Tkinter
Introduction to GUI programming
Tkinter(Tk interface) is the standard Python interface to the Tk-GUI toolkit. Tk & Tkinter are available on all platforms. Graphical User Interface allows a user to interact with a computer program using a pointing device that manipulates small pictures on a computer screen.

GUI Programming in Python:
There are many ways to develop GUI based programs in Python. Top 5 Python GUI Libraries:
1.Tkinter
2. PyQT5
3. PySide
4. Kivy
5. wxPython

Tkinter module:
It is the default Graphical User Interface library for Python. Python once combined with Tkinter provides a quick and simple graphical program applications. The first graphical user interface toolkit is Tk, Tkinter stands for Tk interface.

Tkinter in Python helps in creating GUI Applications with a minimized steps of coding. Tkinter is the only library that is built-in into Python's Standard Library. It is cross-platform, we can write same code on any platform like Windows, Unix, Linux, Mac etc. It is a lightweight, simple and efficient module.

Tk, TCL & Tkinter
Tool Command Language. Tk is very popular thus it has been ported to a variety of other scripting languages, including Perl, Ruby and Python. Tkinter provides us a faster and efficient way in order to build useful apps .

Setup/Configure Tkinter
Explicitly no need to install Tkinter. It is standard library in Python.

Example:
```
import tkinter as tk
abc = tk.Tk() #abc indicates name of the main window object
abc.mainloop()# It is used to start the application
```

Example:
```
from tkinter import * #It import all of the funtions and varibales from tkinter package
abc=Tk() #tk constructor method
Label(abc,text="Hello PYTHON GUI").pack() #Geometry management method
abc.mainloop()
```

PYTHON Tkinter Components / Widgets:

1 Button	2 Canvas	3 Checkbutton		
4 Entry	5 Frame	6 Label		
7 Listbox	8 Menubutton	9 Menu		
10 Message	11 Radiobutton	12 Scale		
13 Scrollbar	14 Text	15 Spinbox	16 tkMessageBox	

Tkinter Button
It is used to add buttons in a Python applications. These buttons are display different components like text or images that convey the purpose of the buttons.

Syntax
```
w = Button ( master, option=value, ... )
```

Example:
```
from tkinter import *
x = Tk()
x.geometry("250x150")
but = Button(x,text = "SampleButton")
but.pack()
x.mainloop()
```

Example:
```
import tkinter
from tkinter import *
from tkinter import messagebox
x = Tk()
x.geometry("350x200")
def click():
    messagebox.showinfo("Hey", "Clicked on Green Button")
Btn1 = Button(x, text="YELLOW", pady=10)
Btn2 = Button(x, text="BLUE", pady=10)
# Adding click Event to the following Button
Btn3 = Button(x, text="GREEN", command=click,pady=10)
Btn1.pack(side = LEFT)
Btn2.pack(side = RIGHT)
Btn3.pack(side = TOP)
x.mainloop()
```

Example:
```
from tkinter import *
from tkinter import ttk
root=Tk()
button=ttk.Button(root,text="Click ME.!!")
button.pack()
def callback():
        print("ClickME")
button.config(command=callback)
Ctrl+Z (Exit)
button.invoke()
button.state(['disabled'])
button.instate(['disabled'])
button.state(['!disabled'])
lg=PhotoImage(file='E:\HTML_Tutorials\HTML5\Images\Butterfly.gif')
button.config(image=lg,compound=LEFT)
slogo=lg.subsample(3,3)
button.config(image=slogo)
```

Python Tkinter Radio-Button:
This widget is used to implement one-of-many selection in the Python applications. It shows multiple choices to the user out of which, the user can select only one out of them. We can associate different

methods with each of the radio-button.

We can display the multiple line text or images on the radio-buttons. To keep track the user's selection the radio-button, it is associated with a single variable. Each button shows a single value for that particular variable.

Syntax
w = Radiobutton(top, options)

Example:
```
from tkinter import *
from tkinter.ttk import *
win = Tk()
win.geometry("250x250")
var = StringVar(win)
options = {" Option_1" : "A",
                "Option_2" : "B",
                "Option_3" : "C",
                "Option_4" : "D"
                }
for (txtt, vall) in options.items():
        Radiobutton(win,text=txtt, variable=var,
value=vall).pack(side=TOP, ipady = 5)
mainloop()
```

Python Tkinter Checkbutton
The Check-Button is used to track the user's choices provided to the application. In other words, we can say that Check-Button is used to implement the on/off selections. It is a standard widget in Tkinter library..

The Checkbutton can contain the text or images. The Checkbutton is mostly used to provide many choices to the user among which, the user needs to choose the one. It generally implements many of many selections.

Syntax
w = checkbutton(master, options)

Example:
```
from tkinter import *
top = Tk()
top.geometry("100x100")
check1 = IntVar()
check2 = IntVar()
check3 = IntVar()
ckbtn1 = Checkbutton(top, text = "PYTHON", variable = check1)
ckbtn2 = Checkbutton(top, text = "CYTHON", variable = check2)
ckbtn3 = Checkbutton(top, text = "JYTHON", variable = check3)
ckbtn1.pack()
ckbtn2.pack()
ckbtn3.pack()
top.mainloop()
```

ComboBox:
It is another widget from tkinter to select a single item at a time

from group of items. It is popularly known as drop down box. It is standard widget in Python Tkinter library.

Syntax:
```
w = Combobox(master, option=value, ...)
```

Example:
```
from tkinter import *
from tkinter import ttk
x = Tk()
x.geometry("150x130")
fframe = Frame(x)
fframe.pack()
vvlist = ["PYTHON", "Machine Learning", "Data Science","CYTHON",
"Django"]
Drop=ttk.Combobox(fframe, values = vvlist)
Drop.set("Select Required Tech.")
Drop.pack(padx = 5, pady = 5)
x.mainloop()
```

List Box:
It is another tkinter widget, It is used to display a list of options for the user to select any number of items at a time.
Syntax:
```
w = Listbox ( master, option, ... )
```

Example:
```
from tkinter import *
x = Tk()
x.geometry("180x200")
frame = Frame(x)
frame.pack()
label = Label(x,text = "Drop of Technologies.")
label.pack()
lbox = Listbox(x)
lbox.insert(1,"PYTHON")
lbox.insert(2, "CYTHON")
lbox.insert(3, "JYTHON")
lbox.insert(4, "IRONPYTHON")
lbox.insert(5, "RUBYPYTHON")
lbox.insert(6, "PYJAVASCRIPT")
lbox.pack()
x.mainloop()
```