Python, Day 9: Excursion into GUIs with tkinter

Andrew Bydlon

January 25, 2019

What are GUIs and how do I make them?

GUI, shortened from **Graphical User Interface**, is as it sounds: a graphically oriented window where information more than just text can be displayed to the user.

There exist many methods to create GUIs from within python:

1. tkinter: Python's "standard" GUI library. What we will focus on today.
2. PyQt: Python with the popular QT open source library implemented.
3. WxPython: Brings wxWidgets over from C++ to Python and allows their use.

There are many more:
`https://wiki.python.org/moin/GuiProgramming`

What is Tkinter?

Tkinter is a module in the standard Python Library. This allows us to load it directly with a simple import command:

```
import tkinter
from tkinter import *
```

Syntax (Usual tkinter syntax)

```
class MyFirstGUI:
    def __init__(self,master):
        self.master = master
        master.title("Title Name")
        OtherCommands

root = Tk()
MyGUI = MyFirstGUI(root)
root.mainloop()
```

What kind of OtherCommands are available? Buttons

Buttons are used to display a button widget to the user. They can yield different results upon pressing that are accessible as functions.

Syntax

```
⋮
    self.MyButton = Button(master, text="Displayed on the Button", \
        command=self.FunctionToRun)
    self.MyButton.pack()
```

Example (Quitting a program)

```
⋮
self.MyExit = Button(master, text="Close", command=self.master.quit)
self.MyExit.pack()
```

Note: master is where the button is being stored.

Labels

Labels are used to display text or images to the user.

Example

```
self.label = Label(master, text="Some information to be displayed")
self.label.pack()
```

You can also use the PhotoImage command to import a picture for display.

Example (A picture)

```
⋮
self.newlabel = Label(self.master, image=PhotoImage(file='/file/path'))
self.newlabel.pack()
```

Frames

Sometimes it can be beneficial to place information within a particular area, or **frame**. This widget helps to organize your program.

Syntax (Frames)

self.MyFrame = Frame(master, width=1000, height=1000, background='white')
 # self.MyFrame.pack_propagate(0)
self.MyFrame.pack()

The propogate command can be used to keep the size of the frame fixed instead of dependent on it's children.

Example (Putting a label in a frame)

```
self.MyFrame = Frame(self.master, width=1000, height=1000)
self.MyFrame.pack()
self.label = Label(self.MyFrame, text="Some information")
self.label.pack()
```

Menu Buttons

Menu buttons are used to display multiple choice type answers to a give problem. The syntax is as follows:

Syntax (Menu Buttons)

w = Menubutton(master, option, ...)

And here is how it works in practice:

Example

```
self.mb = Menubutton(master, text="Condiments", relief=RAISED)
self.mb.menu = Menu(self.mb, tearoff=0)
self.mb["menu"] = self.mb.menu
mayoVar = IntVar()
ketchVar = IntVar()
self.mb.menu.add_checkbutton(label="Mayo", variable=mayoVar)
self.mb.menu.add_checkbutton(label="Ketchup", variable=ketchVar)
self.mb.pack()
```

Let's check out the implementation of all of them

Combining the code of the above slides into one package yields the following:

- The master window has a title of A simple GUI.
- This is our first GUI is a Label.
- Greet and Close are Buttons with corresponding functions attached to them if pushed.
- The large white space is an empty Frame with background set to white.
- condiments is a menu button, which when clicked allows you to select Mayo or Ketchup.

Practice I

Try to get a GUI of your own up and running, with the various objects implemented. Try in particular moving objects from master to a frame of your choice.

Note: You can try further to simply write the GUI not in a function call if you wish. It is good practice.