

Aplicație de Urmărire Cheltuieli – Python + Tkinter

Descriere

Această aplicație este un **Tracker de Cheltuieli** dezvoltată în Python, care permite utilizatorilor să adauge cheltuieli zilnice, să le salveze local și să vizualizeze distribuția cheltuielilor pe categorii sub formă de grafic. Proiectul este realizat folosind:

- **Tkinter** pentru interfața grafică
- **Matplotlib** pentru a vizualiza cheltuielile pe categorii
- **JSON** pentru stocarea datelor local
- **Datetime** pentru a obține data curentă

Funcționalități

Aplicația permite:

1. Adăugarea cheltuielilor: utilizatorii pot introduce suma, categoria și o descriere a cheltuieli.
2. Vizualizarea cheltuielilor pe categorii: graficul arată distribuția cheltuielilor pe categorii sub formă de bar chart.
3. Stocarea datelor: toate cheltuielile sunt salvate într-un fișier `cheltuieli.json`, care este citit și actualizat automat la fiecare adăugare.

Cum se instalează și rulează aplicația

Pași pentru instalare

1. Asigură-te că ai **Python** instalat. Poți verifica acest lucru rulând comanda:
`python --version`
2. Descarcă fișierul `aplicatie_cheltuieli.py`
3. Instalează librăria `matplotlib`, care este folosită pentru crearea graficului, rulând următoarea comandă:

```
pip install matplotlib
```

4. Deschide un terminal și navighează la directorul în care ai salvat fișierul `aplicatie_cheltuieli.py`
5. După ce te afli în directorul corect, rulează aplicația cu comanda:

```
python aplicatie_cheltuieli.py
```

Detalii despre funcționalități

Funcțiile cheie ale aplicației

1. `incarca_cheltuieli()`
 - Citește și încarcă datele din fișierul `cheltuieli.json` (dacă fișierul există).
 - Dacă fișierul nu există, returnează o listă goală.
2. `salveaza_cheltuieli(cheltuieli)`
 - Salvează lista de cheltuieli în fișierul `cheltuieli.json`, folosind funcția `json.dump()`.
3. `adauga_cheltuiala(suma, categorie, descriere)`
 - Aduagă o nouă cheltuială în lista de cheltuieli și salvează automat fișierul.
4. `arata_grafic_cheltuieli()`

- Creează un grafic de tip bar chart care arată cheltuielile pe categorii. Folosește **matplotlib** pentru a vizualiza datele.

5. `trimite()`

- Funcția apelată atunci când utilizatorul apasă butonul „Adaugă Cheltuială”. Validă suma introdusă și adaugă cheltuială la listă.

Exemplu de utilizare

După ce aplicația se deschide:

1. Introdu suma cheltuielii (ex: 50).
2. Selectează categoria din lista derulantă (ex: „Mâncare”).
3. Completează descrierea cheltuielii (ex: „Cumpărată pâine”).
4. Apasă pe butonul „**Adaugă Cheltuială**” pentru a adăuga cheltuiala.
5. Pentru a vizualiza cheltuielile pe categorii, apasă pe „**Arată Grafic**”.

Posibile extensii

- **Salvarea în SQLite:** În loc de fișiere JSON, datele pot fi salvate într-o bază de date SQLite, ceea ce ar permite gestionarea mai eficientă a acestora.
- **Filtrarea cheltuielilor:** Adăugarea unui sistem de filtrare pe perioade (ex: cheltuieli lunare).
- **Exportul cheltuielilor:** Posibilitatea de a exporta cheltuielile într-un fișier CSV sau PDF pentru a le analiza mai ușor.
- **Design mai avansat:** Folosirea unui framework mai avansat pentru GUI, cum ar fi PyQt5 sau CustomTkinter.

