

**RESUME PYTHON
CHAPTER 3**

Diajukan untuk memenuhi kelulusan matakuliah
Pemrograman II pada Program Studi DIV Teknik
Informatika

O l e h :

Gharrieb Nouvaldi
1.18.4.082



**PROGRAM DIPLOMA IV TEKNIK INFORMATIKA
POLITEKNIK POS INDONESIA**

**BANDUNG
2019**

Teori

1. Fungsi merupakan blok program pada Python, dibuat dengan menggunakan kata kunci `def` kemudian diikuti dengan nama fungsinya. Sama seperti blok kode yang lain, kita harus memberikan identasi (tab atau spasi 2x) untuk menuliskan isi fungsi.

Contoh pengaplikasian dari fungsi:

```
#pembuatan fungsi
def Salam(): print "Hay, Selamat Pagi"
#pemanggilan fungsi
salam()
#output: Selamat Pagi
```

2. Package adalah folder yang menyimpan source code dan dapat di import kedalam program

#Cara memanggilnya : `from nama_package import library`.

3. a). Class adalah blue print untuk mendefinisikan suatu objek. Contoh kode program:

```
Class NPM :
def __init__(self, NPM):
self.NPM = NPM
def helo NPM (self):
print ("Helo", NPM)
```

- b). Objek adalah hasil dari sebuah class. Contohnya :

```
Import kelas2B
Cobakelas=kelas2B.aku(npm)
Hasil=Cobakelas.npm2()
```

- c). Atribut merupakan Fungsi-fungsi yang dimiliki oleh class. Contoh penggunaannya:

Class name:

```
def __init__(self,nama):
```

```
    self.nama = nama
```

d). Method adalah Fungsi yang ada pada objek atau instansi class. Contoh penggunaannya:

Class name:

```
def __init__(self,nama): self.nama = nama
```

```
def nama(self)
```

```
    print("hallo", name)
```

4. Contoh penggunaan sebuah library : pada folder Gharrieb

```
def nama () :
```

```
    print ("Gharrieb")
```

Contoh jika ingin memanggil fungsi dari library, maka kita harus import terlebih dahulu:

```
Import Gharrieb
```

```
Gharrieb.nama ()
```

5. Perintah form kalkulator penambahan.

Maksudnya dari from kalkulator import penambahan yaitu, kode tersebut akan memanggil kalkulator dan mengimport penambahan.

Contohnya : from Mahasiswa import NPM

6. Cara memanggil library dalam folder.

Langkah pertama kita harus menuliskan foldernya terlebih dahulu lalu melakukan import nama librarynya.

Contohnya: from Mahasiswa import NPM

7. Cara memanggil class di dalam folder

Cara melakukan pemanggilannya dengan cara memanggil foldernya terlebih dahulu baru setelah itu library yang kita inginkan. Contohnya:

```
from Mahasiswa import NPM
```

nama kelas yang akan digunakan adalah NPM

Keterampilan Pemrograman

```
1. # -*- coding: utf-8 -*-  
"""
```

```
Created on Tue Nov 26 22:30:33 2019
```

```
@author: Azure  
"""
```

```
def soal1():
```

```
    print("###_###_#####_###_###_#####_#####  
    print("###_###_###_###_###_###_###_###_###_###_###  
    print("###_###_#####_#####_###_###_#####  
    print("###_###_###_###_#####_###_###_###_###_###  
    print("###_###_#####_#####_###_###_#####
```

```
# -*- coding: utf-8 -*-  
"""
```

```
Created on Tue Nov 26 22:33:00 2019
```

```
@author: Azure  
"""
```

```
def soal2(npm):
```

```
    npm=int(npm)  
    TwoLastDigit=abs(npm)%100  
    for i in range(TwoLastDigit):  
        print("Halo , ", npm, " _apa_kabar_?")
```

```
2. # -*- coding: utf-8 -*-  
"""
```

```
Created on Tue Nov 26 22:35:04 2019
```

```
@author: Azure  
"""
```

```
def soal3(npm):
```

```

        for i in range(int(str(npm)[4])+int(str(npm)[5])+int(str(npm)[6])):
            print("Halo , "+str(npm)[4]+str(npm)[5]+str(npm)[6]+"_apa_kabar_")

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
soal3(npm)

# -*- coding: utf-8 -*-
"""
Created on Tue Nov 26 22:35:07 2019

@author: Azure
"""

def soal4(npm):
    key=npm%1000
    str_key=str(key)
    print("Halo , "+str_key[0]+"_apa_kabar_?")

```

```

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
soal4(npm)

4. # -*- coding: utf-8 -*-
"""
Created on Tue Nov 26 22:51:29 2019

```

```
@author: Azure
"""
```

```
def soal5(npm):
    A=np[0]
    B=np[1]
    C=np[2]
    D=np[3]
    E=np[4]
    F=np[5]
    G=np[6]

    for x in A,B,C,D,E,F,G:
        print(x)

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
soal5(npm)
```

```
# -*- coding: utf-8 -*-
"""
```

Created on Tue Nov 26 22:51:33 2019

```
@author: Azure
"""
```

```
def soal6(npm):

    A=np[0]
    B=np[1]
    C=np[2]
    D=np[3]
    E=np[4]
    F=np[5]
    G=np[6]
    y=0
```

```

        for x in A,B,C,D,E,F,G:
            y+=int(x)
        print(y)

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
soal6(npm)

0. # -*- coding: utf-8 -*-
"""
Created on Tue Nov 26 22:51:35 2019

@author: Azure
"""

def soal7(npm):

    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)

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
soal7(npm)

```

```

# -*- coding: utf-8 -*-
"""

```

Created on Tue Nov 26 22:54:16 2019

```

@author: Azure
"""

```

```

def soal8(npm):
    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:
        if int(x)%2==0:
            if int(x)==0:
                x=""
            print(x,end=" ")

```

```

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
soal8(npm)

```

```

9. # -*- coding: utf-8 -*-
"""

```


Created on Tue Nov 26 22:54:17 2019

@author: Azure
"""

```
def soal9(npm):
    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:

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

    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
    soal9(npm)
```

-- coding: utf-8 -*-*
"""

Created on Tue Nov 26 22:54:17 2019

@author: Azure
"""

```
def soal10(npm):
    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:
    if int(x) > 1:
        for i in range(2,int(x)):
            if (int(x) % i) == 0:
                break
        else:
            print(int(x),end =""),

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
soal10(npm)

```

```

10. # -*- coding: utf-8 -*-
"""

```

Created on Tue Nov 26 23:01:58 2019

```

@author: Azure
"""

```

#soal1

```

def soal1():
    print("###_###_#####_###_###_#####_#####_")
    print("###_###_###_###_###_###_###_###_#####_###_###_###_")
    print("###_###_#####_#####_###_###_#####_")
    print("###_###_###_###_#####_###_###_###_###_###_")
    print("###_###_#####_#####_###_#####_#####_")

```

#soal2

```

def soal2(npm):
    npm=int(npm)
    TwoLastDigit=abs(npm)%100
    for i in range(TwoLastDigit):
        print("Halo, ", npm, " apa kabar?")

```

#soal3

```

def soal3(npm):
    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])):
        print("Halo , "+str(npm)[4]+str(npm)[5]+str(npm)[6]+" apa kabar")

```

#soal4

```

def soal4(npm):
    npm=input("Masukan NPM: ")
    key=npm%1000
    str_key=str(key)
    print("Halo , "+str_key[0]+" apa kabar?")

```

#soal5

```

def soal5(npm):
    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)

```

#soal6

```

def soal6(npm):
    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]
y=0

for x in A,B,C,D,E,F,G:
    y+=int(x)
print(y)

#soal7
def soal6(npm):
    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)

#soal8
def soal8(npm):
    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 = "")

#soal9
def soal9(npm):
    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 = "")

#soal10
def soal10(npm):
    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 = ""),

```

```

# -*- coding: utf-8 -*-
"""
Created on Tue Nov 26 23:00:02 2019

@author: Azure
"""

import lib3
class Kelas3lib:
    def __init__(self, npm):
        self.npm = npm
    def soal1(self):
        return lib3.soal1()
    def soal2(self):
        return lib3.soal2(self.npm)
    def soal33(self):
        return lib3.soal3(self.npm)
    def soal4(self):
        return lib3.soal4(self.npm)
    def soal5(self):
        return lib3.soal5(self.npm)
    def soal6(self):
        return lib3.soal6(self.npm)
    def soal7(self):
        return lib3.soal7(self.npm)
    def soal8(self):
        return lib3.soal8(self.npm)
    def soal9(self):
        return lib3.soal9(self.npm)
    def soal10(self):
        return lib3.soal10(self.npm)

```

Keterampilan penanganan error

Penanganan error error:

Tipe error: `--init--` missing 1 required positional argument: "npm"

Penyelesaian:

Menambahkan para meter.

Try except

```
def pembagian(a,b):  
    c=a/b  
    return c  
  
d=int(input("angka pertama : "))  
e=int(input("angka kedua : "))  
try:  
    print(pembagian(d,e))  
except:  
    print("jangan masukan angka 0")
```