

```

#!/usr/bin/env python3.9
# -- coding: utf-8 --
from tkinter import *
from tkinter import messagebox
import pickle
import os.path
import smbclient
import subprocess
import unicodecode # Doit être importée
# Lancer le diaporama avec le second programme en Python
def Lancer_Diaporama():
    try:
        exec(open("./Diapo-py.py").read())
    except:
        print("Impossible de lancer le diaporama.")
# Enregistrer la configuration dans le fichier diaporamaini
def Enregistrer():
    global serveursmb
    global partagesmb
    global utilisateur
    global mot_de_passe
    global domaine
    global fichier_diaporama
    serveursmb = Serveur_Entry.get()
    partagesmb = Partage_Entry.get()
    utilisateur = Utilisateur_Entry.get()
    mot_de_passe = Mot_de_passe_Entry.get()
    domaine = Domaine_Entry.get()
    fichier_diaporama = Fichier_du_diaporama_Entry.get()
    # Enregistrer mes variables sous forme d'une liste dans un fichier
    if serveursmb != "vide" and partagesmb != "vide" and utilisateur != "vide"
and mot_de_passe != "vide" and domaine != "vide" and fichier_diaporama !=
"vide":
        variables = [serveursmb, partagesmb, utilisateur, mot_de_passe, domaine,
fichier_diaporama]
        fichierSauvegarde = open("diaporamaini","wb")
        pickle.dump(variables, fichierSauvegarde)
        fichierSauvegarde.close()
        Mon_bouton_Lister_Fichiers['state'] = NORMAL
        Mon_bouton_Lancer_Diaporama['state'] = NORMAL
        menufichier.entryconfigure(0,state=NORMAL)
        menudiaporama.entryconfigure(0,state=NORMAL)
# Lister les fichiers présents dans le partage réseau en utilisant les
paramètres enregistrés dans le fichier diaporamaini
def Lister():
    Lire_Ini()
    global smb
    # Vider la liste de fichiers si elle n'est pas vide
    if Liste_Fichiers.size() != 0:
        Liste_Fichiers.delete(0,END)
    print("Lister les fichiers du partage Samba")
    smb = smbclient.SambaClient(server=serveursmb, share=partagesmb,
username=utilisateur, password=mot_de_passe, domain=domaine)
    dirs = smb.listdir("/")
    i = 0
    while i < len(dirs):
        print (dirs[i])
        Liste_Fichiers.insert(i+1, dirs[i])
        i += 1
    Mon_bouton_Video['state'] = NORMAL
    menuvideo.entryconfigure(0,state=NORMAL)
# Lire les informations contenues dans le fichier diaporamaini
def Lire_Ini():
    global serveursmb

```

```

global partagesmb
global utilisateur
global mot_de_passe
global domaine
global fichier_diaporama
# Lecture du fichier ini
fichierini = "diaporamaini"
# Ouverture du fichier s'il existe et récupération de la liste
if os.path.isfile(fichierini):
    fichierSauvegarde = open(fichierini,"rb")
    variables = pickle.load(fichierSauvegarde)
    fichierSauvegarde.close()
    # Affichage d'informations
    print("Initialisation des variables depuis le fichier ini")
    # Récupération des données dans les variables
    serveursmb = variables[0]
    partagesmb = variables[1]
    utilisateur = variables[2]
    mot_de_passe = variables[3]
    domaine = variables[4]
    fichier_diaporama = variables[5]
else:
    # Le fichier n'existe pas
    print("Fichier " + fichierini + " non trouvé")
    serveursmb = partagesmb = utilisateur = mot_de_passe = domaine =
fichier_diaporama = "vide"
# Transférer et Lire le fichier vidéo
def Video():
    Fichier_video = "vide"
    i=Liste_Fichiers.curselection() ## Récupération de l'index de l'élément
sélectionné
    try:
        Fichier_video = Liste_Fichiers.get(i)
    except:
        print("Sélectionnez un fichier vidéo")
        messagebox.showinfo("Attention !", "Sélectionnez un fichier
vidéo .mp4 .avi .MOV.")
    # Extension du fichier dans la variable ext
    _, ext = os.path.splitext(Fichier_video)
    Extensions_Liste = ['.mp4', '.avi', '.MOV']
    #if Fichier_video != "vide" and ext == ".mp4" or ext == ".avi":
    if Fichier_video != "vide" and ext in Extensions_Liste:
        print("Fichier vidéo : "+Fichier_video)
        print("Extension du fichier : "+ext)
        local_path = "./" + Fichier_video
        smb.download(Fichier_video, local_path)
        print("Fichier vidéo téléchargé")
        # ajout dominique
        nom_de_fichier_avant = Fichier_video
        print ("Nom de fichier avant : " + nom_de_fichier_avant)
        nom_de_fichier_apres = nom_de_fichier(nom_de_fichier_avant)
        print("Nom de fichier après : " + nom_de_fichier_apres)
        if nom_de_fichier_avant != nom_de_fichier_apres:
            print("Le nom de fichier a été modifié")
            print("Le fichier doit être renommé")
            os.rename(nom_de_fichier_avant, nom_de_fichier_apres)
            Fichier_video = nom_de_fichier_apres
        else:
            print("Le nom de fichier n'a pas été modifié")
            print(Fichier_video)
        # ajout dominique
        subprocess.Popen(['/bin/sh', '-c', "mplayer " + Fichier_video + " -loop
0 -fs"])
    else:

```

```

        print("Sélectionnez un fichier vidéo")
def Aide():
    messagebox.showinfo("À propos", "Programme Dominique Renaudeau - Collège
Celles et Melle 2023.")
def nom_de_fichier(nom):
    # Supprimer les espaces et les caractères accentués du nom du fichier
    # Remplacement des espaces
    nom = nom.replace(" ", "_")
    # suppression les caractères accentués
    nom = unidecode.unidecode(nom)
    return nom

os.chdir(os.path.dirname(__file__)) # Changer le répertoire courant par celui du
programme
Lire_Ini() # lire le fichier ini
Fenetre = Tk() # Création de la fenêtre, avec un nom de mon choix Fenetre
# Création des menus
menubar = Menu(Fenetre)
menufichier = Menu(menubar,tearoff=0)
menubar.add_cascade(label="Fichier", menu=menufichier)
menufichier.add_command(label="Lister",state = DISABLED,command=Lister)
menufichier.add_command(label="Enregistrer", command=Enregistrer)
menufichier.add_separator()
menufichier.add_command(label="Quitter", command=Fenetre.destroy)
menudiaporama = Menu(menubar,tearoff=0)
menubar.add_cascade(label="Diaporama", menu=menudiaporama)
menudiaporama.add_command(label="Lancer",state =
DISABLED,command=Lancer_Diaporama)
menuvideo = Menu(menubar,tearoff=0)
menubar.add_cascade(label="Vidéo", menu=menuvideo)
menuvideo.add_command(label="Transférer et lancer la vidéo",state =
DISABLED,command=Video)
menuaide = Menu(menubar,tearoff=0)
menubar.add_cascade(label="Aide", menu=menuaide)
menuaide.add_command(label="À propos",command=Aide)
Fenetre.config(menu=menubar)
# Paramètres de la fenêtre
Fenetre.title('Paramètres Diaporama et vidéo - Programme : D Renaudeau') #Titre
de la fenêtre
Fenetre.geometry("500x400") # On définit la taille de la fenêtre
Fenetre.resizable(width=False,height=False) #Empêcher de redimensionner la
fenêtre
# Création d'un interface graphique avec grid
Serveur_Label= Label(Fenetre, text = 'Votre serveur - Nom ou IP :') #Un label
pour afficher du texte
Serveur_Label.grid(row=0)
Serveur_Entry= Entry(Fenetre) # On définit l'objet Entry (zone de saisie) qui
porte le nom Serveur_Entry
Serveur_Entry.insert(0, serveursmb)
Serveur_Entry.grid(row=0,column=1,ipadx=30) # ipadx=30 pour créer 10 pixels de
plus horizontalement
Partage_Label= Label(Fenetre, text = 'Votre partage réseau :')
Partage_Label.grid(row=1,column=0)
Partage_Entry= Entry(Fenetre)
Partage_Entry.insert(0, partagesmb)
Partage_Entry.grid(row=1,column=1,ipadx=30)
Utilisateur_Label= Label(Fenetre, text = 'Votre utilisateur :')
Utilisateur_Label.grid(row=2,column=0)
Utilisateur_Entry= Entry(Fenetre)
Utilisateur_Entry.insert(0, utilisateur)
Utilisateur_Entry.grid(row=2,column=1,ipadx=30)
Mot_de_passe_Label= Label(Fenetre, text = 'Votre mot de passe :')
Mot_de_passe_Label.grid(row=3,column=0)
Mot_de_passe_Entry= Entry(Fenetre,show="*") # Saisie masquée du mot de passe

```

```

Mot_de_passe_Entry.insert(0, mot_de_passe)
Mot_de_passe_Entry.grid(row=3,column=1,ipadx=30)
Domaine_Label= Label(Fenetre, text = 'Votre nom de domaine:')
Domaine_Label.grid(row=4,column=0)
Domaine_Entry= Entry(Fenetre)
Domaine_Entry.insert(0, domaine)
Domaine_Entry.grid(row=4,column=1,ipadx=30)
Fichier_du_diaporama_Label= Label(Fenetre, text = 'Votre fichier de
diaporama :')
Fichier_du_diaporama_Label.grid(row=5,column=0)
Fichier_du_diaporama_Entry= Entry(Fenetre)
Fichier_du_diaporama_Entry.insert(0, fichier_diaporama)
Fichier_du_diaporama_Entry.grid(row=5,column=1,ipadx=30)
Mon_bouton_Lancer_Diaporama = Button(Fenetre, text = 'Lancer le Diaporama',
state = DISABLED, command = Lancer_Diaporama)
Mon_bouton_Lancer_Diaporama.grid(row=6,column=0,ipadx=0)
Mon_bouton_Enregistrer = Button(Fenetre, text = ' Enregistrer le fichier ini',
command = Enregistrer)
Mon_bouton_Enregistrer.grid(row=6,column=1,ipadx=0)
Mon_bouton_Video = Button(Fenetre, text = ' Transférer et lancer la vidéo',
state = DISABLED, command = Video)
Mon_bouton_Video.grid(row=8,column=0,ipadx=0)
Liste_Fichiers = Listbox(Fenetre)
Liste_Fichiers.grid(row=7,column=1,ipadx=30)
Mon_bouton_Lister_Fichiers = Button(Fenetre, text = ' Lister les fichiers du
partage depuis ini', state = DISABLED, command = Lister)
Mon_bouton_Lister_Fichiers.grid(row=8,column=1,ipadx=0)
if serveursmb != "vide":
    Mon_bouton_Lister_Fichiers['state'] = NORMAL
    Mon_bouton_Lancer_Diaporama['state'] = NORMAL
    menufichier.entryconfigure(0,state=NORMAL)
    menudiaporama.entryconfigure(0,state=NORMAL)
Fenetre.mainloop() # lance la boucle principale

```