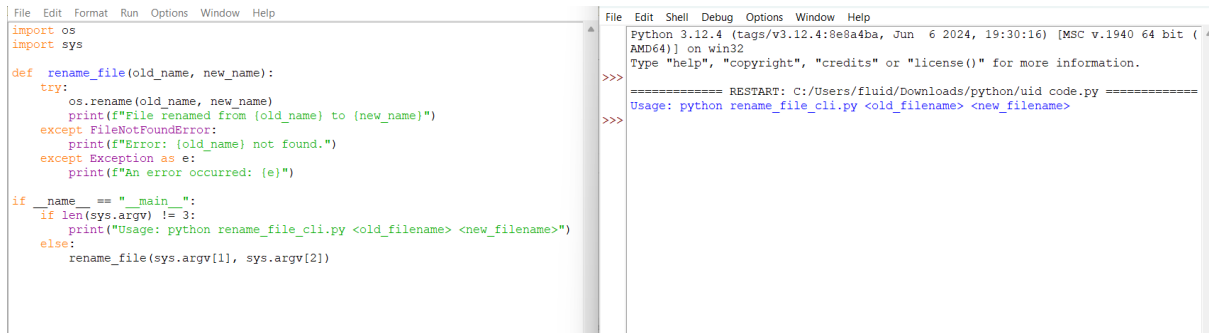


UID

EX NO : 3

1. Command Line Interface



The screenshot displays a Python IDE with two windows. The left window shows the source code for a command-line interface (CLI) for renaming files. The code defines a `rename_file` function that takes `old_name` and `new_name` as arguments, attempts to rename the file using `os.rename`, and handles exceptions for `FileNotFoundError` and general errors. A `__main__` block checks for the correct number of command-line arguments and calls the `rename_file` function. The right window shows the output of running the script, displaying the usage message: `Usage: python rename_file_cli.py <old_filename> <new_filename>`.

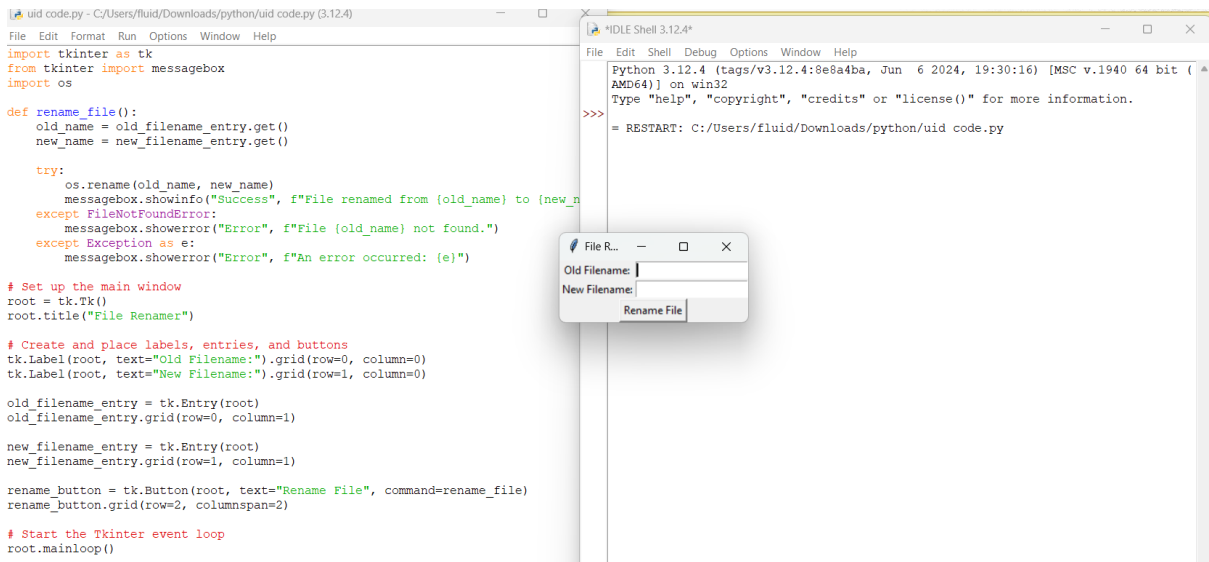
```
File Edit Format Run Options Window Help
import os
import sys

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])

File Edit Shell Debug Options Window Help
Python 3.12.4 (tags/v3.12.4:8e8a4ba, Jun 6 2024, 19:30:16) [MSC v.1940 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/fluid/Downloads/python/uid code.py =====
Usage: python rename_file_cli.py <old_filename> <new_filename>
>>>
```

2. Graphical User Interface



The screenshot displays a Python IDE with two windows. The left window shows the source code for a graphical user interface (GUI) for renaming files. The code uses `tkinter` to create a window titled "File Renamer" with two input fields for "Old Filename" and "New Filename", and a "Rename File" button. The `rename_file` function is called when the button is clicked. The right window shows the output of running the script, displaying the usage message: `Usage: python rename_file_cli.py <old_filename> <new_filename>`. A small "File R..." dialog box is also visible in the foreground.

```
uid code.py - C:/Users/fluid/Downloads/python/uid code.py (3.12.4)
File Edit Format Run Options Window Help
import tkinter as tk
from tkinter import messagebox
import os

def rename_file():
    old_name = old_filename_entry.get()
    new_name = new_filename_entry.get()

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

# Set up the main window
root = tk.Tk()
root.title("File Renamer")

# Create and place labels, entries, and buttons
tk.Label(root, text="Old Filename:").grid(row=0, column=0)
tk.Label(root, text="New Filename:").grid(row=1, column=0)

old_filename_entry = tk.Entry(root)
old_filename_entry.grid(row=0, column=1)

new_filename_entry = tk.Entry(root)
new_filename_entry.grid(row=1, column=1)

rename_button = tk.Button(root, text="Rename File", command=rename_file)
rename_button.grid(row=2, columnspan=2)

# Start the Tkinter event loop
root.mainloop()

File Edit Shell Debug Options Window Help
Python 3.12.4 (tags/v3.12.4:8e8a4ba, Jun 6 2024, 19:30:16) [MSC v.1940 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/fluid/Downloads/python/uid code.py =====
Usage: python rename_file_cli.py <old_filename> <new_filename>
>>>
```

3. Voice User Interface

```
uia code.py - C:/Users/uidq/uownoaas/python/uia code.py (3.12.4)
File Edit Format Run Options Window Help
import speech_recognition as sr
import os

def rename_file_from_voice_command(command):
    # Extracting old and new filename from the command
    try:
        words = command.split(" ")
        old_name = words[1]
        new_name = words[3]

        os.rename(old_name, new_name)
        print(f"File Renamed from {old_name} to {new_name}")
    except Exception as e:
        print(f"Error: {e}")

def listen_for_command():
    recognizer = sr.Recognizer()
    mic = sr.Microphone()

    print("Listening for command to rename a file...")
    with mic as source:
        recognizer.adjust_for_ambient_noise(source)
        audio = recognizer.listen(source)

    try:
        command = recognizer.recognize_google(audio)
        print(f"Command received: {command}")
        rename_file_from_voice_command(command)
    except sr.UnknownValueError:
        print("Sorry, I couldn't understand the command.")
    except sr.RequestError as e:
        print(f"Could not request results from Google Speech Recognition service")

if __name__ == "__main__":
    listen_for_command()
```

```
IDLE Shell 3.12.4
File Edit Shell Debug Options Window Help
Python 3.12.4 (tags/v3.12.4:8e8a4ba, Jun 6 2024, 19:30:16) [MSC v.1940 64 bit AMD64] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/fluid/Downloads/python/uid code.py
Listening for command to rename a file...
Command received: UID code
Error: list index out of range
>>>|
```

```
uia code.py - C:/Users/uidq/uownoaas/python/uia code.py (3.12.4)
File Edit Format Run Options Window Help
import speech_recognition as sr
import os

def rename_file_from_voice_command(command):
    # Extracting old and new filename from the command
    try:
        words = command.split(" ")
        old_name = words[1]
        new_name = words[3]

        os.rename(old_name, new_name)
        print(f"File Renamed from {old_name} to {new_name}")
    except Exception as e:
        print(f"Error: {e}")

def listen_for_command():
    recognizer = sr.Recognizer()
    mic = sr.Microphone()

    print("Listening for command to rename a file...")
    with mic as source:
        recognizer.adjust_for_ambient_noise(source)
        audio = recognizer.listen(source)

    try:
        command = recognizer.recognize_google(audio)
        print(f"Command received: {command}")
        rename_file_from_voice_command(command)
    except sr.UnknownValueError:
        print("Sorry, I couldn't understand the command.")
    except sr.RequestError as e:
        print(f"Could not request results from Google Speech Recognition service")

if __name__ == "__main__":
    listen_for_command()
```

```
IDLE Shell 3.12.4
File Edit Shell Debug Options Window Help
Python 3.12.4 (tags/v3.12.4:8e8a4ba, Jun 6 2024, 19:30:16) [MSC v.1940 64 bit AMD64] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/fluid/Downloads/python/uid code.py
Listening for command to rename a file...
Command received: UID code
Error: list index out of range
>>>|
```