

<https://www.linkedin.com/in/maylinnarahayuningsih/>

–python foundations–

## READ DATASET

```
1 from google.colab import drive
2 drive.mount('/content/drive')
```

Mounted at /content/drive

Isi Kolom :

No, Peserta, Nama, Jurusan, Berat Badan (Kg), Tinggi Badan (Cm)

```
1 import pandas as pd
2
3 # baca file
4 df = pd.read_csv('/content/drive/MyDrive/Startupcampus/Data Assignment Kompi G - Sheet1.csv')
5
6 # Kolom yang akan ditampilkan
7 df_selected = df[['No Peserta', 'Jurusan', 'Berat Badan (Kg)', 'Tinggi Badan (Cm)']]
8
9 # Menampilkan hasil
10 df_selected
11
12 #nama tidak ditampilkan untuk menjaga privasi data
```

	No Peserta	Jurusan	Berat Badan (Kg)	Tinggi Badan (Cm)
0	DS5G09	Manajemen	88	178
1	DS5G16	Administrasi Bisnis	42	150
2	DS5G08	Statistika	55	158
3	DS5G02	Sistem Informasi	60	154
4	DS5G01	Informatika	60	153
5	DS5G05	Sains Data	45	150
6	DS5G23	Matematika	100	178
7	DS5G20	Teknik Informatika	36	150
8	DS5G13	Matematika	45	155
9	DS5G07	Teknik Informatika	50	173
10	DS5G14	Informatika	62	173
11	DS5G12	Informatika	50	156
12	DS5G04	Informatika	48	153
13	DS5G11	Teknik Informatika	56	169
14	DS5G21	Manajemen	49	156
15	DS5G17	Informatika	80	175
16	DS5G19	Matematika	50	149
17	DS5G22	Statistika	43	157
18	DS5G03	Teknik Informatika	70	161
19	DS5G18	Ilmu Ekonomi	44	151
20	DS5G06	Informatika	67	163
21	DS5G15	Informatika	60	173

## Understand the contents of the dataframe

### a. Tipe data

- No Peserta: tipe data **String** (karakter/alphanumeric) karena digunakan sebagai identifikasi atau kode.
- Nama: tipe data **string** karena berisi nama-nama orang.
- Jurusan: Tipe data **string**, berisi nama-nama jurusan atau program studi.
- Berat Badan (Kg): Tipe data bilangan bulat (**integer**) berisi angka-angka yang mewakili berat badan dalam kg
- Tinggi Badan (Cm): Tipe data bilangan bulat (**integer**)berisi angka-angka yang mewakili tinggi badan dalam cm

b jumlah kolom = 23

c. jumlah baris = 5

### d. informasi yang terkandung

Dataset/ Dataframe berisi informasi peserta KOMPI G yang terdiri dari beberapa hal

- No Peserta: nomor identifikasi atau kode unik untuk setiap Peserta dengan default DS5G.
- Nama: Nama lengkap dari peserta kompi G
- Jurusan: Informasi tentang jurusan atau program studi peserta
- Berat Badan (Kg): Data berat badan peserta dalam kilogram(Kg)
- Tinggi Badan (Cm): Data tinggi badan peserta dalam sentimeter(Cm)

## 2. Hitung BMI

```
1 # Import perpustakaan pandas
2 import pandas as pd
3
4 # Baca file dataset
5 df = pd.read_csv('/content/drive/MyDrive/Startupcampus/Data Assignment Kompi G - Sheet1.csv')
6
```

```
7 # Menghitung BMI
8 #note; ['Tinggi Badan (Cm)'] / 100) ----> mengubah Cm ke M karena BMI dalam m(meter)
9
10 df['BMI'] = df['Berat Badan (Kg)'] / ((df['Tinggi Badan (Cm)'] / 100) ** 2)
11
12 # Kolom yang akan ditampilkan
13 df_selected = df[['Nama', 'BMI']]
14
15 # Menampilkan hasil
16 df_selected
17
```

	Nama	BMI
0	Muhamad Fauzi	27.774271
1	Silvia Anisah Lubis	18.666667
2	Bitra Azizy Hamida	22.031726
3	Rih Prajna Nurfatikha	25.299376
4	Fayza Aulia	25.631167
5	Nada Salsabila	20.000000
6	Amos Hatoguan Sihombing	31.561672
7	Anjar Ayuning Lestari	16.000000
8	Rizka Rahmikasari	18.730489
9	Rachmat Faisal Manurung	16.706205
10	Aji Saeful	20.715694
11	Tarida Grace Wahyuni Margaretha Sidabutar	20.545694
12	Fitria Rahmadani	20.504934
13	Muhammad Ilham Pradipta	19.607157
14	Nurul Jumnatha Fajar Nofitasari	20.134780
15	Daffa Rayhan Fadhillah	26.122449
16	Maqhfirotul Khusna	22.521508
17	Oktaviani Esty Susanti	17.444927
18	Maylinna Rahayu Ningsih	27.005131
19	Rafifah Rabbani	19.297399
20	Faiz Fathoni	25.217359
21	Muhammad Cahyo Nugroho	20.047446



Program for BMI!

KLASIFIKASI	BMI (Kg/m²) Principal cut-off points
UNDERWEIGHT	< 18.50
Severe thinness	< 16.00
Moderate thinness	16.00 – 16.99
Mild thinness	17.00 – 18.49
NORMAL	18.50 – 24.99
OVERWEIGHT	≥ 25.00
Pre-obesitas	25.00 – 29.99
OBESITAS	≥ 30.00
Obesitas Klas I	30.00 – 34.99
Obesitas Klas II	35.00 – 39.99
Obesitas Klas III	≥ 40.00

Source: Adapted from WHO, 1995, WHO, 2000 and WHO 2004.  
www.andaka.com

```
1 # Fungsi untuk menghitung BMI
2 def hitung_bmi(berat_badan_kg, tinggi_badan_cm):
3     # Menghitung BMI dengan rumus
4     bmi = berat_badan_kg / ((tinggi_badan_cm / 100) ** 2)
5     return bmi
6
7 # Input berat badan dan tinggi badan
8 berat_badan = float(input("Masukkan berat badan (kg): "))
9 tinggi_badan = float(input("Masukkan tinggi badan (cm): "))
10
11 # Memanggil fungsi hitung_bmi
12 bmi = hitung_bmi(berat_badan, tinggi_badan)
13
14 # Menampilkan hasil BMI
15 print("BMI Anda adalah:", bmi)
16
17 # Menampilkan kategori BMI
18 if bmi < 18.5:
19     print("Kategori BMI: Kurang berat badan")
20 elif bmi >= 18.5 and bmi < 24.9:
21     print("Kategori BMI: Normal")
22 elif bmi >= 25.0 and bmi < 29.9:
23     print("Kategori BMI: Kelebihan berat badan")
24 else:
25     print("Kategori BMI: Obesitas")
26
```

Masukkan berat badan (kg): 68  
Masukkan tinggi badan (cm): 165  
BMI Anda adalah: 24.977043158861342  
Kategori BMI: Obesitas

▼ 5. Status for BMI

- Normal: 18.5-24.9
- Overweight: 25.0-29.9
- Obesity: 30.0-34.9
- Underweight: < 18.5

```
1 import pandas as pd
2
3 # Membaca data awal dari file CSV
4 file_path = '/content/drive/MyDrive/Startupcampus/Data Assignment Kompi G - Sheet1.csv'
5 df = pd.read_csv(file_path)
6
7 # Menghitung BMI dan menambahkannya sebagai kolom baru
8 df['BMI'] = df['Berat Badan (Kg)'] / ((df['Tinggi Badan (Cm)'] / 100) ** 2)
9
10 # Fungsi untuk mengkategorikan status BMI
11 def categorize_bmi(bmi):
12     if bmi < 18.5:
13         return 'Kurang berat badan'
14     elif 18.5 <= bmi < 24.9:
15         return 'Normal'
16     elif 25.0 <= bmi < 29.9:
17         return 'Kelebihan berat badan'
18     else:
19         return 'Obesitas'
20
21 # Menambahkan kolom "Status BMI" dengan status yang sesuai
22 df['Status BMI'] = df['BMI'].apply(categorize_bmi)
23
24 # Kolom yang akan ditampilkan
25 df_selected = df[['No Peserta', 'BMI']]
26
27 # Menampilkan dataframe dengan kolom "Status BMI"
28 df_selected
29
```

	No Peserta	BMI
0	DS5G09	27.774271
1	DS5G16	18.666667
2	DS5G08	22.031726
3	DS5G02	25.299376
4	DS5G01	25.631167
5	DS5G05	20.000000
6	DS5G23	31.561672
7	DS5G20	16.000000
8	DS5G13	18.730489
9	DS5G07	16.706205
10	DS5G14	20.715694
11	DS5G12	20.545694
12	DS5G04	20.504934
13	DS5G11	19.607157
14	DS5G21	20.134780
15	DS5G17	26.122449
16	DS5G19	22.521508
17	DS5G22	17.444927
18	DS5G03	27.005131
19	DS5G18	19.297399
20	DS5G06	25.217359
21	DS5G15	20.047446

▼ Sort By Height

```
1 # jawaban a
2
3 import pandas as pd
4
5 # Membaca data awal dari file CSV
6 file_path = '/content/drive/MyDrive/Startupcampus/Data Assignment Kompi G - Sheet1_with_BMI_and_Status.csv'
7 df = pd.read_csv(file_path)
8
9 # Mengurutkan data berdasarkan tinggi badan dari tertinggi ke terendah
10 df_sorted_by_height = df.sort_values(by='Tinggi Badan (Cm)', ascending=False)
11
12 # Kolom yang akan ditampilkan
13 df_selected = df_sorted_by_height[['No Peserta', 'Tinggi Badan (Cm)', 'BMI']]
14
15 # Menampilkan dataframe dengan kolom "Status BMI"
16 df_selected
```

No Peserta		Tinggi Badan (Cm)	BMI
0	DS5G09	178	27.774271
6	DS5G23	178	31.561672
15	DS5G17	175	26.122449
10	DS5G14	173	20.715694
9	DS5G07	173	16.706205
21	DS5G15	173	20.047446
13	DS5G11	169	19.607157
20	DS5G06	163	25.217359
18	DS5G03	161	27.005131
2	DS5G08	158	22.031726
17	DS5G22	157	17.444927
14	DS5G21	156	20.134780

Sort By name

3	DS5G02	154	25.299376
---	--------	-----	-----------

```
1 import pandas as pd
2
3 # Membaca data awal dari file CSV
4 file_path = '/content/drive/MyDrive/Startupcampus/Data Assignment Kompi G - Sheet1_with_BMI_and_Status.csv'
5 df = pd.read_csv(file_path)
6
7 # Mengurutkan data berdasarkan Nama dari Z ke A
8 df_sorted_by_name = df.sort_values(by='Nama', ascending=False)
9
10 # Kolom yang akan ditampilkan
11 df_selected = df_sorted_by_name[['No Peserta', 'Nama', 'BMI']]
12
13 # Menampilkan dataframe dengan kolom "Status BMI"
14 df_selected
15
```

No Peserta		Nama	BMI
11	DS5G12	Tarida Grace Wahyuni Margaretha Sidabutar	20.545694
1	DS5G16	Silvia Anisah Lubis	18.666667
8	DS5G13	Rizka Rahmikasari	18.730489
3	DS5G02	Rih Prajna Nurfatikha	25.299376
19	DS5G18	Rafifah Rabbani	19.297399
9	DS5G07	Rachmat Faisal Manurung	16.706205
17	DS5G22	Oktaviani Esty Susanti	17.444927
14	DS5G21	Nurul Jumnatha Fajar Nofitasari	20.134780
5	DS5G05	Nada Salsabila	20.000000
13	DS5G11	Muhammad Ilham Pradipta	19.607157
21	DS5G15	Muhammad Cahyo Nugroho	20.047446
0	DS5G09	Muhamad Fauzi	27.774271
18	DS5G03	Maylinna Rahayu Ningsih	27.005131
16	DS5G19	Maqhfirotul Khusna	22.521508
12	DS5G04	Fitria Rahmadani	20.504934
4	DS5G01	Fayza Aulia	25.631167
20	DS5G06	Faiz Fathoni	25.217359
15	DS5G17	Daffa Rayhan Fadhillah	26.122449
2	DS5G08	Bitah Azizy Hamidah	22.031726
7	DS5G20	Anjar Ayuning Lestari	16.000000
6	DS5G23	Amos Hatoguan Sihombing	31.561672
10	DS5G14	Aji Saeful	20.715694



BMI Status outside Normal

```
1 # a. Filter dataframe df berdasarkan: Status BMI di luar Normal.
2
3 # Filter dataframe untuk Status BMI di luar Normal
4 df_filtered_a = df[df['Status BMI'] != 'Normal']
5
6 # Kolom yang akan ditampilkan
7 df_selected = df_filtered_a[['No Peserta', 'BMI', 'Status BMI']]
8
9 # Menampilkan dataframe dengan kolom "Status BMI"
10 df_selected
```

No Peserta	BMI	Status BMI
0	DS5G09 27.774271	Kelebihan berat badan
3	DS5G02 25.299376	Kelebihan berat badan
4	DS5G01 25.631167	Kelebihan berat badan
6	DS5G23 31.561672	Obesitas
7	DS5G20 16.000000	Kurang berat badan
9	DS5G07 16.706205	Kurang berat badan
15	DS5G17 26.122449	Kelebihan berat badan
17	DS5G22 17.444927	Kurang berat badan
18	DS5G03 27.005131	Kelebihan berat badan
20	DS5G06 25.217359	Kelebihan berat badan

▼ Filter for Height > 169 cm and Weight > 60 kg

```
1 # Filter dataframe untuk Tinggi Badan > 169 cm dan Berat Badan > 60 kg
2 df_filtered_b = df[(df['Tinggi Badan (Cm)'] > 169) & (df['Berat Badan (Kg)'] > 60)]
3
4 # Kolom yang akan ditampilkan
5 df_selected = df_filtered_b[['No Peserta', 'Tinggi Badan (Cm)', 'Berat Badan (Kg)']]
6
7 # Menampilkan dataframe dengan kolom "Status BMI"
8 df_selected
```

	No Peserta	Tinggi Badan (Cm)	Berat Badan (Kg)	
0	DS5G09	178	88	
6	DS5G23	178	100	
10	DS5G14	173	62	
15	DS5G17	175	80	

▼ Filter datasets with names containing the letter 'y'

```
1 # Filter dataframe untuk Nama yang mengandung huruf "y"
2 df_filtered_c = df[df['Nama'].str.contains('y', case=False, na=False)]
3
4 # Kolom yang akan ditampilkan
5 df_selected = df_filtered_c[['Nama']]
6
7 # Menampilkan dataframe dengan kolom "Status BMI"
8 df_selected
```

	Nama	
2	Bitu Azizy Hamida	
4	Fayza Aulia	
7	Anjar Ayuning Lestari	
11	Tarida Grace Wahyuni Margaretha Sidabutar	
15	Daffa Rayhan Fadhilla	
17	Oktaviani Esty Susanti	
18	Maylinna Rahayu Ningsih	
21	Muhammad Cahyo Nugroho	

▼ Get Insight From data

```
1 # Kelompokkan tinggi badan dan berat badan serta cari nilai terendah dan tertinggi.
2
3 height_weight_stats = df.groupby(['Tinggi Badan (Cm)', 'Berat Badan (Kg)']).size().reset_index(name='Jumlah Siswa')
4 min_height_weight = height_weight_stats.min()
5 max_height_weight = height_weight_stats.max()
6
7 print("Nilai terendah tinggi badan dan berat badan:")
8 print(min_height_weight)
9 print("\nNilai tertinggi tinggi badan dan berat badan:")
10 print(max_height_weight)
```

```
Nilai terendah tinggi badan dan berat badan:
Tinggi Badan (Cm)    149
Berat Badan (Kg)     36
Jumlah Siswa         1
dtype: int64
```

```
Nilai tertinggi tinggi badan dan berat badan:
Tinggi Badan (Cm)    178
Berat Badan (Kg)     100
Jumlah Siswa         1
dtype: int64
```

```
1 # Hitung rata-rata tinggi badan
2 rata_rata_tinggi = df['Tinggi Badan (Cm)'].mean()
3 print(f"Rata-rata tinggi badan: {rata_rata_tinggi} cm")
4
```

```
Rata-rata tinggi badan: 160.6818181818182 cm
```

```
1 # Hitung rata-rata berat badan
2 rata_rata_berat = df['Berat Badan (Kg)'].mean()
3 print(f"Rata-rata berat badan: {rata_rata_berat} kg")
4
```

```
Rata-rata berat badan: 57.27272727272727 kg
```

```
1 # Hitung jumlah siswa
2 jumlah_siswa = len(df)
3 print(f"Jumlah siswa yang tercatat: {jumlah_siswa} orang")
4
```

```
Jumlah siswa yang tercatat: 22 orang
```

```
1 # 5 orang dengan nilai BMI tertinggi
2 df_sorted_by_bmi = df.sort_values(by='BMI', ascending=False)
3 lima_orang_tertinggi = df_sorted_by_bmi.head(5)
4 print("5 orang dengan nilai BMI tertinggi:")
5 lima_orang_tertinggi
```

5 orang dengan nilai BMI tertinggi:

No	Daerah	Nama	Turunan	Berat Badan (Kg)	Tinggi Badan (Cm)	BMI	Status BMI
1	#	Hitung jumlah orang yang tergolong status BMI					
2		jumlah_underweight = len(df[df['Status BMI'] == 'Kurang berat badan'])					
3		jumlah_normal = len(df[df['Status BMI'] == 'Normal'])					
4		jumlah_overweight = len(df[df['Status BMI'] == 'Kelebihan berat badan'])					
5		jumlah_obesity = len(df[df['Status BMI'] == 'Obesitas'])					
6							
7		print("Jumlah orang yang tergolong status:")					
8		print(f"Kurang berat badan: {jumlah_underweight} orang")					
9		print(f"Normal: {jumlah_normal} orang")					
10		print(f"Kelebihan berat badan: {jumlah_overweight} orang")					
11		print(f"Obesitas: {jumlah_obesity} orang")					
		Jumlah orang yang tergolong status:					
		Kurang berat badan: 3 orang					
		Normal: 12 orang					
		Kelebihan berat badan: 6 orang					
		Obesitas: 1 orang					