Homework #9

01286121 Computer Programming

Software Engineering Program,

Department of Computer Engineering,

School of Engineering, KMITL

By

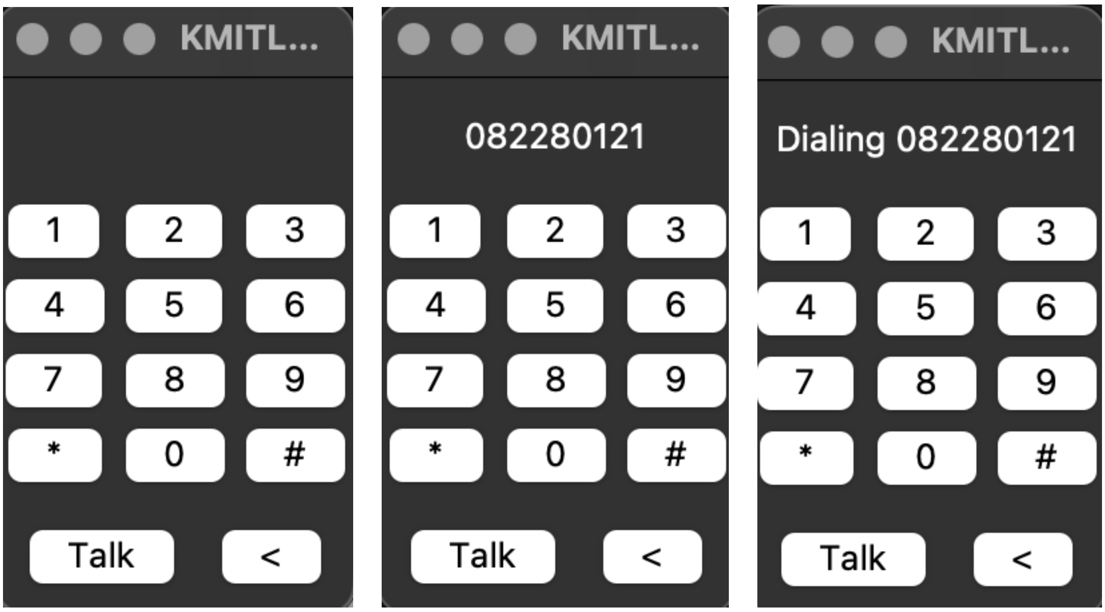
68011278 Ananda Stallard

1.

Code:

```
from tkinter import *

class KMITL_Phone:
    def __init__(self):
        window = Tk()
        window.title("KMITL Phone")

        self.message = StringVar()

        frame1 = Frame(window)
        frame1.grid(row=1, column=1, pady=10)
        Label(frame1, textvariable=self.message).pack()

        frame2 = Frame(window)
        frame2.grid(row=2, column=1)
        Button(frame2, text="1", command=lambda: self.add_digit("1")).grid(row=1, column=1)
        Button(frame2, text="2", command=lambda: self.add_digit("2")).grid(row=1, column=2)
        Button(frame2, text="3", command=lambda: self.add_digit("3")).grid(row=1, column=3)
        Button(frame2, text="4", command=lambda: self.add_digit("4")).grid(row=2, column=1)
        Button(frame2, text="5", command=lambda: self.add_digit("5")).grid(row=2, column=2)
        Button(frame2, text="6", command=lambda: self.add_digit("6")).grid(row=2, column=3)
        Button(frame2, text="7", command=lambda: self.add_digit("7")).grid(row=3, column=1)
        Button(frame2, text="8", command=lambda: self.add_digit("8")).grid(row=3, column=2)
        Button(frame2, text="9", command=lambda: self.add_digit("9")).grid(row=3, column=3)
        Button(frame2, text="*", command=lambda: self.add_digit("*")).grid(row=4, column=1)
        Button(frame2, text="0", command=lambda: self.add_digit("0")).grid(row=4, column=2)
        Button(frame2, text="#", command=lambda: self.add_digit("#")).grid(row=4, column=3)

        frame3 = Frame(window)
        frame3.grid(row=3, column=1, pady=10)
        Button(frame3, text="Talk", command=self.talk).grid(row=1, column=1, padx=5)
        Button(frame3, text="<", command=self.backspace).grid(row=1, column=2, padx=5)

        window.mainloop()

    def add_digit(self, digit):
        self.message.set(self.message.get() + digit)

    def backspace(self):
        self.message.set(self.message.get()[:-1])

    def talk(self):
        self.message.set("Dialing " + self.message.get())

KMITL_Phone()
```

Result:



082280121

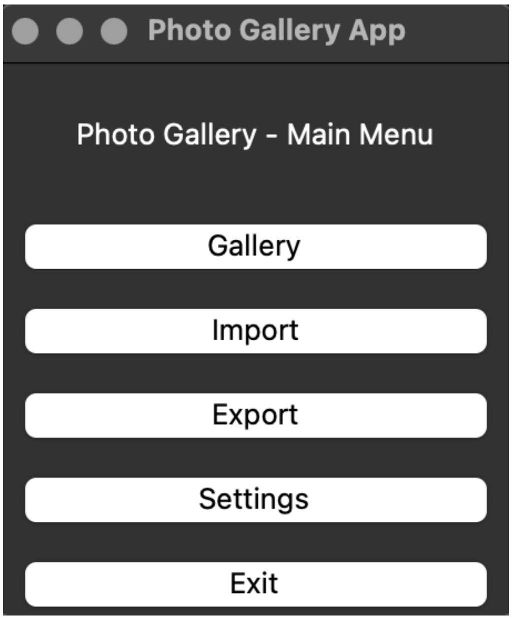
Dialing 082280121

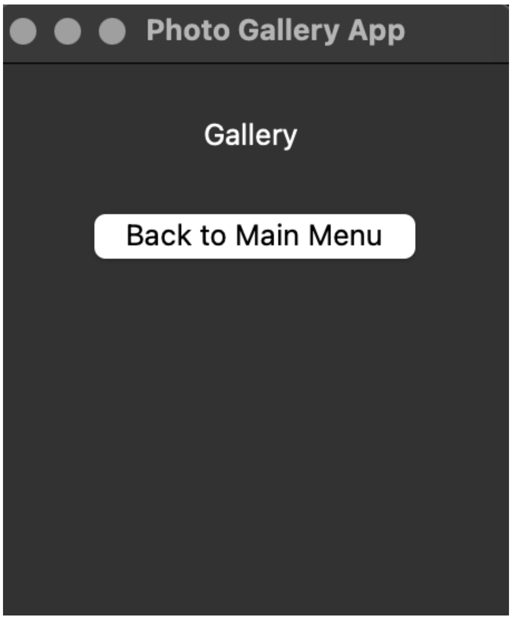
2.

Code:

```
from tkinter import *
from tkinter import ttk

class PhotoGalleryApp:
    def __init__(self, root):
        self.root = root
        self.root.title("Photo Gallery App")

        self.container = Frame(root)
        self.container.pack()

        self.frames = {}

        for F in (MainMenu, Gallery, Import, Export, Settings):
            page_name = F.__name__
            frame = F(parent=self.container, controller=self)
            self.frames[page_name] = frame
            frame.grid(row=0, column=0, sticky="nsew")

        self.show_frame("MainMenu")

    def show_frame(self, page_name):
        frame = self.frames[page_name]
        frame.tkraise()

class MainMenu(Frame):
    def __init__(self, parent, controller):
        Frame.__init__(self, parent)
        Label(self, text="Photo Gallery - Main Menu").pack(padx = 10, pady=20)

        Button(self, text="Gallery", width=20, command=lambda: controller.show_frame("Gallery")).pack(padx = 10, pady=5)
        Button(self, text="Import", width=20, command=lambda: controller.show_frame("Import")).pack(padx = 10, pady=5)
        Button(self, text="Export", width=20, command=lambda: controller.show_frame("Export")).pack(padx = 10, pady=5)
        Button(self, text="Settings", width=20, command=lambda: controller.show_frame("Settings")).pack(padx = 10, pady=5)
        Button(self, text="Exit", width=20, command=controller.root.quit).pack(padx = 10, pady=5)

class Gallery(Frame):
    def __init__(self, parent, controller):
        Frame.__init__(self, parent)
        Label(self, text="Gallery ").pack(padx = 10, pady=20)
        Button(self, text="Back to Main Menu", command=lambda: controller.show_frame("MainMenu")).pack()

class Import(Frame):
    def __init__(self, parent, controller):
        Frame.__init__(self, parent)
        Label(self, text="Import").pack(padx = 10, pady=20)
        Button(self, text="Back to Main Menu", command=lambda: controller.show_frame("MainMenu")).pack()

class Export(Frame):
    def __init__(self, parent, controller):
        Frame.__init__(self, parent)
        Label(self, text="Export").pack(padx = 10, pady=20)
        Button(self, text="Back to Main Menu", command=lambda: controller.show_frame("MainMenu")).pack()

class Settings(Frame):
    def __init__(self, parent, controller):
        Frame.__init__(self, parent)
        Label(self, text="Settings").pack(padx = 10, pady=20)
        Button(self, text="Back to Main Menu", command=lambda: controller.show_frame("MainMenu")).pack()

root = Tk()
app = PhotoGalleryApp(root)
root.mainloop()
```

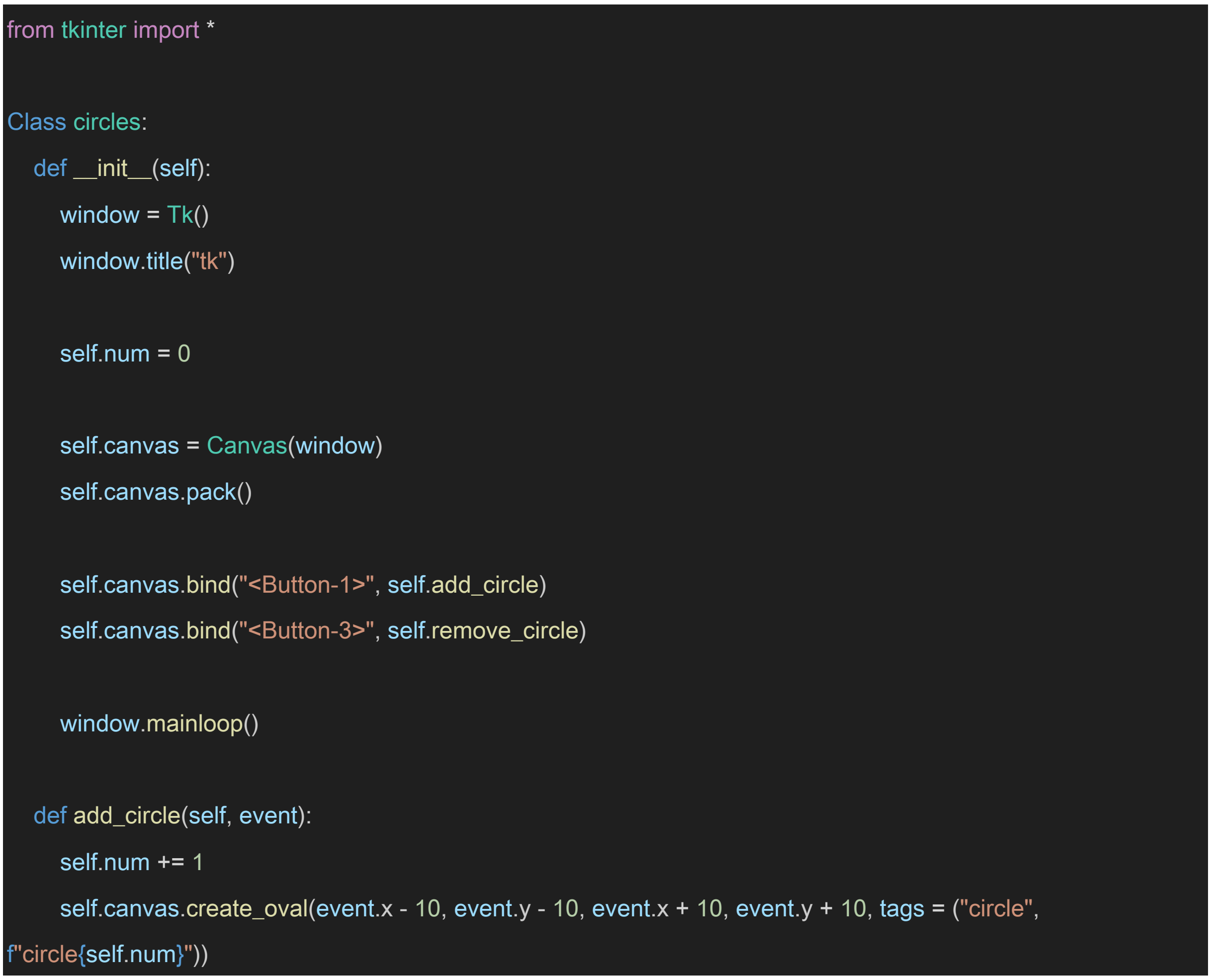
Result:



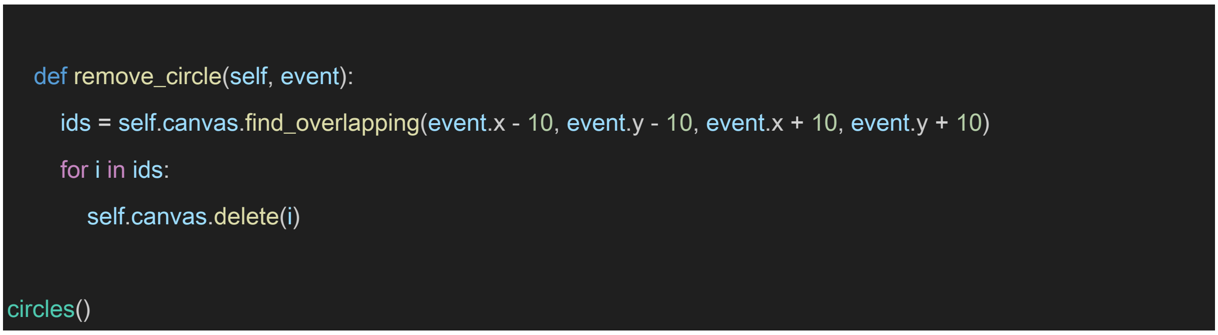


3.

Code:



```
from tkinter import *

Class circles:
  def __init__(self):
    window = Tk()
    window.title("tk")

    self.num = 0

    self.canvas = Canvas(window)
    self.canvas.pack()

    self.canvas.bind("<Button-1>", self.add_circle)
    self.canvas.bind("<Button-3>", self.remove_circle)

    window.mainloop()

  def add_circle(self, event):
    self.num += 1
    self.canvas.create_oval(event.x - 10, event.y - 10, event.x + 10, event.y + 10, tags = ("circle",
f"circle{self.num}"))
```



```
    def remove_circle(self, event):
        ids = self.canvas.find_overlapping(event.x - 10, event.y - 10, event.x + 10, event.y + 10)
        for i in ids:
            self.canvas.delete(i)

circles()
```

Result: