I can't create files directly for download, but I can provide you with the Markdown content. You can copy it into a text editor and save it as a `.md` file. Here's the complete content for your documentation:

```
---
marp: true
size: 4:3
style: |
  section {
    font-size: 9px; /* Base font size for the entire presentation */
  }
  h1 {font-size: 18px; /* Header 1 font size */
  line-height:0; /* Set line height to 1 for tighter spacing */ }
  h2 {
    font-size: 16px; /* Header 2 font size */
    line-height:0; /* Set line height to 1 for tighter spacing */ }
  h3 {font-size: 14px; /* Header 3 font size */ }
  p, li, code {
    font-size: 12px; /* Font size for paragraph, list items, and code */
  }

---
# Slide Automation Tool Documentation

## Overview
This documentation outlines the Slide Automation Tool, which utilizes
Python to automate the creation of PowerPoint presentations based on
cleaned data. The tool is structured into several sections, each focusing
on different aspects of the slide generation process.

## Table of Contents
- [Landscape Section](#landscape-section)
- [Pricing Section](#pricing-section)
- [Project Steps](#project-steps)

---

# Landscape Section

## Introduction
In the slide automation landscape, we utilize 10 basic slides to create 22
sections. Here's a breakdown of the sections:

- Market Trends by Manufacturer
- Market Trends by Brands
- Market Trends by Sectors
- Market Trends by Segments
- Market Concentration By Manufacturer
- Market Concentration By Brands
- Market Concentration By Sectors
- Market Concentration By Segments
- Market Growth By Sectors
- Market Growth By Segments
```

~~~
- Market Growth By Retailer For Region
- Value Vs Avg Price By Sectors
- Value Vs Avg Price By Segments
- Value Vs Avg Price By Retailer For Region
- Share and Growth By Manufacturer/Brands
- Share And Growth By Manufacturer By Sector
- Share And Growth By Brands By Sector
- Share And Growth By Manufacturer By Segment
- Share And Growth By Brands By Segment
- Category Trends
- Share Evolution By Brand
- Category Overview

---

### Project Steps

- Project Flow
![Project Flow](<../Slides_Documentation/duplication_Steps.PNG>)

1. [Import Libraries](#step-1-import-libraries)
2. [Clean DataFrames](#step-2-clean-dataframes)
3. [Create Slides](#step-3-create-slides)
4. [Duplicate Slides](#step-4-duplicate-slides)
5. [Save Presentation](#step-5-save-presentation)

---

# [Step 1: Import Libraries](https://github.com/khaledSeifEleslam/Slide-
Automate/blob/main/general_functions/generalFunctions.ipynb)

```python
# Import necessary libraries for PowerPoint automation and data
manipulation
from pptx import Presentation
import win32com.client as win32
import pandas as pd
import numpy as np
from pathlib import Path
import re
import sys
import time
import shutil
import os
import warnings

# Set default warnings to be ignored
warnings.filterwarnings("ignore")
~~~

Step 2: Clean DataFrames

```python
def clean_dataframes(df_dict):
    """
    Cleans a dictionary of DataFrames by filtering out unwanted rows and
handling NaN values.

    Parameters:
        df_dict (dict): A dictionary containing DataFrames to clean.

    Returns:
        dict: A dictionary containing cleaned DataFrames.
    """
    for key in df_dict.keys():
        df = df_dict[key].copy()  # Create a copy to avoid modifying the
original
        df = df[df['Top Brands'] != 'Others']  # Filter out 'Others' rows
        df = df.fillna(0)  # Replace NaN values with 0
        df_dict[key] = df  # Update the dictionary with the cleaned
DataFrame
    return df_dict
```

Step 3: Create Slides

```python
def create_price_positioning_slide(prs, modified_data, num_of_duplicates, position=0):
    """
    Generates slides for price positioning analysis with bubble chart visualizations.

    Parameters:
        prs (Presentation): PowerPoint presentation object.
        modified_data (dict): Dictionary containing sorted price positioning DataFrames.
        num_of_duplicates (int): Number of duplicate slides to generate.
        position (int): Position index to start adding slides. Default is 0.
    """
    for slide_num in range(num_of_duplicates):
        market = list(modified_data.keys())[slide_num]
        df = modified_data[market].reset_index(drop=True)  # Reset index for the DataFrame
        shapes = prs.slides[slide_num + position].shapes  # Access slide shapes
        charts = [shape for shape in shapes if shape.has_chart]  # Get all charts in slide

        # Update text boxes in the slide
        shapes[4].text = data_source  # Assume data_source is defined elsewhere
        shapes[5].text = f'Brand Price & Index vs Market | Bubble Size by Value Sales | {market} | P12M'
        shapes[5].text_frame.paragraphs[0].font.bold = True

        if charts:
            chart = charts[0].chart  # Assume there is at least one chart
            chart_data = BubbleChartData()
            chart_data.categories = df['Av Price/Unit'].unique().tolist()
            series = chart_data.add_series("Relative Price Index")
            series.has_data_labels = True

            # Add data points to the bubble chart
            for i in range(df.shape[0]):
                series.add_data_point(df['Av Price/Unit'].iloc[i], df['Relative Price'].iloc[i], df['Value Sales'].iloc[i])
            chart.replace_data(chart_data)  # Replace chart data
```

Step 4: Duplicate Slides

```python
def prepare_slide_configuration(modified_data):
    """
    Prepares index and duplication lists for generating PowerPoint slides.

    Parameters:
        modified_data (dict): Dictionary containing modified DataFrames for
slide generation.

    Returns:
        tuple: index list, duplication list, section names list
    """
    index = [0] * 8  # Adjust according to your specific needs
    duplication = [
        len(modified_data['price_positioning']),  # Example for price
positioning slides
        len(modified_data['brand_segments']),      # Example for segments
leadership slides
        # Add more as needed...
    ]

    # Define section names based on duplication
    section_names = [
        "Price Positioning Analysis",
        "Segments Leadership Analysis",
        # Add more as needed...
    ]

    return index, duplication, section_names
```

Step 5: Save Presentation

```python
def save_presentation(prs, filename):
    """
    Saves the PowerPoint presentation and opens it using the PowerPoint
application.

    Parameters:
        prs (Presentation): PowerPoint presentation object to save.
        filename (str): The filename to save the presentation as.
    """
    output_path = os.path.join(os.getcwd(), filename)
    prs.save(output_path)  # Save the presentation
    app = win32.Dispatch("PowerPoint.Application")  # Initialize PowerPoint
application
    app.Presentations.Open(output_path)  # Open the saved presentation
```

Pricing Section

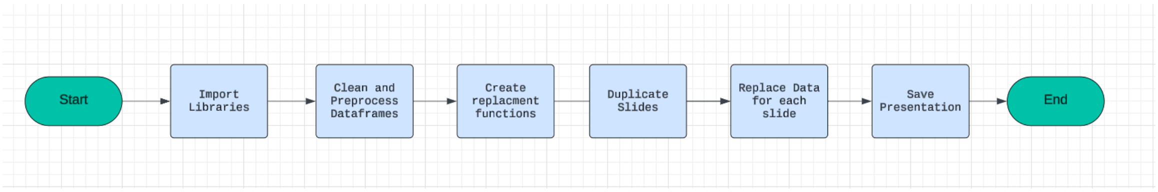
Introduction

In the slide automation landscape using 8 basic slides, we have created 16 sections:

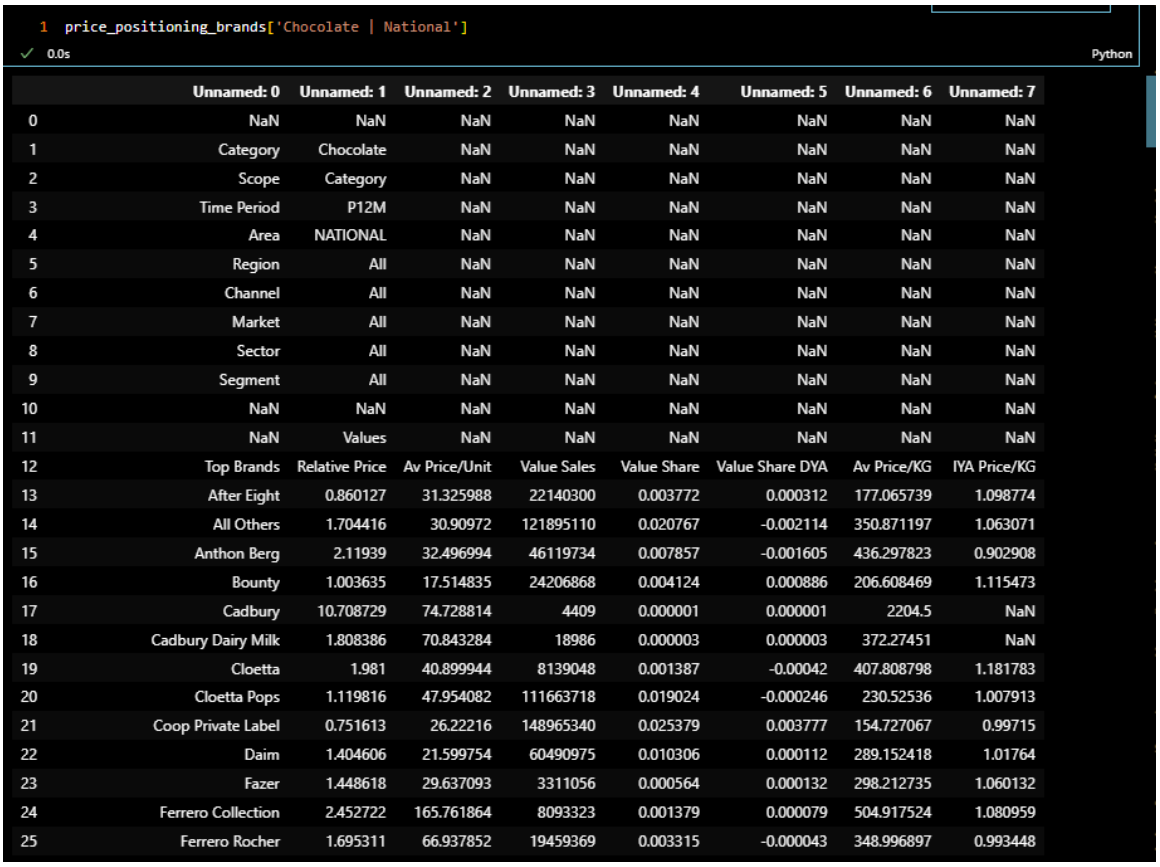
- Price Positioning Analysis
- Segments Leadership Analysis
- Sectors Leadership Analysis
- Sector Avg Price/Vol Comparison
- Sector Shelf Price/Vol Comparison
- Segment Avg Price/Vol Comparison
- Segment Shelf Price/Vol Comparison
- Category Price Point Distribution Analysis P3M
- Category Price Point Distribution Analysis P12M
- Sector Price Point Distribution Analysis P3M
- Sector Price Point Distribution Analysis P12M
- Segment Price Point Distribution Analysis P3M
- Segment Price Point Distribution Analysis P12M
- Price Point Distribution Analysis By Brand
- Price Point Distribution By Brand By Sector
- Price Point Distribution By Brand By Segment

Project Steps

- Project Flow


Start
Import Libraries
Clean and Preprocess Dataframes
Create replacment functions
Duplicate Slides
Replace Data for each slide
Save Presentation
End

1. Import Libraries2. Clean DataFrames3. Create Slides4. Duplicate Slides5. Save Presentation---# Step 1: Import Libraries(Include the same import code as above)---# Step 2: Clean DataFrames(Include the same cleaning code as above)---# Example: Input DataFrame Before Cleaning

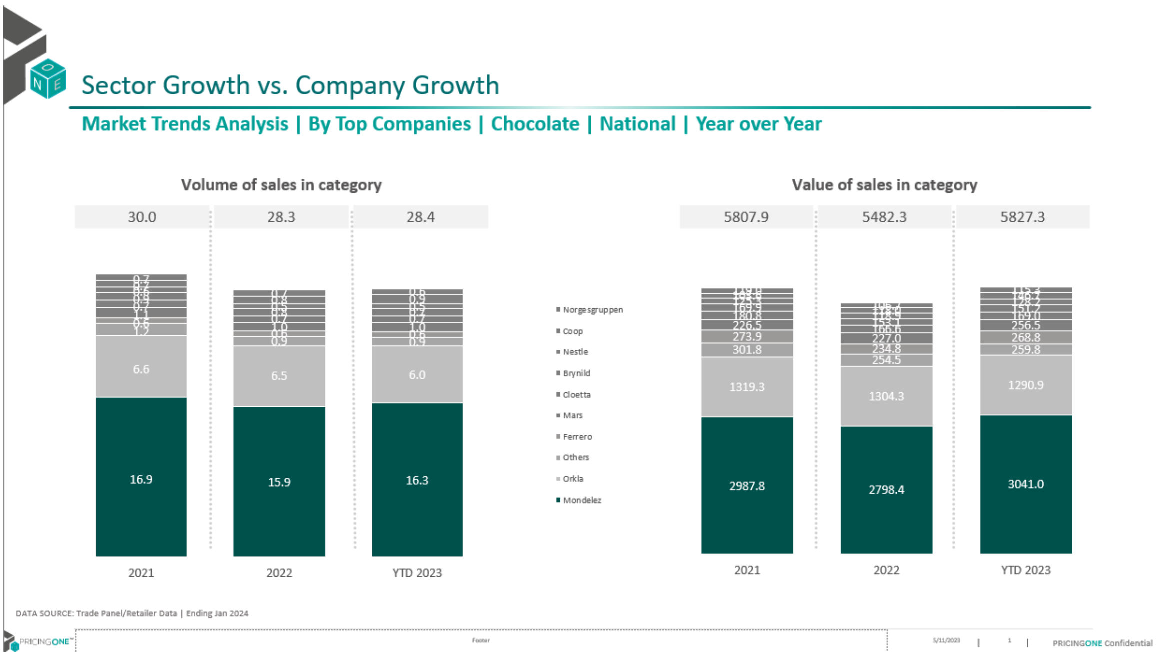


```python
price_positioning_brands['Chocolate | National']
```

```
✓ 0.0s                                                    Python
```

	Unnamed: 0	Unnamed: 1	Unnamed: 2	Unnamed: 3	Unnamed: 4	Unnamed: 5	Unnamed: 6	Unnamed: 7
0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
1	Category	Chocolate	NaN	NaN	NaN	NaN	NaN	NaN
2	Scope	Category	NaN	NaN	NaN	NaN	NaN	NaN
3	Time Period	P12M	NaN	NaN	NaN	NaN	NaN	NaN
4	Area	NATIONAL	NaN	NaN	NaN	NaN	NaN	NaN
5	Region	All	NaN	NaN	NaN	NaN	NaN	NaN
6	Channel	All	NaN	NaN	NaN	NaN	NaN	NaN
7	Market	All	NaN	NaN	NaN	NaN	NaN	NaN
8	Sector	All	NaN	NaN	NaN	NaN	NaN	NaN
9	Segment	All	NaN	NaN	NaN	NaN	NaN	NaN
10	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
11	NaN	Values	NaN	NaN	NaN	NaN	NaN	NaN
12	Top Brands	Relative Price	Av Price/Unit	Value Sales	Value Share	Value Share DYA	Av Price/KG	IYA Price/KG
13	After Eight	0.860127	31.325988	22140300	0.003772	0.000312	177.065739	1.098774
14	All Others	1.704416	30.90972	121895110	0.020767	-0.002114	350.871197	1.063071
15	Anthon Berg	2.11939	32.496994	46119734	0.007857	-0.001605	436.297823	0.902908
16	Bounty	1.003635	17.514835	24206868	0.004124	0.000886	206.608469	1.115473
17	Cadbury	10.708729	74.728814	4409	0.000001	0.000001	2204.5	NaN
18	Cadbury Dairy Milk	1.808386	70.843284	18986	0.000003	0.000003	372.27451	NaN
19	Cloetta	1.981	40.899944	8139048	0.001387	-0.00042	407.808798	1.181783
20	Cloetta Pops	1.119816	47.954082	111663718	0.019024	-0.000246	230.52536	1.007913
21	Coop Private Label	0.751613	26.22216	148965340	0.025379	0.003777	154.727067	0.99715
22	Daim	1.404606	21.599754	60490975	0.010306	0.000112	289.152418	1.01764
23	Fazer	1.448618	29.637093	3311056	0.000564	0.000132	298.212735	1.060132
24	Ferrero Collection	2.452722	165.761864	8093323	0.001379	0.000079	504.917524	1.080959
25	Ferrero Rocher	1.695311	66.937852	19459369	0.003315	-0.000043	348.996897	0.993448

---# Example: Market Trends Slide Output After Replacement Data



Sector Growth vs. Company Growth

Market Trends Analysis | By Top Companies | Chocolate | National | Year over Year

DATA SOURCE: Trade Panel/Retailer Data | Ending Jan 2024

---```### Instructions to Create the File1. **Copy the Markdown content** above.2. **Open a text editor** (like Notepad, VSCode, or any Markdown editor).3. **Paste the content** into the editor.4. **Save the file** with a `.md` extension (for example, `slides_documentation.md`).If you need any more assistance or modifications, just let me know!