

**LAPORAN PRAKTIKUM  
PEMROGRAMAN BERORIENTASI OBJEK (PBO) – [TUGAS BRP]**



Disusun Oleh

ANDRYANO SHEVCHENKO LIMBONG 123140205

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS TEKNOLOGI INFORMASI**

**INSTITUT TEKNOLOGI SUMATERA**

**2025**

**a. Input Soal**



**b. Penjelasan**

Berikan Penjelasan

Pada tugas praktikum ini, diminta untuk membuat sebuah pola segitiga menggunakan karakter (\*) dengan ukuran segitiganya bergantung kepada nilai inputan Height.

Penjelasan dari kode pemrograman, pada line 1 `n = int(input("Tinggi: "))` kode akan dimulai dengan meminta inputan integer tinggi untuk menentukan akan seberapa tinggi segitiga yang akan dibuat.

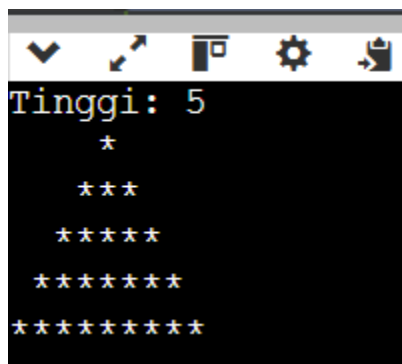
Kemudian inputan tinggi segitiga tersebut akan dimasukkan ke dalam perulangan `for i in range(n):` untuk mencetak tinggi segitiga sesuai dengan inputan tinggi segitiga sebelumnya dan kode `bintang = "*" * (2 * i + 1)` untuk deklarasi variabel bintang untuk menghitung jumlah bintang (\*) yang akan dioutput, dan `print(bintang.center(2*n - 1))` kemudian akan mencetak "\*" sebagai pembentuk segitiga dan memastikan bentuk segitiga simetris.

**c. Source Code**

```
n = int(input("Tinggi: "))

for i in range(n):
    bintang = "*" * (2 * i + 1)
    print(bintang.center(2*n - 1))
```

**d. Hasil**



```
Tinggi: 5
  *
 ***
*****
*****
*****
```

**Soal Nomor 2**

**a. Input soal**

```
1  Masukkan jumlah siswa: 3
2  Masukkan nama siswa ke-1: Ali
3  Masukkan nilai untuk Ali: 85
4  Masukkan nama siswa ke-2: Budi
5  Masukkan nilai untuk Budi: 90
6  Masukkan nama siswa ke-3: Siti
7  Masukkan nilai untuk Siti: 78
8
9  dictionary = {'Ali': 85, 'Budi': 90, 'Siti': 78}
```

## b. Penjelasan

Soal kedua meminta kode program yang berbentuk dictionary yang dimana di dalamnya bisa diinputkan data berupa nama dan nilai.

`data_siswa = {}` adalah deklarasi dictionary yang akan digunakan. Kemudian `jumlah_siswa = int(input("Masukkan jumlah siswa: "))` adalah kode program yang meminta inputan jumlah siswa kepada user. Kemudian kode program `for i in`

```
range(1, jumlah_siswa + 1):  
    nama = input(f"Masukkan nama siswa ke-{i}: ")  
    nilai = int(input(f"Masukkan nilai untuk {nama}: "))
```

Adalah perulangan yang akan digunakan untuk user menginputkan nama siswa dan nilai siswa tersebut ke dalam dictionary dan akan dipasangkan dengan key : value masing-masing sesuai dengan kode program berikut: `data_siswa[nama] = nilai`.

`print("Dictionary siswa:", data_siswa)` berfungsi untuk mengoutput seluruh data yang ada di dalam dictionary.

## c. Source Code

```
data_siswa = {}  
  
jumlah_siswa = int(input("Masukkan jumlah siswa: "))  
  
for i in range(1, jumlah_siswa + 1):  
    nama = input(f"Masukkan nama siswa ke-{i}: ")  
    nilai = int(input(f"Masukkan nilai untuk {nama}: "))  
    data_siswa[nama] = nilai  
  
print("Dictionary siswa:", data_siswa)
```

#### d. Hasil

```
Masukkan jumlah siswa: 3
Masukkan nama siswa ke-1: Andry
Masukkan nilai untuk Andry: 100
Masukkan nama siswa ke-2: Joko
Masukkan nilai untuk Joko: 10
Masukkan nama siswa ke-3: Bowo
Masukkan nilai untuk Bowo: 11
Dictionary siswa: {'Andry': 100, 'Joko': 10, 'Bowo': 11}
```

#### Soal nomor 3

##### a. Input soal

```
akhda@anophylen MINGW64 /c/Aaaaaaaaakhdan/Py
● $ python jokowi.py
Masukkan Nama: Akhdan
Masukkan NIM: 122140099
Masukkan Resolusi di Tahun ini: Jadi presiden
File Me.txt telah berhasil dibuat!
```

##### b. Penjelasan

Soal nomor 3 meminta kode program untuk membuat sebuah file bernama "Me.txt" menggunakan File I/O (Input/Output) di Python, lalu menuliskan beberapa baris teks ke dalamnya.

Line 1 `nama_file = 'D:\\VSCode\\Me.txt'`, kode program deklarasi sekaligus mengakses file Me.txt yang berada di dalam drive D:.

```
with open(nama_file, 'r+') as file:
    isi_file = file.readlines()
```

Line 3 dan 4, adalah syntax yang digunakan untuk mengakses file Me.txt dan dengan syntax r+, program akan dapat melakukan operasi BACA dan TULIS kepada Me.txt secara langsung. Kemudian Line 5 `isi_file.extend([` digunakan untuk menulis atau menimpa isi dari Me.txt dengan isi sebagai berikut:

```
"Masukkan Nama: Andryano Shevchenko Limbong\n",  
"Masukkan NIM: 123140205\n",  
"Masukkan Resolusi Tahun Ini: Jadi Orang Baik, Sopan, Dermawan  
dan Rajin Menabung" ])
```

`file.seek(0)` Code program ini akan kembali ke awal file dan memastikan bahwa code program telah menimpa isi file

`file.writelines(isi_file)` Code program ini digunakan untuk menulis semua baris yang ada di dalam file Me.txt

```
with open(nama_file, 'r') as file:  
  
    print(file.read())
```

Code program di atas merupakan syntax python yang berfungsi untuk membaca isi dari file Me.txt secara keseluruhan.

### c. Source Code

```
nama_file = 'D:\\\\VSCode\\Me.txt'  
  
with open(nama_file, 'r+') as file:  
    isi_file = file.readlines()  
    isi_file.extend([  
        "Masukkan Nama: Andryano Shevchenko Limbong\n",  
        "Masukkan NIM: 123140205\n",  
        "Masukkan Resolusi Tahun Ini: Jadi Orang Baik, Sopan, Dermawan dan Rajin Menabung"  
    ])  
    file.seek(0)  
    file.writelines(isi_file)  
  
with open(nama_file, 'r') as file:  
    print(file.read())
```

#### d. Hasil

```
nama_file = 'D:\VSCode\Me.txt'
Masukkan Nama: Andryano Shevchenko Limbong
Masukkan NIM: 123140205
Masukkan Resolusi Tahun Ini: Jadi Orang Baik, Sopan, Dermawan dan Rajin Menabung
PS D:\VSCode\Phyton>
```

#### Lampiran

1. Lampiran percakapan dengan LLM Blackbox.AI untuk menyelesaikan soal nomor 3

You

syntax code python untuk akses txt

BLACKBOXAI

Untuk mengakses file teks (txt) di Python, Anda dapat menggunakan fungsi built-in `open()`. Berikut adalah beberapa contoh cara untuk membuka, membaca, menulis, dan menutup file teks.

#### 1. Membaca File Teks

Berikut adalah contoh cara membaca isi file teks:

python

```
1 # Membuka file untuk dibaca
2 with open('nama_file.txt', 'r') as file:
3     # Membaca seluruh isi file
4     isi_file = file.read()
5     print(isi_file)
```

- `open('nama_file.txt', 'r')`: Membuka file dengan nama `'nama_file.txt'` dalam mode baca (`'r'`).
- `with`: Menggunakan konteks manajer untuk memastikan file ditutup secara otomatis setelah selesai digunakan.
- `file.read()`: Membaca seluruh isi file.
- `print(isi_file)`: Mencetak isi file ke layar.

## 2. Membaca Baris per Baris

Jika Anda ingin membaca file baris per baris, Anda bisa menggunakan loop:

python



```
1 with open('nama_file.txt', 'r') as file:
2     for baris in file:
3         print(baris.strip()) # strip() untuk menghapus karakter newlin
```

## 3. Menulis ke File Teks

Untuk menulis ke file teks, Anda dapat menggunakan mode `'w'` (write) atau `'a'` (append):

python



```
1 # Menulis ke file (akan menimpa isi file jika sudah ada)
2 with open('nama_file.txt', 'w') as file:
3     file.write("Ini adalah baris pertama.\n")
4     file.write("Ini adalah baris kedua.\n")
5
6 # Menambahkan ke file (tidak menimpa isi file yang sudah ada)
7 with open('nama_file.txt', 'a') as file:
8     file.write("Ini adalah baris ketiga yang ditambahkan.\n")
```



## 4. Membaca File Teks dan Menyimpan ke dalam List

Jika Anda ingin menyimpan setiap baris dari file ke dalam list, Anda bisa menggunakan `readlines()`:

```
python

1 with open('nama_file.txt', 'r') as file:
2     daftar_baris = file.readlines()
3
4 # Menghapus karakter newline dari setiap baris
5 daftar_baris = [baris.strip() for baris in daftar_baris]
6 print(daftar_baris)
```

## 5. Menutup File

Jika Anda tidak menggunakan `with`, Anda harus menutup file secara manual setelah selesai:

```
python

1 file = open('nama_file.txt', 'r')
2 isi_file = file.read()
3 print(isi_file)
4 file.close() # Menutup file
```

## Kesimpulan

Dengan menggunakan fungsi `open()`, Anda dapat dengan mudah mengakses file teks di Python untuk membaca, menulis, atau menambahkan data. Pastikan untuk selalu menutup file setelah selesai, atau lebih baik lagi, gunakan konteks manajer (`with`) untuk mengelola file secara otomatis.

You

tolong buat program untuk mengakses file text (txt) sekaligus menulis ulang dan memprint out hasilnya

BLACKBOXAI

Tentu! Berikut adalah contoh program Python yang mengakses file teks, menulis ulang isi file, dan kemudian mencetak hasilnya. Program ini akan melakukan hal berikut:

1. Membaca isi file teks yang sudah ada.
2. Mengubah isi file (misalnya, menambahkan teks).
3. Menulis ulang isi file dengan perubahan tersebut.
4. Mencetak isi file yang baru.

## Contoh Program

python

```
1 # Nama file yang akan diakses
2 nama_file = 'data.txt'
3
4 # Membaca isi file
5 with open(nama_file, 'r') as file:
6     isi_file = file.readlines() # Membaca semua baris ke dalam list
7
8 # Menampilkan isi file sebelum diubah
9 print("Isi file sebelum diubah:")
10 for baris in isi_file:
11     print(baris.strip()) # Menghapus karakter newline
12
13 # Mengubah isi file (misalnya, menambahkan teks)
14 isi_file.append("Ini adalah baris baru yang ditambahkan.\n")
15
16 # Menulis ulang isi file
17 with open(nama_file, 'w') as file:
18     file.writelines(isi_file) # Menulis kembali semua baris ke file
19
20 # Membaca kembali isi file yang baru
21 with open(nama_file, 'r') as file:
22     isi_file_baru = file.readlines()
23
```

```

21 with open(nama_file, 'r') as file:
22     isi_file_baru = file.readlines()
23
24 # Menampilkan isi file setelah diubah
25 print("\nIsi file setelah diubah:")
26 for baris in isi_file_baru:
27     print(baris.strip()) # Menghapus karakter newline

```

## Penjelasan Program

### 1. Membaca Isi File:

- Program membuka file `data.txt` dalam mode baca (`'r'`) dan membaca semua baris ke dalam list `isi\_file`.

### 2. Menampilkan Isi Sebelum Diubah:

- Program mencetak setiap baris dari `isi\_file` sebelum diubah.

### 3. Mengubah Isi File:

- Program menambahkan baris baru ke dalam list `isi\_file`.

### 4. Menulis Ulang Isi File:

- Program membuka file `data.txt` dalam mode tulis (`'w'`) dan menulis kembali semua baris yang ada di `isi\_file` ke dalam file.

### 5. Membaca Kembali Isi File:

- Program membuka file `data.txt` lagi dalam mode baca untuk mendapatkan isi yang baru.

### 6. Menampilkan Isi Setelah Diubah:

- Program mencetak setiap baris dari `isi\_file\_baru` setelah perubahan.

## Catatan

- Pastikan file `data.txt` sudah ada sebelum menjalankan program ini. Jika file tidak ada, Anda bisa membuatnya terlebih dahulu atau menambahkan logika untuk membuat file baru jika tidak ditemukan.

(Jika menggunakan LLM atau Referensi website dalam pembuatan laporan, baik untuk generate code ataupun penulisan text silahkan sertakan dokumentasinya bisa berupa link ataupun screenshot percakapan. Jika tidak melampirkan dan ketahuan menggunakan nilai = 0)

Contoh :

1. [Link Percakapan LLM](#)
2. [Web Referensi - DuniaIlkom](#)