**Name: Session:**

**Programming I**

**Lab Exercise 9.29.2023**

**Second Taste of Graphics in Python**

When it comes to created graphics in Python, there are a lot of choices. Some of the choices available to you are:

Tkinter

Pygame

Pyglet

PyCairo

wxPython

PyQt

Turtle

PyGTK

For our purpose we will stick with Tkinter and Pygame since they are the most mature and widely used. Tkinter is pretty much “old school” whereas PyGame is a bit more modern and the general consensus is that it is “awesome”.

Here are two examples that will contrast the two:

##### Tkinter example ########

from tkinter import \*

window = Tk()

w = Canvas(window, width=200, height=100)

w.pack()

w.create\_line(0, 0, 200, 100)

w.create\_line(0, 100, 200, 0, fill="red", dash=(4, 4))

w.create\_rectangle(50, 25, 150, 75, fill="blue")

mainloop()

##### PyGame example ########

import sys

#import and init pygame

import pygame

pygame.init()

#create the screen

window = pygame.display.set\_mode((640, 480))

#draw a line

pygame.draw.line(window, (255, 255, 255), (0, 0), (30, 50))

#draw it to the screen

pygame.display.flip()

#input handling (boilerplate code):

while True:

for event in pygame.event.get():

if event.type == pygame.QUIT:

sys.exit(0)

else:

print (event)

One thing you will notice is that the PyGame example does not close its graphic screen gracefully. To correct this we will modify the event handler for the QUIT event as follows:

for event in pygame.event.get():

if event.type == pygame.QUIT:

pygame.display.quit()

sys.exit(0)

else:

print (event)

1. On Ada copy the Lab Exercise 10.21.2020 folder to your MyPython folder on your desktop. The path to the folder is:

\\ADA\Data Files\Programming I

1. The folder contains 14 files [1 graphic (beach.png) and 13 Python (graphics1.py through graphics13.py)]. You are to open each file in IDLE and run it. Write a one sentence description of what it does.

graphics1.py \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

graphics2.py \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

graphics3.py \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

graphics4.py \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

graphics5.py \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

graphics6.py \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

graphics7.py \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

graphics8.py \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

graphics9.py \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

graphics10.py \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

graphics11.py \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

graphics12.py \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

graphics13.py \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_