

▼ Function

Function (fungsi) adalah sebuah blok statement yang akan dijalankan saat ia dipanggil.

Fungsi `len()` untuk mengukur jumlah elemen dalam sebuah list dan fungsi `print()` untuk menampilkan pesan pada layar konsol merupakan contoh dari bentuk fungsi yang telah disediakan oleh bahasa pemrograman Python atau dengan kata lain built-in function.

Selain fungsi yang telah disediakan oleh Python, Python mengizinkan untuk mendeklarasikan fungsi dalam kode yang dituliskan yang dikenal dengan user defined function.

```
def nama_fungsi(argument):
    statement_1
    ...
    statement_n
    return retruned_value
```

- **nama_fungsi** adalah nama untuk fungsi yang anda definisikan, memiliki aturan seperti penamaan variabel.
- **argument** adalah variabel masukan ke fungsi, bisa tanpa variabel, satu variabel atau lebih.
- **statement_1 ... statement_n** adalah algoritma yang telah anda trasnfer dalam bahasa Python yang merupakan inti dari fungsi yang anda definisikan. Seluruh statement adalah menjorok (indent) ke dalam seperti yang pernah anda pelajari pada coditioning dan looping.
- **return_value** adalah variabel output dari fungsi anda, bisa tanpa variabel, satu variabel atau lebih.

▼ Argumen dan Parameter

Parameter adalah variabel yang digunakan dalam deklarasi fungsi

argumen adalah nilai yang diberikan saat memanggil fungsi

```
def print_nama(nama): #'nama adalah parameter'
    print(nama)
print_nama('Budi') #Budi adalah argumen

    Budi
```

▼ Argumen Posisi dan Kata Kunci

Kita bisa menyatakan argumen sebagai argumen posisi atau kata kunci. Beberapa manfaat argumen kata kunci dapat berupa:

- Kita dapat menyebut argumen dengan nama mereka untuk memperjelas apa yang mereka wakili.
- Kita dapat mengatur ulang argumen dengan cara yang membuatnya paling mudah dibaca.

```
def angka(a, b, c):
    print(a, b, c)

#argumen posisi
#argumen berdasarkan posisinya
angka(1, 2, 3)

#argumen berdasarkan kata kuncinya
angka(a=1, b=2, c=3)
angka(c=3, b=2, a=1)
#catatan: jika sudah diberi kunci maka urutan tidak menjadi penting

#campuran antara argumen posisi dan kata kunci
angka(1, b=2, c=3)

#ini tidak diizinkan
#foo(1, b=2, 3) #argumen posisional setelah kata kunci
#foo(1, b=2, a=3) #terdapat dua nilai untuk argumen kunci 'a'

1 2 3
1 2 3
1 2 3
1 2 3
```

▼ Argumen Default

Fungsi dapat memiliki argumen default dengan nilai yang telah ditentukan sebelumnya. Argumen dapat ditinggalkan dan nilai default kemudian diteruskan ke fungsi, atau argumen dapat diubah dengan nilai yang berbeda. Perhatikan bahwa argumen default harus ditentukan sebagai **parameter terakhir** dalam suatu fungsi.

```
#argumen default
def angka2(a, b, c, d=4):
    print(a, b, c, d)

angka2(1, 2, 3, 4)
angka2(1, b=2, c=3, d=100)

#argumen default harus diletakkan diakhir

1 2 3 4
1 2 3 100
```

▼ Praktik Membuat Fungsi

```
x = 1
y = 2
x*y

2

def penjumlahan(x,y): #input
    a=x+y             #proses
    return a           #output

penjumlahan(1,2)      #menggunakan fungsi

3
```

Penjumlahan yang memungkinkan tiga bilangan

```
def penjumlahan(x,y,z=None):
    if (z==None):
        return x+y
    else:
        return x+y+z

print(penjumlahan(1, 2))
print(penjumlahan(1, 2, 3))

3
6
```

Menyimpan hasil fungsi dalam variabel

```
def penjumlahan(x,y):
    return x+y

a = penjumlahan
a(1, 2)

3
```

▼ LATIHAN 1

Buatlah Function pythagoras dengan input argumennya panjang tinggi dan alas segitiga untuk menghasilkan output nilai panjang sisi miring

```
import math
a = float(input("masukkan panjang tinggi: "))
b = float(input("masukkan panjang alas: "))
c = math.sqrt(a ** 2 + b ** 2)
print(f"nilai panjang sisi miringnya adalah {c}")
```

```
masukkan panjang tinggi: 3
masukkan panjang alas: 4
nilai panjang sisi miringnya adalah 5.0
```

atau

```
import math

def pythagoras(panjang_tinggi, panjang_alas):
    panjang_miring = math.sqrt((panjang_tinggi ** 2) + (panjang_alas ** 2))
    return panjang_miring

# Contoh penggunaan fungsi
tinggi = 3
alas = 4
hasil_miring = pythagoras(tinggi, alas)
print("Panjang sisi miring segitiga adalah:", hasil_miring)

    Panjang sisi miring segitiga adalah: 5.0
```

▼ LATIHAN 2

Buatlah function luas permukaan balok dengan input argumennya panjang, lebar, dan tinggi dari balok untuk menghasilkan output nilai luas permukaan balok tersebut

```
panjang = 15
lebar = 5
tinggi = 10

def luas_balok(p, l, t):
    return ((p*l) + (p*t) + (l*t))*2

luas_balok(15, 5, 10)

    550

def luas_permukaan_balok(panjang, lebar, tinggi):
    luas_permukaan_balok = (2*(panjang*lebar+panjang*tinggi+lebar*tinggi))
    return luas_permukaan_balok

#Eksekusi penggunaan fungsi
panjang = int(input("panjang :"))
lebar = int(input("lebar :"))
tinggi = int(input("tinggi :"))
luas_permukaan_balok = luas_permukaan_balok(panjang, lebar, tinggi)
print("luas permukaan baloknya adalah :", luas_permukaan_balok)

    panjang :15
    lebar :10
    tinggi :5
    luas permukaan baloknya adalah : 550
```