Python Coding Challenge

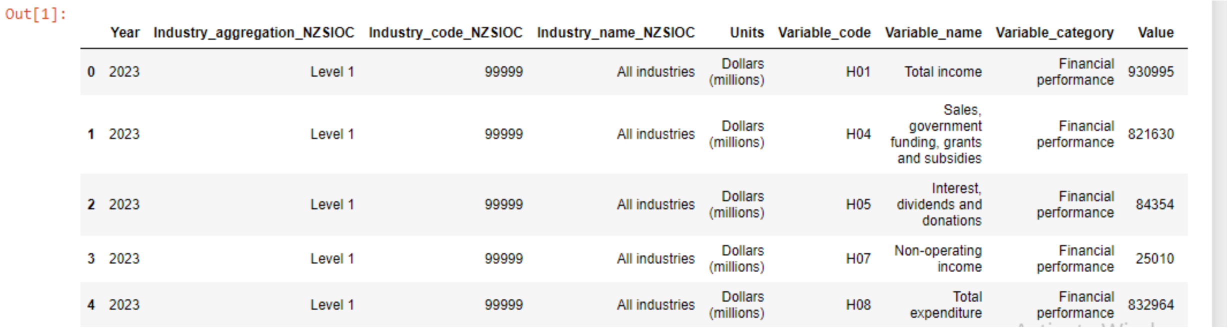
-Sarthak Niranjan Kulkarni (Maverick)

- sarthakkul2311@gmail.com - (+91) 93256 02791

15/11/2024 (Friday)

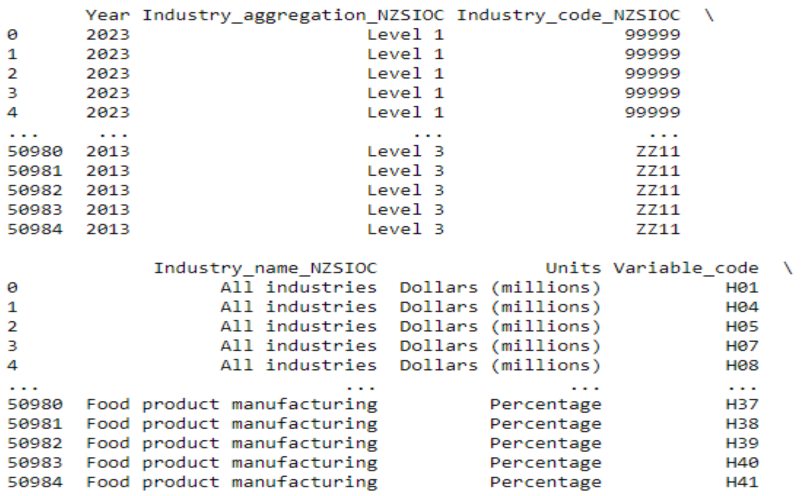
1. Printing rows of the Data

```
→ import pandas as pd

# File path

file_path = r"C:\Users\Sarthak Kulkarni\Desktop\Hexaware Python Training\Data_engineering\Coding_Challenge\Python-Coding-Challenge\annual-enterprise-survey-2023-financial-year-provisional.csv"

# Load the CSV file into a DataFrame

df = pd.read_csv(file_path)

# Display the first few rows of the dataset

df.head()

print(df)
```



Out[1]:

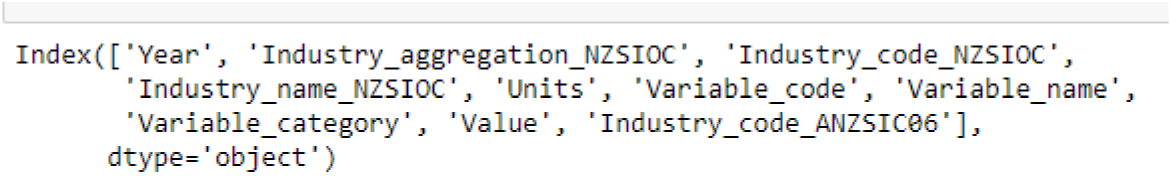
	Year	Industry_aggregation_NZSIOC	Industry_code_NZSIOC	Industry_name_NZSIOC	Units	Variable_code	Variable_name	Variable_category	Value
0	2023	Level 1	99999	All industries	Dollars (millions)	H01	Total income	Financial performance	930995
1	2023	Level 1	99999	All industries	Dollars (millions)	H04	Sales, government funding, grants and subsidies	Financial performance	821630
2	2023	Level 1	99999	All industries	Dollars (millions)	H05	Interest, dividends and donations	Financial performance	84354
3	2023	Level 1	99999	All industries	Dollars (millions)	H07	Non-operating income	Financial performance	25010
4	2023	Level 1	99999	All industries	Dollars (millions)	H08	Total expenditure	Financial performance	832964



```
       Year Industry_aggregation_NZSIOC Industry_code_NZSIOC  \
0      2023                     Level 1                99999
1      2023                     Level 1                99999
2      2023                     Level 1                99999
3      2023                     Level 1                99999
4      2023                     Level 1                99999
...     ...                         ...                  ...
50980  2013                     Level 3                 ZZ11
50981  2013                     Level 3                 ZZ11
50982  2013                     Level 3                 ZZ11
50983  2013                     Level 3                 ZZ11
50984  2013                     Level 3                 ZZ11

              Industry_name_NZSIOC               Units Variable_code  \
0                   All industries  Dollars (millions)           H01
1                   All industries  Dollars (millions)           H04
2                   All industries  Dollars (millions)           H05
3                   All industries  Dollars (millions)           H07
4                   All industries  Dollars (millions)           H08
...                            ...                 ...           ...
50980  Food product manufacturing          Percentage           H37
50981  Food product manufacturing          Percentage           H38
50982  Food product manufacturing          Percentage           H39
50983  Food product manufacturing          Percentage           H40
50984  Food product manufacturing          Percentage           H41
```

2. Printing the column names of the DataFrame.

→ print(df.columns)

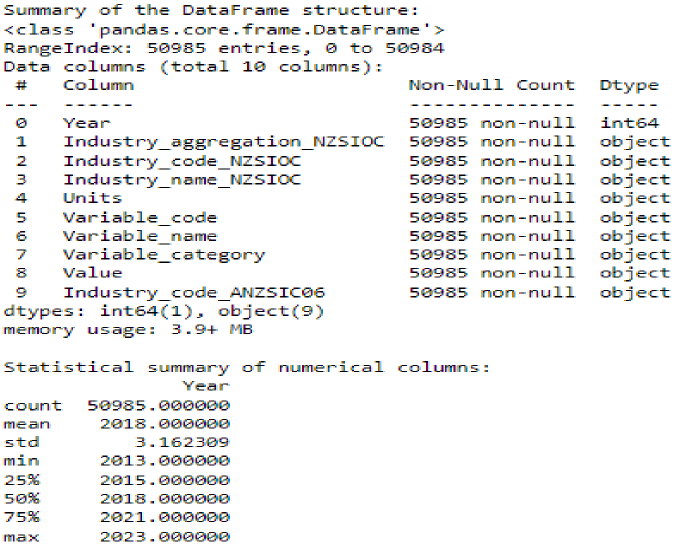


```
Index(['Year', 'Industry_aggregation_NZSIOC', 'Industry_code_NZSIOC',
       'Industry_name_NZSIOC', 'Units', 'Variable_code', 'Variable_name',
       'Variable_category', 'Value', 'Industry_code_ANZSIC06'],
      dtype='object')
```

3. Summary of Data Frame

→ print("Summary of the DataFrame structure:")

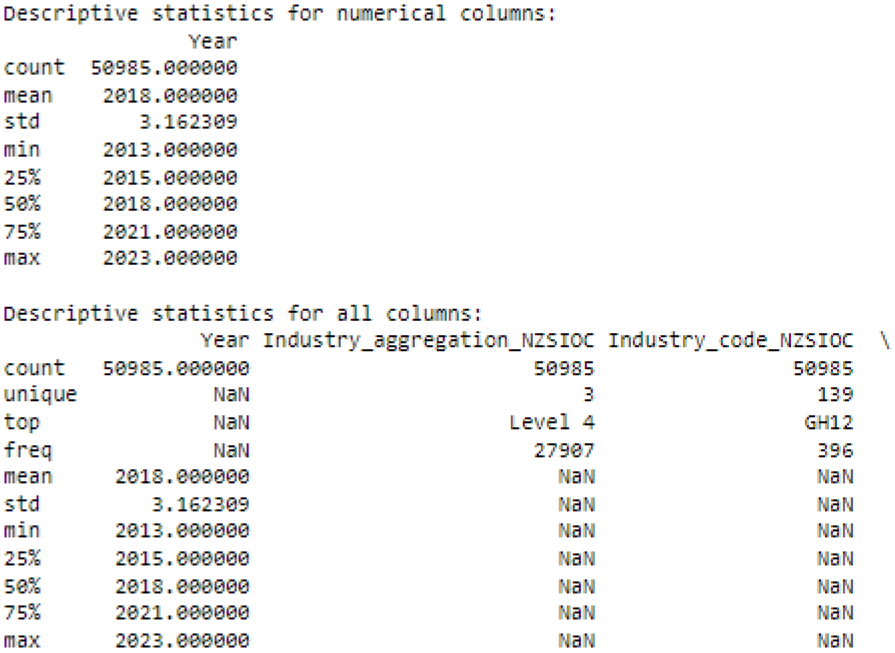
df.info()



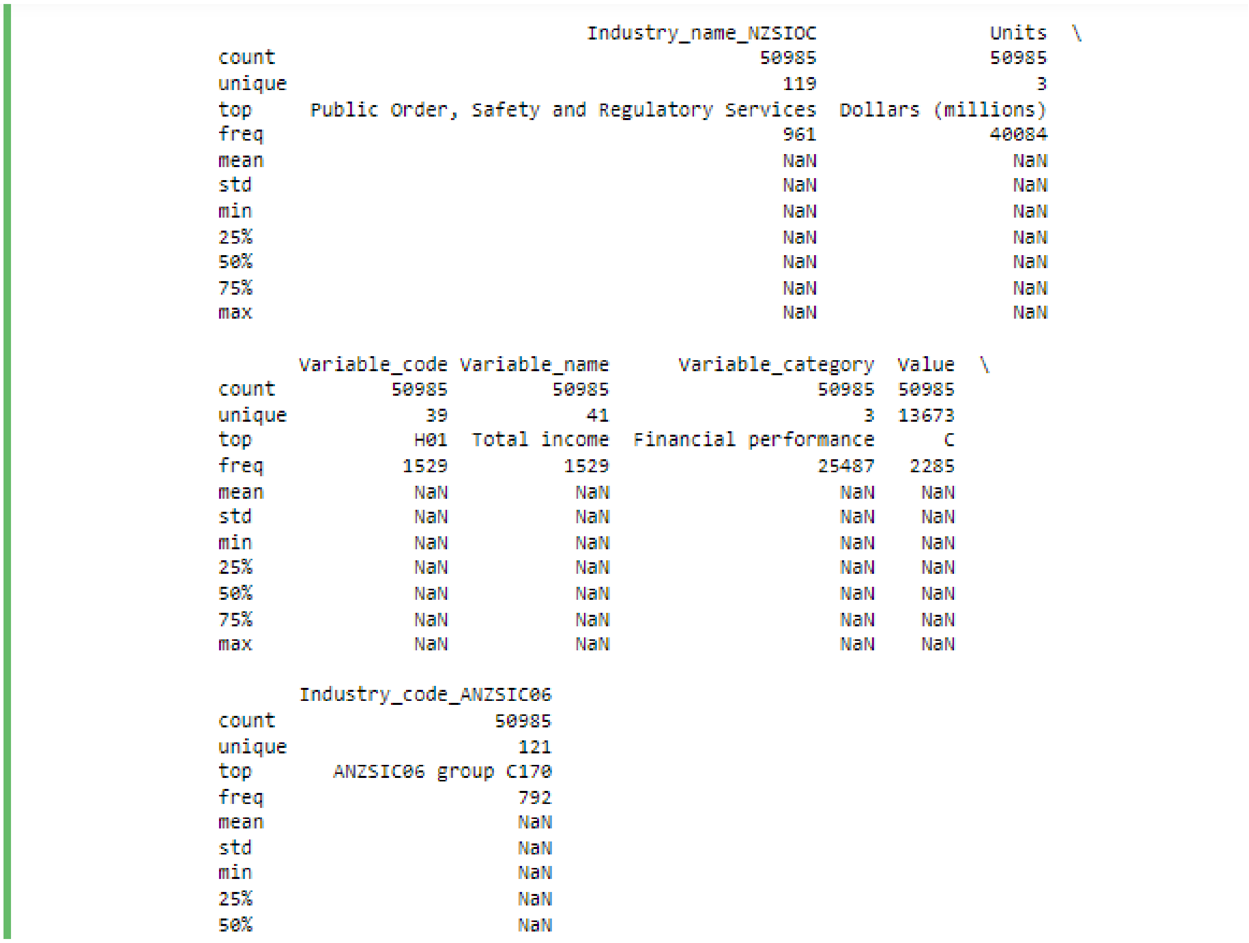
```
Summary of the DataFrame structure:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 50985 entries, 0 to 50984
Data columns (total 10 columns):
 #   Column                       Non-Null Count  Dtype
---  ------                       --------------  -----
 0   Year                         50985 non-null  int64
 1   Industry_aggregation_NZSIOC  50985 non-null  object
 2   Industry_code_NZSIOC         50985 non-null  object
 3   Industry_name_NZSIOC         50985 non-null  object
 4   Units                        50985 non-null  object
 5   Variable_code                50985 non-null  object
 6   Variable_name                50985 non-null  object
 7   Variable_category            50985 non-null  object
 8   Value                        50985 non-null  object
 9   Industry_code_ANZSIC06       50985 non-null  object
dtypes: int64(1), object(9)
memory usage: 3.9+ MB

Statistical summary of numerical columns:
               Year
count  50985.000000
mean    2018.000000
std        3.162309
min     2013.000000
25%     2015.000000
50%     2018.000000
75%     2021.000000
max     2023.000000
```

4. Descriptive Statistical Measures of a DataFrame

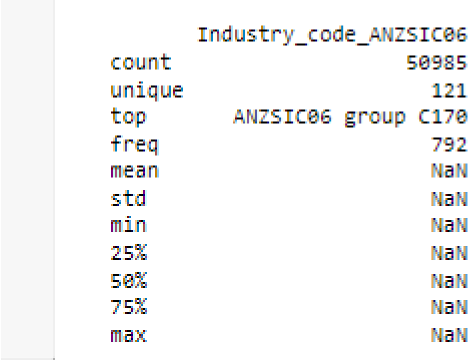
```
→ # Descriptive Statistical Measures of a DataFrame

print("Descriptive statistics for numerical columns:")

print(df.describe())



# Descriptive statistics for all columns (including categorical)

print("\nDescriptive statistics for all columns:")

print(df.describe(include='all'))
```



```
Descriptive statistics for numerical columns:
               Year
count  50985.000000
mean    2018.000000
std        3.162309
min     2013.000000
25%     2015.000000
50%     2018.000000
75%     2021.000000
max     2023.000000

Descriptive statistics for all columns:
                Year Industry_aggregation_NZSIOC Industry_code_NZSIOC  \
count   50985.000000                       50985                50985   
unique           NaN                           3                  139   
top              NaN                     Level 4                 GH12   
freq             NaN                       27907                  396   
mean     2018.000000                         NaN                  NaN   
std         3.162309                         NaN                  NaN   
min      2013.000000                         NaN                  NaN   
25%      2015.000000                         NaN                  NaN   
50%      2018.000000                         NaN                  NaN   
75%      2021.000000                         NaN                  NaN   
max      2023.000000                         NaN                  NaN   
```



```
                                   Industry_name_NZSIOC              Units  \
count                                             50985              50985
unique                                              119                  3
top     Public Order, Safety and Regulatory Services  Dollars (millions)
freq                                                961              40084
mean                                                NaN                NaN
std                                                 NaN                NaN
min                                                 NaN                NaN
25%                                                 NaN                NaN
50%                                                 NaN                NaN
75%                                                 NaN                NaN
max                                                 NaN                NaN

       Variable_code Variable_name      Variable_category  Value  \
count          50985         50985                  50985  50985
unique            39            41                      3  13673
top              H01  Total income  Financial performance      C
freq            1529          1529                  25487   2285
mean             NaN           NaN                    NaN    NaN
std              NaN           NaN                    NaN    NaN
min              NaN           NaN                    NaN    NaN
25%              NaN           NaN                    NaN    NaN
50%              NaN           NaN                    NaN    NaN
75%              NaN           NaN                    NaN    NaN
max              NaN           NaN                    NaN    NaN

       Industry_code_ANZSIC06
count                   50985
unique                    121
top       ANZSIC06 group C170
freq                      792
mean                      NaN
std                       NaN
min                       NaN
25%                       NaN
50%                       NaN
```



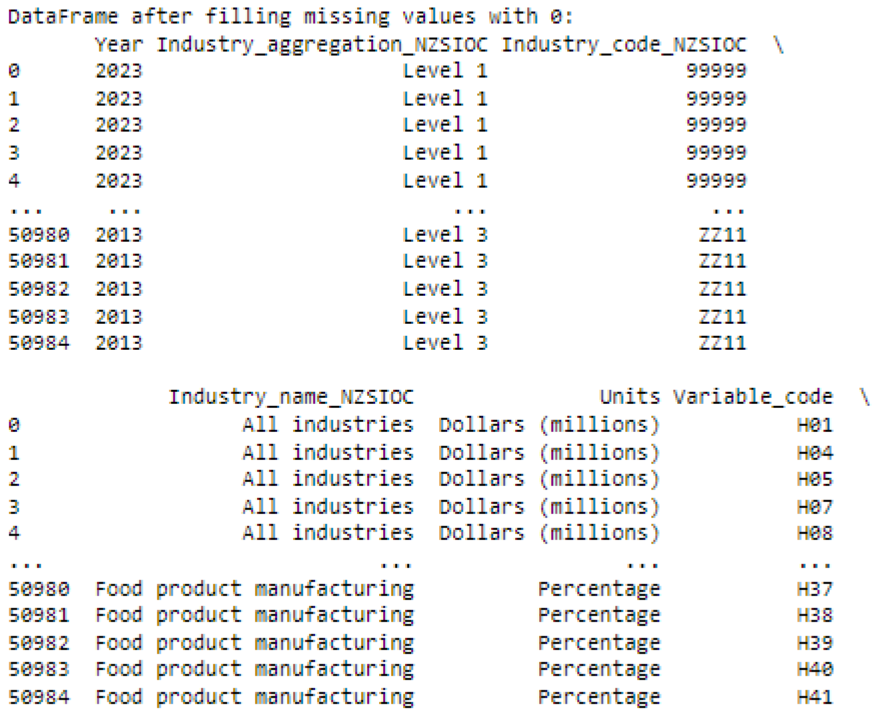
```
       Industry_code_ANZSIC06
count                   50985
unique                    121
top       ANZSIC06 group C170
freq                      792
mean                      NaN
std                       NaN
min                       NaN
25%                       NaN
50%                       NaN
75%                       NaN
max                       NaN
```

5. Missing Data Handing

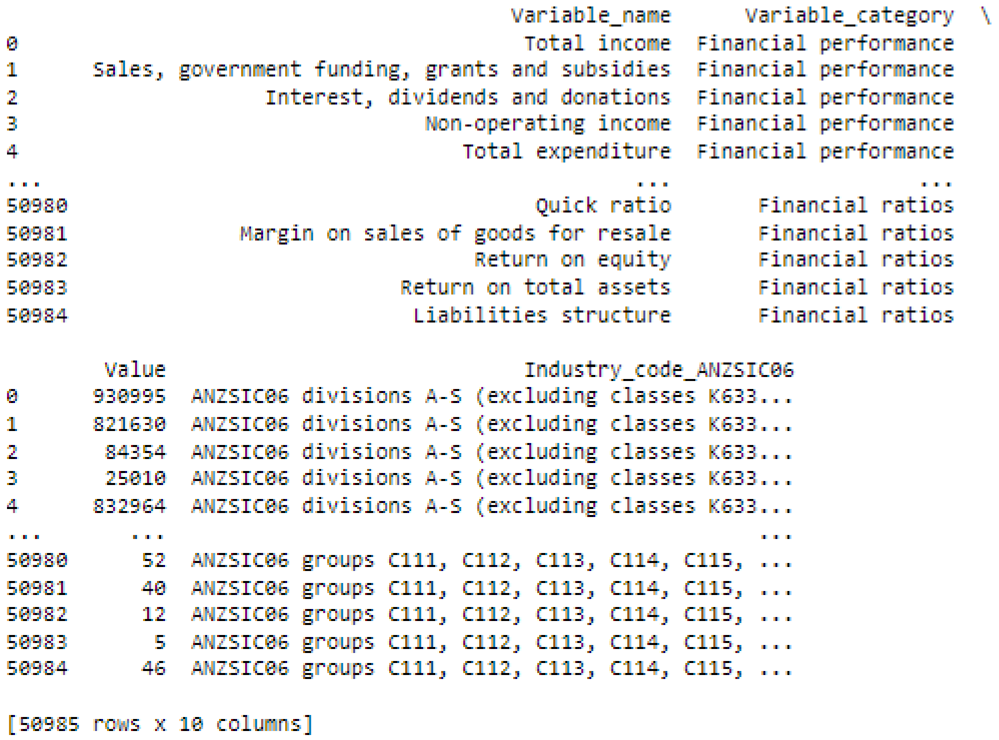
→ df_filled = df.fillna(0)

print("\nDataFrame after filling missing values with 0:")

print(df_filled)



```
DataFrame after filling missing values with 0:
       Year Industry_aggregation_NZSIOC Industry_code_NZSIOC  \
0      2023                     Level 1                99999   
1      2023                     Level 1                99999   
2      2023                     Level 1                99999   
3      2023                     Level 1                99999   
4      2023                     Level 1                99999   
...     ...                         ...                  ...   
50980  2013                     Level 3                 ZZ11   
50981  2013                     Level 3                 ZZ11   
50982  2013                     Level 3                 ZZ11   
50983  2013                     Level 3                 ZZ11   
50984  2013                     Level 3                 ZZ11   

            Industry_name_NZSIOC               Units Variable_code  \
0                 All industries  Dollars (millions)           H01   
1                 All industries  Dollars (millions)           H04   
2                 All industries  Dollars (millions)           H05   
3                 All industries  Dollars (millions)           H07   
4                 All industries  Dollars (millions)           H08   
...                          ...                 ...           ...   
50980  Food product manufacturing          Percentage           H37   
50981  Food product manufacturing          Percentage           H38   
50982  Food product manufacturing          Percentage           H39   
50983  Food product manufacturing          Percentage           H40   
50984  Food product manufacturing          Percentage           H41   
```



```
                                         Variable_name      Variable_category  \
0                                         Total income  Financial performance   
1      Sales, government funding, grants and subsidies  Financial performance   
2                    Interest, dividends and donations  Financial performance   
3                                 Non-operating income  Financial performance   
4                                    Total expenditure  Financial performance   
...                                                ...                    ...   
50980                                      Quick ratio       Financial ratios   
50981                 Margin on sales of goods for resale       Financial ratios   
50982                                  Return on equity       Financial ratios   
50983                            Return on total assets       Financial ratios   
50984                             Liabilities structure       Financial ratios   

        Value                             Industry_code_ANZSIC06  
0      930995  ANZSIC06 divisions A-S (excluding classes K633...  
1      821630  ANZSIC06 divisions A-S (excluding classes K633...  
2       84354  ANZSIC06 divisions A-S (excluding classes K633...  
3       25010  ANZSIC06 divisions A-S (excluding classes K633...  
4      832964  ANZSIC06 divisions A-S (excluding classes K633...  
...       ...                                                ...  
50980      52  ANZSIC06 groups C111, C112, C113, C114, C115, ...  
50981      40  ANZSIC06 groups C111, C112, C113, C114, C115, ...  
50982      12  ANZSIC06 groups C111, C112, C113, C114, C115, ...  
50983       5  ANZSIC06 groups C111, C112, C113, C114, C115, ...  
50984      46  ANZSIC06 groups C111, C112, C113, C114, C115, ...  

[50985 rows x 10 columns]
```

6. Sorting DataFrame values.

→ # 6.Sorting DataFrame values

# Ascending order

sorted_df = df.sort_values(by='Variable_name', ascending=True)

print("DataFrame sorted by Variable_name in ascending order:")

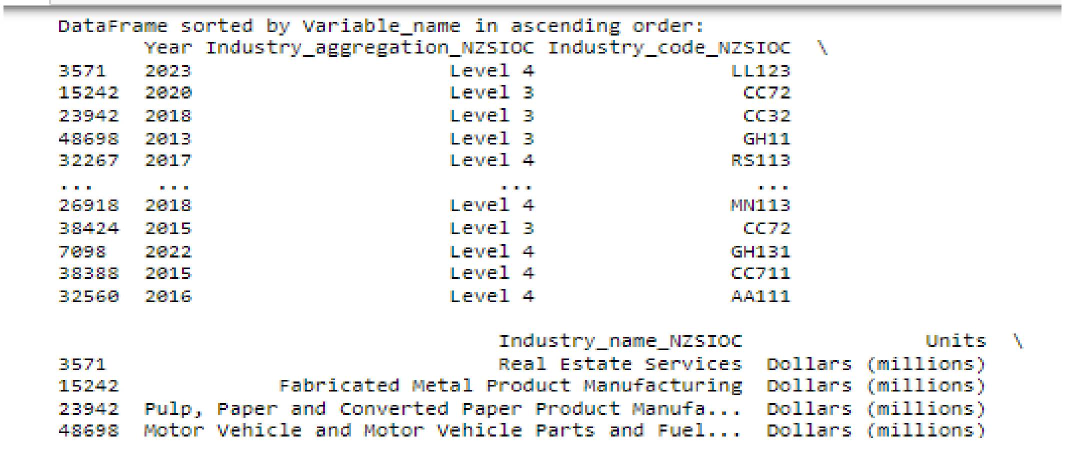
print(sorted_df)

# Descending order

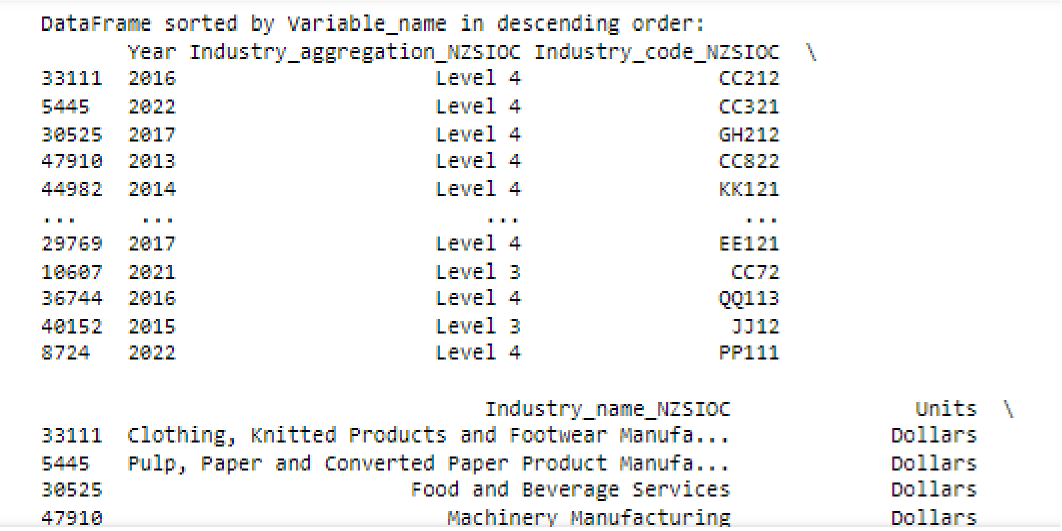
sorted_df_desc = df.sort_values(by='Variable_name', ascending=False)

print("\nDataFrame sorted by Variable_name in descending order:")

print(sorted_df_desc)



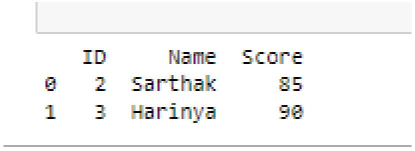
```
DataFrame sorted by Variable_name in ascending order:
       Year Industry_aggregation_NZSIOC Industry_code_NZSIOC  \
3571   2023                     Level 4                LL123
15242  2020                     Level 3                 CC72
23942  2018                     Level 3                 CC32
48698  2013                     Level 3                 GH11
32267  2017                     Level 4                RS113
...     ...                         ...                  ...
26918  2018                     Level 4                MN113
38424  2015                     Level 3                 CC72
7098   2022                     Level 4                GH131
38388  2015                     Level 4                CC711
32560  2016                     Level 4                AA111

                               Industry_name_NZSIOC              Units  \
3571                            Real Estate Services  Dollars (millions)
15242            Fabricated Metal Product Manufacturing  Dollars (millions)
23942  Pulp, Paper and Converted Paper Product Manufa...  Dollars (millions)
48698  Motor Vehicle and Motor Vehicle Parts and Fuel...  Dollars (millions)
```



```
DataFrame sorted by Variable_name in descending order:
       Year Industry_aggregation_NZSIOC Industry_code_NZSIOC  \
33111  2016                     Level 4                CC212
5445   2022                     Level 4                CC321
30525  2017                     Level 4                GH212
47910  2013                     Level 4                CC822
44982  2014                     Level 4                KK121
...     ...                         ...                  ...
29769  2017                     Level 4                EE121
10607  2021                     Level 3                 CC72
36744  2016                     Level 4                QQ113
40152  2015                     Level 3                 JJ12
8724   2022                     Level 4                PP111

                               Industry_name_NZSIOC    Units  \
33111  Clothing, Knitted Products and Footwear Manufa...  Dollars
5445   Pulp, Paper and Converted Paper Product Manufa...  Dollars
30525                        Food and Beverage Services  Dollars
47910                           Machinery Manufacturing  Dollars
```

7. Merge Data Frames.

```
→ df1 = pd.DataFrame({
    'ID': [1, 2, 3],
    'Name': ['Lakshita', 'Sarthak', 'Harinya']
})


df2 = pd.DataFrame({
    'ID': [2, 3, 4],
    'Score': [85, 90, 75]
})


merged_df = df1.merge(df2, on='ID', how='inner')
print(merged_df)
```

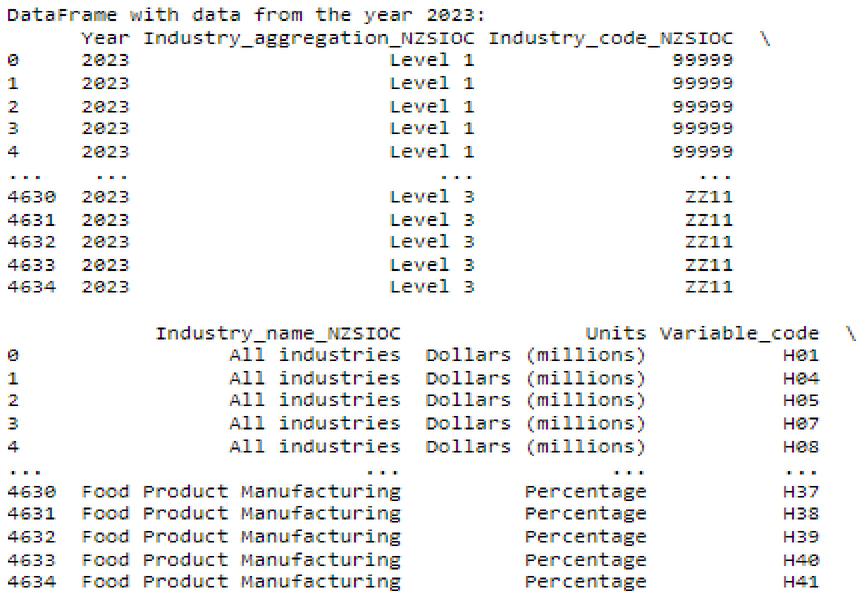


```
   ID     Name  Score
0   2  Sarthak     85
1   3  Harinya     90
```

8. Apply Function.

```
→ def is_year_2023(year):
    return year == 2023


df_2023 = df[df['Year'].apply(is_year_2023)]


print("DataFrame with data from the year 2023:")
```

print(df_2023)



```
DataFrame with data from the year 2023:
      Year Industry_aggregation_NZSIOC Industry_code_NZSIOC  \
0     2023                     Level 1                99999
1     2023                     Level 1                99999
2     2023                     Level 1                99999
3     2023                     Level 1                99999
4     2023                     Level 1                99999
...    ...                         ...                  ...
4630  2023                     Level 3                 ZZ11
4631  2023                     Level 3                 ZZ11
4632  2023                     Level 3                 ZZ11
4633  2023                     Level 3                 ZZ11
4634  2023                     Level 3                 ZZ11

            Industry_name_NZSIOC               Units Variable_code  \
0                 All industries  Dollars (millions)           H01
1                 All industries  Dollars (millions)           H04
2                 All industries  Dollars (millions)           H05
3                 All industries  Dollars (millions)           H07
4                 All industries  Dollars (millions)           H08
...                          ...                 ...           ...
4630  Food Product Manufacturing          Percentage           H37
4631  Food Product Manufacturing          Percentage           H38
4632  Food Product Manufacturing          Percentage           H39
4633  Food Product Manufacturing          Percentage           H40
4634  Food Product Manufacturing          Percentage           H41
```

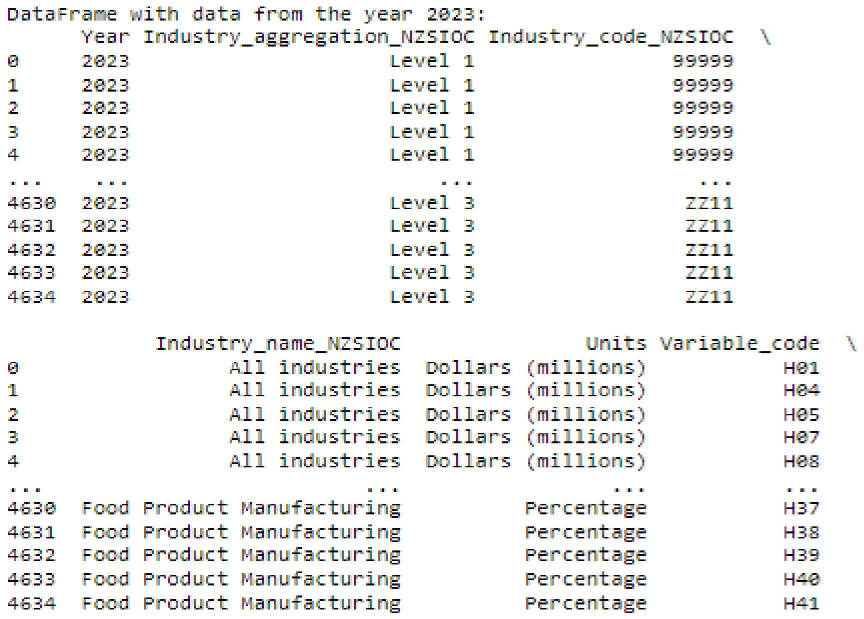
9. By using the lambda operator.

→ # Lambda Function to get rows where 'Year' is 2023

df_2023 = df[df['Year'].apply(lambda x: x == 2023)]

print("DataFrame with data from the year 2023:")

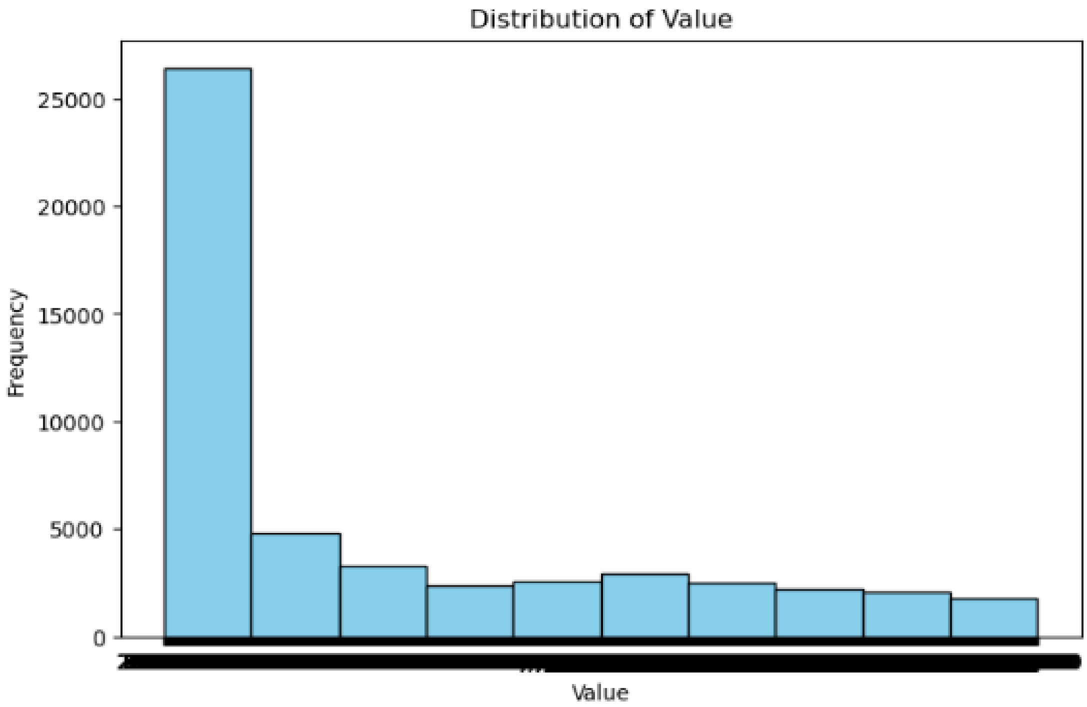
print(df_2023)



```
DataFrame with data from the year 2023:
      Year Industry_aggregation_NZSIOC Industry_code_NZSIOC  \
0     2023                     Level 1                99999
1     2023                     Level 1                99999
2     2023                     Level 1                99999
3     2023                     Level 1                99999
4     2023                     Level 1                99999
...    ...                         ...                  ...
4630  2023                     Level 3                 ZZ11
4631  2023                     Level 3                 ZZ11
4632  2023                     Level 3                 ZZ11
4633  2023                     Level 3                 ZZ11
4634  2023                     Level 3                 ZZ11

            Industry_name_NZSIOC               Units Variable_code  \
0                 All industries  Dollars (millions)           H01
1                 All industries  Dollars (millions)           H04
2                 All industries  Dollars (millions)           H05
3                 All industries  Dollars (millions)           H07
4                 All industries  Dollars (millions)           H08
...                          ...                 ...           ...
4630  Food Product Manufacturing          Percentage           H37
4631  Food Product Manufacturing          Percentage           H38
4632  Food Product Manufacturing          Percentage           H39
4633  Food Product Manufacturing          Percentage           H40
4634  Food Product Manufacturing          Percentage           H41
```

10. Visualizing DataFrame.

```
→ # Visualizing DataFrame

# Simple histogram for the 'Value' column

plt.figure(figsize=(8, 5))

plt.hist(df['Value'], bins=10, color='skyblue', edgecolor='black')

plt.title('Distribution of Value')

plt.xlabel('Value')

plt.ylabel('Frequency')

plt.show()
```


Distribution of Value
Frequency
Value