1. Contoh kasus menggunakan pengulangan for dan while

contoh kasus:

Misalkan kita merupakan badan administrasi yang mengurus registrasi siswa yang telah mendaftar ke universitas tertentu dan mendaftarkannya menjadi mahasiswa/ ter registrasi. yang terjadi adalah perubahan status antara siswa dan mahasiswa.

Kita dapat berasumsi bahwa siswa merupakan list 1 dan mahasiswa adalah list ke 2. Kita ingin memindahkan siswa di list 1 menjadi mahasiswa di list2. Otomatis list pertama akan menjadi hilang.

unconfirmed\_users = ['alice', 'brian', 'candace']

confirmed\_users = []

while unconfirmed\_users:

current\_user = unconfirmed\_users.pop()

print("Verifying user: " + current\_user.title())

confirmed\_users.append(current\_user)

print("\nThe following users have been confirmed:")

for confirmed\_user in confirmed\_users:

print(confirmed\_user.title())

1. .**Fungsi For Else, Range, dan Break Pada Python,**seperti yang kita ketahui bahwa perulangan for disebut juga dengan counted loop (perulangan yang terhitung).

For loop digunakan untuk melakukan perulangan atau interaksi sampai batas range yang telah kita tentukan dan for juga biasa digunakan untuk melooping kode yang sudah diketahui banyak perulangannya. Pada looping for kita juga bisa menambahkan fungsi **else**untuk contoh sederhananya seperti dibawah ini :

for i in range(1,6):

print("angka",i)

else:

print("looping selesai")

# hasil :

# angka 1

# angka 2

# angka 3

# angka 4

# angka 5

# looping selesai

ket :

Pada script diatas **else** berfungsi untuk mencetak hasil akhir dari loopingan yang kita buat, nah jika hasil dari loopingan yang kita buat telah mencapai akhir prosesnya maka perintah else akan di jalankan.

**Range** digunakan sebagai counter, jika yang kita masukan range(1,6) sepeti contoh diatas, maka akan menampilkan perulangan sebanyak 6 element yang dimulai dari element pertama yaitu satu (1), dan untuk struktur dari range tersebut yaitu : **range(range awal, range akhir, selisih)**

Berikutnya kita akan coba menambahkan fungsi **break** pada looping for, **break** berfungsi untuk menghentikan looping, jadi jika kita inggin me-looping sebuah data dan kemudian data tersebut ingin kita stop maka kita bisa menggunakan fungsi **break**, untuk contohnya seperti dibawah ini :

angka = 4

for i in range(0,10):

print(i)

if i is angka:

print("angka ditemukan",i)

break #untuk menghentikan proses

else:

print("angka tidak ditemukan",angka)

# hasil :

# 0

# 1

# 2

# 3

# 4

# angka ditemukan 4

ket :

Pada perulangan for diatas jika kita tidak menggunakan fungsi break maka secara otomatis for akan melooping data sebanyak 10 kali sampai dengan batas yang ditentukan, namun kita juga bisa menghentikan proses loopingan tersebut, jika angka yang kita masukan telah ditemukan, maka proses looping akan berhenti.

seperti pada script diatas kita akan menghentikan proses looping pada angka ke-4 jika loopingan telah sampai di angka ke-4 maka semua proses looping akan berhenti.

* **Fungsi Continue dan Pass Pada Pemrograman Python**kita akan membuat sebuah contoh **continue**seperti dibawah ini :

for i in range(1,5):

if i is 2:

print("nilai",i,"ditemukan")

continue

print("nilai setelah continue") # nilai tidak akan dicetak jika proses terpenuhi

print("nilai saat ini adalah",i)

# hasil :

# nilai saat ini adalah 1

# nilai 2 ditemukan

# nilai saat ini adalah 3

# nilai saat ini adalah 4

ket :

Pada script diatas bisa kita lihat bahwa nilai setelah pass masih dapat ditampilkan, dan pass tersebut tidak melakukan fungsi apapun hanya, fungsi pass untuk mengkonstruksi sebuah loop.