



Direktorat Jenderal Pendidikan Tinggi, Riset, dan, Teknologi
Kementerian Pendidikan, Kebudayaan, Riset, dan Teknologi
Republik Indonesia



MICROCREDENTIAL: ASSOCIATE DATA SCIENTIST

01 November – 10 Desember 2021

Pertemuan ke-4

Tools Proyek Data Science



1



ditjen.dikti



@ditjendikti



ditjen.dikti



Ditjen Diktiristek



<https://dikti.kemdikbud.go.id/>



Ir. Rahmad Dawood, S.Kom., M.Sc., IPM, ASEAN.Eng.



Contak Pengajar:

Email:

rahmad.dawood@Unsyiah.ac.id

Jabatan Akademik: Dosen di Jurusan Teknik Komputer, Universitas Syiah Kuala

Latar Belakang Pendidikan:

- S1: Teknik Komputer, ITS, Surabaya, Indonesia
- S2: Computer Science, Columbia University, New York, USA

Riwayat/Pengalaman Pekerjaan:

- Dosen Teknik Komputer, Universitas Syiah Kuala
- Dosen Teknik Elektro, Universitas Syiah Kuala
- Manajer Pendidikan Tinggi, BRR NAD/Nias
- Sr. Consultant, Context Integration
- Direktur PT. Mitra Sarana Unggul



Deskripsi Pelatihan

Tujuan utama dari modul pelatihan ini adalah untuk membahas data science tools dengan menjelaskan seperangkat kelas dan teknik yang berkaitan dengan keterampilan dasar dalam ilmu komputer, matematika, dan statistik untuk melakukan tugas-tugas yang umumnya terkait dengan data science.



Capaian Pembelajaran

Pada topik ini, kita akan mempelajari:

- Bahasa Pemrograman Python
- *Development Environment*
- Dasar-dasar library Python untuk proyek data science
 - NumPy
 - SciPy
 - Pandas
 - Matplotlib
 - Seaborn
 - Scikit-learn



Mengapa Python?

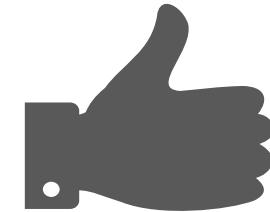


- Bahasa pemrograman tingkat tinggi
- Penulisan kode/sintaks lebih sederhana dan tersedia banyak library
- Bersifat *open-source* dan *cross-platform*
- Diluncurkan oleh Guido Van Rossum pada tahun 1991.

Data Professional



- Data Analyst
- Data Engineer
- Data Scientist
- Business Intelligence
- ML Engineer



- Cocok untuk pemula
- Sederhana tapi *powerful*
- *High-demand skill*



Mengapa Python?

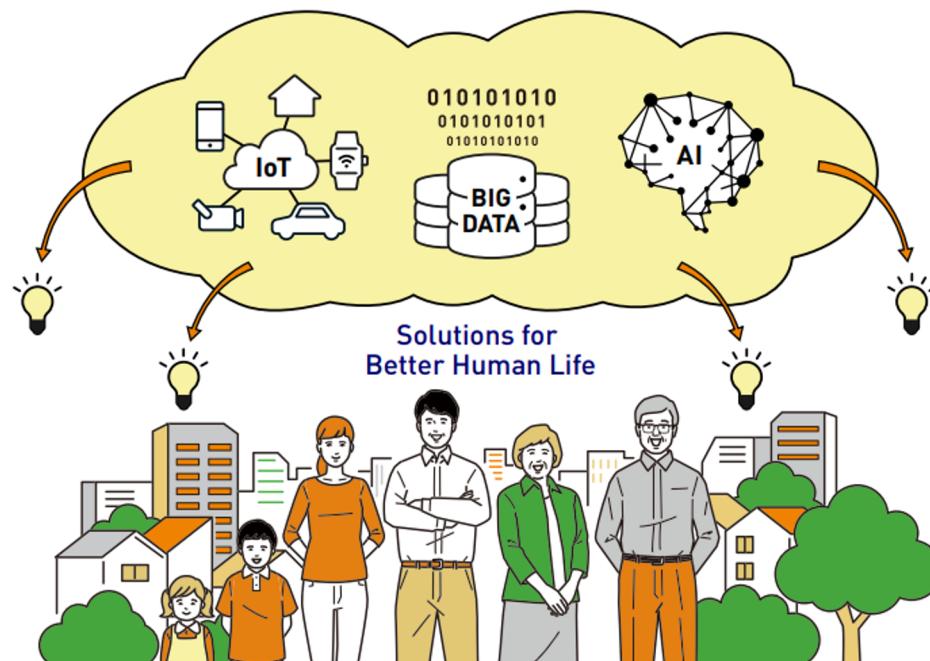


Keunggulan Python:

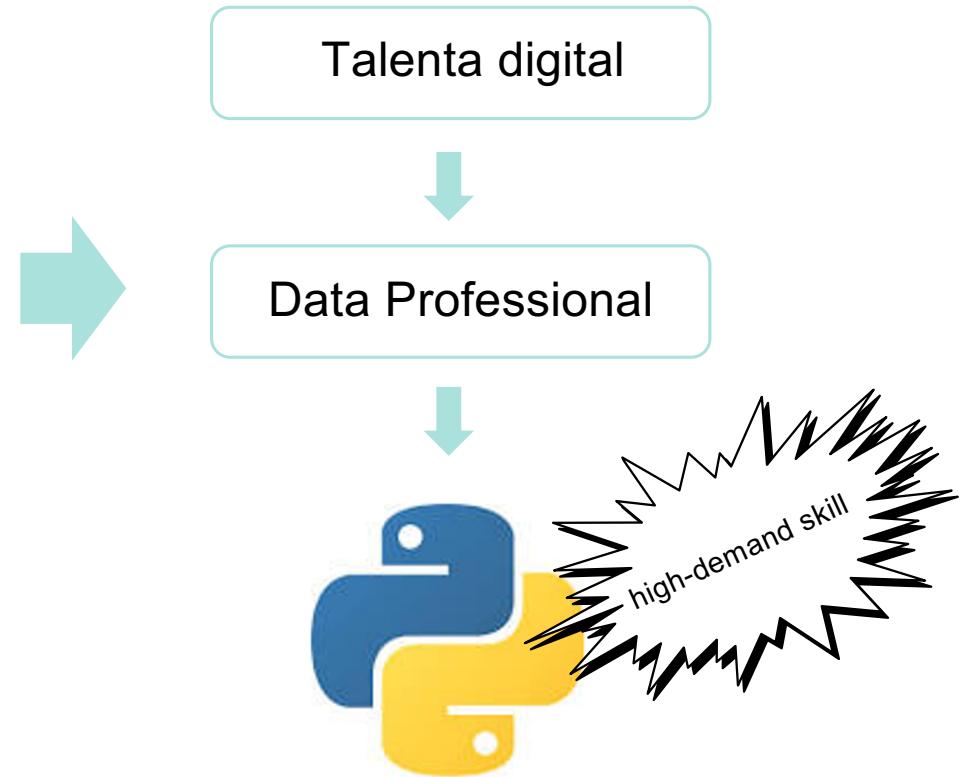
- Keterbacaan, Python mudah dibaca dan dipahami
- Efisien, memiliki library yang lengkap sehingga penulisan coding dapat lebih sederhana
- Multifungsi, dengan menggunakan Python dapat membuat website, aplikasi bidang robotika, aplikasi bidang kecerdasan artifisial.
- Interoperabilitas, Python mampu berinteraksi dengan Bahasa pemrograman lainnya
- Dukungan komunitas, Python merupakan program open source, dan komunitas Python sangat aktif dalam melakukan pengembangan



Mengapa Python?



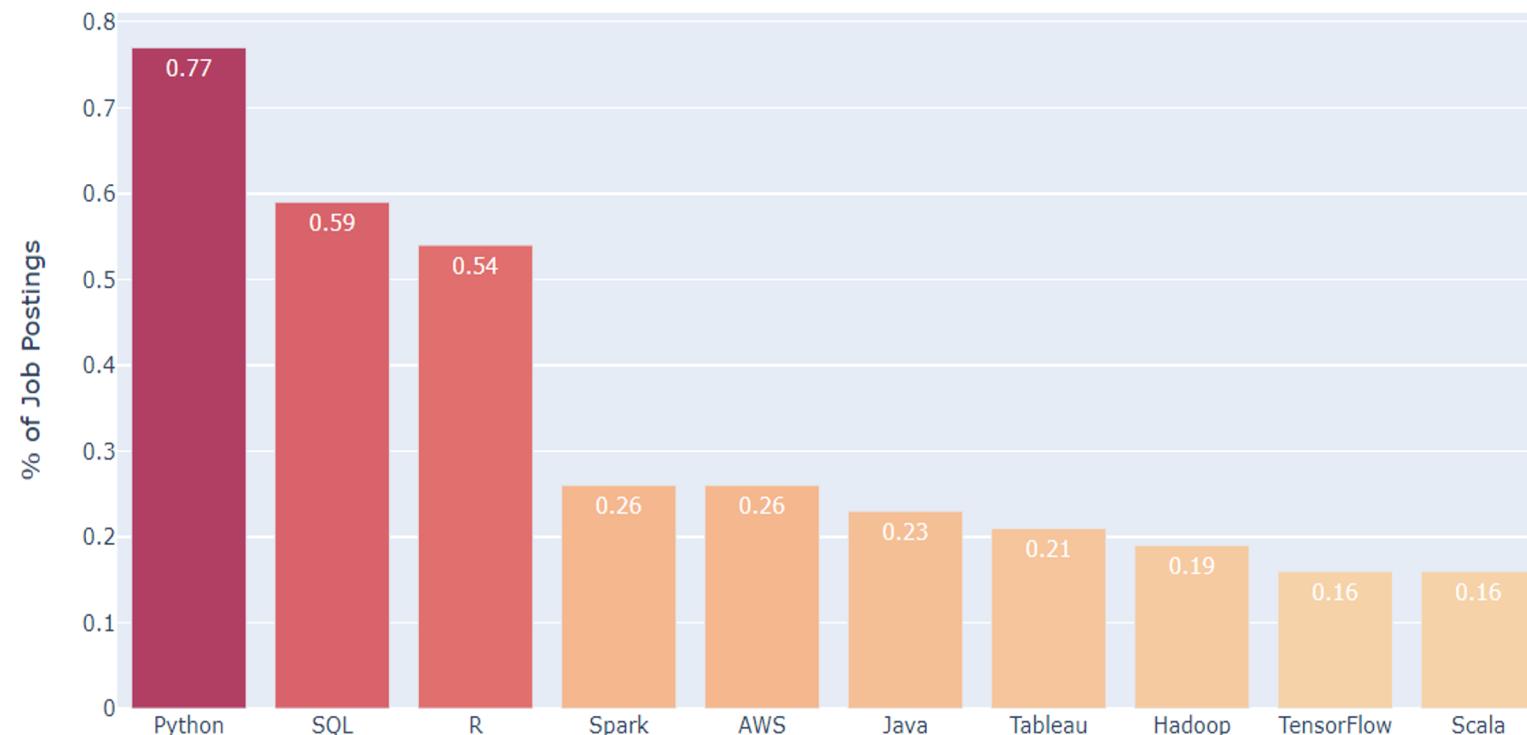
https://www.japan.go.jp/abenomics/_userdata/abenomics/pdf/society_5.0.pdf





Python Menjadi yang Pertama dalam Daftar Keahlian yang Paling Dibutuhkan

10 Most In-Demand Data Science Skills in 2021



(sumber: <https://towardsdatascience.com>)



Python Digunakan pada YouTube

“Google runs millions of lines of Python code. The front-end server that drives youtube.com and YouTubes APIs is primarily written in Python, and it serves millions of requests per second!”
– Dylan Trotter, Youtube Engineer, 2017

<https://opensource.googleblog.com/2017/01/grumpy-go-running-python.html>





Python Digunakan pada Quora

Quora

“We decided that Python was fast enough for most of what we need to do (since we push our performance-critical code to backend servers written in C++ whenever possible). As far as type checking, we ended up writing very thorough unit tests which are worth writing anyway, and achieve most of the same goals.”

– Adam D’Angelo, CEO Quora, 2014

<https://www.quora.com/Why-did-Quora-choose-Python-for-its-development>

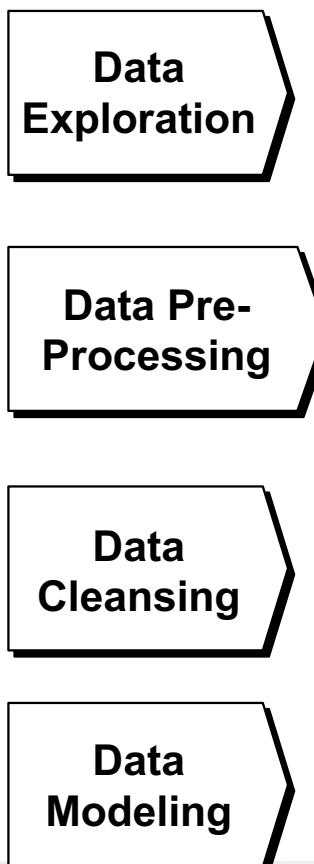


Python Digunakan pada Beberapa Industri





Penerapan Python pada Proyek Data Science



- Scraping, crawling, data mining
- Coding, query
- Seleksi fitur, statistika deskriptif, *class balancing*, visualisasi data
- Transformasi fitur: *Categorical encoding*, *binning*
- Menangani nilai kosong (*missing values*), menghapus baris terduplikasi
- *Data formatting*, menangani data pencilan (*outliers*)
- Melatih data dengan algoritma *machine learning*
- Melakukan klasifikasi, regresi, prediksi, klasterisasi



Memulai Python

- **Python** adalah bahasa *interpreter*, yang dapat mengurangi siklus *edit-test-debug* karena tidak memerlukan langkah kompilasi
- Untuk menjalankan Python, Anda memerlukan *runtime/interpreter environment* untuk mengeksekusi kode:
 - Mode interaktif: Setiap perintah yang Anda tulis akan langsung ditafsirkan dan segera dieksekusi sehingga bisa langsung melihat hasilnya □ **IPython**
 - Mode skrip: Anda memasukkan satu set kode Python ke dalam format .py, program dijalankan baris demi baris





Konsep IPython: REPL Environment

Read

- Proses membaca code

Eval

- Proses evaluasi (eksekusi) code

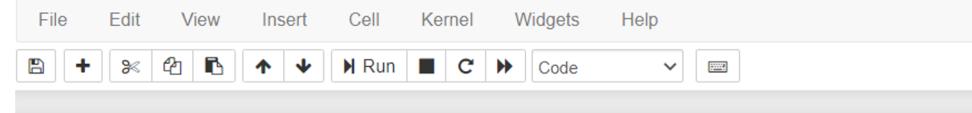
Print

- Proses menampilkan hasil (*output*)

Loop

- Pengulangan proses R-E-P

jupyter Get Started with Python Last Checkpoint: 42 menit yang lalu (unsaved changes)



Membaca (Read) dan mengevaluasi (Eval) code, sekaligus menampilkan hasil (Print)

In [4]: a = 2 + 3

In [5]: print(a)

5

Deklarasi variabel dan fungsi, kemudian akan disimpan dan dapat dipanggil kembali (Loop)

In [6]: pi = 3.14

In [7]: pi

out[7]: 3.14



Pilihan *Development Environment*

Pilih *Development Environment* yang paling mudah dan nyaman:

- Anaconda Distribution (<https://www.anaconda.com/distribution/>)
 - Python
 - Conda lebih dari 1000 library data science
- Miniconda (<https://docs.conda.io/en/latest/miniconda.html>)
 - Python interpreter
 - Conda
- Jupyter Notebook (<https://jupyter.org/>)
- Python installer (<https://www.python.org/downloads/>).
- Google Colaboratory (<https://colab.research.google.com/>).
- Notebooks Azure (<https://notebooks.azure.com/>)
- Github (<https://github.com/>)



Anaconda Distribution



Anaconda Navigator

Sebuah aplikasi *dashboard interface* pada paket Anaconda Distribution

Orange:
Visual
Programming.
data visualization,
machine learning and
data mining toolkit

The screenshot shows the Anaconda Navigator interface with a sidebar on the left containing links for Home, Environments, Projects (beta), Learning, and Community. The main area displays a grid of applications:

- jupyter notebook** (5.0.0): Web-based, interactive computing notebook environment. Edit and run human-readable docs while describing the data analysis.
- IPython qtconsole** (4.3.0): PyQt GUI that supports inline figures, proper multiline editing with syntax highlighting, graphical calltips, and more.
- spyder** (3.1.4): Scientific Python Development Environment. Powerful Python IDE with advanced editing, interactive testing, debugging and introspection features.
- glueviz** (0.10.4): Multidimensional data visualization across files. Explore relationships within and among related datasets.
- orange3** (3.4.1): Component based data mining framework. Data visualization and data analysis for novice and expert. Interactive workflows with a large toolbox.
- rstudio** (1.0.136): A set of integrated tools designed to help you be more productive with R. Includes R essentials and notebooks.

A red circle highlights the **jupyter notebook** icon. A blue box labeled "R programming: analisis statistika dan grafik" has arrows pointing to the **rstudio** and **orange3** icons. A pink box labeled "Jupyter Notebook" has an arrow pointing to the **jupyter notebook** icon.



Jupyter Notebook



- Lingkungan pemrograman interaktif berbasis web yang mendukung berbagai bahasa pemrograman termasuk Python
- Banyak digunakan oleh peneliti dan akademisi untuk pemodelan matematika, pembelajaran mesin, analisis statistik, dan untuk pengajaran pemrograman



Jupyter Notebook

localhost platform

The screenshot shows a Jupyter Notebook interface titled "Get Started with Python". The toolbar includes File, Edit, View, Insert, Cell, Kernel, Widgets, Help, Trusted, and Python 3. A yellow arrow points from the "Code" option in the dropdown menu to the "In [3]" cell, which contains the Python code `print("Hello World!")` and its output "Hello World!". The "Widgets" button in the toolbar is highlighted with a yellow box.

- Skrip dapat ditulis dalam bentuk:
 - **Code** : Algoritma dan formula matematis
 - **Markdown/Heading** : Teks deskripsi, penjelasan code, gambar/image
 - **Raw NBConvert** : Konversi format yang berbeda
- Hasil dapat diketahui langsung setelah menjalankan perintah **Run**



Google Colaboratory

Online/web platform

The screenshot shows the Google Colaboratory interface. At the top, there's a navigation bar with back, forward, and refresh buttons, followed by a URL bar containing the address colab.research.google.com/notebooks/intro.ipynb#scrollTo=gJr_9dXGpJ05. Below the URL bar is a menu bar with CO, Selamat Datang di Colaboratory, File, Edit, Lihat, Sisipkan, Runtime, Fitur, Rantuan, Bagikan, Mengedit, and a user profile icon.

The main area has a sidebar on the left with sections like Daftar isi, Memulai (Ilmu data, Machine learning, Referensi Lainnya, Contoh Machine Learning), and Bagian. A yellow box highlights the '+ Kode' and '+ Teks' buttons in the toolbar above the code cell. A yellow arrow points from the bottom-left towards the 'Run' button in the code cell.

The code cell contains the following Python code:

```
seconds_in_a_day = 24 * 60 * 60
seconds_in_a_day
```

The output of the code is:

```
86400
```

A descriptive text below the code cell explains how to execute it:

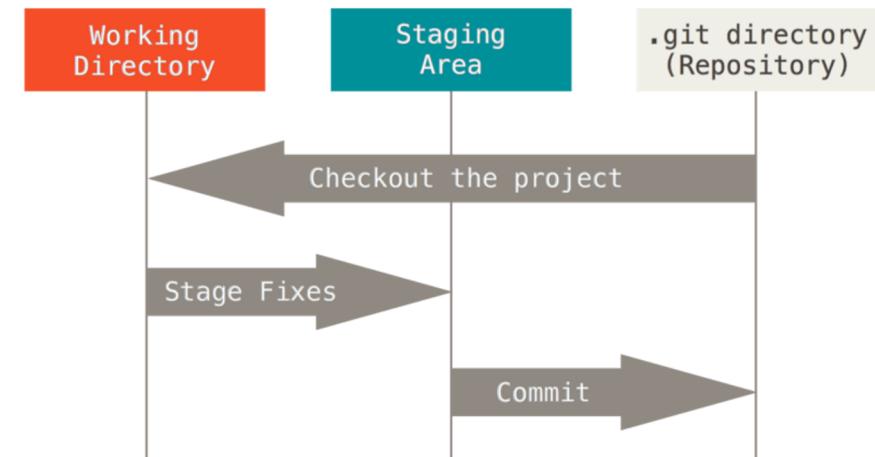
Untuk mengeksekusi kode dalam sel di atas, pilih kode tersebut dengan mengkliknya, kemudian tekan tombol putar di sebelah kiri kode atau gunakan pintasan keyboard "Command/Ctrl+Enter". Untuk mengedit kode, cukup klik sel dan mulai pengeditan.

- Skrip dapat ditulis dalam bentuk:
 - **Code:** Algoritma dan formula matematis
 - **Teks:** Teks deskripsi, penjelasan code
- Dapat digunakan pada <https://colab.research.google.com/> dan hasil dapat diketahui langsung setelah menjalankan perintah *Run*



Bekerja dengan Git

- **Git** merupakan tools yang bersifat *open source* untuk memudahkan bekerja dengan proyek berskala kecil maupun besar (<https://git-scm.com/>)
- Git memiliki tiga status utama tempat file berada: *modified, staged, committed*:
 - **Modified** berarti Anda telah mengubah file tetapi belum menyimpannya ke database Anda
 - **Staged** berarti Anda telah menandai file yang dimodifikasi dalam versi terbaru untuk masuk ke tahap *commit*
 - **Commit** berarti bahwa data disimpan dengan aman di database local Anda





Bekerja dengan Git

- **Inisialisasi:** `git init`
- **Commit:** `git commit -m "first commit"`
- **Branch:** `git branch -M main`
- **Add:** `git remote add origin https://github.com/\[user\]/\[repo\].git`
- **Push:** `git push -u origin main`
- **Pull:** `git pull origin [branch]`



Hello World!

Bahasa C

```
#include <stdio.h>

int main() {
printf("Hello
World!");
return 0;
}
```

Bahasa Python

```
print("Hello
World!")
```

- Lebih sederhana
- Tidak ada kurung kurawal {..}
- Tidak perlu titik koma ;



Tipe Data Python

- float – bilangan riil
- int – bilangan bulat (integer)
- str – string, teks
- bool – True or False

```
In [1]: height = 1.84
In [2]: tall = True
```

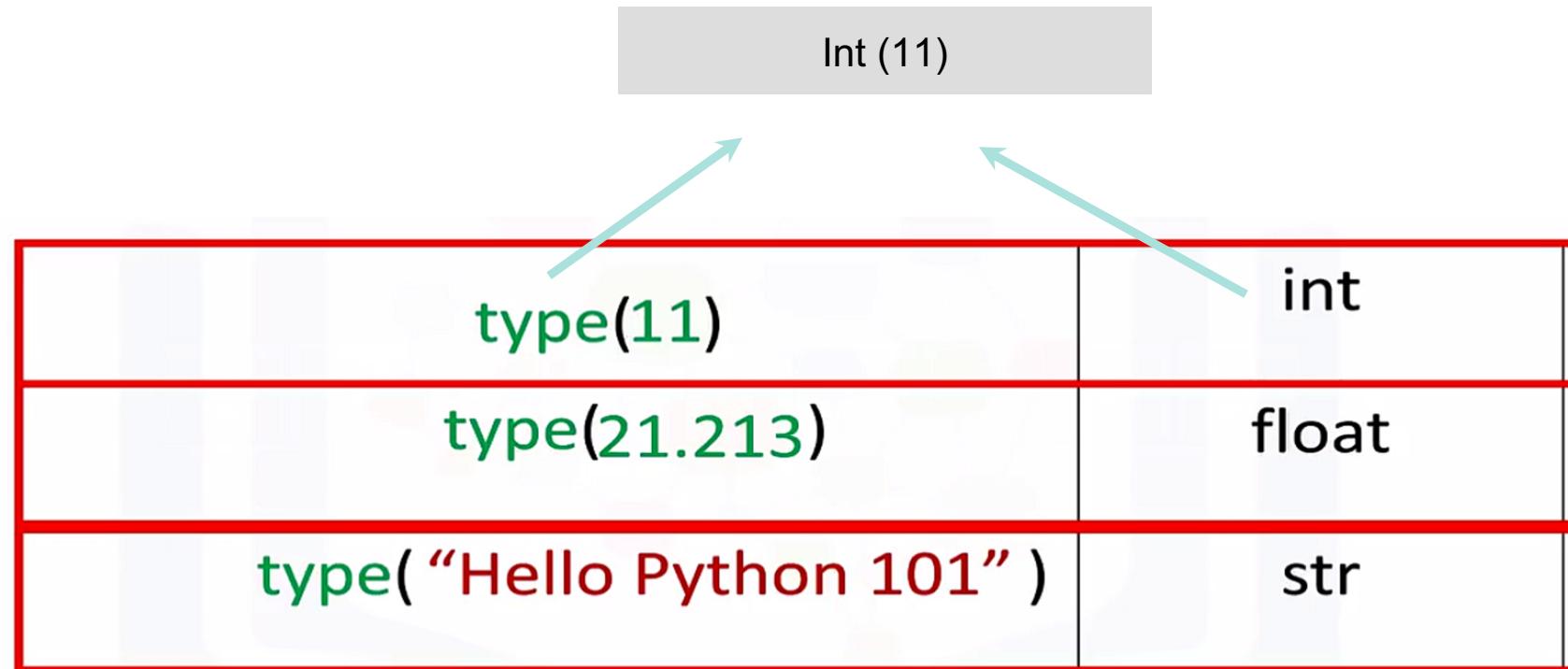
- Masalah
 - Terlalu banyak data masukan untuk tipe data yang sama
 - Tidak nyaman

```
In [3]: height1 = 1.84
In [4]: height2 = 1.79
In [5]: height3 = 1.82
In [6]: height4 = 1.90
```

- Solusi □ Python List



T



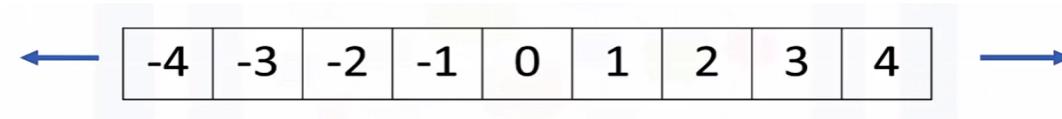


Tipe Data Python

T

Tipe Data Integer

Merupakan bilangan bulat negatif maupun positif



Tipe Data Float

Bilangan bentuk bilangan pecahan dapat negatif maupun positif



Tipe Data String

Merupakan data yang berisi string berupa angka, huruf dan symbol.

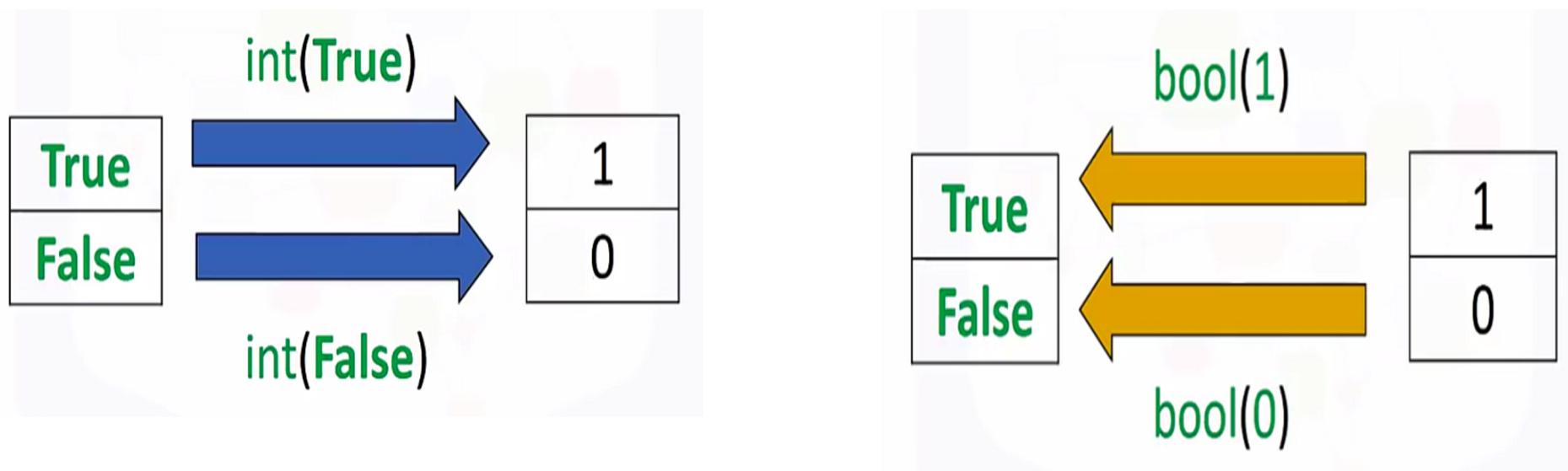


T

Tipe Data Boolean

Merupakan tipe data dengan dua keluaran yaitu: True atau False.

Pada Python, tipe data boolean adalah **bool**.





T

Type Casting

Pada Python, dapat dilakukan pengubahan tipe data.

Misalkan: tipe data integer akan diubah menjadi tipe data float.

`float(2):2.0`

Integer 2 diubah ke tipe Float, menjadi: 2.0

`int(1.1):1`

Float 1.1 diubah ke tipe Integer, menjadi: 1
Perlu diingat: perubahan ini mengakibatkan ada nilai yang hilang, yaitu: 0.1

`int('1'):1`

A merupakan karakter, TIDAK DAPAT diubah menjadi integer

~~`int('A')`~~

`str(1): "1"`

Integer 1 diubah ke tipe string menjadi: '1'
Float 4.5 diubah ke tipe string, menjadi: '4.5'

`str(4.5):'4.5'`



List

T

List merupakan daftar yang memiliki anggota/element yang berurutan. Anggota/element dari list dituliskan di antara tanda kurung siku [...] dan masing-masing anggota/element dipisahkan oleh tanda koma (,).

List bersifat **mutable** (tidak tetap).

```
L = ["Michael Jackson", 10.1, 1982]
```

```
["Michael Jackson", 10.1, 1982, [1, 2], ('A', 1)]
```

List dapat berisikan anggota/element dengan tipe data string, integer, float, bersarang.



List

T

Anggota/element pada list dapat diakses dengan menggunakan indeks. Indeks pada list dapat dibuat dalam bentuk indeks positif maupun indeks negatif.

L = ["Michael Jackson", 10.1, 1982]

0	"Michael Jackson"
1	10.1
2	1982

L[0]: "Michael Jackson"

L[1]: 10

L[2]: 1982

L = ["Michael Jackson", 10.1, 1982]

-3
-2
-1

0	"Michael Jackson"
1	10.1
2	1982

L[-3]: "Michael Jackson"

L[-2]: 10

L[-1]: 1982

Indeks Positif

Indeks Negatif



Python List [a, b, c]

- Koleksi nilai
- Dapat mengandung beberapa tipe data berbeda

```
In [7]: [1.84, 1.79, 1.82, 1.90, 1.80]
Out[7]: [1.84, 1.79, 1.82, 1.90, 1.80]
```

```
In [8]: height = [1.84, 1.79, 1.82,
1.90, 1.80]
```

```
In [9]: height
Out[9]: [1.84, 1.79, 1.82, 1.90, 1.80]
```

```
In [10]: famz = ["Abe", 1.84, "Beb",
1.79, "Cory", 1.82, "Dad", 1.90]
```

```
In [11]: famz
Out[11]: ["Abe", 1.84, "Beb", 1.79,
"Cory", 1.82, "Dad", 1.90]
```

```
["Abe", 1.84]
["Beb", 1.79]
["Cory", 1.82]
["Dad", 1.90]
```



Python List

```
In [1]: height = [1.84, 1.79, 1.82, 1.90, 1.80]
```

```
In [2]: height
```

```
Out[2]: [1.84, 1.79, 1.82, 1.90, 1.80]
```

```
In [3]: weight = [66.5, 60.3, 64.7, 89.5, 69.8]
```

```
In [4]: weight
```

```
Out[4]: [66.5, 60.3, 64.7, 89.5, 69.8]
```

```
In [5]: weight / height ** 2
```

```
TypeError: unsupported operand type(s) for ** or pow(): 'list' and 'int'
```

Problem!



Solusi:

- Library dasar untuk perhitungan saintifik (*scientific computing*) dengan Python (<https://numpy.org/>)
- Alternatif untuk Python List: Numpy Array untuk n -dimensi
- Mudah digunakan dan bersifat *open source*
- Jika library belum terpasang, tuliskan perintah instalasi:

```
pip install numpy
```

- Kemudian impor:

```
import numpy as np
```

NumPy

```
In [6]: import numpy as np

In [7]: np_height = np.array(height)

In [8]: np_height
Out[8]: array([1.84, 1.79, 1.82, 1.9, 1.8])

In [9]: np_weight = np.array(weight)

In [10]: np_weight
Out[10]: array([66.5, 60.3, 64.7, 89.5, 69.8])

In [11]: bmi = np_weight / np_height ** 2

In [12]: bmi
Out[12]: array([19.64201323, 18.81963734,
   19.53266514, 24.79224377, 21.54320988])
```



NumPy



Digital data

- Pengolahan data dapat berupa bermacam-macam bentuk dan formatnya: dokumen, gambar, video, suara, angka, atau teks
- Ketika data-data tersebut diproses, tidak secara mentah-mentah dibaca sebagai video atau audio. Tetapi sudah dilakukan transformasi ke dalam bentuk array atau *matrix of number*
- Array dengan minimal dua dimensi akan membentuk matriks dan dapat menggunakan NumPy

```
import numpy as np  
np.<TAB>
```



NumPy

- NumPy juga dapat digunakan untuk membuat array berdimensi- n

```
In [13]: import numpy as np
```

```
In [14]: np_height = np.array([1.84, 1.79,  
1.82, 1.9, 1.8])
```

```
In [15]: np_weight = np.array([66.5, 60.3,  
64.7, 89.5, 69.8])
```

```
In [16]: type(np_height)  
Out[16]: numpy.ndarray
```

```
In [16]: type(np_weight)  
Out[16]: numpy.ndarray
```

ndarray = n-dimensional array

```
In [17]: np_2d = np.array([[1, 2, 3, 4, 5],  
[6, 7, 8, 9, 10]])
```

```
In [18]: np_2d  
Out[18]: array([[1, 2, 3, 4, 5],  
[6, 7, 8, 9, 10]])
```

```
In [19]: np_2d.shape  
Out[19]: (2, 5)
```

Array berdimensi 2 baris 5 kolom → Matriks $M_{2 \times 5}$

$$M = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 \\ 6 & 7 & 8 & 9 & 10 \end{bmatrix}$$



- **SciPy** (dibaca “*Sigh Pie*”) merupakan library yang bersifat *open source* dan tersedia di <https://www.scipy.org/>
- SciPy dibangun untuk bekerja dengan NumPy array dan menyediakan kumpulan algoritma numerik, termasuk pemrosesan sinyal, optimasi, statistika, dan library Matplotlib untuk visualisasi data.
- Jika library belum terpasang, tuliskan perintah instalasi:

```
pip install scipy
```



pandas

- **Pandas** (Panel Data) merupakan library popular di Python yang digunakan untuk *data structure* dan *data analysis*
- Bersifat *open source* dan tersedia di <https://pandas.pydata.org/>
- Pandas sangat berkaitan dengan NumPy
- Jika library belum terpasang, tuliskan perintah instalasi:

```
pip install pandas
```

- Kemudian impor:

```
import pandas as pd
```

Data Wrangling / Data Munging

- *Reshaping* (mengubah bentuk data)
- *Joining* (menggabungkan data)
- *Splitting* (pemisahan data)
- *Time-series analysis* (data berkala)

Data Cleansing

- Membersihkan data tidak lengkap (*Error*)
- Menangani data pencilan (*outliers*)
- Menghapus data duplikat



Representasi Data di pandas

- Terdapat 2 *data objects*: **Series** dan **DataFrame**
- **Series** □ Data berbentuk 1 dimensi

```
In [13]: np.array([1, 2, 3, 4, 5])  
Out[13]: array([1, 2, 3, 4, 5])
```

- **DataFrame** □ Data berbentuk 2 dimensi atau lebih

```
In [14]: np.array([[1, 2], [3, 4]])  
Out[14]: array([[1, 2],  
                 [3, 4]])
```

Kolom: **Fitur / atribut/variabel**

Baris: sampel

	Negara	Populasi	Area	Ibukota
IN	Indonesia	250	123456	Jakarta
MA	Malaysia	25	3456	KL
SI	Singapura	15	456	Singapura
JP	Jepang	60	5678	Tokyo
TH	Thailand	45	678	Bangkok



pandas

- Pandas dapat mengimpor data dari berbagai format: *comma-separated value* (CSV), file teks, Microsoft Excel, database SQL, dan format HDF5
- Unduh dataset: <http://bit.ly/TabDataset>
- CSV file □ DataFrame

```
import pandas as pd
```

Tab.cs
v

Negara,Populasi,Area,Ibukota
IN,Indonesia,250,123456,Jakarta
MA,Malaysia,25,3456,KL
SI,Singapura,15,456,Singapura
JP,Jepang,60,5678,Tokyo
TH,Thailand,45,678,Bangkok

```
In [1]: Tab = ... # deklarasi tabel
```

```
In [2]: Tab
```

		Negara	Populasi	Area	Ibukota
	IN	Indonesia	250	123456	Jakarta
	MA	Malaysia	25	3456	KL
	SI	Singapura	15	456	Singapura
	JP	Jepang	60	5678	Tokyo
	TH	Thailand	45	678	Bangkok



pandas

```
In [3]: import pandas as pd
```

```
In [4]: Tab = pd.read_csv("Tab.csv")
```

```
In [5]: Tab
```

```
Out[5]:
```

	Unnamed: 0	Negara	Populasi	Area	Ibukota
0	IN	Indonesia	250	123456	Jakarta
1	MA	Malaysia	25	3456	KL
2	SI	Singapura	15	456	Singapura
3	JP	Jepang	60	5678	Tokyo
4	TH	Thailand	45	678	Bangkok

```
In [6]: Tab["Negara"] # akses kolom
```

```
Out[6]:
```

```
0    Indonesia  
1    Malaysia  
2    Singapura  
3    Jepang  
4    Thailand
```

```
Name: Negara, dtype: object
```

```
In [7]: Tab.Ibukota # akses kolom
```

```
Out[7]:
```

```
0    Jakarta  
1        KL  
2    Singapura  
3      Tokyo  
4    Bangkok
```

```
Name: Ibukota, dtype: object
```



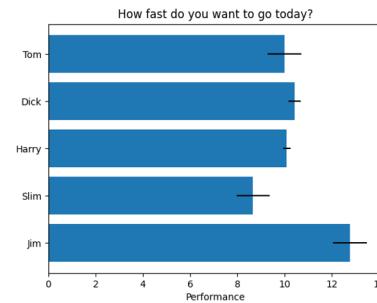
matplotlib

- **Matplotlib** adalah library Python untuk visualisasi data dengan dua dimensi
- Bersifat *open source* dan tersedia di <https://matplotlib.org/>
- Matplotlib berkaitan dengan NumPy dan Pandas
- Jika library belum terpasang, tuliskan perintah instalasi:

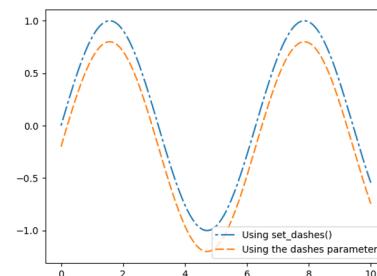
```
pip install matplotlib
```

- Kemudian impor:

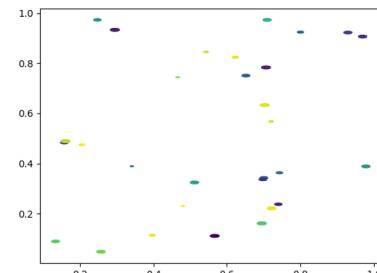
```
import matplotlib.pyplot as plt
```



bar chart



Line chart

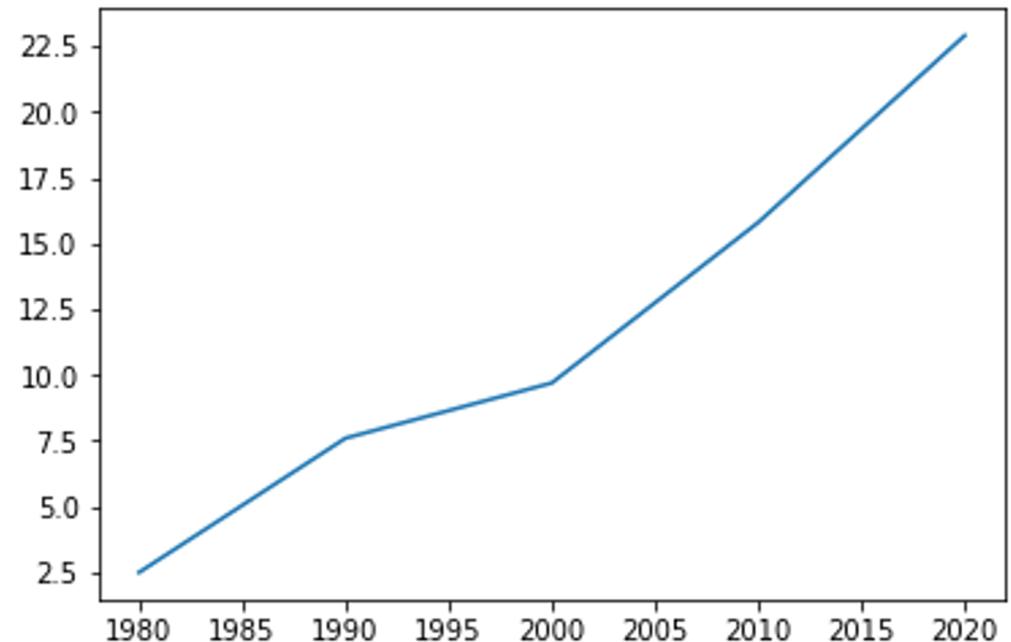


Scatter plot



matplotlib

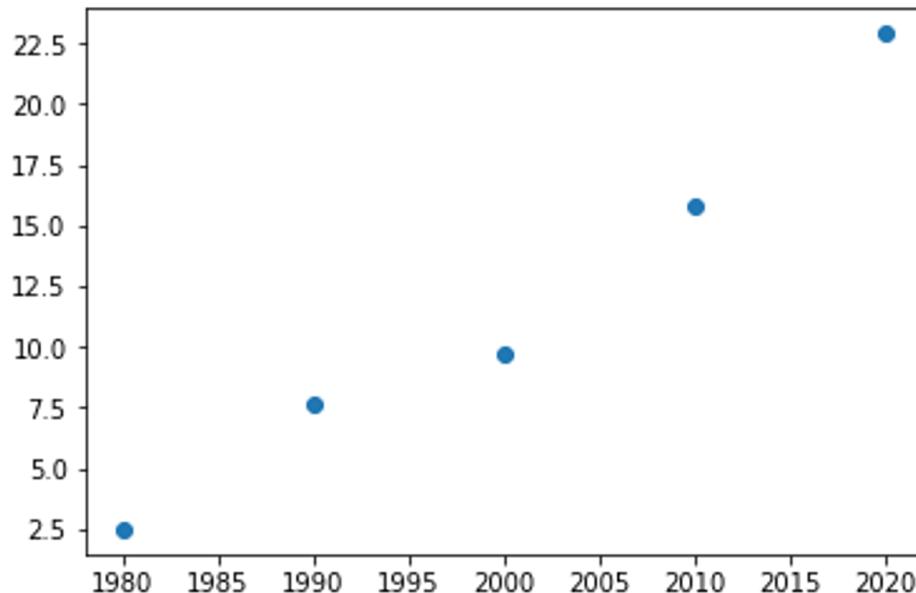
```
In [1]: import matplotlib.pyplot as plt  
  
In [2]: year = [1980, 1990, 2000, 2010, 2020]  
  
In [3]: price = [2.5, 7.6, 9.7, 15.8, 22.9]  
  
In [4]: plt.plot(year, price)  
  
In [5]: plt.show()
```



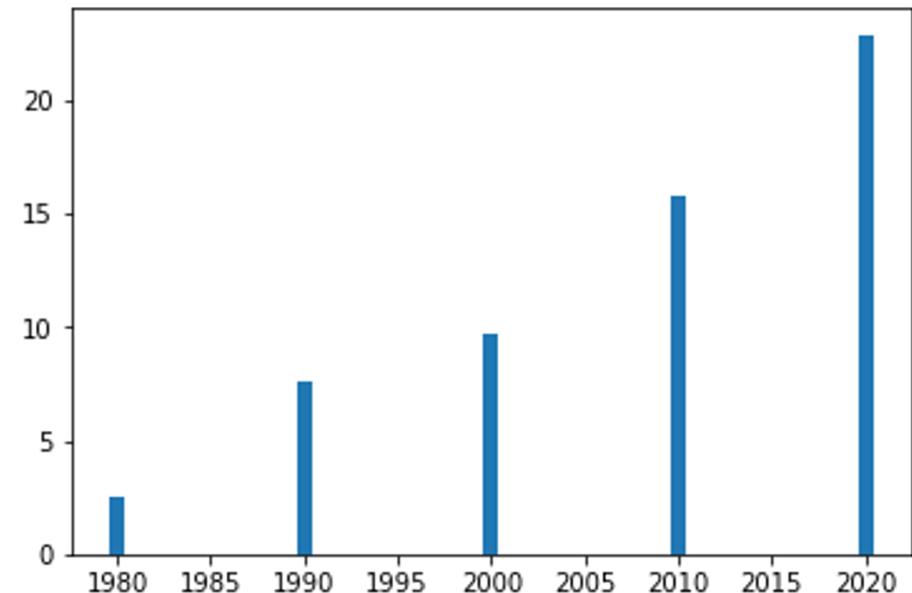


matplotlib

```
In [6]: plt.scatter(year, price)
```



```
In [7]: plt.bar(year, price)
```





- **Seaborn** adalah library visualisasi data Python (serupa dengan Matplotlib) yang menyediakan *high-level interface* untuk menggambar grafik statistika yang menarik dan informatif

- Library ini bersifat *open source* dan tersedia di <https://seaborn.pydata.org/>
- Jika library belum terpasang, tuliskan perintah instalasi:

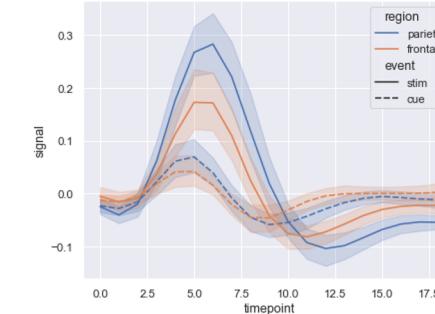
```
pip install seaborn
```

- Kemudian impor:

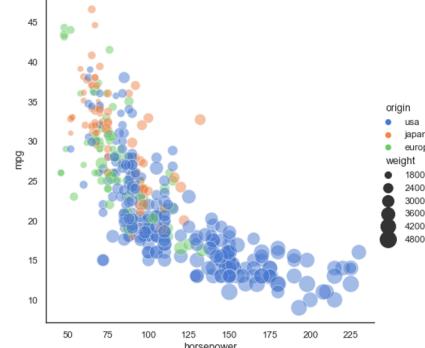
```
import seaborn as sns
```



Heatmap



Line chart



Scatter plot



- **Scikit-learn** adalah library untuk mempraktikkan *machine learning* dan membuat model
- Bersifat *open source* dan tersedia di <https://scikit-learn.org/>
- Scikit-learn diawali dari project SciPy (*Scientific Python*) yang berisi fungsi-fungsi matematis
- Jika library belum terpasang, tuliskan perintah instalasi:

```
pip install sklearn
```

- Kemudian impor:

```
import sklearn
```

Classification

- *Support Vector Machines*
- *Decision Tree*
- *Random Forest*
- *Neural Network*
- *Nearest neighbors*

Clustering

- *K-Means Clustering*
- *Hierarchical Clustering*

Model Selection

- *Cross validation*
- *Metrics*



Summary

Pada topik ini, kita sudah mempelajari:

- Keunggulan Python sebagai Tools dalam proyek data science
- *Development environment* Python yang bervariasi, baik yang bersifat *offline (local computer)* maupun berbasis web (Jupyter Notebook / Google Colaboratory)
- Dasar-dasar library Python untuk proyek data science:
 - NumPy □ library untuk *numerical computation*
 - SciPy □ library untuk perhitungan statistika dan matematis
 - Pandas □ library untuk analisis dan manipulasi data
 - Matplotlib □ library untuk visualisasi data
 - Seaborn □ library untuk visualisasi data dengan *high-level interface*
 - Scikit-learn □ library untuk mempraktikkan *machine learning* dan pemodelan



Tools / Lab Online

1. Python for Data Science: <https://cognitiveclass.ai/courses/python-for-data-science>
2. Introduction to Python for Data Science: <https://www.datacamp.com/courses/intro-to-python-for-data-science>
3. Build real-world applications with Python: <https://docs.microsoft.com/en-us/learn/patterns/python-language/>
4. Introduction to Python for Data Science: <https://learning.edx.org/course/course-v1:Microsoft+DAT208x+3T2018/home>
5. Data Visualization with Python: <https://cognitiveclass.ai/courses/data-visualization-with-python>



Referensi

- Arfika Nurhudatiana. 2019. *Analisis dan Visualisasi Data Dengan Python*. Pintaria.
- Introduction to Python for Data Science: <https://learning.edx.org/course/course-v1:Microsoft+DAT208x+3T2018/home>
- Jake VanderPlas. 2016. *Python Data Science Handbook: Essential Tools for Working with Data*. O'Reilly Media.
- Laura Igual and Santi Segui. 2017. *Introduction to Data Science: A Python Approach to Concepts, Techniques and Applications*. Springer International Publishing.
- Python Tutorial: Learn Python for Data Science (Youtube: DataCamp).
https://www.youtube.com/watch?v=Rf4fZDQ0yw&list=PLjgj6kdf_snaw8QnlhK5f3DzFDFKDU5f4



Quiz / Tugas

Quiz dapat diakses melalui <https://spadadikti.id/>



Terima kasih