



UNIVERSITAS
GADJAH MADA

W6 PEMROGRAMAN LANJUT

Oleh :

Alim Satria Fi'i Wijaya Kusuma (21/483503/SV/20304)

Video



UNIVERSITAS GADJAH MADA

1. TUPLE

```
iniTuple = {1.5,"alim",10,True}  
iniList = [1.5,"alim",10,True]
```

```
print(type(iniTuple))  
print(type(iniList))
```

✓ 0.7s

```
<class 'set'  
<class 'list'
```

```
tuple_1 = 1.5,"alim",10,True  
print(type(tuple_1))
```

✓ 0.3s

```
<class 'tuple'
```

```
list_1 = [50]  
print(type(list_1))
```

✓ 0.2s

```
<class 'list'
```

```
tuple_2 = (50)  
tuple_3 = 50  
tuple_4 = (50,)
```

```
print(type(tuple_2))  
print(type(tuple_3))  
print(type(tuple_4))
```

✓ 0.4s

```
<class 'int'  
<class 'int'  
<class 'tuple'
```

```
tuple_5 = 1,2,3,4,5,6,7,8,9,10  
print(tuple_5[4])  
print(tuple_5[5:8])
```

✓ 0.4s

```
5  
(6, 7, 8)
```

```
tuple_5 = 1,2,3,4,5,6,7,8,9,10  
tuple_6 = 10,20,30
```

```
tuple_7 = tuple_5+tuple_6  
print(tuple_7)
```

✓ 0.6s

```
(1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 10, 20, 30)
```

```
tuple_6 = 10,20,30  
tuple_8 = tuple_6*3  
print(tuple_8)
```

✓ 0.4s

```
(10, 20, 30, 10, 20, 30, 10, 20, 30)
```

Video



UNIVERSITAS GADJAH MADA

2. DICTIONARY

```
kamus_eng = {'apel': 'apple', 'anggur': 'grape', 'jeruk': 'orange', 'belimbing': 'starfruit'}  
print(type(kamus_eng))  
print(kamus_eng['apel'])
```

✓ 0.4s

```
<class 'dict'>  
apple
```

```
nilai_Matek = {'alim': 9.0, 'satria': 6.5}  
print(nilai_Matek['alim'])
```

✓ 0.4s

```
9.0
```

```
nilai_Matek['andi'] = 9.8  
print(nilai_Matek)
```

✓ 0.4s

```
{'alim': 9.0, 'satria': 6.5, 'andi': 9.8}
```

```
nilai_Matek['satria'] = 90  
print(nilai_Matek)
```

✓ 0.4s

```
{'alim': 9.0, 'satria': 90, 'andi': 9.8}
```

```
del nilai_Matek['satria']  
print(nilai_Matek)
```

✓ 0.5s

```
{'alim': 9.0, 'andi': 9.8}
```

```
data_pasien = {'nama': 'alim', 'umur': 22, 'gender': True, 'BB': 60}  
print(data_pasien['gender'])
```

✓ 0.3s

```
True
```

Video



```
data_pasien = {'nama':'alim','umur':22,'gender':True,'BB':60}
item_data = []

for i in data_pasien.keys() :
    item_data.append(i)
print(item_data)
✓ 0.3s

['nama', 'umur', 'gender', 'BB']

data_alim = []

for i in data_pasien.values() :
    data_alim.append(i)
print(data_alim)
✓ 0.3s

['alim', 22, True, 60]
```

```
kamus_eng = {'apel':'apple','anggur':'grape','jeruk':'orange','belimbing':'starfruit'}

for i,j in kamus_eng.items() :
    print(i, "Bahasa inggrisnya adalah ",j)
✓ 0.4s

apel Bahasa inggrisnya adalah  apple
anggur Bahasa inggrisnya adalah  grape
jeruk Bahasa inggrisnya adalah  orange
belimbing Bahasa inggrisnya adalah  starfruit
```


Modul : Tuple



UNIVERSITAS GADJAH MADA

```
my_tuple = {1,2,'a string',(3,4),(5,6),None}
print(my_tuple)
```

```
my_list = [1,2,True,'a string',(3,4),(5,6),None]
print(my_list)
```

✓ 0.3s

```
{1, 2, None, (3, 4), 'a string', (5, 6)}
```

```
[1, 2, True, 'a string', (3, 4), (5, 6), None]
```

```
empty_tuple = ()
print(type(empty_tuple))
```

✓ 0.3s

```
<class 'tuple'>
```

```
tup_1 = (50,)
tup_2 = (50)
```

```
print(type(tup_1))
print(type(tup_2))
```

✓ 0.5s

```
<class 'tuple'>
```

```
<class 'int'>
```

```
tuple_1 = (1,2,4,8)
tuple_2 = 1., .5, .25, .125
```

```
print(tuple_1)
print(tuple_2)
```

✓ 0.5s

```
(1, 2, 4, 8)
```

```
(1.0, 0.5, 0.25, 0.125)
```

Tuple, merupakan sebuah list yang elemen-elemen di dalamnya tidak dapat diubah urutannya (immutable). Tupel menggunakan tanda kurung (dapat juga tidak menggunakan tanda kurung) sedangkan list menggunakan tanda kurung siku.



Akses nilai pada tuple

AKSES NILAI PADA TUPEL

```
tup_1 = ('fisika','kimia',1993,2017)
tup_2 = (1,2,3,4,5,6,7)
print('tup_1[0] ',tup_1[0])
print('tup_2[1:5] ',tup_2[1:5])
```

✓ 0.3s

```
tup_1[0]  fisika
tup_2[1:5]  (2, 3, 4, 5)
```

nilai tuple dapat diakses dengan memanggil tuple dengan indeks elemen didalamnya, didalam program biasa ditulis dengan,

`tuple[index_elemen]`

Update nilai tuple



UPDATE NILAI

```
tup_1 = (12, 34.16)
tup_2 = ('abc', 'xyz')

tup_3 = tup_1 + tup_2
print(tup_3)
```

✓ 0.3s

```
(12, 34.16, 'abc', 'xyz')
```

Penambahan elemen pada tuple dapat dilakukan dengan menambahkan secara langsung dua tuple kedalam tuple yang baru,

```
tuple_3 = tuple_1 + tuple_2
```

Penghapusan nilai tuple



```
tup = {'fisika','kimia',1993,2017}
print(tup)
del tup
print('setelah tuple dihapus ',tup)
```

0.8s Python

```
{1993, 'fisika', 'kimia', 2017}
```

```
NameError                                Traceback (most recent call last)
/home/alimsatria/Documents/MATERI KULIAH/PEMROGRAMAN LANJUT/WE
EK_6/WEEK_6_Modul.ipynb Cell 8' in <module>
      2 print(tup)
      3 del tup
----> 4 print('setelah tuple dihapus ',tup)

NameError: name 'tup' is not defined
```

Tuple dapat dihapus dengan menggunakan perintah `del` yang diimplementasikan sebagai berikut,

```
del tuple
```


Dictionary



UNIVERSITAS GADJAH MADA

DICTIONARY

```
dict_1 = {'cat':'chat','dog':'chien','horse':'cheval'}  
phone_num = {'boss' : 5551234567, 'suzy' : 22657854310}  
empty_dict = {}
```

```
print(dict_1)  
print(phone_num)  
print(empty_dict)
```

✓ 0.4s

```
{'cat': 'chat', 'dog': 'chien', 'horse': 'cheval'}  
{'boss': 5551234567, 'suzy': 22657854310}  
{}
```

Dictionary, merupakan penumpukan data dengan format data yang memiliki key dan value yang dipisahkan dengan tanda titik dua, penulisan dictionary sebagai berikut,

```
dictionary = {key:value}
```



AKSES NILAI PADA DICTIONARY

```
dict_1 = {'cat':'chat','dog':'chien','horse':'cheval'}  
phone_num = {'boss' : 5551234567, 'suzy' : 22657854310}
```

```
print(dict_1['cat'])  
print(phone_num['suzy'])
```

✓ 0.4s

```
chat  
22657854310
```

akses nilai pada dictionary dapat dilakukan dengan memanggil dictionary dan key dari data yang ingin ditarik,

`dictionary['key']`



UPDATE NILAI PADA DICTIONARY

```
dict_1 = {'cat': 'chat', 'dog': 'chien', 'horse': 'cheval'}
```

```
dict_1['cat'] = 'minou'
```

```
print(dict_1)
```

```
dict_1['swan'] = 'cygne'
```

```
print(dict_1)
```

✓ 0.3s

```
{'cat': 'minou', 'dog': 'chien', 'horse': 'cheval'}
```

```
{'cat': 'minou', 'dog': 'chien', 'horse': 'cheval', 'swan': 'cygne'}
```

menambahkan atau merubah nilai pada dictionary dapat dilakukan dengan menambahkan elemen pada key lama (untuk merubah value) atau menggunakan key baru (untuk menambahkan elemen pada dictionary)



Penghapusan nilai item dalam dictionary

```
data_siswa = {'name':'zara','age':7,'class':'first'}
del data_siswa['name']
print(data_siswa)

del data_siswa
print(data_siswa)
```

✖ 0.6s

```
{'age': 7, 'class': 'first'}
```

NameError Traceback (most recent call last)

/home/alimsatria/Documents/MATERI KULIAH/PEMROGRAMAN LANJUT/WEEK_6/WEEK_6_Modul.ipynb Cell 12' in <module>

3 print(data_siswa)

5 del data_siswa

----> 6 print(data_siswa)

NameError: name 'data_siswa' is not defined

```
data_siswa = {'name':'zara','age':7,'class':'first'}
data_siswa.clear()
print(data_siswa)
```

✓ 0.3s

```
{}
```

Terdapat dua cara untuk menghapus nilai pada dictionary, yang pertama dapat menggunakan del. yang kedua dapat menggunakan clear

penggunaan del dapat menyebabkan seluruh dictionary hilang, namun ketika menggunakan clear maka hanya element di dalam dictionary yang hilang.



KEYS, VALUES, DAN ITEMS

```
dict_1 = {'cat': 'chat', 'dog': 'chien', 'horse': 'cheval'}
```

```
for key in dict_1.keys() :  
    print(key, '-->', dict_1[key])
```

```
print('')
```

```
for key in sorted(dict_1.keys()) :  
    print(key, '-->', dict_1[key])
```

✓ 0.8s

```
cat --> chat  
dog --> chien  
horse --> cheval
```

```
cat --> chat  
dog --> chien  
horse --> cheval
```

```
dict_1 = {'cat': 'chat', 'dog': 'chien', 'horse': 'cheval'}
```

```
for french in dict_1.values() :  
    print(french)
```

✓ 0.3s

```
chat  
chien  
cheval
```

```
dict_1 = {'cat': 'chat', 'dog': 'chien', 'horse': 'cheval'}
```

```
for english, french in dict_1.items() :  
    print(english, '-->', french)
```

✓ 0.5s

```
cat --> chat  
dog --> chien  
horse --> cheval
```


Pemanggilan dictionary dapat dilakukan dengan tiga cara, yaitu dengan memanggil keys, memanggil value, dan memanggil item.

- pemanggilan berdasarkan keys akan memanggil key dari dictionary
- pemanggilan berdasarkan values akan memanggil values dari dictionary
- pemanggilan items akan memanggil keys dan values

Tugas



```
tuple_sayur = 'bayam','terong','kubis','labu','brokoli'

sayur = {
    'bayam':1500,
    'terong':2000,
    'kubis':3000,
    'labu':4000,
    'brokoli':4500
}

pembelian = {}

while(1) :

    kode_sayur = int(input("Masukkan kode sayur"))
    jumlah_sayur = int(input("Masukkan jumlah sayur"))

    total_harga_sayur = sayur[tuple_sayur[kode_sayur - 1]] * jumlah_sayur

    pembelian[tuple_sayur[kode_sayur - 1]] = total_harga_sayur

    total_belanja = total_belanja + total_harga_sayur

    loop_flag = input('ada item lain yang ingin dibeli? (y/n)')

    if(loop_flag == 'n') :
        break
```

```
print('daftar belanja : ', pembelian)
print('\ntotal belanja : Rp.', total_belanja)

x = 0

if total_belanja > 20000 and total_belanja < 50000 :
    x = 10
    total_belanja = total_belanja - (total_belanja * 0.1)
elif(total_belanja > 50000 and total_belanja < 100000) :
    x = 15
    total_belanja = total_belanja - (total_belanja * 0.15)
elif(total_belanja > 100000) :
    x = 20
    total_belanja = total_belanja - (total_belanja * 0.2)
else :
    x = 0
    total_belanja = total_belanja

print('\nanda mendapat diskon sebesar ',x,'persen')
print('total belanja menjadi ',total_belanja)

total_harga_sayur = 0 #reset variabel
total_belanja = 0
pembelian.clear()
```

Penjelasan kode tugas



```
tuple_sayur = 'bayam','terong','kubis','labu','brokoli'

sayur = {
    'bayam':1500,
    'terong':2000,
    'kubis':3000,
    'labu':4000,
    'brokoli':4500
}

pembelian = {}
```

- data berupa kode sayur dapat dideklarasikan dalam tuple. Data dalam `tuple_sayur` telah diurutkan sesuai dengan kode sayur.
- data berupa nama sayur dan harga masing-masing sayur ditampung kedalam dictionary `sayur` agar dapat sesuai dengan ketentuan tugas.
- dibuat dictionary kosong `pembelian` untuk menampung sayur apa saja yang nanti akan dibeli dan total harga dari jumlah sayur yang dibeli.

Penjelasan kode tugas



```
while(1) :  
  
    kode_sayur = int(input("Masukkan kode sayur"))  
    jumlah_sayur = int(input("Masukkan jumlah sayur"))  
  
    total_harga_sayur = sayur[tuple_sayur[kode_sayur - 1]] * jumlah_sayur  
  
    pembelian[tuple_sayur[kode_sayur - 1]] = total_harga_sayur  
  
    total_belanja = total_belanja + total_harga_sayur  
  
    loop_flag = input('ada item lain yang ingin dibeli? (y/n)')  
  
    if(loop_flag == 'n') :  
        break
```

- kode dimasukkan ke loop menggunakan while agar dapat menampung order sayur selanjutnya. loop while akan berhenti apabila user memasukan karakter n ketika ditanya apakah ada item lain yang akan dibeli
- syntax input berfungsi agar user dapat memasukkan kode sayur dan jumlah sayur yang hendak dibeli
- variabel total_harga_sayur digunakan untuk menghitung total harga dari setiap jenis sayur dikalikan dengan jumlah sayur yang dibeli per satu jenis sayur
- lalu seluruh pembelian sayur ditotal dan ditampung pada variabel total_belanja

Penjelasan kode tugas



```
print('daftar belanja : ', pembelian)
print('\ntotal belanja : Rp.', total_belanja)

x = 0

if total_belanja > 20000 and total_belanja < 50000 :
    x = 10
    total_belanja = total_belanja - (total_belanja * 0.1)
elif(total_belanja > 50000 and total_belanja < 100000) :
    x = 15
    total_belanja = total_belanja - (total_belanja * 0.15)
elif(total_belanja > 100000) :
    x = 20
    total_belanja = total_belanja - (total_belanja * 0.2)
else :
    x = 0
    total_belanja = total_belanja

print('\nanda mendapat diskon sebesar ',x,'persen')
print('total belanja menjadi ',total_belanja)

total_harga_sayur = 0 #reset variabel
total_belanja = 0
pembelian.clear()
```

Dari hasil perhitungan total_belanja dapat ditentukan diskon yang diperoleh user, kemudian program menampilkan total belanja sebelum diskon, daftar item yang dibeli dan total harga per item dan total_belanja setelah didiskon.

Tugas



UNIVERSITAS GADJAH MADA

```
daftar belanja : {'bayam': 7500, 'kubis': 9000, 'brokoli':  
31500}
```

```
total belanja : Rp. 48000
```

```
anda mendapat diskon sebesar 10 persen  
total belanja menjadi 43200.0
```

```
daftar belanja : {'terong': 16000, 'labu': 20000, 'bayam':  
4500}
```

```
total belanja : Rp. 40500
```

```
anda mendapat diskon sebesar 10 persen  
total belanja menjadi 36450.0
```

```
daftar belanja : {'brokoli': 36000, 'labu': 40000, 'kubis':  
45000, 'terong': 8000}
```

```
total belanja : Rp. 129000
```

```
anda mendapat diskon sebesar 20 persen  
total belanja menjadi 103200.0
```

```
daftar belanja : {'bayam': 7500, 'terong': 14000, 'kubis':  
18000, 'labu': 16000, 'brokoli': 22500}
```

```
total belanja : Rp. 78000
```

```
anda mendapat diskon sebesar 15 persen  
total belanja menjadi 66300.0
```

Perhitungan



```
bayam = 1500
terong = 2000
kubis = 3000
labu = 4000
brokoli = 4500

total_1 = 5*bayam + 3*kubis + 7*brokoli
total_2 = 8*terong + 5*labu + 3*bayam
total_3 = 8*brokoli + 10*labu + 15*kubis + 4*terong
total_4 = 5*bayam + 7*terong + 6*kubis + 4*labu + 5*brokoli

print('perhitungan satu : ', total_1-(0.1*total_1))
print('perhitungan dua : ', total_2-(0.1*total_2))
print('perhitungan tiga : ', total_3-(0.2*total_3))
print('perhitungan empat : ', total_4-(0.15*total_4))
```

✓ 0.6s

Python

```
perhitungan satu : 43200.0
perhitungan dua : 36450.0
perhitungan tiga : 103200.0
perhitungan empat : 66300.0
```