

Tugas pemrograman 2

idam fadilah

23 oktober 2019

Chapter 1

Python

1.1 Variabel

Variabel adalah sebuah tempat untuk menampung value dimemori, ibaratkan sebuah ruangan atau wadah, variabel dibagi dua berdasarkan ruang lingkup yaitu variabel lokal dan global, untuk menentukan variabel global atau lokal itu tergantung dari tempat dideklarasikannya variabel pada program yang sedang dibuat. Variabel global yaitu variabel yang dapat diakses di semua lingkup dalam program yang sedang dibuat, dalam kata lain variabel global ini dapat dikenali oleh semua fungsi dan prosedur, sementara variabel lokal yaitu variabel yang dapat diakses hanya di lingkup khusus, dalam kata lain variabel lokal ini hanya bisa diakses pada fungsi/prosedur dimana variabel itu dideklarasikan.

```
def no():  
    nomer=10  
    print(nomer)  
nomer=2  
no()  
print(nomer)  
no()
```

1.2 Input dan output

Input & output tentunya wajib kita ketahui agar pengguna dan program dapat berinteraksi, berikut merupakan syntax meminta inputan :

- var input("message")

var yaitu variabel yang akan menampung inputan dari user, input merupakan fungsi yang digunakan untuk menangkap inputan dari user, message digunakan untuk mencetak suatu keterangan saat meminta inputan, contoh :

- `nama = input("Masukan nama anda : ")`

sementara untuk menampilkan data bisa menggunakan fungsi `print()`, contoh :

```
print("Halo galaksi")
```

atau kalian juga bisa menampilkan isi suatu variabel, untuk menampilkan data variabel yaitu dengan menuliskan nama variabel tanpa tanda petik, contoh :

- `nama="Idam fadilah"`
`print(nama)`

atau mungkin digabung yaitu dengan menambahkan koma untuk memisahkan antara message dengan variabel, contoh :

- `nama="idam fadilah"`
`print("Halo ",nama," Selamat datang di galaksi bima sakti")`

1.3 Operasi aritmatika dan casting

1.3.1 Operasi aritmatika

Sebagai bahasa pemrograman, pastinya python mempunyai operasi aritmatika antara lain seperti tambah, kurang kali, bagi. symbol yang digunakan dalam operasi aritmatikanya pun tidak jauh berbeda dengan rekan-rekannya. berikut merupakan contoh penggunaan operasi aritmatika pada python Contoh misalkan disini kita mempunyai variable : `a=7` dan `b=3`

1.3.2 Casting

Namun apa yang akan terjadi bila mana ternyata variabel `a` merupakan string dan variabel `b` merupakan integer, contoh `a="7"` dan `b=3`, tentu program akan melemparkan error bukan? Disinilah peran casting digunakan.

Casting yaitu cara/mekanisme untuk mengubah tipe data dari suatu data primitive, maksudnya gimana? Jadi misalkan kita akan menjumlahkan variabel a dan b seperti contoh diatas tetapi logikanya sebuah kata (string) tidak akan bisa dijumlahkan dengan angka ("7" + 3) karena variabel a diapit oleh tanda kutip, ini berarti variabel a bertipe data string untuk itu kita perlu merubah dulu variabel a yang tadinya string menjadi integer. Berikut adalah syntax untuk melakukan casting :

- `int(var/value)` : mengubah tipe data ke integer, contoh `int(angka)`
- `float(var/value)` : mengubah tipe data ke float, contoh `float(hasil)`
- `string(var/value)` : mengubah tipe data ke str, contoh `string(12)`

jadi cara menyelesaikan contoh kasus diatas yaitu dengan menggunakan `int()` untuk merubah tipe data variabel a menjadi integer, contoh :

- ```
a="7"
b=3
int(a)+b
```

maka jika dijalankan program diatas tidak akan error, karena tipe data dari variabel a sudah diubah dari string menjadi integer sehingga program dapat dijalankan dengan lancar

## 1.4 condision (kondisi)

Pengambilan keputusan kadang diperlukan dalam sebuah program untuk menentukan tindakan apa yang akan dilakukan sesuai dengan kondisi yang terjadi, contoh kasus misalkan ada seorang anak bernama idam, seorang manusia biasa yang membutuhkan makan, jika idam lapar maka idam akan makan. Maka dapat dipecah seperti dibawah ini :

Kondisi, jika :  
Idam lapar  
Maka :  
Idam akan makan

Namun kadang kondisi tidak juga sampai disitu, kadang ada saja tambahan opsi sebuah kondisi, misalkan jika idam makan maka idam hidup, namun jika tidak maka idam akan mati. Maka dapat dipecah seperti ini :

Kondisi, jika :  
 Idam makan  
 Maka :  
 Idam akan hidup  
 Jika tidak :  
 Idam akan mati

Contoh diatas dapat ditulis dalam syntax python dengan menggunakan if, pengkondisian if dalam python dibagi menjadi 4, yaitu : if, if else, elif, nested if. Berikut merupakan pembahasannya :

### 1.4.1 if

if digunakan jika misalkan kita mempunyai kondisi yang sederhana.

| <u>sintax</u>                              | <u>Contoh</u>                                                        |
|--------------------------------------------|----------------------------------------------------------------------|
| if <u>kondisi</u> :<br><br><u>perintah</u> | if <u>umur</u> < 15:<br><br>print( <u>"anda tidak boleh masuk"</u> ) |

### 1.4.2 if else

If else digunakan jika kita mempunyai kondisi namun jika kondisi tidak terpenuhi maka kita akan melakukan sesuatu yang lain, penggunaan else dalam kondisi tidak diwajibkan, digunakan hanya jika diperlukan.

| <u>sintax</u>                                                                  | <u>Contoh</u>                                                                                                               |
|--------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------|
| if <u>kondisi</u> :<br><br><u>perintah</u><br><br>else:<br><br><u>perintah</u> | if <u>umur</u> < 15:<br><br>print( <u>"anda tidak boleh masuk"</u> )<br><br>else:<br><br>print( <u>"anda boleh masuk"</u> ) |

### 1.4.3 elif

Elif digunakan jika kita mempunyai 2 atau lebih opsi kondisi

| <u>sintax</u>                                                                    | <u>Contoh</u>                                                                                                                             |
|----------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------|
| <u>if kondisi:</u><br><u>perintah</u><br><u>elif kondisi:</u><br><u>perintah</u> | <u>if umur &lt; 15:</u><br><u>print("anda tidak boleh masuk")</u><br><u>elif umur &gt; 50:</u><br><u>print("anda juga gaboleh masuk")</u> |

#### 1.4.4 nested if

Nested if/if bersarang atau disebut juga if dalam if, digunakan jika misal kita mempunyai Kondisi yang ingin ditambahkan jika kondisi sebelumnya sudah terpenuhi

| <u>sintax</u>                                               | <u>Contoh</u>                                                                                      |
|-------------------------------------------------------------|----------------------------------------------------------------------------------------------------|
| <u>if kondisi:</u><br><u>if kondisi:</u><br><u>perintah</u> | <u>if umur &lt; 15:</u><br><u>if umur &gt; 8 :</u><br><u>print("7 tahun lagi anda bisa masuk")</u> |

### 1.5 loop (perulangan)

Dalam membuat sebuah program, kadang kita perlu satu baris atau satu blok kode beberapa kali, misalkan kita membuat program input nama jika misal namanya salah maka program akan meminta inputan lagi, bagaimana jika misalkan user salah input nama 10 kali atau mungkin lebih tentu kita harus menyiapkan banyak sekali kode. disini fungsi perulangan dipakai, perulangan dalam python ada 2 yaitu for dan while.

#### 1.5.1 for

perulangan yang sudah diketahui akhirnya, pada perulangan ini wajib mencantumkan batas perulangan

```
for x in range(5):
 print("Hello world")
```

## 1.5.2 while

perulangan yang akan terus mengulang selama kondisi true

```
x=0
while x < 5:
 print("Hello world")
 x+=1
```

## 1.6 Error

Error yang didapat :

- IndexError: string index out of range  
penanganan errornya bisa menggunakan while dan fungsi len() untuk memastikan panjang NPM yang di inputkan memiliki panjang yang sama dengan yang ditetapkan
- TypeError: unsupported operand type(s) for +: 'int' and 'str'  
penanganan error ini bisa ditangani menggunakan casting operand kedua menjadi integer
- TypeError: can only concatenate str (not "int") to str  
penanganan error ini bisa ditangani menggunakan casting operand kedua menjadi string

## 1.7 Try Except

dalam membuat sebuah program pasti ada saja kode yang membuat program crash saat dijalankan, contoh jika kita menuliskan  $2/0$  dalam program maka program akan error, salah satu cara menanganinya yaitu dengan menggunakan statement try except, contoh penggunaan :

```
a="1"
b=1

try:
 a+b
except:
 print("Error, hanya bisa menggabungkan string dengan string")
```

## Chapter 2

### Latihan soal

#### 2.1 Soal

##### 2.1.1 Soal 1

```
print("+++++____+++++____+++++____+++++____+++++____+++++")
print("+++++____+++++____+++++____+++++____+++++____+++++____+++++")
print("+++++____+++++____+++++____+++++____+++++____+++++____+++++")
print("+++++____+++++____+++++____+++++____+++++____+++++____+++++")
print("+++++____+++++____+++++____+++++____+++++____+++++____+++++")
```

##### 2.1.2 Soal 2

```
npm=int(input("masukan NPM anda : "))
TwoLastDigit=abs(npm)%100
for i in range(TwoLastDigit):
 print("Halo , ", npm, " apa kabar ?")
```

##### 2.1.3 Soal 3

```
npm=int(input("Masukan NPM : "))
key=str(npm%1000)
print("Halo , "+str(npm)[4]+str(npm)[5]+str(npm)[6]+" apa kabar ?")

for i in range(int(str(npm)[4])+int(str(npm)[5])+int(str(npm)[6])-1):
 print("Halo , "+str(npm)[4]+str(npm)[5]+str(npm)[6]+" apa kabar ?")
```

##### 2.1.4 Soal 4

```
npm=int(input("Masukan NPM : "))
```



```
key=npn%1000
str_key=str(key)
print("Halo , "+str_key[0]+" apa kabar ")
```

### 2.1.5 Soal 5

```
i=0
npm=input("Masukan NPM: ")
while i<1:
 if len(npm) < 7:
 print("NPM Kurang dari 7 digit")
 npm=input("Masukan NPM: ")
 elif len(npm) > 7:
 print("NPM lebih dari 7 digit")
 npm=input("Masukan NPM: ")
 else:
 i=1
a=npn[0]
b=npn[1]
c=npn[2]
d=npn[3]
e=npn[4]
f=npn[5]
g=npn[6]
```

```
for x in a,b,c,d,e,f,g:
 print(x,end=""),
```

### 2.1.6 Soal 6

```
i=0
npm=input("Masukan NPM: ")
while i<1:
 if len(npm) < 7:
 print("NPM Kurang dari 7 digit")
 npm=input("Masukan NPM: ")
 elif len(npm) > 7:
 print("NPM lebih dari 7 digit")
 npm=input("Masukan NPM: ")
 else:
 i=1
a=npn[0]
```

```

b=npm[1]
c=npm[2]
d=npm[3]
e=npm[4]
f=npm[5]
g=npm[6]
y=0

for x in a,b,c,d,e,f,g:
 y+=int(x)
print(y)

```

### 2.1.7 Soal 7

```

i=0
npm=input("Masukan NPM: ")
while i < 1:
 if len(npm) < 7:
 print("NPM Kurang dari 7 digit")
 npm=input("Masukan NPM: ")
 elif len(npm) > 7:
 print("NPM lebih dari 7 digit")
 npm=input("Masukan NPM: ")
 else:
 i=1
a=npm[0]
b=npm[1]
c=npm[2]
d=npm[3]
e=npm[4]
f=npm[5]
g=npm[6]
conv=1

```

```

for x in a,b,c,d,e,f,g:
 conv*=int(x)
print(conv)

```

### 2.1.8 Soal 8

```

i=0
npm=input("Masukan NPM: ")

```

```

while i < 1:
 if len(npm) < 7:
 print("NPM_Kurang_dari_7_digit")
 npm=input("Masukan_NPM: ")
 elif len(npm) > 7:
 print("NPM_lebih_dari_7_digit")
 npm=input("Masukan_NPM: ")
 else:
 i=1
a=npm[0]
b=npm[1]
c=npm[2]
d=npm[3]
e=npm[4]
f=npm[5]
g=npm[6]

for x in a,b,c,d,e,f,g:
 print(x)

```

### 2.1.9 Soal 9

```

i=0
npm=input("Masukan_NPM: ")
while i < 1:
 if len(npm) < 7:
 print("NPM_Kurang_dari_7_digit")
 npm=input("Masukan_NPM: ")
 elif len(npm) > 7:
 print("NPM_lebih_dari_7_digit")
 npm=input("Masukan_NPM: ")
 else:
 i=1
a=npm[0]
b=npm[1]
c=npm[2]
d=npm[3]
e=npm[4]
f=npm[5]
g=npm[6]
conv=1

```

```

for x in a,b,c,d,e,f,g:

 if int(x)%2==0:
 if int(x)==0:
 x=""
 print(x,end=" ")

```

### 2.1.10 Soal 10

```

i=0
npm=input("Masukan NPM: ")
while i<1:
 if len(npm) < 7:
 print("NPM Kurang dari 7 digit")
 npm=input("Masukan NPM: ")
 elif len(npm) > 7:
 print("NPM lebih dari 7 digit")
 npm=input("Masukan NPM: ")
 else:
 i=1
a=npm[0]
b=npm[1]
c=npm[2]
d=npm[3]
e=npm[4]
f=npm[5]
g=npm[6]
conv=1

```

```

for x in a,b,c,d,e,f,g:

 if int(x)%2==1:
 print(x,end=" ")

```

### 2.1.11 Soal 11

```

i=0
npm=input("Masukan NPM: ")
while i<1:
 if len(npm) < 7:
 print("NPM Kurang dari 7 digit")

```

```

 npm=input("Masukan NPM: ")
 elif len(npm) > 7:
 print("NPM lebih dari 7 digit")
 npm=input("Masukan NPM: ")
 else:
 i=1
a=npm[0]
b=npm[1]
c=npm[2]
d=npm[3]
e=npm[4]
f=npm[5]
g=npm[6]
conv=1

for x in a,b,c,d,e,f,g:
 if int(x) > 1:
 for i in range(2,int(x)):
 if (int(x) % i) == 0:
 break
 else:
 print(int(x),end=""),

```

### 2.1.12 Soal 2err

```

a="1"
b=1

```

```

try:
 a+b
except:
 print("Error , hanya bisa menggabungkan string dengan string")

```