

EXPERIMENT 3-CLI, GUI AND VUI

CLI (Command Line Interface)

CLI allows users to interact with the system using text commands typed in the terminal. It is fast, lightweight, and suitable for experienced users.

CODE:

```
import os
import sys

def rename_file(old_name, new_name):
    try:
        os.rename(old_name, new_name)
        print(f"File renamed from {old_name} to {new_name}")
    except FileNotFoundError:
        print(f"Error: {old_name} not found.")
    except Exception as e:
        print(f"An error occurred: {e}")

if __name__ == "__main__":
    if len(sys.argv) != 3:
        print("Usage: python rename_file_cli.py <old_filename>
<new_filename>")
    else:
        rename_file(sys.argv[1], sys.argv[2])
```

OUTPUT:

```
Usage: python rename_file_cli.py <old_filename> <new_filename>
PS C:\Users\ADMIN\OneDrive\Desktop\UID> python rename_file_cli.py
final.txt old.txt
File renamed from final.txt to old.txt
```

HOW IT WORKS:

- User selects create / rename / exit from menu
- Program performs file operation using Python functions
- Output displayed in terminal



GUI (Graphical User Interface)

GUI allows users to interact using graphical elements like buttons, text boxes, and windows. It is user-friendly and easy to use.

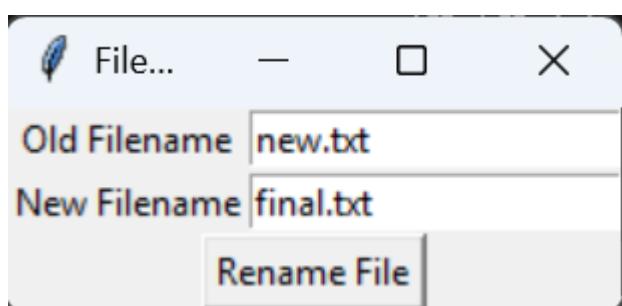
CODE:

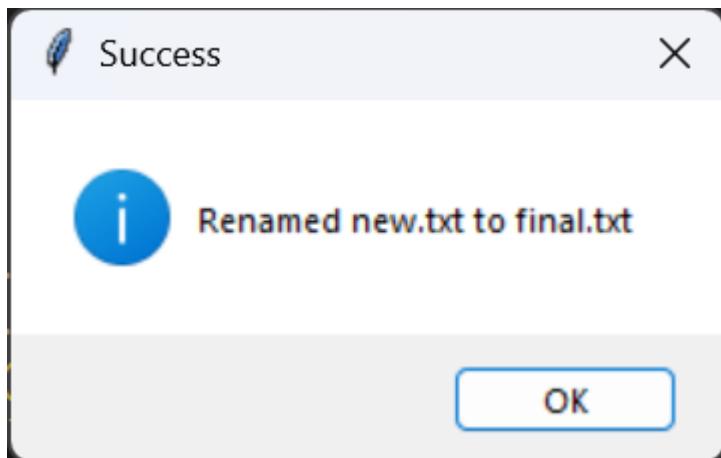
```
import tkinter as tk
from tkinter import messagebox
import os

def rename_file():
    old_name = old_entry.get()
    new_name = new_entry.get()
    try:
        os.rename(old_name, new_name)
```

```
    messagebox.showinfo("Success", f"Renamed {old_name} to  
{new_name}")  
except FileNotFoundError:  
    messagebox.showerror("Error", "File not found")  
  
root = tk.Tk()  
root.title("File Renamer")  
  
tk.Label(root, text="Old Filename").grid(row=0, column=0)  
tk.Label(root, text="New Filename").grid(row=1, column=0)  
  
old_entry = tk.Entry(root)  
new_entry = tk.Entry(root)  
old_entry.grid(row=0, column=1)  
new_entry.grid(row=1, column=1)  
  
tk.Button(root, text="Rename File", command= rename_file).grid(row=2,  
columnspan=2)  
  
root.mainloop()
```

OUTPUT:





HOW IT WORKS:

- User enters file name in text box
- Click buttons (Create, Rename, Delete)
- Files shown in listbox
- Status bar shows result

◆ VUI (Voice User Interface)

VUI allows users to interact with the system using voice commands. It improves accessibility and provides hands-free operation.

CODE:

```
import speech_recognition as sr
import os
import sounddevice as sd
from scipy.io.wavfile import write

samplerate = 44100
```

```
def record(seconds=4):
    print("Speak now...")
    audio = sd.rec(int(seconds * samplerate), samplerate=samplerate,
    channels=1, dtype='int16')
    sd.wait()
    write("temp.wav", samplerate, audio)

def listen(prompt):
    print(prompt)
    record()

r = sr.Recognizer()
with sr.AudioFile("temp.wav") as source:
    audio_data = r.record(source)

try:
    text = r.recognize_google(audio_data)
    print("You said:", text)
    return text.lower()
except:
    print("Could not understand. Try again.")
    return None

def rename_file():
    print("Welcome to the Voice-Controlled File Renamer!")
```

```
old_name = listen("Say the name of the file you want to rename  
(without .txt)")
```

```
if not old_name:
```

```
    return
```

```
new_name = listen("Say the new name for the file (without .txt)")
```

```
if not new_name:
```

```
    return
```

```
old_file = old_name + ".txt"
```

```
new_file = new_name + ".txt"
```

```
if not os.path.exists(old_file):
```

```
    print(f"Error: '{old_file}' not found.")
```

```
    return
```

```
os.rename(old_file, new_file)
```

```
print(f"\033[92m \u2713 File successfully renamed from '{old_file}' to  
'{new_file}'\033[0m")
```

```
rename_file()
```

OUTPUT:

```
Welcome to the Voice-Controlled File Renamer!
Say the name of the file you want to rename (without .txt)
Speak now...
You said: final
Say the new name for the file (without .txt)
Speak now...
You said: voice
 File successfully renamed from 'final.txt' to 'voice.txt'
```

HOW IT WORKS:

- System listens to voice command
- User says create or rename
- Program performs operation
- System speaks result