

Classes

Livre

C'est la classe de base permettant d'accéder aux informations du fichier. S'initialise avec un fichier. Renvoie une `FormatError` si le fichier ne peut pas être traité (différent de `.pdf` et `.epub`)

attributs :

- `path` → objet path-like Chemin qui mène vers le fichier
- `format` → type str. Extension du fichier
- `nomFichier` → type str. Nom complet du fichier
- `toc` → type dict. Si le fichier est un pdf ou un epub, ceci est sa table des matières sous la forme `{page : chapitre, ...}`
- `titre` → type str.
- `Auteur` → type str
- `langue` → type str

méthodes :

- `print_toc(arg – chemin : type str, option - epub : type bool*)` → Crée les différentes tables des matières dans le dossier spécifié par chemin.
*l'epub est en option car sur certaines distributions le module utilisé crée un conflit.
- `del_toc(arg – chemin : type str)` → supprime les tables des matières dans le dossier spécifié par chemin.

Bibliothèque

Classe représentant l'état de la bibliothèque. Contient les informations nécessaires à la liste d'ouvrages et d'auteurs. Il sera nécessaire de créer une instance manuellement si il y a une bibliothèque existante.

attributs :

- `ouvrages` → type dict. Dictionnaire des différents fichiers qui seront gérés. De la forme `{objet Livre : [titre, auteur, langue, fichier], ...}`
- `auteurs` → type dict. Dictionnaire des différents auteurs de la bibliothèque. De la forme `{auteur : [(titre, fichier), ...], ...}`

méthodes :

- `add(arg – chemin, option – log : type bool)` → Ajoute le fichier spécifié par chemin dans la bibliothèque. Log permet de savoir si l'opération sera répertoriée dans le fichier de log.
- `remove(arg – chemin, option – log : type bool)` → Opération analogue à add
- `is_vide()` → retourne un bool. Si la bibliothèque ne contient rien, renvoie True. False sinon.
- `print_ouvrages_auteurs(arg – dossierRapport : str, option - epub : type bool)` → Crée les deux listes d'ouvrages et d'auteurs dans le dossier spécifié.

Rapport

Classe dont la création d'une instance crée automatiquement les rapports dans le dossier spécifié en entrée. S'initialise avec un dossier (type str) contenant les livres et un autre dossier (type str) indiquant l'emplacement des rapports.

On peut indiquer des options `init` (type bool), `update` (type bool), `ancienneBibli` (type objet Bibliothèque). `init` et `update` ne doivent pas être vrais en même temps. Et si `update` est vrai il faut indiquer une ancienne bibliothèque à comparer.

attributs :

dossierRapport → type str. Fichier où devront être déposés les rapports

bibli → type objet Bibliothèque. Contient la bibliothèque ainsi générée

méthodes :

Config

Classe à initialiser au début du programme pour accéder aux différents dossiers.

attributs :

fichierConfig → type str.

DossierRapports → type str.

DossierBibli → type str.

FichierLogs → type str.

Méthodes :

startlog() → permet la création du fichier de log

Dépendances

pathlib, pymupdf, epub, fpdf, langdetect, configparser, logging, aspose.words