

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

1	SEJARAH DAN KARAKTERISTIK PYTHON	1
2	Judul Bagian Kedua	3
3	Judul Bagian Ketiga	5

DAFTAR ISI

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

3.1	Harun Ar - Rasyid	5
3.2	Evietania	11
3.3	Kadek Diva Krishna Murti	17
3.3.1	Pemahaman Teori	17
3.3.2	Ketrampilan Pemrograman	21
3.3.3	Ketrampilan Penanganan Error	25
3.4	Felix Setiawan Lase	25
Index		33

DAFTAR GAMBAR

DAFTAR TABEL

Listings

src/1174027.py	5
src/1174027.py	5
src/1174027.py	6
src/1174027.py	6
src/1174027.py	6
src/1174027.py	7
src/1174027.py	7
src/1174027.py	7
src/1174027.py	7
src/1174027.py	9
src/1174027.py	9
src/1174027.py	9
src/1174027.py	9
src/1174027.py	10
src/1174027.py	10
src/1174027.py	10
src/1174027.py	10
src/1174027.py	10

src/main.py	11
src/main.py	11
src/1174027.py	11
src/1174051_praktek.py	11
src/1174051_praktek.py	12
src/1174051_praktek.py	12
src/1174051_praktek.py	12
src/1174051_praktek.py	12
src/1174051_praktek.py	13
src/1174051_praktek.py	13
src/1174051_praktek.py	13
src/1174051_praktek.py	13
src/1174051_praktek.py	15
src/1174051_praktek.py	15
src/1174051_praktek.py	15
src/1174051_praktek.py	15
src/1174051_praktek.py	16
src/1174051_praktek.py	16
src/1174051_praktek.py	16
src/1174051_praktek.py	16
src/1174051_praktek.py	17
src/mainn.py	17
src/mainn.py	17
src/1174051_praktek.py	17
3.1 namaFungsi merupakan nama dari fungsi yang dibuat.	18
3.2 inputanFungsi merupakan nama dari inputan fungsi yang diterima dari luar fungsi namaFungsi.	18
3.3 return inputanFungsi merupakan kembalian fungsi dari fungsi namaFungsi.	18
3.4 Contoh penggunaan fungsi di Python.	18
3.5 Contoh penggunaan paket atau library.	18
3.6 Contoh penggunaan kelas di python.	19
3.7 Contoh penggunaan objek di python.	19
3.8 Contoh penggunaan atribut di python.	19
3.9 Contoh penggunaan method di python.	19
3.10 Contoh pemanggilan library kelas dari instansi dan pemakaiannya. .	19
3.11 Contoh pemakaian paket dengan perintah from kalkulator import Penambahan.	20

3.12 Contoh kode pemakaian paket fungsi dimana file library ada di dalam folder.	20
3.13 Contoh kode pemakaian paket kelas dimana file library ada di dalam folder.	20
3.14 Jawaban soal No. 1 Ketrampilan Pemrograman.	21
3.15 Jawaban soal No. 2 Ketrampilan Pemrograman.	22
3.16 Jawaban soal No. 3 Ketrampilan Pemrograman.	22
3.17 Jawaban soal No. 4 Ketrampilan Pemrograman.	22
3.18 Jawaban soal No. 5 Ketrampilan Pemrograman.	22
3.19 Jawaban soal No. 6 Ketrampilan Pemrograman.	23
3.20 Jawaban soal No. 7 Ketrampilan Pemrograman.	23
3.21 Jawaban soal No. 8 Ketrampilan Pemrograman.	23
3.22 Jawaban soal No. 9 Ketrampilan Pemrograman.	23
3.23 Jawaban soal No. 10 Ketrampilan Pemrograman.	24
3.24 Jawaban soal No. 11 Ketrampilan Pemrograman.	24
3.25 Jawaban soal No. 12 Ketrampilan Pemrograman.	24
3.26 Contoh fungsi yang menggunakan try except .	25
src/1174026.py	26
src/1174026.py	26
src/1174026.py	26
src/1174026.py	26
src/1174026.py	26
src/1174026.py	27
src/1174026.py	27
src/1174026.py	27
src/1174026.py	28
src/1174026.py	29
src/1174026.py	29
src/1174026.py	29
src/1174026.py	30
src/1174026.py	30
src/1174026.py	30
src/1174026.py	30
src/1174026.py	30
src/1174026.py	31
src/main.py	31
src/main.py	31
src/1174026.py	31

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 Harun Ar - Rasyid

3.1.0.1 Pemahaman Teori

1. Apa itu fungsi, inputan

fungsi dan kembalian fungsi dengan contoh kode program lainnya. Fungsi adalah bagian dari program yang dapat digunakan ulang. Berikut merupakan contoh fungsi dan cara pemanggilannya

```
1 def uji():
2     print("Tugas Web Service")
3
4 uji()
```

Fungsi dapat membaca parameter, parameter adalah nilai yang disediakan kepada fungsi, dimana nilai ini akan menentukan output yang akan dihasilkan fungsi.

```
1 def uji_param(nama):
2     print("Nama saya :"+str(nama))
3
4 uji_param(input("Masukan Nama Kamu : "))
```

Statemen return digunakan untuk keluar dari fungsi. Kita juga dapat menspesifikasikan nilai kembalian.

```

1 def uji_return(a,b):
2     r = a + b
3     return r
4
5 a = 10
6 b = 50
7 c = uji_return(a,b)
8 print(c)

```

2. Apa itu paket dan cara pemanggilan paket atau library dengan contoh kode program lainnya. Untuk memudahkan dalam pemanggilan fungsi yang di butuhkan, agar dapat dipanggil berulang. Cara pemanggilannya

```

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

```

3. Jelaskan Apa itu kelas, apa itu objek, apa itu atribut, apa itu method dan contoh kode program lainnya masing-masing. kelas merupakan sebuah blueprint yang merepresentasikan objek. objek adalah hasil cetakan dari sebuah kelas. method adalah suatu upaya yang digunakan oleh object.

```

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("Zara", 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. Jelaskan cara pemanggilan library kelas dari instansiasi dan pemakaiannya dengan contoh program lainnya. Cara Pemanggilanya

- pertama import terlebih dahulu filenya.
- kemudian buat variabel untuk menampung datanya

- setelah itu panggil nama classnya dan panggil methodnya
- Gunakan perintah print untuk menampilkan hasilnya

```

1 #import belajar
2 #a = 100
3 #b = 50
4
5 #c = belajar.penambahan(a,b)
6 #print(c)

```

5. Jelaskan dengan contoh pemakaian paket dengan perintah from kalkulator import Penambahan disertai dengan contoh kode lainnya. Penggunaan paket from namafile import, itu berfungsi untuk memanggil file dan fungsinya

```

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

```

6. Jelaskan dengan contoh kodenya, pemakaian paket fungsi apabila le library ada di dalam folder. Pemakaian paket adalah perkumpulan fungsi-fungsi. contoh kodenya adalah sebagai berikut :
7. Jelaskan dengan contoh kodenya, pemakaian paket kelas apabila le library ada di dalam folder.

```

1 #Chapter 3

```

3.1.0.2 Ketramplan Pemrograman

1. Buatlah fungsi dengan inputan variabel NPM, dan melakukan print luaran huruf yang dirangkai dari tanda bintang, pagar atau plus dari NPM kita. Tanda bintang untuk NPM mod 3=0, tanda pagar untuk NPM mod 3 =1, tanda plus untuk NPM mod3=2.

```

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

```



```

10  angka5 = {"0": " ##      ## ", "1": " ### ", "2": " ##### ", "3": "
      ##### ", "4": " ##### ", "5": " ##### ", "6": " ##### ", "7": "
      ###      ", "8": " ##### ", "9": " ##### "}
11  angka6 = {"0": " ##      ## ", "1": " ### ", "2": " ##### ", "3": "
      ##### ", "4": " ##### ", "5": " ##### ", "6": " ##### ", "7": "
      ###      ", "8": " ##### ", "9": " ##### "}
12  angka7 = {"0": " ##      ## ", "1": " ### ", "2": " ##### ", "3": "
      ##### ", "4": " ##### ", "5": " ##### ", "6": " ##### ", "7": "
      ###      ", "8": " ##### ", "9": " ##### "}
13  angka8 = {"0": " ##      ## ", "1": " ### ", "2": " ##### ", "3": "
      ##### ", "4": " ##### ", "5": " ##### ", "6": " ##### ", "7": "
      ###      ", "8": " ##### ", "9": " ##### "}
14  angka9 = {"0": " ##### ", "1": " ### ", "2": " ##### ", "3": "
      ##### ", "4": " ##### ", "5": " ##### ", "6": " ##### ", "7": "
      ###      ", "8": " ##### ", "9": " ##### "}
15  angka10 = {"0": " ##### ", "1": " ### ", "2": " ##### ", "3": "
      ##### ", "4": " ##### ", "5": " ##### ", "6": " ##### ", "7": "
      ###      ", "8": " ##### ", "9": " ##### "}
16
17  hasil1 = []
18  hasil2 = []
19  hasil3 = []
20  hasil4 = []
21  hasil5 = []
22  hasil6 = []
23  hasil7 = []
24  hasil8 = []
25  hasil9 = []
26  hasil10 = []
27
28  for x in npm:
29      hasil1.append(angka1[x])
30      hasil2.append(angka2[x])
31      hasil3.append(angka3[x])
32      hasil4.append(angka4[x])
33      hasil5.append(angka5[x])
34      hasil6.append(angka6[x])
35      hasil7.append(angka7[x])
36      hasil8.append(angka8[x])
37      hasil9.append(angka9[x])
38      hasil10.append(angka10[x])
39
40  print(*hasil1, sep=' ')
41  print(*hasil2, sep=' ')
42  print(*hasil3, sep=' ')
43  print(*hasil4, sep=' ')
44  print(*hasil5, sep=' ')
45  print(*hasil6, sep=' ')
46  print(*hasil7, sep=' ')
47  print(*hasil8, sep=' ')
48  print(*hasil9, sep=' ')
49  print(*hasil10, sep=' ')
50
51  penulisan(int(input("Masukan NPM :")))

```

2. Buatlah fungsi dengan inputan variabel berupa NPM, kemudian dengan menggunakan perulangan mengeluarkan print output sebanyak dua digit belakang NPM.

```

1 def perulangan(npm):
2     hitung = 0
3     while (hitung < 27):
4         print("Halo, "+str(npm)+" apa kabar?")
5         hitung = hitung +1
6
7 perulangan(int(input("Masukan NPM : ")))

```

3. Buatlah fungsi dengan dengan input variabel string bernama NPM dan beri luaran output dengan perulangan berupa tiga karakter belakang dari NPM sebanyak penjumlahan tiga digit tersebut.

```

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 < 9):
8         print("Halo, "+bil+" apa kabar?")
9         hitung = hitung +1
10
11 perulangan_3_digit(int(input("Masukan NPM : ")))

```

4. Buatlah fungsi hello word dengan input variabel string bernama NPM dan beri luaran output berupa digit ketiga dari belakang dari variabel NPM menggunakan akses langsung manipulasi string pada baris ketiga dari variabel NPM.

```

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 : ")))

```

5. buat fungsi program dengan input variabel NPM dan melakukan print nomor npm satu persatu kebawah.

```

1 #No 5
2 def down(npm):
3     for i in npm:
4         print (i)

```

6. Buatlah fungsi dengan inputan variabel NPM, didalamnya melakukan penjumlahan dari seluruh digit NPM tersebut, wajib menggunakan perulangan dan atau kondisi.

```

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 : "))

```

7. Buatlah fungsi dengan inputan variabel NPM, didalamnya melakukan melakukan perkalian dari seluruh digit NPM tersebut, wajib menggunakan perulangan dan atau kondisi.

```

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 : "))

```

8. Buatlah fungsi dengan inputan variabel NPM, Lakukan print NPM anda tapi hanya digit genap saja. wajib menggunakan perulangan dan atau kondisi.

```

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()

```

9. Buatlah fungsi dengan inputan variabel NPM, Lakukan print NPM anda tapi hanya digit ganjil saja. wajib menggunakan perulangan dan atau kondisi.

```

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()

```

10. Buatlah fungsi dengan inputan variabel NPM, Lakukan print NPM anda tapi hanya digit yang termasuk bilangan prima saja. wajib menggunakan perulangan dan atau kondisi.

```

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 : ")))

```

11. Buatlah satu library yang berisi fungsi-fungsi dari nomor diatas dengan nama le 3lib.py dan berikan contoh cara pemanggilannya pada le main.py.

```

1 from fungsi_harun import penulisan

```

12. Buatlah satu library class dengan nama le kelas3lib.py yang merupakan mod- i kasi dari fungsi-fungsi nomor diatas dan berikan contoh cara pemanggilannya pada le main.py.

```

1 from 3lib import penulisan

```

3.1.0.3 Ketrampilan Penanganan Error Error yang di dapat dari mengerjakan tugas ini adalah type error, cara menaggulaginya dengan cara mengecek kembali codingannya kemudian run kembali aplikasinya berikut contoh Penggunaan fungsi try dan exception

```

1 #def penanganan_error(a,b):
2 #    try :
3 #        c = a+b
4 #        print(c)
5 #    except TypeError:
6 #        print("We Are Different")

```

3.2 Evietania

3.2.0.1 Pemahaman Teori

1. Apa itu fungsi, inputan

fungsi dan kembalian fungsi dengan contoh kode program lainnya. Fungsi adalah bagian dari program yang dapat digunakan ulang. Berikut merupakan contoh fungsi dan cara pemanggilannya

```

1 def uji():
2     print("Tugas")
3
4 uji()

```

Fungsi dapat membaca parameter, parameter adalah nilai yang disediakan kepada fungsi, dimana nilai ini akan menentukan output yang akan dihasilkan fungsi.

```
1 def uji_param(nama):
2     print("Nama saya :"+str(nama))
3
4 uji_param(input("Masukan Nama Kamu : "))
```

Statemen return digunakan untuk keluar dari fungsi. Kita juga dapat menspesifikasikan nilai kembalian.

```
1 def uji_return(a,b):
2     r = a + b
3     return r
4
5 a = 10
6 b = 50
7 c = uji_return(a,b)
8 print(c)
```

2. Apa itu paket dan cara pemanggilan paket atau library dengan contoh kode program lainnya. Untuk memudahkan dalam pemanggilan fungsi yang dibutuhkan, agar dapat dipanggil berulang. Cara pemanggilannya

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

3. Jelaskan Apa itu kelas, apa itu objek, apa itu atribut, apa itu method dan contoh kode program lainnya masing-masing. kelas merupakan sebuah blueprint yang merepresentasikan objek. objek adalah hasil cetakan dadri sebuah kelas. method adalah suatu upaya yang digunakan oleh object.

```
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("Zara", 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. Jelaskan cara pemanggilan library kelas dari instansiasi dan pemakaiannya dengan contoh program lainnya. Cara Pemanggilanya

- pertama import terlebih dahulu filenya.
- kemudian buat variabel untuk menampung datanya
- setelah itu panggil nama classnya dan panggil methodnya
- Gunakan perintah print untuk menampilkan hasilnya

```
1 #import belajar
2 #a = 100
3 #b = 50
4
5 #c = belajar.penambahan(a,b)
6 #print(c)
```

5. Jelaskan dengan contoh pemakaian paket dengan perintah from kalkulator import Penambahan disertai dengan contoh kode lainnya. Penggunaan paket from namafile import, itu berfungsi untuk memanggil file dan fungsinya

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

6. Jelaskan dengan contoh kodenya, pemakaian paket fungsi apabila
le library ada di dalam folder. Pemakaian paket adalah perkumpulan fungsi-fungsi. contoh kodenya adalah sebagai berikut :
7. Jelaskan dengan contoh kodenya, pemakaian paket kelas apabila
le library ada di dalam folder.

```
1 #Chapter 3
```

3.2.0.2 Ketampilan Pemrograman

1. Buatlah fungsi dengan inputan variabel NPM, dan melakukan print luaran huruf yang dirangkai dari tanda bintang, pagar atau plus dari NPM kita. Tanda bintang untuk NPM mod 3=0, tanda pagar untuk NPM mod 3 =1, tanda plus untuk NPM mod3=2.

```
1 #Chapter 3
2 #No 1
3 def penulisan(npm):
4     npm = list(str(npm))
5
6     angka1 = {"0": " ##### ", "1": " ### ", "2": " ##### ", "3": "
7     ##### ", "4": " ###   ### ", "5": " ##### ", "6": " ##### ", "7":
8     " ##### ", "8": " ##### ", "9": " ##### "}
9
10    angka2 = {"0": " ##### ", "1": " ### ", "2": " ##### ", "3": "
11    ##### ", "4": " ###   ### ", "5": " ##### ", "6": " ##### ", "7":
12    " ##### ", "8": " ##### ", "9": " ##### "}
```

```

8     angka3 = {"0": " ##      ## ", "1": " ### ", "2": "      ## ", "3": "
#####", "4": "      ##      ## ", "5": "      ##      ", "6": "      ##### ", "7":
: "      ##      ", "8": " ##### ", "9": " ##### "}
9     angka4 = {"0": " ##      ## ", "1": "      ## ", "2": "      ## ", "3": "
#####", "4": "      ##      ## ", "5": "      ##      ", "6": "      ##### ", "7":
: "      ##      ", "8": " ##### ", "9": " ##### "}
10    angka5 = {"0": " ##      ## ", "1": "      ## ", "2": "      ##### ", "3": "
#####", "4": " ##### ", "5": " ##### ", "6": "      ##### ", "7":
: "      ##      ", "8": " ##### ", "9": " ##### "}
11    angka6 = {"0": " ##      ## ", "1": "      ## ", "2": "      ##### ", "3": "
#####", "4": "      ##      ## ", "5": " ##### ", "6": "      ##### ", "7":
: "      ##      ", "8": " ##### ", "9": " ##### "}
12    angka7 = {"0": " ##      ## ", "1": "      ## ", "2": "      ## ", "3": "
#####", "4": "      ##      ## ", "5": "      ## ", "6": "      ##### ", "7":
: "      ##      ", "8": " ##### ", "9": " ##### "}
13    angka8 = {"0": " ##      ## ", "1": "      ## ", "2": "      ## ", "3": "
#####", "4": "      ##      ## ", "5": "      ## ", "6": "      ##### ", "7":
: "      ##      ", "8": " ##### ", "9": " ##### "}
14    angka9 = {"0": " ##### ", "1": "      ## ", "2": "      ##### ", "3": "
#####", "4": "      ##      ## ", "5": " ##### ", "6": "      ##### ", "7":
: "      ##      ", "8": " ##### ", "9": " ##### "}
15    angka10 = {"0": " ##### ", "1": "      ## ", "2": "      ##### ", "3": "
#####", "4": "      ##      ## ", "5": "      ## ", "6": "      ##### ", "7":
: "      ##      ", "8": " ##### ", "9": " ##### "}

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

```

```

48     print(*hasil9 , sep=' ')
49     print(*hasil10 , sep=' ')
50
51 penulisan(int(input("Masukan NPM :"))))

```

2. Buatlah fungsi dengan inputan variabel berupa NPM. kemudian dengan menggunakan perulangan mengeluarkan print output sebanyak dua digit belakang NPM.

```

1 def perulangan(npm):
2     hitung = 0
3     while(hitung < 51):
4         print("Halo, "+str(npm)+" apa kabar?")
5         hitung = hitung +1
6
7 perulangan(int(input("Masukan NPM : ")))

```

3. Buatlah fungsi dengan dengan input variabel string bernama NPM dan beri luaran output dengan perulangan berupa tiga karakter belakang dari NPM sebanyak penjumlahan tiga digit tersebut.

```

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 < 9):
8         print("Halo, "+bil+" apa kabar?")
9         hitung = hitung +1
10
11 perulangan_3_digit(int(input("Masukan NPM : ")))

```

4. Buatlah fungsi hello word dengan input variabel string bernama NPM dan beri luaran output berupa digit ketiga dari belakang dari variabel NPM menggunakan akses langsung manipulasi string pada baris ketiga dari variabel NPM.

```

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 : ")))

```

5. buat fungsi program dengan input variabel NPM dan melakukan print nomor npm satu persatu kebawah.

```

1 #No 5
2 def down(npm):
3     for i in npm:
4         print (i)

```


6. Buatlah fungsi dengan inputan variabel NPM, didalamnya melakukan penjumlahan dari seluruh digit NPM tersebut, wajib menggunakan perulangan dan atau kondisi.

```

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 : "))

```

7. Buatlah fungsi dengan inputan variabel NPM, didalamnya melakukan melakukan perkalian dari seluruh digit NPM tersebut, wajib menggunakan perulangan dan atau kondisi.

```

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 : "))

```

8. Buatlah fungsi dengan inputan variabel NPM, Lakukan print NPM anda tapi hanya digit genap saja. wajib menggunakan perulangan dan atau kondisi.

```

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()

```

9. Buatlah fungsi dengan inputan variabel NPM, Lakukan print NPM anda tapi hanya digit ganjil saja. wajib menggunakan perulangan dan atau kondisi.

```

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()

```

10. Buatlah fungsi dengan inputan variabel NPM, Lakukan print NPM anda tapi hanya digit yang termasuk bilangan prima saja. wajib menggunakan perulangan dan atau kondisi.

```

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 : ")))

```

11. Buatlah satu library yang berisi fungsi-fungsi dari nomor diatas dengan nama le epi.py dan berikan contoh cara pemanggilannya pada le main.py.

```

1 from fungsi_epie import penulisan

```

12. Buatlah satu library class dengan nama le kelas3lib.py yang merupakan mod- i kasi dari fungsi-fungsi nomor diatas dan berikan contoh cara pemanggilannya pada le mainn.py.

```

1 from 3lib import penulisan

```

3.2.0.3 Ketrampilan Penanganan Error Error yang di dapat dari mengerjakan tugas ini adalah type error, cara menaggulaginya dengan cara mengecek kembali codingannya kemudian run kembali aplikasinya berikut contoh Penggunaan fungsi try dan exception

```

1 #def penanganan_error(a,b):
2 #     try :
3 #         c = a+b
4 #         print(c)
5 #     except TypeError:
6 #         print("We Are Different")

```

3.3 Kadek Diva Krishna Murti

3.3.1 Pemahaman Teori

1. Pengertian fungsi, inputan fungsi, dan kembalian fungsi serta contoh kode programnya.

- Fungsi adalah blok program untuk melakukan tugas-tugas tertentu yang dilakukan berulang dan dapat digunakan berulang kali dari tempat lain di dalam program.

```
1 def namaFungsi(inputanFungsi):
```

Listing 3.1 namaFungsi merupakan nama dari fungsi yang dibuat.

- Inputan fungsi adalah inputan yang berasal dari luar fungsi yang akan di proses di dalam fungsi itu sendiri.

```
1 def namaFungsi(inputanFungsi):
```

Listing 3.2 inputanFungsi merupakan nama dari inputan fungsi yang diterima dari luar fungsi namaFungsi.

- Kembalian fungsi adalah untuk mengembalikan suatu nilai ekspresi dari proses yang dilakukan fungsi.

```
1 return inputanFungsi
```

Listing 3.3 return inputanFungsi merupakan kembalian fungsi dari fungsi namaFungsi.

Penggunaan fungsi di Python

```
1 #Jawaban No. 1
2 def namaFungsi(inputanFungsi):
3     return inputanFungsi
4
5 output = namaFungsi("Kembalian Fungsi")
6 print(output)
```

Listing 3.4 Contoh penggunaan fungsi di Python.

2. Pengertian paket dan cara pemanggilannya serta contoh kode programnya.

Paket atau library adalah file yang berisi kode program python yang bisa digunakan berulang dimana paket itu dipanggil.

Cara pemanggilan paket atau library yaitu dengan meng-import paket atau library yang akan digunakan. Lalu panggil dengan cara mendefinisikan nama-paket.namafungsinya.

Berikut ini merupakan contoh penggunaan paket atau library.

```
1 #Jawaban No. 2
2 import math
3 print("Nilai pi adalah: ", math.pi)
```

Listing 3.5 Contoh penggunaan paket atau library.

3. Pengertian kelas, objek, atribut, method, dan contoh kode programnya.

- Kelas Kelas adalah cetak biru atau prototipe dari objek dimana kita mendefinisikan atribut dari suatu objek. Contoh penggunaan kelas di python.

```

1 #Jawaban No. 3
2 class Mahasiswa:
3     jumlahMahasiswa = 0
4
5     def __init__(self, npm, nama):
6         self.npm = npm
7         self.nama = nama
8         Mahasiswa.jumlahMahasiswa +=1
9
10    def tampilkanProfil(self):
11        print("NPM :", self.npm)
12        print("Nama :", self.nama)
13        print()
14
15    mahasiswa1 = Mahasiswa("1174006", "Diva Krishna")
16    mahasiswa2 = Mahasiswa("1174001", "Indah Pramesti")
17
18    mahasiswa1.tampilkanProfil()
19    mahasiswa2.tampilkanProfil()
20
21    print("Total mahasiswa adalah ", Mahasiswa.jumlahMahasiswa)

```

Listing 3.6 Contoh penggunaan kelas di python.

- Objek Objek adalah instansi atau perwujudan dari sebuah kelas.

```

1 mahasiswa1 = Mahasiswa("1174006", "Diva Krishna")
2 mahasiswa2 = Mahasiswa("1174001", "Indah Pramesti")

```

Listing 3.7 Contoh penggunaan objek di python.

- Atribut Atribut adalah variabel yang menyimpan data yang berhubungan dengan kelas dan objeknya.

```

1     jumlahMahasiswa = 0

```

Listing 3.8 Contoh penggunaan atribut di python.

- Method Metode adalah fungsi yang didefinisikan di dalam suatu kelas.

```

1     def tampilkanProfil(self):
2         print("NPM :", self.npm)
3         print("Nama :", self.nama)
4         print()

```

Listing 3.9 Contoh penggunaan method di python.

4. Cara pemanggilan library kelas, dan contoh kode programnya.

Berikut ini adalah cara pemanggilan library kelas dari instansi dan pemakaiannya. Library kelasnya adalah Mahasiswa dari file Mahasiswa.py. Lalu dipanggil dengan import. Kemudian instansi dengan mhs1 dan mhs1, dengan 2 parameter.

```

1 #Jawaban No. 4
2 from Mahasiswa import Mahasiswa
3
4 mhs1 = Mahasiswa("1174006", "Diva Krishna")

```

```
5 mhs2 = Mahasiswa("1174001", "Indah Pramesti")
6
7 mhs1.tampilkanProfil()
8 mhs2.tampilkanProfil()
9
10 print("Total mahasiswa adalah ", Mahasiswa.jumlahMahasiswa)
```

Listing 3.10 Contoh pemanggilan library kelas dari instansi dan pemakaiannya. .

5. Penjelasan pemakaian paket disertai dengan contoh kode programnya.

Berikut ini adalah contoh pemakaian paket dengan perintah from kalkulator import Penambahan. Setelah mengimport paketnya, lalu panggil fungsi penambahannya.

```
1 #Jawaban No. 5
2 from kalkulator import Penambahan
3
4 hasil = Penambahan(10, 5)
5 print(hasil)
```

Listing 3.11 Contoh pemakaian paket dengan perintah from kalkulator import Penambahan.

6. Contoh kode pemakaian paket fungsi apabila file library ada di dalam folder.
Berikut ini adalah pemakaian paket fungsi apabila file library ada di dalam folder.

```
1 #Jawaban No. 6
2 from folder import kalkulator
3
4 a=100
5 b=50
6
7 hasil1=kalkulator.Penambahan(a,b)
8 hasil2=kalkulator.Pengurangan(a,b)
9 hasil3=kalkulator.Perkalian(a,b)
10 hasil4=kalkulator.Pembagian(a,b)
11
12 print(hasil1)
13 print(hasil2)
14 print(hasil3)
15 print(hasil4)
```

Listing 3.12 Contoh kode pemakaian paket fungsi dimana file library ada di dalam folder.

7. Contoh kode pemakaian paket kelas apabila file library ada di dalam folder.
Berikut ini adalah pemakaian paket kelas apabila file library ada di dalam folder.

```
1 #Jawaban No. 7
2 from folder.Mahasiswa import Mahasiswa
3
4 mhs1 = Mahasiswa("1174006", "Diva Krishna")
5 mhs2 = Mahasiswa("1174001", "Indah Pramesti")
6
7 mhs1.tampilkanProfil()
```

```

8 mhs2.tampilkanProfil()
9
10 print("Total mahasiswa adalah ", Mahasiswa.jumlahMahasiswa)

```

Listing 3.13 Contoh kode pemakaian paket kelas dimana file library ada di dalam folder.

3.3.2 Ketrampilan Pemrograman

1. Jawaban soal No. 1

```

1 #Jawaban No. 1
2 def printNPM(npm):
3
4     npm = list(str(npm))
5
6     angka1 = {"0": "##### ", "1": " ##", "2": "##### ", "3": "##### ",
7              "4": "      ##", "5": "#####", "6": "##### ", "7": "#####",
8              "8": "##### ", "9": "#####"}
9     angka2 = {"0": "###  ##", "1": "####", "2": "##  ##", "3": "##  ##",
10             "4": "#####", "5": "##  ##", "6": "###  ##", "7": "##  ##",
11             "8": "###  ##", "9": "##  ##"}
12     angka3 = {"0": "###  ##", "1": "###  ##", "2": "###  ##", "3": "###  ##",
13             "4": "###  ##", "5": "#####", "6": "##### ", "7": "###  ##",
14             "8": "###  ##", "9": "###  ##"}
15     angka4 = {"0": "###  ##", "1": "###  ##", "2": "###  ##", "3": "###  ##",
16             "4": "#####", "5": "###  ##", "6": "###  ##", "7": "###  ##",
17             "8": "###  ##", "9": "#####"}
18     angka5 = {"0": "###  ##", "1": "###  ##", "2": "###  ##", "3": "###  ##",
19             "4": "###  ##", "5": "###  ##", "6": "###  ##", "7": "###  ##",
20             "8": "###  ##", "9": "###  ##"}
21     angka6 = {"0": "##### ", "1": "###  ##", "2": "#####", "3": "##### ",
22             "4": "      ##", "5": "##### ", "6": "##### ", "7": "###  ##",
23             "8": "##### ", "9": "##### "}
24
25     hasil1 = []
26     hasil2 = []
27     hasil3 = []
28     hasil4 = []
29     hasil5 = []
30     hasil6 = []
31
32     for x in npm:
33         hasil1.append(angka1[x])
34         hasil2.append(angka2[x])
35         hasil3.append(angka3[x])
36         hasil4.append(angka4[x])
37         hasil5.append(angka5[x])
38         hasil6.append(angka6[x])
39
40     print(*hasil1, sep=' ')
41     print(*hasil2, sep=' ')
42     print(*hasil3, sep=' ')
43     print(*hasil4, sep=' ')
44     print(*hasil5, sep=' ')
45     print(*hasil6, sep=' ')

```

```
35 printNPM(input("Masukan NPM anda: "))
```

Listing 3.14 Jawaban soal No. 1 Ketrampilan Pemrograman.

2. Jawaban soal No. 2

```
1 #Jawaban No. 2
2 def printNPMDuaDijit(npm):
3     ulang = 1
4     sampai = int(npm[5:7])
5     while(ulang <= sampai):
6         print("Halo, "+str(npm)+" apa kabar?")
7         ulang += 1
8
9 printNPMDuaDijit(input("Masukan NPM anda: "))
```

Listing 3.15 Jawaban soal No. 2 Ketrampilan Pemrograman.

3. Jawaban soal No. 3

```
1 #Jawaban No. 3
2 def printNPMTigaDijit(npm):
3     ulang = 1
4     sampai = list(map(int, npm[4:7]))
5     sampai = sum(sampai)
6     while(ulang <= sampai):
7         print("Halo, "+str(npm[-3:])+ " apa kabar?")
8         ulang += 1
9
10 printNPMTigaDijit(input("Masukan NPM anda: "))
```

Listing 3.16 Jawaban soal No. 3 Ketrampilan Pemrograman.

4. Jawaban soal No. 4

```
1 #Jawaban No. 4
2 def printNPMDigitKetiga(npm):
3     print("Output:")
4     print("Halo, "+str(npm[-3])+ " apa kabar?")
5
6 printNPMDigitKetiga(input("Input: "))
```

Listing 3.17 Jawaban soal No. 4 Ketrampilan Pemrograman.

5. Jawaban soal No. 5

```
1 #Jawaban No. 5
2 def printNPMSatuPersatu(npm):
3     npm = list(map(int, npm))
4     for n in npm:
5         print(n)
6
7 printNPMSatuPersatu(input("Masukan NPM anda: "))
```

Listing 3.18 Jawaban soal No. 5 Ketrampilan Pemrograman.

6. Jawaban soal No. 6

```
1 #Jawaban No. 6
2 def printNMPenjumlahan(npm):
3     npm = list(map(int, npm))
4     hasil = 0
5     for x in npm:
6         hasil += x
7
8     print(hasil)
9
10 printNMPenjumlahan(input("Masukan NPM anda: "))
```

Listing 3.19 Jawaban soal No. 6 Ketrampilan Pemrograman.

7. Jawaban soal No. 7

```
1 #Jawaban No. 7
2 def printNMPerkalian(npm):
3     npm = list(map(int, npm))
4     hasil = 0
5     for x in npm:
6         hasil *= x
7
8     print(hasil)
9
10 printNMPerkalian(input("Masukan NPM anda: "))
```

Listing 3.20 Jawaban soal No. 7 Ketrampilan Pemrograman.

8. Jawaban soal No. 8

```
1 #Jawaban No. 8
2 def printNPMDijitGenap(npm):
3     npm = list(map(int, npm))
4     for n in npm:
5         if(n % 2 == 0):
6             if(n != 0):
7                 print(n, end = "")
8
9 printNPMDijitGenap(input("Masukan NPM anda: "))
```

Listing 3.21 Jawaban soal No. 8 Ketrampilan Pemrograman.

9. Jawaban soal No. 9

```
1 #Jawaban No. 9
2 def printNPMDijitGanjil(npm):
3     npm = list(map(int, npm))
4     for n in npm:
5         if(n % 2 != 0):
6             print(n, end = "")
7
8 printNPMDijitGanjil(input("Masukan NPM anda: "))
```

Listing 3.22 Jawaban soal No. 9 Ketrampilan Pemrograman.

10. Jawaban soal No. 10

```

1 #Jawaban No. 10
2 def printNPMDijitPrima(npm):
3     npm = list(map(int, npm))
4     prima = []
5     for n in npm:
6         isPrime = True
7         if n == 0 or n == 1:
8             isPrime = False
9         for x in range(2, n):
10            if n % x == 0:
11                isPrime = False
12        if isPrime:
13            prima.append(n)
14
15    for p in prima:
16        print(p, end = " ")
17
18 printNPMDijitPrima(input("Masukan NPM anda: "))

```

Listing 3.23 Jawaban soal No. 10 Ketrampilan Pemrograman.

11. Jawaban soal No. 11

```

1 lib = __import__('3lib')
2
3 npm = "1174006"
4
5 lib.printNPM(npm)
6 lib.printNPMDuaDijit(npm)
7 lib.printNPMTigaDijit(npm)
8 lib.printNPMDigitKetiga(npm)
9 lib.printNPMSatuPersatu(npm)
10 lib.printNMPenjumlahan(npm)
11 lib.printNMPerkalian(npm)
12 lib.printNPMDijitGenap(npm)
13 lib.printNPMDijitGanjil(npm)
14 lib.printNPMDijitPrima(npm)

```

Listing 3.24 Jawaban soal No. 11 Ketrampilan Pemrograman.

12. Jawaban soal No. 12

```

1 from kelas3lib import kelas3lib
2
3 npm = "1174006"
4
5 k3lib = kelas3lib(npm)
6
7 k3lib.printNPM()
8 k3lib.printNPMDuaDijit()
9 k3lib.printNPMTigaDijit()
10 k3lib.printNPMDigitKetiga()
11 k3lib.printNPMSatuPersatu()
12 k3lib.printNMPenjumlahan()
13 k3lib.printNMPerkalian()

```

```

14 k3lib.printNPMDijitGenap()
15 k3lib.printNPMDijitGanjil()
16 k3lib.printNPMDijitPrima()

```

Listing 3.25 Jawaban soal No. 12 Ketrampilan Pemrograman.

3.3.3 Ketrampilan Penanganan Error

1. Peringatan error yang ditemukan dan penjelasannya serta buat sebuah fungsi try except untuk menanggulangi error.

Peringatan error di praktek ketiga ini, yaitu:

- **Syntax Errors** Syntax Errors adalah suatu keadaan saat kode python mengalami kesalahan penulisan. Solusinya adalah memperbaiki penulisan kode yang salah.
- **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 yang terjadi saat kode melakukan eksekusi terhadap local name atau global name yang tidak terdefinisi. Solusinya adalah memastikan variabel atau function yang dipanggil ada atau tidak salah ketik.
- **Type Error** TypeError adalah exception yang terjadi saat dilakukan eksekusi terhadap suatu operasi atau fungsi dengan type object yang tidak sesuai. Solusinya adalah mengkonversi variabelnya sesuai dengan tipe data yang akan digunakan.

Contoh fungsi yang menggunakan try except

```

1 #Jawaban No. 1
2 def sapa(nama):
3     try:
4         print("Hallo , "+str(nama))
5     except:
6         print("Terjadi error")
7
8 sapa(input("Masukan nama anda: "))

```

Listing 3.26 Contoh fungsi yang menggunakan try except .

3.4 Felix Setiawan Lase

3.4.0.1 Pemahaman Teori

1. Apa itu fungsi, inputan

fungsi dan kembalian fungsi dengan contoh kode program lainnya. Fungsi adalah bagian dari program yang dapat digunakan ulang. Berikut merupakan contoh fungsi dan cara pemanggilannya

```
1 def uji():
2     print("Tugas Web Service")
3
4 uji()
```

Fungsi dapat membaca parameter, parameter adalah nilai yang disediakan kepada fungsi, dimana nilai ini akan menentukan output yang akan dihasilkan fungsi.

```
1 def uji_param(nama):
2     print("Nama saya :"+str(nama))
3
4 uji_param(input("Masukan Nama Kamu : "))
```

Statemen return digunakan untuk keluar dari fungsi. Kita juga dapat menspesifikasikan nilai kembalian.

```
1 def uji_return(a,b):
2     r = a + b
3     return r
4
5 a = 10
6 b = 50
7 c = uji_return(a,b)
8 print(c)
```

2. Apa itu paket dan cara pemanggilan paket atau library dengan contoh kode program lainnya. Untuk memudahkan dalam pemanggilan fungsi yang di butuhkan, agar dapat dipanggil berulang. Cara pemanggilannya

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

3. Jelaskan Apa itu kelas, apa itu objek, apa itu atribut, apa itu method dan contoh kode program lainnya masing-masing. kelas merupakan sebuah blueprint yang merepresentasikan objek. objek adalah hasil cetakan dari sebuah kelas. method adalah suatu upaya yang digunakan oleh object.

```
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("Zara", 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. Jelaskan cara pemanggilan library kelas dari instansiasi dan pemakaiannya dengan contoh program lainnya. Cara Pemanggilanya

- pertama import terlebih dahulu filenya.
- kemudian buat variabel untuk menampung datanya
- setelah itu panggil nama classnya dan panggil methodnya
- Gunakan perintah print untuk menampilkan hasilnya

```

1 #import belajar
2 #a = 100
3 #b = 50
4
5 #c = belajar.penambahan(a,b)
6 #print(c)

```

5. Jelaskan dengan contoh pemakaian paket dengan perintah from kalkulator import Penambahan disertai dengan contoh kode lainnya. Penggunaan paket from namafile import, itu berfungsi untuk memanggil file dan fungsinya

```

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

```

6. Jelaskan dengan contoh kodenya, pemakaian paket fungsi apabila le library ada di dalam folder. Pemakaian paket adalah perkumpulan fungsi-fungsi. contoh kodenya adalah sebagai berikut :
7. Jelaskan dengan contoh kodenya, pemakaian paket kelas apabila le library ada di dalam folder.

3.4.0.2 Ketrampilan Pemrograman

1. Buatlah fungsi dengan inputan variabel NPM, dan melakukan print luaran huruf yang dirangkai dari tanda bintang, pagar atau plus dari NPM kita. Tanda bintang untuk NPM mod 3=0, tanda pagar untuk NPM mod 3 =1, tanda plus untuk NPM mod3=2.

```

1
2 #No 1
3 def speakup(npm):
4     npm = list(str(npm))
5
6     angka1 = {"0": "*****", "1": "***", "2": "*****", "3": "*****", "4": "*****", "5": "*****", "6": "*****", "7": "*****", "8": "*****", "9": "*****"}
7     angka2 = {"0": "*****", "1": "*****", "2": "*****", "3": "*****", "4": "*****", "5": "*****", "6": "*****", "7": "*****", "8": "*****", "9": "*****"}
8     angka3 = {"0": "*****", "1": "*****", "2": "*****", "3": "*****", "4": "*****", "5": "*****", "6": "*****", "7": "*****", "8": "*****", "9": "*****"}
9     angka4 = {"0": "*****", "1": "*****", "2": "*****", "3": "*****", "4": "*****", "5": "*****", "6": "*****", "7": "*****", "8": "*****", "9": "*****"}
10    angka5 = {"0": "*****", "1": "*****", "2": "*****", "3": "*****", "4": "*****", "5": "*****", "6": "*****", "7": "*****", "8": "*****", "9": "*****"}
11    angka6 = {"0": "*****", "1": "*****", "2": "*****", "3": "*****", "4": "*****", "5": "*****", "6": "*****", "7": "*****", "8": "*****", "9": "*****"}
12    angka7 = {"0": "*****", "1": "*****", "2": "*****", "3": "*****", "4": "*****", "5": "*****", "6": "*****", "7": "*****", "8": "*****", "9": "*****"}
13    angka8 = {"0": "*****", "1": "*****", "2": "*****", "3": "*****", "4": "*****", "5": "*****", "6": "*****", "7": "*****", "8": "*****", "9": "*****"}
14    angka9 = {"0": "*****", "1": "*****", "2": "*****", "3": "*****", "4": "*****", "5": "*****", "6": "*****", "7": "*****", "8": "*****", "9": "*****"}
15    angka10 = {"0": "*****", "1": "*****", "2": "*****", "3": "*****", "4": "*****", "5": "*****", "6": "*****", "7": "*****", "8": "*****", "9": "*****"}
16
17    hasil1 = []
18    hasil2 = []
19    hasil3 = []
20    hasil4 = []
21    hasil5 = []
22    hasil6 = []
23    hasil7 = []
24    hasil8 = []
25    hasil9 = []
26    hasil10 = []
27
28    for x in npm:
29        hasil1.append(angka1[x])
30        hasil2.append(angka2[x])
31        hasil3.append(angka3[x])
32        hasil4.append(angka4[x])
33        hasil5.append(angka5[x])
34        hasil6.append(angka6[x])
35        hasil7.append(angka7[x])
36        hasil8.append(angka8[x])

```

```

37     hasil9.append(angka9[x])
38     hasil10.append(angka10[x])
39
40     print(*hasil1, sep=' ')
41     print(*hasil2, sep=' ')
42     print(*hasil3, sep=' ')
43     print(*hasil4, sep=' ')
44     print(*hasil5, sep=' ')
45     print(*hasil6, sep=' ')
46     print(*hasil7, sep=' ')
47     print(*hasil8, sep=' ')
48     print(*hasil9, sep=' ')
49     print(*hasil10, sep=' ')
50
51 speakup(int(input("Masukan NPM : 117")))

```

2. Buatlah fungsi dengan inputan variabel berupa NPM. kemudian dengan menggunakan perulangan mengeluarkan print output sebanyak dua digit belakang NPM.

```

1 def looping(npm):
2     hitung = 0
3     while(hitung < 26):
4         print("Haiiiii, "+str(npm)+" apa kabar?")
5         hitung = hitung +1
6
7 looping(int(input("Masukan NPM : ")))

```

3. Buatlah fungsi dengan dengan input variabel string bernama NPM dan beri luaran output dengan perulangan berupa tiga karakter belakang dari NPM sebanyak penjumlahan tiga digit tersebut.

```

1 #No 3
2 def looping_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 looping_3_digit(int(input("Masukan NPM : ")))

```

4. Buatlah fungsi hello word dengan input variabel string bernama NPM dan beri luaran output berupa digit ketiga dari belakang dari variabel NPM menggunakan akses langsung manipulasi string pada baris ketiga dari variabel NPM.

```

1 #No 4
2 def looping_3_digit_terakhir(npm):
3     npm = str(npm)
4     bil = npm[-3]
5     print("Haiii, "+bil+" apa kabar?")
6
7 looping_3_digit_terakhir(int(input("Masukan NPM : ")))

```

5. buat fungsi program dengan input variabel NPM dan melakukan print nomor npm satu persatu kebawah.

```

1 #No 5
2 def down(npm):
3     for i in npm:
4         print (i)

```

6. Buatlah fungsi dengan inputan variabel NPM, didalamnya melakukan penjumlahan dari seluruh digit NPM tersebut, wajib menggunakan perulangan dan atau kondisi.

```

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

```

7. Buatlah fungsi dengan inputan variabel NPM, didalamnya melakukan melakukan perkalian dari seluruh digit NPM tersebut, wajib menggunakan perulangan dan atau kondisi.

```

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

```

8. Buatlah fungsi dengan inputan variabel NPM, Lakukan print NPM anda tapi hanya digit genap saja. wajib menggunakan perulangan dan atau kondisi.

```

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

```

9. Buatlah fungsi dengan inputan variabel NPM, Lakukan print NPM anda tapi hanya digit ganjil saja. wajib menggunakan perulangan dan atau kondisi.

```

1 even()
2
3 #No 9
4 def odd():
5     npm = [1,1,7,4,0,2,6]
6     for i in npm:
7         if (i%2)==1:

```

10. Buatlah fungsi dengan inputan variabel NPM, Lakukan print NPM anda tapi hanya digit yang termasuk bilangan prima saja. wajib menggunakan perulangan dan atau kondisi.

```

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

```

11. Buatlah satu library yang berisi fungsi-fungsi dari nomor diatas dengan nama le 3lib.py dan berikan contoh cara pemanggilannya pada le main.py.

```

1 from fungsi_harun import penulisan

```

12. Buatlah satu library class dengan nama le kelas3lib.py yang merupakan mod- i kasi dari fungsi-fungsi nomor diatas dan berikan contoh cara pemanggilannya pada le main.py.

```

1 from 3lib import penulisan

```

3.4.0.3 Ketrampilan Penanganan Error Error yang di dapat dari mengerjakan tugas ini adalah type error, cara menaggulaginya dengan cara mengecek kembali codingannya kemudian run kembali aplikasinya berikut contoh Penggunaan fungsi try dan exception

```

1 #def penanganan_error(a,b):
2 #     try :
3 #         c = a+b
4 #         print(c)
5 #     except TypeError:
6 #         print("kita beda")

```


Index

disruptif, xxxi
modern, xxxi