LOG ANALYZER – PYTHON CODE

```
import tkinter as tk
from tkinter import filedialog
import os

def extract_errors_warnings(log_file_path):
    # Read the log file
    with open(log_file_path, 'r') as f:
        log_lines = f.readlines()

    # Extract the warnings and errors
    warnings = []
    errors = []
    for line in log_lines:
        if 'WARNING' in line:
            warnings.append(line)
        elif 'ERROR' in line:
            errors.append(line)

    return warnings, errors

def generate_report(warnings, errors, output_file_path):
    # Create a text report
    with open(output_file_path, 'w') as f:
        f.write("Log Analysis Report\n\n")
        f.write("Warnings:\n")
        for w in warnings:
            f.write(w)
        f.write("\nErrors:\n")
        for e in errors:
            f.write(e)

def select_input_file():
    input_file_path = filedialog.askopenfilename()
    input_file_entry.delete(0, tk.END)
    input_file_entry.insert(0, input_file_path)

def select_output_file():
    output_file_path = filedialog.asksaveasfilename(defaultextension=".txt")
    output_file_entry.delete(0, tk.END)
    output_file_entry.insert(0, output_file_path)

def generate_report_action():
    input_file_path = input_file_entry.get()
    output_file_path = output_file_entry.get()

    # Check if the input file exists
    if not os.path.exists(input_file_path):
        tk.messagebox.showerror("Error", "Input file does not exist")
        return

    # Extract the warnings and errors from the log file
    warnings, errors = extract_errors_warnings(input_file_path)

    # Generate the text report
    generate_report(warnings, errors, output_file_path)

root = tk.Tk()
root.title("Log Analysis Tool by Arun Anbu")
root.geometry("500x200")

label = tk.Label(root, text="Select input and output files")
label.pack(pady=10)

input_file_button = tk.Button(root, text="Select input file", command=select_input_file)
input_file_button.pack(pady=10)

output_file_button = tk.Button(root, text="Select output file",
command=select_output_file)
output_file_button.pack(pady=10)

input_file_entry = tk.Entry(root, width=50)
input_file_entry.pack(pady=10)

output_file_entry = tk.Entry(root, width=50)
output_file_entry.pack(pady=10)

generate_button = tk.Button(root, text="Generate Report",
command=generate_report_action)
generate_button.pack(pady=10)

root.mainloop()
```