

Tugas Chapter 3

Pemrograman II



Alifia Zahra

1184051

D4 TI 2B

Program Studi D4 Teknik Informatika

Applied Bachelor Program of Informatics Engineering

Politeknik Pos Indonesia

Bandung 2019

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

Acknowledgements

Pertama-tama kami panjatkan puji dan syukur kepada Allah SWT yang telah memberikan rahmat dan hidayah-Nya sehingga Modul Praktikum ini dapat diselesaikan.

Abstract

Modul Praktikum ini dibuat dengan tujuan memberikan acuan, bagi mahasiswa dan dosen Pengajar Mata Kuliah. Pada intinya buku ini menjelaskan secara lengkap tentang Standar penilaian mata kuliah pemrograman II di Program Studi D4 Teknik Informatika, dan juga mengatur mekanisme, teknik penulisan, serta penilaiannya. Dengan demikian diharapkan semua pihak yang terlibat dalam aktivitas belajar dan mengajar berjalan lancar dan sesuai dengan standar.

Contents

1	Fungsi dan Kelas	1
1.1	Pemahaman Teori	1
1.1.1	Fungsi, Inputan Fungsi, dan Kembalian Fungsi	1
1.1.2	Library dan Cara Pemanggilannya	3
1.1.3	Kelas, Objek, Atribut, Method, dan Contoh Kodanya	3
1.1.4	Cara Memanggil Library Kelas dari Instansi dan Pemakaiannya dengan Contoh Program Lainnya	4
1.1.5	Contoh Pemakaian Library dengan perintah from kalkulator import penambahan disertai dengan contoh kode lainnya . . .	5
1.1.6	Pemanggilan paket fungsi apabila file library berada di dalam folder	5
1.1.7	Pemakaian paket kelas apabila file library berada di dalam folder	5
1.2	Ketrampilan Pemrograman	6
1.3	Ketrampilan Penanganan Error	15
A	Form Penilaian Jurnal	16
B	FAQ	19
	Bibliography	21

List of Figures

A.1	Form nilai bagian 1.	17
A.2	form nilai bagian 2.	18

Chapter 1

Fungsi dan Kelas

Tujuan pembelajaran pada pertemuan ketiga antara lain:

1. Menenal struktur fungsi di python dalam satu file dan cara pemanggilannya
2. Mengerti cara membuat library fungsi dan melakukan import dan berbagai jenis import
3. Mengerti struktur library kelas python dan cara pemakaiannya
4. Mengatasi Error yang terjadi akibat pemakaian fungsi dan kelas
5. Try Except

Tugas dengan cara dikumpulkan dengan pull request ke github dengan menggunakan latex pada repo yang dibuat oleh asisten IRC. Kode program dipisah dalam folder src NPM.py yang berisi praktek dari masing-masing tugas file terpisah sesuai nomor yang kemudian dipanggil menggunakan input listing ke dalam file latex penjelasan atau nomor pengerjaan. Masing masing soal bernilai 5 dengan total nilai 100. Gunakan bahasa yang baku dan bebas plagiat dengan dibuktikan hasil scan plagiarisme. Serta hasil scrinsut dari komputer sendiri, dan kode hasil sendiri.

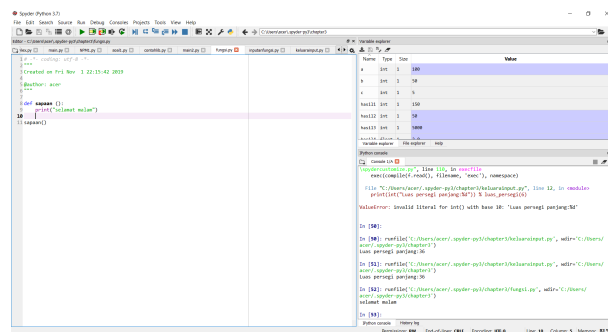
1.1 Pemahaman Teori

1.1.1 Fungsi, Inputan Fungsi, dan Kembalian Fungsi

Fungsi digunakan untuk memecah kode program besar menjadi sub program yang sederhana, disaat membutuhkan kode program besar tersebut kita tinggal memanggil fungsinya. Fitur-fitur dalam program dapat menjadi satu fungsi.

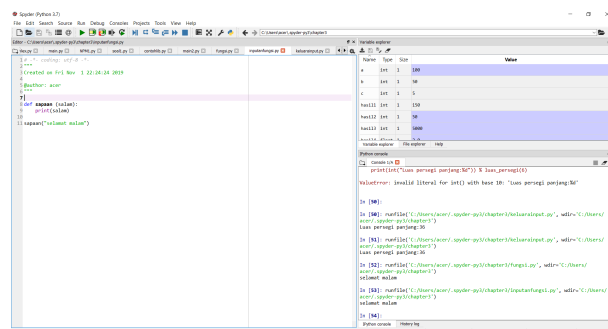
Pada python, fungsi diikuti dengan kata kunci *def* lalu diikuti oleh nama fungsinya.

```
1 # -*- coding: utf-8 -*-
2 """
3 Created on Fri Nov 1 22:15:42 2019
4
5 @author: acer
6 """
7
8 def sapaan ():
9     print("selamat malam")
10
11 sapaan()
```



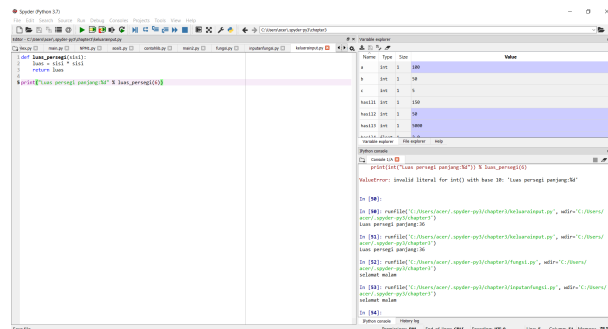
Untuk memberikan inputan pada fungsi, kita dapat menggunakan dan memanfaatkan parameter. Lalu apa itu parameter? Parameter merupakan suatu variabel untuk menampung nilai yang akan diproses dalam fungsi.

```
1 # -*- coding: utf-8 -*-
2 """
3 Created on Fri Nov 1 22:24:24 2019
4
5 @author: acer
6 """
7
8 def sapaan (salam):
9     print(salam)
10
11 sapaan("selamat malam")
```



Sedangkan fungsi yang harus mengembalikan nilai bisa disebut juga keluaran fungsi, kita dapat menggunakan kata kunci *return*.

```
1 def luas_persegi(sisi):
2     luas = sisi * sisi
3     return luas
4
5 print("Luas persegi panjang:%d" % luas_persegi(6))
```



1.1.2 Library dan Cara Pemanggilannya

Library atau paket merupakan sekumpulan file-file module, sedangkan module berisikan kumpulan file dari fungsi, class, dsb. Agar suatu file dianggap paket atau package oleh python maka buatlah file *init.py* terlebih dahulu.

```
1 #math adalah modul bawaan
2
3 import math
4 print("Nilai pi adalah: ",math.pi)
```

1.1.3 Kelas, Objek, Atribut, Method, dan Contoh Kodenya

Suatu prototipe yang dibuat oleh user untuk objek yang mendefinisikan atribut yang menjadi ciri objek suatu kelas. Untuk mendefinisikan suatu class, tulislah *class* sebelum nama class.

Atribut merupakan suatu data anggota sebuah variabel dan dapat diakses melalui notasi titik.

Object Oriented Programming (OOP) adalah paradigma atau teknik pemrograman di mana semua hal dalam program dimodelkan seperti objek dalam dunia nyata. Objek merupakan suatu blueprint, di dunia nyata objek memiliki ciri atau atribut dan juga aksi atau kelakuan.

Method merupakan suatu fungsi atau atribut yang dimiliki suatu class. Method terbagi menjadi dua jenis yaitu function dan procedure.

Function merupakan bagian dari program yang memiliki algoritma tertentu dan mengembalikan suatu nilai.

Sedangkan Procedure mempunyai algoritma yang tidak mengembalikan nilai.

```
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)
```

1.1.4 Cara Memanggil Library Kelas dari Instansi dan Pemakaiannya dengan Contoh Program Lainnya

Berikut adalah contoh suatu file fungsi yang akan dipanggil dengan perintah *form*.

```
1 def salam():
2     nama = input()
3     print ("Hello , Selamat Datang "+nama)
4
5 salam()
6
7 def full_name():
8     first=input()
9     last=input()
10    print ("Hello "+first+last)
11
12 full_name()
```

Berikut adalah contoh pemanggilan library.

```

1 from fungsi1 import salam
2
3 c=salam()
4 print (c)

```

1.1.5 Contoh Pemakaian Library dengan perintah from kalkulator import penambahan disertai dengan contoh kode lainnya

```

1 def Penjumlahan(a,b):
2     r = a + b
3     return r
4 def Pengurangan(a,b):
5     r = a - b
6     return r
7 def Perkalian(a,b):
8     r = a * b
9     return r
10 def Pembagian(a,b):
11     r = a / b
12     return r

```

```

1 from kalkulator import Penjumlahan #memanggil file kalkulator dari
   library kemudian memanggil fungsi penambahan
2
3 c=Penjumlahan(3,2)
4 print (c)

```

1.1.6 Pemanggilan paket fungsi apabila file library berada di dalam folder

Untuk mengakses suatu library yang berada di dalam folder, terlebih dahulu foldernya kita tulis(src) kemudian import nama librarynya (library1).

1.1.7 Pemakaian paket kelas apabila file library berada di dalam folder

Untuk mengakses suatu class dalam sebuah folder, kita perlu menuliskan nama foldernya terlebih dahulu kemudian mengimport nama kelasnya.

1.2 Ketrampilan Pemrograman

```
1 def printNPM(npm):
2
3     npm = list(str(npm))
4
5     angka1 = {"0": "+++++", "1": "++", "2": "+++++", "3": "+++++",
6               "4": "+++", "5": "+++++", "6": "+++++", "7": "+++++",
7               "8": "+++++"}
8     angka2 = {"0": "+++ ++", "1": "++++", "2": "++ ++", "3": "++
9               ++", "4": "++++", "5": "++", "6": "++++", "7": "++
10              ++", "8": "+++ ++"}
11     angka3 = {"0": "+++ ++", "1": "+++ ", "2": "+++ ", "3": "+++
12              ", "4": "+++ ++", "5": "+++++", "6": "+++++", "7": "+++
13              ", "8": "+++++"}
14     angka4 = {"0": "+++ ++", "1": "+++ ", "2": "+++ ", "3": "+++
15              ", "4": "+++++", "5": "+++", "6": "+++ ++", "7": "+++
16              ", "8": "+++++"}
17     angka5 = {"0": "+++ ++", "1": "+++ ", "2": "+++ ", "3": "+++
18              ", "4": "+++", "5": "++ ++", "6": "+++ ++", "7": "+++
19              ", "8": "+++ ++"}
20     angka6 = {"0": "+++++", "1": "+++ ", "2": "+++++", "3": "+++++",
21               "4": "+++", "5": "+++++", "6": "+++++", "7": "+++++",
22               "8": "+++++"}
23
24     hasil1 = []
25     hasil2 = []
26     hasil3 = []
27     hasil4 = []
28     hasil5 = []
29     hasil6 = []
30
31     for x in npm:
32
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=' ')
46
47 printNPM(input("Masukan NPM anda: "))
```

```
2 def perulangan(npm):
```

```

2     hitung = 0
3     while(hitung < 51):
4         print("Halo, 1184051 apa kabar")
5         hitung = hitung + 1
6
7     perulangan(int(input("Masukan NPM: ")))

```

```

31 # -*- coding: utf-8 -*-
2 """
3 Created on Thu Nov  7 03:45:12 2019
4
5 @author: acer
6 """
7
8 def x(npm):
9
10     a = 1
11     b = list(map(int, npm[4:7]))
12     b = sum(b)
13     while(a <= b):
14         print("Halo, "+str(npm[-3:])+ " apa kabar?")
15         a += 1
16
17 x(input("Masukkan NPM Anda: "))

```

```

41 # -*- coding: utf-8 -*-
2 """
3 Created on Thu Nov  7 03:51:50 2019
4
5 @author: acer
6 """
7
8 def x(npm):
9
10     print("Output:")
11     print("Halo, "+str(npm[-3])+ " apa kabar?")
12
13 x(input("Masukkan NPM Anda: "))

```

```

51 # -*- coding: utf-8 -*-
2 """
3 Created on Thu Nov  7 03:55:27 2019
4
5 @author: acer
6 """
7
8 def x(npm):
9
10     npm = list(map(int, npm))
11     for n in npm:
12         print(n)
13
14 x(input("Masukkan NPM Anda: "))

```

```

6: # -*- coding: utf-8 -*-
  """
  Created on Thu Nov  7 03:57:43 2019

  @author: acer
  """
  7
  8
  9 def X(npm):
10
11     npm = list(map(int, npm))
12     hasil = 0
13     for n in npm:
14         hasil += n
15     print(hasil)
16
17 X(input("Masukkan NPM Anda: "))

```

```

7: # -*- coding: utf-8 -*-
  """
  Created on Thu Nov  7 03:59:06 2019

  @author: acer
  """
  7
  8
  9 def X(npm):
10
11     npm = list(map(int, npm))
12     hasil = 0
13     for n in npm:
14         hasil *= n
15     print(hasil)
16
17 X(input("Masukkan NPM Anda: "))

```

```

8: def X(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 X(input("Masukan NPM anda :"))

```

```

9: def X(npm):
  2     npm = list(map(int, npm))
  3     for n in npm:
  4         if (n % 2 != 0):
  5             print(n, end = "")
  6 X(input("Masukan NPM anda :"))

```

```

10: def X(npm):
  2     npm = list(map(int, npm))

```

```

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

```

11. #SOAL1

```

2     def X(npm):
3
4         npm = list(str(npm))
5
6         angka1 = {"0": "+++++", "1": "++", "2": "+++++", "3": "+++++", "4": "+++", "5": "+++++", "6": "+++++", "7": "+++++", "8": "+++++"}
7         angka2 = {"0": "+++ ++", "1": "++++", "2": "++ ++", "3": "++ ++", "4": "++++", "5": "++", "6": "++++", "7": "+++ ++", "8": "+++ ++"}
8         angka3 = {"0": "+++ ++", "1": "+++ ++", "2": "+++ ++", "3": "+++ ++", "4": "+++ ++", "5": "+++++", "6": "+++++", "7": "+++ ++", "8": "+++++"}
9         angka4 = {"0": "+++ ++", "1": "+++ ++", "2": "+++ ++", "3": "+++ ++", "4": "+++++", "5": "+++ ++", "6": "+++ ++", "7": "+++ ++", "8": "+++++"}
10        angka5 = {"0": "+++ ++", "1": "+++ ++", "2": "+++ ++", "3": "+++ ++", "4": "+++ ++", "5": "+++ ++", "6": "+++ ++", "7": "+++ ++", "8": "+++ ++"}
11        angka6 = {"0": "+++++", "1": "+++ ++", "2": "+++++", "3": "+++++", "4": "+++ ++", "5": "+++++", "6": "+++++", "7": "+++ ++", "8": "+++++"}
12
13        hasil1 = []
14        hasil2 = []
15        hasil3 = []
16        hasil4 = []
17        hasil5 = []
18        hasil6 = []
19
20
21        for x in npm:
22
23            hasil1.append(angka1[x])
24            hasil2.append(angka2[x])
25            hasil3.append(angka3[x])
26            hasil4.append(angka4[x])
27            hasil5.append(angka5[x])

```

```

28     hasil6.append(angka6[x])
29
30
31     print(*hasil1, sep=' ')
32     print(*hasil2, sep=' ')
33     print(*hasil3, sep=' ')
34     print(*hasil4, sep=' ')
35     print(*hasil5, sep=' ')
36     print(*hasil6, sep=' ')
37
38
39 X(input("Masukan NPM anda: "))
40
41 #SOAL 2
42 def Y(npm):
43     hitung = 0
44     while(hitung < 65):
45         print("Hallo, 1184065 apa kabar")
46         hitung = hitung + 1
47
48 Y(int(input("Masukan NPM: ")))
49
50 #SOAL3
51 def printNPMTigaDigit(npm):
52
53     ulang = 1
54     sampai = list(map(int, npm[3:7]))
55     sampai = sum(sampai)
56     while(ulang <= sampai):
57         print("Halo, "+str(npm[-3:])+ " apa kabar?")
58         ulang += 1
59
60 printNPMTigaDigit(input("Masukkan NPM Anda: "))
61
62 #SOAL4
63 def printdigit_ketiga(npm):
64
65     print("Output:")
66     print("Halo, "+str(npm[-3])+ " apa kabar?")
67
68 printdigit_ketiga(input("Masukkan NPM Anda:"))
69
70 #SOAL5
71 def satupersatu(npm):
72
73     npm = list(map(int, npm))
74     for n in npm:
75         print(n)
76
77 satupersatu(input("Masukkan NPM Anda: "))
78
79 #SOAL6
80 def printpenjumlahan(npm):
81

```



```

82     npm = list(map(int, npm))
83     hasil = 0
84     for n in npm:
85         hasil += n
86     print(hasil)
87
88 printpenjumlahan(input("Masukkan NPM Anda: "))
89
90 #SOAL7
91 def printperkalian(npm):
92
93     npm = list(map(int, npm))
94     hasil = 0
95     for n in npm:
96         hasil *= n
97     print(hasil)
98
99 printperkalian(input("Masukkan NPM Anda: "))
100
101 #SOAL8
102 def printNPMDigitGenap(npm):
103     npm = list(map(int, npm))
104     for n in npm:
105         if (n % 2 == 0):
106             if (n != 0):
107                 print(n, end = " ")
108 printNPMDigitGenap(input("Masukan NPM anda: "))
109
110 #SOAL9
111 def printNPMDigitGanjil(npm):
112     npm = list(map(int, npm))
113     for n in npm:
114         if (n % 2 != 0):
115             print(n, end = " ")
116 printNPMDigitGanjil(input("Masukan NPM anda: "))
117
118 #SOAL10
119 def printNPMDigitPrima(npm):
120     npm = list(map(int, npm))
121     prima = []
122     for n in npm:
123         isPrime = True
124         if n == 0 or n == 1:
125             isPrime = False
126         for x in range(2, n):
127             if n % x == 0:
128                 isPrime = False
129         if isPrime:
130             prima.append(n)
131
132     for p in prima:
133         print(p, end = " ")
134 printNPMDigitPrima(input("Masukan NPM anda: "))

```

```

12: class kelas3lib:
13:     def __init__(self, npm):
14:         self.npm = npm
15:
16: #SOAL1
17:
18: def printNPM(npm):
19:
20:     npm = list(str(npm))
21:
22:     angka1 = {"0": "##### ", "1": " ##", "2": "##### ", "3": "##### ",
23:               "4": "   ##", "5": "##### ", "6": "##### ", "7": "##### ",
24:               "8": "##### " }
25:     angka2 = {"0": "### ##", "1": "####", "2": "##  ##", "3": "##  ##",
26:               "4": "   #####", "5": "##      ", "6": "###      ", "7": "###      ",
27:               "8": "###  ##" }
28:     angka3 = {"0": "### ##", "1": "   ##", "2": "   ## ", "3": "   ## ",
29:               "4": "   ## ##", "5": "##### ", "6": "##### ", "7": "   ## ",
30:               "8": "   ##### " }
31:     angka4 = {"0": "### ##", "1": "   ##", "2": "   ## ", "3": "   ## ",
32:               "4": "##### ", "5": "   #####", "6": "###  ##", "7": "   ## ",
33:               "8": "   ##### " }
34:     angka5 = {"0": "### ##", "1": "   ##", "2": "   ## ", "3": "##  ##",
35:               "4": "   #####", "5": "##  ##", "6": "###  ##", "7": "   ## ",
36:               "8": "###  ##" }
37:     angka6 = {"0": "##### ", "1": "   ##", "2": "##### ", "3": "##### ",
38:               "4": "   #####", "5": "   ##### ", "6": "##### ", "7": "   ## ",
39:               "8": "   ##### " }
40:
41:     hasil1 = []
42:     hasil2 = []
43:     hasil3 = []
44:     hasil4 = []
45:     hasil5 = []
46:     hasil6 = []
47:
48:     for x in npm:
49:
50:         hasil1.append(angka1[x])
51:         hasil2.append(angka2[x])
52:         hasil3.append(angka3[x])
53:         hasil4.append(angka4[x])
54:         hasil5.append(angka5[x])
55:         hasil6.append(angka6[x])
56:
57:     print(*hasil1, sep=' ')
58:     print(*hasil2, sep=' ')
59:     print(*hasil3, sep=' ')
60:     print(*hasil4, sep=' ')
61:     print(*hasil5, sep=' ')
62:     print(*hasil6, sep=' ')

```

```

43
44 printNPM(input("Masukan NPM anda: "))
45
46 #SOAL2
47 def perulangan(npm):
48     hitung = 0
49     while(hitung < 65):
50         print("Halo , 1184065 apa kabar")
51         hitung = hitung + 1
52
53 perulangan(int(input("Masukan NPM: ")))
54
55 #SOAL3
56 def printNPMTigaDigit(npm):
57
58     ulang = 1
59     sampai = list(map(int , npm[3:7]))
60     sampai = sum(sampai)
61     while(ulang <= sampai):
62         print("Halo , "+str(npm[-3:])+ " apa kabar?")
63         ulang += 1
64
65 printNPMTigaDigit(input("Masukkan NPM Anda: "))
66
67 #SOAL4
68 def printdigit_ketiga(npm):
69
70     print("Output:")
71     print("Halo , "+str(npm[-3])+ " apa kabar?")
72
73 printdigit_ketiga(input("Masukkan NPM Anda:"))
74
75 #SOAL5
76 def satupersatu(npm):
77
78     npm = list(map(int , npm))
79     for n in npm:
80         print(n)
81
82 satupersatu(input("Masukkan NPM Anda: "))
83
84 #SOAL6
85 def printpenjumlahan(npm):
86
87     npm = list(map(int , npm))
88     hasil = 0
89     for n in npm:
90         hasil += n
91     print(hasil)
92
93 printpenjumlahan(input("Masukkan NPM Anda: "))
94
95 #SOAL7
96 def printperkalian(npm):

```

```

97     npm = list(map(int, npm))
98     hasil = 0
99     for n in npm:
100         hasil *= n
101     print(hasil)
102
103
104 printperkalian(input("Masukkan NPM Anda: "))
105
106 #SOAL 8
107 #DigitGenap
108 def printNPMDigitGenap(npm):
109     npm = list(map(int, npm))
110     for n in npm:
111         if (n % 2 == 0):
112             if (n != 0):
113                 print(n, end = " ")
114 printNPMDigitGenap(input("Masukan NPM anda: "))
115
116 #SOAL9
117 #DigitGanjil
118 def printNPMDigitGanjil(npm):
119     npm = list(map(int, npm))
120     for n in npm:
121         if (n % 2 != 0):
122             print(n, end = " ")
123 printNPMDigitGanjil(input("Masukan NPM anda: "))
124
125 #SOAL10
126 def printNPMDigitPrima(npm):
127     npm = list(map(int, npm))
128     prima = []
129     for n in npm:
130         isPrime = True
131         if n == 0 or n == 1:
132             isPrime = False
133         for x in range(2, n):
134             if n % x == 0:
135                 isPrime = False
136         if isPrime:
137             prima.append(n)
138
139     for p in prima:
140         print(p, end = " ")
141 printNPMDigitPrima(input("Masukan NPM anda: "))

```



```

1 from kelas3lib import kelas3lib
2 npm = input("Masukan NPM: ")
3 kelas = kelas3lib.NPM(npm)
4 fungsi = kelas.NPM1() #NPM1 adalah fungsi jadi dapat memanggil
   fungsi NPM yang lain
5
6 print("")

```

1.3 Ketrampilan Penanganan Error

```
1 def hi(aku):  
2     try:  
3         print("Hallo , "+str(aku))  
4     except:  
5         print("Terjadi error")  
6  
7 hi(input("nama aku: "))
```

Appendix A

Form Penilaian Jurnal

gambar A.1 dan A.2 merupakan contoh bagaimana reviewer menilai jurnal kita.

NO	UNSUR	KETERANGAN	MAKS	KETERANGAN
1	Keefektifan Judul Artikel	Maksimal 12 (dua belas) kata dalam Bahasa Indonesia atau 10 (sepuluh) kata dalam Bahasa Inggris	2	a. Tidak lugas dan tidak ringkas (0) b. Kurang lugas dan kurang ringkas (1) c. Ringkas dan lugas (2)
2	Pencantuman Nama Penulis dan Lembaga Penulis		1	a. Tidak lengkap dan tidak konsisten (0) b. Lengkap tetapi tidak konsisten (0,5) c. Lengkap dan konsisten (1)
3	Abstrak	Dalam Bahasa Indonesia dan Bahasa Inggris yang baik, jumlah 150-200 kata. Isi terdiri dari latar belakang, metode, hasil, dan kesimpulan. Isi tertuang dengan kalimat yang jelas.	2	a. Tidak dalam Bahasa Indonesia dan Bahasa Inggris (0) b. Abstrak kurang jelas dan ringkas, atau hanya dalam Bahasa Inggris, atau dalam Bahasa Indonesia saja (1) c. Abstrak yang jelas dan ringkas dalam Bahasa Indonesia dan Bahasa Inggris (2)
4	Kata Kunci	Maksimal 5 kata kunci terpenting dalam paper	1	a. Tidak ada (0) b. Ada tetapi kurang mencerminkan konsep penting dalam artikel (0,5) c. Ada dan mencerminkan konsep penting dalam artikel (1)
5	Sistematika Pembahasan	Terdiri dari pendahuluan, tinjauan pustaka, metode penelitian, hasil dan pembahasan, kesimpulan dan saran, daftar pustaka	1	a. Tidak lengkap (0) b. Lengkap tetapi tidak sesuai sistematika (0,5) c. Lengkap dan bersistem (1)
6	Pemanfaatan Instrumen Pendukung	Pemanfaatan Instrumen Pendukung seperti gambar dan tabel	1	a. Tidak dimanfaatkan (0) b. Kurang informatif atau komplementer (0,5) c. Informatif dan komplementer (1)
7	Cara Pengacuan dan Pengutipan		1	a. Tidak baku (0) b. Kurang baku (0,5) c. Baku (1)
8	Penyusunan Daftar Pustaka	Penyusunan Daftar Pustaka	1	a. Tidak baku (0) b. Kurang baku (0,5) c. Baku (1)
9	Peristilahan dan Kebahasaan		2	a. Buruk (0) b. Baik (1) c. Cukup (2)
10	Makna Sumbangan bagi Kemajuan		4	a. Tidak ada (0) b. Kurang (1) c. Sedang (2) d. Cukup (3) e. Tinggi (4)

Figure A.1: Form nilai bagian 1.

11	Dampak Ilmiah		7	a. Tidak ada (0) b. Kurang (1) c. Sedang (3) d. Cukup (5) e. Besar (7)
12	Nisbah Sumber Acuan Primer berbanding Sumber lainnya	Sumber acuan yang langsung merujuk pada bidang ilmiah tertentu, sesuai topik penelitian dan sudah teruji.	3	a. < 40% (1) b. 40-80% (2) c. > 80% (3)
13	Derajat Kemutakhiran Pustaka Acuan	Derajat Kemutakhiran Pustaka Acuan	3	a. < 40% (1) b. 40-80% (2) c. > 80% (3)
14	Analisis dan Sintesis	Analisis dan Sintesis	4	a. Sedang (2) b. Cukup (3) c. Baik (4)
15	Penyimpulan	Sangat jelas relevasinya dengan latar belakang dan pembahasan, dirumuskan dengan singkat	3	a. Kurang (1) b. Cukup (2) c. Baik (3)
16	Unsur Plagiat		0	a. Tidak mengandung plagiat (0) b. Terdapat bagian-bagian yang merupakan plagiat (-5) c. Keseluruhannya merupakan plagiat (-20)
TOTAL			36	
Catatan : Nilai minimal untuk diterima 25				

Figure A.2: form nilai bagian 2.

Appendix B

FAQ

M : Kalo Intership II atau TA harus buat aplikasi ? D : Ga harus buat aplikasi tapi harus ngoding

M : Pa saya bingung mau ngapain, saya juga bingung mau presentasi apa? D : Makanya baca de, buka jurnal topik ‘ganteng’ nah kamu baca dulu sehari 5 kali ya, 4 hari udah 20 tuh. Bingung itu tanda kurang wawasan alias kurang baca.

M : Pa saya sudah cari jurnal terindeks scopus tapi ga nemu. D : Kamu punya mata de? coba dicolok dulu. Kamu udah lakuin apa aja? tolong di list laporkan ke grup Tingkat Akhir. Tinggal buka google scholar klik dari tahun 2014, cek nama jurnalnya di scimagojr.com beres.

M : Pa saya belum dapat tempat intership, jadi ga tau mau presentasi apa? D : kamu kok ga nyambung, yang dipresentasikan itu yang kamu baca bukan yang akan kamu lakukan.

M : Pa ini jurnal harus yang terindex scopus ga bisa yang lain ? D : Index scopus menandakan artikel tersebut dalam standar semantik yang mudah dipahami dan dibaca serta bukan artikel asal jadi. Jika diluar scopus biasanya lebih sukar untuk dibaca dan dipahami karena tidak adanya proses review yang baik dan benar terhadap artikel.

M : Pa saya tidak mengerti D : Coba lihat standar alasan

M : Pa saya bingung D : Coba lihat standar alasan

M : Pa saya sibuk D : Mbahmu....

M : Pa saya ganteng D : Ndasmu....

M : Pa saya kece D : wes karepmu lah....

Biasanya anda memiliki alasan tertentu jika menghadapi kendala saat proses bimbingan, disini saya akan melakukan standar alasan agar persepsi yang diterima sama dan tidak salah kaprah. Penggunaan kata alasan tersebut antara lain :

1. Tidak Mengerti : anda boleh menggunakan alasan ini jika anda sudah melakukan tahapan membaca dan meresumekan 15 jurnal. Sudah mencoba dan mempraktekkan teorinya dengan mencari di youtube dan google minimal 6 jam sehari selama 3 hari berturut-turut.

2. Bingung : anda boleh mengatakan alasan bingung setelah maksimal dalam berusaha menyelesaikan tugas bimbingan dari dosen(sudah dilakukan semua). Anda belum bisa mengatakan alasan bingung jika anda masih belum menyelesaikan tugas bimbingan dan poin nomor 1 diatas. Setelah anda menyelesaikan tugas bimbingan secara maksimal dan tahap 1 poin diatas, tapi anda masih tetap bingung maka anda boleh memakai alasan ini.

Bibliography