In [1]:
```python
import numpy as np
import pandas as pd
```

In [2]:
```python
df=pd.read_csv(r'C:\Users\moayad\Desktop\Data Analysis\Perfume data analysis\messy_dataset.csv')
```

In [3]:
```python
df.head()
```

Out[3]:

	Customer ID	Sales ID	Product Name	Sentiment	Category	User Type	Gender	Age	Price	Cost Price	Quantity	Payment Method	City	Country	Bottle Size	Discount	Purchase Date	Rating	Oil Content Percentage
0	NaN	sid1	Aqua	Positive	eau de parfum	uni	f	150	325.044124	82.724641	NaN	PayPal	Sydney	NaN	100ml	NaN	1900-01-02	6/12/2022	NaN
1	cid18393	sid2	Dusk	Positive	eau de cologne	Women	F	37	98.222536	58.933522	8.0	Card	Tokyo	Japan	50ml	81.11	2017-07-26	5	2%
2	cid3108	sid3	Aqua	Positive	eau de cologne	Women	M	34	143.470592	100.429415	2.0	Bank Transfer	Tokyo	Japan	30ml	36.22	2019-12-23	1	2%
3	cid181	sid4	Elixir	Positive	eau de cologne	Unisex	F	150	747.300840	176.283608	4.0	Card	London	UK	200ml	15.25	2018-05-06	4	4%
4	cid48531	sid5	Dusk	Positive	eau de cologne	Women	F	30	62.294481	37.376689	2.0	Cash	London	UK	30ml	44.53	2022-07-26	3	4%

In [4]:
```python
df.shape
```

Out[4]:
```
(200001, 19)
```

In [5]:
```python
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 200001 entries, 0 to 200000
Data columns (total 19 columns):
 #   Column                  Non-Null Count   Dtype
---  ------                  --------------   -----
 0   Customer ID             180001 non-null  object
 1   Sales ID                200001 non-null  object
 2   Product Name            200001 non-null  object
 3   Sentiment               200001 non-null  object
 4   Category                200001 non-null  object
 5   User Type               200001 non-null  object
 6   Gender                  200001 non-null  object
 7   Age                     200001 non-null  int64
 8   Price                   200001 non-null  float64
 9   Cost Price              200001 non-null  float64
 10  Quantity                171429 non-null  float64
 11  Payment Method          194302 non-null  object
 12  City                    200001 non-null  object
 13  Country                 181819 non-null  object
 14  Bottle Size             197059 non-null  object
 15  Discount                184616 non-null  float64
 16  Purchase Date           200001 non-null  object
 17  Rating                  200001 non-null  object
 18  Oil Content Percentage  188236 non-null  object
dtypes: float64(4), int64(1), object(14)
memory usage: 29.0+ MB
```

In [6]:
```python
df.columns
```

Out[6]:
```
Index(['Customer ID', 'Sales ID', 'Product Name', 'Sentiment', 'Category',
       'User Type', 'Gender', 'Age', 'Price', 'Cost Price', 'Quantity',
       'Payment Method', 'City', 'Country', 'Bottle Size', 'Discount',
       'Purchase Date', 'Rating', 'Oil Content Percentage'],
      dtype='object')
```

In [7]:
```python
df.duplicated().sum()
```

Out[7]:
```
1
```

In [8]:
```python
Sales_tran=df
```

In [9]:
```python
Sales_tran[Sales_tran.duplicated()]
```

Out[9]:

	Customer ID	Sales ID	Product Name	Sentiment	Category	User Type	Gender	Age	Price	Cost Price	Quantity	Payment Method	City	Country	Bottle Size	Discount	Purchase Date	Rating	Oil Content Percentage
200000	cid6205	sid13	Ciel	Positive	eau de cologne	Women	F	150	523.245238	205.88621	2.0	Card	London	UK	50ml	2.35	2017-12-10	5	3%

In [10]:
```python
Sales_tran=Sales_tran.drop_duplicates()
```

In [11]:
```python
Sales_tran.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 200000 entries, 0 to 199999
Data columns (total 19 columns):
 #   Column                  Non-Null Count   Dtype
---  ------                  --------------   -----
 0   Customer ID             180000 non-null  object
 1   Sales ID                200000 non-null  object
 2   Product Name            200000 non-null  object
 3   Sentiment               200000 non-null  object
 4   Category                200000 non-null  object
 5   User Type               200000 non-null  object
 6   Gender                  200000 non-null  object
 7   Age                     200000 non-null  int64
 8   Price                   200000 non-null  float64
 9   Cost Price              200000 non-null  float64
 10  Quantity                171428 non-null  float64
 11  Payment Method          194301 non-null  object
 12  City                    200000 non-null  object
 13  Country                 181818 non-null  object
 14  Bottle Size             197058 non-null  object
 15  Discount                184615 non-null  float64
 16  Purchase Date           200000 non-null  object
 17  Rating                  200000 non-null  object
 18  Oil Content Percentage  188235 non-null  object
dtypes: float64(4), int64(1), object(14)
memory usage: 30.5+ MB
```

In [12]:
```python
Sales_tran['Customer ID'].isnull().sum()
```

Out[12]:
```
20000
```

In [13]:
```python
Sales_tran.dropna(subset=['Customer ID','Quantity','Payment Method','Country','Bottle Size','Oil Content Percentage'],inplace=True)
```

```
C:\Users\moayad\AppData\Local\Temp\ipykernel_13332\3597757337.py:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  Sales_tran.dropna(subset=['Customer ID','Quantity','Payment Method','Country','Bottle Size','Oil Content Percentage'],inplace=True)
```

In [14]:
```python
pd.options.mode.chained_assignment = None   # To remove wearing meassage above
```

In [15]:
```python
Sales_tran.shape
```

Out[15]:
```
(132011, 19)
```

In [16]:
```python
Sales_tran.Discount.mean()
```

Out[16]:
```
50.120975011283896
```

In [17]:
```python
Sales_tran['Discount']=Sales_tran['Discount'].fillna(Sales_tran.Discount.mean())
```

In [18]:
```python
Sales_tran.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 132011 entries, 1 to 199999
Data columns (total 19 columns):
 #   Column                  Non-Null Count   Dtype
---  ------                  --------------   -----
 0   Customer ID             132011 non-null  object
 1   Sales ID                132011 non-null  object
 2   Product Name            132011 non-null  object
 3   Sentiment               132011 non-null  object
 4   Category                132011 non-null  object
 5   User Type               132011 non-null  object
 6   Gender                  132011 non-null  object
 7   Age                     132011 non-null  int64
 8   Price                   132011 non-null  float64
 9   Cost Price              132011 non-null  float64
 10  Quantity                132011 non-null  float64
 11  Payment Method          132011 non-null  object
 12  City                    132011 non-null  object
 13  Country                 132011 non-null  object
 14  Bottle Size             132011 non-null  object
 15  Discount                132011 non-null  float64
 16  Purchase Date           132011 non-null  object
 17  Rating                  132011 non-null  object
 18  Oil Content Percentage  132011 non-null  object
dtypes: float64(4), int64(1), object(14)
memory usage: 20.1+ MB
```

In [19]:
```python
Sales_tran['Product Name'].unique()
```

Out[19]:
```
array(['Dusk', 'Aqua', 'Elixir', 'Ciel', 'Belle'], dtype=object)
```

In [20]:
```python
Sales_tran['User Type'].unique()
```

Out[20]:
```
array(['Women', 'Unisex', 'uni', 'Men', 'man', 'woman'], dtype=object)
```

In [21]:
```python
Sales_tran['User Type'].replace(['Women','man','uni','woman','Men'],['Woman','Man','Unisex','Woman','Man'],inplace=True)
```

In [22]:
```python
Sales_tran['User Type'].unique()
```

Out[22]:
```
array(['Woman', 'Unisex', 'Man'], dtype=object)
```

In [23]:
```python
Sales_tran.Gender.unique()
```

Out[23]:
```
array(['F', 'M', 'm', 'f'], dtype=object)
```

In [24]:
```python
Sales_tran.Gender.replace(['f','m'],['F','M'],inplace=True)
```

In [25]:
```python
Sales_tran.Gender.unique()
```

Out[25]:
```
array(['F', 'M'], dtype=object)
```

In [26]:
```python
Sales_tran['Payment Method'].unique()
```

Out[26]:
```
array(['Card', 'Bank Transfer', 'Cash'], dtype=object)
```

In [27]:
```python
Sales_tran.City.unique()
```

Out[27]:
```
array(['Tokyo', 'London', 'Dubai', 'Paris', 'New York', 'Sydney'],
      dtype=object)
```

In [28]:
```python
Sales_tran.Country.unique()
```

Out[28]:
```
array(['Japan', 'UK', 'UAE', 'France', 'USA', 'Australia'], dtype=object)
```

In [29]:
```python
Sales_tran['Bottle Size'].unique()
```

Out[29]:
```
array(['50ml', '30ml', '200ml', '75ml', '100ml'], dtype=object)
```

In [30]:
```python
Sales_tran['Purchase Date'].dtypes      # The datatype here it's Object "String"
```

Out[30]:
```
dtype('O')
```

In [31]:
```python
Sales_tran['Purchase Date']=pd.to_datetime(Sales_tran['Purchase Date'])
```

In [32]:
```python
Sales_tran['Year']=Sales_tran['Purchase Date'].dt.year
```

In [33]:
```python
Sales_tran['Month']=Sales_tran['Purchase Date'].dt.month
```

In [34]:
```python
Sales_tran['Quarter']=Sales_tran['Purchase Date'].dt.quarter
```

In [35]:
```python
Sales_tran['Rating'].unique()
```

Out[35]:
```
array(['5', '1', '4', '3', '2', '10', '5/11/2021', '12', '6/12/2022'],
      dtype=object)
```

In [36]:
```python
Sales_tran[Sales_tran.Rating=='5/11/2021']
```

Out[36]:

	Customer ID	Sales ID	Product Name	Sentiment	Category	User Type	Gender	Age	Price	Cost Price	...	City	Country	Bottle Size	Discount	Purchase Date	Rating	Oil Content Percentage	Year	Month	Quarter
19	cid24933	sid20	Elixir	Neutral	eau de cologne	Unisex	M	32	125.371946	87.760362	...	Sydney	Australia	75ml	15.960000	2019-06-17	5/11/2021	2%	2019	6	2
38	cid32917	sid39	Belle	Positive	eau de cologne	Woman	M	33	511.683967	614.020760	...	Paris	France	75ml	61.550000	2018-06-01	5/11/2021	4%	2018	6	2
228	cid13413	sid229	Dusk	Negative	eau de cologne	Woman	M	150	324.448155	559.396580	...	London	UK	100ml	3.970000	2022-03-05	5/11/2021	4%	2022	3	1
247	cid43961	sid248	Elixir	Positive	eau de toilette	Unisex	M	34	178.713190	160.841871	...	New York	USA	30ml	50.120975	1900-01-03	5/11/2021	6%	1900	1	1
285	cid20290	sid286	Aqua	Positive	eau de parfum	Unisex	M	150	432.486359	165.184583	...	New York	USA	30ml	61.500000	2022-06-18	5/11/2021	11%	2022	6	2
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
199728	cid17989	sid199729	Aqua	Positive	eau de toilette	Unisex	M	150	855.367703	227.557389	...	Paris	France	100ml	36.450000	2019-04-08	5/11/2021	15%	2019	4	2
199823	cid14355	sid199824	Ciel	Positive	eau de cologne	Man	F	32	569.814715	683.777658	...	New York	USA	75ml	50.120975	1900-01-05	5/11/2021	2%	1900	1	1
199861	cid49515	sid199862	Dusk	Positive	eau de cologne	Woman	F	31	198.754408	139.128085	...	Sydney	Australia	50ml	0.810000	2017-11-11	5/11/2021	3%	2017	11	4
199918	cid47122	sid199919	Belle	Positive	eau de cologne	Woman	M	37	55.789901	33.473941	...	Dubai	UAE	75ml	68.700000	2019-12-11	5/11/2021	3%	2019	12	4
199994	cid33865	sid199995	Belle	Neutral	eau de cologne	Woman	M	32	373.087184	335.778465	...	New York	USA	50ml	38.310000	2022-02-03	5/11/2021	2%	2022	2	1

3505 rows × 22 columns

In [37]:
```python
Sales_tran[Sales_tran.Rating=='6/12/2022']
```

Out[37]:

	Customer ID	Sales ID	Product Name	Sentiment	Category	User Type	Gender	Age	Price	Cost Price	...	City	Country	Bottle Size	Discount	Purchase Date	Rating	Oil Content Percentage	Year	Month	Quarter
57	cid518	sid58	Ciel	Positive	eau de cologne	Man	F	150	766.089694	75.128486	...	Tokyo	Japan	75ml	42.91	2020-09-26	6/12/2022	3%	2020	9	3
76	cid31964	sid77	Belle	Positive	eau de cologne	Woman	M	43	152.630108	106.841075	...	New York	USA	100ml	24.77	2018-07-12	6/12/2022	3%	2018	7	3
95	cid18307	sid96	Belle	Positive	eau de parfum	Unisex	F	30	261.869571	209.495656	...	Sydney	Australia	100ml	12.27	2022-08-20	6/12/2022	3%	2022	8	3
114	cid40378	sid115	Aqua	Positive	eau de toilette	Woman	F	150	443.482645	103.185920	...	Tokyo	Japan	100ml	50.99	2019-02-18	6/12/2022	3%	2019	2	1
152	cid17323	sid153	Belle	Positive	eau de toilette	Woman	F	42	70.937230	49.656061	...	Sydney	Australia	200ml	59.55	2018-07-18	6/12/2022	12%	2018	7	3
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
199747	cid18453	sid199748	Elixir	Positive	eau de cologne	Man	M	18	248.480089	223.632080	...	New York	USA	75ml	27.14	2021-07-26	6/12/2022	3%	2021	7	3
199785	cid22934	sid199786	Aqua	Negative	eau de cologne	Unisex	M	150	426.804070	168.870315	...	New York	USA	100ml	71.96	2021-11-28	6/12/2022	4%	2021	11	4
199842	cid18190	sid199843	Belle	Neutral	eau de cologne	Unisex	F	150	461.156769	46.674852	...	London	UK	75ml	56.44	2020-07-16	6/12/2022	2%	2020	7	3
199956	cid7479	sid199957	Elixir	Neutral	eau de cologne	Man	F	150	88.034105	40.511103	...	Tokyo	Japan	75ml	82.02	2022-06-02	6/12/2022	4%	2022	6	2
199975	cid44571	sid199976	Aqua	Positive	eau de parfum	Man	M	24	134.020982	80.412589	...	London	UK	100ml	0.35	2021-03-03	6/12/2022	18%	2021	3	1

3443 rows × 22 columns

In [38]:
```python
pd.set_option('display.max_columns', None)
```

In [39]:
```python
Sales_tran[Sales_tran.Rating=='10']
```

Out[39]:

	Customer ID	Sales ID	Product Name	Sentiment	Category	User Type	Gender	Age	Price	Cost Price	Quantity	Payment Method	City	Country	Bottle Size	Discount	Purchase Date	Rating	Oil Content Percentage	Year	Month	Quarter
9	cid46921	sid10	Belle	Positive	eau de cologne	Man	M	150	982.651592	94.037638	2.0	Bank Transfer	London	UK	50ml	13.900000	2020-02-27	10	3%	2020	2	1
37	cid1217	sid38	Dusk	Neutral	eau de parfum	Woman	M	29	132.349742	79.409845	9.0	Bank Transfer	Paris	France	100ml	83.270000	2020-03-09	10	16%	2020	3	1
83	cid42020	sid84	Dusk	Positive	eau de cologne	Man	M	29	82.309047	57.616333	3.0	Card	Dubai	UAE	200ml	92.140000	2017-09-29	10	3%	2017	9	3
104	cid5737	sid105	Elixir	Negative	eau de cologne	Unisex	M	28	63.104951	37.862970	5.0	Cash	London	UK	100ml	50.120975	1900-01-05	10	2%	1900	1	1
137	cid30663	sid138	Aqua	Positive	eau de cologne	Woman	M	44	136.918409	109.534727	2.0	Card	New York	USA	30ml	58.340000	2021-11-08	10	4%	2021	11	4
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
199896	cid4150	sid199897	Belle	Positive	eau de cologne	Unisex	M	150	269.340930	467.350308	10.0	Bank Transfer	London	UK	30ml	3.340000	2019-05-22	10	3%	2019	5	2
199908	cid13380	sid199909	Elixir	Positive	eau de cologne	Unisex	M	150	46.859748	135.246129	4.0	Card	London	UK	200ml	91.600000	2022-02-24	10	3%	2022	2	1
199935	cid44309	sid199936	Ciel	Positive	eau de cologne	Woman	F	150	34.835432	73.918113	1.0	Card	London	UK	50ml	94.670000	2017-04-01	10	3%	2017	4	2
199944	cid29044	sid199945	Dusk	Negative	eau de cologne	Man	M	150	423.533187	265.688745	2.0	Card	Sydney	Australia	75ml	39.690000	2017-01-31	10	3%	2017	1	1
199987	cid18546	sid199988	Ciel	Positive	eau de cologne	Man	M	40	98.228795	58.937277	1.0	Card	New York	USA	200ml	6.190000	2019-03-08	10	2%	2019	3	1

9794 rows × 22 columns

In [40]:
```python
Sales_tran[Sales_tran.Rating=='12']
```

Out[40]:

	Customer ID	Sales ID	Product Name	Sentiment	Category	User Type	Gender	Age	Price	Cost Price	Quantity	Payment Method	City	Country	Bottle Size	Discount	Purchase Date	Rating	Oil Content Percentage	Year	Month	Quarter
23	cid5952	sid24	Belle	Positive	eau de parfum	Man	M	23	185.289732	148.231786	3.0	Cash	Dubai	UAE	50ml	21.360000	2021-11-28	12	18%	2021	11	4
86	cid45560	sid87	Dusk	Positive	eau de cologne	Man	M	19	237.498517	284.998221	7.0	Cash	New York	USA	50ml	56.960000	2022-12-22	12	18%	2022	12	4
87	cid21727	sid88	Aqua	Positive	eau de cologne	Man	F	150	270.576079	293.893286	7.0	Card	New York	USA	200ml	95.760000	2022-11-19	12	3%	2022	11	4
107	cid34036	sid108	Dusk	Negative	eau de cologne	Unisex	M	29	101.771992	61.063195	5.0	Card	Dubai	UAE	75ml	77.500000	2022-06-10	12	3%	2022	6	2
111	cid35105	sid112	Aqua	Positive	eau de cologne	Woman	F	150	711.256941	202.170649	1.0	Card	Sydney	Australia	75ml	72.730000	2020-09-22	12	2%	2020	9	3
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
199957	cid48141	sid199958	Ciel	Positive	eau de cologne	Woman	F	23	103.208482	72.245937	1.0	Card	Dubai	UAE	75ml	50.160000	2017-10-31	12	4%	2017	10	4
199967	cid21959	sid199968	Ciel	Positive	eau de cologne	Woman	M	31	99.236646	59.541988	6.0	Cash	New York	USA	30ml	30.400000	2019-07-25	12	3%	2019	7	3
199978	cid33148	sid199979	Dusk	Neutral	eau de cologne	Man	M	19	102.296007	81.836806	1.0	Card	Sydney	Australia	75ml	32.750000	2021-11-27	12	4%	2021	11	4
199992	cid46881	sid199993	Ciel	Positive	eau de cologne	Unisex	F	150	601.445942	125.087705	2.0	Card	Paris	France	50ml	50.120975	1900-01-03	12	4%	1900	1	1
199996	cid49485	sid199997	Belle	Neutral	eau de cologne	Man	M	22	137.624430	96.337101	2.0	Card	Dubai	UAE	200ml	49.950000	2020-12-20	12	4%	2020	12	4

9587 rows × 22 columns

In [ ]:

In [41]:
```python
Mean_rating=Sales_tran['Rating'][(Sales_tran.Rating=='1' ) | (Sales_tran.Rating=='2' )|(Sales_tran.Rating=='3') | (Sales_tran.Rating=='4' )|(Sales_tran.Rating=='5') ]
```

In [42]:
```python
Mean_rating=Mean_rating.astype('int64')
Mean_rating
```

Out[42]:
```
1         5
2         1
3         4
4         3
5         5
         ..
199989    1
199993    5
199995    1
199998    1
199999    4
Name: Rating, Length: 105682, dtype: int64
```

In [43]:
```python
Mean_rating=round(Mean_rating.mean())
Mean_rating
```

Out[43]:
```
2
```

In [44]:
```python
Sales_tran['Rating']=Sales_tran.Rating.replace(['12','10','6/12/2022','5/11/2021'],[Mean_rating,Mean_rating,Mean_rating,Mean_rating])
```

In [45]:
```python
Sales_tran.Rating.dtype
```

Out[45]:
```
dtype('O')
```

In [46]:
```python
Sales_tran['Rating']=Sales_tran.Rating.astype('int64')
Sales_tran.Rating.dtype
```

Out[46]:
```
dtype('int64')
```

In [47]:
```python
Sales_tran['Oil Content Percentage'].unique()
```

Out[47]:
```
array(['2%', '4%', '3%', '6%', '18%', '19%', '8%', '16%', '14%', '20%',
       '15%', '7%', '12%', '9%', '17%', '10%', '11%', '13%', '5%'],
      dtype=object)
```

In [ ]:

In [48]:
```python
Sales_tran['Sales']=Sales_tran['Price']*Sales_tran['Quantity']
```

In [49]:
```python
Sales_tran['Profit']=(((Sales_tran['Price']*Sales_tran['Quantity'])*(Sales_tran['Discount']-100))-(Sales_tran['Cost Price']))
```

In [ ]:

In [50]:
```python
pd.options.display.float_format = '{:.0f}'.format
```

In [51]:
```python
Product_by_gender=Sales_tran.groupby('Gender')['Sales','Profit'].sum().sort_values(by='Sales',ascending=False)
Product_by_gender
```

```
C:\Users\moayad\AppData\Local\Temp\ipykernel_13332\822453185.py:1: FutureWarning: Indexing with multiple keys (implicitly converted to a tuple of keys) will be deprecated, use a li
st instead.
  Product_by_gender=Sales_tran.groupby('Gender')['Sales','Profit'].sum().sort_values(by='Sales',ascending=False)
```

Out[51]:

Gender	Sales	Profit
M	121458201	-6063928567
F	85477246	-4245765244

In [52]:
```python
Sales_by_year=Sales_tran.groupby('Year')[['Sales']].sum()
Sales_by_year
```

Out[52]:

Year	Sales
1900	16057171
2017	31523785
2018	31439418
2019	32142599
2020	31968925
2021	31089419
2022	32714131

In [53]:
```python
Sales_by_Year_Qu=Sales_tran.groupby(['Year','Quarter'])[['Sales']].sum()
Sales_by_Year_Qu
```

Out[53]:

Year	Quarter	Sales
1900	1	16057171
2017	1	7736138
	2	7856809
	3	7791405
	4	8139434
2018	1	7750133
	2	7701122
	3	8132359
	4	7855805
2019	1	7917980
	2	8048538
	3	8188933
	4	7987148
2020	1	7784472
	2	8096550
	3	8006680
	4	8081223
2021	1	7583056
	2	7920480
	3	7847387
	4	7738495
2022	1	8181748
	2	8176786
	3	8139804
	4	8215794

In [54]:
```python
Rating_by_product=Sales_tran.groupby(['Product Name','Rating'])[['Quantity']].sum()
Rating_by_product
```

Out[54]:

Product Name	Rating	Quantity
Aqua	1	73902
	2	40316
	3	10574
	4	10907
	5	10346
Belle	1	73502
	2	39654
	3	10684
	4	11068
	5	10521
Ciel	1	75033
	2	40299
	3	10370
	4	10365
	5	10550
Dusk	1	74110
	2	38874
	3	10280
	4	10116
	5	10301
Elixir	1	74296
	2	38841
	3	10250
	4	10403
	5	10107

In [55]:
```python
Sales_by_Payment_Method=Sales_tran.groupby('Payment Method')[['Sales ID']].count().sort_values(by='Sales ID',ascending=False)
Sales_by_Payment_Method
```

Out[55]:

Payment Method	Sales ID
Card	92464
Cash	26413
Bank Transfer	13134

In [56]:
```python
Sales_Qty_Bottle=Sales_tran.groupby('Bottle Size')[['Quantity']].sum().sort_values(by='Quantity',ascending=False)
Sales_Qty_Bottle
```

Out[56]:

Bottle Size	Quantity
100ml	217596
50ml	216934
75ml	143922
30ml	74163
200ml	73054

In [57]:
```python
Sales_by_Category=Sales_tran.groupby('Category')[['Quantity']].sum().sort_values(by='Quantity',ascending=False)
Sales_by_Category
```

Out[57]:

Category	Quantity
eau de cologne	542794
eau de parfum	92636
eau de toilette	90239

In [58]:
```python
Sales_by_Category_product=Sales_tran.groupby(['Category','Product Name'])[['Quantity']].sum().sort_values(by='Quantity',ascending=False)
Sales_by_Category_product
```

Out[58]:

Category	Product Name	Quantity
eau de cologne	Ciel	109558
	Belle	109116
	Aqua	109087
	Elixir	107539
	Dusk	107494
eau de parfum	Ciel	18864
	Belle	18608
	Aqua	18494
	Elixir	18484
eau de toilette	Aqua	18464
	Ciel	18195
eau de parfum	Dusk	18186
eau de toilette	Dusk	18001
	Elixir	17874
	Belle	17705

In [59]:
```python
Sales_tran.columns
```

Out[59]:
```
Index(['Customer ID', 'Sales ID', 'Product Name', 'Sentiment', 'Category',
       'User Type', 'Gender', 'Age', 'Price', 'Cost Price', 'Quantity',
       'Payment Method', 'City', 'Country', 'Bottle Size', 'Discount',
       'Purchase Date', 'Rating', 'Oil Content Percentage', 'Year', 'Month',
       'Quarter', 'Sales', 'Profit'],
      dtype='object')
```

In [60]:
```python
Sales_tran.Year.max()
```

Out[60]:
```
2022
```

In [61]:
```python
Sales_tran.Year.min()
```

Out[61]:
```
1900
```

In [62]:
```python
Sales_tran.groupby('City')['Sales'].sum().sort_values(ascending=False)
```

Out[62]:
```
City
Paris      41514523
New York   41462598
London     41187019
Dubai      41147671
Tokyo      20823988
Sydney     20799650
Name: Sales, dtype: float64
```

In [63]:
```python
Total_Sales=round(Sales_tran.Sales.sum())
Total_Sales
```

Out[63]:
```
206935447
```

In [ ]:

Let's find the correlation for the sales, discount, quantity columns in the dataset

In [64]:
```python
Sales_tran.columns
```

Out[64]:
```
Index(['Customer ID', 'Sales ID', 'Product Name', 'Sentiment', 'Category',
       'User Type', 'Gender', 'Age', 'Price', 'Cost Price', 'Quantity',
       'Payment Method', 'City', 'Country', 'Bottle Size', 'Discount',
       'Purchase Date', 'Rating', 'Oil Content Percentage', 'Year', 'Month',
       'Quarter', 'Sales', 'Profit'],
      dtype='object')
```

In [65]:
```python
correlation_matrix=Sales_tran[['Sales','Discount']].corr()     # There are No correlation between Sales And Discount
correlation_matrix
```

Out[65]:

	Sales	Discount
Sales	1	0
Discount	0	1

In [66]:
```python
Sales_tran['Price'].corr(Sales_tran['Quantity'])     # there is weak negative correlation
```

Out[66]:
```
-0.006344334804217529
```

In [67]:
```python
df_eda=Sales_tran
```

In [68]:
```python
df_eda.to_csv('Perfume_EDA',index=True)
```