**WYKRES LINIOWY**

import matplotlib.pyplot as plt

import numpy as np

x = np.linspace(0, 2, 100)

plt.plot(x, x, label='linear')

plt.plot(x, x\*\*2, label='quadratic')

plt.plot(x, x\*\*3, label='cubic')

plt.xlabel('x label')

plt.ylabel('y label')

plt.title("Simple Plot")

plt.legend()

plt.show()

**WYKRES PIERŚCIENIOWY**

import matplotlib.pyplot as plt

import numpy as np

np.random.seed(345)

data = np.random.randint(20, 100, 6)

total = sum(data)

data\_per = data/total\*100

explode = (0.2, 0, 0, 0, 0, 0)

plt.pie(data\_per, explode = explode, labels = [round(i,2) for i in (list(data\_per))])

circle = plt.Circle( (0,0), 0.7, color='white')

p=plt.gcf()

p.gca().add\_artist(circle)

plt.show()

**WYKRES WARSTWOWY**

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

np.random.seed(123)

df = pd.DataFrame(np.random.rand(10, 4), columns=['a', 'b', 'c', 'd'])

df.plot.area()

plt.show()

**WYKRES KOŁOWY**

import matplotlib.pyplot as plt

# Dane

sizes = [20, 30, 40, 10]

labels = ['Kategoria A', 'Kategoria B', 'Kategoria C', 'Kategoria D']

colors = ['red', 'blue', 'green', 'yellow']

explode = (0, 0.1, 0, 0) # Wyróżnienie segmentu Kategoria B

# Tworzenie wykresu kołowego

plt.pie(sizes, explode=explode, labels=labels, colors=colors, autopct='%1.1f%%', shadow=True, startangle=90)

# Dodanie tytułu

plt.title('Przykład wykresu kołowego')

# Równomierne skalowanie osi X i Y, aby koło było okrągłe

plt.axis('equal')

plt.show()

**WYKRES SŁUPKOWY**

słupki pojedyncze

import numpy as np

import matplotlib.pyplot as plt

height = [3, 12, 5, 18, 45]

bars = ('A', 'B', 'C', 'D', 'E')

y\_pos = np.arange(len(bars))

plt.bar(y\_pos, height, color=['black', 'red', 'green', 'blue', 'cyan'])

plt.xticks(y\_pos, bars)

plt.show()

Po 3 słupki dla każdej wartości osi X

import numpy as np

import matplotlib.pyplot as plt

data = [[30, 25, 50, 20],

[40, 23, 51, 17],

[35, 22, 45, 19]]

X = np.arange(4)

plt.bar(X + 0.00, data[0], color='b', width=0.25, label="A")

plt.bar(X + 0.25, data[1], color='g', width=0.25, label="B")

plt.bar(X + 0.50, data[2], color='r', width=0.25, label="C")

labelsbar = np.arange(2015,2019)

plt.xticks(X+0.25,labelsbar)

plt.legend()

plt.show()

Słupki pojedyncze, ale podzielone na dwie części

import numpy as np

import matplotlib.pyplot as plt

N = 5

boys = (20, 35, 30, 35, 27)

girls = (25, 32, 34, 20, 25)

ind = np.arange(N)

width = 0.35

plt.bar(ind, boys, width, label="boys")

plt.bar(ind, girls, width,bottom=boys, label="girls")

plt.ylabel('Contribution')

plt.title('Contribution by the teams')

plt.xticks(ind, ('T1', 'T2', 'T3', 'T4', 'T5'))

plt.yticks(np.arange(0, 81, 10))

plt.legend()

plt.show()

Poziomy

import numpy as np

import matplotlib.pyplot as plt

width = [3, 12, 5, 18, 45]

bars = ('A', 'B', 'C', 'D', 'E')

x\_pos = np.arange(len(bars))

plt.barh(x\_pos, width, color=['black', 'red', 'green', 'blue', 'cyan'])

plt.yticks(x\_pos, bars)

plt.show()

Poziomy po 3 słupki

import numpy as np

import matplotlib.pyplot as plt

data = [[30, 25, 50, 20],

[40, 23, 51, 17],

[35, 22, 45, 19]]

Y = np.arange(4)

plt.barh(Y + 0.00, data[0], color='b', height=0.25, label="A")

plt.barh(Y + 0.25, data[1], color='g', height=0.25, label="B")

plt.barh(Y + 0.50, data[2], color='r', height=0.25, label="C")

labelsbar = np.arange(2015,2019)

plt.yticks(Y + 0.25, labelsbar)

plt.legend()

plt.show()

**WYKRES PUNKTOWY**

import matplotlib.pyplot as plt

house\_prices = [230000, 350000, 480000, 280000, 420000, 610000, 390000, 580000]

square\_meters = [90, 140, 210, 100, 170, 260, 150, 240]

plt.scatter(square\_meters, house\_prices, color='blue', marker='o')

plt.xlabel('Metraż [m2]')

plt.ylabel('Cena domu [PLN]')

plt.title('Związek między metrażem a ceną domu')

plt.show()

**KILKA WYKRESÓW W JEDNYM OKNIE**

import matplotlib.pyplot as plt

import numpy as np

x = np.linspace(0, 2 \* np.pi, 100)

# Tworzenie siatki wykresów 2x2

# Pierwszy wykres (w lewym górnym rogu)

plt.subplot(2, 2, 1)

plt.plot(x, np.sin(x))

plt.title('sin(x)')

# Drugi wykres (w prawym górnym rogu)

plt.subplot(2, 2, 2)

plt.plot(x, np.cos(x))

plt.title('cos(x)')

# Trzeci wykres (w lewym dolnym rogu)

plt.subplot(2, 2, 3)

plt.plot(x, np.tan(x))

plt.title('tan(x)')

# Czwarty wykres (w prawym dolnym rogu)

plt.subplot(2, 2, 4)

plt.plot(x, -np.sin(x))

plt.title('-sin(x)')

# Dopasowanie odstępów między wykresami

plt.tight\_layout()

# Wyświetlenie wykresów

plt.show()

**ZAPIS DO PLIKU**

wywołujemy przed plt.show(), do zapisu jpg potrzebna biblioteka pillow (jest w zestawie piojasa)

plt.savefig('timevsdist.pdf', format='pdf')

**PARAMETRY LEGENDY**

Plt.legend(title=)

Obraz zawierający tekst, zrzut ekranu, numer, Czcionka

Opis wygenerowany automatycznie

**ETYKIETY OSI**

plt.xlabel('Oś X', fontsize=12, color='red', fontweight='bold')

plt.ylabel('Oś Y', fontsize=12, color='green', fontweight='bold')

**ETYKIETY PODZIAŁKI OSI**

xtick\_vals = [0, np.pi, 2 \* np.pi, 3 \* np.pi]

xtick\_labels = ['0', '$\pi$', '$2\pi$', '$3\pi$']

ytick\_vals = [-1, -0.5, 0, 0.5, 1]

ytick\_labels = ['-1', '-1/2', '0', '1/2', '1']

plt.xticks(xtick\_vals, xtick\_labels, fontsize=12, color='red')

plt.yticks(ytick\_vals, ytick\_labels, fontsize=12, rotation=45)

**STYLE I KOLORY LINII**

import numpy as np

import matplotlib.pyplot as plt

x = np.arange(14)

y = np.cos(5 \* x)

plt.plot(x, y + 2, 'blue', linestyle="-", label="niebieski")

plt.plot(x, y + 1, 'red', linestyle=":", label="czerwony")

plt.plot(x, y, 'green', linestyle="--", label="zielony")

plt.legend(title='Legenda:')

plt.show()

Obraz zawierający tekst, zrzut ekranu, Czcionka, numer

Opis wygenerowany automatycznie

**KOLORY**

Obraz zawierający tekst, zrzut ekranu, Wielobarwność, design

Opis wygenerowany automatycznie

**SIATKA**

plt.grid(True, which='both', color='gray', linewidth=0.5, linestyle='--')