Python Coding Challenge

Name: PRIYESHWAR

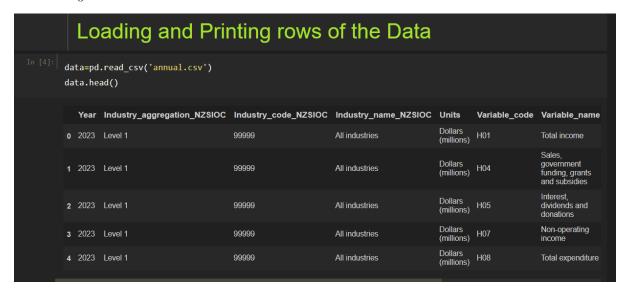
Mail: priyesh2664@gmail.com

Loading and Printing rows of the Data

Read the CSV file into a DataFrame and preview the first few rows using head().

data=pd.read_csv('annual.csv')

data.head()



Summary of Data Frame

Use info() to display data types and count of non-null values per column.

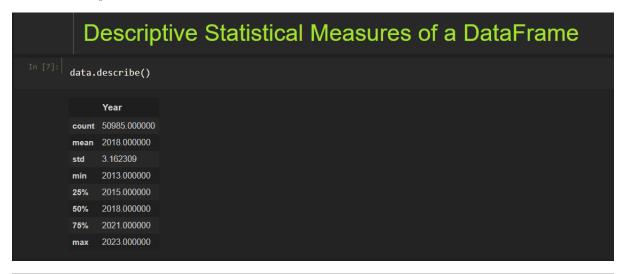
data.info()



Descriptive Statistical Measures of a DataFrame

Generate basic statistics like mean, median, and quartiles with describe().

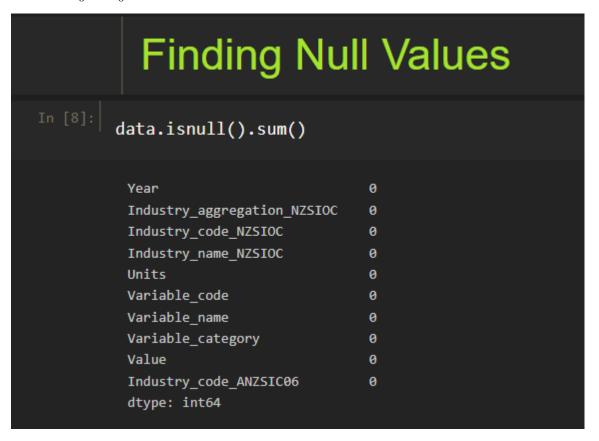
data.describe()



Finding Null Values

Check for missing data by summing nulls in each column using isnull().sum().

data.isnull().sum()



Missing Data Handing

Remove rows containing any null values with dropna().

data.dropna()

Missing Data Handing													
In [9]: data.dropna()													
		Year	Industry_aggregation_NZSIOC	Industry_code_NZSIOC	Industry_name_NZSIOC	Units	Variable_code	Variable_					
	0	2023	Level 1	99999	All industries	Dollars (millions)	H01	Total incon					
	1	2023	Level 1	99999	All industries	Dollars (millions)	H04	Sales, governmer funding, gr and subsid					
	2	2023	Level 1	99999	All industries	Dollars (millions)	H05	Interest, dividends a donations					
	3	2023	Level 1	99999	All industries	Dollars (millions)	H07	Non-opera income					
	4	2023	Level 1	99999	All industries	Dollars (millions)	H08	Total expe					
	50980	2013	Level 3	ZZ11	Food product manufacturing	Percentage	H37	Quick ratio					
	50981	2013	Level 3	ZZ11	Food product manufacturing	Percentage	H38	Margin on of goods for resale					
	50982	2013	Level 3	ZZ11	Food product manufacturing	Percentage	H39	Return on					
	50983	2013	Level 3	ZZ11	Food product manufacturing	Percentage	H40	Return on assets					
								Liabilities					

Sorting DataFrame values

Sort the DataFrame based on the 'Value' column using sort_values().

data.sort_values(by='Value')

S	ort	ing DataFram	e values											
data.sort_values(by='Value')														
	Year	Industry_aggregation_NZSIOC	Industry_code_NZSIOC	Industry_name_NZSIOC	Units	Variable_code	Varia							
27424	2018	Level 4	QQ111	Hospitals	Percentage	H40	Returnasset:							
47238	2013	Level 4	CC411	Printing	Percentage	H40	Return							
47202	2013	Level 3	CC41	Printing	Percentage	H40	Return							
3270	2023	Level 4	KK121	Life Insurance	Percentage	H40	Return							
32685	2016	Level 4	AA131	Dairy Cattle Farming	Percentage	H40	Return assets							
49887	2013	Level 4	LL122	Non-Residential Property Operation	Dollars (millions)	H27	Addition assets							
31347	2017	Level 4	LL122	Non-Residential Property Operation	Dollars (millions)	H27	Additional assets							
40219	2015	Level 3	KK11	Finance	Dollars (millions)	H26	Fixed assets							
40221	2015	Level 3	KK11	Finance	Dollars (millions)	H28	Dispo assets							
40618	2015	Level 4	LL122	Non-Residential Property Operation	Dollars (millions)	H28	Dispo asset							

Merge Data Frames

Combine sales and products tables on product id using inner, left, right, and outer joins.

Inner Join

```
sales = pd.read_csv("sales.csv")
products = pd.read_csv("products.csv")
```

```
inner = pd.merge(sales, products, on="product_id", how="inner")
print( inner)
```

```
Inner Join

In [17]: sales = pd.read_csv("sales.csv")
products = pd.read_csv("products.csv")

inner = pd.merge(sales, products, on="product_id", how="inner")
print( inner)

sale_id product_id quantity sale_date product_name category
0 1 101 2 2023-01-01 Laptop Electronics
1 6 101 1 2023-01-06 Laptop Electronics
2 2 102 1 2023-01-02 Keyboard Electronics
3 3 103 4 2023-01-03 Mouse Electronics
4 7 104 3 2023-01-07 Desk Furniture
```

Left Join

left = pd.merge(sales, products, on="product_id", how="left")
print(left)

```
Left Join
left = pd.merge(sales, products, on="product id", how="left")
print(left)
    sale_id product_id quantity sale_date product_name
                                                           category
                             2 2023-01-01 Laptop Electronics
                             1 2023-01-02
                                              Keyboard Electronics
                   102
                  103
                             4 2023-01-03
                                             Mouse Electronics
                             1 2023-01-04
2 2023-01-05
                                                   NaN
                  108
                                                                NaN
                  109
                                                                NaN
 4
                                                   NaN
                             NaN NaN
1 2023-01-06 Laptop Electronics
3 2023-01-07 Desk Furniture
                  101
                  104
 6
```

Right Join

right = pd.merge(sales, products, on="product_id", how="right")
print(right)

```
Right Join
right = pd.merge(sales, products, on="product_id", how="right")
print(right)
   sale_id product_id quantity sale_date product_name
                                                  category
      1.0 101 2.0 2023-01-01 Laptop Electronics
               101
                       1.0 2023-01-06
                                        Laptop Electronics
      6.0
      2.0
                102
                        1.0 2023-01-02 Keyboard Electronics
                                        Mouse Electronics
      3.0
               103
                       4.0 2023-01-03
                       3.0 2023-01-07
 4
      7.0
               104
                                         Desk Furniture
               105
                       NaN
                                         Chair Furniture
                                 NaN
                                       Monitor Electronics
      NaN
               106
                       NaN
                                 NaN
      NaN
                107
                       NaN
                                        Notebook Stationery
```

Outer Join

```
outer = pd.merge(sales, products, on="product_id", how="outer")
print( outer)
```

```
Outer Join
outer = pd.merge(sales, products, on="product id", how="outer")
print( outer)
      sale_id product_id quantity sale_date product_name
                                       2.0 2023-01-01 Laptop Electronics
          1.0
           6.0 101 1.0 2023-0
2.0 102 1.0 2023-0
3.0 103 4.0 2023-0
4.0 108 1.0 2023-0
5.0 109 2.0 2023-0
7.0 104 3.0 2023-0
NaN 105 NaN
NaN 106 NaN
NaN 107 NaN
                                             1.0 2023-01-06
            6.0
                                                                              Laptop Electronics
                                           1.0 2023-01-02 Keyboard Electronics
4.0 2023-01-03 Mouse Electronics
                                                                          Mouse Electronics
                                             1.0 2023-01-04
                                                                                  NaN

    1.0
    2023-01-04
    NaN
    NaN

    2.0
    2023-01-05
    NaN
    NaN

    3.0
    2023-01-07
    Desk
    Furniture

    NaN
    NaN
    Chair
    Furniture

    NaN
    NaN
    Monitor
    Electronics

                                                                               NaN
 8
                                                               NaN Notebook Stationery
```

Visualizing DataFrame

Plot total quantity sold over time with a line chart using Seaborn's lineplot().

```
import matplotlib.pyplot as plt
import seaborn as sns
plt.figure(figsize=(10, 6))
sns.lineplot(data=outer, x='sale_date', y='quantity', marker='o')
plt.title('Total Quantity Sold Over Time')
plt.xlabel('Sale Date')
plt.ylabel('Quantity Sold')
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```

