

Zeus D. Elderfield
CS3202N - G2
Programming Assignment #1

```
Python
import tkinter as tk

# -----
# Assignment Algorithm
# -----


def reverse_word(word: str) -> str:
    new_word = ""

    index = len(word) - 1

    # Add letter to new word starting from last index
    while index >= 0:
        character = word[index]
        new_word += character
        index -= 1

    return new_word


def reverse_text(text: str) -> str:
    words = text.split() # Divide text into words by whitespace
    reversed_words = []

    # Reverse each word in the text
    for word in words:
        reversed_word = reverse_word(word)
        reversed_words.append(reversed_word)

    # Reconstruct the different words back to a single text
    new_text = " ".join(reversed_words)
    return new_text


# -----
# Theme Variables (Minimalist)
# -----


THEME = {
    "bg": "#fefefe",           # light background
    "fg": "#222222",           # dark text
    "accent": "#4a90e2",        # subtle accent
```

```

"font_main": ("Arial", 12),
"font_label": ("Arial", 10, "bold"),
"padding": 12,
"entry_bg": "#ffffff",      # white input/output boxes
"entry_fg": "#222222",
"button_bg": "#4a90e2",
"button_fg": "#ffffff",
"button_alt_bg": "#e0e0e0",
"button_alt_fg": "#222222"
}
# -----



def process_input():
    text = input_entry.get()
    reversed_text = reverse_text(text)

    output_entry.config(state="normal")
    output_entry.delete(0, tk.END)
    output_entry.insert(0, reversed_text)
    output_entry.config(state="readonly")

def clear_fields():
    input_entry.delete(0, tk.END)
    output_entry.config(state="normal")
    output_entry.delete(0, tk.END)
    output_entry.config(state="readonly")

# Main window
root = tk.Tk()
root.title("Programming Assignment #1")
root.configure(bg=THEME[ "bg"])
root.geometry("400x220")
root.resizable(False, False)

# Input label and entry
input_frame = tk.Frame(root, bg=THEME[ "bg"])
input_frame.pack(fill="x", padx=THEME[ "padding"], pady=(THEME[ "padding"], 0))

tk.Label(
    input_frame,
    text="Enter text:",
    bg=THEME[ "bg"],
    fg=THEME[ "fg"],
    font=THEME[ "font_label"],

```

```
        anchor="w"
    ).pack(fill="x")

input_entry = tk.Entry(
    input_frame,
    font=THEME["font_main"],
    bg=THEME["entry_bg"],
    fg=THEME["entry_fg"],
    relief="solid",
    bd=1
)
input_entry.pack(fill="x", pady=(2, THEME["padding"]))

# Output label and entry
output_frame = tk.Frame(root, bg=THEME["bg"])
output_frame.pack(fill="x", padx=THEME["padding"])

tk.Label(
    output_frame,
    text="Output:",
    bg=THEME["bg"],
    fg=THEME["fg"],
    font=THEME["font_label"],
    anchor="w"
).pack(fill="x")

output_entry = tk.Entry(
    output_frame,
    font=THEME["font_main"],
    bg=THEME["entry_bg"],
    fg=THEME["entry_fg"],
    relief="solid",
    bd=1,
    state="readonly"
)
output_entry.pack(fill="x", pady=(2, THEME["padding"]))

# Buttons at the bottom
button_frame = tk.Frame(root, bg=THEME["bg"])
button_frame.pack(side="bottom", pady=THEME["padding"])

reverse_btn = tk.Button(
    button_frame,
    text="Reverse",
```

```
        command=process_input,
        bg=THEME[ "button_bg"],
        fg=THEME[ "button_fg"],
        font=THEME[ "font_main"],
        relief="flat",
        padx=20,
        pady=6
    )
reverse_btn.pack(side="left", padx=6)

clear_btn = tk.Button(
    button_frame,
    text="Clear",
    command=clear_fields,
    bg=THEME[ "button_alt_bg"],
    fg=THEME[ "button_alt_fg"],
    font=THEME[ "font_main"],
    relief="flat",
    padx=20,
    pady=6
)
clear_btn.pack(side="left", padx=6)

root.mainloop()
```