

Modul Praktikum

Pengenalan Tkinter

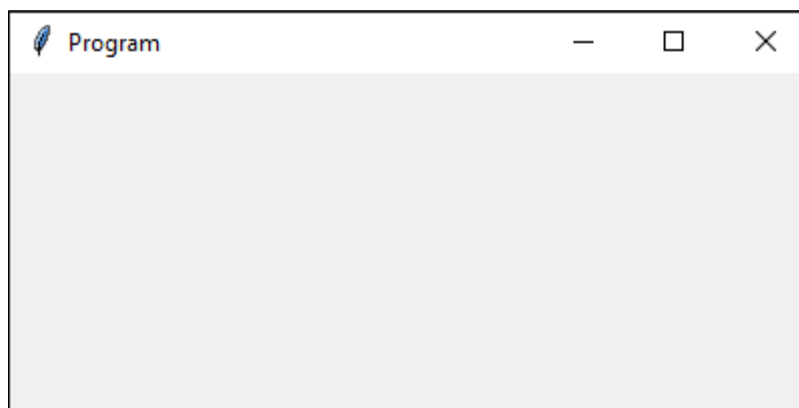
A. Latihan:

1. Menampilkan Form Kosong:

```
from tkinter import Tk

class FormBlanko:
    def __init__(self, parent, title):
        self.parent = parent
        self.parent.geometry("400x200")
        self.parent.title(title)

if __name__ == '__main__':
    root = Tk()
    aplikasi = FormBlanko(root, "Program")
    root.mainloop()
```



Gambar 1. Form Kosong

2. Tampilkan Pesan

```
from tkinter import Tk, Button, Frame, BOTH, YES, messagebox

class FormBlanko:
    def __init__(self, parent, title):
        self.parent = parent
        self.parent.geometry("300x200")
        self.parent.title(title)
        self.parent.protocol("WM_DELETE_WINDOW", self.onKeluar)
        self.aturKomponen()

    def aturKomponen(self):
        # Pasang Frame
        mainFrame = Frame(self.parent, bd=10)
        mainFrame.pack(fill=BOTH, expand=YES)
```

```

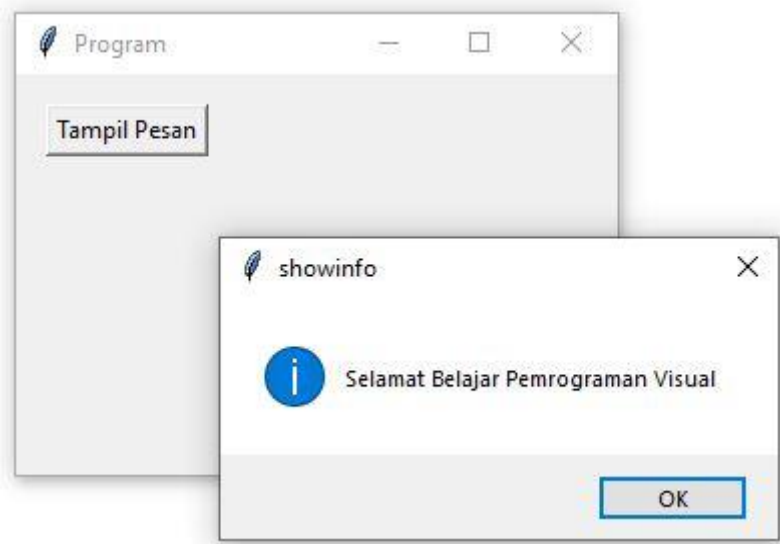
# Pasang Button
self.btnTampil = Button(mainFrame, text='Tampil Pesan',
command=self.onTampil)
self.btnTampil.grid(row=2, column=1, padx=5, pady=5)

def onTampil(self, event=None):
    messagebox.showinfo("showinfo", "Selamat Belajar Pemrograman
Visual")

def onKeluar(self, event=None):
    # memberikan perintah menutup aplikasi
    self.parent.destroy()

if __name__ == '__main__':
    root = Tk()
    aplikasi = FormBlanko(root, "Program")
    root.mainloop()

```



Gambar 2. Form Tampil Pesan

3. Memposisikan Widget di Tkinter Window

Contoh:

```

from tkinter import Tk, Label, Frame, BOTH, YES, W
class FormBlanko:
    def __init__(self, parent, title):
        self.parent = parent
        self.parent.geometry("200x100") # Ukuran Layar Windows
        self.parent.title(title)
        self.parent.protocol("WM_DELETE_WINDOW", self.onKeluar)
        self.aturKomponen()

    def aturKomponen(self):
        mainFrame = Frame(self.parent, bd=10)

```

```

mainFrame.pack(fill=BOTH, expand=YES)

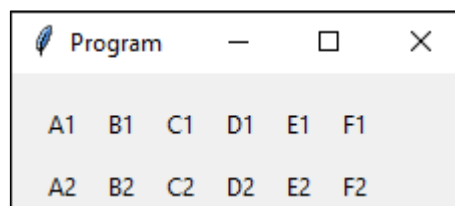
# Label
# row=0 -> baris pertama
# column=0 -> kolom pertama
Label(mainFrame, text='A1').grid(row=0, column=0, sticky=W,
    padx=5, pady=5)
Label(mainFrame, text='B1').grid(row=0, column=1, sticky=W,
    padx=5, pady=5)
Label(mainFrame, text='C1').grid(row=0, column=2, sticky=W,
    padx=5, pady=5)
Label(mainFrame, text='D1').grid(row=0, column=3, sticky=W,
    padx=5, pady=5)
Label(mainFrame, text='E1').grid(row=0, column=4, sticky=W,
    padx=5, pady=5)
Label(mainFrame, text='F1').grid(row=0, column=5, sticky=W,
    padx=5, pady=5)

Label(mainFrame, text='A2').grid(row=1, column=0, sticky=W,
    padx=5, pady=5)
Label(mainFrame, text='B2').grid(row=1, column=1, sticky=W,
    padx=5, pady=5)
Label(mainFrame, text='C2').grid(row=1, column=2, sticky=W,
    padx=5, pady=5)
Label(mainFrame, text='D2').grid(row=1, column=3, sticky=W,
    padx=5, pady=5)
Label(mainFrame, text='E2').grid(row=1, column=4, sticky=W,
    padx=5, pady=5)
Label(mainFrame, text='F2').grid(row=1, column=5, sticky=W,
    padx=5, pady=5)

def onKeluar(self, event=None):
    # memberikan perintah menutup aplikasi
    self.parent.destroy()

if __name__ == '__main__':
    root = Tk()
    aplikasi = FormBlanko(root, "Program")
    root.mainloop()

```



Gambar 3. Posisi Widget Label

4. Entry Data sederhana menggunakan Dictionary

Contoh:

```
import tkinter as tk
from tkinter import ttk

class App:
    def __init__(self, root):
        self.root = root
        self.root.title("Data Entry App")

        # siapkan variabel dictionary kosong utk menyimpan data
        self.data = {}

        # Label dan Textbox Nama
        tk.Label(root, text="Nama:").grid(row=0, column=0, padx=5,
pady=5)
        self.nama_entry = tk.Entry(root)
        self.nama_entry.grid(row=0, column=1, padx=5, pady=5)

        # Label dan Textbox Umur
        tk.Label(root, text="Umur:").grid(row=1, column=0, padx=5,
pady=5)
        self.umur_entry = tk.Entry(root)
        self.umur_entry.grid(row=1, column=1, padx=5, pady=5)

        # Tombol Simpan
        tk.Button(root, text="Simpan",
command=self.save_data).grid(row=2, column=1, padx=5, pady=5)

        # create a TreeView widget to display data
        self.tree = ttk.Treeview(root, columns=("nama", "umur"),
show="headings")
        self.tree.column("nama", width=100)
        self.tree.column("umur", width=100)
        self.tree.heading("nama", text="Nama")
        self.tree.heading("umur", text="Umur")
        self.tree.grid(row=3, column=0, columnspan=2, padx=5, pady=5)

    def save_data(self):
        nama = self.nama_entry.get()
        umur = self.umur_entry.get()

        # save data to dictionary
        data_dict = {
            "nama": nama,
            "umur": umur
        }
```

```

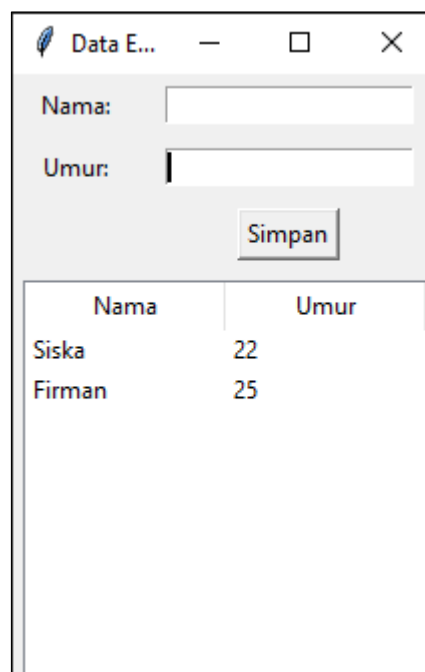
        self.data.update(data_dict)

        # clear entry widgets
        self.nama_entry.delete(0, tk.END)
        self.umur_entry.delete(0, tk.END)

        # update TreeView with new data
        self.tree.insert("", tk.END, values=(nama, umur))

if __name__ == "__main__":
    root = tk.Tk()
    app = App(root)
    root.mainloop()

```



Gambar 4. Entry Data

5. Aplikasi GUI sederhana untuk menghitung Luas dan Keliling Persegi Panjang

```

from tkinter import Frame, Label, Entry, Button, YES, BOTH, END, Tk, W

class FrmPersegi:
    def __init__(self, parent, title):
        self.parent = parent
        self.parent.geometry("400x200")
        self.parent.title(title)
        self.parent.protocol("WM_DELETE_WINDOW", self.onKeluar)
        self.aturKomponen()

```

```

def aturKomponen(self):
    mainFrame = Frame(self.parent, bd=10)
    mainFrame.pack(fill=BOTH, expand=YES)

    # pasang Label
    Label(mainFrame, text='Panjang:').grid(row=0, column=0,
        sticky=W, padx=5, pady=5)
    Label(mainFrame, text="Lebar:").grid(row=1, column=0,
        sticky=W, padx=5, pady=5)
    Label(mainFrame, text="Luas:").grid(row=3, column=0,
        sticky=W, padx=5, pady=5)
    Label(mainFrame, text="Keliling:").grid(row=4, column=0,
        sticky=W, padx=5, pady=5)

    # pasang textbox
    self.txtPanjang = Entry(mainFrame)
    self.txtPanjang.grid(row=0, column=1, padx=5, pady=5)
    self.txtLebar = Entry(mainFrame)
    self.txtLebar.grid(row=1, column=1, padx=5, pady=5)
    self.txtLuas = Entry(mainFrame)
    self.txtLuas.grid(row=3, column=1, padx=5, pady=5)
    self.txtKeliling = Entry(mainFrame)
    self.txtKeliling.grid(row=4, column=1, padx=5, pady=5)

    # Pasang Button
    self.btnHitung = Button(mainFrame, text='Hitung',
        command=self.onHitung)
    self.btnHitung.grid(row=2, column=1, padx=5, pady=5)

# fungsi untuk menghitung luas dan keliling persegi panjang
def onHitung(self, event=None):
    # perhitungan dengan metode Pemrograman Tidak Terstruktur
    panjang = int(self.txtPanjang.get())
    lebar = int(self.txtLebar.get())

    luas = panjang * lebar
    self.txtLuas.delete(0,END)
    self.txtLuas.insert(END,str(luas))

    kel = (2 * panjang) + (2 * lebar)
    self.txtKeliling.delete(0,END)
    self.txtKeliling.insert(END,str(kel))

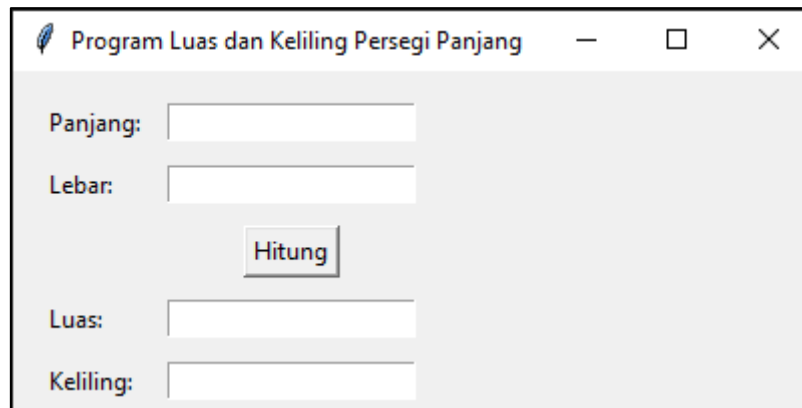
def onKeluar(self, event=None):
    # memberikan perintah menutup aplikasi
    self.parent.destroy()

```

```

if __name__ == '__main__':
    root = Tk()
    aplikasi = FrmPersegi(root, "Program Luas dan Keliling Persegi Panjang")
    root.mainloop()

```



Gambar 5. Desain Aplikasi Hitung Luas & Keliling Persegi Panjang

B. Praktikum

Soal Praktikum:

Salah satu ciri khas pada Pemrograman Terstruktur adalah adanya penggunaan Fungsi

Buatlah 6 Aplikasi untuk menghitung Luas dan Keliling dari bangun datar berikut menggunakan konsep **Pemrograman Terstruktur**:

1. Persegi Panjang
2. Segitiga
3. Persegi / Bujur Sangkar
4. Lingkaran
5. Trapesium
6. Layang-layang
7. Belah Ketupat

C. Tugas

Soal Tugas:

Buatlah 7 aplikasi diatas dengan perhitungan menggunakan konsep **Object Oriented Programming**

Kirim dalam bentuk Laporan Praktikum berupa PDF ke email: freddy.wicaksono@umc.ac.id dengan subject: Laporan **Tugas-1 Visual 2023**