



Algoritma dan Pemrograman Python I

Sistem Informasi

Penulis: Yusri Ikhwani

Universitas: Universitas Islam Kalimantan Muhammad Arsyad Al Banjari

Tanggal: Aug 15, 2022

Versi: 1

Bio: ALGO-I



Victory won't come to us unless we go to it.

DAFTAR ISI

BAB 1	Mengubah Flowcart Menjadi Kode Program	1
1.1	Pendahuluan	1
1.2	Instalasi Interpreter	1
1.3	Menjalankan Python pada Visual Studio Code	3
1.4	Implementasi Penugasan (Assignment) dan Proses	4
BAB 1	Latihan	7
BAB 2	PRAKTIKUM PENGENALAN TIPE DATA	9
2.1	Dasar Teori	9
BAB 2	Latihan	9
BAB 3	PRAKTIKUM if else dan elif (PERCABANGAN BERSYARAT)	12
3.1	Dasar Teori	12
BAB 3	Latihan	12
BAB 4	PRAKTIKUM MATCH AND CASE (PERCABANGAN BERSYARAT)	16
4.1	Dasar Teori	16
BAB 4	Latihan	16
BAB 5	FOR LOOP (PERULANGAN ATAU LOOPING)	20
5.1	Dasar Teori	20
BAB 5	Latihan	20
BAB 6	PRAKTIKUM WHILE IF CONDITION (PERULANGAN / LOOPING)	25
6.1	Dasar Teori	25
BAB 6	Latihan	25
BAB 7	PRAKTIKUM PROCEDURE DAN FUNGSI)	30
7.1	Dasar Teori	30
BAB 7	Latihan	30
BAB 8	PRAKTIKUM ARRAY	34
8.1	Dasar Teori	34
BAB 8	Latihan	36
Daftar Pustaka		38

BAB 1 Mengubah Flowcart Menjadi Kode Program

Tujuan Praktek

- ❑ Memahami cara instalasi perangkat lunak interpreter Python 3
- ❑ Memahami cara mengubah diagram alir menjadi kode dengan bahasa pemrograman Python
- ❑ Memahami cara mengubah diagram alir menjadi kode dengan bahasa pemrograman Python

1.1 Pendahuluan

Setelah memahami cara menggambar diagram alir maka untuk implementasinya digunakan bahasa pemrograman Python. Bahasa ini dipilih karena merupakan bahasa pemrograman yang digunakan pada mata kuliah Algoritma dan Pemrograman saat ini. Bahasa pemrograman **Python** dapat dituliskan dengan berbagai macam text editor seperti notepad, notepad++, vi/vim, sublime text dan text editor lainnya, bahkan dapat juga dituliskan pada interpreter yang sudah mempunyai Integrated Development Environment (IDE) seperti PyCharm, Anaconda, Spyder, Atom dan Visual studio code. Untuk praktikum ini akan menggunakan notepad / notepad++ untuk programmer pemula, namun tidak menutup kemungkinan anda untuk menggunakan IDE Python lain yang pernah anda gunakan. Perlu dipahami bahwa penulisan kode python pada sesi ini bukan best practice atau cara yang terbaik dalam menuliskan kode, melainkan hanya pengenalan dan implementasi cepat dari diagram alir yang sudah digambar pada sesi sebelumnya.

1.2 Instalasi Interpreter

1. Buka web browser kemudian akses laman <http://www.python.org/downloads/windows>
2. Klik tautan bertuliskan Latest **Python** 3 Release - **Python** 3.11.2 untuk mengunduh interpreter python



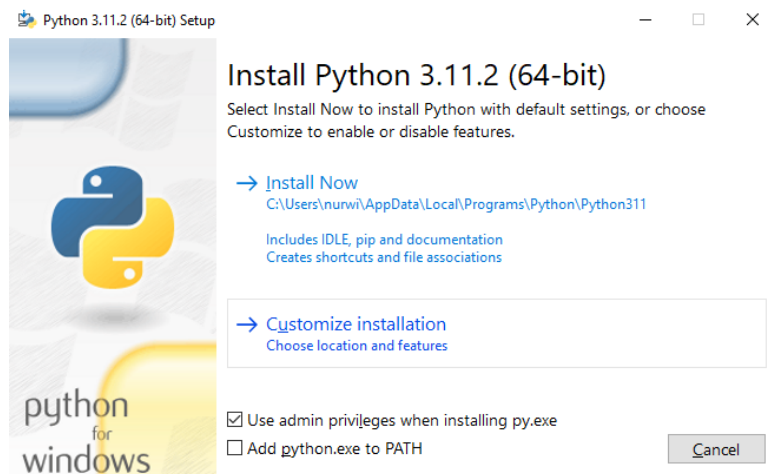
Gambar 1.1: Python Release untuk Windows

3. Klik kanan pada file installer yang baru saja di unduh, kemudian klik run as administrator dan pilih YES



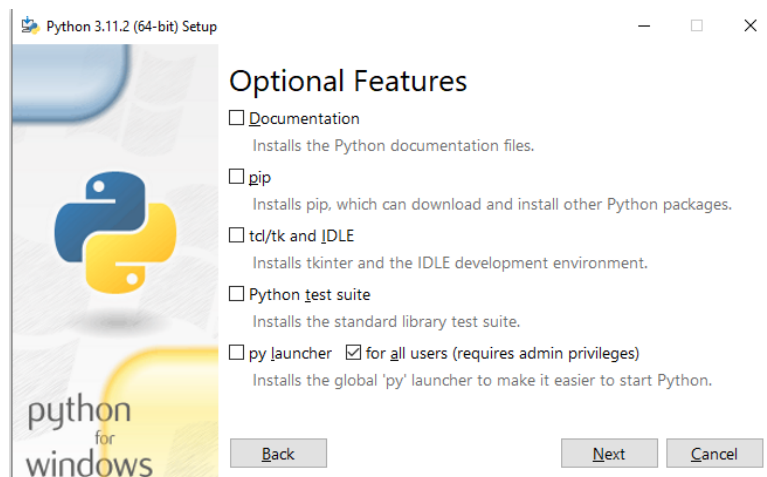
Gambar 1.2: File instalasi Python

4. Jika berhasil maka akan dibawa pada tampilan seperti berikut, Kemudian pilih Customize installation



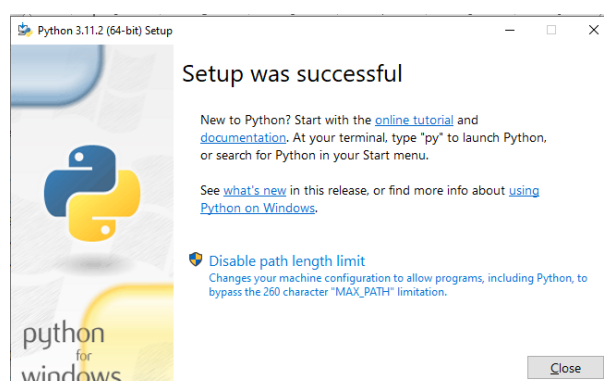
Gambar 1.3: File instalasi Python

5. Hilangkan semua contend pada tampilan, dengan mengklik pada masing-masing fitur, sehingga terlihat seperti berikut:



Gambar 1.4: File instalasi Python

6. Kemudian klik next, hilangkan contend pada fitur “Create shortcut for installed applications”, atur ulang lokasi instalasi dengan mengklik tombol “browse” lalu pilih Local Disk (C:)> Program Files > Python 3. 11.2 lalu klik “Install” (jika sub folder “Python 3. 11.2” belum ada didalam folder Program Files, silahkan buat sub foldernya terlebih dahulu). Jika berhasil maka akan terlihat seperti tampilan berikut, lalu klik close. selesai



7. Selanjutnya adalah melakukan pengaturan pada PATH , sehingga dapat berjalan pada command windows (CMD). Buka command prompt dan ketikkan perintah-perintah berikut:

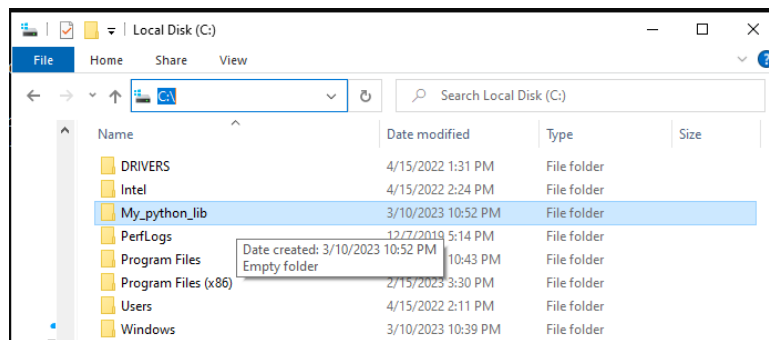
- CD C: \
- set PATH="C:\Program Files\Python 3.11.2";%PATH%
- set PYTHONPATH=%PYTHONPATH%;C:\My_python_lib
- python

8. Jika berhasil, maka akan tampil seperti gambar berikut:

```
C:\>set PATH="C:\Program Files\Python 3.11.2";%PATH%
C:\>set PYTHONPATH=%PYTHONPATH%;C:\My_python_lib
C:\>python
Python 3.11.2 (tags/v3.11.2:87ead1, Feb 7 2023, 16:38:35) [MSC v.1934 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 1 + 1
2
```

Gambar 1.5: Output Python

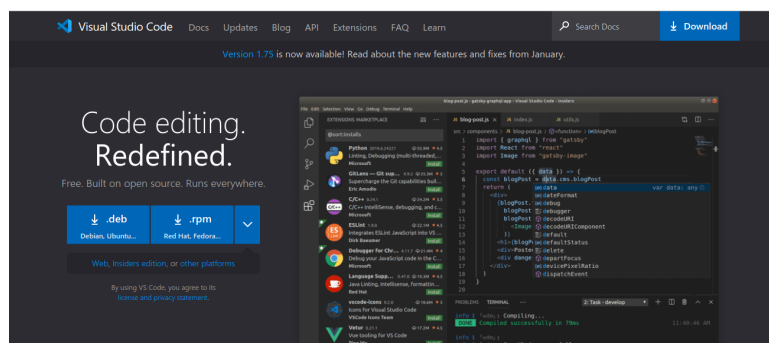
9. jika folder "My_python_lib" belum terdapat pada drive C:\, silahkan buat folder tersebut terlebih dahulu, seperti yang di tunjukkan pada gambar berikut:



Gambar 1.6: Output Python

1.3 Menjalankan Python pada Visual Studio Code

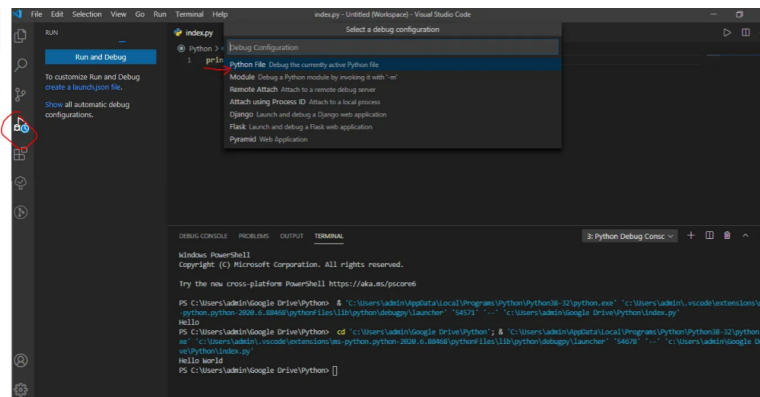
1. Install VS Code terlebih dahulu melalui halaman <https://code.visualstudio.com/>



2. Setelah itu, buka VS Code dan buatlah sebuah file baru dengan nama index.py. Pada contoh berikut, file diletakkan didalam folder pythonsource.

3. Mari kita menuliskan code print("Hello World") pada editor. Kita akan mencoba menjalankan dengan mengklik tanda panah yang dilingkari seperti pada gambar. Anda juga bisa menggunakan shortcut Ctrl +

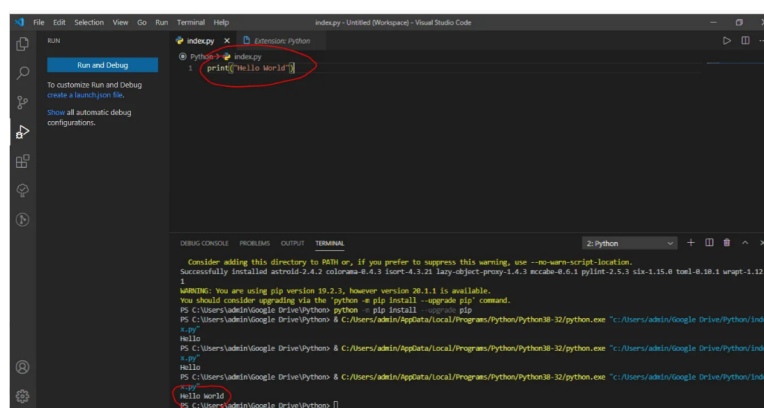
Shift + D



4. Pada bagian Ekstensions (Ctrl + Shift + X) VS Code, carilah ekstension Python dan install seperti pada gambar berikut. Ikuti perintah selanjutnya seperti pada bagian sudut kanan bawah, kita diminta untuk menginstall Linter. Silakan klik install pada Linter tersebut.



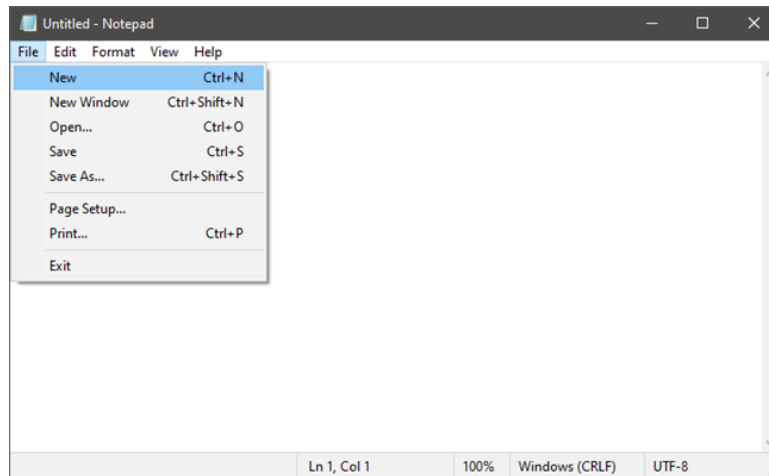
5. Jalankan kembali code ini dengan klik kanan dan pilih “run python file in terminal”. Shortcut : Ctrl + Shift + D
6. Perhatikan pada bagian terminal, jika Anda menemukan tulisan “Hello World”, maka selamat !!!, Anda berhasil menjalankan Python melalui editor VS Code.



1.4 Implementasi Penugasan (Assignment) dan Proses

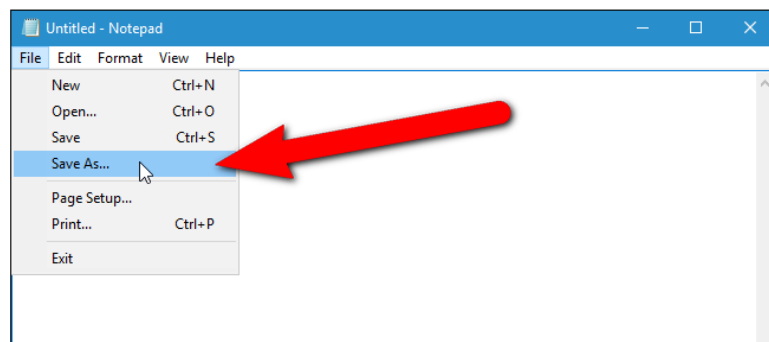
1. Buatlah folder pada C:\Documents and Settings\tulis_nama_anda\Documents\pythonsource

2. Bukalah perangkat lunak Notepad atau Notepad++
3. Klik file;new untuk membuat file baru



Gambar 1.7: File baru pada notepad

4. Klik file>save as untuk menyimpan file



Gambar 1.8: File Save As

5. Tentukan tempat penyimpanan pada folder python source yang telah dibuat pada langkah 1, dan beri nama file
01_halo.py
6. Masukkan kode berikut ini pada notepad

```
print("Hallo Semuanya")
```

7. Buka terminal atau command prompt dan masuklah ke folder atau directory pythonsource
8. Jalankan perintah:

```
python3 01_halo.py
```

9. Jika kode program berhasil, berikut tampilan yang akan muncul

```
station@worked:~/texmf/algo$ python3 01_halo.py
Hallo Semuanya
```

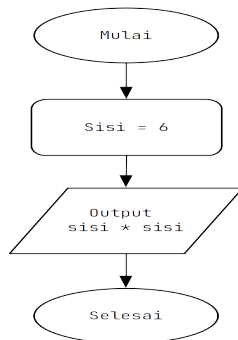
Gambar 1.9: Output Program



Informasi:

Selamat jika kode program anda berhasil, selamat juga bagi yang masih belum berhasil. Seperti peribahasa “Tak ada gading yang tak retak”, tidak ada programmer yang tidak melakukan kesalahan, sila lakukan debugging sesuai kode galat yang tampil.

10. Lanjutkan dengan membuat file Python baru, beri nama kode 02_sisi.py dan ketikkan kode di bawah ini:



```

1      sisi = 6
2      output = sisi * sisi
3      print("Hasil Luas:", output)
  
```



Keterangan Kode Program:

Tabel 1.1: Keterangan Sintaksis Program

Sintaks	Keterangan
<code>sisi = 6</code>	Perintah mengisikan nilai 6 ke variable sisi (assignment), digambarkan dengan bentuk flowcart proses
<code>output = sisi * sisi</code>	Perintah mengisikan jumlah perkalian sisi * sisi ke variable output (assignment), digambarkan dengan bentuk flowcart proses I/O
<code>print("Hasil Luas:", output)</code>	Perintah menampilkan isi dari output

11. Buka kembali terminal atau command prompt dan masuklah ke folder atau directory pythonsource

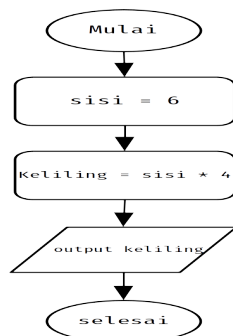
12. Jalankan perintah:

```
python3 02_sisi.py
```

13. Jika kode program berhasil, berikut tampilan yang akan muncul

```
Hasil Luas: 36
```

14. Lanjutkan dengan membuat file python baru, beri nama 03_sisi.py dan ketikkan kode di bawah ini:



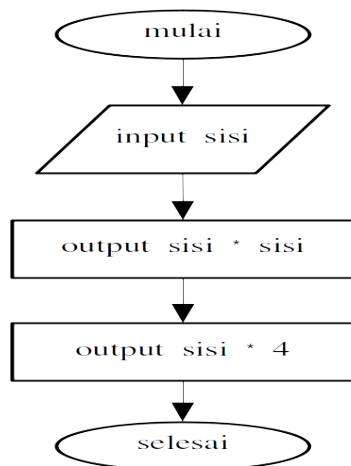
```

1      sisi = 6
2      keliling = sisi * 4
3      output = keliling
4      print("Hasil Keliling:", output)
  
```

15. Jalankan program tersebut, berikut tampilan yang seharusnya muncul:

16. Percobaan terakhir bertujuan untuk kita mengetahui cara menerima inputan dari user dengan perintah `input()`, buat file python baru, beri nama 01_input.py dan ketikkan kode berikut:


```
Hasil Keliling: 24
```



```

1     print("Masukkan Sisi :")
2     sisi = int(input())
3     print("Hasil luas: ", sisi*sisi)
4     print("Hasil keliling: ", sisi*4)
  
```

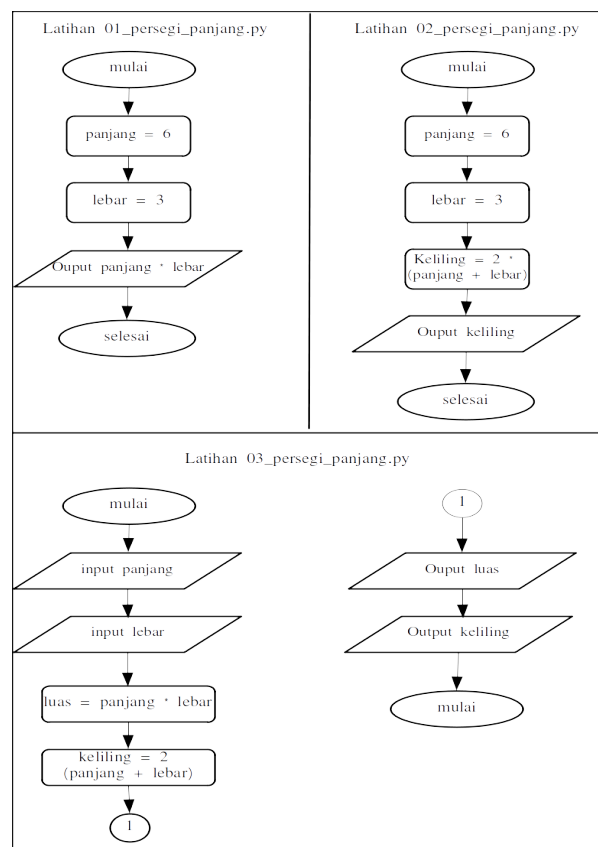
17. Jalankan program tersebut, masukkan sisi = 6, berikut tampilan yang seharusnya muncul:

```

Masukkan Sisi :
6
Hasil luas: 36
Hasil keliling: 24
  
```

📖 BAB 1 Latihan 📖

1. Tuliskan kode program python untuk diagram alir persegi panjang berikut ini(beri penamaan sesuai kolom):



2. Tulis kode program python yang mengubah inputan user berupa suhu dalam Fahrenheit menjadi nilai suhu dalam Celcius, rumus: $Celcius = \frac{5}{9} * (Fahrenheit - 32)$ dan tampilkan hasilnya (Latihan 01_celcius.py)
3. Tulis kode program python yang menerima inputan dua nilai, kemudian hitung dan tampilkan hasil penjumlahan, hasil pengurangan, hasil perkalian, dan hasil pembagiannya (Latihan 1_aritmatik.py)
4. Tuliskan kode program python yang menerima 4 inputan, nilaiHarian, nilaiTugas, nilaiUTS, nilaiUAS. Hitung nilaiAkhir dengan ketentuan bobot nilaiHarian 10%, nilaiTugas 20%, nilaiUTS 30%, dan nilaiUAS 40%, kemudian tampilkan hasilnya (Latihan 01_nilai_kuliah.py)

BAB 2 PRAKTIKUM PENGENALAN TIPE DATA

Tujuan Praktek

- ❑ Mahasiswa bisa mengenal, memahami dan mengenali jenis-jenis data dan variabel dalam python,
- ❑ Memahami memahami data yang diinputkan sesuai dengan tipe variabelnya.

2.1 Dasar Teori

Data adalah penggambaran fakta secara umum dan dilukiskan dalam bentuk symbol, tulisan atau angka. Adapun tipe data tersebut adalah sebagai berikut:

Tabel 2.1: Keterangan tipe data

Tipe Data	Contoh	Penjelasan
Boolean	True atau False	Menyatakan nilai benar (True) yang bernilai 1, atau nilai salah (False) yang bernilai 0
String	"FTI UNISKA Handsome"	Menyatakan karakter kalimat bisa berupa huruf, angka, atau kombinasi keduanya, nilai pada string harus ditulis diantara pada Petik double atau "" atau petik tunggal "
Integer	232 atau 1234	Menyatakan semua bilangan positif
Float	3.14 atau 2.19	Menyatakan semua bilangan yang desimal
Hexadecimal	6a atau 2d3	Menyatakan bilangan dalam format heksa (bilangan berbasis 16)
Complex	1 + 5j	Menyatakan pasangan angka real dan imajiner
List	['abc', 123, 2.23]	Data list dapat menyimpan berbagai tipe data dan isinya bisa diubah-ubah
Tuple	('xyz', 768, 2.23)	Data jenis Tuple yang menyimpan berbagai tipe data tapi isinya tidak bisa diubah
Dictionary	{'nama':'yusri','id':2}	Data jenis Dictionary menyimpan berbagai tipe data berupa pasangan penunjuk dan nilai

🎨 BAB 2 Latihan 🎨

1. Latihan Praktikum 2.1.1

Buatlah Program (inputan string) seperti di bawah ini, dengan nama file latihan2_1_1.py

```
1 print('Program Penampil Data Diri')
2 print('*****')
3 nama = input("Masukkan Nama :")
4 alamat = input("Masukkan Alamat :")
5 hobby = input("Masukkan Hobby :")
6 print("Hallo", nama, "!!!", "kamu tinggal di ", alamat, "dan hobby mu adalah",
    hobby)
```

- Hasil eksekusi program seperti gambar di bawah ini dan masukan nama anda, alamat anda dan hobby anda:

```
Program Penampil Data Diri
*****
Masukkan Nama :yusri
Masukkan Alamat :banjar
Masukkan Hobby :olahraga
Hallo yusri !!! kamu tinggal di banjar dan hobby mu adalah olahraga
```

2. Latihan Praktikum 2.1.2

Buatlah Program (inputan integer) seperti di bawah ini, dengan nama file latihan2_1_2.py

```
1 print('Program Menghitung')
2 print('*****')
3 angka1 = int(input("Ketik Angka Pertama :"))
4 angka2 = int(input("Ketik Angka Kedua :"))
5 hasil = angka1 + angka2
6 print("angka1 + angka2 =", hasil)
```

- Hasil eksekusi program seperti gambar di bawah ini & masukan angka pertama (19), angka kedua (3) :

```
Program Menghitung
*****
Ketik Angka Pertama :19
Ketik Angka Kedua :3
angka1 + angka2 = 22
```

3. Latihan Praktikum 2.1.3

Buatlah Program (inputan float) seperti di bawah ini, dengan nama file latihan2_1_3.py

```
1 print('Program Menghitung Luas Lingkaran')
2 print('*****')
3
4 phi = 3.14
5 r = float(input("Masukkan jari-jari lingkaran: "))
6 luas = phi * r * r
7 print("Luas lingkaran adalah =", luas)
```

- Hasil eksekusi program seperti gambar di bawah ini & masukan jari -jari lingkaran (8) :

```
Program Menghitung Luas Lingkaran
*****
Masukkan panjang jari-jari lingkaran: 8
Luas lingkaran adalah = 200.96
```

4. Latihan Praktikum 2.1.4

Buatlah Program (inputan float) seperti di bawah ini, dengan nama file latihan2_1_4.py

```
1 print('Program Konversi Suhu')
2 print('*****')
3
4 celcius = float(input("Masukkan besaran derajat Suhu celcius: "))
5 fahrenheit = (celcius * 1.8) + 32
6 print("Hasil konversi ke fahrenheit =", fahrenheit)
```

```

Program Konversi Suhu
*****
Masukkan besaran derajat Suhu celcius: 9
Hasil konversi ke fahrenheit = 48.2

```

- Hasil eksekusi program seperti gambar di atas, & masukan suhu derajat celcius (9) :

5. Latihan Praktikum 2.1.5

Lengkapi (modifikasi) program latihan2_1_3.py untuk menghitung keliling lingkaran seperti gambar di bawah ini dan simpan (save) file dengan nama latihan2_1_5.py

Rumus $keliling = 2 * \pi * r$

```

Masukan Jari-jari : 3.6
Luas Lingkaran      = 40.71504079052372
Keliling Lingkaran  = 22.61946710584651

```

6. Latihan Praktikum 2.1.6

Lengkapi (modifikasi) program latihan2_1_4.py dengan menambahkan konversi suhu celcius ke reamur, seperti gambar di bawah ini, dan simpan (save) file dengan nama latihan2_1_6.py

Rumus $reamur = (4 / 5) * celcius$

```

Program Konversi Suhu
*****
Masukkan besaran derajat Suhu celcius: 9
Hasil konversi ke fahrenheit = 48.2
Hasil konversi ke reamur = 7.2

```


BAB 3 PRAKTIKUM if else dan elif (PERCABANGAN BERSYARAT)

Tujuan Praktek

- ❑ Mengetahui syarat logika yang merupakan ungkapan sebagai kondisi untuk mengatur pengekskusi terhadap pernyataan yang mengikuti pernyataan *if else* dan *elif*.

3.1 Dasar Teori

Statemen kendali *if* akan mengecek suatu kondisi dan menentukan apakah kondisi tersebut benar atau salah, kemudian melakukan kegiatan sesuai dengan nilai kondisi tersebut. Jika kondisinya salah maka statemen yang lainnya atau statemen setelah kata cadangan *else* yang akan dikerjakan (*else* diawali dengan titik dua, karena titik dua menunjukkan akhir dari suatu statemen) [1, 2]

Bentuk umum dari statemen *if* adalah sebagai berikut :

```
if <kondisi> <statemen1> else: <statemen2>
```

Dengan kondisi

kondisi yang dites untuk menentukan apakah statemen 1 atau statemen 2 yang akan dikerjakan

Statemen 1

statemen (-statemen) yang akan dikerjakan jika kondisi bernilai *true*

Statement 2

statemen (-statemen) yang akan dikerjakan jika kondisi bernilai *false*.

🎨 BAB 3 Latihan 🎨

1. Latihan Praktikum 3.1.1

Buatlah Program seperti di bawah ini, dengan nama file `latihan3_1_1.py`

```
1 print("Program Menentukan Ganjil atau Genap")
2 print('*****\n')
3 x = int(input("Masukkan bilangan :"))
4
5 if(x > 0):
6     print(x," adalah bilangan positif")
7 else:
8     print(x," adalah bilangan negatif")
```

- Hasil eksekusi program, dimana program dapat menentukan bilangan positif atau bilangan ganjil.

```
Program Menentukan Ganjil atau Genap
*****
Masukkan bilangan :11
11 adalah bilangan positif
```

2. Latihan Praktikum 3.1.2

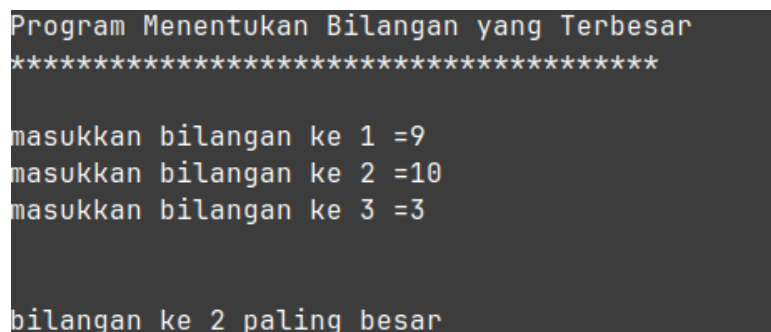
Buatlah Program seperti di bawah ini, dengan nama file latihan3_1_2.py

```

1      print("Program Menentukan Bilangan yang Terbesar")
2      print('*****\n')
3      a = int(input("masukkan bilangan ke 1 ="))
4      b = int(input("masukkan bilangan ke 2 ="))
5      c = int(input("masukkan bilangan ke 3 ="))
6
7      print('\n')
8
9      if(a > b) and (a > c):
10         print("bilangan ke 1 paling besar")
11     elif(b > a) and (b > c):
12         print("bilangan ke 2 paling besar")
13     elif(c > a) and (c > b):
14         print("bilangan ke 3 paling besar")
15     else:
16         print("ada dua atau tiga masukan memiliki nilai sama")

```

- Hasil eksekusi program, dimana program dapat menentukan bilangan terbesar.



```

Program Menentukan Bilangan yang Terbesar
*****

masukkan bilangan ke 1 =9
masukkan bilangan ke 2 =10
masukkan bilangan ke 3 =3

bilangan ke 2 paling besar

```

3. Latihan Praktikum 3.1.3

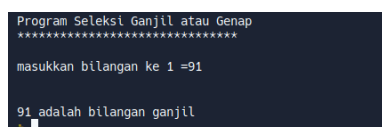
Buatlah Program seperti di bawah ini, dengan nama file latihan3_1_3.py

```

1      print("Program Seleksi Ganjil atau Genap")
2      print('*****\n')
3      bilangan = int(input("masukkan bilangan ke 1 ="))
4      print('\n')
5
6      if bilangan % 2 == 1:
7         print(bilangan, "adalah bilangan ganjil")
8     else:
9         print(bilangan, "adalah bilangan genap")

```

- Hasil eksekusi program seperti gambar di bawah ini.



```

Program Seleksi Ganjil atau Genap
*****

masukkan bilangan ke 1 =91

91 adalah bilangan ganjil
>

```

4. Latihan Praktikum 3.1.4

Buatlah Program seperti di bawah ini, dengan nama file.

latihan3_1_4.py

- Hasil eksekusi program seperti gambar di bawah ini.

```

Program Menghitung Luas
*****

Pilih Menu
=====> 1. Luas Lingkaran
          2. Luas Persergi
          3. Luas Segitiga

Masukkan Pilihan =3

Program Segitiga
=====

Masukkan Alas =4
Masukkan Tinggi =7
Luas adalah : 14.0
  
```

```

1     print("Program Menghitung Luas")
2     print('*****\n')
3     print('\n')
4     print("Pilih Menu")
5     print('=====> 1. Luas Lingkaran')
6     print('          2. Luas Persergi')
7     print('          3. Luas Segitiga')
8     print('\n')
9     pilihan = int(input("Masukkan Pilihan ="))
10    print('\n')
11
12    if pilihan == 1:
13        print("Program Lingkaran")
14        print('*****\n')
15        print('\n')
16        jari = int(input("Masukkan jari-jari ="))
17        luas = 3.14 * jari * jari
18        print("Luas adalah :", luas)
19    elif pilihan == 2:
20        print("Program Persegi Panjang")
21        print('=====\\n')
22        print('\n')
23        panjang = int(input("Masukkan Panjang ="))
24        lebar = int(input("Masukkan Lebar ="))
25        luas = panjang * lebar
  
```

```

26     print("Luas adalah :", luas)
27 elif pilihan == 3:
28     print("Program Segitiga")
29     print('=====\\n')
30     print('\\n')
31     a = int(input("Masukkan Alas ="))
32     t = int(input("Masukkan Tinggi ="))
33     luas = 0.5 * a * t
34     print("Luas adalah :", luas)
35 else:
36     print("Pilihan Menu Tidak Ada")

```

5. Latihan Praktikum 3.1.5

Buatlah program untuk menentukan nilai terkecil bilangan dari 3 bilangan yang di input, dan simpan (save) file dengan nama latihan3_1_5.py

6. Latihan Praktikum 3.1.6

Buatlah program untuk menghitung gaji karyawan dengan aturan :

- Jika MASA KERJA > 5 tahun memperoleh BONUS 15% dari GAJI POKOK
- Jika PEGAWAI TETAP memperoleh UANG TRANSPORT sebesar Rp. 50.000,-
- Jika SUDAH BERKELUARGA memperoleh TUNJANGAN sebesar 10% dari GAJI POKOK
- Sedangkan GAJI BERSIH dihitung dengan rumus : GAJI POKOK + BONUS + TUNJANGAN + UANG TRANSPORT.

Dan simpan (save) file program dengan nama latihan3_1_6.py

7. Latihan Praktikum 3.1.7

Buatlah program kalkulator sederhana dengan memberikan pilihan bagi pengguna (menggunakan IF THEN ELSE) . Operasi-operasinya meliputi : penjumlahan, pengurangan, perkalian, pembagian. Dan simpan (save) file program dengan nama latihan3_1_7.py

BAB 4 PRAKTIKUM MATCH AND CASE (PERCABANGAN BERSYARAT)

Tujuan Praktek

- ❑ Untuk mengetahui statemen *MATCH* dan *CASE* yang berisi ungkapan (pemilih) dan sederetan statemen yang masing-masing diawali dengan satu atau lebih konstanta (disebut konstanta *CASE*) atau dengan wildcard *CASE* `_`.

4.1 Dasar Teori

Pernyataan *CASE* merupakan pernyataan bersyarat yang memiliki bentuk :

```
CASE <syarat-pemilih>
<daftar-label 1> : statemen1
<daftar-label 1> : statemen2
[CASE _ statemen-statemen lain]
```

Syarat pemilih dapat berupa sembarang ungkapan, variabel atau fungsi yang bertipe scalar, kecuali *REAL*. Daftar label 1 boleh berupa satu atau lebih dari satu nilai yang merupakan kemungkinan dari syarat pemilih. Semua pernyataan yang berhubungan dengan daftar label ataupun *CASE* `_` boleh berupa sembarang pernyataan. Pernyataan I yang bersangkutan dengan daftar label 1 di eksekusi bila nilai dari syarat pemilih tercantum dalam daftar label tersebut. Perlu diketahui, *CASE* `_` dalam pernyataan *CASE* disertakan, pernyataan yang mengikuti *CASE* `_` akan dieksekusi hanya dengan nilai dari syarat pemilih tidak ada dalam semua daftar label.

BAB 4 Latihan

1. Latihan Praktikum 4.1.1

Buatlah Program seperti di bawah ini, dengan nama file `latihan4_1_1.py`

```
1      print("Program memeriksa hari dalam bulan Maret 2012")
2      print('*****\n')
3      tgl = int(input("Masukkan tanggal ="))
4
5      match tgl:
6          case 5|12|19|26: print("Hari Senin")
7          case 6|13|20|27: print("Hari Selasa")
8          case 7|14|21|28: print("Hari Rabu")
9          case 1|8|15|22|29: print("Hari Kamis")
10         case 2|9|16|23|20: print("Hari Jumat")
11         case 3|10|17|24|31: print("Hari Sabtu")
12         case 4|11|18|25: print("Hari Minggu")
13         case _:
14             print("Maaf tidak ada tanggal tersebut")
```

- Hasil eksekusi program seperti Gambar berikut:


```

Program memeriksa hari dalam bulan Maret 2012
*****

Masukkan tanggal =15
Hari Kamis

```

2. Latihan Praktikum 4.1.2

Buatlah Program seperti di bawah ini, dengan nama file latihan4_1_2.py

```

1  print("Program Jual Disc")
2  print('*****\n')
3  beli = int(input("Total pembelian ="))
4
5  match beli:
6      case beli if 0 <= beli <= 1000:disc = 100
7      case beli if 1001 <= beli <= 10000:disc = 500
8      case beli if 10001 <= beli <= 30000:disc = 2000
9
10     case _:
11         print("Persediaan tidak mencukupi")
12
13     print("\n")
14     print("Jumlah Pembelian = Rp. ", beli)
15     print("discount = Rp. ", disc)
16     print("")
17     print("Jumlah yang dibayar : Rp. ", beli - disc)

```

- Hasil eksekusi program seperti Gambar berikut:

```

Masukkan Total beli =7000

Jumlah Pembelian = Rp. 7000
discount = Rp. 500

Jumlah yang dibayar : Rp. 6500

```

3. Latihan Praktikum 4.1.3

Buatlah Program seperti di bawah ini, dengan nama file latihan4_1_3.py

```

1  print("Program Menghitung Luas")
2  print('*****\n')
3  print('\n')
4  print("Pilih Menu")
5  print('=====> 1. Luas Lingkaran')
6  print('          2. Luas Persergi')
7  print('          3. Luas Segitiga')
8  print('\n')
9  pilihan = int(input("Masukkan Pilihan ="))
10 print('\n')
11
12 match pilihan:

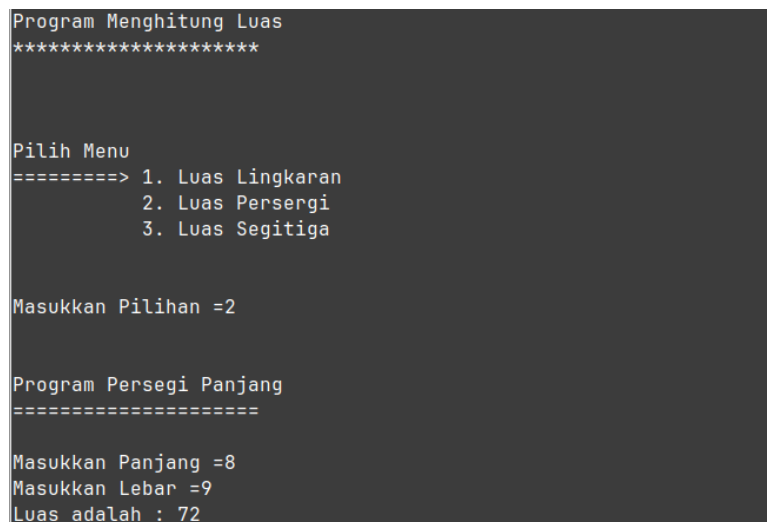
```

```

13     case 1:
14         print("Program Lingkaran")
15         print('*****\n')
16
17         jari = int(input("Masukkan jari-jari ="))
18         luas = 3.14 * jari * jari
19         print("Luas adalah :", luas)
20     case 2:
21         print("Program Persegi Panjang")
22         print('===== \n')
23
24         panjang = int(input("Masukkan Panjang ="))
25         lebar = int(input("Masukkan Lebar ="))
26         luas = panjang * lebar
27         print("Luas adalah :", luas)
28     case 3:
29         print("Program Segitiga")
30         print('===== \n')
31
32         a = int(input("Masukkan Alas ="))
33         t = int(input("Masukkan Tinggi ="))
34         luas = 0.5 * a * t
35         print("Luas adalah :", luas)
36     case _:
37         print("Pilihan Menu Tidak Ada")

```

- Hasil eksekusi program seperti Gambar berikut:



```

Program Menghitung Luas
*****

Pilih Menu
=====> 1. Luas Lingkaran
        2. Luas Persegi
        3. Luas Segitiga

Masukkan Pilihan =2

Program Persegi Panjang
=====

Masukkan Panjang =8
Masukkan Lebar =9
Luas adalah : 72

```

4. Latihan Praktikum 4.1.4

Buatlah program kalkulator sederhana dengan memberikan pilihan bagi pengguna (menggunakan MATCH CASE) . Operasi-operasinya meliputi : penjumlahan, pengurangan, perkalian, pembagian. Dan simpan (save) file program dengan nama latihan4_1_4.py

5. Latihan Praktikum 4.1.5

Diketahui toko buah “SEGAR MANIS” yang menjual 4 jenis buah-buahan dalam jumlah kiloan. Untuk

setiap buah -buahan yang dijual jika melebihi 5 Kg mendapat potongan :

Kiwi : 5%
Mangga : 2.5%
Alpukat : 7%
Apel : 10%

dari total penjualan:

- Untuk buah-buahan Kiwi dan Mangga jika lebih dari 20 Kg mendapat korting lagi hingga 7% dari total penjualan
- Untuk buah-buahan Alpukat jika lebih dari 10 Kg dapat hadiah payung dan untuk buah-buahan Apel jika lebih dari 15 Kg dapat hadiah tas

Dari persoalan diatas buatlah program pemecahannya dan simpan (save) dengan nama file latihan 4_1_5.py

BAB 5 FOR LOOP (PERULANGAN ATAU LOOPING)

Tujuan Praktek

- Untuk mengetahui proses pengulangan yang berfungsi untuk mengeksekusi suatu pernyataan lain secara berulang-ulang, dengan jumlah pengulangan tergantung pada persyaratan yang diberikan.

5.1 Dasar Teori

Pernyataan FOR memungkinkan pengulangan eksekusi terhadap suatu pernyataan, baik berupa pernyataan yang sederhana maupun pernyataan majemuk. Pengendalian pengulangan eksekusi dilakukan oleh variabel kendali. Jumlah pengulangan eksekusi ditentukan oleh nilai awal dan nilai akhir dari variabel kendali.

Berikut syntax dasar untuk struktur for loop dalam bahasa pemrograman Python:

```
# for item in sequence:
# execute expression
```

dimana :

for	awal mulai perulangan for
item	sebuah item individual selama setiap iterasi. item juga merupakan nama variabel sementara yang dapat diubah penamaannya.
in	yang memisahkan tiap-tiap item dengan yang lainnya.
:	instruksi untuk menjalankan body dari code dibawahnya.

🏰 BAB 5 Latihan 🏰

1. Latihan Praktikum 5.1.1

Buatlah Program seperti di bawah ini, dengan nama file latihan5_1_1.py

```
1   jumlah = 0
2   for i in range(26):
3       jumlah = jumlah + i
4
5   rerata = jumlah / 25
6   print("Jumlah :", jumlah)
7   print("Rata-rata :", rerata)
```

- Hasil eksekusi program seperti Gambar berikut:

```
Jumlah : 325
Rata-rata : 13.0
>
```

2. Latihan Praktikum 5.1.2

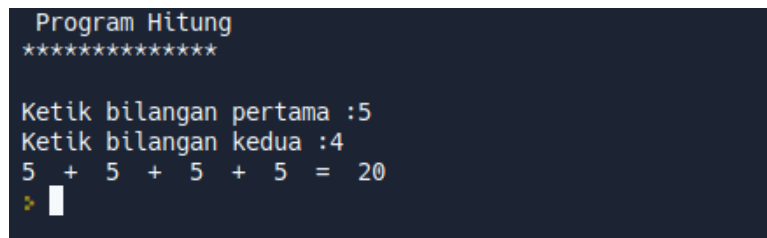
Buatlah Program seperti di bawah ini, dengan nama file latihan5_1_2.py

```

1      print(' Program Hitung ')
2      print('*****\n')
3      bil1 = int(input("Ketik bilangan pertama :"))
4      bil2 = int(input("Ketik bilangan kedua :"))
5
6      hasil = 0
7
8      for i in range(bil1):
9          hasil = hasil + bil2
10
11     bil2 = bil2 - 1
12     for i in range(bil2):
13         print(bil1, " + ", end=' ')
14
15     print(bil1, " = ", hasil)

```

- Hasil eksekusi program seperti Gambar berikut:



```

Program Hitung
*****

Ketik bilangan pertama :5
Ketik bilangan kedua :4
5 + 5 + 5 + 5 = 20
> 

```

3. Latihan Praktikum 5.1.3

Buatlah Program seperti di bawah ini, dengan nama file latihan5_1_3.py

```

1      print('Program Nilai Kurs')
2      print('*****\n')
3
4      a = int(input("Masukkan nilai kurs 1$ = Rp. "))
5
6      b=0
7      n=6
8
9      for i in range(1, n):
10         b=b+a
11         print("Kurs ", i, "$ = Rp. ", b)
12
13     c=b/(n-1)
14     print('Rata-rata 1$ = Rp. ',c)
15     if c > 9000:
16         print("Jual Dollar")
17     elif c < 65000:
18         print("Dollar simpan")
19     else:
20         print("?")

```


- Hasil eksekusi program seperti Gambar berikut:

```

Program Nilai Kurs
*****

Masukkan nilai kurs 1$ = Rp. 9000
Kurs 1 $ = Rp. 9000
Kurs 2 $ = Rp. 18000
Kurs 3 $ = Rp. 27000
Kurs 4 $ = Rp. 36000
Kurs 5 $ = Rp. 45000
Rata-rata 1$ = Rp. 9000.0
Dollar simpan
>

```

4. Latihan Praktikum 5.1.4

Buatlah Program seperti di bawah ini, dengan nama file latihan5_1_4.py

```

1     for i in range(90, 64, -1):
2         print(chr(i), ' ', end="")

```

- Hasil eksekusi program seperti Gambar berikut:

```

Z Y X W V U T S R Q P O N M L K J I H G F E D C B A

```

5. Latihan Praktikum 5.1.5

Buatlah Program seperti di bawah ini, dengan nama file latihan5_1_5.py

```

1     print('Program Membalik Kata dari Belakang')
2     print('*****\n')
3
4     kata = input("Tulis kata yang akan dibalik = ")
5
6     balik = ''
7
8     for i in kata:
9         balik = i + balik
10    print("Hasilnya :", balik)

```

- Hasil eksekusi program seperti Gambar berikut:

```

Program Membalik Kata dari Belakang
*****

Tulis kata yang akan dibalik = yusri
Hasilnya : irsuy
>

```

6. Latihan Praktikum 5.1.6

Buatlah Program seperti di bawah ini, dengan nama file latihan5_1_6.py

```

1     print('Program Menampilkan Bintang')
2     print('=====')

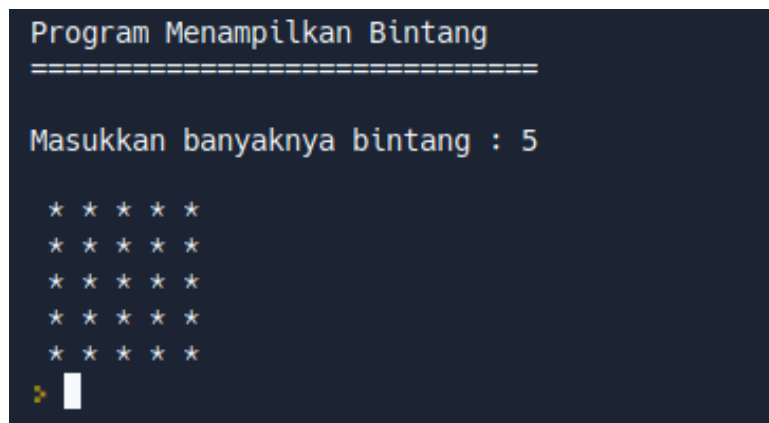
```

```

3     print()
4
5     bintang = int(input('Masukkan banyaknya bintang : '))
6     print()
7
8     for i in range(bintang):
9         for j in range(bintang):
10            print(' *',end='')
11            print()

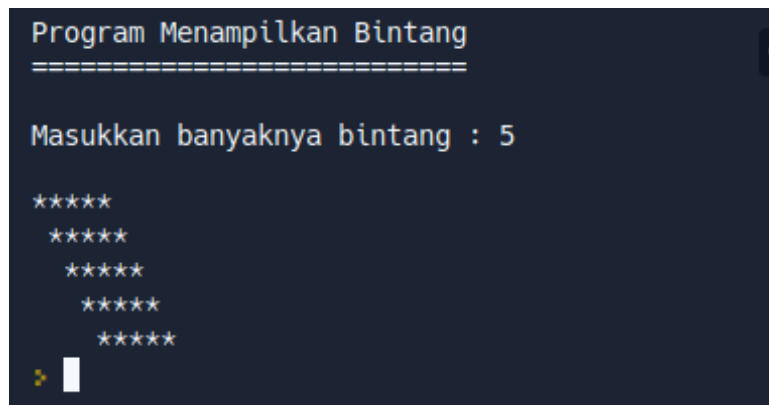
```

- Hasil eksekusi program seperti Gambar berikut:



7. Latihan Praktikum 5.1.7

Lengkapi (modifikasi) program latihan5_1_6.py menjadi tampilan seperti gambar di bawah ini dan simpan (save) file dengan nama latihan5_1_7.py



8. Latihan Praktikum 5.1.8

Buat program untuk membuat daftar Harga Bahan Bakar(B), Solar(S), dan Pertamina(P). Dimana yang di input adalah besaran Liter(L) dan file disimpan (save) dengan nama latihan5_1_8.py

Keterangan:

- Harga Bensin : Rp. 4500/L
- Harga Solar : Rp. 5000/L
- Harga Pertamina : Rp. 7500/L

9. Latihan Praktikum 5.1.9

Buat program untuk menampilkan angka ganjil dari angka 1 – 100, dan file disimpan (save) dengan nama

latihan5_1_9.py

BAB 6 PRAKTIKUM WHILE IF CONDITION (PERULANGAN / LOOPING)

Tujuan Praktek

- ❑ Untuk memahami suatu proses perulangan pada while dan repeat dalam suatu program terstruktur.

6.1 Dasar Teori

- Pernyataan WHILE

Instruksi WHILE berarti selama atau sementara. Jadi selama sebuah kondisi terpenuhi maka proses perulangan tersebut akan berjalan terus. Dengan kata lain, satu atau lebih statemen akan dikerjakan terus selama kondisi bernilai benar.

Bentuk umum:

```
# while condition:
    # body dari while loop berisikan code yang melakukan suatu
    instruksi
```

Pada pemrograman pascal biasanya instruksi WHILE berpasangan dengan pernyataan REPEAT UNTIL, namun didalam python pernyataan tersebut ekuivalen dengan bentuk berikut:

```
REPEAT
.....
UNTIL cond
```

ekivalen dengan:

```
while True:
.....
    if cond:
        break
```

🎨 BAB 6 Latihan 🎨

1. Latihan Praktikum 6.1.1

Buatlah Program seperti di bawah ini, dengan nama file latihan6_1_1.py

```
1   angka = 0
2   while angka < 5:
3       print(angka)
4       angka = angka+1
```

- Hasil eksekusi program seperti Gambar berikut:

2. Latihan Praktikum 6.1.2

Buatlah Program seperti di bawah ini, dengan nama file latihan6_1_2.py

```
0
1
2
3
4
>
```

```
1 angka = 0
2
3 while True:
4     angka = angka + 1
5     print(angka)
6
7     if angka == 5:
8         break
```

- Hasil eksekusi program seperti Gambar berikut:

```
1
2
3
4
5
>
```

3. Latihan Praktikum 6.1.3

Buatlah Program seperti di bawah ini, dengan nama file latihan6_1_3.py

```
1     hasil = 0
2
3     while True:
4         print()
5         print('Program Perhitungan Menggunakan Perulangan')
6         print('+++++')
7         print()
8         bil1 = int(input("Ketik Bilangan Pertama = "))
9         bil2 = int(input("Ketik Bilangan Kedua = "))
10
11         if bil1 >= bil2:
12             bil1 = bil1 - bil2
13             hasil = hasil + 1
14             print("Hasil perhitungan =", hasil)
15             print()
16
17         ulang = input("Mau ulang Program Tekan [Y] / Keluar [T] =")
18
19         if ulang == 'T':
20             break
```

- Hasil eksekusi program seperti Gambar berikut:

4. Latihan Praktikum 6.1.4

Buatlah Program seperti di bawah ini, dengan nama file latihan6_1_4.py


```

Program Perhitungan Menggunakan Perulangan
+++++

Ketik Bilangan Pertama = 10
Ketik Bilangan Kedua = 2
Hasil perhitungan = 5

Mau ulang Program Tekan [Y] / Keluar [T] =

```

```

1      c = 1.609344
2      print('Program Konversi Kilometer ==> Mil')
3      print()
4
5      a = 0
6      while a < 20:
7          a = a + 1
8          b = c * a
9      print(a, "Km = ", round(b, 4), "Mil")

```

- Hasil eksekusi program seperti Gambar berikut:

```

Program Konversi Kilometer ==> Mil

1 Km = 1.6093 Mil
2 Km = 3.2187 Mil
3 Km = 4.828 Mil
4 Km = 6.4374 Mil
5 Km = 8.0467 Mil
6 Km = 9.6561 Mil
7 Km = 11.2654 Mil
8 Km = 12.8748 Mil
9 Km = 14.4841 Mil
10 Km = 16.0934 Mil
11 Km = 17.7028 Mil
12 Km = 19.3121 Mil
13 Km = 20.9215 Mil
14 Km = 22.5308 Mil
15 Km = 24.1402 Mil
16 Km = 25.7495 Mil
17 Km = 27.3588 Mil
18 Km = 28.9682 Mil
19 Km = 30.5775 Mil
20 Km = 32.1869 Mil

```

5. Latihan Praktikum 6.1.5

Buatlah Program seperti di bawah ini, dengan nama file latihan6_1_5.py

```

1      import random
2
3      print("PROGRAM ANGKA ACAK")

```

```
4      print("+++++")
5      print()
6
7      b = "N"
8
9      while (b == "N"):
10         Angka_Acak = (random.randrange(1000))
11         print ("Angka sekarang :", Angka_Acak)
12         print()
13         print("Tekan Sembarang Tombol untuk mulai mengacak")
14         print("Tekan Y untuk berhenti")
15         print()
16
17         x=input()
18
19         if x == "Y":
20             break
21
22     print("Pengacakan Berhenti")
```

- Hasil eksekusi program seperti Gambar berikut:

```
PROGRAM ANGKA ACAK
+++++

Angka sekarang : 296

Tekan Sembarang Tombol untuk mulai mengacak
Tekan Y untuk berhenti

Angka sekarang : 431

Tekan Sembarang Tombol untuk mulai mengacak
Tekan Y untuk berhenti

Angka sekarang : 880

Tekan Sembarang Tombol untuk mulai mengacak
Tekan Y untuk berhenti

Y
Pengacakan Berhenti
```

6. Latihan Praktikum 6.1.6

Lengkapi (modifikasi) program latihan6_1_3.py menjadi tampilan seperti gambar di bawah ini dan simpan (save) file dengan nama latihan6_1_6.py.

```
Program Perhitungan Menggunakan PERulangan
+++++

Ketik Bilangan Pertama = 10
Ketik Bilangan Kedua   = 2
10 - 2 - 2 - 2 - 2 - 2 = 5 kali pengurangannya

Mau ulang Program Tekan [Y] / Keluar [T] = _
```

7. Latihan Praktikum 6.1.7

Buat Program seperti tampilan eksekusi program di bawah ini, dan simpan (save) file dengan nama latihan6_1_7.py

```
MENGHITUNG JUMLAH & RATA-RATA SEJUMLAH DATA NILAI
CACAH DATA TIDAK DIKETAHUI DENGAN PASTI
PEMBACAAN DATA DIHENTIKAN DENGAN DATA SENTINEL
DATA SENTINEL BERUPA SEMBARANG BILANGAN NEGATIF

ISIKAN BILANGAN NEGATIF UNTUK SELESAI
MASUKKAN DATA NILAINYA : 8
MASUKKAN DATA NILAINYA = 9
MASUKKAN DATA NILAINYA = 6
MASUKKAN DATA NILAINYA = 1
MASUKKAN DATA NILAINYA = -9

JUMLAH DATA      = 24
NILAI RATA-RATA   = 6.00
```

BAB 7 PRAKTIKUM PROCEDURE DAN FUNGSI

Tujuan Praktek

- ❑ Mahasiswa mengenal, memahami, dan mampu membuat sub program.

7.1 Dasar Teori

Python merupakan bahasa pemrograman yang cocok digunakan dalam jenis kepentingan website, pengembangan aplikasi dan banyak lainnya kegunaan yang dimiliki. Python juga memiliki dukungan untuk modul dan package, dimana python sendiri memperbolehkan sistem modul dan kode untuk digunakan kembali. Python sangat cocok untuk dipelajari bagi pemula untuk belajar menjadi seorang programmer. Adakalanya ketika kita menyusun kode dalam bahasa pemrograman Python terkadang kita membuat kode secara berulang. Misalnya untuk membaca tabel, menjumlahkan data, penyusunan laporan. Kode inilah yang nantinya akan digunakan untuk menampilkan output hasil running data kita. Didalam bahasa pemrograman Python terdapat dua jenis yaitu Built-In Function dan User Defined Function. Built-In Function adalah sebutan untuk fungsi yang sudah ada secara bawaan dari dalam bahasa pemrograman. Sedangkan User Defined Function adalah fungsi yang kita (sebagai programmer) membuatnya sendiri. Salah satu function yang sering kita gunakan adalah `print()` untuk menampilkan output. Fungsi `print` inilah yang disebut dengan built-in function karena bahasa Python telah menyediakan ragam fungsi bawaan. Fungsi ini belum termasuk library dan package.

Dalam bahasa pemrograman **Python** disediakan fungsi dan prosedur. Keduanya merupakan fasilitas yang telah disediakan oleh bahasa pemrograman. Yang membedakan antara suatu prosedur dengan suatu fungsi dalam sebuah program adalah masukan dan keluarannya. Pada prosedur, tidak ada keluaran sedangkan pada fungsi ada keluaran yang nantinya dapat selanjutnya diolah lagi. Prosedur dan fungsi mempunyai deklarasi sintak yang sama pada bahasa pemrograman python yaitu dengan menggunakan keyword `def`. Yang perlu diingat adalah nama variabel yang digunakan untuk masukan dari fungsi jangan digunakan untuk proses yang dilakukan dalam fungsi karena jika nilai variabel tersebut akan diisi dengan nilai terakhir dari proses yang dilakukan. Lebih baik mendeklarasikan variabel baru yang khusus digunakan untuk proses dalam fungsi tersebut. Banyak contoh fungsi dan prosedur yang disediakan oleh bahasa pemrograman Python sebagai langkah kamu belajar bahasa pemrograman python.

Sebelum kita masuk dalam pengenalan lebih lanjut mengenai procedure Python, kita akan kenalkan terlebih dahulu mengenai procedure dasarnya. Procedure pada bahasa pemrograman Python dibuat dengan kata kunci atau keyword `def`. Setelah itu, diikuti dengan nama Procedurenya.

Kurang lebih gambarannya seperti ini. Sama seperti blok kode yang lain, kita juga harus memberikan indentasi (tab atau spasi 2x) untuk menuliskan isi fungsi. Nah, sekarang kita langsung mulai ke aplikasi contohnya dalam bahasa pemrograman Python. Pasti akan lebih mudah dan aplikatif tentunya menggunakan bahasa pemrograman Python. Mari kita mulai operasinya!

🌊 BAB 7 Latihan 🌊

1. Latihan Praktikum 7.1.1

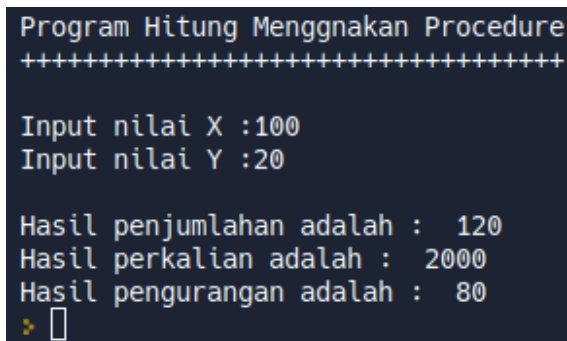
Buatlah Program procedure seperti di bawah ini, dengan nama file `latihan7_1_1.py`

```

1  def Operasi_hitung(A, B, C):
2      C = A + B
3      print()
4      print("Hasil penjumlahan adalah : ", C)
5      C = A * B
6      print("Hasil perkalian adalah : ", C)
7      C = A - B
8      print("Hasil pengurangan adalah : ", C)
9
10     Z = 0
11     print("Program Hitung Menggnakan Procedure")
12     print("+++++")
13     print()
14     X = int(input("Input nilai X :"))
15     Y = int(input("Input nilai Y :"))
16
17     Operasi_hitung(X,Y,Z)

```

- Hasil eksekusi program seperti Gambar berikut:



```

Program Hitung Menggnakan Procedure
+++++

Input nilai X :100
Input nilai Y :20

Hasil penjumlahan adalah : 120
Hasil perkalian adalah : 2000
Hasil pengurangan adalah : 80
> 

```

2. Latihan Praktikum 7.1.2

Buatlah Program procedure seperti di bawah ini, dengan nama file latihan7_1_2.py

```

1  def Hitung_Panjang(nama):
2      n = len(nama)
3      print(n, "Karakter")
4
5      print()
6      print("Program Hitung Huruf Nama")
7      print("+++++")
8      print()
9      nama = input("Siapakah Nama Lengkap Anda :")
10     print("panjang nama anda adalah : " ,end='')
11     Hitung_Panjang(nama)
12     print("Catatan :")
13     print("Spasi dalam nama dianggap Satu Karakter")

```

- Hasil eksekusi program seperti Gambar berikut:

```

Program Hitung Huruf Nama
+++++

Siapakah Nama Lengkap Anda :yusri ikhwani
panjang nama anda adalah :13 Karakter
Catatan :
Spasi dalam nama dianggap Satu Karakter
> 

```

3. Latihan Praktikum 7.1.3

Buatlah Program procedure seperti di bawah ini, dengan nama file latihan7_1_3.py

```

1  def A_Pangkat_N_Positif(A, N):
2      hasil = A**N
3      return hasil
4
5  def Pangkat_Positif(A, N):
6      hasil = A_Pangkat_N_Positif(A, N)
7      return hasil
8
9  def Pangkat_Negatif(A,N):
10     return A_Pangkat_N_Positif(A, N)
11
12 def masukan_Data():
13     A = float(input("Masukan Nilai A : "))
14     N = float(input("Masukan Nilai N : "))
15
16     if N < 0:
17         print("A pangkat N (negatif) = ", Pangkat_Negatif(A,N))
18     else:
19         print("A pangkat N = ", Pangkat_Positif(A,N))
20
21 print('Menghitung A pangkat N dengan A riil & N integer')
22 print('A dan N boleh bilangan negatif')
23 print(' ++++++ ')
24 print()
25 masukan_Data()

```

- Hasil eksekusi program seperti Gambar berikut:

```

Menghitung A pangkat N dengan A riil & N integer
A dan N boleh bilangan negatif
+++++

Masukan Nilai A : 8
Masukan Nilai N : 3
A pangkat N = 512.0
> 

```

4. Latihan Praktikum 7.1.4

Buatlah Program function seperti di bawah ini, dengan nama file latihan7_1_4.py

```

1      def tambah(a,b):
2          tambah = a+b
3          return tambah
4
5      def kali(a,b):
6          kali = a*b
7          return kali
8
9      while True:
10         print()
11         print('Program Perhitungan Menggunakan Function')
12         print('+++++')
13         print()
14         bil1 = int(input("Ketik Bilangan Pertama = "))
15         bil2 = int(input("Ketik Bilangan Kedua = "))
16
17         print(bil1, "+", bil2, " = ", tambah(bil1,bil2))
18         print(bil1, "x", bil2, " = ", "%.2f" % kali(bil1,bil2))
19         print()
20         ulang = input("Mau ulang Program Tekan [Y] / Keluar [T] =")
21
22         if ulang == 'T':
23             break

```

- Hasil eksekusi program seperti Gambar berikut:

```

Program Perhitungan Menggunakan Function
+++++

Ketik Bilangan Pertama = 8
Ketik Bilangan Kedua = 4
8 + 4 = 12
8 x 4 = 32.00

Mau ulang Program Tekan [Y] / Keluar [T] =Y

```

BAB 8 PRAKTIKUM ARRAY

Tujuan Praktek

- ❑ Mengetahui beberapa tipe bentukan dalam bahasa Python, yaitu Array & Record.
- ❑ Mampu membuat dan menggunakan tipe data bentukan tersebut dalam masalah yang sesuai dan mengimplementasikannya dalam bahasa Python

8.1 Dasar Teori

Dalam bahasa Python terdapat bermacam -macam tipe data. Tipe data – tipe data tersebut dikelompokkan ke dalam dua jenis, yaitu : tipe data dasar dan tipe data bentukan. Tipe data dasar meliputi integer, float, complex, Boolean, frozenset dan string. Tipe data bentukan antara lain : tipe enumerated, subrange, rccord dan array.

- Enumerated

Contoh:

```
Warna = ('merah', 'hijau', 'biru')
Bentuk = ('segitiga', 'bujursangkar', 'lingkaran')
x = enumerate(Warna)
y = enumerate(Bentuk)
```

Warna dan bentuk adalah nama tipe enumerated yang masing-masing mempunyai elemen-elemen sesuai dengan yang dituliskan.

- Tipe Subrange

Contoh:

```
from subrange import subrange

r = subrange(1, 5)           # subrange of integers 1..5

print r                     # 1..5
print r.min, r.max          # 1 5
print repr(r)               # subrange(1, 5)
print "{:04b}".format(r)    # 0001..0101
print 4 in r                 # True
print [i for i in r]        # [1, 2, 3, 4, 5]
print r < 6                  # True
```

r nama tipe subrange yang masing -masing mempunyai rentang nilai atau elemen. Elemen digit adalah semua elemen mulai dari 1 sampai dengan 5.

- Tipe Record

Contoh :

r adalah sebuah record yang elemennya terdiri dari attribute1, attribute2 dan attribute3 yang masing-masing menunjukkan dua string "hello" dan "there" serta satu integer 3.14. Record hanya dapat diakses melalui elemen-elemennya. Pengaksesan menggunakan bantuan . (titik). Sebelum record digunakan harus dibuat instansi objek recordnya.


```

class Record(object):
    __slots__ = "attribute1", "attribute2", "attribute3",

    def items(self):
        "dict style items"
        return [
            (field_name, getattr(self, field_name))
            for field_name in self.__slots__]

    def __iter__(self):
        "iterate over fields tuple/list style"
        for field_name in self.__slots__:
            yield getattr(self, field_name)

    def __getitem__(self, index):
        "tuple/list style getitem"
        return getattr(self, self.__slots__[index])

>>> r = Record()
>>> r.attribute1= "hello"
>>> r.attribute2= "there"
>>> r.attribute3= 3.14

>>> print r.items()
[('attribute1', 'hello'), ('attribute2', 'there'), ('attribute3', 3.1400000000000000)]
>>> print tuple(r)
('hello', 'there', 3.1400000000000001)

```

- Tipe Array

Contoh :

```

class StaticArray:

    def __init__(self, n):
        self.data = [None] * n

    def get_at(self, i):
        if not (0 <= i < len(self.data)): raise IndexError
        return self.data[i]

    def set_at(self, i, x):
        if not (0 <= i < len(self.data)): raise IndexError
        self.data[i] = x

array1 = StaticArray(20)
array2 = StaticArray(20)
array3 = StaticArray(20)
array4 = StaticArray(20)

```

array1 dan array2 bertipe string dan array 3 bertipe float dan integer. Array dapat diakses melalui nilai indeks, yang dituliskan dalam kurung siku, dan nilai tersebut masih berada dalam domain harganya yang dituliskan dengan tipe data yang telah ditentukan.

BAB 8 Latihan

1. Latihan Praktikum 8.1.1

Buatlah Program seperti di bawah ini, dengan nama file latihan8_1_1.py

```

1      class StaticArray:
2
3          def __init__(self, n):
4              self.data = [None] * n
5
6          def get_at(self, i):
7              if not (0 <= i < len(self.data)): raise IndexError
8              return self.data[i]
9
10         def set_at(self, i, x):
11             if not (0 <= i < len(self.data)): raise IndexError
12             self.data[i] = x
13
14
15     array1 = StaticArray(20)
16     array2 = StaticArray(20)
17     array3 = StaticArray(20)
18     array4 = StaticArray(20)
19     rerata = StaticArray(20)
20
21     print(' Program Nilai Mahasiswa ')
22     print(' ##### ')
23     print()
24     n = int(input('Masukkan Jumlah Data Mahasiswa : '))
25     print()
26     for k in range(n):
27         print('ENTRY NILAI UJIAN MAHASISWA ')
28         print('-----')
29         nama = input("Masukkan Nama :")
30         array1.set_at(k, (nama))
31         tugas = float(input("Masukkan Nilai Tugas :"))
32         array2.set_at(k, (tugas))
33         uts = float(input("Masukkan Nilai UTS :"))
34         array3.set_at(k, (uts))
35         uas = float(input("Masukkan Nilai UAS :"))
36         array4.set_at(k, (uas))
37         print()
38         rerata.set_at(k, ((0.30 * array2.get_at(k))) + ((0.30 * array3.get_at(k))) +
39                         ((0.40 * array4.get_at(k))))
40
41     print('LAPORAN UJIAN MAHASISWA')
42     print('-----')

```

```

43     print('No', '\t', 'Nama', '\t', 'Tugas', '\t', 'UTS', '\t', 'UAS', '\t', 'Nilai
        Akhir')
44     for i in range(n):
45         print(i, '\t', array1.get_at(i), '\t', array2.get_at(i), '\t', array3.get_at(i)
        ,
46             '\t', array4.get_at(i), '\t', rerata.get_at(i), end='\n')

```

- Hasil eksekusi program seperti Gambar berikut:

```

Program Nilai Mahasiswa
#####

Masukkan Jumlah Data Mahasiswa : 2

ENTRY NILAI UJIAN MAHASISWA
-----
Masukkan Nama :toni
Masukkan Nilai Tugas :80
Masukkan Nilai UTS :90
Masukkan Nilai UAS :60

ENTRY NILAI UJIAN MAHASISWA
-----
Masukkan Nama :roni
Masukkan Nilai Tugas :80
Masukkan Nilai UTS :50
Masukkan Nilai UAS :90

LAPORAN UJIAN MAHASISWA
-----

```

No	Nama	Tugas	UTS	UAS	Nilai Akhir
0	toni	80.0	90.0	60.0	75.0
1	roni	80.0	50.0	90.0	75.0

Daftar Pustaka

- [1] Qiang Li, Liwen Chen, and Yong Zeng. “The Mechanism and Effectiveness of Credit Scoring of P2P Lending Platform: Evidence from Renrendai.com”. In: *China Finance Review International* 8.3 (2018), pp. 256–274.
- [2] Vincenzo Quadrini. “Financial Frictions in Macroeconomic Fluctuations”. In: *FRB Richmond Economic Quarterly* 97.3 (2011), pp. 209–254.