



22A0111053 : Pemrograman

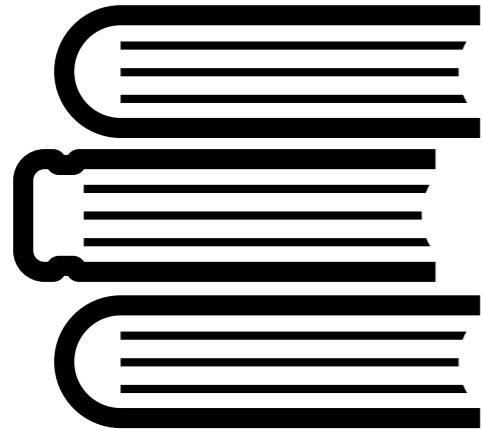
Pertemuan 6-7 : Lists dan Tuples

Zaitun, S.Si., M.Mat.



LIST DAN TUPLES

- **List**
 - Tentang list
 - Function pada list
 - Method pada list
 - Pengirisan list (slicing)
- **Tuples**
 - Tentang tuple
 - Function pada tuple
 - Method pada tuple
 - Pengirisan tuple (slicing)
- **Nested List**
- **Latihan**



LIST

```
my_list = ['Frankel', 2023, True]  
print(my_list)
```



TENTANG LIST

List adalah salah satu jenis tipe data pada komputer yang merupakan struktur data yang terdiri atas data berurut (tipe data urutan). List dituliskan dengan kurung siku '[]' (**square brackets**) yang di dalamnya dapat berupa tipe data apa saja. Karena merupakan tipe data, list juga dapat ditugaskan dalam variabel.

```
ex_list = ['String', 2023, True]  
print(ex_list)
```

```
>> ['String', 2023, True]
```



TAMBAHAN

jenis tanda kurung, yaitu:

1. tanda kurung/kurung lengkung (*round brackets*): ()
2. tanda kurung siku/kurung tegak (*square brackets*): []
3. tanda kurung kurawal (*curly brackets*): { }
4. tanda kurung sudut (*angle brackets*): <>



FUNCTION PADA LIST

Penulisan function pada list sama sekali tidak berbeda yaitu dengan menuliskan nama *function* kemudian memasukkan input di dalamnya. Berikut contoh function pada list.

Function	Penulisan	Keterangan
len	len(list)	Jumlah item dalam list
max	max(list)	Nilai terbesar pada list (hanya untuk tipe data numerik)
min	min(list)	Nilai terkecil pada list (hanya untuk tipe data numerik)
sum	sum(list)	Jumlah total data list (hanya untuk tipe data numerik)



CONTOH FUNCTION

```
# function len  
int_list = [2,5,6,2]  
a = len(int_list)  
print(a)
```

```
>> 4
```

```
# function max  
int_list = [2,5,6,2]  
a = max(int_list)  
print(a)
```

```
>> 6
```



CONTOH FUNCTION 2

```
# function min  
int_list = [2,5,6,2]  
a = min(int_list)  
print(a)
```

```
>> 2
```

```
# function sum  
int_list = [2,5,6,2]  
a = sum(int_list)  
print(a)
```

```
>> 15
```



METHOD PADA LIST

Method pada list ditulis dengan cara memberikan nama list kemudian kode method yang diinginkan.

Listed.method(properti)

Method	Penulisan	Keterangan
count	listed.count(item)	Menghitung kemunculan data
index	listed.index(item)	Mendapatkan posisi index dari data
sort	listed.sort()	Mengurutkan data sesuai alphabet dan angka
reverse	listed.reverse()	Membalik urutan data
append	listed.append(item)	Menambahkan item di urutan terakhir
extend	listed.extend(list)	Menambahkan list baru di urutan terakhir
insert	listed.insert(pos, item)	Menyisipkan item baru di posisi ditentukan
remove	listed.remove(item)	Menghapus kemunculan pertama item
clear	listed.clear()	Menghapus seluruh data
Pop	listed.pop(int)	Menghapus data pada posisi yang ditentukan



CONTOH METHOD COUNT

```
listed = ['pema', 'ursa', 'ulon', 'pema', 'ozai']
a = listed.count('pema')
print(a)
```

```
>> 2
```

```
listed = [2,3,6,4,1,2,4,3,2,1,4]
a = listed.count(4)
print(a)
```

```
>> 3
```



CONTOH METHOD INDEX

```
listed = ['pema', 'ursa', 'ulon', 'pema', 'ozai']
a = listed.index('ursa')
print(a)
```

```
>> 1
```

```
listed = [2,3,6,4,1,2,4,3,2,1,4]
a = listed.index(1)
print(a)
```

```
>> 4
```



CONTOH METHOD SORT

```
listed = ['pema', 'ursa', 'ulon', 'pema', 'ozai']
listed.sort()
print(listed)
```

```
>> ['ozai', 'pema', 'pema', 'ulon', 'ursa']
```

```
listed = ['pema', 'ursa', 'ulon', 'pema', 'ozai']
listed.sort(reverse=True)
print(listed)
```

```
>> ['ursa', 'ulon', 'pema', 'pema', 'ozai']
```



CONTOH METHOD SORT

```
listed = [2,3,6,4,1,2,4,3,2,1,4]
listed.sort()
print(listed)
```

```
>> [1, 1, 2, 2, 2, 3, 3, 4, 4, 4, 6]
```

```
listed = [2,3,6,4,1,2,4,3,2,1,4]
listed.sort(reverse=True)
print(listed)
```

```
>> [6, 4, 4, 4, 3, 3, 2, 2, 2, 1, 1]
```



CONTOH METHOD REVERSE

```
listed = ['pema', 'ursa', 'ulon', 'pema', 'ozai']
listed.reverse()
print(listed)
```

```
>> ['ozai', 'pema', 'ulon', 'ursa', 'pema']
```

```
listed = [2,3,6,4,1,2,4,3,2,1,4]
listed.reverse()
print(listed)
```

```
>> [4, 1, 2, 3, 4, 2, 1, 4, 6, 3, 2]
```



CONTOH METHOD APPEND

```
listed = ['pema', 'ursa', 'ulon', 'pema', 'ozai']
listed.append('azula')
print(listed)
```

```
>> ['pema', 'ursa', 'ulon', 'pema', 'ozai', 'azula']
```

```
listed = [2,3,6,4,1,2,4,3,2,1,4]
listed.append(19)
print(listed)
```

```
>> [2, 3, 6, 4, 1, 2, 4, 3, 2, 1, 4, 19]
```



CONTOH METHOD EXTEND

```
listed = ['pema', 'ursa', 'ulon', 'azula']
name = ['ila', 'mai', 'tamin']
listed.extend(name)
print(listed)
```

```
>> ['pema', 'ursa', 'ulon', 'azula', 'ila', 'mai', 'tamin']
```

```
listed = ['pema', 'ursa', 'ulon', 'azula']
number = [1,2,3,4]
listed.extend(number)
print(listed)
```

```
>> ['pema', 'ursa', 'ulon', 'azula', 1,2,3,4]
```



CONTOH METHOD INSERT

```
listed = ['pema', 'ursa', 'ulon', 'azula']
listed.insert(1, 'ila')
print(listed)
```

```
>>['pema', 'ila', 'ursa', 'ulon', 'azula']
```

```
listed = [1,2,3,4]
listed.insert(3,5)
print(listed)
```

```
>> [1,2,3,5,4]
```



CONTOH METHOD REMOVE

```
listed = ['pema', 'ila', 'ursa', 'pema', 'azula']
listed.remove('ursa')
print(listed)
```

```
>> ['pema', 'ila', 'pema', 'azula']
```

```
listed = ['pema', 'ila', 'ursa', 'pema', 'azula']
listed.remove('pema')
print(listed)
```

```
>> ['ila', 'ursa', 'pema', 'azula']
```



CONTOH METHOD CLEAR

```
listed = ['pema', 'ila', 'ursa', 'pema', 'azula']
listed.clear()
print(listed)
```

```
>> []
```



CONTOH METHOD POP

```
listed = ['pema', 'ila', 'ursa', 'pema', 'azula']
listed.pop(1)
print(listed)
```

```
>> ['pema', 'ursa', 'pema', 'azula']
```

```
listed = ['pema', 'ila', 'ursa', 'pema', 'azula']
listed.pop()
print(listed)
```

```
>> ['pema', 'ila', 'ursa', 'pema']
```



TAMBAHAN METHOD

Terdapat operator pada list yang sedikit berbeda penggunaannya pada tipe data numerik.

```
list1 = [1,2,4]
list2 = [5,6]
listed = list1 + list2
print(listed)
```

```
>> [1,2,4,5,6]
```

```
list1 = ['pema','ila','ursa','azula']
List2 = [1,2,4]
listed = list1 + list2
print(listed)
```

```
>> ['pema','ila','ursa','azula',1,2,4]
```



TAMBAHAN METHOD 2

```
list1 = [1,2,4]*2  
  
print(list1)
```

```
>> [1,2,4,1,2,4]
```

```
list1 = ['pema','ila']*3  
print(list1)
```

```
>> ['pema','ila','pema','ila','pema','ila']
```

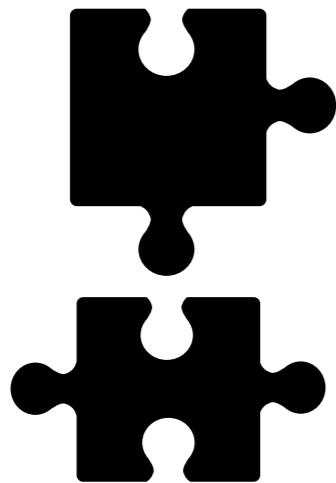


PENGIRISAN LIST (SLICING)

Pada pertemuan sebelumnya telah pernah dibahas pengirisan, yaitu pada string. Misal

$$a = [0, 1, 2, 3, 4, 5]$$

input	Output
<code>a[0]</code>	<code>[0]</code>
<code>a[5]</code>	<code>[5]</code>
<code>a[-1]</code>	<code>[5]</code>
<code>a[-2]</code>	<code>[4]</code>
<code>a[2:]</code>	<code>[2, 3, 4, 5]</code>
<code>a[:5]</code>	<code>[0, 1, 2, 3, 4]</code>
<code>a[-3:]</code>	<code>[3, 4, 5]</code>
<code>a[:-2]</code>	<code>[0, 1, 2, 3]</code>
<code>a[1:4]</code>	<code>[1, 2, 3]</code>
<code>a[1:-1]</code>	<code>[1, 2, 3, 4]</code>
<code>a[::-1]</code>	<code>[5, 4, 3, 2, 1, 0]</code>
<code>a[::-2]</code>	<code>[5, 3, 1]</code>



TUPLE

```
my_tuple = ('Frankel', 2023, True)  
print(my_tuple)
```



TENTANG TUPLE

- Tuple merupakan tipe data urutan
- Ditulis dengan kurung biasa '()' (**round brackets**)
- Dapat diisi dengan tipe data apa saja.
- Tidak dapat diubah isinya (ex: **insert and remove not Allowed**)
- Dapat dilakukan pengirisan (slicing)
- Dapat diakses isinya berdasarkan indeksnya.

```
tp = ('String', 2023, True)
```

```
print(tp)
```

```
>> ('String', 2023, True)
```



FUNCTION PADA TUPLE

Penulisan function pada tuple masih sama dengan list dengan menuliskan nama *function* kemudian memasukkan input di dalamnya. Berikut contoh function pada tuple.

Function	Penulisan	Keterangan
len	len (Tuple)	Jumlah item dalam tuple
max	max (Tuple)	Nilai terbesar pada tuple (hanya untuk tipe data numerik)
min	min (Tuple)	Nilai terkecil pada tuple (hanya untuk tipe data numerik)
sum	sum (Tuple)	Jumlah total data tuple (hanya untuk tipe data numerik)



CONTOH FUNCTION

```
# function len  
int_tup = (2,5,6,2)  
a = len(int_tup)  
print(a)
```

```
>> 4
```

```
# function max  
int_tup = (2,5,6,2)  
a = max(int_tup)  
print(a)
```

```
>> 6
```



CONTOH FUNCTION 2

```
# function min  
int_tup = (2,5,6,2)  
a = min(int_tup)  
print(a)
```

```
>> 2
```

```
# function sum  
int_tup = (2,5,6,2)  
a = sum(int_tup)  
print(a)
```

```
>> 15
```



METHOD PADA TUPLE

Method pada tuple identik dengan list. Ditulis dengan nama tuple kemudian dilanjutkan dengan nama *method*.

Tuple.method (properti)

Method	Penulisan	Keterangan
count	tuple.count(item)	Menghitung kemunculan data dalam tuple
index	tuple.index(item)	Mendapatkan posisi index dari data
sort	x	Mengurutkan data sesuai alphabet dan angka
reverse	x	Membalik urutan data
append	x	Menambahkan item di urutan terakhir
extend	x	Menambahkan data baru di urutan terakhir
insert	x	Menyisipkan item baru di posisi ditentukan
remove	x	Menghapus kemunculan pertama item
clear	x	Menghapus seluruh data
pop	x	Menghapus data pada posisi yang ditentukan



CONTOH METHOD COUNT

```
mytup = ('pema', 'ursa', 'ulon', 'pema', 'ozai')
a = mytup.count('pema')
print(a)
```

```
>> 2
```

```
mytup = (2, 3, 6, 4, 1, 2, 4, 3, 2, 1, 4)
a = mytup.count(4)
print(a)
```

```
>> 3
```



CONTOH METHOD INDEX

```
mytup = ('pema', 'ursa', 'ulon', 'pema', 'ozai')
a = mytup.index('ursa')
print(a)
```

```
>> 1
```

```
mytup = (2, 3, 6, 4, 1, 2, 4, 3, 2, 1, 4)
a = mytup.index(1)
print(a)
```

```
>> 4
```



TAMBAHAN

Terdapat operator pada tuple yang masih identik dengan list.

```
tup1 = (1,2,4)
tup2 = (5,6)
mytup = tup1 + tup2
print(mytup)
```

```
>> (1,2,4,5,6)
```

```
tup1 = ('pema','ila','ursa','azula')
tup2 = (1,2,4)
mytup = tup1 + tup2
print(mytup)
```

```
>> ('pema','ila','ursa','azula',1,2,4)
```



TAMBAHAN 2

```
tup1 = (1,2,4)*2  
print(tup1)
```

```
>> (1,2,4,1,2,4)
```

```
mytup = ('pema','ila')*3  
print(mytup)
```

```
>> ('pema','ila','pema','ila','pema','ila')
```



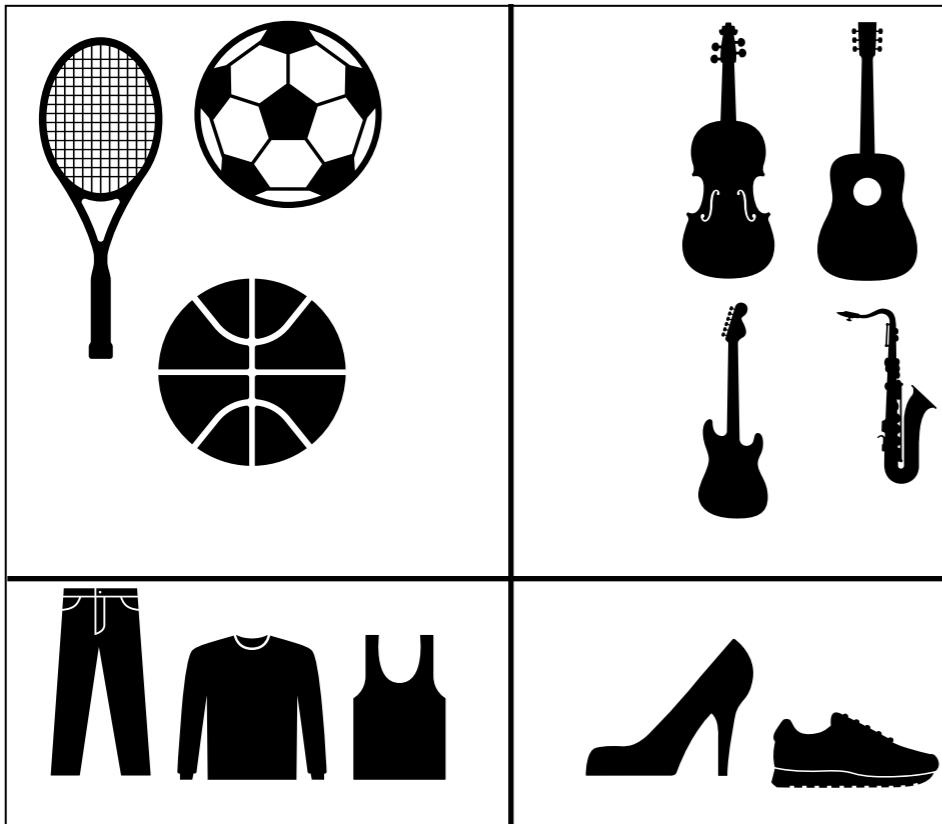
PENGIRISAN TUPLE (SLICING)

Pada pertemuan sebelumnya telah pernah dibahas pengirisan, yaitu pada string. Misal

$$t = (0, 1, 2, 3, 4, 5)$$

Input	Output
<code>t[0]</code>	0
<code>t[5]</code>	5
<code>t[-1]</code>	5
<code>t[-2]</code>	4
<code>t[2:]</code>	(2, 3, 4, 5)
<code>t[:5]</code>	(0, 1, 2, 3, 4)
<code>t[-3:]</code>	(3, 4, 5)
<code>t[:-2]</code>	(0, 1, 2, 3)
<code>t[1:4]</code>	(1, 2, 3)
<code>t[1:-1]</code>	(1, 2, 3, 4)
<code>t[::-1]</code>	(5, 4, 3, 2, 1, 0)
<code>t[::-2]</code>	(5, 3, 1)

NESTED LIST



```
my_data = ['Frankel'  
          (89, 87, 90),  
          2023,  
          (True, False)]  
  
print(my_data)
```



TENTANG NESTED LIST

Nested list atau list bersarang merupakan tipe data list namun di dalam list terdapat data urutan. Contoh:

```
kelas = [[ 'A' ,31] , [ 'B' ,31] , [ 'C' ,30] , [ 'D' ,30]]
```

```
kelas2 = [ ('A' ,31) , ('B' ,31) , ('C' ,30) , ('D' ,30)]
```

kelas[0][0]	Indeks 0 pada kelas data ke-0	A
kelas[1][1]	Indeks 1 pada kelas data ke-1	31
kelas[2][0]	Indeks 2 pada kelas data ke-0	C
kelas[3][1]	Indeks 3 pada kelas data ke-1	30
kelas[-1][0]	Indeks terakhir pada kelas data ke-0	D
kelas[0][-1]	Indeks 0 pada kelas data terakhir	31
kelas[1][0:]	Indeks 1 pada kelas data ke 0 hingga terakhir	['B' ,31]
kelas[-1][:-1]	Indeks terakhir, data ke 0 hingga 0	['D']



LATIHAN LIST

Tentukan output dari kode di bawah ini.

```
data = ['Von Neumann', 2023, 'Aktif']
nilai= [90, 89, 93, 97]

print('Nama: ' + data[0])
print('angkatan: ', data[1])
print('status: ' + data[2])
```

Tentukan kode untuk input di bawah ini.

```
>> Von Neumann status Kuliah: Aktif
>> Data terbesar nilai adalah: 97
>> Data terkecil nilai adalah: 89
>> Rata-rata nilai adalah: 92.25
```



LATIHAN TUPLE

Tentukan output dari kode di bawah ini.

```
data = ('mahasiswa1', 2023, 'Aktif')
nilai= (90, 89, 93, 97)

print(data[1])
print(data[-1])
print(nilai[1:-1])
```

Tentukan kode untuk input di bawah ini.

```
>> Jumlah nilai mahasiswa adalah: 4
>> Data terbesar nilai adalah: 97
>> Data terkecil nilai adalah: 89
>> Rata-rata nilai adalah: 92.25
```



LATIHAN NESTED LIST

Tentukan output dari kode di bawah ini.

```
data = [ ('Thomas', 2023, 'Aktif'),  
        (90, 89, 93, 97),  
        (20, 22),  
        ('S1-Reguler', 'Ganjil')]  
  
print(data[0][0])  
print(data[-1][0])  
print(data[2][0:])
```

Tentukan kode untuk input di bawah ini.

```
>> Program pendidikan Thomas: S1-Reguler  
>> Angkatan : 2023-Ganjil  
>> Jumlah nilai Thomas adalah: 4  
>> Data terbesar Thomas adalah: 97  
>> Data terkecil Thomas adalah: 89  
>> Rata-rata nilai Thomas adalah: 92.25
```



REFERENSI

<https://www.bhutanpythoncoders.com>

<https://www.w3schools.com/>