

LAPORAN PRAKTIKUM
PEMROGRAMAN 1
MODUL 6



Oleh:

Nama : Alfian Mutakim

Nim : 2211104024

Kelas : SE06A

PRODI S1 REKAYASA PERANGKAT LUNAK
FAKULTAS INFORMATIKA
INSTITUT TEKNOLOGI TELKOM PURWOKERTO
2023

I. DASAR TEORI

Method adalah suatu bagian dari program yang digunakan untuk menjalankan suatu tugas tertentu dan letaknya terpisah dari bagian program yang menggunakannya. Suatu method dipanggil/digunakan dengan tujuan khusus, yaitu untuk mengerjakan suatu tugas tertentu, dimana tugas – tugas tersebut dapat berupa tugas input (menyimpan hasil ke dalam suatu array atau file) dan/atau output (menampilkan hasil di layar monitor) ataupun melakukan penyeleksian dan perhitungan.

1. PEMBAHASAN.

Method terbagi menjadi 2 macam, yakni method dengan pengembalian (function) dan method tanpa pengembalian (procedure).

a. FUNCTION.

Fungsi pada python dibuat dengan kata kunci 'def' kemudian diikuti dengan nama fungsinya. Penamaan fungsi harus mudah dipahami oleh siapa saja termasuk orang awam. Fungsi dapat dipanggil dengan cara memanggil nama fungsinya langsung. Fungsi juga dapat dipanggil pada fungsi lain, bahkan bisa memanggil dirinya sendiri. Fungsi yang memanggil dirinya sendiri, disebut fungsi rekursif. Pada fungsi kita menggunakan 'return' untuk mengembalikan nilai.

Cara mendeklarasikan fungsi sebagai berikut:

```
def hitng_luas_persegi(sisi):  
    hasil = sisi * sisi  
    return hasil  
  
print("Luas Persegi : %d" % hitng_luas_persegi(5))
```

b. PROSEDUR.

Dalam python, fungsi yang tidak mengembalikan nilai disebut prosedur. Cara mendeklarasikan prosedur adalah sebagai berikut:

```
def hitung_luas_segitiga(alas, tinggi):  
    print(f"Luas Segitiga: {alas*tinggi/2}")  
  
hitung_luas_segitiga(10, 5)
```

c. PARAMETER.

Parameter adalah variabel yang menampung nilai untuk diproses di dalam fungsi. Dengan adanya parameter suatu fungsi akan bersifat dinamis. Parameter diberikan pada saat deklarasi fungsi.



```
def fungsi(parameter):
    print parameter
```

Jenis – jenis parameter :

- Parameter masukan: adalah parameter yang digunakan menampung nilai yang akan dijadikan masukan (input) ke dalam suatu fungsi.
- Parameter keluaran: adalah parameter yang digunakan menampung nilai yang akan dijadikan keluaran (output) yang akan dikirimkan ke bagian yang memanggil fungsi tersebut.
- Parameter masukan/keluaran: adalah parameter yang digunakan menampung nilai yang akan dijadikan masukan (input) ke dalam suatu fungsi selain itu juga menampung nilai yang akan dijadikan keluaran (output) yang akan dikirimkan ke bagian yang memanggil fungsi tersebut.

Contoh:

```
def salam(ucapan):
    print(ucapan)
```

kemudian panggil fungsi tersebut:

```
salam("asalamualaikum")
```

3 Ketika program dijalankan maka akan muncul kata “asalamualaikum”. Kata “asalamualaikum” adalah parameter. Lalu bagaimana jika parameternya lebih dari satu? Kita dapat memisahkan parameter dengan tanda koma (’,’)

Contoh:

```
print('-----')
def hitung_luas_segitiga(alas, tinggi):
    hasil = (alas * tinggi) / 2
    print("Luas Segitiga %d" % hasil)
```

Jalankan aplikasi dengan memanggil fungsi beserta parameternya:

```
hitung_luas_segitiga(10, 5)
```

ketika program dijalankan akan muncul:

```
ers/User/Documents/Praktikum
Luas Segitiga 25
PS C:\Users\User\Documents
```

d. PERBEDAAN FUNGSI & PROSEDUR.

- Fungsi hanya akan mengembalikan satu nilai ke bagian yang memangginya.
- Fungsi hanya mengerjakan satu tugas.
- Prosedur dapat mengembalikan lebih dari satu nilai atau bahkan tidak sama sekali ke bagian yang memangginya.
- Prosedur dapat mengerjakan lebih dari satu tugas

II. GUIDED

1. Buatlah program yang akan menghitung luas dan keliling dari persegi berdasarkan masukan dari pengguna! Dengan output sebagai berikut:

```
Masukan panjang sisi: 20
Keliling persegi: 80
Luas persegi: 400
```

- a. Berikut kode program menggunakan function:

```
# Menggunakan Function
print('-----')
def keliling_persegi(sisi):
    keliling = 4 * sisi
    return keliling

def luas_persegi(sisi):
    luas = sisi * sisi
    return luas

s = int(input("Masukkan sisi :"))
print("Keliling : %d" % keliling_persegi(s))
print("Luas : %d" % luas_persegi(s))
```

Output yang dikeluarkan:

```
Masukkan sisi :4
Keliling : 16
Luas : 16
PS C:\Users\User\Documents\B
```

- b. Berikut kode program menggunakan prosedur:

```
# Menggunakan Prosedur
def keliling_persegi(sisi):
    keliling = 4 * sisi
    luas = sisi * sisi
    print("Keliling : %d" % keliling)
    print("Luas : %d" % luas)

s = int(input("Masukkan panjang sisi :"))
keliling_persegi(s)
```

Output yang dikeluarkan:

```
ers/User/Documents/Praktikum Pemograman
Masukkan panjang sisi :3
Keliling : 12
Luas : 9
PS C:\Users\User\Documents\Praktikum
```

2. Membuat sebuah program perbandingan bilangan (lebih kecil dan lebih besar), menggunakan function! Dengan output sebagai berikut:

```
Masukan bilangan 1: 10
Masukan bilangan 2: 20
bilangan yang lebih besar adalah
20
```

Berikut kode programnya:

```
# Latihan perbandingan bilangan function
print('-----')
def bilangan1(bil1, bil2):
    if bil1 > bil2:
        print("bilangan yang lebih besar adalah ", bil1)
    elif bil1==bil2:
        print("idak ada bilangan yang paling besar")
    else:
        print("bilangan yang lebih besar adalah ", bil2)

awal = int(input("Masukkan Bilangan 1 : "))
akhir = int(input("Masukkan Bilangan 2 : "))
bilangan1(awal, akhir)
```

output yang dihasilkan:

```
-----
Masukkan Bilangan 1 : 4
Masukkan Bilangan 2 : 10
bilangan yang lebih besar adalah 10
PS C:\Users\User\Documents\Praktikum Pemog
```

III. UNGUIDED

1. Buatlah program untuk menampilkan nilai bilangan ganjil atau genap dari bilangan yang dimasukkan dengan menggunakan method procedure.

Kode program:

```
def hitung(angka):  
    if angka ==0:  
        print("bilangan yang anda masukkan adalah bilangan nol")  
    else:  
        if angka%2==0:  
            print("bilangan yang anda masukkan adalah bilangan genap")  
        else:  
            print("bilangan yang anda masukkan adalah bilangan ganjil")  
  
bilangan = int(input("Masukkan Bilangan : "))  
hitung(bilangan)
```

output yang dihasilkan:

```
PS C:\Users\User\Documents\Praktikum Pemograman1\lapModul6_SE06>  
Masukkan Bilangan : 4  
bilangan yang anda masukkan adalah bilangan genap  
PS C:\Users\User\Documents\Praktikum Pemograman1\lapModul6_SE06>  
Masukkan Bilangan : 0  
bilangan yang anda masukkan adalah bilangan nol  
PS C:\Users\User\Documents\Praktikum Pemograman1\lapModul6_SE06>
```

Kode program tersebut merupakan sebuah program untuk menampilkan nilai bilangan ganjil atau genap dari bilangan yang dimasukkan oleh pengguna, fungsi yang disebut "hitung" dengan parameter "angka". Fungsinya adalah untuk memeriksa apakah sebuah bilangan yang dimasukkan oleh pengguna merupakan bilangan nol, bilangan genap, atau bilangan ganjil.

Jika angka yang dimasukkan adalah 0, maka program akan mencetak "bilangan yang anda masukkan adalah bilangan nol". Jika angka tersebut bukan 0, maka program akan melanjutkan pengecekan apakah angka tersebut genap atau ganjil. Jika angka tersebut habis dibagi 2 ($\text{angka} \% 2 == 0$), maka program akan mencetak "bilangan yang anda masukkan adalah bilangan genap". Jika angka tersebut tidak habis

dibagi 2, maka program akan mencetak "bilangan yang anda masukkan adalah bilangan ganjil".

Setelah itu, program akan meminta pengguna memasukkan sebuah bilangan dan akan memanggil fungsi "hitung" dengan parameter bilangan yang dimasukkan oleh pengguna. Fungsi "hitung" kemudian akan mengevaluasi bilangan tersebut dan mencetak hasilnya berdasarkan kondisi yang dijelaskan di atas.

2. Buatlah program untuk menghitung luas lingkaran dan keliling dengan menggunakan method function. Jari – jari adalah masukan dari pengguna.

Kode program:

```
# Menggunakan Function
print('-----')
def keliling_lingkaran(jari):
    keliling = 2 * 3.14 * jari
    return keliling

def luas_lingkaran(jari):
    luas = 3.14 * (jari*jari)
    return luas

s = int(input("Masukkan jari-jari :"))
print("Keliling : %d" % keliling_lingkaran(s))
print("Luas : %d" % luas_lingkaran(s))
```

Output yang dihasilkan:

```
-----
Masukkan jari-jari :20
Keliling : 125
Luas : 1256
PS C:\Users\User\Documents\Praktikum
```

Program Python di atas memiliki dua fungsi yaitu `keliling_lingkaran` dan `luas_lingkaran` yang digunakan untuk menghitung keliling dan luas lingkaran berdasarkan jari-jari yang diberikan.

Fungsi `keliling_lingkaran` mengambil jari-jari sebagai argumen dan menghitung keliling lingkaran dengan rumus $2 * 3.14 * \text{jari}$. Kemudian, nilai keliling dikembalikan sebagai hasil fungsi.

Fungsi `luas_lingkaran` juga mengambil jari-jari sebagai argumen dan menghitung luas lingkaran dengan rumus $3.14 * (\text{jari} * \text{jari})$. Nilai luas dikembalikan sebagai hasil fungsi.

Pada bagian utama kode, program meminta pengguna untuk memasukkan jari-jari lingkaran menggunakan `input` dan menyimpan nilainya dalam variabel `s`. Kemudian, program mencetak keliling lingkaran dengan memanggil fungsi `keliling_lingkaran(s)` dan mengeksekusinya. Hal yang sama dilakukan untuk mencetak luas lingkaran dengan memanggil fungsi `luas_lingkaran(s)`.

3. Buatlah sebuah kalkulator sederhana untuk melakukan kalkulasi 2 bilangan dengan menggunakan method function atau procedure.

Kode program:

```
# KALKULATOR SEDERHANA

def penjumlahan ():
    bil1 = int(input("Bilangan pertama: "))
    bil2 = int(input("Bilangan kedua: "))
    jumlah = bil1 + bil2
    print("hasil penjumlahan : %d" %jumlah)

def kali():
    bil1 = int(input("Bilangan pertama: "))
    bil2 = int(input("Bilangan kedua: "))
    perkalian = bil1 * bil2
    print("hasil perkalian : %d" %perkalian)

def bagi():
    bil1 = int(input("Bilangan pertama: "))
    bil2 = int(input("Bilangan kedua: "))
    pembagian = bil1 / bil2
    print("hasil pembagian : %d" %pembagian)

def kurang():
    bil1 = int(input("Bilangan pertama: "))
    bil2 = int(input("Bilangan kedua: "))
    pengurangan = bil1 - bil2
    print("hasil pengurangan : %d" %pengurangan)

def perpangkatan():
    bil1 = int(input("Bilangan pertama: "))
    bil2 = int(input("Bilangan kedua: "))
    pangkat = bil1**bil2
    print("hasil pangkat : %d" %pangkat)
```

```

while True:
    print("=====Kalkulator Sederhana=====")
    print()
    print("1. Penjumlahan")
    print("2. Perkalian")
    print("3. Pembagian")
    print("4. Pengurangan")
    print("5. Pangkat")
    print()
    a = int(input("Masukkan pilihan anda: "))
    if a == 1:
        penjumlahan()
    elif a == 2:
        kali()
    elif a == 3:
        bagi()
    elif a == 4:
        kurang()
    elif a == 5:
        perpangkatan()
    else:
        print("pilihan yang anda masukkan tidak valid")

    ulangi = input("Apakah Anda ingin mengulangi program? (ya/tidak): ")
    if ulangi.lower() != "ya":
        break

```

Output yang dikeluarkan:

```

C:\Users\User\Documents\Praktikum Pemograman1\lapModul6_SE06A_2211104024_Alfian Mutakir>
=====Kalkulator Sederhana=====

1. Penjumlahan
2. Perkalian
3. Pembagian
4. Pengurangan
5. Pangkat

Masukkan pilihan anda: 1
Bilangan pertama: 2
Bilangan kedua: 5
hasil penjumlahan : 7
Apakah Anda ingin mengulangi program? (ya/tidak): ya
=====Kalkulator Sederhana=====

1. Penjumlahan
2. Perkalian
3. Pembagian
4. Pengurangan
5. Pangkat

Masukkan pilihan anda: 4
Bilangan pertama: 8
Bilangan kedua: 2
hasil pengurangan : 6
Apakah Anda ingin mengulangi program? (ya/tidak): tidak
PS C:\Users\User\Documents\Praktikum Pemograman1\lapModul6_SE06A_2211104024_Alfian Mutakir>

```

Kode program tersebut adalah sebuah kalkulator sederhana yang memungkinkan pengguna untuk melakukan operasi penjumlahan, perkalian, pembagian, pengurangan, dan perpangkatan. Cara kerja kode program tersebut:

1. Program akan menampilkan opsi-opsi operasi yang tersedia kepada pengguna.
2. Pengguna diminta untuk memasukkan pilihan operasi yang ingin mereka jalankan.
3. Berdasarkan pilihan pengguna, program akan memanggil fungsi yang sesuai, yaitu ``penjumlahan()``, ``kali()``, ``bagi()``, ``kurang()``, atau ``perpangkatan()``.
4. Setelah operasi selesai dieksekusi, hasilnya akan ditampilkan.
5. Setelah itu, pengguna akan ditanya apakah mereka ingin menggunakan kalkulator lagi.
6. Jika pengguna menjawab "ya", program akan kembali ke awal dan menanyakan pilihan operasi baru.
7. Jika pengguna menjawab "tidak" atau jawaban lain, program akan berhenti.

Dengan menggunakan loop ``while True`` dan `break` statement, program dapat berulang atau berhenti sesuai dengan keinginan pengguna. Ini memungkinkan pengguna untuk terus menggunakan kalkulator atau keluar dari program jika mereka sudah selesai.