

```
import pandas as pd #pandas (pd): Data manipulation ke liye use hota hai, jaise CSV file load karna aur process karna.

import plotly.express as px #plotly.express (px): Data visualization library jo easy aur quick plots banata hai.
import plotly.graph_objects as go #plotly.graph_objects (go): Advanced and customizable graphs banane ke liye
import plotly.io as pio #plotly.io (pio): Graph templates ko customize karne ke liye
import plotly.colors as colors
pio.templates.default = "plotly_white" #pio.templates.default = "plotly_white": Default theme white rakha gaya hai graphs ke liye

data = pd.read_csv("Sample - Superstore.csv", encoding='latin-1')
#encoding='latin-1': Special characters ko properly read karne ke liye encoding use hui hai
data.head()
```

	Row ID	Order ID	Order Date	Ship Date	Ship Mode
Customer ID \					
0	1	CA-2016-152156	11/8/2016	11/11/2016	Second Class
CG-12520					
1	2	CA-2016-152156	11/8/2016	11/11/2016	Second Class
CG-12520					
2	3	CA-2016-138688	6/12/2016	6/16/2016	Second Class
DV-13045					
3	4	US-2015-108966	10/11/2015	10/18/2015	Standard Class
S0-20335					
4	5	US-2015-108966	10/11/2015	10/18/2015	Standard Class
S0-20335					

	Customer Name	Segment	Country	City	...	\
0	Claire Gute	Consumer	United States	Henderson	...	
1	Claire Gute	Consumer	United States	Henderson	...	
2	Darrin Van Huff	Corporate	United States	Los Angeles	...	
3	Sean O'Donnell	Consumer	United States	Fort Lauderdale	...	
4	Sean O'Donnell	Consumer	United States	Fort Lauderdale	...	

	Postal Code	Region	Product ID	Category	Sub-
Category \					
0	42420	South	FUR-B0-10001798	Furniture	Bookcases
1	42420	South	FUR-CH-10000454	Furniture	Chairs
2	90036	West	OFF-LA-10000240	Office Supplies	Labels
3	33311	South	FUR-TA-10000577	Furniture	Tables
4	33311	South	OFF-ST-10000760	Office Supplies	Storage

Quantity \	Product Name	Sales
0	Bush Somerset Collection Bookcase	261.9600
2		
1	Hon Deluxe Fabric Upholstered Stacking Chairs,...	731.9400
3		
2	Self-Adhesive Address Labels for Typewriters b...	14.6200
2		
3	Bretford CR4500 Series Slim Rectangular Table	957.5775
5		
4	Eldon Fold 'N Roll Cart System	22.3680
2		

	Discount	Profit
0	0.00	41.9136
1	0.00	219.5820
2	0.00	6.8714
3	0.45	-383.0310
4	0.20	2.5164

[5 rows x 21 columns]

Let's start by looking at the descriptive statistics of the dataset

```
data.describe()
```

	Row ID	Postal Code	Sales	Quantity
Discount \				
count	9994.000000	9994.000000	9994.000000	9994.000000
mean	4997.500000	55190.379428	229.858001	3.789574
std	2885.163629	32063.693350	623.245101	2.225110
min	1.000000	1040.000000	0.444000	1.000000
25%	2499.250000	23223.000000	17.280000	2.000000
50%	4997.500000	56430.500000	54.490000	3.000000
75%	7495.750000	90008.000000	209.940000	5.000000
max	9994.000000	99301.000000	22638.480000	14.000000

Profit
0.800000

```
count    9994.000000
mean      28.656896
std       234.260108
min      -6599.978000
25%        1.728750
50%        8.666500
75%       29.364000
max       8399.976000
```

The dataset has an order date column. We can use this column to create new columns like order month, order year, and order day, which will be very valuable for sales and profit analysis according to time periods. So let's add these columns:

```
data.head()
```

Row ID	Order ID	Order Date	Ship Date	Ship Mode
Customer ID \				
0 1	CA-2016-152156	11/8/2016	11/11/2016	Second Class
CG-12520				
1 2	CA-2016-152156	11/8/2016	11/11/2016	Second Class
CG-12520				
2 3	CA-2016-138688	6/12/2016	6/16/2016	Second Class
DV-13045				
3 4	US-2015-108966	10/11/2015	10/18/2015	Standard Class
S0-20335				
4 5	US-2015-108966	10/11/2015	10/18/2015	Standard Class
S0-20335				

Customer Name	Segment	Country	City	...	\
0 Claire Gute	Consumer	United States	Henderson	...	
1 Claire Gute	Consumer	United States	Henderson	...	
2 Darrin Van Huff	Corporate	United States	Los Angeles	...	
3 Sean O'Donnell	Consumer	United States	Fort Lauderdale	...	
4 Sean O'Donnell	Consumer	United States	Fort Lauderdale	...	

Postal Code	Region	Product ID	Category	Sub-
0 42420	South	FUR-B0-10001798	Furniture	Bookcases
1 42420	South	FUR-CH-10000454	Furniture	Chairs
2 90036	West	OFF-LA-10000240	Office Supplies	Labels
3 33311	South	FUR-TA-10000577	Furniture	Tables
4 33311	South	OFF-ST-10000760	Office Supplies	Storage

Product Name	Sales
Quantity \	

```

0          Bush Somerset Collection Bookcase  261.9600
2
1  Hon Deluxe Fabric Upholstered Stacking Chairs,...  731.9400
3
2  Self-Adhesive Address Labels for Typewriters b...  14.6200
2
3          Bretford CR4500 Series Slim Rectangular Table  957.5775
5
4          Eldon Fold 'N Roll Cart System  22.3680
2

```

```

      Discount    Profit
0         0.00  41.9136
1         0.00 219.5820
2         0.00   6.8714
3         0.45 -383.0310
4         0.20   2.5164

```

```
[5 rows x 21 columns]
```

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
```

```
RangeIndex: 9994 entries, 0 to 9993
```

```
Data columns (total 21 columns):
```

#	Column	Non-Null Count	Dtype
0	Row ID	9994 non-null	int64
1	Order ID	9994 non-null	object
2	Order Date	9994 non-null	object
3	Ship Date	9994 non-null	object
4	Ship Mode	9994 non-null	object
5	Customer ID	9994 non-null	object
6	Customer Name	9994 non-null	object
7	Segment	9994 non-null	object
8	Country	9994 non-null	object
9	City	9994 non-null	object
10	State	9994 non-null	object
11	Postal Code	9994 non-null	int64
12	Region	9994 non-null	object
13	Product ID	9994 non-null	object
14	Category	9994 non-null	object
15	Sub-Category	9994 non-null	object
16	Product Name	9994 non-null	object
17	Sales	9994 non-null	float64
18	Quantity	9994 non-null	int64
19	Discount	9994 non-null	float64
20	Profit	9994 non-null	float64

```
dtypes: float64(3), int64(3), object(15)
```

```
memory usage: 1.6+ MB
```

# Converting Date Columns

```
data['Order Date'] = pd.to_datetime(data['Order Date'])
data['Ship Date'] = pd.to_datetime(data['Ship Date'])
#Date Conversion: Order Date aur Ship Date columns ko datetime format
me convert kiya gaya hai for date-based analysis.
```

## Adding New Date-Based Columns

```
data['Order Month'] = data['Order Date'].dt.month
data['Order Year'] = data['Order Date'].dt.year
data['Order Day of Week'] = data['Order Date'].dt.dayofweek

#Order Month: Order date se month extract karte hain.
#Order Year: Order date se year extract hota hai.
#Order Day of Week: Week ka day (0 for Monday, 6 for Sunday) extract
kiya gaya hai.
```

```
data.head()
```

Row ID	Order ID	Order Date	Ship Date	Ship Mode
Customer ID \				
0 1	CA-2016-152156	2016-11-08	2016-11-11	Second Class CG-
12520				
1 2	CA-2016-152156	2016-11-08	2016-11-11	Second Class CG-
12520				
2 3	CA-2016-138688	2016-06-12	2016-06-16	Second Class DV-
13045				
3 4	US-2015-108966	2015-10-11	2015-10-18	Standard Class SO-
20335				
4 5	US-2015-108966	2015-10-11	2015-10-18	Standard Class SO-
20335				

	Customer Name	Segment	Country	City	...	\
0	Claire Gute	Consumer	United States	Henderson	...	
1	Claire Gute	Consumer	United States	Henderson	...	
2	Darrin Van Huff	Corporate	United States	Los Angeles	...	
3	Sean O'Donnell	Consumer	United States	Fort Lauderdale	...	
4	Sean O'Donnell	Consumer	United States	Fort Lauderdale	...	

	Category	Sub-Category	\
0	Furniture	Bookcases	
1	Furniture	Chairs	
2	Office Supplies	Labels	
3	Furniture	Tables	
4	Office Supplies	Storage	

Product Name	Sales
--------------	-------

Quantity \		
0	Bush Somerset Collection Bookcase	261.9600
2		
1	Hon Deluxe Fabric Upholstered Stacking Chairs,...	731.9400
3		
2	Self-Adhesive Address Labels for Typewriters b...	14.6200
2		
3	Bretford CR4500 Series Slim Rectangular Table	957.5775
5		
4	Eldon Fold 'N Roll Cart System	22.3680
2		

	Discount	Profit	Order Month	Order Year	Order Day of Week
0	0.00	41.9136	11	2016	1
1	0.00	219.5820	11	2016	1
2	0.00	6.8714	6	2016	6
3	0.45	-383.0310	10	2015	6
4	0.20	2.5164	10	2015	6

[5 rows x 24 columns]

## Monthly Sales Analysis

```

sales_by_month = data.groupby('Order Month')
['Sales'].sum().reset_index()
fig = px.line(sales_by_month,
              x='Order Month',
              y='Sales',
              title='Monthly Sales Analysis')
fig.show()

{"config":{"plotlyServerURL":"https://plot.ly"},"data":
[{"hovertemplate":"Order
Month=%{x}<br>Sales=%{y}<extra></extra>","legendgroup":"","line":
{"color":"#636efa","dash":"solid"},"marker":
{"symbol":"circle"},"mode":"lines","name":"","orientation":"v","showle
gend":false,"type":"scatter","x":
[1,2,3,4,5,6,7,8,9,10,11,12],"xaxis":"x","y":
[94924.8356,59751.2514,205005.4888,137762.1286,155028.8117,152718.6793
,147238.097,159044.063,307649.9457,200322.9847,352461.071,325293.5035]
,"yaxis":"y"}],"layout":{"autosize":true,"legend":
{"tracegroupgap":0},"template":{"data":{"bar":[{"error_x":
{"color":"#2a3f5f"},"error_y":{"color":"#2a3f5f"},"marker":{"line":
{"color":"white","width":0.5},"pattern":
{"fillmode":"overlay","size":10,"solidity":0.2},"type":"bar"}],"barpo
lar":[{"marker":{"line":{"color":"white","width":0.5},"pattern":
{"fillmode":"overlay","size":10,"solidity":0.2},"type":"barpolar"}],"
carpet":[{"aaxis":

```

```

"endlinecolor": "#2a3f5f", "gridcolor": "#C8D4E3", "linecolor": "#C8D4E3",
"minorgridcolor": "#C8D4E3", "startlinecolor": "#2a3f5f", "baxis":
{"endlinecolor": "#2a3f5f", "gridcolor": "#C8D4E3", "linecolor": "#C8D4E3",
"minorgridcolor": "#C8D4E3", "startlinecolor": "#2a3f5f", "type": "carpet"
}], "choropleth": [{"colorbar":
{"outlinewidth": 0, "ticks": ""}, "type": "choropleth"}], "contour":
[{"colorbar": {"outlinewidth": 0, "ticks": ""}, "colorscale":
[[0, "#0d0887"], [0.1111111111111111, "#46039f"],
[0.2222222222222222, "#7201a8"], [0.3333333333333333, "#9c179e"],
[0.4444444444444444, "#bd3786"], [0.5555555555555556, "#d8576b"],
[0.6666666666666666, "#ed7953"], [0.7777777777777778, "#fb9f3a"],
[0.8888888888888888, "#fdca26"],
[1, "#f0f921"]], "type": "contour"}], "contourcarpet": [{"colorbar":
{"outlinewidth": 0, "ticks": ""}, "type": "contourcarpet"}], "heatmap":
[{"colorbar": {"outlinewidth": 0, "ticks": ""}, "colorscale":
[[0, "#0d0887"], [0.1111111111111111, "#46039f"],
[0.2222222222222222, "#7201a8"], [0.3333333333333333, "#9c179e"],
[0.4444444444444444, "#bd3786"], [0.5555555555555556, "#d8576b"],
[0.6666666666666666, "#ed7953"], [0.7777777777777778, "#fb9f3a"],
[0.8888888888888888, "#fdca26"],
[1, "#f0f921"]], "type": "heatmap"}], "heatmapgl": [{"colorbar":
{"outlinewidth": 0, "ticks": ""}, "colorscale": [[0, "#0d0887"],
[0.1111111111111111, "#46039f"], [0.2222222222222222, "#7201a8"],
[0.3333333333333333, "#9c179e"], [0.4444444444444444, "#bd3786"],
[0.5555555555555556, "#d8576b"], [0.6666666666666666, "#ed7953"],
[0.7777777777777778, "#fb9f3a"], [0.8888888888888888, "#fdca26"],
[1, "#f0f921"]], "type": "heatmapgl"}], "histogram": [{"marker": {"pattern":
{"fillmode": "overlay", "size": 10, "solidity": 0.2}}, "type": "histogram"}],
"histogram2d": [{"colorbar": {"outlinewidth": 0, "ticks": ""}, "colorscale":
[[0, "#0d0887"], [0.1111111111111111, "#46039f"],
[0.2222222222222222, "#7201a8"], [0.3333333333333333, "#9c179e"],
[0.4444444444444444, "#bd3786"], [0.5555555555555556, "#d8576b"],
[0.6666666666666666, "#ed7953"], [0.7777777777777778, "#fb9f3a"],
[0.8888888888888888, "#fdca26"],
[1, "#f0f921"]], "type": "histogram2d"}], "histogram2dcontour":
[{"colorbar": {"outlinewidth": 0, "ticks": ""}, "colorscale":
[[0, "#0d0887"], [0.1111111111111111, "#46039f"],
[0.2222222222222222, "#7201a8"], [0.3333333333333333, "#9c179e"],
[0.4444444444444444, "#bd3786"], [0.5555555555555556, "#d8576b"],
[0.6666666666666666, "#ed7953"], [0.7777777777777778, "#fb9f3a"],
[0.8888888888888888, "#fdca26"],
[1, "#f0f921"]], "type": "histogram2dcontour"}], "mesh3d": [{"colorbar":
{"outlinewidth": 0, "ticks": ""}, "type": "mesh3d"}], "parcoords": [{"line":
{"colorbar": {"outlinewidth": 0, "ticks": ""}, "type": "parcoords"}], "pie":
[{"automargin": true, "type": "pie"}], "scatter": [{"fillpattern":
{"fillmode": "overlay", "size": 10, "solidity": 0.2}, "type": "scatter"}], "scatter3d": [{"line": {"colorbar": {"outlinewidth": 0, "ticks": ""}, "marker":
{"colorbar":
{"outlinewidth": 0, "ticks": ""}, "type": "scatter3d"}], "scattercarpet":

```

```
[{"marker":{"colorbar":
{"linewidth":0,"ticks":"","type":"scattercarpet"}}, "scattergeo":
[{"marker":{"colorbar":
{"linewidth":0,"ticks":"","type":"scattergeo"}}, "scattergl":
[{"marker":{"colorbar":
{"linewidth":0,"ticks":"","type":"scattergl"}}, "scattermapbox":
[{"marker":{"colorbar":
{"linewidth":0,"ticks":"","type":"scattermapbox"}}, "scatterpolar":
[{"marker":{"colorbar":
{"linewidth":0,"ticks":"","type":"scatterpolar"}}, "scatterpolargl":
[{"marker":{"colorbar":
{"linewidth":0,"ticks":"","type":"scatterpolargl"}}, "scatterternary":
[{"marker":{"colorbar":
{"linewidth":0,"ticks":"","type":"scatterternary"}}, "surface":
[{"colorbar":{"linewidth":0,"ticks":"","colorscale":
[[0,"#0d0887"],[0.1111111111111111,"#46039f"],
[0.2222222222222222,"#7201a8"],[0.3333333333333333,"#9c179e"],
[0.4444444444444444,"#bd3786"],[0.5555555555555556,"#d8576b"],
[0.6666666666666666,"#ed7953"],[0.7777777777777778,"#fb9f3a"],
[0.8888888888888888,"#fdca26"],
[1,"#f0f921"]], "type":"surface"}}, "table":{"cells":{"fill":
{"color":"#EBF0F8"}, "line":{"color":"white"}}, "header":{"fill":
{"color":"#C8D4E3"}, "line":
{"color":"white"}}, "type":"table"}}, "layout":{"annotationdefaults":
{"arrowcolor":"#2a3f5f", "arrowhead":0, "arrowwidth":1}, "autotypenumbers":
"strict", "coloraxis":{"colorbar":
{"linewidth":0,"ticks":"","colorscale":{"diverging":
[[0,"#8e0152"],[0.1,"#c51b7d"],[0.2,"#de77ae"],[0.3,"#f1b6da"],
[0.4,"#fde0ef"],[0.5,"#f7f7f7"],[0.6,"#e6f5d0"],[0.7,"#b8e186"],
[0.8,"#7fbcb4"],[0.9,"#4d9221"],[1,"#276419"]], "sequential":
[[0,"#0d0887"],[0.1111111111111111,"#46039f"],
[0.2222222222222222,"#7201a8"],[0.3333333333333333,"#9c179e"],
[0.4444444444444444,"#bd3786"],[0.5555555555555556,"#d8576b"],
[0.6666666666666666,"#ed7953"],[0.7777777777777778,"#fb9f3a"],
[0.8888888888888888,"#fdca26"],[1,"#f0f921"]], "sequentialminus":
[[0,"#0d0887"],[0.1111111111111111,"#46039f"],
[0.2222222222222222,"#7201a8"],[0.3333333333333333,"#9c179e"],
[0.4444444444444444,"#bd3786"],[0.5555555555555556,"#d8576b"],
[0.6666666666666666,"#ed7953"],[0.7777777777777778,"#fb9f3a"],
[0.8888888888888888,"#fdca26"],[1,"#f0f921"]]}}, "colorway":
["#636efa", "#EF553B", "#00cc96", "#ab63fa", "#FFA15A", "#19d3f3", "#FF6692",
"#B6E880", "#FF97FF", "#FECB52"], "font":{"color":"#2a3f5f"}, "geo":
{"bgcolor":"white", "lakecolor":"white", "landcolor":"white", "showlakes":
true, "showland":true, "subunitcolor":"#C8D4E3"}, "hoverlabel":
{"align":"left"}, "hovermode":"closest", "mapbox":
{"style":"light"}, "paper_bgcolor":"white", "plot_bgcolor":"white", "polar":
{"angularaxis":
{"gridcolor":"#EBF0F8", "linecolor":"#EBF0F8", "ticks":"","bgcolor":"white", "radialaxis":
```



```
{
  "gridcolor": "#EBF0F8", "linecolor": "#EBF0F8", "ticks": ""}, "scene": {
  "xaxis": {
    "backgroundcolor": "white", "gridcolor": "#DFE8F3", "gridwidth": 2, "linecolor": "#EBF0F8", "showbackground": true, "ticks": "", "zerolinecolor": "#EBF0F8"},
    "yaxis": {
      "backgroundcolor": "white", "gridcolor": "#DFE8F3", "gridwidth": 2, "linecolor": "#EBF0F8", "showbackground": true, "ticks": "", "zerolinecolor": "#EBF0F8"},
      "zaxis": {
        "backgroundcolor": "white", "gridcolor": "#DFE8F3", "gridwidth": 2, "linecolor": "#EBF0F8", "showbackground": true, "ticks": "", "zerolinecolor": "#EBF0F8"}},
    "shapedefaults": {"line": {"color": "#2a3f5f"}}, "ternary": {"aaxis": {"gridcolor": "#DFE8F3", "linecolor": "#A2B1C6", "ticks": ""}, "baxis": {"gridcolor": "#DFE8F3", "linecolor": "#A2B1C6", "ticks": ""}, "bgcolor": "white", "caxis": {"gridcolor": "#DFE8F3", "linecolor": "#A2B1C6", "ticks": ""}}, "title": {"x": 5.0e-2, "xaxis": {"automargin": true, "gridcolor": "#EBF0F8", "linecolor": "#EBF0F8", "ticks": "", "title": {"standoff": 15}, "zerolinecolor": "#EBF0F8", "zerolinewidth": 2}, "yaxis": {"automargin": true, "gridcolor": "#EBF0F8", "linecolor": "#EBF0F8", "ticks": "", "title": {"standoff": 15}, "zerolinecolor": "#EBF0F8", "zerolinewidth": 2}}}, "title": {"text": "Monthly Sales Analysis", "xaxis": {"anchor": "y", "autorange": true, "domain": [0, 1], "range": [1, 12], "title": {"text": "Order Month", "type": "linear"}, "yaxis": {"anchor": "x", "autorange": true, "domain": [0, 1], "range": [43489.59475555556, 368722.72764444444], "title": {"text": "Sales", "type": "linear"}}}}
```

*#Data Grouping:*

*#data.groupby('Order Month')['Sales'].sum() se har month ki total sales nikalte hain.*

*#.reset\_index() data ko structured format me rakhta hai.*

*#px.line: Monthly sales trend show karne ke liye line chart banaya gaya hai.*

*#fig.show(): Graph display karta hai.*

## Sales Analysis by Category

```
sales_by_category = data.groupby('Category')
['Sales'].sum().reset_index()
```

```
fig = px.pie(sales_by_category,
             values='Sales',
             names='Category',
             hole=0.5,
             color_discrete_sequence=px.colors.qualitative.Pastel)
```

```
fig.update_traces(textposition='inside', textinfo='percent+label')
fig.update_layout(title_text='Sales Analysis by Category',
title_font=dict(size=24))

fig.show()
```

Sales Analysis by Category

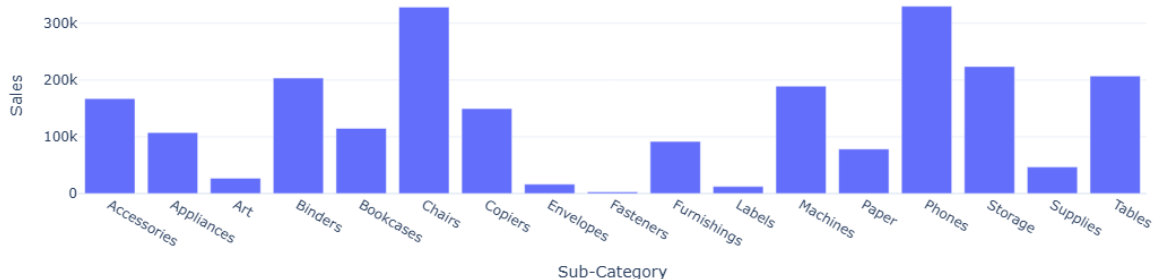


*#groupby('Category'): Category-wise sales nikalte hain.*  
*#Pie Chart:*  
*#px.pie: Sales proportions ko pie chart me show karta hai.*  
*#hole=0.5: Donut-style chart banata hai.*  
*#Pastel Colors: Chart me soft color palette use kiya gaya hai.*

## Sales Analysis by Sub-Category

```
sales_by_subcategory = data.groupby('Sub-Category')
['Sales'].sum().reset_index()
fig = px.bar(sales_by_subcategory,
             x='Sub-Category',
             y='Sales',
             title='Sales Analysis by Sub-Category')
fig.show()
```

Sales Analysis by Sub-Category



# Monthly Profit Analysis

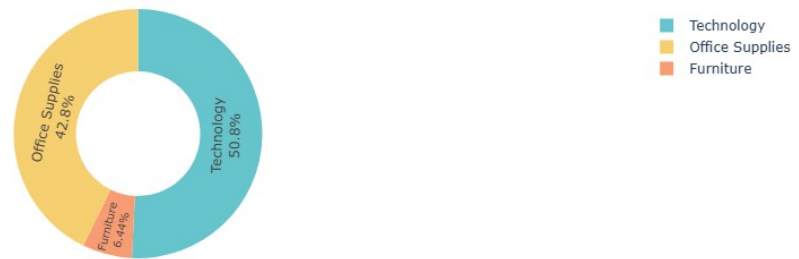
```
profit_by_month = data.groupby('Order Month')  
['Profit'].sum().reset_index()  
fig = px.line(profit_by_month,  
              x='Order Month',  
              y='Profit',  
              title='Monthly Profit Analysis')  
fig.show()
```



# Profit Analysis by Category

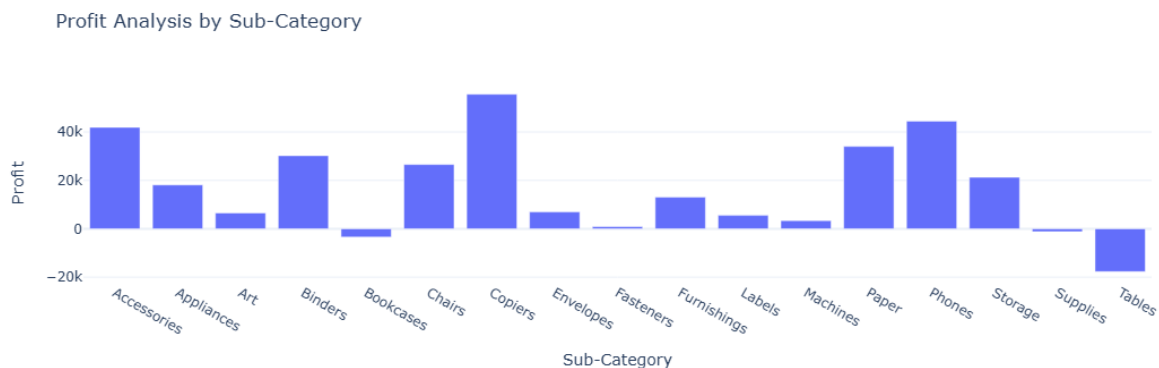
```
profit_by_category = data.groupby('Category')  
['Profit'].sum().reset_index()  
  
fig = px.pie(profit_by_category,  
             values='Profit',  
             names='Category',  
             hole=0.5,  
             color_discrete_sequence=px.colors.qualitative.Pastel)  
  
fig.update_traces(textposition='inside', textinfo='percent+label')  
fig.update_layout(title_text='Profit Analysis by Category',  
                  title_font=dict(size=24))  
fig.show()
```

## Profit Analysis by Category



## Profit Analysis by Sub-Category

```
profit_by_subcategory = data.groupby('Sub-Category')  
['Profit'].sum().reset_index()  
fig = px.bar(profit_by_subcategory, x='Sub-Category',  
             y='Profit',  
             title='Profit Analysis by Sub-Category')  
fig.show()
```



## Sales and Profit Analysis by Customer Segment

```
sales_profit_by_segment = data.groupby('Segment').agg({'Sales': 'sum',  
                                                       'Profit': 'sum'}).reset_index()  
  
color_palette = colors.qualitative.Pastel  
  
fig = go.Figure()  
fig.add_trace(go.Bar(x=sales_profit_by_segment['Segment'],  
                    y=sales_profit_by_segment['Sales'],  
                    name='Sales',  
                    marker_color=color_palette[0])))
```

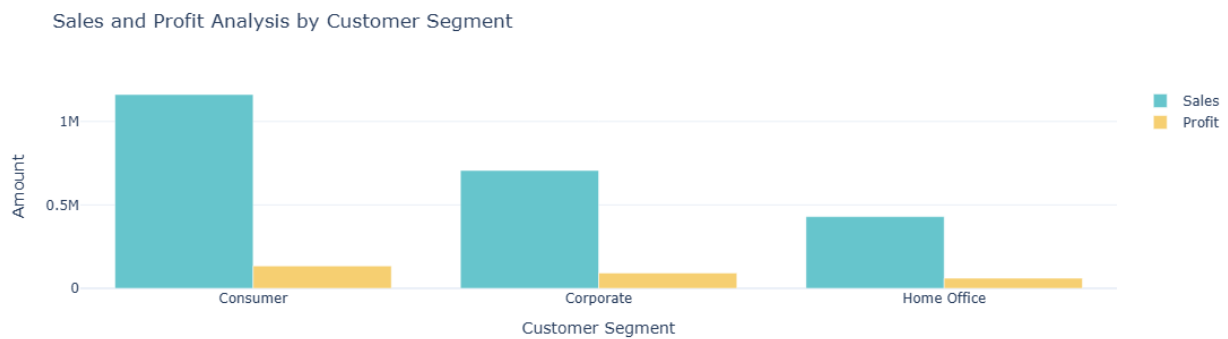
```

fig.add_trace(go.Bar(x=sales_profit_by_segment['Segment'],
                    y=sales_profit_by_segment['Profit'],
                    name='Profit',
                    marker_color=color_palette[1]))

fig.update_layout(title='Sales and Profit Analysis by Customer Segment',
                  xaxis_title='Customer Segment',
                  yaxis_title='Amount')

fig.show()

```



## analyse sales-to-profit ratio

```

sales_profit_by_segment = data.groupby('Segment').agg({'Sales': 'sum',
'Profit': 'sum'}).reset_index()
sales_profit_by_segment['Sales_to_Profit_Ratio'] =
sales_profit_by_segment['Sales'] / sales_profit_by_segment['Profit']
print(sales_profit_by_segment[['Segment', 'Sales_to_Profit_Ratio']])

```

	Segment	Sales_to_Profit_Ratio
0	Consumer	8.659471
1	Corporate	7.677245
2	Home Office	7.125416