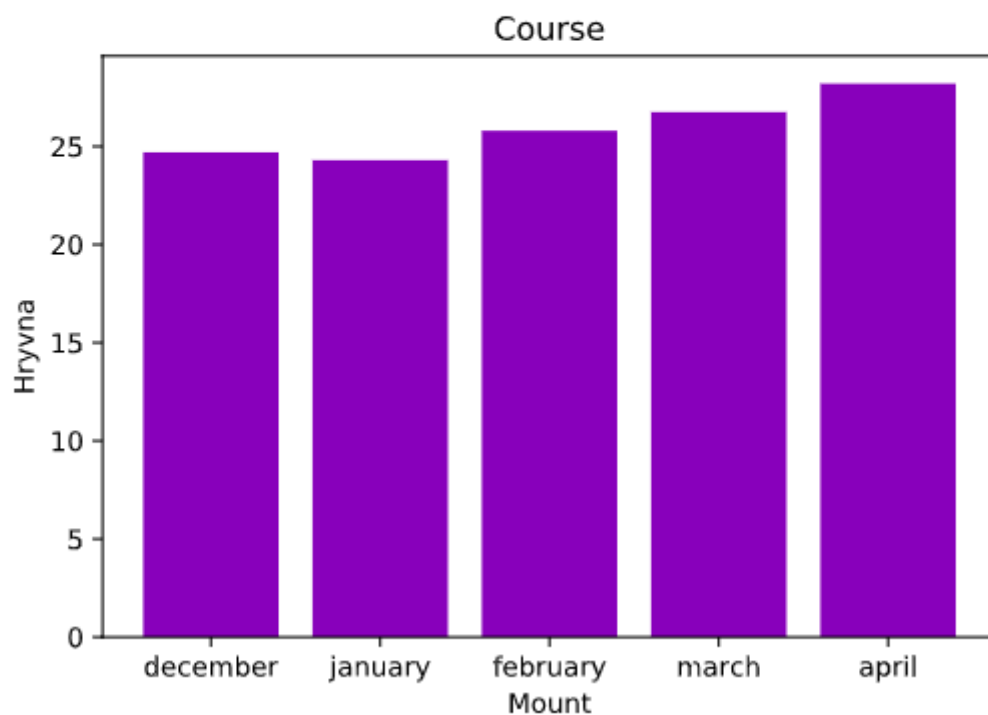
Файл програми

```
[34]: import matplotlib.pyplot as plt
import numpy as np
%matplotlib inline
%config InlineBackend.figure_format = 'svg'
x = np.linspace(-50, 50, 100)
fig=plt.figure()
ax= fig.add_axes([0.1,0.1,0.9,0.9])
ax.plot(x, 2*(x**2)+3, color="#880088", ls="--",label="2x^2+3",alpha=0.5)
ax.set_ylim([0, 100])
ax.set_xlim([-50, 50])
ax.legend(loc=3)
ax.set_title("Графік функції")
ax.set_xlabel("X")
ax.set_ylabel("Y")
fig.savefig("MyFile.png",dpi=800)
```



```
[36]: mounts = ["december", "january", "february", "march", "april"]  
      counts = [24.7, 24.3, 25.8, 26.75, 28.20]  
      plt.bar(mounts, counts, color="#880088")  
      plt.title("Course")  
      plt.xlabel("Mount")  
      plt.ylabel("Hryvna")
```

```
[36]: Text(0, 0.5, 'Hryvna')
```



4. Контрольні запитання

1. Модуль `matplotlib`. Призначення та базові принципи застосування. Наведіть приклади.
2. Опишіть базові принципи та механізми роботи з гістограмами у `matplotlib`.
3. Опишіть що таке легенди та як їх можна розташовувати.
4. Опишіть існуючі види налаштування відображення графіків, наведіть приклади.
5. Які види діаграм підтримуються у `matplotlib`, та якими методами будуються, наведіть приклади.