Tugas Capstone Bengkel Koding

Nama : Syallom Christian

NIM : A11.2022.14384

Link file all : https://drive.google.com/drive/folders/1b5TbkWHwxtVNNOFI72Z9qYzi8ObxFzeK?usp=sharing

Link github : https://github.com/syallomchristian/Capstone_Project_BengKod_DataScience

1. Import Lib

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np
```

2. Import Dataset

```
from google.colab import drive
drive.mount('/content/drive', force_remount=True)

import sys
sys.path.append('/content/drive/My Drive/Project_CAPSTONE_BengKod')

df = pd.read_csv('/content/drive/My Drive/Project_CAPSTONE_BengKod/ObesityDataSet.csv')
```

```
Mounted at /content/drive
```

```
df.info()
df.head()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2111 entries, 0 to 2110
Data columns (total 17 columns):
 #   Column                          Non-Null Count  Dtype
---  ------                          --------------  -----
 0   Age                             2097 non-null   object
 1   Gender                          2102 non-null   object
```

```
 2   Height                          2099 non-null   object
 3   Weight                          2100 non-null   object
 4   CALC                            2106 non-null   object
 5   FAVC                            2100 non-null   object
 6   FCVC                            2103 non-null   object
 7   NCP                             2099 non-null   object
 8   SCC                             2101 non-null   object
 9   SMOKE                           2106 non-null   object
 10  CH2O                            2105 non-null   object
 11  family_history_with_overweight  2098 non-null   object
 12  FAF                             2103 non-null   object
 13  TUE                             2102 non-null   object
 14  CAEC                            2100 non-null   object
 15  MTRANS                          2105 non-null   object
 16  NObeyesdad                      2111 non-null   object
dtypes: object(17)
memory usage: 280.5+ KB
```

	Age	Gender	Height	Weight	CALC	FAVC	FCVC	NCP	SCC	SMOKE	CH2O	family_h
0	21	Female	1.62	64	no	no	2	3	no	no	2	
1	21	Female	1.52	56	Sometimes	no	3	3	yes	yes	3	
2	23	Male	1.8	77	Frequently	no	2	3	no	no	2	
3	27	Male	1.8	87	Frequently	no	3	3	no	no	2	
4	22	Male	1.78	89.8	Sometimes	no	2	1	no	no	2	

Next steps: Generate code with df | View recommended plots | New interactive sheet

3. EDA

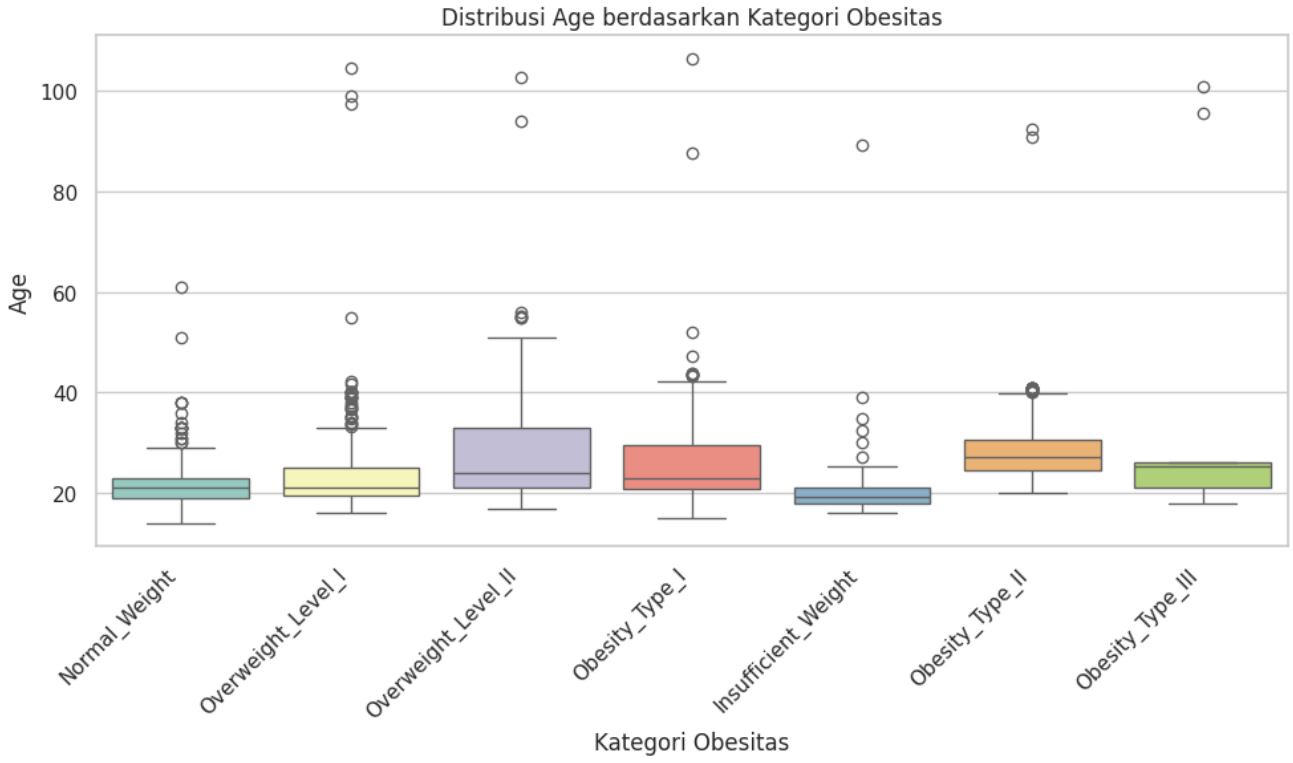
```
# Ringkasan informasi dataset
info = df.info()

# Cek nilai yang hilang
missing_values = df.isnull().sum()

# Statistik deskriptif untuk kolom numerik
desc_stats = df.describe()

info, missing_values, desc_stats
```

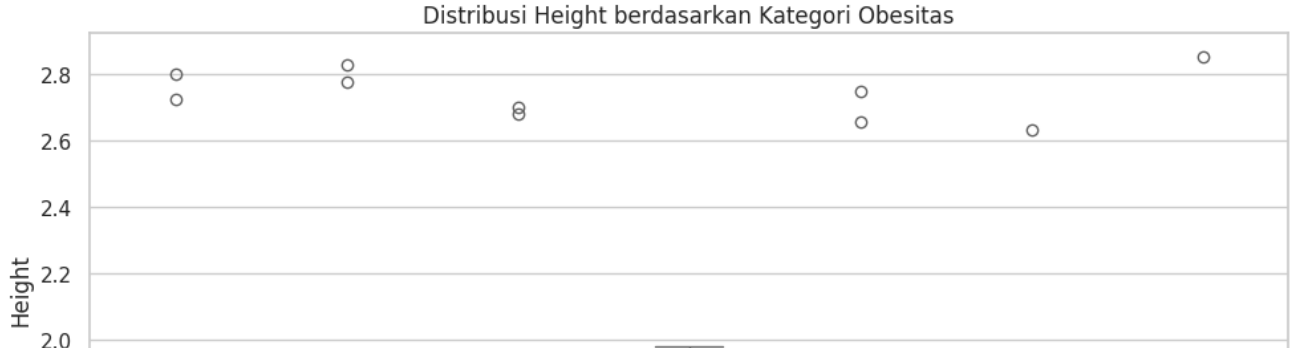
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2111 entries, 0 to 2110
Data columns (total 17 columns):
 #   Column                          Non-Null Count  Dtype
---  ------                          --------------  -----
```

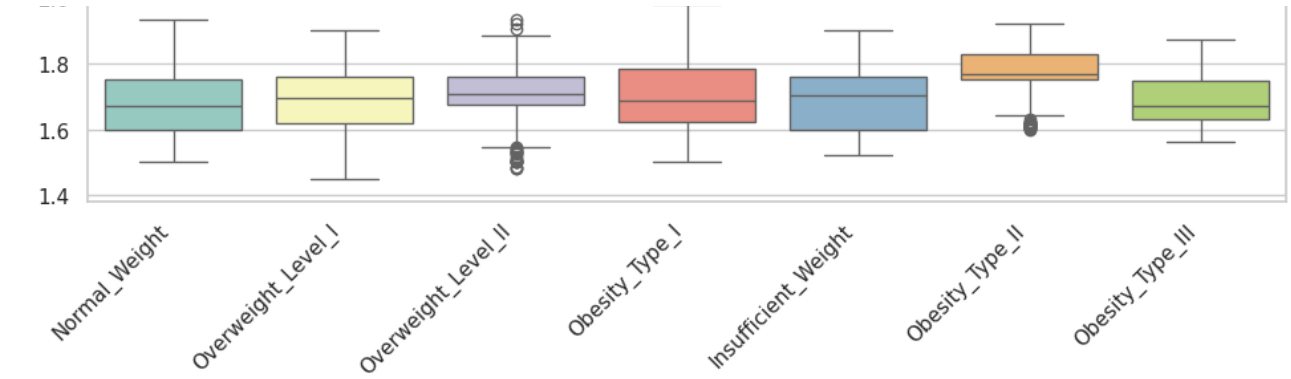
```
 0   Age                             2097 non-null   object
 1   Gender                          2102 non-null   object
 2   Height                          2099 non-null   object
 3   Weight                          2100 non-null   object
 4   CALC                            2106 non-null   object
 5   FAVC                            2100 non-null   object
 6   FCVC                            2103 non-null   object
 7   NCP                             2099 non-null   object
 8   SCC                             2101 non-null   object
 9   SMOKE                           2106 non-null   object
 10  CH2O                            2105 non-null   object
 11  family_history_with_overweight  2098 non-null   object
 12  FAF                             2103 non-null   object
 13  TUE                             2102 non-null   object
 14  CAEC                            2100 non-null   object
 15  MTRANS                          2105 non-null   object
 16  NObeyesdad                      2111 non-null   object
dtypes: object(17)
memory usage: 280.5+ KB
(None,
 Age                               14
 Gender                             9
 Height                            12
 Weight                            11
 CALC                               5
 FAVC                              11
 FCVC                               8
 NCP                               12
 SCC                               10
 SMOKE                              5
 CH2O                               6
 family_history_with_overweight    13
 FAF                                8
 TUE                                9
 CAEC                              11
 MTRANS                             6
 NObeyesdad                         0
 dtype: int64,
          Age Gender Height Weight       CALC  FAVC  FCVC   NCP   SCC SMOKE  \
 count   2097   2102   2099   2100       2106  2100  2103  2099  2101  2106
 unique  1394      3   1562   1518          5     3   808   637     3     3
 top       18   Male    1.7     80  Sometimes   yes     3     3    no    no
 freq     124   1056     58     58       1386  1844   647  1183  1997  2054

         CH2O family_history_with_overweight   FAF   TUE       CAEC  \
 count   2105                            2098  2103  2102       2100
 unique  1263                               3  1186  1130          5
 top        2                             yes     0     0  Sometimes
 freq     441                            1705   404   552       1747

                       MTRANS      NObeyesdad
 count                   2105            2111
 unique                     6               7
```

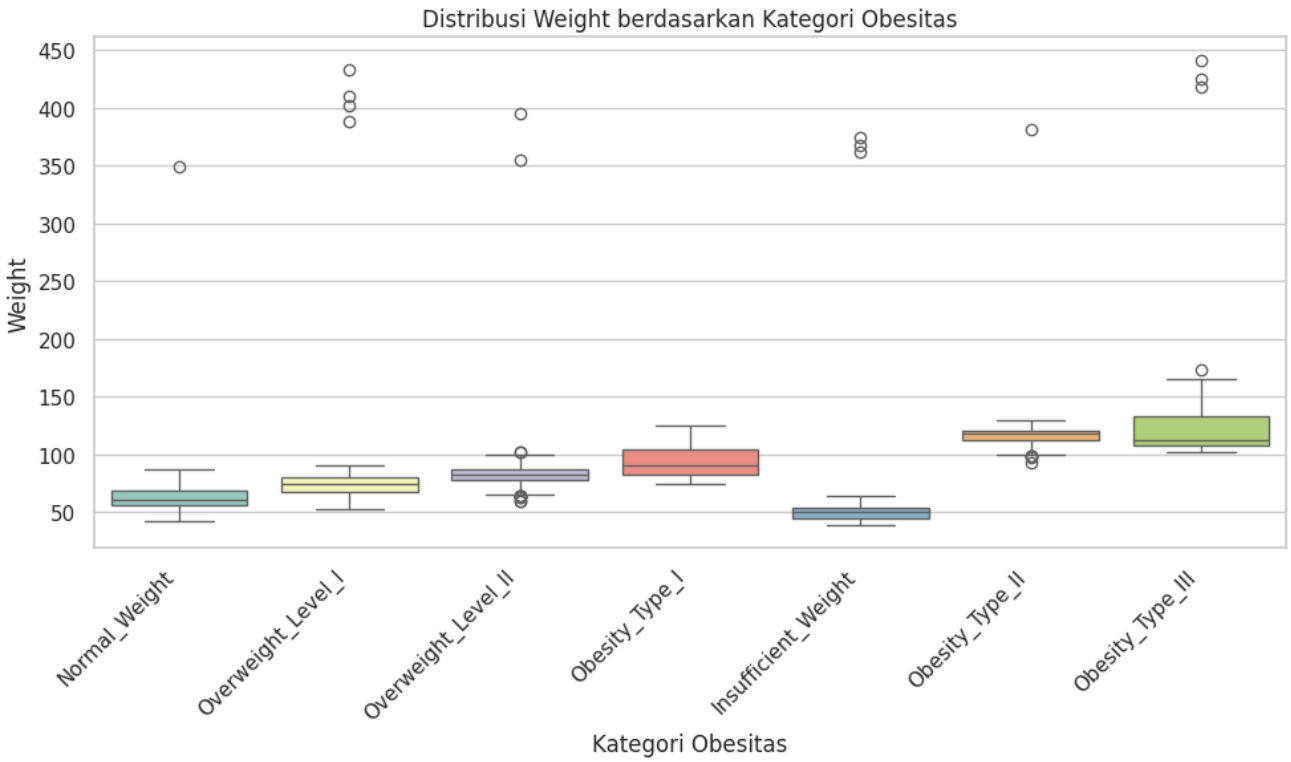
```
numerical_columns = ['Age', 'Height', 'Weight', 'FCVC', 'NCP', 'CH2O', 'FAF', 'TUE']
```

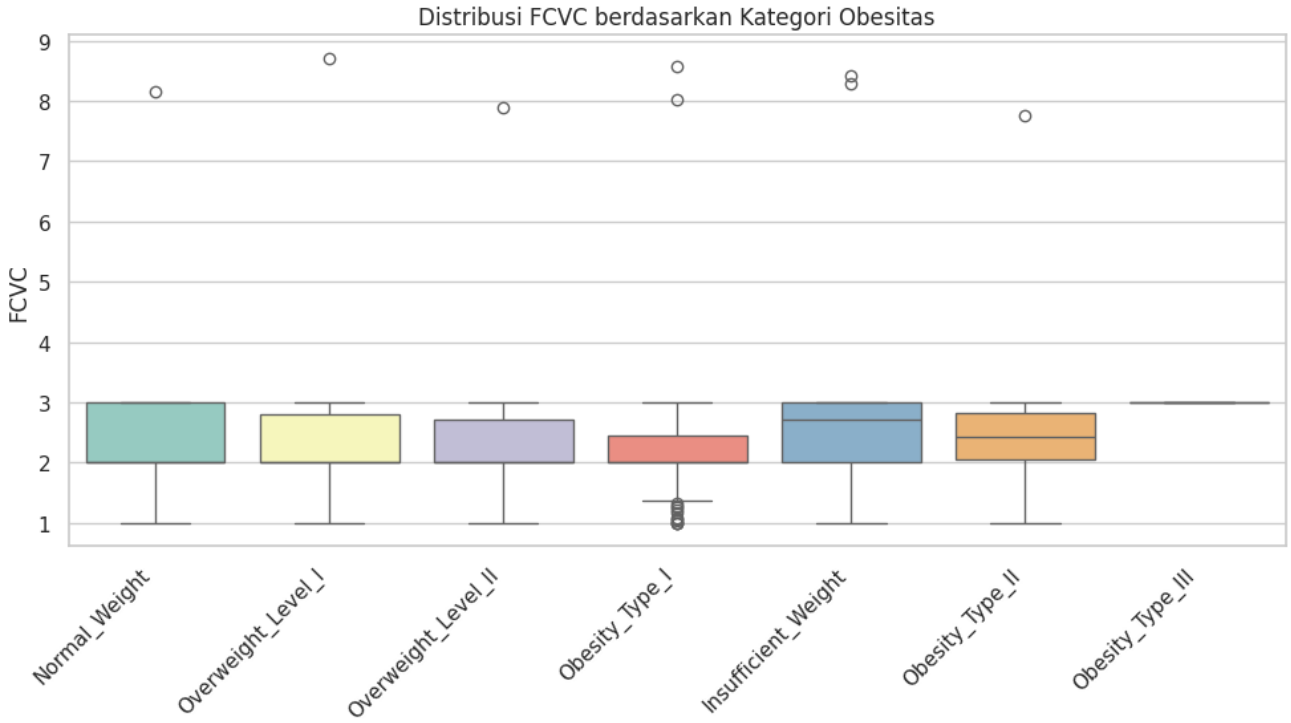
```
# Ubah nilai tidak valid menjadi NaN pada kolom numerik
for col in numerical_columns:
    df[col] = pd.to_numeric(df[col], errors='coerce')

# Hapus NaN
df_cleaned = df.dropna(subset=numerical_columns + ['NObeyesdad']) # Hapus baris dengan Na

# Buat boxplot
sns.set_theme(style="whitegrid")
for col in numerical_columns:
    plt.figure(figsize=(10, 6))
    sns.boxplot(x='NObeyesdad', y=col, hue='NObeyesdad', data=df_cleaned, palette='Set3',
    plt.title(f'Distribusi {col} berdasarkan Kategori Obesitas')
    plt.xlabel('Kategori Obesitas')
    plt.ylabel(col)
    plt.xticks(rotation=45, ha='right')      # Putar dan rapikan label
    plt.subplots_adjust(bottom=0.25)         # Tambah jarak bawah agar tidak bertabrakan
    plt.tight_layout()
    plt.show()
```

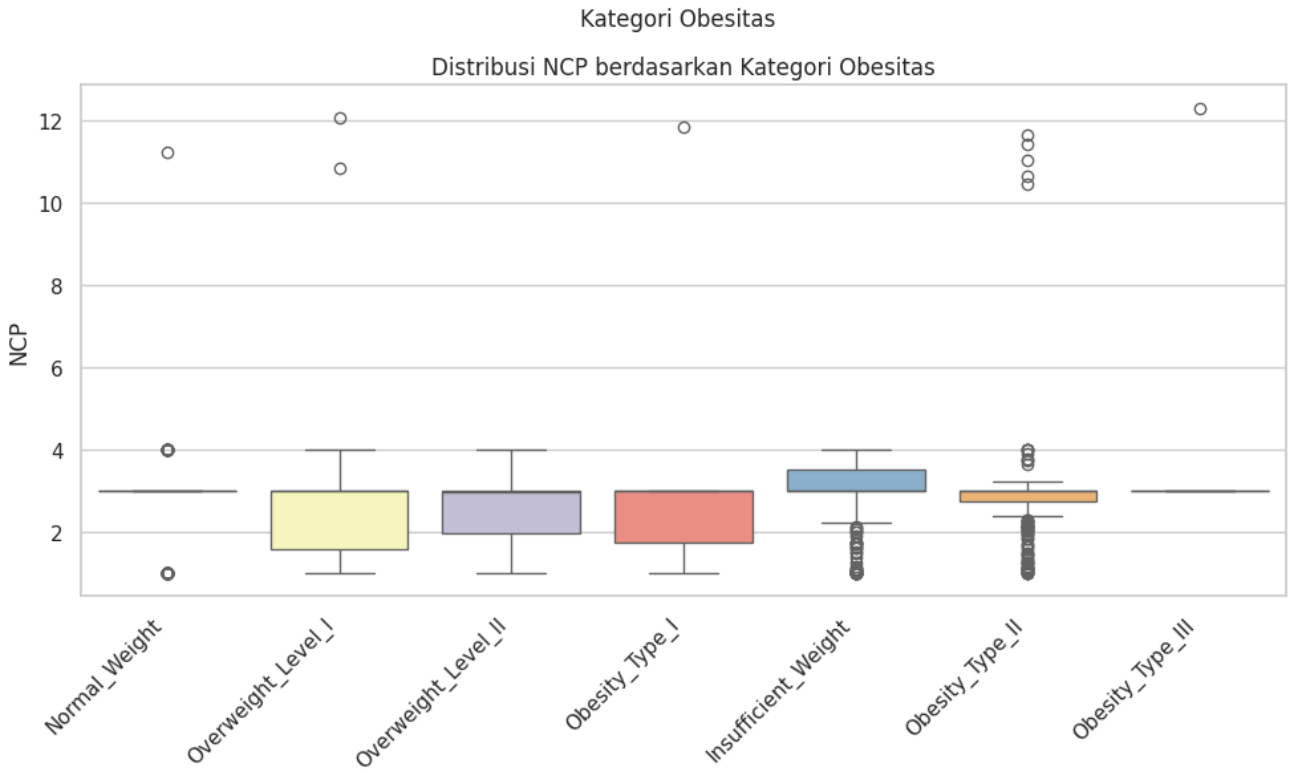


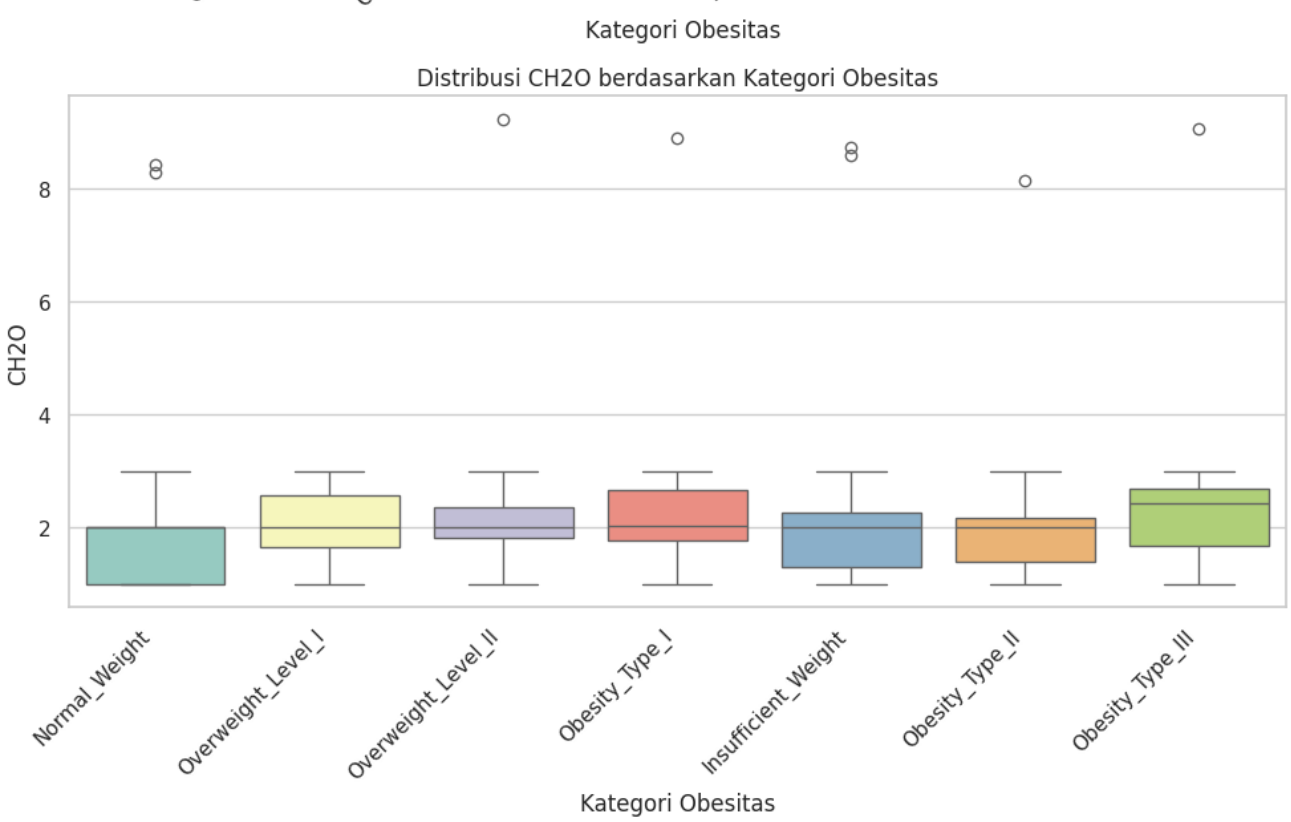


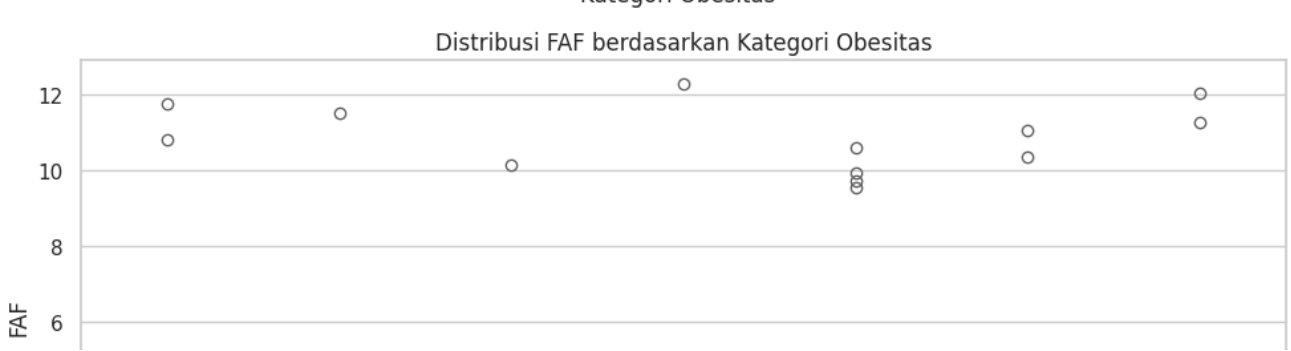

1.8
1.6
1.4
Normal_Weight
Overweight_Level_I
Overweight_Level_II
Obesity_Type_I
Insufficient_Weight
Obesity_Type_II
Obesity_Type_III
Kategori Obesitas

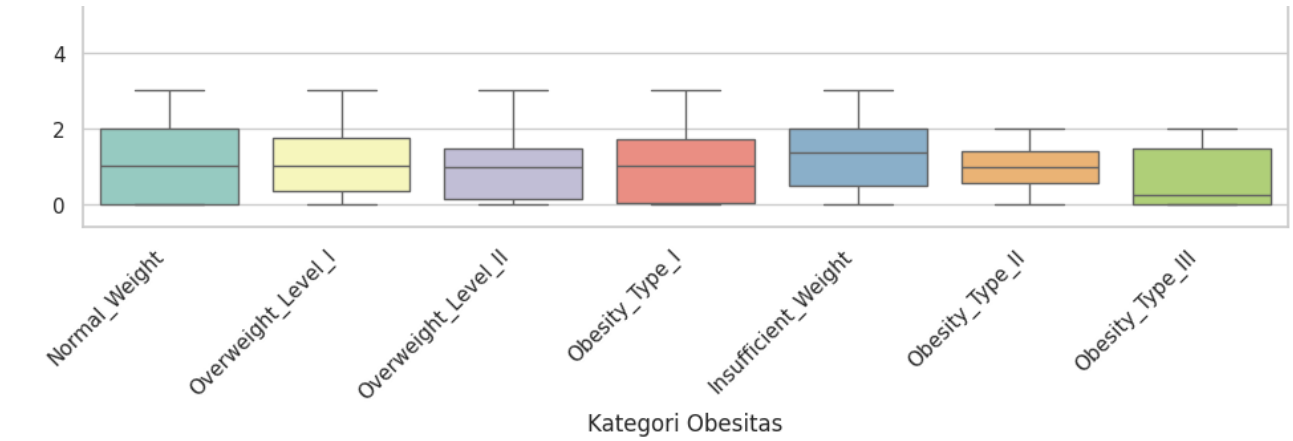

Distribusi Weight berdasarkan Kategori Obesitas
Weight
450
400
350
300
250
200
150
100
50
Normal_Weight
Overweight_Level_I
Overweight_Level_II
Obesity_Type_I
Insufficient_Weight
Obesity_Type_II
Obesity_Type_III
Kategori Obesitas

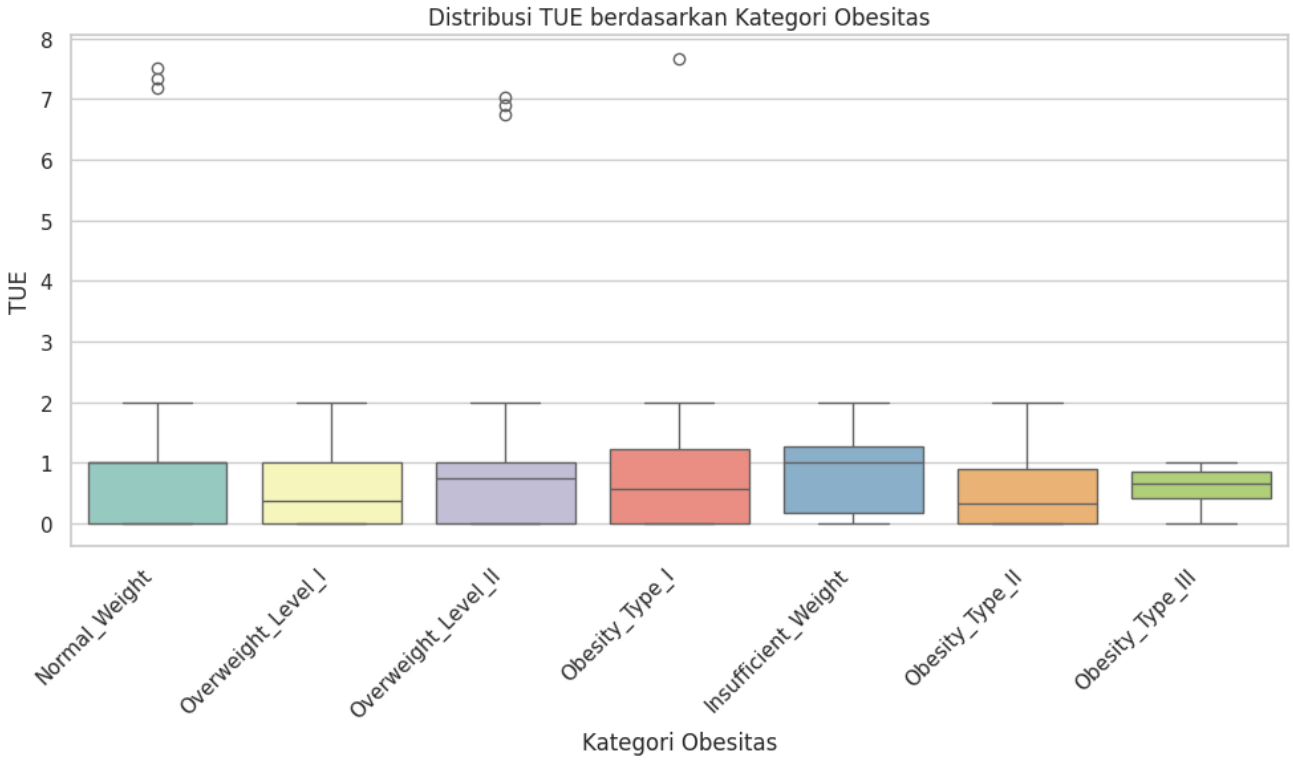

Distribusi FCVC berdasarkan Kategori Obesitas
FCVC
9
8
7
6
5
4
3
2
1
Normal_Weight
Overweight_Level_I
Overweight_Level_II
Obesity_Type_I
Insufficient_Weight
Obesity_Type_II
Obesity_Type_III


Kategori Obesitas
Distribusi NCP berdasarkan Kategori Obesitas
NCP
12
10
8
6
4
2
Normal_Weight
Overweight_Level_I
Overweight_Level_II
Obesity_Type_I
Insufficient_Weight
Obesity_Type_II
Obesity_Type_III


Kategori Obesitas
Distribusi CH2O berdasarkan Kategori Obesitas
CH2O
8
6
4
2
Normal_Weight
Overweight_Level_I
Overweight_Level_II
Obesity_Type_I
Insufficient_Weight
Obesity_Type_II
Obesity_Type_III


Kategori Obesitas
Distribusi FAF berdasarkan Kategori Obesitas
FAF
12
10
8
6


4
2
0
Normal_Weight
Overweight_Level_I
Overweight_Level_II
Obesity_Type_I
Insufficient_Weight
Obesity_Type_II
Obesity_Type_III
Kategori Obesitas


Distribusi TUE berdasarkan Kategori Obesitas
TUE
8
7
6
5
4
3
2
1
0
Normal_Weight
Overweight_Level_I
Overweight_Level_II
Obesity_Type_I
Insufficient_Weight
Obesity_Type_II
Obesity_Type_III
Kategori Obesitas

```
# Cek missing values per kolom
missing_values = df.isnull().sum()
#print("Jumlah Missing Values per Kolom:\n", missing_values)
print(missing_values)
```

```
Age                               22
Gender                             9
Height                            22
Weight                            19
CALC                               5
FAVC                              11
FCVC                              18
NCP                               22
SCC                               10
SMOKE                              5
CH2O                              15
family_history_with_overweight    13
FAF                               19
TUE                               15
CAEC                              11
MTRANS                             6
NObeyesdad                         0
dtype: int64
```

```
# Cek unique values per kolom
unique_values = df.nunique()
print("\nJumlah Unique Values per Kolom:\n", unique_values)
```

```
Jumlah Unique Values per Kolom:
 Age                               1393
Gender                                3
Height                             1561
Weight                             1517
CALC                                  5
FAVC                                  3
```

```
FCVC                                807
NCP                                 636
SCC                                   3
SMOKE                                 3
CH2O                               1262
family_history_with_overweight        3
FAF                                1185
TUE                                1129
CAEC                                  5
MTRANS                                6
NObeyesdad                            7
dtype: int64
```

```
print("\nUnique values pada semua kolom:")
for col in df.columns:
    print(f"- {col}: {df[col].unique()}")
```

```
Unique values pada semua kolom:
- Age: [21.       23.       27.       ... 22.524036 24.361936 23.664709]
- Gender: ['Female' 'Male' '?' nan]
- Height: [1.62     1.52     1.8      ... 1.752206 1.73945  1.738836]
- Weight: [ 64.        56.        77.       ... 133.689352 133.346641 133.472641]
- CALC: ['no' 'Sometimes' 'Frequently' '?' 'Always' nan]
- FAVC: ['no' 'yes' '?' nan]
- FCVC: [2.         3.         1.               nan 8.14899274 8.42397393
 2.450218   2.880161   2.00876    2.596579   2.591439   2.392665
 1.123939   2.027574   2.658112   2.88626    2.714447   2.750715
 1.4925     2.205439   2.059138   2.310423   2.823179   2.052932
 2.596364   2.767731   2.815157   2.737762   2.524428   2.971574
 1.0816     1.270448   1.344854   2.959658   2.725282   2.844607
 2.44004    2.432302   2.592247   2.449267   2.929889   2.015258
 1.031149   1.592183   1.21498    1.522001   2.703436   2.362918
 2.14084    2.5596     2.336044   1.813234   2.724285   2.71897
 1.133844   1.757466   2.979383   2.204914   2.927218   2.88853
 2.890535   2.530066   2.241606   1.003566   2.652779   2.897899
 2.483979   2.945967   2.478891   2.784464   1.005578   2.938031
 2.842102   1.889199   2.943749   2.33998    1.950742   2.277436
 2.371338   2.984425   2.977018   2.663421   2.753752   2.318355
 2.594653   2.886157   2.967853   2.619835   1.053534   2.530233
 2.8813     2.824559   2.762325   2.070964   2.68601    2.794197
 2.720701   2.880792   2.674431   2.55996    1.212908   1.140615
 2.562409   2.004146   2.690754   2.051283   2.19005    2.21498
 2.91548    2.708965   2.853513   2.580872   2.508835   2.896562
 2.911877   2.910733   2.966126   2.613249   2.627031   2.919751
 2.494451   1.69427    1.601236   1.204855   1.052699   2.910345
 2.866383   2.913486   2.432886   2.883745   2.707666   2.919584
 2.969205   2.486189   1.642241   1.567101   1.036414   1.649974
 1.118436   2.673638   2.120185   2.34222    2.86099    2.559571
 2.424977   1.786841   1.303878   1.889883   2.984004   2.749268
 1.202075   8.28511134 2.341133   1.206276   2.81646    1.758394
 2.577427   2.052152   2.954996   2.555401   2.108711   2.915279
 1.570089   1.94313    2.903545   1.75375    2.543563   2.39728
```

```
 2.37464    2.278644   1.620845   2.061952   2.838969   2.568063
 2.652958   1.27785    1.729824   1.452524   2.303367   2.948425
 2.291846   1.906194   1.834155   2.048582   2.948248   2.869436
 2.293705   2.510583   2.366949   2.615788   2.217267   2.801514
 2.188722   2.971351   2.086093   1.901611   1.977298   2.446872
 2.839048   2.21232    2.427689   1.078529   1.064162   1.993101
 2.620963   2.95118    2.021446   2.000466   2.5621     2.96008
 2.53915    2.244142   2.253371   2.851664   1.31415    1.321028
 2.253998   2.778079   2.838037   2.814453   2.013782   2.459976
 2.643183   2.22399    2.104105   1.972545   2.286481   2.971588
 2.872121   2.109162   2.178889   1.142468   2.047069   2.843709
 2.416044   2.146598   1.766849   1.188089   1.910176   2.956671
 2.002796   2.288604   2.138334   2.029634   2.048216   2.8557
 2.995599   2.987148   1.887951   2.786008   2.342323   1.874935
 2.213135   2.273548   2.780699   1.687569   1.989905   1.947405
 2.162519   2.923916   2.99448    2.507841   1.836554   1.773265
 2.388168   2.286146   2.487167   2.185938   2.206399   1.952987
 2.908757   2.628791   2.749629   1.595746   2.885178   2.372494
 8.7067947  2.793561   2.992329   2.927409   2.706134   2.010684
 2.300408   2.119643   2.901924   2.451009   2.754646   2.417635
 2.512719   1.771693   1.57223    2.661556   2.097373   2.061461
 1.317729   1.882235   2.951591   2.067817   2.54527    2.694281
```

```
# Cek data duplikat
# Jumlah total baris duplikat (seluruh baris sama persis)
total_duplikat = df.duplicated().sum()
print("Total baris duplikat:", total_duplikat)
print("--------------------------------")

# Tampilkan baris yang terduplikat
duplikat = df[df.duplicated()]
print("Baris duplikat:")
print(duplikat)

# Ambil satu contoh baris duplikat
if not duplikat.empty:
    ref = duplikat.iloc[0]
    matching_cols = df.columns[(df == ref).all(axis=0)]
    print("Kolom yang identik di baris duplikat contoh:", list(matching_cols))
```

```
Total baris duplikat: 18
--------------------------------
Baris duplikat:
      Age  Gender  Height  Weight       CALC FAVC  FCVC  NCP SCC SMOKE  CH2O  \
98   21.0  Female    1.52    42.0  Sometimes   no   3.0  1.0  no    no   1.0
174  21.0    Male    1.62    70.0  Sometimes  yes   2.0  1.0  no    no   3.0
179  21.0    Male    1.62    70.0  Sometimes  yes   2.0  1.0  no    no   3.0
184  21.0    Male    1.62    70.0  Sometimes  yes   2.0  1.0  no    no   3.0
309  16.0  Female    1.66    58.0         no   no   2.0  1.0  no    no   1.0
460  18.0  Female    1.62    55.0         no  yes   2.0  3.0  no    no   1.0
663  21.0  Female    1.52    42.0  Sometimes  yes   3.0  1.0  no    no   1.0
763  21.0    Male    1.62    70.0  Sometimes  yes   2.0  1.0  no    no   3.0
```

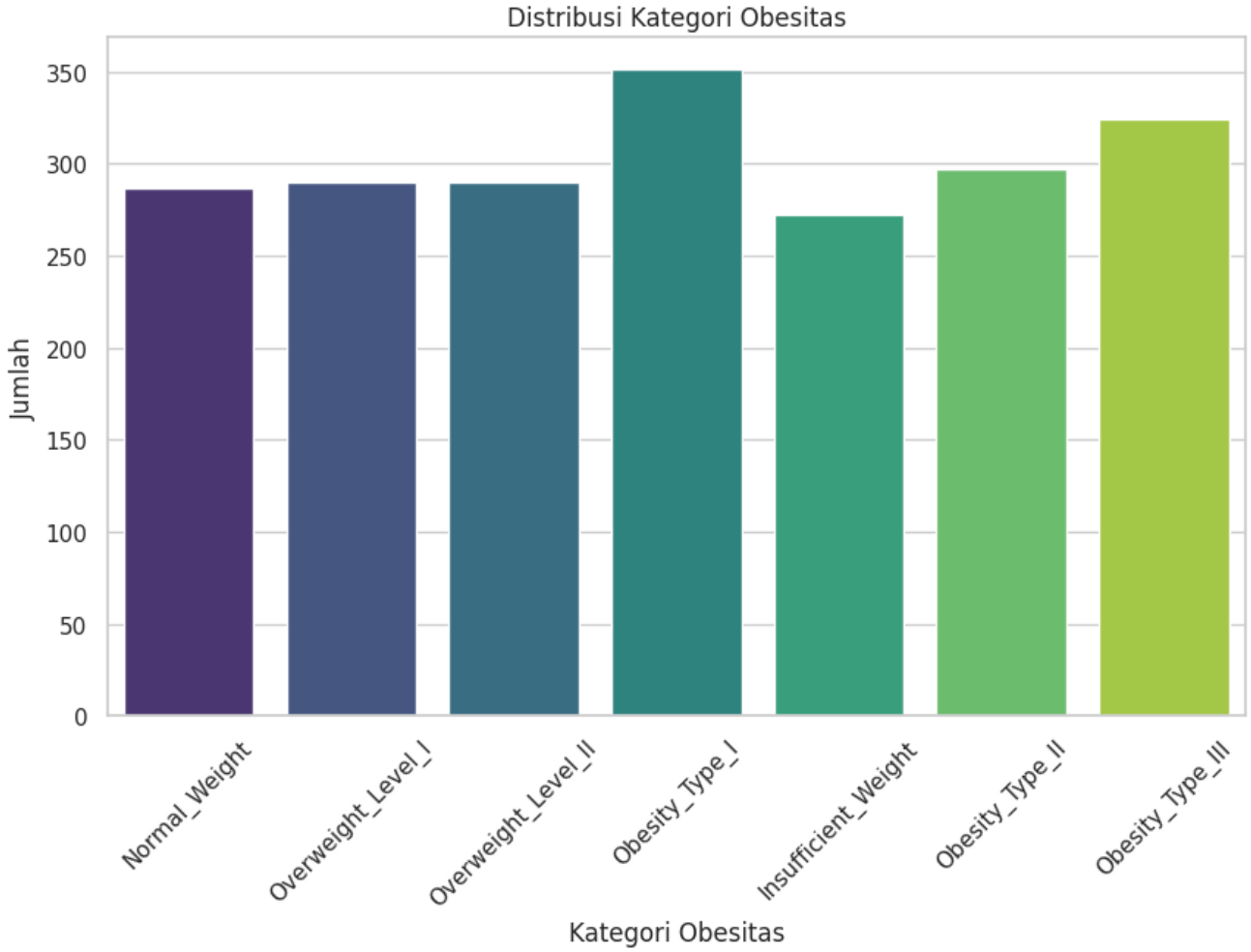
```
     764  21.0    Male    1.62    70.0  Sometimes  yes   2.0  1.0  no    no   3.0
     824  21.0    Male    1.62    70.0  Sometimes  yes   2.0  1.0  no    no   3.0
     830  21.0    Male    1.62    70.0  Sometimes  yes   2.0  1.0  no    no   3.0
     831  21.0    Male    1.62    70.0  Sometimes  yes   2.0  1.0  no    no   3.0
     832  21.0    Male    1.62    70.0  Sometimes  yes   2.0  1.0  no    no   3.0
     833  21.0    Male    1.62    70.0  Sometimes  yes   2.0  1.0  no    no   3.0
     834  21.0    Male    1.62    70.0  Sometimes  yes   2.0  1.0  no    no   3.0
     921  21.0    Male    1.62    70.0  Sometimes  yes   2.0  1.0  no    no   3.0
     922  21.0    Male    1.62    70.0  Sometimes  yes   2.0  1.0  no    no   3.0
     923  21.0    Male    1.62    70.0  Sometimes  yes   2.0  1.0  no    no   3.0

         family_history_with_overweight  FAF  TUE        CAEC  \
     98                               no  0.0  0.0  Frequently
     174                              no  1.0  0.0          no
     179                              no  1.0  0.0          no
     184                              no  1.0  0.0          no
     309                              no  0.0  1.0   Sometimes
     460                             yes  1.0  1.0  Frequently
     663                              no  0.0  0.0  Frequently
     763                              no  1.0  0.0          no
     764                              no  1.0  0.0          no
     824                              no  1.0  0.0          no
     830                              no  1.0  0.0          no
     831                              no  1.0  0.0          no
     832                              no  1.0  0.0          no
     833                              no  1.0  0.0          no
     834                              no  1.0  0.0          no
     921                              no  1.0  0.0          no
     922                              no  1.0  0.0          no
     923                              no  1.0  0.0          no

                        MTRANS           NObeyesdad
     98   Public_Transportation  Insufficient_Weight
     174  Public_Transportation   Overweight_Level_I
     179  Public_Transportation   Overweight_Level_I
     184  Public_Transportation   Overweight_Level_I
     309                Walking        Normal_Weight
     460  Public_Transportation        Normal_Weight
     663  Public_Transportation  Insufficient_Weight
     763  Public_Transportation   Overweight_Level_I
     764  Public_Transportation   Overweight_Level_I
     824  Public_Transportation   Overweight_Level_I
     830  Public_Transportation   Overweight_Level_I
     831  Public_Transportation   Overweight_Level_I
     832  Public_Transportation   Overweight_Level_I
     833  Public_Transportation   Overweight_Level_I
```

```
# Cek keseimbangan data pada kolom target 'NObeyesdad'
class_distribution = df['NObeyesdad'].value_counts()
print("\nDistribusi Keseimbangan Data (NObeyesdad):\n", class_distribution)
```

```
plt.figure(figsize=(10, 6))
sns.countplot(data=df, x='NObeyesdad', hue='NObeyesdad', palette='viridis', legend=False)
```

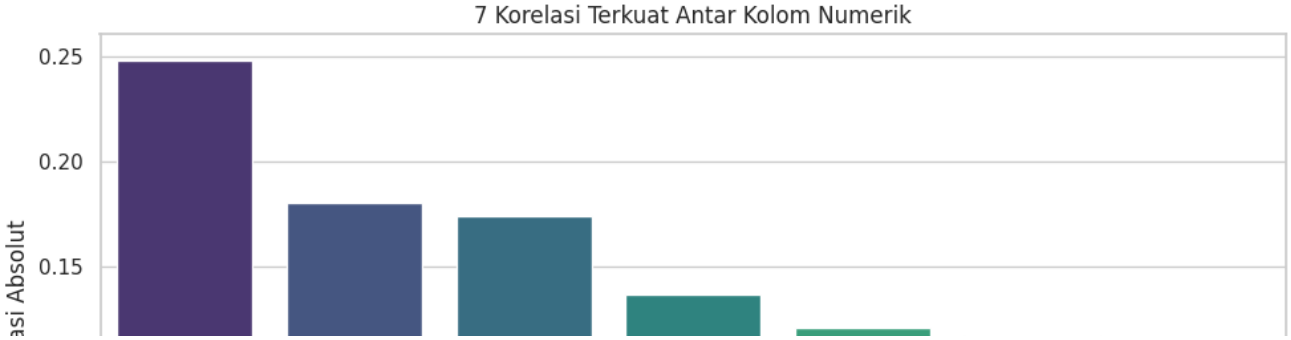
```
plt.title('Distribusi Kategori Obesitas')
plt.xlabel('Kategori Obesitas')
plt.ylabel('Jumlah')
plt.xticks(rotation=45)
plt.show()
```

```
Distribusi Keseimbangan Data (NObeyesdad):
 NObeyesdad
Obesity_Type_I         351
Obesity_Type_III       324
Obesity_Type_II        297
Overweight_Level_I     290
Overweight_Level_II    290
Normal_Weight          287
Insufficient_Weight    272
Name: count, dtype: int64
```



```
# Konversi kolom numerik ke tipe data numerik
for col in numerical_columns:
    df[col] = pd.to_numeric(df[col], errors='coerce')

# Hapus baris dengan missing values di kolom numerik
df_cleaned = df.dropna(subset=numerical_columns)

# Hitung matriks korelasi
correlation_matrix = df_cleaned[numerical_columns].corr()

# Ambil nilai korelasi absolut dan urutkan
abs_corr_matrix = np.abs(correlation_matrix)
upper_triangle = abs_corr_matrix.where(np.triu(np.ones(abs_corr_matrix.shape), k=1).astyp
strong_correlations = upper_triangle.unstack().sort_values(ascending=False)
strong_correlations = strong_correlations.dropna()  # Hapus NaN

# Pilih 7 korelasi teratas
top_7_correlations = strong_correlations.head(7)

# Siapkan data untuk diagram batang
correlation_data = pd.DataFrame({
    'Pair': [f"{pair[0]} - {pair[1]}" for pair in top_5_correlations.index],
    'Correlation': top_5_correlations.values
})

# Visualisasikan dalam diagram batang
plt.figure(figsize=(10, 6))
sns.barplot(x='Pair', y='Correlation', data=correlation_data, palette='viridis')
plt.title('7 Korelasi Terkuat Antar Kolom Numerik')
plt.xlabel('Pasangan Kolom')
plt.ylabel('Korelasi Absolut')
plt.xticks(rotation=45, ha='right')  # Rotasi label sumbu x agar terbaca
plt.tight_layout()  # Untuk mencegah label tumpang tindih
plt.show()
```

```
<ipython-input-53-4972a78b9657>:28: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.

  sns.barplot(x='Pair', y='Correlation', data=correlation_data, palette='viridis')
```



Top 7 Korelasi Tertinggi (dalam nilai absolut)

Pasangan Fitur	Korelasi	Analisis
Weight – Height	0.248	Positif lemah: Orang dengan tinggi badan lebih besar cenderung memiliki berat lebih tinggi, meskip
NCP – Height	0.180	Korelasi lemah: Orang dengan tinggi tertentu cenderung memiliki pola makan besar tertentu, tetapi
TUE – Age	0.174	Korelasi lemah: Usia memengaruhi durasi penggunaan teknologi; kemungkinan, kelompok usia muc
FAF – Height	0.136	Korelasi sangat lemah: Tinggi badan sedikit berkorelasi dengan aktivitas fisik, bisa jadi orang lebih t
CH2O – Height	0.121	Korelasi sangat lemah: Tinggi badan sedikit berkaitan dengan konsumsi air harian, tetapi tidak sign
CH2O – Weight	0.117	Korelasi sangat lemah: Berat badan memiliki sedikit hubungan dengan konsumsi air, tetapi tidak cu
FCVC – Weight	0.111	Korelasi sangat lemah: Konsumsi sayuran berkaitan sedikit dengan berat badan; bisa berarti diet se

Kesimpulan Analisis Korelasi

- Tidak ada korelasi yang **kuat** antar fitur numerik (semuanya < 0.3).
- Korelasi tertinggi pun (`Weight - Height`) hanya 0.25, yang termasuk **lemah**.
- Ini menunjukkan bahwa **tidak ada dua fitur numerik** dalam dataset ini yang sangat linear satu sama lain.
- Hal ini baik untuk pemodelan, karena tidak ada multikolinearitas tinggi yang bisa merusak performa model prediktif berbasis regresi atau pohon keputusan.