Problem Statement: Analyze and Provide Insights on Amazon Sales Report

Problem Description:

The provided dataset contains information about sales transaction on Amazon, including details such as order

ID, date, status, fulfilment method, sales channel, product category, size,quantity, amount, shipping details,

and more. The objective is to conduct a comprehensive analysis of the data and extract actionable insights to

support business decision-making

In [1]:
```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
import warnings as warnings
```

To Import Dataset Into Python

In [2]:
```
original_dataframe=pd.read_csv(r"E:\Study\Github\PowerBi-Project-4\Amazon Sale Report.csv")
```

In [3]:
```
# We will create a duplicate dataframe to avoid modification on original dataset
```

In [4]:
```
df=original_dataframe.copy()
```

In [5]:
```
df.head() #This are top 5 rows our data
```

Out[5]:

	index	Order ID	Date	Status	Fulfilment	Sales Channel	ship-service-level	Category	Size	Courier Status	...	currency	Amount	ship-ci
0	0	405-8078784-5731545	04-30-22	Cancelled	Merchant	Amazon.in	Standard	T-shirt	S	On the Way	...	INR	647.62	MUMB
1	1	171-9198151-1101146	04-30-22	Shipped - Delivered to Buyer	Merchant	Amazon.in	Standard	Shirt	3XL	Shipped	...	INR	406.00	BENGALUR
2	2	404-0687676-7273146	04-30-22	Shipped	Amazon	Amazon.in	Expedited	Shirt	XL	Shipped	...	INR	329.00	NAVI MUMB
3	3	403-9615377-8133951	04-30-22	Cancelled	Merchant	Amazon.in	Standard	Blazzer	L	On the Way	...	INR	753.33	PUDUCHERR
4	4	407-1069790-7240320	04-30-22	Shipped	Amazon	Amazon.in	Expedited	Trousers	3XL	Shipped	...	INR	574.00	CHENN

5 rows × 21 columns

In [6]:
```
df.tail() # This are bottom 5 rows of our dataframe
```

Out[6]:

	index	Order ID	Date	Status	Fulfilment	Sales Channel	ship-service-level	Category	Size	Courier Status	...	currency	Amount	shi
128971	128970	406-6001380-7673107	05-31-22	Shipped	Amazon	Amazon.in	Expedited	Shirt	XL	Shipped	...	INR	517.0	HYDER
128972	128971	402-9551604-7544318	05-31-22	Shipped	Amazon	Amazon.in	Expedited	T-shirt	M	Shipped	...	INR	999.0	GURU
128973	128972	407-9547469-3152358	05-31-22	Shipped	Amazon	Amazon.in	Expedited	Blazzer	XXL	Shipped	...	INR	690.0	HYDER
128974	128973	402-6184140-0545956	05-31-22	Shipped	Amazon	Amazon.in	Expedited	T-shirt	XS	Shipped	...	INR	1199.0	
128975	128974	408-7436540-8728312	05-31-22	Shipped	Amazon	Amazon.in	Expedited	T-shirt	S	Shipped	...	INR	696.0	

5 rows × 21 columns

Sanity chek of data

In [7]:
```
df.shape # This is shape of our data means we have 128976 rows and 21 columns
```

Out[7]: (128976, 21)

In [8]:
```
df.describe()
```

Out[8]:

	index	Qty	Amount	ship-postal-code	New	PendingS
count	128976.000000	128976.000000	121176.000000	128941.000000	0.0	0.0
mean	64486.130427	0.904401	648.562176	463945.677744	NaN	NaN
std	37232.897832	0.313368	281.185041	191458.488954	NaN	NaN
min	0.000000	0.000000	0.000000	110001.000000	NaN	NaN
25%	32242.750000	1.000000	449.000000	382421.000000	NaN	NaN
50%	64486.500000	1.000000	605.000000	500033.000000	NaN	NaN
75%	96730.250000	1.000000	788.000000	600024.000000	NaN	NaN
max	128974.000000	15.000000	5584.000000	989898.000000	NaN	NaN

In [9]:
```
df.info() # While inspecting the data we got to know that 'DATE' column is in object data type it shold be in d
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 128976 entries, 0 to 128975
Data columns (total 21 columns):
 #   Column              Non-Null Count   Dtype  
---  ------              --------------   -----  
 0   index               128976 non-null  int64  
 1   Order ID            128976 non-null  object 
 2   Date                128976 non-null  object 
 3   Status              128976 non-null  object 
 4   Fulfilment          128976 non-null  object 
 5   Sales Channel       128976 non-null  object 
 6   ship-service-level  128976 non-null  object 
 7   Category            128976 non-null  object 
 8   Size                128976 non-null  object 
 9   Courier Status      128976 non-null  object 
 10  Qty                 128976 non-null  int64  
 11  currency            121176 non-null  object 
 12  Amount              121176 non-null  float64
 13  ship-city           128941 non-null  object 
 14  ship-state          128941 non-null  object 
 15  ship-postal-code    128941 non-null  float64
 16  ship-country        128941 non-null  object 
 17  B2B                 128976 non-null  bool   
 18  fulfilled-by        39263 non-null   object 
 19  New                 0 non-null       float64
 20  PendingS            0 non-null       float64
dtypes: bool(1), float64(4), int64(2), object(14)
memory usage: 19.8+ MB
```

To change wrong data type in dataset

In [10]:
```
# Ignore warnings
warnings.filterwarnings('ignore')

# Convert dates to datetime format, handling different date formats
df['Date'] = pd.to_datetime(df['Date'], dayfirst=False, errors='coerce').fillna(pd.to_datetime(df['Date'], dayf

# Format the dates to 'dd/mm/yyyy'
df['Date'] = df['Date'].dt.strftime('%d/%m/%Y')

# Convert the formatted string dates back to datetime objects
df['Date'] = pd.to_datetime(df['Date'], format='%d/%m/%Y')
```

In [11]:
```
df.info() # Now 'Date' column is in  correct datetime format
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 128976 entries, 0 to 128975
Data columns (total 21 columns):
 #   Column              Non-Null Count   Dtype         
---  ------              --------------   -----         
 0   index               128976 non-null  int64         
 1   Order ID            128976 non-null  object        
 2   Date                128976 non-null  datetime64[ns]
 3   Status              128976 non-null  object        
 4   Fulfilment          128976 non-null  object        
 5   Sales Channel       128976 non-null  object        
 6   ship-service-level  128976 non-null  object        
 7   Category            128976 non-null  object        
 8   Size                128976 non-null  object        
 9   Courier Status      128976 non-null  object        
 10  Qty                 128976 non-null  int64         
 11  currency            121176 non-null  object        
 12  Amount              121176 non-null  float64       
 13  ship-city           128941 non-null  object        
 14  ship-state          128941 non-null  object        
 15  ship-postal-code    128941 non-null  float64       
 16  ship-country        128941 non-null  object        
 17  B2B                 128976 non-null  bool          
 18  fulfilled-by        39263 non-null   object        
 19  New                 0 non-null       float64       
 20  PendingS            0 non-null       float64       
dtypes: bool(1), datetime64[ns](1), float64(4), int64(2), object(13)
memory usage: 19.8+ MB
```

To find out null values

In [12]:
```
df.isnull()
```

Out[12]:

	index	Order ID	Date	Status	Fulfilment	Sales Channel	ship-service-level	Category	Size	Courier Status	...	currency	Amount	ship-city	ship-state
0	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False
1	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False
2	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False
3	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False
4	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
128971	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False
128972	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False
128973	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False
128974	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False
128975	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False

128976 rows × 21 columns

In [12]:
```
df.isnull().sum() # We have some null value in currency,amount,ship,city,ship
```

Out[12]:
```
index                   0
Order ID                0
Date                    0
Status                  0
Fulfilment              0
Sales Channel           0
ship-service-level      0
Category                0
Size                    0
Courier Status          0
Qty                     0
currency             7800
Amount               7800
ship-city              35
ship-state             35
ship-postal-code       35
ship-country           35
B2B                     0
fulfilled-by        89713
New                128976
PendingS           128976
dtype: int64
```

In [13]:
```
df.isnull().mean()*100
```

Out[13]:
```
index                 0.000000
Order ID              0.000000
Date                  0.000000
Status                0.000000
Fulfilment            0.000000
Sales Channel         0.000000
ship-service-level    0.000000
Category              0.000000
Size                  0.000000
Courier Status        0.000000
Qty                   0.000000
currency              6.047637
Amount                6.047637
ship-city             0.027137
ship-state            0.027137
ship-postal-code      0.027137
ship-country          0.027137
B2B                   0.000000
fulfilled-by         69.557902
New                 100.000000
PendingS            100.000000
dtype: float64
```

In [ ]:
```
```

In [14]:
```
df['fulfilled-by'].fillna(value='Other', inplace=True) # We will replace null values by 'Other'
df['currency'].fillna(value='INR', inplace=True) # We will replace null values by 'INR'
```

In [15]:
```
df.isnull().mean()*100
```

Out[15]:
```
index                 0.000000
Order ID              0.000000
Date                  0.000000
Status                0.000000
Fulfilment            0.000000
Sales Channel         0.000000
ship-service-level    0.000000
Category              0.000000
Size                  0.000000
Courier Status        0.000000
Qty                   0.000000
currency              0.000000
Amount                6.047637
ship-city             0.027137
ship-state            0.027137
ship-postal-code      0.027137
ship-country          0.027137
B2B                   0.000000
fulfilled-by          0.000000
New                 100.000000
PendingS            100.000000
dtype: float64
```

We will drop below two colums which are 100% blank.

In [16]:
```
df.drop(columns=['New'], inplace=True)
```

```
df.drop(columns=['PendingS'], inplace=True)
```

In [17]:
```
df_null_state = df[df['ship-state'].isnull()]
df_null_state
```

Out[17]:

	index	Order ID	Date	Status	Fulfilment	Sales Channel	ship-service-level	Category	Size	Courier Status	Qty	currency	Amount	s
1872	1871	404-0566904-4825137	2022-04-29	Shipped	Amazon	Amazon.in	Expedited	Trousers	L	Shipped	1	INR	493.0	
1873	1872	404-0566904-4825137	2022-04-29	Shipped	Amazon	Amazon.in	Expedited	Shirt	L	Shipped	1	INR	458.0	
2090	1871	404-0566904-4825137	2022-04-29	Shipped	Amazon	Amazon.in	Expedited	Trousers	L	Shipped	1	INR	493.0	
2091	1872	404-0566904-4825137	2022-04-29	Shipped	Amazon	Amazon.in	Expedited	Shirt	L	Shipped	1	INR	458.0	
8753	8752	406-4003386-8768363	2022-04-25	Shipped	Amazon	Amazon.in	Expedited	Shirt	M	Shipped	1	INR	432.0	
11216	11215	402-0107720-7057168	2022-04-23	Shipped	Amazon	Amazon.in	Expedited	T-shirt	S	Shipped	1	INR	654.0	
13253	13252	407-4532637-8415521	2022-04-22	Cancelled	Merchant	Amazon.in	Standard	Shirt	S	On the Way	0	INR	380.0	
15689	15688	404-9229894-8608305	2022-04-21	Shipped	Amazon	Amazon.in	Expedited	Shirt	M	Shipped	1	INR	442.0	
16788	16787	402-4919636-4333150	2022-04-20	Shipped - Delivered to Buyer	Merchant	Amazon.in	Standard	Shirt	3XL	Shipped	1	INR	376.0	
18352	18351	405-4927647-8064368	2022-04-19	Shipped	Amazon	Amazon.in	Expedited	T-shirt	XS	Shipped	1	INR	1112.0	
22931	22930	402-8628677-0457954	2022-04-16	Shipped - Returned to Seller	Merchant	Amazon.in	Standard	T-shirt	S	Shipped	1	INR	654.0	
24987	24986	406-4079063-8291520	2022-04-15	Cancelled	Amazon	Amazon.in	Expedited	Shirt	XXL	Unshipped	1	INR	399.0	
30380	30379	404-7506843-7913132	2022-04-12	Shipped - Delivered to Buyer	Merchant	Amazon.in	Standard	T-shirt	L	Shipped	1	INR	1299.0	
37964	37963	407-3064376-9158743	2022-04-08	Cancelled	Merchant	Amazon.in	Standard	Shirt	M	On the Way	0	INR	380.0	
37965	37964	407-3064376-9158743	2022-04-08	Cancelled	Merchant	Amazon.in	Standard	Shirt	S	On the Way	0	INR	380.0	
60987	60986	171-3257610-9237139	2022-05-22	Shipped - Delivered to Buyer	Merchant	Amazon.in	Standard	Shirt	L	Shipped	1	INR	376.0	
63041	63040	402-1049475-3611523	2022-05-20	Shipped	Amazon	Amazon.in	Expedited	Shirt	XXL	Shipped	1	INR	459.0	
73676	73675	405-1356730-8598722	2022-05-11	Shipped - Delivered to Buyer	Merchant	Amazon.in	Standard	T-shirt	XL	Shipped	1	INR	699.0	
73747	73746	407-4664354-2179554	2022-05-11	Shipped	Amazon	Amazon.in	Expedited	T-shirt	M	Shipped	1	INR	666.0	
73772	73771	406-7680604-5439529	2022-05-11	Shipped	Amazon	Amazon.in	Expedited	Shirt	4XL	Shipped	1	INR	869.0	
73773	73772	406-7680604-5439529	2022-05-11	Shipped	Amazon	Amazon.in	Expedited	Shirt	4XL	Shipped	1	INR	869.0	
73912	73911	405-3420451-5230744	2022-05-11	Cancelled	Amazon	Amazon.in	Expedited	T-shirt	M	Cancelled	0	INR	NaN	

80012	80011	171-4552355-0255565	2022-05-07	Shipped - Returned to Seller	Merchant	Amazon.in	Standard	T-shirt	XL	Shipped	1	INR	607.0
80093	80092	403-2857451-7335536	2022-05-07	Shipped	Amazon	Amazon.in	Expedited	Shirt	S	Shipped	1	INR	467.0
80456	80455	171-4691098-8489159	2022-05-06	Shipped - Delivered to Buyer	Merchant	Amazon.in	Standard	T-shirt	M	Shipped	1	INR	759.0
84009	84008	405-8605864-8021901	2022-05-04	Shipped - Returned to Seller	Merchant	Amazon.in	Standard	Shirt	XXL	Shipped	1	INR	368.0
84758	84757	402-8651786-0683548	2022-05-04	Shipped	Amazon	Amazon.in	Expedited	Trousers	XL	Shipped	1	INR	487.0
104202	104201	403-3190636-2013146	2022-06-18	Shipped - Delivered to Buyer	Merchant	Amazon.in	Standard	T-shirt	S	Shipped	1	INR	792.0
106499	106498	402-8673941-7883531	2022-06-16	Cancelled	Amazon	Amazon.in	Expedited	Shirt	XL	Cancelled	0	INR	NaN
124216	124215	405-0340492-2359532	2022-06-04	Cancelled	Amazon	Amazon.in	Expedited	Shirt	L	Unshipped	1	INR	491.0
124358	124357	405-5884153-9925116	2022-06-04	Shipped	Amazon	Amazon.in	Expedited	Shirt	L	Shipped	1	INR	486.0
124359	124358	405-5884153-9925116	2022-06-04	Shipped	Amazon	Amazon.in	Expedited	T-shirt	L	Shipped	1	INR	874.0
124360	124359	405-5884153-9925116	2022-06-04	Shipped	Amazon	Amazon.in	Expedited	T-shirt	L	Shipped	1	INR	832.0
125386	125385	403-5172380-9787567	2022-06-03	Shipped	Amazon	Amazon.in	Expedited	Shirt	L	Shipped	1	INR	376.0
126621	126620	403-4249038-6582716	2022-06-02	Shipped - Delivered to Buyer	Merchant	Amazon.in	Standard	Blazzer	M	Shipped	1	INR	735.0

From above we can see that 'ship-city','ship-state','ship-postal code' everywhere is null values so there is no other option to drop this rows. which is less that 1% of date

To complete case analysis we have to discard rows and colums which are blank.As this data is missing completely at random. approx 6% data is till missing we will drop missing values rows

In [19]:
```
df.dropna(inplace=True)
```

In [20]:
```
df.isnull().mean()*100
```

Out[20]:
```
index                0.0
Order ID             0.0
Date                 0.0
Status               0.0
Fulfilment           0.0
Sales Channel        0.0
ship-service-level   0.0
Category             0.0
Size                 0.0
Courier Status       0.0
Qty                  0.0
currency             0.0
Amount               0.0
ship-city            0.0
ship-state           0.0
ship-postal-code     0.0
ship-country         0.0
B2B                  0.0
fulfilled-by         0.0
dtype: float64
```

Now we can see that we do not have any null values here. Now we will find if we have any duplicate values

In [21]:
```
df.duplicated().sum() # we have 155 duplicate values
```

Out[21]:
```
155
```

In [22]:
```
df.drop_duplicates(inplace=True) #we have drop duplicat values
```

In [23]:
```
df.duplicated().sum()
```

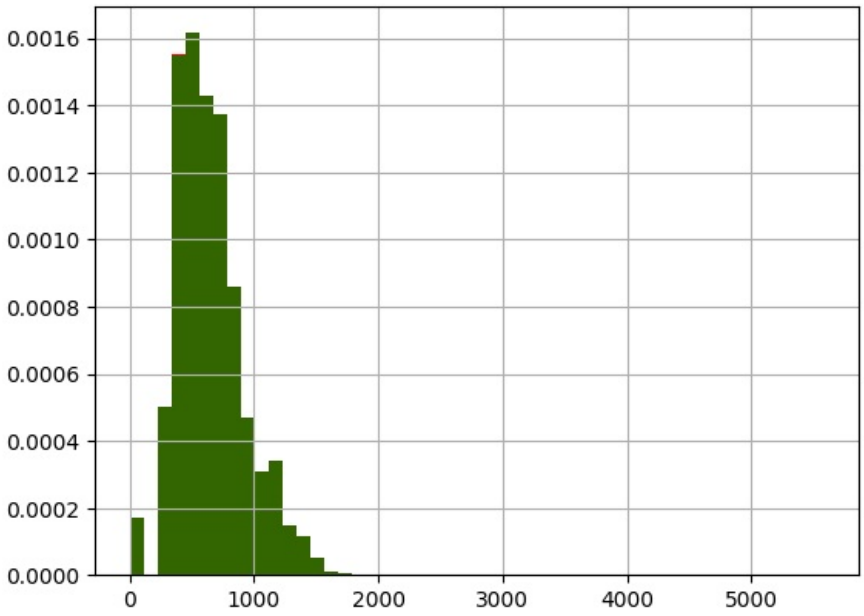
Out[23]:
```
0
```

In [30]:
```
df.shape
```

Out[30]:
```
(120988, 19)
```

In [26]:
```
fig=plt.figure()
ax=fig.add_subplot(111)
original_dataframe['Amount'].hist(bins=50, ax=ax,density=True,color='Red') # This is our original dataframe
df['Amount'].hist(bins=50, ax=ax,color='green',density=True,alpha=0.8) # this is our new dataframe
```

Out[26]:
```
<Axes: >
```



From above chart we can clearly see that the distribution is not changed before and aftter drop null values.It is same throughout. as we are not able to red histogram which was our original dataset

Now we are done with our cleaning part we will do visualization part in powerbi.To do that we will export this cleaned data into sql

In [41]:
```
import sqlalchemy as sa
import pandas as pd
```

In [42]:
```
import mysql.connector as sql
import pandas as pd
```

In [44]:
```
engine = create_engine('mysql+mysqlconnector://root:root@localhost:3306/Amazon') #To create a MySQL engine
```

In [45]:
```
df.to_sql('Amazon', engine, if_exists='replace', index=False)  # To save the DataFrame to a SQL table
```

Out[45]: 120988

In [ ]: