Importing Libraries

```
import pandas as pd
import numpy as np
import plotly.express as px
```

Read csv **file**

```
df = pd.read_csv('apple.csv')
```

Returnig info of csv file

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 62 entries, 0 to 61
Data columns (total 11 columns):
 #   Column               Non-Null Count  Dtype  
---  ------               --------------  -----  
 0   Product Name         62 non-null     object 
 1   Product URL          62 non-null     object 
 2   Brand                62 non-null     object 
 3   Sale Price           62 non-null     int64  
 4   Mrp                  62 non-null     int64  
 5   Discount Percentage  62 non-null     int64  
 6   Number Of Ratings    62 non-null     int64  
 7   Number Of Reviews    62 non-null     int64  
 8   Upc                  62 non-null     object 
 9   Star Rating          62 non-null     float64
 10  Ram                  62 non-null     object 
dtypes: float64(1), int64(5), object(5)
memory usage: 5.5+ KB
```

Describing csv file

```
df.describe()
```

	Sale Price	Mrp	Discount Percentage	Number Of Ratings	Number Of Reviews	Star Rating
count	62.000000	62.000000	62.000000	62.000000	62.000000	62.000000
mean	80073.887097	88058.064516	9.951613	22420.403226	1861.677419	4.575806
std	34310.446132	34728.825597	7.608079	33768.589550	2855.883830	0.059190
min	29999.000000	39900.000000	0.000000	542.000000	42.000000	4.500000
25%	49900.000000	54900.000000	6.000000	740.000000	64.000000	4.500000
50%	75900.000000	79900.000000	10.000000	2101.000000	180.000000	4.600000
75%	117100.000000	120950.000000	14.000000	43470.000000	3331.000000	4.600000
max	140900.000000	149900.000000	29.000000	95909.000000	8161.000000	4.700000

First 10 rows of csv file

```
df.head(10)
```

	Product Name	Product URL	Brand	Sale Price	Mrp	Discount Percentage	Number Of Ratings	Number Of Reviews	Upc	Star Rating	Ram
0	APPLE iPhone 8 Plus (Gold, 64 GB)	https://www.flipkart.com/apple-iphone-8-plus-g...	Apple	49900	49900	0	3431	356	MOBEXRGV7EHHTGUH	4.6	2 GB
1	APPLE iPhone 8 Plus (Space Grey, 256 GB)	https://www.flipkart.com/apple-iphone-8-plus-s...	Apple	84900	84900	0	3431	356	MOBEXRGVAC6TJT4F	4.6	2 GB
2	APPLE iPhone 8 Plus (Silver, 256 GB)	https://www.flipkart.com/apple-iphone-8-plus-s...	Apple	84900	84900	0	3431	356	MOBEXRGVGETABXWZ	4.6	2 GB
3	APPLE iPhone 8 (Silver, 256 GB)	https://www.flipkart.com/apple-iphone-8-silver...	Apple	77000	77000	0	11202	794	MOBEXRGVMZWUHCBA	4.5	2 GB
4	APPLE iPhone 8 (Gold, 256 GB)	https://www.flipkart.com/apple-iphone-8-gold-2...	Apple	77000	77000	0	11202	794	MOBEXRGVPK7PFEJZ	4.5	2 GB
5	APPLE iPhone 8 Plus (Silver, 64 GB)	https://www.flipkart.com/apple-iphone-8-plus-s...	Apple	49900	49900	0	3431	356	MOBEXRGVQGYYP8FV	4.6	2 GB
6	APPLE iPhone 8 Plus (Space Grey, 64 GB)	https://www.flipkart.com/apple-iphone-8-plus-s...	Apple	49900	49900	0	3431	356	MOBEXRGVQKBREZP8	4.6	2 GB
7	APPLE iPhone 8 (Space Grey, 256 GB)	https://www.flipkart.com/apple-iphone-8-space-...	Apple	77000	77000	0	11202	794	MOBEXRGVZFZGZEWV	4.5	2 GB

Next steps: Generate code with df | View recommended plots | New interactive sheet

Last 10 rows of csv file

```
df.tail(10)
```

	Product Name	Product URL	Brand	Sale Price	Mrp	Discount Percentage	Number Of Ratings	Number Of Reviews	Upc	Star Rating	Ram
52	APPLE iPhone SE (White, 64 GB)	https://www.flipkart.com/apple-iphone-se-white...	Apple	29999	39900	24	95807	8154	MOBFWQ6BGWDVGF3E	4.5	2 GB
53	APPLE iPhone SE (Black, 128 GB)	https://www.flipkart.com/apple-iphone-se-black...	Apple	34999	44900	22	95909	8161	MOBFWQ6BHUEVZPXD	4.5	2 GB
54	APPLE iPhone SE (White, 128 GB)	https://www.flipkart.com/apple-iphone-se-white...	Apple	34999	44900	22	95807	8154	MOBFWQ6BJEHMUUZY	4.5	2 GB
55	APPLE iPhone SE (Red, 128 GB)	https://www.flipkart.com/apple-iphone-se-red-1...	Apple	34999	44900	22	95909	8161	MOBFWQ6BJTVFKPEJ	4.5	2 GB
56	APPLE iPhone 11 (Black,	https://www.flipkart.com/apple-iphone-11-black...	Apple	54999	59900	8	43470	3331	MOBFWQ6BKRYBP5X8	4.6	4 GB

Return not null data

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

	0
Product Name	0
Product URL	0
Brand	0
Sale Price	0
Mrp	0
Discount Percentage	0
Number Of Ratings	0
Number Of Reviews	0
Upc	0
Star Rating	0
Ram	0

dtype: int64

All iphones and their ratings

```
df[['Product Name','Star Rating']]
```

	Product Name	Star Rating
0	APPLE iPhone 8 Plus (Gold, 64 GB)	4.6
1	APPLE iPhone 8 Plus (Space Grey, 256 GB)	4.6
2	APPLE iPhone 8 Plus (Silver, 256 GB)	4.6
3	APPLE iPhone 8 (Silver, 256 GB)	4.5
4	APPLE iPhone 8 (Gold, 256 GB)	4.5
...	...	...
57	APPLE iPhone SE (Black, 64 GB)	4.5
58	APPLE iPhone 11 (Purple, 64 GB)	4.6
59	APPLE iPhone 11 (White, 64 GB)	4.6
60	APPLE iPhone 11 (Black, 64 GB)	4.6
61	APPLE iPhone 11 (Red, 64 GB)	4.6

62 rows × 2 columns

Top 5 star rating apple phones

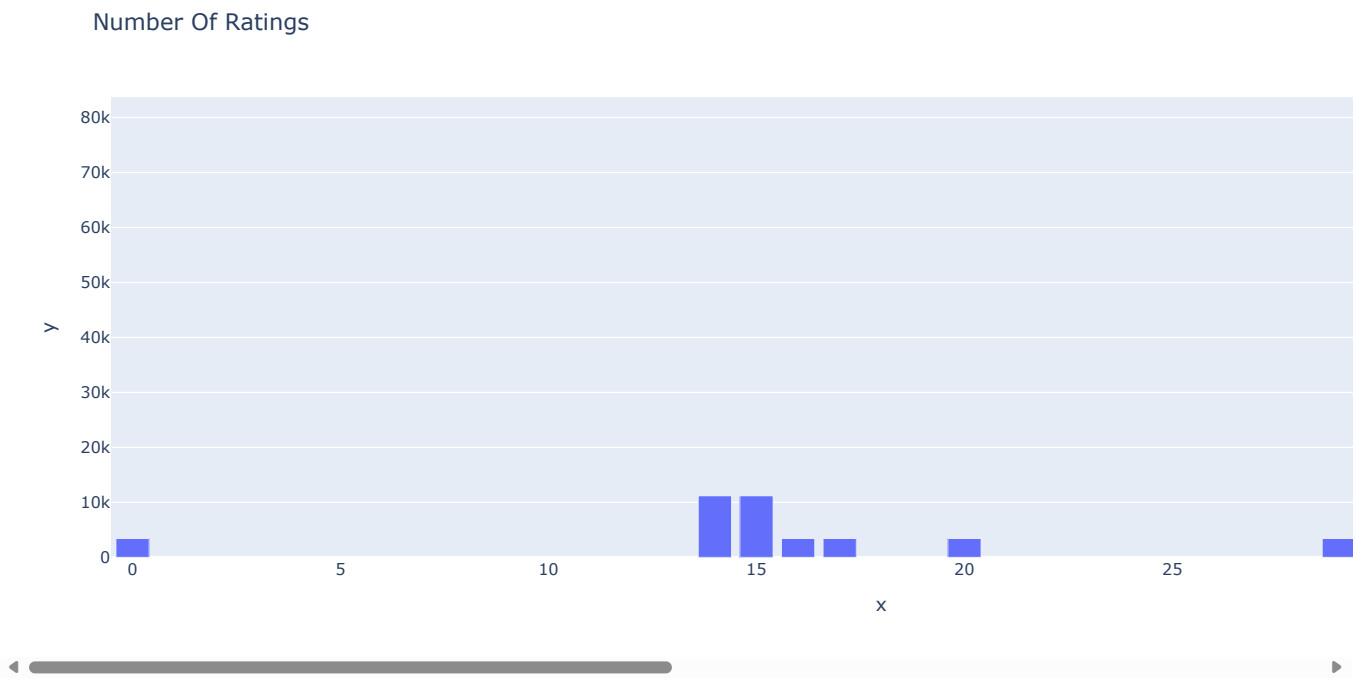
```
df_sorted = df.sort_values('Star Rating', ascending=False)
df_sorted.head(5)
```

	Product Name	Product URL	Brand	Sale Price	Mrp	Discount Percentage	Number Of Ratings	Number Of Reviews	Upc	Star Rating	Ram
20	APPLE iPhone 11 Pro Max (Midnight Green, 64 GB)	https://www.flipkart.com/apple-iphone-11-pro-m...	Apple	117100	117100	0	1078	101	MOBFKCTSRYPAQNYT	4.7	4 GB
17	APPLE iPhone 11 Pro Max (Space	https://www.flipkart.com/apple-iphone-11-pro-m...	Apple	117100	117100	0	1078	101	MOBFKCTSKDMKCGQS	4.7	4 GB

Next steps: Generate code with df_sorted | View recommended plots | New interactive sheet

Top 10 Iphones and their rating according to their Number Of Ratings

```
label = df_sorted.head(10).index
counts = df['Number Of Ratings'].head(10)
fig = px.bar(x=label, y=counts, title='Number Of Ratings')
fig.show()
```



Most expensive Iphone

```
most_expensive = df.loc[df['Sale Price'].idxmax()]
most_expensive
```

	24
Product Name	APPLE iPhone 12 Pro (Silver, 512 GB)
Product URL	https://www.flipkart.com/apple-iphone-12-pro-s...
Brand	Apple
Sale Price	140900
Mrp	149900
Discount Percentage	6
Number Of Ratings	542
Number Of Reviews	42
Upc	MOBFWBYZ5UY6ZBVA
Star Rating	4.5
Ram	4 GB

dtype: object

Least expensive Iphone

```
least_expensive = df.loc[df['Sale Price'].idxmin()]
least_expensive
```

	52
Product Name	APPLE iPhone SE (White, 64 GB)
Product URL	https://www.flipkart.com/apple-iphone-se-white...
Brand	Apple
Sale Price	29999
Mrp	39900
Discount Percentage	24
Number Of Ratings	95807
Number Of Reviews	8154
Upc	MOBFWQ6BGWDVGF3E
Star Rating	4.5
Ram	2 GB

dtype: object

Lowest 5 star rating apple phones

```
df_sorted = df.sort_values('Star Rating')
df_sorted.head(5)
```

	Product Name	Product URL	Brand	Sale Price	Mrp	Discount Percentage	Number Of Ratings	Number Of Reviews	Upc	Star Rating	Ram
30	APPLE iPhone 12 Pro (Graphite, 128 GB)	https://www.flipkart.com/apple-iphone-12-pro-g...	Apple	110900	119900	7	545	42	MOBFWBYZBZ7Y56WD	4.5	6 GB
38	APPLE iPhone 12 Mini (Red, 64 GB)	https://www.flipkart.com/apple-iphone-12-mini-...	Apple	59900	69900	14	740	64	MOBFWBYZNVWGWN2U	4.5	6 GB
	APPLE										

Next steps: Generate code with df_sorted | View recommended plots | New interactive sheet

Apple phone have highest number of reviews

```
df_highest_reviews = df.sort_values('Number Of Reviews', ascending=False)
df_highest_reviews.head(1)
```

	Product Name	Product URL	Brand	Sale Price	Mrp	Discount Percentage	Number Of Ratings	Number Of Reviews	Upc	Star Rating	Ram
	Apple										

Next steps: Generate code with df_highest_reviews | View recommended plots | New interactive sheet

Apple phone have lowest number of reviews

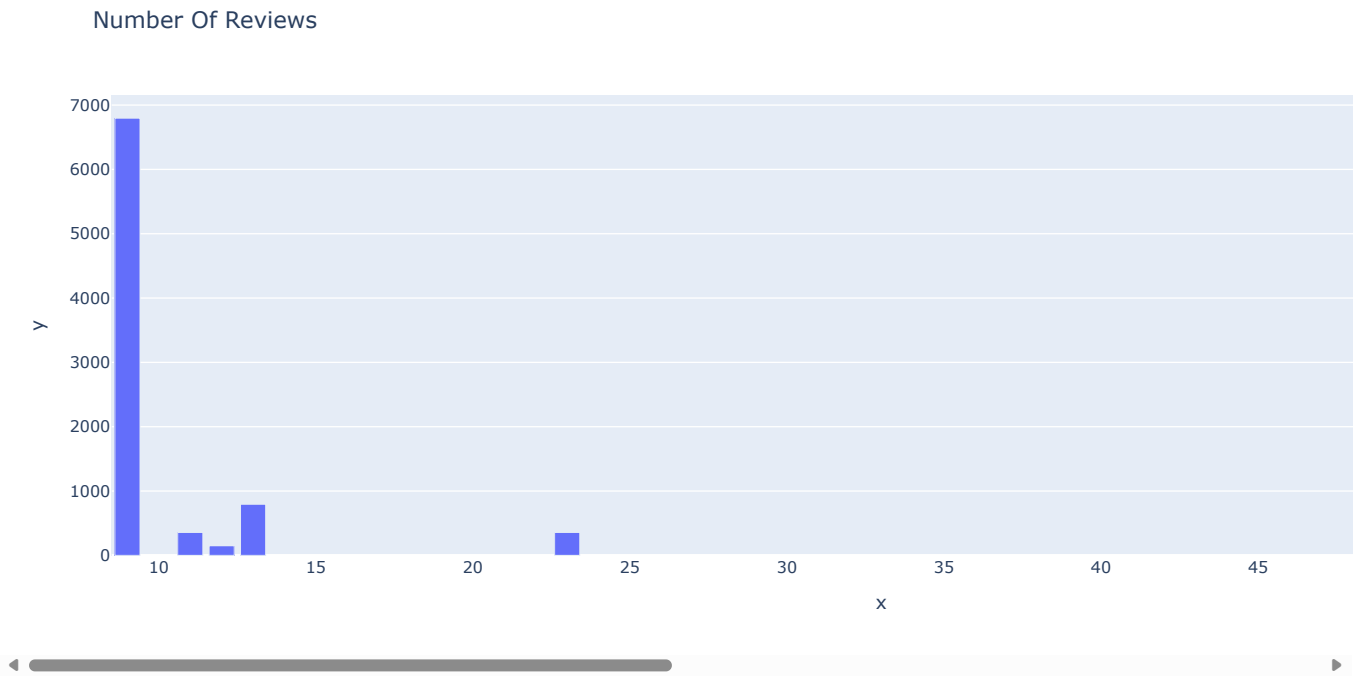
```
df_lowest_reviews = df.sort_values('Number Of Reviews')
df_lowest_reviews.head(1)
```

	Product Name	Product URL	Brand	Sale Price	Mrp	Discount Percentage	Number Of Ratings	Number Of Reviews	Upc	Star Rating	Ram
	APPLE										

Next steps: Generate code with df_lowest_reviews | View recommended plots | New interactive sheet

Top 10 Iphones and their reviews , according to their Number Of Review

```
label = df_highest_reviews.head(10).index
counts = df['Number Of Reviews'].head(10)
fig = px.bar(df,x=label,y=counts,title='Number Of Reviews')
fig.show()
```



Relationship between sale price and number of ratings

```
correlation = df['Sale Price'].corr(df['Number Of Ratings'])
print(correlation)
```

```
-0.7015259181182026
```

Relationship between sale price and number of reviews

```
correlation = df['Sale Price'].corr(df['Number Of Reviews'])
print(correlation)
```

```
-0.6960291835220087
```

Relationship between discount percentage and number of ratings

```
correlation = df['Discount Percentage'].corr(df['Number Of Ratings'])
print(correlation)
```

```
0.6848270553540624
```

Relationship between discount percentage and number of reviews

```
correlation = df['Discount Percentage'].corr(df['Number Of Reviews'])
print(correlation)
```

```
0.6858769720978277
```

Most expensive and least expensive apple phone

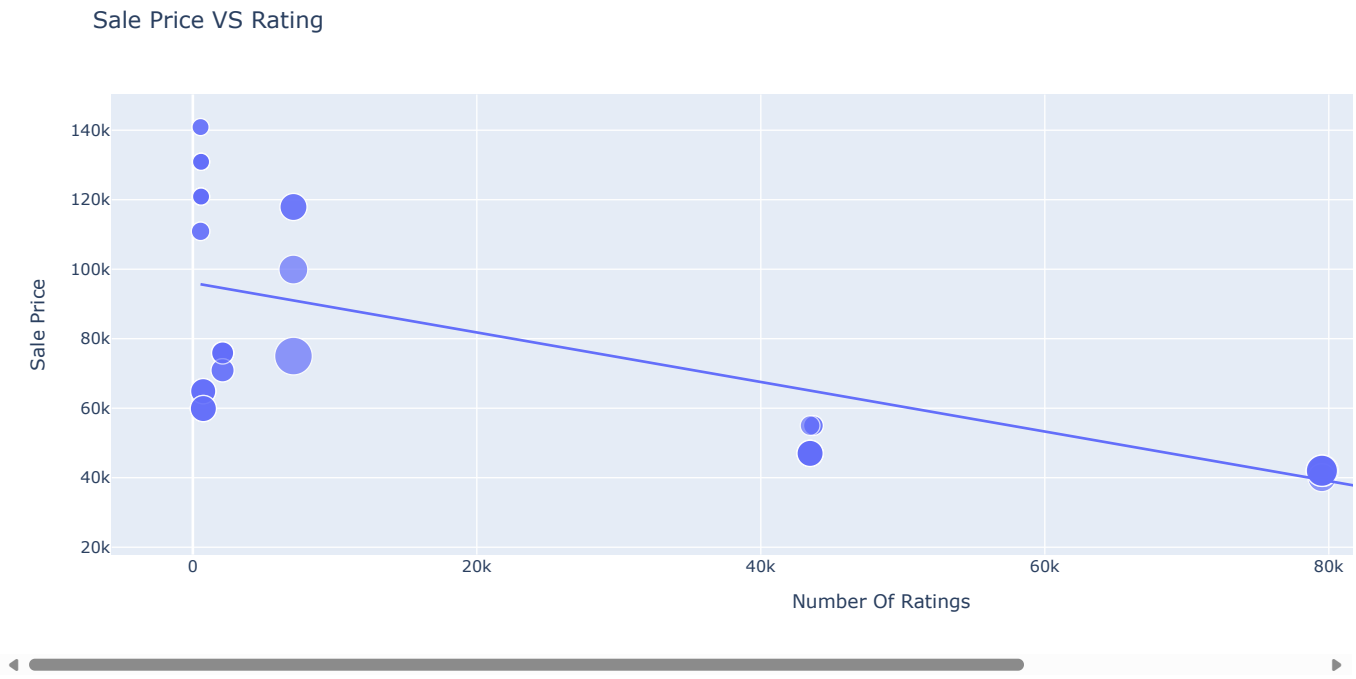
```
df_sorted = df.sort_values('Sale Price', ascending=False)
most_expensive_phone = df_sorted.iloc[0]
most_expensive_phone
```

	41
Product Name	APPLE iPhone 12 Pro (Pacific Blue, 512 GB)
Product URL	https://www.flipkart.com/apple-iphone-12-pro-p...
Brand	Apple
Sale Price	140900
Mrp	149900
Discount Percentage	6
Number Of Ratings	545
Number Of Reviews	42
Upc	MOBFWBYZTHSXKMGW
Star Rating	4.5
Ram	4 GB

dtype: object

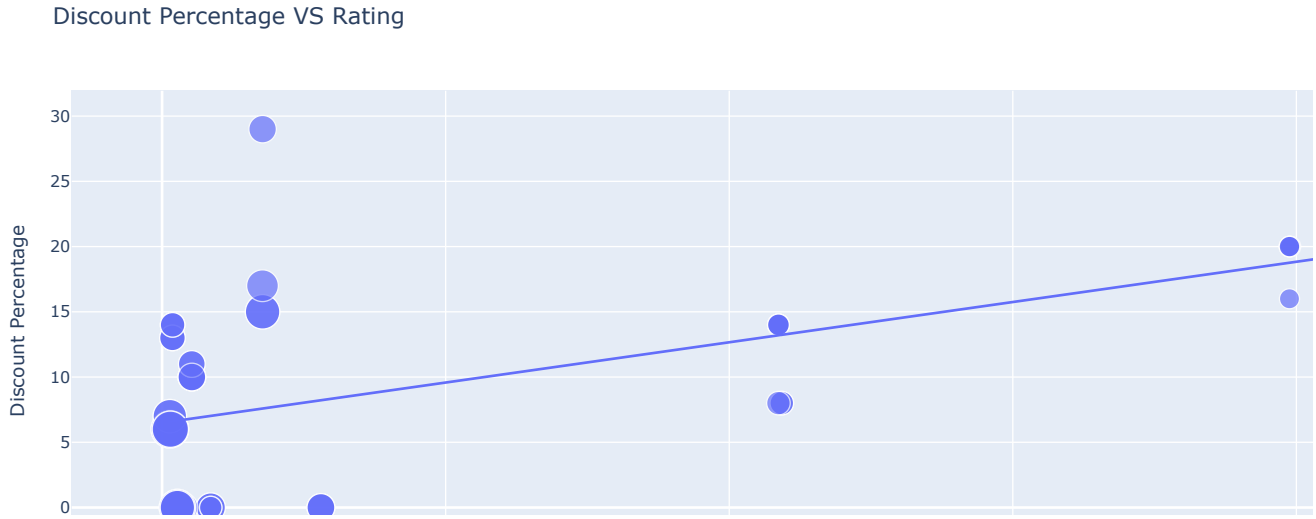
Iphone Sale Price and Rating relationship

```
fg = px.scatter(data_frame=df, x='Number Of Ratings', y='Sale Price', size='Discount Percentage', trendline='ols',title='Sale Price VS Ra
fg.show()
```



Iphone Discount Percentage and Rating Relationship

```
fg = px.scatter(data_frame=df, x='Number Of Ratings', y='Discount Percentage', size='Sale Price', trendline='ols',title='Discount Percent
fg.show()
```



Most Expensive and Least Expensive Iphone

```
product_sale_price = df[['Product Name','Sale Price']]
product_sale_price
```

	Product Name	Sale Price
0	APPLE iPhone 8 Plus (Gold, 64 GB)	49900
1	APPLE iPhone 8 Plus (Space Grey, 256 GB)	84900
2	APPLE iPhone 8 Plus (Silver, 256 GB)	84900
3	APPLE iPhone 8 (Silver, 256 GB)	77000
4	APPLE iPhone 8 (Gold, 256 GB)	77000
...	...	...
57	APPLE iPhone SE (Black, 64 GB)	29999
58	APPLE iPhone 11 (Purple, 64 GB)	46999
59	APPLE iPhone 11 (White, 64 GB)	46999
60	APPLE iPhone 11 (Black, 64 GB)	46999
61	APPLE iPhone 11 (Red, 64 GB)	46999

62 rows × 2 columns

Next steps: Generate code with product_sale_price | View recommended plots | New interactive sheet

Sale prices of iphone

```
sale_price = product_sale_price['Sale Price']
sale_price
```

	Sale Price
0	49900
1	84900
2	84900