

# **CERDAS MENGUASAI PYTHON**



---

# CERDAS MENGUASAI PYTHON

## Dalam 24 Jam

---

**Rolly M. Awangga**  
Informatics Research Center



**Kreatif Industri Nusantara**

***Penulis:***

Rolly Maulana Awangga

ISBN : 978-602-53897-0-2

***Editor:***

M. Yusril Helmi Setyawan

***Penyunting:***

Syafrial Fachrie Pane

Khaera Tunnisa

Diana Asri Wijayanti

***Desain sampul dan Tata letak:***

Deza Martha Akbar

***Penerbit:***

Kreatif Industri Nusantara

***Redaksi:***

Jl. Ligar Nyawang No. 2

Bandung 40191

Tel. 022 2045-8529

Email : awangga@kreatif.co.id

***Distributor:***

Informatics Research Center

Jl. Sariasih No. 54

Bandung 40151

Email : irc@poltekpos.ac.id

Cetakan Pertama, 2019

Hak cipta dilindungi undang-undang

Dilarang memperbanyak karya tulis ini dalam bentuk dan dengan cara  
apapun tanpa ijin tertulis dari penerbit

*‘Jika Kamu tidak dapat  
menahan lelahnya  
belajar, Maka kamu harus  
sanggup menahan  
perihnya Kebodohan.’  
Imam Syafi’i*

# CONTRIBUTORS

---

ROLLY MAULANA AWANGGA, Informatics Research Center., Politeknik Pos Indonesia, Bandung, Indonesia



# CONTENTS IN BRIEF

---

<b>1</b>	<b>SEJARAH DAN KARAKTERISTIK PYTHON</b>	<b>1</b>
<b>2</b>	<b>Judul Bagian Kedua</b>	<b>3</b>
<b>3</b>	<b>Judul Bagian Ketiga</b>	<b>5</b>





# DAFTAR ISI

---

Daftar Gambar	xi
Daftar Tabel	xiii
Foreword	xvii
Kata Pengantar	xix
Acknowledgments	xxi
Acronyms	xxiii
Glossary	xxv
List of Symbols	xxvii
Introduction	xxix
<i>Rolly Maulana Awangga, S.T., M.T.</i>	
<b>1 SEJARAH DAN KARAKTERISTIK PYTHON</b>	<b>1</b>
<b>2 Judul Bagian Kedua</b>	<b>3</b>
<b>3 Judul Bagian Ketiga</b>	<b>5</b>
	ix

3.1	D. Irga B. Naufal Fakhri	5
3.1.1	Pemahaman Teori	5
3.1.2	Keterampilan Pemograman	8
3.1.3	Ketrampilan Penanganan Error	11
3.2	Nurul Izza Hamka   1174062   Teori	11

# DAFTAR GAMBAR

---



# DAFTAR TABEL

---



# Listings

---

3.1	Contoh kode fungsi inputan ke fungsi.	5
3.2	Contoh kode fungsi outputan ke fungsi.	5
3.3	Contoh import package atau library.	6
3.4	Contoh import package atau library.	6
3.5	Contoh package atau library.	7
3.6	Contoh import package atau library.	7
3.7	Contoh import package atau library.	7
3.8	Contoh import package atau library didalam folder.	7
3.9	Contoh import package atau library didalam folder.	7
	src/1174066.py	8
	src/1174066.py	8
	src/1174066.py	8
	src/1174066.py	9
	src/1174066.py	9
	src/1174066.py	9
	src/1174066.py	9
	src/1174066.py	9



src/1174066.py	10
src/main_1174066.py	10
src/main_1174066.py	10
src/main_1174066.py	11
src/1174062.py	12
src/kelas3lib.py	12
src/1174062.py	12
src/1174062.py	13
src/1174062.py	13
src/1174062.py	13
src/1174062.py	13
src/1174062.py	14
src/1174062.py	15
src/1174062.py	15
src/1174062.py	15
src/1174062.py	15
src/1174062.py	16
src/1174062.py	16
src/1174062.py	16
src/1174062.py	16

# FOREWORD

---

Sepatah kata dari Kaprodi, Kabag Kemahasiswaan dan Mahasiswa



# KATA PENGANTAR

---

Buku ini diciptakan bagi yang awam dengan git sekalipun.

R. M. AWANGGA

*Bandung, Jawa Barat*  
*Februari, 2019*



# ACKNOWLEDGMENTS

---

Terima kasih atas semua masukan dari para mahasiswa agar bisa membuat buku ini lebih baik dan lebih mudah dimengerti.

Terima kasih ini juga ditujukan khusus untuk team IRC yang telah fokus untuk belajar dan memahami bagaimana buku ini mendampingi proses Intership.

R. M. A.



# ACRONYMS

---

ACGIH	American Conference of Governmental Industrial Hygienists
AEC	Atomic Energy Commission
OSHA	Occupational Health and Safety Commission
SAMA	Scientific Apparatus Makers Association





# GLOSSARY

---

git	Merupakan manajemen sumber kode yang dibuat oleh linus torvald.
bash	Merupakan bahasa sistem operasi berbasiskan *NIX.
linux	Sistem operasi berbasis sumber kode terbuka yang dibuat oleh Linus Torvald



# SYMBOLS

---

- $A$  Amplitude
- $\&$  Propositional logic symbol
- $a$  Filter Coefficient
  
- $\mathcal{B}$  Number of Beats



# INTRODUCTION

---

ROLLY MAULANA AWANGGA, S.T., M.T.

Informatics Research Center  
Bandung, Jawa Barat, Indonesia

Pada era disruptif saat ini. git merupakan sebuah kebutuhan dalam sebuah organisasi pengembangan perangkat lunak. Buku ini diharapkan bisa menjadi penghantar para programmer, analis, IT Operation dan Project Manajer. Dalam melakukan implementasi git pada diri dan organisasinya.

Rumusnya cuman sebagai contoh aja biar keren[?].

$$ABCDEF\alpha\beta\Gamma\Delta\sum_{def}^{abc} \tag{I.1}$$



# BAB 1

---

## SEJARAH DAN KARAKTERISTIK PYTHON

---





## BAB 2

---

## JUDUL BAGIAN KEDUA

---



## BAB 3

---

# JUDUL BAGIAN KETIGA

---

### 3.1 D. Irga B. Naufal Fakhri

#### 3.1.1 Pemahaman Teori

##### 1. Fungsi

Fungsi adalah blok blok kode yang terorganisir yang dapat digunakan kembali didalam program yang digunakan untuk melakukan suatu perintah yang telah diberikan. untuk membuat fungsi kita harus menggunakan def kemudian nama fungsinya dan (variable)nya diakhiri oleh tanda :

```
1 #Contoh fungsi inputan
2 def printteks( str ):
3     print ( str )
4     return
5
6 printteks("Ini adalah Contoh fungsi")
```

**Listing 3.1** Contoh kode fungsi inputan ke fungsi.

Fungsi juga berguna untuk melemparkan variable contohnya

```

1 #Contoh fungsi outputan
2 def output( str ):
3     full = "Ini adalah contoh dari " + str
4     return full
5
6 print( output("fungsi outputan"))

```

**Listing 3.2** Contoh kode fungsi outputan ke fungsi.

## 2. Paket(Package) atau Library

Paket atau yang biasa disebut dengan library adalah kumpulan kode-kode fungsi atau method pada python yang dapat dipanggil kedalam program python yang kita buat. Package berada di file terpisah dari main program cara memanggil package: Pastikan file package ada didalam folder yang sama lalu ditambah import dengan nama filenya tanpa extensi (.py)

```

1 #Contoh import library atau package
2 import fungsi_1174066
3
4 input( fungsi_1174066.hello("Dirga"))

```

**Listing 3.3** Contoh import package atau library.

## 3. Kelas (Class), Objek (Object), Atribut (Attribute), dan Method

Kelas(Class) adalah sebuah blueprint(cetakan) dari sebuah objek. Objek(Object) adalah hasil cetakan dari sebuah kelas(class). Atribut(Attribute) adalah nilai data yang ada didalam sebuah object. Method adalah sesuatu yang bisa dilakukan oleh object.

```

1 #Contoh Class
2 class Mahasiswa:
3     def __init__( self , nama ,npm, kelas ) :
4         self.nama = nama
5         self.npm = npm
6         self.kelas = kelas
7
8     def datadiri( self ):
9         print ( "Nama: ", self.nama, ", NPM: ", self.npm, ",
10             Kelas: ", self.kelas )
11
12 #Pembuatan Object
13 mhs1 = Mahasiswa("D. Irga",11174066,"D4TI2C")
14 mhs1.datadiri() ;

```

**Listing 3.4** Contoh import package atau library.

## 4. Cara memanggil library dari instansiasi

Cara memanggilnya:

- Pertama kita import filenya
- kemudian buat variabelnya jika menggunakan variable untuk menampung data

- Kemudian panggil nama classnya(file) dan panggil fungsinya
- Kemudian menggunakan perintah print untuk menampilkan data

```

1 #Contoh package nomor 4
2 def Tambah(x,y):
3     z = x + y
4     return z

```

**Listing 3.5** Contoh package atau library.

```

1 #Contoh cara memanggil library atau package
2 import fungsi_1174066
3
4 x = 3
5 y = 7
6 print(fungsi_1174066.Tambah(x,y))

```

**Listing 3.6** Contoh import package atau library.

5. Contoh pemakaian paket dengan perintah from kalkulator import Penambahan  
Pemakaian package(paket) dengan perintah from namafilenya import berfungsi untuk memanggil fungsi dari nama filenya

```

1 #Contoh cara memanggil library atau package dari from import
2 from fungsi_1174066 import Tambah
3
4 x = 3
5 y = 7
6 print(Tambah(x,y))

```

**Listing 3.7** Contoh import package atau library.

6. Jelaskan dengan contoh kode, pemakaian paket fungsi didalam folder  
Jika file paket ada didalam folder maka kita harus menambahkan lokasi filenya ada didalam folder apa dengan cara menggunakan namafolder.namafile

```

1 #Contoh cara memanggil library atau package dari folder
2 from 1174066.fungsi_1174066 import Tambah
3
4 x = 3
5 y = 7
6 print(Tambah(x,y))

```

**Listing 3.8** Contoh import package atau library didalam folder.

7. Jelaskan dengan contoh kode, pemakaian paket fungsi didalam folder  
Jika file paket ada didalam folder maka kita harus menambahkan lokasi filenya ada didalam folder apa dengan cara menggunakan namafolder.namafile

```

1 #Contoh cara memanggil library atau package dari folder
2 from 1174066.fungsi_1174066 import Tambah
3
4 x = 3

```

```

5 y = 7
6 print(Tambah(x,y))

```

**Listing 3.9** Contoh import package atau library didalam folder.

### 3.1.2 Keterampilan Pemograman

#### 1. Jawaban nomor 1

```

1 #Jawaban nomor 1
2 def npm(npm):
3     text = str(npm)
4     from PIL import Image, ImageDraw, ImageFont
5     import numpy as np
6     myfont = ImageFont.truetype("verdanab.ttf", 12)
7     size = myfont.getsize(text)
8     img = Image.new("1",size,"black")
9     draw = ImageDraw.Draw(img)
10    draw.text((0, 0), text, "white", font=myfont)
11    pixels = np.array(img, dtype=np.uint8)
12    if int(text) % 3 == 0:
13        chars = np.array([' ', '*'], dtype="U1")[pixels]
14    elif int(text) % 3 == 1:
15        chars = np.array([' ', '#'], dtype="U1")[pixels]
16    elif int(text) % 3 == 2:
17        chars = np.array([' ', '+'], dtype="U1")[pixels]
18    strings = chars.view('U' + str(chars.shape[1])).flatten()
19    print( "\n".join( strings ))

```

#### 2. Jawaban nomor 2

```

1 #Jawaban nomor 2
2 def loop(npm):
3     i = 1
4     while(i <= int(npm[5:])):
5         print("Hallo, "+str(npm)+" apa kabar?")
6         i += 1
7
8 loop(input("Masukan NPM : "))

```

#### 3. Jawaban nomor 3

```

1 #Jawaban nomor 3
2 def looptambah(npm):
3     i = 1
4     akhir = int(npm[4:5]) + int(npm[5:6]) + int(npm[6:7])
5     while(i <= akhir):
6         print("Hallo, "+str(npm[4:7])+" apa kabar?")
7         i += 1
8
9 looptambah(input("Input: "))

```

#### 4. Jawaban nomor 4

```
1 #Jawaban nomor 4
2 def hallo(npm):
3     print("Halo, "+str(npm[-3])+ " apa kabar?")
4
5 hallo(input("Input: "))
```

## 5. Jawaban nomor 5

```
1 #Jawaban nomor 5
2 def array(npm):
3     for x in npm:
4         print(x)
```

## 6. Jawaban nomor 6

```
1 #Jawaban nomor 6
2 def jumlah(npm):
3     npm = list(map(int, npm))
4     hasil = 0
5     for x in npm:
6         hasil += x
7     print("Hasil perjumlahan dari npm anda adalah: " + str(hasil))
8
9 jumlah(input("NPM: "))
```

## 7. Jawaban nomor 7

```
1 #Jawaban nomor 7
2 def kali(npm):
3     npm = list(map(int, npm))
4     hasil = 0
5     for x in npm:
6         hasil *= x
7     print("Hasil perkalian dari npm anda adalah: " + str(hasil))
8
9 kali(input("NPM: "))
```

## 8. Jawaban nomor 8

```
1 #Jawaban nomor 8
2 def printgenap(npm):
3     npm = list(map(int, npm))
4     for x in npm:
5         if(x % 2 == 0):
6             if(x != 0):
7                 print(x, end = " ")
8
9 printgenap(input("NPM: "))
```

## 9. Jawaban nomor 9

```
1 #Jawaban nomor 9
2 def printganjil(npm):
3     npm = list(map(int, npm))
```



```

4     for x in npm:
5         if(x % 2 != 0):
6             print(x, end = "")
7
8     printganjil(input("NPM: "))

```

## 10. Jawaban nomor 10

```

1  #Jawaban nomor 10
2  def printprima(npm):
3      npm = list(map(int, npm))
4      prima = []
5      for n in npm:
6          bilPrima = True
7          if n == 0 or n == 1:
8              bilPrima = False
9          for x in range(2, n):
10             if n % x == 0:
11                 bilPrima = False
12             if bilPrima:
13                 prima.append(n)
14
15     for p in prima:
16         print(p, end = "")
17
18     printprima(input("NPM: "))

```

## 11. Jawaban nomor 11

```

1  #Pemanggilan library 3lib
2  from lib_1174066 import *
3
4  npm(1174066)
5  loop(input("Masukan NPM : "))
6  looptambah(input("Input: "))
7  hallo(input("Input: "))
8  array(input("NPM: "))
9  jumlah(input("NPM: "))
10 kali(input("NPM: "))
11 printgenap(input("NPM: "))
12 printganjil(input("NPM: "))
13 printprima(input("NPM: "))

```

## 12. Jawaban nomor 12

```

1  #Pemanggilan kelas 3lib
2  from kelas3lib_1174066 import kelas3lib
3  npm = "1174066"
4  kelas = kelas3lib(npm)
5
6  kelas.npmprint()
7  kelas.loop()
8  kelas.looptambah()
9  kelas.hallo()
10 kelas.array()
11 kelas.jumlah()

```

```

12 kelas.kali()
13 kelas.printgenap()
14 kelas.printganjil()
15 kelas.printprima()

```

### 3.1.3 Ketrampilan Penanganan Error

- Syntax Errors

Syntax Errors adalah kesalahan pada penulisan syntax atau kode. Solusinya adalah memperbaiki penulisan syntax atau kode

- Zero Division Error

ZeroDivisonError adalah exceptions yang terjadi saat eksekusi program menghasilkan perhitungan matematika pembagian dengan angka nol (0). Solusinya adalah tidak membagi suatu yang hasilnya nol.

- Name Error

NameError adalah exception saat kode melakukan eksekusi terhadap local name atau global name yang tidak terdefinisi atau tidak ada. Solusinya adalah memastikan variabel atau function yang akan dipanggil ada didalam program atau tidak salah mengetikannya.

- Type Error

TypeError adalah exception saat melakukan eksekusi terhadap suatu operasi atau fungsi dengan type object yang tidak sesuai. Solusinya adalah mengkoversi variabelnya sesuai dengan tipe data sesuai dengan yang akan digunakan.

```

1 #Pemanggilan kelas 3lib
2 from kelas3lib_1174066 import kelas3lib
3 npm = "1174066"
4 kelas = kelas3lib(npm)
5
6 kelas.npmprint()
7 kelas.loop()
8 kelas.looptambah()
9 kelas.hallo()
10 kelas.array()
11 kelas.jumlah()
12 kelas.kali()
13 kelas.printgenap()
14 kelas.printganjil()
15 kelas.printprima()

```

## 3.2 Nurul Izza Hamka | 1174062 | Teori

1. Fungsi pada python menggunakan kata kunci def. setelah menulis kata def, kita tulis lagi nama fungsi kemudian diikuti dengan parameter yang diberi tanda

kurung dan diakhirnya diberi tanda titi dua (:). Setelah itu kita menulis lagi fungsi yang akan di panggil untuk di jalankan. Inputan fungsi adalah untuk memanggil fungsi dengan menuliskan nama fungsi. Kembalian fungsi adalah keluaran fungsi dan untuk kembali ke baris selanjutnya untuk memanggil fungsi tadi.

```
1 def Izzah() :
2     print ("Apa Kabar Izzah")
3 Izzah()
```

2. Paket adalah sebuah file yang berisi kode program python yang bisa digunakan berulang ketika sebuah paket itu dipanggil.

```
1 #from kelas3lib import *
2 #print(penulisan(int(input("Masukan NPM kamu : "))))
```

3. Apa itu kelas, apa itu objek, apa itu atribut, apa itu method :

- Kelas adalah sebuah objek yang di dalamnya terdapat sebuah metode atau seperangkat atribut.
- Objek adalah struktur data yang di definisikan dalam kelas, objek ini memiliki atribut dan juga aksi /behaviour.
- Atribut adalah data dari variable kelas dan juga method.
- Method adalah sebuah kode yang di gunakan untuk melakukan perintah.

```
1 #class Employee:
2 #     'Common base class for all employees'
3 #     empCount = 0
4
5 #     def __init__(self, name, salary):
6 #         self.name = name
7 #         self.salary = salary
8 #         Employee.empCount += 1
9
10 #     def displayCount(self):
11 #         print ("Total Employee %d" % Employee.empCount)
12
13 #     def displayEmployee(self):
14 #         print ("Name : ", self.name, ", Salary : ", self.salary)
15
16
17 #This would create first object of Employee class"
18 #emp1 = Employee("Izza", 2000)
19 #This would create second object of Employee class"
20 #emp2 = Employee("Manni", 5000)
21 #emp1.displayEmployee()
22 #emp2.displayEmployee()
23 #print ("Total Employee %d" % Employee.empCount)
```

#### 4. Pemanggilan library kelas dari instansiasi

- Pertama, lakukan import file.
- Membuat variable untuk menyimpan data.
- Lakukan pemanggilan data dan class.
- Ketikkan print untuk melihat hasilnya.

```

1 #import belajar
2 #a = 10
3 #b = 5
4
5 #c = belajar.penambahan(a,b)
6 #print(c)

```

#### 5. Penggunaan paket from kalkulator import untuk penambahan berfungsi untuk memanggil file yang di masukkan dan juga fungsinya.

```

1 #import belajar
2 #a = 10
3 #b = 5
4
5 #c = belajar.penambahan(a,b)
6 #print(c)

```

#### 6. Pemakaian paket fungsi adalah sebuah kumpulan fungsi-fungsi.

#### 7. Pemakaian paket kelas apabila file library ada di dalam folder.

```

1 # Employee.empCount += 1

```

#### 8. No1

```

1 #No 1
2 def penulisan(npm):
3     npm = list(str(npm))
4
5     angka1 = {"0": " ##### ", "1": " ### ", "2": " ##### ", "3": "
##### ", "4": " ###   ## ", "5": " ##### ", "6": " ##### ", "7": "
##### ", "8": " ##### ", "9": " ##### "}
6     angka2 = {"0": " ##### ", "1": " ### ", "2": " ##### ", "3": "
##### ", "4": " ###   ## ", "5": " ##### ", "6": " ##### ", "7": "
##### ", "8": " ##### ", "9": " ##### "}
7     angka3 = {"0": " ##   ## ", "1": " ### ", "2": "   ## ", "3": "
##### ", "4": " ###   ## ", "5": " ##### ", "6": " ##### ", "7": "
   ##   ", "8": " ##### ", "9": " ##### "}
8     angka4 = {"0": " ##   ## ", "1": " ### ", "2": "   ## ", "3": "
##### ", "4": " ###   ## ", "5": " ##### ", "6": " ##### ", "7": "
   ##   ", "8": " ##### ", "9": " ##### "}

```

```

9     angka5 = {"0": " ##      ## ", "1": " ### ", "2": " ##### ", "3": "
##### ", "4": " ##### ", "5": " ##### ", "6": " ##### ", "7": "
##### ", "8": " ##### ", "9": " ##### "}
10    angka6 = {"0": " ##      ## ", "1": " ### ", "2": " ##### ", "3": "
##### ", "4": " ##### ", "5": " ##### ", "6": " ##### ", "7": "
##### ", "8": " ##### ", "9": " ##### "}
11    angka7 = {"0": " ##      ## ", "1": " ### ", "2": " ##### ", "3": "
##### ", "4": " ##### ", "5": " ##### ", "6": " ##### ", "7": "
##### ", "8": " ##### ", "9": " ##### "}
12    angka8 = {"0": " ##      ## ", "1": " ### ", "2": " ##### ", "3": "
##### ", "4": " ##### ", "5": " ##### ", "6": " ##### ", "7": "
##### ", "8": " ##### ", "9": " ##### "}
13    angka9 = {"0": " ##### ", "1": " ### ", "2": " ##### ", "3": "
##### ", "4": " ##### ", "5": " ##### ", "6": " ##### ", "7": "
##### ", "8": " ##### ", "9": " ##### "}
14    angka10 = {"0": " ##### ", "1": " ### ", "2": " ##### ", "3": "
##### ", "4": " ##### ", "5": " ##### ", "6": " ##### ", "7": "
##### ", "8": " ##### ", "9": " ##### "}

15
16    hasil1 = []
17    hasil2 = []
18    hasil3 = []
19    hasil4 = []
20    hasil5 = []
21    hasil6 = []
22    hasil7 = []
23    hasil8 = []
24    hasil9 = []
25    hasil10 = []
26
27    for x in npm:
28        hasil1.append(angka1[x])
29        hasil2.append(angka2[x])
30        hasil3.append(angka3[x])
31        hasil4.append(angka4[x])
32        hasil5.append(angka5[x])
33        hasil6.append(angka6[x])
34        hasil7.append(angka7[x])
35        hasil8.append(angka8[x])
36        hasil9.append(angka9[x])
37        hasil10.append(angka10[x])
38
39    print(*hasil1, sep=' ')
40    print(*hasil2, sep=' ')
41    print(*hasil3, sep=' ')
42    print(*hasil4, sep=' ')
43    print(*hasil5, sep=' ')
44    print(*hasil6, sep=' ')
45    print(*hasil7, sep=' ')
46    print(*hasil8, sep=' ')
47    print(*hasil9, sep=' ')
48    print(*hasil10, sep=' ')

```

## 9. no2

1 #No 2

```

2 def perulangan(npm):
3     hitung = 0
4     while(hitung < 62):
5         print("Halo, "+str(npm)+" apa kabar?")
6         hitung = hitung +1
7
8 perulangan(int(input("Masukan NPM : ")))

```

## 10. no3

```

1 #No 3
2 def perulangan_3_digit(npm):
3     hitung = 0
4     npm = str(npm)
5     bil = npm[4:7]
6
7     while(hitung < 8):
8         print("Halo, "+bil+" apa kabar?")
9         hitung = hitung +1
10
11 perulangan_3_digit(int(input("Masukan NPM : ")))

```

## 11. no4

```

1 #No 4
2 def perulangan_3_digit_terakhir(npm):
3     npm = str(npm)
4     bil = npm[-3]
5     print("Halo, "+bil+" apa kabar?")
6
7 perulangan_3_digit_terakhir(int(input("Masukan NPM : ")))

```

## item no5

```

1 #No 5
2 def down(npm):
3     for i in npm:
4         print(i)
5
6 down(input("Masukan NPM : "))

```

## 12. no6

```

1 #No 6
2 def penjumlahan(npm):
3     jumlah = 0
4     for i in npm:
5         jumlah += int(i)
6     print(str(jumlah)+" Adalah hasil penjumlahan")
7
8 penjumlahan(input("Masukan NPM : "))

```

## 13. no7

```

1 #No 7
2 def perkalian(npm):
3     jumlah = 0
4     for i in npm:
5         jumlah *= int(i)
6     print(str(jumlah)+" Adalah hasil perkalian")
7
8 perkalian(input("Masukan NPM : "))

```

## 14. no8

```

1 #No 8
2 def genap():
3     npm = [1,1,7,4,0,2,7]
4     for i in npm:
5         if (i % 2) == 0:
6             print("Bilangan Genapnya : "+str(i))
7 genap()

```

## 15. no9

```

1 #No 9
2 def ganjil():
3     npm = [1,1,7,4,0,2,7]
4     for i in npm:
5         if (i%2)==1:
6             print("Bilangan Ganjilnya : "+str(i))
7 ganjil()

```

## 16. no10

```

1 #No 10
2 def prima(npm):
3     npm = str(npm)
4     bil = npm[2]
5     num = int(bil)
6     if num > 1:
7         for i in range(2,num):
8             if (num%i)==0:
9                 print("Bukan Bilangan Prima")
10                break
11            else:
12                print("Bilangan Primanya :"+str(num))
13        else:
14            print("Tidak Ada Bilangan Prima")
15 prima(int(input("Masukan NPM : ")))

```