

MODUL 5

Tipe Data



CAPAIAN PEMBELAJARAN

Mampu menggunakan tipe data List, Tuple, Set dan Dictionary dalam python untuk membuat program



KEBUTUHAN ALAT/BAHAN/SOFTWARE

1. Python
2. VS Code



DASAR TEORI

Python menyediakan sejumlah tipe data yang berupa kumpulan data baik teratur (ordered) maupun tidak teratur (unordered). Tipe yang teratur adalah List dan Tuple, sedangkan yang tidak teratur adalah Set dan Dictionary.

Kumpulan Data Berurut:

List

List adalah tipe data yang berisi satu atau beberapa nilai di dalamnya. Nilai – nilai ini sering juga disebut item, elemen, atau anggota list. List dibuat dengan menempatkan semua item di dalam tanda kurung [], dipisahkan oleh tanda koma. Anggota list bisa berisi satu tipe data, atau campuran. Index posisi data di dalam list dimulai dari 0

Index List:

0	1	2	3	4	5	6	7	8	9	10	11
S	T	M	I	K		A	K	A	K	O	M
-12	-11	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1

```
# list kosong
my_list = []

# list berisi integer
my_list = [1,2,3,4,5]

# list berisi tipe campuran
my_list = [1, 3.5, "Hello"]
# baca isi list ke 1
print(my_list[1])          #output 3.5
#slice list, bisa dari kiri atau kanan (menggunakan bil negatif)
my_list = ['a','b','c','d','e']
print(my_list[:2])         # output ['a', 'b']
print(my_list[2:4])        # output ['c', 'd']
print(my_list[-3])         # output c
print(my_list[-3:])        # output ['c', 'd', 'e']
print(my_list[:-3])        # output ['a', 'b']
```

List juga bisa berisi list lain. Ini disebut list bersarang

```
# list bersarang
my_list = ["hello", [2,4,6], ['a','b']]
# ambil data '6'
Print(my_list[1][2])      #output 6
```

Tuple

Tuple mirip dengan list. Bedanya, tuple bersifat immutable, sehingga anggotanya tidak bisa diubah.

Tuple dibuat dengan meletakkan semua anggota di dalam tanda kurung (), masing-masing dipisahkan oleh tanda koma. Menggunakan tanda kurung sebenarnya hanya opsional, tapi kita sebaiknya tetap menggunakannya untuk kemudahan pembacaan kode. Tuple dapat berisi tipe data yang sama maupun campuran.

```
# tuple bersarang
my_tuple = ("hello", [1, 2, 3], (4, 5, 6))
print(my_tuple) # Output: ("hello", [1, 2, 3], (4, 5, 6))
# Tuple bisa tidak menggunakan tanda ()
# Output (1, 2, 3)
my_tuple = 1, 2, 3
```

Kumpulan Data Tidak Berurut

Set

Set adalah salah satu tipe data di Python yang tidak berurut (unordered). Set memiliki anggota yang unik (tidak ada duplikasi). Jadi misalnya kalau kita meletakkan dua anggota yang sama di dalam set, maka otomatis set akan menghilangkan yang salah satunya.

Set bisa digunakan untuk melakukan operasi himpunan matematika seperti gabungan, irisan, selisih, dan lain – lain. Set bisa berisi data campuran, baik integer, float, string, dan lain sebagainya. Akan tetapi set tidak bisa berisi list, set, dan dictionary.

Set dibuat dengan meletakkan anggota – anggotanya di dalam tanda kurung kurawal { }, dipisahkan menggunakan tanda koma. Kita juga bisa membuat set dari list dengan memasukkan list ke dalam fungsi set()

```
# set data campuran
my_set = {1, 2.0, "Python", (3,4,5)}
print(my_set) # output {1, 2.0, "Python", (3,4,5)}

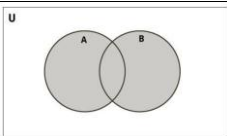
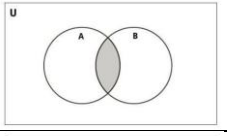
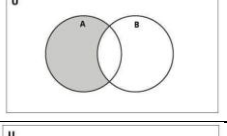
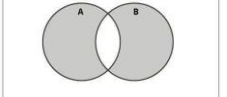
#memasukkan List ke dalam Set
my_set = set([1,2,3])
print(my_set)      # output {1,2,3}

# bila kita mengisi duplikasi, set akan menghilangkan salah satu
my_set = {1,2,2,3,3,3}
print(my_set)      # output: {1,2,3}
#untuk membuat set kosong
a = set()
```

Operasi Set di python:

Misal terdapat data sbb:

```
A = {1, 2, 3, 4, 5}
B = {4, 5, 6, 7, 8}
```

Operasi	Diagram	Syntax	output
Union		Print (A B) A.union(B) B.union(A)	{1, 2, 3, 4, 5, 6, 7, 8}
Irisan		Print (A & B) A.intersection(B) B.intersection(A)	{4,5}
Selisih (Difference)		Print (A - B) A.difference(B) Print (B - A) B.difference(A)	{1,2,3} {8,6,7}
Komplemen		print(A ^ B) A.symmetric_difference(B) B.symmetric_difference(A)	{1, 2, 3, 6, 7, 8}

Dictionary

Dictionary adalah tipe data yang anggotanya terdiri dari pasangan `kunci:nilai` (`key:value`) . Dictionary bersifat tidak berurut (`unordered`) sehingga anggotanya tidak memiliki indeks. Dictionary dibuat dengan menempatkan anggotanya di dalam tanda kurung kurawal `{ }`, dipisahkan oleh tanda koma.

Anggota dictionary terdiri dari pasangan `kunci:nilai` . Kunci harus bersifat unik, tidak boleh ada dua kunci yang sama dalam dictionary.

```
# Membuat dictionary kosong
my_dict = {}

# dictionary dengan kunci integer
my_dict = {1: 'sepatu', 2: 'tas'}

# dictionary dengan kunci campuran
my_dict = {'warna': 'merah', 1: [2,3,5]}

# membuat dictionary menggunakan fungsi dict()
my_dict = dict([(1:'sepatu'), (2:'bola')])
```

Dictionary tidak menggunakan indeks. Anggota dictionary diakses dengan menggunakan kuncinya. Selain itu, bisa juga diakses dengan menggunakan fungsi `get()`.

Dengan menggunakan fungsi `get()`, bila kunci tidak ada di dalam dictionary, maka akan dikembalikan `None`. Bila tidak menggunakan fungsi `get()`, maka akan terjadi error `KeyError` bila kunci yang hendak diakses tidak ada di dalam dictionary.

```
print(my_dict["warna"])
print(my_dict.get("warna"))
```

Dictionary bersifat mutable. Kita bisa menambahkan atau mengubah nilai dari anggotanya menggunakan operator penugasan. Bila kunci sudah ada, maka nilainya yang akan diupdate. Bila kunci belum ada, maka akan ditambahkan sebagai kunci baru.



PRAKTIK

1. Buka VS Code

2. Buat file baru di dalam direktori c:\python dengan nama list1.py

3. Ketikkan script berikut ini:

```
1  #Deklarasi List
2  list_nilai = [1,2,3,4,5,6,7,8,9,10]
3  #tampilkan list
4  print("List Nilai : \n",list_nilai)
5  #menggunakan len
6  print("\nBanyak Item list nilai (len) : ", len(list_nilai))
7  #Menggunakan append
8  print("\nMenambah element 12 dan 5 dalam list (append) : ")
9  list_nilai.append(12)
10 list_nilai.append(5)
11 print("List Nilai : \n",list_nilai)
12 #menggunakan count
13 print("\nBanyak Nilai 5 Dalam List (count) : \n ", list_nilai.count(5))
14 #Menggunakan index
15 print("\nAngka 8 ada di posisi index ke : \n", list_nilai.index(8))
16 #Menggunakan insert
17 list_nilai.insert(1,12)
18 print("\nTambah Nilai List (Insert) : \n", list_nilai)
19 #Menggunakan pop
20 list_nilai.pop()
21 list_nilai.pop()
22 print("\nList Nilai (pop) : \n",list_nilai)
23 #Menggunakan remove
24 list_nilai.remove(2)
25 list_nilai.remove(1)
26 print("\nList Nilai (remove) : \n", list_nilai)
27 #Menggunakan extend
28 list_nilai2 = [4,3,8,7]
29 list_nilai.extend(list_nilai2)
30 print("\nList Nilai (extend) : \n", list_nilai)
31 #Menggunakan reverse
32 list_nilai.reverse()
33 print("\nList Nilai (reverse) : \n", list_nilai)
34 #Menggunakan sort
35 list_nilai.sort()
36 print("\nList Nilai (sort) : \n", list_nilai)
37 #Menggunakan Max
38 print("\nNilai Terbanyak (max) : \n", max(list_nilai))
39 #Menggunakan min
40 print("\nNilai terendah (min) : \n", min(list_nilai))
41 #Menggunakan Sum
42 print("\nTotal penjumlahan nilai dari list_nilai (sum) : \n", sum(list_nilai))
```

4. Simpan script tersebut

5. Kembali ke command prompt, masuk ke c:\python

6. Untuk menjalankan ketikkan : py list1.py atau python list1.py, amati hasilnya!

Jelaskan fungsi tiap metode yang dioperasikan pada list tersebut (len, sort, dll)! Apa perbedaan Pop dan Remove ?

7. Buat program baru bernama dict.py, kemudian ketikkan script berikut ini:

```

1 my_dict = {'nama':'Budi', 'usia':27}
2
3
4 print(my_dict['nama'])
5 print(my_dict.get('usia'))
6 print(my_dict['alamat'] )

```

Apa yang terjadi saat script tersebut dijalankan? Mengapa?

8. Modifikasilah program dict.py sebagai berikut (baris 2 dan 3) :

```

1 my_dict = {'nama':'Budi', 'usia':27}
2 my_dict['alamat'] = "jogja"
3 my_dict['usia'] = 40
4 print(my_dict['nama'])
5 print(my_dict.get('usia'))
6 print(my_dict['alamat'] )

```

Apa yang terjadi saat script tersebut dijalankan? Apa fungsi statemen di baris 2 dan 3?

9. Buat program baru dengan nama olehdata.py, kemudian ketikkan script berikut ini:
(script cukup panjang sehingga anda harus teliti dalam menuliskannya)

```

1 # import lib operating system untuk fungsi bersihkan layar
2 import os
3 #buat variabel global
4 data = []
5 ke = 0
6
7 def tambahdata():
8     bersihkanlayar()
9     global ke #gunakan statemen global agar var ke ter-assign
10    jawab = 'y'
11    while(jawab.lower() == 'y'):
12        ke+=1
13        print("Data Ke-"+str(ke))
14        nama = input("Nama : ")
15        nilai = int(input("Nilai : "))
16        #memasukkan nama dan nilai ke dalam list dengan bentuk dictionary
17        data.append({"nama":nama,"nilai":nilai})
18        #meminta masukan lanjut atau tidak
19        jawab = input("Isi Lagi (y/t) ? : ")

```

```

20
21 def tampildata():
22     bersihkanlayar()
23     no = 0
24     for dt in data:
25         no+=1
26         print(no,"%s \t %d" %(dt['nama'], dt['nilai']))
27     tunggu()
28
29 def hapusdata():
30     global data
31     namadihapus = input("Isikan Nama Yang Akan Dihapus : ")
32     #metode khusus untuk menghapus data dictionary dalam list
33     data = [i for i in data if not (i['nama'] == namadihapus)]
34     print("Data "+namadihapus+" Sudah dihapus...")
35     tunggu()
36
37 def bersihkanlayar():
38     os.system("cls")
39
40 def tunggu():
41     a = input("Tekan ENTER untuk lanjut...")
42
43
44 #program utama
45 pilih = 1
46 while(pilih != 0):
47     bersihkanlayar()
48     print("==MENU==")
49     print("1. Tambah Data")
50     print("2. Lihat Data")
51     print("3. Hapus Data")
52     print("0. Keluar")
53     print
54     pilih = int(input("Pilihan Anda : "))
55     if (pilih == 1):
56         tambahdata()
57     elif (pilih == 2):
58         tampildata()
59     elif (pilih == 3):
60         hapusdata()
61     else :
62         print("selesai")
63

```

10. Jalankan program tersebut dan pelajari cara kerjanya! Tambahkan tampilan nilai rata-rata dari semua data yang dimasukkan pada fungsi tampildata()



LATIHAN



TUGAS

Buatlah program sesuai kasus berikut ini :

1. Diketahui suatu list :

```
kps = ['s', 't', 'm', 'i', 'k', ' ', 'a', 'k', 'a', 'k', 'o', 'm']
```

bagaimana cara menampilkan list yang hanya berisi :

- ['s', 't', 'm', 'i', 'k']
- ['a', 'k', 'a', 'k', 'o', 'm']
- ['m', 'i', 'k']

2. Diketahui data mahasiswa anggota UKM berbasis set sbb :

```
kesenian = {'adi', 'siti', 'rudi', 'doni'}
```

```
mapala = {'rudi', 'sigit', 'doni', 'andi'}
```

Gunakan operator Set untuk menampilkan :

- Anggota kesenian yang tidak ikut mapala
- Mahasiswa yang ikut di kedua UKM tsb
- Semua data mahasiswa anggota UKM
- Anggota Mapala yang tidak ikut kesenian
- Mahasiswa yang hanya ikut 1 UKM saja



REFERENSI
