



Modul Praktikum **Kecerdasan Buatan**



Instalasi dan Setup Environment

Environment

Environment adalah sistem komputer yang digunakan untuk membangun dan mengembangkan suatu perangkat lunak. Environment yang dimaksud untuk sebagian aplikasi mencakup sistem operasi, sistem basis data, compiler, software, modul, dan lain sebagainya. Dalam praktikum ini kita akan menggunakan Bahasa Pemrograman Python.

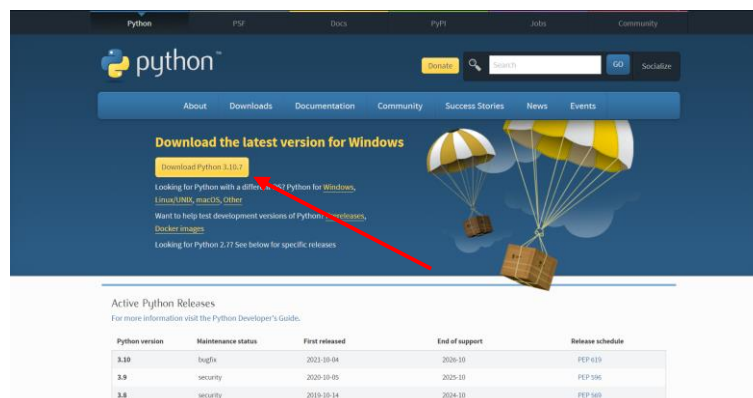
Tools dan IDEs

Dalam pengerjaan proyek AI dibutuhkan IDE dan tools untuk menunjang pengerjaan proyek. Terdapat banyak IDEs dan tools yang tersedia, berikut merupakan IDEs dan tools yang sering digunakan dalam proyek machine learning [Anaconda](#), [Spyder](#), [Python](#), [VS Code](#), [PyCharm](#), [Atom](#), [Jupyter Notebook](#), [Google Colabs](#) dan masih banyak lagi. Dalam praktikum ini kita akan menggunakan PyCharm dan Google Colabs.

Python

Python dapat diunduh di link [ini](#). Silakan pilih Python versi terbaru. Pada saat modul ini dibuat versi terbarunya adalah Python versi 3.10.7. Unduh sesuai operating sistem yang digunakan. Berikut merupakan Langkah-langkah mengunduh Python.

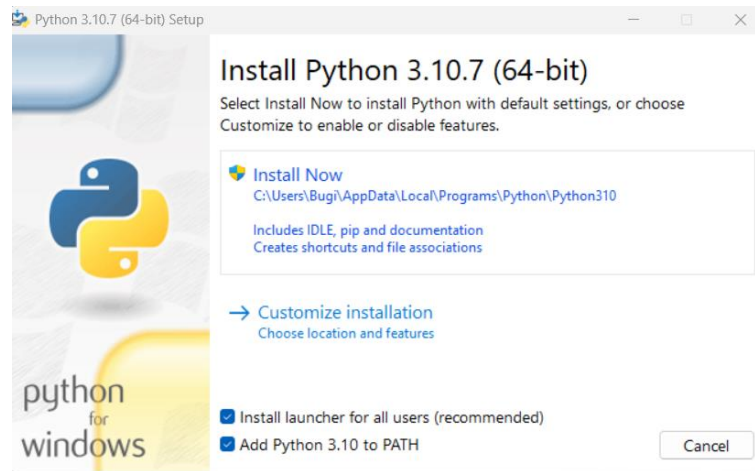
1. Tekan tombol **Download** berwarna kuning



2. Kemudian buka file **python-3.10.7-amd64.exe**



- Centang **"Add Python 3.10 to PATH"** dan tekan tombol **Install Now**



Silakan atur agar pengaturan sama dengan gambar.

- Jika sudah selesai, cek apakah Python sudah terinstal dengan menjalankan perintah `python --version` pada cmd.

```
Command Prompt
Microsoft Windows [Version 10.0.22622.590]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Bugi>python --version
Python 3.10.7

C:\Users\Bugi>
```

Jika keluar tampilan seperti di atas maka instalasi telah berhasil.



PIP

pip adalah package installer untuk Python. pip dapat digunakan untuk menginstal packages dari Python Package Index dan index lain. Ketika menginstal Python maka pip akan secara otomatis terinstal. Secara default pip akan menyimpan package pada `C:\Users\user\AppData\Local\Programs\Python\Python310\Lib\site-packages`. Berikut merupakan penggunaan pip.

1. Cek versi >> `py -m pip --version`

```
C:\Users\Bugi>py -m pip --version
pip 22.2.2 from C:\Users\Bugi\AppData\Local\Programs\Python\Python310\lib\site-packages\pip (python 3.10)
```

2. Cek package terinstal >> `pip list`

```
C:\Users\Bugi>pip list
Package      Version
-----
pip          22.2.2
setuptools   63.2.0
```

3. Install package >> `pip install [package name]`

```
C:\Users\Bugi>pip install numpy
Collecting numpy
  Downloading numpy-1.23.3-cp310-cp310-win_amd64.whl (14.6 MB)
----- 14.6/14.6 MB 380.2 kB/s eta 0:00:00
Installing collected packages: numpy
Successfully installed numpy-1.23.3
```

4. Install package versi spesifik >> `pip install [package name]==[version]`

```
C:\Users\Bugi>pip install numpy==1.22.1
WARNING: Retrying (Retry(total=4, connect=None, read=None, redirect=None, status=None)) after connection broken by 'ReadTimeoutError("HTTPSConnectionPool(host='pypi.org', port=443): Read timed out. (read timeout=15)")': /simple/numpy/
Collecting numpy==1.22.1
  Downloading numpy-1.22.1-cp310-cp310-win_amd64.whl (14.7 MB)
----- 14.7/14.7 MB 933.3 kB/s eta 0:00:00
Installing collected packages: numpy
Successfully installed numpy-1.22.1
```

5. Upgrade versi package >> `pip install [package name] --upgrade`

```
C:\Users\Bugi>pip install numpy --upgrade
Requirement already satisfied: numpy in c:\users\bugi\appdata\local\programs\python\python310\lib\site-packages (1.22.1)
Collecting numpy
  Using cached numpy-1.23.3-cp310-cp310-win_amd64.whl (14.6 MB)
Installing collected packages: numpy
  Attempting uninstall: numpy
    Found existing installation: numpy 1.22.1
    Uninstalling numpy-1.22.1:
      Successfully uninstalled numpy-1.22.1
  Successfully installed numpy-1.23.3
```

6. Uninstall package >> `pip uninstall [package name]`

```
C:\Users\Bugi>pip uninstall numpy
Found existing installation: numpy 1.23.3
Uninstalling numpy-1.23.3:
  Would remove:
    c:\users\bugi\appdata\local\programs\python\python310\lib\site-packages\numpy-1.23.3.dist-info\*
    c:\users\bugi\appdata\local\programs\python\python310\lib\site-packages\numpy\*
    c:\users\bugi\appdata\local\programs\python\python310\scripts\f2py.exe
Proceed (Y/n)? y
  Successfully uninstalled numpy-1.23.3
```



7. Cek lokasi package >> `pip show [package name]`

```
C:\Users\Bugi>pip show numpy
Name: numpy
Version: 1.23.3
Summary: NumPy is the fundamental package for array computing with Python.
Home-page: https://www.numpy.org
Author: Travis E. Oliphant et al.
Author-email:
License: BSD
Location: c:\users\bugi\appdata\local\programs\python\python310\lib\site-packages
Requires:
Required-by:
```

JupyterLab

Pada praktikum ini kita akan menggunakan JupyterLab/Jupyter Notebook agar tipe file .ipynb (Interactive Python Notebook) dapat digunakan. Jupyter dapat diunduh melalui **conda (Anaconda)**, **mamba**, **pip**, **pipenv**, atau **docker**. Tetapi untuk tutorial kita akan menggunakan **pip** karena sudah melakukan penginstalan Python seperti di atas.

JupyterLab vs Jupyter Notebook

Perbedaan antara JupyterLab dan Jupyter Notebook dapat dilihat dari penggunaannya. Tampilan di bawah ini merupakan tampilan Jupyter Notebook yang sangat ringan menggunakan web browser saja, tanpa penggunaan lain selain melihat file .ipynb.

The screenshot shows a Jupyter Notebook interface with a title bar 'jupyter tutorial Last Checkpoint: 3 minutes ago (autosaved)' and a 'Logout' button. The notebook content includes a title 'PyCon 2018: Using pandas for Better (and Worse) Data Science', a GitHub link, and three code cells. The first cell imports matplotlib and pandas, and the second cell reads a CSV file. The third cell prints the head of the DataFrame. The output of the third cell is a table with 4 rows and 10 columns.

```
In [1]: import matplotlib.pyplot as plt
import pandas as pd
pd.__version__

Out[1]: '0.24.1'
```

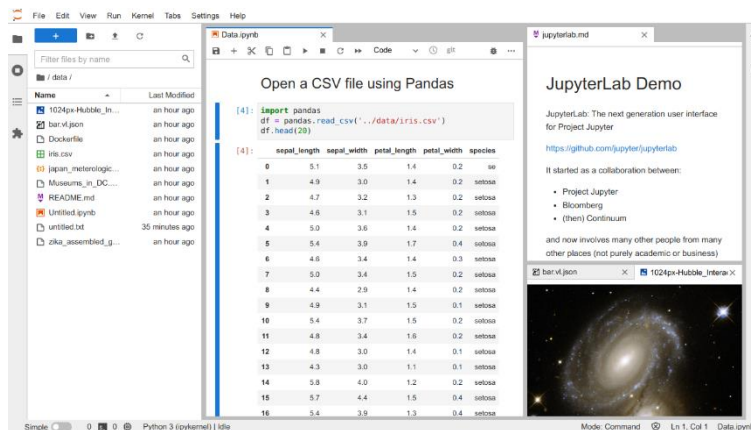
Dataset: Stanford Open Policing Project ([video](#))

```
In [2]: # ri stands for Rhode Island
ri = pd.read_csv('police.csv')

In [3]: # what does each row represent?
ri.head()
```

```
Out[3]:
```

	stop_date	stop_time	county_name	driver_gender	driver_age_raw	driver_age	driver_race	violation_raw	violation	search...
0	2005-01-02	01:55	NaN	M	1985.0	20.0	White	Speeding	Speeding	
1	2005-01-18	08:15	NaN	M	1965.0	40.0	White	Speeding	Speeding	
2	2005-01-23	23:15	NaN	M	1972.0	33.0	White	Speeding	Speeding	
3	2005-02-20	17:15	NaN	M	1986.0	19.0	White	Call for Service	Other	



JupyterLab dapat memberikan fasilitas penggunaan tab seperti screenshot di atas, seperti IDE (text editor), dapat menampilkan format file lain (gambar, json, csv, markdown, pdf, svg, etc) hingga dapat melakukan real-time collaboration menggunakan framework Yjs (tutorial dapat dicek pada docs [disini](#)).

Instalasi JupyterLab

Pada CMD, (Win + R, ketik "cmd", Enter): `pip install jupyterlab`

```
Command Prompt
Microsoft Windows [Version 10.0.19044.1889]
(c) Microsoft Corporation. All rights reserved.

C:\Users\super>pip install jupyterlab
Collecting jupyterlab
  Downloading jupyterlab-3.4.6-py3-none-any.whl (8.8 MB)
----- 8.8/8.8 MB 4.7 MB/s eta 0:00:00
Collecting notebook<7
  Downloading notebook-6.4.12-py3-none-any.whl (9.9 MB)
----- 9.9/9.9 MB 4.8 MB/s eta 0:00:00
Collecting jupyter-core
  Downloading jupyter_core-4.11.1-py3-none-any.whl (88 kB)
----- 88.4/88.4 kB 5.2 MB/s eta 0:00:00
Collecting tornado>=6.1.0
  Downloading tornado-6.2-cp37-abi3-win_amd64.whl (425 kB)
----- 425.3/425.3 kB 6.7 MB/s eta 0:00:00
Collecting ipython
  Downloading ipython-8.5.0-py3-none-any.whl (752 kB)
----- 752.0/752.0 kB 6.7 MB/s eta 0:00:00
Collecting jupyterlab-server<=2.10
  Downloading jupyterlab_server-2.15.1-py3-none-any.whl (54 kB)
----- 54.1/54.1 kB 2.7 MB/s eta 0:00:00
Collecting jupyter-server<=1.16
  Downloading jupyter_server-1.18.1-py3-none-any.whl (344 kB)
----- 344.9/344.9 kB 10.5 MB/s eta 0:00:00
Collecting packaging
  Downloading packaging-21.3-py3-none-any.whl (40 kB)
----- 40.8/40.8 kB ? eta 0:00:00
Collecting Jinja2>=2.1
  Downloading Jinja2-3.1.2-py3-none-any.whl (133 kB)
```

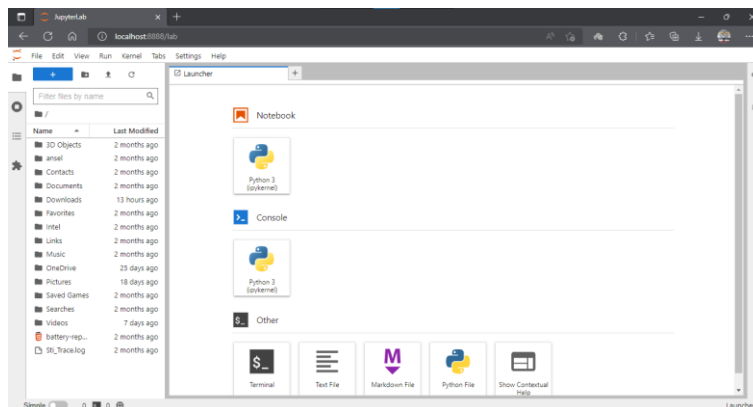


Untuk menjalankannya, pada CMD gunakan: `jupyter-lab`

```
Command Prompt - jupyter-lab
C:\Users\super>jupyter-lab
[I 2022-09-12 09:58:45.837 ServerApp] jupyterlab | extension was successfully linked.
[I 2022-09-12 09:58:45.853 ServerApp] nbclassic | extension was successfully linked.
[I 2022-09-12 09:58:45.869 ServerApp] Writing Jupyter server cookie secret to C:\Users\super\AppData\Roaming\jupyter\runt
time\jupyter_cookie_secret
[I 2022-09-12 09:58:46.494 ServerApp] notebook_shim | extension was successfully linked.
[I 2022-09-12 09:58:46.540 ServerApp] notebook_shim | extension was successfully loaded.
[I 2022-09-12 09:58:46.540 LabApp] JupyterLab extension loaded from C:\Users\super\AppData\Local\Programs\Python\Python3
10\lib\site-packages\jupyterlab
[I 2022-09-12 09:58:46.540 LabApp] JupyterLab application directory is C:\Users\super\AppData\Local\Programs\Python\Pyth
on310\share\jupyterlab
[I 2022-09-12 09:58:46.540 ServerApp] jupyterlab | extension was successfully loaded.
[I 2022-09-12 09:58:46.556 ServerApp] nbclassic | extension was successfully loaded.
[I 2022-09-12 09:58:46.556 ServerApp] Serving notebooks from local directory: C:\Users\super
[I 2022-09-12 09:58:46.556 ServerApp] Jupyter Server 1.18.1 is running at:
[I 2022-09-12 09:58:46.556 ServerApp] http://localhost:8888/lab?token=8fbb5612e5f7bf0e4607a43caf040367d7f60e856eece9a
[I 2022-09-12 09:58:46.556 ServerApp] or http://127.0.0.1:8888/lab?token=8fbb5612e5f7bf0e4607a43caf040367d7f60e856eece
9a
[I 2022-09-12 09:58:46.556 ServerApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirm
ation).
[C 2022-09-12 09:58:46.603 ServerApp]

To access the server, open this file in a browser:
file:///C:/Users/super/AppData/Roaming/jupyter/runtime/jpservice-7772-open.html
Or copy and paste one of these URLs:
http://localhost:8888/lab?token=8fbb5612e5f7bf0e4607a43caf040367d7f60e856eece9a
or http://127.0.0.1:8888/lab?token=8fbb5612e5f7bf0e4607a43caf040367d7f60e856eece9a
```

Maka akan otomatis tampil web-browser seperti berikut.



Instalasi Jupyter Notebook

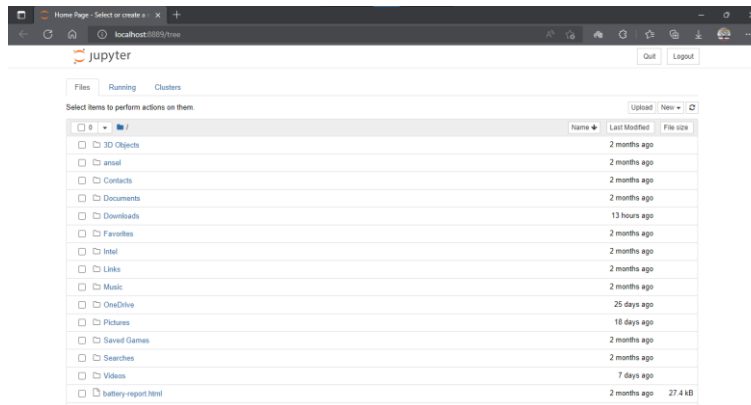
Pada CMD, (Win + R, ketik "cmd", Enter): `pip install notebook`

Untuk menjalankannya, pada CMD gunakan: `jupyter notebook`

NOTE: Jika telah menginstall JupyterLab, maka sudah otomatis dapat menggunakan command `jupyter notebook` tanpa instalasi terlebih dahulu.



Tampilan Jupyter Notebook

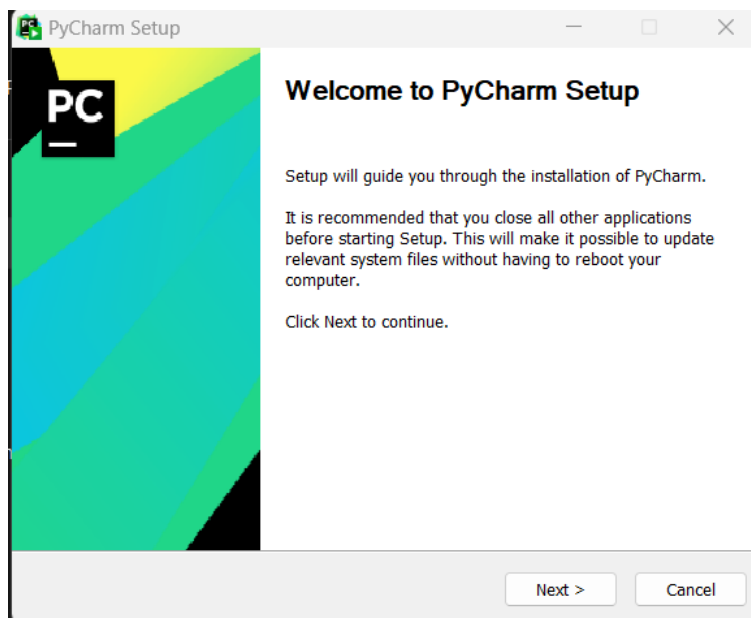


Pycharm

Pada praktikum ini kita akan menggunakan Pycharm Professional agar tipe file .ipynb dapat digunakan. Pycharm Community versi 2022.2.1 diunduh pada link [ini](#).

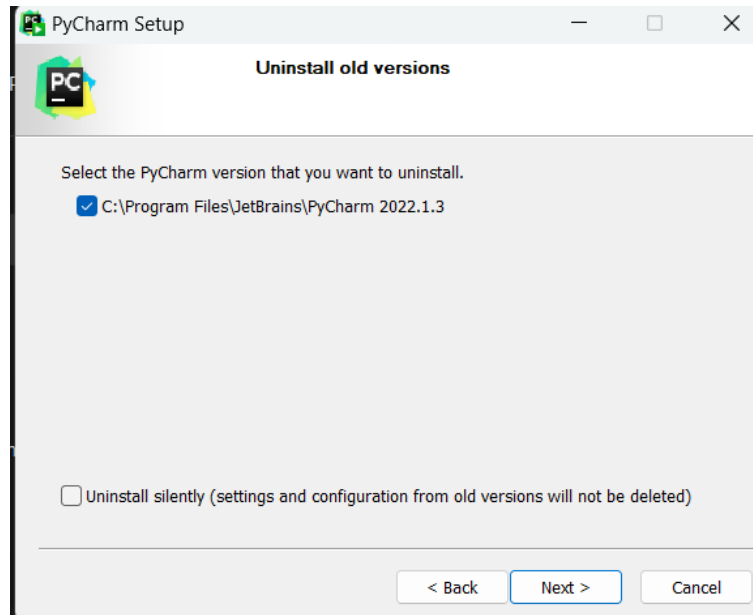
Instalasi:

1. Setelah mengunduh file, buka file **pycharm-professional-2022.1.3.exe**
2. Tekan tombol **next**





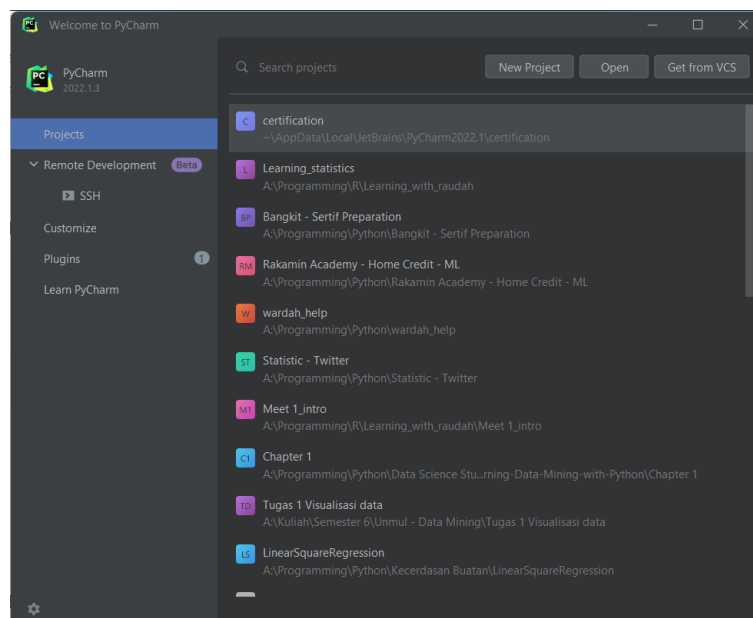
- Centang cek list box seperti gambar di bawah dan tekan tombol **next**



- Lanjutkan sesuai instruksi sampai selesai

Penggunaan:

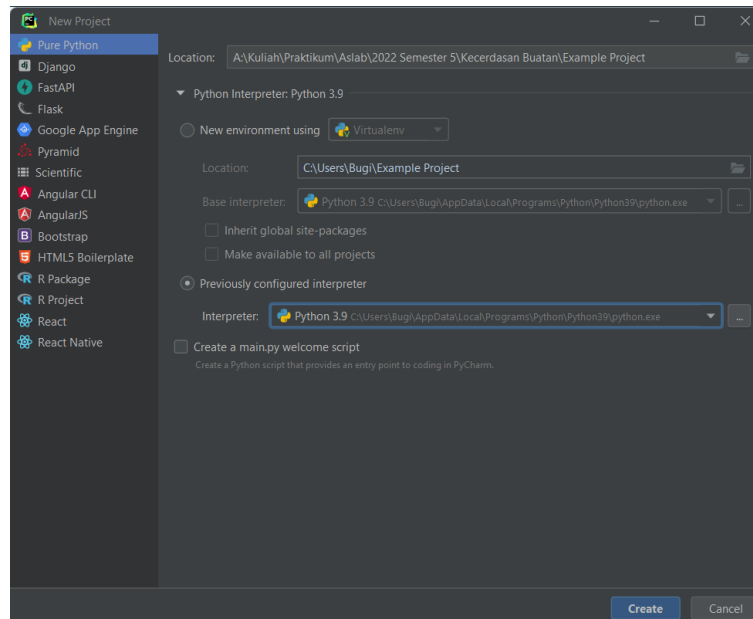
- Setelah instalasi selesai buka Pycharm
- Tekan **New Project**



Tampilan Project kalian kosong karena belum pernah membuka Project apa pun.

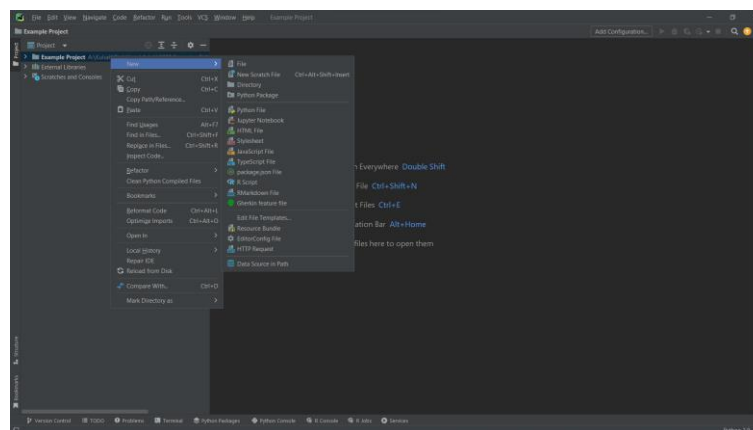


3. Pilih lokasi tempat Project ingin disimpan pada bagian **Location**.



Jika tampilan kalian tidak seperti gambar di atas, maka tekan tulisan Python Interpreter di bawah Location. Kemudian pilih **Previously configured interpreter**. Setelah itu tekan tombol **Create**.

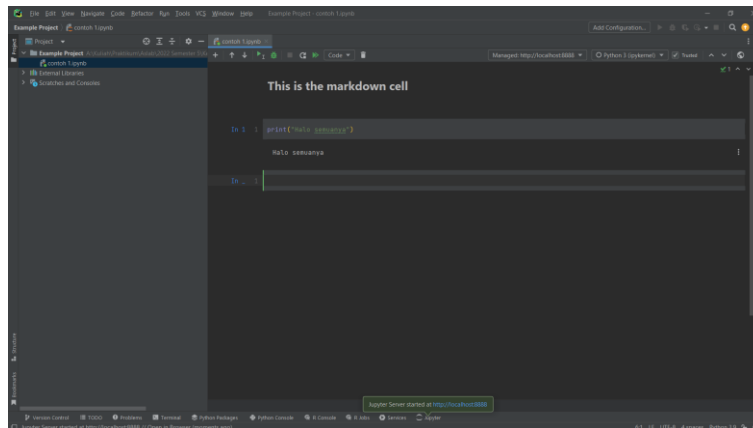
4. Buat file baru





Di bagian sebelah kiri terdapat folder sesuai dengan nama yang telah ditentukan sebelumnya. Klik kanan pada folder tersebut > **new > Jupyter Notebook/Python File**. Setelah itu beri nama file dan tekan **Enter**.

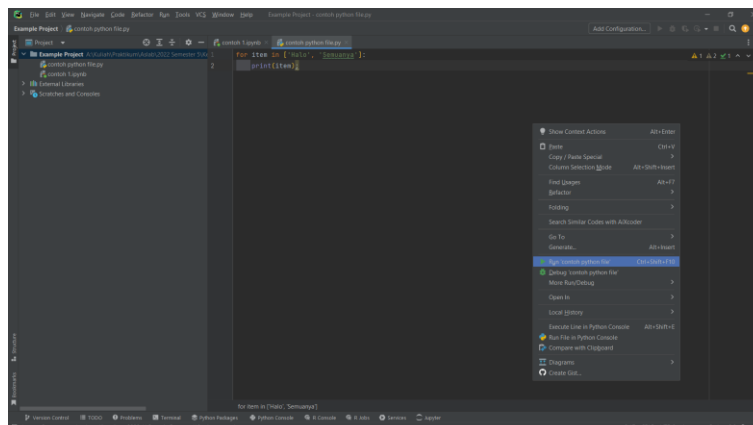



5. Run file .ipynb

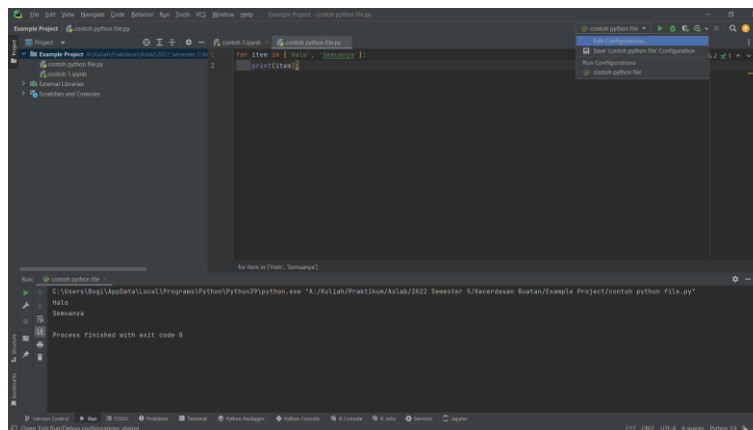


Tekan tombol  untuk menjalankan satu cell. Tekan tombol  untuk menjalankan semua cell dalam file. Atau dapat menggunakan shortcut **Ctrl + Enter** untuk run 1 cell dan **Shift + Enter** untuk run 1 cell dan lanjut ke cell selanjutnya.

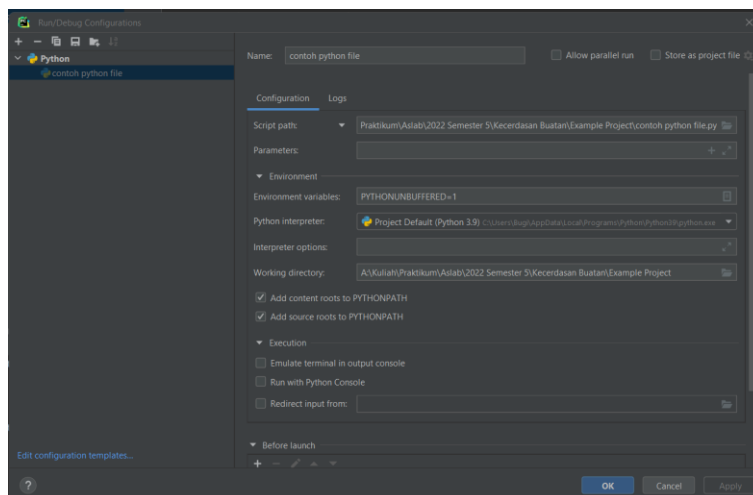
6. Run file .py



Klik kanan pada script/file .py dan tekan tombol  **Run 'contoh python file'** atau menggunakan shortcut **Ctrl + Shift + F10**.



Setelah run file kita dapat mengatur file yang akan di run pada bagian configuration di atas kanan aplikasi seperti gambar di atas kemudian pilih Edit Configurasi.



Gambar di atas adalah tampilan konfigurasi. Bagian yang perlu diperhatikan adalah Script path, Python Interpreter, Working directory, dan Execution.

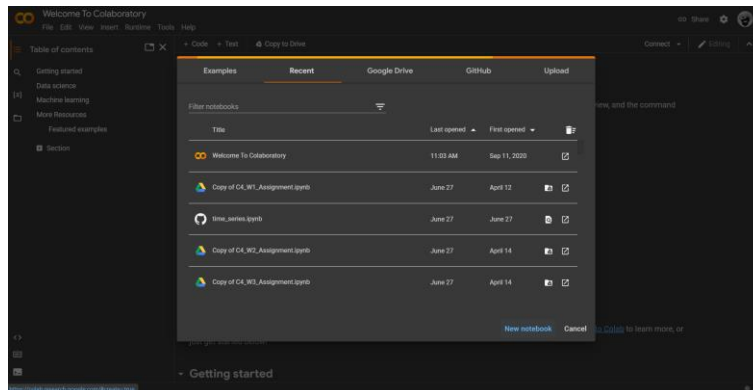
- Script path >> script file yang dipilih
- Python Interpreter >> interpreter/environment yang dipilih
- Working directory >> tempat project yang dibuka (berguna untuk pembuatan path ketika ingin membuka file lain seperti dataset)
- Execution >> tampilan script akan ditampilkan



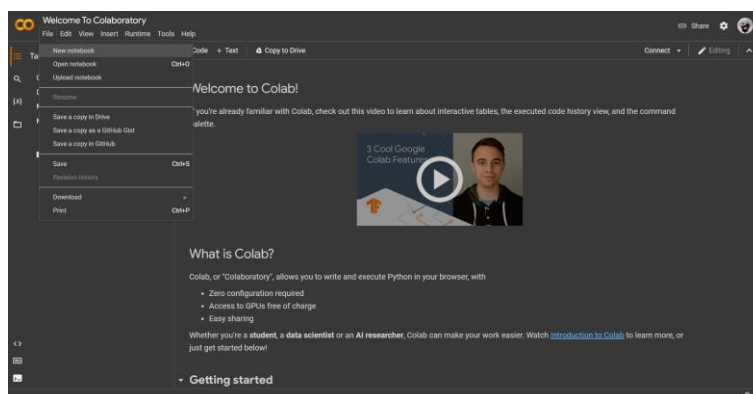
Google Colab

Colab merupakan IDE online dari Google. Colab memberikan akses GPU pada 1 akun dalam satu waktu. Colab digunakan sebagai alternative bagi developer dengan hardware kurang mumpuni. Colab dapat terintegrasi dengan Google Drive dan GitHub. Colab dapat diakses pada link [ini](https://colab.research.google.com/). Berikut merupakan penggunaan Google Colab.

1. Membuat notebook



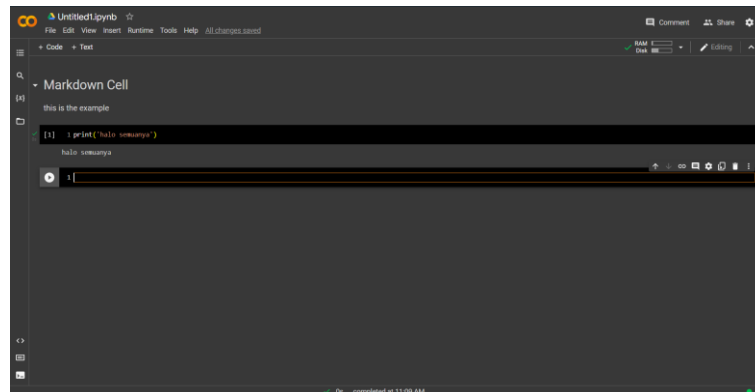
Tekan tombol **New notebook** untuk membuat file baru atau




Pada bagian kiri atas buka menu **file > New notebook**

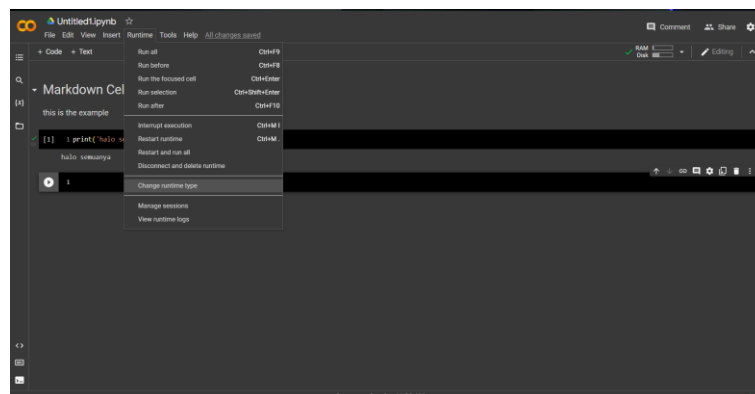


2. Run program

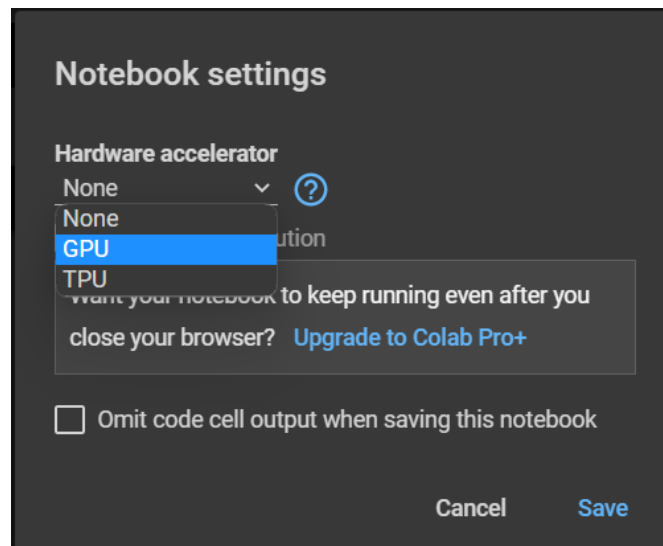


Run program dengan menekan tombol  pada sisi kiri cell. Atau dapat menggunakan shortcut **Ctrl + Enter** untuk run 1 cell dan **Shift + Enter** untuk run 1 cell dan lanjut ke cell selanjutnya.

3. Mengatur Mesin yang digunakan

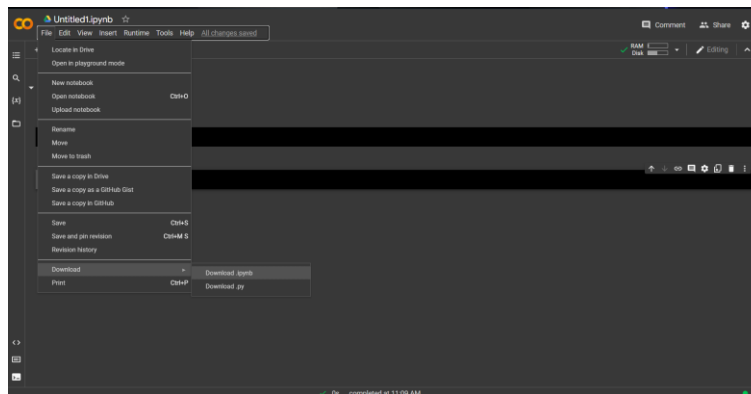


Pada sisi atas kiri pilih menu **Runtime > Change runtime type**



Kemudian pilih tipe mesin yang ingin digunakan

4. Download file



Pada bagian kiri atas pilih menu **file > Download**. Pilih format file yang ingin kalian download.



Virtual Environment

Virtual environment adalah Python environment seperti Python interpreter, libraries dan scripts yang terinstal secara terisolasi. Secara default adalah semua libraries terinstal pada "sistem" Python. Praktikum ini akan mengatur virtual environment menggunakan Pyscharm/Jupyter.

Jupyter-Lab / Jupyter Notebook VENV

Berikut merupakan cara mengatur virtual environment dengan Jupyter.

1. Tentukan lokasi penempatan Virtual Environment menggunakan command `cd` (change directory) pada cmd:

Contoh:

C:\Users\super> `mkdir venv_jupyterlab` (untuk membuat direktori baru)

Kemudian ketik: `python -m venv <nama virtual environment>`

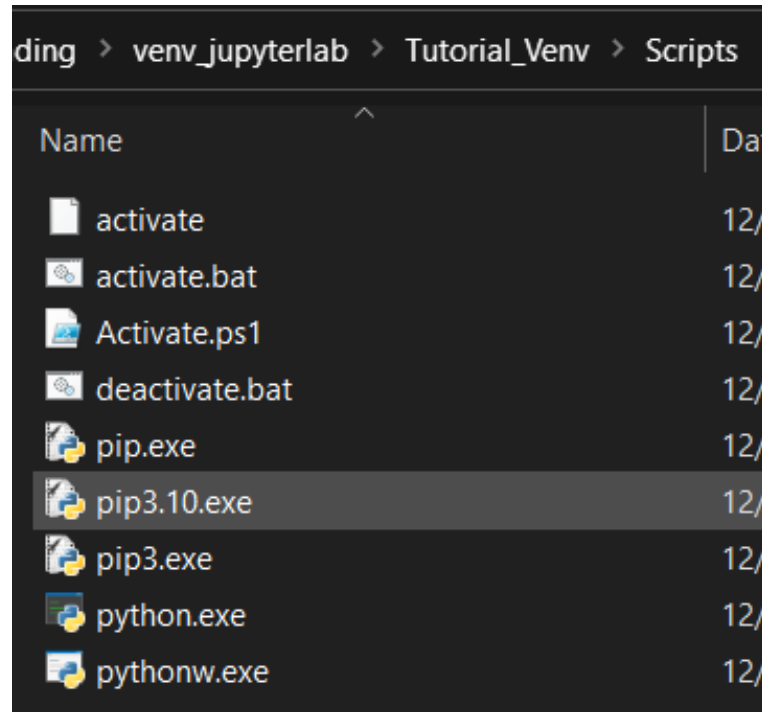
```
C:\Users\super>cd venv_jupyterlab
C:\Users\super\venv_jupyterlab>python -m venv Tutorial_Venv
C:\Users\super\venv_jupyterlab>
```

Setelah itu akan loading beberapa saat, kemudian selesai. Dapat di cek pada direktori yang bersangkutan, terdapat beberapa file seperti berikut:

ding > venv_jupyterlab > Tutorial_Venv	
Name	Date modified
Include	12/09/2022 10:27
Lib	12/09/2022 10:27
Scripts	12/09/2022 10:27
pyvenv.cfg	12/09/2022 10:27



2. Aktivasi file `activate.bat` pada cmd dengan cara:
`Tutorial_Venv\Scripts\activate`



Maka akan muncul (Tutorial_Venv) seperti berikut:

```
C:\Users\super\venv_jupyterlab>Tutorial_Venv\Scripts\activate
(Tutorial_Venv) C:\Users\super\venv_jupyterlab>
```

3. Untuk pengunduhan package pada pip saat Virtual Environment aktif, gunakan ipykernel sebagai tempat sementara untuk diinstallkan ke venv yang ingin dipakai. Penggunaannya sebagai berikut:

`pip install ipykernel <nama package>`

Contoh: `pip install ipykernel pandas`

Dapat digunakan berkali-kali sebelum dimasukkan ke venv.

```
(Tutorial_Venv) C:\Users\super\venv_jupyterlab>pip install ipykernel pandas
Collecting ipykernel
  Using cached ipykernel-6.15.2-py3-none-any.whl (132 kB)
Collecting pandas
  Using cached pandas-1.4.4-cp310-cp310-win_amd64.whl (10.0 MB)
Collecting ipython>=7.23.1
  Using cached ipython-8.5.0-py3-none-any.whl (752 kB)
Collecting jupyter-client>=6.1.12
  Using cached jupyter_client-7.3.5-py3-none-any.whl (132 kB)
Collecting debugpy>=1.0
  Using cached debugpy-1.6.3-cp310-cp310-win_amd64.whl (4.6 MB)
Collecting matplotlib-inline>=0.1
  Using cached matplotlib_inline-0.1.6-py3-none-any.whl (9.4 kB)
Collecting nest-asyncio
  Using cached nest_asyncio-1.5.5-py3-none-any.whl (5.2 kB)
```



4. Kemudian install pada venv dengan cara:

```
python -m ipykernel install --name=<nama venv>
```

Contoh: `python -m ipykernel install --name=Tutorial_Venv`

```
(Tutorial_Venv) C:\Users\super\venv_jupyterlab>python -m ipykernel install --name=Tutorial_Venv  
Installed kernelspec Tutorial_Venv in C:\ProgramData\jupyter\kernels\tutorial_venv
```

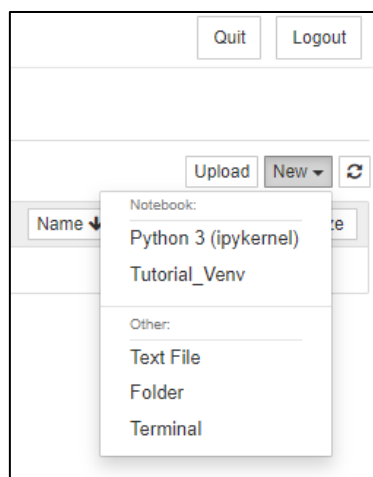
5. Deactivate sebelum memulai JupyterLab / Jupyter Notebook kembali.

```
(Tutorial_Venv) C:\Users\super\venv_jupyterlab>deactivate  
C:\Users\super\venv_jupyterlab>jupyter notebook
```

6. Dapat di cek pada bagian New yaitu terdapat Virtual Environment yang telah kita buat sebelumnya yaitu Tutorial_Venv.

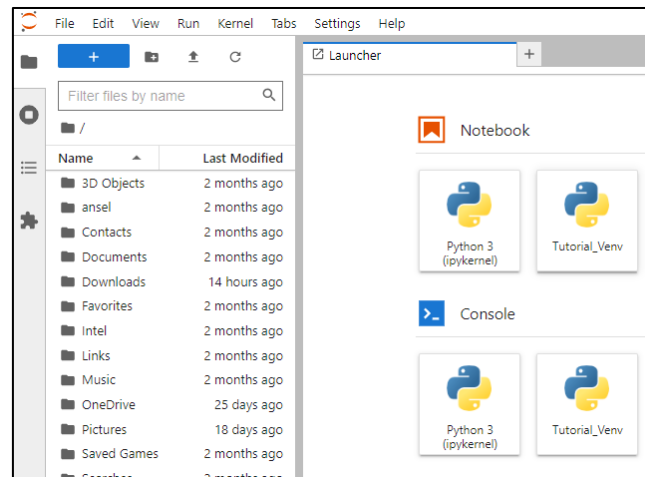
Kiri:

Jupyter Notebook



Kanan:

Jupyter-Lab





7. Untuk mengecek isi package yang telah kita install, tekan Virtual Environment yang telah dibuat, dan dapat gunakan code berikut:

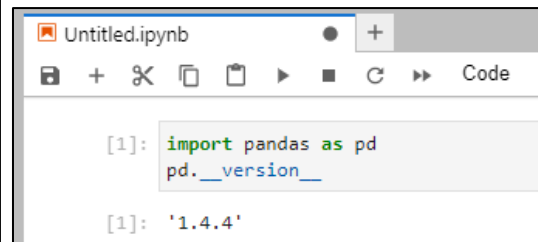
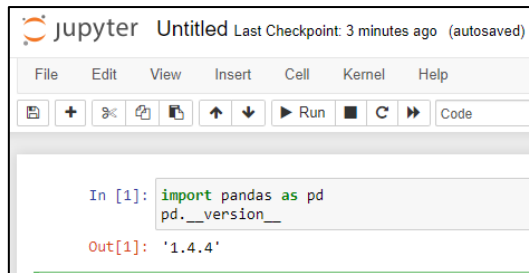
```
import pandas as pd  
pd.__version__
```

Kiri:

Jupyter Notebook

Kanan:

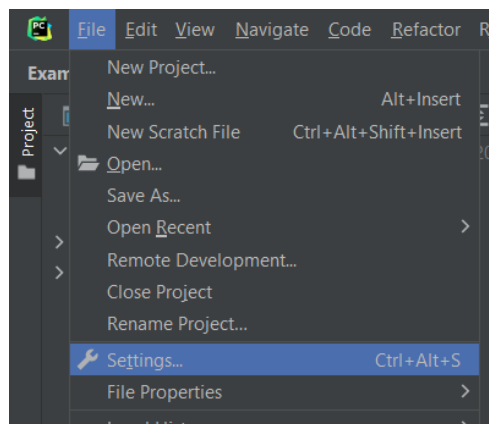
Jupyter-Lab



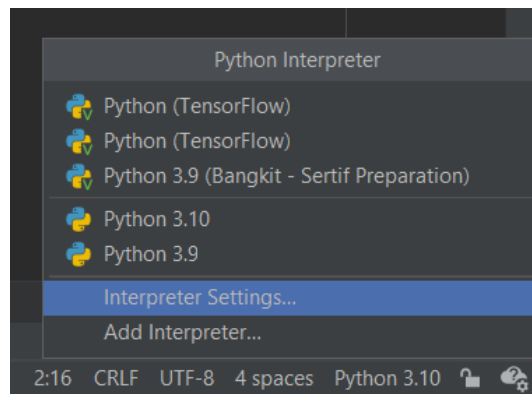
PyCharm VENV

Berikut merupakan cara mengatur virtual environment dengan Pycharm.

1. Membuka Pengaturan Interpreter

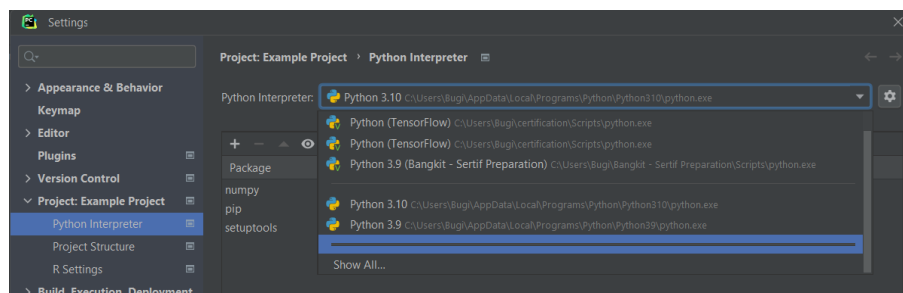


Pada bagian kiri atas buka menu **file > Setting**. atau

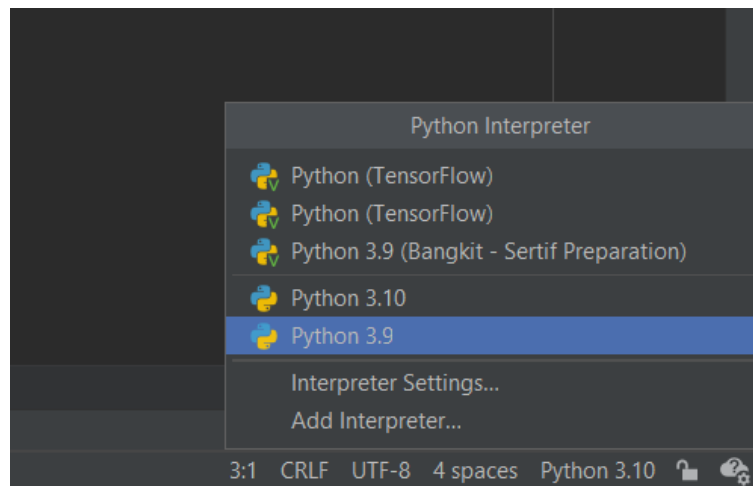


Pada bagian kanan bawah pilih nama interpreter yang digunakan

2. Memilih Interpreter



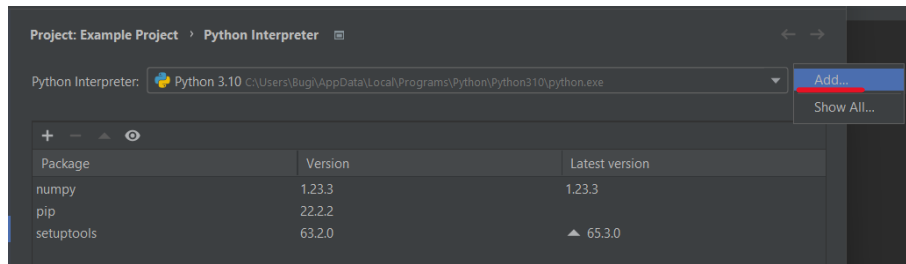
Pada window setting Pycharm pilih **Project > Python Interpreter**. Kemudian pilih interpreter yang ada pada bagian atas di Python Interpreter. atau




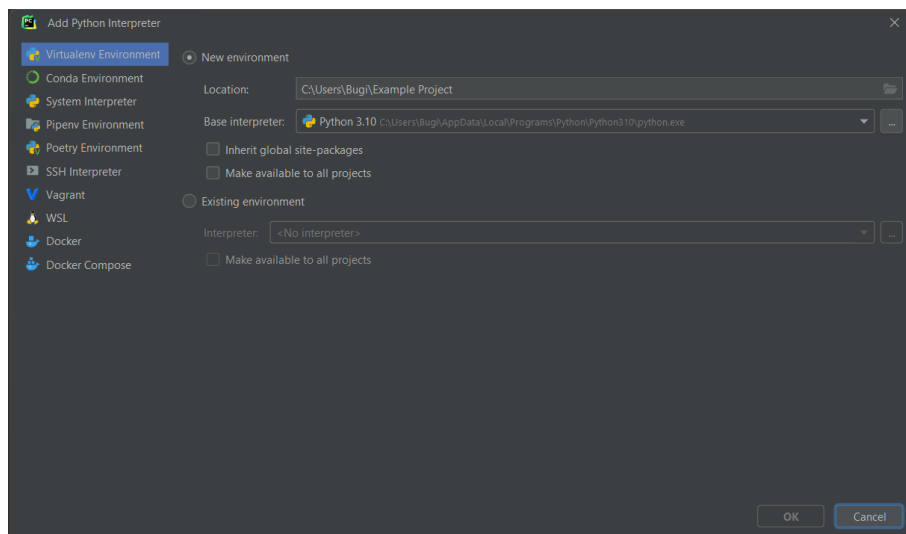
Pada bagian kanan bawah Pycharm pilih langsung Interpreter.



3. Menambahkan Interpreter

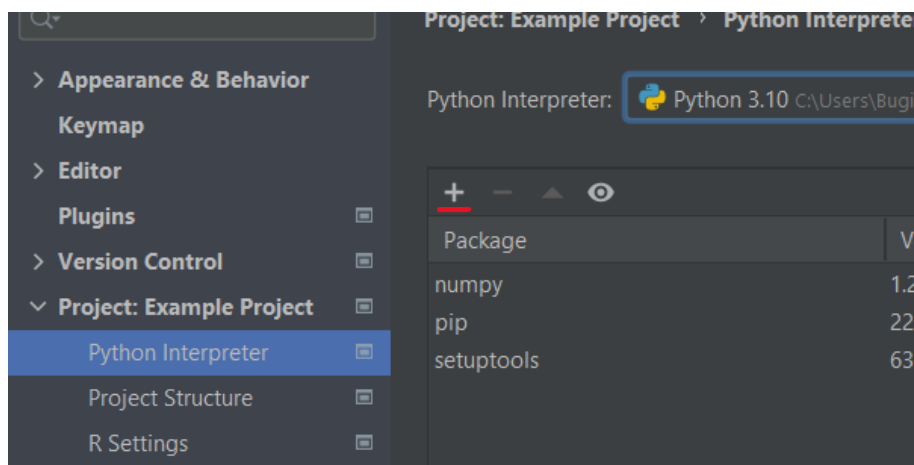


Pada bagian atas kanan tekan tombol  > add

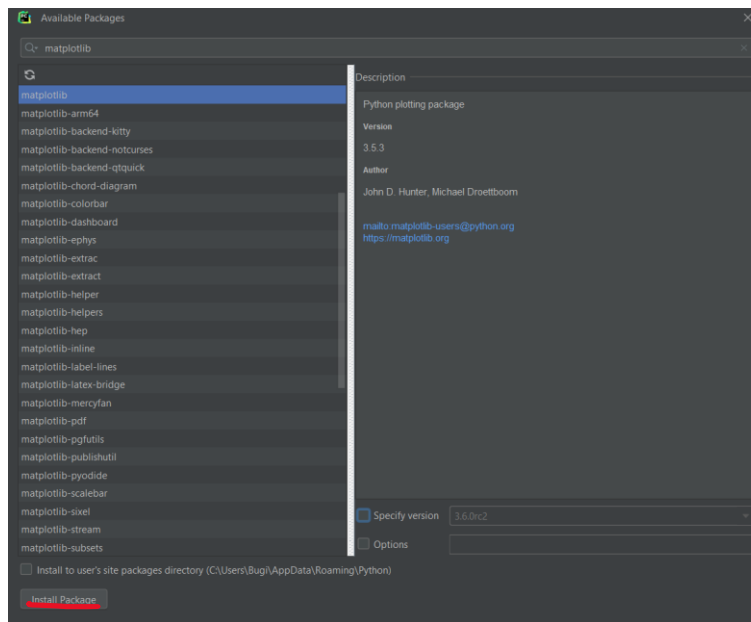


Setelah tampilan seperti gambar di atas, pilih jenis environment yang ingin dibuat. Untuk Virtualenv tentukan lokasi environment akan disimpan.

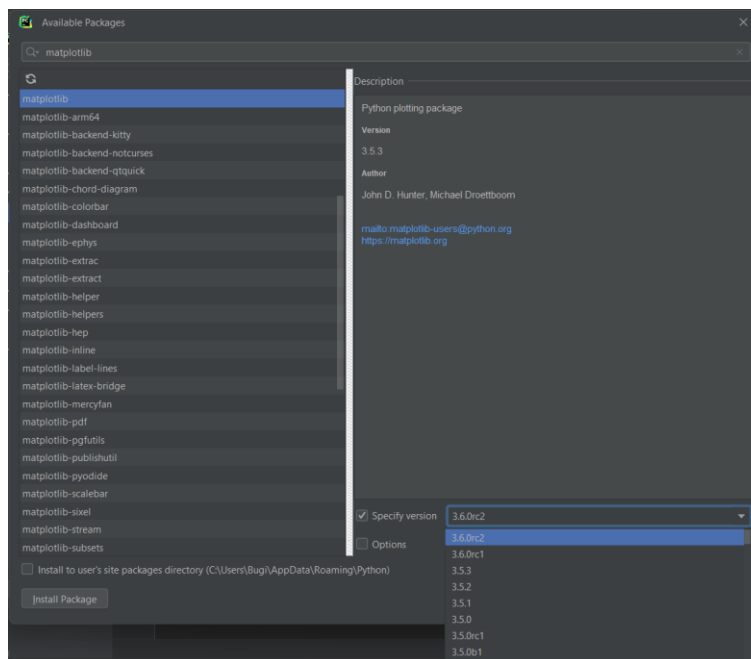
4. Menambahkan Package



Pada bagian atas kiri tekan tombol .



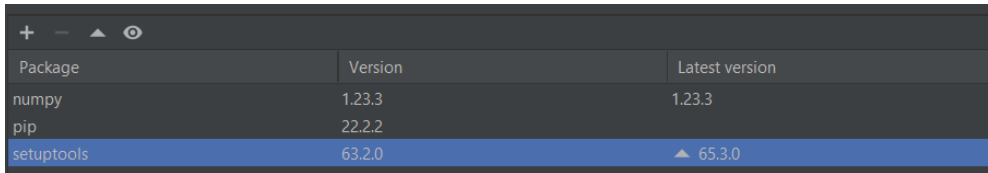
Untuk memudahkan pencarian package, ketikkan nama package pada kolom pencarian. Setelah tekan tombol **Install Package**. Kita dapat menentukan letak package diinstal, mengikuti default sistem atau Pycharm.



Versi package dapat ditentukan dengan mencentang pilihan **Specify version** dan pilih versi yang ingin digunakan.



5. Upgrade versi package



Package	Version	Latest version
numpy	1.23.3	1.23.3
pip	22.2.2	
setuptools	63.2.0	▲ 65.3.0

Pilih package yang ingin diupgrade dan terdapat tanda ▲ pada kolom **Latest version**. Kemudian tekan tombol ▲ pada bagian atas. Setelah itu tekan tombol **OK** di kanan bawah.



Source

1. [Software Environment - Medium](#)
2. [pip](#)
3. [Virtual Environment](#)
4. [JupyterLab ReadTheDocs](#)
5. [Python In Office](#)