**Proiect_Python_Audit_Financiar_with_GUI_by_ing.Popescu_Vlad_Gabriel**

Acest proiect realizează un audit financiar complet asupra unei firme folosind Python. Include testări financiare automate, vizualizări grafice și interfață grafică pentru utilizatori non-tehnici.

:file_folder: Structura Directorului

audit_project/

```
— data/                          # Aici punem fișierele CSV/Excel de la firmă
|   ├— cash.csv
|   ├— payable.csv
|   ├— receivable.csv
|   ├— cost_sales.csv
|   ├— operating_expenses.csv
|   └— ppe.csv
 ├— output/                      # Salvăm rezultatele (Excel, grafice, etc.)
|        └— raport_audit.xlsx
 ├— output/charts/
|   └— cost_of_sales_hist.png, operating_expenses_bar.png, receivable_heatmap.png
 ├— scripts/
|   ├— __init__.py
|   ├— main.py                   # Scriptul principal care coordonează tot
|   ├— main_gui.py          # Scriptul genereaza Interfață grafică (GUI)
|   ├— auditor.py                # Clasa principală de audit
|   └— utils.py            # Funcții ajutătoare (curățare, sampling, etc.)
|   └— generate_charts.py     # Script care genereaza graficele Seaborn/Matplotlib
|
 ├— requirements.txt         # Librării necesare
└— README.md                # Documentație proiect
└— Proiect_Python_Audit_Financiar_with_GUI.pdf  # Referat proiect
```

:hammer: Funcționalități principale

- Audit pentru următoarele componente:
  - Cash
  - Payable
  - Receivable
  - Cost of Sales
  - Operating Expenses
  - PPE (Active fixe)
- Eșantionare randomizată a datelor
- Vizualizări grafice (heatmap, histogram, bar chart)
- Export rezultate în CSV și Excel
- Interfață grafică intuitivă (Tkinter)

:rocket: Cum rulezi proiectul

1. Instalare dependințe (folosește mediu virtual recomandat):
```bash
pip install -r requirements.txt
```

2. Lansare GUI:
```bash
python scripts/main_gui.py
```

3. Executare audit complet (CLI):
```bash
python scripts/run_audit.py
```

4. Generare grafice (salvate automat în outputs/charts):
```bash
python scripts/generate_charts.py
```

:chart_with_upwards_trend: Grafice generate

- **Receivable Correlation Heatmap**
- **Operating Expenses Bar Chart**
- **Cost of Sales Histogram**

:notebook: Exemple de fișiere CSV de intrare (`data/`)

Fiecare fișier trebuie să conțină coloane relevante, de exemplu:
- `cash.csv` → `begin_balance`, `inflows`, `outflows`, `end_balance`
- `receivable.csv` → `invoice_amount`, `received_amount`

:white_check_mark: Validare
- Proiectul este modular, testabil și extensibil
- Codul folosește docstring-uri și `type hints`
- Organizare clară pe funcții și clase reutilizabile

:inbox_tray: Output
- Fișiere `.csv` și `.xlsx` pentru fiecare test
- Grafice `.png` pentru fiecare analiză vizuală

Realizat cu :heart: folosind Python 3 și biblioteci open-source.

Resurse Necesare

-windows.10
- Python 3.8+
- Librarii: `pandas`, `matplotlib`, `seaborn`, `tk`, `openpyxl`

Licență


Realizat de ing.Popescu Vlad Gabriel
Data: 10.05.2025