



“PERULANGAN”

6.1. Tujuan Pembelajaran

- Mahasiswa mampu memahami apa itu Perulangan.
- Mahasiswa mampu menganalisis Perulangan serta memecahkan masalah dengan menggunakan Perulangan.
- Mahasiswa mampu membuat Perulangan sesuai studi kasus yang ada dan mengeksekusinya ke dalam program.

6.2. Penjelasan

A. Pengertian

Perulangan dalam bahasa pemrograman merupakan instruksi yang digunakan untuk memerintahkan komputer melakukan sesuatu secara berulang-ulang dengan jumlah yang ditentukan atau selama masih memenuhi suatu kondisi yang telah ditentukan. Dalam bahasa pemrograman Python terdapat dua macam perulangan, yaitu perulangan **for** dan perulangan **while**.

B. Perulangan For

Perulangan **for** disebut juga sebagai *counted loop* (perulangan yang terhitung). Perulangan **for** biasanya digunakan untuk mengulangi kode yang sudah diketahui banyak perulangannya

Contoh Program :

```
ulang = 10
for i in range(ulang):
    print("Perulangan ke-" + str(i))
```



Output :

```
Perulangan ke-0
Perulangan ke-1
Perulangan ke-2
Perulangan ke-3
Perulangan ke-4
Perulangan ke-5
Perulangan ke-6
Perulangan ke-7
Perulangan ke-8
Perulangan ke-9
```

Penjelasan :

- Variabel **ulang** digunakan untuk menentukan banyak perulangan, yaitu 10.
- Variabel **i** berfungsi untuk menampung indeks
- Fungsi **range()** berfungsi untuk membuat list dengan range dari 0-10.
- Fungsi **str()** berfungsi merubah tipe data integer ke string.

Contoh Lain Penggunaan For :

```
simpan = [12, "udin petot", 14.5, True, 'A']
for i in simpan:
    print(i)
```

Output :

```
12
udin petot
14.5
True
A
```

Penjelasan :

- Terdapat sebuah list bernama **simpan** yang berisi nilai-nilai dengan tipe data yang bervariasi.
- Perulangan **for** digunakan untuk melakukan iterasi atau perulangan melalui setiap nilai yang ada dalam list **simpan**.
- Variabel **i** disini berfungsi untuk menampung nilai dari list tersebut.
- **Print(i)** berfungsi untuk menampilkan isi dari list tersebut.



Contoh Penggunaan For di dalam For :

```
for i in range(1, 4):  
    for j in range(1, 4):  
        print(f"{i} x {j} = {i * j}")  
    print()
```

Output :

```
1 x 1 = 1  
1 x 2 = 2  
1 x 3 = 3  
  
2 x 1 = 2  
2 x 2 = 4  
2 x 3 = 6  
  
3 x 1 = 3  
3 x 2 = 6  
3 x 3 = 9
```

Penjelasan :

- `for i in range(1, 4)` perulangan ini mengontrol baris dalam tabel perkalian. •
 - `for j in range(1, 4)` perulangan ini mengontrol kolom dalam tabel perkalian. •
- Pada perulangan kedua, program menghitung hasil perkalian dari variabel `i` dan `j`.

C. Perulangan While

Perulangan **while** disebut juga sebagai *uncounted loop* (perulangan yang tak terhitung). Perulangan **while** memiliki syarat dan akan terus mengulang apabila memenuhi syarat tersebut. Oleh karena itu, perulangan **while** tidak tentu berapa banyak perulangannya.

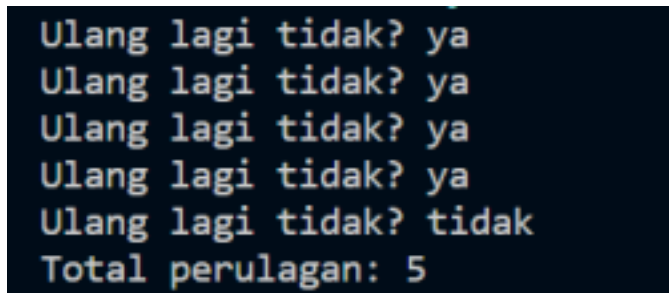
Bentuk umum :

```
while(syarat):  
    # jalankan kode ini
```

Contoh Program :

```
jawab = 'ya'  
hitung = 0  
while(jawab == 'ya'):  
    hitung += 1  
    jawab = input("Ulang lagi tidak? ")  
print(f"Total perulangan: {hitung}")
```

Output :



```
Ulang lagi tidak? ya  
Ulang lagi tidak? ya  
Ulang lagi tidak? ya  
Ulang lagi tidak? ya  
Ulang lagi tidak? tidak  
Total perulangan: 5
```

Penjelasan :

- Variabel **jawab** yang bernilai 'ya' berfungsi sebagai syarat perulangan dan variabel **hitung** dengan nilai 0 berfungsi untuk menghitung jumlah perulangan. • Perulangan **while(jawab == 'ya')** berfungsi agar perulangan selalu berjalan selama variabel **jawab** berisi 'ya'. Jika variabel **jawab** bernilai selain 'ya', maka perulangan akan berhenti.
- Setiap kali perulangan **while** berjalan, nilai **hitung** akan ditambah 1. • Pengguna akan diminta untuk memasukkan jawaban melalui **input**. • Setelah program berhenti, program akan mencetak total jumlah perulangan.



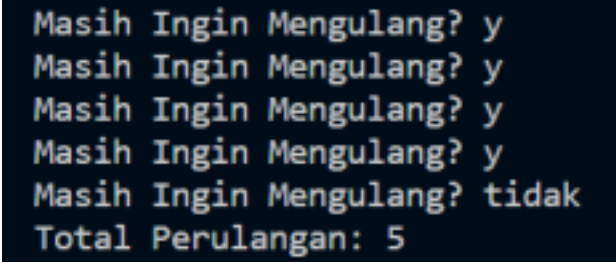
D. Break

Fungsi **break** adalah perintah khusus yang dipakai untuk memaksa sebuah perulangan berhenti sebelum waktunya. Perintah **break** bisa digunakan untuk jenis perulangan baik **while** maupun perulangan **for** pada bahasa pemrograman Python.

Contoh Penggunaan Break :

```
hitung = 0
while True:
    hitung += 1
    ulang = input("Masih Ingin Mengulang? ")
    if ulang == "tidak" or ulang == "Tidak":
        break
    print(f"Total Perulangan: {hitung}")
```

Output :



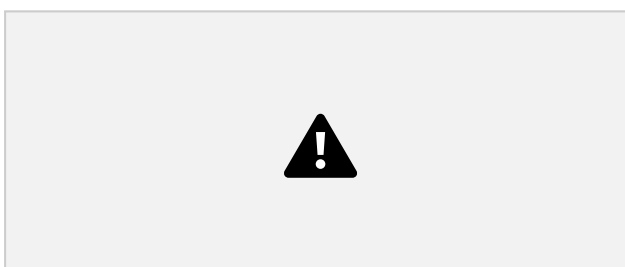
E. Continue

Fungsi **continue** adalah perintah yang digunakan di dalam loop untuk melewati iterasi saat ini dan langsung lanjut ke iterasi berikutnya, tanpa menjalankan kode yang ada setelah pernyataan continue dalam iterasi tersebut.

Contoh Penggunaan Continue:

```
print("Daftar bilangan ganjil dari 1-10")
for i in range(10):
    if i % 2 == 0:
        continue
    print(i)
```

Output :





F. Flowchart Perulangan

Berikut adalah contoh dari flowchart perulangan.



Studi Kasus

1. Buatlah program yang dapat menentukan dan menghitung jumlah bilangan prima dengan *range* bilangan mulai dari 1!
2. Buatlah program perulangan *while* yang menjumlahkan semua inputan integer positif, jika diinput negatif maka program berhenti!