

# Programming II:

Intermediate Python



# Hello, everyone!

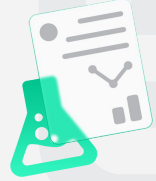


## Sebelum kita memulai kelas, kita awali dengan:

1. Berdoa
2. siapkan diri



# Pertemuan lalu...

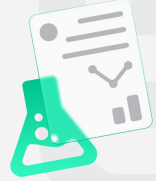


## Let's Recall

pertemuan sebelumnya kita sudah membahas pengenalan programming python. Ada yang inget apa aja?



# Target



1. Introduction to Iteration and Loop
2. Flowchart of loop
3. While statement
4. For statement
5. Multiple loop
6. What is functional programming?
7. Create functions in Python



# Tools



<https://colab.research.google.com/>

# **1. Introduction to Iteration and Loop**

# Konsep Loop

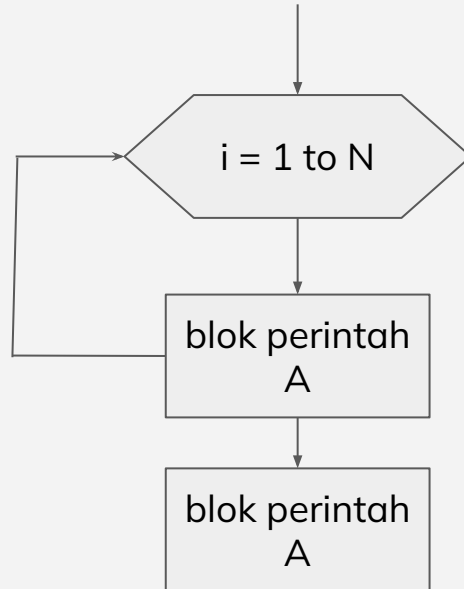
- Loop adalah struktur pemrograman yang mengulangi urutan instruksi tertentu sampai kondisi tertentu terpenuhi.
- Pada looping perlu adanya *exit condition*. Jika *exit condition* yang tidak tepat, loop bisa menjadi infinite loop, artinya program akan terus berjalan tanpa henti. Kejadian tersebut dapat menyebabkan program menjadi tidak responsif atau mengalami *hang*.
- Jenis struktur loop:
  - for loops
  - while loops

## **2. Flowchart of loop**



# For loop

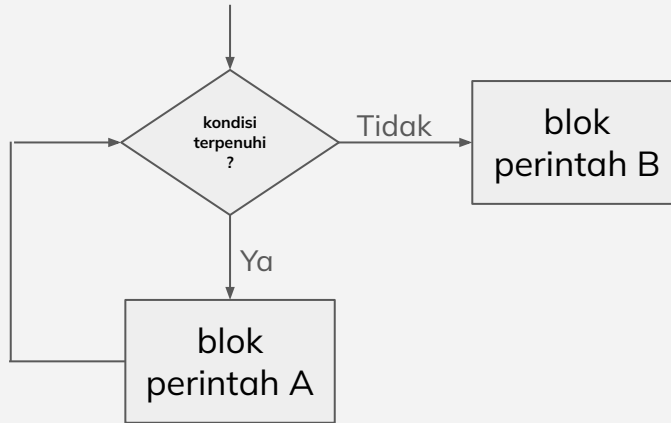
- Pengulangan berdasarkan banyaknya pengulangan sudah ditetapkan/diketahui sebelumnya
- Looping : For – Next



**FOR i = 1 to N**  
**BLOK PERINTAH A**  
**NEXT i**  
**BLOK PERINTAH B**

# While loop

- Pengulangan berdasarkan syarat tertentu yang dipenuhi
- Looping : While – Do



```
While <ekspresi> Do  
    BLOK PERINTAH A  
END WHILE  
BLOK PERINTAH
```

### **3. While statement**

# Konsep While

- Dalam konteks pemrograman Python, dengan menggunakan while loop dapat membuat program melakukan tugas secara berulang-ulang selama kondisi tertentu terpenuhi.
- 'while' adalah salah satu tipe loop atau perulangan dalam Python yang dapat menjalankan blok kode berulang kali selama kondisi yang ditentukan masih memenuhi.
- Penggunaan python 'while' perlu menentukan kriteria tertentu yang bisa meng-update kondisi di dalam while loop.
- Tujuan dari penentuan kriteria ini adalah mencegah program berjalan selamanya atau infinite loop yang menyebabkan program menjadi hang. Dengan demikian, selalu pastikan kondisi yang ditentukan di dalam while loop akan terpenuhi pada suatu titik sehingga loop bisa berhenti.

# While loop statement Python

- Pada gambar 1, struktur 'while' diikuti oleh kondisi yang akan dievaluasi. Selama kondisi ini bernilai benar atau 'True', blok kode di dalam loop akan terus dijalankan. Namun, begitu kondisi menjadi salah atau 'False', loop akan berhenti.

▼ While loop

```
[ ] while kondisi:
    # blok kode yang dijalankan
```

✓  
0d [2] i = 1  
while i < 20:  
 print(i)  
 i += 8

1  
9  
17

(1)

(2)

# While loop statement Python

- Pada contoh di samping, Terdapat dua list, yaitu 'tenant' yang berisi nama pengguna platform dan 'loyal\_tenant' yang berisi pengguna loyal. Tahap 'i' dengan 0 dan menjalankan perulangan 'while' selama 'i' kurang dari jumlah elemen dalam 'tenant'.

```
✓ [5] tenant = ['Siska', 'Putra', 'Ayu', 'Roy', 'Rizki']  
0d loyal_tenant = ['Putra', 'Ayu', 'Siska']  
  
i=0  
while i < len(tenant):  
    if tenant[i] in loyal_tenant:  
        print(tenant[i], 'loyal tenant.')  
    else:  
        print (tenant[i], 'tidak loyal tenant.')  
    i += 1  
  
Siska loyal tenant.  
Putra loyal tenant.  
Ayu loyal tenant.  
Roy tidak loyal tenant.  
Rizki tidak loyal tenant.
```

## **4. For statement**

# Konsep Dasar For Loop

- for...loop statement adalah salah satu konsep paling fundamental digunakan untuk **iterasi kumpulan data tanpa kondisi apapun**.
- For loop memiliki prinsip dasar sederhana dimana statement akan berjalan atau berulang kali melakukan blok kode tertentu berdasarkan jumlah elemen dalam kumpulan atau koleksi data.
- Command/perintah pada for...loop statement ini adalah:

```
for... in...  
print ( )
```

Artinya untuk menggunakan for...loop gunakan command/perintah for ... lalu ketik in ... lalu ketik print untuk menampilkan

▼ for loop



```
for... in...  
print ( )
```



# Konsep Dasar For Loop

- Fungsi range() dapat digunakan dengan dua atau tiga argumen
  - fungsi range(10) menghasilkan deret angka dari 0 sampai 9.
  - range(start,stop): *output* yang akan keluar adalah angka mulai dari 'start' hingga 'stop - 1'.
  - range(start, stop, step): *output* yang akan keluar adalah mulai dari 'start' meningkatkan dengan 'step' di setiap langkah, dan berhenti sebelum 'stop'.

## for loop

```
0d [19] for i in range(10):  
      print(i)
```

0  
1  
2  
3  
4  
5  
6  
7  
8  
9

## range(start, stop)

```
0d [21] for a in range(10,15):  
      print(a)
```

10  
11  
12  
13  
14

## range(start, stop, step)

```
0d [22] for b in range(1,64,8):  
      print(b)
```

1  
9  
17  
25  
33  
41  
49  
57

# Konsep Dasar For Loop

- For loop dapat digunakan untuk melakukan iterasi melalui setiap karakter dalam sebuah string.
- For loop dengan string berfungsi untuk melakukan pemrosesan atau analisis karakter-demi-karakter dalam sebuah string.

## ✓ for loop string

```
[24] for text in 'belajar kampus merdeka':  
      print(text)
```

b  
e  
l  
a  
j  
a  
r

k  
a  
m  
p  
u  
s

m  
e  
r  
d  
e  
k  
a


## 5. Multiple loop

# Konsep Multiple Loop

- Multiple loop adalah loop yang berada di dalam loop lain.
- Bahasa pemrograman Python memungkinkan penggunaan satu lingkaran di dalam loop lain.

# Ilustrasi Multiple Loop

## multiple loop for

```
0d  for i in range(1, 5):  
    # multiple loop  
    # to iterate from 1 to 10  
    for j in range(1, 11):  
        # print multiplication  
        print(i + j, end=' ')  
    print()
```

```
 2 3 4 5 6 7 8 9 10 11  
3 4 5 6 7 8 9 10 11 12  
4 5 6 7 8 9 10 11 12 13  
5 6 7 8 9 10 11 12 13 14
```

- For pertama akan melakukan perulangan angka dari 1 hingga 4
- For kedua akan melakukan perulangan angka dari 1 hingga 10
- Selanjutnya, Untuk setiap iterasi for pertama dan for kedua akan dieksekusi sepuluh kali. Karena kita mencetak tabel penjumlahan hingga sepuluh.
- Dalam setiap iterasi perulangan akan menghitung penjumlahan dua angka.

## Multiple loop While loop dan for loop

```
0d [14] tenant = ['Siska', 'Putra', 'Ayu', 'Roy', 'Rizki']  
for user in tenant:  
    count = 0  
    while count < 3:  
        print(user, end=' ')  
        count = count + 1  
    print()
```

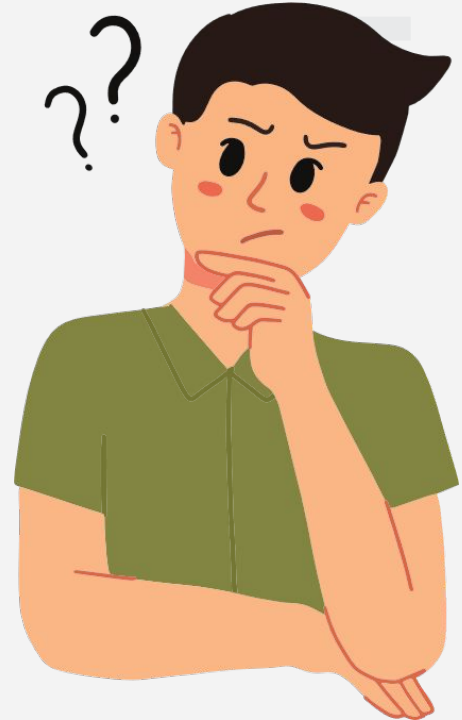
```
Siska Siska Siska  
Putra Putra Putra  
Ayu Ayu Ayu  
Roy Roy Roy  
Rizki Rizki Rizki
```

- Contoh ini akan *looping* list tenant menggunakan perulangan for
- Pada while nilai awal count adalah 0, selama count kurang dari 3. disetiap iterasi count akan ditambah dengan 1
- Dalam setiap iterasi perulangan for, perulangan for dijalankan sebanyak tiga kali untuk mencetak tenant saat ini sebanyak tiga kali

## **6. What is functional programming?**

# Apaitu *functional programming* ?

- suatu bagian dari program yang digunakan untuk menjalankan suatu tugas tertentu dan letaknya terpisah dari bagian program yang menggunakannya.
- Suatu fungsi/prosedur dipanggil/digunakan dengan tujuan khusus, yaitu untuk mengerjakan suatu tugas tertentu, dapat berupa :
  - Tugas input (menyimpan hasil ke dalam suatu array atau file) dan/atau
  - Tugas output (menampilkan hasil di layar monitor) ataupun
  - melakukan penyeleksian dan perhitungan
- Suatu Fungsi dapat memberikan suatu hasil balik ke program yang memanggilnya atau tidak memberikan hasil balik sama sekali.
- Hasil balik ini biasanya berupa suatu nilai yang dibutuhkan oleh bagian program yang memanggilnya.



## **7. Create functions in Python**

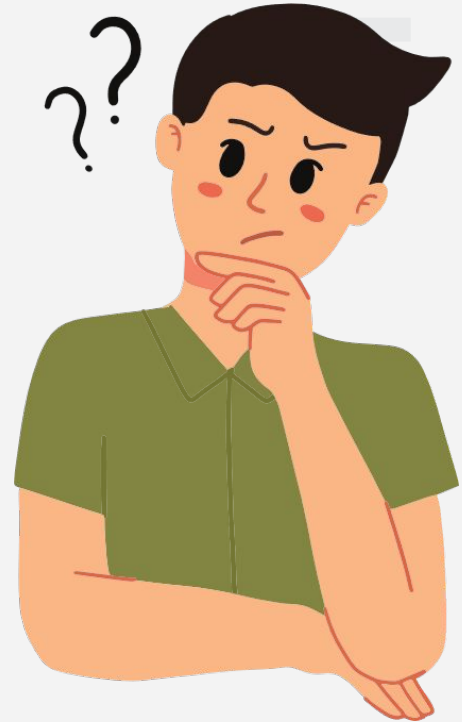


# Apaitu *functional programming* di Python?

- Function dalam Python berguna untuk menjalankan sebuah blok berisikan script Python yang berfungsi untuk menjalankan suatu task tertentu.
- Jenis-jenis function di python lho, yaitu:
  - Function tanpa Parameter
  - Function dengan Parameter

function in python

```
def fungsi():  
    print("contoh fungsi")
```



# Function tanpa Parameter

- Function ini bertugas untuk menjalankan script tanpa syarat tertentu.

## Function Tanpa Parameter

```
✓ [1] def selisih():  
0d     diff = 5 - 3  
     return diff
```

```
✓ [2] selisih()  
0d
```

2

# Function dengan Parameter

- Parameter adalah nilai yang dimasukkan ke dalam function, seperti syarat atau konstanta yang harus dipenuhi
- Jika tidak dipenuhi maka program tidak akan berjalan dan menghasilkan output yang diinginkan
- contoh gambar disamping line 6 artinya kita sedang membuat fungsi operasi akar kuadrat dengan sqrt dan parameter yang didefinisikan adalah x. kita harus input x terlebih dahulu dan akan keluar output dari fungsi tersebut.
- contoh gambar disamping line 8 artinya kita sedang membuat fungsi kuadrat dimana diberi x, dengan  $y = (x^2) + (2x) - 15$ . fungsi didefinisikan kuadratik dan parameter x

## Function dengan Parameter

```
0d [6] def sqrt(x):  
      akar = x**(1/2)  
      return akar
```

```
0d [7] sqrt(25)  
  
5.0
```

```
0d [8] def kuadratik(x):  
      y = (x**2)+(2*x)-15  
      print ("jika x=",x , "maka y=",y)  
      return
```

```
0d [9] kuadratik(10)  
  
jika x= 10 maka y= 105
```

**NEXT!!**

Programming III:  
Functions in Python



# Terima Kasih

