This Python script creates a **Personal Diary Application** using the **Tkinter** library. The application provides an interface for users to write, save, and view diary entries. Below is a description of the key features, components, and functionality.

**Description of Features:**

**1. Write Diary Entries:**

* Users can write multi-line diary entries in a Text widget.
* The content is cleared after the entry is saved to ensure a clean slate for new inputs.

**2. Save Entries:**

* Entries are saved to a diary\_entries directory as .txt files.
* The filenames are automatically generated based on the current date using the format: diary\_YYYY-MM-DD.txt.
* If the diary\_entries folder does not exist, it is created dynamically.

**3. View Saved Entries:**

* Users can view all previously saved diary entries in a new window.
* Saved entries are listed in a Listbox.
* Users can select and open an entry to view its content.

**Core Components:**

**Tkinter Widgets:**

1. **Text**:
   * Used for multi-line text input to write diary entries.
   * Allows users to enter large amounts of text easily.
2. **Button**:
   * "Save Entry" Button:
     + Saves the current text to a file.
   * "View Entries" Button:
     + Opens a new window listing all saved entries.
3. **Listbox**:
   * Displays the list of saved diary entries in a new window.
   * Allows users to select and open specific entries.
4. **Toplevel**:
   * Used to open secondary windows for viewing entries or content.
5. **Label**:
   * Provides instructions or headings in the application.

**File Handling:**

* **Directory Creation**:
  + Ensures the diary\_entries folder exists before saving any files.
* **File Writing**:
  + Diary content is saved with proper formatting.
* **File Reading**:
  + Content is retrieved when viewing a saved entry.

**How the Application Works:**

**1. Writing and Saving Entries:**

* Users type their diary content in the Text widget.
* Clicking "Save Entry":
  + Checks if the Text widget is empty. If empty, it shows a warning.
  + If not empty, the content is saved in the diary\_entries folder with a filename based on the current date.
  + The Text widget is cleared after saving.

**2. Viewing Previous Entries:**

* Clicking "View Entries":
  + Opens a new window showing a list of all saved entries using a Listbox.
  + If no entries exist, it informs the user with a messagebox.

**3. Opening an Entry:**

* Users select an entry from the list and click "Open Entry."
* The selected file is opened in a new window, displaying its content in a Text widget.

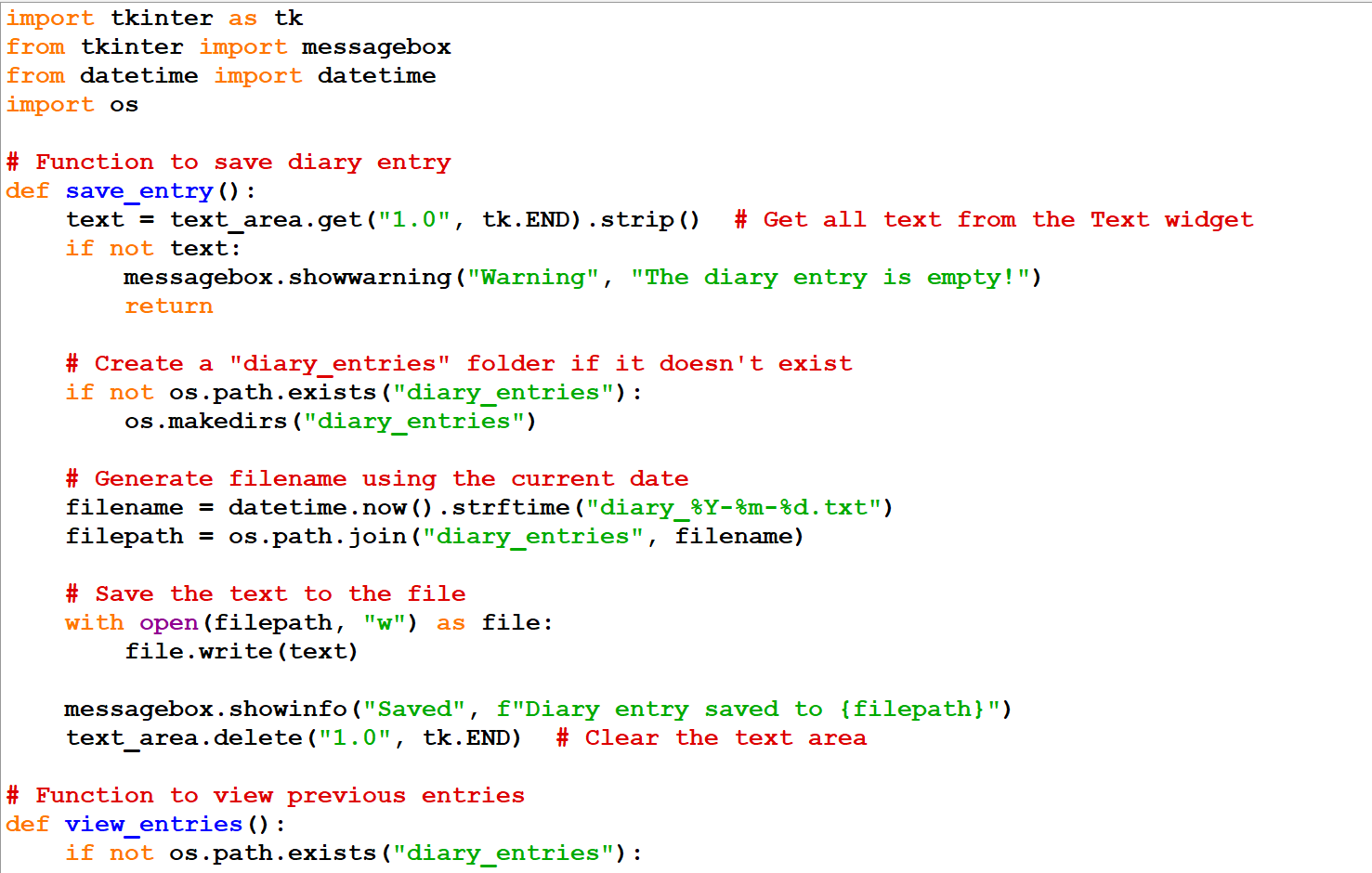
**Error Handling:**

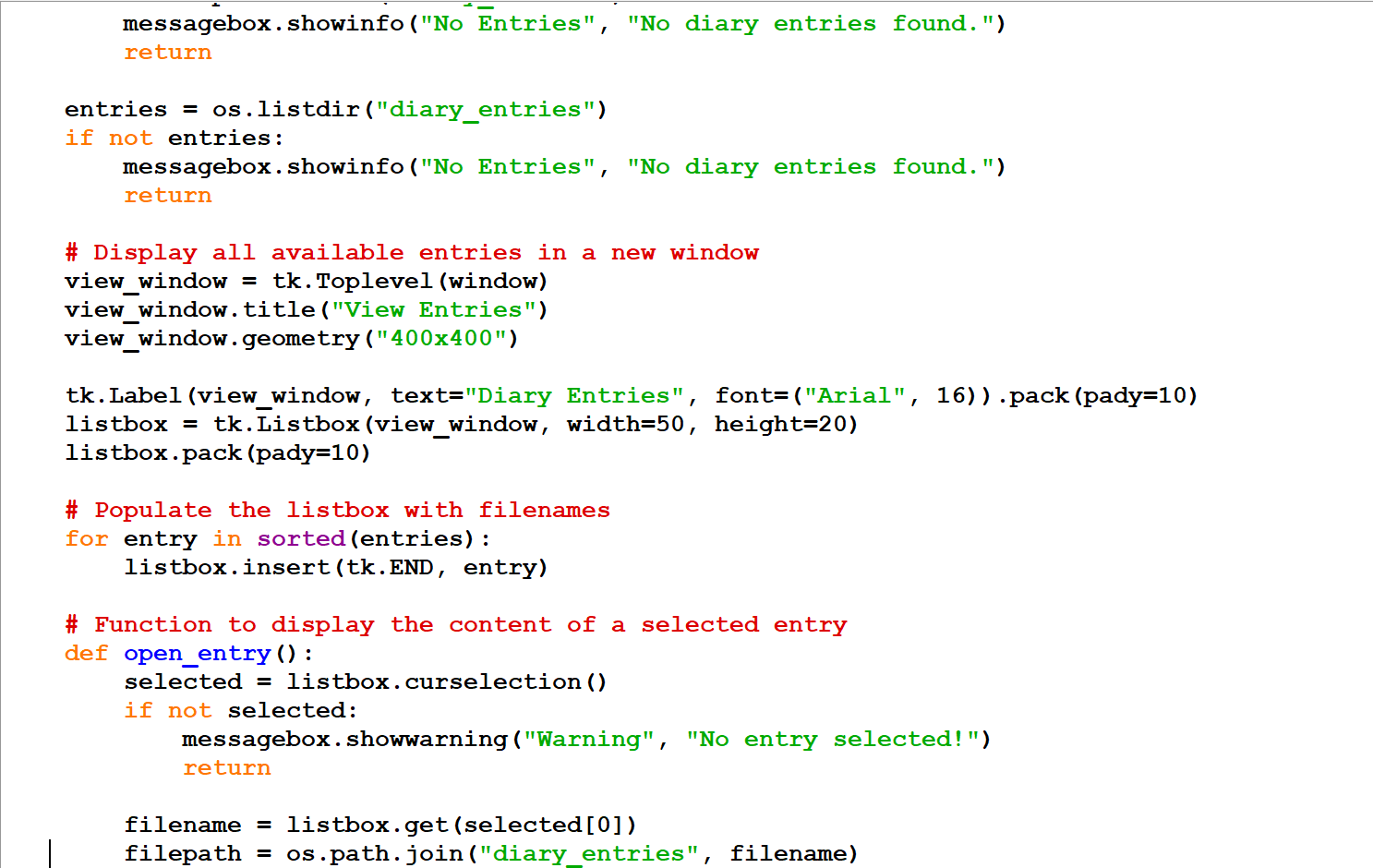
* Warns users if:
  + They try to save an empty diary entry.
  + No diary entries are found when attempting to view them.
  + No entry is selected in the listbox before clicking "Open Entry."
* Displays user-friendly messages using messagebox.

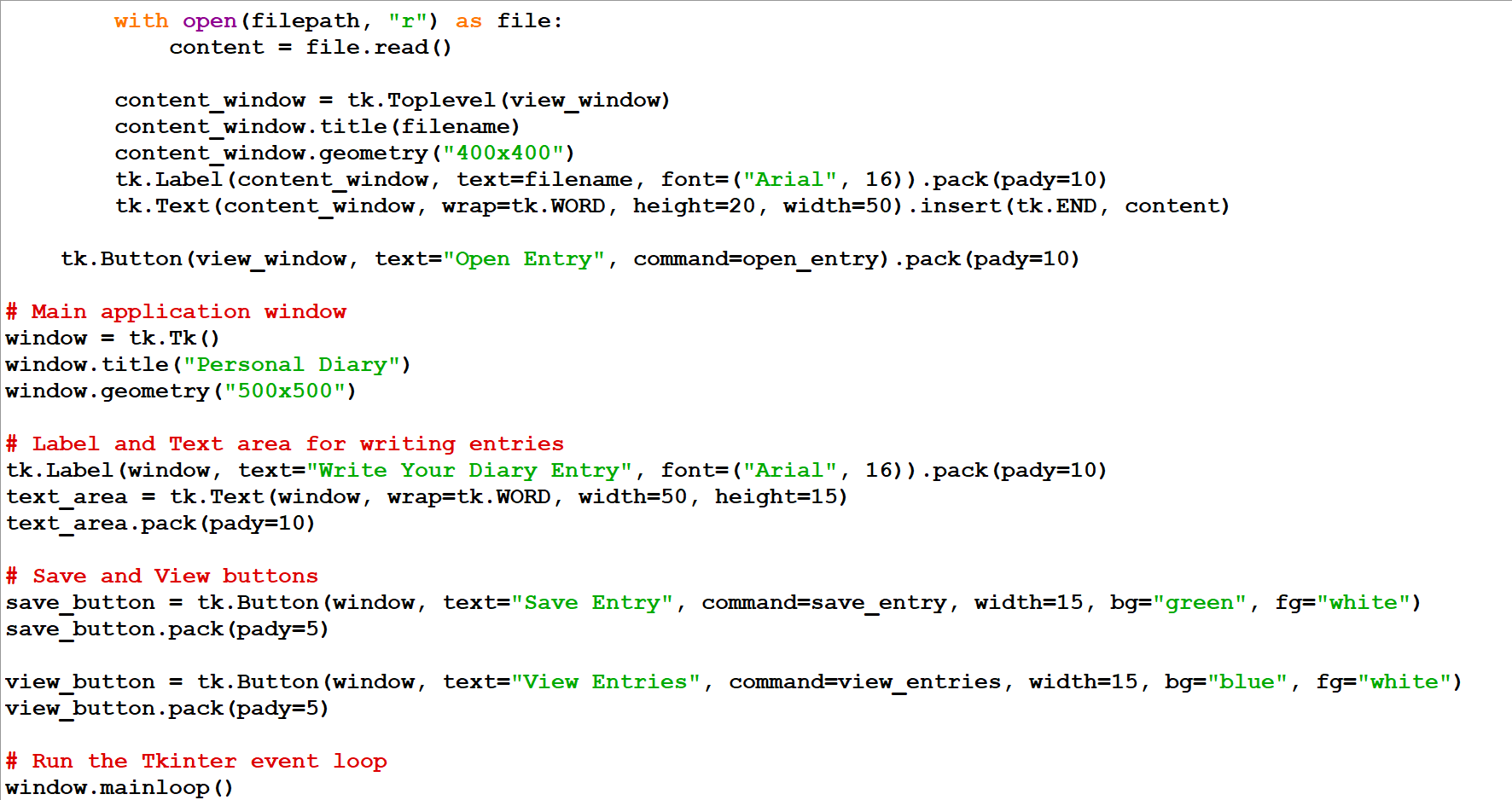
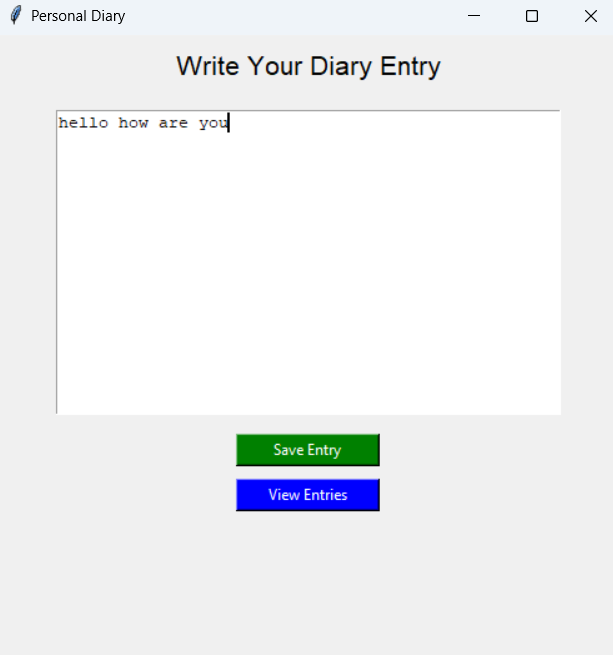
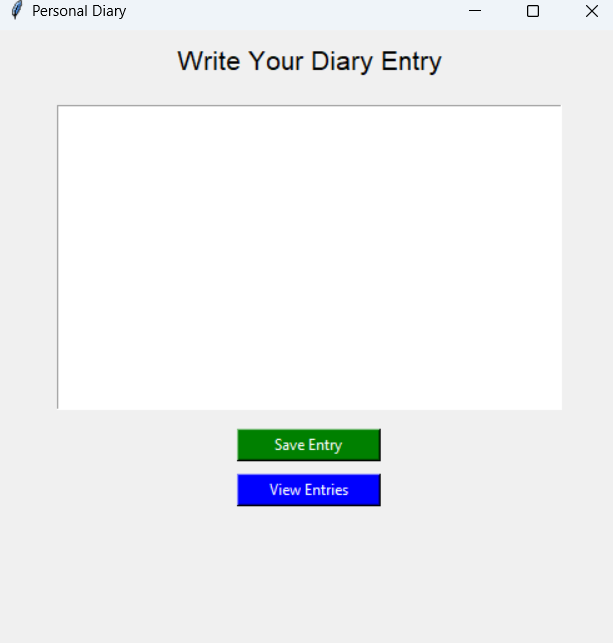
**Possible Enhancements:**

1. **Date Navigation**:
   * Add functionality to navigate through entries by date.
2. **Password Protection**:
   * Secure the diary with a login system or password.
3. **Search Functionality**:
   * Allow users to search for specific entries by keywords.
4. **Rich Text Formatting**:
   * Enable text styling (e.g., bold, italic) in diary entries.
5. **Cloud Integration**:
   * Save entries in a cloud storage system for access from multiple devices.

**SCREENSHOTS**

****

****

****