

Week 3 Homework 3

Hermulia Hadie Putra

Jelaskan secara detail definisi masing-masing dari data type disertai contoh kodenya! (silahkan cari referensi dari internet)

Jawab:

"A data type or simply type is a classification of data which tells the compiler or interpreter how the programmer intends to use the data". (Wikipedia)

Secara sederhana, tipe data adalah cara kita memberitahu komputer untuk mengelompokkan data berdasarkan apa yang dipahami oleh komputer.

Dalam bahasa pemrograman Python, terdapat 9 tipe data dasar:

- ❖ String
- ❖ Integer
- ❖ Float
- ❖ Complex Number
- ❖ Boolean
- ❖ List
- ❖ Tuple
- ❖ Set
- ❖ Dictionary

Pembagian 9 jenis tipe data di atas tidak mutlak, karena banyak referensi yang menggabungkan tipe data Integer, Float dan Complex Number sebagai satu tipe data saja, yakni tipe data Number. Jika dibuat seperti ini maka jenis tipe data python menjadi 7 jenis tipe data.

1. String

Tipe data string adalah tipe data untuk menampung data teks, seperti "Indonesia", atau "Saya sedang belajar bahasa Python".

Di dalam bahasa Python, terdapat 3 cara untuk membuat tipe data string:

- Menggunakan tanda kutip satu (')
- Menggunakan tanda kutip dua (")
- Menggunakan tanda kutip satu atau dua sebanyak 3 kali (' ' ') atau (" " ")

```
1 foo = "Duniaikom"
2 print(foo)
3 bar = 'Duniaikom'
4 print(bar)
```

Cara pembuatan string menggunakan 3 kali tanda kutip dipakai untuk membuat multiline string, dimana teks yang diinput bisa tersambung ke beberapa baris.

```
1 foo = '''Teks ini
2 akan dipecah
3 ke dalam 3 baris'''
4 print(foo)
```

Hasilnya:

```
1 Teks ini
2 akan dipecah
3 ke dalam 3 baris
```

Untuk menulis karakter khusus seperti tab atau pindah baris, bisa menggunakan escape character, yakni karakter *backslash* (\) dan diikuti dengan satu karakter khusus. Sebagai contoh, untuk menulis karakter 'tab' bisa dipakai \t, dan untuk membuat karakter new line (pindah baris) bisa menggunakan tanda \n

```
1 foo = 'Teks ini\nakan dipecah\nke dalam 3 baris'
2 print(foo)
```

Hasilnya:

```
1 Teks ini
2 akan dipecah
3 ke dalam 3 baris
```

2. Number (Integer, Float, Complex Number)

Dalam bahasa Python, tipe data number terdiri dari 3 jenis:

- Integer (int): Tipe data bilangan bulat, seperti 1, 300, 59000000
- Float: Tipe data bilangan desimal / pecahan, seperti 0.43, 0.0002, 999.99
- Complex Number: Tipe data bilangan kompleks atau bilangan imajiner, seperti 5j, 54j, 1j

Dari ketiga tipe data ini, tipe data Complex Number adalah tipe data yang cukup unik dan jarang tersedia di bahasa pemrograman lain. Dalam teori matematika, complex number atau bilangan kompleks atau bilangan imajiner adalah sebutan untuk angka yang mengandung nilai akar kuadrat dari -1. Angka akar kuadrat dari -1 ini ditulis dalam Python dengan huruf j. Bilangan 5j sama artinya dengan $5\sqrt{-1}$.

Sama seperti tipe data lain di dalam Python, kita bisa langsung menginput angka-angka ini ke dalam sebuah variabel. Berikut contohnya:

```
1 foo = 100
2 bar = 30.23
3 baz = 4j
4
5 print(foo)
6 print(bar)
7 print(baz)
```

Hasilnya:

```
1 100
2 30.23
3 4j
```

Untuk memeriksa tipe data dari sebuah variabel, terdapat function `type()` bawaan Python. Berikut contoh penggunaannya:

```
1 foo = 100
2 bar = 30.23
3 baz = 4j
4
5 print(type(foo))
```

```
6 print(type(bar))
7 print(type(baz))
```

Hasilnya:

```
1 <class 'int'>
2 <class 'float'>
3 <class 'complex'>
```

Dari hasil ini bisa terlihat bahwa variabel foo yang berisi angka 100 merupakan tipe data int. Variabel bar yang berisi angka 30.23 merupakan tipe data float, dan variabel baz yang diisi dengan angka 4j merupakan tipe data complex.

3. Boolean

Dari seluruh tipe data yang ada di dalam bahasa pemrograman Python, mungkin tipe data Boolean yang paling sederhana. Tipe data ini hanya bisa diisi dengan salah satu dari 2 nilai: True atau False. Tipe data boolean banyak dipakai untuk percabangan kode program atau untuk memutuskan apa yang mesti dijalankan ketika sebuah kondisi terjadi.

Sebagai contoh, kita bisa membuat kode program untuk menentukan apakah sebuah angka genap atau ganjil berdasarkan input dari pengguna. Untuk keperluan ini kita harus mengecek terlebih dahulu apakah angka tersebut bisa dibagi 2 (untuk angka genap), atau tidak bisa dibagi 2 (untuk angka ganjil). Tipe data boolean bisa dipakai untuk menampung kondisi seperti ini, yakni benar atau salah (True atau False). Berikut contoh penulisannya:

```
1 foo = True
2 bar = False
3
4 print(foo)
5 print(bar)
```

Hasilnya:

```
1 True
2 False
```

Dalam kode program diatas, nilai boolean True diinput ke dalam variabel foo dan nilai boolean False diinput ke dalam variabel bar.

Yang juga harus diperhatikan adalah penulisan huruf besar atau kecil. Dalam bahasa Python, penulisannya harus persis seperti di atas, jika diinput sebagai true atau TRUE, akan menghasilkan error.

Selain diinput manual, tipe data boolean lebih sering di dapat sebagai hasil dari operasi perbandingan, seperti apakah suatu angka lebih besar dari angka lainnya, apakah lebih kecil, atau sama dengan. Berikut contoh penggunaan operasi perbandingan ini

```
1 foo = 12 < 10
2 print(foo)
3 foo = 12 > 10
4 print(foo)
5 foo = "A" == "a"
6 print(foo)
```

Hasilnya:

```
1 False
2 True
3 False
```

Di baris pertama, hasil operasi perbandingan $12 < 10$ disimpan ke dalam variabel `foo`. Apakah $12 < 10$? Salah. Sehingga nilai variabel `foo` akan berisi boolean `False`. Di baris ketiga, operasi perbandingannya di balik, yakni apakah $12 > 10$? Benar. Sehingga nilai variabel `foo` akan berisi boolean `True`.

4. List

Secara sederhana, tipe data List adalah sebuah array, yakni tipe data yang berisi kumpulan tipe data lain. Namun berbeda seperti array dalam bahasa pemrograman lain, List bisa diisi dengan berbagai jenis data, tidak harus tipe data yang sama.

Untuk membuat tipe data List, gunakan tanda kurung siku, kemudian setiap anggota list dipisah dengan tanda koma. Berikut contohnya:

```
1 foo = ["Belajar", "Python", "di", "Duniaikom"]
2 bar = [100, 200, 300, 400]
3 baz = ["Python", 200, 6.99, True]
4
5 print(foo)
6 print(bar)
7 print(baz)
```

Hasilnya:

```
1 ['Belajar', 'Python', 'di', 'Duniaikom']
2 [100, 200, 300, 400]
3 ['Python', 200, 6.99, True]
```

Dalam kode diatas, variabel `foo` di baris pertama diisi dengan list yang terdiri dari 4 tipe data string. Variabel `bar` di baris kedua diisi dengan list yang terdiri dari 4 tipe data integer atau angka bulat. Dan terakhir tipe data `baz` diisi dengan berbagai jenis tipe data.

Perintah `print` bisa dipakai untuk menampilkan semua isi List secara langsung. Ini sangat praktis karena di dalam bahasa pemrograman lain biasanya kita tidak bisa menampilkan isi array secara langsung, tapi harus menggunakan fungsi khusus atau diakses menggunakan perulangan.

5. Tuple

Tipe data Tuple sangat mirip seperti tipe data List, dimana tipe data Tuple juga terdiri dari kumpulan tipe data lain. Bedanya, anggota di dalam tipe data Tuple tidak bisa diubah setelah di deklarasikan.

Untuk membuat tipe data Tuple, gunakan tanda kurung biasa, kemudian setiap anggota list dipisah dengan tanda koma. Berikut contohnya:

```
1 foo = ("Belajar", "Python", "di", "Duniaikom")
2 bar = (100, 200, 300, 400)
3 baz = ("Python", 200, 6.99, True)
4
5 print(foo)
6 print(bar)
7 print(baz)
```

Hasilnya:

```
1 ('Belajar', 'Python', 'di', 'Duniaikom')
2 (100, 200, 300, 400)
3 ('Python', 200, 6.99, True)
```

Nyaris tidak berbeda dengan List, namun tipe data Tuple dibuat menggunakan tanda kurung bulat, bukan tanda kurung siku sebagaimana List.

6. Set

Tipe data Set Python adalah tipe data yang berisi kumpulan tipe data dan dipakai untuk mengolah himpunan (set). Jika dibandingkan dengan tipe data berbentuk array lain di Python, tipe data set berbeda dalam hal index, pengurutan dan keunikan nilai (unique value).

Tipe data set tidak memiliki index, sehingga tidak ada mekanisme pengurutan. Artinya, ketika kita menginput beberapa nilai ke dalam tipe data set, posisi nilai tersebut bisa berada di mana saja, tidak persis seperti apa yang tertulis. Akibat tidak memiliki index, maka kita tidak bisa menambah nilai baru ke dalam tipe data set dengan cara menulis nomor index (seperti di dalam tipe data list).

Ciri khas selanjutnya dari tipe data set adalah tidak bisa menerima nilai ganda, setiap nilai di dalam set harus unik. Jika kita menginput beberapa nilai yang sama, hanya satu yang tersimpan di dalam set.

Untuk membuat tipe data set, gunakan tanda kurung kurawal, kemudian setiap anggota set dipisah dengan tanda koma. Berikut contohnya:

```
1 foo = {"Belajar", "Python", "di", "Duniaikom"}
2 bar = {100, 200, 300, 400}
3 baz = {"Python", 200, 6.99, True}
4
5 print(foo)
6 print(bar)
7 print(baz)
```

Hasilnya:

```
1 {'Belajar', 'di', 'Duniaikom', 'Python'}
2 {200, 100, 400, 300}
3 {200, True, 6.99, 'Python'}
```

Selain perbedaan tanda kurung, cara penulisan tipe data set tidak berbeda dengan tipe data list dan tuple.

Jika diperhatikan urutan data yang tampil, hampir semuanya tidak terurut sesuai penulisan. Ini karena di dalam tipe data set setiap anggota atau element tidak memiliki index sehingga posisinya tidak bisa dipastikan.

7. Dictionary

Tipe data dictionary adalah tipe data array dimana key atau index dari array bisa berbentuk string, tidak hanya number saja. Dalam bahasa pemrograman lain (seperti PHP) dictionary ini dikenal juga dengan sebutan associative array. Berikut format dasar penulisan dictionary dalam bahasa Python:

```
nama_variabel = { "key1": "value1", "key2": "value2", "key3": "value3" }
```

Tipe data dictionary ini cocok dipakai untuk kelompok data yang kompleks (terdiri dari banyak element).

Pembuatan dictionary menggunakan tanda kurung kurawal { dan } . Selain itu, setiap element merupakan pasangan dari key dan value. Antar satu element dengan element lain dipisah dengan tanda koma seperti contoh berikut:

```
1     foo = { 1: "Belajar", 2: "Python", 3: "di Duniailkom" }
2     bar = { "mengapa": "Belajar", "apa": "Python", "dimana": "di Duniailkom" }
3     baz = { 1: "Belajar", "apa": "Python", "dimana": "di Duniailkom" }
4
5     print(type(foo))
6     print(type(bar))
7     print(type(baz))
8
9     print(foo)
10    print(bar)
11    print(baz)
```

Hasilnya:

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

{1: 'Belajar', 2: 'Python', 3: 'di Duniailkom'}
{'mengapa': 'Belajar', 'apa': 'Python', 'dimana': 'di Duniailkom'}
{1: 'Belajar', 'apa': 'Python', 'dimana': 'di Duniailkom'}
```

Dari contoh tersebut, terdapat 3 buah variabel: foo, bar dan baz yang masing-masingnya diisi dengan tipe data dictionary. Untuk variabel foo diisi dengan dictionary dimana semua key-nya berupa angka (1, 2 dan 3). Jika ditulis seperti ini, foo mirip seperti tipe data list. Untuk variabel bar, diisi semua key berbentuk string (“mengapa”, “apa”, dan “dimana”). Terakhir untuk variabel baz mengkombinasikan penulisan key antara number dan string (1, “apa”, dan “dimana”).

Perintah di baris 5 – 7 membuktikan bahwa ketiga variabel bertipe dict (singkatan dari dictionary). Untuk menampilkan seluruh isi dictionary, kita bisa memakai perintah print seperti di baris 9 – 11.