CAHIER DES CHARGES : STAGE

Mots clés

* Archivage automatique
* Batch
* PowerShell
* Script
* Excel CSV
* SWOT
* UTF-8

Contexte

Le but ici est de pouvoir créer une arborescence de fichiers pour étudiants du CESI et en suite pouvoir archiver des documents ou fichiers concernant les étudiants dans cette arborescence.

Fonctionnalités à mettre en place

* Archivage automatique
* Modification des fichiers
* Suppression
* Presse bouton (Facile à utiliser pour le personnel CESI)
* Optimiser

Contraintes

* Langage de développement
* Chemin de stockage
* Lecture des fichiers
* Espaces dans la nomination des dossiers, noms de fichiers et code
* Caractères spéciales (lettres avec accent)

Problématique

Créer une automatisation de fichier facile à utiliser

Hypothèses

* Créer des petits bouts de codes et faire des appels ?
* Utiliser Batch ou PowerShell ?

Pistes de solutions

* J’ai déjà un bout de code qui me permet de créer une arborescence pour chaque dossiers étudiants créer.
* Également j’ai déjà un script qui a pour but de récupérer des fichiers dans un emplacement et de les déposer (Archiver, déplacer ou copier) dans le dépôt (Emplacement) appeler dossier Apprenant. Les fichiers doivent être déposer dans le ou les dossier(s) qui les correspondent.
* Comment rendre l’automatisation, modification, suppression et ajout des fichier facile et simple d’utilisation.
* La commande chcp 65001 >nul pour les caractères spéciaux.

* Syntaxe de nomination des fichiers à respecter « NOM Prenom\_Rubrique\_descripstion du fichier ou document.ext »

Pistes d’améliorations

* A l’intérieur du code insérer les commentaires suivant au début du code
* Description du code
* Version
* Date
* Nom
* Dans le script Arbo\_Etudiants mettre une ligne ou condition qui nous dira si un dossier avec le même nom a déjà été créer.
* Horodatage sur les fichiers archiver
* Avoir une structure de code ce qui me permettre d’avoir plusieurs bouts de code bien organiser.

@echo off

setlocal EnableDelayedExpansion

:: 1. Définir chemins et extensions

:: 2. Vérifier que Etudiants.csv existe

:: 3. Construire la liste temporaire des étudiants

:: 4. Boucle sur chaque fichier .pdf / .doc / .docx

:: Appeler :classer

:: 5. Sous-programme :classer

:: Chercher l'étudiant

:: Extraire la rubrique

:: Construire le chemin de destination

:: Copier le fichier

* Vérifier que tous les dossiers d'apprenants et leurs sous-dossiers existent dans L:\Scripts\Apprenants. (Sinon ➔ prévoir une création avec le script Arbo\_Etudiants)

Problème Rencontrer

ÔØî ├ëtudiant "WAFOJulie\_COMMERCIALISATION\_Contrat.pdf" non reconnu.

ÔØî ├ëtudiant "CLASSOlive\_COMMERCIALISATION.docx" non reconnu.

ÔØî ├ëtudiant "CLASSOlive\_FACTURATION.docx" non reconnu.

ÔØî ├ëtudiant "HOPEStage\_COMMERCIALISATION.docx" non reconnu.

ÔØî ├ëtudiant "HOPEStage\_PEDAGOGIE-IR.docx" non reconnu.

ÔØî ├ëtudiant "WAFOJulie\_COMMERCIALISATION\_Contrat.docx" non reconnu.

ÔØî ├ëtudiant "CLASSOlive\_COMMERCIALISATION.docx" non reconnu.

ÔØî ├ëtudiant "CLASSOlive\_FACTURATION.docx" non reconnu.

ÔØî ├ëtudiant "HOPEStage\_COMMERCIALISATION.docx" non reconnu.

ÔØî ├ëtudiant "HOPEStage\_PEDAGOGIE-IR.docx" non reconnu.

ÔØî ├ëtudiant "WAFOJulie\_COMMERCIALISATION\_Contrat.docx" non reconnu.

­ƒƒó Classement termin├®.

Appuyez sur une touche pour continuer...

Avec tout les essaies et vérifications effectues le problème viens de ma fonction classer

*:: === SOUS-PROGRAMME : CLASSER ===*

:classer

set "file=%~1"

set "full=%~2"

set "nomprenom="

set "rubrique="

for /f %%N in (%tmpList%) do (

    set "etudiant=%%N"

    echo "!file!" | findstr /b /i "!etudiant!" >nul

    if !errorlevel! == 0 (

        set "nomprenom=!etudiant!"

        goto :found

    )

)

* Il faut **éviter de recréer les dossiers déjà créés dans le passé**, même si le script est relancé dans les années futures

@echo off

setlocal enabledelayedexpansion

rem === Dossiers ===

set "base\_dir=L:\Scripts\Apprenants"

set "fichier\_apprenants=apprenants.txt"

set "fichier\_log=dossiers\_crees.log"

rem === Vérifie si le fichier log existe, sinon le créer ===

if not exist "%fichier\_log%" (

type nul > "%fichier\_log%"

)

rem === Lire le fichier apprenants ligne par ligne ===

for /f "tokens=1,2 delims=;" %%A in (%fichier\_apprenants%) do (

set "nomprenom=%%A"

set "formation=%%B"

set "dossier=%nomprenom%\_%formation%"

rem Nettoyer les espaces fin/début

for /f "tokens=\* delims= " %%x in ("!dossier!") do set "dossier=%%x"

rem Vérifier si ce dossier a déjà été créé

findstr /i /c:"!dossier!" "%fichier\_log%" >nul

if errorlevel 1 (

echo Création de !dossier!

rem Créer les dossiers

mkdir "%base\_dir%\!dossier!\PEDAGOGIE\IR" 2>nul

mkdir "%base\_dir%\!dossier!\FACTURATION" 2>nul

rem Ajouter au fichier log

echo !dossier!>>"%fichier\_log%"

) else (

echo Dossier déjà créé : !dossier!

)

)

echo.

echo Fin du traitement.

pause

# RENOMMAGE AUTOMATIQUE DES FICHIERS

Pour réaliser un renommage automatique il va falloir prendre en compte une typologie fixe pour les différentes catégories de fichiers. De tel sorte que le scripte en reconnaissant un fichier puisse savoir comment le renommer

## Problématique

* Comment reconnaitre la catégorie du fichier pour ensuite le renommer ?

## Contraintes

* Moodle
* Imposer une typologie de nommage aux Enseignants comme étudiants, au service scolarité et pédagogique

## Piste de solutions

* Renommer automatiquement les fichiers déposés par les étudiants pour qu’ils suivent une typologie homogène, par exemple :

NomPrenom\_RUBRIQUE\_DESCRIPTION.ext

* Comment reconnaître la catégorie (rubrique) du fichier pour le renommer : par détection sémantique via mots-clés associés

Exemple :

* Si un fichier contient "facture", "paiement", "bordereau" → **FACTURATION**
* S’il contient "rapport", "tp", "exercice" → **TP**
* S’il contient "stage", "entreprise" → **STAGE**

On peut donc créer une table de correspondance mots-clés ↔ rubrique que le script pourra utiliser pour catégoriser automatiquement les fichiers.

Moodle ne contrôle pas bien les noms de fichiers Il faut traiter automatiquement les dépôts après téléchargement

* Une typologie officielle doit être définie et diffusée
* Elle doit inclure : liste des rubriques, nommage type, exemples

Exemple de dictionnaireRubriques.csv

rubrique;motcle

FACTURATION;facture

FACTURATION;paiement

FACTURATION;bordereau

TP;tp

TP;exercice

TP;corrige

STAGE;stage

STAGE;entreprise

PROJET;projet

PROJET;maquette

COMPTERENDU;compte rendu

COMPTERENDU;compterendu

**Extrait du fonctionnement du script**

1. Lire tous les fichiers du dossier source
2. Pour chaque fichier :
   * Tenter de retrouver un nom d’étudiant (via Etudiants.csv)
   * Analyser le nom de fichier pour détecter un mot-clé connu (via dictionnaireRubriques.csv)
   * Si un mot-clé correspond → affecter la rubrique
   * Renommer le fichier selon la norme :

NomPrenom\_RUBRIQUE\_DESCRIPTION.ext

* + Si non reconnu → le déplacer dans un dossier ÀVérifier

rubrique;motcle

COMMERCIALISATION;commercial

COMMERCIALISATION;prospection

COMMERCIALISATION;client

COMMERCIALISATION;vente

COMMERCIALISATION;offre

CONTRACTUALISATION;contrat

CONTRACTUALISATION;signature

CONTRACTUALISATION;avenant

CONTRACTUALISATION;engagement

FACTURATION;facture

FACTURATION;paiement

FACTURATION;bordereau

FACTURATION;reglement

PEDAGOGIE\Activites de Synthes;synthese

PEDAGOGIE\Activites de Synthes;dossier

PEDAGOGIE\Activites de Synthes;travail final

PEDAGOGIE\ADMINISTRATION;inscription

PEDAGOGIE\ADMINISTRATION;dossier administratif

PEDAGOGIE\ADMINISTRATION;carte étudiant

PEDAGOGIE\Engagement Citoyen;citoyen

PEDAGOGIE\Engagement Citoyen;bénévolat

PEDAGOGIE\Engagement Citoyen;associatif

PEDAGOGIE\Engagement Citoyen;engagement

PEDAGOGIE\Entretiens Individuels;entretien

PEDAGOGIE\Entretiens Individuels;suivi personnel

PEDAGOGIE\Entretiens Individuels;rdv

PEDAGOGIE\IR;IR

PEDAGOGIE\IR;intervention rapide

PEDAGOGIE\IR;urgence

PEDAGOGIE\MI;MI

PEDAGOGIE\MI;mission

PEDAGOGIE\MI;intervention

PEDAGOGIE\Suivi Entreprise;entreprise

PEDAGOGIE\Suivi Entreprise;tuteur

PEDAGOGIE\Suivi Entreprise;visite

PEDAGOGIE\Suivi Entreprise;suivi pro

## Problématique

A chaque fois qu’un fichier est renommé il change d’extension. C’est-à-dire l’extension initiale du fichier (PDF, doc, …) devient un fichier texte.

# Gestion des promotions

La règle ici est la même que pour les dossiers et fichiers apprenant ;

* Création d’une arborescence pour les dossiers apprenant chaque début d’année scolaire.
* Transférer des fichiers dans leur destination de dossier respective.
* Archiver les dossiers des promotions plus à jour.

Et il y a des détails qui s’appliquent également dans ce cas.

# Interface pour les utilisateurs

Avec des toute les fonctionnalités des dossiers promotions et apprenants. L’utilisation du langage python pour la réalisation de l’interface, on parlera alors de TK Interface(tkinter).

## Problématique

Le code python n’arrive pas à reconnaitre le chemin des fichiers CSV, ce qui empêche le bonne fonctionnement et la bonne exécution des script pourtant quand on exécute les scripts ils fonctionnent bien.