



# Tipe Data pada Python - List, Slicing, Tuple, Set, dan Dictionary

## List

List adalah jenis kumpulan data teratur (ordered sequence), dan merupakan salah satu variabel yang sering digunakan pada Python. Serupa, namun tak sama dengan array pada bahasa pemrograman lainnya. Bedanya, elemen List pada Python tidak harus memiliki tipe data yang sama. Mendeklarasikan List cukup mudah dengan kurung siku dan elemen yang dipisahkan dengan koma.

Setiap data di dalamnya dapat diakses dengan indeks yang dimulai dari 0.

```
1. a = [1, 2.2, 'python']
```

Python mengenal slicing operator `[]` yang dapat melakukan ekstraksi sebuah item atau beberapa item yang berada dalam range tertentu pada tipe data urutan (sequences), misalnya list, string dan tuple. Beberapa tipe urutan juga mendukung "extended slicing" dengan parameter ketiga berupa "step".

- **`x[0]`** artinya mengambil elemen paling awal, dengan index 0 dari List `x`.
- **`x[5]`** artinya mengambil elemen dengan index 5 dari List `x`.
- **`x[-1]`** artinya mengambil elemen dengan index paling belakang ke-1 dari List `x`.
- **`x[3:5]`** artinya membuat list dari anggota elemen List `x` dengan index 3 hingga sebelum index 5 (tidak termasuk elemen dengan index 5, dalam hal ini hanya index 3-4).
- **`x[:5]`** artinya membuat list dari anggota elemen List `x` paling awal hingga sebelum index 5 (tidak termasuk elemen dengan index 5, dalam hal ini hanya index 0-4).
- **`x[-3:]`** artinya membuat list dari anggota elemen List `x` mulai index ke-3 dari belakang hingga paling belakang.
- **`x[1:7:2]`** artinya membuat list dari anggota elemen List `x` dengan index 1 hingga sebelum index 7, dengan "step" 2 (dalam hal ini hanya index 1, 3, 5).

```
1. x = [5,10,15,20,25,30,35,40]
2. print(x[5])
3. print(x[-1])
4. print(x[3:5])
5. print(x[:5])
6. print(x[-3:])
7. print(x[1:7:2])
```

Output:



DIBANTU



4U

`[20, 25]``[5, 10, 15, 20, 25]``[30, 35, 40]``[10, 20, 30]`

Elemen pada list dapat diubah atau ditambahkan. Misalnya untuk melakukan perubahan kemudian penambahan:

```
1. x = [1,2,3]
2. x[2]=4
3. print(x)
```

Output:

`[1, 2, 4]`

```
1. x = [1,2,3]
2. x[2]=4
3. x.append(5)
4. print(x)
```

Output:

`[1, 2, 4, 5]`

**Untuk menghapus item pada list, gunakan fungsi `del`.** Ingat bahwa Indeks Python dimulai dari 0:

```
1. binatang = ['kucing', 'rusa', 'badak', 'gajah']
2. del binatang[2]
3. print(binatang)
```

Output:

`['kucing', 'rusa', 'gajah']`

DIBANTU



Coba tambahkan lagi kode berikut:

```
1. del binatang[2]
2. print(binatang)
```

Pada baris terbawah kode di atas, maka output akan menjadi:

Output:

```
['kucing', 'rusa']
```

## Slicing pada String

Karena string mirip dengan list, maka slicing operator [ ] juga dapat digunakan pada string untuk mengambil isinya atau bahkan substring. Sebuah string utuh bersifat mutable (bisa diubah), namun elemennya bersifat immutable (tidak bisa diubah).

```
1. s = "Hello World!"
2. print(s[4])           #ambil karakter kelima dari string s
3. print(s[6:11])        #ambil karakter ketujuh hingga sebelas dari string s
4. s[5]="d"              #ubah karakter keenam dari string s menjadi "d", seharusnya gagal karena immutable
5. s = "Halo Dunia!"      #ubah isi string s menjadi "Halo Dunia!", seharusnya berhasil karena mutable
6. print(s)
```

Output:

```
'o'
```

```
'World'
```

Traceback (most recent call last):

File "<stdin>", line 1, in <module>

TypeError: 'str' object does not support item assignment

```
'Halo Dunia!'
```

## Tuple

Tuple adalah jenis dari list yang tidak dapat diubah elemennya. Umumnya tuple digunakan untuk data yang bersifat sekali tulis, dan dapat dieksekusi lebih cepat. Tuple didefinisikan dengan kurung dan elemen dipisahkan dengan koma.



DIBANTU



Seperti list, kita dapat melakukan slicing, namun pada tuple kita tidak dapat melakukan perubahan:

```
1. t = (5, 'program', 1+3j)
2. print(t[1])
3. print(t[0:3])
4. print(t[0]=10)
```

Output:

```
'program'
(5, 'program', (1+3j))
Traceback (most recent call last):
File "<stdin>", line 1, in <module>
TypeError: 'tuple' object does not support item assignment
```

## Set

Set adalah kumpulan item bersifat unik dan tanpa urutan (unordered collection). Didefinisikan dengan kurawal dan elemennya dipisahkan dengan koma. Pada Set kita dapat melakukan union dan intersection, sekaligus otomatis melakukan penghapusan data duplikat.

```
1. a = {1,2,2,3,3,3}
2. print(a)
```

Output:

```
{1, 2, 3}
```

Karena set bersifat unordered, maka kita tidak bisa mengambil sebagian data / elemen datanya menggunakan proses slicing.

```
1. a = {1,2,3}
2. print(a[1])
```

Output:



DIBANTU



```
File "<string>", line 501, in runCode
```

```
File "<interactive input>", line 1, in <module>
```

```
TypeError: 'set' object does not support indexing
```

## Dictionary

Dictionary pada Python adalah kumpulan pasangan kunci-nilai (pair of key-value) yang bersifat tidak berurutan. Dictionary dapat digunakan untuk menyimpan data kecil hingga besar. Untuk mengakses datanya, kita harus mengetahui kuncinya (key). Pada Python, dictionary didefinisikan dengan kurawal dan tambahan definisi berikut:

1. Setiap elemen pair key-value dipisahkan dengan koma (,).
2. Key dan Value dipisahkan dengan titik dua (:).
3. Key dan Value dapat berupa tipe variabel/obyek apapun.

```
1. d = {1:'value','key':2}
2. print(type(d))
```

Output:

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

```
1. d = {1:'value','key':2}
2. print(type(d))
3. print("d[1] = ", d[1]);
4. print("d['key'] = ", d['key']);
```

Output:

```
<class 'dict'>
d[1] = value
d['key'] = 2
```

Dictionary bukan termasuk dalam implementasi urutan (sequences), sehingga tidak bisa dipanggil dengan urutan indeks. Misalnya dalam contoh berikut dicoba dengan indeks 2, tetapi menghasilkan error (KeyError) karena tidak ada kunci (key) 2:



DIBANTU



```
3. print("d[1] = ", d[1]);
4. print("d['key'] = ", d['key']);
5.
6. # Generates error
7. print("d[2] = ", d[2]);
```

Output:

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

```
d[1] = value
```

```
d['key'] = 2
```

```
-----
KeyError Traceback (most recent call last)
```

```
<ipython-input-7-4b566e677ca2> in <module>()
```

```
1 d = {'value':'key':2}
```

```
----> 2 print("d[2] = ", d[2]);
```

```
KeyError: 2
```

## Konversi (conversion, cast) antar tipe data

Kita dapat melakukan konversi tipe data bawaan dengan menggunakan fungsi konversi tipe bawaan (standard type) misalnya: `int()`, `float()`, `str()`, dll.

```
1. print(float(5))
```

Output:

```
5.0
```

Konversi float ke int akan bersifat floor/truncating atau menghilangkan nilai di belakang koma.

```
1. print(int(10.6))
```

Output:



DIBANTU



```
1. print(int(-10.6))
```

Output:

```
-10
```

Konversi dari-dan-ke string akan melalui pengujian dan dipastikan validitasnya.

```
1. print(float('2.5'))
```

Output:

```
2.5
```

```
1. print(str(25))
```

Output:

```
'25'
```

```
1. print(int('1p'))
```

Output:

```
Traceback (most recent call last):
  File "<string>", line 301, in runcode
  File "<interactive input>", line 1, in <module>
ValueError: invalid literal for int() with base 10: '1p'
```



DIBANTU



```
1. print(set([1,2,3]))
```

Output:

```
{1, 2, 3}
```



```
1. print(tuple({5,6,7}))
```

Output:

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



```
1. print(list('hello'))
```

Output:

```
['h', 'e', 'l', 'l', 'o']
```

Untuk konversi ke dictionary, data harus memenuhi persyaratan key-value. Berikut adalah dua contoh konversi:

List dari beberapa List yang isinya pasangan nilai menjadi Dictionary.

Serta konversi List dari beberapa Tuple yang isinya pasangan nilai menjadi Dictionary.



```
1. print(dict([[1,2],[3,4]]))
```

Output:

```
{1: 2, 3: 4}
```



```
1. print(dict([(3,26),(4,44)]))
```



DIBANTU





{3: 26, 4: 44}

< [Sebelumnya](#)

[Selanjutnya](#) >



Dicoding Space  
Jl. Batik Kumeli No.50, Sukaluyu,  
Kec. Cibeunying Kaler, Kota Bandung  
Jawa Barat 40123

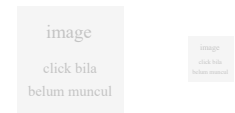


Decode Ideas  
Discover Potential

[Tentang Kami](#)

- [Blog](#)
- [Reward](#)
- [Showcase](#)
- [Hubungi Kami](#)
- [FAQ](#)

Penghargaan



© 2022 Dicoding | Dicoding adalah merek milik PT Presentologics, perusahaan induk dari PT Dicoding Akademi Indonesia.

[Terms](#) • [Privacy](#)



DIBANTU