

Python Fundamental

Ch.1 : Programming Language

Ch.2 : Python

Instruktur : Ahmad Rio Adriansyah

Nurul Fikri Komputer

Program Komputer

- Sederetan instruksi terstruktur yang dieksekusi oleh komputer untuk melakukan pekerjaan tertentu.
- Analoginya seperti resep masakan
 - Program = resep
 - Variabel = bahan
 - Statement = cara penggunaan
 - Komputer = juru masak

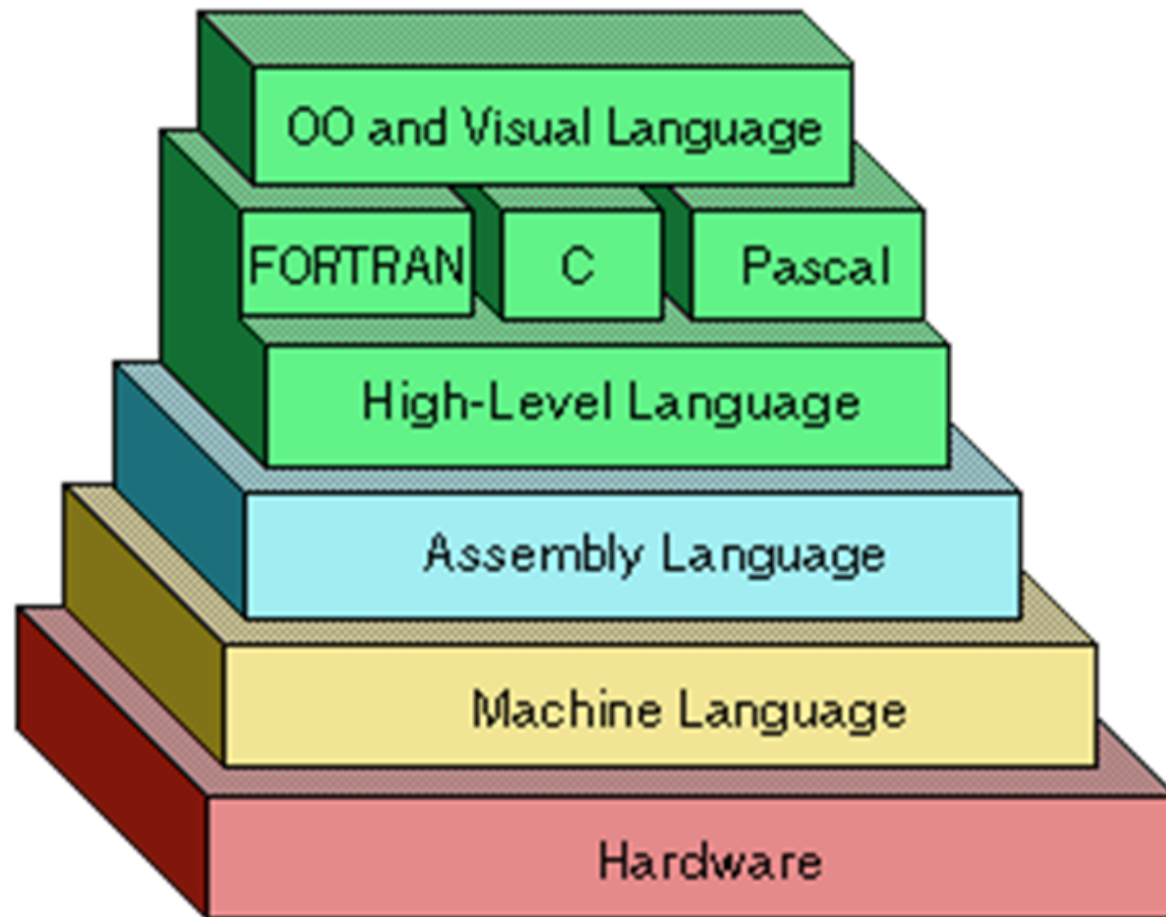
Program Komputer

- Untuk apa?
 - Membantu pekerjaan manusia
 - Mengotomasi sesuatu yang rutin
 - Hiburan (multimedia, game,dll)

Programmer

- Orang yang membuat program
- Menyusun instruksi dengan bahasa pemrograman tertentu agar dapat diproses oleh komputer
- Karena komputer hanya mengerti bahasa mesin (biner, nol dan satu)

Bahasa Pemrograman



Compiler dan Interpreter

- Bahasa pemrograman high-level lebih mudah dipahami manusia, tapi sulit dipahami oleh mesin
- Butuh penerjemah yang menerjemahkan bahasa pemrograman high-level ke bahasa mesin
- Compiler dan Interpreter menerjemahkan ke bahasa mesin dengan cara yang berbeda

Compiler

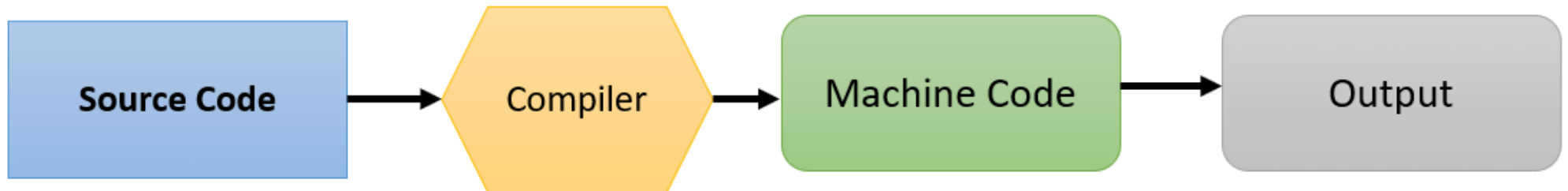
- Membaca keseluruhan kode program (source code) dan menterjemahkannya menjadi satu set instruksi lengkap dalam bahasa mesin sebelum instruksinya dijalankan.
- Saat menjalankan program, yang digunakan adalah instruksi dalam bahasa mesin (object code/machine code). Tidak perlu diterjemahkan lagi jadi prosesnya lebih cepat.

Interpreter

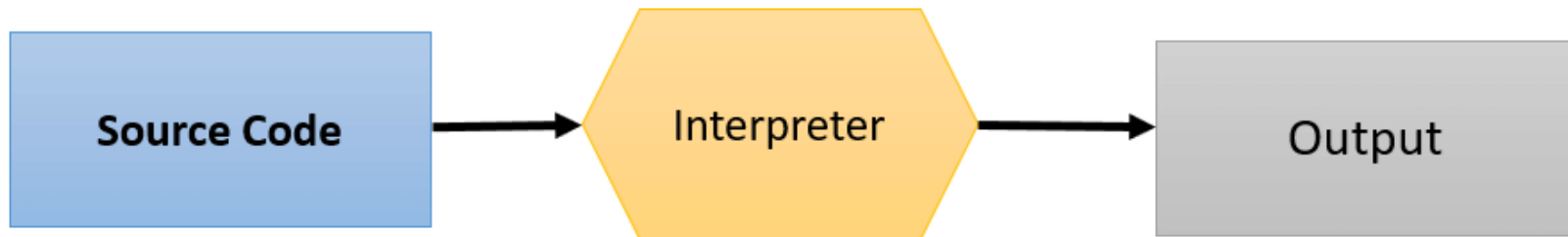
- Instruksi-instruksi dalam kode program dibaca, diterjemahkan, dan langsung dilaksanakan satu per satu.
- Tidak ada hasil terjemahan dalam bahasa mesin jadi setiap kali mau menjalankan program dibutuhkan interpreter.

Compiler dan Interpreter

How Compiler Works



How Interpreter Works



Python

- Python adalah bahasa pemrograman yang:
 - High level
 - Interpreted (butuh interpreter)
 - Dinamis berorientasi objek
 - General purpose
- Diciptakan oleh Guido van Rossum (BDFL sampai tahun 2018)
- Diambil dari nama acara komedi Monty Python



Kenapa Python?

- Bahasanya mudah dipelajari karena dekat dengan bahasa manusia
- Komunitasnya besar, banyak modul yang bisa digunakan
- Populer (urutan no.1 tahun 2019)

Worldwide, Dec 2018 compared to a year ago:

Rank	Change	Language	Share	Trend
1	↑	Python	25.36 %	+5.2 %
2	↓	Java	21.56 %	-1.1 %
3	↑	Javascript	8.4 %	+0.0 %
4	↑	C#	7.63 %	-0.4 %
5	↓↓↓	PHP	7.31 %	-1.3 %
6		C/C++	6.4 %	-0.4 %
7		R	4.01 %	-0.3 %
8		Objective-C	3.21 %	-0.9 %
9		Swift	2.69 %	-0.7 %
10		Matlab	2.06 %	-0.3 %

Versi Python

- Ada 2 versi python yang sering digunakan :
 - Python 2 (2.7)
 - Python 3 (versi terbaru 3.7.3 per Maret 2019)
- **Python 2 is legacy, python 3 is the future.**
- Tahun 2016, 71.9% project menggunakan python 2. Tahun 2017 turun ke 63.7%

PYTHON 2

PYTHON 3



Legacy

It is still entrenched in the software at certain companies



Library

Many older libraries built for Python 2 are not forwards-compatible

0100
0001

ASCII

Strings are stored as ASCII by default



$$5/2=2$$

It rounds your calculation down to the nearest whole number

`print "hello"`

Python 2 print statement



Future



It will take over Python 2 by 2020

Library



Many of today's developers are creating libraries strictly for use with Python 3

Unicode

0000
0000
0100
0001

Text strings are Unicode by default

$$5/2=2.5$$



The expression 5 / 2 will return the expected result

`print ("hello")`

The print statement has been replaced with a `print ()` function

Zen of Python

The Zen of Python, by Tim Peters

Beautiful is better than ugly.

Explicit is better than implicit.

Simple is better than complex.

Complex is better than complicated.

Flat is better than nested.

Sparse is better than dense.

Readability counts.

Special cases aren't special enough to break the rules.

Although practicality beats purity.

Errors should never pass silently.

Unless explicitly silenced.

In the face of ambiguity, refuse the temptation to guess.

There should be one-- and preferably only one --obvious way to do it.

Although that way may not be obvious at first unless you're Dutch.

Now is better than never.

Although never is often better than *right* now.

If the implementation is hard to explain, it's a bad idea.

If the implementation is easy to explain, it may be a good idea.

Namespaces are one honking great idea -- let's do more of those!

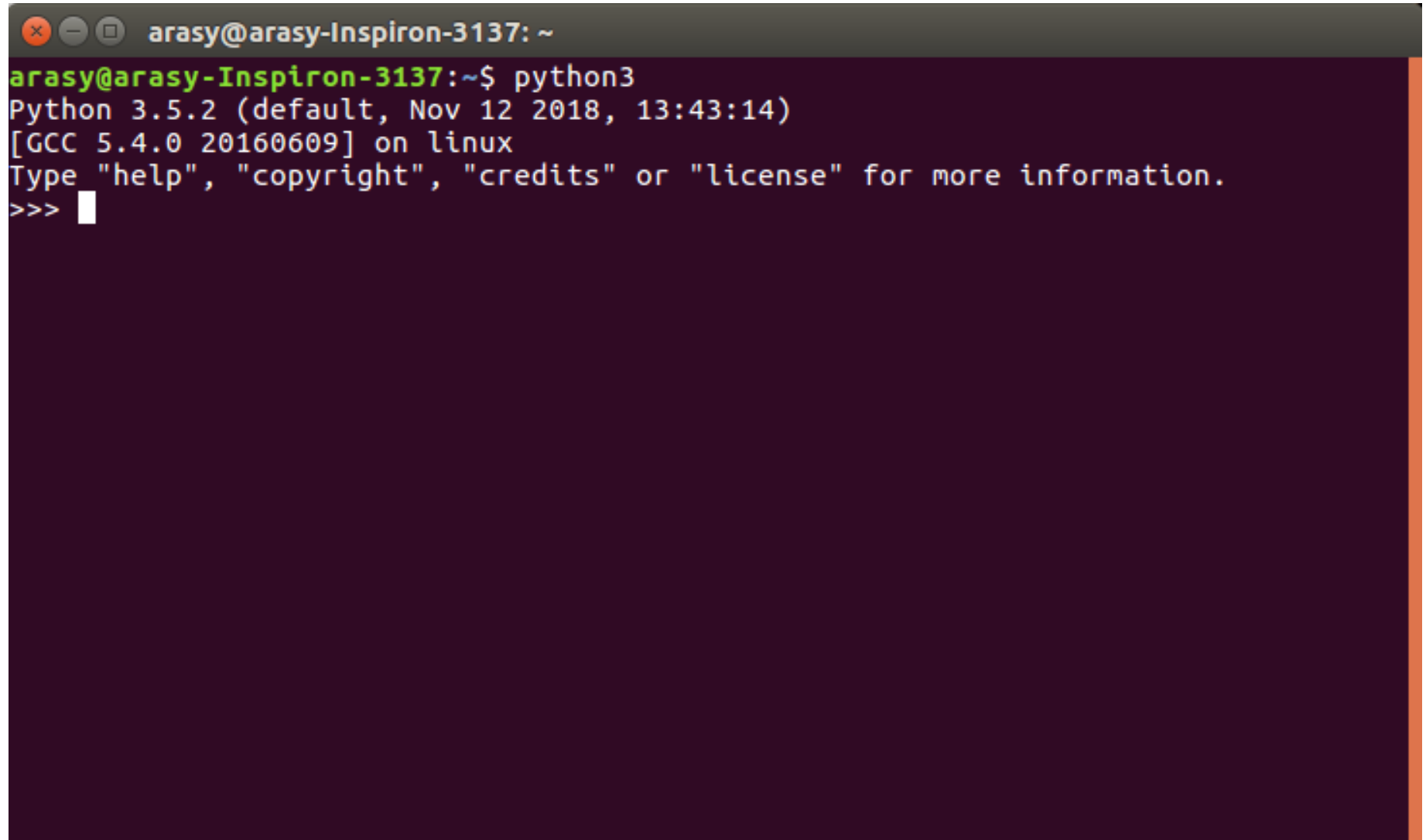
Instalasi

- Windows
 - Buka python.org
 - Pilih versi python yang diinginkan
 - Install
 - Jangan lupa dicentang pilihan memasukkan python ke path!
- Linux
 - Tidak perlu pusing, sudah ada

Menjalankan Python

- Ada 2 mode untuk menjalankan python
 - Melalui shell interaktif (python shell)
 - Melalui text file yang dieksekusi (python script)
- Python shell
 - Windows
 - Masuk ke start menu
 - Ketik python
 - Linux
 - Masuk ke terminal (Ctrl+Alt+T)
 - Ketik python (python 2) atau python3 (python 3)

Python Shell

A terminal window with a dark purple background and a grey title bar. The title bar contains three window control icons (close, minimize, maximize) and the text 'arasy@arasy-Inspiron-3137: ~'. The terminal text is as follows:

```
arasy@arasy-Inspiron-3137:~$ python3
Python 3.5.2 (default, Nov 12 2018, 13:43:14)
[GCC 5.4.0 20160609] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> 
```

Konstanta dan Variabel

- Dalam setiap bahasa pemrograman, pasti digunakan konstanta dan variabel sebagai data yang diolah oleh komputer
- Konstanta adalah nilai yang dituliskan langsung
 - Konstanta/data Numerik
 - Konstanta/data String
- Variabel adalah suatu besaran yang nilainya bisa berubah, digunakan untuk merepresentasikan nilai-nilai konstanta.

Konstanta

- Konstanta Numerik :
 - Bilangan bulat (integer) : nilai numerik yang tidak mengandung titik desimal seperti 10, 50, -15, 311
 - Bilangan desimal (floating point) : bilangan riil yang mengandung titik desimal seperti 0.2, 1.25, -11.759
- Konstanta String
 - Sederetan karakter alfanumerik (huruf dan angka)
 - Dituliskan diantara tanda petik (' ') atau petik dua (" ")
 - Contohnya "Nama", 'A', "Inspiron 3176", 'XL'

Variabel

- Berfungsi seperti wadah untuk konstanta dan nilai-nilai lainnya
- Program menggunakan variabel untuk menyimpan data ke dalam memori secara sementara
- Biasanya (di bahasa pemrograman lain) variabel harus dideklarasikan dulu sebelum digunakan. Di python hal ini tidak perlu dilakukan
- Nama variabel (identifier) harus memenuhi aturan tertentu

Aturan Penamaan Variabel

- Harus dimulai dengan huruf atau karakter garis bawah/underscore (_)
- Tidak boleh diawali angka
- Hanya boleh mengandung alfanumerik dan underscore (A-Z, a-z, _)
- Case sensitive
- Tidak boleh sama seperti reserved words di python (keywords)

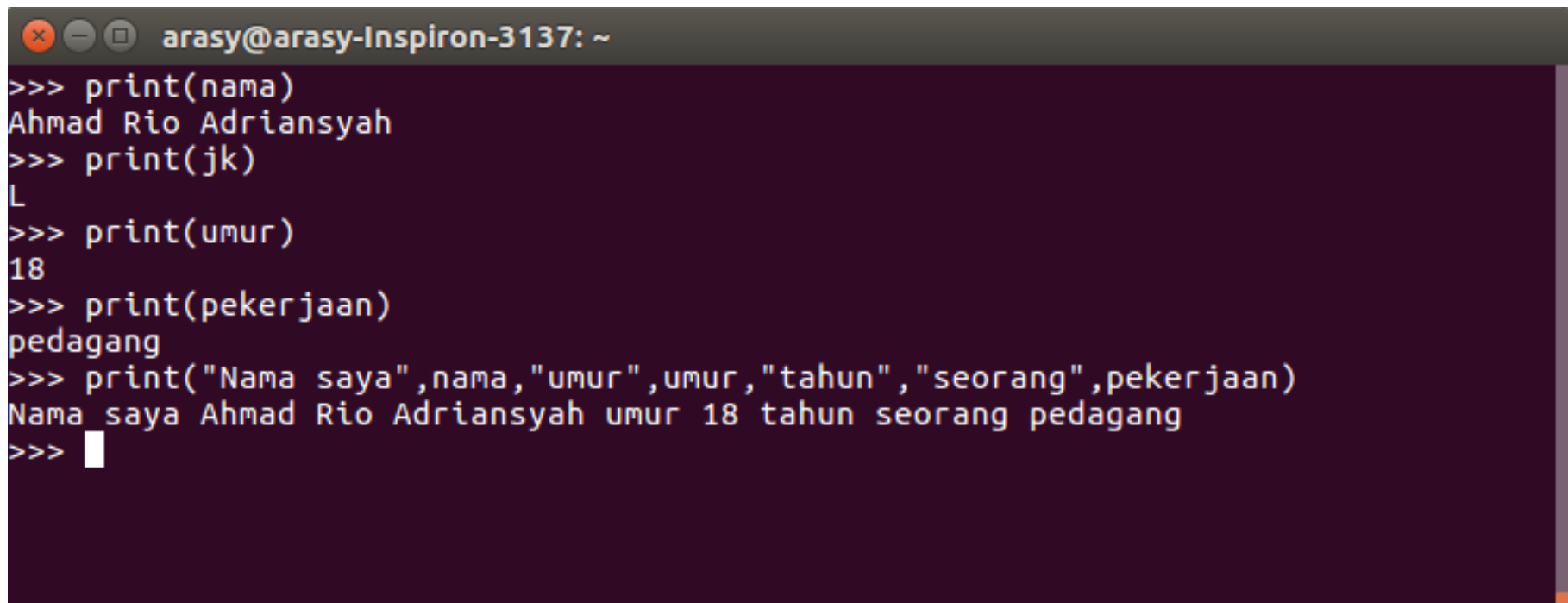
Latihan

- Buat informasi tentang data diri anda dalam variabel-variabel yang sesuai

```
arasy@arasy-Inspiron-3137: ~  
arasy@arasy-Inspiron-3137:~$ python3  
Python 3.5.2 (default, Nov 12 2018, 13:43:14)  
[GCC 5.4.0 20160609] on linux  
Type "help", "copyright", "credits" or "license" for more information.  
>>> nama = "Ahmad Rio Adriansyah"  
>>> jk = 'L'  
>>> umur = 18  
>>> pekerjaan = 'pedagang'  
>>> 
```

Latihan

- Tampilkan informasi tadi dengan memanggil variabelnya
- Gunakan fungsi `print()` untuk menampilkan di console



```
arasy@arasy-Inspiron-3137: ~  
>>> print(nama)  
Ahmad Rio Adriansyah  
>>> print(jk)  
L  
>>> print(umur)  
18  
>>> print(pekerjaan)  
pedagang  
>>> print("Nama saya",nama,"umur",umur,"tahun","seorang",pekerjaan)  
Nama saya Ahmad Rio Adriansyah umur 18 tahun seorang pedagang  
>>> 
```


Exit dari Python Shell

- Untuk keluar dari python shell user dapat melakukan :
 - Menekan Ctrl+D
 - Mengetikkan `exit()` dan menjalankan perintah tersebut
- Silahkan coba keluar dan masuk kembali ke dalam python shell

Python Script

- Cara kedua menjalankan kode python adalah dengan menuliskannya dalam teks
- Buka teks editor, ketik informasi yang tadi ditulis
- Simpan dalam nama “**biodata.py**”
- Buka terminal, jalankan file yang kita buat tadi dengan perintah

```
$ python biodata.py
```

Python Script

- Bisa juga python script dijalankan sebagai executable file di linux.
- Cari posisi direktori python. Defaultnya di `/usr/bin/python`, atau cari dengan
`$ whereis python`
- Tambahkan `#!/usr/bin/python` di baris pertama python script
- Ubah atribut filenya agar menjadi executable
`$ chmod 755 biodata.py`
- Jalankan filenya
`$./biodata.py`

Format Bahasa Python

- Python dituliskan dalam blok kode yang disebut sebagai fungsi
- Ada fungsi utama di bahasa pemrograman
 - Di C++ ada `main { }`
 - Di pascal ada `Begin End`
 - Di python menggunakan `__main__`
- Blok fungsi dalam python ditandai dengan indentasi (spasi/tab), biasanya digunakan 4 spasi

Format Bahasa Python

```
1 #!/usr/bin/python3
2 print("Bismillah")
3
4 def mainFunction():
5     print("Ini adalah fungsi utama python")
6     print("Tidak harus pakai seperti ini sih")
7     print("Cuma digunakan untuk menunjukkan indentasi")
8
9 if __name__ == '__main__':
10     mainFunction()
```

- Note : nomor di kiri adalah nomor baris, bawaan dari text editor atau IDE nya, bukan termasuk bagian dari kode. Jangan diketik.
- Simpan sebagai “**fungsiutama.py**” dan jalankan