

Tugas Praktikum Pemrograman II

(Chapter 2)



Disusun oleh :

Dinda Anik Masruro

D4 TI 2C

1.18.4.003

PROGRAM DIPLOMA IV POLITEKNIK POS INDONESIA

POLITEKNIK POS INDONESIA

BANDUNG

2019

1 Tugas Teori

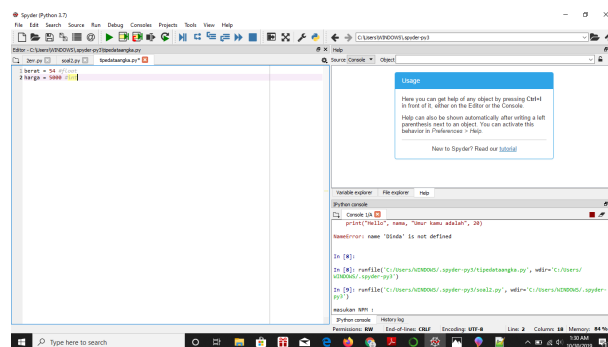
1.1 Jenis-Jenis Variabel dan Cara Pemakaian Variabel

Variabel merupakan suatu wadah untuk menyimpan data, sedangkan data yang disimpan di dalam variabel disebut tipe data. Tipe data terbagi menjadi beberapa macam, yaitu:

1. Tipe Data Angka

Tipe ini terbagi menjadi beberapa jenis lagi, yaitu:

- (a) Integer (bilangan bulat), Contoh : 1,2,3,dst.
- (b) Float (bilangan pecahan), contoh : 1.5, 2.1, dst.



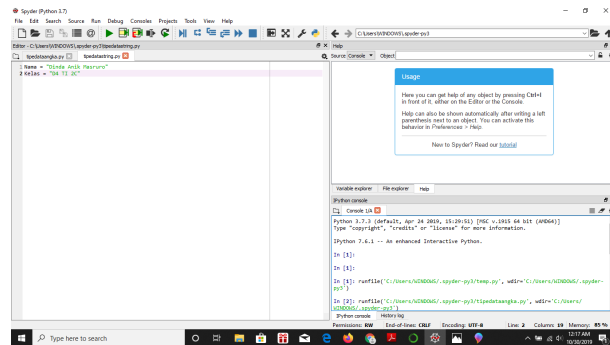
berat = 54 *#float*

harga = 5000 *#int*

2. Tipe Data Teks

Tipe ini terbagi menjadi dua jenis, yaitu:

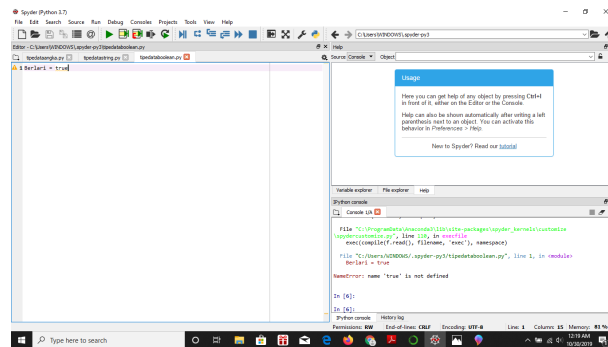
- (a) String (kumpulan karakter), Contoh : "Saya sedang makan"
- (b) Varchar (karakter), Contoh : a,b,c,dst.



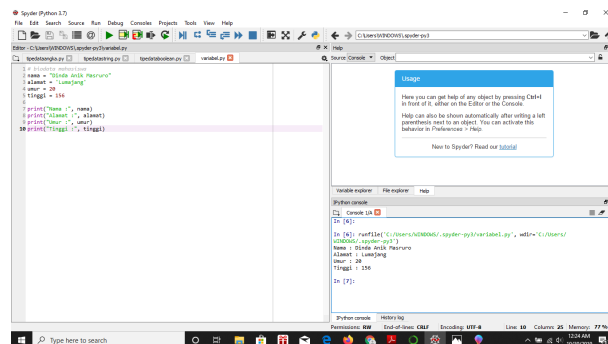
Penulisan tipe ini harus diapit oleh tanda petik.

3. Tipe Data Boolean

Tipe ini adalah tipe yang hanya memiliki dua nilai yaitu True dan False atau 0 dan 1.



Adapun contoh penggunaan variabel menggunakan kode python adalah:



```
# biodata mahasiswa
nama = "Dinda Anik Masruro"
alamat = 'Lumajang'
umur = 20
tinggi = 156

print("Nama :", nama)
print("Alamat :", alamat)
print("Umur :", umur)
print("Tinggi :", tinggi)
```

1.2 Kode Untuk Meminta Input Dari User dan Melakukan Output ke Layar

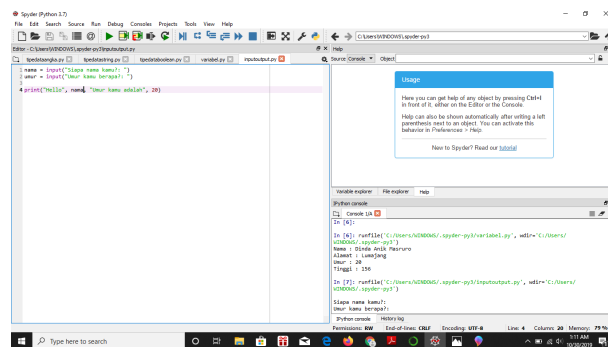
Input merupakan suatu masukan yang akan kita berikan ke program. Sedangkan hasil yang ditampilkan disebut output.

1. Cara mengambil input

Python menyediakan fungsi input, menggunakan kode input()

2. Cara menampilkan output

Untuk menampilkan sebuah output teks kita menggunakan kode print()



```
nama = input("Siapa nama kamu?: ")
umur = input("Umur kamu berapa?: ")

print("Hello", nama, "Umur kamu adalah", 20)
```

1.3 Operator Dasar Aritmatika, Tambah, Kali, Kurang, Bagi, dan Bagaimana Mengubah String ke Integer dan Integer Ke String

Operator merupakan suatu simbol-simbol yang digunakan untuk operasi tertentu. Dan operator aritmatika termasuk pada operator yang paling sering digunakan.

Operator	Simbol
Penjumlahan	+
Pengurangan	-
Perkalian	*
Pembagian	/
Sisa Bagi	%
Pemangkatan	**

Adapun contoh penggunaannya adalah:

```

1 a = int(input('Masukkan nilai a: '))
2 b = int(input('Masukkan nilai b: '))
3
4 # operator penjumlahan
5 c = a + b
6 print('Hasil %d + %d = %d' % (a,b,c))
7
8 # operator pengurangan
9 c = a - b
10 print('Hasil %d - %d = %d' % (a,b,c))
11
12 # operator perkalian
13 c = a * b
14 print('Hasil %d * %d = %d' % (a,b,c))
15
16 # operator pembagian
17 c = a / b
18 print('Hasil %d / %d = %d' % (a,b,c))

```

Console Output:

```

In [43]: runfile('C:/Users/acer/.spyder-py3/
operator.py', wdir='C:/Users/acer/.spyder-py3')

Masukkan nilai a: 5
Masukkan nilai b: 6
Hasil 5 + 6 = 11
Hasil 5 - 6 = -1
Hasil 5 * 6 = 30
Hasil 5 / 6 = 0

```

Cara mengubah string ke integer dan integer ke string

```

1 a=150
2 b=151
3 string = str(a)
4 print(string)
5 integer = int(b)
6 print(integer)
7

```

Console Output:

```

In [14]: runfile('C:/Users/acer/.spyder-py3/
konversint.py', wdir='C:/Users/acer/.spyder-py3')
150

In [15]: runfile('C:/Users/acer/.spyder-py3/
konversint.py', wdir='C:/Users/acer/.spyder-py3')
150

In [16]: runfile('C:/Users/acer/.spyder-py3/
konversint.py', wdir='C:/Users/acer/.spyder-py3')
150
151

In [17]:

```

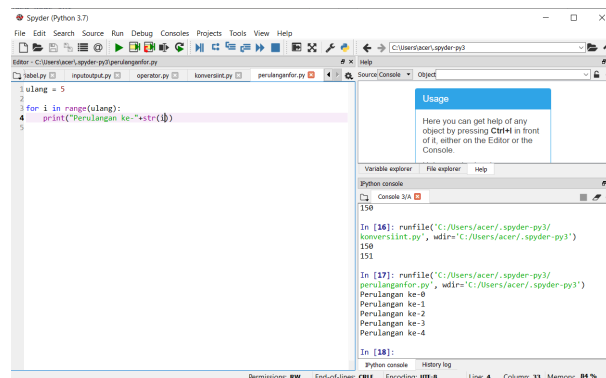
1.4 Penjelasan Syntax Untuk Perulangan dan Jenis-Jenisnya serta Contoh Kode dan Cara Menggunakannya di Python

Perulangan berfungsi untuk melakukan sesuatu secara berulang atau terus menerus. Pada bahasa pemrograman terdapat dua jenis perulangan, yaitu For dan While. Perulangan For disebut juga perulangan yang terhitung (counted loop) sedangkan perulangan while disebut perulangan yang tak terhitung (uncounted loop).

Secara umum python mengeksekusi program secara berbaris, tetapi untuk perulangan satu baris dieksekusi beberapa kali. Perulangan memerlukan tes kondisi, jika hasil tes true maka blok tersebut akan terus dieksekusi sedangkan jika false maka akan keluar dari blok perulangan dan mengeksekusi blok selanjutnya.

1. Perulangan For

Perulangan ini biasanya digunakan untuk mengetahui kode yang sudah banyak perulangannya. Adapun contoh kodenya adalah:



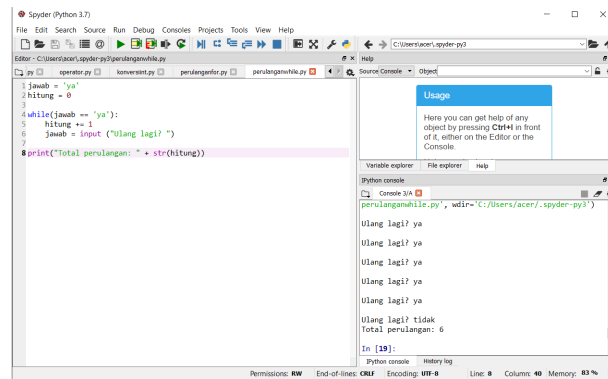
```
ulang = 5
```

```
for i in (ulang):
    print("Perulangan ke—" + str(i))
```

2. Perulangan While

Bila kondisi yang diuji salah, maka loop tidak akan pernah dieksekusi

```
umur = 19
```



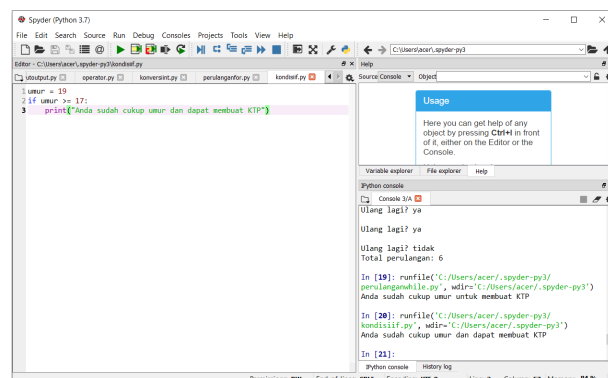
```
if umur > 17:
    print("Anda sudah cukup umur untuk membuat KTP")
```

1.5 Cara Menggunakan Syntax Untuk Memilih Kondisi dan Contoh Syntax Kondisi Di Dalam Kondisi

Python memiliki tiga jenis kondisional yang dapat digunakan untuk membangun suatu alur logika. Yaitu if, ifelse, dan ifelifelse.

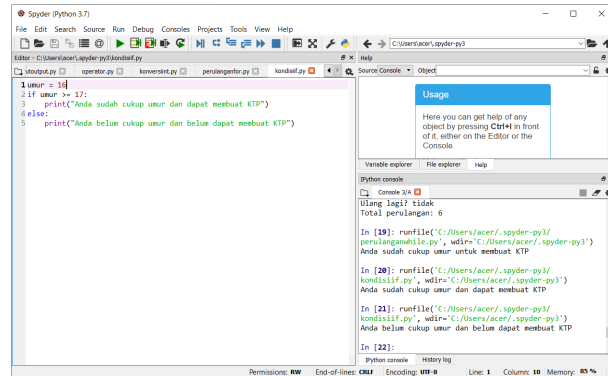
1. Kondisi If

Jika kondisi utama true, maka perintah akan dijalankan



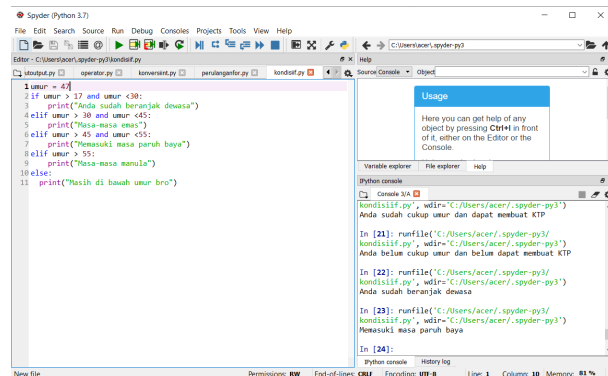
2. Kondisi If Else

Untuk memeriksa kondisi utama, else digunakan untuk menangani kondisi selain kondisi yang telah ditentukan.



3. Kondisi if elif else

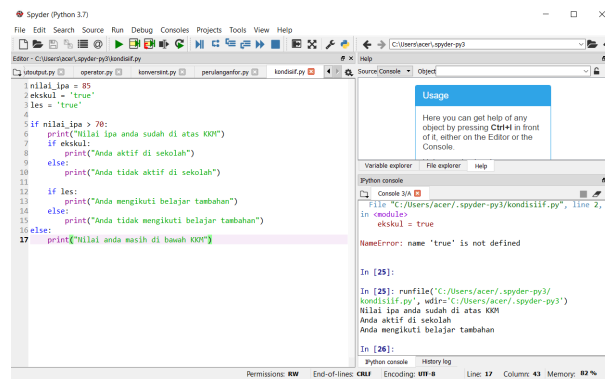
Bila anda akan mendefinisikan cukup banyak kondisi, maka gunakan elif di bawah statement if dan diatas statement else.



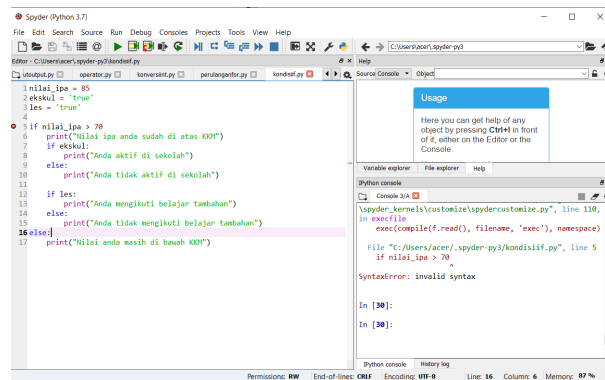
4. If di dalam if (If bersarang)

Suatu kondisional dapat dsimpan di dalam if lain, berikut adalah contohnya:

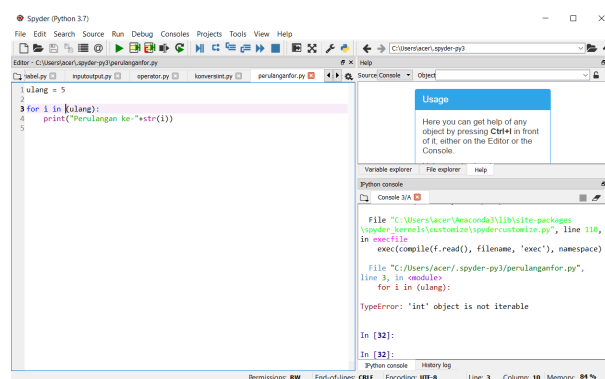
1.6 Jenis error yang sering ditemui di Python dalam mengerjakan Syntax di atas



Biasanya error terjadi dikarenakan ada kesalahan dalam pengetikan syntax, contohnya sebagai berikut:



Gambar di atas merupakan contoh dari kesalahan syntax titik dua pada line 5.

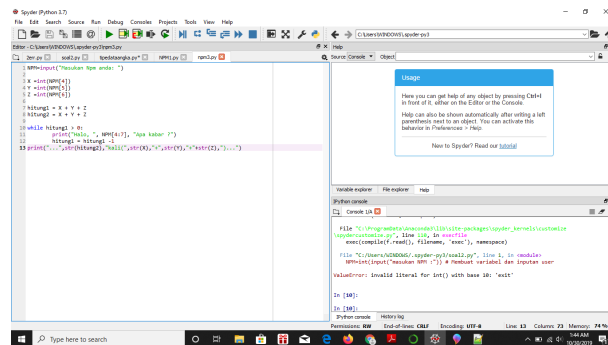


Gambar di atas merupakan contoh dari kesalahan syntax range, jika nilai bertipe data integer maka harus menggunakan range setelah in.

[illegible]


```
NPM=int(input("masukan NPM:")) # Membuat variabel dan inputan
Tld=NPM%100 #
for i in range(Tld): # pengulangan i dalam batas (range) untuk
    print("Halo ", NPM, " apa kabar?") #output
```

- ```
X =int (NPM[4])
```



**Y =int (NPM[ 5])**

**Z =int (NPM[ 6])**

**hitung1 = X + Y + Z**

**hitung2 = X + Y + Z**

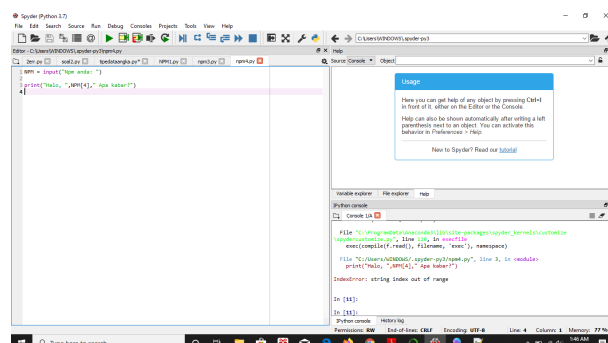
**while hitung1 > 0:**

**print ("Halo ,", NPM[4:7] , "Apa kabar?")**

**hitung1 = hitung1 -1**

**print (" ... ",str (hitung2) ," kali (",str (X) ,"+" ,str (Y) ,"+" +str (Z) ,"**

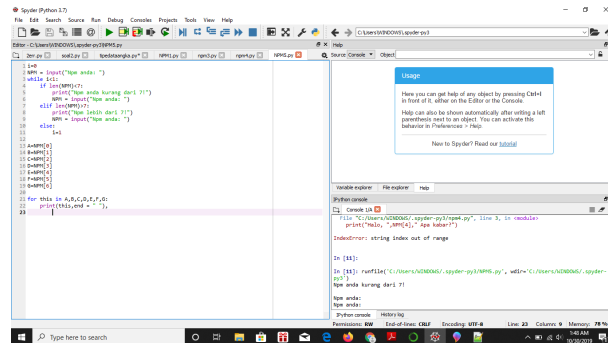
4. Program hello word dengan input nama yang disimpan dalam sebuah variabel string bernama **NPM** dan beri luaran output berupa digit ketiga dari belakang dari variabel NPM.



**NPM = input ("Npm\_anda: ")**

```
print ("Halo , _" ,NPM[4] , " _Apa_kabar?")
```

5. (untuk soal no 5 dan selanjutnya wajib menggunakan perulangan dan kondisi)  
buat program dengan mengisi variabel alfabet dengan nomor npm satu persatu  
berurut.



```
i=0
```

```
NPM = input ("Npm_anda : _")
```

```
while i < 1:
```

```
 if len (NPM) < 7:
```

```
 print ("Npm_anda_kurang_dari_7!")
```

```
 NPM = input ("Npm_anda : _")
```

```
 elif len (NPM) > 7:
```

```
 print ("Npm_lebih_dari_7!")
```

```
 NPM = input ("Npm_anda : _")
```

```
 else :
```

```
 i = 1
```

```
A=NPM[0]
```

```
B=NPM[1]
```

```
C=NPM[2]
```

```
D=NPM[3]
```

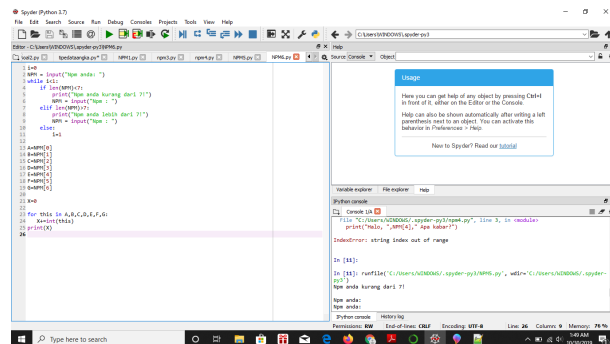
```
E=NPM[4]
```

```
F=NPM[5]
```

G=NPM[ 6 ]

```
for this in A,B,C,D,E,F,G:
 print(this ,end = " ") ,
```

6. Dari soal no 5, Lakukan penjumlahan dari seluruh variabel tersebut



i=0

NPM = input("Npm anda : ")

while i < 1:

if len(NPM) < 7:

print("Npm anda kurang dari 7!")

NPM = input("Npm : ")

elif len(NPM) > 7:

print("Npm anda lebih dari 7!")

NPM = input("Npm : ")

else :

i=1

A=NPM[ 0 ]

B=NPM[ 1 ]

C=NPM[ 2 ]

D=NPM[ 3 ]

E=NPM[ 4 ]

G=NPM[6]

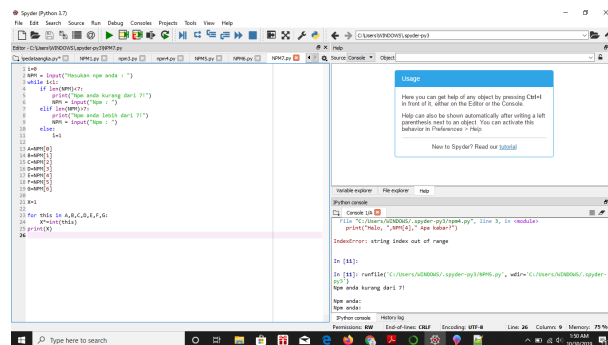
$$X=0$$

**for this in A,B,C,D,E,F,G:**

```
X+=int (this)
```

```
print(X)
```

7. Dari soal no 5, Lakukan perkalian dari seluruh variabel tersebut


$$i=0$$

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

```
while i < 1:
```

```
if len(NPM) < 7:
```

```
print("Npm_anda_kurang_dari_7!")
```

```
NPM = input("Npm: ")
```

```
elif len(NPM) > 7:
```

```
print("Npm_anda_lebih_dari_7!")
```

```
NPM = input("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 ]

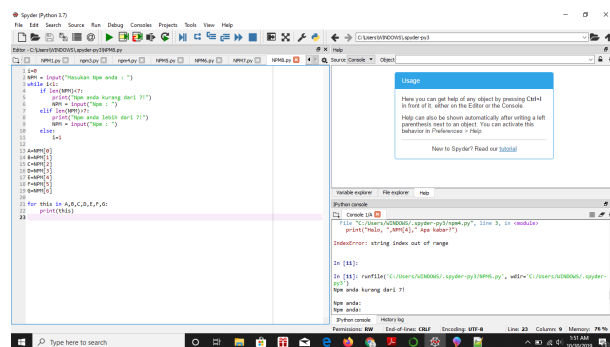
X=1

**for this in A,B,C,D,E,F,G:**

**X\*=int( this )**

**print (X)**

8. Dari soal no 5, Lakukan print secara vertikal dari NPM anda menggunakan variabel diatas.



i=0

NPM = **input( "Masukan \_Npm\_ anda : \_"**)

**while i < 1:**

**if len (NPM) < 7:**

**print ( "Npm\_anda \_kurang \_dari \_7!")**

**NPM = input( "Npm\_ : \_"**)

**elif len (NPM) > 7:**

**print ( "Npm\_anda \_lebih \_dari \_7!")**



```

 print("Npm anda lebih dari 7!")
 NPM = input("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]

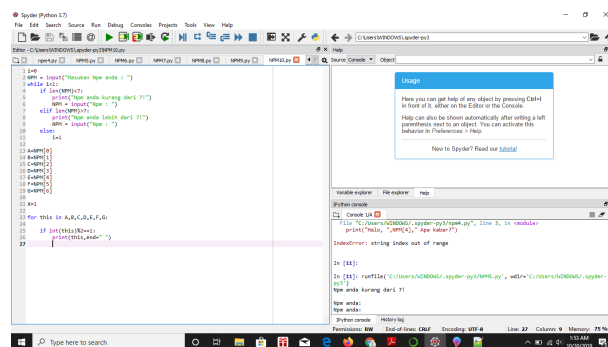
X=1

for this in A,B,C,D,E,F,G:

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

```

10. Dari soal no 5, Lakukan print NPM anda tapi hanya dijit ganjil saja.



```

1 i=0
2 NPM = input("Masukan Npm anda : ")
3 while i<len(NPM):
4 if len(NPM)%2==1:
5 print("Npm anda kurang dari 7!")
6 elif len(NPM)%2==0:
7 print("Npm anda lebih dari 7!")
8 NPM = input("Npm : ")
9 i+=1
10
11 A=NPM[0]
12 B=NPM[1]
13 C=NPM[2]
14 D=NPM[3]
15 E=NPM[4]
16 F=NPM[5]
17 G=NPM[6]
18
19 for this in A,B,C,D,E,F,G:
20 if int(this)%2==1:
21 print(this,end=" ")
22

```

```

i=0
NPM = input("Masukan Npm anda : ")
while i < 1:
 if len(NPM) < 7:
 print("Npm anda kurang dari 7!")
 NPM = input("Npm : ")
 elif len(NPM) > 7:
 print("Npm anda lebih dari 7!")
 NPM = input("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]

```

```

X=1

```

```

for this in A,B,C,D,E,F,G:

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

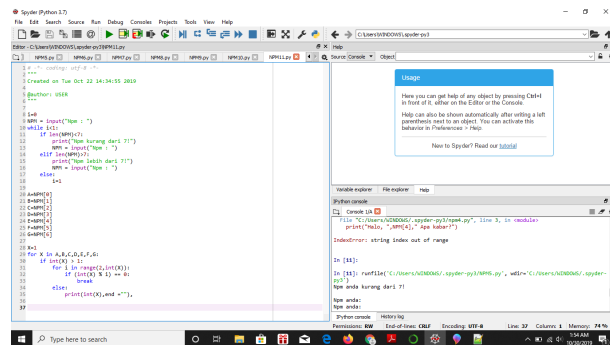
```

11. Dari soal no 5, Lakukan print NPM anda tapi hanya dijit yang termasuk bilangan prima saja.

```

-*- coding: utf-8 -*-

```



” ” ”

*Created on Tue Oct 22 14:34:55 2019*

*@author: USER*

” ” ”

i=0

**NPM = input("Npm: ")**

**while i<1:**

**if len(NPM)<7:**

**print("Npm kurang dari 7!")**

**NPM = input("Npm: ")**

**elif len(NPM)>7:**

**print("Npm lebih dari 7!")**

**NPM = input("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 ]

X=1

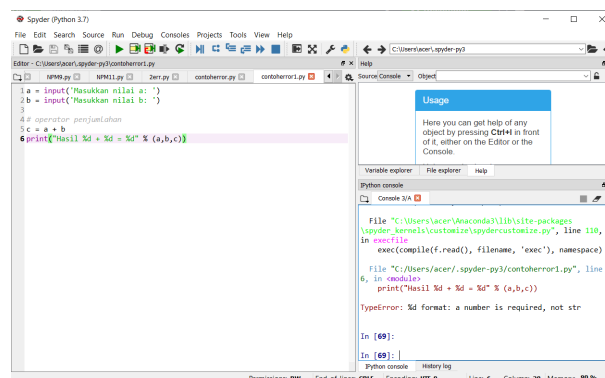
```
for X in A,B,C,D,E,F,G:
 if int(X) > 1:
 for i in range(2,int(X)):
 if (int(X) % i) == 0:
 break
 else :
 print (int(X) ,end ="") ,
```

### 3 Ketrampilan Penanganan Error

Bagian Penanganan error dari script python.

1. Tuliskan peringatan error yang didapat dari mengerjakan praktek kedua ini, dan jelaskan cara penanganan error tersebut.

Peringatan error yang didapat adalah seperti berikut, dijelaskan bahwa penggunaan persen d untuk integer dan bukan string. Tetapi tipe data yang dimasukkan atau diinputkan adalah string, maka terjadi error.



```

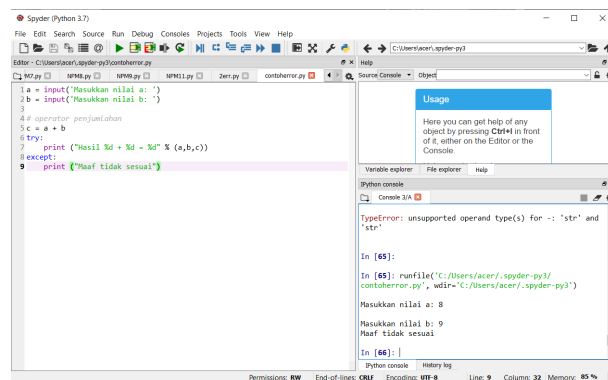
a = input('Masukkan nilai a: ')
b = input('Masukkan nilai b: ')

operator penjumlahan
c = a + b

print("Hasil %d + %d = %d" % (a, b, c))

```

Error akan diatasi oleh try except, dengan cara menangkap error lalu memprint peringatan error tersebut.



```

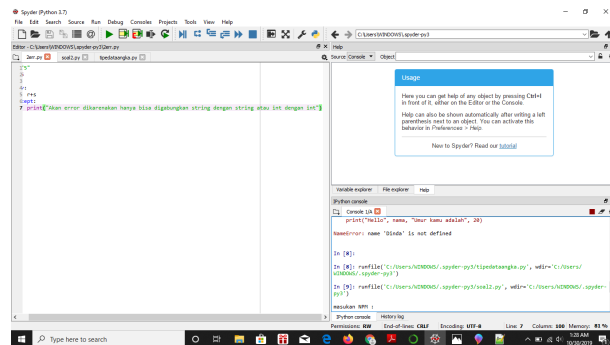
a = input('Masukkan nilai a: ')
b = input('Masukkan nilai b: ')

operator penjumlahan
c = a + b

try:
 print("Hasil %d + %d = %d" % (a, b, c))
except:
 print("Maaf tidak sesuai")

```

2. Membuat file 2err.py dan mengisinya dengan script pengisian variabel sebagai string dan pengisian variabel sebagai interger. Kemudian jumlahkan antara variabel integer dan string dan tangkap jenis errornya, gunakan try except untuk menunjukkan error tersebut dengan bahasa indonesia.



**r="5"**

**s=5**

**try :**

**r+s**

**except :**

**print("Akan\_error\_dikarenakan\_hanya\_bisa\_digabungkan\_string**