Python-Coding Challenge

Submitted By-

Subrat Shukla, DE Batch1

Dataset given : Annual enterprise survey: 2023 financial year (provisional)

Loading the given Dataset:

[1]:
```
#Loading the given Dataset
import pandas as pd

data = pd.read_csv("annual-enterprise-survey-2023-financial-year-provisional.csv")
data.head()
```

[1]:

	Year	Industry_aggregation_NZSIOC	Industry_code_NZSIOC	Industry_name_NZSIOC	Units	Variable_code	Variable_name	Variabl
0	2023	Level 1	99999	All industries	Dollars (millions)	H01	Total income	p
1	2023	Level 1	99999	All industries	Dollars (millions)	H04	Sales, government funding, grants and subsidies	p
2	2023	Level 1	99999	All industries	Dollars (millions)	H05	Interest, dividends and donations	p
3	2023	Level 1	99999	All industries	Dollars (millions)	H07	Non-operating income	p
4	2023	Level 1	99999	All industries	Dollars (millions)	H08	Total expenditure	p

Question 1: Printing Rows of the Data

[6]:
```
#Question 1: Printing Rows of the Data

#printing first 5 rows
data.head(5)
```

[6]:

	Year	Industry_aggregation_NZSIOC	Industry_code_NZSIOC	Industry_name_NZSIOC	Units	Vari
0	2023	Level 1	99999	All industries	Dollars (millions)	
1	2023	Level 1	99999	All industries	Dollars (millions)	
2	2023	Level 1	99999	All industries	Dollars (millions)	
3	2023	Level 1	99999	All industries	Dollars (millions)	
4	2023	Level 1	99999	All industries	Dollars (millions)	

[7]:
```
#printing last 5 rows
data.tail(5)
```

[7]:

	Year	Industry_aggregation_NZSIOC	Industry_code_NZSIOC	Industry_name_NZSIOC
50980	2013	Level 3	ZZ11	Food product manufacturing
50981	2013	Level 3	ZZ11	Food product manufacturing
50982	2013	Level 3	ZZ11	Food product manufacturing
50983	2013	Level 3	ZZ11	Food product manufacturing
50984	2013	Level 3	ZZ11	Food product manufacturing

Question 2: Printing the column names of the DataFrame

[11]:
```
#Question 2: Printing the column names of the DataFrame
list(data.columns)
```

[11]:
```
['Year',
 'Industry_aggregation_NZSIOC',
 'Industry_code_NZSIOC',
 'Industry_name_NZSIOC',
 'Units',
 'Variable_code',
 'Variable_name',
 'Variable_category',
 'Value',
 'Industry_code_ANZSIC06']
```

Question 3: Summary of Data Frame

[12]:
```
#Question 3: Summary of Data Frame
data.info()
```

```
<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
```

Question 4: Descriptive Statistical Measures of a DataFrame

[13]:
```
#Question 4: Descriptive Statistical Measures of a DataFrame
data.describe()
```

[13]:

	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

Question 5: Missing Data Handling

[31]:
```
#Question 5: Missing Data Handling

# Check for missing data
print("Missing Data : ")
print(data.isnull().sum())
```

```
Missing Data :
Year                          0
Industry_aggregation_NZSIOC   0
Industry_code_NZSIOC          0
Industry_name_NZSIOC          0
Units                         0
Variable_code                 0
Variable_name                 0
Variable_category             0
Value                         0
Industry_code_ANZSIC06        0
NewColumn                     0
Category                      0
NewCategory                   0
dtype: int64
```

[32]:
```
# Dropping rows with missing values
cleaned_data = data.dropna()

# Filling missing values with a default value (e.g., 0)
data_filled = data.fillna(0)
data_filled.head()
```

[32]:

	Year	Industry_aggregation_NZSIOC	Industry_code_NZSIOC	Industry_name_NZSIOC	Units	Variab
0	2023	Level 1	99999	All industries	Dollars (millions)	
1	2023	Level 1	99999	All industries	Dollars (millions)	
2	2023	Level 1	99999	All industries	Dollars (millions)	
3	2023	Level 1	99999	All industries	Dollars (millions)	
4	2023	Level 1	99999	All industries	Dollars (millions)	

Question 6: Sorting DataFrame values

[33]:
```
#Question 6: Sorting DataFrame values

# Sorting by first numerical column
numerical_column = data.select_dtypes(include='number').columns
if not numerical_column.empty:
    num_col = numerical_column[0]
    sorted_data = data.sort_values(by=num_col)
    print(f"\nData sorted by column '{num_col}':")
    print(sorted_data.head())
else:
    print("No numerical columns available.")
```

```
Data sorted by column 'Year':
       Year Industry_aggregation_NZSIOC Industry_code_NZSIOC  \
50968  2013                    Level 3                 ZZ11
50967  2013                    Level 3                 ZZ11
50966  2013                    Level 3                 ZZ11
50965  2013                    Level 3                 ZZ11
50964  2013                    Level 3                 ZZ11
```

```
            Industry_name_NZSIOC              Units Variable_code  \
50968  Food product manufacturing  Dollars (millions)           H25
50967  Food product manufacturing  Dollars (millions)           H24
50966  Food product manufacturing  Dollars (millions)           H23
50965  Food product manufacturing  Dollars (millions)           H22
50964  Food product manufacturing  Dollars (millions)           H21

                  Variable_name      Variable_category  Value  \
50968             Current assets     Financial position    0.0
50967               Total assets     Financial position    0.0
50966  Surplus before income tax  Financial performance    0.0
50965             Closing stocks  Financial performance    0.0
50964             Opening stocks  Financial performance    0.0

                                  Industry_code_ANZSIC06  NewColumn Category
50968  ANZSIC06 groups C111, C112, C113, C114, C115, ...     4052169      Low
50967  ANZSIC06 groups C111, C112, C113, C114, C115, ...     4052169      Low
50966  ANZSIC06 groups C111, C112, C113, C114, C115, ...     4052169      Low
50965  ANZSIC06 groups C111, C112, C113, C114, C115, ...     4052169      Low
50964  ANZSIC06 groups C111, C112, C113, C114, C115, ...     4052169      Low
```

Question 7: Apply Function

[25]:
```python
#Question 7: Apply Function
# Convert 'Value' column to numeric, replacing non-numeric values with NaN
data['Value'] = pd.to_numeric(data['Value'], errors='coerce')
data['Value'] = data['Value'].fillna(0)
def check_value(value):
    if value > 20000:
        return "High"
    else:
        return "Low"

data['Category'] = data['Value'].apply(check_value)
data.head()
```

Variable_name	Variable_category	Value	Industry_code_ANZSIC06	NewColumn	Category
Total income	Financial performance	930995.0	ANZSIC06 divisions A-S (excluding classes K633...	4092529	High
Sales, government funding, grants and subsidies	Financial performance	821630.0	ANZSIC06 divisions A-S (excluding classes K633...	4092529	High
Interest, dividends and donations	Financial performance	84354.0	ANZSIC06 divisions A-S (excluding classes K633...	4092529	High
Non-operating income	Financial performance	25010.0	ANZSIC06 divisions A-S (excluding classes K633...	4092529	High
Total expenditure	Financial performance	832964.0	ANZSIC06 divisions A-S (excluding classes K633...	4092529	High

Question 8: By using the lambda operator

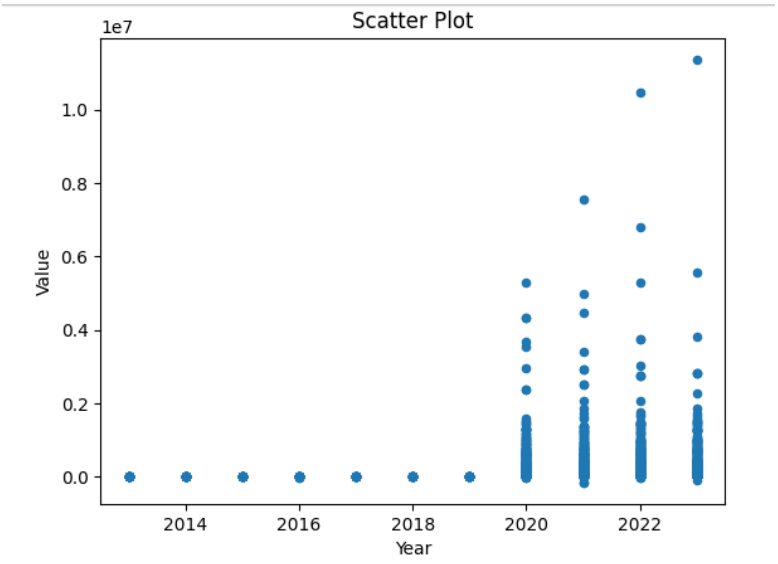
[26]:
```
#Question 8: By using the lambda operator
data['NewCategory'] = data['Value'].apply(lambda x: "High" if x > 20000 else "Low")
data.head()
```

Variable_category	Value	Industry_code_ANZSIC06	NewColumn	Category	NewCategory
Financial performance	930995.0	ANZSIC06 divisions A-S (excluding classes K633...	4092529	High	High
Financial performance	821630.0	ANZSIC06 divisions A-S (excluding classes K633...	4092529	High	High
Financial performance	84354.0	ANZSIC06 divisions A-S (excluding classes K633...	4092529	High	High
Financial performance	25010.0	ANZSIC06 divisions A-S (excluding classes K633...	4092529	High	High
Financial performance	832964.0	ANZSIC06 divisions A-S (excluding classes K633...	4092529	High	High

Question 9: Visualizing DataFrame

[27]:
```
#Question 9: Visualizing DataFrame
import matplotlib.pyplot as plt

data.plot(x=data.select_dtypes(include='number').columns[0],
          y=data.select_dtypes(include='number').columns[1],
          kind='scatter')
plt.title("Scatter Plot")
plt.show()
```



Question 10: What is the number of columns in the dataset?

[28]:
```
#Question 10: What is the number of columns in the dataset?
num_columns = data.shape[1]
num_columns
```

[28]:
```
13
```

Question 11: How is the dataset indexed?

[34]:
```
#Question 11:  How is the dataset indexed?
data.index
```

[34]:
```
RangeIndex(start=0, stop=50985, step=1)
```

--Thank You!