

LAPORAN PRAKTIKUM

PEMROGRAMAN KOMPUTER (PYTHON)



Disusun Oleh:
Rafli Pratama
H1101241008

PROGRAM STUDI SISTEM INFORMASI
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS TANJUNGPURA PONTIANAK
2024

Praktikum 7

FUNGSI (lanjutan)

1.1. Konsep Dasar

Salah satu faktor penentu yang dapat mempercepat proses belajar suatu bahasa pemrograman adalah dengan memahami penggunaan fungsi-fungsi (built-in) yang disediakan oleh bahasa pemrograman tersebut.

1.1.1 Fungsi String

Kali ini, kita akan membahas lebih jauh tentang penggunaan fungsi-fungsi yang telah disediakan oleh Python di dalam kelas string.

a) Fungsi `capitalize()`

Metode ini digunakan untuk merubah huruf pertama yang terdapat di dalam string menjadi huruf kapital.

Contoh:

```
#fungsi capitalize
firstName = 'rafli pratama'
print(firstName.capitalize())
```

Output :

```
PS D:\Kuliah\Tugas Kuliah\Semester
liah\Tugas Kuliah\Semester 1\Matkul
Rafli pratama
PS D:\Kuliah\Tugas Kuliah\Semester
```

b) Fungsi `center()`

Argumen pertama adalah lebar string yang diinginkan, sedangkan argumen kedua adalah karakter yang akan mengisi ruang kiri dan kanan.

Contoh:

```
1 #fungsi center
2 saya = 'Welcome'
3 print(saya.center(50, "-"))
```

Output :

```
liah\Tugas Kuliah\Semester 1\Matkul\Praktikum Pemogra
-----Welcome-----
PS D:\Kuliah\Tugas Kuliah\Semester 1\Matkul\Praktikum
```

c) Fungsi `count()`

Fungsi ini akan menghitung berapa kali substring muncul di dalam string.

Contoh:

```
#fungsi count
firstName = 'Rafli Pratama'
print(f'jumlah huruf a pada {firstName} ada {firstName.count('a')}')
```

Output :

```
PS D:\Kuliah\Tugas Kuliah\Semester 1\Matkul\Praktikum Pemograman
jumlah huruf a pada Rafli Pratama ada 4
PS D:\Kuliah\Tugas Kuliah\Semester 1\Matkul\Pr
```

d) Fungsi len()

Fungsi len() sebenarnya bukan merupakan anggota dari kelas string, tapi penggunaannya berkaitan dengan string. Ini merupakan fungsi built-in yang disimpan di dalam namespace global. Fungsi ini digunakan untuk memperoleh jumlah atau panjang karakter yang terdapat di string.

Contoh:

```
#fungsi len
firstName = 'Rafli Pratama'
print(f'Jumlah huruf pada nama {firstName} adalah {len(firstName)}')
```

Output :

```
PS D:\Kuliah\Tugas Kuliah\Semester 1\Matkul\Praktikum Pemograman
Jumlah huruf pada nama Rafli Pratama adalah 13
PS D:\Kuliah\Tugas Kuliah\Semester 1\Matkul\Praktikum Pem
```

e) Fungsi replace()

Fungsi ini digunakan untuk mengganti substring yang lama menjadi baru. Argumen pertama adalah substring lama dan argumen kedua adalah substring baru yang akan menggantikan substring lama.

Contoh:

```
str = 'Saya pergi dari yogyakarta'
print(str)
a= str.replace('pergi', 'pulang')
print(a)
```

Output :

```
PS D:\Kuliah\Tugas Kuliah\Semester 1\Matkul\Praktikum Pemograman
Saya pergi dari yogyakarta
Saya pulang dari yogyakarta
PS D:\Kuliah\Tugas Kuliah\Semester 1\Matkul\Praktikum Pem
```

f) Fungsi upper() dan lower()

Fungsi upper() digunakan untuk mengubah semua huruf yang ada di dalam string menjadi huruf kapital, sedangkan metode lower() digunakan mengubah semua huruf yang ada di dalam string menjadi huruf kecil

Contoh :

```

userName = 'rafli Pratama'
a = userName.upper()
b = userName.lower()
print(a)
print(b)

```

Output :

```

python.exe "d:/Kuliah/Tugas Kuliah/Semester 1\Matk
RAFLI PRATAMA
rafli pratama
PS D:\Kuliah\Tugas Kuliah\Semester 1\Matk

```

1.1.2 Fungsi Numerik

Selain string, tipe fundamental lain yang banyak dijumpai dalam kasus-kasus nyata di dunia pemrograman adalah numerik. Python juga menyediakan fungsi-fungsi untuk melakukan operasi-operasi terhadap suatu bilangan.

a) Fungsi abs()

Fungsi ini akan mengembalikan nilai absolut dari suatu bilangan diberikan.

Contoh:

```

a = -7
b = abs(a)
print(a,b)

```

Output :

```

python.exe "d:/Kuliah/Tugas Kuliah/Semester 1\Matk
-7 7
PS D:\Kuliah\Tugas

```

b) Fungsi min() dan max()

Fungsi max() digunakan untuk menentukan nilai maksimum dari nilai-nilai yang dilewatkan sebagai parameternya. Sedangkan, fungsi min() digunakan untuk menentukan nilai minimum dari nilai-nilai yang dilewatkan sebagai parameternya.

Contoh:

```

listAngka = [1,5,6,-6,5,10]
a = min(listAngka)
b = max(listAngka)
print(a,b)

```

Output :

```

Kuliah/Tugas Kuliah/Semester 1\Matk
-6 10
PS D:\Kuliah\Tugas Kuliah\Semester 1\Matk

```

c) Fungsi pow()

Fungsi ini digunakan untuk menghitung nilai argumen pertama pangkat argumen kedua.

Contoh:

```
a = 5
c = 2
b = pow(a, c)
print(a)
print(b)
```

d) Fungsi ceil()

Fungsi ini digunakan untuk melakukan pembulatan bilangan decimal ke bilangan bulat di atasnya. Fungsi ini tersimpan dalam modul math.

Contoh:

```
import math
a = 4.45
b = -2.76
print(math.ceil(a))
print(math.ceil(b))
```

Output :

```
PS D:\Kuliah\Tugas Kuliah\Semester 1\Matkul/Pr  
5  
-2  
PS D:\Kuliah\Tugas Kuliah\Semester 1\Matkul/Pr
```

e) Fungsi sqrt()

Fungsi ini ada di dalam modul math, digunakan untuk menghitung nilai akar kuadrat nilai yang melewati parameternya.

Contoh:

```
import math
a = 64
b = 16
print(f'Akar dari {a} adalah {math.sqrt(a)}')
print(f'Akar dari {b} adalah {math.sqrt(b)}')
```

Output :

```
PS D:\Kuliah\Tugas Kuliah\Semester 1\Matkul/Pr  
Akar dari 64 adalah 8.0  
Akar dari 16 adalah 4.0  
PS D:\Kuliah\Tugas Kuliah\Semester 1\Matkul/Pr
```

f) Fungsi choice()

Fungsi ini digunakan untuk mengambil nilai acak dari suatu objek sequence (list, tuple dan sebagainya). Setiap proses pemanggilan fungsi akan mengembalikan nilai yang berbeda, tapi bisa juga sama. Fungsi ini tersimpan dalam modul random.

Contoh:

```
import random
listAngka = [11,23,34,32,12,32,53,53,]
print(random.choice(listAngka))
```

Output :

```
PS D:\Kuliah\Tugas Kuliah\Semester 1\Mat
liah\Tugas Kuliah\Semester 1\Matkul\Pra
12
PS D:\Kuliah\Tugas Kuliah\Semester 1\Mat
liah\Tugas Kuliah\Semester 1\Matkul\Pra
53
PS D:\Kuliah\Tugas Kuliah\Semester 1\Mat
```

g) Fungsi randrange()

Fungsi ini digunakan untuk mengambil nilai acak dengan rentang diberikan pada parameternya.

Contoh:

```
import random
a = random.randrange(10,100)
print(a)
```

Output :

```
liah\Tugas Kuliah\Semester 1\Matkul/
10
PS D:\Kuliah\Tugas Kuliah\Semester 1\
liah\Tugas Kuliah\Semester 1\Matkul/
78
PS D:\Kuliah\Tugas Kuliah\Semester 1\
```

1.1.3 Membuat Modul Sendiri

Modul Python adalah file-file .py yang terdiri dari kode Python. Kode Python manapun bisa dianggap sebagai modul. Beberapa modul dapat ditemukan di Python Standard Library dan sudah terinstal bersamaan dengan instalasi Python. Modul lainnya dapat diinstal melalui Python's package manager pip. Selain itu, kamu dapat membuat modul Python kamu sendiri yang terdiri dari file Python .py. Buatlah kumpulan fungsi-fungsi yang disimpan dalam sebuah file python (.py) luastigonometri seperti kode program di bawah. Nama file akan menjadi nama modul. Dengan demikian, saat ini anda telah memiliki modul dengan nama luasgeometri yang sudah siap digunakan di dalam kode program manapun.

Contoh:

```

def pertambahan ():
    print('Anda memilih operasi pertambahan')
    a = int(input('Masukkan angka pertama : '))
    b = int(input('Masukkan angka kedua : '))
    print(f'Hasil pertambahan dari {a} + {b} adalah {a+b}')
    return a + b

def pengurangan():
    print('Anda memilih operasi pengurangan')
    a = int(input('Masukkan angka pertama : '))
    b = int(input('Masukkan angka kedua : '))
    print(f'Hasil pengurangan dari {a} - {b} adalah {a - b}')
    return a - b

def perkalian():
    print('Anda memilih operasi perkalian')
    a = int(input('Masukkan angka pertama : '))
    b = int(input('Masukkan angka kedua : '))
    print(f'Hasil perkalian dari {a} x {b} adalah {a*b}')
    return a * b

import operasiAritmatika
print('-----Selamat Datang-----')
while True:
    operasi = int(input('Silahkan pilih operasi yang anda inginkan :\n'))
    if operasi == 1:
        operasiAritmatika.pertambahan()
    elif operasi == 2:
        operasiAritmatika.pengurangan()
    elif operasi == 3:
        operasiAritmatika.perkalian()
    elif operasi == 4:
        operasiAritmatika.pembagian()
    elif operasi == 5:
        operasiAritmatika.modulo()
    elif operasi == 6:
        operasiAritmatika.pangkat()
    elif operasi == 7:
        print('Anda memilih keluar!!! Terima kasih')
        break
    else:
        print('Silahkan pilih operasi yang benar')

```

Untuk dapat menggunakan fungsi-fungsi yang ada di dalam modul luasgeometri yang telah dibuat. Kita perlu mengimpor atau menyertakan nama modul tersebut dengan statemen import seperti kode program di bawah:

Ketika menggunakan statemen import, semua kode yang ada di dalam modul akan diimpor. Pada saat tertentu, kita hanya ingin mengimpor bagian-bagian kode tertentu saja. Untuk melakukan hal ini, kita perlu menggunakan statemen from ... import seperti yang terlihat pada kode berikut:

1.2. Percobaan Praktikum

1.2.1 Soal 1

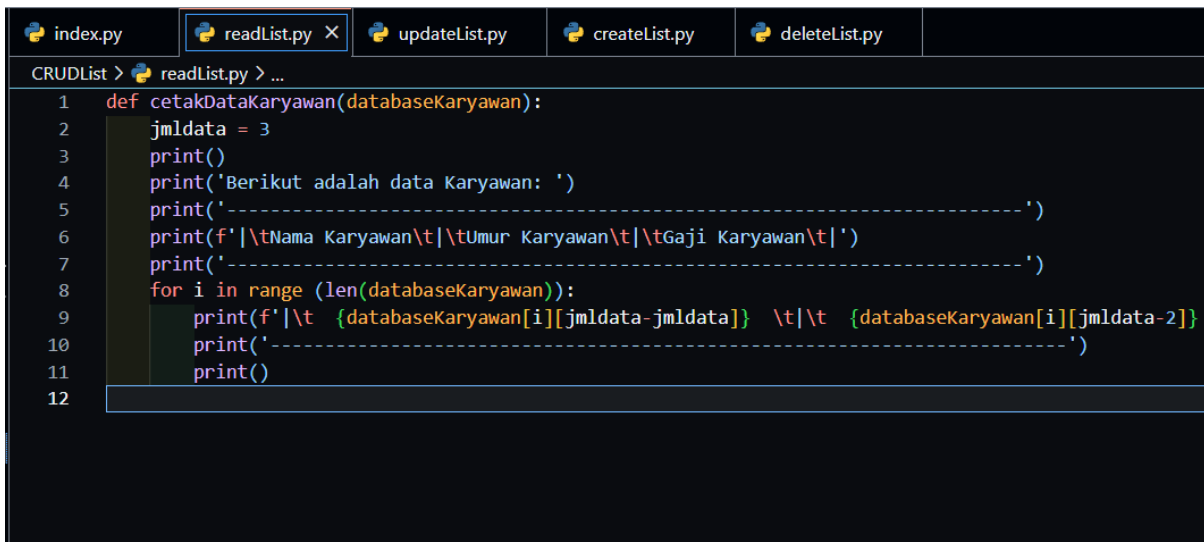
Buatlah modul untuk memanipulasi data pada sebuah list dengan fungsi dasar CRUD (create, read, update, delete). Buat program utamanya pada file kode yang berbeda.



```
index.py readList.py updateList.py createList.py X deleteList.py
CRUDList > createList.py > dataKaryawan > daftarKaryawan
1 def dataKaryawan():
2     listKaryawan = []
3     def daftarKaryawan():
4         namaKaryawan = input('Masukkan Nama Karyawan : ').upper()
5         umur = int(input('Masukkan Umur Karyawan : '))
6         gaji = int(input('Masukkan gaji pokok karyawan : '))
7         return namaKaryawan, umur, gaji
8
9     def simpanDataKaryawan(namaKaryawan, umur, gaji):
10        listKaryawan.append(namaKaryawan)
11        listKaryawan.append(umur)
12        listKaryawan.append(gaji)
13        print('Data Karyawan berhasil di simpan')
14        print('-----')
15
16        namaKaryawan, umur, gaji= daftarKaryawan()
17        simpanDataKaryawan(namaKaryawan, umur, gaji)
18        return listKaryawan
19
```

Penjelasan :

Pada modul create, program akan membuat fungsi dataKaryawan yang berisi list kosong dan fungsi daftarKaryawan lalu di dalam nya terdapat inputan user yang berisi nama, umur, dan gaji pokok karyawan, dan mereturn nya, lalu dibuat juga fungsi simpanDataKaryawan yang menerima parameter umur dan gaji lalu mengappend nya ke list kosong yang sudah di buat di atas setelah itu kita return list tersebut



```
index.py readList.py X updateList.py createList.py deleteList.py
CRUDList > readList.py > ...
1 def cetakDataKaryawan(databaseKaryawan):
2     jmldata = 3
3     print()
4     print('Berikut adalah data Karyawan: ')
5     print('-----')
6     print(f'|\\tNama Karyawan\\t|\\tUmur Karyawan\\t|\\tGaji Karyawan\\t|')
7     print('-----')
8     for i in range (len(databaseKaryawan)):
9         print(f'|\\t {databaseKaryawan[i][jmldata-jmldata]} \\t|\\t {databaseKaryawan[i][jmldata-2]}')
10        print('-----')
11        print()
12
```

Penjelasan : pada modul read, didalamnya terdapat fungsi cetakDataKaryawan yang menerima parameter databaseKaryawan, setelah itu melakukan perulangan dan mencetak seluruh data yang ada di dalam list databaseKaryawan


```

index.py  readList.py  updateList.py X  createList.py  deleteList.py
CRUDList > updateList.py > editData
7         namaKaryawan = input('Masukkan nama karyawan yang ingin di edit : ').upper()
8         for karyawan in databaseKaryawan:
9             if namaKaryawan == karyawan[0]:
10                print(f'Data ditemukan {karyawan} silahkan edit data tersebut')
11                karyawan.clear()
12                namaKaryawanBaru = input('Masukkan Nama karyawan yang baru : ').upper()
13                umur = int(input('Masukkan umur karyawan : '))
14                gaji = int(input('Masukkan gaji pokok karyawan : '))
15                karyawan.append(namaKaryawanBaru)
16                karyawan.append(umur)
17                karyawan.append(gaji)
18                print('Data berhasil di edit')
19                print('-----')
20                print()
21            else:
22                print('Data karyawan tidak ditemukan.')
23                print()
24

```

Penjelasan : Pada modul update terdapat fungsi editData yang menerima parameter databaseKaryawan lalu di dalam nya kita melakukan pengecekan yang jika data di dalam list databaseKaryawan kosong maka data akan menampilkan bahwa data masih kosong, jika tidak user akan di minta memasukkan kata kunci berupa nama karyawan, jika ada nama karyawan yang sama dengan yang ada di dalam databaseKaryawan, maka data tersebut dapat di ganti dan di update

```

index.py  readList.py  updateList.py  createList.py  deleteList.py X
CRUDList > deleteList.py > ...
1  def hapusData(databaseKaryawan):
2      namaKaryawan = input('Masukkan nama karyawan yang ingin dihapus: ').upper()
3      dataDitemukan = False
4
5      for i in range(len(databaseKaryawan)):
6          if namaKaryawan == databaseKaryawan[i][0].upper():
7              dataDitemukan = True
8              print(f'Data ditemukan: {databaseKaryawan[i]}, silakan hapus data tersebut.')
9              permission = input('Apakah Anda ingin menghapus data ini? y/n: ').lower()
10
11              if permission == 'y':
12                  del databaseKaryawan[i]
13                  print('Data berhasil dihapus.')
14                  print()
15              else:
16                  print('Data tidak dihapus.')
17                  print()
18              break
19
20      if dataDitemukan == False:
21          print('Data karyawan tidak ditemukan.')
22          print()
23

```

Penjelasan : modul delete, sama dengan modul update hanya saja pada modul ini terdapat fungsiData yang cara kerja nya sama dengan fungsi editData di modul update, tetapi di akhir terdapat pilihan lagi sebagai konfirmasi ulang jika user memilih y, maka data akan di hapus dan selain itu data tidak akan di hapus.

```
index.py x readList.py updatelist.py createList.py deletelist.py
CRUDList > index.py > ...
1 import createList
2 import readList
3 import updatelist
4 import deletelist
5 databaseKaryawan = []
6
7 print('Selamat datang di program CRUD Karyawan')
8 while True :
9     print('Silahkan pilih menu operasi yang anda inginkan : ')
10    menu = int(input('1. Membuat data karyawan\n2. Menampilkan database karyawan\n3. Mengupdate data
11    if menu == 1:
12        data = int(input('Berapa banyak karyawan yang ingin di daftarkan? '))
13        for i in range(data):
14            isiDb = createList.dataKaryawan()
15            databaseKaryawan.append(isiDb)
16        elif menu == 2:
17            readList.cetakDataKaryawan(databaseKaryawan)
18        elif menu == 3:
19            print('Untuk mengupdate data karyawan, silahkan masukkan nama karyawan!!!')
20            updatelist.editData(databaseKaryawan)
21        elif menu == 4:
22            print('Untuk menghapus data karyawan, silahkan masukkan nama karyawan !!!')
23            deletelist.hapusData(databaseKaryawan)
24        elif menu == 5:
25            print('Anda memilih keluar !!! Terima kasih')
26            break
27
28
```

Penjelasan : Pada program ini akan mengimport semua modul yang telah di buat,lalu membuat list kosong untuk menyimpan semua data karyawan, lalu akan masuk ke dalam perulangan while True, jika user menginputkan 1, maka program akan meminta user memasukkan jumlah karyawan yang ingin di data dan membuat perulangan sesuai jumlah tersebut dan di dalam perulangan itu akan di jalankan fungsi dataKaryawan yang ada di modul createList

Jika user menginputkan 2, maka program pada modul readList akan di jalankan, jika 3 maka program pada modul updateList akan di jalankan, jika 4 maka program pada modul deleteList akan di jalankan dan jika memilih 5 maka program akan keluar dari perulangan

Output :

A. Create

```
python.exe "d:/Kuliah/Tugas Kuliah/Semester 1/Matkul/Prakti
Selamat datang di program CRUD Karyawan
Silahkan pilih menu operasi yang anda inginkan :
1. Membuat data karyawan
2. Menampilkan database karyawan
3. Mengupdate data karyawan
4. Menghapus data karyawan
5. Keluar
Pilihah anda : 1
Berapa banyak karyawan yang ingin di daftarkan? 2
Masukkan Nama Karyawan : Rafli
Masukkan Umur Karyawan : 17
Masukkan gaji pokok karyawan : 3000000
Data Karyawan berhasil di simpan
```

B. Read

Pilihah anda : 2

Berikut adalah data Karyawan:

Nama Karyawan	Umur Karyawan	Gaji Karyawan
RAFLI	17	3000000
PRATAMA	18	2800000

C. Update

Pilihah anda : 3

Untuk mengupdate data karyawan, silahkan masukkan nama karyawan!!!

Masukkan nama karyawan yang ingin di edit : rafli

Data ditemukan ['RAFLI', 17, 3000000] silahkan edit data tersebut

Masukkan Nama karyawan yang baru : liraf

Masukkan umur karyawan : 17

Masukkan gaji pokok karyawan : 4000000

Data berhasil di edit

LIRAF	17	4000000
PRATAMA	18	2800000

D. Delete

Pilihah anda : 4

Untuk menghapus data karyawan, silahkan masukkan nama karyawan !!!

Masukkan nama karyawan yang ingin dihapus: liraf

Data ditemukan: ['LIRAF', 17, 4000000], silakan hapus data tersebut.

Apakah Anda ingin menghapus data ini? y/n: y

Data berhasil dihapus.

Silahkan pilih menu operasi yang anda inginkan :

1. Membuat data karyawan
2. Menampilkan database karyawan
3. Mengupdate data karyawan
4. Menghapus data karyawan
5. Keluar

Pilihah anda : 2

Berikut adalah data Karyawan:

Nama Karyawan	Umur Karyawan	Gaji Karyawan
PRATAMA	18	2800000

1.2.2 Soal 2

Buatlah program kalkulator sederhana dimana operasi aritmatikanya menggunakan yang dibuat sendiri. Dan panggil menggunakan program utamanya pada file kode yang berbeda.

```
index.py  operasiAritmatika.py X
Kalkulator > operasiAritmatika.py > perkalian

1  def pertambahan ():
2      print('Anda memilih operasi pertambahan')
3      a = int(input('Masukkan angka pertama : '))
4      b = int(input('Masukkan angka kedua : '))
5      print(f'Hasil pertambahan dari {a} + {b} adalah {a+b}')
6      return a + b
7
8  def pengurangan():
9      print('Anda memilih operasi pengurangan')
10     a = int(input('Masukkan angka pertama : '))
11     b = int(input('Masukkan angka kedua : '))
12     print(f'Hasil pengurangan dari {a} - {b} adalah {a - b}')
13     return a - b
14
15  def perkalian():
16     print('Anda memilih operasi perkalian')
17     a = int(input('Masukkan angka pertama : '))
18     b = int(input('Masukkan angka kedua : '))
19     print(f'Hasil perkalian dari {a} x {b} adalah {a*b}')
20     return a * b
21
22  def pembagian ():
23     print('Anda memilih operasi pembagian')
24     a = int(input('Masukkan angka pertama : '))
25     b = int(input('Masukkan angka kedua : '))
26     print(f'Hasil pembagian dari {a} : {b} adalah {a/b}')
27     return a / b
28
29  def modulo ():
30     print('Anda memilih operasi Modulo')
31     a = int(input('Masukkan angka pertama : '))
32     b = int(input('Masukkan angka kedua : '))
33     print(f'Hasil modulo dari {a} mod {b} adalah {a%b}')
34     return a % b
35
36  def pangkat ():
37     print('Anda memilih operasi perpangkatan')
38     a = int(input('Masukkan angka pertama : '))
39     b = int(input('Masukkan angka kedua : '))
40     print(f'Hasil perpangkatan dari {a} ** {b} adalah {pow(a,b)}')
41     return pow(a,b)
```

Penjelasan : Pada modul ini kita akan membuat fungsi yang sesuai dengan operasi matematika pada umumnya seperti pertambahan, pengurangan, perkalian, pembagian, modulo, dan pangkat, lalu di setiap fungsi tersebut user akan di minta memasukkan dua buah inputan dan dua buah inputan tersebut akan di hitung sesuai dengan operasi aritmatika yang di pilih

```
index.py  X  operasiAritmatika.py
Kalkulator > index.py > ...
1  import operasiAritmatika
2  print('-----Selamat Datang-----')
3  while True:
4      operasi = int(input('Silahkan pilih operasi yang anda inginkan :\n1.
5      if operasi == 1:
6          operasiAritmatika.pertambahan()
7      elif operasi == 2:
8          operasiAritmatika.perkurangan()
9      elif operasi == 3:
10         operasiAritmatika.perkalian()
11     elif operasi == 4:
12         operasiAritmatika.pembagian()
13     elif operasi == 5:
14         operasiAritmatika.modulo()
15     elif operasi == 6:
16         operasiAritmatika.pangkat()
17     elif operasi == 7:
18         print('Anda memilih keluar!!! Terima kasih')
19         break
20     else:
21         print('Silahkan pilih operasi yang benar')
```

Pada modul ini kita memanggil atau mengimport modul operasiAritmatika yang sebelumnya sudah kita buat, kita akan masuk ke dalam perulangan while True, jika user memilih 1 maka fungsi penambahan akan dijalankan, jika 2 maka fungsi pengurangan akan di jalankan, jika 3 maka fungsi perkalian akan di jalankan, jika 4 maka fungsi pembagian akan dijalankan, jika 5 maka fungsi modulo akan dijalankan, jika 6 maka fungsi pangkat akan di jalankan, jika user memilih 7 maka kita akan keluar dari perulangan dan jika selain angka angka tersebut, user akan di minta memasukkan angka yang benar

Output :

```
Silahkan pilih operasi yang anda inginkan :
1. Pertambahan
2. Pengurangan
3. Perkalian
4. Pembagian
5. Modulo
6. Perpangkatan
7. Keluar
1
Anda memilih operasi penambahan
Masukkan angka pertama : 5
Masukkan angka kedua : 4
Hasil penambahan dari 5 + 4 adalah 9
```

```
2
Anda memilih operasi pengurangan
Masukkan angka pertama : 5
Masukkan angka kedua : 7
Hasil pengurangan dari 5 - 7 adalah -2
```

```
3
Anda memilih operasi perkalian
Masukkan angka pertama : 2
Masukkan angka kedua : 5
Hasil perkalian dari 2 x 5 adalah 10
```

```

4
Anda memilih operasi pembagian
Masukkan angka pertama : 10
Masukkan angka kedua : 5
Hasil pembagian dari 10 : 5 adalah 2.0

5
Anda memilih operasi Modulo
Masukkan angka pertama : 10
Masukkan angka kedua : 3
Hasil modulo dari 10 mod 3 adalah 1

6
Anda memilih operasi perpangkatan
Masukkan angka pertama : 2
Masukkan angka kedua : 5
Hasil perpangkatan dari 2 ** 5 adalah 32

7.Keluar
7
Anda memilih keluar!!! Terima kasih

```

1.2.3 Soal 3

Buatlah program menghitung luas bangun datar dan volume bangun ruang (7 menu) dimana operasi hitung luas dan volumenya terpisah pada modul yang berbeda dan dipanggil pada program utama :

```

LuasDanVolume > bangunDatar.py > luasBangunDatar > lingkaran
1 def luasBangunDatar():
2
3     def persegi():
4         a = int(input('Masukkan sisi persegi: '))
5         luasPersegi = a * a
6         print(f'Luas persegi dengan sisi {a} cm dan {a} cm adalah {luasPersegi} cm2')
7         return luasPersegi
8
9     def persegiPanjang():
10        a = int(input('Masukkan panjang: '))
11        b = int(input('Masukkan lebar: '))
12        luasPersegiPanjang = a * b
13        print(f'Luas persegi panjang dengan panjang {a} cm dan lebar {b} cm adalah {luasPersegiPanjang} cm2')
14        return luasPersegiPanjang
15
16    def segitiga():
17        a = int(input('Masukkan sisi alas: '))
18        t = int(input('Masukkan tinggi: '))
19        luasSegitiga = (a * t) / 2
20        print(f'Luas segitiga dengan alas {a} cm dan tinggi {t} cm adalah {luasSegitiga} cm2')
21        return luasSegitiga
22
23    def jajargenjang():
24        a = int(input('Masukkan alas : '))
25        t = int(input('Masukkan tinggi : '))
26        luasJajarGenjang = a * t
27        print(f'Luas jajargenjang dengan alas {a} cm dan tinggi {t} cm adalah {luasJajarGenjang} cm2')
28        return luasJajarGenjang
29
30    def belahKetupat():
31        d1 = int(input('Masukkan Diagonal 1 : '))
32        d2 = int(input('Masukkan Diagonal 2 : '))
33        luasBelahKetupat = 0.5 * d1 * d2
34        print(f'Luas belah ketupat dengan diagonal 1 {d1} cm dan diagonal 2 {d2} cm adalah {luasBelahKetupat} cm2')
35        return luasBelahKetupat
36

```

```
bangunDatar.py X bangunRuang.py index.py
LuasDanVolume > bangunDatar.py > luasBangunDatar > lingkaran
37 def layangLayang ():
38     d1 = int('Masukkan Diagonal 1 : ')
39     d2 = int('Masukkan Diagonal 2 : ')
40     luasLayangLayang = 0.5 * d1 * d2
41     print(f'Luas layang-layang dengan diagonal 1 {d1} cm dan diagonal 2 {d2} cm adalah {luasLayangLayang} cm2')
42     return luasLayangLayang
43
44 def trapesium():
45     a = int(input('Masukkan panjang: '))
46     b = int(input('Masukkan lebar: '))
47     t = int(input('Masukkan tinggi : '))
48     luasTrapesium = (a + b)/2 * t
49     print(f'Luas Trapesium dengan sisi {a} cm dan {b} cm dengan tinggi {t}cm adalah {trapesium} cm2')
50     return luasTrapesium
51
52 def lingkaran():
53     phi = 3.14
54     r = int(input('Masukkan jari jari lingkaran : '))
55     luasLingkaran = phi * pow(r,r)
56     print(f'Luas Lingkaran dengan jari-jari {r} cm adalah {luasLingkaran} cm2')
57     return luasLingkaran
58
```

```
bangunDatar.py X bangunRuang.py index.py
LuasDanVolume > bangunDatar.py > luasBangunDatar
49
50 bangunDatar = int(input('Silahkan pilih bangun datar:\n1. Persegi\n2. Persegi panjang\n3. Segitiga
51 if bangunDatar == 1:
52     persegi()
53     print()
54 elif bangunDatar == 2:
55     persegiPanjang()
56     print()
57 elif bangunDatar == 3:
58     segitiga()
59     print()
60 elif bangunDatar == 4:
61     jajarGenjang()
62     print()
63 elif bangunDatar == 5:
64     belahKetupat()
65     print()
66 elif bangunDatar == 6:
67     layangLayang()
68     print()
69 elif bangunDatar == 7:
70     trapesium()
71     print()
72 elif bangunDatar == 8:
73     lingkaran()
74     print()
75 else:
76     print('Silahkan masukkan pilihan yang benar')
77
78
```

Penjelasan :

Pada modul bangunDatar berisi fungsi fungsi menghitung luas luas bangun daar seperti pada bangun datar persegi, persegi panjang, segitiga, jajar genjang, belah ketupat, layang layang, trapesium dan lingkaran, didalam setiap masing masing fungsi tersbeut user akan diminta memasukkan ukuran ukuran yang di minta seperti nilai tinggi, panjang, luas alas dan lain lain, lalu user akan di minta menentukan ingin menghitung luas bangun datar apa

```
bangunDatar.py  bangunRuang.py X  index.py
LuasDanVolume > bangunRuang.py > volumeBangunRuang
1 def volumeBangunRuang():
2     bangunRuang = int(input('Silahkan pilih bangun datar:\n1. Kubus\n2. Balok\n3. Limas Segi Empat\n4. Prisma Segi Tiga\
3     def kubus():
4         a = int(input('Masukkan sisi a: '))
5         volumeKubus = a * a * a
6         print(f'Volume kubus dengan sisi {a}cm adalah {volumeKubus} cm3')
7         return volumeKubus
8
9     def balok():
10        p = int(input('Masukkan panjang: '))
11        l = int(input('Masukkan lebar: '))
12        t = int(input('Masukkan tinggi: '))
13        volumeBalok = p*l*t
14        print(f'Volume balok dengan panjang{p} cm lebar {l}cm dan tinggi {t}cm adalah {volumeBalok}cm3')
15        return volumeBalok
16
17    def limasSegiEmpat():
18        p = int(input('Masukkan sisi alas: '))
19        l = int(input('Masukkan sisi alas: '))
20        t = int(input('Masukkan tinggi limas: '))
21        la = p * l
22        volumeLimasSegiEmpat = 1/3 * la * t
23        print(f'Volume limas segi empat dengan luas alas {la} cm2 dan tinggi {t} cm adalah {volumeLimasSegiEmpat}cm3')
24        return volumeLimasSegiEmpat
25
26    def prismaSegiTiga():
27        a = int(input('Masukkan alas: '))
28        t = int(input('Masukkan tinggi alas: '))
29        T = int(input('Masukkan tinggi prisma: '))
30        la = 1/2 * a * T
31        volumePrismaSegiTiga = la * T
32        print(f'Volume prisma segitiga dengan luas alas {la}cm2 dan tinggi {T}cm adalah {volumePrismaSegiTiga}cm3')
33        return volumePrismaSegiTiga
34
```

```
bangunDatar.py  bangunRuang.py X  index.py
LuasDanVolume > bangunRuang.py > volumeBangunRuang
34
35    def limasSegiTiga():
36        a = int(input('Masukkan alas segitiga: '))
37        t = int(input('Masukkan tinggi segitiga: '))
38        T = int(input('Masukkan tinggi prisma: '))
39        la = 1/2 * a * T
40        volumeLimasSegiTiga = 1/3 * la * T
41        print(f'Volume limas segitiga dengan luas alas {la}cm2 dan tinggi {T}cm adalah {volumeLimasSegiTiga}cm3')
42        return volumeLimasSegiTiga
43
44    def silinder():
45        phi = 3.14
46        r = int(input('Masukkan jari jari alas: '))
47        t = int(input('Masukkan tinggi silinder: '))
48        volumeSilinder = phi * r*r * t
49        print(f'Volume silinder dengan jari jari {r} cm dan tinggi {t} cm adalah {volumeSilinder}cm3')
50        return volumeSilinder
51
52    def kerucut():
53        phi = 3.14
54        r = int(input('Masukkan jari jari alas: '))
55        t = int(input('Masukkan tinggi silinder: '))
56        volumeKerucut = 1/3 * phi * pow(r,r) * t
57        print(f'Volume kerucut dengan jari jari {r} cm dan tinggi {t} cm adalah {volumeKerucut}cm3')
58        return volumeKerucut
59
60    def bola():
61        phi = 3.14
62        r = int(input('Masukkan jari jari alas: '))
63        volumeBola = 4/3 * phi * r*r*r
64        print(f'Volume bola dengan jari jari {r} cm adalah {volumeBola}cm3')
65        return volumeBola
66
```



```
bangunDatar.py  bangunRuang.py X  index.py
LuasDanVolume > bangunRuang.py > volumeBangunRuang
60     if bangunRuang == 1:
61         kubus()
62         print()
63     elif bangunRuang == 2:
64         balok()
65         print()
66     elif bangunRuang == 3:
67         limasSegiEmpat()
68         print()
69     elif bangunRuang == 4:
70         prismaSegiTiga()
71         print()
72     elif bangunRuang == 5:
73         limasSegiTiga()
74         print()
75     elif bangunRuang == 6:
76         silinder()
77         print()
78     elif bangunRuang == 7:
79         kerucut()
80         print()
81     elif bangunRuang == 8:
82         bola()
83         print()
84     else:
85         print('Silahkan masukkan pilihan yang benar')
```

Penjelasan :

Pada modul bangunRuang berisi fungsi fungsi menghitung volume bangun ruang seperti bangun ruang kubus, balok, limas segi empat, prisma segi tiga, limas segitiga, silinder, kerucut, bola, lalu di dalam setiap fungsi tersebut user akan di minta memasukkan ukuran ukuran bangun ruang yang dibutuhkan seperti tinggi, lebar dan luas alas bangun ruang, lalu user akan di minta menentukan ingin menghitung volume bangun ruang yang mana.

```
bangunDatar.py  bangunRuang.py  index.py X
LuasDanVolume > index.py > ...
1  import bangunDatar
2  import bangunRuang
3  while True:
4      operasi = int(input('Silahkan pilih Operasi : \n1. Mengitung Luas Bangun datar \n2. Menghitung Volume Bangun ruang\n3. Kelu
5      if operasi == 1:
6          print('Anda memilih opsi 1. menghitung luas bangun datar, Silahkan pilih bangun datar yang anda inginkan')
7          bangunDatar.luasBangunDatar()
8      elif operasi == 2:
9          print('Anda memilih opsi 2. menghitung luas bangun ruang, Silahkan pilih bangun ruang yang anda inginkan')
10         bangunRuang.volumeBangunRuang()
11     elif operasi == 3:
12         print('Anda memilih keluar, Terima kasih')
13         break
14     else:
15         print('Silahkan masukkan pilihan operasi yang tepat')
16
```

Penjelasan :

Pada modul index, tempat kita mengimport modul modul yang sudah di buat, dalam program ini akan masuk ke dalam perulangan while True, dan didalamnya kita akan di minta menentukan ingin menghitung luas bangun datar atau volume bangun ruang, jika user memilih 1, maka fungsi luasBangunDatar akan di jalankan, jika user memilih 2 maka fungsi volumeBangunRuang yang akan di jalankan, jika user memilih 3 maka program akan selesai dan keluar dari perulangan while dan jika selain itu maka user akan di minta memasukkan pilihan yang benar

Output :

```
PS D:\Kuliah\Tugas Kuliah\Semester 1\Matkul\Praktikum Pemograman Komputer\Laporan Harian\Pertemuan ke-11> &
Python312/python.exe "d:/Kuliah/Tugas Kuliah/Semester 1/Matkul/Praktikum Pemograman Komputer/Laporan Harian
Silahkan pilih Operasi :
1. Mengitung Luas Bangun datar
2. Menghitung Volume Bangun ruang
3. Keluar
Pilihan anda : 1
Anda memilih opsi 1. menghitung luas bangun datar, Silahkan pilih bangun datar yang anda inginkan
Silahkan pilih bangun datar:
1. Persegi
2. Persegi panjang
3. Segitiga
4. Jajar Genjang
5. Belah Ketupat
6. Layang - Layang
7. Trapesium
8. Lingkaran
pilihan anda : 3
Masukkan sisi alas: 4
Masukkan tinggi: 6
Luas segitiga dengan alas 4 cm dan tinggi 6 cm adalah 12.0 cm2

Silahkan pilih Operasi :
1. Mengitung Luas Bangun datar
Anda memilih opsi 2. menghitung luas bangun ruang, Silahkan pilih bangun ruang yang anda inginkan
Silahkan pilih bangun datar:
1. Kubus
2. Balok
3. Limas Segi Empat
4. Prisma Segi Tiga
5. Limas Segi Tiga
6. Silinder
7. Kerucut
8. Bola
9. Pilihan Anda : 5
Masukkan alas segitiga : 4
Masukkan tinggi segitiga : 6
Masukkan tinggi prisma : 12
Volume limas segitiga dengan luas alas 24.0cm2 dan tinggi 12cm adalah 96.0cm3

Silahkan pilih Operasi :
1. Mengitung Luas Bangun datar
2. Menghitung Volume Bangun ruang
3. Keluar
Pilihan anda : 3
Anda memilih keluar, Terima kasih
PS D:\Kuliah\Tugas Kuliah\Semester 1\Matkul\Praktikum Pemograman Komputer\Laporan Harian\Pertemuan ke-11> |
```

1.3. Kesimpulan dan Saran

1.3.1. Kesimpulan

Dari praktikum ini dapat disimpulkan :

1. Kita dapat membuat sebuah program yang dapat di panggil berkali kali dengan menggunakan import modul
2. Setiap string dan int memiliki fungsinya masing-masing dan fungsi tersebut dapat kita manfaatkan untuk mempercepat proses pemograman kita
3. Pada python sudah ter include banyak sekali modul modul yang dapat kita manfaatkan dengan cara mengimport modul tersebut

1.3.2. Saran

Saran terhadap praktikum ini yaitu:

1. Selalu perhatikan bahwa ada beberapa fungsi yang di haruskan menerima parameter dan ada yang tidak memerlukan parameter, kita dapat mengetahui nya lewat membaca dokumentasi pyhton
2. Saat membuat modul, pastikan kita mengimport nya sesuai dengan alamat atau lokasi tempat modul kita di buat
3. Kita dapat mengetahui modul modul yang tersedia dengan cara membaca nya di dokumentasi resmi pyhton

DAFTAR PUSTAKA

Modul *Perkuliahan dan Praktikum Algoritma dan Pemrograman* Oleh Ilhamsyah, S.Si., M.Cs.