Lab Report of Python Programming

Lab 7: GUI Programming Credit hour: 2

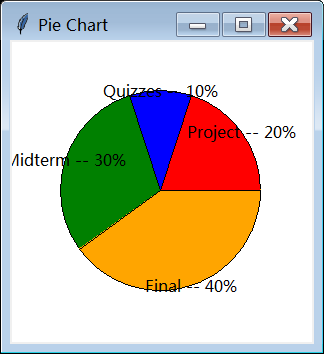
Student Name: 徐政辉 Student ID:2017329621139

## ￣￣￣￣￣￣￣￣￣￣￣￣￣￣￣￣￣￣￣￣￣￣￣￣￣￣￣￣￣￣￣￣￣￣

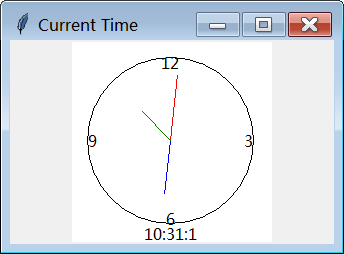
1. Objective

Know how to create a GUI application with Tkinter.

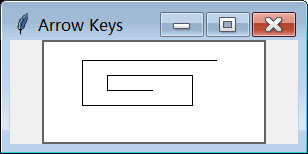
1. Lab content
   1. (Display a pie chart) Write a program that uses a pie chart to display the percentages of the overall grade represented by the project, quizzes, the midterm exam, and the final exam. Suppose that project is weighted as 20 percent of the grade and is displayed in red, quizzes are 10 percent and are displayed in blue, the midterm exam is 30 percent and is displayed in green, and the final exam is 40 percent and is displayed in orange.



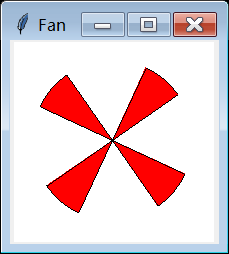
* 1. (Display a clock) Write a program that displays a clock to show the current time. To obtain the current time, use the datetime class.



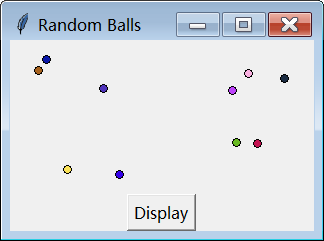
* 1. (Draw lines using the arrow keys) Write a program that draws line segments using the arrow keys. The line starts from the center of the frame and draws toward east, north, west, or south when the Right arrow key, Up arrow key, Left arrow key, or Down arrow key is clicked.



* 1. (Display a running fan) Write a program that displays a fan running.



* 1. (Display balls with random colors) Write a program that displays ten balls with random colors and placed at random locations.



1. Code list

2.1.py

from tkinter import \*

window=Tk()

canvas=Canvas(window,width=400,height=400)

canvas.pack()

canvas.create\_arc(10,10,390,390,start=0,extent=360\*0.2,fill="red",tags="Project")

canvas.create\_arc(10,10,390,390,start=360\*0.2,extent=360\*0.1,fill="blue",tags="Quizzes")

canvas.create\_arc(10,10,390,390,start=360\*0.3,extent=360\*0.3,fill="green",tags="Midterm")

canvas.create\_arc(10,10,390,390,start=360\*0.6,extent=360\*0.4,fill="yellow",tags="Final")

canvas.create\_text(300,160,text="Project--20%")

canvas.create\_text(200,40,text="Quizzes--10%")

canvas.create\_text(80,190,text="Midterm--30%")

canvas.create\_text(200,300,text="Final--40%")

window.mainloop()

2.2.py

import time

from datetime import datetime

from tkinter import \*

import math

def get\_time():

hour=datetime.now().hour

minute=datetime.now().minute

second=datetime.now().second

return hour,minute,second

hour,minute,second=get\_time()

print(hour,minute,second)

window=Tk()

window.title("current time")

canvas=Canvas(window,width=380,height=400)

canvas.pack()

canvas.create\_oval(10,10,370,370)

canvas.create\_text(20,185,text="9",font= "time 10 bold")

canvas.create\_text(185,20,text="12",font= "time 10 bold")

canvas.create\_text(185,360,text="6",font= "time 10 bold")

canvas.create\_text(360,185,text="3",font= "time 10 bold")

theta1=2\*math.pi-math.pi\*abs(hour-12)/6-math.pi/6\*minute/60-2\*math.pi/360\*second/60

print(theta1\*180/math.pi)

theta2=2\*math.pi-math.pi\*minute/30-2\*math.pi/60\*second/60

theta3=2\*math.pi-math.pi\*second/30

x,y=0,80

def transform(x,y,theta):

x1=x\*math.cos(theta)-y\*math.sin(theta)

y1=x\*math.sin(theta)+y\*math.cos(theta)

return x1,y1

x1,y1=transform(x, y, theta1)

x,y=0,110

x2,y2=transform(x,y,theta2)

x,y=0,150

x3,y3=transform(x,y,theta3)

canvas.create\_line(185,185,x1+185,185-y1,fill="green")

canvas.create\_line(185,185,x2+185,185-y2,fill="blue")

canvas.create\_line(185,185,x3+185,185-y3,fill="red")

canvas.create\_text(180,385,text="%d:%d:%d"%(hour,minute,second),font = "time 14 bold")

window.mainloop()

2.3.py

from tkinter import \*

class keyArrow:

def \_\_init\_\_(self):

self.x=200

self.y=200

self.window = Tk()

self.window.title("Arrow Key")

self.canvas=Canvas(self.window,width=400,height=400)

self.canvas.pack()

self.canvas.bind("<Key>",self.keyevent)

self.canvas.focus\_set()

self.window.mainloop()

def keyevent(self,event):

print(event.keysym)

if event.keysym=="Up":

self.canvas.create\_line(self.x,self.y,self.x,self.y-5)

self.y-=5

elif event.keysym=="Down":

self.canvas.create\_line(self.x,self.y,self.x,self.y+5)

self.y+=5

elif event.keysym=="Left":

self.canvas.create\_line(self.x,self.y,self.x-5,self.y)

self.x-=5

elif event.keysym=="Right":

self.canvas.create\_line(self.x,self.y,self.x+5,self.y)

self.x+=5

keyArrow()

2.4.py

from tkinter import \*

import time

theta=0

window=Tk()

canvas=Canvas(window,width=400,height=400)

canvas.pack()

while 1:

theta+=1

arc1=canvas.create\_arc(10,10,390,390,start=theta,extent=30,fill="red")

arc2=canvas.create\_arc(10,10,390,390,start=theta+90,extent=30,fill="red")

arc3=canvas.create\_arc(10,10,390,390,start=theta+180,extent=30,fill="red")

arc4=canvas.create\_arc(10,10,390,390,start=theta+270,extent=30,fill="red")

canvas.update()

time.sleep(0.01)

canvas.delete(arc1)

canvas.delete(arc2)

canvas.delete(arc3)

canvas.delete(arc4)

window.mainloop()

2.5.py

from tkinter import \*

import random

class displayball:

def \_\_init\_\_(self):

self.window=Tk()

self.window.title("Random Balls")

self.width=600

self.height=400

self.canvas=Canvas(self.window,width=self.width,height=self.height)

self.canvas.pack()

self.display()

self.button=Button(self.window,text="display",command=self.deleteball)

self.button.pack()

self.window.mainloop()

def randomcolor(self):

color=""

code=['0','1','2','3','4','5','6','7','8','9','A','B','C','D','E','F']

for i in range(6):

color+=code[random.randint(0,15)]

return "#"+color

def display(self):

x1,y1=random.randint(0,600),random.randint(0,400)

self.ball1=self.canvas.create\_oval(x1,y1,x1+8,y1+8,fill=self.randomcolor())

x2,y2=random.randint(0,600),random.randint(0,400)

self.ball2=self.canvas.create\_oval(x2,y2,x2+8,y2+8,fill=self.randomcolor())

x3,y3=random.randint(0,600),random.randint(0,400)

self.ball3=self.canvas.create\_oval(x3,y3,x3+8,y3+8,fill=self.randomcolor())

x4,y4=random.randint(0,600),random.randint(0,400)

self.ball4=self.canvas.create\_oval(x4,y4,x4+8,y4+8,fill=self.randomcolor())

x5,y5=random.randint(0,600),random.randint(0,400)

self.ball5=self.canvas.create\_oval(x5,y5,x5+8,y5+8,fill=self.randomcolor())

x6,y6=random.randint(0,600),random.randint(0,400)

self.ball6=self.canvas.create\_oval(x6,y6,x6+8,y6+8,fill=self.randomcolor())

x7,y7=random.randint(0,600),random.randint(0,400)

self.ball7=self.canvas.create\_oval(x7,y7,x7+8,y7+8,fill=self.randomcolor())

x8,y8=random.randint(0,600),random.randint(0,400)

self.ball8=self.canvas.create\_oval(x8,y8,x8+8,y8+8,fill=self.randomcolor())

x9,y9=random.randint(0,600),random.randint(0,400)

self.ball9=self.canvas.create\_oval(x9,y9,x9+8,y9+8,fill=self.randomcolor())

x10,y10=random.randint(0,600),random.randint(0,400)

self.ball10=self.canvas.create\_oval(x10,y10,x10+8,y10+8,fill=self.randomcolor())

def deleteball(self):

self.canvas.delete(self.ball1)

self.canvas.delete(self.ball2)

self.canvas.delete(self.ball3)

self.canvas.delete(self.ball4)

self.canvas.delete(self.ball5)

self.canvas.delete(self.ball6)

self.canvas.delete(self.ball7)

self.canvas.delete(self.ball8)

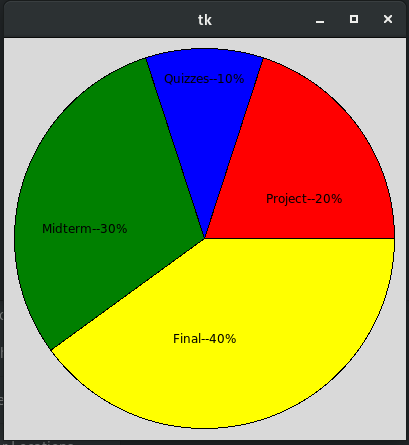
self.canvas.delete(self.ball9)

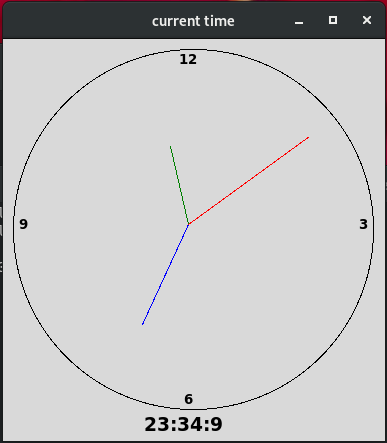
self.canvas.delete(self.ball10)

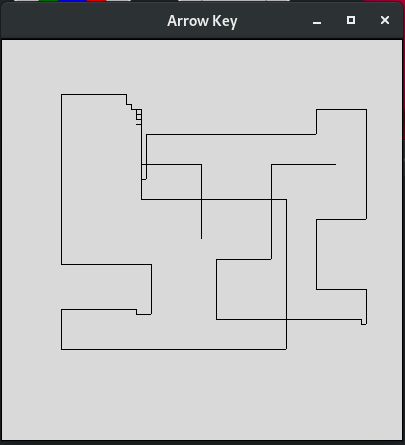
self.display()

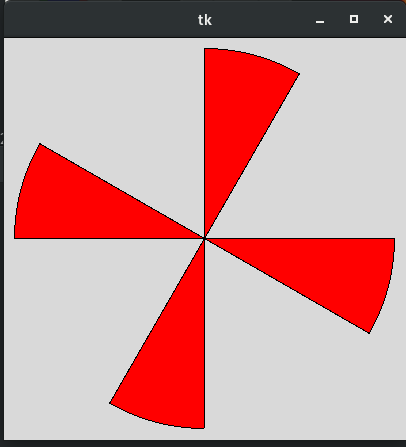
displayball()

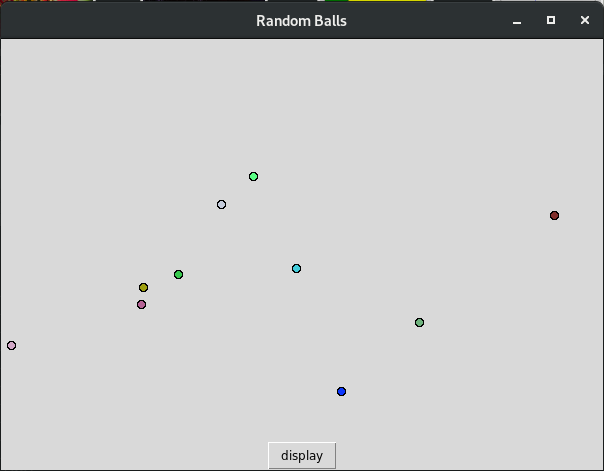
1. Output











1. Analysis and conclusions

This lab work is very simple but its very interesting that I can use the code to get a lot of GUI program and maybe some of them is very useful I my later study or life. This lab told me how to handle the main operation of tkinter and review the content that I learned before. It’s meaningful and its easy.