

```
In [3]: import tkinter
from tkinter import *
root=Tk()
label1=Label (root, text="Hello world")
label1.pack ()
root.mainloop ()
```

```
In [10]: from tkinter import *
root=Tk()
frm1=Frame(root)
frm1.pack(side=TOP)
frm2=Frame(root)
frm2.pack(side=BOTTOM)

#to add button1 on   frame1 syntax is given.. fg=foreground colour/text colour
b1=Button(frm1,text="Click Here",fg="RED")
b2=Button(frm2,text="Press on this",fg="BLUE")

# to add buttons on respective frames .pack() function is used
b1.pack()
b2.pack()
root.mainloop()
```

```
In [11]: from tkinter import *
root=Tk()
l1=Label (root, text="Frist Name")
l2=Label (root, text="Last Name")
entry1=Entry (root)
entry2=Entry (root)
l1.grid (row=0, column=0)
l2.grid (row=1, column=0)
entry1.grid (row=0, column=1)
entry2.grid (row=1,column=1)
root.mainloop ()
```

```
In [5]: # Tkinter: Self Adjusting Widgets
```

```
from tkinter import *
root=Tk()
label1=Label (root, text='First Name', bg='YELLOW', fg='RED')

label1.pack (side=TOP, fill=X)
label2=Label (root, text='Second Name', fg='BLUE', bg='GREEN')

label2.pack (side=LEFT, fill=Y)
root.mainloop ()
```

```
In [8]: from tkinter import *
root=Tk()
def function1():
    print("Hello You just clicked here")
b1=Button (root, text='click', command=function1)
b1.pack()
root.mainloop()
```

```
Hello You just clicked here
Hello You just clicked here
Hello You just clicked here
Hello You just clicked here
Hello You just clicked here
Hello You just clicked here
```

```
In [12]: from tkinter import *
root=Tk ()
class mybutton:
    def __init__(self, root):
        frm=Frame(root)
        frm.pack ()

        self.butt1=Button (frm,text="click",fg="RED",command=self.printdata)
        self.butt1.pack (sid=TOP)

        self.butt2=Button(frm,text="Exit",command=frm.quit)
        self.butt2.pack ()

    def printdata(self):
        print("Hello How r You?")

b=mybutton(root)
root.mainloop()

Hello How r You?
```

```
In [2]: from tkinter import *
root=Tk()
def fun1():
    print("You just clicked menu")
def fun2():
    print("Hi ")

mymenu=Menu(root)
submenu=Menu(mymenu)
root.config(menu=mymenu)

#They are attached to a parent menu (using add_cascade), instead of a top-level window.
mymenu.add_cascade(label="File",menu=submenu)
submenu.add_command(label="NewNotebook",command=fun1)
submenu.add_command(label="open",command=fun2)
submenu.add_separator ()

submenu.add_command(label="save",command=fun2)
newmenu=Menu(mymenu)
mymenu.add_cascade(label="edit",menu=newmenu)
newmenu.add_command(label="Undo",command=fun1)
root.mainloop()

from tkinter import *
root=Tk()
def fun1():
    print("You just clicked menu")
def fun2():
    print("Hi ")

toolbar1=Frame (root, bg="RED")
butt1=Button (toolbar1,text="print",command=fun1)
butt1.pack (side=LEFT, padx=2, pady=2)
butt2=Button(toolbar1, text="##",command=fun2)
butt2.pack(side=LEFT, padx=3, pady=3)
butt3=Button(toolbar1, text="++",command=fun2)
butt3.pack(side=LEFT, padx=3, pady=3)

toolbar1.pack(side=TOP,fill=X)
root.mainloop()
```

```
In [ ]: from tkinter import *
root=Tk()
statusbar1=Label(root, text='MITWPU', anchor=E, bd=4, relief=SUNKEN)
statusbar1.pack(side=BOTTOM, fill=X)
root.mainloop()
```

```
In [ ]: from tkinter import *
root=Tk()
statusbar1=Label(root, text='Hi This is status', anchor=W, bd=1, relief=RIDGE)
statusbar1.pack(side=BOTTOM, fill=X)
root.mainloop()
```

Devanshu Surana
1032210755
PC-12.



Python Programming Lab Assignment - 11

Problem Statement: Create a program to take input of your date of birth and output your age.

Aim: Write a python GUI program to import tkinter package and create a program to take input of your date of birth and output of your age.

Objectives: To learn and implement Python GUI using Tkinter.

Theory:

Tkinter commonly comes bundled with python, using TK and is Python's standard GUI framework. It is famous for its simplicity and graphical user interface. It's open source and available under the python license.

Steps:

- 1) You import the key component i.e the tkinter module
- 2) Initialize window manager with the tkinter, Tk() method and assign it to a variable. This method creates a blank window with close, maximize and minimize buttons on the top as a usual GUI should have.
- 3) Then as an optional step, you will rename the title of the window as you like with window title (title-of-the-window)
- 4) Make use of widget called label which is used to insert some text into the window
- 5) Make use of Tkinter's geometry management attribute called pack().
- 6) Use mainloop() to display the window.

Widgets

- Button: Has a property of switching on/off when a user clicks the button.

Syntax: `button_widget = tk.Button(widget, option = placeholder)`

- Canvas: You use it to draw shapes in GUI.

Syntax: `canvas_widget = tk.Canvas(widget, option = placeholder)`

- Check button: Check button records on-off or true-false state.

Syntax: `checkbox_widget = tk.Checkbutton(widget, option = placeholder)`

- Frame = used as containers in the Tkinter for grouping

Syntax: `frame_widget = tk.Frame(widget, option = placeholder)`

Platform: Windows / Ubuntu - Python Editor (Jupyter Notebook)

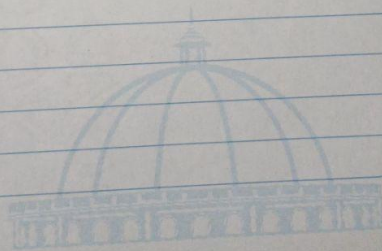
Conclusion: studied python GUI using Tkinter.

FAQ's

Ans 1) Tkinter is the standard GUI library for python. Python when combined with Tkinter provides a fast and easy way to create GUI application. Tkinter provides a powerful object-oriented interface to the TK GUI toolkit.

Ans 2) In order to create Tkinter application, we generally create an instance of Tkinter frame. i.e tk(), helps display the root window and manage all other components of Tkinter application.

Ans 3) The pack geometry manager packs widgets relative to the earlier widget. Tkinter literally packs all the widgets one after the other in a window. We can use options like fill, expands and size to control this geometry manager. The pack manager is some ~~what~~ what limited, but its much easier to use in a few situations.



```
In [ ]: from tkinter import *
root=Tk()
statusbar=Label(root, text='Hi This is status', anchor=W,bd=1,relief=RIDGE)
statusbar1.pack(side=BOTTOM, fill=X)
root.mainloop()
```

```
In [ ]: import tkinter.messagebox
tkinter.messagebox.showinfo ('title', 'message want to display')
res=tkinter.messagebox.askquestion('title',' question')
```

```
In [ ]: from tkinter import *
import tkinter.messagebox
root=Tk()
tkinter.messagebox.showinfo ('title', 'This is awesome')
res=tkinter.messagebox.askquestion('Que.1','Do you want Tea?')
if res=='yes':
    print("Here is tea")
else:
    print("Ok")
    root.mainloop()
```

```
In [ ]: from tkinter import *
master=Tk()
root=Tk()
cn1=Canvas (master, width=300, height=300, bg='YELLOW', bd=3)
cn1.pack ()
n1=cn1.create_line (0, 0,100,100)
n2=cn1.create_line (50, 50, 50, 100, fill='RED')
n3=cn1.create_rectangle (100, 100, 200, 200, fill='PINK')
root.mainloop ()
```

```
In [1]: from tkinter import *

root = Tk()

root.title("TEST")
root.geometry("400x400")

def submitf():
    a=ue.get()
    b=pe.get()
    if a == b:
        printl = Label(root,text="Unlocked")
        printl.grid(row =3,column=1)
    else:
        printe = Label(root,text="Wrong pw")
        printe.grid(row =3,column=1)

user = Label(root, text="User Id")
user.grid(row=0,column=0,padx=20)

pw = Label(root, text="Password")
pw.grid(row=1,column=0,padx=20)

ue = Entry(root,width=30)
ue.grid(row=0,column=1,padx=5)

pe = Entry(root,width=30)
pe.grid(row=1,column=1,padx=5)

submit_btn = Button(root, text="Submit", command=submitf)
submit_btn.grid(row=2,column=0,columnspan=2)

root.mainloop()
```

```
In [ ]:
```