Data Science

Pokok Bahasan : Tools Proyek Data Science

Dosen Pengampu : Hary Sabita, ST., MTI

26 April 2022

Outline :

1. Pengantar Python
2. Library Python: NumPy, SciPy, Pandas, Matplotlib, Seaborn, Scikit-Learn
3. Integrated Development Environment
4. Web Integrated Development Environment

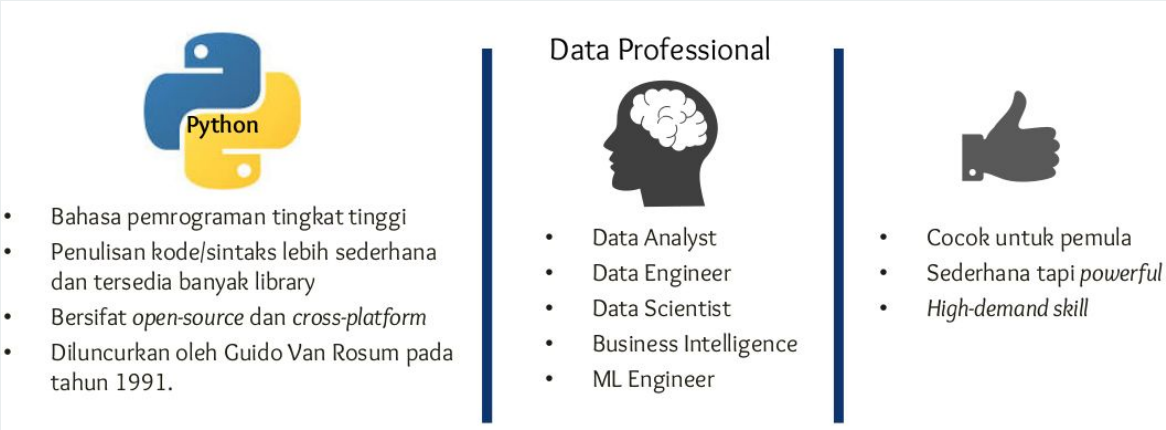
Next - Apa tujuan mempelajari ini ??

Tujuan yang akan dicapai :

1. Mampu menyelesaikan permasalahan menggunakan Python
2. Mampu menganalisis dan menginterpretasikan data menggunakan library python

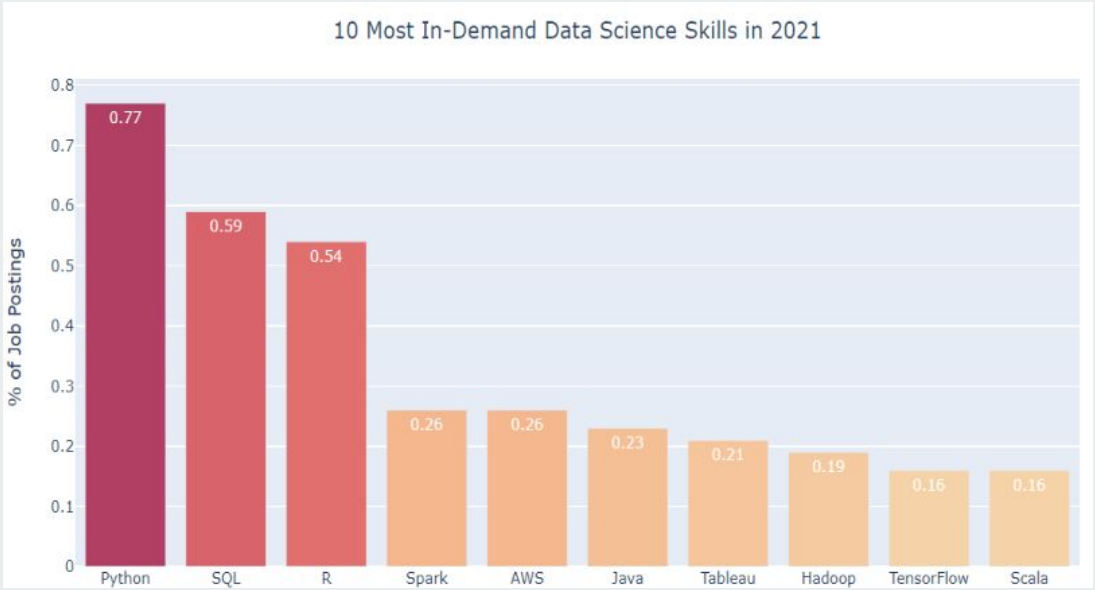
Next - Mengapa python

Mengapa Python?



Next - Keahlian yang dibutuhkan saat ini

Daftar Keahlian Pertama yang dibutuhkan

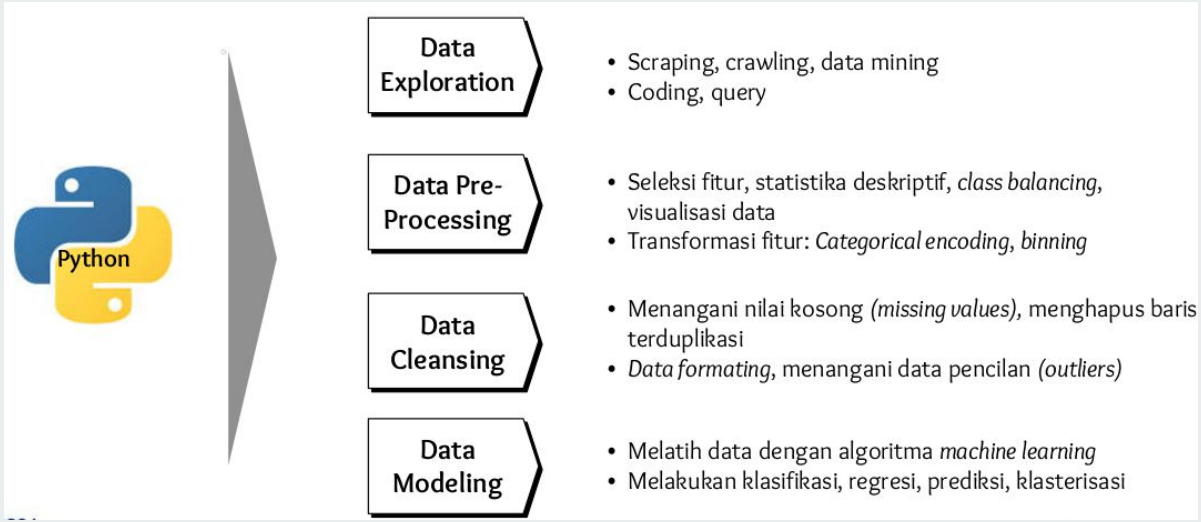


Next - Siapa pengguna python

Python digunakan pada banyak Industri

Next - bagaimana menerapkan python

Penerapan Python pada Proyek Data Science



Next - Memulai Python

Memulai Python

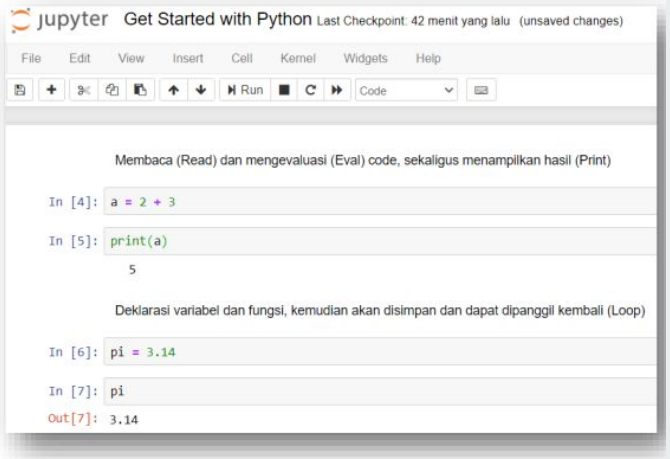
- Python adalah bahasa *interpreter*, yang dapat mengurangi siklus *edit-test-debug* karena tidak memerlukan langkah kompilasi
- Untuk menjalankan Python, Anda memerlukan *runtime/interpreter environment* untuk mengeksekusi kode:
  - Mode interaktif: Setiap perintah yang Anda tulis akan langsung ditafsirkan 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

Python interpreter

Next - Konsep Ipython

Konsep IPython: REPL Environment

- **Read** • Proses membaca code
- **Eval** • Proses evaluasi (eksekusi) code
- **Print** • Proses menampilkan hasil *(output)*
- **Loop** • Pengulangan proses R-E-P



Next - Pilihan Environment

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/)

Next - Hello World

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 ;

Next - Tipe Data Python

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

Next - List

Python list [a,b,c]

- Koleksi nilai-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]
```

```
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]
```

Problem!

```
In [5]: weight / height ** 2
TypeError: unsupported operand type(s) for ** or pow(): 'list' and 'int'
```

Next - NumPy

Solusi NumPy

- 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
```

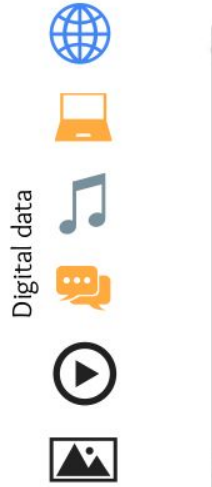
- Kemudian impor:

```
import numpy as np
```

```
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])
```

Next - NumPy

Solusi Numpy



- 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>
```

Next - NumPy

Solusi 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_{2x5}

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

Next - SciPy

SciPy

- SciPy (dibaca "*Sigh Pie*") merupakan library yang bersifat *open source* dan tersedia di https://www.scipy.org/
- SciPy dibangun untuk 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
```

Next - Pandas

Pandas

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

Next - Representasi Pandas

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

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

Next - Representasi Pandas

Representasi Data di 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.csv

```
,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

Next - Matplotlib

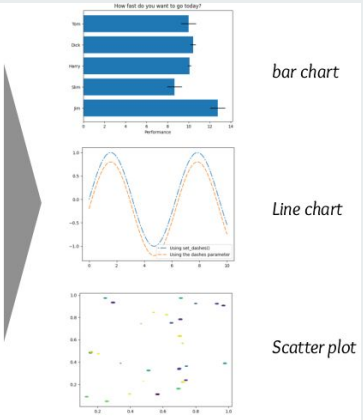
Matplotlib

- 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
```

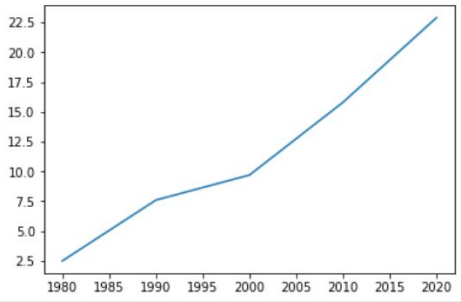


bar chart

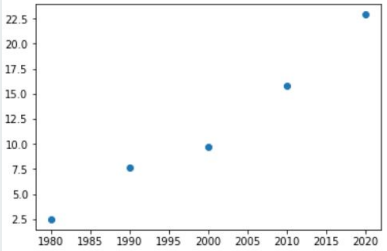
Line chart

Scatter plot

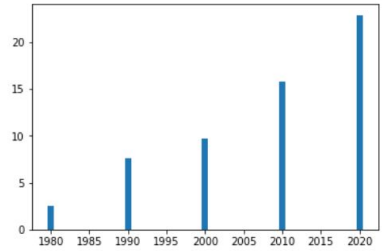
```
In [1]: import matplotlib.pyplot as plt

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()
```




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




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



Next - Seaborn

Seaborn

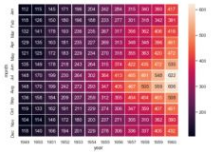
seaborn

- 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:

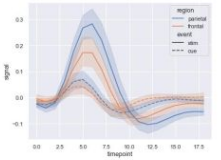
```
pip install seaborn
```

- Kemudian impor:

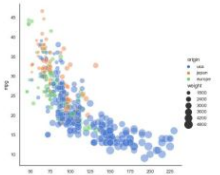
```
import seaborn as sns
```



Heatmap



Line chart



Scatter plot

Next - Scikit Learn

Scikit Learn

- 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*

Next..

Pertemuan berikutnya akan membahas business understanding..

TERIMA KASIH

Tugas Kelompok (Tugas 5)

1. Tampilkan data yang telah kalian ambil pada Kaggle dengan menggunakan library Pandas pada google colab
2. Presentasikan tugas pada pertemuan 7