

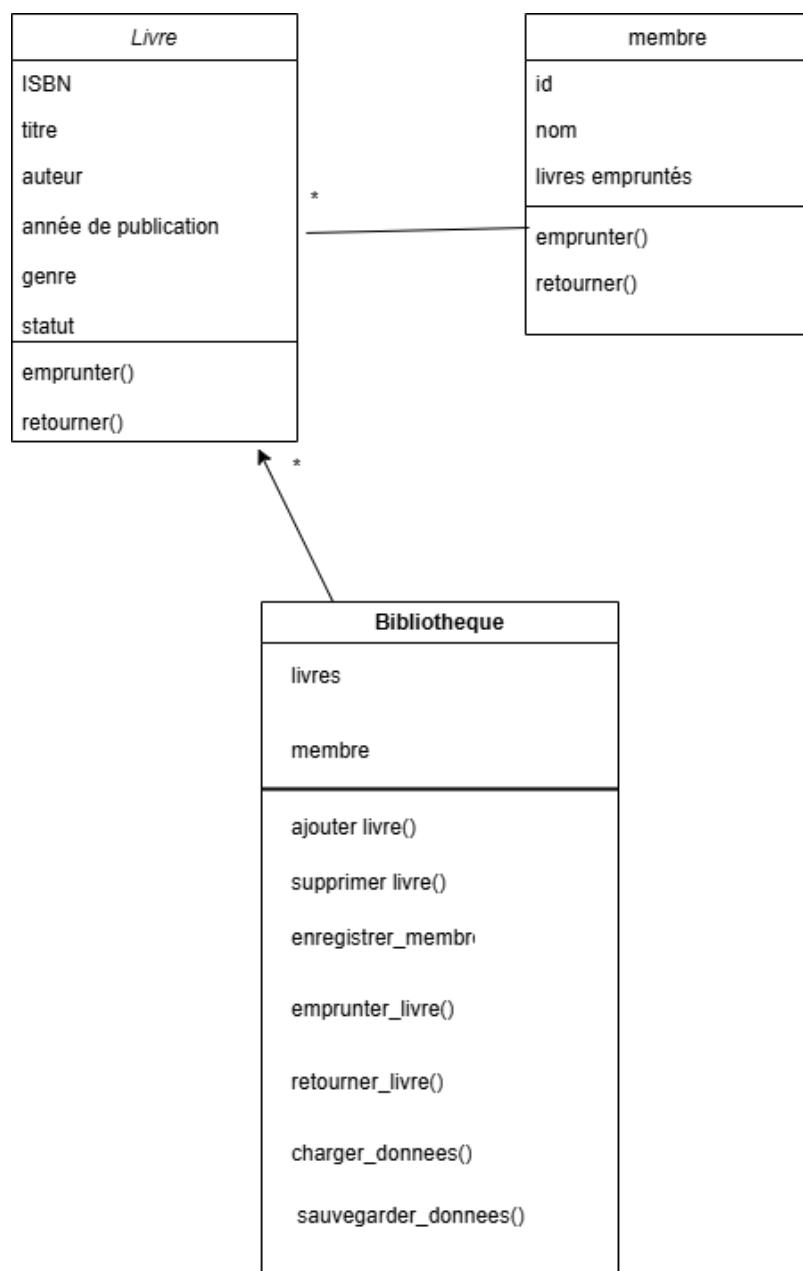
Rapport de Projet : Système de Gestion de Bibliothèque en Python

1-Introduction :

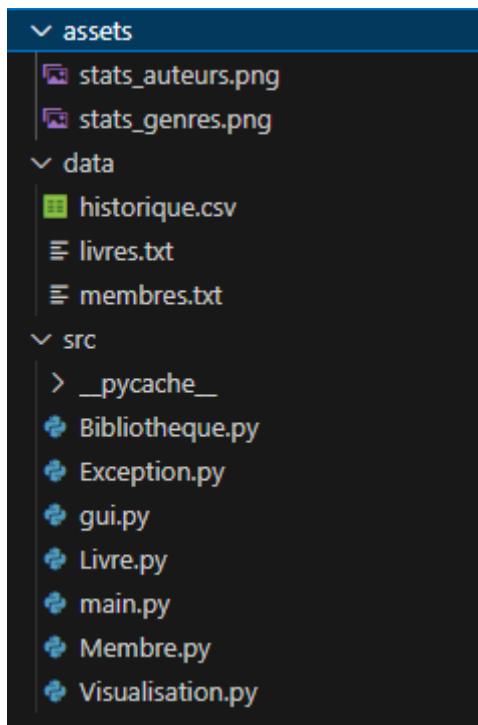
Ce projet vise à développer une application de gestion de bibliothèque en Python, intégrant :

- Un système POO pour la logique métier.
- Une interface en ligne de commande (CLI) et une interface graphique (Tkinter).
- La gestion d'erreurs via des exceptions personnalisées.
- La persistance des données (fichiers JSON/CSV).
- Un tableau de bord statistique avec Matplotlib.

2-Architecture du Projet :



2.2-Structure des fichiers :



3-Fonctionnalités Implémentées :

3.1 Gestion des Ressources

Le système offre des fonctionnalités complètes d'ajout et de suppression de livres et de membres. Chaque opération est rigoureusement contrôlée afin de garantir l'intégrité des données. Par exemple, l'emprunt d'un livre s'accompagne d'un ensemble de vérifications systématiques portant sur :

- ✓ *Disponibilité du livre*
- ✓ *Existence du membre et du livre*
- ✓ *Quota d'emprunts autorisé*

Ces contrôles sont réalisés grâce à des exceptions personnalisées, qui assurent une gestion rigoureuse des cas d'erreur :

- **MembreInexistantError :**
Levée si l'identifiant du membre n'existe pas dans le système. Elle est vérifiée avant toute opération d'emprunt ou de retour.
- **LivreInexistantError :**
Levée si le nom du livre n'existe pas dans le système. Il est vérifié avant chaque emprunt ou retour.
- **LivreIndisponibleError :**
Levée si tous les exemplaires d'un livre sont déjà empruntés. Cette vérification repose sur le champ stock associé à chaque livre dans le dictionnaire.
- **QuotaEmpruntDepasseError :**
Levée si un membre tente d'emprunter plus de 4 livres simultanément. Le système impose une limite stricte à 4 emprunts actifs.

3.2 Interfaces Utilisateur

L'interface en ligne de commande offre un menu complet avec des invites claires, tandis que l'interface graphique (Tkinter) présente une organisation en onglets thématiques. Les deux interfaces

DOUAE BENACHOUR

partagent la même logique métier, assurant une expérience cohérente.

gestion de bibliothèque

Livres Membres Emprunts Statistiques

Ajouter un Livre

ISBN :

Titre :

Auteur :

Année de publication :

Genre :

Quantité :

gestion de bibliothèque

Livres Membres Emprunts Statistiques

Ajouter un membre

ID :

Nom complet :

gestion de bibliothèque

Livres Membres Emprunts Statistiques

Emprunter un livre

Id membre :

Titre du livre emprunté :

Rendre un livre

Id membre :

Titre du livre emprunté :

DOUAE BENACHOUR



3.3 Persistance des Données

Les données sont sauvegardées automatiquement dans des fichiers JSON (livres et membres) et CSV (historique). Le système inclut des mécanismes de validation pour prévenir la corruption des données lors des opérations de lecture/écriture.

Two screenshots of text files in a 'Bloc-notes' (Notepad) application. The first file, 'membres.txt', contains the following JSON-like data:

```
2;salma;
```

The second file, 'livres.txt', contains the following JSON-like data:

```
1;la boite à merveille;ahmed sefreoui;3;roman;disponible;4
```

A	B	C	D
23/06/2025	1234	1	emprunt
23/06/2025	1234	1	retour
23/06/2025	2345	0	emprunt
23/06/2025	2345	0	retour
23/06/2025	1	2	emprunt
23/06/2025	1	2	retour

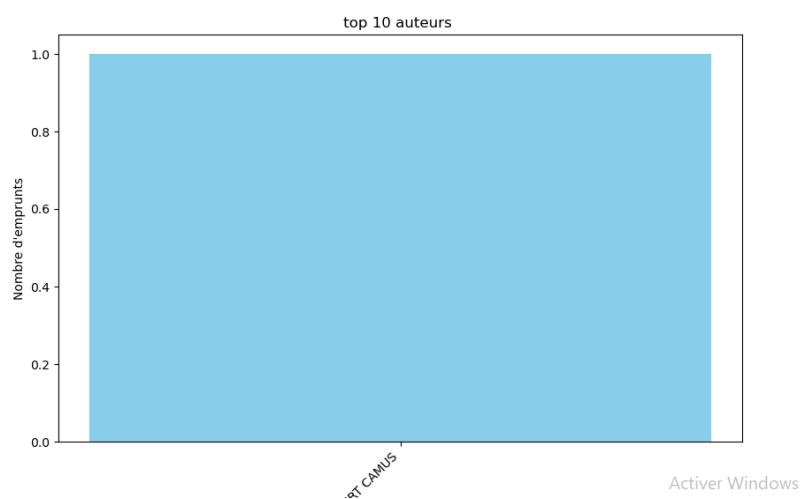
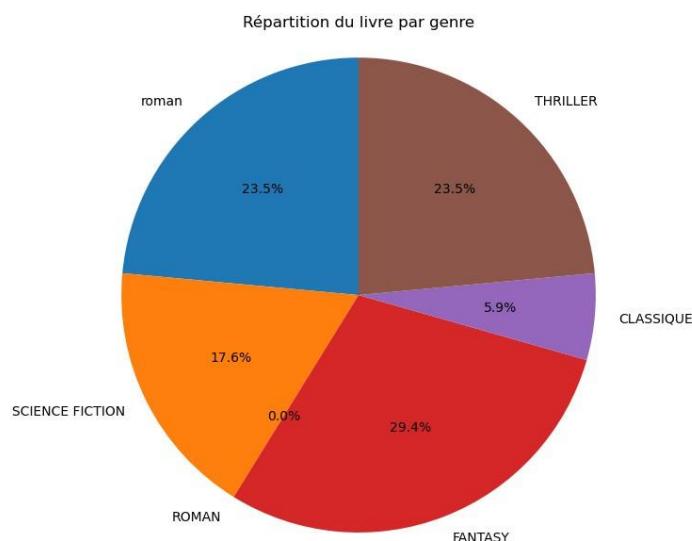
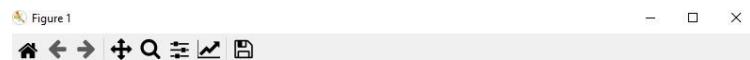
3.4 Analyses Statistiques

DOUAE BENACHOUR

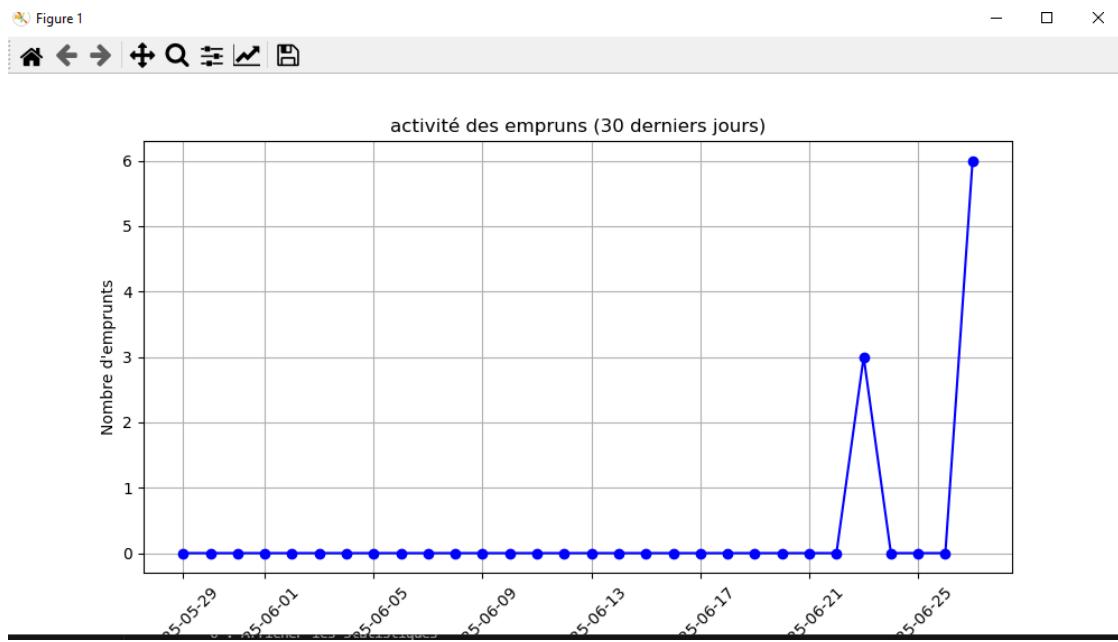
Le module de statistiques génère des visualisations professionnelles :

- Répartition des livres par genre (diagramme circulaire)
- Popularité des auteurs (histogramme)
- Activité récente (graphique temporel)

Ces visualisations aident à comprendre l'utilisation de la bibliothèque et à prendre des décisions éclairées.



DOUAE BENACHOUR



4-défis Techniques et Solutions :

Le développement a rencontré plusieurs défis techniques. L'optimisation des performances a été atteinte grâce à des structures de données adaptées (dictionnaires) permettant des accès en temps constant.

La sérialisation des données a nécessité une attention particulière pour garantir la compatibilité entre les différentes versions de l'application. Des tests rigoureux ont été mis en place pour valider l'intégrité des données après chaque opération de sauvegarde ou de chargement.

5 Bilan du Projet

L'application répond à tous les objectifs initiaux avec une implémentation robuste et bien structurée. Les choix techniques (architecture modulaire, exceptions personnalisées, séparation des interfaces) ont prouvé leur efficacité durant les tests.