

# Laporan Tugas

## Chapter I



Muchamad Innal Kariem

NPM : 1.18.4.036

Program Diploma IV Teknik Informatika

*Applied Bachelor Program of Informatics Engineering*

Bandung 2019

# Chapter 1

## Mengenai Python dan Anaconda

### 1.1 Teori

#### 1.1.1 Sejarah Python

Python adalah bahasa pemrograman yang dinamis, yang mendukung pemrograman berbasis suatu objek. Python itu sendiri dikembangkan oleh Guido Van Rossum pada tahun 1990-an di CWI, Amsterdam. Bahasa pemrograman ini merupakan kelanjutan dari bahasa pemrograman ABC. Nama python itu sendiri diambil dari kegemaran Guido Van Rossum pada salah satu acara humor di televisi pada era 1980-an yang berjudul “Monty Python’s Flying Circus”. Bahasa pemrograman python menggunakan metode pemrosesan interpreted, yaitu code pemrograman akan diproses baris per baris langsung dari kode program. Dan bahasa pemrograman ini juga disebut sebagai bahasa pemrograman tingkat tinggi serta bisa dipakai untuk berbagai jenis tujuan. Pada tahun 1995 Guido pindah ke CNRI yang berada di Virginia Amerika untuk melanjutkan perkembangan bahasa pemrograman python itu sendiri, dan merilis beberapa versi terbaru dari python. Hampir semua pengembangan python dirilis menggunakan lisensi GFL-compatible.

#### 1.1.2 Perbedaan Python 2 dan 3

Python yang digunakan sekarang adalah Python versi 2 dan versi Python 3, Namun pada setiap pengembangan versi tentu saja memiliki peningkatan kualitas seperti yang terjadi di Python versi 2 dan versi Python 3. Perbedaan python versi 2 dan python versi 3 yaitu ketika membuat kodingan di Python 2 dan di compile di shell python versi 3 akan terjadi error karena script python versi 2 sudah tidak kompatibel di shell python versi 3, karena di python versi 3 memerlukan tanda kurung () sedangkan di python versi 2 tidak memerlukannya. Di python versi 3 akan terlihat lebih rapih

dibandingkan di python versi 2 . Dan pada python versi 2 dilengkapi dengan berbagai fitur programatikal sedangkan pada python versi 3 itu sendiri melakukan perapian pada codebase dan penghapusan redudansi.

### **1.1.3 Implementasi dan Penggunaan Python di Perusahaan Dunia**

#### **1. spotify**

Spotify adalah suatu layanan musik streaming yang sedang booming memanfaatkan bahasa pemrograman python untuk analisis data dan backend. Pada backend spotify berkomunikasi dengan OMQ. OMQ itu sendiri adalah suatu framework dan library open source untuk networking. Untuk menginterpretasikan analisis data tersebut spotify menggunakan luigi, dan modul python yang sinkron dengan hadoop. Modul open source ini menangani satu library dengan library lainnya agar saling bekerjasama, serta dapat mengkonsolidasi error log secara cepat.

#### **2. Google**

Google ini sudah menggunakan bahasa pemrograman python ini sudah sejak dari awal berdirinya. Dan pada saat ini bahasa pemrograman python merupakan salah satu bahasa pemrograman server-side resmi di google. Meskipun ada script yang ditulis untuk google menggunakan bahasa perl dan bash, maka nantinya script tersebut akan diubah ke python terlebih dahulu, karena kemudahan dalam perawatannya.

#### **3. Industrial Light and Magic**

Industrial Light and Magic ini merupakan studio special efek yang dibutuhkan untuk film star wars saja. Karena infrastruktur awal industrial light and magic ini menggunakan C dan C++, maka akan lebih mudah mengintegrasikan bahasa pemrograman python ketimbang bahasa pemrograman lainnya. Dengan menggunakan bahasa pemrograman python ini industrial light and magic dengan mudah mengemas komponen software dan dapat meningkatkan aplikasi grafisnya.

#### **4. Netflix**

Netflix adalah suatu layanan pemutaran film yang dapat dilakukan oleh pengguna dimanapun dan kapanpun. Pada netflix bahasa pemrograman yang digunakan adalah bahasa pemrograman python, bahasa pemrograman ini digunakan pada Central Alert Gateway yang akan me-reroute alert dan mengirimkannya pada individu yang akan melihatnya serta juga dapat secara otomatis reboot atau menghentikan proses yang dianggap bermasalah. Selain itu python juga digunakan untuk menelusuri riwayat dan perubahan pengaturan keamanan.

#### 5. instagram

Instagram adalah suatu aplikasi mobile berbasis IOS, android dan windows phone, dimana pengguna dapat berbagi foto dan video melalui instagram ini. Pada instagram ini menggunakan bahasa pemrograman python dalam task queue-nya atau fitur dimana setiap pengguna dapat berbagi foto atau video ke beberapa social network lainnya seperti facebook, twitter, dan lain-lainnya. Selain perusahaan diatas ada beberapa perusahaan pengguna Python lain yaitu : Pinterest, Disqus, Dropbox, Uber, Reddit, Quora, Facebook (Bahasa ke-3 setelah PHP (Hack) dan C++, digunakan untuk manajemen infrastruktur).

## 1.2 Instalasi

### 1.2.1 Instalasi Anaconda 3

Hal yang harus diperhatikan sebelum melakukan instalasi *Anaconda Python*

1. Perhatikan versi dari sistem operasi yang digunakan (versi 32bit atau 64bit)
2. Download file anaconda yang sesuai dengan versi sistem operasi (32bit atau 64bit)
3. *Download Anaconda Python* <https://www.anaconda.com/distribution/>

Berikut langkah-langkah instalasi anaconda.

1. Buka aplikasi *installer Anaconda* tersebut lalu akan muncul gambar *installer anaconda*.

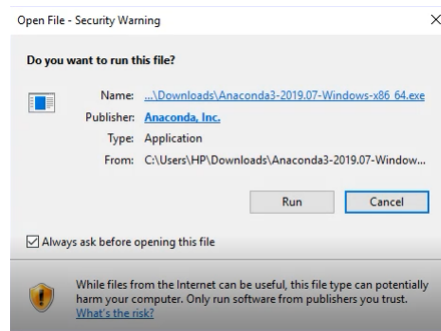


Figure 1.1: Run Setup Anaconda

2. Tunggu sampai *setup loading* selesai

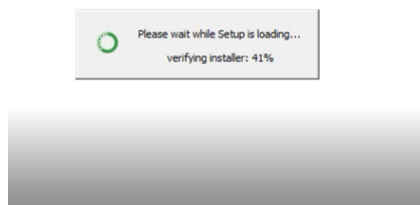


Figure 1.2: Setup Loading

3. Jika *setup loading* sudah selesai, maka selanjutnya klik *next*

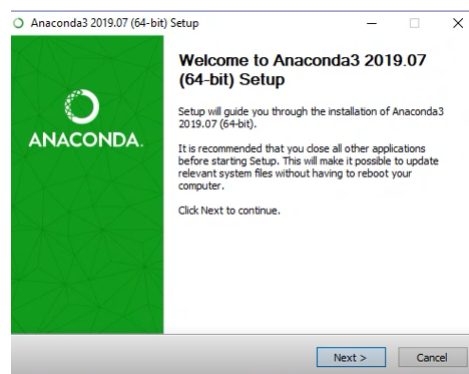


Figure 1.3: Welcome to Anaconda Setup

4. Pada *License Agreement* klik *I Agree* gambar *License Agreement*.

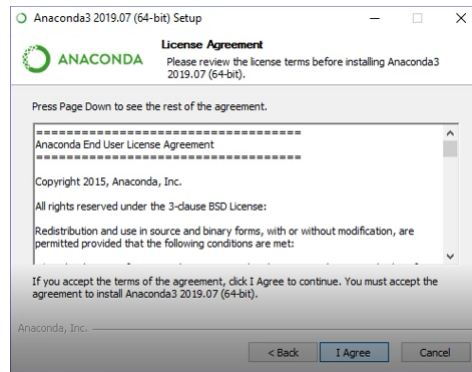


Figure 1.4: *License Agreement*

5. Kemudian pilih *Just Me(Recommended)* agar sesuai dengan komputer yang anda gunakan, kemudian klik *next* gambar *Just Me(recommended)*.

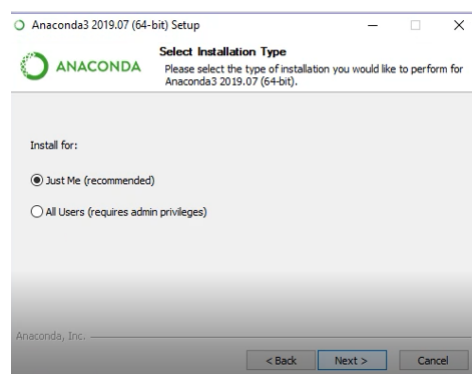


Figure 1.5: *Just Me(recommended)*

6. Kemudian pilih lokasi tempat penyimpanan *installan anaconda* gambar *Pilih lokasi*.

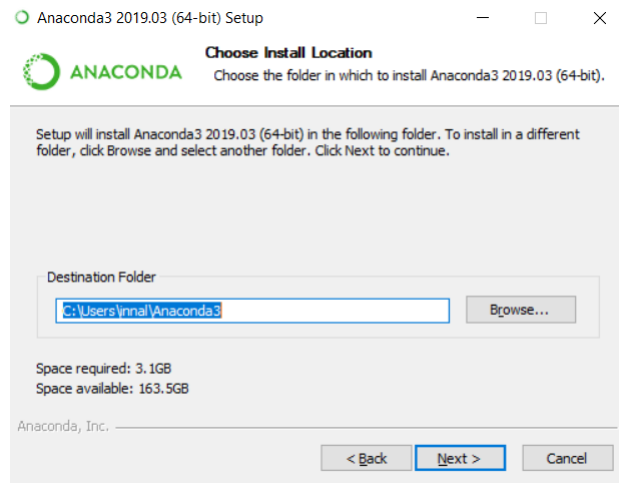


Figure 1.6: *Pilih lokasi*

7. Kemudian centang *Add Anaconda to my Path environment variable*, agar saat *menginstall selenium* langsung ke *path anaconda* tidak ke aplikasi yang lain. Klik *install* gambar *Centang Anaconda to my PATH*.

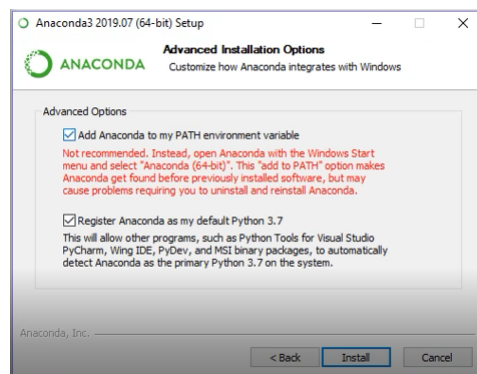


Figure 1.7: *Centang Anaconda to my PATH*

8. Tunggu sampai proses *installasi* selesai gambar *Installation Complete*.

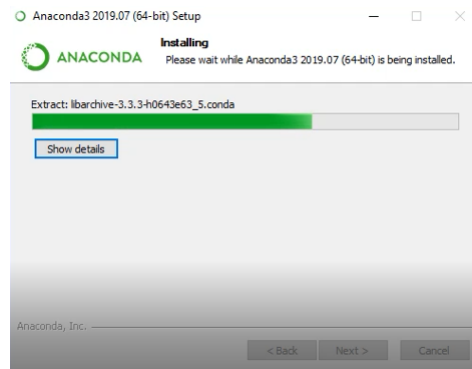


Figure 1.8: *Installation Complete*

9. Apabila instalasi telah selesai klik *next*

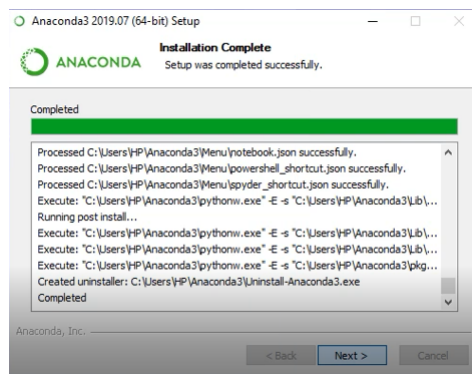


Figure 1.9: *Installation Complete*

10. klik *next*

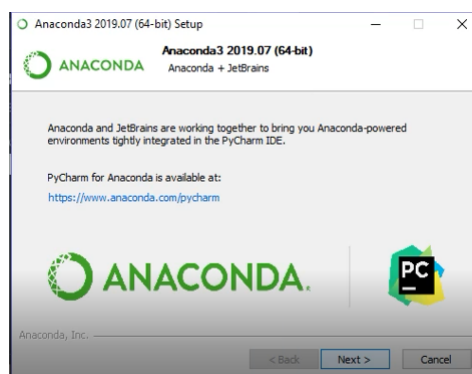


Figure 1.10: *Anaconda+JetBrains*

11. Jika sudah klik *finish* gambar *Thanks fo install Anaconda.*



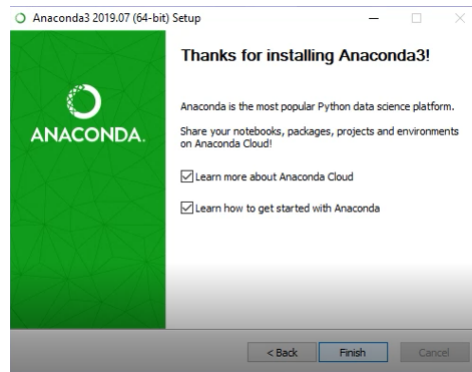


Figure 1.11: *Thanks for install Anaconda*

## 1.2.2 Instalasi Pip

1. buka anaconda promt
2. ketikkan `conda install -c anaconda pip`

```

Command Prompt - conda install -c anaconda pip
Microsoft Windows [Version 10.0.18362.418]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\innal>conda install -c anaconda pip
Collecting package metadata (repodata.json): done
Solving environment: done

## Package Plan ##

  environment location: C:\Users\innal\Anaconda3

  added / updated specs:
    - pip

The following packages will be downloaded:

  package | build | size | channel
  -----|-----|-----|-----
  pip-19.2.3 | py37_0 | 1.9 MB | anaconda
  -----|-----|-----|-----
  Total: | | 1.9 MB |

The following packages will be SUPERSEDED by a higher-priority channel:

  pip | pkgs/main --> anaconda

```

Figure 1.12: *Install pip*

3. ketik `y`, lalu enter. Tunggu hingga proses instalasi selesai.

```
Command Prompt

environment location: C:\Users\innal\Anaconda3

added / updated specs:
- pip

The following packages will be downloaded:

package | build
-----|-----
pip-19.2.3 | py37_0 1.9 MB anaconda
-----|-----
Total: 1.9 MB

The following packages will be SUPERSEDED by a higher-priority channel:

pip pkgs/main --> anaconda

Proceed ([y]/n)? y

Downloading and Extracting Packages
pip-19.2.3 | 1.9 MB | ##### | 100%
Preparing transaction: done
Verifying transaction: done
Executing transaction: done
```

Figure 1.13: *Install pip Selesai*

4. jika telah selesai, lakukan pengecekan versi pip dengan mengetikkan pip -V

```
Command Prompt

added / updated specs:
- pip

The following packages will be downloaded:

package | build
-----|-----
pip-19.2.3 | py37_0 1.9 MB anaconda
-----|-----
Total: 1.9 MB

The following packages will be SUPERSEDED by a higher-priority channel:

pip pkgs/main --> anaconda

Proceed ([y]/n)? y

Downloading and Extracting Packages
pip-19.2.3 | 1.9 MB | ##### | 100%
Preparing transaction: done
Verifying transaction: done
Executing transaction: done

C:\Users\innal>pip -V
pip 19.2.3 from C:\Users\innal\Anaconda3\lib\site-packages\pip (python 3.7)

C:\Users\innal>
```

Figure 1.14: *Melihat Versi pip*

## 1.2.3 Setting Environment

1. Buka file explorer

2. Klik kanan pada This pc, lalu pilih properties

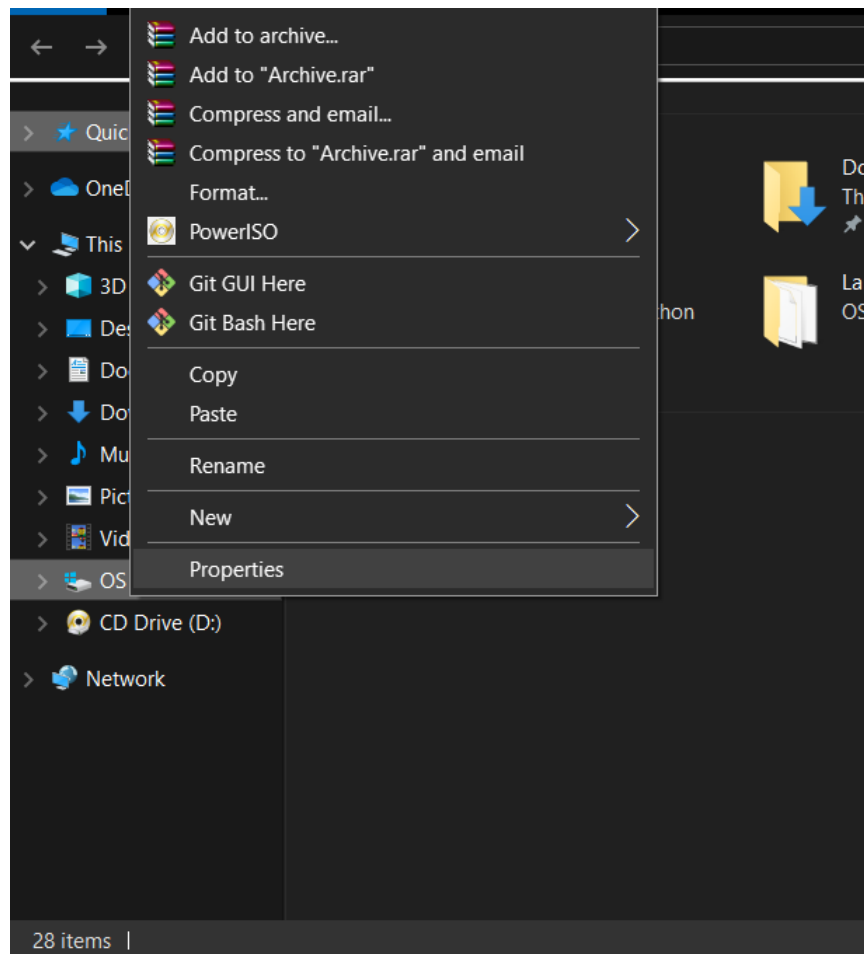


Figure 1.15: *Properties*

3. Pilih menu Advanced system settings

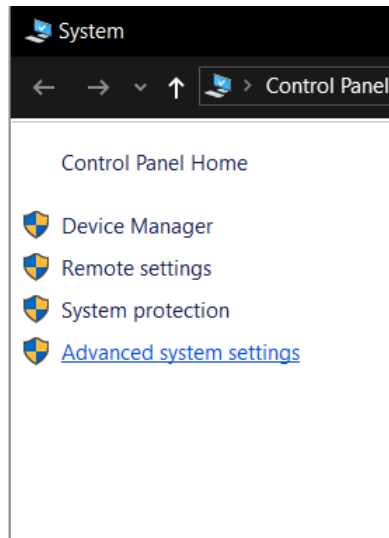


Figure 1.16: *Advanced system settings*

#### 4. Pilih Environment Variables

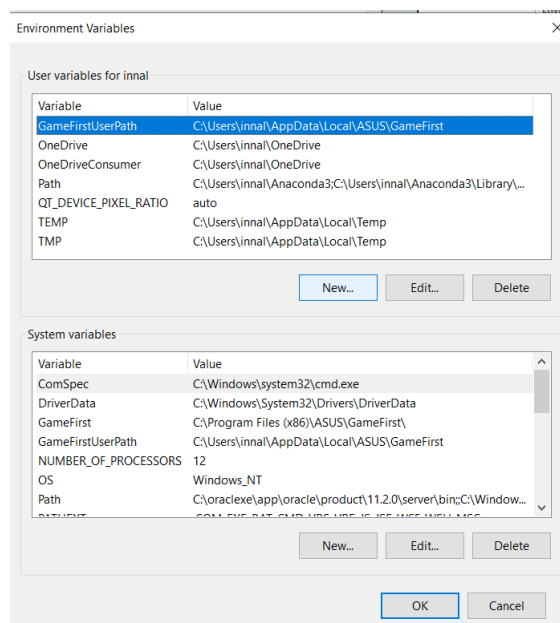


Figure 1.17: *Environment Variables*

#### 5. Pilih Path

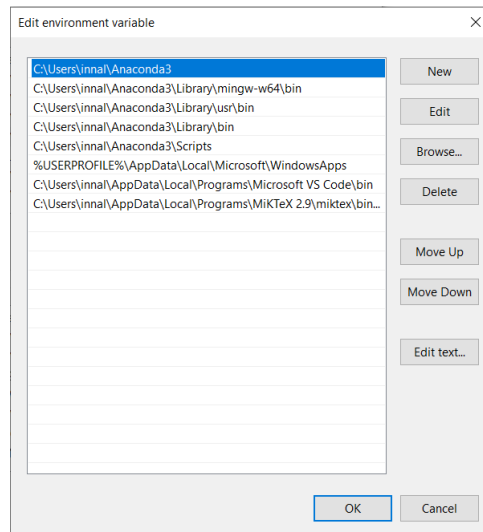


Figure 1.18: *Path*

6. lalu pilih environment variable yang ingin ditambahkan, klik OK

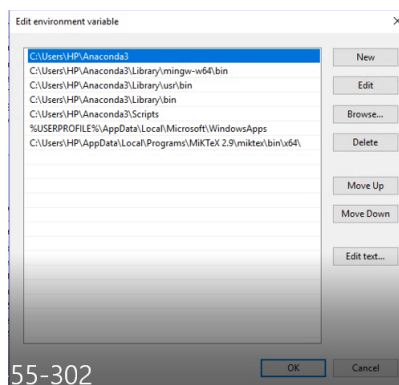
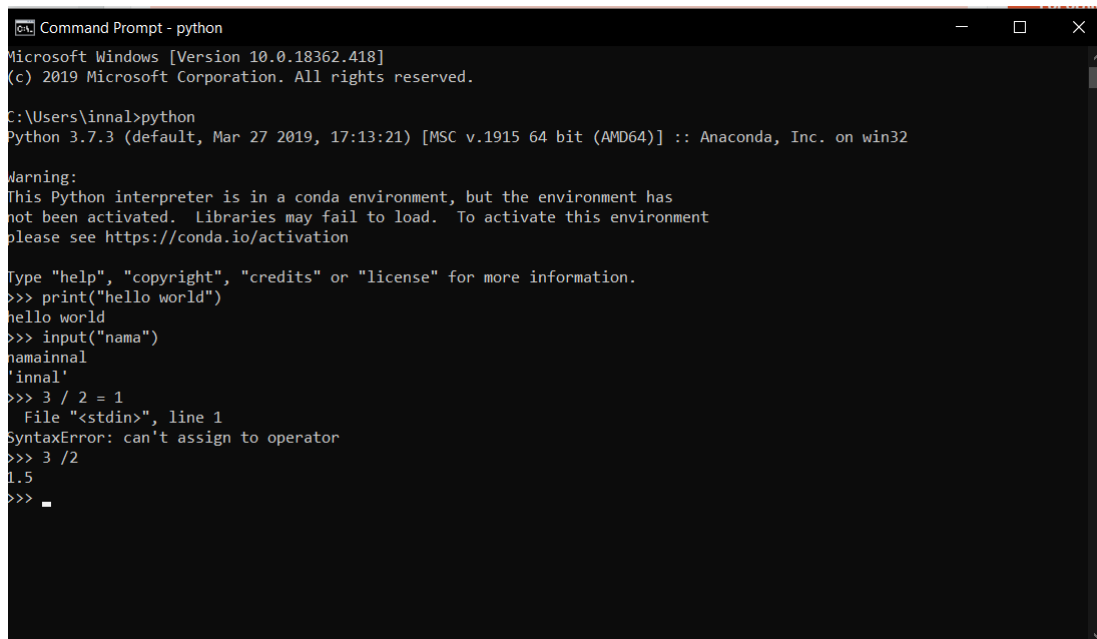


Figure 1.19: *Edit Environment Variable*

#### 1.2.4 Command Line Interface/Interpreter

1. Buka command prompt lalu ketikkan python
2. Buatlah perintah print, input, perkalian, dan pembagian
3. Bisa juga menjalankan file .py yang telah dibuat di IDE dengan cara python namafile.py, lalu klik enter



```
Command Prompt - python
Microsoft Windows [Version 10.0.18362.418]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\innal>python
Python 3.7.3 (default, Mar 27 2019, 17:13:21) [MSC v.1915 64 bit (AMD64)] :: Anaconda, Inc. on win32

Warning:
This Python interpreter is in a conda environment, but the environment has
not been activated. Libraries may fail to load. To activate this environment
please see https://conda.io/activation

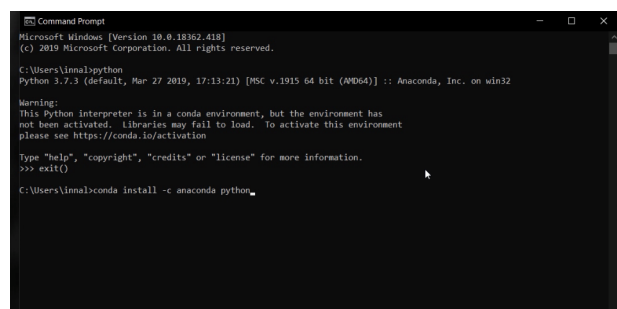
Type "help", "copyright", "credits" or "license" for more information.
>>> print("hello world")
hello world
>>> input("nama")
innal
'innal'
>>> 3 / 2 = 1
File "<stdin>", line 1
SyntaxError: can't assign to operator
>>> 3 / 2
1.5
>>> _
```

Figure 1.20: *CLI in Command Prompt*

## 1.2.5 Update Anaconda dan Spyder

### 1.2.5.1 Anaconda

1. Ketikan `conda install -c anaconda python` di cmd



```
Command Prompt
Microsoft Windows [Version 10.0.18362.418]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\innal>python
Python 3.7.3 (default, Mar 27 2019, 17:13:21) [MSC v.1915 64 bit (AMD64)] :: Anaconda, Inc. on win32

Warning:
This Python interpreter is in a conda environment, but the environment has
not been activated. Libraries may fail to load. To activate this environment
please see https://conda.io/activation

Type "help", "copyright", "credits" or "license" for more information.
>>> exit()
C:\Users\innal>conda install -c anaconda python
```

Figure 1.21: *Update python*

2. Ketikan `y` agar melanjutkan update

```

Command Prompt - conda install -c anaconda python

openssl-1.1.1b | he774522_1 | 5.7 MB | anaconda
python-3.7.3 | h8c8aaf0_0 | 17.8 MB | anaconda
qt-5.9.7 | vc16h73c8ide_0 | 92.3 MB | anaconda
-----
Total: | | 119.4 MB

The following NEW packages will be INSTALLED:
conda-package-handling-1.3.11-py37_0

The following packages will be UPDATED:
conda | pkgs/main::conda-4.6.11-py37_0 --> anaconda::conda-4.7.12-py37_0

The following packages will be SUPERSEDED by a higher-priority channel:
ca-certificates | pkgs/main --> anaconda
certifi | pkgs/main --> anaconda
openssl | pkgs/main --> anaconda
python | pkgs/main --> anaconda
qt | pkgs/main --> anaconda

Proceed ([y]/n)? y

```

Figure 1.22: *Update python yes*

### 3. Update sesesai

```

Command Prompt

_anaconda_depends | anaconda/win-64::_anaconda_depends-2019.03-py37_0
tbb | anaconda/win-64::tbb-2019.4-h74a7933_0

The following packages will be UPDATED:
ca-certificates | 2019.1.23-0 --> 2019.8.28-0
certifi | 2019.3.9-py37_0 --> 2019.9.11-py37_0
openssl | 1.1.1b-he774522_1 --> 1.1.1-he774522_0

The following packages will be SUPERSEDED by a higher-priority channel:
anaconda | pkgs/main::anaconda-2019.03-py37_0 --> anaconda::anaconda-custom-py37_1

Proceed ([y]/n)? y

Downloading and Extracting Packages
certifi-2019.9.11 | 155 KB | ##### | 100%
_anaconda_depends-2019.03 | 5 KB | ##### | 100%
tbb-2019.4 | 173 KB | ##### | 100%
ca-certificates-2019.8.28 | 165 KB | ##### | 100%
anaconda-custom-py37_1 | 2 KB | ##### | 100%
openssl-1.1.1 | 5.7 MB | ##### | 100%

Preparing transaction: done
Verifying transaction: done
Executing transaction: done

C:\Users\innal>

```

Figure 1.23: *Update python selesai*

## 1.2.5.2 Spyder

### 1. Ketikan conda install -c anaconda spyder di cmd

```

Command Prompt - conda install -c anaconda spyder

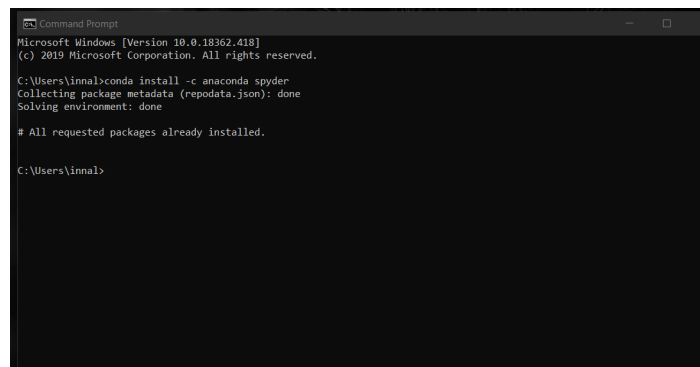
Microsoft Windows [Version 10.0.18362.418]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\innal>conda install -c anaconda spyder
Collecting package metadata (repodata.json): /

```

Figure 1.24: *Update spyder*

2. update selesai



```
Microsoft Windows [Version 10.0.19362.418]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\innal>conda install -c anaconda spyder
Collecting package metadata (repodata.json): done
Solving environment: done

# All requested packages already installed.

C:\Users\innal>
```

Figure 1.25: *Update spyder selesai*

## 1.2.6 Run Script Hello World di Spyder

1. Buka anaconda navigator, lalu klik launch

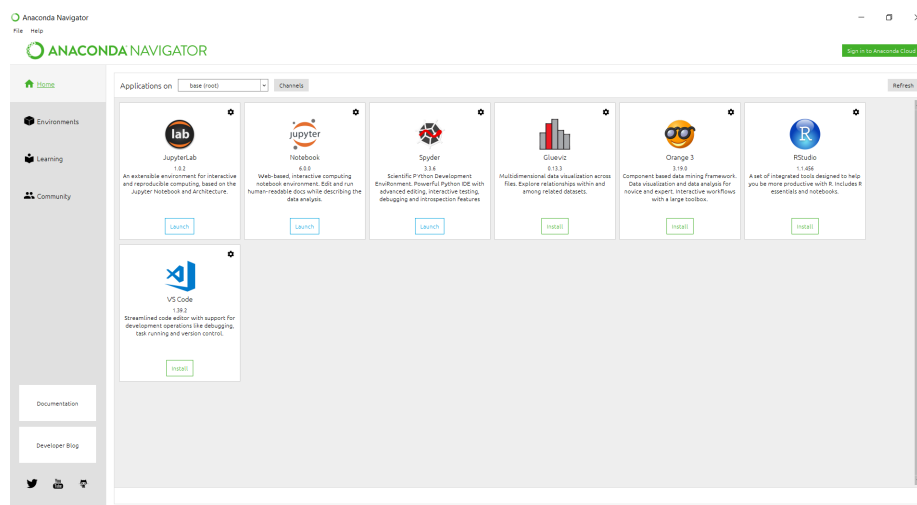


Figure 1.26: *CLI in Command Prompt*

2. ketikkan `print("Hello World")` dan run spyder



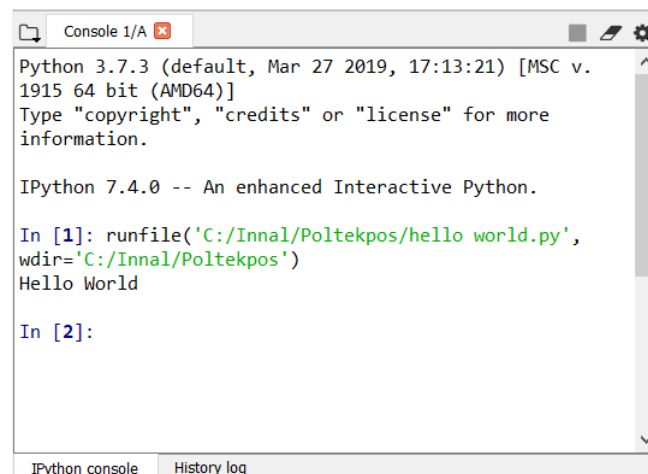
```

1 # -*- coding: utf-8 -*-
2 """
3 Created on Thu Oct 17 00:01:37 2019
4
5 @author: innal
6 """
7
8 print("Hello World")

```

Figure 1.27: *Print Hello World*

3. hasilnya akan seperti ini



The screenshot shows a console window titled 'Console 1/A'. It displays the output of a Python script. The first part shows the Python version (3.7.3) and the IPython version (7.4.0). Then, it shows the execution of the script 'C:/Innal/Poltekpos/hello world.py' with the output 'Hello World'. The console also shows the prompt 'In [2]:'.

```

Python 3.7.3 (default, Mar 27 2019, 17:13:21) [MSC v.
1915 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more
information.

IPython 7.4.0 -- An enhanced Interactive Python.

In [1]: runfile('C:/Innal/Poltekpos/hello world.py',
wdir='C:/Innal/Poltekpos')
Hello World

In [2]:

```

Figure 1.28: *Hello World*

## 1.2.7 Automatic Login SIAP

Buka Spyder lalu tuliskan script sebagai berikut.

```

from selenium.webdriver import Firefox
from selenium.webdriver.firefox.options import Options
from selenium.webdriver.common.desired_capabilities import DesiredCapabilities
from selenium.webdriver.firefox.firefox_binary import FirefoxBinary

print("Masukkan Npm Anda:")
npm = input()
print("Masukkan Password SIAP Anda:")
paswd = input('')

```

```

opsi = Options()

opsi.headless = False
binary = FirefoxBinary("C:\\Program Files\\Mozilla Firefox\\firefox.exe")
cap = DesiredCapabilities().FIREFOX
cap['marionette'] = True

browser=Firefox(executable_path='geckodriver.exe',
options=opsi,capabilities=cap,firefox_binary=binary)
browser.get('http://siap.poltekpos.ac.id/siap/besan.depan.php')

name = browser.find_element_by_name('user_name')
word = browser.find_element_by_name('user_pass')
login = browser.find_element_by_name('login')

name.send_keys(npm)
word.send_keys(paswd)
login.click()

```

Kemudian save program dengan nama WA.py dan run program

```

1 from selenium.webdriver import Firefox
2 from selenium.webdriver.firefox.options import Options
3 from selenium.webdriver.common.desired_capabilities import DesiredCapabilities
4 from selenium.webdriver.firefox.firefox_binary import FirefoxBinary
5
6 print("Masukkan Npm Anda:")
7 npm = input()
8 print("Masukkan Password SIAP Anda:")
9 paswd = input('')
10
11 opsi = Options()
12
13 opsi.headless = False
14 binary = FirefoxBinary("C:\\Program Files\\Mozilla Firefox\\firefox.exe")
15 cap = DesiredCapabilities().FIREFOX
16 cap['marionette'] = True
17
18 browser=Firefox(executable_path='geckodriver.exe',options=opsi,capabilities=cap,firefox_binary=binary)
19 browser.get('http://siap.poltekpos.ac.id/siap/besan.depan.php')
20
21 name = browser.find_element_by_name('user_name')
22 word = browser.find_element_by_name('user_pass')
23 login = browser.find_element_by_name('login')
24
25 name.send_keys(npm)
26 word.send_keys(paswd)
27 login.click()

```

Figure 1.29: *Automatic Login SIAP*

Setelah di running maka kita akan diminta menginputkan npm dan Password akun SIAP

```

IPython console
Console 2/A

Python 3.7.3 (default, Mar 27 2019, 17:13:21) [MSC v.1915 64 bit (AMD64)]
Type "copyright", "credits" or "license()" for more information.

IPython 7.4.0 -- An enhanced Interactive Python.

In [1]: runfile('C:/Innal/Poltekpos/WA.py', wdir='C:/Innal/Poltekpos')
Masukkan Npm Anda:

1184036
Masukkan Password SIAP Anda:

samsung12345

```

Figure 1.30: Hasil Running

Program selanjutnya akan membuka mozilla secara otomatis dan mengetikkan npm serta password yang telah diinputkan oleh user dan kemudian mengklik login.

SIAP - Politeknik Pos Indonesia

Politeknik POS INDONESIA Logistics Expert

Statistik Perwalan Semester Ganjil 2019/2020

Program Studi	Pengisian KRS						Perwalan							
	Hari Ini	Sebelumnya	Jumlah	Hari Ini	Sebelumnya	Jumlah								
D3 Teknik Informatika	0	0	97	792	110	97	792	0	0	92	731	110	92	731
D4 Teknik Informatika	0	0	272	1750	284	272	1750	0	0	270	1676	284	270	1676
D3 Manajemen Informatika	0	0	46	329	67	46	329	0	0	46	324	67	46	324
D3 Akuntansi	0	0	103	895	103	103	895	0	0	102	858	103	102	858
D4 Akuntansi Keuangan	0	0	186	1394	214	186	1394	0	0	184	1380	214	184	1380
D3 Manajemen Pemasaran	0	0	71	618	96	71	618	0	0	63	550	96	63	550
D4 Manajemen Perusahaan	0	0	267	1455	347	267	1455	0	0	263	1412	347	263	1412
D3 Logistik Bisnis	0	0	286	2499	352	286	2499	0	0	281	2441	352	281	2441
D4 Logistik Bisnis	0	0	633	4351	733	633	4351	0	0	604	4182	733	604	4182
Jumlah	0	0	1961	14083	2306	1961	14083	0	0	1905	13554	2306	1905	13554

Log Pengisian KRS dan Perwalan Semester Ganjil 2019/2020

No	NPM	Nama	Jml MK	Waktu	No	NPM	Nama	Jml MK	Waktu
1	6194163	Rachmad Hendrawan	10	16-10-2019 (18:48:21)	1	5193061	Alfan Rakahandy Afkri	10	16-10-2019 (22:09:48)
2	6174061	Bima Satra Rahmadhani	2	16-10-2019 (17:25:23)	2	6184212	Afrida Kristina Sinaga	9	16-10-2019 (19:33:18)
3	6164003	Al Prasetyo	8	16-10-2019 (17:13:22)	3	6164130	Hafiz Rasydy Sandra	8	16-10-2019 (15:52:44)
4	6194160	Novia Ratnasari	10	16-10-2019 (15:01:52)	4	6194160	Novia Ratnasari	10	16-10-2019 (15:52:20)
5	6164130	Hafiz Rasydy Sandra	8	16-10-2019 (14:11:02)	5	6184127	Rafi Pratama Andito	6	16-10-2019 (14:32:13)
6	5173137	Wira Cristian Anugrah Pardede	8	16-10-2019 (14:05:06)	6	6184064	NANCY MARIA MEDELNE	6	16-10-2019 (14:31:15)
7	5173131	Dina Wulandari	10	16-10-2019 (14:01:12)	7	5173131	Dina Wulandari	10	16-10-2019 (14:08:50)
8	5173137	Wira Cristian Anugrah Pardede	8	16-10-2019 (13:27:34)	8	5173137	Wira Cristian Anugrah Pardede	8	16-10-2019 (14:08:29)

Login

Pengguna:

Kata Sandi:

☐ Login

Figure 1.31: Automatic Input dan Login

Login selesai

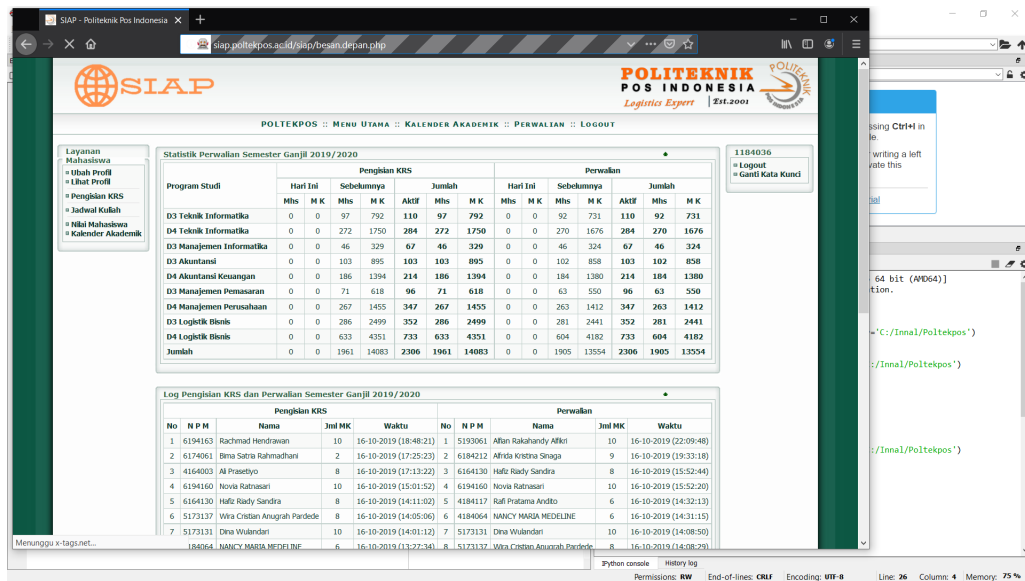


Figure 1.32: Automatic Input dan Login Selesai

## 1.2.8 Pemakaian Variable Explorer

Variable explorer akan secara otomatis terisi ketika kita membuat sebuah variable, pada variable explorer kita bisa melihat nama variable, tipe data, length, dan value dari variable yang kita buat tersebut.

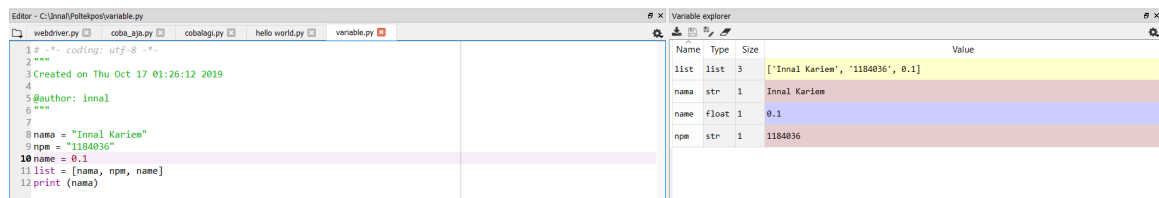


Figure 1.33: Variable Explorer

## 1.3 Indentasi

### 1.3.1 Penjelasan Indentasi

Indentasi, bagian code paragraf yang posisinya menjorok ke dalam pada baris-baris paragraf code. untuk mengatur indentasi dapat menggunakan tab atau spasi. Indentasi biasa digunakan oleh bahasa pemrograman python sebagai pengganti briket ( ) untuk membuka dan menutup fungsi. Error indentasi dapat terjadi apabila syntax tidak menggunakan tab atau space. Contoh yang benar (menggunakan tab/spasi sebagai indentasi):

```
# percabangan if
if username == 'crot':
    print("Selamat Datang ganteng")
    print("Silahkan ambil air wudhu")
```

```
# blok percabangan for
for i in range(10):
    print i
```

Contoh yang salah (tidak menggunakan tab/spasi):

```
# percabangan if
if username == 'crot':
print("Selamat Datang ganteng")
print("Silahkan ambil air wudhu")
```

```
# percabangan for
for i in range(10):
print i
```

### 1.3.2 Jenis-Jenis Error Indentasi

IndentationError: unexpected indent. Error diatas terjadi apabila syntax kekurangan tab atau spasi.

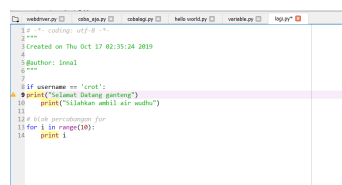
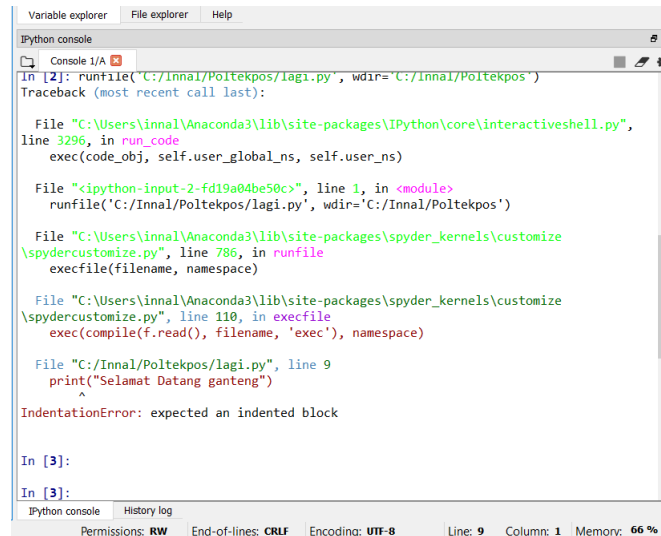


Figure 1.34: *Indentasi*

Apabila di running akan memunculkan error sebagai berikut.



```
Variable explorer | File explorer | Help
IPython console
Console 1/A
In [2]: runfile("C:/Innal/Poltekpos/lagi.py", wdir="C:/Innal/Poltekpos")
Traceback (most recent call last):

  File "C:\Users\innal\Anaconda3\lib\site-packages\IPython\core\interactiveshell.py",
line 3296, in run_code
    exec(code_obj, self.user_global_ns, self.user_ns)

  File "<ipython-input-2-fd19a04be50c>", line 1, in <module>
    runfile('C:/Innal/Poltekpos/lagi.py', wdir='C:/Innal/Poltekpos')

  File "C:\Users\innal\Anaconda3\lib\site-packages\spyder_kernels\customize
\spydercustomize.py", line 786, in runfile
    execfile(filename, namespace)

  File "C:\Users\innal\Anaconda3\lib\site-packages\spyder_kernels\customize
\spydercustomize.py", line 110, in execfile
    exec(compile(f.read(), filename, 'exec'), namespace)

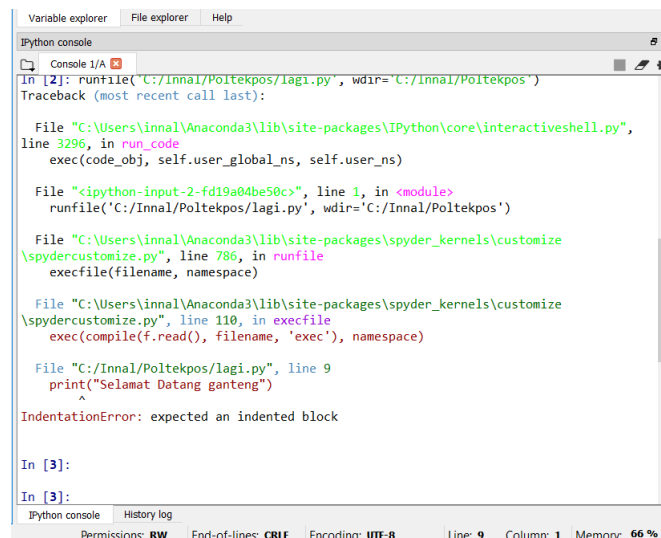
  File "C:/Innal/Poltekpos/lagi.py", line 9
    print("Selamat Datang ganteng")
    ^
IndentationError: expected an indented block

In [3]:
In [3]:
IPython console | History log
Permissions: RW | End-of-lines: CRLF | Encoding: UTF-8 | Line: 9 | Column: 1 | Memory: 66 %
```

Figure 1.35: *Error Indentasi*

### 1.3.3 Cara Membaca Error

Jika terjadi error maka cari di console line berapa yang error, pada gambar berikut terdapat error indentasi pada line 9.



```
Variable explorer | File explorer | Help
IPython console
Console 1/A
In [2]: runfile("C:/Innal/Poltekpos/lagi.py", wdir="C:/Innal/Poltekpos")
Traceback (most recent call last):

  File "C:\Users\innal\Anaconda3\lib\site-packages\IPython\core\interactiveshell.py",
line 3296, in run_code
    exec(code_obj, self.user_global_ns, self.user_ns)

  File "<ipython-input-2-fd19a04be50c>", line 1, in <module>
    runfile('C:/Innal/Poltekpos/lagi.py', wdir='C:/Innal/Poltekpos')

  File "C:\Users\innal\Anaconda3\lib\site-packages\spyder_kernels\customize
\spydercustomize.py", line 786, in runfile
    execfile(filename, namespace)

  File "C:\Users\innal\Anaconda3\lib\site-packages\spyder_kernels\customize
\spydercustomize.py", line 110, in execfile
    exec(compile(f.read(), filename, 'exec'), namespace)

  File "C:/Innal/Poltekpos/lagi.py", line 9
    print("Selamat Datang ganteng")
    ^
IndentationError: expected an indented block

In [3]:
In [3]:
IPython console | History log
Permissions: RW | End-of-lines: CRLF | Encoding: UTF-8 | Line: 9 | Column: 1 | Memory: 66 %
```

Figure 1.36: *Error*

### 1.3.4 Cara Menangani Error

Untuk Menangani error indentasi dapat dilakukan dengan cara menambahkan tab atau space pada line yang error. Berikut syntax yang error.

```

1 # -*- coding: utf-8 -*-
2 """
3 Created on Thu Oct 17 02:35:24 2019
4
5 @author: innal
6 """
7
8 if username == 'crot':
9 print("Selamat Datang ganteng")
10     print("Silahkan ambil air wudhu")
11
12 # blok percabangan for
13 for i in range(10):
14     print i

```

Figure 1.37: *Syntax Error*

Berikut syntax yang tidak error menggunakan tab/spasi (indentasi).

```

1 # -*- coding: utf-8 -*-
2 """
3 Created on Thu Oct 17 02:35:24 2019
4
5 @author: innal
6 """
7 username = input()
8
9 if username == 'crot':
10     print("Selamat Datang ganteng")
11     print("Silahkan ambil air wudhu")
12
13 for i in range(10):
14     print (i)

```

Name	Type	Size	Value
1	int	1	9
username	str	1	1

```

File "C:\Users\innal\AppData\Local\Temp\spyder_packages\spyder_kernels\customize\spydercustomize.py", line 110, in execFile
exec(compile(f.read(), filename, 'exec'), namespace)

File "C:/Innal/Poltekpos/lagi.py", line 8, in <module>
if username == 'crot':

NameError: name 'username' is not defined

In [4]:
In [4]: runfile('C:/Innal/Poltekpos/lagi.py', wdir='C:/Innal/Poltekpos')

1
0
1
2
3
4
5
6
7
8
9

In [5]:

```

Figure 1.38: *Syntax yang Telah Diperbaiki*