```
1 #a. Performing exploratory analysis on the data to understand the patterns
```

```
1 import pandas as pd
2 import numpy as np
3 import matplotlib.pyplot as plt
4 import matplotlib.gridspec as gridspec
5 import seaborn as sns
6
```

```
1 # data = constant_data.DataConstant
2 dataset = pd.read_excel("../Data/Womens Clothing Reviews Data.xlsx")
3 dataset
```

	Product ID	Category	Subcategory1	SubCategory2	Location	Channel	Customer Age	Review Title	Review Text
0	767	Initmates	Intimate	Intimates	Mumbai	Mobile	33	NaN	Absolutely wonderful - silky and sexy and comf...
1	1080	General	Dresses	Dresses	Bangalore	Mobile	34	NaN	Love this dress! it's sooo pretty. i happene...
2	1077	General	Dresses	Dresses	Gurgaon	Mobile	60	Some major design flaws	I had such high hopes for this dress and reall...
3	1049	General Petite	Bottoms	Pants	Chennai	Web	50	My favorite buy!	I love, love, love this jumpsuit. it's fun, fl...
4	847	General	Tops	Blouses	Bangalore	Web	47	Flattering shirt	This shirt is very flattering to all due to th...
...	...	...	...	...	...	...	...	...	...

```
1 dataset.shape
```

```
(23486, 11)
```

```
1 dataset.head()
```

	Product ID	Category	Subcategory1	SubCategory2	Location	Channel	Customer Age	Review Title	Review Text	Rati
0	767	Initmates	Intimate	Intimates	Mumbai	Mobile	33	NaN	Absolutely wonderful - silky and sexy and comf...	
1	1080	General	Dresses	Dresses	Bangalore	Mobile	34	NaN	Love this dress! it's	

```
1 #Make columns names in proper format by assigning proper variables
2 dataset.columns=['Product_ID', 'Category', 'SubCategory1', 'SubCategory2', 'Location',
3        'Channel', 'Customer_Age', 'Review_Title', 'Review_Text', 'Rating',
4        'Recommend_Flag']
```

```
1 dataset.head()
```

	Product_ID	Category	SubCategory1	SubCategory2	Location	Channel	Customer_Age	Review_Title	Rev
0	767	Initmates	Intimate	Intimates	Mumbai	Mobile	33	NaN	

```
1 #Check columns name
2 dataset.columns
```

```
Index(['Product_ID', 'Category', 'SubCategory1', 'SubCategory2', 'Location',
       'Channel', 'Customer_Age', 'Review_Title', 'Review_Text', 'Rating',
       'Recommend_Flag'],
      dtype='object')
```

```
1 # Find duplicate rows
2 duplicated_rows = dataset.duplicated()
3
4 # Count the number of duplicate rows
5 number_of_duplicate_rows = duplicated_rows.sum()
6
7 print("Number of duplicated rows:", number_of_duplicate_rows)
```

```
Number of duplicated rows: 3
```

```
1 #Remove duplicate data
2 dataset=dataset.drop_duplicates()
```

```
1 dataset.shape
```

```
(23483, 11)
```

```
1 # Validate with duplicate rows again
2 duplicated_rows = dataset.duplicated()
3
4 # Count the number of duplicate rows
5 number_of_duplicate_rows = duplicated_rows.sum()
6
7 print("Number of duplicated rows:", number_of_duplicate_rows)
```

```
Number of duplicated rows: 0
```

```
1 #Check null values in dataset
2 dataset.isnull().sum()
```

```
Product_ID           0
Category            14
SubCategory1        14
SubCategory2        14
Location             0
Channel              0
Customer_Age         0
Review_Title      3807
Review_Text        842
Rating               0
Recommend_Flag       0
dtype: int64
```

```
1 rows_missing_category = dataset[dataset['Category'].isnull()]
2 # Display rows with missing 'Category' as a DataFrame
3 rows_missing_category
```

	Product_ID	Category	SubCategory1	SubCategory2	Location	Channel	Customer_Age	Review_Title
9444	72	NaN	NaN	NaN	Chennai	Web	25	My favorite socks!!!
13767	492	NaN	NaN	NaN	Gurgaon	Web	23	So soft!
13768	492	NaN	NaN	NaN	Mumbai	Web	49	Wardrobe staple

```
1 dataset['Category'].fillna(dataset['Category'].mode()[0], inplace=True)
2 dataset['SubCategory1'].fillna(dataset['SubCategory1'].mode()[0], inplace=True)
3 dataset['SubCategory2'].fillna(dataset['SubCategory2'].mode()[0], inplace=True)
```

```
C:\Users\pholl\AppData\Local\Temp\ipykernel_8740\757871235.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-co
  dataset['Category'].fillna(dataset['Category'].mode()[0], inplace=True)
C:\Users\pholl\AppData\Local\Temp\ipykernel_8740\757871235.py:2: 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-co
  dataset['SubCategory1'].fillna(dataset['SubCategory1'].mode()[0], inplace=True)
C:\Users\pholl\AppData\Local\Temp\ipykernel_8740\757871235.py:3: 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-co
  dataset['SubCategory2'].fillna(dataset['SubCategory2'].mode()[0], inplace=True)
```

```
1 dataset.loc[dataset['Product_ID']==492]
```

	Product_ID	Category	SubCategory1	SubCategory2	Location	Channel	Customer_Age	Review_Title
13767	492	General	Tops	Dresses	Gurgaon	Web	23	So soft!

```
1 dataset.isnull().sum()
```

```
Product_ID           0
Category             0
SubCategory1         0
SubCategory2         0
Location             0
Channel              0
Customer_Age         0
Review_Title      3807
Review_Text        842
Rating               0
Recommend_Flag       0
dtype: int64
```

```
1 rows_missing_Review_Text  = dataset[dataset['Review_Text'].isnull()]
2 # Display rows with missing 'Review_Text ' as a DataFrame
3 rows_missing_Review_Text
```

	Product_ID	Category	SubCategory1	SubCategory2	Location	Channel	Customer_Age	Review_Title
92	861	General Petite	Tops	Knits	Gurgaon	Mobile	23	NaN
93	1081	General	Dresses	Dresses	Gurgaon	Mobile	31	NaN
98	1133	General	Jackets	Outerwear	Mumbai	Mobile	50	NaN
135	861	General	Tops	Knits	Gurgaon	Web	35	NaN

```
# remove rows where Review_Title and Review_Text have NaN values
dataset.dropna(subset=['Review_Title', 'Review_Text'], how='all', inplace=True)
```

```
C:\Users\pholl\AppData\Local\Temp\ipykernel_8740\566270713.py:2: 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-co
  dataset.dropna(subset=['Review_Title', 'Review_Text'], how='all', inplace=True)
```

```
dataset.isnull().sum()
```

```
Product_ID           0
Category             0
SubCategory1         0
SubCategory2         0
Location             0
Channel              0
Customer_Age         0
Review_Title      2966
Review_Text          1
Rating               0
Recommend_Flag       0
dtype: int64
```

```
rows_missing_Review_Text  = dataset[dataset['Review_Text'].isnull()]
# Display rows with missing 'Review_Text ' as a DataFrame
rows_missing_Review_Text
```

	Product_ID	Category	SubCategory1	SubCategory2	Location	Channel	Customer_Age	Review_Title
10220	1096	General	Dresses	Dresses	Chennai	Mobile	30	Such a beautiful

```
rows_missing_Review_Title  = dataset[dataset['Review_Title'].isnull()]
# Display rows with missing 'Review_Text ' as a DataFrame
rows_missing_Review_Title
```

	Product_ID	Category	SubCategory1	SubCategory2	Location	Channel	Customer_Age	Review_Title
0	767	Initmates	Intimate	Intimates	Mumbai	Mobile	33	NaN
1	1080	General	Dresses	Dresses	Bangalore	Mobile	34	NaN
11	1095	General Petite	Dresses	Dresses	Mumbai	Mobile	39	NaN
30	1060	General Petite	Bottoms	Pants	Mumbai	Web	33	NaN

```
import pandas as pd

# Replace "NaN" values in both columns with empty strings
dataset['Review_Text'].fillna('', inplace=True)
dataset['Review_Title'].fillna('', inplace=True)

```

```
# Merge the columns and store the result in a new column called "Merged_Review"
dataset['Merged_Review'] = dataset['Review_Title'] + ' ' + dataset['Review_Text']

# Drop the individual "Review_Text" and "Review_Title" columns if needed
dataset.drop(['Review_Text', 'Review_Title'], axis=1, inplace=True)
```

```
C:\Users\pholl\AppData\Local\Temp\ipykernel_8740\984468918.py:4: 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-co
  dataset['Review_Text'].fillna('', inplace=True)
C:\Users\pholl\AppData\Local\Temp\ipykernel_8740\984468918.py:5: 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-co
  dataset['Review_Title'].fillna('', inplace=True)
C:\Users\pholl\AppData\Local\Temp\ipykernel_8740\984468918.py:8: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-co
  dataset['Merged_Review'] = dataset['Review_Title'] + ' ' + dataset['Review_Text']
C:\Users\pholl\AppData\Local\Temp\ipykernel_8740\984468918.py:11: 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-co
  dataset.drop(['Review_Text', 'Review_Title'], axis=1, inplace=True)
```

```
dataset.isnull().sum()
```

```
Product_ID        0
Category          0
SubCategory1      0
SubCategory2      0
Location          0
Channel           0
Customer_Age      0
Rating            0
Recommend_Flag    0
Merged_Review     0
dtype: int64
```

```
dataset.loc[dataset['Product_ID']==767]
```

	Product_ID	Category	SubCategory1	SubCategory2	Location	Channel	Customer_Age	Rating	Recommen
0	767	Initmates	Intimate	Intimates	Mumbai	Mobile	33	4	

```
dataset.shape
```

```
(22642, 10)
```

```
#Check info
dataset.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 22642 entries, 0 to 23485
Data columns (total 10 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   Product_ID      22642 non-null  int64
 1   Category        22642 non-null  object
 2   SubCategory1    22642 non-null  object
 3   SubCategory2    22642 non-null  object
 4   Location        22642 non-null  object
 5   Channel         22642 non-null  object
 6   Customer_Age    22642 non-null  int64
 7   Rating          22642 non-null  int64
 8   Recommend_Flag  22642 non-null  int64
 9   Merged_Review   22642 non-null  object
dtypes: int64(4), object(6)
memory usage: 1.9+ MB
```

```
dataset.describe()
```

	Product_ID	Customer_Age	Rating	Recommend_Flag
count	22642.000000	22642.000000	22642.000000	22642.000000
mean	919.340164	43.279790	4.183597	0.818876
std	202.265815	12.327023	1.115751	0.385129
min	1.000000	18.000000	1.000000	0.000000
25%	861.000000	34.000000	4.000000	1.000000
50%	936.000000	41.000000	5.000000	1.000000
75%	1078.000000	52.000000	5.000000	1.000000
max	1205.000000	99.000000	5.000000	1.000000

```
1 dataset.columns
```

```
Index(['Product_ID', 'Category', 'SubCategory1', 'SubCategory2', 'Location',
       'Channel', 'Customer_Age', 'Rating', 'Recommend_Flag', 'Merged_Review'],
      dtype='object')
```

```
1 # Value counts for 'Category'
2 category_counts = dataset['Category'].value_counts()
3 print("Value Counts for Category:")
4 print(category_counts)
```

```
Value Counts for Category:
General           13379
General Petite     7837
Initmates          1426
Name: Category, dtype: int64
```

```
1 # Value counts for 'SubCategory1'
2 subcategory1_counts = dataset['SubCategory1'].value_counts()
3 print("Value Counts for SubCategory1:")
4 print(subcategory1_counts)
```

```
Value Counts for SubCategory1:
Tops        10061
Dresses      6146
Bottoms      3662
Intimate     1653
Jackets      1002
Trend         118
Name: SubCategory1, dtype: int64
```

```
1 # Value counts for 'SubCategory2'
2 subcategory2_counts = dataset['SubCategory2'].value_counts()
3 print("Value Counts for SubCategory2:")
4 print(subcategory2_counts)
```

```
Value Counts for SubCategory2:
Dresses           6159
Knits             4626
Blouses           2983
Sweaters          1380
Pants             1350
Jeans             1104
Fine gauge        1059
Skirts             903
Jackets            683
Lounge             669
Swim               332
Outerwear          319
Shorts             304
Sleep              214
Legwear            158
Intimates          147
Layering           132
Trend              118
Casual bottoms       1
Chemises             1
Name: SubCategory2, dtype: int64
```

```
1 # Value counts for 'Location'
2 Location_counts = dataset['Location'].value_counts()
```

```
print("Value Counts for Location:")
print(Location_counts)
```

```
Value Counts for Location:
Gurgaon      8491
Mumbai       6859
Bangalore    5050
Chennai      2242
Name: Location, dtype: int64
```

```
# Value counts for 'Channel'
Channel_counts = dataset['Channel'].value_counts()
print("Value Counts for Channel:")
print(Channel_counts)
```

```
Value Counts for Channel:
Web       13096
Mobile     9546
Name: Channel, dtype: int64
```

```
# Value counts for 'Customer_Age'
Customer_Age_counts = dataset['Customer_Age'].value_counts()
print("Value Counts for Customer_Age:")
print(Customer_Age_counts)
```

```
Value Counts for Customer_Age:
39    1226
35     851
36     801
34     766
38     751
      ...
93       2
90       2
86       2
99       2
92       1
Name: Customer_Age, Length: 77, dtype: int64
```

```
# Value counts for 'Recommend_Flag'
Recommend_Flag_counts = dataset['Recommend_Flag'].value_counts()
print("Value Counts for Recommend_Flag:")
print(Recommend_Flag_counts)
```

```
Value Counts for Recommend_Flag:
1    18541
0     4101
Name: Recommend_Flag, dtype: int64
```

```
# Value counts for 'Rating'
Rating_counts = dataset['Rating'].value_counts()
print("Value Counts for Rating:")
print(Rating_counts)
```

```
Value Counts for Rating:
5    12541
4     4908
3     2823
2     1549
1      821
Name: Rating, dtype: int64
```

```
#Check the correlation between numerical features.
dataset.corr()
```

```
C:\Users\pholl\AppData\Local\Temp\ipykernel_8740\3955366760.py:2: FutureWarning: The def
  dataset.corr()
```

	Product_ID	Customer_Age	Rating	Recommend_Flag
Product_ID	1.000000	0.017646	-0.018425	-0.014855
Customer_Age	0.017646	1.000000	0.029926	0.034184
Rating	-0.018425	0.029926	1.000000	0.792570
Recommend_Flag	-0.014855	0.034184	0.792570	1.000000

```
# Save the cleaned dataset to a new CSV file
dataset.to_excel('Womens Clothing Reviews Data New.xlsx', index=False)
```