

LAPORAN PRAKTIKUM
MODUL I
VARIABEL, TIPE DATA, DAN OPERATOR



Disusun Oleh :

Fauziyah Ulur Rosyad Prawidani

NIM : 19104033

Dosen :

Ariq Cahya Wardhana, S.Kom., M.Kom

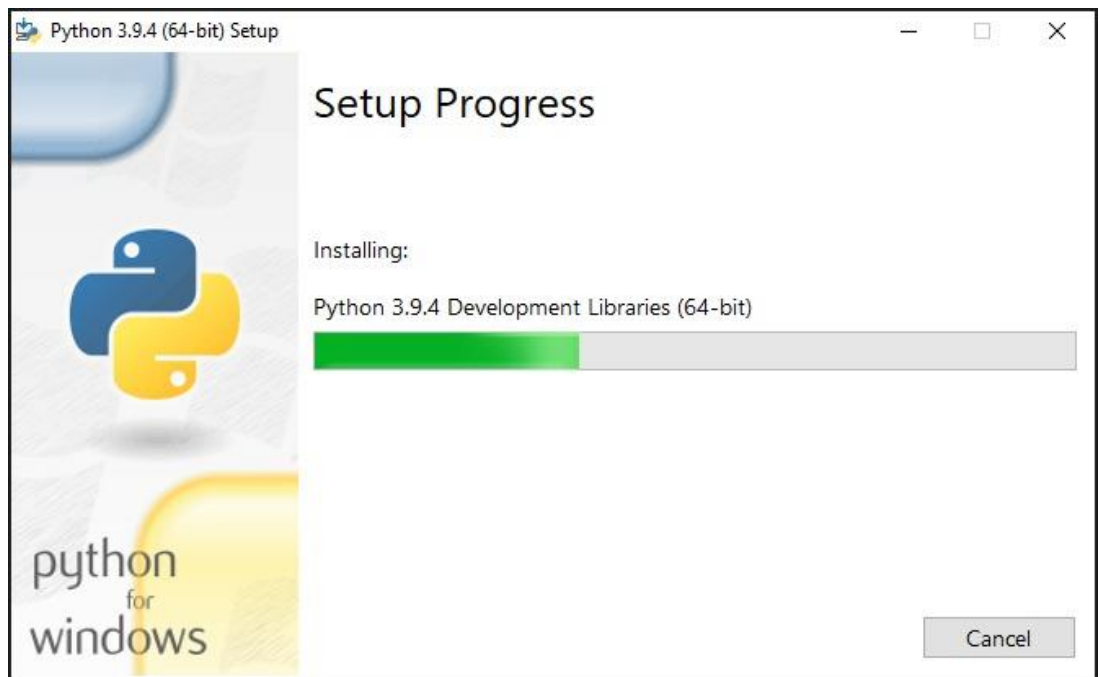
PROGRAM STUDI S1 REKAYASA PERANGKAT LUNAK
FAKULTAS INFORMATIKA
INSTITUT TEKNOLOGI TELKOM
PURWOKERTO
2021

1. Instalasi Software Python

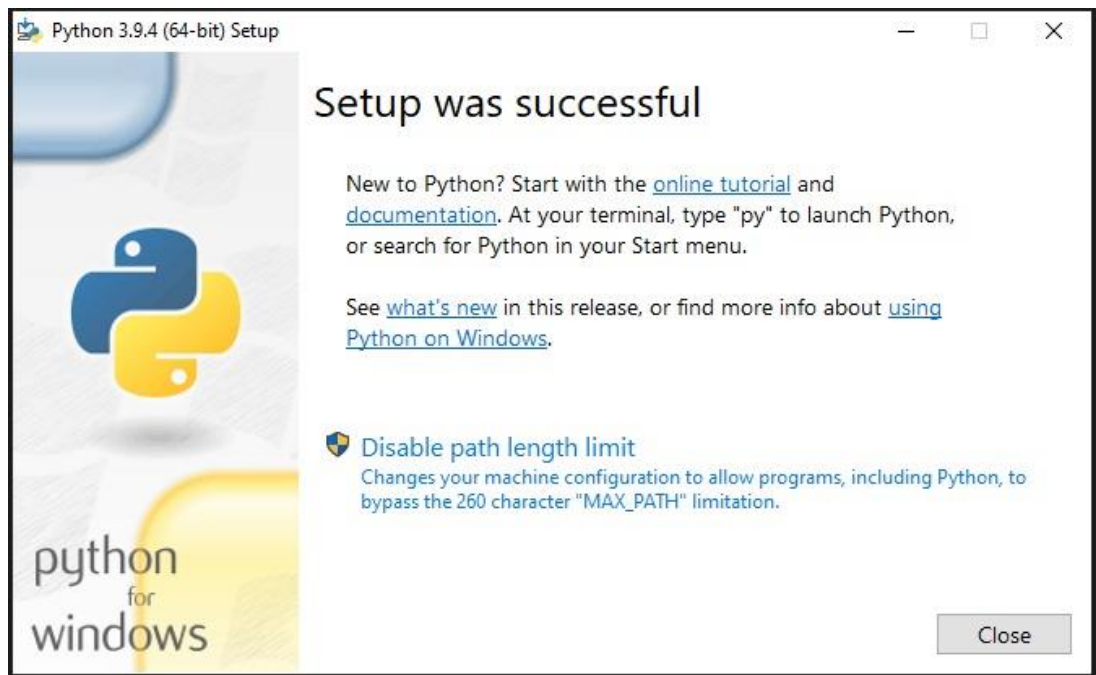
- 1) Download Python 3.5.0-amd64 atau versi lain yang sesuai dengan kebutuhan.
- 2) Buka file download tersebut, maka akan tampil jendela pertama pada instalasi Python dan centang pada checkbox “Add Python 3.5 to PATH”. Lalu klik Install Now.



- 3) Lalu akan muncul progress penginstalasian Python

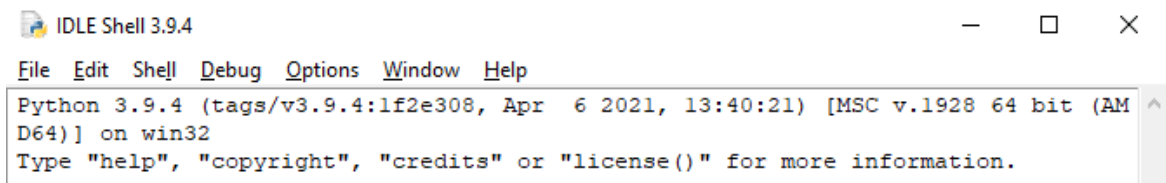


4) Python sudah terinstall



A. Menggunakan IDLE (Python Shell)

Untuk menjalankan Python Shell, tuliskan python 3 lalu akan muncul tampilan IDLE Python seperti di bawah ini.



Selanjutnya Python IDLE sudah dapat dijalankan sesuai instruksi yang diberikan.

```
>>> nama = "Liam"
>>> nama
'Liam'
>>> print(nama)
Liam
>>> print(nama, " berumur ", umur, " tahun")
Liam berumur 20 tahun
>>> |
```

2. Membuat dan Mengeksekusi Kode Program Pada Python

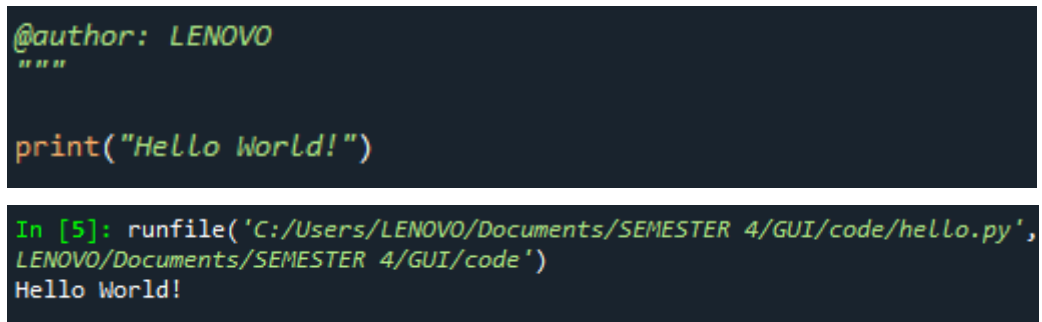
A. Membuat Kode Program

Untuk membuat kode program dengan nama hello.py, langkah-langkah yang diperlukan adalah sebagai berikut:

- 1) Buat direktori tempat penyimpanan
- 2) Jalankan program teks editor
- 3) Tuliskan kode

```
Print('Hello World!')
```

- 4) Simpan file tersebut dengan nama hello.py



The image shows a code editor window with a dark background. The first part shows a Python script with a comment and a print statement: `@author: LENOVO`, `"""`, and `print("Hello World!")`. The second part shows a terminal window with the command `In [5]: runfile('C:/Users/LENOVO/Documents/SEMESTER 4/GUI/code/hello.py', LENOVO/Documents/SEMESTER 4/GUI/code')` and the output `Hello World!`.

3. Variabel dan Objek

Python merupakan Bahasa dengan dynamic typing yaitu variabelnya tidak dibatasi oleh tipe datanya. Berikut adalah contoh kode yang menunjukkan variabel x yang pertama berisi tipe integer. Lalu, variabel tersebut digunakan untuk menampung nilai dari tipe data yang lain, seperti bool dan string. Sehingga satu variable dapat berubah-ubah tipe datanya sesuai dengan kebutuhan.

```
>>> x = 9
>>> type(x)
<class 'int'>
>>> x = True
>>> type(x)
<class 'bool'>
>>> x = 'contoh'
>>> type(x)
<class 'str'>
```

Lalu, buatlah variabel x bernilai 9, dan y memiliki nilai yang berbeda kemudian panggil id nya.

```
>>> x = 9
>>> id(x)
2108763892272
>>> y = 3
>>> id(y)
2108763892080
```

Gunakan perintah 'del' untuk menghapus variabel y.

```
>>> x = 9
>>> id(x)
2108763892272
>>> y = 3
>>> id(y)
2108763892080
>>> del y
>>> y
Traceback (most recent call last):
  File "<pyshell#15>", line 1, in <module>
    y
NameError: name 'y' is not defined
>>> x
9
>>> id(x)
2108763892272
```

Kemudian pindahkan objek variabel x dari objek '9' ke objek 'True'. Dengan demikian objek lama (9) akan diklaim sebagai sampah karena objek tersebut tidak ditunjuk oleh variable apapun.

```
>>> x = True
>>>
```

4. Python Bersifat Case-Sensitive

Penulisan kode program pada python bersifat case sensitive. Dengan demikian misal variable 'Posisi' akan berbeda dengan variable 'posisi'.

```

...
>>> posisi = (300, 300)
>>> posisi
(300, 300)
>>>
>>> Posisi
Traceback (most recent call last):
  File "<pyshell#23>", line 1, in <module>
    Posisi
NameError: name 'Posisi' is not defined
>>>

```

5. Perintah Program (Statement)

Titik koma pada python hanya diberikan pada saat ada dua atau lebih statement pada satu baris yang sama.

```

>>> a = 1; b = 1; c = 3
>>> print(a); print(b); print(c)
1
1
3

```

Secara umum perintah program ditulis dalam satu baris kode, tetapi jika perintah yang dituliskan panjang maka anda dapat memecah perintah tersebut menjadi beberapa baris. Dimana setiap baris harus dihubungkan dengan tanda backslash (\).

```

>>> x = 9
>>> if isinstance(x, int) and \
    x > 0 and \
    x%2 == 1 :
    print("%d adalah bilangan bulat ganjil positif" %x)

9 adalah bilangan bulat ganjil positif
>>>

```

Tetapi tanda backslash tidak diperlukan jika kita menulis perintah kode dalam bentuk array atau kode yang terdapat diantara tanda (...), [...] atau {...}.

```

9 adalah bilangan bulat ganjil positif
>>> print("Pemrogram GUI" + "dengan Python dan PyQt")
Pemrogram GUIDengan Python dan PyQt
>>> data = [
    100,
    200,
    300
]
>>> kamus = {
    'one' : 'satu',
    'book' : 'buku',
    'what' : 'apa'
}
>>> data
[100, 200, 300]
>>> kamus
{'one': 'satu', 'book': 'buku', 'what': 'apa'}

```

6. Tipe Numerik

A. Bilangan Bulat

Dalam python terapat dua tipe bilangan bulat yaitu int dan bool. Selain tipe integral primitive python juga dapat menggunakan bilangan integral dengan basis decimal (10), biner (2), octal (8) maupun heksadesimal (16).

```
>>> a = 0b1001 #bil biner
>>> b = 0o23 #bil oktal
>>> c = 0x2f #bil heksadesimal
>>> a
9
>>> b
19
>>> c
47
```

Tipe bilangan bulat yang kedua adalah tipe Boolean, dimana seperti yang telah kita ketahui tipe data boolean bernilai true atau false saja.

```
>>> a = True
>>> type(a)
<class 'bool'>
>>> int(a)
1
```

Proses perhitungan dan penambahan bilangan pada python akan menghasilkan objek baru, hal ini terlihat dari id nya.

```
>>> a = 12
>>> id(a)
2108763892368
>>> a +=3
>>> a
15
>>> id(a)
2108763892464
```

B. Bilangan Riil

Untuk tipe bilangan riil, python menyediakan tipe float, desimal. Desimal dan complex. Tipe bilangan float menggunakan titik untuk tanda desimalnya.

```
>>> a = 12.432
>>> a
12.432
>>> a*3
37.296
```

Sedangkan untuk tipe decimal hampir sama dengan tipe data float, akan tetapi tipe decimal digunakan untuk melakukan perhitungan dengan nilai koma yang lebih presisi.

7. Tipe String

Tipe data string dalam python direpresentasikan dengan tipe str. Objek string dapat dibuat dengan tiga cara yaitu:

- 1) Menggunakan tanda petik tunggal
- 2) Menggunakan tanda petik ganda
- 3) Menggunakan tanda petik tunggal ataupun ganda yang direpetisi sebanyak 3 kali

```
>>>
>>> s1 = 'Pemrograman Python'
>>> s2 = "Pemrograman Python 2"
>>> s3 = '''Pemrograman Python 3'''
>>>
```

Objek dalam string tidak dapat dirubah, tiap karakter di dalam string dapat diakses dengan tanda [] diikuti nomor array-nya.

```
>>> s1[0], s1[1], s1[2]
('P', 'e', 'm')
```

Dalam string dapat diberikan karakter khusus antara lain \n untuk memberikan enter, \' untuk memberikan petik tunggal, \t untuk memberikan tab. Pada dasarnya karakter khusus dalam python harus diawali dengan backslash (\) diikuti dengan karakter khususnya.

```
>>> data = 'p023\tbolpoin\t\nj2\tpenghapus\t\t3000'
>>> print(data)
p023      bolpoin
j2        penghapus          3000
```

A. Membandingkan String

Untuk membandingkan kesamaan string python menggunakan operator ==. Sedangkan untuk membandingkan id objek string menggunakan is. Selain kedua operator tersebut, python juga dapat menggunakan operator lainnya untuk membandingkan tipe data string.


```
>>> s1 = 'Python'
>>> s2 = 'PYTHON'
>>> s1 == s2
False
>>> s1 != s2
True
>>> s1 < s2
False
>>> s1 > s2
True
```

B. Mengekstrak Substring

Substring di dalam string dapat diekstrak dengan menggunakan operator slice (:) dengan menyertakan indeks awal dan akhir sebagai penanda.

```
>>> s = 'Pemrograman Python dan PyQt'
>>> s1 = s[0:11]
>>> s1
'Pemrograman'
>>> len(s1)
11
```

Kode tersebut mengambil substring dari variable s mulai dari indeks ke 0 sampai indeks ke 11. Jika kita tidak menyertakan indeks maka string yang akan diekstrak adalah sepanjang string tersebut / string sisanya.

```
>>> s = s[:11]
>>> s = s[:8]
>>> s = s[8:]
>>> s = s[0:11:2]
>>> s = s[0:11:1]
>>> s = s[0:11:3]
>>> s
''
```

C. Membuat String dengan Format Tertentu

Python pada dasarnya juga dapat menggabungkan tipe data atau format lain ke dalam string yang telah dibuat. Antara lain dengan menggunakan \$d, %f, %s dan lain sebagainya.

```
>>> s = 'balonku ada %d, meletus %d tinggal %f' % (5,1,4)
>>> s
'balonku ada 5, meletus 1 tinggal 4.000000'
```

8. Tipe Koleksi

Tipe koleksi biasa disebut dengan tipe container. Beberapa tipe koleksi antara lain list, dictionary, tuple dan set. Objek list dibuat dengan menggunakan tanda [], setiap objek yang berada di dalamnya dipisahkan dengan menggunakan koma dan dapat terdiri dari berbagai macam tipe data.

```
>>> list = ['piring', 'ani', 'ada', 2]
>>> list
['piring', 'ani', 'ada', 2]
```

Model dan cara akses list dapat digabungkan dengan fungsi perulangan dasar seperti for, while dan lain sebagainya.

```
>>> list = ['piring', 'ani', 'ada', 2]
>>> list
['piring', 'ani', 'ada', 2]
>>> for item in list :
    print(item)

piring
ani
ada
2
```

Untuk menghapus atau merubah elemen pada list anda dapat menggunakan perintah “del namaList[‘indeks_list’]” sedangkan untuk merubah dapat menggunakan perintah “namaList[‘indeks’] = value baru”. Untuk menambahkan elemen pada list anda dapat menggunakan perintah “namaList.extend([list])”.

```
>>> list.insert(1, 'kaca')
>>> list
['piring', 'kaca', 'ani', 'ada', 2]
>>> del list[0]
>>> list
['kaca', 'ani', 'ada', 2]

>>> list.extend(['besar'])
>>> list
['kaca', 'ani', 'ada', 2, 'besar']
>>> list[5] = 'kecil'
```