Assignment - 10

Aim : Implement file sysy

Theory :

File manager :
The File Manager is a system software responsible for the creation, deletion, modification of the files and managing their access, security and the resources used by them. These functions are performed in collaboration with the Device Manager.

The File Manager has big responsibilities in it’s hands. It is in charge of the physical components of the computer system, information resources and the policies to store and distribute the files. It’s responsibilities include :

File management in an operating system is formally defined as manipulating files in a computer system, which includes creating, modifying, and deleting files. Therefore, file management is one of the simple but crucial features offered by the operating system. The operating system’s file management function entails software that handles or maintains files (binary, text, PDF, docs, audio, video, etc.) included in computer software.

Features of file manager :

1. Providing security to application software and system.
2. Memory management
3. Disk management.
4. I/O operations.
5. File management, etc

Operations of file manager :

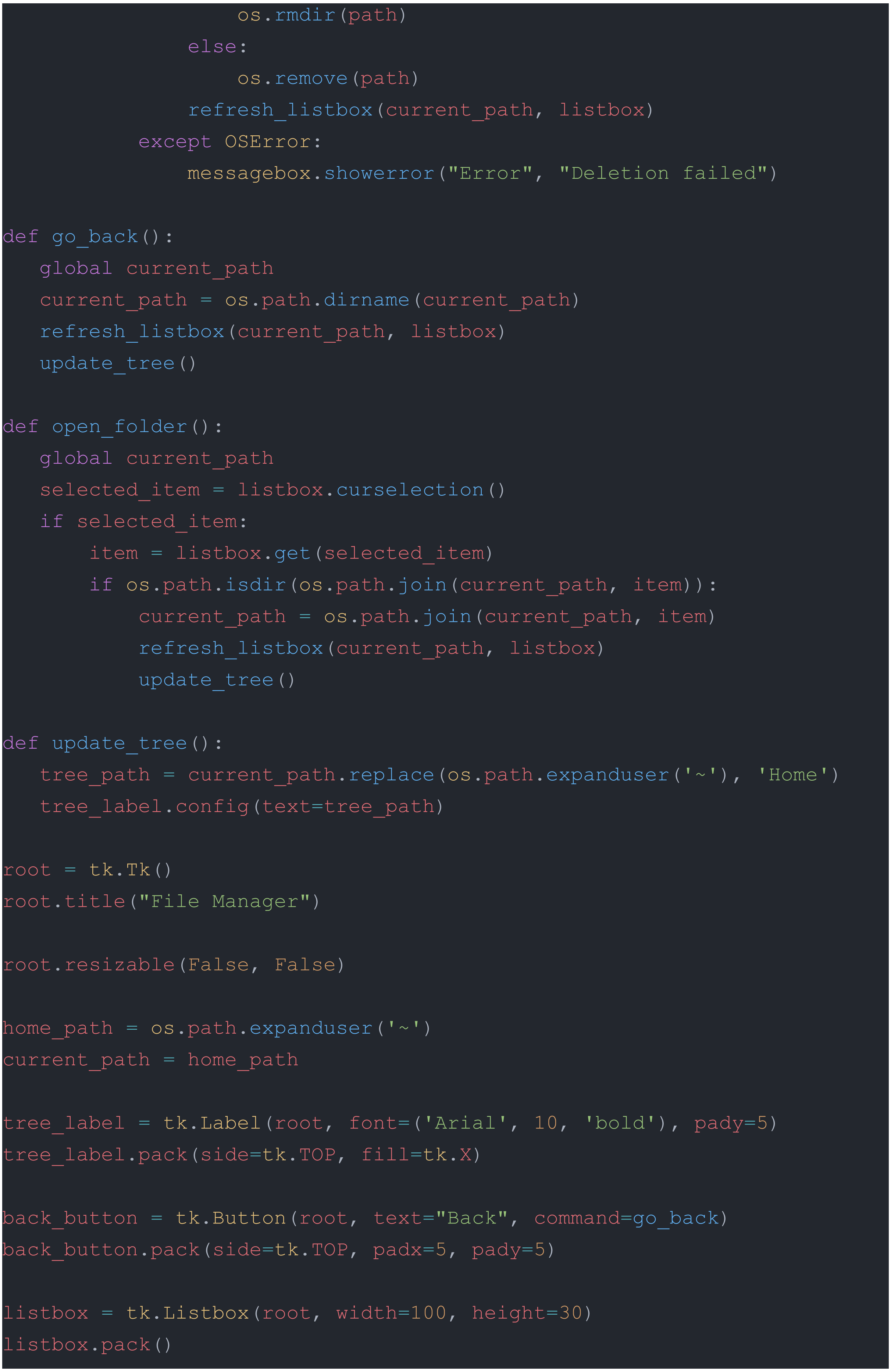
1. File creation
2. File modification
3. File deletion
4. File transfer
5. File renaming
6. File copying and moving
7. Changing file creation

Examples of file browsers :

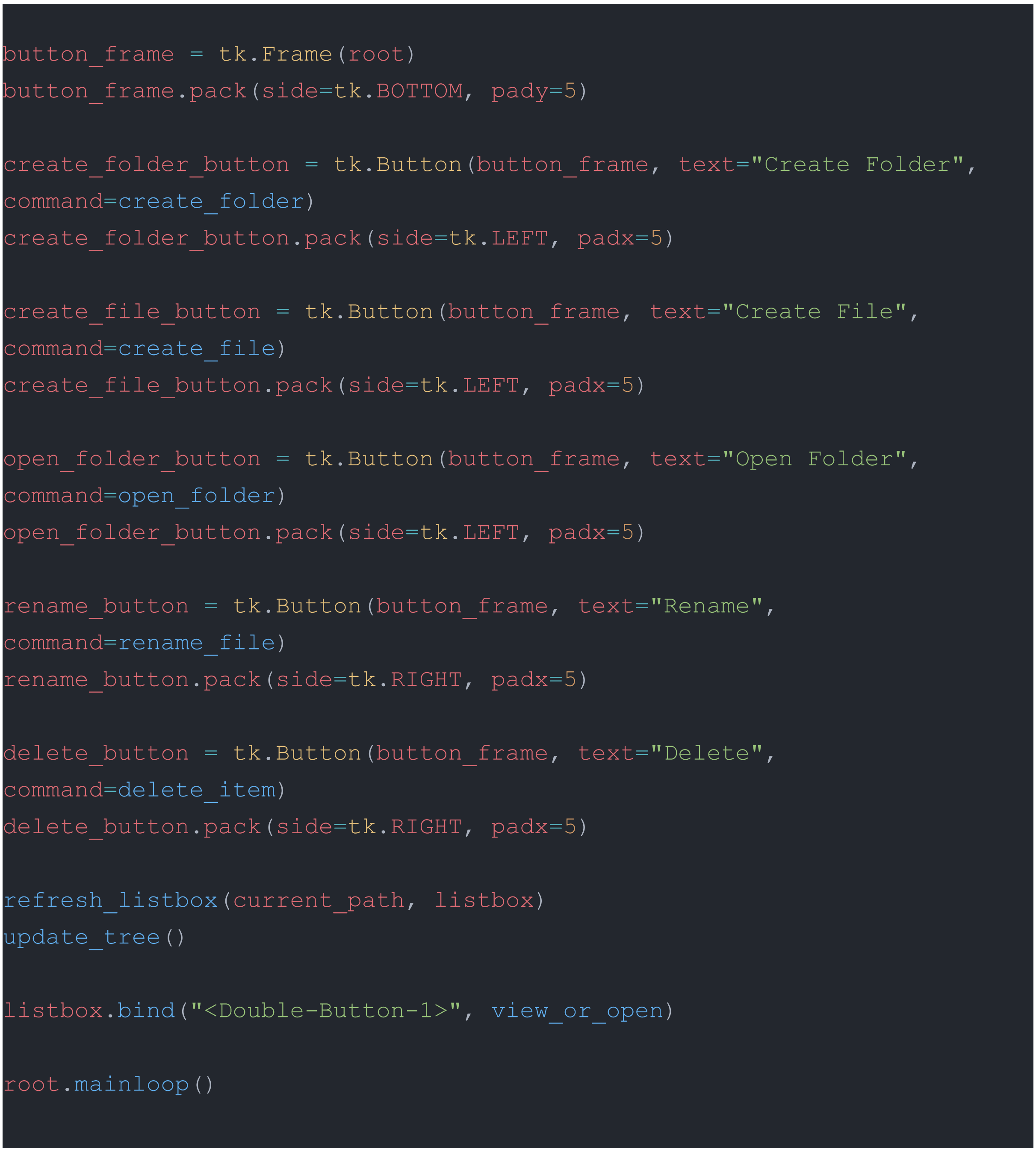
1. Windows file manager (This PC)
2. Finder
3. Dolphin
4. One Drive
5. GNOME Files, etc

Code and Output :

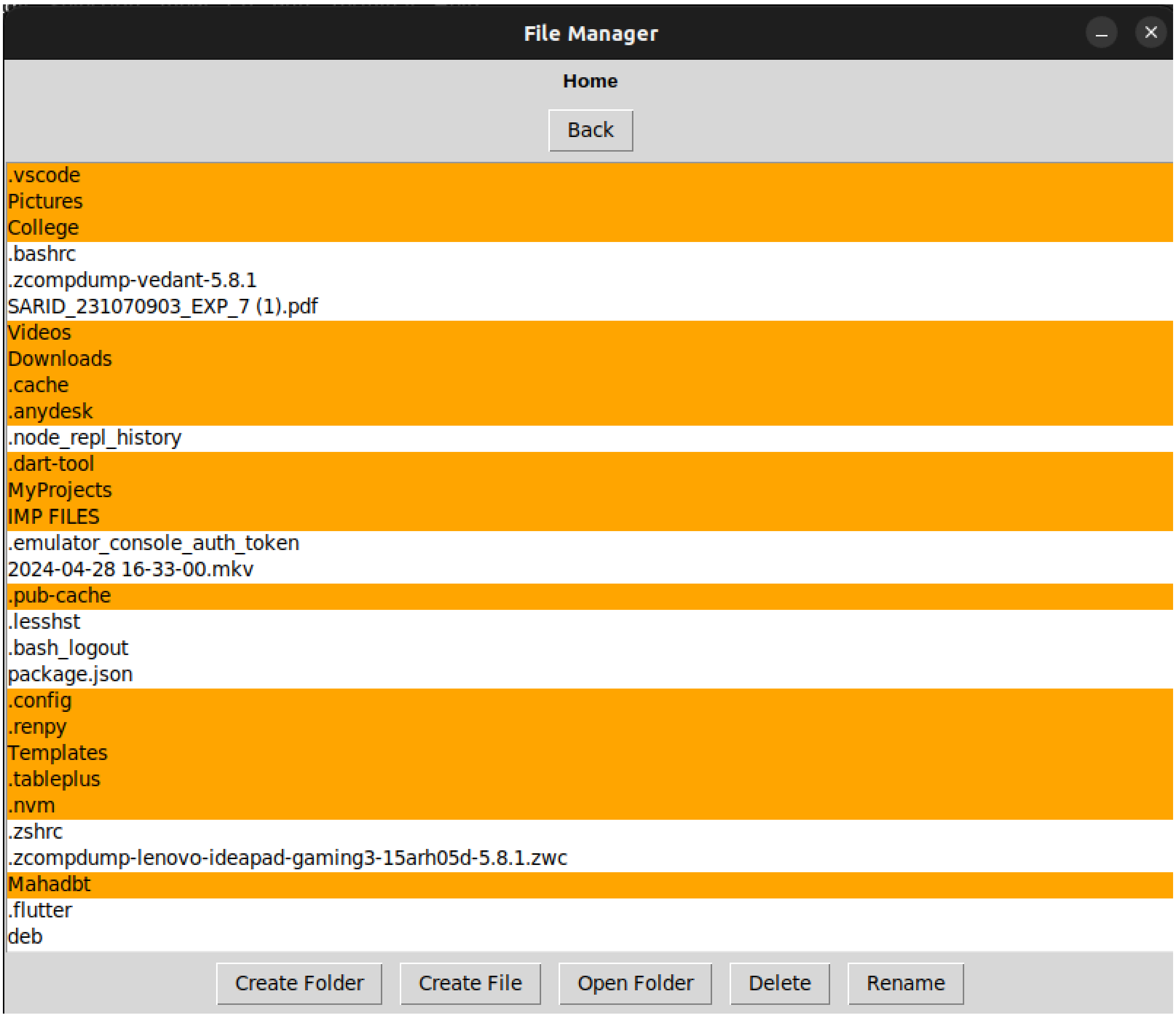
```
import os
import tkinter as tk
from tkinter import messagebox, simpledialog
import subprocess

def refresh_listbox(path, listbox):
   listbox.delete(0, tk.END)
   for item in os.listdir(path):
       listbox.insert(tk.END, item)
   for i in range(listbox.size()):
       item = listbox.get(i)
       if os.path.isdir(os.path.join(path, item)):
           listbox.itemconfig(i, {'bg': 'orange'})

def create_folder():
   folder_name = simpledialog.askstring("Input", "Enter folder name:")
   if folder_name:
       try:
           os.mkdir(os.path.join(current_path, folder_name))
           refresh_listbox(current_path, listbox)
       except OSError:
           messagebox.showerror("Error", "Creation of the directory
failed")

def create_file():
   file_name = simpledialog.askstring("Input", "Enter file name:")
   if file_name:
       try:
           with open(os.path.join(current_path, file_name), 'w') as f:
               pass
           refresh_listbox(current_path, listbox)
       except OSError:
           messagebox.showerror("Error", "Creation of the file failed")

def view_or_open(event):
   selected_item = listbox.curselection()
   if selected_item:
       item = listbox.get(selected_item)
       if os.path.isdir(os.path.join(current_path, item)):
           open_folder()
       else:
           view_file()

def view_file():
   selected_item = listbox.curselection()
   if selected_item:
       item = listbox.get(selected_item)
       file_path = os.path.join(current_path, item)
       if os.path.isfile(file_path):
           try:
               subprocess.Popen(['xdg-open', file_path])
           except:
               messagebox.showerror("Error", "Failed to open file")

def rename_file():
   selected_item = listbox.curselection()
   if selected_item:
       item = listbox.get(selected_item)
       new_name = simpledialog.askstring("Rename", f"Enter new name for
{item}:")
       if new_name:
           try:
               os.rename(os.path.join(current_path, item),
os.path.join(current_path, new_name))
               refresh_listbox(current_path, listbox)
           except OSError:
               messagebox.showerror("Error", "Rename failed")

def delete_item():
   selected_item = listbox.curselection()
   if selected_item:
       item = listbox.get(selected_item)
       confirm = messagebox.askyesno("Delete", f"Are you sure you want
to delete {item}?")
       if confirm:
           try:
               path = os.path.join(current_path, item)
               if os.path.isdir(path):
```



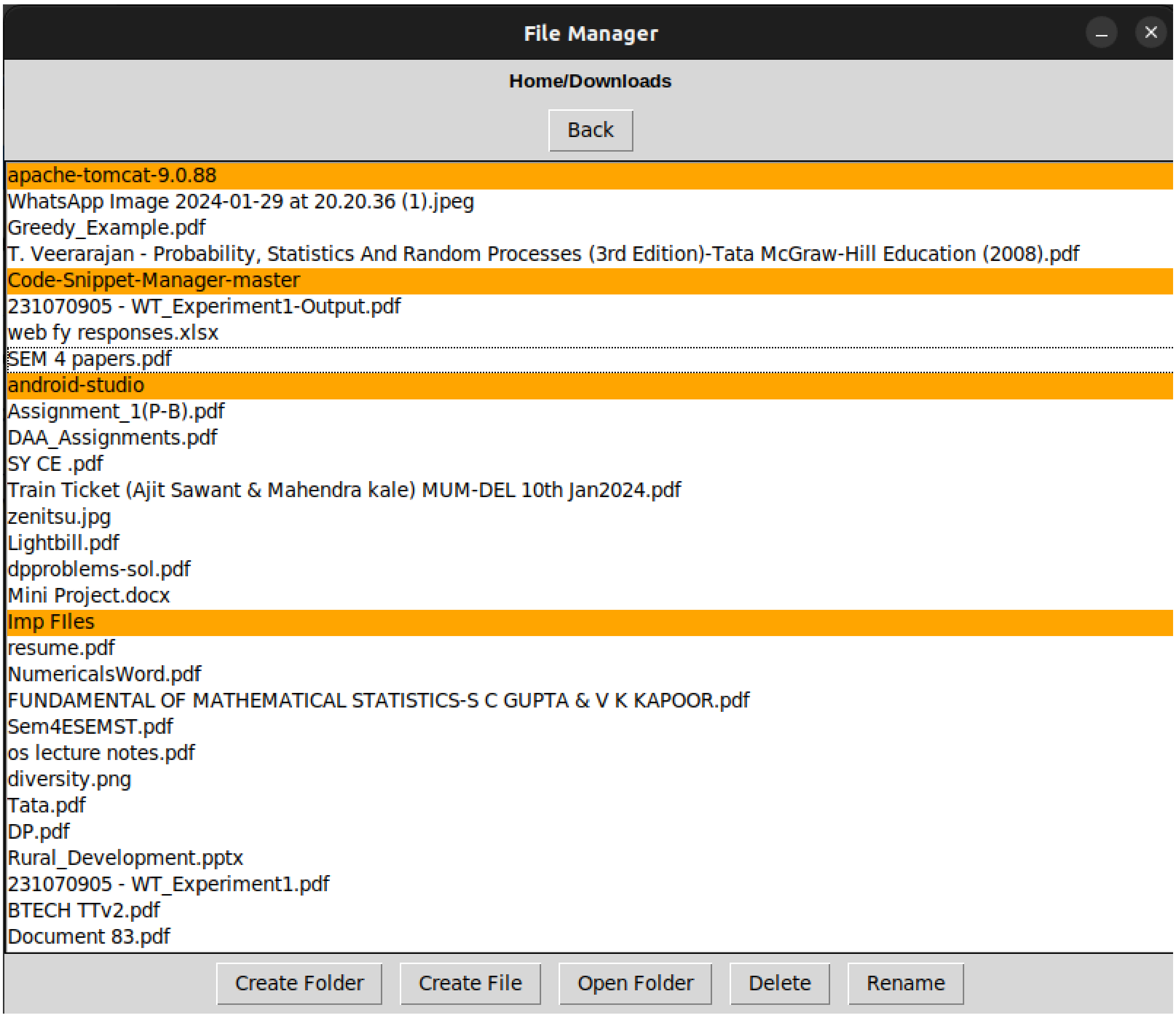
```
                    os.rmdir(path)
                else:
                    os.remove(path)
                refresh_listbox(current_path, listbox)
            except OSError:
                messagebox.showerror("Error", "Deletion failed")

def go_back():
    global current_path
    current_path = os.path.dirname(current_path)
    refresh_listbox(current_path, listbox)
    update_tree()

def open_folder():
    global current_path
    selected_item = listbox.curselection()
    if selected_item:
        item = listbox.get(selected_item)
        if os.path.isdir(os.path.join(current_path, item)):
            current_path = os.path.join(current_path, item)
            refresh_listbox(current_path, listbox)
            update_tree()

def update_tree():
    tree_path = current_path.replace(os.path.expanduser('~'), 'Home')
    tree_label.config(text=tree_path)

root = tk.Tk()
root.title("File Manager")

root.resizable(False, False)

home_path = os.path.expanduser('~')
current_path = home_path

tree_label = tk.Label(root, font=('Arial', 10, 'bold'), pady=5)
tree_label.pack(side=tk.TOP, fill=tk.X)

back_button = tk.Button(root, text="Back", command=go_back)
back_button.pack(side=tk.TOP, padx=5, pady=5)

listbox = tk.Listbox(root, width=100, height=30)
listbox.pack()
```



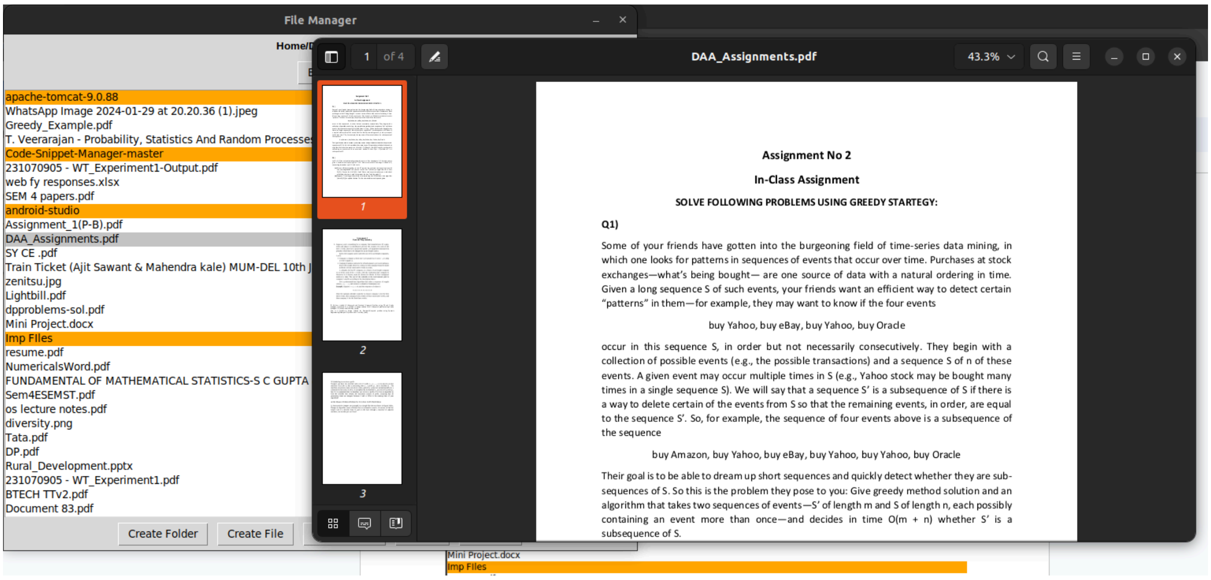
```
button_frame = tk.Frame(root)
button_frame.pack(side=tk.BOTTOM, pady=5)

create_folder_button = tk.Button(button_frame, text="Create Folder",
command=create_folder)
create_folder_button.pack(side=tk.LEFT, padx=5)

create_file_button = tk.Button(button_frame, text="Create File",
command=create_file)
create_file_button.pack(side=tk.LEFT, padx=5)

open_folder_button = tk.Button(button_frame, text="Open Folder",
command=open_folder)
open_folder_button.pack(side=tk.LEFT, padx=5)

rename_button = tk.Button(button_frame, text="Rename",
command=rename_file)
rename_button.pack(side=tk.RIGHT, padx=5)

delete_button = tk.Button(button_frame, text="Delete",
command=delete_item)
delete_button.pack(side=tk.RIGHT, padx=5)

refresh_listbox(current_path, listbox)
update_tree()

listbox.bind("<Double-Button-1>", view_or_open)

root.mainloop()
```



```
.vscode
Pictures
College
.bashrc
.zcompdump-vedant-5.8.1
SARID_231070903_EXP_7 (1).pdf
Videos
Downloads
.cache
.anydesk
.node_repl_history
.dart-tool
MyProjects
IMP FILES
.emulator_console_auth_token
2024-04-28 16-33-00.mkv
.pub-cache
.lesshst
.bash_logout
package.json
.config
.renpy
Templates
.tableplus
.nvm
.zshrc
.zcompdump-lenovo-ideapad-gaming3-15arh05d-5.8.1.zwc
Mahadbt
.flutter
deb
```



```
apache-tomcat-9.0.88
WhatsApp Image 2024-01-29 at 20.20.36 (1).jpeg
Greedy_Example.pdf
T. Veerarajan - Probability, Statistics And Random Processes (3rd Edition)-Tata McGraw-Hill Education (2008).pdf
Code-Snippet-Manager-master
231070905 - WT_Experiment1-Output.pdf
web fy responses.xlsx
SEM 4 papers.pdf
android-studio
Assignment_1(P-B).pdf
DAA_Assignments.pdf
SY CE .pdf
Train Ticket (Ajit Sawant & Mahendra kale) MUM-DEL 10th Jan2024.pdf
zenitsu.jpg
Lightbill.pdf
dpproblems-sol.pdf
Mini Project.docx
Imp FIles
resume.pdf
NumericalsWord.pdf
FUNDAMENTAL OF MATHEMATICAL STATISTICS-S C GUPTA & V K KAPOOR.pdf
Sem4ESEMST.pdf
os lecture notes.pdf
diversity.png
Tata.pdf
DP.pdf
Rural_Development.pptx
231070905 - WT_Experiment1.pdf
BTECH TTv2.pdf
Document 83.pdf
```



apache-tomcat-9.0.88
WhatsApp Image 2024-01-29 at 20.20.36 (1).jpeg
Greedy_Example.pdf
T. Veerarajan - Probability, Statistics And Random Processe
Code-Snippet-Manager-master
231070905 - WT_Experiment1-Output.pdf
web fy responses.xlsx
SEM 4 papers.pdf
android-studio
Assignment_1(P-B).pdf
DAA_Assignments.pdf
SY CE .pdf
Train Ticket (Ajit Sawant & Mahendra kale) MUM-DEL 10th J
zenitsu.jpg
Lightbill.pdf
dpproblems-sol.pdf
Mini Project.docx
Imp FIles
resume.pdf
NumericalsWord.pdf
FUNDAMENTAL OF MATHEMATICAL STATISTICS-S C GUPTA
Sem4ESEMST.pdf
os lecture notes.pdf
diversity.png
Tata.pdf
DP.pdf
Rural_Development.pptx
231070905 - WT_Experiment1.pdf
BTECH TTv2.pdf
Document 83.pdf
Mini Project.docx
Imp Files

Assignment No 2

In-Class Assignment

SOLVE FOLLOWING PROBLEMS USING GREEDY STARTEGY:

Q1)

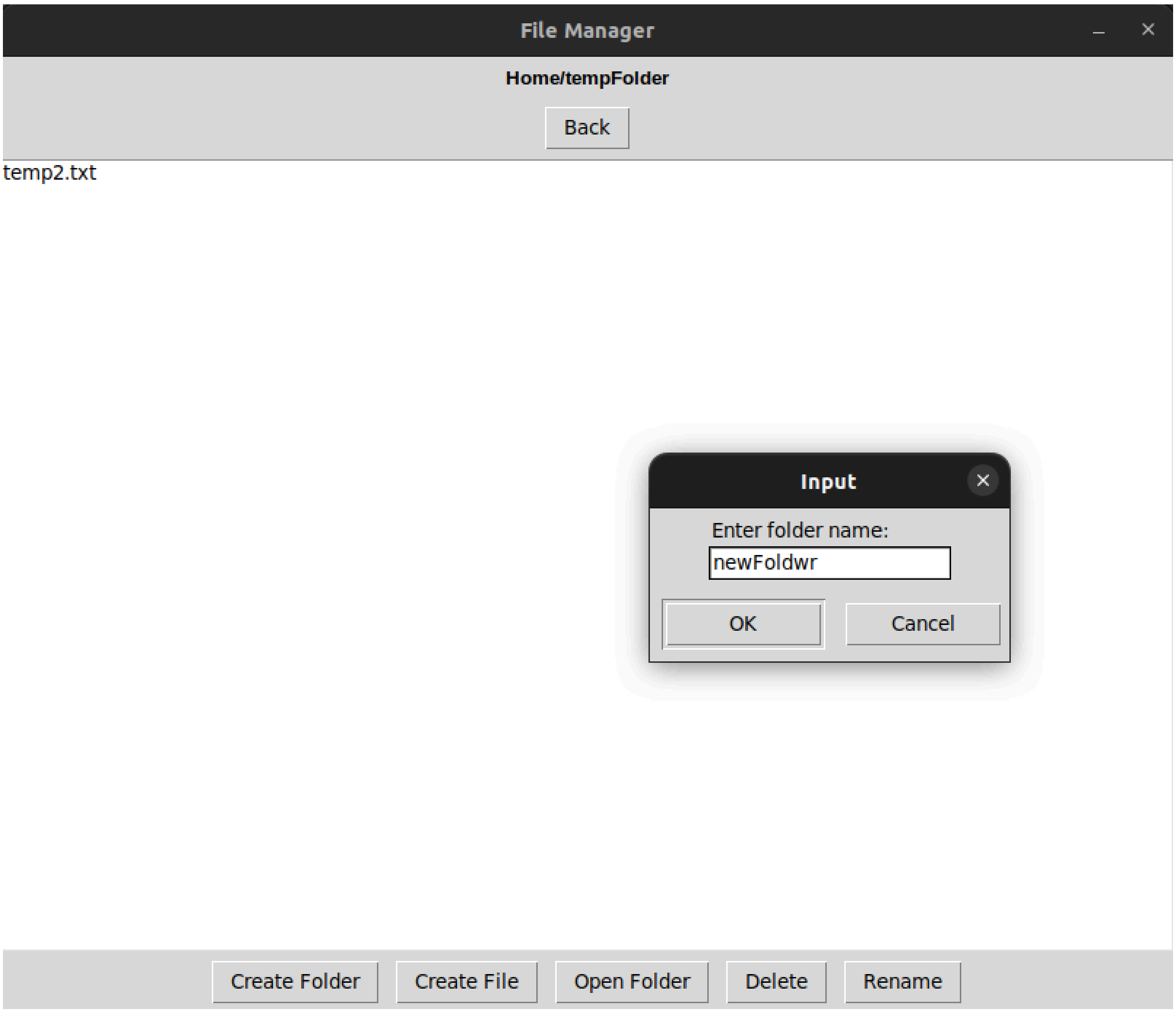
Some of your friends have gotten into the burgeoning field of time-series data mining, in which one looks for patterns in sequences of events that occur over time. Purchases at stock exchanges—what's being bought— are one source of data with a natural ordering in time. Given a long sequence S of such events, your friends want an efficient way to detect certain "patterns" in them—for example, they may want to know if the four events

buy Yahoo, buy eBay, buy Yahoo, buy Oracle

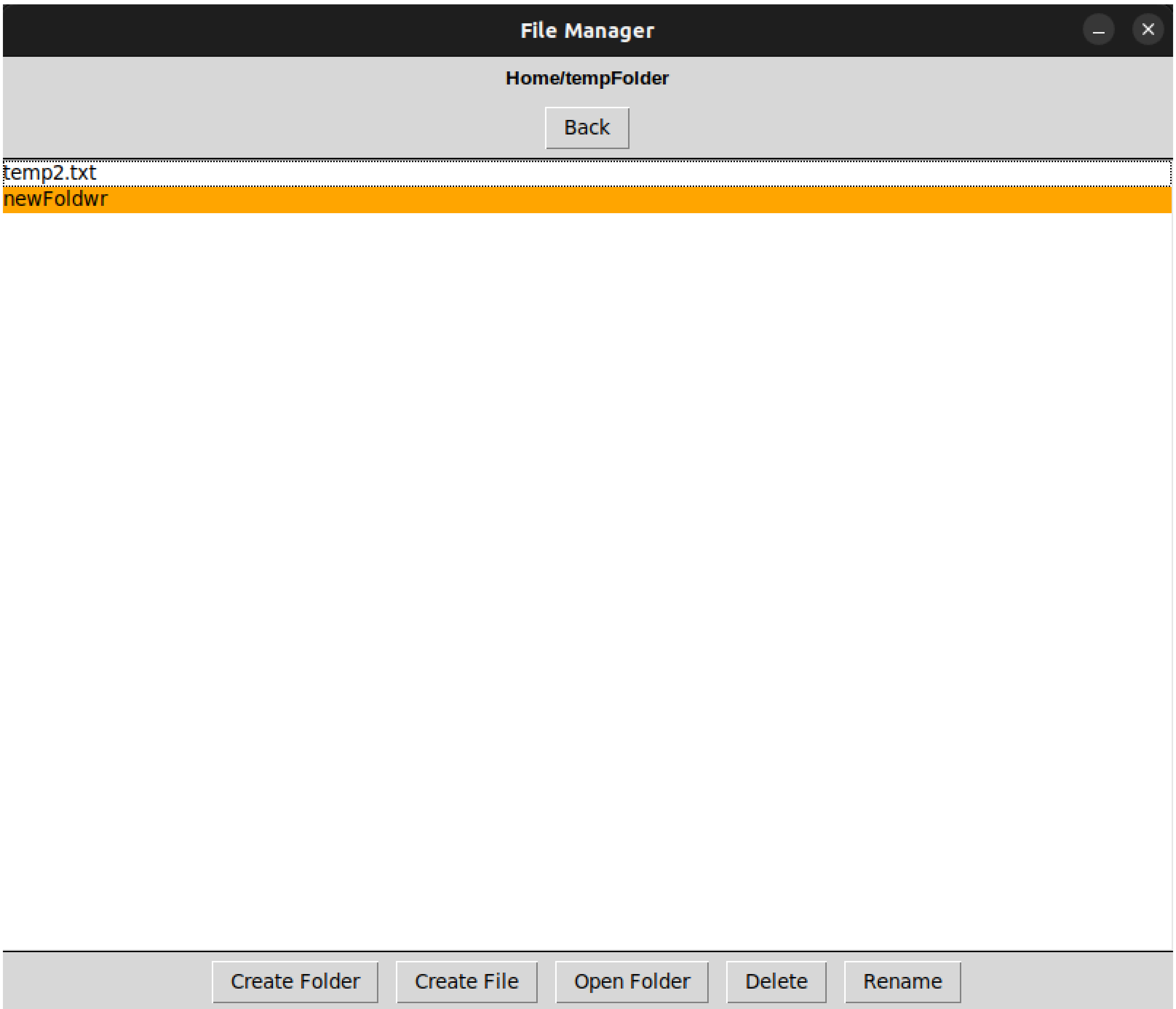
occur in this sequence S, in order but not necessarily consecutively. They begin with a collection of possible events (e.g., the possible transactions) and a sequence S of n of these events. A given event may occur multiple times in S (e.g., Yahoo stock may be bought many times in a single sequence S). We will say that a sequence S' is a subsequence of S if there is a way to delete certain of the events from S so that the remaining events, in order, are equal to the sequence S'. So, for example, the sequence of four events above is a subsequence of the sequence

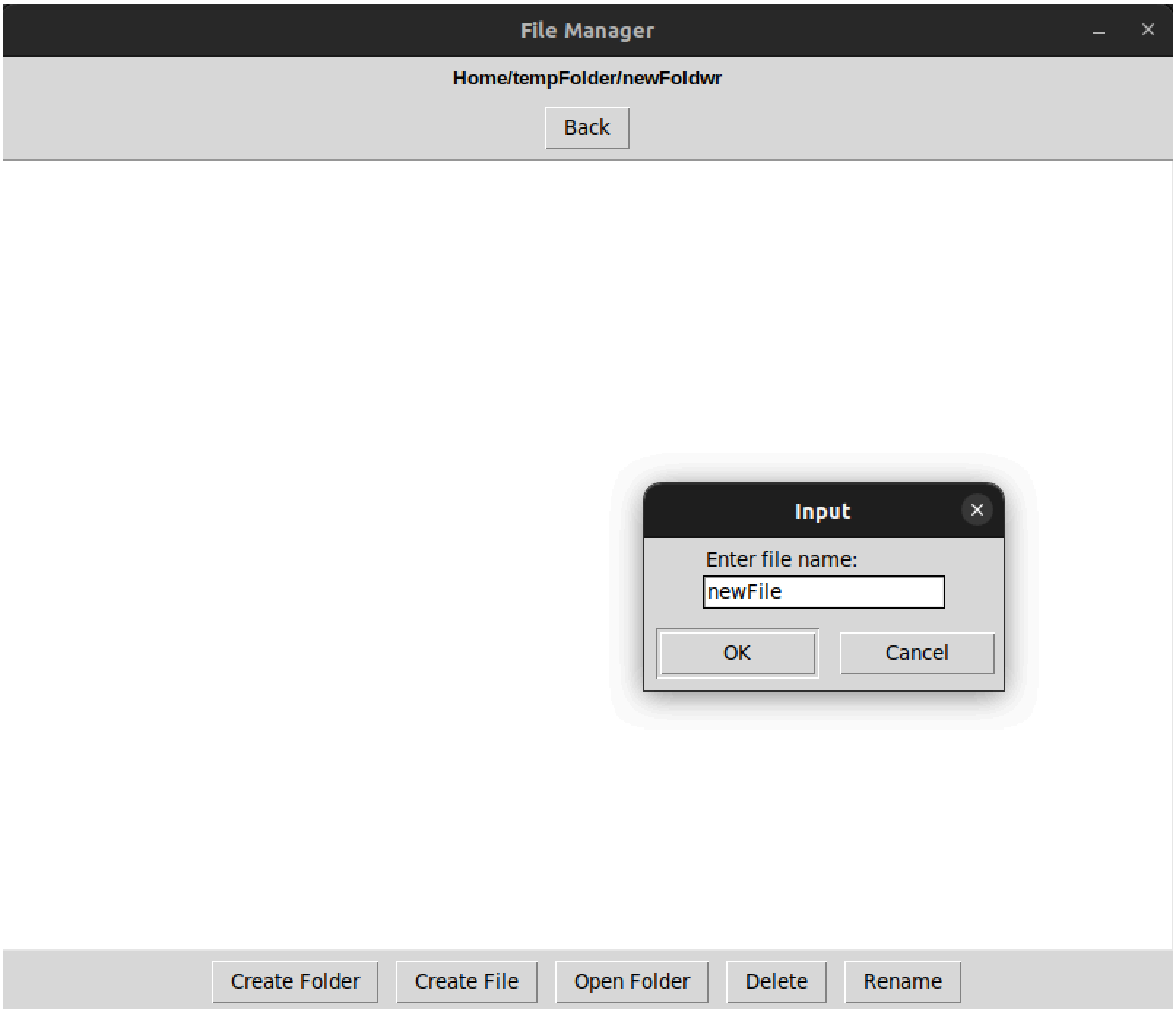
buy Amazon, buy Yahoo, buy eBay, buy Yahoo, buy Yahoo, buy Oracle

Their goal is to be able to dream up short sequences and quickly detect whether they are subsequences of S. So this is the problem they pose to you: Give greedy method solution and an algorithm that takes two sequences of events—S' of length m and S of length n, each possibly containing an event more than once—and decides in time O(m + n) whether S' is a subsequence of S.



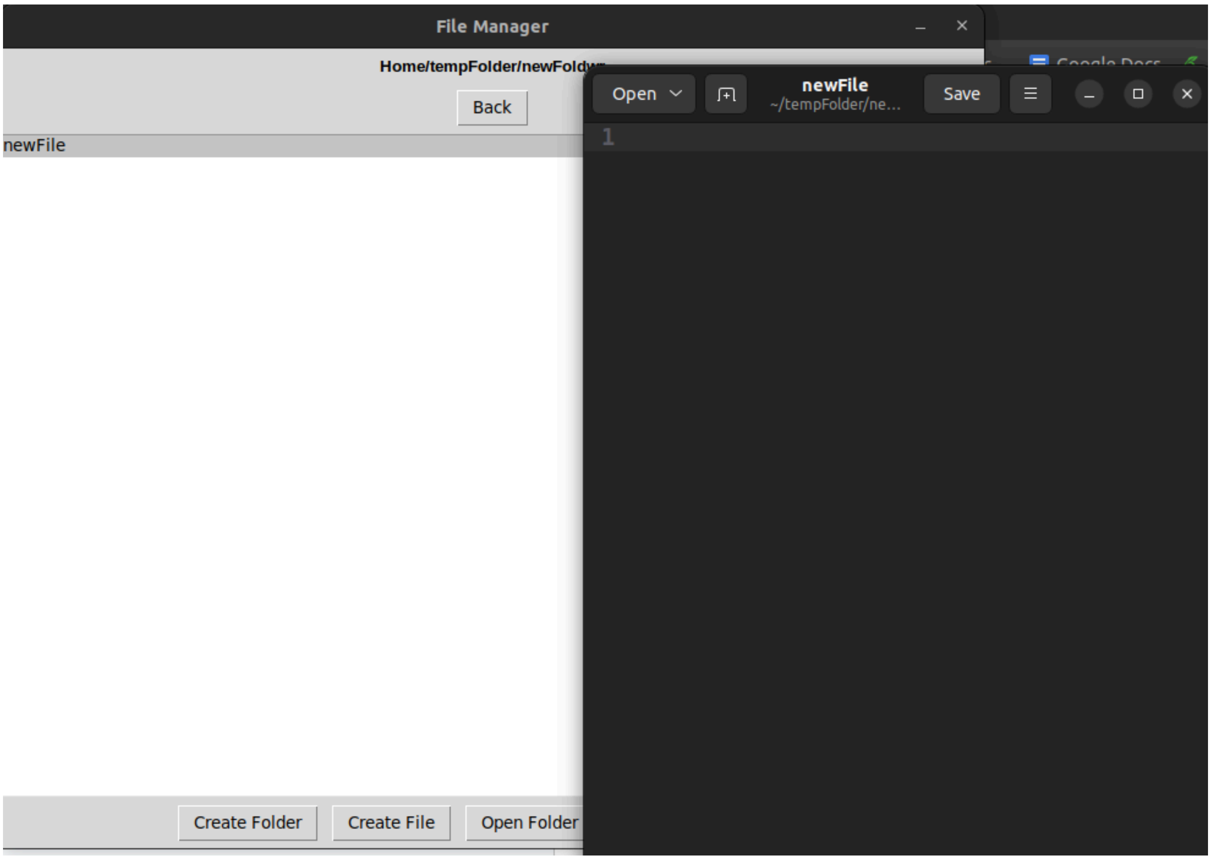
File Manager
Home/tempFolder
Back
temp2.txt
Input
Enter folder name:
newFoldwr
OK
Cancel
Create Folder
Create File
Open Folder
Delete
Rename

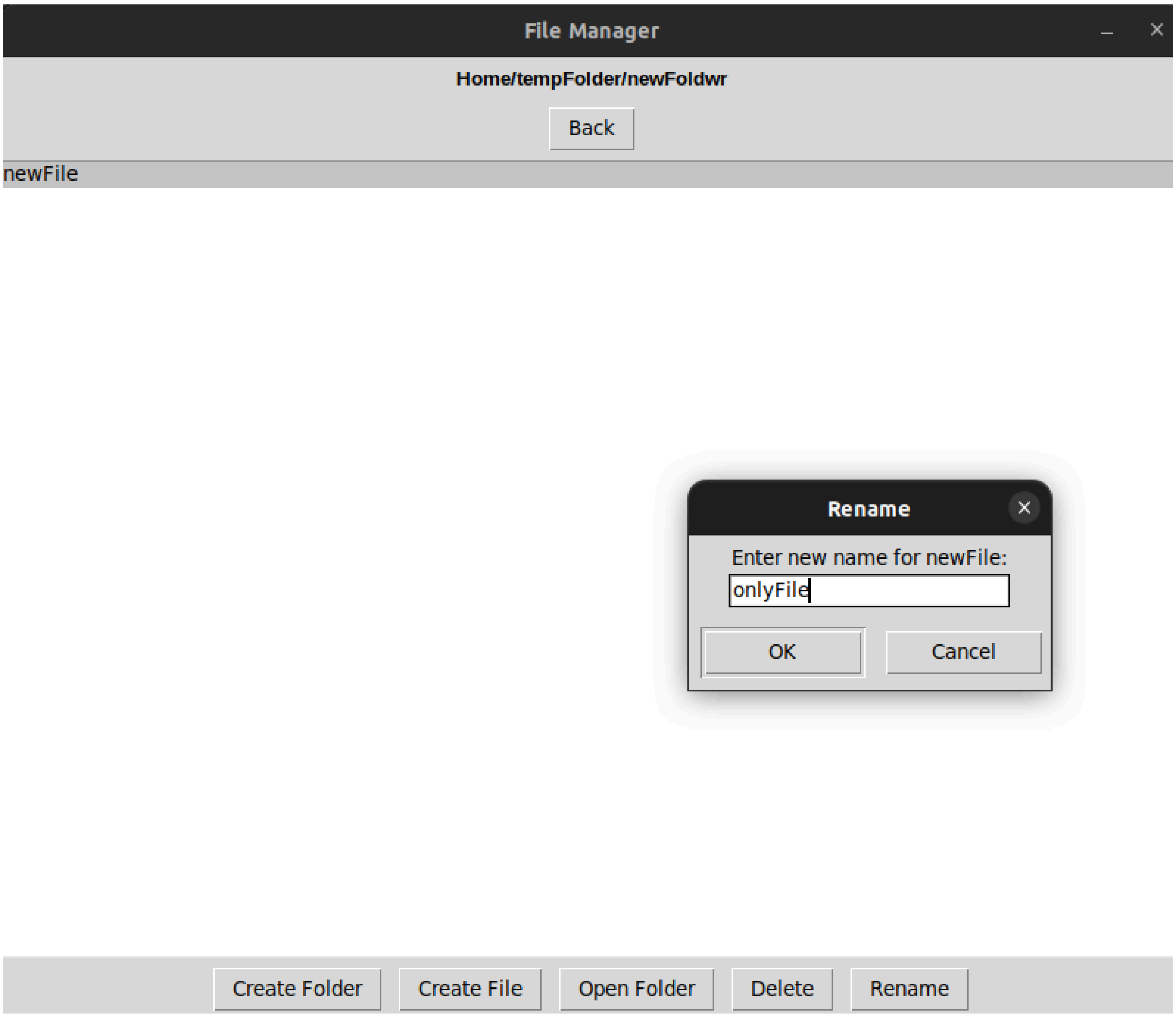

File Manager
Home/tempFolder
Back
temp2.txt
newFoldwr
Create Folder
Create File
Open Folder
Delete
Rename



Home/tempFolder/newFoldwr

Enter file name: newFile


File Manager
Home/tempFolder/newFolder
Back
newFile
Create Folder
Create File
Open Folder
Open
newFile
~/tempFolder/ne...
Save
1
Google Docs

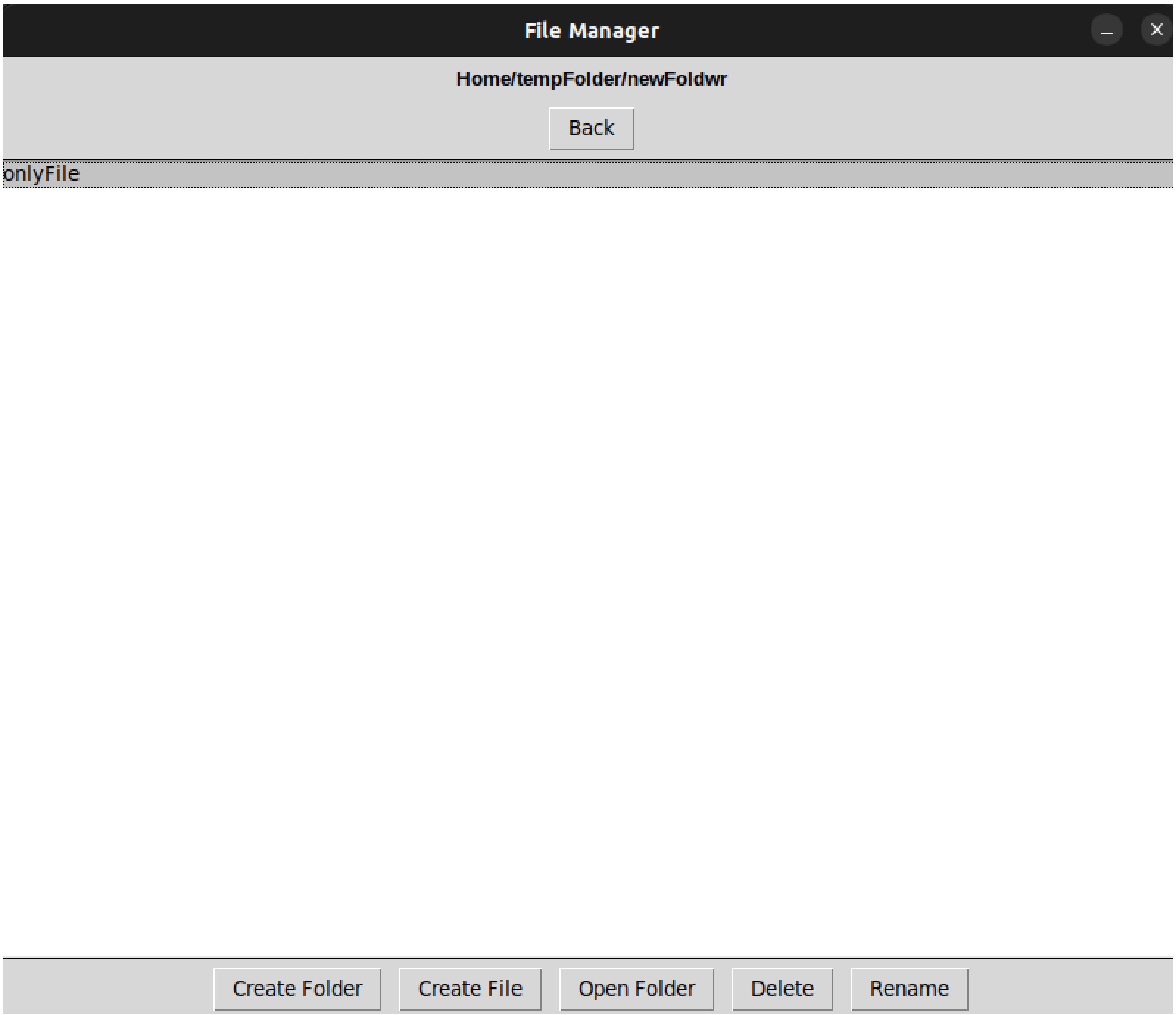


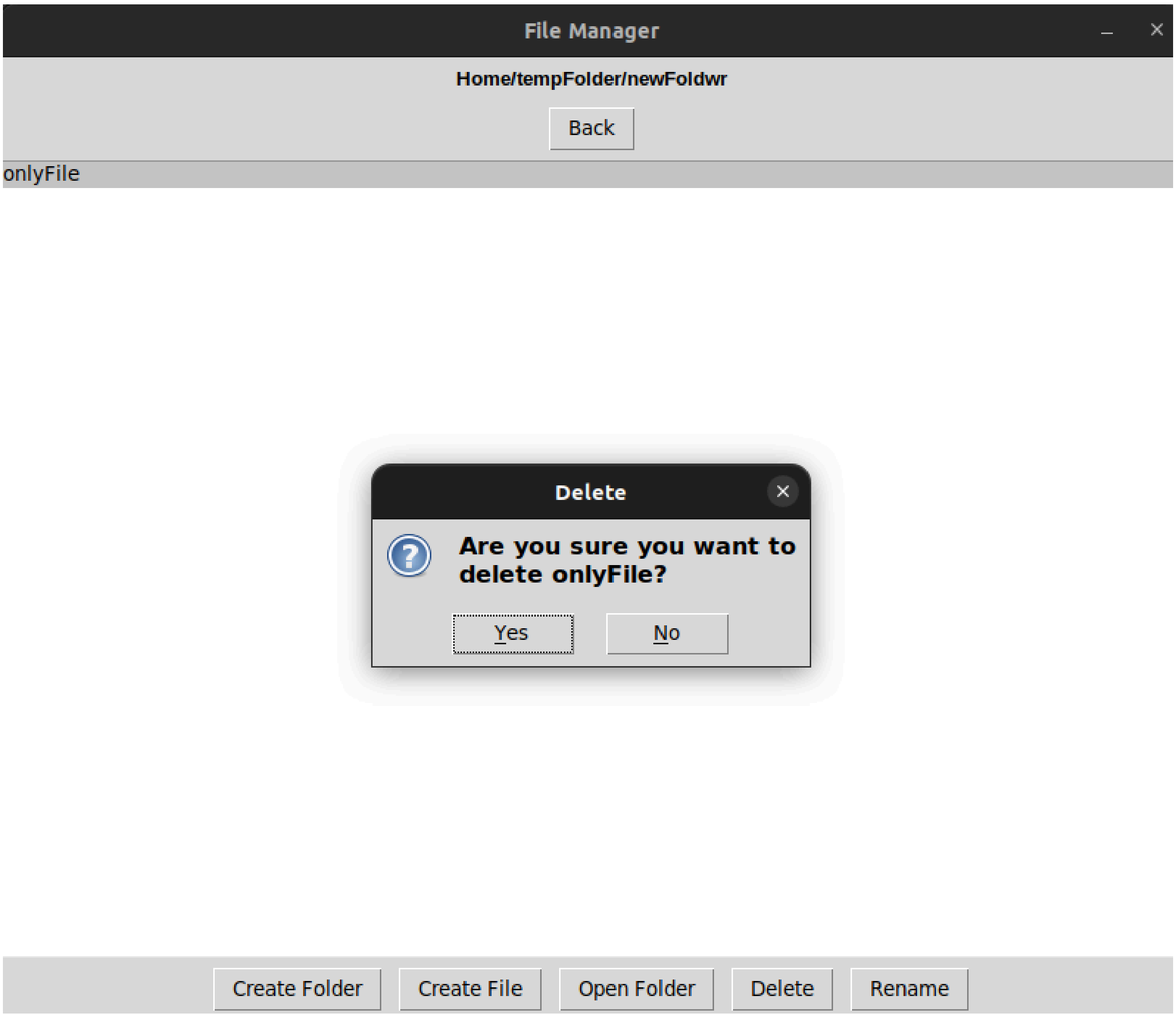
Home/tempFolder/newFoldwr

newFile

Enter new name for newFile:

onlyFile

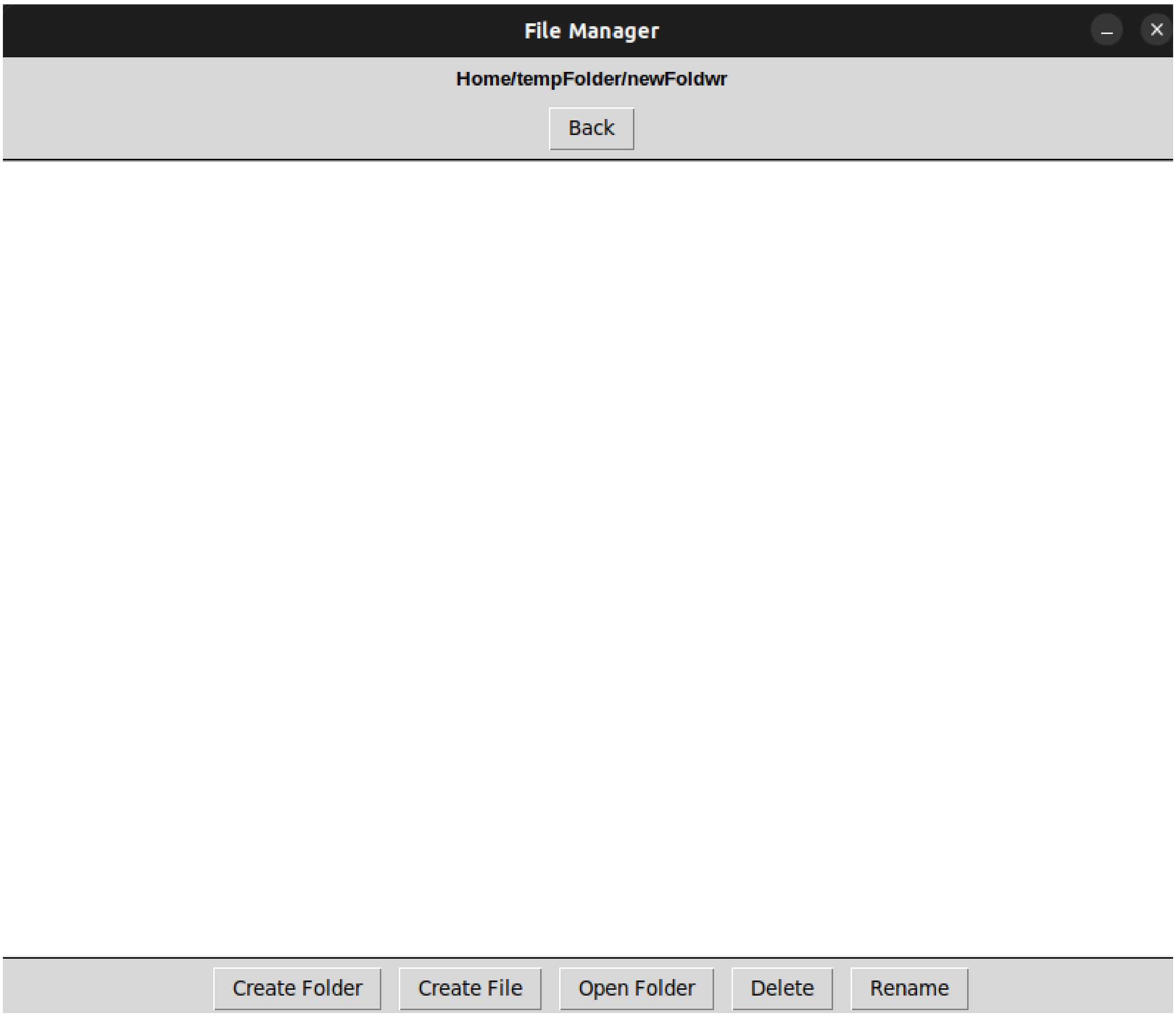

File Manager
Home/tempFolder/newFoldwr
Back
onlyFile
Create Folder
Create File
Open Folder
Delete
Rename

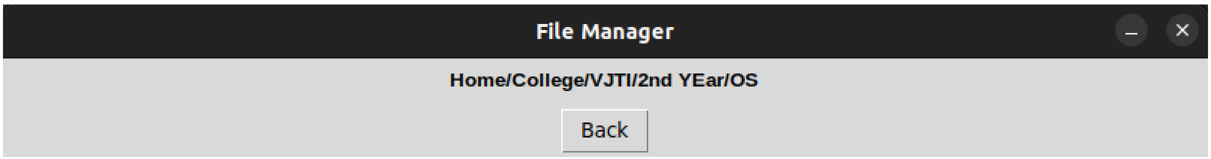


Home/tempFolder/newFoldwr

onlyFile

Are you sure you want to delete onlyFile?





Conclusion : Here in this experiment, we learnt about file manager, its key features, operations and examples. And we implemented simple file manager using Tkinter, os and subprocess library in python.