

Program Fresh Graduate Academy Digital Talent Scholarship 2019 | Machine Learning

Pemrograman Python : Percabangan

M. Ramli & M. Soleh





Bagian 1

Operator Perbandingan



Operator Perbandingan

- Operator perbandingan digunakan untuk membandingkan dua bilangan.
- Operator perbandingan seperti: sama dengan (==), tidak sama dengan (!=), lebih besar (>), lebih besar sama dengan (>=), lebih kecil (<), lebih kecil sama dengan (<=)
- Operator perbandingan akan memberikan hasil berupa tipe boolean (True atau false)

```
a=6  
a==7
```



False

```
a=6  
a==6
```

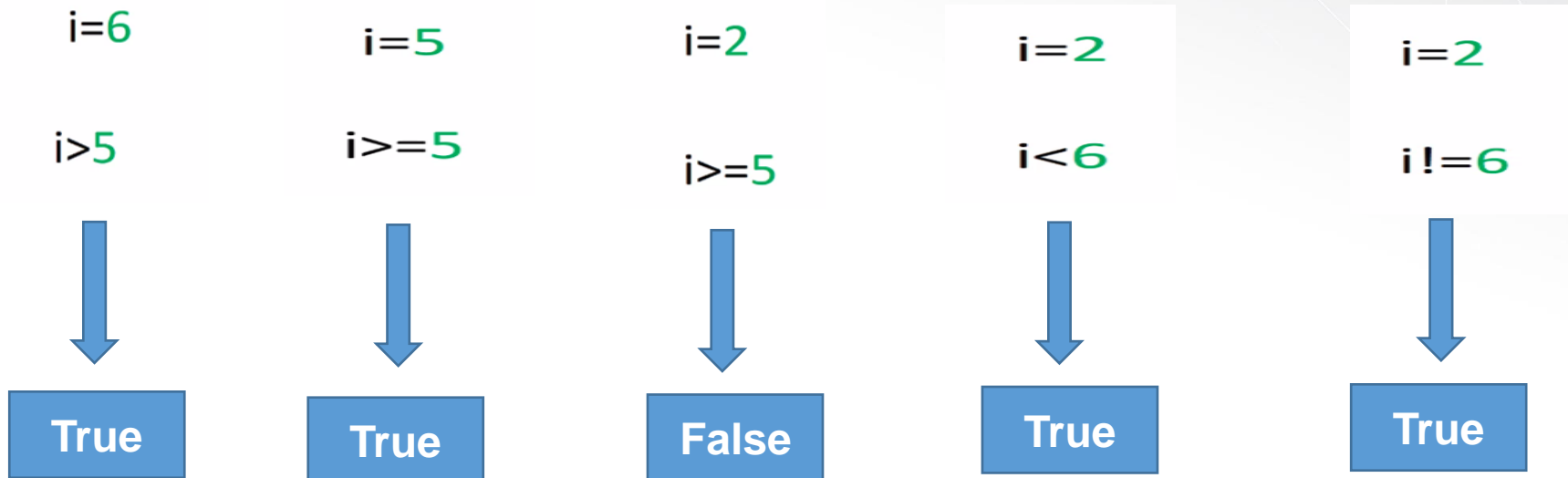


True



Operator Perbandingan

Contoh operator perbandingan





Operator Perbandingan

Operator perbandingan dapat digunakan untuk membandingkan tipe data string.

```
"AC/DC"=="Michael Jackson"
```

False

```
"AC/DC"!="Michael Jackson"
```

True



Bagian 2

Percabangan

Percabangan

Pada percabangan dapat memasukkan pernyataan yang berbeda dan memberikan keluaran yang berbeda.

Bentuk Umum:

Jika BENAR maka....

Jika SALAH maka...

Contoh penulisan percabangan pada Python (menggunakan **IF**) :

Kondisi



```
if (age>18):
```



Titik dua (:)

Akan dilakukan jika
kondisi TRUE



```
print("you can enter" )
```

Akan dilakukan jika
kondisi FALSE



```
print("move on")
```

Percabangan

Contoh ke-1 penggunaan percabangan dan kondisi:

```
age=17
```

```
if (age>18):
```

```
    print("you can enter" )
```

```
print("move on")
```

Apakah $17 > 18$?
Jawab: FALSE

Karena FALSE, maka pilihan
“move on” yang dikerjakan

Percabangan

Contoh ke-2 penggunaan percabangan dan kondisi:

```
age=19
```

```
if (age>18):
```

Apakah $19 > 18$?
Jawab: TRUE

```
print("you can enter" )
```

Karena TRUE, maka yang dijalankan adalah "you can enter" kemudian "move on"

```
print("move on")
```

Percabangan

Percabangan bisa menggunakan perintah **IF..... ELSE....**

Perintah setelah ELSE akan dilakukan jika kondisi adalah FALSE

Contoh ke-1:

```
age=17
```

```
if (age>18):
```

```
    print("you can enter" )
```

```
else:
```

```
    print("go see Meat Loaf" )
```

```
    print("move on")
```

Apakah $17 > 18$?
Jawab: FALSE

Karena FALSE, maka yang dijalankan adalah "go see meat loaf" kemudian "move on"

Percabangan

Percabangan bisa menggunakan perintah **IF..... ELSE.....**

Perintah setelah ELSE akan dilakukan jika kondisi adalah FALSE

Contoh ke-2:

```
age=19
```

```
if (age>18):
```

```
    print("you can enter" )
```

```
else:
```

```
    print("go see Meat Loaf" )
```

```
print("move on")
```

Apakah $19 > 18$?
Jawab: TRUE

Karena TRUE, maka yang dijalankan adalah "you can enter" kemudian "move on"

Percabangan

Percabangan bisa menggunakan perintah **IF..... ELSEIF....ELSE....**

Perintah ini memberikan lebih dari 1 (satu) kondisi yang dapat diseleksi.

Contoh ke-1:

```
age=18
```

```
if (age>18):
```

```
    print("you can enter" )
```

```
elif(age==18):
```

```
    print("go see Pink Floyd" )
```

```
else:
```

```
    print("go see Meat Loaf" )
```

```
print("move on")
```

Apakah $18 > 18$?
Jawab: FALSE, maka
seleksi kondisi pada
ELSEIF

Apakah $18 = 18$?
Jawab: TRUE

Karena TRUE, maka yang
dijalankan adalah "go see
Pink Floyd" kemudian "move
on"



Bagian 3

Operator Logika

Operator Logika

Merupakan operator yang akan memberikan hasil Boolean (True atau False).

Berikut adalah tabel untuk **operator logika OR**

A	B	A or B
False	False	False
False	True	True
True	False	True
True	True	True

Operator Logika

Operator logika OR

Contoh:

```
album_year = 1990
```

```
if (album_year < 1980) or (album_year > 1989):
```

```
    print ("The Album was made in the 70 's or 90's")
```

```
else:
```

```
    print("The Album was made in the 1980's ")
```

Jika salah satu kondisi TRUE,
maka akan menjalankan
perintah sebelum ELSE

Jika kedua FALSE, maka akan
menjalankan perintah setelah
ELSE

Output program:
The Album was made in the 70's or 90's

Operator Logika

Berikut adalah tabel untuk **operator logika AND**

A	B	A & B (AND)
False	False	False
False	True	False
True	False	False
True	True	True

Operator Logika

Operator logika AND

Contoh:

```
album_year = 1983
```

```
if(album_year > 1979) and (album_year < 1990):
```

```
    print ("This album was made in the 80's ")
```

This album was made in the 80's

Output program

Jika kedua kondisi TRUE, maka akan menjalankan
print ("This album was made in the 80's")

Jika salah satu kondisi TRUE
atau FALSE, maka akan keluar
dari program

Program Fresh Graduate Academy Digital Talent Scholarship 2019 | Machine Learning

Pemrograman Python : Perulangan





Bagian 1

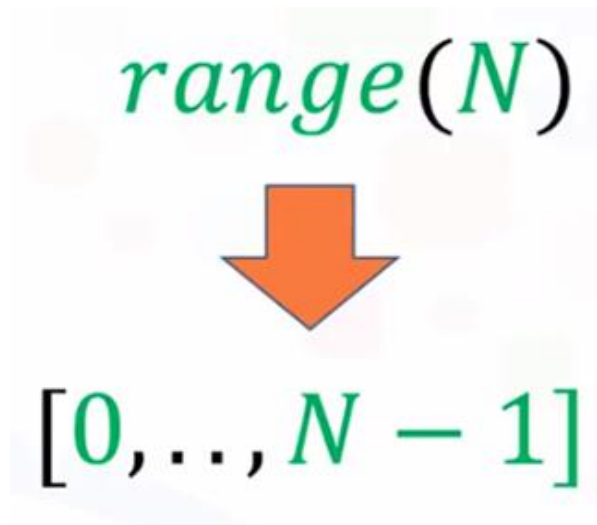
Fungsi Range

Fungsi Range

Fungsi “Range”

Digunakan untuk menghasilkan deret angka (seperti list) sesuai kondisi batas bawah dan batas atas yang diberikan.

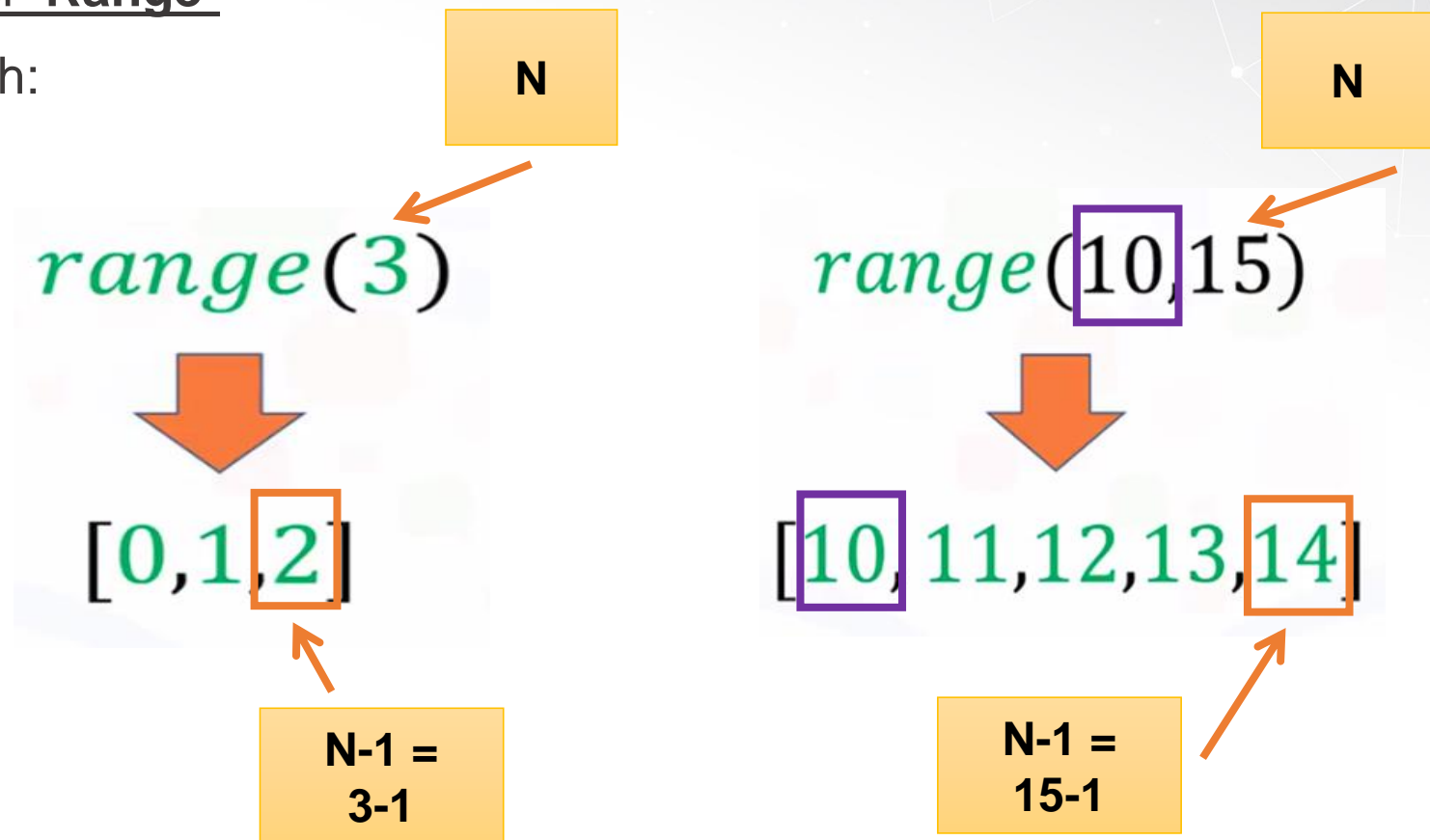
Bentuk Umumnya adalah:



Fungsi Range

Fungsi "Range"

Contoh:






Bagian 2

Perulangan dengan FOR

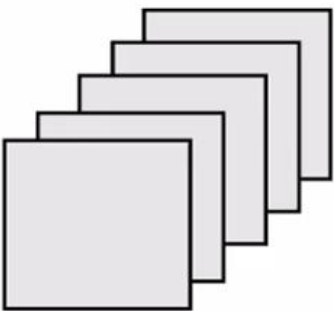
Bentuk Umum:

FOR IN :

Contoh: akan mengganti 5 (lima) kotak berwarna dengan kotak berwarna putih saja.



squares	red	yellow	green	purple	blue
	0	1	2	3	4



```
squares=["red", "yellow ", "green", "purple", "blue "]
```

```
for i in range(0,5):  
    squares[i]="white"
```



Bagian 3

Perulangan dengan WHILE

Perulangan dengan WHILE

Perulangan WHILE digunakan untuk melakukan perulangan berdasarkan kondisi tertentu, dan kondisi tersebut harus terpenuhi.

```
squares=['orange','orange','purple','blue']
```

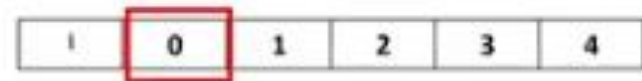
```
Newsquares=[]
```

```
i=0
```

```
while(squares[i]=='orange'):
```

```
    Newsquares.append(squares[i])
```

```
    i=i+1
```



```
while( squares[i] =='orange'):
```

