

Hands-On

Hands-On ini digunakan pada kegiatan Microcredential Associate Data Scientist 2021

Pertemuan 5

Pertemuan 5 (lima) pada Microcredential Associate Data Scientist 2021 menyampaikan materi mengenai Mengumpulkan Data, Menelaah Data dengan metode Statistik

Pengambilan Data dari API Kaggle

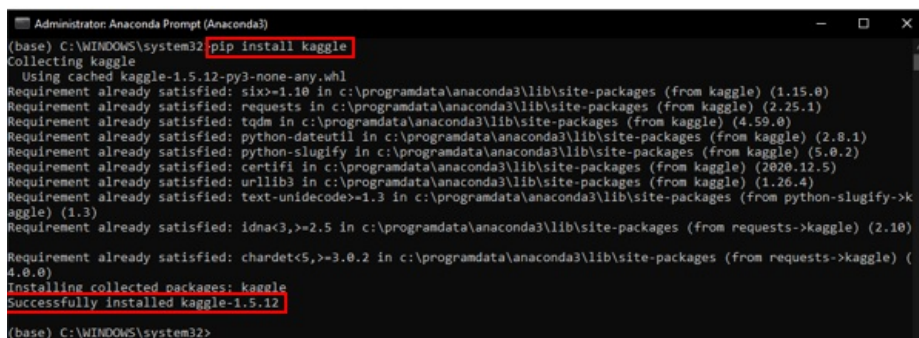
Salah satu portal yang menyediakan dataset untuk project Data Science adalah Kaggle (<https://www.kaggle.com/>). Pada latihan ini, silakan peserta mengunduh dataset mengenai bunga Iris dengan menggunakan kata kunci: "iris species" yang disediakan oleh UCI Machine Learning (UCIML)

1. Install Modul kaggle:

```
In [1]: # Install modul kaggle secara inline (di dalam notebook)
!pip install kaggle
```

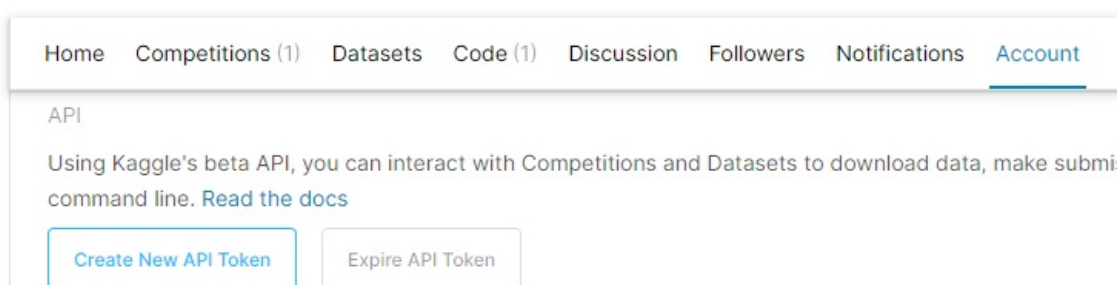
```
Requirement already satisfied: kaggle in c:\users\bunsm\anaconda3\lib\site-packages (1.5.12)
Requirement already satisfied: python-dateutil in c:\users\bunsm\anaconda3\lib\site-packages (from kaggle) (2.8.1)
Requirement already satisfied: python-slugify in c:\users\bunsm\anaconda3\lib\site-packages (from kaggle) (5.0.2)
Requirement already satisfied: certifi in c:\users\bunsm\anaconda3\lib\site-packages (from kaggle) (2020.12.5)
Requirement already satisfied: tqdm in c:\users\bunsm\anaconda3\lib\site-packages (from kaggle) (4.59.0)
Requirement already satisfied: requests in c:\users\bunsm\anaconda3\lib\site-packages (from kaggle) (2.25.1)
Requirement already satisfied: urllib3 in c:\users\bunsm\anaconda3\lib\site-packages (from kaggle) (1.26.4)
Requirement already satisfied: six>=1.10 in c:\users\bunsm\anaconda3\lib\site-packages (from kaggle) (1.15.0)
Requirement already satisfied: text-unidecode>=1.3 in c:\users\bunsm\anaconda3\lib\site-packages (from python-slugify->kaggle) (1.3)
Requirement already satisfied: chardet<5,>=3.0.2 in c:\users\bunsm\anaconda3\lib\site-packages (from requests->kaggle) (4.0.0)
Requirement already satisfied: idna<3,>=2.5 in c:\users\bunsm\anaconda3\lib\site-packages (from requests->kaggle) (2.10)
```

```
In [2]: # Install modul kaggle secara eksternal melalui anaconda prompt:
```



```
Administrator: Anaconda Prompt (Anaconda3)
(base) C:\WINDOWS\system32>pip install kaggle
Collecting kaggle
  Using cached kaggle-1.5.12-py3-none-any.whl
Requirement already satisfied: six>=1.10 in c:\programdata\anaconda3\lib\site-packages (from kaggle) (1.15.0)
Requirement already satisfied: requests in c:\programdata\anaconda3\lib\site-packages (from kaggle) (2.25.1)
Requirement already satisfied: tqdm in c:\programdata\anaconda3\lib\site-packages (from kaggle) (4.59.0)
Requirement already satisfied: python-dateutil in c:\programdata\anaconda3\lib\site-packages (from kaggle) (2.8.1)
Requirement already satisfied: certifi in c:\programdata\anaconda3\lib\site-packages (from kaggle) (2020.12.5)
Requirement already satisfied: urllib3 in c:\programdata\anaconda3\lib\site-packages (from kaggle) (1.26.4)
Requirement already satisfied: text-unidecode>=1.3 in c:\programdata\anaconda3\lib\site-packages (from python-slugify->kaggle) (1.3)
Requirement already satisfied: idna<3,>=2.5 in c:\programdata\anaconda3\lib\site-packages (from requests->kaggle) (2.10)
Requirement already satisfied: chardet<5,>=3.0.2 in c:\programdata\anaconda3\lib\site-packages (from requests->kaggle) (4.0.0)
Installing collected packages: kaggle
Successfully installed kaggle-1.5.12
(base) C:\WINDOWS\system32>
```

2. Create Token API kaggle:



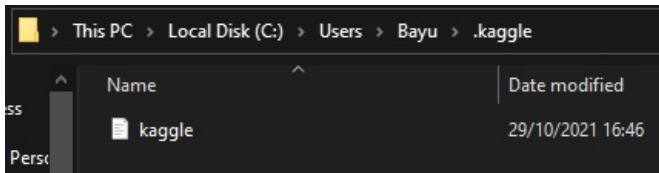
Home Competitions (1) Datasets Code (1) Discussion Followers Notifications Account

API

Using Kaggle's beta API, you can interact with Competitions and Datasets to download data, make submissions, and more. [Read the docs](#)

[Create New API Token](#) [Expire API Token](#)

1. Login Kaggle.com
2. Kemudian pada menu Profile --> Account
3. Klik Create New Api Token
4. Maka akan terdownload file kaggle.json



Kaggle API secara default mengasumsikan bahwa file kaggle.json tersebut berada di dalam folder:

- ~/.kaggle/ (Linux/Mac)
- C:\Users\kaggle\ (Windows)

Jika folder tersebut belum ada:

1. Buat folder di direktori C:\Users\kaggle\
2. letakkan file kaggle.json kedalam folder tersebut

3. Download Dataset dari Kaggle:

```
usage: kaggle datasets [-h]
                        {list,files,download,create,version,init,metadata,status} ...

optional arguments:
  -h, --help            show this help message and exit

commands:
  {list,files,download,create,version,init,metadata, status}
  list                  List available datasets
  files                 List dataset files
  download              Download dataset files
  create                Create a new dataset
  version               Create a new dataset version
  init                  Initialize metadata file for dataset creation
  metadata              Download metadata about a dataset
  status               Get the creation status for a dataset
```

Dokumentasi Kaggle Commands selengkapnya [Disini](#)

In [3]: `# Mencari dataset yang tersedia di kaggle --> pilih data provider dari UCIML`
`!kaggle datasets list -s Iris`

ref	downloadCount	voteCount	usabilityRating	title	size	lastUpdated
uciml/iris	227087	2681	0.7941176	Iris Species	4KB	201
6-09-27 07:38:05						
arshid/iris-flower-dataset	40680	371	0.8235294	Iris Flower Dataset	1010B	201
8-03-22 15:18:06						
vikrishnan/iris-dataset	2936	26	0.7647059	Iris Dataset	999B	201
7-08-03 16:00:44						
therohk/ireland-historical-news	2984	157	1.0	Irish Times - Waxy-Wany News	52MB	202
1-09-25 10:52:48						
chuckyin/iris-datasets	1775	14	0.7352941	Iris datasets	1KB	201
7-03-10 09:35:43						
rtatman/iris-dataset-json-version	5643	43	0.75	Iris Dataset (JSON Version)	1KB	201
8-04-06 20:21:31						
parulpandey/palmer-archipelago-antarctica-penguin-data	10122	116	0.9705882	Palmer Archipelago (Antarctica) penguin data	11KB	202
0-06-09 10:14:54						
conorrot/irish-weather-hourly-data	1872	40	0.8235294	Irish Weather (hourly data)	67MB	202
0-06-29 20:15:18						
saurabh00007/iriscsv	17190	58	0.4117647	Iris.csv	1KB	201
7-11-09 07:34:35						
jillanisofttech/iris-dataset-uci	39	12	1.0	Iris dataset uci	1KB	202
1-11-06 15:11:47						
fleanend/birds-songs-numeric-dataset	713	25	0.9411765	Birds' Songs Numeric Dataset	25MB	201
9-04-01 09:09:46						
kamrankausar/iris-data	1120	13	0.64705884	iris_data	1KB	201
7-11-30 10:26:01						
jeffheaton/iris-computer-vision	313	9	0.875	Iris Computer Vision	5MB	202
0-11-24 21:23:29						

styven/iris-dataset				Iris dataset	1KB	201
7-11-04 14:10:12	797	8	0.29411766			
arslanali4343/iris-species				Iris Species	2KB	202
0-07-02 06:09:09	62	13	0.5625			
olgabelitskaya/flower-color-images				Flower Color Images	50MB	202
0-10-01 22:48:07	8379	161	0.75			
naureenmohammad/mmu-iris-dataset				MMU iris dataset	30MB	202
0-07-25 18:38:33	646	19	0.5625			
rutujavaidya/iris-dataset				Iris Dataset	1KB	202
1-07-25 17:37:14	36	6	0.4117647			
shantanuss/iris-flower-dataset				IRIS flower dataset	1KB	202
0-01-18 19:43:18	200	3	0.9411765			
ashishs0ni/iris-dataset				Iris_dataset	1KB	201
8-08-05 14:26:19	603	7	0.64705884			

In [4]:

```
# Download dan ekstrak dataset, secara default akan berada dalam satu direktori dengan notebook ini
!kaggle datasets download uciml/iris --unzip
```

Downloading iris.zip to C:\Users\bunsm\Desktop\Data Science\Code Python\1636807287718_Tugas-Mandiri-Pert-5

```
0%|          | 0.00/3.60k [00:00<?, ?B/s]
100%|#####| 3.60k/3.60k [00:00<00:00, 409kB/s]
```

Atau bisa juga menggunakan link dari kaggle

Latihan (1)

Silahkan Download sebuah dataset menggunakan API Kaggle

In [5]:

```
#Latihan (1)
#Langkah nya seperti contoh diatas
```

PENGUNAAN LIBRARY PANDAS dan NUMPY

Pada materi ini, peserta sudah mendapatkan pemahaman mengenai data dan dataset. Penggunaan library pada Python memberikan kemudahan dalam proses data understanding. Beberapa library yang digunakan adalah library Pandas dan Numpy.

Latihan (2)

Lakukan import Library Pandas dan Library Numpy

In [6]:

```
#Latihan(2)
#Import Library Pandas

import pandas as pd

#Import Library Numpy

import numpy as np
```

DATAFRAME

DataFrame adalah struktur data 2 dimensi yang berbentuk tabular (mempunyai baris dan kolom). Hampir semua data tidak hanya memiliki 1 kolom tetapi lebih dari 1 kolom, sehingga lebih cocok menggunakan pandas DataFrame untuk mengolahnya.

Penggunaan dataframe pada Python dengan menggunakan syntaks: df.

Latihan (3)

Panggil file (load dataset) dengan format .csv untuk dataset mengenai bunga Iris yang sudah peserta unduh dari Kaggle, dan akan disimpan di dalam dataframe df. Lalu tampilkan 5 baris awal dataset dengan function head()

In [7]:

```
#latihan(3)
#Panggil file (load file bernama Iris.csv) dan simpan dalam dataframe Lalu tampilkan 5 baris awal dataset dengan

dataset ='iris.csv'
df = pd.read_csv(dataset)
df.head()
```

```
Out[7]:
```

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	1	5.1	3.5	1.4	0.2	Iris-setosa
1	2	4.9	3.0	1.4	0.2	Iris-setosa
2	3	4.7	3.2	1.3	0.2	Iris-setosa
3	4	4.6	3.1	1.5	0.2	Iris-setosa
4	5	5.0	3.6	1.4	0.2	Iris-setosa

Telaah Data

Pada telaah data, dapat dilakukan untuk mengetahui:

- tipe data dari setiap kolom
- deskripsi statistik data

Latihan (4)

Tampilkan tipe data dari kolom yang ada pada dataset

```
In [8]: #latihan(4)
#Tampilkan tipe data dari kolom yang ada pada dataset

print(df.dtypes)
```

```
Id                int64
SepalLengthCm    float64
SepalWidthCm     float64
PetalLengthCm    float64
PetalWidthCm     float64
Species          object
dtype: object
```

Latihan (5)

Apakah tipe Data dari kolom berikut ini: (silakan diisi pada cell di bawah ini)

```
In [9]: #Latihan (5)
#Tipe Data dari kolom yang ada di dataset
print(df.dtypes)
#Kolom "Id" memiliki tipe data = <isikan jawaban di sini>
print(df.dtypes.Id)
#Kolom "SepalLengthCm" memiliki tipe data = <isikan jawaban di sini>
print(df.dtypes.SepalLengthCm)
#Kolom "Species" memiliki tipe data = <isikan jawaban di sini>
print(df.dtypes.Species)
```

```
Id                int64
SepalLengthCm    float64
SepalWidthCm     float64
PetalLengthCm    float64
PetalWidthCm     float64
Species          object
dtype: object
int64
float64
object
```

Latihan (6)

Hitunglah ukuran (jumlah baris dan kolom) dari dataset. Dengan menggunakan method function

```
In [10]: #Latihan (6)
#Hitung ukuran (jumlah baris dan kolom) dari dataset
df.shape
```

```
Out[10]: (150, 6)
```

Latihan (7)

Berapakah jumlah baris, dan jumlah kolom pada dataset? (silakan diisi pada cell di bawah ini)

```
In [11]: #Latihan (7)

#Jumlah Baris pada dataset adalah = <isikan jawaban di sini>
print(df.shape[0])

#Jumlah kolom pada dataset adalah = <isikan jawaban di sini>
print(df.shape[1])
```

```
150
6
```

Latihan (8)

Tampilkan data yang hanya berisi kolom "Id" dan kolom "Species" dalam bentuk dataframe.

```
In [12]: #Latihan (8)
#Tampilkan data untuk kolom "Id" dan kolom "Species" dalam bentuk dataframe

df[['Id', 'Species']]
```

```
Out[12]:
```

	Id	Species
0	1	Iris-setosa
1	2	Iris-setosa
2	3	Iris-setosa
3	4	Iris-setosa
4	5	Iris-setosa
...
145	146	Iris-virginica
146	147	Iris-virginica
147	148	Iris-virginica
148	149	Iris-virginica
149	150	Iris-virginica

150 rows × 2 columns

Latihan (9)

Tampilkan data dengan dataframe, dan data yang ditampilkan adalah data pada baris dengan indeks 0 (nol) sampai dengan indeks 9 (sembilan)

```
In [13]: #Latihan (9)
#Tampilkan data dengan dataframe, dan data yang ditampilkan adalah baris dengan indeks 0 (nol) sampai dengan indeks 9
df_noid = df.iloc[0:10]
df_noid
```

```
Out[13]:
```

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
--	----	---------------	--------------	---------------	--------------	---------

0	1	5.1	3.5	1.4	0.2	Iris-setosa
1	2	4.9	3.0	1.4	0.2	Iris-setosa
2	3	4.7	3.2	1.3	0.2	Iris-setosa
3	4	4.6	3.1	1.5	0.2	Iris-setosa
4	5	5.0	3.6	1.4	0.2	Iris-setosa
5	6	5.4	3.9	1.7	0.4	Iris-setosa
6	7	4.6	3.4	1.4	0.3	Iris-setosa
7	8	5.0	3.4	1.5	0.2	Iris-setosa
8	9	4.4	2.9	1.4	0.2	Iris-setosa
9	10	4.9	3.1	1.5	0.1	Iris-setosa

Latihan (10)

Tampilkan data hanya kolom "Id" dan kolom "Species" dengan dataframe, dan yang ditampilkan adalah data pada baris dengan indeks 11 (sebelas) sampai dengan indeks 15 (limabelas)

```
In [14]: #Latihan (10)
#Tampilkan data hanya kolom "Id" dan kolom "Species", pada baris dengan indeks 0 (nol) sampai dengan indeks 9 (sebelas)
df_noid = df.iloc[11:16]
df_noid
```

```
Out[14]:
```

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
11	12	4.8	3.4	1.6	0.2	Iris-setosa
12	13	4.8	3.0	1.4	0.1	Iris-setosa
13	14	4.3	3.0	1.1	0.1	Iris-setosa
14	15	5.8	4.0	1.2	0.2	Iris-setosa
15	16	5.7	4.4	1.5	0.4	Iris-setosa

Latihan (11)

Pada DataFrame dapat menampilkan beberapa baris pertama/terakhir dari dataset yang di load. Gunakan Method head() dan tail().

Latihan: Tampilkan data pada 8 (delapan) baris pertama dari dataset, dengan dataframe.

```
In [15]: #Latihan (11)
#Tampilkan data pada 8 (delapan) baris pertama dari dataset, dengan dataframe
df.head(8)
```

```
Out[15]:
```

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	1	5.1	3.5	1.4	0.2	Iris-setosa
1	2	4.9	3.0	1.4	0.2	Iris-setosa
2	3	4.7	3.2	1.3	0.2	Iris-setosa
3	4	4.6	3.1	1.5	0.2	Iris-setosa
4	5	5.0	3.6	1.4	0.2	Iris-setosa
5	6	5.4	3.9	1.7	0.4	Iris-setosa
6	7	4.6	3.4	1.4	0.3	Iris-setosa
7	8	5.0	3.4	1.5	0.2	Iris-setosa

Latihan (12)

Tampilkan data pada 3 (tiga) baris terakhir dari dataset, dengan dataframe.

```
In [17]: #Latihan (12)
```

```
#Tampilkan data pada 3 (tiga) baris terakhir dari dataset, dengan dataframe
```

```
df.tail(3)
```

Out[17]:

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
147	148	6.5	3.0	5.2	2.0	Iris-virginica
148	149	6.2	3.4	5.4	2.3	Iris-virginica
149	150	5.9	3.0	5.1	1.8	Iris-virginica

Deskripsi Statistik Data

DataFrame method describe() menampilkan statistik dasar setiap kolom data yang bertipe numerik, mencakup banyaknya data (count), rerata aritmetik (mean), simpangan baku (std), nilai terkecil (min), kuartil pertama (25%), kuartil kedua/median (50%), kuartil ketiga (75%), dan nilai terbesar (max).

Latihan (13)

Hitung korelasi dari dataset. Dengan menggunakan method function

In [18]:

```
#Latihan (13)
#Hitung korelasi dataset

df.corr()
```

Out[18]:

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm
Id	1.000000	0.716676	-0.397729	0.882747	0.899759
SepalLengthCm	0.716676	1.000000	-0.109369	0.871754	0.817954
SepalWidthCm	-0.397729	-0.109369	1.000000	-0.420516	-0.356544
PetalLengthCm	0.882747	0.871754	-0.420516	1.000000	0.962757
PetalWidthCm	0.899759	0.817954	-0.356544	0.962757	1.000000

Latihan (14)

Berdasarkan pada perhitungan korelasi di Latihan (11), apakah yang dapat Bapak/Ibu simpulkan sementara? Silakan tuliskan simpulan sementara Bapak/Ibu pada cell di bawah ini.

In [19]:

```
#latihan (14)
#Simpulan Sementara Hasil Korelasi di latihan (13)

print('Ada beberapa yang minus menandakan korelasi negatif dan positif menandakan korelasi positif')
```

Ada beberapa yang minus menandakan korelasi negatif dan positif menandakan korelasi positif

Latihan (15)

Hitung korelasi untuk kolom berikut ini: PetalLengthCm, PetalWidthCm

In [20]:

```
#Latihan (15)
#Hitung korelasi dataset untuk kolom PetalLengthCm, PetalWidthCm
#df[['PetalLengthCm', 'PetalWidthCm']].corr()

df.corr()['PetalLengthCm']['PetalWidthCm']
```

Out[20]: 0.9627570970509662

Latihan (16)

Latihan (16)

Method "describe" secara otomatis melakukan komputasi statistik untuk semua continous variable. Secara default "describe" melakukan ignore terhadap variabel bertipe objek.

Komputasi statistik yang dilakukan terdiri dari: count, mean, std, min, max, 25%, 75%, max.

Latihan: Gunakan method describe pada dataset yang sudah di load untuk semua continous variabel. (Dataset Iris.csv)

```
In [21]: #Latihan (16)
# Penggunaan Metode describe untuk komputasi statistik
df.describe()
```

```
Out[21]:
```

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm
count	150.000000	150.000000	150.000000	150.000000	150.000000
mean	75.500000	5.843333	3.054000	3.758667	1.198667
std	43.445368	0.828066	0.433594	1.764420	0.763161
min	1.000000	4.300000	2.000000	1.000000	0.100000
25%	38.250000	5.100000	2.800000	1.600000	0.300000
50%	75.500000	5.800000	3.000000	4.350000	1.300000
75%	112.750000	6.400000	3.300000	5.100000	1.800000
max	150.000000	7.900000	4.400000	6.900000	2.500000

Latihan (17)

Gunakan method describe pada dataset yang sudah di load untuk data bertipe objek. (Dataset Iris.csv)

```
In [44]: #Latihan (17)
#Gunakan method describe pada dataset yang sudah di load untuk data bertipe objek
df.describe()
```

```
Out[44]:
```

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm
count	150.000000	150.000000	150.000000	150.000000	150.000000
mean	75.500000	5.843333	3.054000	3.758667	1.198667
std	43.445368	0.828066	0.433594	1.764420	0.763161
min	1.000000	4.300000	2.000000	1.000000	0.100000
25%	38.250000	5.100000	2.800000	1.600000	0.300000
50%	75.500000	5.800000	3.000000	4.350000	1.300000
75%	112.750000	6.400000	3.300000	5.100000	1.800000
max	150.000000	7.900000	4.400000	6.900000	2.500000

Latihan 18

Gunakan method describe pada dataset yang sudah di load untuk semua type data (continous variabel dan type object).

```
In [22]: #Latihan (18)
#Gunakan method describe pada dataset yang sudah di load untuk semua type data
df.describe(include = 'all')
```

```
Out[22]:
```

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
count	150.000000	150.000000	150.000000	150.000000	150.000000	150
unique	NaN	NaN	NaN	NaN	NaN	3
top	NaN	NaN	NaN	NaN	NaN	Iris-virginica
freq	NaN	NaN	NaN	NaN	NaN	50
mean	75.500000	5.843333	3.054000	3.758667	1.198667	NaN
std	43.445368	0.828066	0.433594	1.764420	0.763161	NaN
min	1.000000	4.300000	2.000000	1.000000	0.100000	NaN

25%	38.250000	5.100000	2.800000	1.600000	0.300000	NaN
50%	75.500000	5.800000	3.000000	4.350000	1.300000	NaN
75%	112.750000	6.400000	3.300000	5.100000	1.800000	NaN
max	150.000000	7.900000	4.400000	6.900000	2.500000	NaN

Latihan (19)

Hitunglah nilai mean dari dataset.

```
In [23]: #Latihan (19)
#Hitung nilai Mean dari dataset

df.mean()
```

```
Out[23]: Id                75.500000
SepalLengthCm          5.843333
SepalWidthCm           3.054000
PetalLengthCm          3.758667
PetalWidthCm           1.198667
dtype: float64
```

Latihan (20)

Hitung nilai mean dari dataset untuk kolom PetalLengthCm.

```
In [24]: #Latihan (20)
#Hitung nilai Mean untuk kolom PetalLengthCm

df['PetalLengthCm'].mean()
```

```
Out[24]: 3.75866666666666693
```

Latihan (21)

Carilah nilai minimal dari dataset untuk kolom SepalWidthCm.

```
In [25]: #Latihan (21)
#Cari nilai minimal untuk kolom SepalWidthCm

df['SepalWidthCm'].min()
```

```
Out[25]: 2.0
```

Method Groupby

Method groupby memungkinkan analisis dilakukan secara per kelompok nilai atribut tertentu.

Latihan (22)

Hitunglah nilai mean dari dataset untuk kolom SepalLengthCm per Species dengan menggunakan metode groupby.

```
In [26]: #Latihan (22)
#Hitung nilai mean dari dataset untuk SepalLengthCm per Species dengan metode groupby

df.groupby('Species')['SepalLengthCm'].mean()
```

```
Out[26]: Species
Iris-setosa      5.006
Iris-versicolor  5.936
Iris-virginica   6.588
Name: SepalLengthCm, dtype: float64
```

Method Value Count

value_counts() menghasilkan frekuensi setiap nilai unik di dalam kolom, dan yang tertinggi count-nya adalah merupakan modus pada kolom tersebut.

Latihan (23)

Hitunglah frekuensi pada kolom 'Species' dengan menggunakan metode value_counts().

```
In [27]: #Latihan (23)
#Hitung frekuensi pada kolom 'Species' dengan menggunakan metode value_counts()

df['Species'].value_counts()
```

```
Out[27]: Iris-virginica      50
Iris-versicolor      50
Iris-setosa           50
Name: Species, dtype: int64
```

Latihan (24)

Tampilkan perhitungan frekuensi pada kolom 'Species' dengan menggunakan metode value_counts() dalam bentuk dataframe.

```
In [28]: #Latihan (24)
#Perhitungan frekuensi pada kolom 'Species' dengan menggunakan metode value_counts() dalam bentuk dataframe

df['Species'].value_counts()
```

```
Out[28]: Iris-virginica      50
Iris-versicolor      50
Iris-setosa           50
Name: Species, dtype: int64
```

Latihan (25)

Hitunglah frekuensi pada kolom 'PetalLenghCm' dengan menggunakan metode value_counts() dan dalam bentuk dataframe.

```
In [52]: #Latihan (25)
# Hitung frekuensi pada kolom 'PetalLenghCm' dengan menggunakan metode value_counts()

df['PetalLenghCm'].value_counts()
```

```
Out[52]: 1.5      14
1.4      12
5.1       8
4.5       8
1.6       7
1.3       7
5.6       6
4.0       5
4.9       5
4.7       5
5.0       4
1.7       4
4.8       4
4.4       4
4.2       4
4.1       3
```

5.7	3
5.5	3
6.1	3
3.9	3
4.6	3
5.8	3
5.2	2
1.9	2
6.0	2
1.2	2
4.3	2
5.3	2
5.4	2
3.3	2
6.7	2
3.5	2
5.9	2
3.6	1
3.8	1
1.0	1
3.0	1
6.3	1
6.6	1
3.7	1
1.1	1
6.4	1
6.9	1

Name: PetalLengthCm, dtype: int64

Loading [MathJax]/jax/output/CommonHTML/fonts/TeX/fontdata.js