GUI

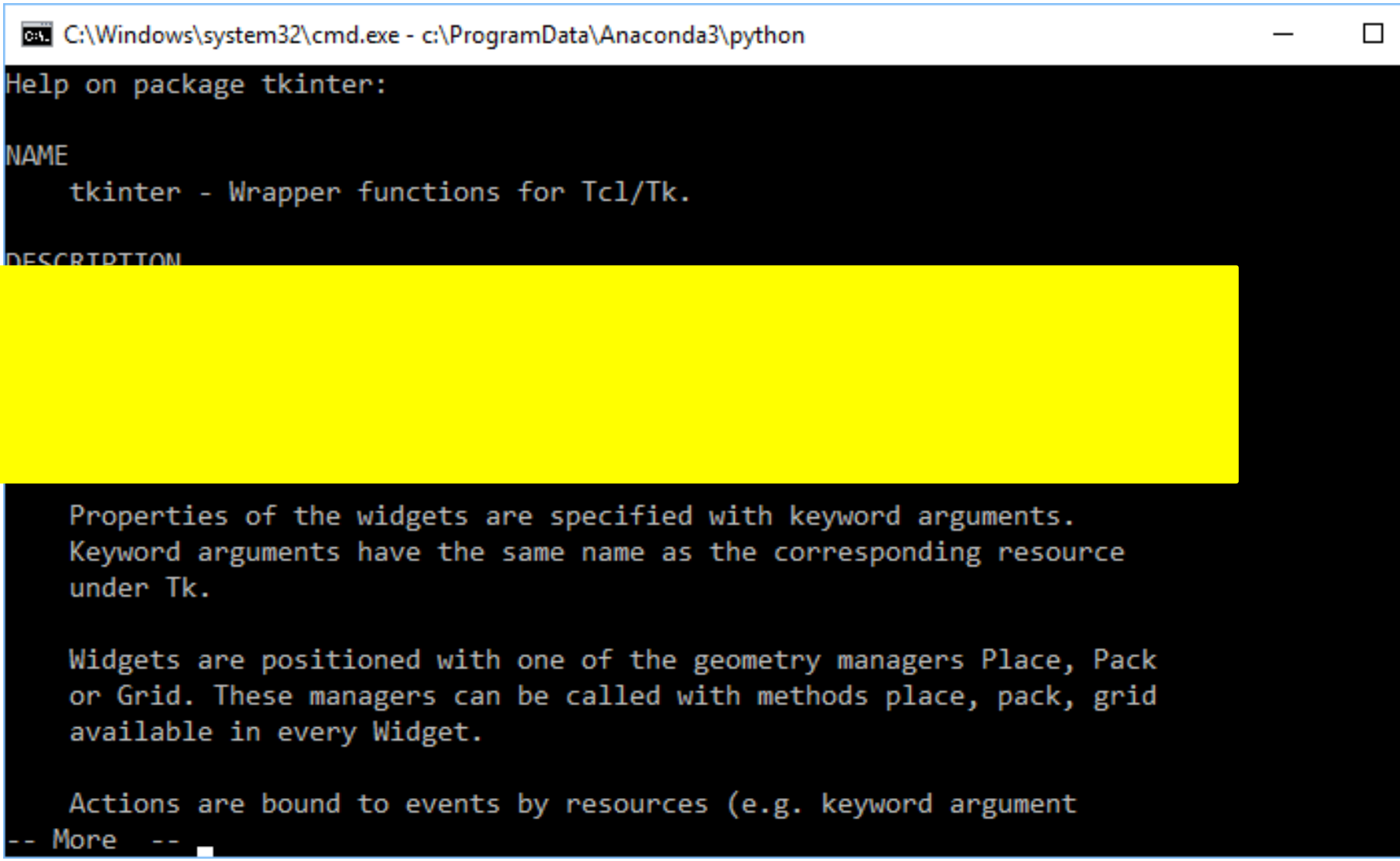
- Import modules

```
from tkinter import *
```

  - The module `tkinter` has submodules

- List the modules contained in tkinter.
  - From `help()> modules` in command-line python

Package tkinter



Classes in tkinter

```
CLASSES
    builtins.object
        CallWrapper
        Event
        Grid
        Image
            BitmapImage
            PhotoImage
        Misc
            BaseWidget
                Toplevel(BaseWidget, Wm)
                Widget(BaseWidget, Pack, Place, Grid)
                    Button
                    Canvas(Widget, XView, YView)
                    Checkbutton
                    Entry(Widget, XView)
                    Frame
                    Label
                    LabelFrame
                    Listbox(Widget, XView, YView)
                    Menu
                    Menubutton
                        OptionMenu
                    Message
                    PanedWindow
                    Radiobutton
                    Scale
                    Scrollbar
                    Spinbox(Widget, XView)
                    Text(Widget, XView, YView)
                Tk(Misc, Wm)
            Pack
            Place
            Variable
                BooleanVar
                DoubleVar
                IntVar
                StringVar
            Wm
            XView
            YView
    builtins.str(builtins.object)
        EventType(builtins.str, enum.Enum)
    enum.Enum(builtins.object)
        EventType(builtins.str, enum.Enum)
```

Try to show one simple GUI

```
import tkinter
from tkinter.constants import *
tk = tkinter.Tk()
frame = tkinter.Frame(tk, relief=RIDGE, borderwidth=2)
frame.pack(fill=BOTH,expand=1)
label = tkinter.Label(frame, text="Hello, World")
label.pack(fill=X, expand=1)
button = tkinter.Button(frame,text="Exit",command=tk.destroy)
button.pack(side=BOTTOM)
tk.mainloop()
```

- Make sure first it works on your computer
  - Save it in a file
  - Run it on your command-line
- Read the Button library and add a couple of functions to the code
  - Ex) .bell() and one more?

Or, you may simply run it on IDE pyzo

Let’s try to learn GUI

GUI

- import a module
- Look up the API of “Python tkinter” from google
  - What is Tk()?
    - **Toplevel widget** of Tk which represents mostly the main windows of an application.
    - It has an **Tcl interpreter**.
- Any functions associated with a Tk() object?
  - .bell()
  - .mainloop()
  - .destroy()
  - Functions inherited from Wm:
    - .title()
    - .geometry()
- What is the function, `.mainloop()`?
  - Executing the main event handler

Please check out the web: https://docs.python.org/3/library/tk.html

GUI

- On a Tk() object,
- What is Canvas()?
  - Any functions?
    - .create_rectangle()
    - .create_polygon()
    - .create_text()
    - .pack()
- What about Frame()?
  - An functions?
    - .pack()

Please check out the web: https://docs.python.org/3/library/tk.html

First GUI Exercise

- Canvas
  - Graphics on Canvas
- Frames, later
  - Menus on Frames
- Canvas and Frames can work together!

Canvas Widgets

Note:
can = Canvas (Tk(), x, y)

```
can.create_line(0, 100, 200, 0, fill="red", dash=(4, 4))

tmp = can.create_rectangle(50, 25, 150, 75, fill="blue")

can.delete(tmp) # remove

can.delete(ALL) # remove all items

can.create_rectangle(50, 20, 150, 80, fill="#476042")
can.create_rectangle(65, 35, 135, 65, fill="yellow")

can.create_oval(50,50,100,100)


points = [100, 140, 110, 110, 140, 100, 110, 90, 100, 60, 90,
90, 60, 100, 90, 110]

can.create_polygon(points, outline=python_green,
            fill='yellow', width=3)
```

Canvas Widgets

Note:
can = Canvas (Tk(), x, y)

```
coord = 10, 50, 240, 210
arc = can.create_arc(coord, start=0, extent=150, fill="blue")



img = PhotoImage(file="rocks.ppm")
can.create_image(20,20, anchor=NW, image=img)
```

Try how this code works

```
from tkinter import *

root = Tk() # constructor: an object of Tk is constructed and labeled by "root"
________ # Add a title here
can = Canvas(root)
# constructor of Canvas: construct an object of Canvas
________ # Set width=700 and height=500 on background with hex code #aa88ff
can.create_rectangle(10,10, 100,100, fill="#ff7711")
# x,y coordinators of the upper left cornder and lower right
________ # Create another rectangle with(100,100, 500,250)
can.create_oval(70,70, 130,130)
________ # Create another circle with (320,220, 380,280)
can.pack()
root.mainloop()
```

- Run first without the lines ___
- Then, code to add the requests in the lines ___

- Hint:
  - Look up the library or aka API

Drawer

```
from tkinter import *

canvas_width = 500
canvas_height = 150

def paint( event ):
   python_green = "#476042"
   x1, y1 = ( event.x - 1 ), ( event.y - 1 )
   x2, y2 = ( event.x + 1 ), ( event.y + 1 )
   w.create_oval( x1, y1, x2, y2, fill = python_green )

master = Tk()
master.title( "Painting using Ovals" )
w = Canvas(master,
           width=canvas_width,
           height=canvas_height)
w.pack(expand = YES, fill = BOTH)
w.bind( "<B1-Motion>", paint )

message = Label( master, text = "Press and Drag the mouse to draw"
message.pack( side = BOTTOM )

mainloop()
```

Homework 3

- Choose a pet, e.g., dog, elephant, bird, etc, in your mind
- Simplify it to draw using
  - Any available functions
  - On a canvas

- Evaluation
  - Based on voting
  - Each student votes for
    - 3 products in the top tier
    - 3 products in the second tier
  - Those who receive top tier vote can earn 5 points, and those who receive second tier vote earn 3 points.
  - Student who votes for a person who is finally selected as the final top tier will also earn 3 points, and student who votes for a person who is finally selected as the second top tier will earn 1 point.