Roll NO:2019BTECS00022

Name: Prachi R. Chobhare

Topic: Python GUI and Multitheading

Problem no:1

from tkinter import \*

def click(num):

global val

val=val+str(num)

data.set(val)

def clear():

global val

val=""

data.set("")

def equal():

global val

result=str(eval(val))

data.set(result)

cal = Tk()

cal.title("Calculator")

cal.geometry("361x381+500+200")

val=""

data=StringVar()

display= Entry(cal,textvariable=data,bd=29,justify="right",bg="violet",font=("arial",20))

display.grid(row=0,columnspan=4)

btn7=Button(cal,text="7",bg="violet",font=("arial",12,"bold"),bd=12,height=2,width=6,command=lambda:click(7))

btn7.grid(row=1,column=0)

btn8=Button(cal,text="8",bg="violet",font=("arial",12,"bold"),bd=12,height=2,width=6,command=lambda:click(8))

btn8.grid(row=1,column=1)

btn9=Button(cal,text="9",bg="violet",font=("arial",12,"bold"),bd=12,height=2,width=6,command=lambda:click(9))

btn9.grid(row=1,column=2)

btn\_plus=Button(cal,text="+",bg="violet",font=("arial",12,"bold"),bd=12,height=2,width=6,command=lambda:click('+'))

btn\_plus.grid(row=1,column=3)

btn4=Button(cal,text="4",bg="violet",font=("arial",12,"bold"),bd=12,height=2,width=6,command=lambda:click(4))

btn4.grid(row=2,column=0)

btn5=Button(cal,text="5",bg="violet",font=("arial",12,"bold"),bd=12,height=2,width=6,command=lambda:click(5))

btn5.grid(row=2,column=1)

btn6=Button(cal,text="6",bg="violet",font=("arial",12,"bold"),bd=12,height=2,width=6,command=lambda:click(6))

btn6.grid(row=2,column=2)

btn\_sub=Button(cal,text="-",bg="violet",font=("arial",12,"bold"),bd=12,height=2,width=6,command=lambda:click('-'))

btn\_sub.grid(row=2,column=3)

btn1=Button(cal,text="1",bg="violet",font=("arial",12,"bold"),bd=12,height=2,width=6,command=lambda:click(1))

btn1.grid(row=3,column=0)

btn2=Button(cal,text="2",bg="violet",font=("arial",12,"bold"),bd=12,height=2,width=6,command=lambda:click(2))

btn2.grid(row=3,column=1)

btn3=Button(cal,text="3",bg="violet",font=("arial",12,"bold"),bd=12,height=2,width=6,command=lambda:click(3))

btn3.grid(row=3,column=2)

btn\_multi=Button(cal,text="x",bg="violet",font=("arial",12,"bold"),bd=12,height=2,width=6,command=lambda:click('\*'))

btn\_multi.grid(row=3,column=3)

btn\_clear=Button(cal,text="C",bg="violet",font=("arial",12,"bold"),bd=12,height=2,width=6,command=lambda:clear())

btn\_clear.grid(row=4,column=0)

btn0=Button(cal,text="0",bg="violet",font=("arial",12,"bold"),bd=12,height=2,width=6,command=lambda:click(0))

btn0.grid(row=4,column=1)

btn\_div=Button(cal,text="/",bg="violet",font=("arial",12,"bold"),bd=12,height=2,width=6,command=lambda:click('/'))

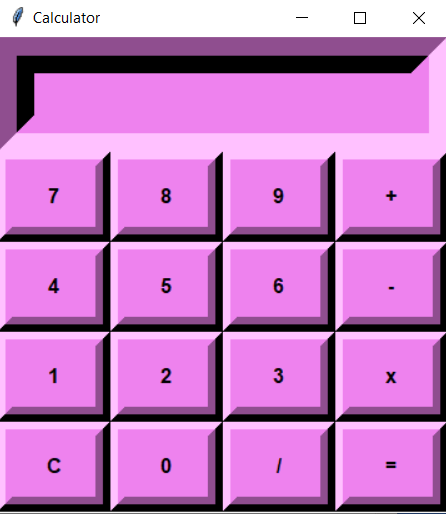
btn\_div.grid(row=4,column=2)

btn\_eq=Button(cal,text="=",bg="violet",font=("arial",12,"bold"),bd=12,height=2,width=6,command=lambda:equal())

btn\_eq.grid(row=4,column=3)

cal.mainloop()

Output:



Problem no:2

import tkinter as tk

def form\_data():

result.config(text=f'Name: {name.get()}\nAge: {n.get()}\nEmail Id: {e.get()}')

form = tk.Tk()

form.title('Register Form')

form.geometry('300x300')

tk.Label(form, text='Enter your name').pack()

name = tk.Entry(form)

name.pack()

tk.Label(form, text='Enter your age').pack()

n = tk.Entry(form)

n.pack()

tk.Label(form, text='Enter your Email').pack()

e = tk.Entry(form)

e.pack()

submit = tk.Button(form, bg='blue', text='Register', command=form\_data)

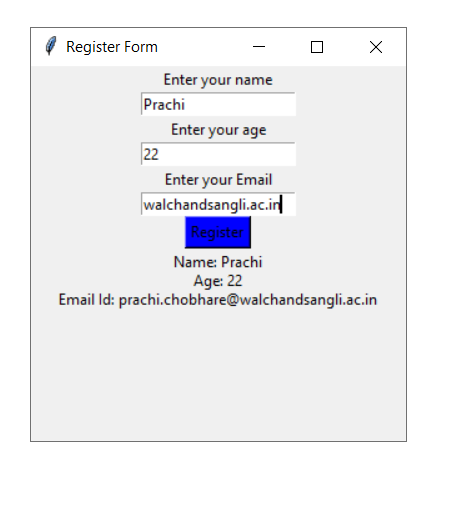
submit.pack()

result = tk.Label(form)

result.pack()

form.mainloop()

Output:



Problem no:3

from tkinter import \*

import random

from tkinter import messagebox

screen= Tk()

screen.geometry("600x600")

screen.title("Color Game")

timeleft=30

def nextcolor():

global s

global timeleft

if timeleft>0:

k.focus\_set()

if k.get().lower()==colors[1].lower():

s+=1

k.delete(0,END)

random.shuffle(colors)

text.config(fg=str(colors[1]),text=str(colors[0]))

score=Label(text="score :"+ str(s),font=("Arial",15),fg="red")

score.place(x=220,y=90)

if timeleft == 0:

messagebox.showinfo("Game over!!!! Your tme is Up and your score is (s)")

def countdown():

global timeleft

if timeleft > 0:

timeleft-=1

tm=Label(text="Timeleft :"+ str(timeleft),font=("Arial",15),fg="red")

tm.place(x=220,y=120)

tm.after(1000,countdown)

def start(event):

t2.config(text="")

if timeleft==30:

countdown()

nextcolor()

colors=['Red','blue','Green','Yellow','pink','Purple']

s=0

t=Label(screen,text='Remember! You have to type color name',font=('Aria',17),fg='red')

t.place(x=180,y=30)

t2=Label(screen,text='Please enter to start: ',font=('Aria',17),fg='red')

t2.place(x=180,y=80)

k=Entry(screen,font=('Aria',17),fg='black',width=15)

k.place(x=200,y=400)

text=Label(screen,font=('Aria',100))

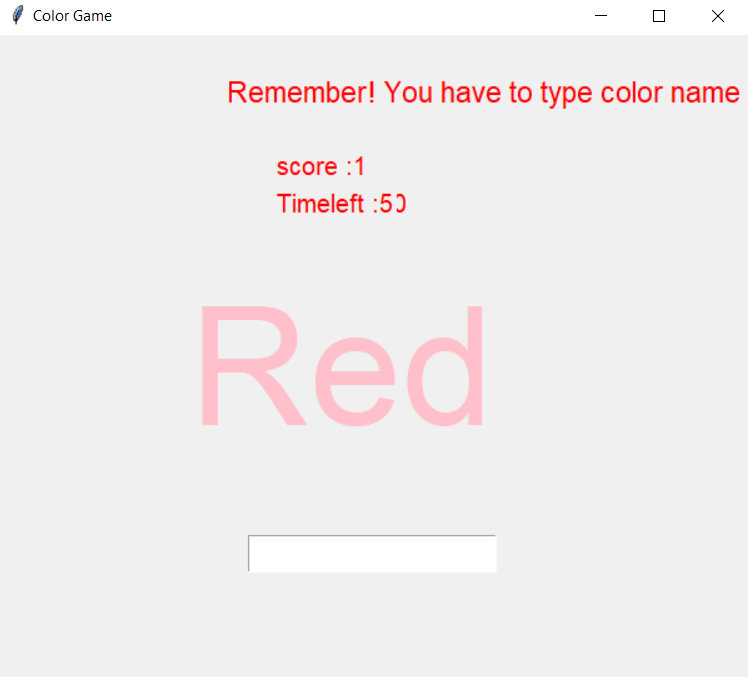
text.place(x=150,y=190)

screen.bind('<Return>',start)

k.focus\_set()

screen.mainloop

Output:



Problem no:4

from tkinter import \*

from tkinter.messagebox import showinfo

from tkinter.filedialog import askopenfilename, asksaveasfilename

import os

def newFile():

global file

notep.title("Untitle - Notepad")

file=None

TextArea.delete(1.0,END)

def openFile():

global file

file= askopenfilename(defaultextension=".txt",filetypes=[("All Files","\*.\*"),("Text Documents",".txt")])

if file== "":

file=None

else:

notep.title(os.path.basename(file)+" - Notepad")

TextArea.delete(1.0,END)

f=open(file,"r")

TextArea.insert(1.0,f.read())

f.close()

def saveFile():

global file

if file == None:

file=asksaveasfilename(initialfile ="Untitled.txt",defaultextension=".txt",filetypes=[("All Files","\*.\*"),("Text Documents",".txt")])

if file=="":

file=None

else:

f=open("file","w")

f.write(TextArea.get(1.0,END))

f.close()

notep.title(os.path.basename(file)+" - Notepad")

else:

f=open("file","w")

f.write(TextArea.get(1.0,END))

f.close()

def quitFile():

notep.destroy()

def Cut():

TextArea.event\_generate("<<Cut>>")

def Copy():

TextArea.event\_generate("<<Copy>>")

def Paste():

TextArea.event\_generate("<<Paste>>")

def about():

showinfo("Notepad" ,"Notepad By Vaishnavi")

if \_\_name\_\_ == '\_\_main\_\_':

notep=Tk()

notep.title("Untitled - Notepad")

notep.geometry("644x788")

TextArea= Text(notep,font="lucida 14")

file=None

TextArea.pack(expand=True,fill=BOTH)

Scroll=Scrollbar(TextArea)

Scroll.pack(side=RIGHT,fill=Y)

Scroll.config(command=TextArea.yview)

TextArea.config(yscrollcommand=Scroll.set)

#menu

menubar= Menu(notep)

#file

filemenu= Menu(menubar,tearoff=0)

filemenu.add\_command(label="New" ,command=newFile)

filemenu.add\_command(label="Open" ,command=openFile)

filemenu.add\_command(label="Save" ,command=saveFile)

filemenu.add\_command(label="Exit" ,command=quitFile)

menubar.add\_cascade(label="File", menu=filemenu)

#Edit

editmenu= Menu(menubar,tearoff=0)

editmenu.add\_command(label="Cut" ,command=Cut)

editmenu.add\_command(label="Copy" ,command=Copy)

editmenu.add\_command(label="Paste" ,command=Paste)

menubar.add\_cascade(label="Edit", menu=editmenu)

#help

helpmenu=Menu(menubar,tearoff=0)

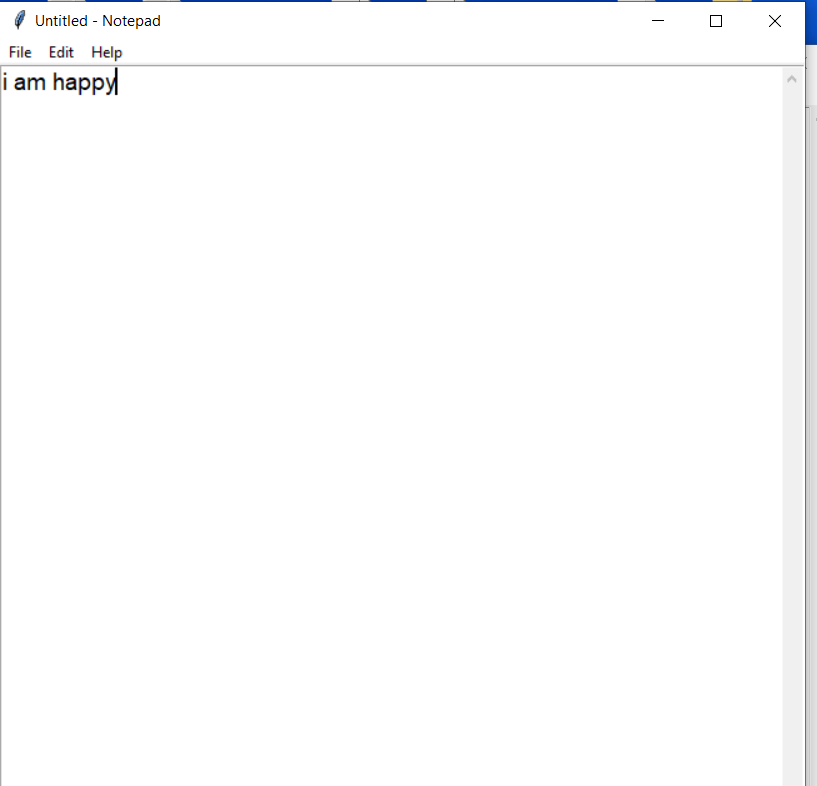
helpmenu.add\_command(label="About Notepad",command=about)

menubar.add\_cascade(label="Help", menu=helpmenu)

notep.config(menu=menubar)

notep.mainloop()

Output:



Problem no:5

import time

from tkinter import \*

clock=Tk()

clock.geometry("359x150+0+0")

clock.configure(background="black")

clock.resizable(0,0)

clock.overrideredirect(1)

def start():

text=time.strftime("%H:%M:%S")

label.config(text=text)

label.after(200,start)

label=Label(clock,font=("ds-digital",50,"bold"),bg="black",fg="red",bd=50)

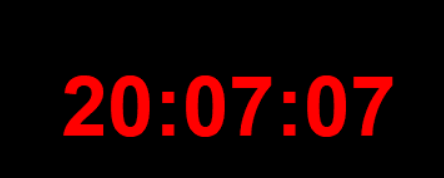
label.grid(row=0,column=1)

start()

print("done")

clock.mainloop()

Output:



Problem no:6

from tkinter import \*

from tkinter.ttk import \*

if \_\_name\_\_ == '\_\_main\_\_':

root=Tk()

#getting screen height and width in pixels

height2=root.winfo\_screenmmheight()

width2=root.winfo\_screenmmwidth()

#getting screen Depth

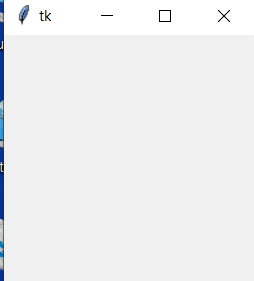
screendepth=root.winfo\_screendepth()

print("\n Width x Height = %d x %d (in mm)\n"%(width2,height2))

print("\nScreen Depth= ",screendepth)

mainloop()

Output:



Problem no:7

# python -m pip install requests

import concurrent.futures

import requests

import threading

import time

thread\_local = threading.local()

def get\_session():

if not hasattr(thread\_local, "session"):

thread\_local.session = requests.Session()

return thread\_local.session

def download\_site(url):

session = get\_session()

with session.get(url) as response:

print(f"Read {len(response.content)} from {url}")

def download\_all\_sites(sites):

with concurrent.futures.ThreadPoolExecutor(max\_workers=5) as executor:

executor.map(download\_site, sites)

if \_\_name\_\_ == "\_\_main\_\_":

sites = [

"https://www.jython.org",

"http://olympus.realpython.org/dice",

] \* 5

start\_time = time.time()

download\_all\_sites(sites)

duration = time.time() - start\_time

print(f"\nDownloaded {len(sites)} in {duration} seconds")

Output:

