

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»

Кафедра інформаційних систем та мереж



Звіт

Про виконання лабораторної роботи №8
з дисципліни «Спеціалізовані мови програмування»
на тему « Візуалізація та обробка даних за допомогою спеціалізованих
бібліотек Python »

Виконала:

Студентка гр. РІ-21

Зузяк Л. Р.

Прийняв:

Щербак С.С.

Львів 2024

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

План роботи

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)

Виконавши ці завдання, ви створите багатофункціональний додаток для візуалізації CSV-наборів даних за допомогою Matplotlib. Цей проект покращить ваші навички візуалізації даних, дозволяючи досліджувати результати з різноманітними наборами даних

Виконання:

```
from visualizer.csv_loader import CSVLoader
from visualizer.data_analyzer import DataAnalyzer
from visualizer.basic_visuals import BasicVisuals
from visualizer.advanced_visuals import AdvancedVisuals
from visualizer.multi_subplots import MultiSubplots
from visualizer.exporter import Exporter
from visualizer.config import Config

def main():
    print("=== Data Visualizer Application ===")
```

```

# Step 1: Load CSV data
csv_path = input("Enter the path to your CSV file (e.g., datasets/example.csv): ")

loader = CSVLoader(csv_path)
data = loader.load_data()

if data is None:
    print("Failed to load data. Exiting.")
    return

# Step 2: Analyze data
analyzer = DataAnalyzer(data)
print("\n=== Data Summary ===")
print(analyzer.get_summary_statistics())

print("\n=== Column Extremes ===")
extremes = analyzer.get_column_extremes()
for column, values in extremes.items():
    print(f"{column}: Min={values['min']}, Max={values['max']}")

# Step 3: Choose visualization type
print("\n=== Visualization Options ===")
print("1. Line Chart")
print("2. Bar Chart")
print("3. Scatter Plot")
print("4. Histogram")
print("5. Multiple Subplots")

choice = input("Select a visualization type (1-5): ")

# Step 4: Visualize data
visuals = None
if choice in ["1", "2"]:
    visuals = BasicVisuals(data)
elif choice in ["3", "4"]:
    visuals = AdvancedVisuals(data)
elif choice == "5":
    visuals = MultiSubplots(data)
else:
    print("Invalid choice. Exiting.")
    return

if choice == "1":
    x_col = input("Enter the column for X-axis: ")
    y_col = input("Enter the column for Y-axis: ")
    visuals.line_chart(x_col, y_col)

```

```

elif choice == "2":
    x_col = input("Enter the column for X-axis: ")
    y_col = input("Enter the column for Y-axis: ")
    visuals.bar_chart(x_col, y_col)
elif choice == "3":
    x_col = input("Enter the column for X-axis: ")
    y_col = input("Enter the column for Y-axis: ")
    visuals.scatter_plot(x_col, y_col)
elif choice == "4":
    col = input("Enter the column for the histogram: ")
    bins = int(input("Enter the number of bins (default 10): ") or 10)
    visuals.histogram(col, bins)
elif choice == "5":
    x_col = input("Enter the column for X-axis: ")
    y_cols = input("Enter the columns for Y-axis (comma-separated): ")
    y_cols = y_cols.split(',')
    y_cols = [col.strip() for col in y_cols]
    visuals.multiple_charts(x_col, y_cols)

# Step 5: Export visualization
export_choice = input("\nDo you want to export the visualization? (yes/no): ")
export_choice = export_choice.strip().lower()
if export_choice == "yes":
    exporter = Exporter(Config.OUTPUT_DIR)
    filename = input("Enter the filename for export (without extension): ")
    format_choice = input("Enter the format (png/svg, default png): ")
    format_choice = format_choice.strip().lower() or "png"
    exporter.save_plot(filename, format_choice)
else:
    print("Visualization not exported.")

print("\n=== Done ===")
print("Thank you for using the Data Visualizer!")

if __name__ == "__main__":
    main()

```

Main.py

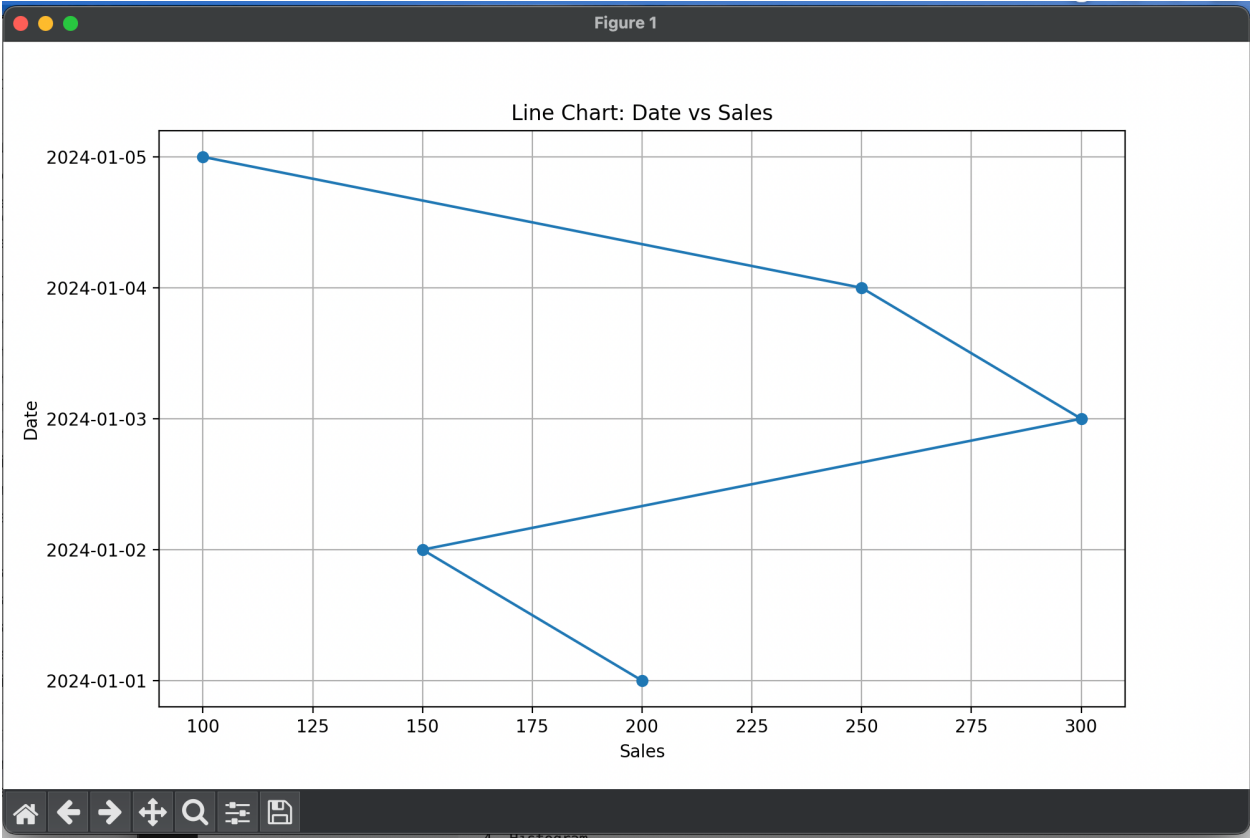
Результат виконання:

```
o (data_visualizer_env) admin@admins-MacBook-Air spl_8 % python3 main.py
=== Data Visualizer Application ===
Enter the path to your CSV file (e.g., datasets/example.csv): datasets/example.csv
Data successfully loaded from datasets/example.csv

=== Data Summary ===
      Sales      Profit
count  5.000000  5.000000
mean   200.000000 52.000000
std     79.056942 23.874673
min     100.000000 20.000000
25%     150.000000 40.000000
50%     200.000000 50.000000
75%     250.000000 70.000000
max     300.000000 80.000000

=== Column Extremes ===
Sales: Min=100, Max=300
Profit: Min=20, Max=80

=== Visualization Options ===
1. Line Chart
2. Bar Chart
3. Scatter Plot
4. Histogram
5. Multiple Subplots
Select a visualization type (1-5): 1
Enter the column for X-axis: Sales
Enter the column for Y-axis: Date
2024-11-28 20:53:01.460 python3[37808:19024471] +[IMKClient subclass]: chose IMKClient_Legacy
2024-11-28 20:53:01.460 python3[37808:19024471] +[IMKInputSession subclass]: chose IMKInputSession_Legac
y
█
```



Висновок: в ході виконання лабораторної роботи я розробила додаток для візуалізації CSV-наборів даних за допомогою Matplotlib та базових принципів ООП (наслідування, інкапсуляція, поліморфізм)