МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ

КАФЕДРА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

**ЗВІТ З ЛАБОРАТОРНОЇ РОБОТИ №14**

з дисципліни «Програмування мовою Python»

на тему «Робота з даними різних форматів»

Виконав студент групи КН-33

Глумний Тимур

Перевірила Парфененко Ю.В.

Варіант 3

Суми 2024

1. **Постановка задачі**

**Завдання:**

1. ) написати програму побудови графіка функції ( 1 бал);
2. ) написати програму візуалізації даних з порталу відкритих даних у вигляді графіків та діаграм, дані для побудови графіків повинні зчитуватися з .csv файлу (2 бали);
3. ) написати програму побудови кругової діаграми (1 бал).

**Завдання 1**

Побудуйте графік функції. Оберіть суцільний тип лінії, задайте колір та товщину графіку та позначте осі, виведіть назву графіка на екран, вставте легенду. Використайте бібліотеку Matplotlib.

Зображення, що містить текст, Шрифт, почерк, каліграфія

Автоматично згенерований опис

1. **Код програми:**

import matplotlib.pyplot as plt

import numpy as np

# Задаємо значення x у діапазоні від -3 до 3

x = np.linspace(-3, 3, 500)

# Обчислюємо значення Y(x)

y = 2\*\*x \* np.sin(10 \* x)

# Налаштування графіка

plt.plot(x, y, color='red', linewidth=2, linestyle='-', label=r'$Y(x) = 2^x \cdot \sin(10x)$')

# Додаємо назву, підписи осей та легенду

plt.title("Графік функції Y(x)", fontsize=15)

plt.xlabel("x", fontsize=12, color='green')

plt.ylabel("Y(x)", fontsize=12, color='green')

plt.legend()

plt.grid(True)

# Показуємо графік

plt.show()

1. **Результат роботи програми**

Зображення, що містить текст, Графік, ряд, схема

Автоматично згенерований опис

**Завдання 2**

1. **Постановка задачі**

написати програму візуалізації даних з порталу відкритих даних у вигляді графіків та діаграм, дані для побудови графіків повинні зчитуватися з .csv файлу (2 бали);

1. Код програми

import csv

import matplotlib.pyplot as plt

# Функція для зчитування даних із CSV-файлу

def get\_population\_data(filename):

with open(filename, mode='r') as file:

csv\_reader = csv.reader(file)

headers = next(csv\_reader) # Пропускаємо заголовок

years = [header.split('[')[0] for header in headers[4:]] # Отримуємо роки із заголовків

population\_ukraine = []

population\_france = []

# Обробка рядків файлу

for line in csv\_reader:

country = line[2]

data = [int(value) if value else 0 for value in line[4:]]

if country == 'Ukraine':

population\_ukraine = data

elif country == 'France':

population\_france = data

return years, population\_ukraine, population\_france

# Шлях до вхідного файлу

file\_path = 'Lab14.2.csv'

# Зчитування даних

years, ukraine\_population, france\_population = get\_population\_data(file\_path)

# Нормалізуємо чисельність населення до мільйонів

ukraine\_in\_millions = [value / 1\_000\_000 for value in ukraine\_population]

france\_in\_millions = [value / 1\_000\_000 for value in france\_population]

# Побудова графіка

plt.figure(figsize=(10, 6))

plt.plot(years, ukraine\_in\_millions, color='darkblue', linestyle='-', marker='s', label='Україна')

plt.plot(years, france\_in\_millions, color='crimson', linestyle='--', marker='d', label='Франція')

# Додавання підписів, легенди та заголовка

plt.title('Динаміка чисельності населення України та Франції (2003–2022)', fontsize=16, color='darkgreen')

plt.xlabel('Рік', fontsize=12)

plt.ylabel('Населення (млн осіб)', fontsize=12)

plt.xticks(rotation=45, fontsize=10)

plt.yticks(fontsize=10)

# Налаштування формату значень осі Y

plt.gca().get\_yaxis().set\_major\_formatter(plt.FuncFormatter(lambda x, \_: f'{x:.1f} млн'))

# Відображення сітки та легенди

plt.grid(visible=True, linestyle=':', linewidth=0.7, alpha=0.8)

plt.legend(fontsize=12)

# Відображення графіка

plt.tight\_layout()

plt.show()

1. Результат роботи програми

Зображення, що містить текст, ряд, Графік, схема

Автоматично згенерований опис

**Завдання 3**

1. **Постановка задачі**

Побудуйте кругову діаграму на основі даних з предметної області лабораторної роботи №12. Використайте бібліотеку Matplotlib.

1. **Код програми:**

import matplotlib.pyplot as plt

# Дані про інфляцію в Україні (річні %)

years = [

1993, 1994, 1995, 1996, 1997, 1998, 1999, 2000, 2001, 2002, 2003,

2004, 2005, 2006, 2007, 2008, 2009, 2010, 2011, 2012, 2013, 2014, 2015,

2016, 2017, 2018, 2019

]

inflation\_rates = [

4734.91, 891.19, 376.75, 80.33, 15.94, 10.58, 22.68, 28.20, 11.96,

0.76, 5.18, 9.05, 13.57, 9.05, 12.84, 25.23, 15.88, 9.37, 7.96, 0.57,

0, 12.07, 48.70, 13.91, 14.44, 10.95, 7.89

]

# Усунення від'ємних значень

inflation\_rates = [rate if rate > 0 else 0 for rate in inflation\_rates]

# Перевірка, чи збігаються кількість років і значень

if len(years) != len(inflation\_rates):

print("Кількість років не відповідає кількості даних!")

else:

# Налаштування кольорів

colors = plt.cm.viridis([i / len(inflation\_rates) for i in range(len(inflation\_rates))])

# Побудова кругової діаграми

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

wedges, labels, percentages = ax.pie(

inflation\_rates, labels=years, autopct='%1.1f%%', startangle=140, colors=colors,

wedgeprops={'edgecolor': 'gray'}

)

# Налаштування підписів

for label in labels:

label.set\_fontsize(9)

for percentage in percentages:

percentage.set\_fontsize(9)

# Додавання заголовка

ax.set\_title("Річна інфляція в Україні (1993-2019)", fontsize=14, pad=20)

# Вирівнювання кругової діаграми

ax.axis('equal')

# Відображення

plt.show()

1. **Результат роботи програми**

Зображення, що містить знімок екрана, коло, Барвистість, Графіка

Автоматично згенерований опис

**https://github.com/doomedd/Python-laba-14**