https://docs.google.com/document/d/1NbIsYdL9apD8kkINfMadSDXwMih4sMArZXKlJJ4YGhM/edit?usp=sharing

#Observer opening

event\_handler = MyHandler()

observer = Observer()

observer.schedule(event\_handler, source\_folder, recursive=False)

observer.start()

passed\_files = []

failed\_files = []

#Fuzzy

matches\_test = process.extract(descriptor, my\_dict.keys(), limit=2)

matches = ", ".join([match[0] for match in matches])

matches\_label.config(text=matches)

#filename/ filepath dictionary check

class MyHandler(FileSystemEventHandler):

def on\_modified(self, event):

for file\_name in os.listdir(source\_folder):

file\_path = os.path.join(source\_folder, file\_name)

basename, extension = os.path.splitext(file\_name)

descriptor = basename[-7:]

if descriptor in my\_dict:

path = my\_dict[descriptor]

destination\_path = os.path.join(path, file\_name)

passed\_files.append((file\_path, destination\_path))

else:

failed\_files.append(file\_name)

for folder\_name in os.listdir(source\_folder):

folder\_path = os.path.join(source\_folder, folder\_name)

folder\_tail = folde\_name[-7:]

If folder\_tail in my\_dict:

path = my\_dict[descriptor]

destination = os.path.join(path, folder\_name)

passed\_folders.append((folder\_path, destination\_path))

Else:

failed\_folders.append(file\_name)

Def handle overwrite

#Fuzzy

For passed\_files in os.listdir(source\_folder)

If passed\_files not in os.listdir(destination\_folder)

shutil.move(file\_path, destination)

For passed\_folders os.listdir(source\_folder)

If passed\_folder not in os.lstdir(destination\_folder)

shutil.move(folder\_path, destination)

Else

#Import modules

import os

import shutil

import json

from tkinter import \*

from tkinter import Frame, Tk, Label, scrolledtext, BOTH, END

import fuzzywuzzy

from fuzzywuzzy import process

from watchdog.observers import Observer

import logging

from watchdog.events import FileSystemEventHandler

import time

#Log creation

logging.basicConfig(level=logging.INFO,

#import files

with open("/Users/Kyle/Applications/OrganizingApp/Assets/dict.json", "r") as json\_file:

my\_dict = json.load(json\_file)

source\_folder = '/Users/Kyle/Applications/OrganizingApp/Assets/Start'

#Window creation

window Tk()

window.configure(bg="#1E1E1E", bd=0)

window.geometry("430x450")

Gui\_Header = Label(window, bd=0, text="Organizer:", font=("Avenir Next", 24),bg="#1E1E1E", fg="white")

Gui\_Header.pack(padx=28, pady=(10,0), anchor="w")

Gui\_Subtitle = Label(window, bd=0, text="Unmoved Files:", font=("Avenir Next", 12),bg="#1E1E1E", fg="#AABBAA")

Gui\_Subtitle.pack(padx=28, pady=(0, 10), anchor="w")

#FolderTree Creation

# Create all the necessary folders

for path in my\_dict.values():

os.makedirs(path, exist\_ok=True)

#Gui with Overwrite and Skip Options

HeyDuplicate\_label = Label(table\_frame, text="A file with the name '{file\_name}' already exists in the destination folder", bd=0, font=("Avenir Next", 14),

bg="#1E1E1E", fg="white", anchor="w")

choice1\_label = Label(table\_frame, text= "Overwrite", bd=0, font=("Avenir Next", 14),

bg="#1E1E1E", fg="white", anchor="w")

choice2\_label = Label(table\_frame, text="Skip", bd=0, font=("Avenir Next", 14),

bg="#1E1E1E", fg="white", anchor="w")

HeyDuplicate\_label.grid(row=0, column=0, padx=(0, 90), pady=0, sticky="w")

choice1.grid(row=0, column=2, padx=(0, 90), pady=0, sticky="w")

overwrite\_button = Button(table\_frame, text="Overwrite", command=handle\_overwrite)

overwrite\_button.grid(row=2, column=2, padx=(0, 90), pady=10, sticky="w")

skip\_button = Button(table\_frame, text="Skip", command=handle\_skip)

skip\_button.grid(row=2, column=3, padx=10, pady=10, sticky="w")

#Fuzzy

For passed\_files in os.listdir(source\_folder)

If passed\_files not in os.listdir(destination\_folder)

shutil.move(file\_path, destination)

For passed\_folders os.listdir(source\_folder)

If passed\_folder not in os.lstdir(destination\_folder)

shutil.move(folder\_path, destination)

Else window tk

#Window creation

window = Tk()

window.configure(bg="#1E1E1E", bd=0)

window.geometry("430x450")

my\_notebook = ttk.Notebook (window)

my\_notebook.pack (pady=15)

my\_frame1 = Frame (my\_notebook, width=430, height=450, bg="#1E1E1E")

my\_frame2 = Frame (my\_notebook, width=430, height=450, bg="#1E1E1E")

my\_frame1.pack(fill="both", expand=1)

my\_frame2.pack(fill="both", expand=1)

my\_notebook.add(my\_frame1, text="Doubles Trouble")

my\_notebook.add(my\_frame2, text="Confidence\_Keys")

Gui\_Header = Label(window, bd=0, text="Organizer:", font=("Avenir Next", 24),bg="#1E1E1E", fg="white")

Gui\_Header.pack(padx=28, pady=(10,0), anchor="w")

Gui\_Subtitle = Label(window, bd=0, text="Unmoved Files:", font=("Avenir Next", 12),bg="#1E1E1E", fg="#AABBAA")

Gui\_Subtitle.pack(padx=28, pady=(0, 10), anchor="w")

# Create a frame for the table

table\_frame1 = Frame(my\_frame1, bg="#1E1E1E")

table\_frame1.pack(padx=28, pady=(10, 0), anchor="w")

table\_frame2 = Frame(my\_frame2, bg="#1E1E1E")

table\_frame2.pack(padx=28, pady=(10, 0), anchor="w")

# Create label widgets for the headers

headers = ["File Name", "", "Options "]

for i, header in enumerate(headers):

header\_label = Label(table\_frame1, text=header, bd=0, font=("Avenir Next", 12), bg="#1E1E1E", fg="white")

if i == 0:

header\_label.grid(row=0, column=i, padx=(0, 70), pady=0, sticky="w")

elif i == len(headers) - 1:

header\_label.grid(row=0, column=i, padx=(0, 0), pady=0, sticky="e")

else:

header\_label.grid(row=0, column=i, padx=10, pady=10)

HeyDuplicate\_label = Label(table\_frame1, text="A file with the name '{file\_name}' already exists in the destination folder", bd=0, font=("Avenir Next", 14),

bg="#1E1E1E", fg="white", anchor="w")

choice1\_label = Label(table\_frame1, text= "Overwrite", bd=0, font=("Avenir Next", 14),

bg="#1E1E1E", fg="white", anchor="w")

choice2\_label = Label(table\_frame1, text="Skip", bd=0, font=("Avenir Next", 14),

bg="#1E1E1E", fg="white", anchor="w")

HeyDuplicate\_label.grid(row=0, column=0, padx=(0, 90), pady=0, sticky="w")

choice1\_label.grid(row=0, column=2, padx=(0, 90), pady=0, sticky="w")

choice2\_label.grid(row=1, column=2, padx=10, pady=0, sticky="w")

overwrite\_button = Button(table\_frame1, text="Overwrite", command=handle\_overwrite)

overwrite\_button.grid(row=2, column=2, padx=(0, 90), pady=10, sticky="w")

skip\_button = Button(table\_frame1, text="Skip", command=handle\_skip)

skip\_button.grid(row=2, column=3, padx=10, pady=10, sticky="w")

def handle\_overwrite():

If choice1

shutil.move(file\_path, destination)

def handle\_skip():

If choice2

break

file\_label = Label(table\_frame, text=file\_name, bd=0, font=("Avenir Next", 14),

bg="#1E1E1E", fg="white", anchor="w")

key2\_label = Label(table\_frame, text="", bd=0, font=("Avenir Next", 14),

bg="#1E1E1E", fg="white", anchor="w")

matches\_label = Label(table\_frame, bd=0, font=("Avenir Next", 14),

bg="#1E1E1E", fg="white", anchor="w")

# Place the labels in the table

file\_label.grid(row=i+1, column=0, padx=(0, 90), pady=0, sticky="w")

key2\_label.grid(row=i+1, column=1, padx=10, pady=0, sticky="w")

matches\_label.grid(row=i+1, column=2, padx=(50,0), pady=(0,0), sticky="w"

#loop gui window

window.mainloop()

#Observer Closing statements

try:

while observer.is\_alive():

time.sleep(1)

except KeyboardInterrupt:

observer.stop()

observer.join()

# Labels for failed files and proximity scores

file\_label = Label(table\_frame, text=file\_name, bd=0, font=("Avenir Next", 14),

bg="#1E1E1E", fg="white", anchor="w")

key2\_label = Label(table\_frame, text="", bd=0, font=("Avenir Next", 14),

bg="#1E1E1E", fg="white", anchor="w")

matches\_label = Label(table\_frame, bd=0, font=("Avenir Next", 14),

bg="#1E1E1E", fg="white", anchor="w")

# Place the labels in the table

file\_label.grid(row=i+1, column=0, padx=(0, 90), pady=0, sticky="w")

key2\_label.grid(row=i+1, column=1, padx=10, pady=0, sticky="w")

matches\_label.grid(row=i+1, column=2, padx=(50,0), pady=(0,0), sticky="w"

#Create Headers

table\_frame = Frame(window, bg="#1E1E1E")

table\_frame.pack(padx=28, pady=(10, 0), anchor="w")

# Create label widgets for the headers

headers = ["File Name", "", "Options "]

for i, header in enumerate(headers):

header\_label = Label(table\_frame, text=header, bd=0, font=("Avenir Next", 12), bg="#1E1E1E", fg="white")

if i == 0:

header\_label.grid(row=0, column=i, padx=(0, 70), pady=0, sticky="w")

elif i == len(headers) - 1:

header\_label.grid(row=0, column=i, padx=(0, 0), pady=0, sticky="e")

else:

header\_label.grid(row=0, column=i, padx=10, pady=10)

#loop gui window

window.mainloop()

#loop gui window

window.mainloop()