



KONSEP PEMROGRAMAN

[Python 3.6.2]

Abstract

Instalasi Python, Data Types (Structure), Operator

Puji Winar Cahyo
STMIK Jend A Yani [Yogyakarta]

A. Tujuan Pembelajaran.

Mahasiswa diharapkan dapat :

- 1) Memahami cara instalasi python
- 2) Memahami tipe data (Data Structure)
- 3) Memahami variable
- 4) Memahami operator


B. Pengantar

Python Installation

1. Untuk melakukan instalasi python download file master python di :

<https://www.python.org/downloads/>

Kemudian akan memperoleh file master python sesuai versi yang telah didownload

Name	Date modified	Type	Size
 python-3.6.2	9/24/2017 6:44 PM	Application	29,793 KB

2. Klik 2 kali pada master python yang telah berhasil didownload tersebut. Kemudian Next Sehingga python berhasil terinstall (Default installation in C:)
3. Setting path python :
 - a. My Computer → Properties
 - b. Open System Properties → Advanced Tab
 - c. Environment Variables → System Variables, Click New
 - d. Variable Name : PYTHON_HOME, Variable Value = *Path Python Installation*
4. Test Python, Open cmd dan ketikan python:

```
C:\Users\Cahyo>python
Python 3.6.2 (v3.6.2:5fd33b5926, Jul 16 2017, 20:11:06)
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

5. Test Cetak Hello Word, Untuk mencetak output di python menggunakan perintah print kurung bulat, print(*output*)

```
>>> print("Hello Word")
>>> Hello Word
```

C. Implementasi

1. Pengenalan tipe data :

Setiap data pasti memiliki tipe, di dalam bahasa pemrograman python setiap data dapat dilakukan check tipe data yang dimiliki dengan menggunakan perintah `type(data)`.

Berikut berbagai macam tipe data yang tersedia dalam bahasa pemrograman python, diantaranya adalah :

a) Tipe data **Number**

1) Int

Int (integer) merupakan tipe data bilangan bulat. Seperti contoh data 30, maka dapat dilakukan pengecekan sebagai berikut :

```
>>> type(30)
>>> <class 'int'>
```

Long int merupakan bilangan yang mempunyai tipe data sama seperti integer (bilangan bulat), hanya saja memiliki cakupan yang lebih panjang, (dalam python 3 long int tetap didefinisikan sebagai int)

2) Float

Float merupakan tipe data bilangan real. Seperti contoh data 30.0, maka dapat dilakukan pengecekan sebagai berikut :

```
>>> type(30.0)
>>> <class 'float'>
```

3) Complex

Tipe data complex merupakan tipe data untuk bilangan complex.

```
>>> type(2+3j)
>>> <class 'complex'>
```

4) Boolean

Tipe data boolean adalah tipe data yang hanya bernilai True atau False, seperti contoh berikut.

```
>>> type(True)
>>> <class 'bool'>
>>> type(False)
>>> <class 'bool'>
```

b) Tipe data **String**

String merupakan data yang berasal dari kumpulan karakter yang berbentuk teks, untuk sebab itu data string pasti berada diantara tanda petik “”. Dengan contoh kalimat “Hello Word”

```
>>> type("Hello Word")
>>> class 'str'>
```

c) Tipe Data **None**

None digunakan untuk mendefinisikan no value/tidak ada.

```
>>> type("None")
>>> class 'NoneType'>
```

d) **Data Structure**

1) List

List merupakan kumpulan data dari beberapa tipe data (sejenis ataupun bervariasi), definisi list selalu berada diantara tanda kurung siku [*isi list*].

Contoh sebagai berikut.

```
>>> type([1,2,3,4,5])
>>> class 'list'>
>>> type([1,2,3.90,"Hello World",5])
>>> class 'list'>
```

2) Tuple

Tuple merupakan kumpulan data dari beberapa tipe data (sejenis ataupun bervariasi), definisi tuple selalu berada diantara tanda kurung bulat (*isi tuple*). Contoh sebagai berikut.

```
>>> type((1,2,3,4,5))
>>> class 'tuple'>
>>> type((1,2,3.90,"Hello World",5))
>>> class 'tuple'>
```

3) Set

Set merupakan kumpulan data dari beberapa tipe data (sejenis ataupun bervariasi, akan tetapi secara unique), definisi set selalu berada diantara tanda kurung kurawal *{isi set}*. Contoh sebagai berikut.

```
>>> type({"banana","apple","banana","grape"})
>>> class 'set'>
>>> print({"banana","apple","banana","grape"})
>>> {"banana","apple","grape"} //unique
```

4) Dictionary

Dictionary hampir sama seperti set akan tetapi isi dari dictionary terdiri dari key value pairs, sehingga setiap value akan memiliki key bisa bertipe (integer/string/float), seperti contoh sebagai berikut.

```
>>> type({1:"banana",2:"apple",3:"banana",4:"grape"})
>>> class 'dict'>
>>> type({"a":"banana","b":"apple","c":"banana","d":"grape"})
>>> class 'dict'>
```

2. Operator

Bahasa pemrograman python mendukung beberapa operator, diantaranya adalah operator aritmatika, operator perbandingan, assignment operator, operator logika, bitwise operator, membership operator, operator identifikasi.

1. Operator Aritmatika

Operasi	Keterangan	Contoh	Hasil
+	Penambahan	4+5	10
-	Pengurangan	4-5	-1
*	Perkalian	4*5	20
/	Pembagian	4/5	0.8
//	Pembagian pembulatan kebawah apabila hasilnya positive, pembulatan keatas apabila hasilnya negative	4//5	0
%	Modulus (hasil bagi)	4%5	4
**	Perpangkatan	4**5	1024

2. Operator perbandingan

Operasi	Keterangan	Contoh	Hasil
<code>==</code>	Sama dengan	<code>4 == 5</code>	False
<code>!=</code>	Tidak sama dengan	<code>4 != 5</code>	True
<code>></code>	Lebih besar dari	<code>4 > 5</code>	False
<code><</code>	Lebih kecil dari	<code>4 < 5</code>	True
<code>>=</code>	Lebih besar sama dengan	<code>4 >= 5</code>	False
<code><=</code>	Lebih kecil sama dengan	<code>4 <= 5</code>	True

3. Assignment operator

Operasi	Keterangan	Contoh	Hasil
<code>=</code>	$a = b + c$ (nilai a dihasilkan dari $b + c$)	<code>a = 4+5</code>	9
<code>+=</code>	<code>c += a</code> sama dengan <code>c = c + a</code>	<code>c = 0</code> <code>c += 1</code>	1
<code>-=</code>	<code>c -= a</code> sama dengan <code>c = c - a</code>	<code>c = 5</code> <code>c -= 1</code>	4
<code>*=</code>	<code>c *= a</code> sama dengan <code>c = c * a</code>	<code>c = 5</code> <code>c *= 5</code>	25
<code>/=</code>	<code>c /= a</code> sama dengan <code>c = c / a</code>	<code>c = 5</code> <code>c /= 5</code>	1
<code>%=</code>	<code>c %= a</code> sama dengan <code>c = c % a</code>	<code>c = 5</code> <code>c %= 2</code>	1
<code>**=</code>	<code>c //= a</code> sama dengan <code>c = c // a</code>	<code>c = 5</code> <code>c //= 2</code>	2

4. Operator Logika

Operasi	Keterangan	Contoh	Hasil
<code>and</code>	Logika AND	<code>(True and False)</code>	False
<code>or</code>	Logika OR	<code>(True or False)</code>	True
<code>not</code>	Logika not	<code>not(True)</code>	False

5. Bitwise operator

Assumsikan bahwa $a = 60$ dan $b = 13$

Maka akan menghasilkan bilangan biner

$a = 0011\ 1100$

$b = 0000\ 1101$

Operasi	Keterangan	Contoh	Hasil
&	Bitwise AND	$(a \& b)$	0000 1100 (12)
	Bitwise OR	$(a b)$	0011 1101 (61)
^	Bitwise XOR	$(a \wedge b)$	0011 0001 (49)
~	Negasi bit	$\sim a$	1100 0011 (-61)
<<	Binary Left Shift	$a \ll 2$	1111 0000 (240)
>>	Binary Right Shift	$a \gg 2$	0000 1111 (15)

6. Membership operator

Operasi	Keterangan	Contoh	Hasil
in	Ditemukan	"a" in ["a","b"]	True
		"a" in ["b","c"]	False
not in	Tidak Ditemukan	"a" not in ["a","b"]	False
		"a" not in ["b","c"]	True

7. Operator identifikasi

Operasi	Keterangan	Contoh	Hasil
is	Sama	$a = 20$ $b = 20$ $(a \text{ is } b)$	True
		$a = 20$ $b = 21$ $(a \text{ is } b)$	False
is not	Beda	$a = 20$ $b = 21$ $(a \text{ is not } b)$	True
		$a = 20$ $b = 20$ $(a \text{ is not } b)$	False

3. Pengenalan variable dan assignment

Variable adalah definisi yang menampung data, semisal definisi a yang berisi data 30 maka a adalah variable, sehingga dapat dituliskan seperti assignment berikut.

```
>>> a = 30
```

dilakukan check tipe data pada variable a sebagai berikut :

```
>>> type(a)
>>> <type 'int'>
```

Latihan 1

Buatlah program sederhana untuk menangani perhitungan luas segitiga dengan :

- Alas = 20
- Tinggi = 4
- $\text{Luas} = \frac{1}{2} (\text{Alas}) \times \text{Tinggi}$

Dari keterangan tersebut maka buatlah :

1. Pseudo Code
2. Flow Chart
3. Penerapan Code Kedalam Bahasa Python