Let's Do It

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np

%matplotlib inline
```

Load the dataset

```
df = pd.read_csv('/content/sample_data/clean_data.csv')
df.head()
```

	order_id	order_status	customer	order_date	order_quantity	sales	discount	discount_value	product
0	3	Order Finished	Muhammed Mac Intyre	2010-10-13	6	523080	0.04	20923	Off
1	293	Order Finished	Barry French	2012-10-01	49	20246040	0.07	1417223	Off
2	483	Order Finished	Clay Rozendal	2011-07-10	30	9931519	0.08	794522	
3	515	Order Finished	Carlos Soltero	2010-08-28	19	788540	0.08	63083	Off

Check Missing and Duplicated Value

```
df.isnull().sum()
```

```
order_id               0
order_status           0
customer               0
order_date             0
order_quantity         0
sales                  0
discount               0
discount_value         0
product_category       0
product_sub_category   0
dtype: int64
```

```
df.duplicated().sum()
```

```
0
```

First Insight

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5499 entries, 0 to 5498
Data columns (total 10 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   order_id              5499 non-null   int64
 1   order_status          5499 non-null   object
 2   customer              5499 non-null   object
 3   order_date            5499 non-null   object
 4   order_quantity        5499 non-null   int64
 5   sales                 5499 non-null   int64
```

```
 6   discount              5499 non-null   float64
 7   discount_value        5499 non-null   int64
 8   product_category      5499 non-null   object
 9   product_sub_category  5499 non-null   object
dtypes: float64(1), int64(4), object(5)
memory usage: 429.7+ KB
```

```
df['order_date'] = pd.to_datetime(df['order_date'])
print(df.dtypes)
```

```
order_id                         int64
order_status                    object
customer                        object
order_date              datetime64[ns]
order_quantity                   int64
sales                            int64
discount                       float64
discount_value                   int64
product_category                object
product_sub_category            object
dtype: object
```

```
df.describe()
```

	order_id	order_quantity	sales	discount	discount_value
count	5499.000000	5499.000000	5.499000e+03	5499.000000	5.499000e+03
mean	29970.202219	25.521549	3.532838e+06	0.049915	1.735048e+05
std	17243.318085	14.485352	7.305121e+06	0.031783	4.183615e+05
min	3.000000	1.000000	6.460000e+03	0.000000	0.000000e+00
25%	15044.500000	13.000000	2.826700e+05	0.020000	7.739000e+03
50%	29927.000000	26.000000	8.546400e+05	0.050000	3.191700e+04
75%	44646.500000	38.000000	3.298741e+06	0.080000	1.329000e+05
max	59973.000000	50.000000	1.781221e+08	0.170000	7.441778e+06

Numerical Analysis and Visualization

```
df['order_quantity'].describe()
```

```
count    5499.000000
mean       25.521549
std        14.485352
min         1.000000
25%        13.000000
50%        26.000000
75%        38.000000
max        50.000000
Name: order_quantity, dtype: float64
```

```
df['order_quantity'].mean()
```

```
25.5215493726132
```

```
df['order_quantity'].median()
```

```
26.0
```

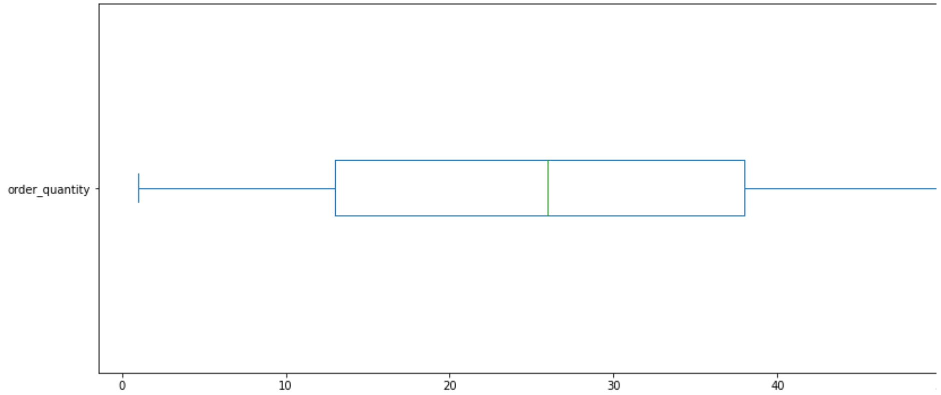
```
df['order_quantity'].min()
```

```
1
```

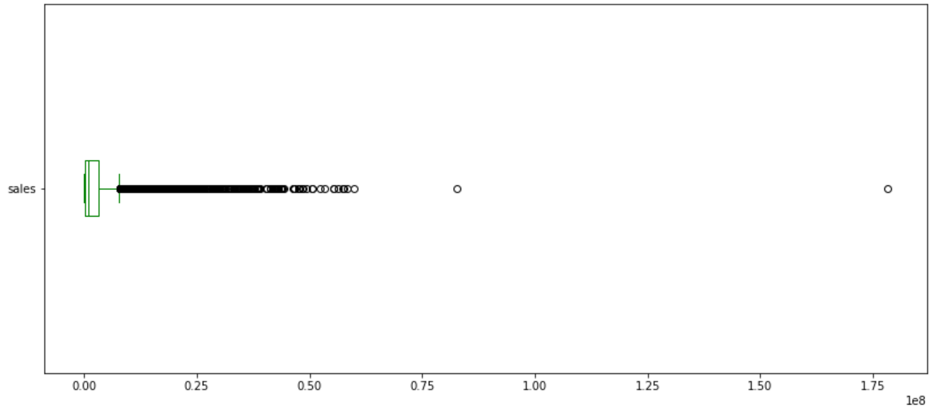
```
df['order_quantity'].max()
```

```
50
```

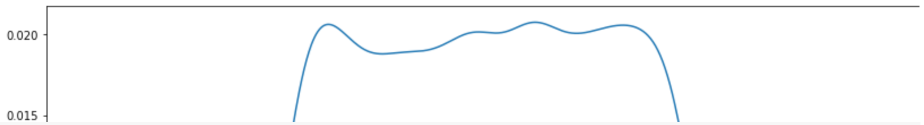
```
df['order_quantity'].plot(kind='box', vert=False, figsize=(14,6))
plt.show()
```



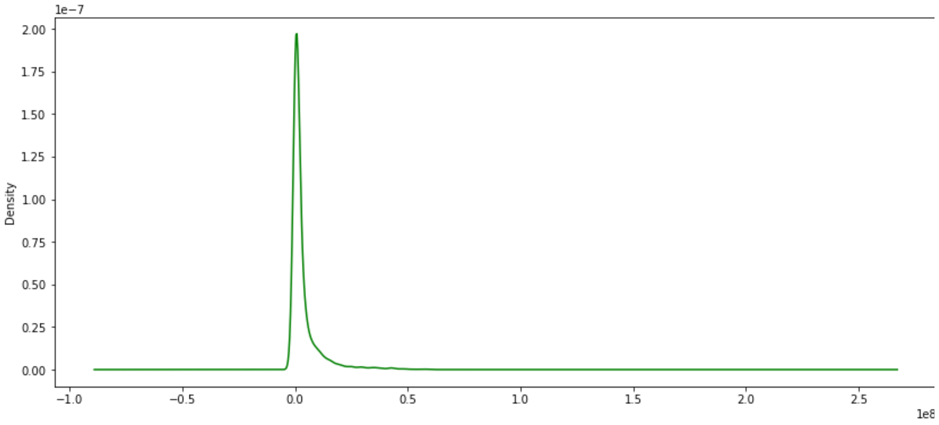
```
df['sales'].plot(kind='box', vert=False, figsize=(14,6), color='g')
plt.show()
```



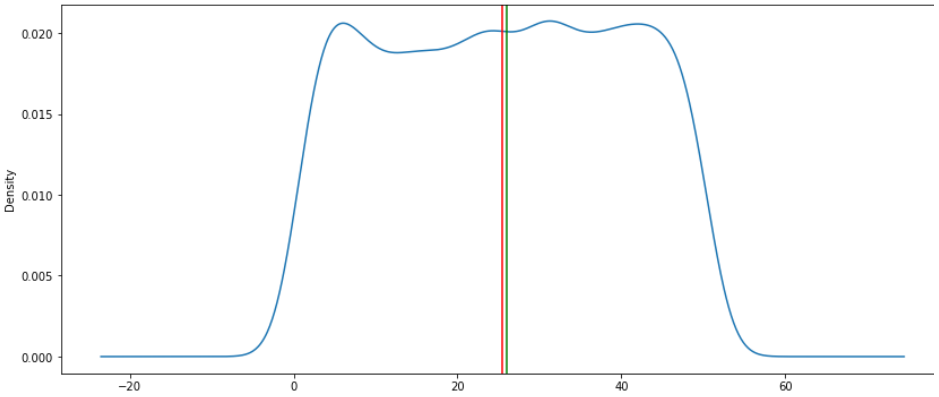
```
df['order_quantity'].plot(kind='density', figsize=(14,6))
plt.show()
```



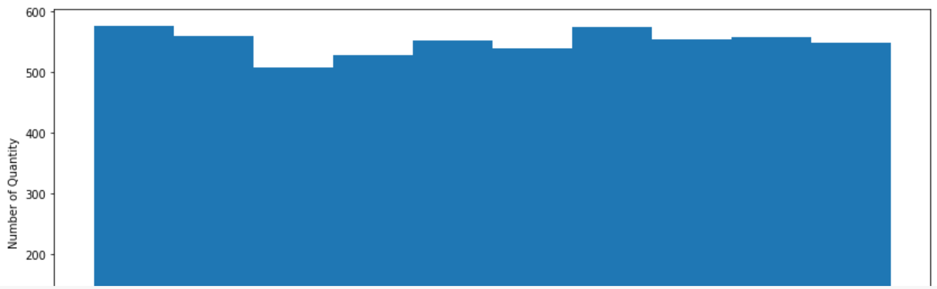
```
df['sales'].plot(kind='density', figsize=(14,6), color='g')
plt.show()
```



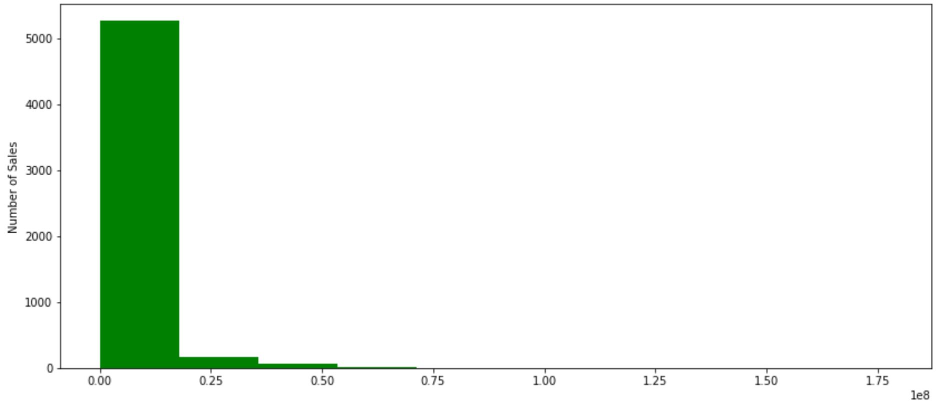
```
ax = df['order_quantity'].plot(kind='density', figsize=(14,6))
ax.axvline(df['order_quantity'].mean(), color='red')
ax.axvline(df['order_quantity'].median(), color='green')
plt.show()
```



```
ax = df['order_quantity'].plot(kind='hist', figsize=(14,6))
ax.set_ylabel('Number of Quantity')
ax.set_xlabel('dollars')
plt.show()
```



```
ax = df['sales'].plot(kind='hist', figsize=(14,6), color='g')
ax.set_ylabel('Number of Sales')
plt.show()
```



Categorycal Analysis

Order Status

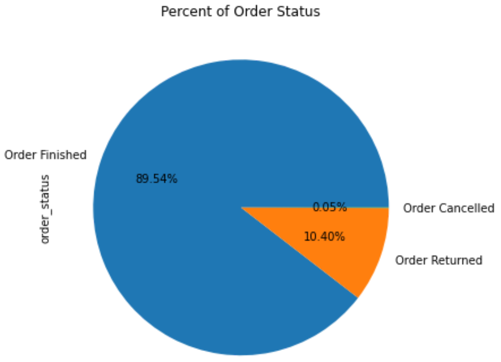
```
df.head()
```

	order_id	order_status	customer	order_date	order_quantity	sales	discount	discount_value	product_
0	3	Order Finished	Muhammed Mac Intyre	2010-10-13	6	523080	0.04	20923	Off
1	293	Order Finished	Barry French	2012-10-01	49	20246040	0.07	1417223	Off
2	483	Order Finished	Clay Rozendal	2011-07-10	30	9931519	0.08	794522	
3	515	Order Finished	Carlos Soltero	2010-08-28	19	788540	0.08	63083	Off

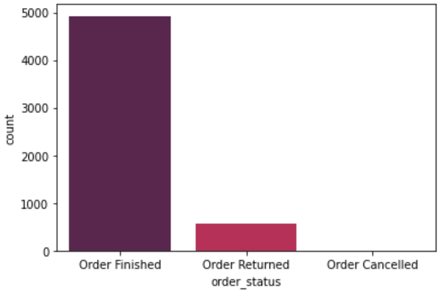
```
df['order_status'].value_counts()
```

```
Order Finished     4924
Order Returned      572
Order Cancelled       3
Name: order_status, dtype: int64
```

```
df['order_status'].value_counts().plot(kind='pie', figsize=(6,6),autopct='%.2f%%')
plt.title('Percent of Order Status')
plt.show()
```



```
sns.countplot(x = 'order_status',data = df,order = df['order_status'].value_counts().head(10).index, palette = 'ro
plt.show()
```



Product Category

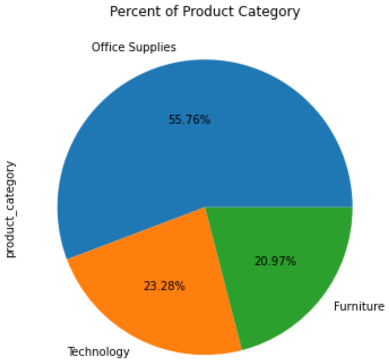
```
df.head()
```

	order_id	order_status	customer	order_date	order_quantity	sales	discount	discount_value	product_
0	3	Order Finished	Muhammed Mac Intyre	2010-10-13	6	523080	0.04	20923	Off
1	293	Order Finished	Barry French	2012-10-01	49	20246040	0.07	1417223	Off
2	483	Order Finished	Clay Rozendal	2011-07-10	30	9931519	0.08	794522	
3	515	Order Finished	Carlos Soltero	2010-08-28	19	788540	0.08	63083	Off

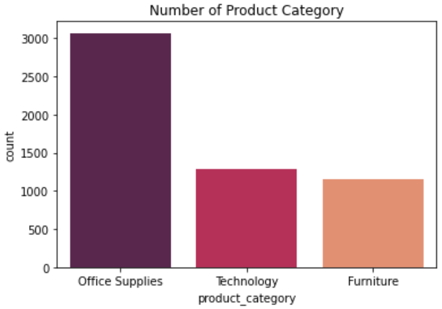
```
df['product_category'].value_counts()
```

```
Office Supplies     3066
Technology          1280
Furniture           1153
Name: product_category, dtype: int64
```

```
df['product_category'].value_counts().plot(kind='pie', figsize=(6,6),autopct='%.2f%%')
plt.title('Percent of Product Category')
plt.show()
```



```
sns.countplot(x = 'product_category',data = df,order = df['product_category'].value_counts().head(10).index, palet
plt.title('Number of Product Category')
plt.show()
```

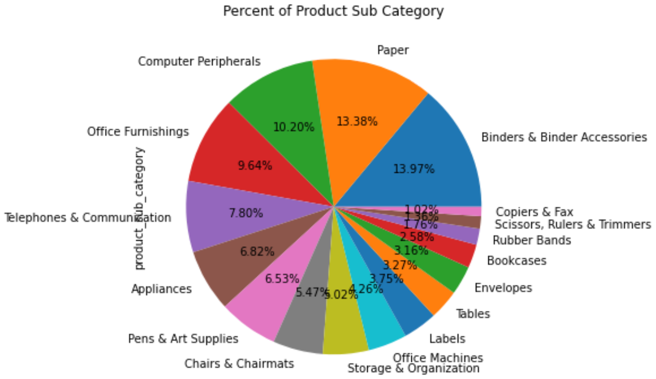


Product Sub Category

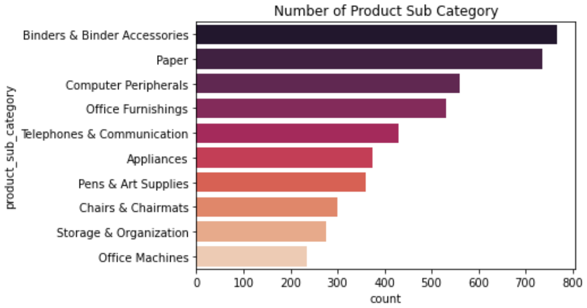
```
df['product_sub_category'].value_counts()
```

```
    Binders & Binder Accessories    768
    Paper                           736
    Computer Peripherals            561
    Office Furnishings              530
    Telephones & Communication      429
    Appliances                      375
    Pens & Art Supplies             359
    Chairs & Chairmats              301
    Storage & Organization          276
    Office Machines                 234
    Labels                          206
    Tables                          180
    Envelopes                       174
    Bookcases                       142
    Rubber Bands                     97
    Scissors, Rulers & Trimmers      75
    Copiers & Fax                    56
    Name: product_sub_category, dtype: int64
```

```
df['product_sub_category'].value_counts().plot(kind='pie', figsize=(6,6),autopct='%.2f%%')
plt.title('Percent of Product Sub Category')
plt.show()
```



```
sns.countplot(y = 'product_sub_category',data = df,order = df['product_sub_category'].value_counts().head(10).inde
plt.title('Number of Product Sub Category')
plt.show()
```

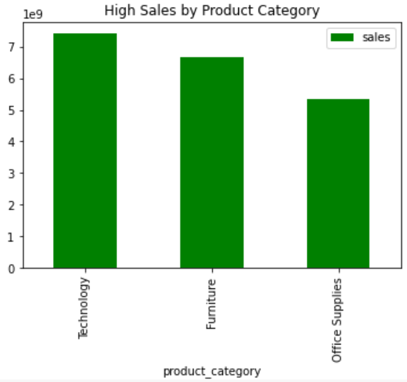


Deeper analysis

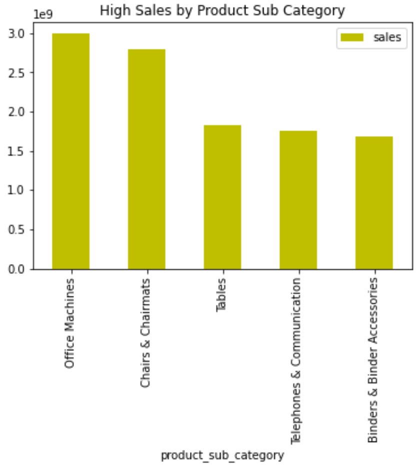
```
df.head()
```

	order_id	order_status	customer	order_date	order_quantity	sales	discount	discount_value	product_
0	3	Order Finished	Muhammed Mac Intyre	2010-10-13	6	523080	0.04	20923	Off
1	293	Order Finished	Barry French	2012-10-01	49	20246040	0.07	1417223	Off
2	483	Order Finished	Clay Rozendal	2011-07-10	30	9931519	0.08	794522	
3	515	Order Finished	Carlos Soltero	2010-08-28	19	788540	0.08	63083	Off

```
df.groupby(['product_category']).sum()[['sales']].sort_values(by="sales",ascending=False).nlargest(n=5, columns=['
plt.title('High Sales by Product Category')
plt.show()
```

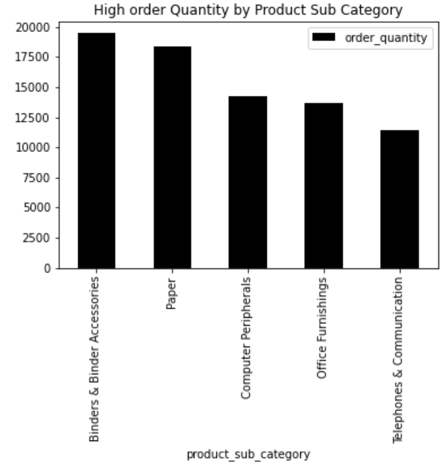


```
df.groupby(['product_sub_category']).sum()[['sales']].sort_values(by="sales",ascending=False).nlargest(n=5, colum
plt.title('High Sales by Product Sub Category')
plt.show()
```

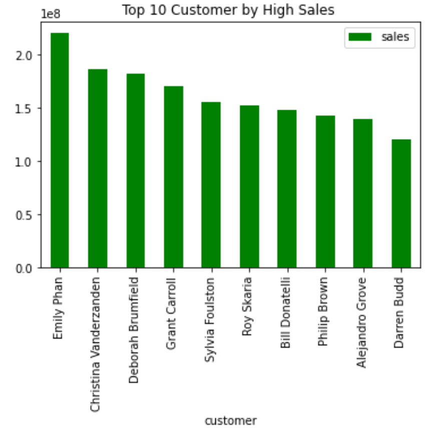


```
df.groupby(['product_category']).sum()[['order_quantity']].sort_values(by="order_quantity",ascending=False).nlarge
plt.title('High Order Quantity by Product Category')
plt.show()
```

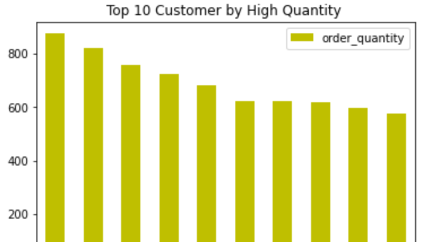
```
df.groupby(['product_sub_category']).sum()[['order_quantity']].sort_values(by="order_quantity",ascending=False).nl
plt.title('High order Quantity by Product Sub Category')
plt.show()
```



```
df.groupby(['customer']).sum()[['sales']].sort_values(by="sales",ascending=False).nlargest(n=10, columns=['sales'
plt.title('Top 10 Customer by High Sales ')
plt.show()
```



```
df.groupby(['customer']).sum()[['order_quantity']].sort_values(by="order_quantity").nlargest(n=10, columns=['order
plt.title('Top 10 Customer by High Quantity ')
plt.show()
```

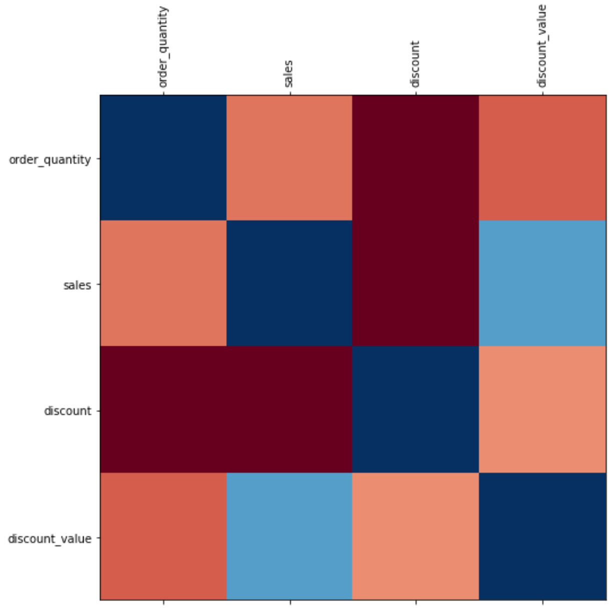


Relationship between the columns?

```
new_df = df.drop(columns='order_id')
corr = new_df.corr()

corr
```

	order_quantity	sales	discount	discount_value
order_quantity	1.000000	0.223802	-0.012348	0.188098
sales	0.223802	1.000000	-0.012213	0.771908
discount	-0.012348	-0.012213	1.000000	0.257164
discount_value	0.188098	0.771908	0.257164	1.000000

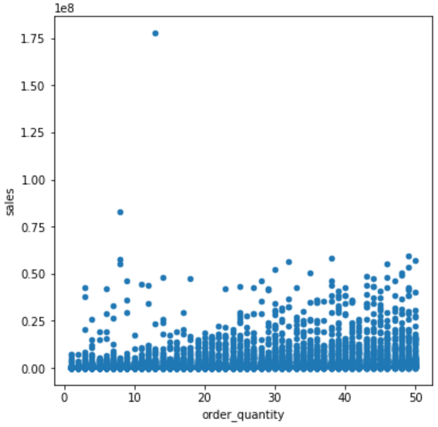
```
fig = plt.figure(figsize=(8,8))
plt.matshow(corr, cmap='RdBu', fignum=fig.number)
plt.xticks(range(len(corr.columns)), corr.columns, rotation='vertical');
plt.yticks(range(len(corr.columns)), corr.columns);
```



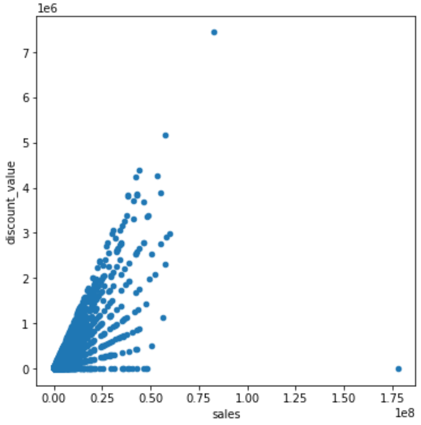
```
new_df.head()
```

	order_status	customer	order_date	order_quantity	sales	discount	discount_value	product_category
0	Order Finished	Muhammed Mac Intyre	2010-10-13	6	523080	0.04	20923	Office Supplies
1	Order Finished	Barry French	2012-10-01	49	20246040	0.07	1417223	Office Supplies
2	Order Finished	Clay Rozendal	2011-07-10	30	9931519	0.08	794522	Technology

```
new_df.plot(kind='scatter', x='order_quantity', y='sales', figsize=(6,6))
plt.show()
```

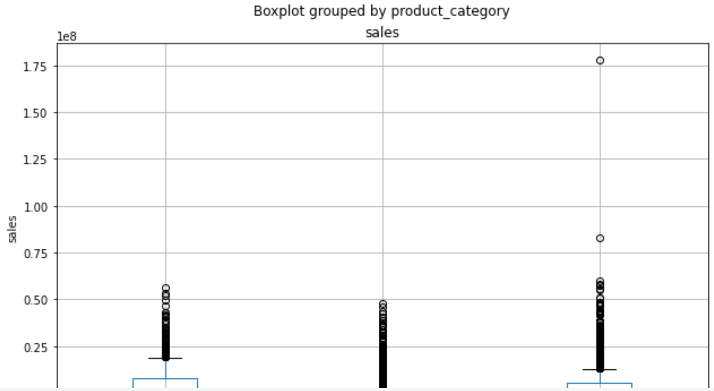


```
new_df.plot(kind='scatter', x='sales', y='discount_value', figsize=(6,6))
plt.show()
```



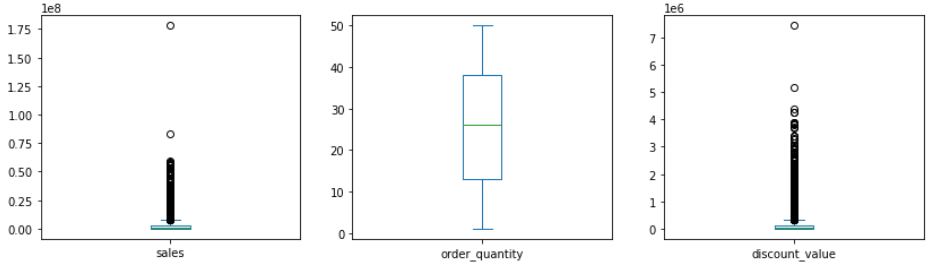
```
ax = new_df[['sales', 'product_category']].boxplot(by='product_category', figsize=(10,6))
ax.set_ylabel('sales')
plt.show()
```

```
/usr/local/lib/python3.7/dist-packages/matplotlib/cbook/__init__.py:1376: VisibleDeprecationWarning: Creatin
  X = np.atleast_1d(X.T if isinstance(X, np.ndarray) else np.asarray(X))
```



```
boxplot_cols = ['order_date', 'sales', 'order_quantity', 'discount_value']

new_df[boxplot_cols].plot(kind='box', subplots=True, layout=(2,3), figsize=(14,8))
```

```
sales             AxesSubplot(0.125,0.536818;0.227941x0.343182)
order_quantity    AxesSubplot(0.398529,0.536818;0.227941x0.343182)
discount_value    AxesSubplot(0.672059,0.536818;0.227941x0.343182)
dtype: object
```

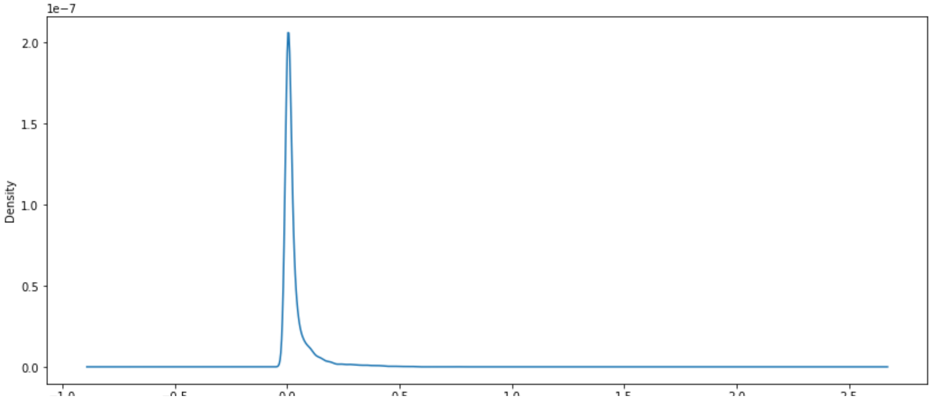


Columns Wrangling

```
new_df['revenue'] = new_df['sales'] - new_df['discount_value']

new_df.head()
```

	order_status	customer	order_date	order_quantity	sales	discount	discount_value	product_category
0	Order Finished	Muhammed Mac Intyre	2010-10-13	6	523080	0.04	20923	Office Supplies
1	Order Finished	Barry French	2012-10-01	49	20246040	0.07	1417223	Office Supplies
2	Order Finished	Clay Rozendal	2011-07-10	30	9931519	0.08	794522	Technology
3	Order Finished	Carlos Soltero	2010-08-28	19	788540	0.08	63083	Office Supplies

```
new_df['revenue'].plot(kind='density', figsize=(14,6))
plt.show()
```



Final Analyst

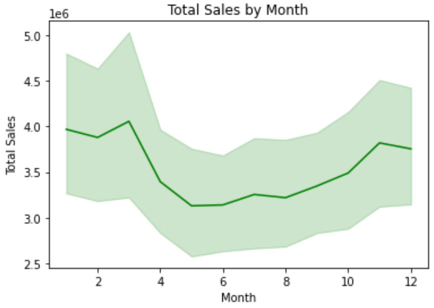
```
new_df['year'] = new_df['order_date'].apply(lambda order_date:order_date.year)
new_df['month'] = new_df['order_date'].apply(lambda order_date: order_date.month)
```

```
new_df.head()
```

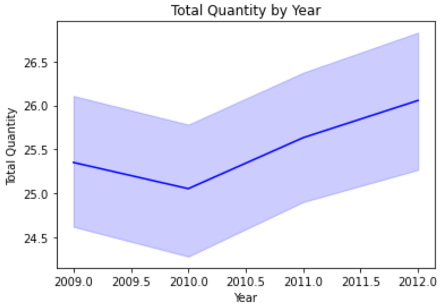
	order_status	customer	order_date	order_quantity	sales	discount	discount_value	product_category
0	Order Finished	Muhammed Mac Intyre	2010-10-13	6	523080	0.04	20923	Office Supplies
1	Order Finished	Barry French	2012-10-01	49	20246040	0.07	1417223	Office Supplies
2	Order Finished	Clay Rozendal	2011-07-10	30	9931519	0.08	794522	Technology
3	Order Finished	Carlos Soltero	2010-08-28	19	788540	0.08	63083	Office Supplies
4	Order Finished	Carl Jackson	2011-06-17	12	187080	0.03	5612	Office Supplies

```
sns.lineplot(x=new_df['year'],y=new_df['sales'],color='g')


plt.title("Total Sales by Year")
plt.xlabel("Year")
plt.ylabel("Total Sales")

plt.show()
```

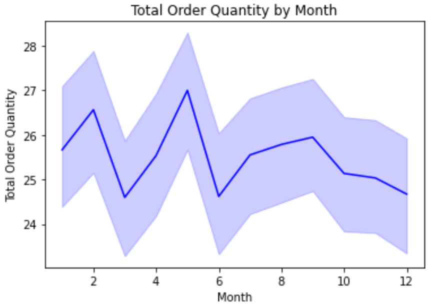
```
sns.lineplot(x=new_df['month'],y=new_df['sales'], color='g')

plt.title("Total Sales by Month")
plt.xlabel("Month")
plt.ylabel("Total Sales")

plt.show()
```



```
sns.lineplot(x=new_df['year'],y=new_df['order_quantity'], color='b')


plt.title("Total Quantity by Year")
plt.xlabel("Year")
plt.ylabel("Total Quantity")

plt.show()
```



```
sns.lineplot(x=new_df['month'],y=new_df['order_quantity'], color='b')

plt.title("Total Order Quantity by Month")
plt.xlabel("Month")
plt.ylabel("Total Order Quantity")

plt.show()
```



✓ 0 d selesai pada 11.24