**Documentatie exercitiul 2**

***Prerequisites și configurare:***

Este nevoie să aveți instalată o versiune a Python pe calculator.

Deschideti cmd-ul și rulati **pip install matplodlid** pentru a descarca si instala o biblioteca ce genereaza grafice.

Apoi rulati comanda **pip install pandas** pentru a descarca si instala o biblioteca ce ne va permite sa manipulam si sa afisam datele.

Pentru a efectua si bonusul, trebuie sa instalam si Flask. Astfel, rulați comanda **pip install flask**.

Rulați **python app.py**. Vă va apărea în cmd o adresă http. Copiati-o si apoi lipiti-o intr-un browser precum Chrome. Ați deschis astfel o pagina web in care cele trei grafice vor fi disponibile.

***Explicarea codului din ex2.py:***

**matplotlib.pyplot as plt**: Este biblioteca utilizată pentru a crea și a desena grafice

**pandas as pd**: Este biblioteca folosită pentru manipularea și analiza datelor, inclusiv citirea fișierelor CSV.

**pd.read\_csv("data.csv"):** Citește fișierul CSV și returnează un obiect DataFrame

**df**: Aceasta este variabila principală care conține datele.

**df.plot():** Creează un grafic pentru toate valorile din DataFrame df. Fiecare coloană din DataFrame devine un set de date pentru un graf.

**plt.title("Toate valorile"):** Setează titlul graficului.

**plt.xlabel("Index") și plt.ylabel("Valori"):** Setează etichetele axelor X și Y.

**plt.legend(title="Legenda"):** Adaugă o legendă la grafic cu titlul "Legenda".

**df.head(6):** Selectează primele 6 rânduri din DataFrame df. Acest subset de date va fi utilizat pentru a crea graficul.

**style=['r--', 'b-', 'g:', 'm', 'y']:** Setează stilurile diferite de linii și culori pentru fiecare coloană în parte:

* 'r--': Linia roșie întreruptă.
* 'b-': Linia albastră solidă.
* 'g:': Linia verde punctată.
* 'm': Linia magenta solidă.

**df[['Durata' , 'Puls']].tail(12):** Selectează ultimele 12 rânduri ale coloanelor Durata și Puls din DataFrame **df**.

**.plot():** Creează un grafic pentru aceste coloane selectate. Acestea vor fi reprezentate pe axa Y, iar pe axa X vor fi indexurile rândurilor.

**BONUS:**

Această aplicație web este dezvoltată folosind Flask și Python. Ea permite afișarea de grafice generate dintr-un fișier CSV. Aplicația are o interfață simplă, cu o rută principală care redirecționează către o pagină HTML și alte rute care generează grafice pe baza unui parametru din URL.

ATENTIE!!!

Fișierul **index.html** trebuie sa se afle intr-un director **templates** care se afla in acelasi director cu fisierul **app.py**.

**Explicatie pentru codul app.py:**

1. Inițializarea Aplicației Flask:

* Instanța aplicației este creată cu **Flask(\_\_name\_\_)**, permițând configurarea rutelor și gestionarea cererilor HTTP.
* Fișierul CSV este localizat în calea absolută dată de **os.path.join(os.getcwd(), 'data.csv')**.

1. Setarea Backend-ului Matplotlib:

* Matplotlib este configurat să folosească backend-ul '**Agg**' pentru a funcționa fără o interfață grafică. Aceasta este necesar pentru aplicațiile web deoarece nu se bazează pe display-ul fizic al graficelor.

1. Definirea Rutelor:

* Ruta Principală (/):
  + Aceasta încarcă pagina index.html folosind **render\_template("index.html").** Pagina HTML este stocată în directorul **templates**.
* Rutele pentru Generarea Grafice (**/plot/<int:plot\_id>):**
  + Graficele sunt generate pe baza valorii plot\_id specificate în URL.
  + plot\_id == 0: Un grafic general cu toate valorile din fișierul CSV.
  + plot\_id == 1: Primele 6 rânduri ale datelor sunt plotate cu stiluri de culoare diferite.
  + plot\_id == 2: Ultimele 12 rânduri ale coloanelor Durata și Puls sunt plotate cu culori specifice.
* Pentru fiecare plot\_id, datele sunt încărcate într-un DataFrame pandas, apoi graficele sunt create cu Matplotlib.
* Graficele sunt salvate într-un buffer de memorie (**io.BytesIO())** și transformate într-un format PNG pentru a putea fi trimise clientului.
* Utilizăm send\_file(buf, mimetype="image/png") pentru a trimite graficul generat către client sub forma unui fișier PNG.

1. Pornirea Aplicației:

* Aplicația este pornită cu app.run(debug=True) pe 127.0.0.1 și portul 5000, cu modul debug activat pentru o mai bună gestionare a erorilor și reîmprospatare automată.

**Explicație despre interfața HTML (index.html):**

**<!DOCTYPE html>**: Declarația pentru tipul documentului HTML5.

**<html lang="en">**: Specifică limba paginii (engleză).

**<meta charset="UTF-8">**: Setează codificarea caracterelor la UTF-8.

**<meta name="viewport" content="width=device-width, initial-scale=1.0">**: Asigură că pagina web se adaptează la lățimea dispozitivului și se scalează corect.

**<title>**: Setează titlul paginii în tab-ul browserului.

**<h1>**: Titlul principal al paginii.

**<ul>**: Lista neordonată.

**<li>**: Element listat individual.

**<a href="/plot/0">...**: Linkuri către rutele pentru generarea graficelor.

***Bibliografie:***

<https://matplotlib.org/stable/tutorials/pyplot.html>

<https://www.geeksforgeeks.org/graph-plotting-in-python-set-1/>

<https://www.w3schools.com/graphics/>

<https://blog.hubspot.com/website/html-a-href>

<https://chatgpt.com/share/675e3c39-2460-8010-abbb-3da778bc9711>