Harsh Kasliwal
2003085
C21

EXPERIMENT 9

Aim: Python program to demonstrate use of Tkinter interface module. Create a GUI based application using widgets Entry, Label, Text, Button, RadioButton, CheckButton, ListBox, Menu, Spinbox (any five).

THEORY:

Tkinter Programming

Tkinter is the standard GUI library for Python. Python when combined with Tkinter provides a fast and easy way to create GUI applications. Tkinter provides a powerful object-oriented interface to the Tk GUI toolkit.

Creating a GUI application using Tkinter is an easy task. All you need to do is perform the following steps −

- Import the *Tkinter* module.
- Create the GUI application main window.
- Add one or more of the above-mentioned widgets to the GUI application.
- Enter the main event loop to take action against each event triggered by the user.

Tkinter Widgets

Tkinter provides various controls, such as buttons, labels and text boxes used in a GUI application. These controls are commonly called widgets.

There are currently 15 types of widgets in Tkinter. We present these widgets as well as a brief description in the following table −

Sr.No.	Operator & Description
1	Button

	The Button widget is used to display buttons in your application.
2	Canvas The Canvas widget is used to draw shapes, such as lines, ovals, polygons and rectangles, in your applica
3	Checkbutton The Checkbutton widget is used to display a number of options as checkboxes. The user can select multip at a time.
4	Entry The Entry widget is used to display a single-line text field for accepting values from a user.
5	Frame The Frame widget is used as a container widget to organize other widgets.
6	Label The Label widget is used to provide a single-line caption for other widgets. It can also contain images.
7	Listbox The Listbox widget is used to provide a list of options to a user.
8	Menubutton The Menubutton widget is used to display menus in your application.
9	Menu The Menu widget is used to provide various commands to a user. These commands are contain Menubutton.
10	Message The Message widget is used to display multiline text fields for accepting values from a user.
11	Radiobutton The Radiobutton widget is used to display a number of options as radio buttons. The user can select only at a time.
12	Scale The Scale widget is used to provide a slider widget.
13	Scrollbar

	The Scrollbar widget is used to add scrolling capability to various widgets, such as list boxes.
14	Text The Text widget is used to display text in multiple lines.
15	Toplevel The Toplevel widget is used to provide a separate window container.
16	Spinbox The Spinbox widget is a variant of the standard Tkinter Entry widget, which can be used to select fro number of values.
17	PanedWindow A PanedWindow is a container widget that may contain any number of panes, arranged horizontally or v
18	LabelFrame A labelframe is a simple container widget. Its primary purpose is to act as a spacer or container for comple layouts.
19	tkMessageBox This module is used to display message boxes in your applications.

L

CODE:

```
import datetime
import tkinter as tk
window=tk.Tk()
window.geometry("500x500")
window.title(" Age Calculator App ")
name = tk.Label(text = "Name")
name.grid(column=0,row=1)
year = tk.Label(text = "Year")
year.grid(column=0,row=2)
month = tk.Label(text = "Month")
month.grid(column=0,row=3)
date = tk.Label(text = "Date")
date.grid(column=0,row=4)
nameEntry = tk.Entry()
nameEntry.grid(column=1,row=1)
yearEntry = tk.Entry()
yearEntry.grid(column=1,row=2)
monthEntry = tk.Entry()
monthEntry.grid(column=1,row=3)
dateEntry = tk.Entry()
dateEntry.grid(column=1,row=4)
def getInput():
    name=nameEntry.get()
    monkey =
Person(name,datetime.date(int(yearEntry.get()),int(monthEntry.get()),
int(dateEntry.get())))

    textArea = tk.Text(master=window,height=10,width=25)
    textArea.grid(column=1,row=6)
    answer = " Heyy {monkey}!. You are {age} years
old;)".format(monkey=name, age=monkey.age())
    textArea.insert(tk.END,answer)
button=tk.Button(window,text="Calculate
Age",command=getInput,bg="pink")
button.grid(column=1,row=5)
```

```
class Person:
    def __init__(self,name,birthdate):
        self.name = name
        self.birthdate = birthdate
    def age(self):
        today = datetime.date.today()
        age = today.year-self.birthdate.year
        return age
window.mainloop()
```

OUTPUT: