## 3BIT

**Мета:** Розробка додатка для візуалізації CSV-наборів даних за допомогою Matplotlib та базових принципів ООП (наслідування, інкапсуляція, поліморфізм)

## Умова завдання:

Завдання 1: Вибір CSV-набору даних

Оберіть CSV-набір даних, який ви хочете візуалізувати. Переконайтеся, що він містить відповідні дані для створення змістовних візуалізацій.

Завдання 2: Завантаження даних з CSV

Напишіть код для завантаження даних з CSV-файлу в ваш додаток Python. Використовуйте бібліотеки, такі як Pandas, для спрощення обробки даних.

Завдання 3: Дослідження даних

Визначте екстремальні значення по стовцям

Завдання 4: Вибір типів візуалізацій

Визначте, які типи візуалізацій підходять для представлення вибраних наборів даних. Зазвичай це може бути лінійні графіки, стовпчикові діаграми, діаграми розсіювання, гістограми та секторні діаграми.

Завдання 5: Підготовка даних

Попередньо обробіть набір даних за необхідністю для візуалізації. Це може включати виправлення даних, фільтрацію, агрегацію або трансформацію.

Завдання 6: Базова візуалізація

Створіть базову візуалізацію набору даних, щоб переконатися, що ви можете відображати дані правильно за допомогою Matplotlib. Розпочніть з простої діаграми для візуалізації однієї змінної.

Завдання 7: Розширені візуалізації

Реалізуйте більш складні візуалізації, виходячи з характеристик набору. Поекспериментуйте з різними функціями Matplotlib та налаштуваннями.

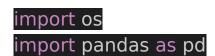
Завдання 8: Декілька піддіаграм

Навчіться створювати кілька піддіаграм в межах одного малюнка для відображення декількох візуалізацій поруч для кращого порівняння.

Завдання 9: Експорт і обмін

Реалізуйте функціональність для експорту візуалізацій як зображень (наприклад, PNG, SVG) або інтерактивних веб-додатків (наприклад, HTML)

Текст програми:



```
from .factories.VisualizationFactory import VisualizationFactory
file path = "/Users/marichka/labs-python/lab 8/data/people.csv"
print(f"Trying to load file from: {file path}")
import os
import pandas as pd
from .factories.VisualizationFactory import VisualizationFactory
class CSVVisualization:
def init (self):
 self.visualization factory = VisualizationFactory()
 def load data(self, file path: str) -> pd.DataFrame:
if not os.path.isfile(file path):
  print(f"Файл {file path} не з<u>найдено.")</u>
   return None
 try:
 data = pd.read_csv(file path)
   print("Дані успішно завантажено.")
     return data
 except Exception as e:
 print(f"Помилка при завантаженні CSV: {e}")
  return None
 def visualize data(self, data: pd.DataFrame, visualization type:
str, x: str, y: str):
try:
      strategy =
self.visualization factory.create strategy(visualization type)
if x not in data.columns or y not in data.columns:
  print(f"Одна з колонок ({x}, {y}) не знайдена в
даних.")
  return
  data_x = pd.to_numeric(data[x], errors="coerce")
 data y = pd.to numeric(data[y], errors="coerce")
      if data x.isnull().any() or data y.isnull().any():
```

```
print("У даних є некоректні значення.")
return
strategy.visualize(data_x, data_y)
except ValueError as e:
print(f"ValueError: {e}")
except Exception as e:
print(f"Помилка: {e}")
```

import os import sys

from shared.logger import Logger

```
project_root =
os.path.abspath(os.path.join(os.path.dirname(__file__), "..", ".."))
sys.path.append(project_root)
```

from .UI.console.ConsoleUI import ConsoleUI

```
def main():
Logger.log("Lab 8 запущена")

ui = ConsoleUI()

ui.display menu()
```

**Висновки:** виконавши ці завдання, я створила багатофункціональний додаток для візуалізації CSV-наборів даних за допомогою Matplotlib.