Python Assignment

A Simple registration form using Tkinter: Write a program where User has to fill up the required information and that information is automatically written into an excel file.

Submitted By-

Name- Nimish Bhardwaj

Sec-B

Roll No.-22

Redg No.-201700315

Contents-

* About GUI
* The Packages-Tkinter and Openpyxl
* Source code
* Input and Output

# What is GUI?

The graphical user interface is a form of [user interface](https://en.wikipedia.org/wiki/User_interface" \o "User interface) that allows [users](https://en.wikipedia.org/wiki/User_(computing)" \o "User (computing)) to [interact with electronic devices](https://en.wikipedia.org/wiki/Human%E2%80%93computer_interaction" \o "Human–computer interaction) through graphical [icons](https://en.wikipedia.org/wiki/Computer_icon" \o "Computer icon) and visual indicators such as secondary notation, instead of [text-based user interfaces](https://en.wikipedia.org/wiki/Text-based_user_interface" \o "Text-based user interface), typed command labels or text navigation. GUIs were introduced in reaction to the perceived steep [learning curve](https://en.wikipedia.org/wiki/Learning_curve" \o "Learning curve) of [command-line interfaces](https://en.wikipedia.org/wiki/Command-line_interface" \o "Command-line interface) (CLIs),which require commands to be typed on a [computer keyboard](https://en.wikipedia.org/wiki/Computer_keyboard" \o "Computer keyboard).

The actions in a GUI are usually performed through [direct manipulation](https://en.wikipedia.org/wiki/Direct_manipulation" \o "Direct manipulation) of the graphical elements.[*[better source needed](https://en.wikipedia.org/wiki/Wikipedia:NOTRS" \o "Wikipedia:NOTRS)*] Beyond computers, GUIs are used in many handheld [mobile devices](https://en.wikipedia.org/wiki/Mobile_device" \o "Mobile device) such as [MP3](https://en.wikipedia.org/wiki/MP3" \o "MP3) players, portable media players, gaming devices, [smartphones](https://en.wikipedia.org/wiki/Smartphone" \o "Smartphone) and smaller household, office and [industrial controls](https://en.wikipedia.org/wiki/Distributed_control_system" \o "Distributed control system). The term *GUI* tends not to be applied to other lower-[display resolution](https://en.wikipedia.org/wiki/Display_resolution" \o "Display resolution) [types of interfaces](https://en.wikipedia.org/wiki/User_interface" \l "Types" \o "User interface), such as [video games](https://en.wikipedia.org/wiki/Video_game" \o "Video game) (where *head-up display* ([HUD](https://en.wikipedia.org/wiki/HUD_(video_gaming)" \o "HUD (video gaming))) is preferred), or not including flat screens, like [volumetric displays](https://en.wikipedia.org/wiki/Volumetric_display" \o "Volumetric display) because the term is restricted to the scope of two-dimensional display screens able to describe generic information, in the tradition of the [computer science](https://en.wikipedia.org/wiki/Computer_science" \o "Computer science) research at the Xerox Palo Alto Research Center.

# Python GUI – tkinter

Python offers multiple options for developing GUI (Graphical User Interface). Out of all the GUI methods, tkinter is most commonly used method. It is a standard Python interface to the Tk GUI toolkit shipped with Python. Python with tkinter outputs the fastest and easiest way to create the GUI applications. Creating a GUI using tkinter is an easy task.  
**To create a tkinter:**

1. Importing the module – tkinter
2. Create the main window (container)
3. Add any number of widgets to the main window
4. Apply the event Trigger on the widgets.

Importing tkinter is same as importing any other module in the python code. Note that the name of the module in Python 2.x is ‘Tkinter’ and in Python 3.x is ‘tkinter’.

import tkinter

There are two main methods used you the user need to remember while creating the Python application with GUI.

1. **Tk(screenName=None,  baseName=None,  className=’Tk’,  useTk=1):** To create a main window, tkinter offers a method ‘Tk(screenName=None,  baseName=None,  className=’Tk’,  useTk=1)’. To change the name of the window, you can change the className to the desired one. The basic code used to create the main window of the application is:

m=tkinter.Tk() where m is the name of the main window object

1. **mainloop():** There is a method known by the name mainloop() is used when you are ready for the application to run. mainloop() is an infinite loop used to run the application, wait for an event to occur and process the event till the window is not closed.

m.mainloop()

* Openpyxl Library-

openpyxl is a Python library to read/write Excel 2010 xlsx/xlsm/xltx/xltm files. It was born from lack of existing library to read/write natively from Python the Office Open XML format. All kudos to the PHPExcel team as openpyxl was initially based on PHPExcel.

Source Code-

----------------------------------------------------------------------------

from openpyxl import \*

from tkinter import \*

wb = load\_workbook('C:\\Users\\Nim\_Ish\\Desktop\\Book1.xlsx')

sheet = wb.active

def excel():

sheet.column\_dimensions['A'].width = 30

sheet.column\_dimensions['B'].width = 20

sheet.column\_dimensions['C'].width = 40

sheet.column\_dimensions['D'].width = 50

sheet.cell(row=1, column=1).value = "Name"

sheet.cell(row=1, column=2).value = "Contact Nmber"

sheet.cell(row=1, column=3).value = "Email id"

sheet.cell(row=1, column=4).value = "Address"

def clear():

name\_field.delete(0, END)

contact\_no\_field.delete(0, END)

email\_id\_field.delete(0, END)

address\_field.delete(0, END)

def insert():

if (name\_field.get() == "" and

contact\_no\_field.get() == "" and

email\_id\_field.get() == "" and

address\_field.get() == ""):

print("empty input")

else:

current\_row = sheet.max\_row

current\_column = sheet.max\_column

sheet.cell(row=current\_row + 1, column=1).value = name\_field.get()

sheet.cell(row=current\_row + 1, column=2).value = contact\_no\_field.get()

sheet.cell(row=current\_row + 1, column=3).value = email\_id\_field.get()

sheet.cell(row=current\_row + 1, column=4).value = address\_field.get()

wb.save('C:\\Users\\Nim\_Ish\\Desktop\\Book1.xlsx')

name\_field.focus\_set()

clear()

if \_\_name\_\_ == "\_\_main\_\_":

root = Tk()

root.configure(background='grey')

root.title("registration form")

root.geometry("500x300")

excel()

heading = Label(root, text="Form", bg="grey")

name = Label(root, text="Name", bg="grey")

contact\_no = Label(root, text="Contact No.", bg="grey")

email\_id = Label(root, text="Email id", bg="grey")

address = Label(root, text="Address", bg="grey")

heading.grid(row=0, column=1)

name.grid(row=1, column=0)

contact\_no.grid(row=2, column=0)

email\_id.grid(row=3, column=0)

address.grid(row=4, column=0)

name\_field = Entry(root)

contact\_no\_field = Entry(root)

email\_id\_field = Entry(root)

address\_field = Entry(root)

name\_field.grid(row=1, column=1, ipadx="100")

contact\_no\_field.grid(row=2, column=1, ipadx="100")

email\_id\_field.grid(row=3, column=1, ipadx="100")

address\_field.grid(row=4, column=1, ipadx="100")

excel()

submit = Button(root, text="Submit", fg="Black",

bg="Red", command=insert)

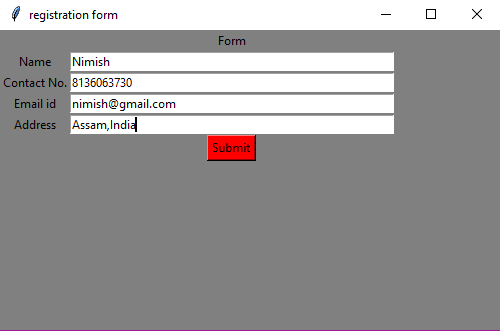
submit.grid(row=8, column=1)

# start the GUI

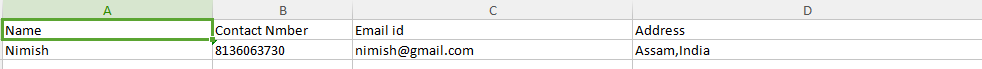
root.mainloop()

----------------------------------------------------------------------------

Input and Output-



The Registration Form for data entry.



The Data gets Entered in the Excel file.