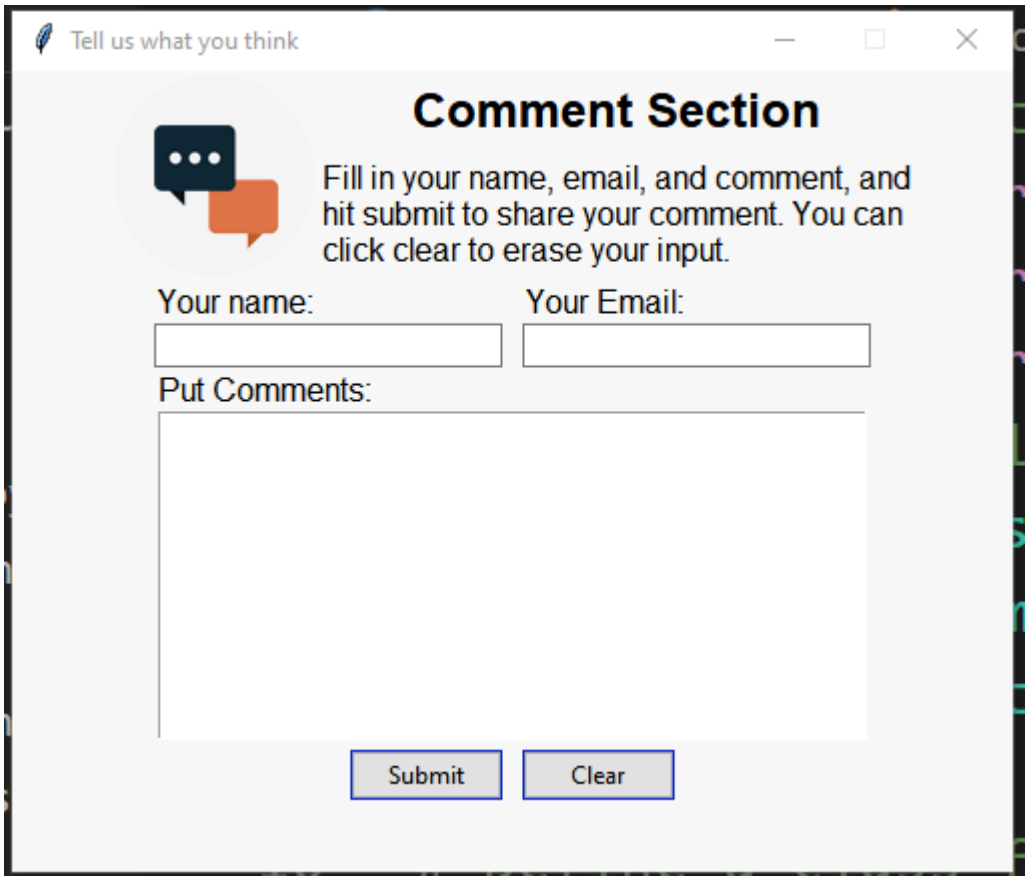
```
# Import the tkinter module and its submodules
from tkinter import *
from tkinter import ttk
from tkinter import messagebox
# Import the modules for handling images and URLs
from urllib.request import urlopen
from PIL import Image, ImageTk
from io import BytesIO

# Define a class for the feedback app
class Feedback:

    # Define the constructor method for the class
    def __init__(this_instance, mainframe):
        # Set the title, size, and background color of the main window
        mainframe.title('Tell us what you think')
        mainframe.resizable(False, False)
        mainframe.configure(background='#f7f7f7')

        # Create a style object to customize the widgets
        this_instance.style = ttk.Style()
        this_instance.style.configure('TFrame', background='#f7f7f7')
        this_instance.style.configure('TButton', background='#0825C8')
        this_instance.style.configure('TLabel', background='#f7f7f7', font=('Arial', 12))
        this_instance.style.configure('Header.TLabel', font=('Arial', 18, 'bold'))

        # Create a frame for the header section
        this_instance.header_frame = ttk.Frame(mainframe)
        this_instance.header_frame.pack()

        # Load the comment icon from the internet
        logo_url = "https://external-content.duckduckgo.com/iu/?u=https%3A%2F%2Ftemplatic.com%2Fwp-
themes%2Fuploads%2F2015%2F12%2Fcomment-images-
icon.png&f=1&nofb=1&ipt=f0b202869eacdc8420200576fcbfc848cea7406d30537aa3bc837c773353fc4a&ipo=images"
        image_data = urlopen(logo_url).read()
        image = Image.open(BytesIO(image_data))
        python_logo = ImageTk.PhotoImage(image)

        # Create labels for the icon, the app name, and the instructions
        ttk.Label(this_instance.header_frame, image=python_logo).grid(row=0, column=0, rowspan=2)
        ttk.Label(this_instance.header_frame, text='Comment Section', style='Header.TLabel').grid(row=0,
column=1)
        ttk.Label(this_instance.header_frame, wraplength=300,
                  text='Fill in your name, email, and comment, and hit submit to share your comment. You
can click clear to erase your input.').grid(row=1, column=1)

        # Create a frame for the content section
        this_instance.content_in_frame = ttk.Frame(mainframe)
        this_instance.content_in_frame.pack()

        # Create labels for the name, email, and comments fields
        ttk.Label(this_instance.content_in_frame, text='Your name:').grid(row=0, column=0, padx=5,
sticky='sw')
        ttk.Label(this_instance.content_in_frame, text='Your Email:').grid(row=0, column=1, padx=5,
sticky='sw')
        ttk.Label(this_instance.content_in_frame, text='Put Comments:').grid(row=2, column=0, padx=5,
sticky='sw')

        # Create entry widgets for the name and email fields
        this_instance.comment_name = ttk.Entry(this_instance.content_in_frame, width=24, font=('Arial',
10))
```

```
        this_instance.comment_email = ttk.Entry(this_instance.content_in_frame, width=24, font=('Arial',
10))
        # Create a text widget for the comments field
        this_instance.comments = Text(this_instance.content_in_frame, width=50, height=10, font=('Arial',
10))

        # Arrange the widgets using the grid geometry manager
        this_instance.comment_name.grid(row=1, column=0, padx=5)
        this_instance.comment_email.grid(row=1, column=1, padx=5)
        this_instance.comments.grid(row=3, column=0, columnspan=2, padx=5)

        # Create buttons for submitting and clearing the input
        ttk.Button(this_instance.content_in_frame, text='Submit',
command=this_instance.submit).grid(row=4, column=0, padx=5, pady=5, sticky='e')
        ttk.Button(this_instance.content_in_frame, text='Clear', command=this_instance.clear).grid(row=4,
column=1, padx=5, pady=5, sticky='w')

        # Keep a reference to the image to prevent it from being garbage collected
        this_instance.python_logo = python_logo

    # Define the method for submitting the input
    def submit(this_instance):
        # Print the input to the console
        print(f'Name: {this_instance.comment_name.get()}')
        print(f'Email: {this_instance.comment_email.get()}')
        print(f'Comments: {this_instance.comments.get(1.0, "end")}')
        # Clear the input fields
        this_instance.clear()
        # Show a message box to thank the user
        messagebox.showinfo(title='Comment info', message='Thanks for your comment!')

    # Define the method for clearing the input
    def clear(this_instance):
        # Delete the text from the entry and text widgets
        this_instance.comment_name.delete(0, 'end')
        this_instance.comment_email.delete(0, 'end')
        this_instance.comments.delete(1.0, 'end')

# Define the main function
def main():
    # Create the root window
    root = Tk()
    # Set the size of the window
    root.geometry('500x400')
    # Create an instance of the feedback class
    feedback = Feedback(root)
    # Start the main loop
    root.mainloop()

# Check if the script is run directly
if __name__ == '__main__':
    # Call the main function
    main()
```


Tell us what you think
Comment Section
Fill in your name, email, and comment, and hit submit to share your comment. You can click clear to erase your input.
Your name:
Your Email:
Put Comments:
Submit
Clear