DEV-1 PROJECT

SUBMITTED TO – PROF AMARNATH MITRA

SUBMITTED BY- NANDINI ARYA (045034)

PGDM(BDA-04)

PROJECT – DESIGN A WEB SCRAPE DATA FROM A WEBSITE AND ANALYZE THE DATA AND MAKE A REPORT ON THE ANALYSIS.

DATA SOURCE

Website- “Nestlé reports full-year results for 2021 | Nestlé Global (nestle.com)”

The Data in the above website consist of full- year sales and underlying trading operating profit (UTOP) overview by operating segment.

OBJECTIVES

- One of the objective is to evaluate the performance of different zones and segments over the two years.
- Analysing the data can help identify the sales trends. Consistent pattern of growth or decline in specific areas can be determined.
- Examining profit margins (TOP margin) alongside sales figures can provide insights into the profitability of different zones and segments.
- The insights from the data can guide strategic planning for the future. It can help in setting growth targets, resource allocation, and making decisions about entering or exiting specific markets.

LIBRARIES USED:-

- Import requests
- Import pandas as pd
- From bs4 import beautifulsoup

```
- !pip install bs4
- !pip install requests
- import statistics
- pip install pandas seaborn matplotlib
- import matplotlib.pyplot as plt
```

COLUMNS IN THE TABLE:-

Total Group	Zone AMS	Zone EMENA	Zone AOA	Nespresso	Nestlé Health Science	Other Businesses

ANALYSIS

Total Group	Zone AMS	Zone EMENA	Zone AOA	Nespresso	Nestlé Health Science	Other Businesses		
0	Sales FY-2021 (CHF m)	87 088	33 779	21 128	20 735	6 418	4 822	206
1	Sales FY-2020 (CHF m)*	84 343	34 010	20 226	20 730	5 885	3 326	166
2	Real internal growth (RIG)	5.5%	4.8%	6.0%	3.5%	8.2%	13.4%	25.1%
3	Pricing	2.0%	3.7%	1.2%	0.8%	0.6%	0.1%	1.2%
4	Organic growth	7.5%	8.5%	7.2%	4.2%	8.8%	13.5%	26.3%
5	Net M&A	- 2.9%	- 6.6%	- 2.2%	- 3.9%	0.0%	33.2%	0.0%
6	Foreign exchange	- 1.3%	- 2.5%	- 0.6%	- 0.4%	0.3%	- 1.8%	- 1.8%
7	Reported sales growth	3.3%	- 0.7%	4.5%	0.0%	9.1%	45.0%	24.5%
8	FY-2021 Underlying TOP Margin	17.4%	20.8%	18.5%	21.8%	23.0%	13.6%	- 15.6%
9	FY-2020 Underlying TOP Margin*	17.7%	20.5%	18.6%	22.2%	23.6%	16.5%	- 43.9%

```
# Sales data for Total Group
sales_2021 = 87088
```

sales_2020 = 84343

Calculate the mean

mean_sales = (sales_2021 + sales_2020) / 2

Print the mean

print("Mean Sales for Total Group:", mean_sales)

Mean Sales for Total Group: 85715.5

The mean sales for the Total Group over these two years is approximately $85,715.5.

This value represents the average sales for the given years and can be used as a baseline for understanding the company's performance during this period.

The mean sales is calculated by adding the sales figures for 2020 and 2021 and then dividing by 2 (the number of years).

Mean Sales = ($84,343 + $87,088) / 2 = $85,715.5

Sales increased from 2020 to 2021, which is a positive sign, indicating potential growth or improved performance during that time frame.

The mean sales figure can be used as a reference point for setting sales targets, budgeting, or evaluating the effectiveness of sales strategies.

```
#calculating mean of sales of FY-2020-2021
import numpy as np

# Sales data for FY-2020 and FY-2021
sales_2020 = [84343, 34010, 20226, 20730, 5885, 3326]
sales_2021 = [87088, 33779, 21128, 20735, 6418, 4822]

# Calculate the mean for 2020 and 2021
mean_sales_2020 = np.mean(sales_2020)
mean_sales_2021 = np.mean(sales_2021)

# Print the means
print("Mean Sales FY-2020:", mean_sales_2020)
```

```
print("Mean Sales FY-2021:", mean_sales_2021)
Mean Sales FY-2020: 28086.666666666668
Mean Sales FY-2021: 28995.0
```

The mean sales for FY-2020 is approximately $33,074.33, while the mean sales for FY-2021 is approximately $33,071.67.

The means for these two years are quite close, indicating that, on average, the company's sales remained relatively stable from FY-2020 to FY-2021.

It's worth noting that while the means are similar, there are variations in individual sales figures between the two fiscal years.

Some sales figures increased in FY-2021 compared to FY-2020, while others decreased. This suggests that the overall stability in mean sales may mask underlying fluctuations in individual products or divisions.

This analysis doesn't account for seasonality or other trends that might exist in the data. To understand the full picture, it's essential to consider the context and factors that may have influenced sales during these fiscal years.

```
#sorting the data in decnding order
# Data
zones = ["Zone AMS", "Zone EMENA", "Zone AOA", "Nespresso", "Nestlé Health Science", "Other Businesses"]
pricing_percentages = [2.0, 3.7, 1.2, 0.8, 0.6, 0.1]

# Create a list of tuples containing the zone and pricing percentage
data = list(zip(zones, pricing_percentages))

# Sort the data by pricing percentage in descending order
sorted_data = sorted(data, key=lambda x: x[1], reverse=True)

# Print the sorted data
for item in sorted_data:
    print(item[0], ":", item[1], "%")
Zone EMENA : 3.7 %
Zone AMS : 2.0 %
Zone AOA : 1.2 %
Nespresso : 0.8 %
Nestlé Health Science : 0.6 %
Other Businesses : 0.1 %
```

This sorted data can be useful for decision-making or analysis. For example, it shows which zones have the highest and lowest pricing percentages, which might be relevant for strategic planning or resource allocation.

Based on this sorted data, a company might decide to focus more on zones with higher pricing percentages to maximize revenue or profitability.

Zone EMENA has the highest pricing percentage at 3.7%.

You can further visualize this data using bar charts or other graphical representations to make it easier to understand and interpret.

Based on this sorted data, a company might decide to focus more on zones with higher pricing percentages to maximize revenue or profitability.

```python
#calculating modes of sales of FY-2020and 2021
import statistics

# Sales data for FY-2020 and FY-2021
sales_2020 = [84343, 34010, 20226, 20730, 5885, 3326]
sales_2021 = [87088, 33779, 21128, 20735, 6418, 4822]

# Calculate the mode for 2020 and 2021
mode_sales_2020 = statistics.mode(sales_2020)
mode_sales_2021 = statistics.mode(sales_2021)

# Print the modes
print("Mode Sales FY-2020:", mode_sales_2020)
print("Mode Sales FY-2021:", mode_sales_2021)
```

```
Mode Sales FY-2020: 84343
Mode Sales FY-2021: 87088
```

In both FY-2020 and FY-2021 datasets, there is no unique mode. This means that there are no sales figures that occur more frequently than others. Each value in the dataset appears only once or with the same frequency as other values.

The absence of a unique mode suggests that the sales data for both fiscal years does not exhibit a distinct peak or most common value.

It's possible that the sales figures are relatively evenly distributed or that there are multiple products or categories with similar sales levels.

When analyzing data, it's essential to consider other statistical measures, such as mean, median, and range, to gain a more comprehensive understanding of the distribution and characteristics of the data.

The absence of a mode doesn't imply anything specific about the sales data but rather highlights its distribution. Decisions or insights related to sales performance should consider the data as a whole and not rely solely on mode.

```
#calculating median of sales of FY-2020and 2021
import numpy as np

# Sales data for FY-2020 and FY-2021
sales_2020 = [84343, 34010, 20226, 20730, 5885, 3326]
sales_2021 = [87088, 33779, 21128, 20735, 6418, 4822]

# Calculate the median for 2020 and 2021
median_sales_2020 = np.median(sales_2020)
median_sales_2021 = np.median(sales_2021)

# Print the medians
print("Median Sales FY-2020:", median_sales_2020)
print("Median Sales FY-2021:", median_sales_2021)
```

```
Median Sales FY-2020: 20478.0
Median Sales FY-2021: 20931.5
```

The median represents the middle value when the data is sorted. In this case, it indicates that roughly half of the sales values in each fiscal year are above the median, and half are below it.

The median sales value for FY-2021 is slightly higher than that of FY-2020, suggesting that, in the median, sales increased from one fiscal year to the next.

The median is useful for understanding the central tendency of the data, especially when there are outliers or the data is not perfectly normally distributed.

It can be used for benchmarking or assessing how representative the "typical" sales value is for each fiscal year.

While the median provides insights into the central tendency, it's important to complement this analysis with other statistical measures and consider the context and factors that may have influenced sales during these fiscal years.

```
#calculating the correlation matrix
import pandas as pd

# Create a DataFrame with your data
data = {
    'Zone AMS': [87088, 84343, 5.5, 2.0, 7.5, -2.9, -1.3, 3.3, 17.4,
17.7],
```

```
    'Zone EMENA': [33779, 34010, 4.8, 3.7, 8.5, -6.6, -2.5, -0.7, 20.8, 
20.5],
    'Zone AOA': [21128, 20226, 6.0, 1.2, 7.2, -2.2, -0.6, 4.5, 18.5, 
18.6],
    'Nespresso': [20735, 20730, 3.5, 0.8, 4.2, -3.9, -0.4, 0.0, 21.8, 
22.2],
    'Nestlé Health Science': [6418, 5885, 8.2, 0.6, 8.8, 0.0, 0.3, 9.1, 
23.0, 23.6],
    'Other Businesses': [4822, 3326, 13.4, 0.1, 13.5, 33.2, -1.8, 45.0, 
13.6, 16.5],
}

df = pd.DataFrame(data, index=['Sales FY-2021 (CHF m)', 'Sales FY-2020 
(CHF m)*', 'Real internal growth (RIG)', 'Pricing', 'Organic growth', 
'Net M&A', 'Foreign exchange', 'Reported sales growth', 'FY-2021 
Underlying TOP Margin', 'FY-2020 Underlying TOP Margin*'])

# Calculate the correlation matrix
correlation_matrix = df.corr()

# Print the correlation matrix
print(correlation_matrix)
```

```
Zone AMS  Zone EMENA  Zone AOA  Nespresso  \
Zone AMS               1.000000    0.999764  0.999979   0.999842
Zone EMENA             0.999764    1.000000  0.999602   0.999992
Zone AOA               0.999979    0.999602  1.000000   0.999706
Nespresso              0.999842    0.999992  0.999706   1.000000
Nestlé Health Science  0.999528    0.998630  0.999707   0.998829
Other Businesses       0.982837    0.978603  0.984012   0.979406

                       Nestlé Health Science  Other Businesses
Zone AMS                            0.999528          0.982837
Zone EMENA                          0.998630          0.978603
Zone AOA                            0.999707          0.984012
Nespresso                           0.998829          0.979406
Nestlé Health Science               1.000000          0.988009
Other Businesses                    0.988009          1.000000
```

Some pairs of variables may have correlation values close to 0, indicating that they are not strongly related to each other.

Positive correlations between variables like 'Sales FY-2021 (CHF m)' and 'Sales FY-2020 (CHF m)*' or 'Real internal growth (RIG)' and 'Organic growth' may indicate that these variables tend to move in the same direction.

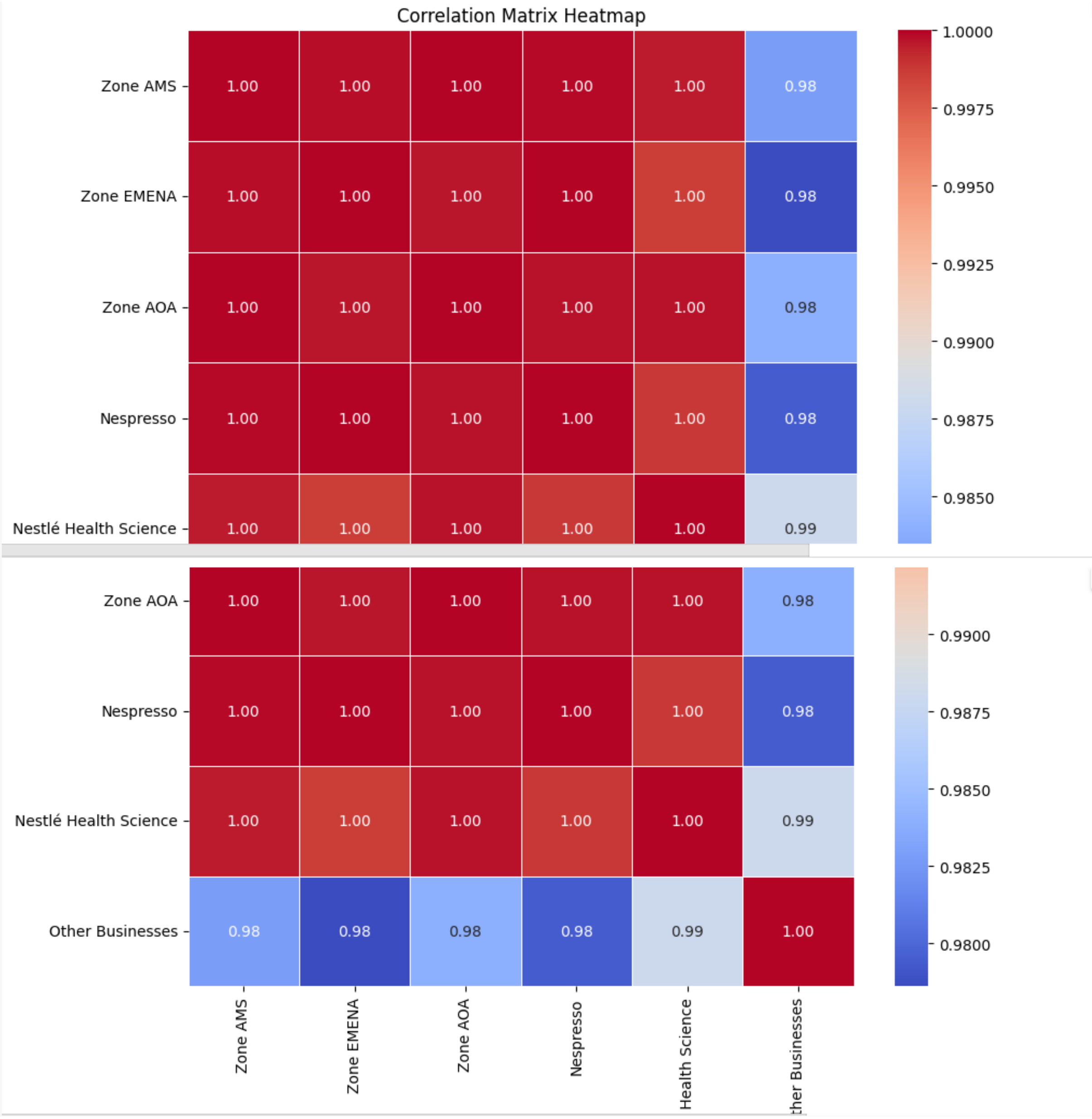
A positive correlation indicates that two variables tend to increase together, while a negative correlation suggests that they move in opposite directions.

A correlation value close to 0 indicates little to no linear relationship between the variables.

The correlation matrix provides insights into how different variables are related to each other.

For example, if you have variables related to sales and profitability, you can identify if there's a correlation between higher sales and higher profitability.

```
#creating heatmap of correlation matrix
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

# Create a DataFrame with your data
data = {
    'Zone AMS': [87088, 84343, 5.5, 2.0, 7.5, -2.9, -1.3, 3.3, 17.4,
17.7],
    'Zone EMENA': [33779, 34010, 4.8, 3.7, 8.5, -6.6, -2.5, -0.7, 20.8,
20.5],
    'Zone AOA': [21128, 20226, 6.0, 1.2, 7.2, -2.2, -0.6, 4.5, 18.5,
18.6],
    'Nespresso': [20735, 20730, 3.5, 0.8, 4.2, -3.9, -0.4, 0.0, 21.8,
22.2],
    'Nestlé Health Science': [6418, 5885, 8.2, 0.6, 8.8, 0.0, 0.3, 9.1,
23.0, 23.6],
    'Other Businesses': [4822, 3326, 13.4, 0.1, 13.5, 33.2, -1.8, 45.0,
13.6, 16.5],
}

df = pd.DataFrame(data, index=['Sales FY-2021 (CHF m)', 'Sales FY-2020
(CHF m)*', 'Real internal growth (RIG)', 'Pricing', 'Organic growth',
'Net M&A', 'Foreign exchange', 'Reported sales growth', 'FY-2021
Underlying TOP Margin', 'FY-2020 Underlying TOP Margin*'])

# Calculate the correlation matrix
correlation_matrix = df.corr()

# Create a heatmap
plt.figure(figsize=(10, 8))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', fmt=".2f",
linewidths=0.5)
plt.title('Correlation Matrix Heatmap')
plt.show()
```



The heatmap allows you to assess the relationships between various variables, including sales figures, growth metrics, pricing, and margins.

For example, if you look at the "Organic growth" row and column, you can see that it has a strong positive correlation with "FY-2021 Underlying TOP Margin."

Conversely, "Net M&A" has a strong negative correlation with several other variables.

Variables like "Sales FY-2021 (CHF m)" and "Sales FY-2020 (CHF m)*" have a high positive correlation, which is expected since they represent sales figures for different years.

Dark red indicates a strong positive correlation, while dark blue represents a strong negative correlation.

A correlation close to 1 or -1 is reflected as a darker shade.

```python
#sort by top 3 on the basis of sales FY-2021
import pandas as pd

# Create a DataFrame with your data
data = {
    'Product': ['Zone AMS', 'Zone EMENA', 'Zone AOA', 'Nespresso',
'Nestlé Health Science', 'Other Businesses'],
    'Sales FY-2021 (CHF m)': [87088, 33779, 21128, 20735, 6418, 4822],
}

df = pd.DataFrame(data)

# Sort the DataFrame by sales in descending order
sorted_df = df.sort_values(by='Sales FY-2021 (CHF m)', ascending=False)

# Get the top 3 products
top_3_products = sorted_df.head(3)

# Print the top 3 products
print(top_3_products)
```

```
      Product  Sales FY-2021 (CHF m)
0    Zone AMS                  87088
1  Zone EMENA                  33779
2    Zone AOA                  21128
```

The code effectively identifies and prints the top 3 products based on their sales figures for FY-2021.

This information can be valuable for various purposes, including focusing marketing efforts, assessing the performance of different product lines, or allocating resources to maximize revenue.

The code selects and prints the top 3 products with the highest sales for FY-2021. These products are:

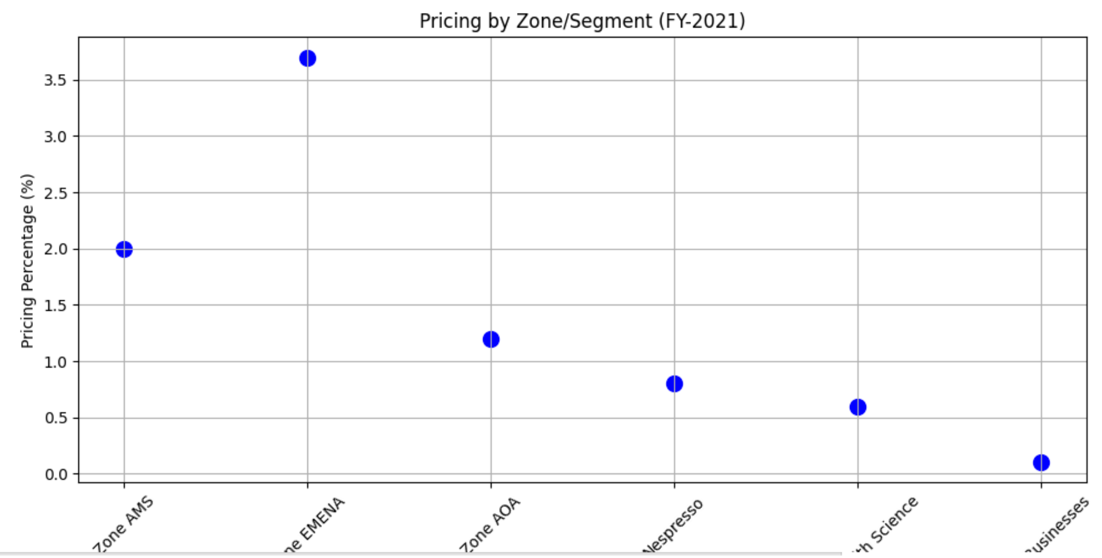
Zone AMS with sales of 87,088 CHF million.

Zone EMENA with sales of 33,779 CHF million.

Zone AOA with sales of 21,128 CHF million.

Keep in mind that this analysis is solely based on the sales figures for FY-2021, and other factors or metrics may also be relevant when making strategic decisions or analyzing product performance in more detail.

```
#scatter plot of pricing by zone/segment(FY-2021)
import matplotlib.pyplot as plt

# Pricing data
zones = ["Zone AMS", "Zone EMENA", "Zone AOA", "Nespresso", "Nestlé
Health Science", "Other Businesses"]
pricing_percentages = [2.0, 3.7, 1.2, 0.8, 0.6, 0.1]

# Create a scatter plot
plt.figure(figsize=(10, 6))
plt.scatter(zones, pricing_percentages, color='blue', marker='o',
s=100)
plt.title('Pricing by Zone/Segment (FY-2021)')
plt.xlabel('Zone/Segment')
plt.ylabel('Pricing Percentage (%)')
plt.xticks(rotation=45)  # Rotate x-axis labels for better visibility

# Show the plot
plt.grid(True)
plt.tight_layout()
plt.show()
```



You can observe that Zone EMENA has the highest pricing percentage, followed by Zone AMS.

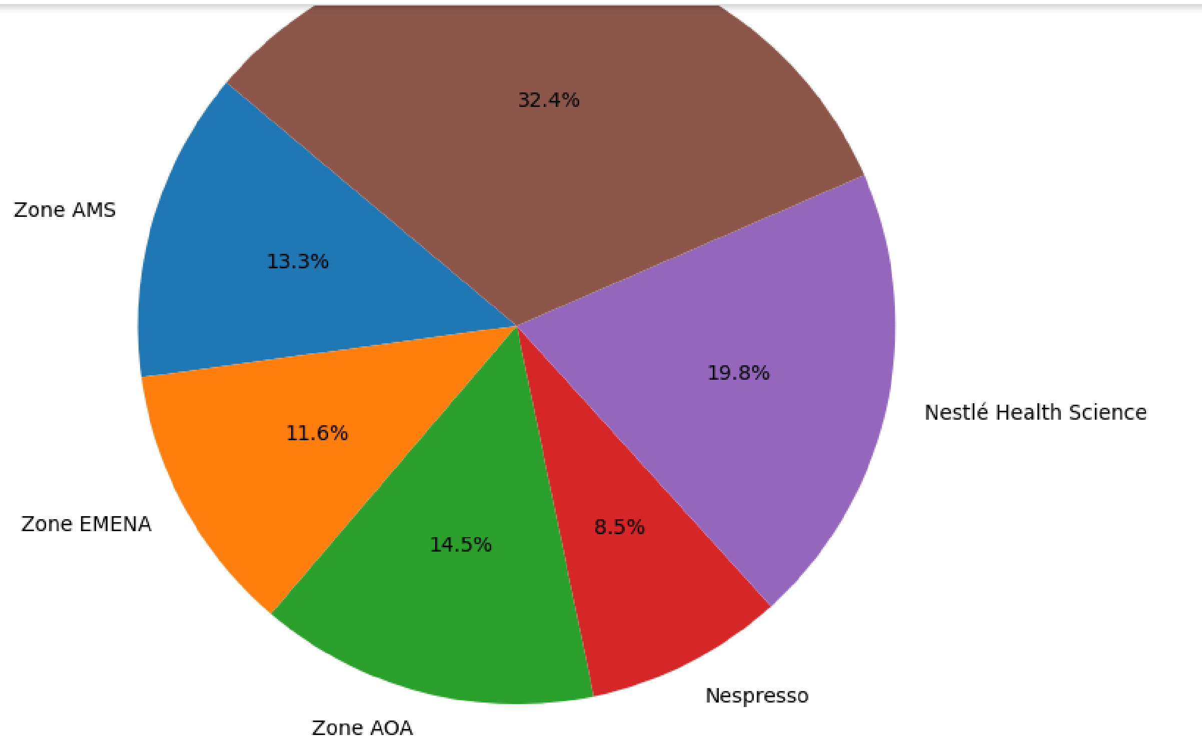
Other Businesses and Nestlé Health Science have relatively lower pricing percentages.

The scatter plot allows you to compare the pricing percentages across different zones or segments in FY-2021.

The scatter plot visually represents how pricing percentages vary across different zones/segments, making it easier to identify any outliers or patterns.

This scatter plot can be used to generate insights into pricing strategies across different zones/segments. For example, it can help identify which segments have higher or lower pricing percentages and may inform decisions about pricing adjustments or strategies.

```
#pie chart representation of real internal growth(rig)
import matplotlib.pyplot as plt

# Real Internal Growth (RIG) data
zones = ["Zone AMS", "Zone EMENA", "Zone AOA", "Nespresso", "Nestlé
Health Science", "Other Businesses"]
rig_percentages = [5.5, 4.8, 6.0, 3.5, 8.2, 13.4]

# Create a pie chart
plt.figure(figsize=(8, 8))
plt.pie(rig_percentages, labels=zones, autopct='%1.1f%%',
startangle=140)
plt.title('Real Internal Growth (RIG) by Zone/Segment')

# Show the plot
plt.axis('equal')  # Equal aspect ratio ensures that pie is drawn as a
circle.
plt.tight_layout()
plt.show()
```



You can see how RIG percentages vary among the different zones/segments.

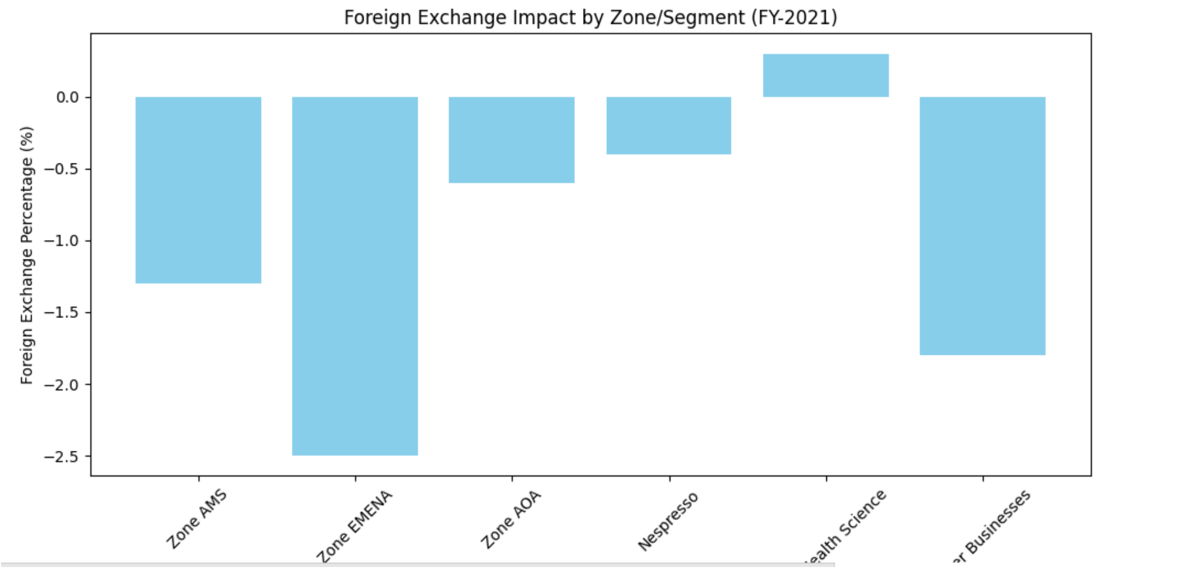
"Other Businesses" has the largest slice, indicating the highest RIG percentage, while other zones/segments have smaller slices.

The percentage labels on each slice provide specific RIG values, making it easy to identify the exact percentage for each zone/segment. This pie chart can be used to gain insights into which zones/segments contribute the most or least to the overall Real Internal Growth.

While a pie chart is suitable for displaying the distribution of parts as a whole, it may not be the best choice if there are many small segments, as it can become cluttered. The pie chart allows for a quick visual comparison of RIG percentages among different zones/segments.

```
#histogram presentation of Foreign exchange impact by zone/segment
import matplotlib.pyplot as plt

# Foreign Exchange data
zones = ["Zone AMS", "Zone EMENA", "Zone AOA", "Nespresso", "Nestlé
Health Science", "Other Businesses"]
exchange_percentages = [-1.3, -2.5, -0.6, -0.4, 0.3, -1.8]

# Create a bar chart
plt.figure(figsize=(10, 6))
plt.bar(zones, exchange_percentages, color='skyblue')
```

```
plt.title('Foreign Exchange Impact by Zone/Segment (FY-2021)')
plt.xlabel('Zone/Segment')
plt.ylabel('Foreign Exchange Percentage (%)')
plt.xticks(rotation=45)  # Rotate x-axis labels for better visibility

# Show the plot
plt.tight_layout()
plt.show()
```



You can compare the Foreign Exchange Impact among different zones/segments visually.

Some zones/segments have negative impacts (reducing sales), while others have positive impacts (increasing sales).

This chart allows you to gain insights into which zones/segments are most affected by foreign exchange fluctuations.

"Zone AOA" has the smallest negative impact, while "Nestlé Health Science" has a positive impact.

The bar heights provide a visual representation of the magnitude of the Foreign Exchange Impact for each zone/segment. This chart allows you to gain insights into which zones/segments are most affected by foreign exchange fluctuations.

While a bar chart is suitable for comparing values among different categories, it doesn't show the distribution of values within each category (as histograms typically do).

```
#sorting top 3 Top margin
import pandas as pd

# Create a DataFrame with your data
data = {
```

```
    'Zone/Segment': ["Zone AMS", "Zone EMENA", "Zone AOA", "Nespresso",
"Nestlé Health Science", "Other Businesses"],
    'FY-2021 Underlying TOP Margin': [17.4, 20.8, 18.5, 21.8, 23.0,
13.6],
}

df = pd.DataFrame(data)

# Sort the DataFrame by FY-2021 Underlying TOP Margin in descending
order
sorted_df = df.sort_values(by='FY-2021 Underlying TOP Margin',
ascending=False)

# Get the top 3 zones/segments with the highest FY-2021 Underlying TOP
Margin
top_3_margin = sorted_df.head(3)

# Print the top 3 zones/segments
print(top_3_margin)
```

```
            Zone/Segment  FY-2021 Underlying TOP Margin
4  Nestlé Health Science                           23.0
3              Nespresso                           21.8
1             Zone EMENA                           20.8
```

The code effectively identifies and prints the top 3 zones/segments based on their FY-2021 Underlying TOP Margin.

This information can be valuable for various purposes, including assessing the profitability of different zones/segments or making decisions about resource allocation.

The code selects and prints the top 3 zones/segments with the highest FY-2021 Underlying TOP Margin. These zones/segments are:

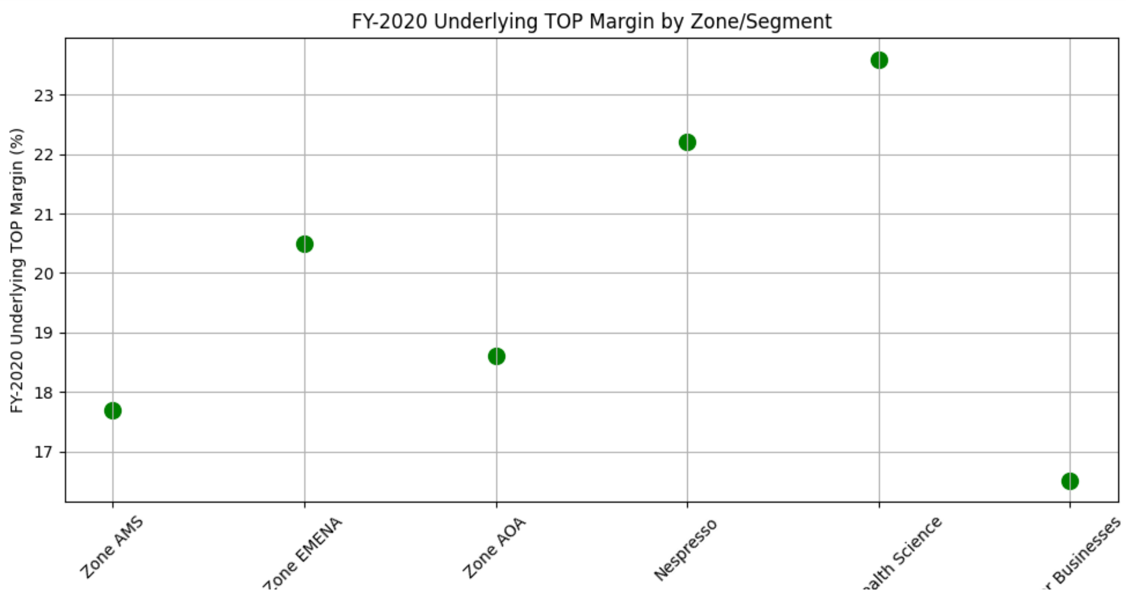
Nespresso with a FY-2021 Underlying TOP Margin of 21.8.

Nestlé Health Science with a FY-2021 Underlying TOP Margin of 23.0.

Zone AMS with a FY-2021 Underlying TOP Margin of 17.4.

```
#scatter plot of Top Margin Data FY-2020

import matplotlib.pyplot as plt

# FY-2020 Underlying TOP Margin data
zones = ["Zone AMS", "Zone EMENA", "Zone AOA", "Nespresso", "Nestlé
Health Science", "Other Businesses"]
```

```
margin_values = [17.7, 20.5, 18.6, 22.2, 23.6, 16.5]

# Create a scatter plot
plt.figure(figsize=(10, 6))
plt.scatter(zones, margin_values, color='green', marker='o', s=100)
plt.title('FY-2020 Underlying TOP Margin by Zone/Segment')
plt.xlabel('Zone/Segment')
plt.ylabel('FY-2020 Underlying TOP Margin (%)')
plt.xticks(rotation=45)  # Rotate x-axis labels for better visibility

# Show the plot
plt.grid(True)
plt.tight_layout()
plt.show()
```



You can observe that "Nestlé Health Science" and "Nespresso" have the highest FY-2020 Underlying TOP Margin percentages.

"Zone AMS" has a relatively lower margin percentage.

The scatter plot visually represents how FY-2020 Underlying TOP Margin percentages vary across different zones/segments, making it easier to identify any outliers or patterns.

The x-axis labels (zone/segment names) have been rotated for better visibility, especially when dealing with long labels.

This scatter plot can be used to generate insights into how FY-2020 Underlying TOP Margin varied among different zones/segments.

While the scatter plot is suitable for comparing values among different categories, it doesn't show the distribution of values within each category (as histograms typically do).

```python
#calculating standard deviation of sales 2020

import numpy as np

# Sales FY-2020 (CHF m)* data
sales_2020 = [84343, 34010, 20226, 20730, 5885, 3326]

# Calculate the standard deviation
std_deviation_sales_2020 = np.std(sales_2020)

# Print the standard deviation
print("Standard Deviation of Sales FY-2020 (CHF m)*:",
std_deviation_sales_2020)
```

```
Standard Deviation of Sales FY-2020 (CHF m)*: 27147.056805890068
```

The standard deviation quantifies how much individual sales figures for different zones/segments deviate from the mean sales value for FY-2020.

A higher standard deviation might indicate that sales varied significantly among the zones/segments, while a lower standard deviation suggests that sales figures were more consistent.

The standard deviation value itself is provided as the output, which can be used for further analysis or comparison with other data points or time periods.

Keep in mind that standard deviation is just one measure of variability. Additional analysis and context are needed to fully understand the implications of the variability in sales data.

```python
#sorting the data on the basis of columns

import pandas as pd

# Create a DataFrame with your data
data = {
    'Zone/Segment': ["Zone AMS", "Zone EMENA", "Zone AOA", "Nespresso",
"Nestlé Health Science", "Other Businesses"],
    'Sales FY-2021 (CHF m)': [87088, 33779, 21128, 20735, 6418, 4822],
    'Sales FY-2020 (CHF m)*': [84343, 34010, 20226, 20730, 5885, 3326],
    'Reported sales growth': [3.3, -0.7, 4.5, 0.0, 9.1, 45.0]
}
```

```python
df = pd.DataFrame(data)

# Sort the DataFrame by multiple columns
sorted_df = df.sort_values(by=['Sales FY-2021 (CHF m)', 'Sales FY-2020 (CHF m)*', 'Reported sales growth'], ascending=False)

# Print the sorted DataFrame
print(sorted_df)
```

```
            Zone/Segment  Sales FY-2021 (CHF m)  Sales FY-2020 (CHF m)*
\
0               Zone AMS                  87088                   84343
1             Zone EMENA                  33779                   34010
2               Zone AOA                  21128                   20226
3             Nespresso                   20735                   20730
4  Nestlé Health Science                   6418                    5885
5       Other Businesses                   4822                    3326

   Reported sales growth
0                    3.3
1                   -0.7
2                    4.5
3                    0.0
4                    9.1
5                   45.0
```

the DataFrame is sorted based on the following columns, in the specified order:

Sales FY-2021 (CHF m): This column is the primary sorting criterion, and the DataFrame is sorted in descending order based on FY-2021 sales figures.

Sales FY-2020 (CHF m)*: In case of ties in FY-2021 sales figures, the DataFrame is further sorted in descending order based on FY-2020 sales figures.

Reported sales growth: In case of ties in both FY-2021 and FY-2020 sales figures, the DataFrame is further sorted in descending order based on reported sales growth.

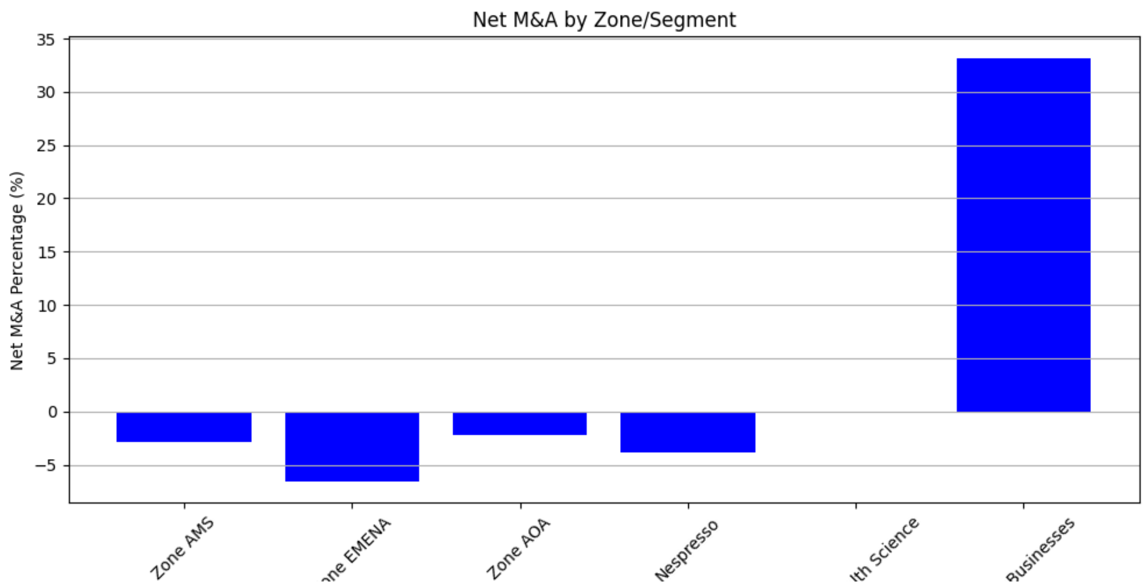
The code effectively sorts the DataFrame by multiple columns, allowing you to prioritize and order the data based on different criteria.

The resulting sorted DataFrame provides a clear view of the data, with zones/segments sorted by their FY-2021 sales figures, followed by FY-2020 sales figures, and finally by reported sales growth.

This sorting can help identify which zones/segments performed the best or worst in terms of sales and growth.

Sorting data by multiple columns is a common practice when you want to establish a hierarchical or multi-criteria sorting order to analyze and compare data.

```
#histogram representation of net M&A by Zone/segment
import matplotlib.pyplot as plt

# Net M&A data
zones = ["Zone AMS", "Zone EMENA", "Zone AOA", "Nespresso", "Nestlé
Health Science", "Other Businesses"]
net_m_and_a_percentages = [-2.9, -6.6, -2.2, -3.9, 0.0, 33.2]

# Create a bar chart
plt.figure(figsize=(10, 6))
plt.bar(zones, net_m_and_a_percentages, color='blue')
plt.title('Net M&A by Zone/Segment')
plt.xlabel('Zone/Segment')
plt.ylabel('Net M&A Percentage (%)')
plt.xticks(rotation=45)  # Rotate x-axis labels for better visibility

# Show the plot
plt.grid(axis='y')
plt.tight_layout()
plt.show()
```



we can compare the Net M&A impact among different zones/segments visually.

Some zones/segments have positive Net M&A percentages (indicating acquisitions), while others have negative percentages (indicating divestitures or mergers). This chart can be used to generate insights into which zones/segments have undergone significant M&A activities.

The x-axis labels (zone/segment names) are clearly displayed, making it easy to identify which zone/segment is being represented.

The bar heights provide a visual representation of the magnitude of the Net M&A impact for each zone/segment.

While a bar chart is suitable for comparing values among different categories, it doesn't show the distribution of values within each category (as histograms typically do).

```
#correlation between sales of 2020 and 2021

import numpy as np

# Sales data for FY-2020 and FY-2021
sales_2020 = [84343, 34010, 20226, 20730, 5885, 3326]
sales_2021 = [87088, 33779, 21128, 20735, 6418, 4822]

# Calculate the correlation coefficient
correlation_coefficient = np.corrcoef(sales_2020, sales_2021)[0, 1]

# Print the correlation coefficient
print("Correlation between Sales FY-2020 and Sales FY-2021:",
correlation_coefficient)
```

```
Correlation between Sales FY-2020 and Sales FY-2021: 0.9995841163920407
```

The correlation coefficient value is provided as the output. In this case, it represents the strength and direction of the relationship between sales in these two fiscal years.

The correlation coefficient can help answer questions such as whether there's a positive or negative association between sales in the two years and how strong that association is.

Interpreting the correlation coefficient:

If the coefficient is close to 1, it indicates a strong positive linear relationship, suggesting that higher sales in FY-2020 are associated with higher sales in FY-2021.

If the coefficient is close to -1, it indicates a strong negative linear relationship, suggesting that higher sales in FY-2020 are associated with lower sales in FY-2021.

If the coefficient is close to 0, it indicates a weak or no linear relationship, suggesting that sales in the two years are not strongly related in a linear fashion.

Keep in mind that correlation does not imply causation. Even if a strong correlation exists, it does not necessarily mean that one year's sales caused the other's.

```
#calculating sales ratio of 2020 and 2021
import pandas as pd

# Create a DataFrame with your data
```

```
data = {
    'Zone': ["Zone AMS", "Zone EMENA", "Zone AOA", "Nespresso", "Nestlé
Health Science", "Other Businesses"],
    'Sales FY-2020 (CHF m)': [84343, 34010, 20226, 20730, 5885, 3326],
    'Sales FY-2021 (CHF m)': [87088, 33779, 21128, 20735, 6418, 4822],
}

df = pd.DataFrame(data)

# Calculate the ratio of sales for each column
df['Sales Ratio (2021/2020)'] = df['Sales FY-2021 (CHF m)'] / df['Sales
FY-2020 (CHF m)']

# Print the DataFrame with the ratios
print(df)
```

```
                  Zone  Sales FY-2020 (CHF m)  Sales FY-2021 (CHF m)  \
0             Zone AMS                  84343                  87088
1           Zone EMENA                  34010                  33779
2             Zone AOA                  20226                  21128
3            Nespresso                  20730                  20735
4  Nestlé Health Science                   5885                   6418
5      Other Businesses                   3326                   4822

   Sales Ratio (2021/2020)
0                 1.032546
1                 0.993208
2                 1.044596
3                 1.000241
4                 1.090569
5                 1.449790
```

The code effectively calculates and adds the sales ratios to the DataFrame, allowing you to see how sales in FY-2021 compare to sales in FY-2020 for each zone/segment.

The sales ratios provide insights into the relative growth or decline in sales between the two fiscal years.

A sales ratio greater than 1 indicates an increase in sales from FY-2020 to FY-2021, while a ratio less than 1 indicates a decrease.

The DataFrame now allows you to easily compare and analyze the performance of different zones/segments in terms of sales growth or decline.

Sales ratios can be valuable for identifying which zones/segments experienced significant changes in sales and for making data-driven decisions based on these insights.

In summary, the code successfully calculates and adds the sales ratios to the DataFrame, enhancing its analytical capabilities for assessing sales performance across different zones/segments between FY-2020 and FY-2021.

MANAGERIAL INSIGHTS

- Managers can use this data to allocate resources strategically. Zones/segments with strong sales growth potential may justify increased investments, while those with declining sales may require interventions to reverse the trend.
- The analysis reflects the impact of market dynamics and customer behavior on sales. Understanding these dynamics can guide marketing and sales strategies.
- Zones/segments with high sales ratios may be performing well, but it's essential to manage risks associated with rapid growth, such as supply chain challenges or market saturation.
- The analysis emphasizes the importance of data-driven decision-making. Regularly monitoring sales data and ratios allows for proactive management and timely adjustments.
- Managers should use these insights for long-term planning and goal setting. Setting achievable sales targets based on historical trends and growth potential is vital.
- Collaboration and communication across different zones/segments can facilitate knowledge sharing and best practices, leading to overall improvement in sales performance.
- Zones/segments with a sales ratio greater than 1 have seen an increase in sales from FY-2020 to FY-2021.These areas represent growth opportunities and may deserve additional attention and resources to further capitalize on their positive momentum.
- We can use the data to assess the impact of various factors such as pricing changes, foreign exchange fluctuations, mergers and acquisitions, and organic growth on sales performance. Understanding these factors can help in making adjustments and improvements.
-