

# GUITraker Manual

- 1. Introduction**
- 2. Requirements**
- 3. Instructions**
- 4. Use**
- 5. Validation**
- 6. Thanks**

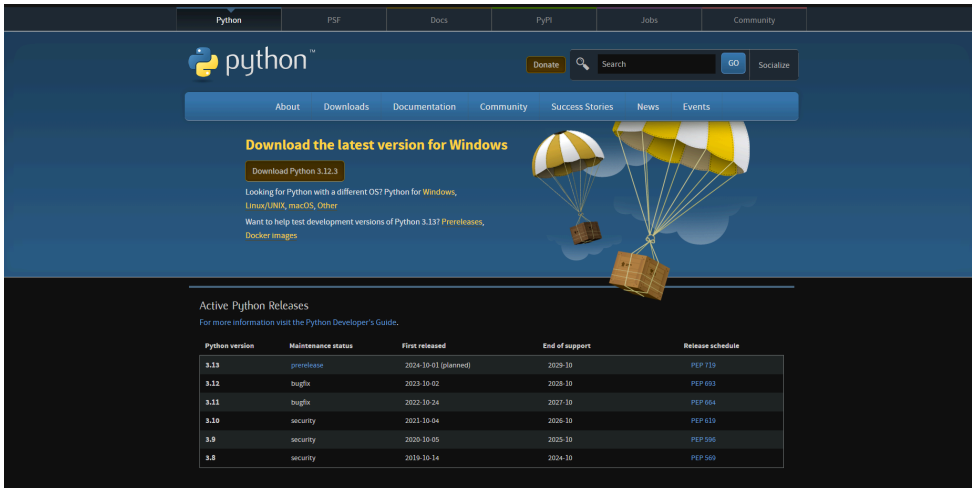
## 1. Introduction

Habit Tracker is a GUI application designed to help you track and maintain your daily habits. With this application, add the habits you want to complete each day, easy to use and simple to understand.

## 2. Requirements

- Operating system: Windows, macOS or Linux
- Python 3.x
- PIL (Pillow)
- Tkinter

To install Python: <https://www.python.org/downloads/>



## To install PIL (Pillow)

Write in the terminal: *pip install pillow*

```
PROBLEMS OUTPUT DEBUG-CONSOLE TERMINAL PORTS Python + - [ ] [ ] ... ^ x
PS C:\Users\adria\Documents\Homework\Python\Final Project> & C:/Users/adria/AppData/Local/Programs/Python/Python312/python.exe "c:/Users/adria/Documents/Homework/Python/Final Project/Home 4-2/GUITracker.py"
PS C:\Users\adria\Documents\Homework\Python\Final Project> pip install pillow
Requirement already satisfied: pillow in c:\users\adria\appdata\local\programs\python\python312\lib\site-packages (10.3.0)

[notice] A new release of pip is available: 23.2.1 -> 24.0
[notice] To update, run: python.exe -m pip install --upgrade pip
PS C:\Users\adria\Documents\Homework\Python\Final Project> 
```

## To install Tkinter

Go to the link:

<https://lambertk.academic.wlu.edu/breezypythongui/>

Click on Downloads

Home About the Author Installation Downloads Links Overview Tutorial Quick Reference


**Programmers** import breezypythongui into their applications and start writing GUI-based programs the easy way.

**Definition:** breezypythongui is a module of classes that takes the pain out of writing GUI-based programs.

**breezypythongui** is not a GUI builder, editor, or development environment. Instead, it **simplifies the user interface and event model** for programmers by subclassing various classes in Python's tkinter framework.

- **Easy to use** because you don't need to know the details of tkinter resources.
- **Easy to install** because it is contained in a Python module.
- **Flexible and extensible** because the source code is provided.

### GUIDE TO BREEZYPYTHONGUI

Overview	What breezypythongui is and why you will want to use it.
 Downloads	Copyright and other rules governing the use of the material on this website. Obtain breezypythongui source code and demo programs for free.
Installation	How to install breezypythongui on your computer.
Tutorial	How to use breezypythongui.
Quick Reference	A quick guide to breezypythongui classes and methods.
Books	Python textbooks for beginners.
About Me	Meet the author of breezypythongui.

**breezypythongui now runs on Python 2.x.x OR Python 3.x.x!**

Go down until you find:

Home Courses ▾ Publications ▾ Software ▾ Links


link to this website.

- Source code: You may modify the source code to better suit your own purposes. However, if you wish to publish or otherwise make available the finished product, you must
  - give it a new name
  - prominently acknowledge that it is a modification of this software
  - provide a link to this website
  - make the modification available to others on the same terms as the original is made available to you
  - inform me via email so I can track spin-offs

Any questions can be directed to the author at [lambertk@wlu.edu](mailto:lambertk@wlu.edu).

**You must read the terms and conditions above before you are authorized to download or use the source code listed at the end of this page.**

### Downloads

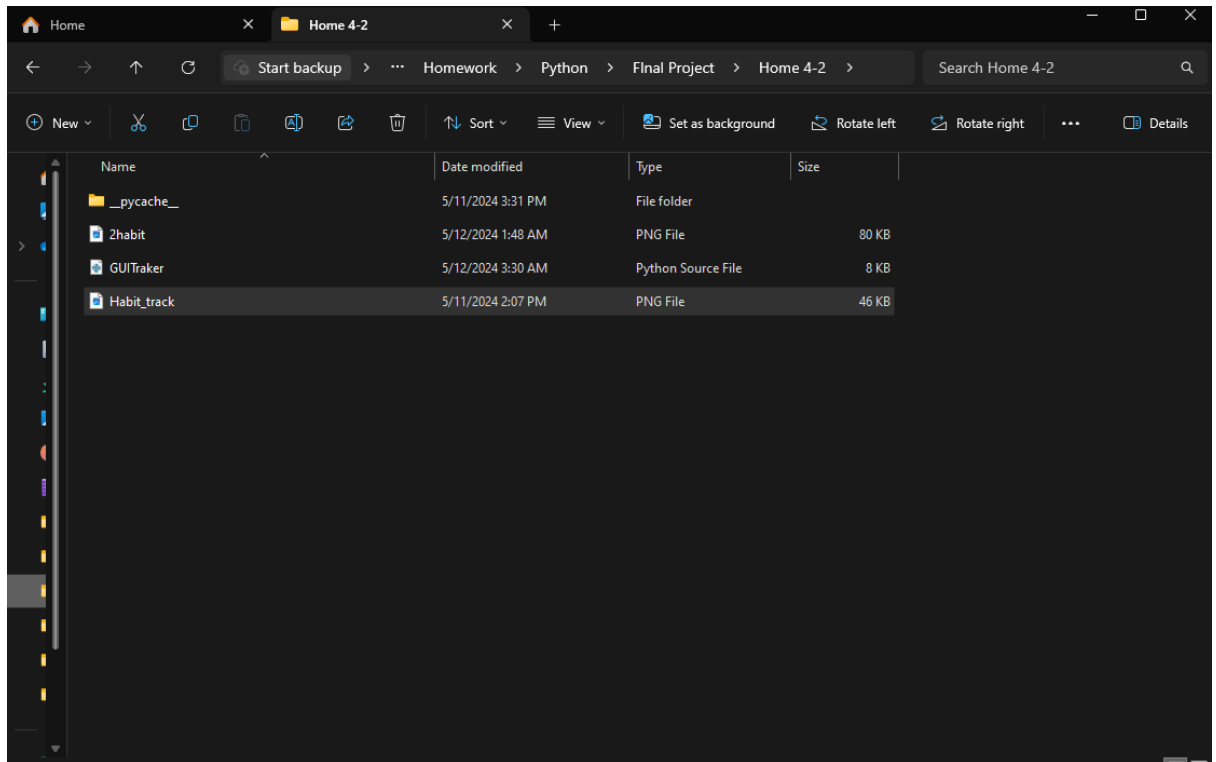
 breezypythongui_2_or_3	Version 1.1, works with either Python 2.x.x or Python 3.x.x.
breezypythongui	Version 1.0 (Python 3.x.x only).
Demo Programs	Example programs that appear in the eBook Easy GUI Programming in Python.
Example Programs	Source code for the programs shown on this Web site.

© Ken Lambert. All rights reserved.

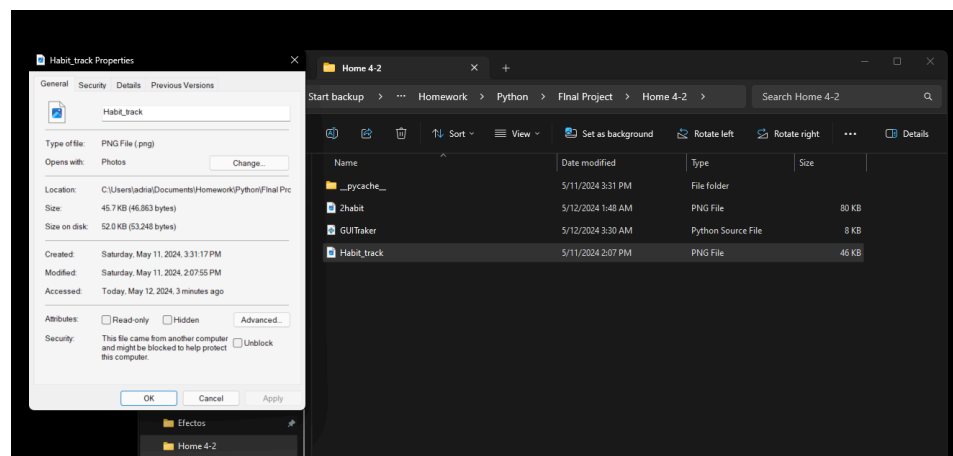
copy all the folders that are in the zip, and paste them in the same folder as the program, otherwise you will not be able to run the program.

### 3. Instructions

1. Copy all the folders that are in the ZIP, make sure they are in the same folder as **Tkinter**



2. To get the images in place, follow these steps:
  - A) First see where they are located and copy the location ("**habit\_track.png**") and ("**2habit.png**")



example:C:\Users\adria\Documents\Homework\Python\Final Project\Home 4-2

go to the program and search for ("1 IMAGE") and ("2 IMAGE") and replace:

```
image_path =  
"C:\\Users\\adria\\Documents\\Homework\\Python\\Final  
Project\\Home 4-2\\habit_track.png"
```

```
# 1111      IIIII MM      MM      AA      GGGGGG EEEEEEE  
#   11      II  MM MM  MM MM      A  A      GG      EE  
#   11      II  MM  MM  MM      AAAAAA  GG  GGG  EEEE  
#   11      II  MM      MM      AA      AA      GG      GG  EE  
# 111111      iiiiii MM      MM      AA      AA      GGGGG  EEEEEEE  
  
image_path = "C:\\Users\\adria\\Documents\\Homework\\Python\\Final Project\\Home 4-2\\habit_track.png"  
image = Image.open(image_path)  
photo = ImageTk.PhotoImage(image)  
self.image_label = tk.Label(self, image=photo)  
self.image_label.image = photo  
self.image_label.pack()
```

```
112  
113  
114  
115  
116  
117  
118  
119  
120  
121  
122  
123  
# 2222      IIIII MM      MM      AA      GGGGGG EEEEEEE  
# 22  22      II  MM MM  MM MM      A  A      GG      EE  
#   22      II  MM  MM  MM      AAAAAA  GG  GGG  EEEE  
#   22      II  MM      MM      AA      AA      GG      GG  EE  
# 22222222      iiiiii MM      MM      AA      AA      GGGGG  EEEEEEE  
  
image_path2 = "C:\\Users\\adria\\Documents\\Homework\\Python\\Final Project\\Home 4-2\\2habit.png"  
habits_window = tk.Toplevel(self)  
habits_window.title("Habits")  
habits_window.geometry("700x600")
```

By the direction of your image,

**IMPORTANT** make sure you have (\\) in each (\\) otherwise the images will not be seen, as shown in the image, also that at the end of the location, add ("habit\_track.png") at the end of the 1 image, the same with the 2 image ("2habit.png")

## 4. Use

### Main window

When you start the program, you will see the main window with a welcome image and three buttons: "Add habit", "View habits" and "Exit".

### Add habit

To add a new habit, follow these steps:

Click the "Add Habit" button.

In the text box, type the name of the habit you want to add.

Click the "Add" button.

The new habit will be added to the list of habits and will appear in the Habits window.

### Viewing habits

To view your habits and mark the days when you completed them, follow these steps:

Click the "View habits" button.

In the habits window, you will see a list of all your habits with checkboxes for each day of the week.

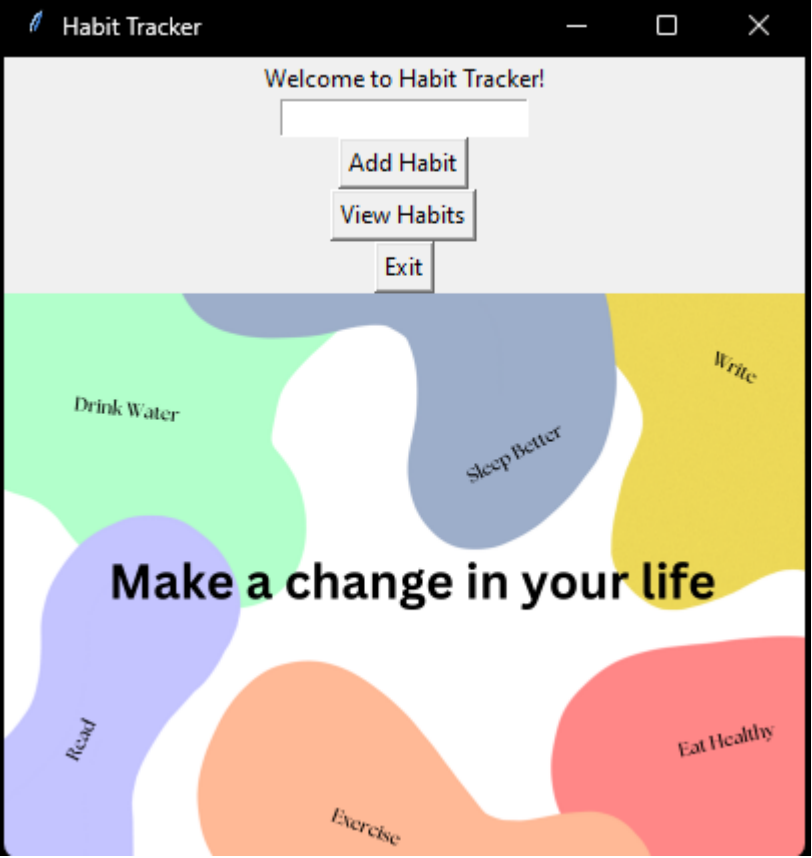
Check the checkboxes corresponding to the days you completed each habit.

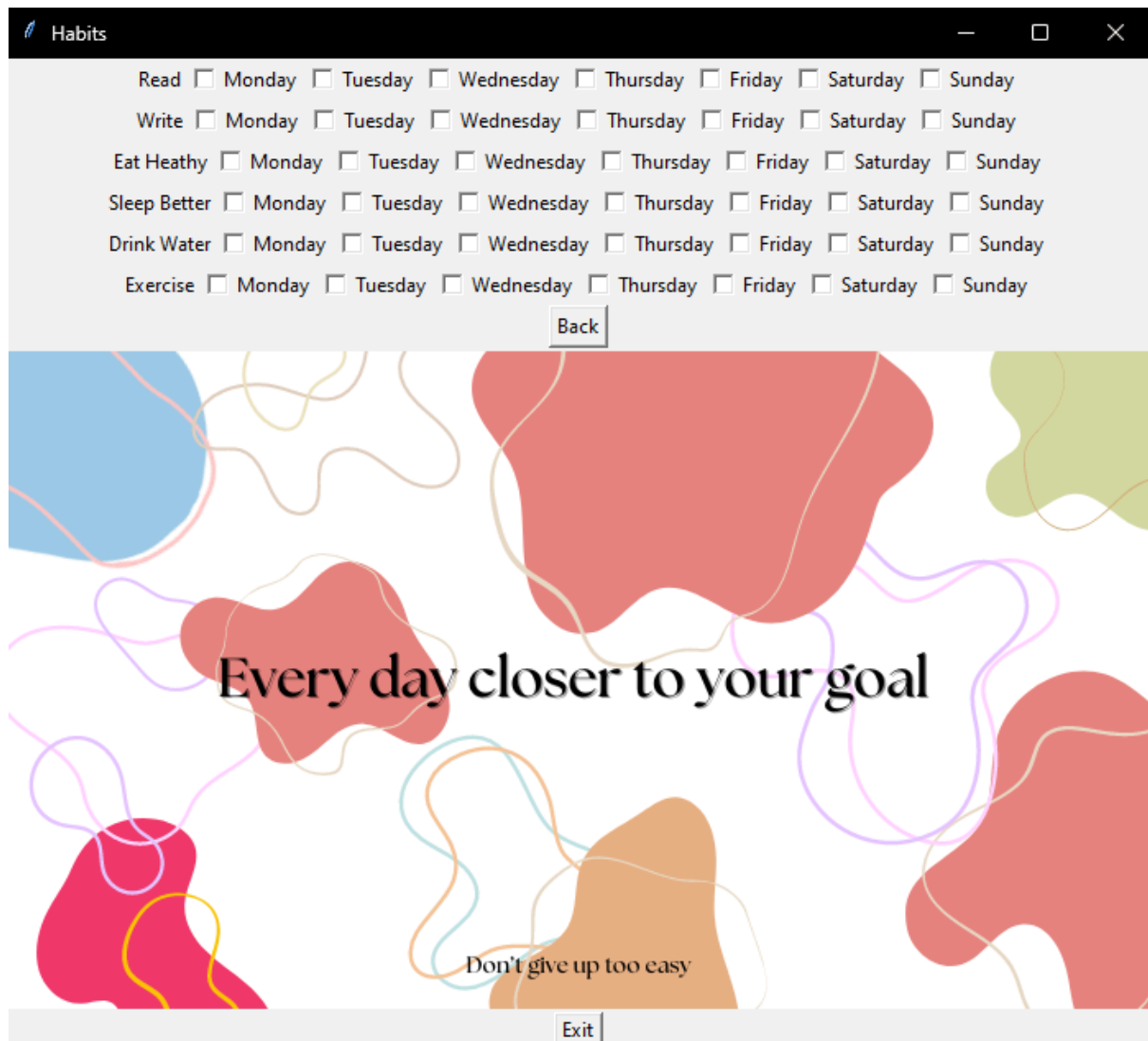
When you are done, click the "Back" button to return to the main window.

### Exit

To exit the program, click the "Exit" button.

## 5. Result





## 6. Thanks

This section is more to thank you for this year, in which I thank you for your time and however you continue to teach, I wish you the best and enjoy a good vacation.

Adrian Valdez (CEO of ROMX)