

```

# --- INSTALLATION DES OUTILS ---
!pip install pikepdf -q

import os
import pikepdf
from google.colab import drive
from datetime import datetime

# --- 1. CONNEXION AU DRIVE ---
if not os.path.exists('/content/drive'):
    drive.mount('/content/drive')

# --- 2. CONFIGURATION ---
nom_dossier_source = "A_Fusionner"
nom_fichier_final = "Document_Assemble.pdf"

sauver_dans_dossier_source = True
type_de_tri = "interne"

dossier_source = f"/content/drive/MyDrive/{nom_dossier_source}"
chemin_final = os.path.join(dossier_source, nom_fichier_final) if
sauver_dans_dossier_source else f"/content/drive/MyDrive/{nom_fichier_final}"

# --- 3. FONCTION DE LECTURE DES MÉTADONNÉES ---
def obtenir_date_tri(chemin):
    try:
        with pikepdf.open(chemin) as pdf:
            meta = pdf.docinfo
            if '/CreationDate' in meta:
                return str(meta['/CreationDate'])
    except:
        pass
    return str(os.path.getmtime(chemin))

# --- 4. EXÉCUTION DU TRAITEMENT ---
print(f"🚀 Analyse du dossier '{nom_dossier_source}'...")

if not os.path.exists(dossier_source):
    print(f"❌ Erreur : Dossier introuvable.")
else:
    # FILTRE : On ignore le fichier final déjà présent pour ne pas fusionner le
    résultat avec les sources
    fichiers = [f for f in os.listdir(dossier_source) if f.lower().endswith('.pdf') and
f != nom_fichier_final]

    if not fichiers:
        print("⚠️ Aucun PDF trouvé dans le dossier.")
    else:
        try:
            if type_de_tri == "interne":
                print("🔍 Lecture de la mémoire interne des PDF (Date de création)...")
                fichiers.sort(key=lambda x:
obtenir_date_tri(os.path.join(dossier_source, x)))
            elif type_de_tri == "date":
                print("📅 Tri par date de fichier (Google Drive) activé.")
                fichiers.sort(key=lambda x:
obtenir_date_tri(os.path.join(dossier_source, x)))
            else:
                print("🔤 Tri alphabétique (par nom) activé.")
                fichiers.sort()
        except Exception as e:
            print(f"⚠️ Souci lors du tri ({e}).")
            fichiers.sort()

```

```
print(f"📄 Fusion de {len(fichiers)} fichiers en cours...")
nouveau_pdf = pikepdf.Pdf.new()
fichiers_ajoutes = 0

for f in fichiers:
    chemin_pdf = os.path.join(dossier_source, f)
    try:
        with pikepdf.open(chemin_pdf) as pdf_a_ajouter:
            nouveau_pdf.pages.extend(pdf_a_ajouter.pages)
            print(f"✅ Ajouté : {f}")
            fichiers_ajoutes += 1
    except Exception as e:
        print(f"❌ Impossible de lire {f} (Fichier endommagé).")

if fichiers_ajoutes > 0:
    nouveau_pdf.save(chemin_final, linearize=True)
    print(f"\n🌟 OPÉRATION RÉUSSIE !")
    print(f"📍 Emplacement : {chemin_final.replace('/content/drive/MyDrive/',
'Mon Drive / ')}")
```