Ex - GroupBy

Introduction:

GroupBy can be summarized as Split-Apply-Combine.

Special thanks to: https://github.com/justmarkham for sharing the dataset and materials.

Check out this Diagram

Step 1. Import the necessary libraries

import pandas as pd

Step 2. Import the dataset from this <u>address</u>.

Step 3. Assign it to a variable called drinks.

drinks = pd.read_csv('drinks.csv')
drinks.head()

→		country	beer_servings	spirit_servings	wine_servings	total_lit
	0	Afghanistan	0	0	0	
	1	Albania	89	132	54	
	2	Algeria	25	0	14	
	3	Andorra	245	138	312	
	4	Angola	217	57	45	
	4 (

Step 4. Which continent drinks more beer on average?

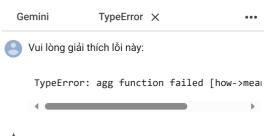
drinks.groupby('continent')['beer_servings'].mean().idxmax()



Step 5. For each continent print the statistics for wine consumption.

drinks.groupby('continent')['wine_servings'].describe()

_		count	mean	std	min	25%	50%	75%	max
	continent								
	AF	53.0	16.264151	38.846419	0.0	1.0	2.0	13.00	233.0
	AS	44.0	9.068182	21.667034	0.0	0.0	1.0	8.00	123.0
	EU	45.0	142.22222	97.421738	0.0	59.0	128.0	195.00	370.0
	ОС	16.0	35.625000	64.555790	0.0	1.0	8.5	23.25	212.0
	SA	12.0	62.416667	88.620189	1.0	3.0	12.0	98.50	221.0
	1								



Hypothesis

The error message "TypeError: Could not convert string 'AlgeriaAngolaBenin...' to numeric" indicates that the mean() function was applied to a column containing string values, specifically the 'country' column. The 'country' column was likely included in the grouping or aggregation process, preventing the calculation of the mean for numerical columns.

The traceback shows that the error occurs in cell 7 where drinks.groupby('continent').mean() is called. Since you're grouping by 'continent' and then calculating the mean, pandas tries to calculate the mean for all columns in the DataFrame, including the 'country' column which contains string data.

Suggested Changes



Explanation of changes:

By adding numeric_only=True to the mean() function, you tell pandas to only calculate the mean for numeric columns. This fix will skip non-numeric columns like 'country' during the calculation, resolving the TypeError.

Xếp hạng câu trả lời này 🏚 🗇

- Print the median alcohol consumption per continent for every column
- Okay, to print the median alcohol consumption per continent for every column, you can use the following code:

drinks.groupby('continent').median(numer:

Explanation:

- drinks.groupby('continent'): This groups the data in the drinks DataFrame by the 'continent' column.
- .median(numeric_only=True): This
 calculates the median for each numerical
 column within each continent group. Similar
 to the previous fix, the numeric_only=True
 is key to avoiding the TypeError by
 instructing pandas to consider only numeric
 columns

This code will output a DataFrame where each row represents a continent, and the columns show the

Step 6. Print the mean alcohol consumption per continent for every column

drinks.groupby('continent').mean(numeric_only=True)

→		beer_servings	spirit_servings	wine_servings	total_litres_
	continent				
	AF	61.471698	16.339623	16.264151	
	AS	37.045455	60.840909	9.068182	
	EU	193.777778	132.555556	142.222222	
	ОС	89.687500	58.437500	35.625000	
	SA	175.083333	114.750000	62.416667	
					>

Step 7. Print the median alcohol consumption per continent for every column

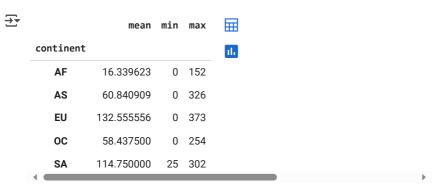
drinks.groupby('continent').median(numeric_only=True)

→		beer_servings	spirit_servings	wine_servings	total_litres_
	continent				
	AF	32.0	3.0	2.0	
	AS	17.5	16.0	1.0	
	EU	219.0	122.0	128.0	
	ОС	52.5	37.0	8.5	
	SA	162.5	108.5	12.0	•
					· · · · · · · · · · · · · · · · · · ·

Step 8. Print the mean, min and max values for spirit consumption.

This time output a DataFrame

drinks.groupby('continent')['spirit_servings'].agg(['mean', 'min', 'max'])



median values for 'beer_servings', 'spirit_servings', 'wine_servings', and 'total_litres_of_pure_alcohol' for that continent.

Xếp hạng câu trả lời này 凸 切



- Print the mean, min and max values for spirit consumption.
- Okay, to print the mean, min, and max values for spirit consumption, and output the result as a DataFrame, you can use the agg function with a dictionary to specify the desired aggregations:

drinks.groupby('continent')['spirit_serv:

Hãy thận trọng khi sử dụng mã

Explanation:

- 1. drinks.groupby('continent') ['spirit_servings']: This selects the 'spirit_servings' column and groups the data by 'continent'.
- 2. .agg(['mean', 'min', 'max']):This applies the specified aggregation functions ('mean', 'min', 'max') to the 'spirit_servings' column within each continent group. The result will be a DataFrame with the aggregated values for each continent.

This code will output a DataFrame with 'continent' as the index, and columns for 'mean', 'min', and 'max' representing the respective values for spirit consumption within each continent.

Xếp hạng câu trả lời này 🖒 🗇

Nhập câu lệnh tại đây



0/2000

Gemini có thể mắc sai sót, vì vậy, hãy xác minh câu trả lời và thận trọng khi sử dụng mã. $\underline{\text{Tìm}}$ hiểu thêm