```
import pandas as pd
import plotly.express as px
import plotly.graph_objects as go
import numpy as np
import plotly.io as pio
import plotly.offline as iplot
pio.templates.default = "plotly_white"
pd.set_option("display.float_format", sep=lambda*x*:'%.2f')
print(df.head)

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

[5 rows x 21 columns]

   Customer ID  Customer Name    Segment  ...       Country             City  \
0     CG-12520    Claire Gute   Consumer  ...  United States        Henderson
1     CG-12520    Claire Gute   Consumer  ...  United States        Henderson
2     DV-13045  Darrin Van Huff  Corporate  ...  United States      Los Angeles
3     SO-20335   Sean O'Donnell   Consumer  ...  United States  Fort Lauderdale
4     SO-20335   Sean O'Donnell   Consumer  ...  United States  Fort Lauderdale

       State  Postal Code  Region  Product ID  Category  Sub-Category  \
0   Kentucky        42420   South  FUR-BO-  Furniture     Bookcases
1   Kentucky        42420   South  FUR-CH-  Furniture        Chairs
2 California        90036    West  OFF-LA-  Office Supplies    Labels
3    Florida        33311   South  FUR-TA-  Furniture        Tables
4    Florida        33311   South  OFF-ST-  Office Supplies   Storage

                                        Product Name  Sales  Quantity  Discount  \
0  Bush Somerset Collection Bookcase  261.9600         2      0.00
1  Hon Deluxe Fabric Upholstered Stacking Chairs,...  731.9400         3      0.00
2  Self-Adhesive Address Labels for Typewriters b...   14.6200         2      0.00
3  Bretford CM4680 Series Slim Rectangular Table  957.5775         5      0.45
4  Eldon Fold N Roll Cart System  22.3680         2      0.20

   Profit
0  41.9136
1 219.5820
2   6.8714
3 -383.031
4   2.5104

[5 rows x 21 columns]
```

```
data.head()
```

```
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(5), int64(5), object(11)
memory usage: 1.6 MB
```

```
data['Order Date'] = pd.to_datetime(data['Order Date'])
data['Ship Date'] = pd.to_datetime(data['Ship Date'])
print(data['Order Date'])

0   2016-11-08
1   2016-11-08
2   2016-06-12
3   2015-10-11
4   2015-10-11
Name: Order Date, Length: 9994, dtype: datetime64[ns]
```

```
print(data['Ship Date'])

0   2016-11-11
1   2016-11-11
2   2016-06-16
3   2015-10-18
4   2015-10-18
Name: Ship Date, Length: 9994, dtype: datetime64[ns]
```

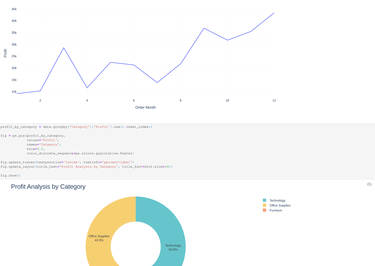
```
data['Order Month'] = data['Order Date'].dt.month
data['Order Year'] = data['Order Date'].dt.year
data['Order Day Of Week'] = data['Order Date'].dt.dayofweek
print(data['Order Month'])

0    11
1    11
2     6
3    10
4    10
Name: Order Month, Length: 9994, dtype: int32
```

```
print(data['Order Year'])

0    2016
1    2016
2    2016
3    2015
4    2015
Name: Order Year, Length: 9994, dtype: int32
```

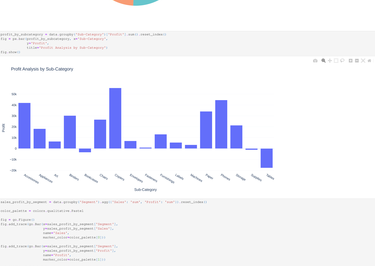
```
print(data['Order Day Of Week'])

0    1
1    1
2    4
3    4
4    4
Name: Order Day Of Week, Length: 9994, dtype: int32
```

```
data.head()

   Row ID  Order ID  Order Date  Ship Date  Ship Mode  Customer ID  Customer Name  Segment  Country  City  ...  Category  Sub-Category  Product Name  Sales  Quantity  Discount  Profit  Order Month  Order Year  Order Day Of Week
0       1  CA-2016-152156  2016-11-08  2016-11-11  Second Class  CG-12520  Claire Gute  Consumer  United States  Henderson  ...  Furniture  Bookcases  Bush Somerset Collection Bookcase  261.9600  2  0.00  41.9136  11  2016  1
1       2  CA-2016-152156  2016-11-08  2016-11-11  Second Class  CG-12520  Claire Gute  Consumer  United States  Henderson  ...  Furniture  Chairs  Hon Deluxe Fabric Upholstered Stacking Chairs,...  731.9400  3  0.00  219.5820  11  2016  1
2       3  CA-2016-138688  2016-06-12  2016-06-16  Second Class  DV-13045  Darrin Van Huff  Corporate  United States  Los Angeles  ...  Office Supplies  Labels  Self-Adhesive Address Labels for Typewriters b...  14.6200  2  0.00  6.8714  6  2016  4
3       4  US-2015-108966  2015-10-11  2015-10-18  Standard Class  SO-20335  Sean O'Donnell  Consumer  United States  Fort Lauderdale  ...  Furniture  Tables  Bretford CM4680 Series Slim Rectangular Table  957.5775  5  0.45  -383.031  10  2015  4
4       5  US-2015-108966  2015-10-11  2015-10-18  Standard Class  SO-20335  Sean O'Donnell  Consumer  United States  Fort Lauderdale  ...  Office Supplies  Storage  Eldon Fold N Roll Cart System  22.3680  2  0.20  2.5104  10  2015  4

5 rows × 27 columns
```

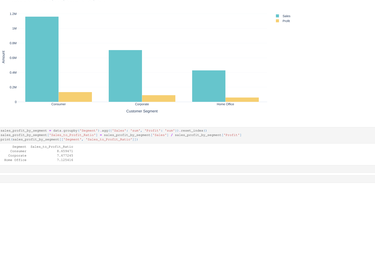
```
sales_by_month = data.groupby("Order Month")["Sales"].sum().reset_index()
fig = px.line(sales_by_month,
              x="Order Month",
              y="Sales",
              title="Monthly Sales Analysis")
fig.show()
```




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

fig = px.pie(sales_by_category,
             values="Sales",
             names="Category",
             hole=0.4,
             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=20))

fig.show()
```



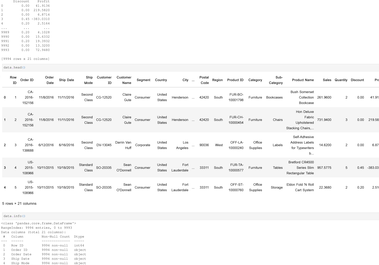

```
sales_by_subcategory = data.groupby("Sub-Category")["Sales"].sum().reset_index()
fig = px.bar(sales_by_subcategory,
             x="Sub-Category",
             y="Sales",
             title="Sales Analysis by Sub-Category")
fig.show()
```



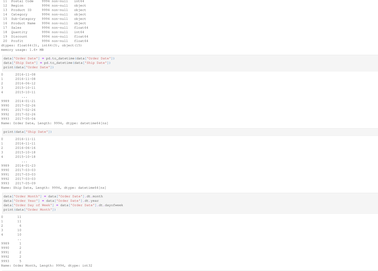

```
profit_by_month = data.groupby("Order Month")["Profit"].sum().reset_index()
fig = px.line(profit_by_month,
              x="Order Month",
              y="Profit",
              title="Monthly Profit Analysis")
fig.show()
```



```
profit_by_category = data.groupby("Category")["Profit"].sum().reset_index()

fig = px.pie(profit_by_category,
             values="Profit",
             names="Category",
             hole=0.4,
             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=20))

fig.show()
```




```
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_profit_by_segment = data.groupby("Segment").agg({"Sales": "sum", "Profit": "sum"}).reset_index()

fig.update_layout(title_font=dict(size=20))

fig.add_trace(go.Bar(sales_profit_by_segment["Segment"],
                     sales_profit_by_segment["Sales"],
                     name="Sales",
                     marker_color=px.colors.qualitative.Pastel))

fig.add_trace(go.Bar(sales_profit_by_segment["Segment"],
                     sales_profit_by_segment["Profit"],
                     name="Profit",
                     marker_color=px.colors.qualitative.Pastel))

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

fig.show()
```



```
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              8.609672
1Corporate              8.577143
2Home Office            7.120419
```