



LIST

List

- *List* dalam bahasa pemrograman **Python** adalah salah satu metode penanganan kumpulan data dengan tipe data gabungan, selain *tuple*, *set*, dan *dictionary*.
- **List** merupakan sekumpulan data yang terdapat dalam suatu **variabel**.
- *List* berbeda dari *Array* dalam pemrograman C/C++ dan Java.
 - Array* = dapat menyimpan banyak data dengan **tipe yang sama**.
 - List* = dapat menyimpan banyak data dengan **tipe yang berbeda-beda**.
- Indeks/elemen data *list* ditulis di dalam tanda kurung **kotak/siku** “[...]”.
- Indeks data *list* dimulai dari **0**, 1, 2, 3, ..., dst.

Format Penulisan List

```
nama_list = [index-0, index-1, ..., index-n]
```

Contoh:

```
Angka = [ 0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
```

List dengan tipe data yang **sama**

```
data = [ 1, 2, 3, "Hello", 4.9, "Robonesia", 5, 6]
```

List dengan tipe data yang **berbeda**

Mengakses Elemen List Dengan Indeks Positif (1)

```
# List dengan tipe data yang sama
angka = [0,1,2,3,4,5,6,7,8,9]

# Menampilkan nilai suatu index pada suatu list
print("Elemen list ke-0 = ", angka[0])
print("Elemen list ke-7 = ", angka[7])

# Menampilkan 4 data pertama sebuah list
print('Empat data pertama = ', angka[:4])

# Menampilkan 4 data terakhir sebuah list
print('Empat data terakhir = ', angka[10-4:])

# Menampilkan data pada rentang indeks tertentu
print('Data rentang index 3 s/d 8 = ', angka[3:8])
```

```
# output
Elemen list ke-0 = 0
Elemen list ke-7 = 7

Empat data pertama = [0, 1, 2, 3]

Empat data terakhir = [6, 7, 8, 9]

Data rentang index 3 s/d 8 = [3, 4, 5, 6, 7]
```

Mengakses Elemen List Dengan Indeks Positif (2)

```
# List dengan tipe data yang berbeda-beda
data_list = [ 1, 2, 3, "Hello", 4.9, "Robonesia", 5, 6.8]

print("Tiga data awal = ", data_list[:3])
print('Data setelah INDEKS ke-3 = ', data_list[3:])
print("Tiga data terakhir = ", data_list[len(data_list)-3:])
```

```
# output
Tiga data awal =  [1, 2, 3]
Data setelah INDEKS ke-3 =  ['Hello', 4.9, 'Robonesia', 5, 6.8]
Tiga data terakhir =  ['Robonesia', 5, 6.8]
```

Mengakses Elemen List Dengan Indeks **Negatif**

```
# List dengan tipe data yang sama
angka = [0,1,2,3,4,5,6,7,8,9]

print("Elemen list ke -10 = ", angka[-10])
print("Elemen list ke -8 = ", angka[-8])
print("Menampilkan 3 elemen pertama = ", angka[:3])
print("Menampilkan elemen setelah indeks -7 = ", angka[-7:])
```

List	0	1	2	3	4	5	6	7	8	9
Indeks positif	0	1	2	3	4	5	6	7	8	9
Indeks negatif	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1

```
# output
Elemen list ke -10 = 0
Elemen list ke -8 = 2
Menampilkan 3 elemen pertama = [0, 1, 2]
Menampilkan elemen setelah indeks -7 = [3, 4, 5, 6, 7, 8, 9]
```

Mengetahui Panjang List



Fungsi **len()** Cara Mengetahui Panjang List

```
# List dengan tipe data yang berbeda-beda
data_list = [ 1, 2, 3, "Hello", 4.9, "Robonesia", 5, 6.8]

# Mengetahui Panjang Suatu List
A = len(data_list)
print(A)
```

```
# output
Panjang Data_list = 8
```


Modifikasi List Menggunakan Indeks



1. Mengubah Elemen Dalam List

```
data_list1 = [ 7, 5, 9, "Belajar", 7.9, "Python", 10, 7.8]
print(data_list1)                # Tampilkan data list
data_list1[6]= "di Robonesia"    # Mengubah elemen ke-6
print(data_list1)                # Tampilkan data yang telah berubah
data_list1[0]= "Ayo ..."      # Mengubah elemen ke-0
print(data_list1)                # Tampilkan data yang telah berubah
```

```
# output
[7, 5, 9, 'Belajar', 7.9, 'Python', 10, 7.8]
[7, 5, 9, 'Belajar', 7.9, 'Python', 'di Robonesia', 7.8]
['Ayo ...', 5, 9, 'Belajar', 7.9, 'Python', 'di Robonesia', 7.8]
```

2. Menyisipkan elemen ke dalam List

```
data_list2 = [8, 2, "Belajar", 7.9, "Python", 10.2]
print(data_list2)                                # Tampilkan data list
data_list2[:0]=["Sahabat Robonesia"]             # Menyisipkan elemen pada indeks pertama
data_list2[len(data_list2):]=["Yuks!"]           # Menyisipkan elemen pada indeks terakhir
print(data_list2)
```

```
# output
[8, 2, 'Belajar', 7.9, 'Python', 10.2]
['Sahabat Robonesia', 8, 2, 'Belajar', 7.9, 'Python', 10.2, 'Yuks!']
```

3. Menghapus Elemen dalam List

```
data_list3 = [ "Batam", 8, "Riau", "Jogja", 7.9, "Wonogiri", 10.2]
print(data_list3)
data_list3[4:6]=[]      # Menghapus elemen indeks 4 s/d 5
print(data_list3)
```

```
# output
['Batam', 8, 'Riau', 'Jogja', 7.9, 'Wonogiri', 10.2]
['Batam', 8, 'Riau', 'Jogja', 10.2]
```

Modifikasi List Menggunakan Fungsi



1. Menyisipkan elemen ke dalam List (1)

Menyisipkan elemen baru ke dalam suatu list dapat dilakukan menggunakan fungsi **append()**, **insert()**, dan **extend()**.

A. Fungsi **append()**, menyisipkan data baru di akhir list

```
A = [45, 90, 135]
print("List A = ", A)          # Tampilkan data list A
A.append(180)                   # Sisipkan angka 180 di akhir list
print("List A baru = ", A)     # Tampilkan data list A setelah disisipi
```

```
# output
List A = [45, 90, 135]
List A baru = [45, 90, 135, 180]
```

1. Menyisipkan elemen ke dalam List (2)

B. Fungsi **insert()**, menyisipkan data baru **pada indeks tertentu**

Sintak:

```
nama_list.insert(index, data_sisipan)
```

```
B = [30, 90, 120, 150]
print("List B = ", B)      # Tampilkan data list B
B.insert(1, "Enam puluh")  # Sisipkan "Enam puluh" pada indeks ke-1
print("List B baru = ", B) # Tampilkan data list B setelah disisipi
```

```
# output
List B = [30, 90, 120, 150]
List B baru = [30, 'Enam puluh', 90, 120, 150]
```

1. Menyisipkan elemen ke dalam List (3)

C. Fungsi `extend()`, menyisipkan suatu list ke dalam list lainnya

```
C = [70, 140, 210]
D = ["Apel", "Jeruk", "Mangga"]
print("List C = ", C)           # Tampilkan data list C
print("List D = ", D)           # Tampilkan data list D
C.extend(D)                     # Menyisipkan list D ke dalam list C
print("List baru setelah di extend = ", C) # List baru setelah di extend
```

```
# output
List C = [70, 140, 210]
List D = ['Apel', 'Jeruk', 'Mangga']
List baru setelah di extend = [70, 140, 210, 'Apel', 'Jeruk', 'Mangga']
```


2. Menghapus Element List

Menghapus elemen dalam suatu list dapat dilakukan menggunakan fungsi **del** dan **remove()**.

Sintak:

```
del nama_list[index]  
nama_list.remove(nilai_elemen)
```

```
kota = ["Jogja", "Klaten", "Solo", "Sukoharjo", "Wonogiri"]  
print(kota)           # Tampilkan list kota  
del kota[2]           # Hapus elemen indeks 2  
print(kota)  
kota.remove("Klaten")  # Hapus elemen bernilai "Klaten"  
print(kota)
```

```
# output  
['Jogja', 'Klaten', 'Solo', 'Sukoharjo', 'Wonogiri']  
['Jogja', 'Klaten', 'Sukoharjo', 'Wonogiri']  
['Jogja', 'Sukoharjo', 'Wonogiri']
```

3. Mengurutkan & Membalik Elemen List (1)

A. Mengurutkan list data **integer** & Membalik Elemennya

```
bilangan = [25, 19, 35, 47, 77, 63, 81, 94, 101]
print("List angka = ", bilangan)
bilangan.sort()                # Urutkan elemen list dengan 'sort'
print("Hasil sort = ", bilangan)

AA = sorted(bilangan)          # Urutkan elemen list dengan 'sorted'
print("Hasil sorted = ", AA)
bilangan.reverse()             # Balik urutan list elemen
print("Hasil reverse = ", bilangan)
```

```
# output
List angka =  [25, 19, 35, 47, 77, 63, 81, 94, 101]
Hasil sort =  [19, 25, 35, 47, 63, 77, 81, 94, 101]
Hasil sorted = [19, 25, 35, 47, 63, 77, 81, 94, 101]
Hasil reverse = [101, 94, 81, 77, 63, 47, 35, 25, 19]
```

3. Mengurutkan & Membalik Elemen List (2)

B. Mengurutkan list data **string** & Membalik Elemennya

```
sayur = ['bayam', 'kangkung', 'wortel', 'sawi', 'tomat', 'kubis']
print("List sayur = ", sayur)
sayur.sort()                                # Urutkan elemen list dengan 'sort'
print("Hasil sort = ", sayur)

BB = sorted(sayur)                          # Urutkan elemen list dengan 'sorted'
print("Hasil sorted = ", BB)
sayur.reverse()                             # Balik urutan list elemen
print("Hasil reverse = ", sayur)
```

```
# output
List sayur = ['bayam', 'kangkung', 'wortel', 'sawi', 'tomat', 'kubis']
Hasil sort = ['bayam', 'kangkung', 'kubis', 'sawi', 'tomat', 'wortel']
Hasil sorted = ['bayam', 'kangkung', 'kubis', 'sawi', 'tomat', 'wortel']
Hasil reverse = ['wortel', 'tomat', 'sawi', 'kubis', 'kangkung', 'bayam']
```

3. Mengurutkan & Membalik Elemen List (3)

Catatan:

Fungsi `sort()` dan `sorted()` hanya bisa diaplikasikan pada list yang elemen-elemennya memiliki **tipe data yang sama**.

4. Operasi Aritmatika Pada List

```
data = ["Robonesia"]*3
print(data)
data2 = ["Robonesia"] + ["More than robotics learning"]
print(data2)
```

```
# output
['Robonesia', 'Robonesia', 'Robonesia']
['Robonesia', 'More than robotics learning']
```

5. Menggunakan Operator **in** Pada List

```
programming = ["java", "C", "C++", "python", "PHP"]  
print("Apakah pemrograman C++ ada = ", "C++" in programming)  
print("Apakah pemrograman Jawa ada = ", "Jawa" in programming)  
print("Apakah pemrograman Python ada = ", "python" in programming)
```

```
# output  
Apakah pemrograman C++ ada = True  
Apakah pemrograman Jawa ada = False  
Apakah pemrograman Python ada = True
```

List Bersarang (List 2-Dimensi)



List Bersarang (Dua dimensi) - 1

A. Membuat list 2-dimensi

```
data1_2D = [ ["Raspberry", "Blueberry", "Blackberry"], [3, 5, 7] ]  
print(data1_2D)          # Tampilkan list 2D 2x3 (2 baris, 3 kolom)
```

```
# output  
[['Raspberry', 'Blueberry', 'Blackberry'], [3, 5, 7]]
```

B. Mengakses list 2-dimensi

```
data2_2D = [ ["Lampung", "Palembang", "Jambi"], [1, 2, 3] ]  
print(data2_2D)          # Tampilkan list 2D 2x3 (2 baris, 3 kolom)  
print(data2_2D[0][0:])    # Menampilkan elemen pada baris-1  
print(data2_2D[1][0:])    # Menampilkan elemen pada baris-2
```

```
# output  
[['Lampung', 'Palembang', 'Jambi'], [1, 2, 3]]  
['Lampung', 'Palembang', 'Jambi']  
[1, 2, 3]
```


Ilustrasi List Dua Dimensi (1)

Raspberry	Blueberry	Blackberry	← - - - - Baris 0
3	5	7	← - - - - Baris 1

Kolom 0 Kolom 1 Kolom 2

Gambar 1. Visualisasi list dua dimensi - Jumlah elemen **sama**

List Bersarang (Dua dimensi) - 2

C. List 2-dimensi dengan jumlah elemen berbeda

```
data3_2D = ["Bali", "Surabaya", "Semarang", "Bandung"], ["Apel", "Peer"], ["Red", "Green", "Blue"]
print(data3_2D)           # Tampilkan list 2D
print(data3_2D[0][0:])    # Menampilkan elemen pada baris-1
print(data3_2D[1][0:])    # Menampilkan elemen pada baris-2
print(data3_2D[2][0:])    # Menampilkan elemen pada baris-3
```

```
# Output
[['Bali', 'Surabaya', 'Semarang', 'Bandung'], ['Apel', 'Peer'], ['Red', 'Green', 'Blue']]
['Bali', 'Surabaya', 'Semarang', 'Bandung']
['Apel', 'Peer']
['Red', 'Green', 'Blue']
```

Ilustrasi List Dua Dimensi (2)

Bali	Surabaya	Semarang	Bandung	← - - - - Baris 0
Apel	Peer			← - - - - Baris 1
Red	Green	Blue		← - - - - Baris 2

Kolom 0 Kolom 1 Kolom 2 Kolom 3

Gambar 2. Visualisasi list dua dimensi - Jumlah elemen **berbeda**

Terima Kasih