Homework 9

01286121 Computer Programming

Software Engineering Program,

Department of Computer Engineering,

School of Engineering, KMITL

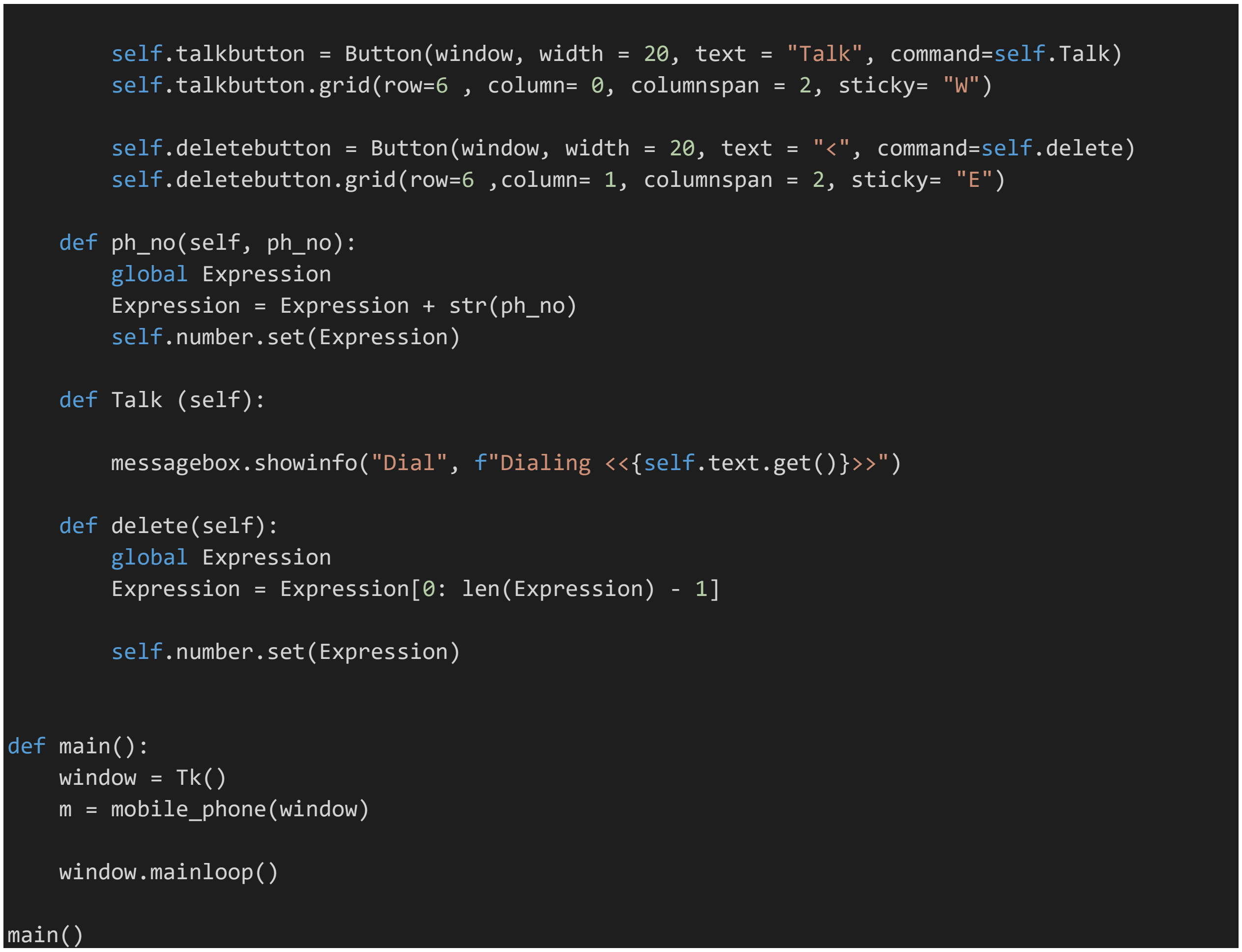
By

65011693   Soe Moe Htet

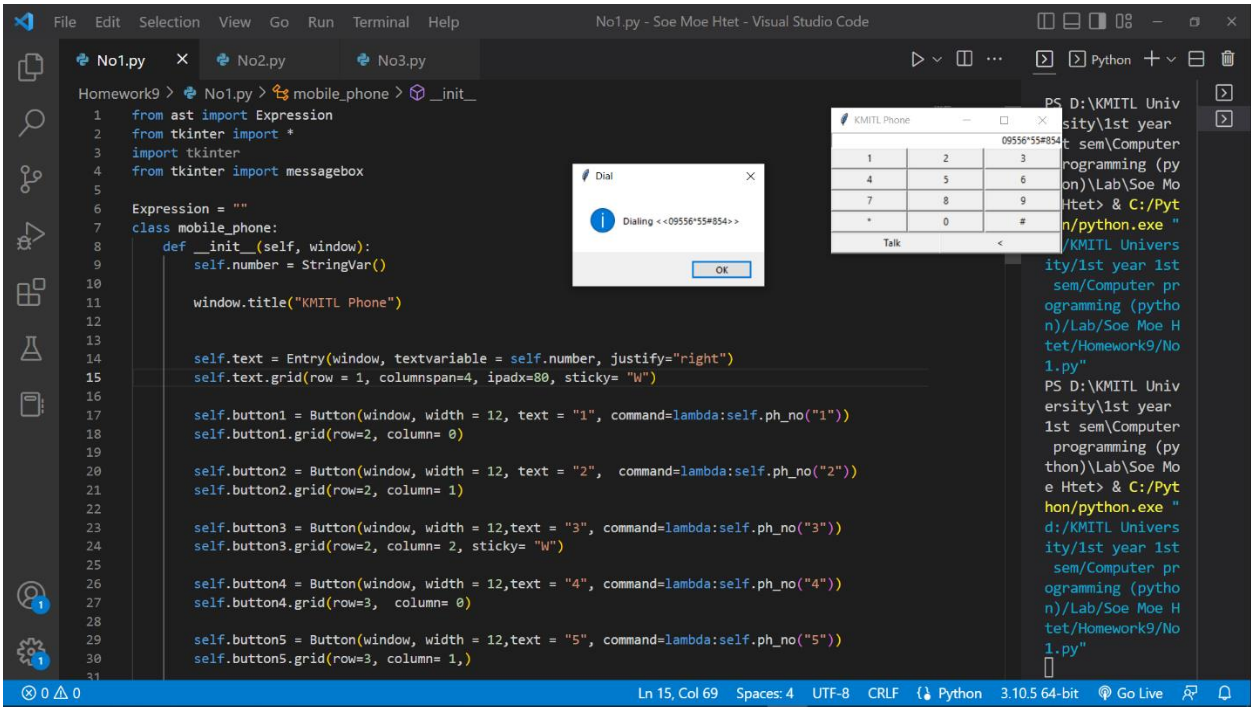
No.1

Code

```
from tkinter import *
from tkinter import messagebox

Expression = ""
class mobile_phone:
    def __init__(self, window):
        self.number = StringVar()

        window.title("KMITL Phone")


        self.text = Entry(window, textvariable = self.number, justify="right")
        self.text.grid(row = 1, columnspan=4, ipadx=80, sticky= "W")

        self.button1 = Button(window, width = 12, text = "1", command=lambda:self.ph_no("1"))
        self.button1.grid(row=2, column= 0)

        self.button2 = Button(window, width = 12, text = "2",  command=lambda:self.ph_no("2"))
        self.button2.grid(row=2, column= 1)

        self.button3 = Button(window, width = 12,text = "3", command=lambda:self.ph_no("3"))
        self.button3.grid(row=2, column= 2, sticky= "W")

        self.button4 = Button(window, width = 12,text = "4", command=lambda:self.ph_no("4"))
        self.button4.grid(row=3,  column= 0)

        self.button5 = Button(window, width = 12,text = "5", command=lambda:self.ph_no("5"))
        self.button5.grid(row=3, column= 1,)

        self.button6 = Button(window, width = 12,text = "6", command=lambda:self.ph_no("6"))
        self.button6.grid(row=3, column= 2,sticky= "W")

        self.button7 = Button(window, width = 12,text = "7", command=lambda:self.ph_no("7"))
        self.button7.grid(row=4, column= 0, )

        self.button8 = Button(window, width = 12,text = "8", command=lambda:self.ph_no("8"))
        self.button8.grid(row=4, column= 1, )

        self.button9 = Button(window, width = 12,text = "9", command=lambda:self.ph_no("9"))
        self.button9.grid(row=4, column=2, sticky= "W")

        self.button10 = Button(window, width = 12,text = "*", command=lambda:self.ph_no("*"))
        self.button10.grid(row=5, column= 0, )

        self.button0 = Button(window, width = 12,text = "0", command=lambda:self.ph_no("0"))
        self.button0.grid(row=5, column= 1, )

        self.button11 = Button(window, width = 12,text = "#", command=lambda:self.ph_no("#"))
        self.button11.grid(row=5, column=2, sticky= "E")
```



```
        self.talkbutton = Button(window, width = 20, text = "Talk", command=self.Talk)
        self.talkbutton.grid(row=6 , column= 0, columnspan = 2, sticky= "W")

        self.deletebutton = Button(window, width = 20, text = "<", command=self.delete)
        self.deletebutton.grid(row=6 ,column= 1, columnspan = 2, sticky= "E")

    def ph_no(self, ph_no):
        global Expression
        Expression = Expression + str(ph_no)
        self.number.set(Expression)

    def Talk (self):

        messagebox.showinfo("Dial", f"Dialing <<{self.text.get()}>>")

    def delete(self):
        global Expression
        Expression = Expression[0: len(Expression) - 1]

        self.number.set(Expression)


def main():
    window = Tk()
    m = mobile_phone(window)

    window.mainloop()

main()
```



```
from ast import Expression
from tkinter import *
import tkinter
from tkinter import messagebox

Expression = ""
class mobile_phone:
    def __init__(self, window):
        self.number = StringVar()

        window.title("KMITL Phone")


        self.text = Entry(window, textvariable = self.number, justify="right")
        self.text.grid(row = 1, columnspan=4, ipadx=80, sticky= "W")

        self.button1 = Button(window, width = 12, text = "1", command=lambda:self.ph_no("1"))
        self.button1.grid(row=2, column= 0)

        self.button2 = Button(window, width = 12, text = "2",  command=lambda:self.ph_no("2"))
        self.button2.grid(row=2, column= 1)

        self.button3 = Button(window, width = 12,text = "3", command=lambda:self.ph_no("3"))
        self.button3.grid(row=2, column= 2, sticky= "W")

        self.button4 = Button(window, width = 12,text = "4", command=lambda:self.ph_no("4"))
        self.button4.grid(row=3,  column= 0)

        self.button5 = Button(window, width = 12,text = "5", command=lambda:self.ph_no("5"))
        self.button5.grid(row=3, column= 1,)
```

09556*55#854

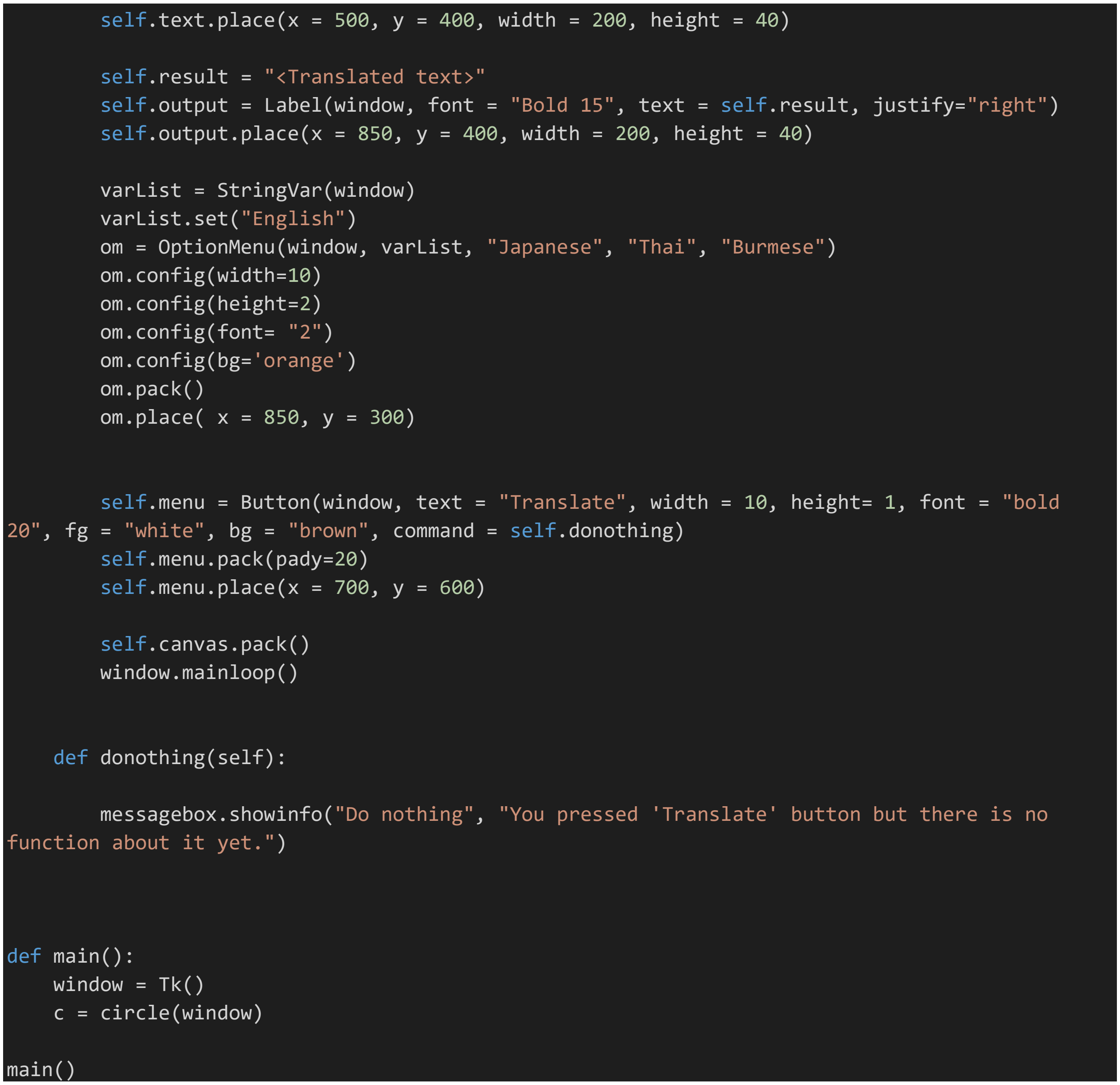
Dialing <<09556*55#854>>

```
PS D:\KMITL Univ
sity\1st year
t sem\Computer
rogramming (py
on)\Lab\Soe Mo
Htet> & C:/Pyt
n/python.exe "
/KMITL Univers
ity/1st year 1st
 sem/Computer pr
ogramming (pytho
n)/Lab/Soe Moe H
tet/Homework9/No
1.py"
PS D:\KMITL Univ
ersity\1st year
1st sem\Computer
 programming (py
thon)\Lab\Soe Mo
e Htet> & C:/Pyt
hon/python.exe "
d:/KMITL Univers
ity/1st year 1st
 sem/Computer pr
ogramming (pytho
n)/Lab/Soe Moe H
tet/Homework9/No
1.py"
```

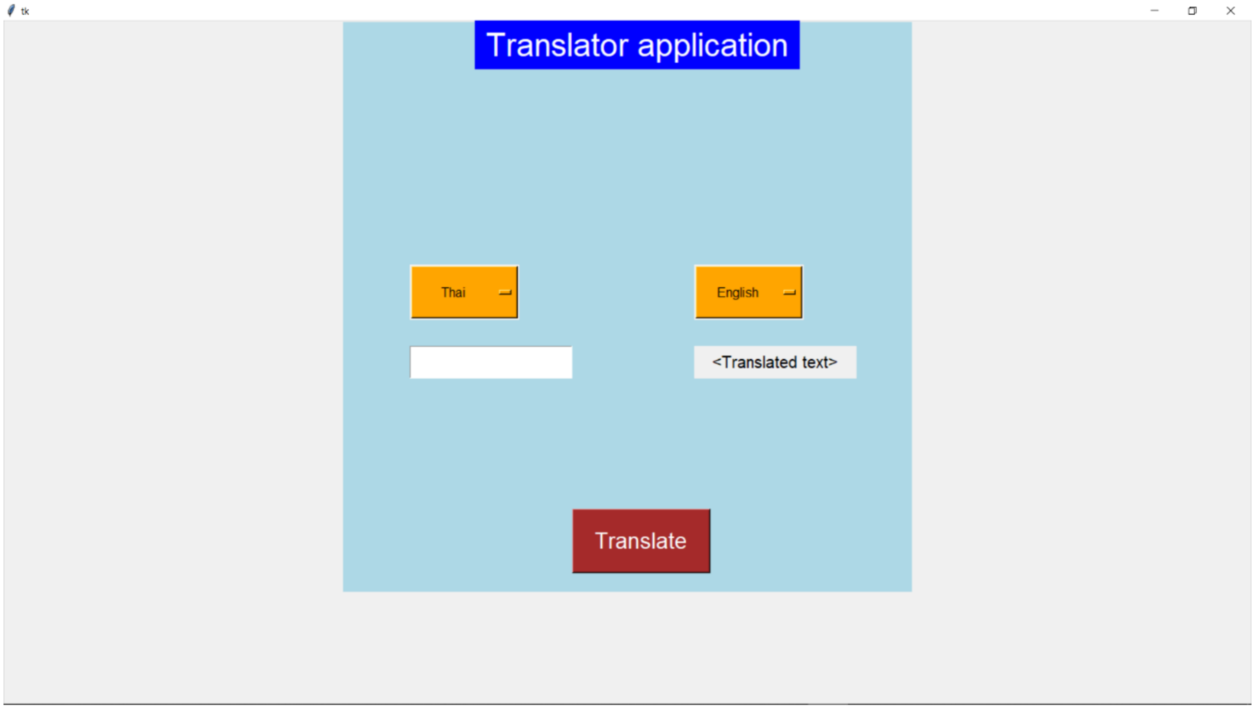
No.2

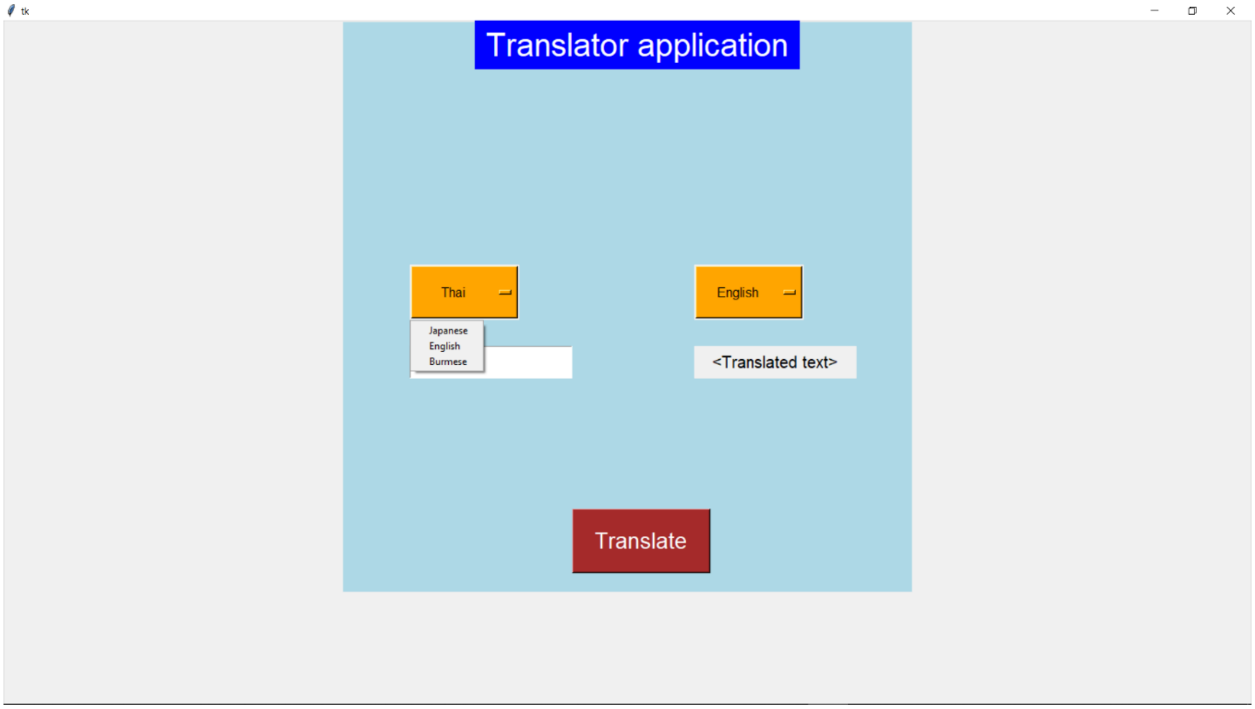
Code

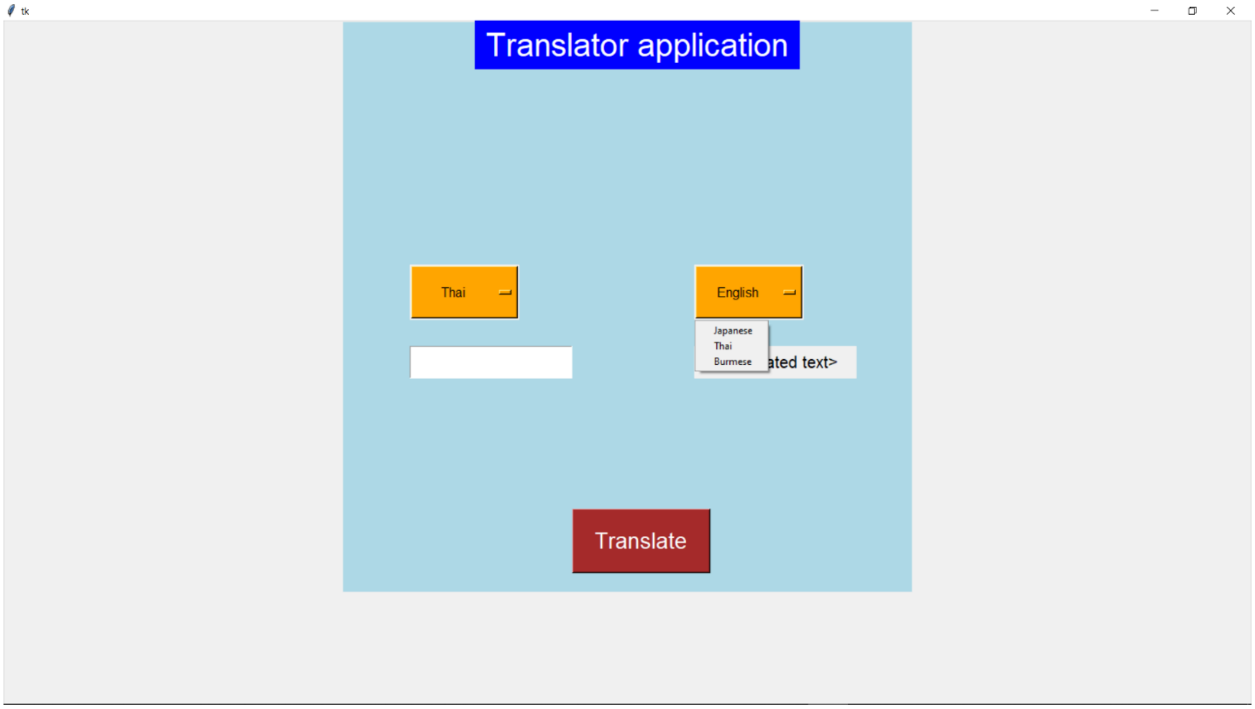
```
from tkinter import *
from tkinter import messagebox
class circle:
    def __init__(self, window):
        self.window = window

        self.canvas = Canvas(
            window,
            width = 700,
            height = 700,
            bg = "light blue"
            )

        self.title = Label(window, width = 8, height= 1, text = "Translator application", font
= "Bold 30", bg = "blue", fg='#ffffff')
        self.title.place(x = 580, y = 0, width = 400, height = 60)




        # menubar = Menu(window)
        # filemenu = Menu(menubar, tearoff=0)
        # filemenu.add_command(label="New",           command = self.donothing)
        # filemenu.add_command(label="Open",          command = self.donothing)
        # filemenu.add_command(label="Save",          command = self.donothing)
        # filemenu.add_command(label="Save as...",    command = self.donothing)
        # filemenu.add_command(label="Close",         command = self.donothing)

      # filemenu.add_separator()

      # filemenu.add_command(label="Exit", command=window.quit)
      # menubar.add_cascade(label="File", menu=filemenu)

        # menubar.place(x = 300, y = 0)
        varList = StringVar(window)
        varList.set("Thai")
        om = OptionMenu(window, varList, "Japanese", "English", "Burmese")
        om.config(width=10)
        om.config(height=2)
        om.config(font= "2")
        om.config(bg='orange')
        om.pack()
        om.place( x = 500, y = 300)

        self.number = StringVar()


        self.text = Entry(window, font = "Bold 20", textvariable = self.number,
justify="right")
```

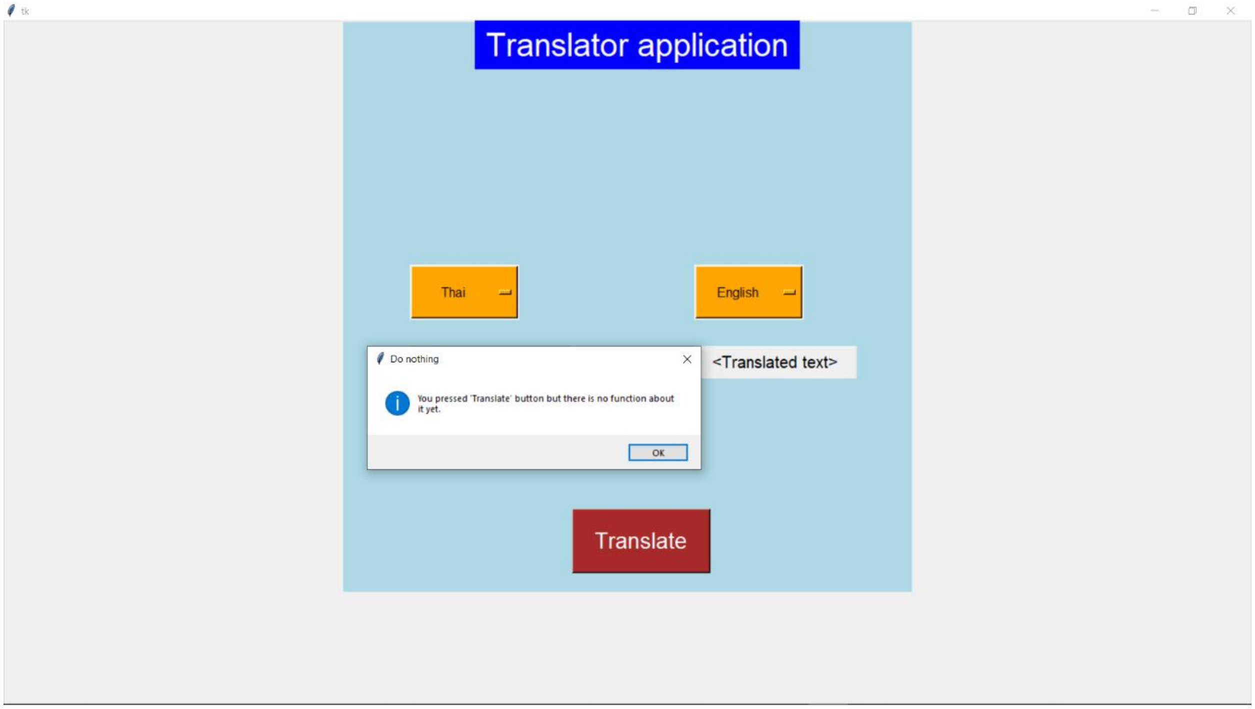


```
        self.text.place(x = 500, y = 400, width = 200, height = 40)

        self.result = "<Translated text>"
        self.output = Label(window, font = "Bold 15", text = self.result, justify="right")
        self.output.place(x = 850, y = 400, width = 200, height = 40)

        varList = StringVar(window)
        varList.set("English")
        om = OptionMenu(window, varList, "Japanese", "Thai", "Burmese")
        om.config(width=10)
        om.config(height=2)
        om.config(font= "2")
        om.config(bg='orange')
        om.pack()
        om.place( x = 850, y = 300)


        self.menu = Button(window, text = "Translate", width = 10, height= 1, font = "bold 
20", fg = "white", bg = "brown", command = self.donothing)
        self.menu.pack(pady=20)
        self.menu.place(x = 700, y = 600)

        self.canvas.pack()
        window.mainloop()


    def donothing(self):

        messagebox.showinfo("Do nothing", "You pressed 'Translate' button but there is no 
function about it yet.")



def main():
    window = Tk()
    c = circle(window)

main()
```


tk
Translator application
Thai
English
<Translated text>
Translate


tk
Translator application
Thai
English
Japanese
English
Burmese
<Translated text>
Translate



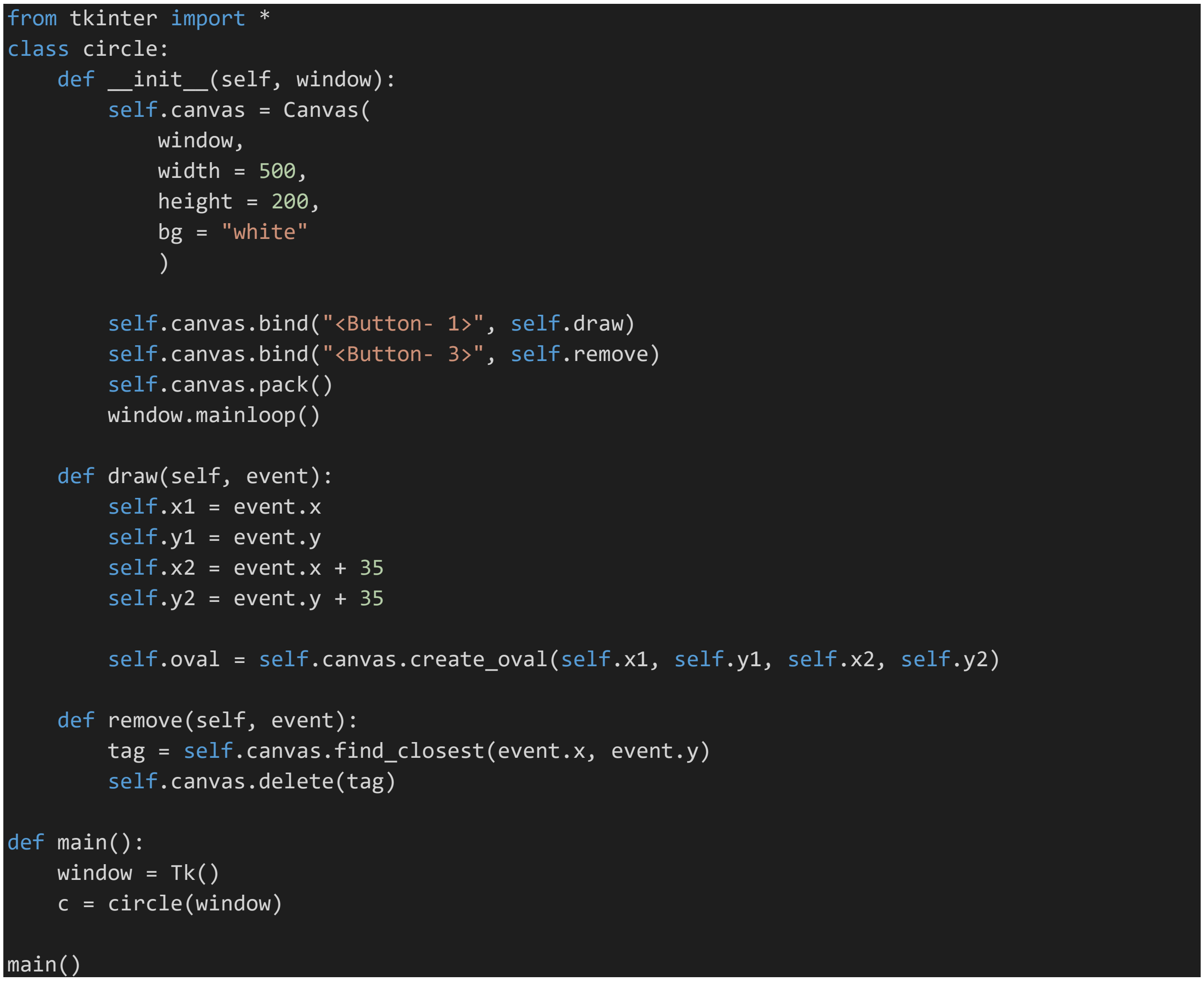


Do nothing

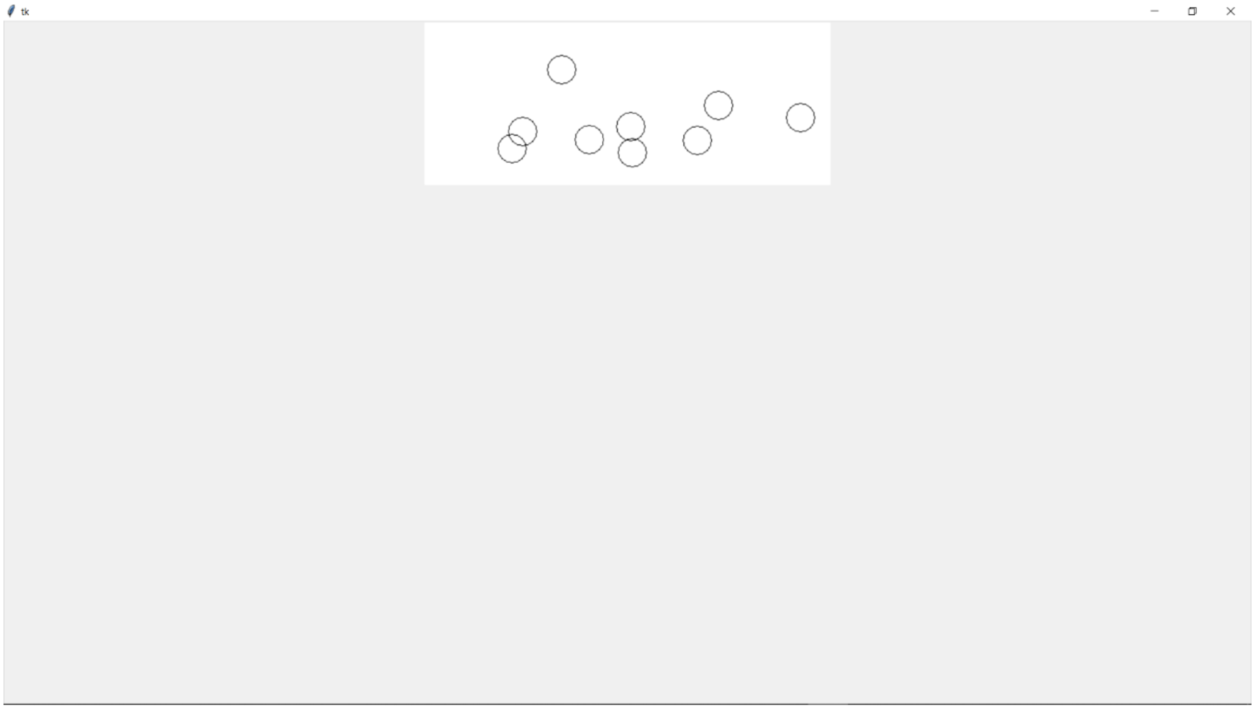
You pressed 'Translate' button but there is no function about it yet.

No.3

Code



```
from tkinter import *
class circle:
    def __init__(self, window):
        self.canvas = Canvas(
            window,
            width = 500,
            height = 200,
            bg = "white"
            )

        self.canvas.bind("<Button- 1>", self.draw)
        self.canvas.bind("<Button- 3>", self.remove)
        self.canvas.pack()
        window.mainloop()

    def draw(self, event):
        self.x1 = event.x
        self.y1 = event.y
        self.x2 = event.x + 35
        self.y2 = event.y + 35

        self.oval = self.canvas.create_oval(self.x1, self.y1, self.x2, self.y2)

    def remove(self, event):
        tag = self.canvas.find_closest(event.x, event.y)
        self.canvas.delete(tag)

def main():
    window = Tk()
    c = circle(window)

main()
```


tk