

# LAPORAN PRAKTIKUM

PEMROGRAMAN VISUAL

2023



Prepared By:

AHMAD NUR CAHYADI

190511094 / R3

**LAPORAN PRAKTIKUM**  
**PEMROGRAMAN VISUAL**



Disusun Oleh :

Nama : Ahmad Nur Cahyadi  
NIM : 190511094  
Kelas : TI20C

**JURUSAN TEKNIK INFORMATIKA**  
**FAKULTAS TEKNIK**  
**UNIVERSITAS MUHAMMADIYAH CIREBON**  
**2023**

## **KATA PENGANTAR**

Puji syukur kita panjatkan kehadiran Allah Swt. yang telah memberikan rahmat dan hidayah-Nya sehingga saya dapat menyelesaikan tugas yang berjudul "Laporan Praktikum Pemrograman Visual " ini tepat pada waktunya.

Adapun tujuan dari penulisan dari laporan ini adalah untuk memenuhi tugas pada mata kuliah Pemrograman Visual. Selain itu, laporan ini juga bertujuan untuk menambah wawasan tentang Object Oriented Programming (OOP) menggunakan Python tkinter.

Terlebih dahulu, saya mengucapkan terima kasih kepada Bapak Freddy Wicaksono M. Kom, selaku Dosen Pemrograman Visual yang telah memberikan tugas ini sehingga dapat menambah pengetahuan dan wawasan sesuai dengan bidang studi yang saya tekuni ini.

Saya juga mengucapkan terima kasih kepada semua pihak yang tidak dapat saya sebutkan semua, terima kasih atas bantuannya sehingga sehingga saya dapat menyelesaikan tugas ini.

Kemudian, saya menyadari bahwa tugas yang saya tulis ini masih jauh dari kata sempurna. Oleh karena itu, kritik dan saran yang membangun kami butuhkan demi kesempurnaan laporan ini

Cirebon, Maret 2023

Penyusun

## I. PENDAHULUAN

### A. Latar Belakang

Praktikum adalah kegiatan dengan mendukung kegiatan dalam belajar mengajar serta diharapkan dapat memaksimalkan serta memberi pemahaman mengenai tentang teori. Untuk manfaat nyata, praktikum membantu mahasiswa untuk mendapatkan pemahaman lebih konkrit atas teori yang telah mereka pelajari.

Untuk mahasiswa Teknik Informatika, praktikum selalu menjadi pilar penting dalam pemahaman teori mahasiswanya. Karena dalam penerapannya, mahasiswa Teknik Informatika selalu dihadapkan dengan praktik pembuatan aplikasi ataupun sistem. Salah satunya yang telah kami lakukan kemarin, yaitu praktikum pembuatan aplikasi perhitungan luas dan keliling pada bangun datar menggunakan python tkinter.

### B. Tujuan

Tujuan pada praktikum kali ini adalah sebagai berikut :

1. Memahami konsep python tkinter
2. Memahami konsep Object Oriented Programming (OOP)
3. Membuat aplikasi perhitungan luas dan keliling bangun datar

## II. DASAR TEORI

### A. Sistem Informasi

Secara umum pengertian Sistem Informasi adalah suatu yang menyediakan informasi untuk manajemen pengambilan keputusan atau kebijakan dan menjalankan operasional dari kombinasi orang-orang, teknologi dan prosedur-prosedur yang terorganisasi.

Sistem informasi juga dapat di artikan sebagai kombinasi dari teknologi untuk mendukung operasi dan manajemen. Sebuah sistem merupakan piranti penting dan sangat dibutuhkan oleh suatu perusahaan atau instansi resmi. Hal ini karena dengan adanya sistem yang terintegrasi, kinerja suatu perusahaan atau instansi akan lebih terarah dan sistematis. Namun untuk mendapatkan dampak positif dari penggunaan sistem, semua unsur – unsur yang terkait didalamnya harus berkerja sama guna mencapai tujuan yang sudah ditentukan sebelumnya.

## Pengertian Sistem Informasi (SI) Menurut Para Ahli

- John F. Nash

Pengertian sistem Informasi menurut John F. Nash adalah kombinasi dari manusia, fasilitas atau alat teknologi, media, prosedur dan pengendalian yang bermaksud menata jaringan komunikasi yang penting, proses atau transaksi tertentu dan rutin, membantu manajemen dan pemakai intern dan ekstern dan menyediakan dasar pengambilan keputusan yang tepat.

- Robert A. Leitch

Menurut Robert A. Leitch, pengertian sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan.

- Henry Lucas

Pengertian sistem informasi menurut Henry Lucas adalah suatu kegiatan dari prosedur-prosedur yang diorganisasikan, apabila dieksekusi akan menyediakan informasi untuk mendukung pengambilan keputusan dan pengendalian di dalam.

## Ciri - ciri sistem informasi :

1. **Baru**, artinya bahwa suatu siste informasi yang didapat merupakan informasi yang baru dan segar bagi penerima
2. **Tambahan**, adalah informasi dapat diperbaharui atau memberikan tambahan terhadap informasi yang sebelumnya telah ada.
3. **Kolektif**, adalah informasi yang dapat menjadi suatu koreksi dari informasi yang salah sebelumnya.
4. **Penegas**, adalah informasi yang dapat mempertegas informasi yang telah ada

## B. Object Oriented Programming (OOP)

Object-Oriented Programming atau OOP adalah suatu metode programming yang berorientasi pada objek. Program-program tersebut merupakan gabungan dari beberapa komponen kecil yang telah ada sebelumnya, sehingga dapat mempermudah pekerjaan seorang programmer dalam melakukan pengembangan.

OOP adalah cocok di gunakan untuk membuat program yang menerapkan kehidupan sehari-hari. Dengan OOP para pengembang bisa menjadi software pada objek-objek tertentu.

Prinsip pada Object Oriented Programming :

### 1. Encapsulation

Pengertian Encapsulation pada OOP adalah konsep tentang pengikatan data atau metode berbeda yang di satukan atau “dikapsulkan” menjadi satu unit data. Encapsulation dapat memudahkan dalam pembacaan kode karena informasi yang di sajikan tidak perlu di baca secara rinci dan sudah merupakan satu kesatuan.

Encapsulation juga sering di gunakan karena terdapat fitur information-hiding mechanism. Mekanisme ini menghilangkan akses publik ke atribut-atribut yang terdapat di dalam “kapsul” tersebut. Metode ini dapat memudahkan kamu dalam mendefinisikan atribut apa saja yang dapat di baca dan di perbarui.

### 2. Inheritance

Prinsip inheritance pada OOP adalah di mana kita dapat membentuk class baru yang “mewarisi” atau memiliki bagian-bagian dari class yang sudah ada sebelumnya. Konsep ini menggunakan sistem hirarki atau bertingkat.

Seperti sebuah Drop-Down Menu yang ada di kebanyakan website, di mana semakin spesifik submenunya, semakin spesifik pula kontennya. Demikian juga dengan Inheritance OOP di mana semakin spesifik subclassnya, semakin sedikit pula komponen yang dapat di warisi class tersebut.

### 3. Abstract Class

Seperti namanya, prinsip abstract class OOP adalah class-class yang memiliki informasi abstrak dan metode-metode dari sekumpulan data. Abstract Class tidak bisa di ubah dan berlaku juga sebagai kerangka dalam penciptaan berbagai subclass (berperan seperti Superclass yang di bahas di konsep Inheritance).

Suatu abstract class memiliki informasi dan metode yang dapat di turunkan ke subclass-nya. Kemudian, seluruh subclass tersebut akan mengikuti apa saja metode turunan dari abstract class tersebut.

### 4. Polymorphism

Prinsip polymorphism pada OOP adalah konsep di mana suatu objek berbeda-beda dapat di akses melalui satu interface. Sebuah objek polymorphic dapat beradaptasi dengan metode apapun yang di implementasikan pada objek tersebut, dan setiap class memiliki interpretasinya tersendiri terhadap interfacenya.

## C. Python Tkinter

Python adalah salah satu bahasa pemrograman yang dapat melakukan eksekusi sejumlah instruksi multi guna secara langsung (interpretatif) dengan metode orientasi objek (Object Oriented Programming) serta menggunakan semantik dinamis untuk memberikan tingkat keterbacaan syntax. Sebagian lain mengartikan Python sebagai bahasa yang kemampuan, menggabungkan kapabilitas, dan sintaksis kode yang sangat jelas, dan juga dilengkapi dengan fungsionalitas pustaka standar yang besar serta komprehensif. Walaupun Python tergolong bahasa pemrograman dengan level tinggi, nyatanya Python dirancang sedemikian rupa agar mudah dipelajari dan dipahami.

Python sendiri menampilkan fitur-fitur menarik sehingga layak untuk Anda pelajari. Pertama, Python memiliki tata bahasa dan script yang sangat mudah untuk dipelajari. Python juga memiliki sistem pengelolaan data dan memori otomatis. Selain itu modul pada Python selalu diupdate. Ditambah lagi, Python juga memiliki banyak fasilitas pendukung. Python banyak diaplikasikan pada berbagai sistem

operasi seperti Linux, Microsoft Windows, Mac OS, Android, Symbian OS, Amiga, Palm dan lain-lain.

Tkinter merupakan library yang sudah sering kamu dengar untuk membuat aplikasi antarmuka (GUI) python. Tkinter menyediakan cara cepat dan mudah yang berorientasikan objek yang kuat dalam membuat aplikasi python berbasis GUI. Tkinter biasanya secara default di-bundle dengan Python. Jadi ketika kamu install Python, Tkinter juga akan ikut terinstal pula. Tkinter sebenarnya bentuk OOP dari TCL/TK. TCL (Tool Command Language) adalah sebuah bahasa pemrograman dan TK adalah library yang digunakan oleh TCL untuk membuat aplikasi GUI.

#### D. Luas dan Keliling Bangun Datar

Berikut ini adalah berbagai macam rumus luas dan keliling dari bermacam bentuk bangun datar yang harus kamu ketahui. Panjang batas bangun datar disebut keliling bangun datar. Satuan keliling sama dengan satuan panjang, yaitu m, cm, mm, dll. Sedangkan bagian dari bidang yang diapit oleh gambar tertutup sederhana disebut bidang bidang dan pengukuran bidang bidang yang tertutup disebut luasnya. Luas diukur dalam satuan persegi.

1. Persegi Panjang

$$Keliling = 2(p + l)$$

$$Luas = p \times l$$

2. Persegi

$$Keliling = 4 \times s$$

$$Luas = s \times s$$

3. Segitiga

$$Keliling = a + b + c \Rightarrow (a, b, c) \text{ merupakan sisi dari segitiga}$$

$$Luas = \frac{a \times t}{2}$$

4. Layang - layang

$$Keliling = 2(x + y) \Rightarrow x \text{ merupakan sisi yang pendek, } y \text{ merupakan sisi yang panjang}$$

$$Luas = \frac{d1 \times d2}{2}$$

5. Belah Ketupat

$$Keliling = 4 \times s$$



$$Luas = a \times t$$

#### 6. Trapesium

$Keliling = a + b + c + d \Rightarrow$  (a, b, c, d) merupakan sisi pada trapesium

$$Luas = \frac{(jumlah\ sisi\ sejajar) \times t}{2}$$

#### 7. Lingkaran

$$Keliling = 2\pi r$$

$$Luas = \pi r^2$$

### III. PEMBAHASAN

Berikut hasil dari praktikum kemarin :

#### 1. Persegi Panjang

- Source code nya :

```
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))

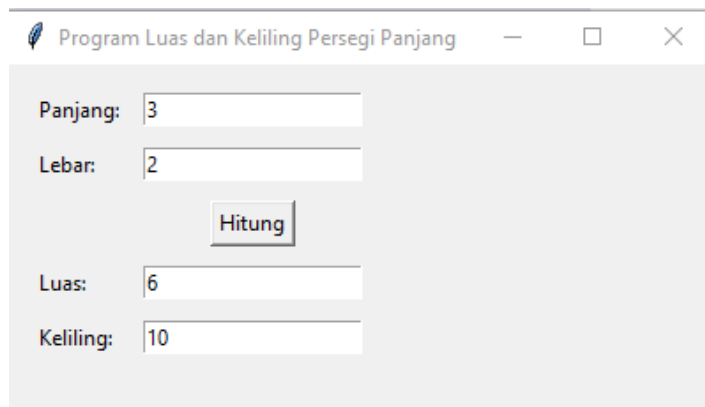
    kel = (2 * panjang) + (2 * lebar)
    self.txtKeliling.delete(0,END)
    self.txtKeliling.insert(END,str(kel))

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()

```

- Hasilnya :



The screenshot shows a graphical user interface for calculating the area and perimeter of a rectangle. The window has a title bar with the text 'Program Luas dan Keliling Persegi Panjang'. Inside the window, there are two input fields: 'Panjang' with the value '3' and 'Lebar' with the value '2'. Below these is a button labeled 'Hitung'. Under the button, there are two more input fields: 'Luas' with the value '6' and 'Keliling' with the value '10'.

## 2. Segitiga

- Source Code nya :

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

class FrmSegitiga:
    def __init__(self, parent, title):
        self.parent = parent
        self.parent.geometry("400x500")
        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='Alas:').grid(row=0, column=0, sticky=W, padx=5,
pady=5)
        Label(mainFrame, text="Tinggi:").grid(row=1, column=0, sticky=W,
padx=5, pady=5)
        Label(mainFrame, text='Sisi 1:').grid(row=2, column=0, sticky=W,
padx=5, pady=5)
        Label(mainFrame, text='Sisi 2:').grid(row=3, column=0, sticky=W,
padx=5, pady=5)
        Label(mainFrame, text='Sisi 3:').grid(row=4, column=0, sticky=W,
padx=5, pady=5)
        Label(mainFrame, text="Luas:").grid(row=6, column=0, sticky=W, padx=5,
pady=5)
        Label(mainFrame, text="Keliling:").grid(row=7, column=0, sticky=W,
padx=5, pady=5)

        # pasang textbox
        self.txtAlas = Entry(mainFrame)
        self.txtAlas.grid(row=0, column=1, padx=5, pady=5)

        self.txtTinggi = Entry(mainFrame)
        self.txtTinggi.grid(row=1, column=1, padx=5, pady=5)

        self.txtSisiSatu = Entry(mainFrame)
        self.txtSisiSatu.grid(row=2, column=1, padx=5, pady=5)

        self.txtSisiDua = Entry(mainFrame)
        self.txtSisiDua.grid(row=3, column=1, padx=5, pady=5)

        self.txtSisiTiga = Entry(mainFrame)
        self.txtSisiTiga.grid(row=4, column=1, padx=5, pady=5)
```

```

self.txtLuas = Entry(mainFrame)
self.txtLuas.grid(row=6, column=1, padx=5, pady=5)

self.txtKeliling = Entry(mainFrame)
self.txtKeliling.grid(row=7, column=1, padx=5, pady=5)

# Pasang Button
self.btnHitung = Button(mainFrame, text='Hitung',
command=self.onHitung)
self.btnHitung.grid(row=5, 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
    sisiSatu = int(self.txtSisiSatu.get())
    sisiDua = int(self.txtSisiDua.get())
    sisiTiga = int(self.txtSisiTiga.get())
    alas = int(self.txtAlas.get())
    tinggi = int(self.txtTinggi.get())

    luas = (alas * tinggi) / 2
    self.txtLuas.delete(0,END)
    self.txtLuas.insert(END,str(luas))

    kel = sisiSatu + sisiDua + sisiTiga
    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 = FrmSegitiga(root, "Program Luas dan Keliling Segitiga")
    root.mainloop()

```

- Hasilnya :

The screenshot shows a graphical user interface for calculating the area and perimeter of a triangle. The window title is 'Program Luas dan Keliling Segitiga'. It features five input fields for the triangle's dimensions: 'Alas' (base) with value 2, 'Tinggi' (height) with value 3, 'Sisi 1' (side 1) with value 1, 'Sisi 2' (side 2) with value 2, and 'Sisi 3' (side 3) with value 5. A 'Hitung' (Calculate) button is positioned below the input fields. The results are displayed at the bottom: 'Luas' (Area) is 3.0 and 'Keliling' (Perimeter) is 8.

### 3. Persegi

- Source Code :

```
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='Sisi:').grid(row=0, column=0, sticky=W, padx=5,
pady=5)
        Label(mainFrame, text="Luas:").grid(row=2, column=0, sticky=W, padx=5,
pady=5)
        Label(mainFrame, text="Keliling:").grid(row=3, column=0, sticky=W,
padx=5, pady=5)

        # pasang textbox
        self.txtSisi = Entry(mainFrame)
        self.txtSisi.grid(row=0, column=1, padx=5, pady=5)

        self.txtLuas = Entry(mainFrame)
        self.txtLuas.grid(row=2, column=1, padx=5, pady=5)
```

```

self.txtKeliling = Entry(mainFrame)
self.txtKeliling.grid(row=3, column=1, padx=5, pady=5)

# Pasang Button
self.btnHitung = Button(mainFrame, text='Hitung',
command=self.onHitung)
self.btnHitung.grid(row=1, 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
    sisi = int(self.txtSisi.get())

    luas = sisi ** 2
    self.txtLuas.delete(0,END)
    self.txtLuas.insert(END,str(luas))

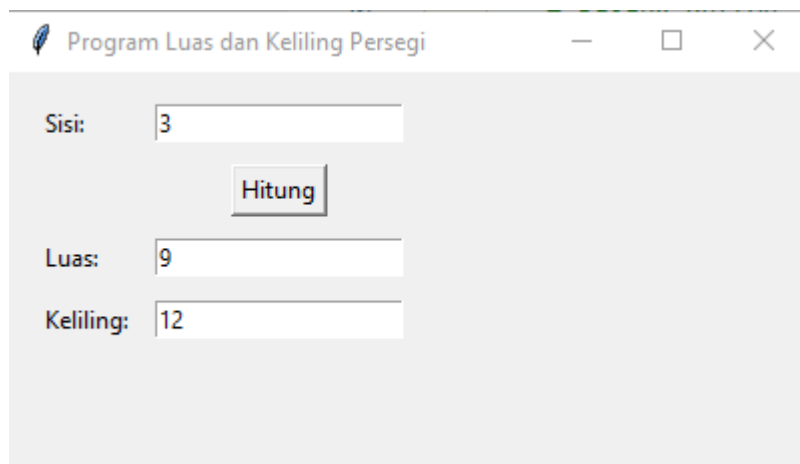
    kel = 4 * sisi
    self.txtKeliling.delete(0,END)
    self.txtKeliling.insert(END,str(kel))

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")
    root.mainloop()

```

- Hasil :



#### 4. Lingkaran

- Source Code :

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

class FrmLingkaran:
    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='Jari-jari:').grid(row=0, column=0, sticky=W,
        padx=5, pady=5)
        Label(mainFrame, text="Luas:").grid(row=2, column=0, sticky=W, padx=5,
        pady=5)
        Label(mainFrame, text="Keliling:").grid(row=3, column=0, sticky=W,
        padx=5, pady=5)

        # pasang textbox
        self.txtJari = Entry(mainFrame)
        self.txtJari.grid(row=0, column=1, padx=5, pady=5)

        self.txtLuas = Entry(mainFrame)
        self.txtLuas.grid(row=2, column=1, padx=5, pady=5)

        self.txtKeliling = Entry(mainFrame)
        self.txtKeliling.grid(row=3, column=1, padx=5, pady=5)

        # Pasang Button
        self.btnHitung = Button(mainFrame, text='Hitung',
        command=self.onHitung)
        self.btnHitung.grid(row=1, 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
            r = int(self.txtJari.get())
            phi = 3.14
```

```

luas = phi * (r ** 2)
self.txtLuas.delete(0,END)
self.txtLuas.insert(END,str(luas))

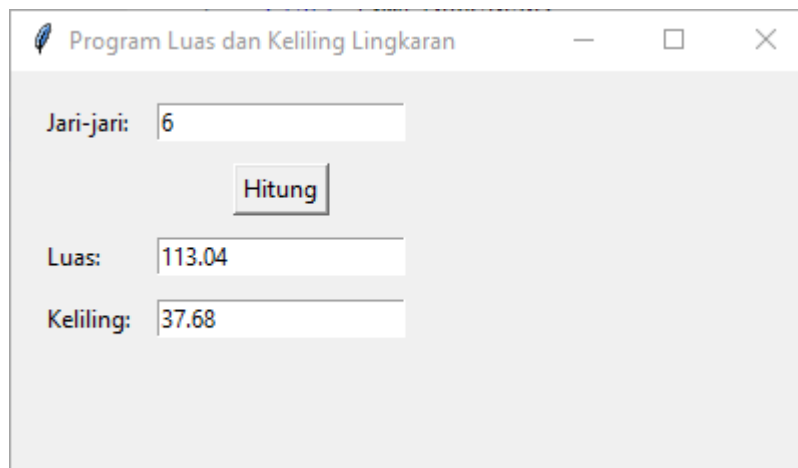
kel = 2 * phi * r
self.txtKeliling.delete(0,END)
self.txtKeliling.insert(END,str(kel))

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

if __name__ == '__main__':
    root = Tk()
    aplikasi = FrmLingkaran(root, "Program Luas dan Keliling Lingkaran")
    root.mainloop()

```

- Hasil :



## 5. Trapesium

- Source Code :

```

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

class FrmTrapesium:
    def __init__(self, parent, title):
        self.parent = parent
        self.parent.geometry("400x500")
        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='Sisi A:').grid(row=0, column=0, sticky=W,
padx=5, pady=5)
Label(mainFrame, text='Sisi B:').grid(row=1, column=0, sticky=W,
padx=5, pady=5)
Label(mainFrame, text='Sisi C:').grid(row=2, column=0, sticky=W,
padx=5, pady=5)
Label(mainFrame, text='Sisi D:').grid(row=3, column=0, sticky=W,
padx=5, pady=5)
Label(mainFrame, text="Tinggi:").grid(row=4, column=0, sticky=W,
padx=5, pady=5)
Label(mainFrame, text="Luas:").grid(row=6, column=0, sticky=W, padx=5,
pady=5)
Label(mainFrame, text="Keliling:").grid(row=7, column=0, sticky=W,
padx=5, pady=5)

# pasang textbox
self.txtA = Entry(mainFrame)
self.txtA.grid(row=0, column=1, padx=5, pady=5)

self.txtB = Entry(mainFrame)
self.txtB.grid(row=1, column=1, padx=5, pady=5)

self.txtC = Entry(mainFrame)
self.txtC.grid(row=2, column=1, padx=5, pady=5)

self.txtD = Entry(mainFrame)
self.txtD.grid(row=3, column=1, padx=5, pady=5)

self.txtTinggi = Entry(mainFrame)
self.txtTinggi.grid(row=4, column=1, padx=5, pady=5)

self.txtLuas = Entry(mainFrame)
self.txtLuas.grid(row=6, column=1, padx=5, pady=5)

self.txtKeliling = Entry(mainFrame)
self.txtKeliling.grid(row=7, column=1, padx=5, pady=5)

# Pasang Button
self.btnHitung = Button(mainFrame, text='Hitung',
command=self.onHitung)
self.btnHitung.grid(row=5, 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
a = int(self.txtA.get())
b = int(self.txtB.get())
c = int(self.txtC.get())
d = int(self.txtD.get())
tinggi = int(self.txtTinggi.get())

luas = ((a + c) * tinggi) / 2
self.txtLuas.delete(0,END)
self.txtLuas.insert(END,str(luas))

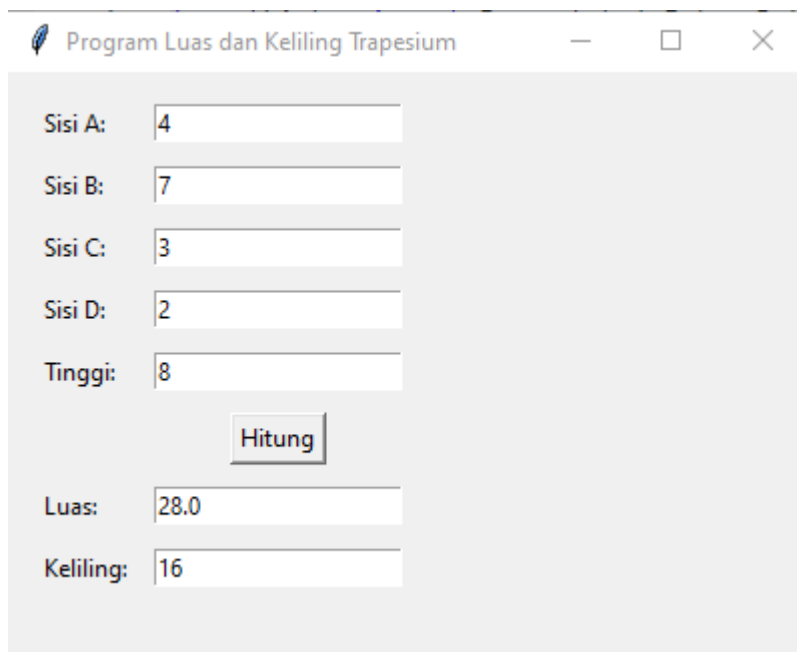
kel = a + b + c + d
self.txtKeliling.delete(0,END)
self.txtKeliling.insert(END,str(kel))

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

if __name__ == '__main__':
    root = Tk()
    aplikasi = FrmTrapesium(root, "Program Luas dan Keliling Trapesium")
    root.mainloop()

```

- Hasil :



Program Luas dan Keliling Trapesium

Sisi A: 4

Sisi B: 7

Sisi C: 3

Sisi D: 2

Tinggi: 8

Hitung

Luas: 28.0

Keliling: 16

## 6. Layang – layang

- Source Code :

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

class FrmLayang:
    def __init__(self, parent, title):
        self.parent = parent
        self.parent.geometry("400x500")
        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='Sisi X:').grid(row=0, column=0, sticky=W,
        padx=5, pady=5)
        Label(mainFrame, text='Sisi Y:').grid(row=1, column=0, sticky=W,
        padx=5, pady=5)
        Label(mainFrame, text='d1:').grid(row=2, column=0, sticky=W, padx=5,
        pady=5)
        Label(mainFrame, text='d2:').grid(row=3, column=0, sticky=W, padx=5,
        pady=5)
        Label(mainFrame, text="Luas:").grid(row=5, column=0, sticky=W, padx=5,
        pady=5)
        Label(mainFrame, text="Keliling:").grid(row=6, column=0, sticky=W,
        padx=5, pady=5)

        # pasang textbox
        self.txtX = Entry(mainFrame)
        self.txtX.grid(row=0, column=1, padx=5, pady=5)

        self.txtY = Entry(mainFrame)
        self.txtY.grid(row=1, column=1, padx=5, pady=5)

        self.txtd1 = Entry(mainFrame)
        self.txtd1.grid(row=2, column=1, padx=5, pady=5)

        self.txtd2 = Entry(mainFrame)
        self.txtd2.grid(row=3, column=1, padx=5, pady=5)

        self.txtLuas = Entry(mainFrame)
        self.txtLuas.grid(row=5, column=1, padx=5, pady=5)
```

```

self.txtKeliling = Entry(mainFrame)
self.txtKeliling.grid(row=6, column=1, padx=5, pady=5)

# Pasang Button
self.btnHitung = Button(mainFrame, text='Hitung',
command=self.onHitung)
self.btnHitung.grid(row=4, 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
    x = int(self.txtX.get())
    y = int(self.txtY.get())
    d1 = int(self.txtd1.get())
    d2 = int(self.txtd2.get())

    luas = (d1 * d2) / 2
    self.txtLuas.delete(0,END)
    self.txtLuas.insert(END,str(luas))

    kel = 2 * (x + y)
    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 = FrmLayang(root, "Program Luas dan Keliling Layang-Layang")
    root.mainloop()

```

- Hasil :

The screenshot shows a graphical user interface for a program that calculates the area and perimeter of a kite. The window has a title bar with the text 'Program Luas dan Keliling Layang-Layang'. Inside the window, there are four input fields labeled 'Sisi X:', 'Sisi Y:', 'd1:', and 'd2:'. The values entered are 4, 8, 11, and 23 respectively. Below these input fields is a button labeled 'Hitung'. Below the button, there are two output fields: 'Luas:' with the value 126.5 and 'Keliling:' with the value 24.

## 7. Belah ketupat

- Source Code :

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

class FrmKetupat:
    def __init__(self, parent, title):
        self.parent = parent
        self.parent.geometry("400x300")
        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='Sisi:').grid(row=0, column=0, sticky=W, padx=5,
pady=5)
        Label(mainFrame, text='d1:').grid(row=1, column=0, sticky=W, padx=5,
pady=5)
        Label(mainFrame, text='d2:').grid(row=2, column=0, sticky=W, padx=5,
pady=5)
        Label(mainFrame, text="Luas:").grid(row=4, column=0, sticky=W, padx=5,
pady=5)
        Label(mainFrame, text="Keliling:").grid(row=5, column=0, sticky=W,
padx=5, pady=5)

        # pasang textbox
        self.txtSisi = Entry(mainFrame)
        self.txtSisi.grid(row=0, column=1, padx=5, pady=5)

        self.txtd1 = Entry(mainFrame)
        self.txtd1.grid(row=1, column=1, padx=5, pady=5)

        self.txtd2 = Entry(mainFrame)
        self.txtd2.grid(row=2, column=1, padx=5, pady=5)

        self.txtLuas = Entry(mainFrame)
        self.txtLuas.grid(row=4, column=1, padx=5, pady=5)

        self.txtKeliling = Entry(mainFrame)
        self.txtKeliling.grid(row=5, column=1, padx=5, pady=5)
```

```

        # Pasang Button
        self.btnHitung = Button(mainFrame, text='Hitung',
command=self.onHitung)
        self.btnHitung.grid(row=3, 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
        sisi = int(self.txtSisi.get())
        d1 = int(self.txtd1.get())
        d2 = int(self.txtd2.get())

        luas = (d1 * d2) / 2
        self.txtLuas.delete(0,END)
        self.txtLuas.insert(END,str(luas))

        kel = 4 * sisi
        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 = FrmKetupat(root, "Program Luas dan Keliling Belah Ketupat")
    root.mainloop()

```

- Hasil :

The screenshot shows a graphical user interface for a Python program. The window has a title bar with the text 'Program Luas dan Keliling Belah Ketupat'. Inside the window, there are three input fields labeled 'Sisi:', 'd1:', and 'd2:'. The 'Sisi:' field contains the value '13', 'd1:' contains '24', and 'd2:' contains '12'. Below these input fields is a button labeled 'Hitung'. Underneath the button, there are two more input fields: 'Luas:' containing '144.0' and 'Keliling:' containing '52'.

## IV. PENUTUP

### a. Kesimpulan

Tujuan dalam praktikum sudah tercapai dengan berhasilnya aplikasi perhitungan luas dan keliling bangun datar seperti dalam pembahasan di atas. Aplikasi tersebut dapat dibuat menggunakan python dengan library tkinter untuk membuat aplikasi GUI.

### b. Saran

Rapikan kembali penulisan kode-kode dalam syntax python, dan perbaiki komentar dalam setiap proses dalam syntax tersebut, agar developer dapat mengetahui syntax tersebut untuk apa fungsinya.

## V. DAFTAR PUSTAKA

- <https://bsi.today/pengertian-sistem-informasi/>
- <https://www.jagoanhosting.com/blog/oop-adalah/>
- [https://www.baktikominfo.id/id/informasi/pengetahuan/bahasa\\_pemrograman\\_python\\_pengertian\\_sejarah\\_kelebihan\\_dan\\_kekurangannya-954](https://www.baktikominfo.id/id/informasi/pengetahuan/bahasa_pemrograman_python_pengertian_sejarah_kelebihan_dan_kekurangannya-954)
- <https://dqlab.id/library-python-tkinter-untuk-membuat-aplikasi-dengan-bahasa-pemrograman-berbasis-gui>
- <https://www.kelaspintar.id/blog/tips-pintar/rumus-luas-dan-keliling-bangun-datar-11922/>