

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	xxiii
Kata Pengantar	xxv
Acknowledgments	xxvii
Acronyms	xxix
Glossary	xxxi
List of Symbols	xxxiii
Introduction	xxxv
<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
3.3.4	Hasil Screenshoot	25
3.4	Felix Setiawan Lase	32
3.5	Dwi Septiani Tsaniyah	38
3.6	Dwi Yulianingsih	41
3.6.1	Pemahaman Teori	41
3.6.2	Praktek	43
3.6.3	Penanganan Error	46
3.7	Choirul Anam	46
3.8	Muh. Rifky Prananda	53
3.9	Habib Abdul Rasyid	58
3.9.1	Pemahaman Teori	58
3.9.2	Ketrampilan Pemrograman	61
3.9.3	Ketrampilan Penanganan Error	65
3.10	Dezha Martha	66
3.10.1	Pemahaman Teori	66
3.10.2	Praktek	69
3.10.3	Praktek	72
3.11	Damara Benedikta	73
3.11.1	Pemahaman Teori	73
3.11.2	Praktek	75
3.11.3	Ketrampilan Penanganan Error	79
	Index	81

DAFTAR GAMBAR

3.1	SS Bebas Plagiarisme	47
3.2	SS Bebas Plagiarisme	66
3.3	SS plagiarisme	79

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	32
src/1174026.py	32
src/1174026.py	32
src/1174026.py	33
src/1174026.py	33
src/1174026.py	33
src/1174026.py	34
src/1174026.py	34
src/1174026.py	34
src/1174026.py	35
src/1174026.py	36
src/1174026.py	36
src/1174026.py	36
src/1174026.py	36
src/1174026.py	36
src/1174026.py	37
src/1174026.py	37
src/1174026.py	37
src/main.py	37
src/main.py	38
src/1174026.py	38

src/1174003/1174003.py	38
src/1174003/1174003.py	38
src/1174003/1174003.py	38
src/1174003/1174003.py	38
src/1174003/1174003.py	39
src/1174003/1174003.py	39
src/1174003/1174003.py	40
src/1174003/1174003.py	40
src/1174003/1174003.py	41
src/1174009/tugas.py	41
src/1174009/tugas.py	41
src/1174009/tugas.py	41
src/1174009/tugas.py	41
src/1174009/tugas.py	42
src/1174009/tugas.py	42
src/1174009/tugas.py	42
src/1174009/tugas.py	42
src/1174009/tugas.py	43
src/1174009/1174009.py	43
src/1174009/1174009.py	44
src/1174009/1174009.py	44
src/1174009/1174009.py	44
src/1174009/1174009.py	44
src/1174009/1174009.py	45
src/1174009/1174009.py	45
src/1174009/1174009.py	45
src/1174009/1174009.py	45
src/1174009/1174009.py	45
src/1174009/main.py	46
src/1174009/main.py	46
src/1174009/tugas.py	46
src/1174004.py	47
src/1174004.py	47
src/1174004.py	47
src/1174004.py	48
src/1174004.py	48
src/1174004.py	48
src/1174004.py	49

src/1174004.py	49
src/1174004.py	49
src/1174004.py	50
src/1174004.py	50
src/1174004.py	51
src/1174004.py	51
src/1174004.py	51
src/1174004.py	51
src/1174004.py	52
src/1174004.py	52
src/1174004.py	52
src/main.py	52
src/main.py	52
src/1174004.py	53
src/1174017.py	53
src/1174017.py	53
src/1174017.py	53
src/1174017.py	53
src/1174017.py	53
src/1174017.py	54
src/1174017.py	54
src/1174017.py	54
src/1174017.py	54
src/1174017.py	54
src/1174017.py	55
src/1174017.py	56
src/1174017.py	56
src/1174017.py	56
src/1174017.py	56
src/1174017.py	56
src/1174017.py	57
src/1174017.py	57
src/1174017.py	57
src/1174017.py	57
src/1174017.py	57
src/main.py	58
src/main.py	58
src/1174017.py	58
3.27 namaFungsi merupakan nama dari fungsi yang dibuat.	58
3.28 inputanFungsi merupakan nama dari inputan fungsi yang diterima dari luar fungsi namaFungsi.	59

3.29	return inputanFungsi merupakan kembalian fungsi dari fungsi namaFungsi.	59
3.30	Contoh penggunaan fungsi di Python.	59
3.31	Contoh penggunaan paket atau library.	59
3.32	Contoh penggunaan kelas di python.	59
3.33	Contoh penggunaan objek di python.	60
3.34	Contoh penggunaan atribut di python.	60
3.35	Contoh penggunaan method di python.	60
3.36	Contoh pemanggilan library kelas dari instansi dan pemakaiannya. .	60
3.37	Contoh pemakaian paket dengan perintah from kalkulator import Penambahan.	61
3.38	Contoh kode pemakaian paket fungsi dimana file library ada di dalam folder.	61
3.39	Contoh kode pemakaian paket kelas dimana file library ada di dalam folder.	61
3.40	Jawaban soal No. 1 Ketrampilan Pemrograman.	61
3.41	Jawaban soal No. 2 Ketrampilan Pemrograman.	62
3.42	Jawaban soal No. 3 Ketrampilan Pemrograman.	62
3.43	Jawaban soal No. 4 Ketrampilan Pemrograman.	63
3.44	Jawaban soal No. 5 Ketrampilan Pemrograman.	63
3.45	Jawaban soal No. 6 Ketrampilan Pemrograman.	63
3.46	Jawaban soal No. 7 Ketrampilan Pemrograman.	63
3.47	Jawaban soal No. 8 Ketrampilan Pemrograman.	64
3.48	Jawaban soal No. 9 Ketrampilan Pemrograman.	64
3.49	Jawaban soal No. 10 Ketrampilan Pemrograman.	64
3.50	Jawaban soal No. 11 Ketrampilan Pemrograman.	64
3.51	Jawaban soal No. 12 Ketrampilan Pemrograman.	65
3.52	Contoh fungsi yang menggunakan try except .	66
	src/1174025/1174025.py	66
	src/1174025/1174025.py	66
	src/1174025/1174025.py	67
	src/1174025/1174025.py	67
	src/1174025/1174025.py	67
	src/1174025/1174025.py	68
	src/1174025/1174025.py	68
	src/1174025/1174025.py	68
	src/1174025/1174025.py	68
3.53	Jawaban soal no 1.	69

3.54 Jawaban soal no 2.	70
3.55 Jawaban soal no 3.	70
3.56 Jawaban soal no 4.	70
3.57 Jawaban soal no 5.	70
3.58 Jawaban soal no 6.	70
3.59 Jawaban soal no 7.	71
3.60 Jawaban soal no 8.	71
3.61 Jawaban soal no 9.	71
3.62 Jawaban soal no 10.	71
3.63 Jawaban soal no 11.	72
3.64 Jawaban soal no 12.	72
3.65 Penggunaan Try Except .	73
src/1174012/1174012.py	73
src/1174012/1174012.py	73
src/1174012/1174012.py	73
src/1174012/1174012.py	74
src/1174012/1174012.py	74
src/1174012/1174012.py	74
src/1174012/1174012.py	75
src/1174012/1174012.py	75
src/1174012/1174012.py	75
src/1174012/1174012.py	75
src/1174012/1174012.py	76
src/1174012/1174012.py	76
src/1174012/1174012.py	77
src/1174012/1174012.py	77
src/1174012/1174012.py	77
src/1174012/1174012.py	77
src/1174012/1174012.py	77
src/1174012/1174012.py	77
src/1174012/main.py	78
src/1174012/main.py	78

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": "###  ",
11             "7": "      ##", "8": "###  ##", "9": "##  ##"}
12     angka3 = {"0": "###  ##", "1": "###", "2": "      ## ", "3": "      ## ",
13             "4": "      ##  ##", "5": "##### ", "6": "##### ",
14             "7": "      ##  ", "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.printNMPPenjumlahan(npm)
11 lib.printNMPPerkalian(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.printNMPPenjumlahan()
13 k3lib.printNMPPerkalian()

```

```

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

```

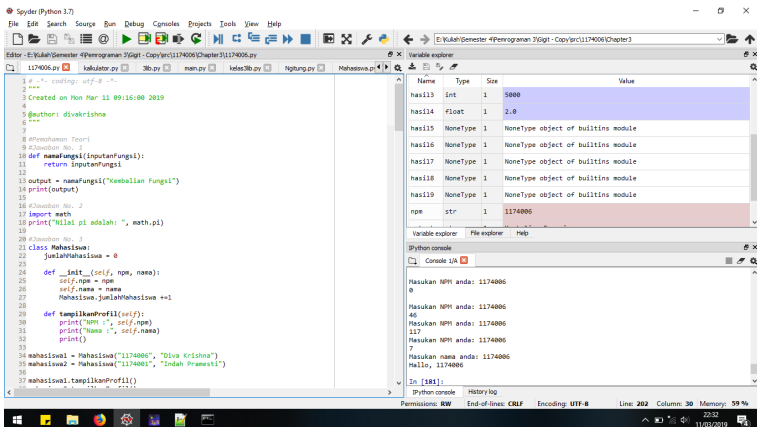
1 #Jawaban No. 1
2 def sapa(nama):
3     try:
4         print("Hallo , "+str(nama))
5     except:
6         print("Terjadi error")
7
8 sapa(input("Masukan nama anda: "))

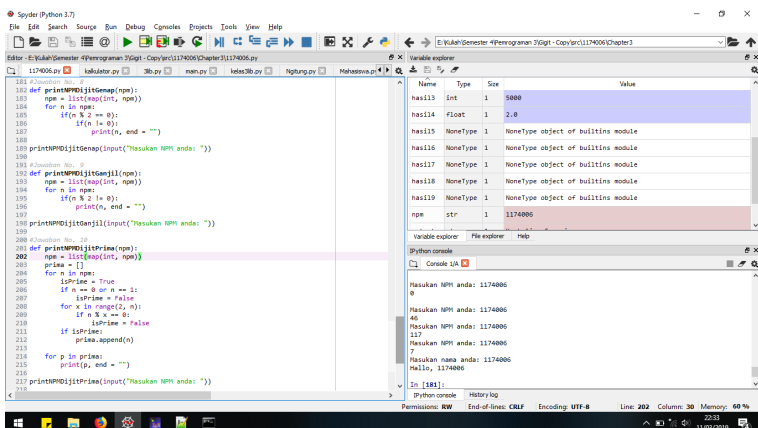
```

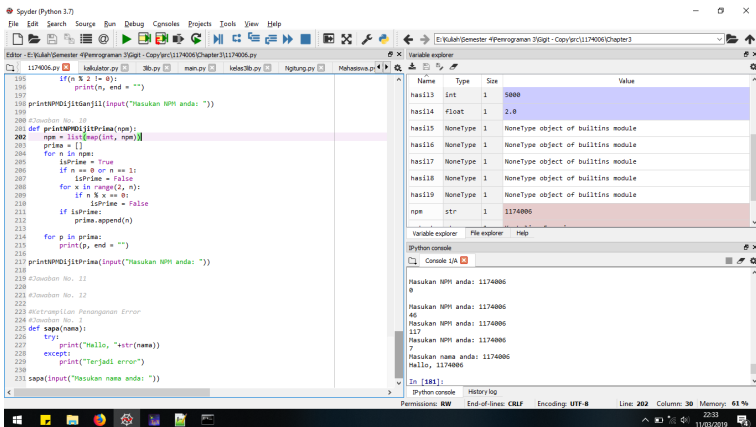
Listing 3.26 Contoh fungsi yang menggunakan try except .

3.3.4 Hasil Screenshoot

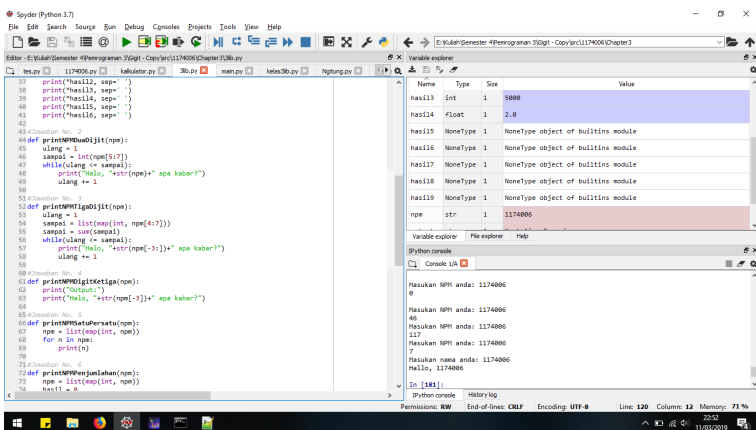
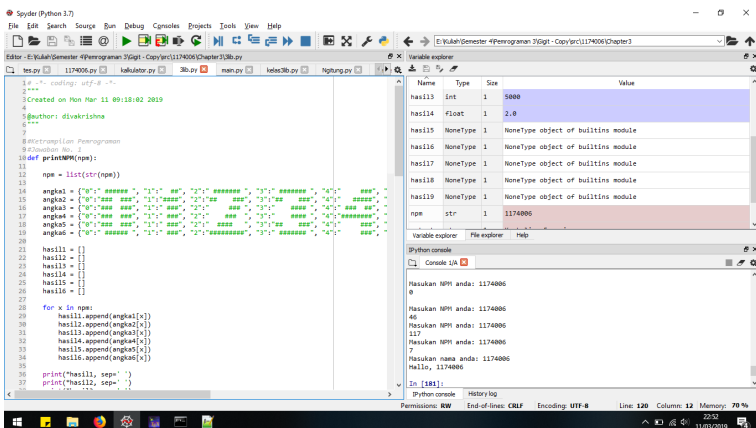
1. Screenshoot Kodingan 1174006.py

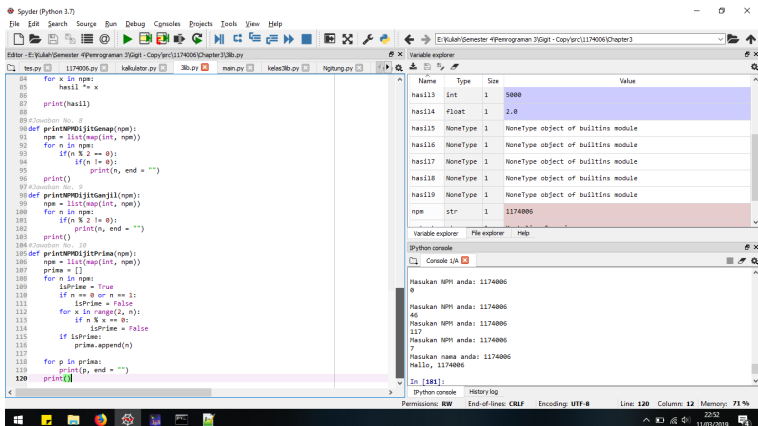
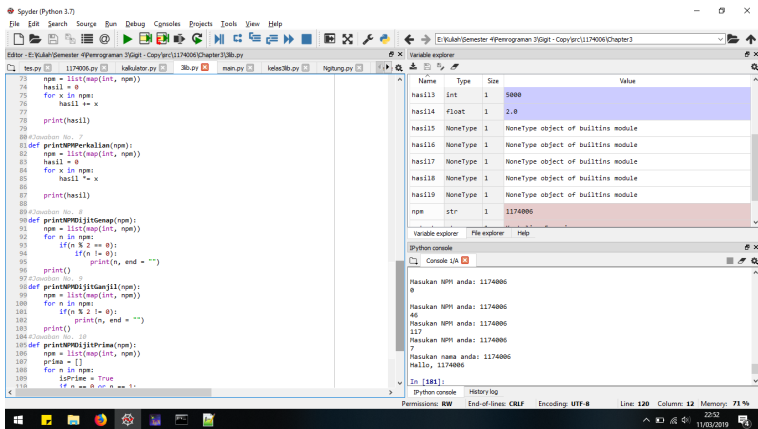




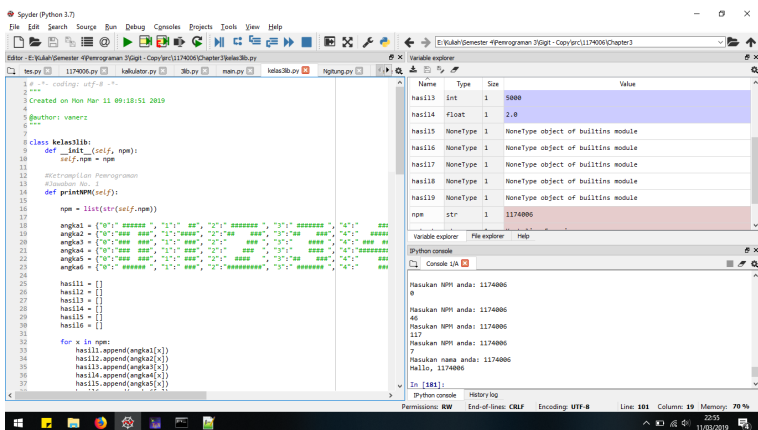


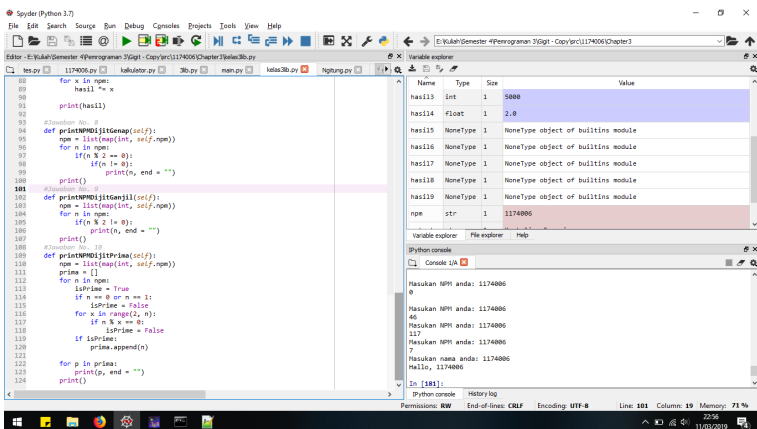
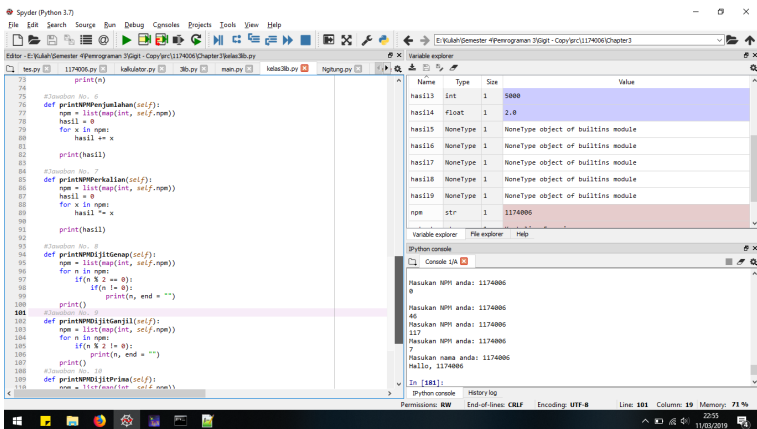
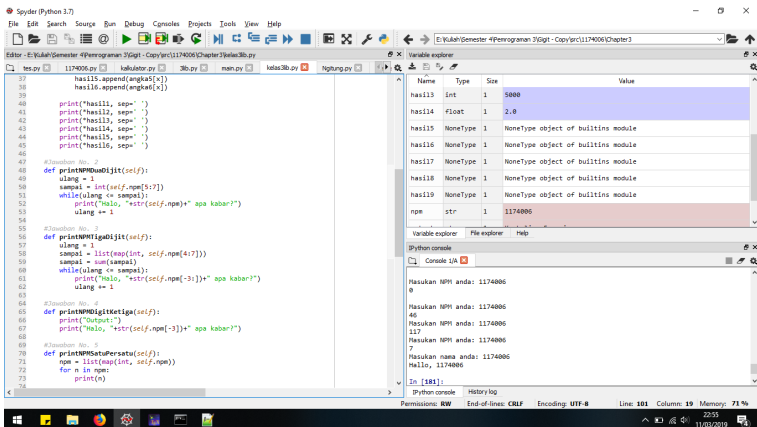
2. Screenshoot Kodingan 3lib.py

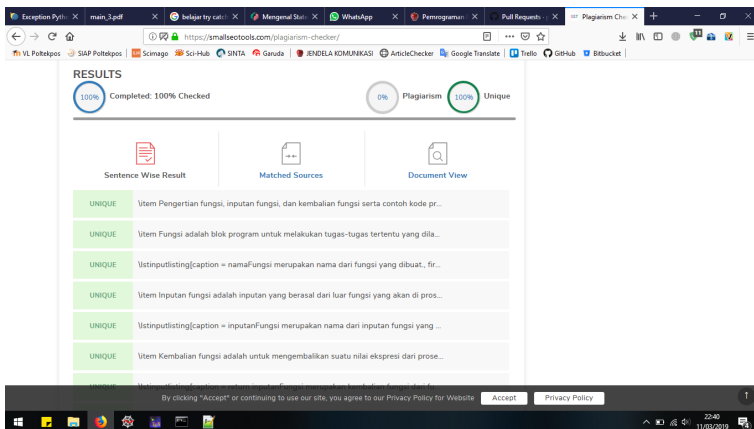




3. Screenshoot Kodingan kelas3lib.py







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 mepresentasikan objek. objek adalah hasil cetakan dadri sebuah kelas. method adalah suatu upaya yang digunakan oleh objek.

```
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 #empl = Employee("Zara", 2000)
19 #This would create second object of Employee class"
20 #emp2 = Employee("Manni", 5000)
21 #empl.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 Ketramampilan 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": "
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    angka7 = {"0": " ***** ", "1": " *** ", "2": " ***** ", "3": "
25    ***** ", "4": " ***** ", "5": " ***** ", "6": " ***** ", "7": "
26    ***** ", "8": " ***** ", "9": " ***** "}
27    angka8 = {"0": " ***** ", "1": " *** ", "2": " ***** ", "3": "
28    ***** ", "4": " ***** ", "5": " ***** ", "6": " ***** ", "7": "
29    ***** ", "8": " ***** ", "9": " ***** "}
30    angka9 = {"0": " ***** ", "1": " *** ", "2": " ***** ", "3": "
31    ***** ", "4": " ***** ", "5": " ***** ", "6": " ***** ", "7": "
32    ***** ", "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  speak(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 dijit 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 dijit 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 dijit yang termasuk bilangan prima saja. wajib menggunakan perulangan dan atau kondisi.

```

1
2 odd()
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")
```

3.5 Dwi Septiani Tsaniyah

1. Apa itu fungsi, inputan fungsi dan kembalian fungsi dengan contoh kode program lainnya. Fungsi memiliki tujuan agar kita dapat memecah program besar menjadi sub-sub program yang lebih sederhana. Pada saat kita membutuhkan suatu fitur maka kita tinggal memanggil fungsi yang telah kita buat. Fungsi pada python dibuat dengan kata kunci def dan diikuti dengan nama fungsi yang kita buat seperti contoh dibawah :

```
1 #No.1
```

Inputan fungsi merupakan masukan yang kita berikan pada program dan program akan menampilkan hasil dari inputan yang kita masukkan. contoh dari inputan fungsi sebagai berikut :

```
1     npm = input("Masukan NPM :")
2     npm = list(str(npm))
```

Pengembalian fungsi memiliki tujuan untuk mengembalikan nilai dari hasil yang telah di proses. Dalam hal ini menggunakan kata kunci return yang diikuti dengan nilai atau variabel yang akan dikembalikan.

```
1 def jawabanNo1() :
```

2. Apa itu paket dan cara pemanggilan paket atau library dengan contoh kode program lainnya. Library atau paket adalah modul-modul yang menyusun python. Modul-modul tersebut ditulis oleh berbagai orang dari seluruh dunia dan memiliki fungsi masing-masing untuk melakukan suatu hal. contoh kode programnya adalah sebagai berikut : from bte import penulisan print (penulisan(int(input("Masukan NPM :"))))

```
1 def jawabanNo1() :
2
3     npm = input("Masukan 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": "##### "}

```

3. Jelaskan Apa itu kelas, apa itu objek, apa itu atribut, apa itu method dan contoh kode program lainnya masing-masing. kelas adalah Prototype yang ditentukan oleh pengguna untuk objek yang mendefinisikan seperangkat atribut yang menjadi ciri objek kelas apa pun. Objek ialah instansiasi atau perwujudan dari sebuah kelas. Bila kelas adalah prototipenya, dan objek adalah barang jadinya. Attribute adalah menggambarkan data yang dapat memberikan informasi kelas atau objek dimana attribut tersebut berada.

```

1 angka5 = {"0": "### ##", "1": " ##", "2": " ### ", "3": " ##", "4": " ##", "5": " ##", "6": "### ##", "7": " ##", "8": " ##", "9": " ## "}
2 angka6 = {"0": "##### ", "1": " ##", "2": "##### ", "3": "##### ", "4": " ##", "5": "##### ", "6": "##### ", "7": "### ", "8": " #####", "9": "##### "}
3
4 hasil1 = []
5 hasil2 = []
6 hasil3 = []
7 hasil4 = []
8 hasil5 = []
9 hasil6 = []
10
11 for x in npm:

```

4. Jelaskan cara pemanggilan library kelas dari instansiasi dan pemakaiannya dengan contoh program lainnya. cara pemanggilan library kelas dari instansiasi dan pemakaiannya adalah dengan cara meng-import library yang ada di dalam satu folder dan menggunakan kode berikut :

```

1 hasil1.append(angka1[x])
2 hasil2.append(angka2[x])
3 hasil3.append(angka3[x])
4 hasil4.append(angka4[x])
5 hasil5.append(angka5[x])

```

```

6         hasil6.append(angka6[x])
7
8     print(*hasil1, sep=' ')
9     print(*hasil2, sep=' ')
10    print(*hasil3, sep=' ')
11    print(*hasil4, sep=' ')

```

5. Jelaskan dengan contoh pemakaian paket dengan perintah from kalkulator import Penambahan disertai dengan contoh kode lainnya. contoh kodenya adalah sebagai berikut :

```

1         print("Halo, " + x + " Apa Kabar?")
2         hitung = hitung +1
3     ulang_3_digit_terakhir(int(input("Masukkan NPM: ")))
4
5 #no.4
6 def digit_3_dari_belakang(npm):
7     npm = str(npm)
8     x = npm[-3]
9     print("Halo, " + x + " Apa Kabar?")
10    digit_3_dari_belakang(int(input("Masukkan NPM: ")))
11
12 #no5
13 def bawah(npm):
14     for i in npm:
15         print(i)
16
17    bawah(input("Masukkan NPM: "))
18
19 #no.6
20 def tambahkan(npm):
21     jumlah = 0
22     for i in npm:
23         jumlah += int(i)
24     print("Hasil Penjumlahan NPM adalah : " +str(jumlah))
25    tambahkan(input("Masukkan NPM: "))

```

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 :

```

1         kalikan *= int(i)
2         print("Hasil Perkalian NPM adalah : " +str(kalikan))
3     kalikan(input("Masukkan NPM: "))
4
5 #N0.8
6 def digit_genap(npm):
7     npm = list(map(int, npm))
8     for n in npm:
9         if(n % 2 == 0):
10            if(n != 0):
11                print(n, end = "")

```

7. Jelaskan dengan contoh kodenya, pemakaian paket kelas apabila le library ada di dalam folder.

```

1 #Keterampilan Penanganan error
2 #No.1
3 def halo(nama):
4     try:
5         print("Hallo , "+str(nama))
6     except:
7         print("Ada Yang Error")
8
9 halo(input("Cantumkan Nama Anda : "))

```

3.6 Dwi Yulianingsih

3.6.1 Pemahaman Teori

1. Apa itu fungsi, inputan fungsi dan kembalian fungsi dengan contoh kode program lainnya. Fungsi memiliki tujuan agar kita dapat memecah program besar menjadi sub-sub program yang lebih sederhana. Pada saat kita membutuhkan suatu fitur maka kita tinggal memanggil fungsi yang telah kita buat. Fungsi pada python dibuat dengan kata kunci def dan diikuti dengan nama fungsi yang kita buat seperti contoh dibawah :

```

1 def fungsi(x):

```

Inputan fungsi merupakan masukan yang kita berikan pada program dan program akan menampilkan hasil dari inputan yang kita masukkan. contoh dari inputan fungsi sebagai berikut :

```

1 print(fungsi(3))
2 print(fungsi(5))
3 print(fungsi(9))

```

Pengembalian fungsi memiliki tujuan untuk mengembalikan nilai dari hasil yang telah di proses. Dalam hal ini menggunakan kata kunci return yang diikuti dengan nilai atau variabel yang akan dikembalikan.

```

1     return 5 * x

```

2. Apa itu paket dan cara pemanggilan paket atau library dengan contoh kode program lainnya. Library atau paket adalah modul-modul yang menyusun python. Modul-modul tersebut ditulis oleh berbagai orang dari seluruh dunia dan memiliki fungsi masing-masing untuk melakukan suatu hal. contoh kode programnya adalah sebagai berikut :

```

1 #no2
2 import mymodule
3
4 a = mymodule.person1["age"]
5 print(a)

```

3. Jelaskan Apa itu kelas, apa itu objek, apa itu atribut, apa itu method dan contoh kode program lainnya masing-masing. kelas adalah Prototype yang ditentukan oleh pengguna untuk objek yang mendefinisikan seperangkat atribut yang menjadi ciri objek kelas apa pun. Objek ialah instansiasi atau perwujudan dari sebuah kelas. Bila kelas adalah prototipenya, dan objek adalah barang jadinya. Atribut adalah data anggota (variabel kelas dan variabel contoh) dan metode, diakses melalui notasi titik. Sedangkan method fungsi yang didefinisikan di dalam suatu kelas.

```

1 class Person:
2     def __init__(self, name, age):
3         self.name = name
4         self.age = age
5
6     def myfunc(self):
7         print("Hello my name is " + self.name)
8
9 p1 = Person("uwi", 19)
10 p1.myfunc()

```

4. Jelaskan cara pemanggilan library kelas dari instansiasi dan pemakaiannya dengan contoh program lainnya. cara pemanggilan library kelas dari instansiasi dan pemakaiannya adalah dengan cara meng-import library yang ada di dalam satu folder dan menggunakan kode berikut :

```

1 #no4
2 #import mymodule
3 #c = 300
4 #d = 10
5
6 #e = belajar.pengurangan(c,d)
7 #print (e)

```

5. Jelaskan dengan contoh pemakaian paket dengan perintah from kalkulator import Penambahan disertai dengan contoh kode lainnya. Penggunaan paket berfungsi untuk memanggil file dan fungsinya. contoh kodenya adalah sebagai berikut :

```

1 #no5
2 #from mymodule import *
3 #print (penulisan(int(input("write your NPM"))))

```

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 :

```

1 #no6
2 from folder import kalkulator
3
4 a = 100
5 b = 50
6
7 hasil=kalkulator.Penambahan(a,b)

```

```

8 hasil2=kalkulator.Pengurangan(a,b)
9 hasil3=kalkulator.Perkalian(a,b)
10 hasil4=kalkulator.Pembagian(a,b)

```

7. Jelaskan dengan contoh kodenya, pemakaian paket kelas apabila le library ada di dalam folder.

```

1 #no7
2 #from folder.Mahasiswa import Mahasiswa
3
4 #mhs1.tampilkanprofil()
5 #mhs2.tampilkanprofil()
6 #print("totalnya adalah", Mahasiswa.jumlahMahasiswa)

```

3.6.2 Praktek

1. jawaban no 1

```

1 #No.1
2 def jawabanNo1():
3
4     npm = input("Masukan NPM :")
5     npm = list(str(npm))
6
7     angka1 = {"0": "##### ", "1": " ##", "2": "##### ", "3": "##### ",
8               "4": " #####", "5": "#####", "6": "##### ",
9               "7": "#####", "8": " ##### ", "9": "##### "}
10    angka2 = {"0": "### ##", "1": "####", "2": "##   ##", "3": "###   ##",
11              "4": " ## ##", "5": "##   ", "6": "###   ",
12              "7": "#####", "8": "##  ##", "9": "##   ## "}
13    angka3 = {"0": "### ##", "1": " ##", "2": "   ## ", "3": "   ## ",
14              "4": " ##  ##", "5": "##   ", "6": "###   ",
15              "7": "   ## ", "8": " ##### ", "9": "##   ## "}
16    angka4 = {"0": "### ##", "1": " ##", "2": "   ## ", "3": "   ## ",
17              "4": "#####", "5": "##### ", "6": "##### ",
18              "7": "   ## ", "8": "#####", "9": "##### "}
19    angka5 = {"0": "### ##", "1": " ##", "2": "   ## ", "3": "   ## ",
20              "4": "   ## ", "5": "   ## ", "6": "### ##",
21              "7": "   ## ", "8": "##  ##", "9": "   ## "}
22    angka6 = {"0": "##### ", "1": " ##", "2": "##### ", "3": "##### ",
23              "4": "   ##", "5": "##### ", "6": " ##### ",
24              "7": "###   ", "8": " ##### ", "9": "##### "}
25
26    hasil1 = []
27    hasil2 = []
28    hasil3 = []
29    hasil4 = []
30    hasil5 = []
31    hasil6 = []
32
33    for x in npm:
34        hasil1.append(angka1[x])
35        hasil2.append(angka2[x])
36        hasil3.append(angka3[x])
37        hasil4.append(angka4[x])
38        hasil5.append(angka5[x])
39        hasil6.append(angka6[x])

```



```

26     hasil5.append(angka5[x])
27     hasil6.append(angka6[x])
28
29     print(*hasil1, sep=' ')
30     print(*hasil2, sep=' ')
31     print(*hasil3, sep=' ')
32     print(*hasil4, sep=' ')
33     print(*hasil5, sep=' ')
34     print(*hasil6, sep=' ')
35
36 jawabanNo1()

```

2. jawaban no 2

```

1 #no.2
2 def mbaleni(npm):
3     hitung = 0
4     while(hitung <9):
5         print("Halo, " + str(npm) + " Apa Kabar?")
6         hitung = hitung +1
7 mbaleni(int(input("Masukkan NPM: ")))

```

3. jawaban no 3

```

1 #no.3
2 def mbaleni3digitakhir(npm):
3     hitung = 0
4     npm = str(npm)
5     x = npm[4:7]
6     while(hitung < 9):
7         print("Halo, " + x + " Apa Kabar?")
8         hitung = hitung +1
9 mbaleni3digitakhir(int(input("Masukkan NPM: ")))

```

4. jawaban no 4

```

1 #no.4
2 def digit3daribelakang(npm):
3     npm = str(npm)
4     x = npm[-3]
5     print("Halo, " + x + " Apa Kabar?")
6 digit3daribelakang(int(input("Masukkan NPM: ")))

```

5. jawaban no 5

```

1 #no5
2 def bawah(npm):
3     for i in npm:
4         print(i)
5
6 bawah(input("Masukkan NPM: "))

```

6. jawaban no 6

```

1 #no.6
2 def plus(npm):
3     jumlah = 0
4     for i in npm:
5         jumlah += int(i)
6     print ("Hasil Penjumlahan NPM adalah : " +str(jumlah))
7 plus(input("Masukkan NPM: "))

```

7. jawaban no 7

```

1 #No.7
2 def kali(npm):
3     kalikan = 0
4     for i in npm:
5         kalikan *= int(i)
6     print ("Hasil Perkalian NPM adalah : " +str(kalikan))
7 kali(input("Masukkan NPM: "))

```

8. jawaban no 8

```

1 #NO.8
2 def genap(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 genap(input("Masukkan NPM: "))

```

9. jawaban no 9

```

1 def ganjil(npm):
2     npm = list(map(int , npm))
3     for n in npm:
4         if(n % 2 !=0):
5             print(n, end = "")
6 ganjil(input("Masukkan NPM: "))

```

10. jawaban no 10

```

1 #No.10
2 def prima(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 prima(input("Masukkan NPM: "))

```

11. jawaban 11

```

1 lib = __import__('3lib')
2 npm = "1174009"
3
4 hasil1 = lib.jawabanNo1(npm)
5 hasil2 = lib.jawabanNo2(npm)
6 hasil3 = lib.jawabanNo3(npm)
7 hasil4 = lib.jawabanNo4(npm)
8 hasil5 = lib.jawabanNo5(npm)
9 hasil6 = lib.jawabanNo6(npm)
10 hasil7 = lib.jawabanNo7(npm)
11 hasil8 = lib.jawabanNo8(npm)
12 hasil9 = lib.jawabanNo9(npm)
13 hasil10 = lib.jawabanNo10(npm)
14 print()

```

12. jawaban 12

```

1 from kelas3lib import kelas3lib
2
3 npm = "1174009"
4
5 kel3lib = kelas3lib(npm)
6
7 kel3lib.jawabanNo1()
8 kel3lib.jawabanNo2()
9 kel3lib.jawabanNo3()
10 kel3lib.jawabanNo4()
11 kel3lib.jawabanNo5()
12 kel3lib.jawabanNo6()
13 kel3lib.jawabanNo7()
14 kel3lib.jawabanNo8()
15 kel3lib.jawabanNo9()
16 kel3lib.jawabanNo10()

```

3.6.3 Penanganan Error

untuk menangani error yang terjadi bisa menggunakan source kode dibawah ini :

```

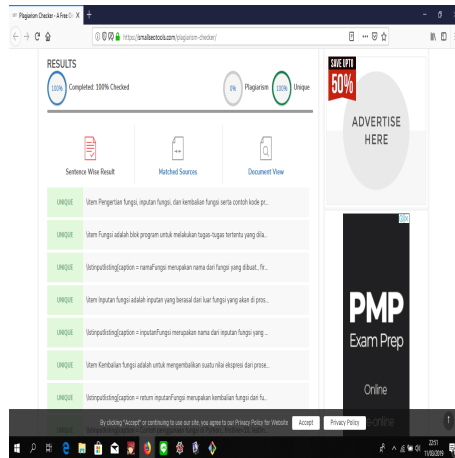
1 #penanganan Error
2 #def penanganan_error(c,d):
3 #     try :
4 #         c = c+d
5 #         print(e)
6 #     except TypeError:
7 #         print("beda ya gais")

```

3.7 Choirul Anam

3.7.0.1 Pemahaman Teori

1. Apa itu fungsi, inputan



Gambar 3.1 SS Bebas Plagiarisme

fungsi dan kembalian fungsi dengan contoh kode program lainnya. Fungsi adalah suatu perintah diaman perintah tersebut dapat di panggil berulang ulang. Berikut merupakan contoh fungsi dan cara pemanggilannya

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

Fungsi juga bisa membaca parameter, dimana parameter adalah nilai yang di inputkan atau di lemparkan untuk kebutuhan 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_choi 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. atribut adalah apa saja yang dapat dilakukan oleh objek

```

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

- import file.
- 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_choi 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

```

3.7.0.2 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 #No 1
2 def penulisan(npm):
3     npm = list(str(npm))
4
5     angka1 = {"0": " ##### ", "1": " ### ", "2": " ##### ", "3": "
6     ##### ", "4": " ###   ### ", "5": " ##### ", "6": " ##### ", "7": "
7     ##### ", "8": " ##### ", "9": " ##### "}
8     angka2 = {"0": " ##### ", "1": " ### ", "2": " ##### ", "3": "
9     ##### ", "4": " ###   ### ", "5": " ##### ", "6": " ##### ", "7": "
10    ##### ", "8": " ##### ", "9": " ##### "}
11    angka3 = {"0": " ##   ## ", "1": " ### ", "2": "   ## ", "3": "
12    ##### ", "4": "   ##   ## ", "5": " ##### ", "6": " ##### ", "7": "
13    ##### ", "8": " ##### ", "9": " ##### "}
14    angka4 = {"0": " ##   ## ", "1": " ### ", "2": "   ## ", "3": "
15    ##### ", "4": "   ##   ### ", "5": " ##### ", "6": " ##### ", "7": "
16    ##### ", "8": " ##### ", "9": " ##### "}
17    angka5 = {"0": " ##   ## ", "1": " ### ", "2": "   ## ", "3": "
18    ##### ", "4": "   ##### ", "5": " ##### ", "6": " ##### ", "7": "
19    ##### ", "8": " ##### ", "9": " ##### "}
20    angka6 = {"0": " ##   ## ", "1": " ### ", "2": "   ## ", "3": "
21    ##### ", "4": "   ##   ## ", "5": " ##### ", "6": " ##### ", "7": "
22    ##### ", "8": " ##### ", "9": " ##### "}
23    angka7 = {"0": " ##   ## ", "1": " ### ", "2": "   ## ", "3": "
24    ##### ", "4": "   ##   ## ", "5": " ##### ", "6": " ##### ", "7": "
25    ##### ", "8": " ##### ", "9": " ##### "}
26    angka8 = {"0": " ##   ## ", "1": " ### ", "2": "   ## ", "3": "
27    ##### ", "4": "   ##   ## ", "5": " ##### ", "6": " ##### ", "7": "
28    ##### ", "8": " ##### ", "9": " ##### "}
29    angka9 = {"0": " ##### ", "1": " ### ", "2": "   ## ", "3": "
30    ##### ", "4": "   ##   ## ", "5": " ##### ", "6": " ##### ", "7": "
31    ##### ", "8": " ##### ", "9": " ##### "}
32    angka10 = {"0": " ##### ", "1": " ### ", "2": "   ## ", "3": "
33    ##### ", "4": "   ##   ## ", "5": " ##### ", "6": " ##### ", "7": "
34    ##### ", "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=' ')
49
50     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 def perulangan_3_digit(npm):
2     hitung = 0

```

```

3     npm = str(npm)
4     bil = npm[4:7]
5
6     while(hitung < 9):
7         print("Halo, "+bil+" apa kabar?")
8         hitung = hitung +1
9
10    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    def perulangan_3_digit_terakhir(npm):
2        npm = str(npm)
3        bil = npm[-3]
4        print("Halo, "+bil+" apa kabar?")
5
6    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    def down(npm):
2        for i in npm:
3            print (i)
4
5    down(input("Masukan NPM : "))

```

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

```

1    def penjumlahan(npm):
2        jumlah = 0
3        for i in npm:
4            jumlah += int(i)
5        print(str(jumlah)+" Adalah hasil penjumlahan")
6
7    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    def perkalian(npm):
2        jumlah = 0
3        for i in npm:
4            jumlah *= int(i)
5        print(str(jumlah)+" Adalah hasil perkalian")
6
7    perkalian(input("Masukan NPM : "))

```


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

```

1 def genap():
2     npm = [1,1,7,4,0,2,7]
3     for i in npm:
4         if (i % 2) == 0:
5             print("Bilangan Genapnya : "+str(i))
6 genap()

```

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

```

1 def ganjil():
2     npm = [1,1,7,4,0,2,7]
3     for i in npm:
4         if (i%2)==1:
5             print("Bilangan Ganjilnya : "+str(i))
6 ganjil()

```

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

```

1 def prima(npm):
2     npm = str(npm)
3     bil = npm[2]
4     num = int(bil)
5     if num > 1:
6         for i in range(2,num):
7             if (num%i)==0:
8                 print("Bukan Bilangan Prima")
9                 break
10            else:
11                print("Bilangan Primanya :"+str(num))
12    else:
13        print("Tidak Ada Bilangan Prima")
14    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

```

3.7.0.3 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 except:
2     print("mas choi ganteng")
3 finally:
4     print("memang ganteng")
```

3.8 Muh. Rifky Prananda

3.8.0.1 Pemahaman Teori

1. Apa itu fungsi, inputan

fungsi dan kembalian fungsi dengan contoh kode program lainnya. Fungsi adalah salah satu 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 bisa membaca parameter, parameter yaitu sebuah nilai yang tersedia untuk fungsi, dimana pada nilai ini menentukan output yang dihasilkan fungsi.

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

Statemen return dapat digunakan untuk bisa 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 bisa lebih memudahkan dalam pemanggilan fungsi yang dibutuhkan, agar dapat dipanggil berulang-ulang. 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 adalah sebuah blueprint yang mempresentasikan suatu objek. objek yaitu hasil cetakan dari sebuah kelas itu sendiri. method adalah sifat atau perilaku dari object.

```
1 #class Employee:
2 #    'Common base class for all employees'
3 #    empCount = 0
```

4. Jelaskan cara pemanggilan library kelas dari instansiasi dan pemakaiannya dengan contoh program lainnya. Cara Pemanggilanya

- yang pertama import terlebih dahulu filenya.
- selanjutnya buat variabel untuk dapat menampung datanya
- kemudian panggil nama classnya dan panggil methodnya
- Gunakan suatu 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 dapat memanggil file dan fungsinya

```
1 #No 1
2 def penulisan(npm):
```

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

7. Jelaskan dengan contoh kodenya, pemakaian paket kelas apabila le library ada di dalam folder.

```
1 print(*hasil4 , sep=' ')
```

3.8.0.2 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 #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=' ')

```

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

```

1 #No 2
2 def perulangan(npm):
3     hitung = 0
4     while(hitung < 17):
5         print("Halo, "+str(npm)+" apa kabar?")
6         hitung = hitung +1

```

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

```

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?")

```

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. 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")

```

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

```

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")

```

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

```

3.8.0.3 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")

```

3.9 Habib Abdul Rasyid

3.9.1 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 3.27 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 3.28 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 3.29 return inputanFungsi merupakan kembalian fungsi dari fungsi namaFungsi.

Penggunaan fungsi di Python

```
1 def namaFungsi(inputanFungsi):
2     return inputanFungsi
3
4 output = namaFungsi("Kembalian Fungsi")
5 print(output)
```

Listing 3.30 Contoh penggunaan fungsi di Python.

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 import math
2 print("Nilai pi adalah: ", math.pi)
```

Listing 3.31 Contoh penggunaan paket atau library.

2. 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 class Mahasiswa:
2     jumlahMahasiswa = 0
3
4     def __init__(self, npm, nama):
5         self.npm = npm
6         self.nama = nama
7         Mahasiswa.jumlahMahasiswa +=1
8
9     def tampilkanProfil(self):
10        print("NPM :", self.npm)
11        print("Nama :", self.nama)
12        print()
```



```

13
14 mahasiswa1 = Mahasiswa("1174002", "Habib Abdul Rasyid")
15 mahasiswa2 = Mahasiswa("1174099", "Abdul Rasyid")
16
17 mahasiswa1.tampilkanProfil()
18 mahasiswa2.tampilkanProfil()
19
20 print("Total mahasiswa adalah ", Mahasiswa.jumlahMahasiswa)

```

Listing 3.32 Contoh penggunaan kelas di python.

- Objek Objek adalah instansi atau perwujudan dari sebuah kelas.

```

1 mahasiswa1 = Mahasiswa("1174002", "Habib Abdul Rasyid")
2 mahasiswa2 = Mahasiswa("1174099", "Abdul Rasyid")

```

Listing 3.33 Contoh penggunaan objek di python.

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

```

1 jumlahMahasiswa = 0

```

Listing 3.34 Contoh penggunaan atribut di python.

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

```

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

```

Listing 3.35 Contoh penggunaan method di python.

3. 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 from Mahasiswa import Mahasiswa
2
3 mhs1 = Mahasiswa("1174002", "Habib Abdul Rasyid")
4 mhs2 = Mahasiswa("1174099", "Abdul")
5
6 mhs1.tampilkanProfil()
7 mhs2.tampilkanProfil()
8
9 print("Total mahasiswa adalah ", Mahasiswa.jumlahMahasiswa)

```

Listing 3.36 Contoh pemanggilan library kelas dari instansi dan pemakaiannya. .

4. 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 from kalkulator import Penambahan
2
3 hasil = Penambahan(10, 5)
4 print(hasil)

```

Listing 3.37 Contoh pemakaian paket dengan perintah from kalkulator import Penambahan.

5. 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 from folder import kalkulator
2
3 a=50
4 b=25
5
6 hasil1=kalkulator.Penambahan(a,b)
7 hasil2=kalkulator.Pengurangan(a,b)
8 hasil3=kalkulator.Perkalian(a,b)
9 hasil4=kalkulator.Pembagian(a,b)
10
11 print(hasil1)
12 print(hasil2)
13 print(hasil3)
14 print(hasil4)

```

Listing 3.38 Contoh kode pemakaian paket fungsi dimana file library ada di dalam folder.

6. 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 from folder.Mahasiswa import Mahasiswa
2
3 mhs1 = Mahasiswa("1174002", "Habib Abdul Rasyid")
4 mhs2 = Mahasiswa("1174099", "Abdul")
5
6 mhs1.tampilkanProfil()
7 mhs2.tampilkanProfil()
8
9 print("Total mahasiswa adalah ", Mahasiswa.jumlahMahasiswa)

```

Listing 3.39 Contoh kode pemakaian paket kelas dimana file library ada di dalam folder.

3.9.2 Ketrampilan Pemrograman

1. Jawaban soal No. 1

```

1 def jawabanNo1():
2
3     npm = input("Masukan 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": "****",
8             "4": "****", "5": "****", "6": "****", "7": "****", "8": "****", "9": "****"}
9     angka3 = {"0": "****", "1": "****", "2": "****", "3": "****",
10            "4": "****", "5": "****", "6": "****", "7": "****", "8": "****", "9": "****"}
11    angka4 = {"0": "****", "1": "****", "2": "****", "3": "****",
12            "4": "****", "5": "****", "6": "****", "7": "****", "8": "****", "9": "****"}
13    angka5 = {"0": "****", "1": "****", "2": "****", "3": "****",
14            "4": "****", "5": "****", "6": "****", "7": "****", "8": "****", "9": "****"}
15    angka6 = {"0": "****", "1": "****", "2": "****", "3": "****",
16            "4": "****", "5": "****", "6": "****", "7": "****", "8": "****", "9": "****"}
17
18    hasil1 = []
19    hasil2 = []
20    hasil3 = []
21    hasil4 = []
22    hasil5 = []
23    hasil6 = []
24
25    for x in npm:
26        hasil1.append(angka1[x])
27        hasil2.append(angka2[x])
28        hasil3.append(angka3[x])
29        hasil4.append(angka4[x])
30        hasil5.append(angka5[x])
31        hasil6.append(angka6[x])
32
33    print(*hasil1, sep=' ')
34    print(*hasil2, sep=' ')
35    print(*hasil3, sep=' ')
36    print(*hasil4, sep=' ')
37    print(*hasil5, sep=' ')
38    print(*hasil6, sep=' ')
39
40    jawabanNo1()

```

Listing 3.40 Jawaban soal No. 1 Ketrampilan Pemrograman.

2. Jawaban soal No. 2

```

1 def jawabanNo2(npm):
2     hitung = 0
3     while(hitung < 2):
4         print("Hello, " + str(npm) + " Apa Kabar?")
5         hitung = hitung +1
6     jawabanNo2(int(input("Masukkan NPM: ")))

```

Listing 3.41 Jawaban soal No. 2 Ketrampilan Pemrograman.

3. Jawaban soal No. 3

```

1 def jawabanNo3(npm):

```

```

2     hitung = 0
3     npm = str(npm)
4     x = npm[4:7]
5     while(hitung < 2):
6         print("Hello, " + x + " Apa Kabar?")
7         hitung = hitung +1
8 jawabanNo3(int(input("Masukkan NPM: ")))

```

Listing 3.42 Jawaban soal No. 3 Ketrampilan Pemrograman.

4. Jawaban soal No. 4

```

1 def jawabanNo4(npm):
2     npm = str(npm)
3     x = npm[-3]
4     print("Hello, " + x + " Apa Kabar?")
5 jawabanNo4(int(input("Masukkan NPM: ")))

```

Listing 3.43 Jawaban soal No. 4 Ketrampilan Pemrograman.

5. Jawaban soal No. 5

```

1 def jawabanNo5(npm):
2     for i in npm:
3         print(i)
4
5 jawabanNo5(input("Masukkan NPM: "))

```

Listing 3.44 Jawaban soal No. 5 Ketrampilan Pemrograman.

6. Jawaban soal No. 6

```

1 def jawabanNo6(npm):
2     jumlah = 0
3     for i in npm:
4         jumlah += int(i)
5     print("Hasil Penjumlahan NPM adalah : " +str(jumlah))
6 jawabanNo6(input("Masukkan NPM: "))

```

Listing 3.45 Jawaban soal No. 6 Ketrampilan Pemrograman.

7. Jawaban soal No. 7

```

1 def jawabanNo7(npm):
2     kalikan = 0
3     for i in npm:
4         kalikan *= int(i)
5     print("Hasil Perkalian NPM adalah : " +str(kalikan))
6 jawabanNo7(input("Masukkan NPM: "))

```

Listing 3.46 Jawaban soal No. 7 Ketrampilan Pemrograman.

8. Jawaban soal No. 8

```

1 def jawabanNo8(npm):
2     npm = list(map(int, npm))
3     for n in npm:
4         if(n % 2 == 0):
5             if(n != 0):
6                 print(n, end = "")
7 jawabanNo8(input("Masukkan NPM: "))

```

Listing 3.47 Jawaban soal No. 8 Ketrampilan Pemrograman.

9. Jawaban soal No. 9

```

1 def jawabanNo9(npm):
2     npm = list(map(int, npm))
3     for n in npm:
4         if(n % 2 != 0):
5             print(n, end = "")
6 jawabanNo9(input("Masukkan NPM: "))

```

Listing 3.48 Jawaban soal No. 9 Ketrampilan Pemrograman.

10. Jawaban soal No. 10

```

1 def jawabanNo10(npm):
2     npm = list(map(int, npm))
3     prima = []
4     for n in npm:
5         bilPrima = True
6         if n == 0 or n == 1:
7             bilPrima = False
8         for x in range(2, n):
9             if n % x == 0:
10                bilPrima = False
11        if bilPrima:
12            prima.append(n)
13
14    for p in prima:
15        print(p, end = "")
16 jawabanNo10(input("Masukkan NPM: "))

```

Listing 3.49 Jawaban soal No. 10 Ketrampilan Pemrograman.

11. Jawaban soal No. 11

```

1 lib = __import__('3lib')
2
3 npm = "1174002"
4
5 hasil1 = lib.jawabanNo1(npm)
6 hasil2 = lib.jawabanNo2(npm)
7 hasil3 = lib.jawabanNo3(npm)
8 hasil4 = lib.jawabanNo4(npm)
9 hasil5 = lib.jawabanNo5(npm)
10 hasil6 = lib.jawabanNo6(npm)
11 hasil7 = lib.jawabanNo7(npm)
12 hasil8 = lib.jawabanNo8(npm)

```

```

13 hasil9 = lib.jawabanNo9(npm)
14 hasil10 = lib.jawabanNo10(npm)
15 print()

```

Listing 3.50 Jawaban soal No. 11 Ketrampilan Pemrograman.

12. Jawaban soal No. 12

```

1 from kelas3lib import kelas3lib
2
3 npm = "1174002"
4
5 kel3lib = kelas3lib(npm)
6
7 kel3lib.jawabanNo1()
8 kel3lib.jawabanNo2()
9 kel3lib.jawabanNo3()
10 kel3lib.jawabanNo4()
11 kel3lib.jawabanNo5()
12 kel3lib.jawabanNo6()
13 kel3lib.jawabanNo7()
14 kel3lib.jawabanNo8()
15 kel3lib.jawabanNo9()
16 kel3lib.jawabanNo10()

```

Listing 3.51 Jawaban soal No. 12 Ketrampilan Pemrograman.

3.9.3 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.

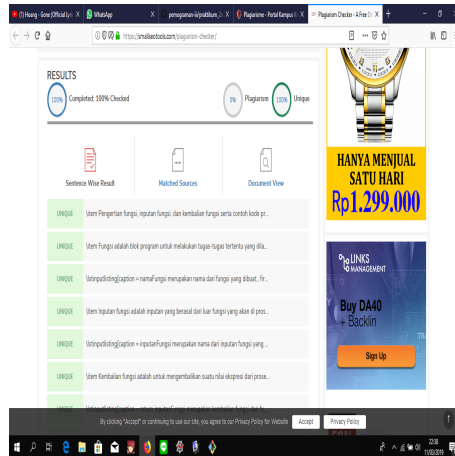
Contoh fungsi yang menggunakan try except

```

1 def halo(nama):
2     try:
3         print("Hallo, "+str(nama))
4     except:
5         print("Ada Yang Error")
6
7 halo(input("Cantumkan Nama Anda : "))

```

Listing 3.52 Contoh fungsi yang menggunakan try except .



Gambar 3.2 SS Bebas Plagiarisme

3.10 Dezha Martha

3.10.1 Pemahaman Teori

1. Apa itu fungsi, inputan fungsi dan kembalian fungsi dengan contoh kode program lainnya. Fungsi memiliki tujuan agar kita dapat memecah program besar menjadi sub-sub program yang lebih sederhana. Pada saat kita membutuhkan suatu fitur maka kita tinggal memanggil fungsi yang telah kita buat. Fungsi pada python dibuat dengan kata kunci def dan diikuti dengan nama fungsi yang kita buat seperti contoh dibawah :

```

1 def namaFnsgi(inputanFungsi):

```

Inputan fungsi merupakan masukan yang kita berikan pada program dan program akan menampilkan hasil dari inputan yang kita masukkan. contoh dari inputan fungsi sebagai berikut :

```

1 output = namaFnsgi("Kembalian Fungsi")

```

Pengembalian fungsi memiliki tujuan untuk mengembalikan nilai dari hasil yang telah di proses. Dalam hal ini menggunakan kata kunci return yang diikuti dengan nilai atau variabel yang akan dikembalikan.

```
1 return inputanFungsi
```

2. Apa itu paket dan cara pemanggilan paket atau library dengan contoh kode program lainnya. Library atau paket adalah modul-modul yang menyusun python. Modul-modul tersebut ditulis oleh berbagai orang dari seluruh dunia dan memiliki fungsi masing-masing untuk melakukan suatu hal. contoh kode programnya adalah sebagai berikut :

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

3. Jelaskan Apa itu kelas, apa itu objek, apa itu atribut, apa itu method dan contoh kode program lainnya masing-masing. kelas adalah Prototype yang ditentukan oleh pengguna untuk objek yang mendefinisikan seperangkat atribut yang menjadi ciri objek kelas apa pun. Objek ialah instansiasi atau perwujudan dari sebuah kelas. Bila kelas adalah prototipenya, dan objek adalah barang jadinya. Atribut adalah data anggota (variabel kelas dan variabel contoh) dan metode, diakses melalui notasi titik. Sedangkan method fungsi yang didefinisikan di dalam suatu kelas.

```
1
2 #No.3
3 class Mahasiswa:
4     jumlahMahasiswa = 0
5
6     def __init__(self, npm, nama):
7         self.npm = npm
8         self.nama = nama
9         Mahasiswa.jumlahMahasiswa +=1
10
11     def tampilProfil(self):
12         print("NPM :", self.npm)
13         print("Nama :", self.nama)
14         print()
15
16 mahasiswa1 = Mahasiswa("1174025", "Dezha Martha")
17 mahasiswa2 = Mahasiswa("1174026", "Jungle Supanti")
18
19 mahasiswa1.tampilProfil()
20 mahasiswa2.tampilProfil()
21
22 print("Total mahasiswa adalah ", Mahasiswa.jumlahMahasiswa)
```

4. Jelaskan cara pemanggilan library kelas dari instansiasi dan pemakaiannya dengan contoh program lainnya. cara pemanggilan library kelas dari instansiasi dan pemakaiannya adalah dengan cara meng-import library yang ada di dalam satu folder dan menggunakan kode berikut :


```
1 #No.4
2 from file.Mahasiswa import Mahasiswa
3
4 mhs1 = Mahasiswa("1174025", "Dezha Martha")
5 mhs2 = Mahasiswa("1174026", "Jungle Supanti")
6
7 mhs1.tampilProfil()
8 mhs2.tampilProfil()
9
10 print("Total mahasiswa adalah ", Mahasiswa.jumlahMahasiswa)
```

5. Jelaskan dengan contoh pemakaian paket dengan perintah from kalkulator import Penambahan disertai dengan contoh kode lainnya. contoh kodenya adalah sebagai berikut :

```
1 #No.5
2 from file.kalkulator import penambahan
3
4 hasil = penambahan(7, 6)
5 print(hasil)
```

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 :

```
1 #No.6
2 from file 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)
```

7. Jelaskan dengan contoh kodenya, pemakaian paket kelas apabila le library ada di dalam folder.

```
1 #No.7
2 from file.Mahasiswa import Mahasiswa
3
4 mhs1 = Mahasiswa("1174025", "Dezha Martha")
5 mhs2 = Mahasiswa("1174026", "Jungle Supanti")
6
7 mhs1.tampilProfil()
8 mhs2.tampilProfil()
9
10 print("Total mahasiswa adalah ", Mahasiswa.jumlahMahasiswa)
```

3.10.2 Praktek

1. jawaban no 1

```

1 #No.1
2 def jawabanNo1():
3
4     npm = input("Masukan NPM :")
5     npm = list(str(npm))
6
7     angka1 = {"0": "++++++ ", "1": " ++", "2": "++++++ ", "3": "
            ++++++ ", "4": "++++", "5": "++++++", "6": "++++++",
            "7": "++++++", "8": "++++ ", "9": "+++++ "}
8     angka2 = {"0": "+++ ++", "1": "++++", "2": "++ ++", "3": "
            ++ ++", "4": " ++ ++", "5": "++ ", "6": "+++ ",
            "7": "++++++", "8": "++ ++", "9": "++ ++"}
9     angka3 = {"0": "+++ ++", "1": "+++", "2": "++ ++", "3": "
            +++ ", "4": "++ ++", "5": "++ ", "6": "++++ ",
            "7": "++ ", "8": "++++ ", "9": "++ ++"}
10    angka4 = {"0": "+++ ++", "1": "+++", "2": "++ ++", "3": "
            +++ ", "4": "++++++", "5": "++++++ ", "6": "++++++",
            "7": "++ ", "8": "++++++", "9": "++++++"}
11    angka5 = {"0": "+++ ++", "1": "+++", "2": "++ ++", "3": "
            ++ ++", "4": "++", "5": "++", "6": "+++ ++",
            "7": "++ ", "8": "++ ++", "9": "++"}
12    angka6 = {"0": "++++++ ", "1": "+++", "2": "++++++", "3": "
            ++++++ ", "4": "++", "5": "++++++", "6": "+++++ ",
            "7": "++++ ", "8": "++++ ", "9": "+++++ "}
13
14    hasil1 = []
15    hasil2 = []
16    hasil3 = []
17    hasil4 = []
18    hasil5 = []
19    hasil6 = []
20
21    for x in npm:
22        hasil1.append(angka1[x])
23        hasil2.append(angka2[x])
24        hasil3.append(angka3[x])
25        hasil4.append(angka4[x])
26        hasil5.append(angka5[x])
27        hasil6.append(angka6[x])
28
29    print(*hasil1, sep=' ')
30    print(*hasil2, sep=' ')
31    print(*hasil3, sep=' ')
32    print(*hasil4, sep=' ')
33    print(*hasil5, sep=' ')
34    print(*hasil6, sep=' ')
35
36    jawabanNo1()

```

Listing 3.53 Jawaban soal no 1.

2. jawaban no 2

```

1 def ulang(npm):
2     hitung = 0
3     while(hitung <25):
4         print("Halo there , " + str(npm) + " wat shup?")
5         hitung = hitung +1
6     ulang(int(input("Masukkan NPM: ")))

```

Listing 3.54 Jawaban soal no 2.

3. jawaban no 3

```

1 def ulang3digitakhir(npm):
2     hitung = 0
3     npm = str(npm)
4     x = npm[4:7]
5     while(hitung < 7):
6         print("Halo , " + x + " Apa Kabar?")
7         hitung = hitung +1
8     ulang3digitakhir(int(input("Masukkan NPM: ")))

```

Listing 3.55 Jawaban soal no 3.

4. jawaban no 4

```

1 #no.4
2 def digit3daribelakang(npm):
3     npm = str(npm)
4     x = npm[-3]
5     print("Halo , " + x + " Apa Kabar anda?")
6     digit3daribelakang(int(input("Masukkan NPM: ")))

```

Listing 3.56 Jawaban soal no 4.

5. jawaban no 5

```

1 #no5
2 def kebawah(npm):
3     for i in npm:
4         print (i)
5
6     kebawah(input("Masukkan NPM: "))

```

Listing 3.57 Jawaban soal no 5.

6. jawaban no 6

```

1 #no.6
2 def tambah(npm):
3     jumlah = 0
4     for i in npm:
5         jumlah += int(i)
6     print ("Hasil Penjumlahan NPM adalah : " +str(jumlah))
7     tambah(input("Masukkan NPM: "))

```

Listing 3.58 Jawaban soal no 6.

7. jawaban no 7

```

1 #No.7
2 def kali(npm):
3     kalikan = 0
4     for i in npm:
5         kalikan *= int(i)
6     print ("Hasil Perkalian NPM adalah : " +str(kalikan))
7 kali(input("Masukkan NPM: "))

```

Listing 3.59 Jawaban soal no 7.

8. jawaban no 8

```

1 #N0.8
2 def digitgenap(npm):
3     npm = list(map(int, npm))
4     for n in npm:
5         if(n % 2 == 0):
6             if(n != 0):

```

Listing 3.60 Jawaban soal no 8.

9. jawaban no 9

```

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

```

Listing 3.61 Jawaban soal no 9.

10. jawaban no 10

```

1 #No.10
2 def digitprima(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 digitprima(input("Masukkan NPM: "))

```

Listing 3.62 Jawaban soal no 10.

11. jawaban 11

```
1 lib = __import__('3lib')
2
3 npm = "1174025"
4
5 hasil1 = lib.jwbNo1(npm)
6 hasil2 = lib.jwbNo2(npm)
7 hasil3 = lib.jwbNo3(npm)
8 hasil4 = lib.jwbNo4(npm)
9 hasil5 = lib.jwbNo5(npm)
10 hasil6 = lib.jwbNo6(npm)
11 hasil7 = lib.jwbNo7(npm)
12 hasil8 = lib.jwbNo8(npm)
13 hasil9 = lib.jwbNo9(npm)
14 hasil10 = lib.jwbNo10(npm)
```

Listing 3.63 Jawaban soal no 11.

12. jawaban 12

```
1 from kelas3lib import kelas3lib
2
3 npm = "1174025"
4
5 kel3lib = kelas3lib(npm)
6
7 kel3lib.JwbNo1()
8 kel3lib.JwbNo2()
9 kel3lib.JwbNo3()
10 kel3lib.JwbNo4()
11 kel3lib.JwbNo5()
12 kel3lib.JwbNo6()
13 kel3lib.JwbNo7()
14 kel3lib.JwbNo8()
15 kel3lib.JwbNo9()
16 kel3lib.JwbNo10()
```

Listing 3.64 Jawaban soal no 12.

3.10.3 Praktek

1. Peringatan error yang ada dan penjelasannya.

- Syntax errors syntax error adalah suatu keadaan saat kode yang dijalankan pada Python mengalami kesalahan penulisan, penempatan dll. Solusi nya adalah dengan menemukan letak sumber masalah dan memperbaikinya.
- Zero Division Error ZeroDivisionError adalah exceptions yang terjadi eksekusi program menghasilkan perhitungan matematika pembagian dengan angka nol (0). Solusiya 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. Solusi

nya ialah 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 tipe objek yang tidak sesuai. Solusinya adalah mengkonversi variabelnya sesuai dengan tipe data yang akan digunakan.

Contoh fungsi yang menggunakan try except

```

1 #Keterampilan Penanganan error
2 def halo(nama):
3     try:
4         print("Hallo , "+str(nama))
5     except:
6         print("Ada yang error nih")
7
8 halo(input("Nama ei nama : "))

```

Listing 3.65 Penggunaan Try Except .

3.11 Damara Benedikta

3.11.1 Pemahaman Teori

1. Apa itu fungsi, inputan fungsi dan kembalian fungsi dengan contoh kode program lainnya. Fungsi memiliki tujuan agar kita dapat memecah program besar menjadi sub-sub program yang lebih sederhana. Pada masing-masing fitur pada program dapat dibuat dalam satu fungsi. Pada saat kita membutuhkan suatu fitur maka kita tinggal memanggil fungsi yang telah kita buat. Fungsi pada python dibuat dengan menggunakan kata kunci `def` dan diikuti dengan nama fungsi yang telah kita buat seperti contoh dibawah ini :

```

1 def namaFungsi(inputanFungsi):

```

Inputan fungsi merupakan masukan yang kita berikan pada program dan program akan menampilkan hasil dari inputan yang telah kita masukkan atau akan menampilkan hasil pada proses selanjutnya. contoh dari inputan fungsi sebagai berikut :

```

1     return inputanFungsi

```

Pengembalian fungsi memiliki tujuan untuk mengembalikan nilai dari hasil yang telah di proses. Dalam hal ini menggunakan kata kunci `return` yang diikuti dengan nilai atau variabel yang akan dikembalikan.

```

1 def namaFungsi(inputanFungsi):
2     return inputanFungsi
3
4 output = namaFungsi("Kembalian Fungsi")
5 print(output)

```

2. Apa itu paket dan cara pemanggilan paket atau library dengan contoh kode program lainnya. Library atau paket adalah modul-modul yang menyusun python. Modul-modul tersebut ditulis oleh berbagai orang dari seluruh dunia dan memiliki fungsi masing-masing untuk melakukan suatu hal. contoh kode programnya adalah sebagai berikut :

```
1 import math
2 print("Nilai pi adalah: ", math.pi)
```

3. Jelaskan Apa itu kelas, apa itu objek, apa itu atribut, apa itu method dan contoh kode program lainnya masing-masing. kelas adalah Prototype atau blueprint untuk menciptakan suatu object yang mendefinisikan seperangkat atribut yang menjadi ciri objek kelas apa pun. Objek ialah instansiasi atau perwujudan dari sebuah kelas. Bila kelas adalah prototipenya, dan objek adalah hasil dari class jadinya. Atribut merupakan data dari anggota (variabel kelas, variabel contoh) dan metode, yang diakses dengan notasi titik. Sedangkan method fungsi yang didefinisikan di dalam suatu kelas.

```
1 class Mahasiswa:
2     jumlahMahasiswa = 0
3
4     def _init_(self, npm, nama):
5         self.npm = npm
6         self.nama = nama
7         Mahasiswa.jumlahMahasiswa +=1
8
9     def tampilkanProfil(self):
10        print("NPM :", self.npm)
11        print("Nama :", self.nama)
12        print()
13
14 mahasiswa1 = Mahasiswa("1174012", "Damara Benedikta")
15 mahasiswa2 = Mahasiswa("1174077", "Abdul")
16
17 mahasiswa1.tampilkanProfil()
18 mahasiswa2.tampilkanProfil()
19
20 print("Total mahasiswa adalah ", Mahasiswa.jumlahMahasiswa)
```

4. Jelaskan cara pemanggilan library kelas dari instansiasi dan pemakaiannya dengan contoh program lainnya. cara pemanggilan library kelas dari instansiasi dan pemakaiannya adalah dengan cara meng-import library yang ada di dalam satu folder dengan menggunakan kode berikut :

```
1 from Mahasiswa import Mahasiswa
2
3 mhs1 = Mahasiswa("1174012", "Damara Benedikta")
4 mhs2 = Mahasiswa("1174077", "Abdul")
5
6 mhs1.tampilkanProfil()
7 mhs2.tampilkanProfil()
8
9 print("Total mahasiswa adalah ", Mahasiswa.jumlahMahasiswa)
```

5. Jelaskan dengan contoh pemakaian paket dengan perintah from kalkulator import Penambahan disertai dengan contoh kode lainnya. contoh kodenya adalah sebagai berikut :

```
1 from kalkulator import Penambahan
2
3 hasil = Penambahan(10, 5)
4 print(hasil)
```

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 :

```
1 from folder import kalkulator
2
3 a=50
4 b=25
5
6 hasil1=kalkulator.Penambahan(a,b)
7 hasil2=kalkulator.Pengurangan(a,b)
8 hasil3=kalkulator.Perkalian(a,b)
9 hasil4=kalkulator.Pembagian(a,b)
10
11 print(hasil1)
12 print(hasil2)
13 print(hasil3)
14 print(hasil4)
```

7. Jelaskan dengan contoh kodenya, pemakaian paket kelas apabila le library ada di dalam folder.

```
1 from folder.Mahasiswa import Mahasiswa
2
3 mhs1 = Mahasiswa("1174012", "Damara Benedikta")
4 mhs2 = Mahasiswa("1174077", "Abdul")
5
6 mhs1.tampilkanProfil()
7 mhs2.tampilkanProfil()
8
9 print("Total mahasiswa adalah ", Mahasiswa.jumlahMahasiswa)
```

3.11.2 Praktek

1. jawaban no 1

```
1 def jwbNo1():
2
3     npm = input("Masukan NPM :")
4     npm = list(str(npm))
5
6     angka1 = {"0": "##### ", "1": " ##", "2": "##### ", "3": "##### ",
7               "4": " #####", "5": "#####", "6": "#####",
8               "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
13    hasil1 = []
14    hasil2 = []
15    hasil3 = []
16    hasil4 = []
17    hasil5 = []
18    hasil6 = []
19
20    for x in npm:
21        hasil1.append(angka1[x])
22        hasil2.append(angka2[x])
23        hasil3.append(angka3[x])
24        hasil4.append(angka4[x])
25        hasil5.append(angka5[x])
26        hasil6.append(angka6[x])
27
28    print(*hasil1, sep=' ')
29    print(*hasil2, sep=' ')
30    print(*hasil3, sep=' ')
31    print(*hasil4, sep=' ')
32    print(*hasil5, sep=' ')
33    print(*hasil6, sep=' ')
34
35    jwbNo1()

```

2. jawaban no 2

```

1  def jwbNo2(npm):
2      hitung = 0
3      while(hitung <12):
4          print("Halo, " + str(npm) + " Apa Kabar?")
5          hitung = hitung +1
6  jwbNo2(int(input("Masukkan NPM: ")))

```

3. jawaban no 3

```

1  def jwbNo3(npm):
2      hitung = 0
3      npm = str(npm)
4      x = npm[4:7]
5      while(hitung < 3):

```

```

6         print("Halo , " + x + " Apa Kabar?")
7         hitung = hitung +1
8     jwbNo3(int(input("Masukkan NPM: ")))

```

4. jawaban no 4

```

1     def jwbNo4(npm):
2         npm = str(npm)
3         x = npm[-3]
4         print("Halo , " + x + " Apa Kabar?")
5     jwbNo4(int(input("Masukkan NPM: ")))

```

5. jawaban no 5

```

1     def jwbNo5(npm):
2         for i in npm:
3             print (i)
4
5     jwbNo5(input("Masukkan NPM: "))

```

6. jawaban no 6

```

1     def jwbNo6(npm):
2         jumlah = 0
3         for i in npm:
4             jumlah += int(i)
5         print ("Hasil Penjumlahan NPM adalah : " +str(jumlah))
6     jwbNo6(input("Masukkan NPM: "))

```

7. jawaban no 7

```

1     def jwbNo7(npm):
2         kalikan = 0
3         for i in npm:
4             kalikan *= int(i)
5         print ("Hasil Perkalian NPM adalah : " +str(kalikan))
6     jwbNo7(input("Masukkan NPM: "))

```

8. jawaban no 8

```

1     def jwbNo8(npm):
2         npm = list(map(int , npm))
3         for n in npm:
4             if(n % 2 == 0):
5                 if(n != 0):
6                     print(n, end = "")
7     jwbNo8(input("Masukkan NPM: "))

```

9. jawaban no 9

```

1     def jwbNo9(npm):
2         npm = list(map(int , npm))
3         for n in npm:
4             if(n % 2 !=0):
5                 print(n, end = "")
6     jwbNo9(input("Masukkan NPM: "))

```

10. jawaban no 10

```

1 def jwbNo10(npm):
2     npm = list(map(int, npm))
3     prima = []
4     for n in npm:
5         bilPrima = True
6         if n == 0 or n == 1:
7             bilPrima = False
8         for x in range(2, n):
9             if n % x == 0:
10                bilPrima = False
11        if bilPrima:
12            prima.append(n)
13
14        for p in prima:
15            print(p, end = " ")
16 jwbNo10(input("Masukkan NPM: "))

```

11. jawaban 11

```

1 lib = __import__('3lib')
2
3 npm = "1174012"
4
5 hs11 = lib.jwbNo1(npm)
6 hs12 = lib.jwbNo2(npm)
7 hs13 = lib.jwbNo3(npm)
8 hs14 = lib.jwbNo4(npm)
9 hs15 = lib.jwbNo5(npm)
10 hs16 = lib.jwbNo6(npm)
11 hs17 = lib.jwbNo7(npm)
12 hs18 = lib.jwbNo8(npm)
13 hs19 = lib.jwbNo9(npm)
14 hs110 = lib.jwbNo10(npm)
15 print()

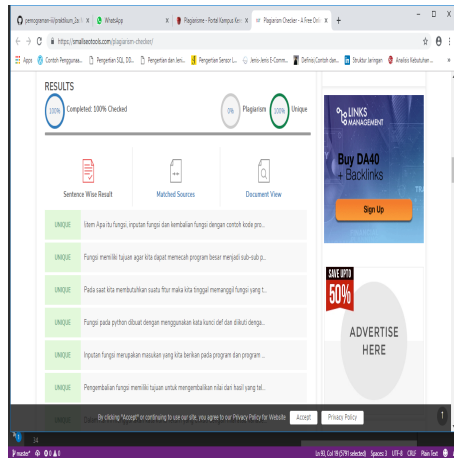
```

12. jawaban 12

```

1 from kelas3lib import kelas3lib
2
3 npm = "1174012"
4
5 kel3lib = kelas3lib(npm)
6
7 kel3lib.jwbNo1()
8 kel3lib.jwbNo2()
9 kel3lib.jwbNo3()
10 kel3lib.jwbNo4()
11 kel3lib.jwbNo5()
12 kel3lib.jwbNo6()
13 kel3lib.jwbNo7()
14 kel3lib.jwbNo8()
15 kel3lib.jwbNo9()
16 kel3lib.jwbNo10()

```



Gambar 3.3 SS plagiarisme

3.11.3 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 merupakan suatu exception yang terjadi saat akan mengeksekusi suatu operasi atau fungsi dengan type object yang tidak sesuai. Solusinya adalah mengkonversi variabelnya sesuai dengan tipe data yang akan digunakan.

Index

disruptif, xxxv
modern, xxxv