

sharepoint

Modules

[sharepy](#)

Classes

[builtins.object](#)

[Sharepoint](#)

```
class Sharepoint(builtins.object)
    Sharepoint(site_url)
```

Methods defined here:

```
__init__(self, site_url)
    Classe représentant une connexion à un site SharePoint.

    Args:
        site_url (str): L'URL du site SharePoint.

    Attributes:
        site_url (str): L'URL du site SharePoint.
        connection: La connexion établie avec SharePoint.

authenticate(self)
    Authentifie l'utilisateur sur le site SharePoint.

    Cette méthode demande à l'utilisateur de saisir son nom d'utilisateur et son mot de passe,
    puis tente de se connecter au site SharePoint avec les informations fournies.

    Returns:
        object: L'objet de connexion établi avec SharePoint.

    Raises:
        Exception: Si l'email ou le mot de passe est invalide.

    Notes:
        Cette méthode effectue plusieurs tentatives de connexion en cas d'erreur d'authentification,
        et finit par quitter si le nombre maximal de tentatives est atteint.
```

Data descriptors defined here:

```
__dict__
    dictionary for instance variables (if defined)

__weakref__
    list of weak references to the object (if defined)
```

files

Modules

[pandas](#)

Classes

[builtins.object](#)

[File](#)

```
class File(builtins.object)
    File(s, site_url)
```

Methods defined here:

```
__init__(self, s, site_url)
    Initialise une instance de la classe File.

    Args:
        s (object): Session d'authentification pour SharePoint.
        site_url (str): URL du site SharePoint.
```

create_df(self, file_name, file)
Crée un DataFrame à partir d'un fichier CSV ou Excel.

Args:
file_name (str): Nom du fichier.
file (BytesIO): Fichier sous forme de BytesIO.

Returns:
tuple: Liste de DataFrames et liste des noms de feuilles (si le fichier est Excel).

display_folders(self, root_items, folder_name)
Affiche les fichiers CSV trouvés dans un dossier SharePoint et ses sous-dossiers.

Args:
root_items (list): Liste des dossiers racines.
folder_name (str): Nom du dossier recherché.

get_folders(self, folder)
Récupère les dossiers et fichiers dans un dossier SharePoint et ses sous-dossiers.

Args:
folder (str): Chemin relatif du dossier SharePoint.

Returns:
list: Liste des dossiers et fichiers trouvés avec leurs chemins.

rework(self)
Demande à l'utilisateur s'il souhaite analyser un autre fichier.

Returns:
int: 1 si l'utilisateur souhaite analyser un autre fichier, 0 sinon.

select_folders(self, root_items, folder_name)
Sélectionne les dossiers contenant un nom spécifique dans une liste de dossiers.

Args:
root_items (list): Liste des dossiers racines.
folder_name (str): Nom du dossier recherché.

Returns:
list: Liste des chemins des dossiers correspondants.

upload(self, file_input="")
Télécharge un fichier depuis SharePoint.

Args:
file_input (str, optional): Chemin relatif du fichier SharePoint. Si vide, l'utilisateur est invité à entrer le lien.

Returns:
tuple: Nom du fichier téléchargé, fichier sous forme de BytesIO.

Data descriptors defined here:

__dict__
dictionary for instance variables (if defined)

__weakref__
list of weak references to the object (if defined)

analyse

Modules

[os](#) [pandas](#) [matplotlib.pyplot](#) [seaborn](#)

Classes

[builtins.object](#)

[Analyse](#)

class **Analyse**([builtins.object](#))

[Analyse](#)(df)

Methods defined here:

__init__(self, df)
Initialise une instance de la classe [Analyse](#).

Args:
df (DataFrame): Le DataFrame à analyser.

check_column_patterns(self, df, pattern_df)
Cette fonction évalue la qualité de la donnée en fonction de pattern.

Args:

df (DataFrame): Le DataFrame de la donnée
pattern_df (DataFrame): Le DataFrame avec les pattern

Returns:
DataFrame: Les analyses, les valeurs en erreurs.

create_directory(self, file_input)

Cette fonction crée une hiérarchie de dossier s'ils n'existent pas.

Args:
file_input (str): Le chemin à créer

execute(self)

Cette fonction effectue l'analyse en elle-meme.

Elle calcule le nombre de donnée de chaque colonne, les valeurs nulles, les tailles des champs, les types, Top 10, Flop 10...

Returns:
DataFrame: avec l'analyse d'un CSV.

plot(self, result, name)

Cette fonction crée un diagramme en histogramme à partir de résultats de l'évaluation de qualité.

Args:
result (DataFrame): Le DataFrame de la donnée
name (str): chemin et nom du graphique

Returns:
plt: crée un image dans le repertoire name

Data descriptors defined here:

__dict__
dictionary for instance variables (if defined)

__weakref__
list of weak references to the object (if defined)

main

Modules

[pandas](#)

Functions

main

Ce script permet d'effectuer des opérations sur un site SharePoint à l'aide des classes Sharepoint, File et Analyse.

Usage:

- Crée une instance de la classe Sharepoint en spécifiant l'URL du site SharePoint.
- Authentifie l'utilisateur en appelant la méthode authenticate de l'instance Sharepoint.
- Récupère la connexion établie avec SharePoint à partir de l'attribut connection de l'instance Sharepoint.
- Demande à l'utilisateur de saisir le nom du sharepoint.
- Commence une boucle while pour afficher un menu à l'utilisateur.
- Si l'utilisateur choisit "1", le programme effectue des opérations sur des dossiers :
 - Initialise une liste files_error pour stocker les fichiers en erreur.
 - Définit le chemin racine root_path pour les dossiers.
 - Crée une instance de la classe File en utilisant la connexion à SharePoint et l'URL du sharepoint.
 - Récupère les dossiers racines à partir de root_path.
 - Demande à l'utilisateur d'entrer un nom de dossier ou un chemin cible.
 - Sélectionne les dossiers correspondants à partir de root_items en utilisant la méthode select_folders de la classe File.
 - Itère sur chaque fichier dans la liste folder_list.
 - Charge le fichier en utilisant la méthode upload de la classe File.
 - Crée une liste de DataFrame df_list et une liste de noms de feuilles sheet_name en utilisant la méthode create_df de la classe
 - Itère sur chaque DataFrame et son nom de feuille correspondant.
 - Crée une instance de la classe Analyse en utilisant le DataFrame.
 - Exécute l'analyse en utilisant la méthode execute de la classe Analyse.
 - Crée un répertoire pour le fichier d'entrée en utilisant la méthode create_directory de la classe Analyse.
 - Enregistre le résultat de l'analyse dans un fichier CSV.
 - Efface la sortie de la console.
 - Gère les exceptions et ajoute les fichiers en erreur à la liste files_error.
 - Si des fichiers en erreur existent, les enregistre dans un fichier CSV.
- Si l'utilisateur choisit "2", le programme effectue des opérations sur des fichiers :
 - Crée une instance de la classe File en utilisant la connexion à SharePoint et l'URL du sharepoint.
 - Commence une boucle while pour demander à l'utilisateur d'uploader des fichiers.
 - Charge le fichier en utilisant la méthode upload de la classe File.
 - Crée une liste de DataFrame df_list et une liste de noms de feuilles sheet_name en utilisant la méthode create_df de la classe
 - Demande à l'utilisateur s'il souhaite évaluer la qualité des données en utilisant un dictionnaire de données.
 - Itère sur chaque DataFrame et son nom de feuille correspondant.
 - Crée une instance de la classe Analyse en utilisant le DataFrame.
 - Exécute l'analyse en utilisant la méthode execute de la classe Analyse.
 - Crée un répertoire pour le fichier d'entrée en utilisant la méthode create_directory de la classe Analyse.
 - Enregistre le résultat de l'analyse dans un fichier CSV.
 - Si l'évaluation de la qualité des données est demandée, effectue cette évaluation en utilisant la méthode check_column_pattern
 - Enregistre les résultats de l'évaluation et les incohérences dans des fichiers CSV.
 - Affiche un graphique basé sur les résultats de l'évaluation.
- Si l'utilisateur n'entre ni "1" ni "2", affiche un message d'erreur.
- La boucle while continue jusqu'à ce que l'utilisateur choisisse une option valide.