

flipkart-product-analysis-final

September 19, 2024

```
[1]: #importing necessary libraries
import numpy as np
import pandas as pd
```

```
[2]: df = pd.read_csv('/Users/randhirkumar/Downloads/flipkart_com-e-commerce_sample.
˓→csv')
df.head()
```

```
[2]:          uniq_id      crawl_timestamp \
0  c2d766ca982eca8304150849735ffef9  2016-03-25 22:59:23 +0000
1  7f7036a6d550aaa89d34c77bd39a5e48  2016-03-25 22:59:23 +0000
2  f449ec65dc0c041b6ae5e6a32717d01b  2016-03-25 22:59:23 +0000
3  0973b37acd0c664e3de26e97e5571454  2016-03-25 22:59:23 +0000
4  bc940ea42ee6bef5ac7cea3fb5cfbee7  2016-03-25 22:59:23 +0000

          product_url \
0  http://www.flipkart.com/alisha-solid-women-s-c...
1  http://www.flipkart.com/fabhomedecor-fabric-do...
2  http://www.flipkart.com/aw-bellies/p/itmeh4grg...
3  http://www.flipkart.com/alisha-solid-women-s-c...
4  http://www.flipkart.com/sicons-all-purpose-arn...

          product_name \
0  Alisha Solid Women's Cycling Shorts
1  FabHomeDecor Fabric Double Sofa Bed
2  AW Bellies
3  Alisha Solid Women's Cycling Shorts
4  Sicons All Purpose Arnica Dog Shampoo

          product_category_tree          pid \
0  ["Clothing >> Women's Clothing >> Lingerie, Sl...  SRTEH2FF9KEDEFGF
1  ["Furniture >> Living Room Furniture >> Sofa B...  SBEEH3QGU7MFYJFY
2  ["Footwear >> Women's Footwear >> Ballerinas >...  SHOEH4GRSUBJGZXE
3  ["Clothing >> Women's Clothing >> Lingerie, Sl...  SRTEH2F6HUZMQ6SJ
4  ["Pet Supplies >> Grooming >> Skin & Coat Care...  PSOEH3ZYDMSYARJ5

  retail_price  discounted_price \
```

```

0      999.0      379.0
1     32157.0     22646.0
2      999.0      499.0
3      699.0      267.0
4      220.0      210.0

                           image  is_FK_Advantage_product  \
0  ["http://img5a.flixcart.com/image/short/u/4/a/...
1  ["http://img6a.flixcart.com/image/sofa-bed/j/f...
2  ["http://img5a.flixcart.com/image/shoe/7/z/z/r...
3  ["http://img5a.flixcart.com/image/short/6/2/h/...
4  ["http://img5a.flixcart.com/image/pet-shampoo/...

                           description      product_rating  \
0  Key Features of Alisha Solid Women's Cycling S...  No rating available
1  FabHomeDecor Fabric Double Sofa Bed (Finish Co...  No rating available
2  Key Features of AW Bellies Sandals Wedges Heel...  No rating available
3  Key Features of Alisha Solid Women's Cycling S...  No rating available
4  Specifications of Sicons All Purpose Arnica Do...  No rating available

      overall_rating      brand  \
0  No rating available      Alisha
1  No rating available  FabHomeDecor
2  No rating available        AW
3  No rating available      Alisha
4  No rating available      Sicons

      product_specifications
0  {"product_specification":>[{"key":>"Number of ...
1  {"product_specification":>[{"key":>"Installati...
2  {"product_specification":>[{"key":>"Ideal For"...
3  {"product_specification":>[{"key":>"Number of ...
4  {"product_specification":>[{"key":>"Pet Type",...

```

[3]: *#copy the dataset to prevent loss of original dataset*
`cdf=df.copy()`

[4]: `cdf.sample(5)`

[4]:

	uniq_id	crawl_timestamp
12970	e698627a87d584134c46f64022f746be	2016-01-01 19:37:22 +0000
5557	7fc429e79a2583a1a5dff35f5d114071	2015-12-01 06:13:00 +0000
14606	c5cbfb14618003448ec51fb053762abd	2015-12-12 11:46:53 +0000
11017	3430af01c7a055b054f40cd4a2213111	2015-12-31 09:19:31 +0000
3304	d420ca42226a38112f950076a6448709	2016-04-03 12:45:50 +0000

product_url \

```
12970 http://www.flipkart.com/radiant-bay-breathaki...
5557 http://www.flipkart.com/jadoo-collections-allo...
14606 http://www.flipkart.com/my-foot-women-flats/p...
11017 http://www.flipkart.com/speedwav-single-pipe-c...
3304 http://www.flipkart.com/rpb-casual-sleeveless-...
```

```
product_name \
12970 Radiant Bay Breathhtaking Black and White Yello...
5557 Jadoo Collections Alloy Necklace
14606 My Foot Women Flats
11017 Speedwav Single Pipe Car Air Pressure -Nissan ...
3304 RPB Casual Sleeveless Solid Girl's Top
```

```
product_category_tree pid \
12970 ["Jewellery >> Rings"] RNGEDAPWYKJJRSZY
5557 ["Jewellery >> Necklaces & Chains >> Necklaces"] NKCEAX7HHE5HZCHG
14606 ["Footwear >> Women's Footwear >> Flats"] SNDEAN424G99YS2N
11017 ["Automotive >> Accessories & Spare parts >> S... VHREB3ZYGQYX5KHF
3304 ["Clothing >> Kids' Clothing >> Girls Wear >> ... TOPEGZAECT2CMGNZ
```

```
retail_price discounted_price \
12970 47390.0 37912.0
5557 599.0 299.0
14606 1699.0 1016.0
11017 1500.0 873.0
3304 999.0 449.0
```

```
image \
12970 ["http://img6a.flixcart.com/image/ring/w/j/z/r...
5557 ["http://img5a.flixcart.com/image/necklace-cha...
14606 ["http://img5a.flixcart.com/image/sandal/q/s/n...
11017 ["http://img6a.flixcart.com/image/vehicle-horn...
3304 ["http://img5a.flixcart.com/image/top/k/g/m/to...
```

```
is_FK_Advantage_product \
12970 False
5557 False
14606 True
11017 False
3304 False
```

```
description product_rating \
12970 Radiant Bay Breathhtaking Black and White Yello... No rating available
5557 Jadoo Collections Alloy Necklace - Buy Jadoo C... No rating available
14606 Flipkart.com: Buy My Foot Women Flats only for... No rating available
11017 Speedwav Single Pipe Car Air Pressure -Nissan ... No rating available
3304 Key Features of RPB Casual Sleeveless Solid Gi... No rating available
```

```

          overall_rating           brand \
12970  No rating available      Radiant Bay
5557   No rating available  Jadoo Collections
14606  No rating available           NaN
11017  No rating available      Speedwav
3304   No rating available        RPB

          product_specifications
12970  {"product_specification":>[{"key":>"Collection...
5557   {"product_specification":>[{"key":>"Brand", "v...
14606  {"product_specification":>[{"key":>"Ideal For"...
11017  {"product_specification":>[{"key":>"Brand", "v...
3304   {"product_specification":>[{"key":>"Sleeve", "...
```

Data Cleaning Process

[6]: `cdf.describe()`

```

[6]:      retail_price  discounted_price
count    19922.000000    19922.000000
mean     2979.206104    1973.401767
std      9009.639341    7333.586040
min      35.000000     35.000000
25%     666.000000    350.000000
50%     1040.000000   550.000000
75%     1999.000000   999.000000
max     571230.000000  571230.000000
```

[7]: *#Checking Number Of Null Rows In Each Columns*
`cdf.isnull().sum()`

```

[7]: uniq_id                  0
crawl_timestamp              0
product_url                  0
product_name                  0
product_category_tree         0
pid                          0
retail_price                 78
discounted_price              78
image                         3
is_FK_Advantage_product      0
description                  2
product_rating                0
overall_rating                0
brand                         5864
product_specifications        14
```

```
dtype: int64
```

```
[8]: cdf.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 20000 entries, 0 to 19999
Data columns (total 15 columns):
 #   Column           Non-Null Count  Dtype  
 ---  --  
 0   uniq_id          20000 non-null   object  
 1   crawl_timestamp  20000 non-null   object  
 2   product_url      20000 non-null   object  
 3   product_name     20000 non-null   object  
 4   product_category_tree  20000 non-null   object  
 5   pid               20000 non-null   object  
 6   retail_price     19922 non-null   float64 
 7   discounted_price 19922 non-null   float64 
 8   image             19997 non-null   object  
 9   is_FK_Advantage_product  20000 non-null   bool    
 10  description       19998 non-null   object  
 11  product_rating    20000 non-null   object  
 12  overall_rating    20000 non-null   object  
 13  brand             14136 non-null   object  
 14  product_specifications 19986 non-null   object  
dtypes: bool(1), float64(2), object(12)
memory usage: 2.2+ MB
```

```
[9]: #For Better Analysis we extract year,month,day From crawl_timestamp And Add ↵Them As New Columns
# As Datatype of crawl_timestamp is object to extract desire output we need to ↵convert into datetime
from datetime import datetime
cdf['crawl_timestamp']=pd.to_datetime(cdf['crawl_timestamp'])
cdf['Year']=cdf['crawl_timestamp'].dt.year
cdf['Month']=cdf['crawl_timestamp'].dt.month
cdf['Days']=cdf['crawl_timestamp'].dt.day
```

```
[10]: # As we have extract year, month,days from crawl_timestamp we can now delete ↵crawl_timestamp column to reduce dimension
cdf.drop(columns='crawl_timestamp',inplace=True)
```

```
[11]: # product_url,image both columns are not useful so,its better to drop them from ↵dataset
cdf.drop(columns=['product_url','image'],inplace=True)
```

```
[12]: # Let's extract the different category from product_category_tree such as ↵main_category,secondary_category,tertiary_category,quaternary_category
```

```

cdf['main_category']=cdf['product_category_tree'].apply(lambda x: x.
    ↪split('>>')[0][2::])
def secondary_category (value):
    try:
        return value.split(">>")[1]
    except IndexError:
        return "None"
def tertiary_category(value):
    try:
        return value.split('>>')[2]
    except IndexError:
        return "None"
def quaternary_category(value):
    try:
        return value.split('>>')[3][:len(value.split('>>')[3])-2]
    except IndexError:
        return "None"

cdf['secondary_category']=cdf['product_category_tree'].apply(secondary_category)
cdf['tertiary_category']=cdf['product_category_tree'].apply(tertiary_category)
cdf['quaternary_category']=cdf['product_category_tree'].
    ↪apply(quaternary_category)

```

[13]: *'''As we have done with extraction of
 ↪main_category,secondary_category,tertiary_catgeory,quaternary_catgeory from
 ↪product_catgeory_tree
 we can delete product_catgeory_tree from dataset'''*

```

cdf.drop(columns='product_category_tree',inplace=True)

```

[14]: *# product_specification and description are not useful for our analysis so we
 ↪delete these columns from dataset to reduce dimension*

```

cdf.drop(columns=['product_specifications','description'],inplace=True)

```

[15]: `cdf.isnull().sum()`

uniq_id	0
product_name	0
pid	0
retail_price	78
discounted_price	78
is_FK_Advantage_product	0
product_rating	0
overall_rating	0
brand	5864
Year	0
Month	0
Days	0

```

main_category          0
secondary_category     0
tertiary_category      0
quaternary_category    0
dtype: int64

```

Now Let's deal with missing value of dataset

To fill missing value in any column instead of direct filling we use segmented imputation for better result and futher analysis

```

[18]: #filling missing value in retail_price by median which is robust to data
# For segment imputation we use product_name to decide the retail_price of ↴
#missing value
# If in case missing value remains we use direct median for fill then
cdf['retail_price']=cdf.groupby(['product_name'])['retail_price'].transform(lambda x: x.fillna(x.median()))
cdf['retail_price']=cdf['retail_price'].transform(lambda x: x.fillna(x.median()))

```

```

[19]: #filling missing value in discounted_price by median which is robust to data
# For segment imputation we use product_name to decide the discount_price of ↴
#missing value
# If in case missing value remains we use direct median for fill them
cdf['discounted_price']=cdf.groupby(['product_name'])['discounted_price'].transform(lambda x: x.fillna(x.median()))
cdf['discounted_price']=cdf['discounted_price'].transform(lambda x: x.fillna(x.median()))

```

```

[20]: # Replacing the No rating available to 0 for rating analysis
cdf['product_rating']=cdf['product_rating'].replace('No rating available',0)
cdf['overall_rating']=cdf['overall_rating'].replace('No rating available',0)

```

```

[21]: # Filling missing value of brand column using segmented imputation
cdf['brand']=cdf.groupby(['main_category'])['brand'].transform(lambda x: x.fillna(x.mode()[0] if not x.mode().empty else "unknown"))

```

```

[22]: cdf.isnull().sum()

```

```

[22]: uniq_id          0
product_name         0
pid                 0
retail_price         0
discounted_price     0
is_FK_Advantage_product 0
product_rating       0
overall_rating       0

```

```

brand          0
Year           0
Month          0
Days           0
main_category  0
secondary_category 0
tertiary_category 0
quaternary_category 0
dtype: int64

```

Now in our dataset there is no missing value so let's do deep dive into analysis

[24]: `cdf.sample(5)`

```

[24]:          uniq_id \
18898  eeb7239b49f9be65c1b55859710eef46
8401   635993d34a562b33208fc052398c2b42
456    d6883e16c9863a8d09a42c0d06af7ae
12932  5cb9567a5491f3aaee65f158f4b34282
8465   21d2f40c61d848898ab8db0e54e1a3fe

                           product_name          pid \
18898          Himmlisch AAD0487 Car Armrest  VAREJGRCZU6PJUCQ
8401           Go Hooked Plant Container Set  PCSDYQEWHCZUKEZA
456    WearYourShine by PCJ The Adriyel Gold Diamond ...  RNGECEVYCZNGJBHY
12932  DailyObjects Back Cover for Apple iPad 2, 3, 4  ACCE6GFCQGEX4CDA
8465    Rega IT Hp Pavilion DV2507TX DV2508TU 6 Cell 6...  ACCDXFPYVJCMFGET

          retail_price  discounted_price  is_FK_Advantage_product  product_rating \
18898        1799.0          999.0                False                  0
8401        800.0          500.0                False                  0
456        28963.0         28963.0               False                  0
12932       1999.0          1685.0               False                  0
8465       2699.0          1299.0               False                  0

          overall_rating      brand  Year  Month  Days \
18898            0  Himmlisch  2016     6     9
8401            0  Go Hooked  2016     1     7
456             0  WearYourShine by PCJ  2016     1     6
12932            0  DailyObjects  2016     1     1
8465            0  Rega IT  2016     1     7

          main_category  secondary_category \
18898      Automotive  Car & Bike Accessories
8401       Tools & Hardware  Tools
456        Jewellery  Rings"]
12932  Mobiles & Accessories  Tablet Accessories

```

8465	Computers	Laptop Accessories
	tertiary_category	quaternary_category
18898	Car Interior & Exterior	Car Interio
8401	Gardening Tools	Plant Containers & Set
456	None	None
12932	Cases & Covers	DailyObjects Cases & Covers
8465	Batteries	Rega IT Batteries

[25]: `cdf.describe()`

	retail_price	discounted_price	Year	Month	\
count	20000.000000	20000.000000	20000.000000	20000.000000	
mean	2973.631450	1969.162350	2015.484250	7.435550	
std	8996.711609	7322.079548	0.499764	4.875165	
min	35.000000	35.000000	2015.000000	1.000000	
25%	669.000000	350.000000	2015.000000	2.000000	
50%	1039.500000	550.000000	2015.000000	12.000000	
75%	1999.000000	999.000000	2016.000000	12.000000	
max	571230.000000	571230.000000	2016.000000	12.000000	
	Days				
count	20000.000000				
mean	11.084300				
std	10.423998				
min	1.000000				
25%	1.000000				
50%	7.000000				
75%	20.000000				
max	31.000000				

price analysis:

[27]: `cdf['discount_percentage']=((cdf['retail_price']-cdf['discounted_price'])/cdf['retail_price'])*100`

[28]: `cdf['retail_price'].value_counts()`

999.0	1287
499.0	697
599.0	641
799.0	583
1299.0	558
	...
1013.0	1
1555.0	1

```
878.0      1
4398.0      1
5795.0      1
Name: count, Length: 2248, dtype: int64
```

```
[29]: cdf['retail_price'].min()
cdf['retail_price'].max()
# In our dataset retail_price has two many outliers because our value of
# retail_price vary from 35 to 600000 approx
# For that lets create a new column where we distinguish the product into three
# category daily_use_products,occasional_product,luxury_products and analyze
# according to it
def product_category(value):
    if value>=0 and value<5000:
        return "Daily Use Product"
    elif value>=5000 and value<=50000:
        return "Occasional Product"
    else:
        return "Luxury Product"
cdf['product_category']=cdf['retail_price'].apply(lambda x: product_category(x))
```

```
[30]: cdf.head()
```

	uniq_id	product_name
0	c2d766ca982eca8304150849735ffef9	Alisha Solid Women's Cycling Shorts
1	7f7036a6d550aaa89d34c77bd39a5e48	FabHomeDecor Fabric Double Sofa Bed
2	f449ec65dc041b6ae5e6a32717d01b	AW Bellies
3	0973b37acd0c664e3de26e97e5571454	Alisha Solid Women's Cycling Shorts
4	bc940ea42ee6bef5ac7cea3fb5cfbee7	Sicons All Purpose Arnica Dog Shampoo

	pid	retail_price	discounted_price	is_FK_Advantage_product
0	SRTEH2FF9KEDEFGF	999.0	379.0	False
1	SBEEH3QGU7MFYJFY	32157.0	22646.0	False
2	SHOEH4GRSUBJGZX	999.0	499.0	False
3	SRTEH2F6HUZMQ6SJ	699.0	267.0	False
4	PSOEH3ZYDMSYARJ5	220.0	210.0	False

	product_rating	overall_rating	brand	Year	Month	Days
0	0	0	Alisha	2016	3	25
1	0	0	FabHomeDecor	2016	3	25
2	0	0	AW	2016	3	25
3	0	0	Alisha	2016	3	25
4	0	0	Sicons	2016	3	25

	main_category	secondary_category	tertiary_category
0	Clothing	Women's Clothing	Lingerie, Sleep & Swimwear
1	Furniture	Living Room Furniture	Sofa Beds & Futons

```

2      Footwear      Women's Footwear      Ballerinas
3      Clothing      Women's Clothing      Lingerie, Sleep & Swimwear
4  Pet Supplies      Grooming      Skin & Coat Care

                           quaternary_category  discount_percentage \
0                           Short          62.062062
1  FabHomeDecor Fabric Double Sofa Bed (Finish C...          29.576764
2                           AW Bellies          50.050050
3                           Short          61.802575
4                           Shampo          4.545455

  product_category
0  Daily Use Product
1  Occasional Product
2  Daily Use Product
3  Daily Use Product
4  Daily Use Product

```

```
[31]: cdf.to_csv('/Users/randhirkumar/Downloads/filpkart_dasboard.csv', index=False)
```

```
[32]: #extract all three type of product from dataset and futher analyze them
daily_use_product=cdf[cdf['product_category']=='Daily Use Product']
pd.DataFrame(daily_use_product)
daily_use_product.head()
```

```

[32]:          uniq_id \
0  c2d766ca982eca8304150849735ffef9
2  f449ec65dc041b6ae5e6a32717d01b
3  0973b37acd0c664e3de26e97e5571454
4  bc940ea42ee6bef5ac7cea3fb5cfbee7
5  c2a17313954882c1dba461863e98adf2

                           product_name          pid \
0  Alisha Solid Women's Cycling Shorts  SRTEH2FF9KEDEFGF
2                           AW Bellies  SHOEH4GRSUBJGZXE
3  Alisha Solid Women's Cycling Shorts  SRTEH2F6HUZMQ6SJ
4  Sicons All Purpose Arnica Dog Shampoo  PSOEH3ZYDMSYARJ5
5  Eternal Gandhi Super Series Crystal Paper Weig...  PWTEB7H2E4KCYUE3

  retail_price  discounted_price  is_FK_Advantage_product  product_rating \
0      999.0          379.0            False                  0
2      999.0          499.0            False                  0
3      699.0          267.0            False                  0
4      220.0          210.0            False                  0
5      430.0          430.0            False                  0

  overall_rating      brand  Year  Month  Days \

```

```

0          0      Alisha  2016      3    25
2          0      AW     2016      3    25
3          0      Alisha  2016      3    25
4          0      Sicons  2016      3    25
5          0  Eternal Gandhi  2016      3    25

                           main_category  secondary_category \
0                         Clothing  Women's Clothing
2                         Footwear  Women's Footwear
3                         Clothing  Women's Clothing
4                         Pet Supplies  Grooming
5  Eternal Gandhi Super Series Crystal Paper Weig...  None

                           tertiary_category  quaternary_category  discount_percentage \
0  Lingerie, Sleep & Swimwear                  Short  62.062062
2          Ballerinas                  AW Bellies  50.050050
3  Lingerie, Sleep & Swimwear                  Short  61.802575
4          Skin & Coat Care                 Shampo  4.545455
5                  None                  None  0.000000

  product_category
0  Daily Use Product
2  Daily Use Product
3  Daily Use Product
4  Daily Use Product
5  Daily Use Product

```

```
[33]: occasional_product=cdf[cdf['product_category']=='Occasional Product']
pd.DataFrame(occasional_product)
occasional_product.head()
```

```

[33]:          uniq_id \
1  7f7036a6d550aaa89d34c77bd39a5e48
7  8542703ca9e6ebdf6d742638dfb1f2ca
16 849ab05698081a374215d0b7d18047d0
19 7ee3e337474a29b38cfe1f5d4020d633
42 f5b22eaa5f79e67209612997f4aa215d

                           product_name  pid \
1  FabHomeDecor Fabric Double Sofa Bed  SBEEH3QGU7MFYJFY
7  FabHomeDecor Fabric Double Sofa Bed  SBEEH3QGYGHFUEXN
16  FabHomeDecor Fabric Double Sofa Bed  SBEEH3QGAYAEPRCG
19  FabHomeDecor Fabric Double Sofa Bed  SBEEH3QGWRGG3J6Q
42  Himmilisch ST381 Magnetic Sun Shade For Maruti ...  SUDEJFBRVYEUZUDH

  retail_price  discounted_price  is_FK_Advantage_product  product_rating \
1        32157.0          22646.0                  False                  0

```

```

7      32157.0      22646.0      False      0
16     32157.0      22646.0      False      0
19     32157.0      22646.0      False      0
42      6999.0      1899.0      False      0

      overall_rating      brand      Year      Month      Days      main_category  \
1                  0  FabHomeDecor  2016       3      25  Furniture
7                  0  FabHomeDecor  2016       3      25  Furniture
16                 0  FabHomeDecor  2016       3      25  Furniture
19                 0  FabHomeDecor  2016       3      25  Furniture
42                 0   Himmlisch  2016       5      26  Automotive

      secondary_category      tertiary_category  \
1  Living Room Furniture  Sofa Beds & Futons
7  Living Room Furniture  Sofa Beds & Futons
16  Living Room Furniture  Sofa Beds & Futons
19  Living Room Furniture  Sofa Beds & Futons
42  Accessories & Spare parts  Car Interior & Exterior

      quaternary_category      discount_percentage  \
1  FabHomeDecor Fabric Double Sofa Bed (Finish C...  29.576764
7  FabHomeDecor Fabric Double Sofa Bed (Finish C...  29.576764
16  FabHomeDecor Fabric Double Sofa Bed (Finish C...  29.576764
19  FabHomeDecor Fabric Double Sofa Bed (Finish C...  29.576764
42                           Car Interio  72.867553

      product_category
1  Occasional Product
7  Occasional Product
16  Occasional Product
19  Occasional Product
42  Occasional Product

```

```
[34]: luxury_product=cdf[cdf['product_category']=='Luxury Product']
pd.DataFrame(luxury_product)
luxury_product.head()
```

```
[34]:                               uniq_id  \
108  710ed5f2393a4b9e8823aa0029f71f93
116  329c5f4d7aced63e1ce3e88f41d5e7e6
134  08452abdadb3db1e686b94a9c52fc7b6
7915 c4b045288524a8770c760ed2bbca2ed5
9782 022a7f08d051e2cfca917ef0a6d3a19a

                               product_name      pid  \
108  TAG Heuer CAU1116.BA0858 Formula 1 Analog Watc...  WATEA6UGFGKZKDJC
116  Breitling AB011010/BB08 131S Chronomat 44 Anal...  WATE9DX9E4YSRJSY
```

```

134      Cartier W6701005 Analog Watch - For Boys, Men WATEAK8SE2RUK7GM
7915  Audeze Lcd2 In Bamboo, High Quality Planar Mag... ACCEEV5UV5NV4PKZ
9782  The Curve Full Sleeve Solid Men's Quilted Jacket JCKEAEA7REDBGJYV

      retail_price  discounted_price  is_FK_Advantage_product  product_rating \
108      107750.0      107750.0                  False            0
116      571230.0      571230.0                  False            0
134      201000.0      201000.0                  False            0
7915      116292.0      116292.0                  False            0
9782      69999.0      48999.0                  False            0

      overall_rating  brand  Year  Month  Days \
108          0  Italian Fashion  2015     12      4
116          0  Italian Fashion  2015     12      4
134          0  Italian Fashion  2015     12      4
7915          0  Audeze  2016      3     11
9782          0  Regular  2016      1      7

      main_category  secondary_category \
108      Watches      Wrist Watches
116      Watches      Wrist Watches
134      Watches      Wrist Watches
7915  Mobiles & Accessories  Mobile Accessories
9782      Clothing      Men's Clothing

      tertiary_category  quaternary_category  discount_percentage \
108  TAG Heuer Wrist Watches"]          None      0.000000
116  Breitling Wrist Watches"]          None      0.000000
134  Cartier Wrist Watches"]          None      0.000000
7915      Headphones  Audeze Headphone      0.000000
9782  Winter & Seasonal Wear          Jacket      30.000429

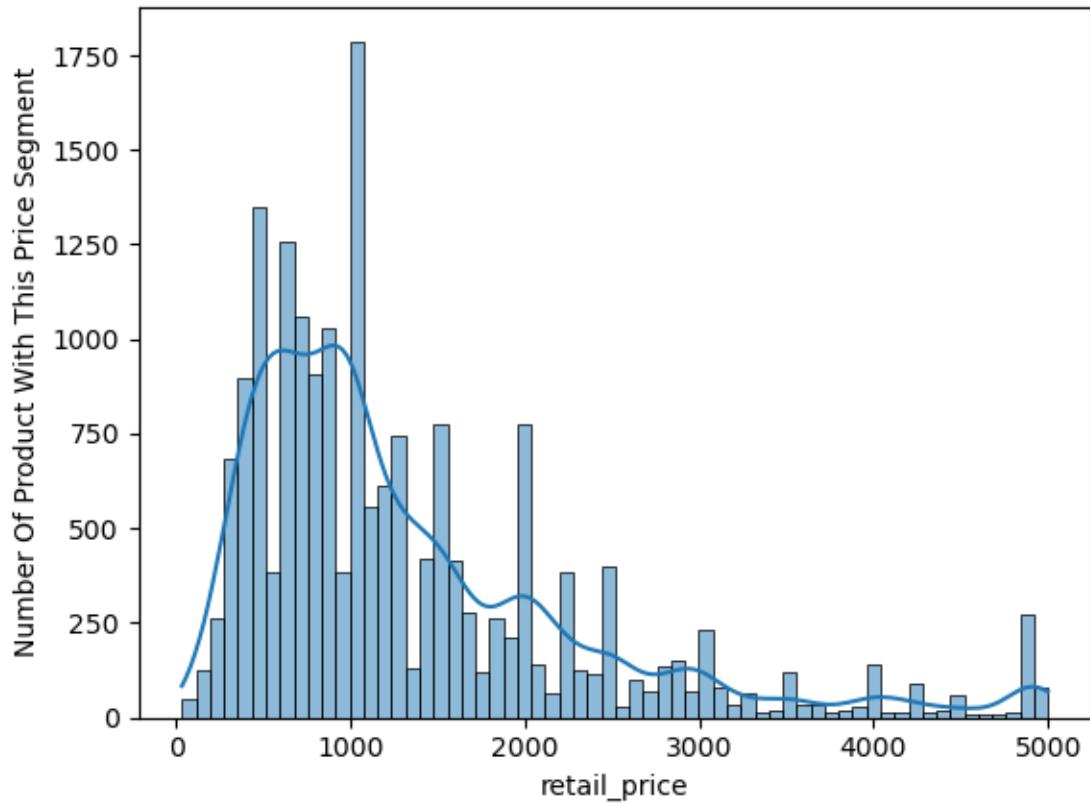
      product_category
108      Luxury Product
116      Luxury Product
134      Luxury Product
7915      Luxury Product
9782      Luxury Product

```

```
[35]: #importing visualization tools
from matplotlib import pyplot as plt
%matplotlib inline
import seaborn as sns
cdf.replace([np.inf, -np.inf], np.nan, inplace=True)
```

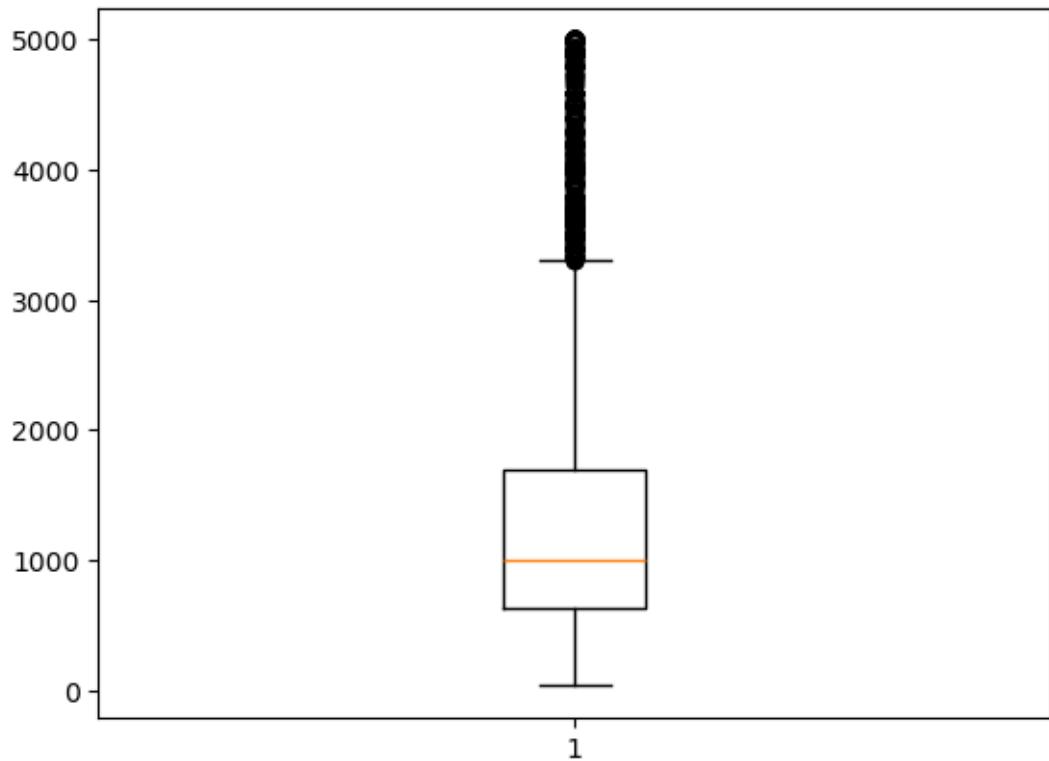
```
[36]: plt.ylabel('Number Of Product With This Price Segment')
sns.histplot(daily_use_product.retail_price,kde=True)
```

```
[36]: <Axes: xlabel='retail_price', ylabel='Number Of Product With This Price Segment'>
```



```
[37]: plt.boxplot(daily_use_product.retail_price)
```

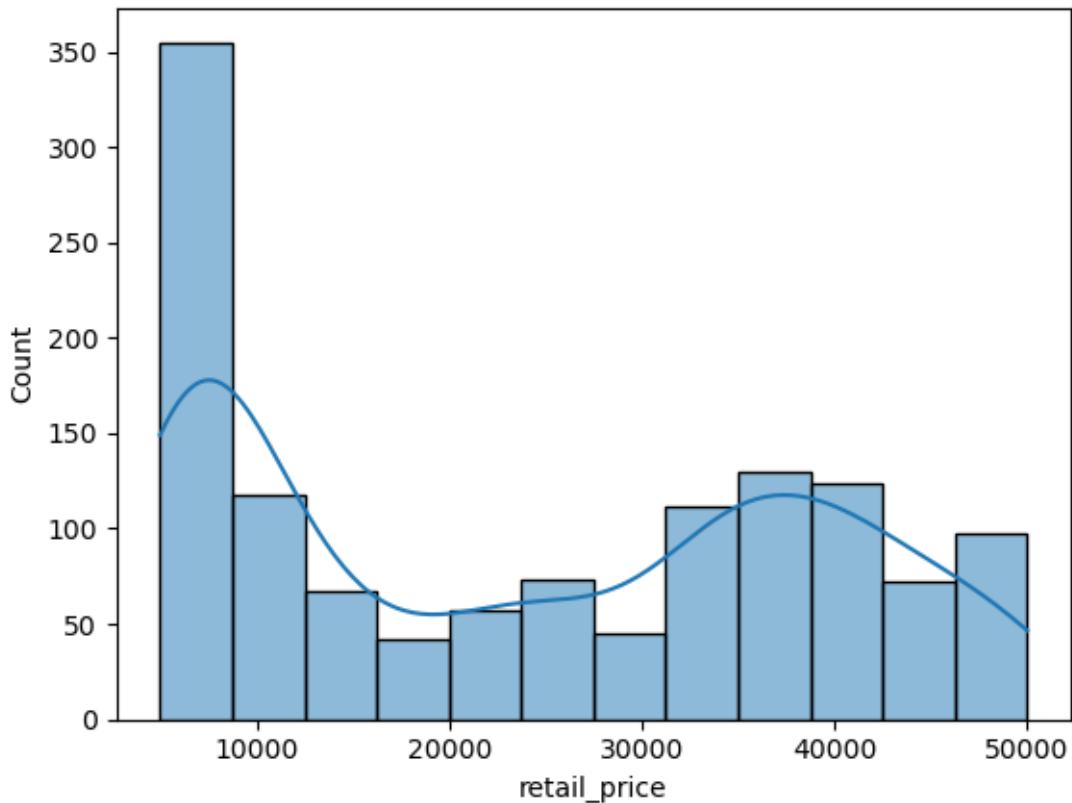
```
[37]: {'whiskers': [<matplotlib.lines.Line2D at 0x138bdb5c0>,
 <matplotlib.lines.Line2D at 0x138bdb890>],
 'caps': [<matplotlib.lines.Line2D at 0x138bdbaa0>,
 <matplotlib.lines.Line2D at 0x138bdbda0>],
 'boxes': [<matplotlib.lines.Line2D at 0x138bdb350>],
 'medians': [<matplotlib.lines.Line2D at 0x138c14110>],
 'fliers': [<matplotlib.lines.Line2D at 0x138c14410>],
 'means': []}
```



The Daily Use Product category is right skewed (positive skewed)

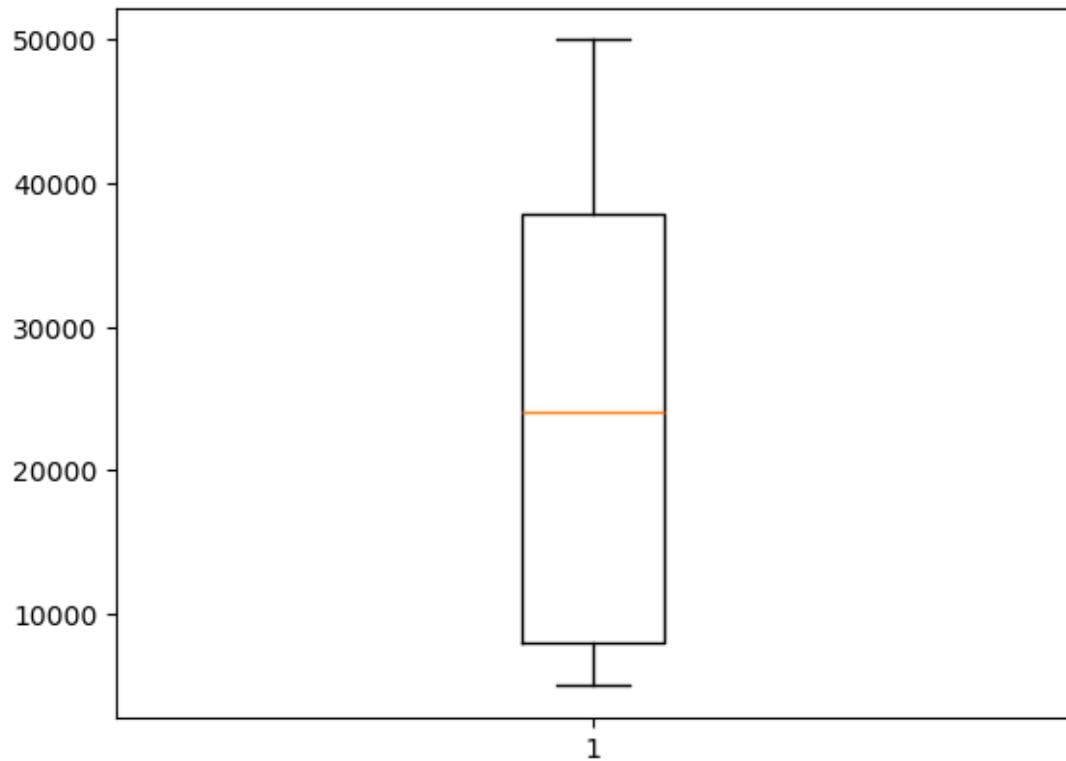
```
[39]: sns.histplot(occasional_product.retail_price,kde=True)
```

```
[39]: <Axes: xlabel='retail_price', ylabel='Count'>
```



```
[40]: plt.boxplot(occasional_product.retail_price)
```

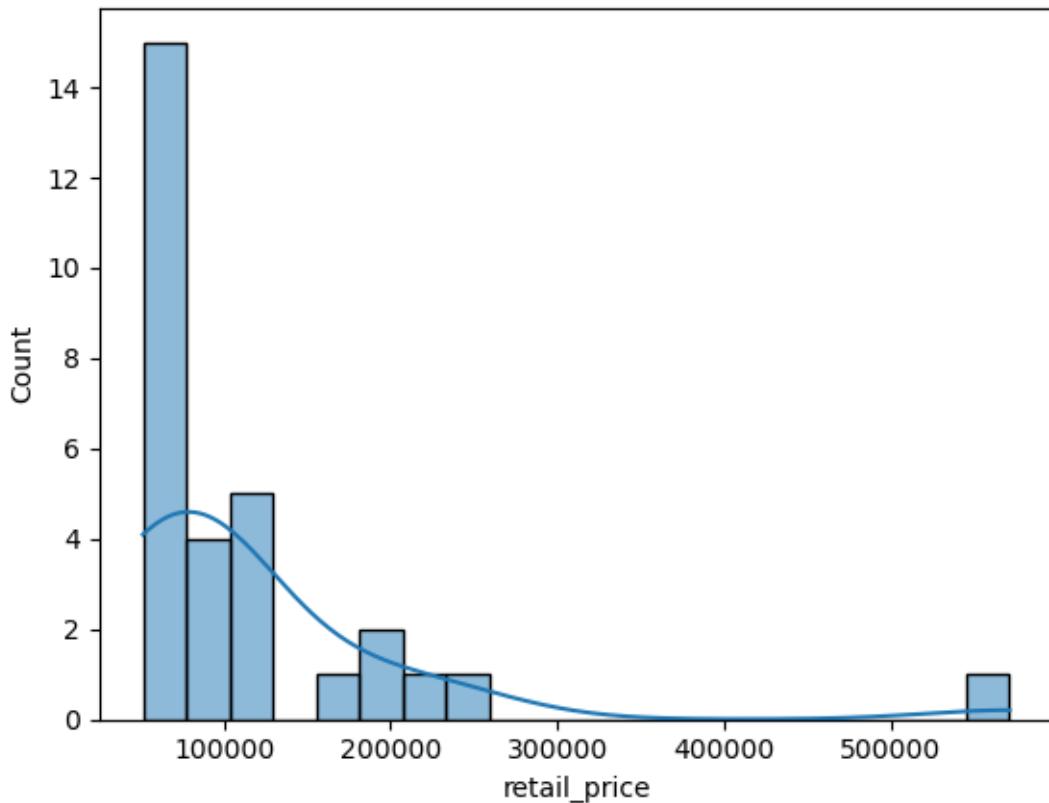
```
[40]: {'whiskers': [<matplotlib.lines.Line2D at 0x138d07590>,
   <matplotlib.lines.Line2D at 0x138d07860>],
  'caps': [<matplotlib.lines.Line2D at 0x138d07b60>,
   <matplotlib.lines.Line2D at 0x138d07e30>],
  'boxes': [<matplotlib.lines.Line2D at 0x138d07350>],
  'medians': [<matplotlib.lines.Line2D at 0x138d54170>],
  'fliers': [<matplotlib.lines.Line2D at 0x138d54470>],
  'means': []}]
```



The occasional product category is not proper normal distribution but yeah it is normal distributed

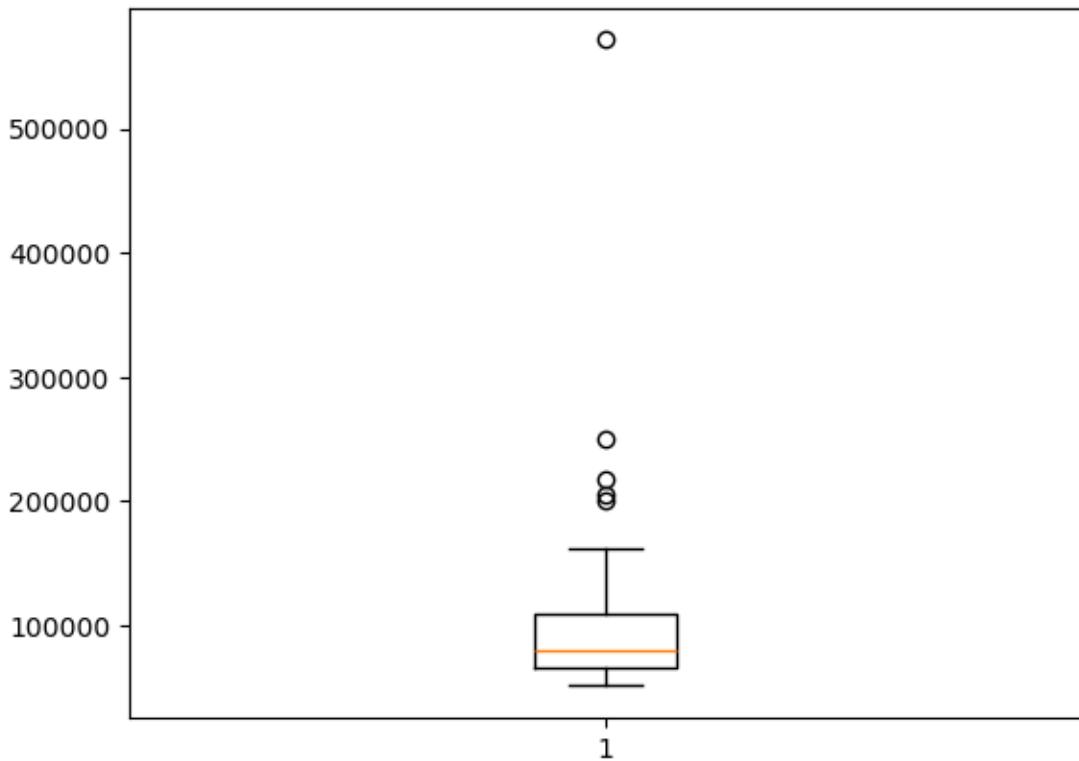
```
[42]: sns.histplot(luxury_product.retail_price,kde=True)
```

```
[42]: <Axes: xlabel='retail_price', ylabel='Count'>
```



```
[43]: plt.boxplot(luxury_product.retail_price)
```

```
[43]: {'whiskers': [matplotlib.lines.Line2D at 0x138def2f0>,
 <matplotlib.lines.Line2D at 0x138def5f0>],
 'caps': [matplotlib.lines.Line2D at 0x138def8f0>,
 <matplotlib.lines.Line2D at 0x138defbc0>],
 'boxes': [matplotlib.lines.Line2D at 0x138def050>],
 'medians': [matplotlib.lines.Line2D at 0x138defe60>],
 'fliers': [matplotlib.lines.Line2D at 0x138e201d0>],
 'means': []}
```



The luxury product category is positive skewed

Daily Use Product
Analysis:

PRICE ANALYSIS:

[47]: `daily_use_product.describe()`

```
[47]:      retail_price  discounted_price      Year      Month \
count    18678.000000    18678.000000  18678.000000  18678.000000
mean    1331.189046     724.164846  2015.464182    7.669129
std     996.265200     596.074020   0.498729    4.816287
min     35.000000     35.000000  2015.000000   1.000000
25%    630.000000     349.000000  2015.000000   3.000000
50%    999.000000     499.000000  2015.000000  12.000000
75%   1699.000000     899.000000  2016.000000  12.000000
max   4999.000000    4999.000000  2016.000000  12.000000

                           Days  discount_percentage
count    18678.000000          18678.000000
mean     11.370115          41.431596
```

```
std      10.454388      23.548481
min      1.000000      0.000000
25%     1.000000      23.094688
50%     7.000000      47.534430
75%    20.000000      60.040027
max     31.000000      95.273818
```

```
[48]: #How average and median price vary for different products category
average_retail_price_for_daily_use_products=daily_use_product.
    ↪groupby(['main_category'])['retail_price'].apply(lambda x: x.mean())
median_retail_price_for_daily_use_products=daily_use_product.
    ↪groupby(['main_category'])['retail_price'].apply(lambda x: x.median())

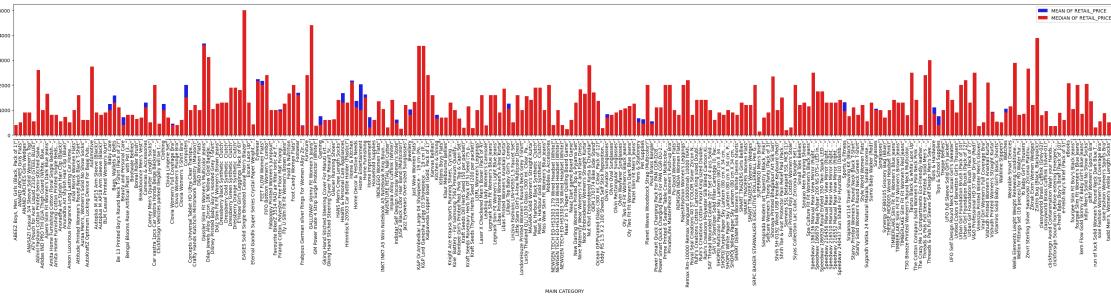
average_retail_price_for_daily_use_products
```

```
[48]: main_category
ABEEZ Boys, Men, Girls (Black, Pack of 1)"]           399.0
ANAND ARCHIES Girls Flats"]                          499.0
ANAND ARCHIES Girls Wedges"]                         899.0
ANASAZI Casual 3/4 Sleeve Solid Women's Top"]        899.0
ATV Pouch for Acer Liquid Z330 (STEEL BLUE)"]       549.0
...
run of luck Solid Women's Round Neck Dark Blue T..."] 299.0
soie Fashion Women's Full Coverage Bra"]                540.0
soie Fashion Women's Sports Bra"]                      880.0
tadd Men's, Women's Ankle Length Socks"]                499.0
xy decor Cotton Sofa Cover (white Pack of 6)"]        1699.0
Name: retail_price, Length: 246, dtype: float64
```

```
[49]: plt.figure(figsize=(42,5))
plt.xticks(rotation=90)

sns.barplot(y=average_retail_price_for_daily_use_products.
    ↪values,x=average_retail_price_for_daily_use_products.
    ↪index,color='blue',label='MEAN OF RETAIL PRICE')
sns.barplot(y=median_retail_price_for_daily_use_products.
    ↪values,x=median_retail_price_for_daily_use_products.
    ↪index,color='red',label="MEDIAN OF RETAIL PRICE")
plt.xlabel("MAIN CATEGORY")
plt.legend(loc="upper right")
```

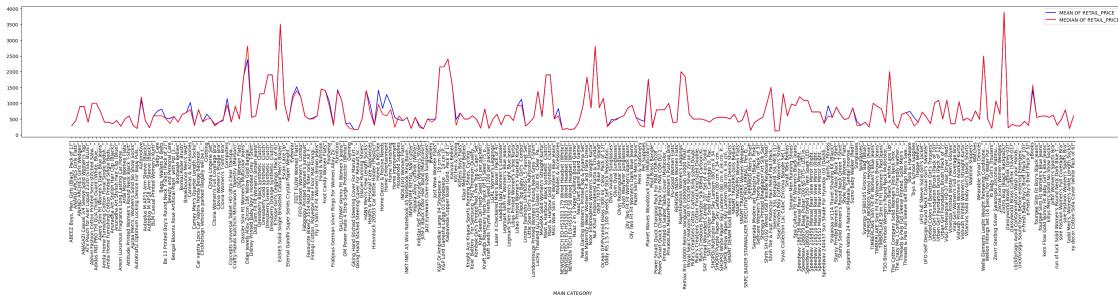
```
[49]: <matplotlib.legend.Legend at 0x138dc2720>
```



```
[50]: # how discount vary by different main category in daily use product category
average_discounted_price_for_daily_use_products=daily_use_product.
    ↪groupby(['main_category'])['discounted_price'].apply(lambda x: x.mean())
median_discounted_price_for_daily_use_products=daily_use_product.
    ↪groupby(['main_category'])['discounted_price'].apply(lambda x: x.median())
```

```
[51]: pd.option_context('mode.use_inf_as_na', True)
plt.figure(figsize=(42,5))
plt.xticks(rotation=90)
sns.lineplot(y=average_discounted_price_for_daily_use_products.
    ↪values,x=average_discounted_price_for_daily_use_products.
    ↪index,color='blue',label='MEAN OF RETAIL PRICE')
sns.lineplot(y=median_discounted_price_for_daily_use_products.
    ↪values,x=median_discounted_price_for_daily_use_products.
    ↪index,color='red',label="MEDIAN OF RETAIL PRICE")
plt.xlabel("MAIN CATEGORY")
plt.legend(loc="upper right")
```

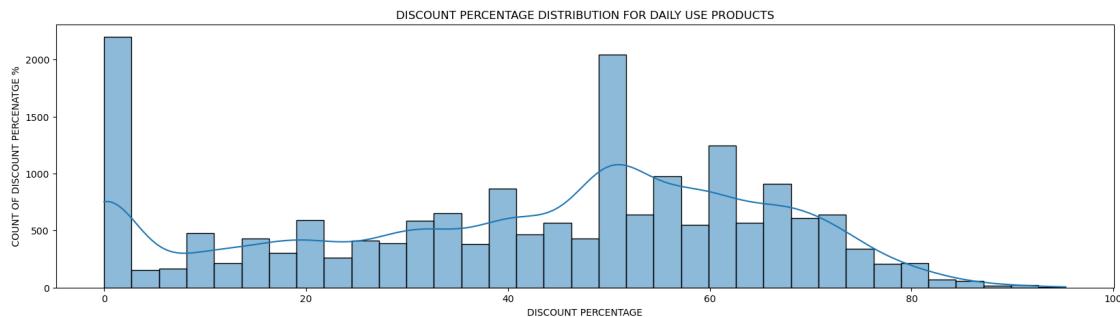
```
[51]: <matplotlib.legend.Legend at 0x13ab20170>
```



```
[52]: # how discount percentage vary for different product in daily use product
plt.figure(figsize=(20,5))
sns.histplot(daily_use_product['discount_percentage'],kde=True)
plt.title('DISCOUNT PERCENTAGE DISTRIBUTION FOR DAILY USE PRODUCTS')
```

```
plt.xlabel("DISCOUNT PERCENTAGE")
plt.ylabel(" COUNT OF DISCOUNT PERCENTAGE % ")
```

[52]: Text(0, 0.5, ' COUNT OF DISCOUNT PERCENTAGE % ')



[53]: *# top 30 products with average highest discount in daily use products*
`discount=daily_use_product.groupby(['main_category'])['discount_percentage'].
 ↪mean()
 top_30_discounted_daily_use_product=discount.nlargest(30)
 top_30_discounted_daily_use_product`

main_category	
NEWGEN TECH EO-HS3303 218 Wired Headset (White)"]	80.080080
Himmlisch 20503 Car Bottle Holder (Plastic)"]	76.982294
Threads & Pals Full Sleeve Self Design Men's Swe..."]	76.584390
Shonaya Printed Bhagalpuri Art Silk Sari"]	76.282051
Yo Baby Girl's Trousers"]	76.000000
THERISE MD0005 Wired Headset (Pink)"]	75.075075
GALLOWAY skinny Fit Women's Jeans"]	75.017049
Little Stars Girl's A-line Multicolor Dress"]	74.984326
Speedwav 240437 Sun Shade For Hyundai i10 (Dashb..."]	74.428571
killys Men's Solid No Show Socks"]	73.094027
Threads & Pals Full Sleeve Printed, Solid Men's ..."]	72.977481
classyworld Brass Cufflink (Grey-02)"]	72.116603
Gking Hand Stiched Steering Cover For Maruti Ert..."]	71.953255
Gking Hand Stiched Steering Cover For Renault Sc..."]	71.953255
Autoplus M AP18 Arm Sleeve (Black)"]	71.839800
The Cotton Company Solid Women's Polo Neck Pink ..."]	71.479628
fourgee Slim Fit Women's Blue Jeans"]	70.600000
Sunglasses"]	67.788106
AutoKraftZ Optimum Locking Device For Bajaj Puls..."]	66.777963
SMART TRADERS Women Flats"]	66.722269
INDILEGO Women Flats"]	66.716705
classyworld Brass Cufflink (Silver-01)"]	65.693431
xy decor Cotton Sofa Cover (white Pack of 6)"]	64.743967

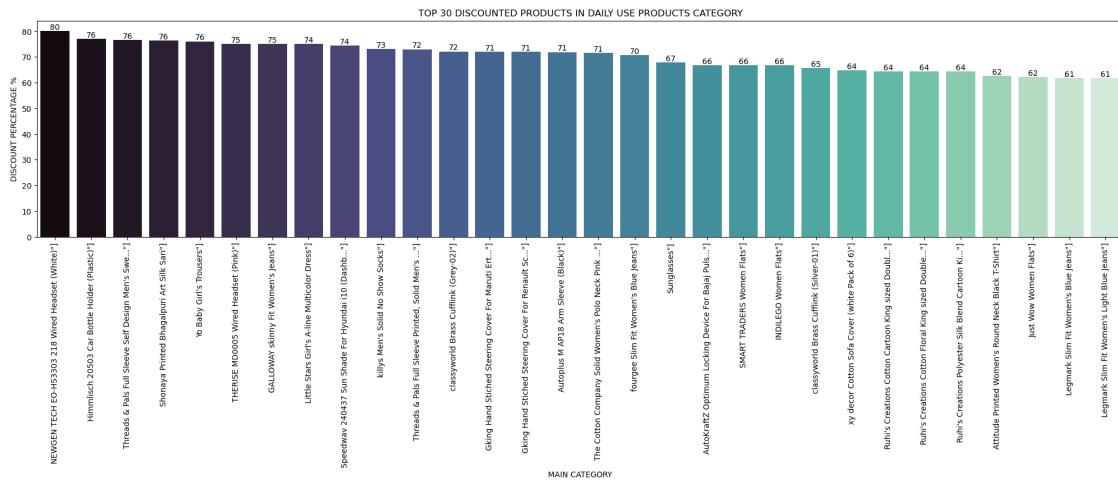
```
Ruhi's Creations Cotton Cartoon King sized Double...] 64.331665
Ruhi's Creations Cotton Floral King sized Double...] 64.331665
Ruhi's Creations Polyester Silk Blend Cartoon Ki...] 64.331665
Attitude Printed Women's Round Neck Black T-Shirt...] 62.539087
Just Wow Women Flats"] 62.196970
Legmark Slim Fit Women's Blue Jeans"] 61.726079
Legmark Slim Fit Women's Light Blue Jeans"] 61.726079
Name: discount_percentage, dtype: float64
```

```
[54]: plt.figure(figsize=(25,5))
plt.xticks(rotation=90)
ax=sns.barplot(x=top_30_discounted_daily_use_product.
    ↪index,y=top_30_discounted_daily_use_product.values,palette='mako')
plt.xlabel("MAIN CATEGORY")
plt.ylabel("DISCOUNT PERCENTAGE %")
plt.title('TOP 30 DISCOUNTED PRODUCTS IN DAILY USE PRODUCTS CATEGORY')
#ADDING DATA LABEL TO BAR
for conatiner in ax.containers:
    ax.bar_label(conatiner,fmt='%d')
```

```
/var/folders/vc/4h3d9s6d639f63qzbhd805b80000gn/T/ipykernel_1158/2175370551.py:3:
FutureWarning:
```

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `x` variable to `hue` and set `legend=False` for the same effect.

```
ax=sns.barplot(x=top_30_discounted_daily_use_product.index,y=top_30_discounted
_daily_use_product.values,palette='mako')
```



```
[55]: #Are there any significant price differences between FK Advantage products and others?
fk_advantage_product_in_daily_use_products=daily_use_product.
groupby(["is_FK_Advantage_product"])['retail_price'].describe()
fk_advantage_product_in_daily_use_products
```

```
[55]:          count      mean       std    min    25%    50%  \
is_FK_Advantage_product
False          17932.0  1334.98037  1004.110959  35.0  619.0  999.0
True           746.0   1240.05496   779.405160  79.0  750.0  999.0

                           75%      max
is_FK_Advantage_product
False          1699.0  4999.0
True           1599.0  4995.0
```

```
[56]: #Are there any significant price differences between FK Advantage products and others?
discounted_fk_advantage_product_in_daily_use_products=daily_use_product.
groupby(["is_FK_Advantage_product"])['discounted_price'].describe()
discounted_fk_advantage_product_in_daily_use_products
```

```
[56]:          count      mean       std    min    25%    50%  \
is_FK_Advantage_product
False          17932.0  719.336215  594.522441  35.0  349.0  499.0
True           746.0   840.233244  621.443299  79.0  419.0  714.0

                           75%      max
is_FK_Advantage_product
False          899.0  4999.0
True           999.0  4995.0
```

From analysis we can conclude that those products which are fk_advantage_products they have heigher discounted price,so from customer pov we should do shopping whenever there is fk_advantagee availabel.

RATING ANALYSIS:

```
[59]: #correlection between retail_price,discounted_price,discount_percent
corr_daily_use_products=daily_use_product[['retail_price','discounted_price','discount_percent']]
corr()
corr_daily_use_products
```

```
[59]:          retail_price  discounted_price  discount_percentage
retail_price          1.000000          0.779443          0.236543
discounted_price        0.779443          1.000000         -0.289154
discount_percentage     0.236543         -0.289154          1.000000
```

```
[60]: sns.heatmap(corr_daily_use_products,vmin=-1,vmax=1,annot=True)
```

```
[60]: <Axes: >
```

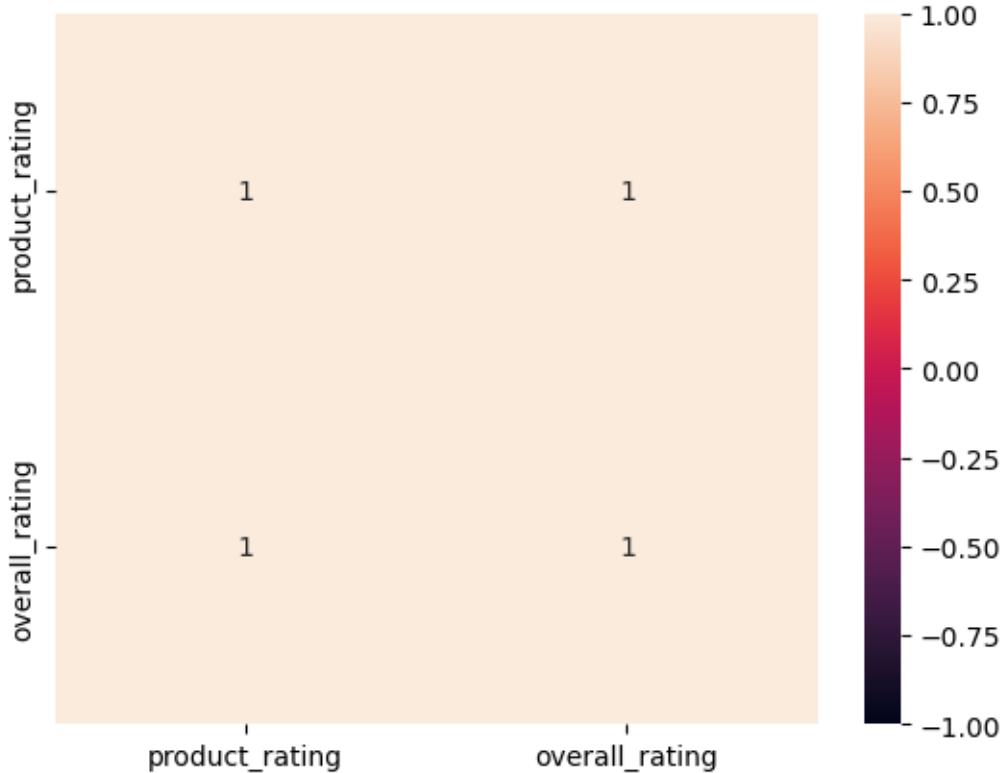


CONCLUSION: From above heat map we find that retail_price and discounted_price is positively related and their bonding is decent and retail_price and discount_percentatge are not so related their realation is too week but discounted_price and discount_percentatge is negitively related to each other but not with tight realation.

RATING ANALYSIS:

```
[63]: rating_corr=daily_use_product[['product_rating','overall_rating']].corr()
sns.heatmap(rating_corr,vmin=-1,vmax=1,annot=True)
```

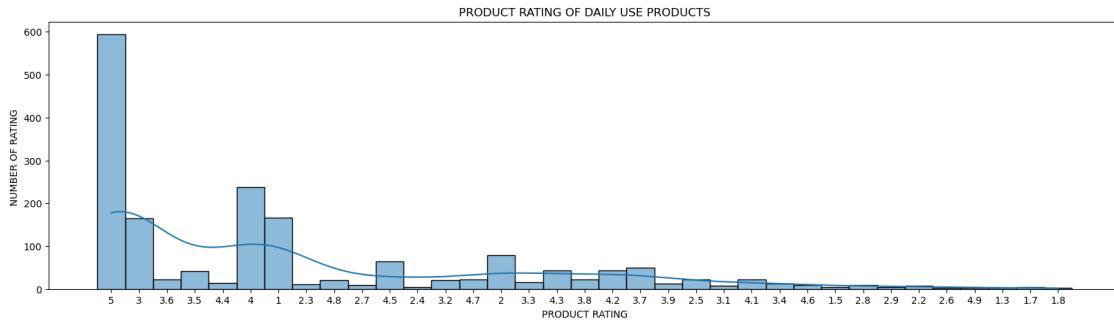
```
[63]: <Axes: >
```



Equal relation between product_rating and overall_rating which show that customers which give product_rating also give overall_rating and vice-versa

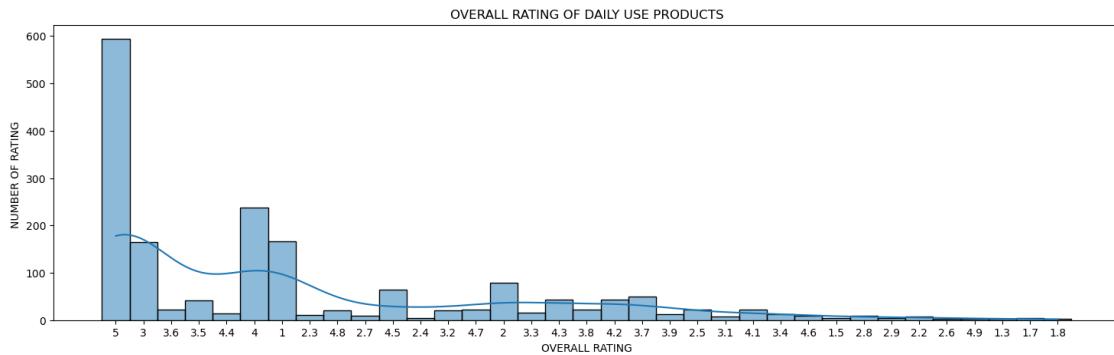
```
[65]: #how rating vary for different products of daily use product category after
      ↵removing zero rating for better results
errorfree_rating_daily_use_product=daily_use_product[~((daily_use_product['product_rating']==0)
      ↵& (daily_use_product['overall_rating']==0))]
errorfree_rating_daily_use_product
plt.figure(figsize=(20,5))
sns.histplot(errorfree_rating_daily_use_product.product_rating,kde=True)
plt.xlabel("PRODUCT RATING")
plt.ylabel("NUMBER OF RATING ")
plt.title("PRODUCT RATING OF DAILY USE PRODUCTS")
```

```
[65]: Text(0.5, 1.0, 'PRODUCT RATING OF DAILY USE PRODUCTS')
```



```
[66]: plt.figure(figsize=(18,5))
sns.histplot(errorfree_rating_daily_use_product.overall_rating,kde=True)
plt.xlabel("OVERALL RATING")
plt.ylabel("NUMBER OF RATING ")
plt.title("OVERALL RATING OF DAILY USE PRODUCTS")
```

```
[66]: Text(0.5, 1.0, 'OVERALL RATING OF DAILY USE PRODUCTS')
```

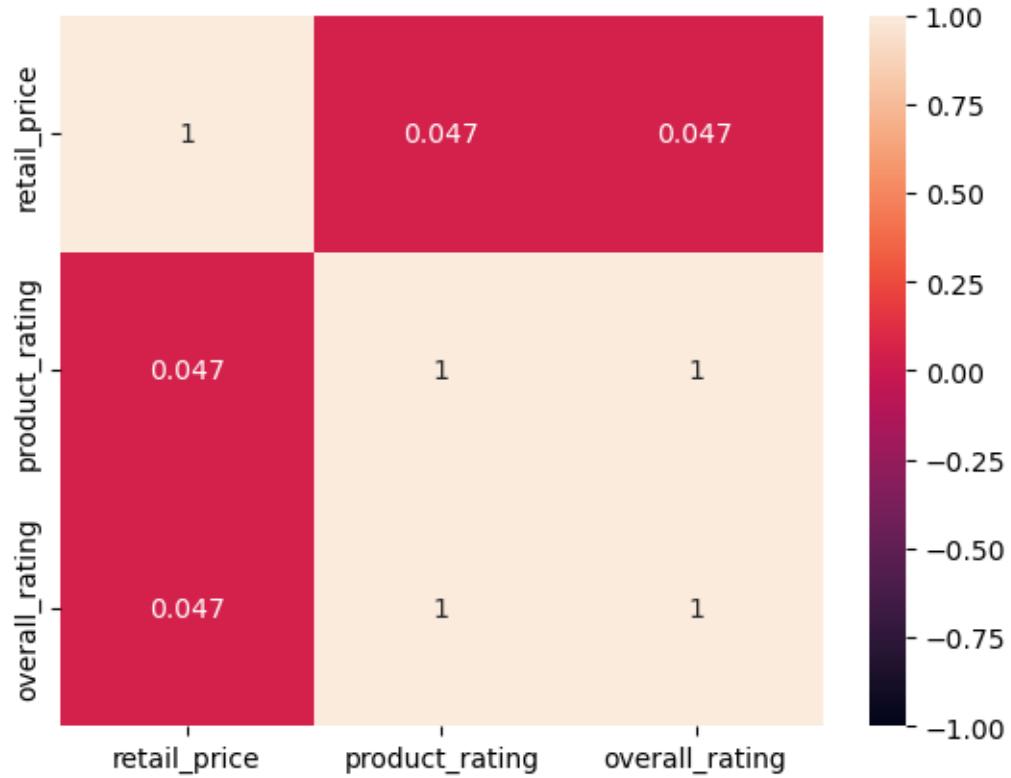


```
[67]: #how retail_price,product_rating,overall_rating related to each other
rating_corr_daily_use_products=errorfree_rating_daily_use_product[['retail_price','product_rating', 'overall_rating']]
rating_corr_daily_use_products
```

	retail_price	product_rating	overall_rating
retail_price	1.000000	0.047371	0.047371
product_rating	0.047371	1.000000	1.000000
overall_rating	0.047371	1.000000	1.000000

```
[68]: sns.heatmap(rating_corr_daily_use_products,vmin=-1,vmax=1,annot=True)
```

```
[68]: <Axes: >
```



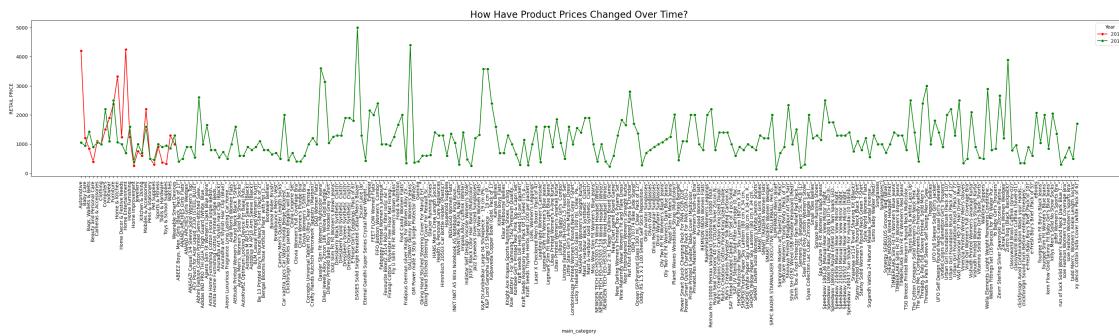
CONCLUSION: All those product which have product_rating have also overall_rating but there is too weak relation between retail_price and product_rating

```
[70]: #How have product prices changed over time?
daily_use_product_price_over_year=daily_use_product.
    ↪groupby(['Year','main_category'])['retail_price'].median()
daily_use_product_price_over_year
```

```
[70]: Year  main_category
2015  Automotive          4200.0
      Baby Care           1220.0
      Bags, Wallets & Belts 849.5
      Beauty and Personal Care 390.0
      Cameras & Accessories 1100.0
      ...
2016  run of luck Solid Women's Round Neck Dark Blue T..."] 299.0
      soie Fashion Women's Full Coverage Bra"]                540.0
      soie Fashion Women's Sports Bra"]                      880.0
      tadd Men's, Women's Ankle Length Socks"]                499.0
      xy decor Cotton Sofa Cover (white Pack of 6)"]        1699.0
Name: retail_price, Length: 270, dtype: float64
```

```
[71]: plt.figure(figsize=(42,6))
plt.xticks(rotation=90)
color=['#FF0000','#008000']
plt.ylabel("RETAIL PRICE")
plt.title("How Have Product Prices Changed Over Time?", fontsize=20)
sns.lineplot(y=daily_use_product_price_over_year.
             ↪values, x=daily_use_product_price_over_year.index.
             ↪get_level_values('main_category'),
             hue=daily_use_product_price_over_year.index.
             ↪get_level_values('Year'), palette=color, marker='o')
```

```
[71]: <Axes: title={'center': 'How Have Product Prices Changed Over Time?'}, xlabel='main_category', ylabel='RETAIL PRICE'>
```



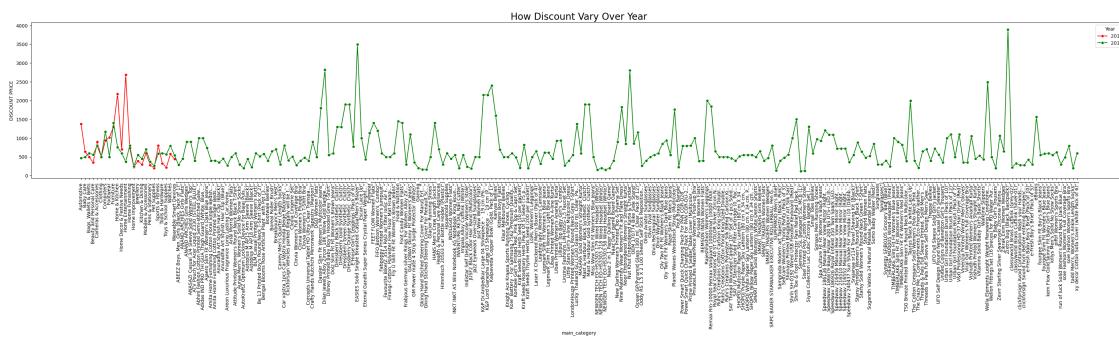
Above visualization shows that there are lots of new product added in 2016 as compare to 2015 and price is also changed in between these year

```
[73]: # how discount vary over year
daily_use_product_discount_price_over_year=daily_use_product.
             ↪groupby(['Year','main_category'])['discounted_price'].median()
daily_use_product_discount_price_over_year
```

```
[73]: Year  main_category
2015  Automotive          1380.0
      Baby Care            640.0
      Bags, Wallets & Belts 499.0
      Beauty and Personal Care 350.0
      Cameras & Accessories 899.0
      ...
2016  run of luck Solid Women's Round Neck Dark Blue T..."] 299.0
      soie Fashion Women's Full Coverage Bra"]                 486.0
      soie Fashion Women's Sports Bra"]                      792.0
      tadd Men's, Women's Ankle Length Socks"]                199.0
      xy decor Cotton Sofa Cover (white Pack of 6)"]          599.0
Name: discounted_price, Length: 270, dtype: float64
```

```
[74]: plt.figure(figsize=(42,6))
plt.xticks(rotation=90)
color=['#FF0000','#008000']
plt.ylabel("DISCOUNT PRICE")
plt.title("How Discount Vary Over Year", fontsize=20)
sns.lineplot(y=daily_use_product_discount_price_over_year.
             ↪values, x=daily_use_product_discount_price_over_year.index.
             ↪get_level_values('main_category'),
             ↪hue=daily_use_product_discount_price_over_year.index.
             ↪get_level_values('Year'), palette=color, marker='o')
```

```
[74]: <Axes: title={'center': 'How Discount Vary Over Year'}, xlabel='main_category',
ylabel='DISCOUNT PRICE'>
```

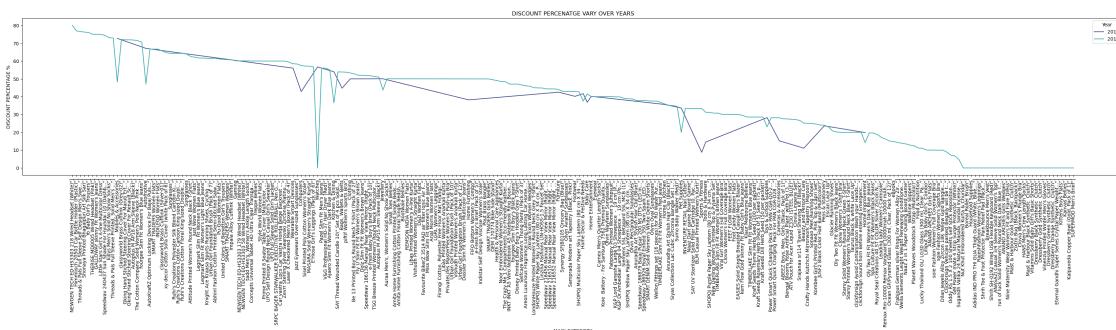


```
[75]: #How discount_percenatage of main_category vary from year to year,month to
      ↪month
      ↪#year to year
year_to_year_discount_change_daily_use_product=daily_use_product.
      ↪groupby(['Year','main_category'])['discount_percentage'].median().
      ↪sort_values(ascending=False)
year_to_year_discount_change_daily_use_product
```

```
[75]: Year  main_category
2016  NEWGEN TECH EO-HS3303 218 Wired Headset (White)"]      80.080080
      Himmlisch 20503 Car Bottle Holder (Plastic)"]          76.982294
      Threads & Pals Full Sleeve Self Design Men's Swe..."]    76.584390
      Shonaya Printed Bhagalpuri Art Silk Sari"]            76.282051
      Yo Baby Girl's Trousers"]                            76.000000
      ...
      Eternal Gandhi Super Series Crystal Paper Weight..."]    0.000000
      Kittens Boys Flats"]                                0.000000
      Kanvas Bellies"]                                0.000000
      Kalpaveda Copper Bowl (Gold, Pack of 1)"]          0.000000
      SUPERMOD Men's Brief"]                            0.000000
Name: discount_percentage, Length: 270, dtype: float64
```

```
[76]: plt.figure(figsize=(42,6))
plt.xticks(rotation=90)
unique_years = year_to_year_discount_change_daily_use_product.index.
    ↪get_level_values('Year').nunique()
sns.lineplot(y=year_to_year_discount_change_daily_use_product.
    ↪values,x=year_to_year_discount_change_daily_use_product.index.
    ↪get_level_values('main_category'),
    hue=year_to_year_discount_change_daily_use_product.index.
    ↪get_level_values('Year')
    ,palette=sns.color_palette("mako", unique_years))
plt.xlabel("MAIN CATEGORY")
plt.ylabel("DISCOUNT PERCENTAGE %")
plt.title("DISCOUNT PERCENTAGE VARY OVER YEARS")
```

[76]: Text(0.5, 1.0, 'DISCOUNT PERCENTAGE VARY OVER YEARS')



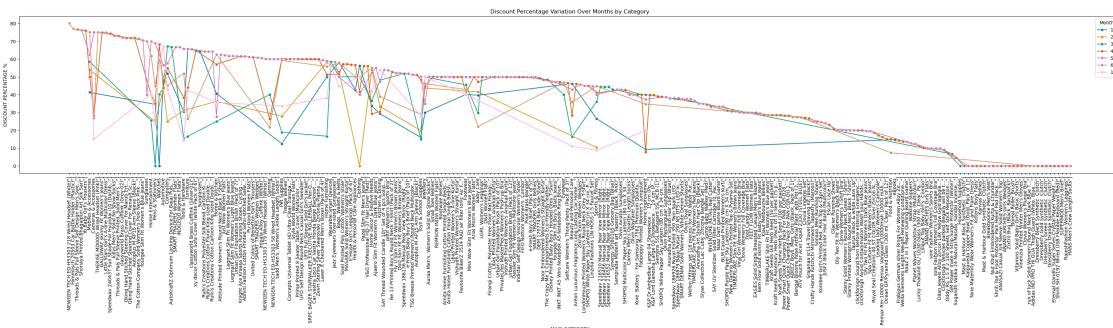
```
[77]: month_to_month_discount_change_daily_use_product=daily_use_product.
    ↪groupby(['Month','main_category'])['discount_percentage'].median().
    ↪sort_values(ascending=False)
month_to_month_discount_change_daily_use_product
```

Month	main_category	discount_percentage
6	NEWGEN TECH EO-HS3303 218 Wired Headset (White)"]	80.080080
	Himmlisch 20503 Car Bottle Holder (Plastic)"]	76.982294
4	Threads & Pals Full Sleeve Self Design Men's Swe..."]	76.584390
5	Shonaya Printed Bhagalpuri Art Silk Sari"]	76.282051
4	Yo Baby Girl's Trousers"]	76.000000
		...
1	Watches	0.000000
4	Shrih SH-0192 Wired USB Flexible Keyboard (Red)"]	0.000000
	SUPERMOD Men's Brief"]	0.000000
	RajeshFashion Women's Leggings"]	0.000000
5	HANS Men's Crew Length Socks"]	0.000000

Name: discount_percentage, Length: 382, dtype: float64

```
[78]: plt.figure(figsize=(42,6))
plt.xticks(rotation=90)
unique_years = month_to_month_discount_change_daily_use_product.index.
    ↪get_level_values('Month').nunique()
sns.lineplot(y=month_to_month_discount_change_daily_use_product.
    ↪values,x=month_to_month_discount_change_daily_use_product.index.
    ↪get_level_values('main_category'),
    hue=month_to_month_discount_change_daily_use_product.index.
    ↪get_level_values('Month')
    ,palette=sns.color_palette("colorblind", unique_years),marker='o')
plt.xlabel("MAIN CATEGORY")
plt.ylabel("DISCOUNT PERCENTAGE %")
plt.title("Discount Percentage Variation Over Months by Category")
```

[78]: Text(0.5, 1.0, 'Discount Percentage Variation Over Months by Category')



```
[79]: #let's check if there any seasonal sale and which product has heighest sale in
    ↪which season
season_sale_daily_use_product=daily_use_product.
    ↪pivot_table(values='retail_price',index=('Month','main_category'),aggfunc='sum')
season_sale_daily_use_product
```

Month	main_category	retail_price
1	Automotive	3702.0
	Baby Care	13027.0
	Bags, Wallets & Belts	110958.5
	Beauty and Personal Care	68042.0
	Clothing	2111014.0
...		...
12	Sports & Fitness	43986.0
	Tools & Hardware	15103.0
	Toys & School Supplies	74816.5
	Watches	223074.0
	Wearable Smart Devices	999.0

```
[382 rows x 1 columns]
```

```
[80]: highest_sales_by_month_daily_use_product= season_sale_daily_use_product.  
      ↪groupby('Month').apply(lambda x: x.loc[x['retail_price'].idxmax()])  
highest_sales_by_month_daily_use_product.  
      ↪sort_values(by='retail_price',ascending=False)
```

```
[80]:      retail_price  
Month  
12      3296120.5  
1       2111014.0  
3       762961.0  
4       551607.0  
6       469791.0  
5       360868.0  
2       85338.0
```

BY ANALYZE THIS MONTH SALE FACTOR WE FIND THAT MOST SALE OF DAILY USE PRODUCT CATEGORY are purchased in 12,1,3 MONTH AND 12,1 MONTH SALE MAY BE HIGH DUE TO NEW YEAR SALE OFFER OR NEW YEAR CAN BE A FACTOR FOR CUSTOMERS TO PURCHASE PRODUCT.

Product Popularity:

```
[83]: #Which product categories have the most products?  
most_products_daily_use_product=daily_use_product.  
      ↪groupby('main_category')['main_category'].size()  
top30_daily_use_product=most_products_daily_use_product.nlargest(30)  
top30_daily_use_product
```

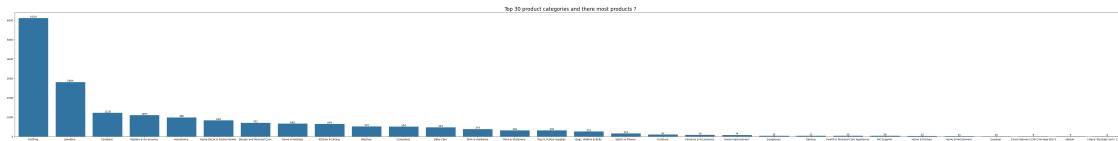
```
[83]: main_category  
Clothing                  6109  
Jewellery                 2804  
Footwear                  1218  
Mobiles & Accessories    1095  
Automotive                 980  
Home Decor & Festive Needs 826  
Beauty and Personal Care  702  
Home Furnishing            663  
Kitchen & Dining           644  
Watches                   519  
Computers                  504  
Baby Care                  467  
Tools & Hardware            379  
Pens & Stationery           309  
Toys & School Supplies      309  
Bags, Wallets & Belts       251
```

```

Sports & Fitness           154
Furniture                   97
Cameras & Accessories      79
Home Improvement             74
Sunglasses                   35
Gaming                       31
Health & Personal Care Appliances 30
Pet Supplies                  30
Home & Kitchen                20
Home Entertainment             15
Eyewear                      10
Clovia Women's Full Coverage Bra"] 9
eBooks                        9
Lilliput Top Baby Girl's Combo"] 8
Name: main_category, dtype: int64

```

```
[84]: plt.figure(figsize=(80,9))
ax=sns.barplot(x=top30_daily_use_product.index,y=top30_daily_use_product.values)
plt.title("Top 30 product categories and there most products?",fontsize=20)
for container in ax.containers:
    ax.bar_label(container,fmt='%d')
```



```
[85]: #product popularity over time
product_popularity_of_daily_use_product=daily_use_product.
    ↪groupby(['Year','main_category'])['pid'].count()
product_popularity_of_daily_use_product
```

```

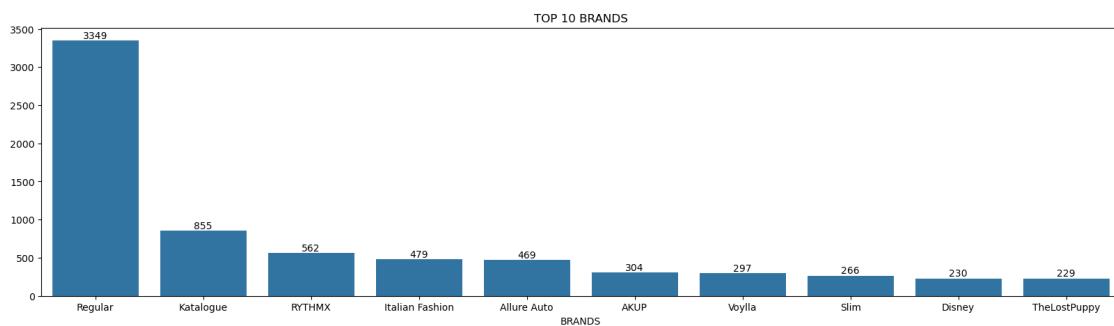
[85]: Year  main_category
2015  Automotive           645
      Baby Care              250
      Bags, Wallets & Belts    42
      Beauty and Personal Care 512
      Cameras & Accessories    67
      ...
2016  run of luck Solid Women's Round Neck Dark Blue T..."] 1
      soie Fashion Women's Full Coverage Bra"]                 1
      soie Fashion Women's Sports Bra"]                      1
      tadd Men's, Women's Ankle Length Socks"]                1
      xy decor Cotton Sofa Cover (white Pack of 6)"]          1
Name: pid, Length: 270, dtype: int64

```

```
[86]: #What are the top 10 most frequently occurring brands?
top_10_brands_of_daily_use_product=daily_use_product.groupby('brand')['brand'].
    ↪value_counts().nlargest(10)
top_10_brands_of_daily_use_product
```

```
[86]: brand
Regular           3349
Katalogue         855
RYTHMX            562
Italian Fashion   479
Allure Auto       469
AKUP              304
Voylla            297
Slim              266
Disney             230
TheLostPuppy      229
Name: count, dtype: int64
```

```
[87]: plt.figure(figsize=(20,5))
ax=sns.barplot(y=top_10_brands_of_daily_use_product.
    ↪values,x=top_10_brands_of_daily_use_product.index.get_level_values('brand'))
plt.xlabel("BRANDS")
plt.title('TOP 10 BRANDS')
for container in ax.containers:
    ax.bar_label(container,fmt='%d')
```



Category Exploration:

```
[89]: #How many distinct product categories exist in daily use product category?
main_category_products_of_daily_use_product=daily_use_product.
    ↪groupby('main_category')['main_category'].nunique()
main_category_products_of_daily_use_product
```

```
[89]: main_category
ABEEZ Boys, Men, Girls (Black, Pack of 1)"]
```

```

ANAND ARCHIES Girls Flats"] 1
ANAND ARCHIES Girls Wedges"] 1
ANASAZI Casual 3/4 Sleeve Solid Women's Top"] 1
ATV Pouch for Acer Liquid Z330 (STEEL BLUE)"] 1
..
run of luck Solid Women's Round Neck Dark Blue T..."] 1
soie Fashion Women's Full Coverage Bra"] 1
soie Fashion Women's Sports Bra"] 1
tadd Men's, Women's Ankle Length Socks"] 1
xy decor Cotton Sofa Cover (white Pack of 6)"] 1
Name: main_category, Length: 246, dtype: int64

```

There are 246 different product category are exist for daily use product tp know this use main_category_products_of_daily_use_product.value_counts()

```
[91]: #What are the most common product subcategories within each category in daily_use_product ?
most_common_product_first_category_daily=daily_use_product.
    .groupby(['main_category'])['secondary_category'].nunique().
    .sort_values(ascending=False)
most20_first_category_daily=most_common_product_first_category_daily.head(20)
most20_first_category_daily
```

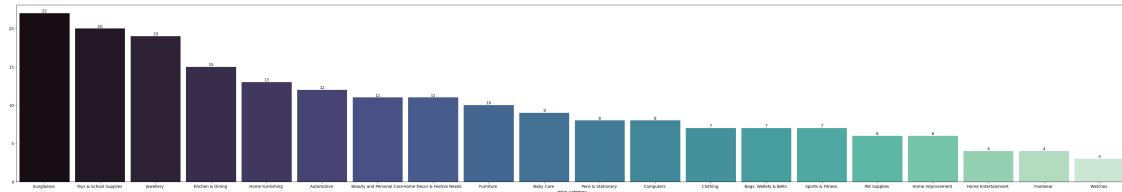
```
[91]: main_category
Sunglasses 22
Toys & School Supplies 20
Jewellery 19
Kitchen & Dining 15
Home Furnishing 13
Automotive 12
Beauty and Personal Care 11
Home Decor & Festive Needs 11
Furniture 10
Baby Care 9
Pens & Stationery 8
Computers 8
Clothing 7
Bags, Wallets & Belts 7
Sports & Fitness 7
Pet Supplies 6
Home Improvement 6
Home Entertainment 4
Footwear 4
Watches 3
Name: secondary_category, dtype: int64
```

```
[92]: plt.figure(figsize=(50,8))
ax=sns.barplot(y=most20_first_category_daily.
                 ↪values,x=most20_first_category_daily.index.get_level_values('main_category')
                 ,width=0.90,palette='mako')
for container in ax.containers:
    ax.bar_label(container, fmt='%d')
```

/var/folders/vc/4h3d9s6d639f63qzbhd805b80000gn/T/ipykernel_1158/3671279258.py:2:
FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `x` variable to `hue` and set `legend=False` for the same effect.

```
ax=sns.barplot(y=most20_first_category_daily.values,x=most20_first_category_da
ily.index.get_level_values('main_category'))
```



```
[93]: most_common_product_second_category_daily=daily_use_product.
       ↪groupby(['secondary_category'])['tertiary_category'].nunique().
       ↪sort_values(ascending=False)
most20_second_category_daily=most_common_product_second_category_daily.head(20)
most20_second_category_daily
```

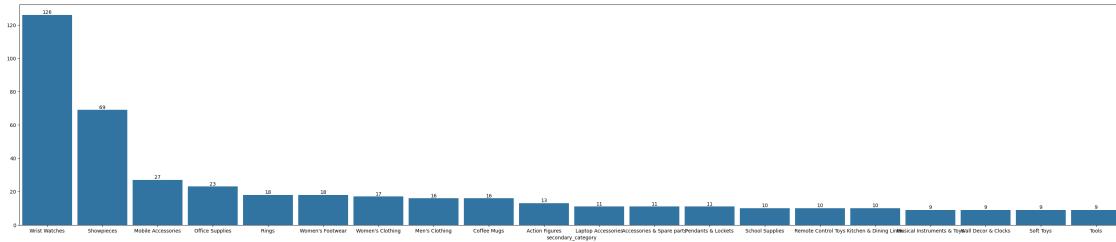
```
[93]: secondary_category
Wrist Watches           126
Showpieces              69
Mobile Accessories       27
Office Supplies          23
Rings                    18
Women's Footwear         18
Women's Clothing          17
Men's Clothing            16
Coffee Mugs              16
Action Figures            13
Laptop Accessories        11
Accessories & Spare parts 11
Pendants & Lockets        11
School Supplies            10
Remote Control Toys        10
```

```

Kitchen & Dining Linen          10
Musical Instruments & Toys      9
Wall Decor & Clocks             9
Soft Toys                         9
Tools                            9
Name: tertiary_category, dtype: int64

```

```
[94]: plt.figure(figsize=(40,8))
ax=sns.barplot(y=most20_second_category_daily.
                 ↪values,x=most20_second_category_daily.index.
                 ↪get_level_values('secondary_category')
                 ,width=0.90)
for container in ax.containers:
    ax.bar_label(container, fmt='%d')
```



```
[95]: most_common_product_third_category_daily=daily_use_product.
        ↪groupby(['tertiary_category'])['quaternary_category'].nunique().
        ↪sort_values(ascending=False)
most20_third_category_daily=most_common_product_third_category_daily.
        ↪nlargest(20)
most20_third_category_daily
```

```
[95]: tertiary_category
Casual Shoes                      119
T-Shirts                          88
Cases & Covers                   79
Brooches                          74
USB Gadgets                       73
Geometry & Pencil Boxes          60
Ethnic                            55
Hand Bags                         47
Backpacks                         40
Routers                           39
Formal Shoes                      38
Screen Protectors                 29
Bulbs                            29
Vanity Boxes                      24
```

```

Curtains           24
Ballerinas        22
AKUP Coffee Mugs  22
Trousers          22
Slippers & Flip Flops 21
Cables            19
Name: quaternary_category, dtype: int64

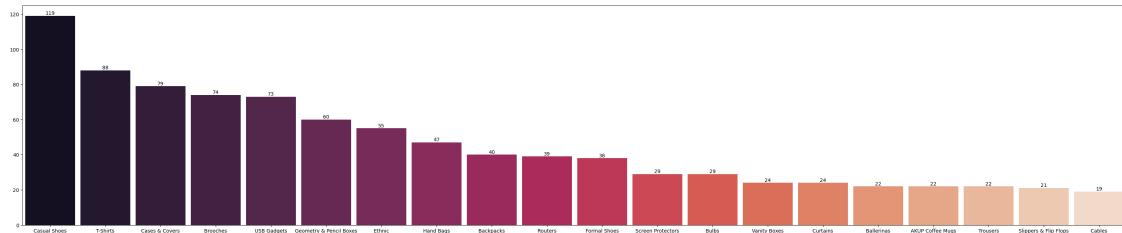
```

```
[96]: plt.figure(figsize=(40,8))
ax=sns.barplot(y=most20_third_category_daily.
                 ↪values,x=most20_third_category_daily.index.
                 ↪get_level_values('tertiary_category')
                 ,width=0.90,palette='rocket')
for container in ax.containers:
    ax.bar_label(container, fmt='%d')
```

```
/var/folders/vc/4h3d9s6d639f63qzbhd805b80000gn/T/ipykernel_1158/4287864878.py:2:
FutureWarning:
```

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `x` variable to `hue` and set `legend=False` for the same effect.

```
ax=sns.barplot(y=most20_third_category_daily.values,x=most20_third_category_da
ily.index.get_level_values('tertiary_category'))
```



```
[97]: # What are the most common product subcategories within each category?
level_1_category_daily_use_product=daily_use_product.
    ↪groupby(['main_category', 'secondary_category']).size().
    ↪sort_values(ascending=False)
level_1_category_daily_use_product
```

main_category	secondary_category
Clothing	Women's Clothing
3865	
	Men's Clothing
1722	

```

Jewellery                                Necklaces & Chains
1594
Automotive                                Accessories & Spare parts
896
Mobiles & Accessories                    Tablet Accessories
795
...
Leading lady Women's Camisole"]           None
1
Legmark Slim Fit Women's Blue Jeans"]     None
1
Legmark Slim Fit Women's Light Blue Jeans"] None
1
Libas Printed Women's A-line Kurta"]       None
1
xy decor Cotton Sofa Cover (white Pack of 6)"] None
1
Length: 434, dtype: int64

```

```
[98]: level_2_category_daily_use_product=daily_use_product.
      ↪groupby(['secondary_category','tertiary_category']).size().
      ↪sort_values(ascending=False)
level_2_category_daily_use_product
```

```

[98]: secondary_category      tertiary_category
Women's Clothing            Western Wear           1969
Necklaces & Chains         Necklaces"]           1564
Women's Clothing            Lingerie, Sleep & Swimwear 1207
Men's Clothing              T-Shirts              903
Tablet Accessories          Cases & Covers        796
...
Remote Control Toys         Toyhouse Remote Control Toys"]   1
                           Taaza Garam Remote Control Toys"]   1
                           Silverlit Remote Control Toys"]   1
                           Nikko Remote Control Toys"]   1
Office Supplies             Boards & Dusters        1
Length: 915, dtype: int64

```

```
[99]: level_3_category_daily_use_product=daily_use_product.
      ↪groupby(['tertiary_category','quaternary_category']).size().
      ↪sort_values(ascending=False)
level_3_category_daily_use_product
```

```

[99]: tertiary_category      quaternary_category
Necklaces"]                   None                  1564
Western Wear                  Shirts, Tops & Tunic 1228
Lingerie, Sleep & Swimwear   Bra                  1036

```

None	None	757
Car Interior & Exterior	Car Interio	636
		...
Ethnic	Urban Monk Creations Ethnic	1
	UFC Mart Ethnic	1
	Total Furnishing Ethnic	1
	The Ethnic Story Ethnic	1
Hand Bags	BUTTERFILES Hand Bag	1

Length: 2658, dtype: int64

Conclusion: From above analysis we find that there is most sales in “Clothing” category in which subcategory “women’s clothing” is on top followed by ‘men’s clothing’ if we futher go deep down we will find that “western wear” are on peek and in ‘western wear’ “Shirts, Tops & Tunics” are mostly odered followed by ‘Dresses & skirts’. Second most purchased thing is ‘Jewellery’ and in jewellery ’ Necklaces & Chains ’ are mostly ordered and in this category (Necklaces & Chains) mostly necklaces are placed.

Brand Analysis:

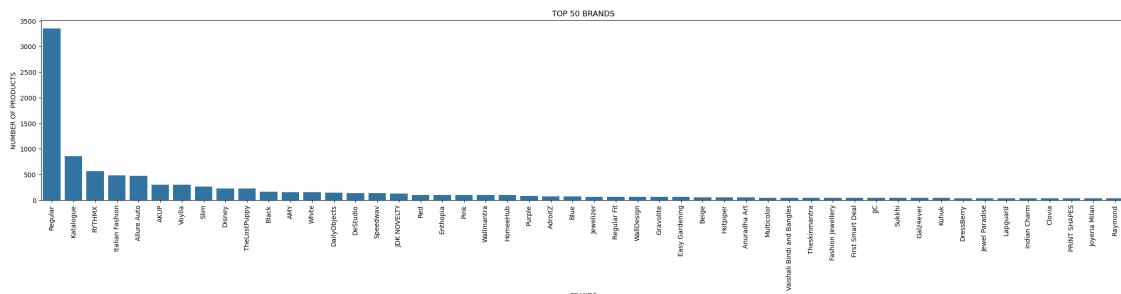
```
[102]: #Which brands have the most products in the daily use product category?
most_brands_daily_use_product=daily_use_product.groupby('brand').size()
top50_brands_daily_use_product= most_brands_daily_use_product.nlargest(50)
top50_brands_daily_use_product
```

brand	
Regular	3349
Katalogue	855
RYTHMX	562
Italian Fashion	479
Allure Auto	469
AKUP	304
Voylla	297
Slim	266
Disney	230
TheLostPuppy	229
Black	166
AMY	156
White	155
DailyObjects	144
DeStudio	138
Speedwav	137
JDK NOVELTY	125
Red	103
Enthopia	101
Pink	100
Wallmantra	97
HomeeHub	95
Purple	83

AdroitZ	74
Blue	71
Jewelizer	67
Regular Fit	66
WallDesign	65
Gravolite	63
Easy Gardening	59
Beige	54
Hotpiper	53
Anuradha Art	51
Multicolor	47
Vaishali Bindi and Bangles	46
Theskinmantra	45
Fashion Jewellery	44
First Smart Deal	42
JJC	41
Sukkhi	41
Galz4ever	40
Kuhuk	40
DressBerry	38
Jewel Paradise	38
Lapguard	38
Indian Charm	37
Clovia	36
PRINT SHAPES	36
Joyeria Milan	35
Raymond	35

```
plt.figure(figsize=(30,5))
plt.xticks(rotation=90)
sns.barplot(x=top50_brands_daily_use_product.
             index,y=top50_brands_daily_use_product.values)
plt.xlabel('BRANDS')
plt.ylabel('NUMBER OF PRODUCTS')
plt.title("TOP 50 BRANDS")
```

[103]: Text(0.5, 1.0, 'TOP 50 BRANDS')



```
[104]: most_brands_daily_use_product.nsmallest(50)
```

```
[104]: brand
3kFactory           1
5 FEELINGS          1
A Bit of Me         1
A R ENTERPRISES    1
A To Z Traders     1
A-maze              1
AAKAR               1
ABB                 1
ABEEZ               1
ABIDA               1
ABSTAR              1
ADIWALK             1
AHAANA FASHION     1
ALIFS               1
ALLURING            1
AM Creation          1
APG                 1
APOLLO+              1
AQUAPOLO             1
ARIHANT              1
ARMOIRE              1
AROVI               1
ASIAN                1
ATIS AMCO             1
AUROSHIKHA            1
AUTOSiTY              1
AYESHA               1
AYMH                1
Aadi                 1
Aaditri28             1
Aadyaa Collections    1
Aakar                1
AaliyaB              1
Aamore Decor           1
AapnoCrafts            1
Aapti                 1
Aarnaa               1
Aarti Collections      1
Aarya 24kt              1
Asaan                 1
Aavaya Fashion          1
Abomedecor             1
```

```

Abhooshan          1
Abony             1
Abqa Hydrabadi Bangle 1
Abstract Mood     1
Acromax           1
Action            1
Addyvero          1
Aden & M         1
dtype: int64

```

```

[105]: #Are there any brands that consistently offer higher discounts?
#Finding which brands give what percentage of discount to their product
higher_discount_brands_daily_use_product=daily_use_product.
    ↪groupby('brand')[ 'discount_percentage' ].mean()
top30_discount_brands_daily_use_product=_
    ↪higher_discount_brands_daily_use_product.nlargest(30)
top30_discount_brands_daily_use_product

```

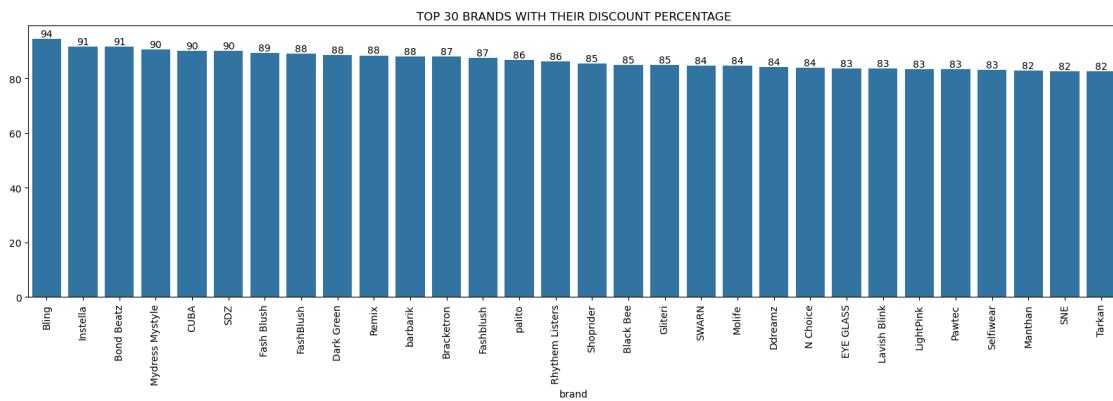
```

[105]: brand
Bling           94.548458
Instella        91.719745
Bond Beatz      91.596639
Mydress Mystyle 90.518987
CUBA            90.045023
SDZ             90.045023
Fash Blush      89.256770
FashBlush       88.983744
Dark Green      88.529638
Remix           88.430269
barbarik        88.035214
Bracketron      87.939394
Fashblush       87.488410
palito          86.733333
Rhythem Listers 86.243122
Shoprider        85.542771
Black Bee        85.034014
Gliteri          85.000000
SWARN            84.723149
Molife           84.680523
Ddreamz          84.236864
N Choice         84.016803
EYE GLASS        83.546415
Lavish Blink     83.541771
LightPink        83.472454
Pawtec           83.388926
Selfiwear        83.208333
Manthan          82.880823

```

```
SNE           82.655103
Tarkan        82.541271
Name: discount_percentage, dtype: float64
```

```
[106]: plt.figure(figsize=(20,5))
plt.xticks(rotation=90)
ax=sns.barplot(x=top30_discount_brands_daily_use_product.
    ↪index,y=top30_discount_brands_daily_use_product.values)
plt.title('TOP 30 BRANDS WITH THEIR DISCOUNT PERCENTAGE',loc='center')
for x in ax.containers:
    ax.bar_label(x,fmt='%d')
```



```
[107]: #To check the brands consistency for discount_percentage we can find standard deviation to know more.
consistent_higher_discount_brands_daily=daily_use_product.
    ↪groupby('brand')['discount_percentage'].std()

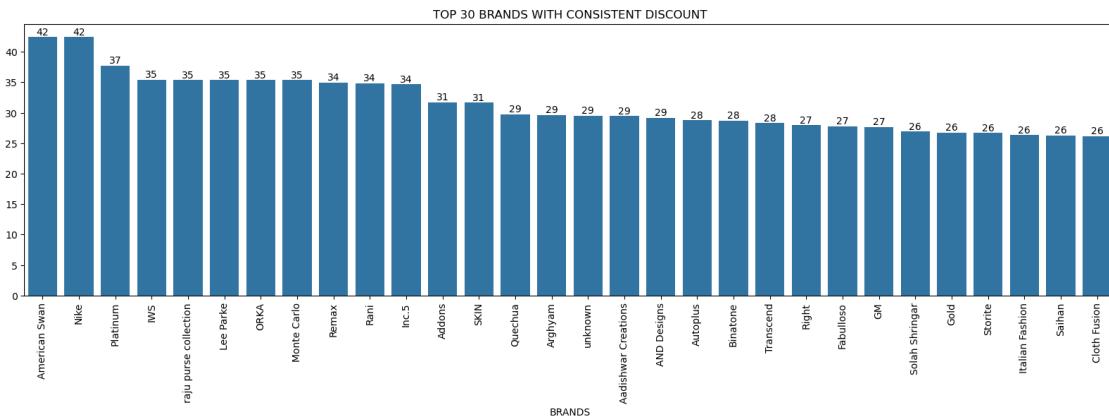
top30_consistent_higher_discount_brands_daily=_
    ↪consistent_higher_discount_brands_daily.sort_values(ascending=False).head(30)
top30_consistent_higher_discount_brands_daily
```

```
[107]: brand
American Swan      42.444861
Nike                42.426407
Platinum           37.737520
IWS                35.414363
raju purse collection 35.390730
Lee Parke           35.390730
ORKA                35.373026
Monte Carlo         35.355339
Remax               34.945151
Rani                34.864170
```

Inc.5	34.641016
Addons	31.739101
SKIN	31.689150
Quechua	29.741324
Arghyam	29.651344
unknown	29.518041
Aadishwar Creations	29.470914
AND Designs	29.168155
Autoplus	28.739649
Binatone	28.689255
Transcend	28.314630
Right	27.970829
Fabuloso	27.788636
GM	27.669074
Solah Shringar	26.943803
Gold	26.758368
Storite	26.654282
Italian Fashion	26.367177
Saihan	26.293431
Cloth Fusion	26.147121

Name: discount_percentage, dtype: float64

```
[108]: plt.figure(figsize=(20,5))
plt.xticks(rotation=90)
ax=sns.barplot(x=top30_consistent_higher_discount_brands_daily.
                 index,y=top30_consistent_higher_discount_brands_daily.values)
plt.title('TOP 30 BRANDS WITH CONSISTENT DISCOUNT',loc='center')
plt.xlabel('BRANDS')
for x in ax.containers:
    ax.bar_label(x,fmt='%d')
```



CONCLUSION: As from above visualization about brands and their discount percentage we find that Bling 94.548458 Instella 91.719745 Bond Beatz 91.596639 Mydress Mystyle 90.518987 CUBA

90.045023 these are the top 5 brands with their discount percentage which give heighest discount on their product. But if we talk about consistent brands which give heigher discount are different compare to brands with heigher discount and these are: American Swan 42.444861 Nike 42.426407 Platinum 37.737520 IWS 35.414363 raju purse collection 35.390730

so,finally we find that American Swan give consistent heigher discounts on their products.

OCCASIONAL PRODUCT ANALYSIS

[111]: `occasional_product.sample(10)`

[111]: `uniq_id \`

```
11637 f7fa98e84f86d29c6d50c4191f3b827b
13353 63636c881ad26ac1246a7fdb191bcd36
13046 ec03c741b671699360292af03321a58f
13113 f61b0e45a9d584cb8335dfc4b5c9ce84
18463 db23c8e84b772989effb123716c7254c
10379 7888fb346e28b3b2f0ae3cbd66967e01
95      f80ba21b9e77fe3a4d0e944f86305f8d
244      d6a4b62067d04cab8397e76e01656d0a
12794      fa5110af297d851ca95bf254dd76f3a4
13121      d11e0598db7c4fa35da2d0294c9f3daa
```

`product_name pid \`

```
11637 Durian Fabric 1 Seater Sofa SOFEE8CG7UJ62T8F
13353 BlueStone The Nabha Yellow Gold Diamond 14 K Ring RNGE9HE5ZYYRGYZ
13046 Fullcutdiamond FCDR5376R Yellow Gold Diamond 1... RNGEC8PZZJWYDH6F
13113 Gunjan Diamond Gold Diamond 18K Yellow Gold 18... RNGEE7ADESACX75T
18463 GAGA Canvas Painting PTGEGUZFRVHMXTFH
10379 Indianbeauty Self Design, Printed Fashion Geor... SAREGTVFGRYRDCNH
95      Reckler Slim Fit Men's Jeans JEAECN2PWUKFA9GJ
244      Magnum Footwear Lifestyle Boots SHOEBZ3DYJZDTGAP
12794      Karatcraft Linnea Yellow Gold Diamond 18 K Ring RNGE7M92QYXEPXGH
13121      Remington Hair Clipper HC5810-Genius Trimmer F... SHVD2H7URCYAHAAH
```

`retail_price discounted_price is_FK_Advantage_product product_rating \`

11637	23400.0	15210.0	False	0
13353	42939.0	29942.0	False	0
13046	48069.0	33649.0	False	0
13113	36100.0	27000.0	False	0
18463	10461.0	5780.0	False	0
10379	5000.0	899.0	False	0
95	5398.0	2599.0	False	0
244	5999.0	3199.0	False	0
12794	41432.0	30653.0	False	0
13121	7599.0	5975.0	False	4.9

`overall_rating brand Year Month Days \`

11637	0	Durian	2015	12	31
13353	0	BlueStone	2016	1	1
13046	0	Fullcutdiamond	2016	1	1
13113	0	Gunjan Diamond	2016	1	1
18463	0	GAGA	2016	3	18
10379	0	Indianbeauty	2016	3	16
95	0	Slim	2015	12	4
244	0	Katalogue	2015	12	4
12794	0	Karatcraft	2016	1	1
13121	4.9	Remington	2016	1	1

		main_category	secondary_category	\
11637		Furniture	Living Room	
13353		Jewellery	Rings"]	
13046		Jewellery	Rings"]	
13113		Jewellery	Rings"]	
18463	Home Decor & Festive Needs		Wall Decor & Clocks	
10379		Clothing	Women's Clothing	
95		Clothing	Men's Clothing	
244		Footwear	Women's Footwear	
12794		Jewellery	Rings"]	
13121	Health & Personal Care Appliances		Personal Care Appliances	

		tertiary_category	quaternary_category	discount_percentage	\
11637	Sofas & Sectionals"]		None	35.000000	
13353		None	None	30.268520	
13046		None	None	29.998544	
13113		None	None	25.207756	
18463	Paintings	GAGA Painting		44.747156	
10379	Ethnic Wear	Saree		82.020000	
95	Jeans	Reckler Jeans		51.852538	
244	Casual Shoes	Boots		46.674446	
12794		None		26.016123	
13121	Trimmers"]		None	21.371233	

	product_category
11637	Occasional Product
13353	Occasional Product
13046	Occasional Product
13113	Occasional Product
18463	Occasional Product
10379	Occasional Product
95	Occasional Product
244	Occasional Product
12794	Occasional Product
13121	Occasional Product

PRICE ANALYSIS:

```
[113]: occasional_product.describe()
```

```
[113]:    retail_price  discounted_price      Year      Month      Days \
count    1292.000000    1292.000000  1292.000000  1292.000000  1292.000000
mean    24102.942724  17878.239938  2015.782508   3.982198   6.603715
std     15110.551526  11874.994702   0.412700   4.401305   8.573061
min     5000.000000   208.000000  2015.000000   1.000000   1.000000
25%    8027.000000   5999.000000  2016.000000   1.000000   1.000000
50%    24063.500000  18163.000000  2016.000000   1.000000   1.000000
75%    37877.500000  27905.000000  2016.000000   6.000000   7.000000
max    49999.000000  47620.000000  2016.000000  12.000000  31.000000

      discount_percentage
count            1292.000000
mean            28.216496
std             19.084493
min             0.000000
25%            19.998317
50%            26.322011
75%            33.727273
max            96.533333
```

```
[114]: #How average and median price vary for different products category
average_retail_price_for_occasional_products=occasional_product.
    ↪groupby(['main_category'])['retail_price'].apply(lambda x: x.mean())
median_retail_price_for_occasional_products=occasional_product.
    ↪groupby(['main_category'])['retail_price'].apply(lambda x: x.median())

average_retail_price_for_occasional_products
```

```
[114]: main_category
883 Police Full Sleeve Solid Men's Jacket"]           7499.000000
Asics Gel-Cumulus 17 Running Shoes"]                  9499.000000
Asics Gel-Kayano 22 Running Shoes"]                  12499.000000
Automation & Robotics                           19999.000000
Automotive                                         7689.906250
BALAJI EXPORTS Bottled Wine Cooler (9 Bottles)"]    10000.000000
Baby Care                                         6818.812500
Bags, Wallets & Belts                           10638.357143
Beauty and Personal Care                         12714.375000
Behringer Xenyx 502 Analog Sound Mixer"]           10000.000000
BuildTrack PIR Wireless Motion Sensor - One Swit..."] 6500.000000
COIRFIT Single Coir Mattress"]                     9743.500000
Cameras & Accessories                           6974.333333
Carbanao Chrome Grill Chevrolet Cruze Car Grill ..."] 6000.000000
```

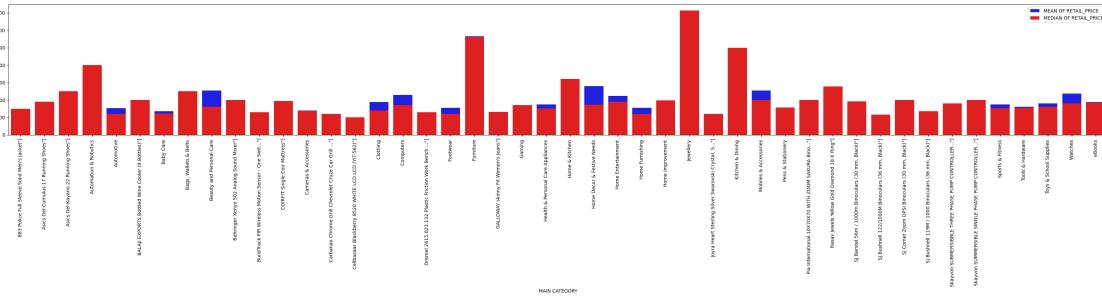
Cellbazaar Blackberry 8520 WHITE LCD LCD (YIT-562)"]	5000.000000
Clothing	9392.875000
Computers	11444.067568
Dremel 2615.023.132 Plastic Friction Work Bench ..."]	6500.000000
Footwear	7777.111111
Furniture	28241.133333
GALLOWAY skinny Fit Women's Jeans"]	6599.000000
Gaming	8499.000000
Health & Personal Care Appliances	8701.461538
Home & Kitchen	15396.250000
Home Decor & Festive Needs	14010.669903
Home Entertainment	11119.500000
Home Furnishing	7740.864865
Home Improvement	9566.142857
Jewellery	33230.202201
Joyra Heart Sterling Silver Swarovski Crystal, S..."]	5995.000000
Kitchen & Dining	23696.333333
Mobiles & Accessories	12709.333333
Pens & Stationery	7889.000000
Pia International 10X70X70 WITH ZOOM SAKURA Bino..."]	9999.000000
Rasav Jewels Yellow Gold Diamond 18 K Ring"]	13903.000000
SJ Barstel 56m / 1000m Binoculars (30 mm, Black)"]	9649.000000
SJ Bushnell 122/1000M Binoculars (36 mm, Black)"]	5779.000000
SJ Comet Zoom DPSI Binoculars (30 mm, Black)"]	9999.000000
SJ Bushnell 119M / 1000 Binoculars (36 mm, Black)"]	6749.000000
Skayvon SUBMERSIBBLE THREE PHASE PUMP CONTROLLER..."]	8999.000000
Skayvon SUMMERSIBLE SINGLE PHASE PUMP CONTROLLER..."]	9990.000000
Sports & Fitness	8728.916667
Tools & Hardware	8093.000000
Toys & School Supplies	9029.476190
Watches	11815.500000
eBooks	9449.833333

Name: retail_price, dtype: float64

```
[115]: plt.figure(figsize=(42,5))
plt.xticks(rotation=90)

sns.barplot(y=average_retail_price_for_occasional_products.
            ↪values,x=average_retail_price_for_occasional_products.
            ↪index,color='blue',label='MEAN OF RETAIL PRICE')
sns.barplot(y=median_retail_price_for_occasional_products.
            ↪values,x=median_retail_price_for_occasional_products.
            ↪index,color='red',label="MEDIAN OF RETAIL PRICE")
plt.xlabel("MAIN CATEGORY")
plt.legend(loc="upper right")
```

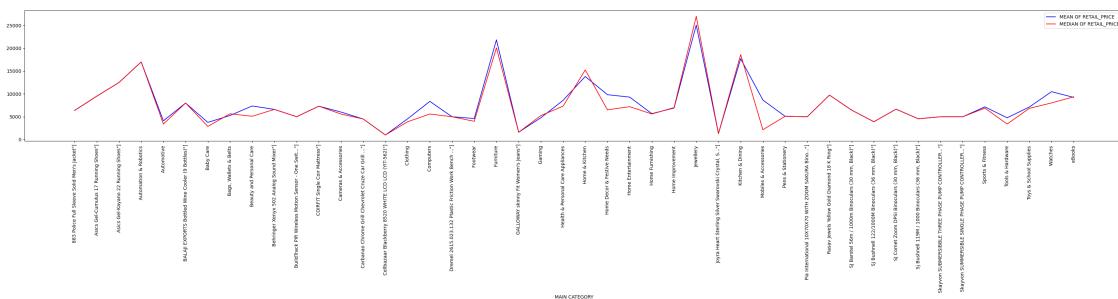
[115]: <matplotlib.legend.Legend at 0x14fd8bce0>



```
[116]: # how discount vary by different main category in occasional product category
average_discounted_price_for_occasional_product=occasional_product.
    ↪groupby(['main_category'])['discounted_price'].apply(lambda x: x.mean())
median_discounted_price_for_occasional_product=occasional_product.
    ↪groupby(['main_category'])['discounted_price'].apply(lambda x: x.median())
```

```
[117]: pd.option_context('mode.use_inf_as_na', True)
plt.figure(figsize=(42,5))
plt.xticks(rotation=90)
sns.lineplot(y=average_discounted_price_for_occasional_product.
             ↴values,x=average_discounted_price_for_occasional_product.
             ↴index,color='blue',label='MEAN OF RETAIL PRICE')
sns.lineplot(y=median_discounted_price_for_occasional_product.
             ↴values,x=median_discounted_price_for_occasional_product.
             ↴index,color='red',label="MEDIAN OF RETAIL PRICE")
plt.xlabel("MAIN CATEGORY")
plt.legend(loc="upper right")
```

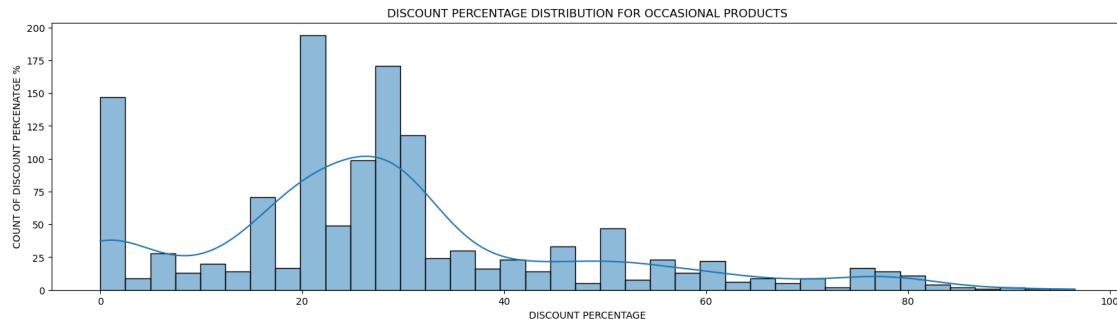
[117]: <matplotlib.legend.Legend at 0x1591786e0>



```
[118]: # how discount percentage vary for different product in occasional product
plt.figure(figsize=(20,5))
sns.histplot(occasional_product['discount_percentage'], kde=True)
plt.title('DISCOUNT PERCENTAGE DISTRIBUTION FOR OCCASIONAL PRODUCTS')
```

```
plt.xlabel("DISCOUNT PERCENTAGE")
plt.ylabel(" COUNT OF DISCOUNT PERCENTAGE % ")
```

[118]: Text(0, 0.5, ' COUNT OF DISCOUNT PERCENTAGE % ')



[119]: *# top 30 products with average highest discount in occasional products*
 discount=occasional_product.groupby(['main_category'])['discount_percentage'].
 ↪mean()
 top_30_discounted_occasional_product=discount.nlargest(30)
 top_30_discounted_occasional_product

main_category	
Cellbazaar Blackberry 8520 WHITE LCD LCD (YIT-562)"]	80.000000
Joyra Heart Sterling Silver Swarovski Crystal, S..."]	77.998332
GALLOWAY skinny Fit Women's Jeans"]	75.769056
Clothing	50.420073
Skayvon SUMMERSIBLE SINGLE PHASE PUMP CONTROLLER..."]	50.050050
Pia International 10X70X70 WITH ZOOM SAKURA Bino..."]	50.005001
Bags, Wallets & Belts	49.723842
Mobiles & Accessories	49.646323
Automotive	46.710497
Baby Care	44.806397
Skayvon SUBMERSIBLE THREE PHASE PUMP CONTROLLER..."]	44.549394
Tools & Hardware	44.027050
Beauty and Personal Care	41.672509
Gaming	39.394603
Footwear	38.618116
Pens & Stationery	35.179995
Behringer Xenyx 502 Analog Sound Mixer"]	34.000000
SJ Comet Zoom DPSI Binoculars (30 mm, Black)"]	33.503350
SJ Barstel 56m / 1000m Binoculars (30 mm, Black)"]	33.164058
Sj Bushnell 119M / 1000 Binoculars (36 mm, Black)"]	32.597422
SJ Bushnell 122/1000M Binoculars (36 mm, Black)"]	32.531580
Rasav Jewels Yellow Gold Diamond 18 K Ring"]	29.993527
Computers	29.742531

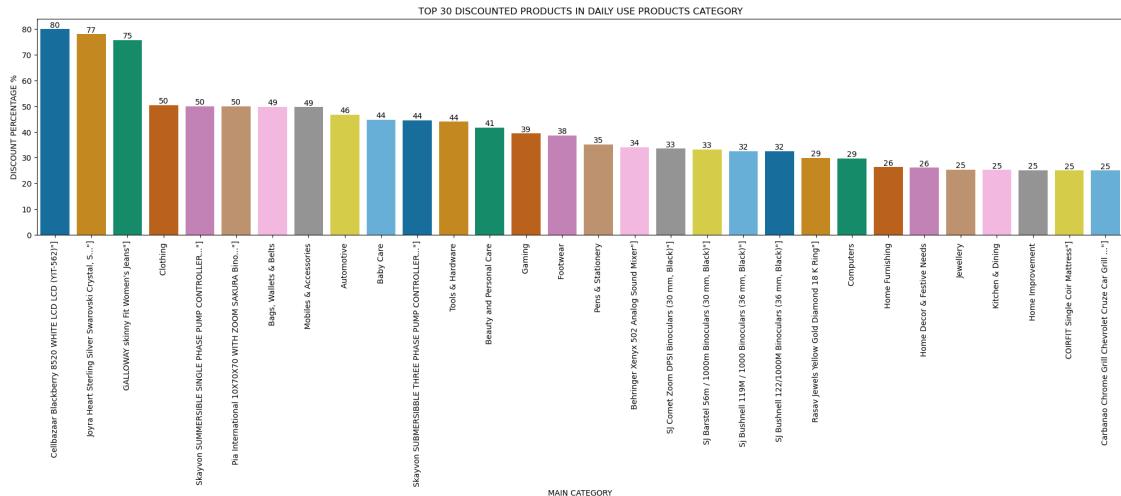
Home Furnishing	26.340620
Home Decor & Festive Needs	26.091300
Jewellery	25.278608
Kitchen & Dining	25.248311
Home Improvement	25.110694
COIRFIT Single Coir Mattress"]	25.006357
Carbanao Chrome Grill Chevrolet Cruze Car Grill ..."]	25.000000
Name: discount_percentage, dtype: float64	

```
[120]: plt.figure(figsize=(25,5))
plt.xticks(rotation=90)
ax=sns.barplot(x=top_30_discounted_occasional_product.
    ↪index,y=top_30_discounted_occasional_product.values,palette='colorblind')
plt.xlabel("MAIN CATEGORY")
plt.ylabel("DISCOUNT PERCENTAGE %")
plt.title('TOP 30 DISCOUNTED PRODUCTS IN DAILY USE PRODUCTS CATEGORY')
for conatiner in ax.containers:
    ax.bar_label(conatiner,fmt='%d')
```

/var/folders/vc/4h3d9s6d639f63qzbhd805b80000gn/T/ipykernel_1158/1106444037.py:3:
FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `x` variable to `hue` and set `legend=False` for the same effect.

```
    ax=sns.barplot(x=top_30_discounted_occasional_product.index,y=top_30_discounted_occasional_product.values,palette='colorblind')
```



```
[121]: #Are there any significant price differences between FK Advantage products and others?
fk_advantage_product_in_occasional_products=occasional_product.
groupby(["is_FK_Advantage_product"])['retail_price'].describe()
fk_advantage_product_in_occasional_products
```

```
[121]:          count      mean       std      min      25%  \
is_FK_Advantage_product
False           1253.0  24489.063049  15124.524985  5000.0  8200.0
True            39.0   11697.589744   7586.760070  5000.0  6597.5

          50%      75%      max
is_FK_Advantage_product
False           24922.0  38106.0   49999.0
True            9600.0   13200.0   38890.0
```

```
[122]: #Are there any significant price differences between FK Advantage products and others?
discounted_fk_advantage_product_in_occasional_products=occasional_product.
groupby(["is_FK_Advantage_product"])['discounted_price'].describe()
discounted_fk_advantage_product_in_occasional_products
```

```
[122]:          count      mean       std      min      25%  \
is_FK_Advantage_product
False           1253.0  18133.883480  11894.062698  208.0  5999.0
True            39.0   9664.871795   7642.260120  1399.0  5325.0

          50%      75%      max
is_FK_Advantage_product
False           18834.0  28050.0   47620.0
True            7527.0  11006.0   38890.0
```

From above analysis we find that there are few product which are fk_advantage products in occasional category as compare to daily use product catgeory and not same discount scean is exist for occasional product category

```
[124]: #correlecion between retail_price,discounted_price,discount_percent
corr_occasional_products=occasional_product[['retail_price','discounted_price','discount_percent']]
corr()
corr_occasional_products
```

```
[124]:          retail_price  discounted_price  discount_percentage
retail_price           1.000000          0.970886         -0.199998
discounted_price        0.970886          1.000000         -0.376209
discount_percent        -0.199998         -0.376209          1.000000
```

```
[125]: sns.heatmap(corr_occasional_products,vmin=-1,vmax=1,annot=True)
```

```
[125]: <Axes: >
```



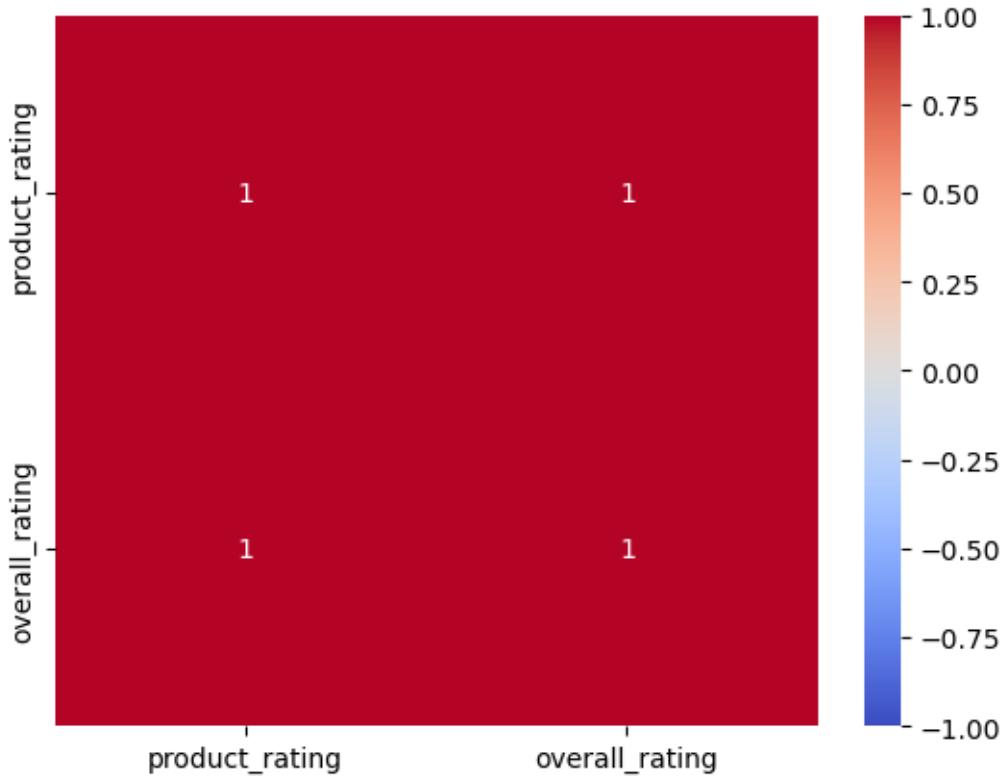
RATING ANALYSIS:

```
[127]: #how rating are related to each other in occasional product category
rating_corr_occa=occasional_product[['product_rating','overall_rating']].corr()
rating_corr_occa
```

```
[127]:      product_rating  overall_rating
product_rating          1.0          1.0
overall_rating          1.0          1.0
```

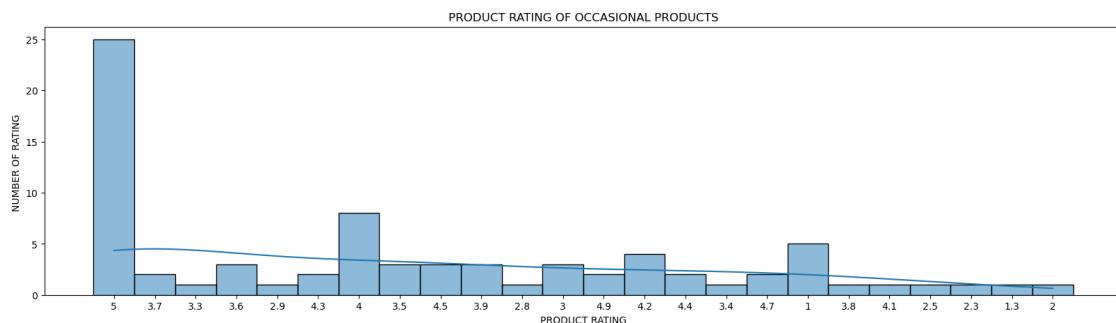
```
[128]: sns.heatmap(rating_corr,vmin=-1,vmax=1,cmap='coolwarm',annot=True)
```

```
[128]: <Axes: >
```



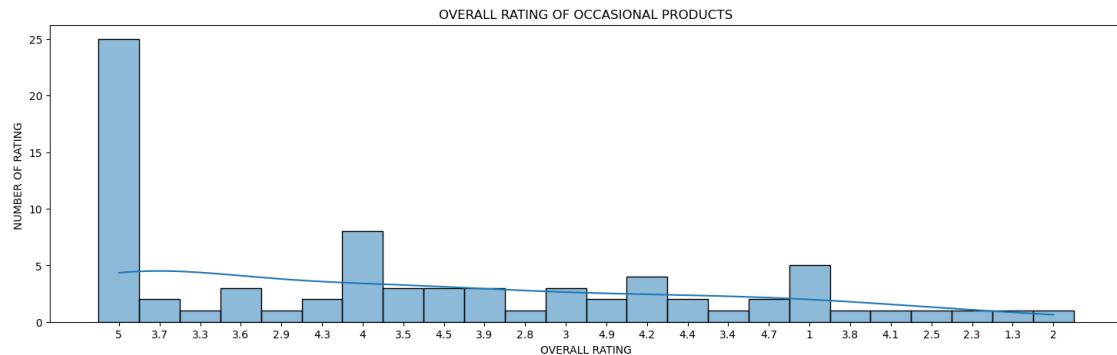
```
[129]: #how rating vary for different products of occasional product category after
    ↪removing zero rating for better results
errorfree_rating_occasional_product=occasional_product[~((occasional_product['product_rating']==
    ↪& (occasional_product['overall_rating']==0)))]
errorfree_rating_occasional_product
plt.figure(figsize=(20,5))
sns.histplot(errorfree_rating_occasional_product.product_rating,kde=True)
plt.xlabel("PRODUCT RATING")
plt.ylabel("NUMBER OF RATING ")
plt.title("PRODUCT RATING OF OCCASIONAL PRODUCTS")
```

```
[129]: Text(0.5, 1.0, 'PRODUCT RATING OF OCCASIONAL PRODUCTS')
```



```
[130]: plt.figure(figsize=(18,5))
sns.histplot(errorfree_rating_occasional_product.overall_rating,kde=True)
plt.xlabel("OVERALL RATING")
plt.ylabel("NUMBER OF RATING ")
plt.title("OVERALL RATING OF OCCASIONAL PRODUCTS")
```

```
[130]: Text(0.5, 1.0, 'OVERALL RATING OF OCCASIONAL PRODUCTS')
```

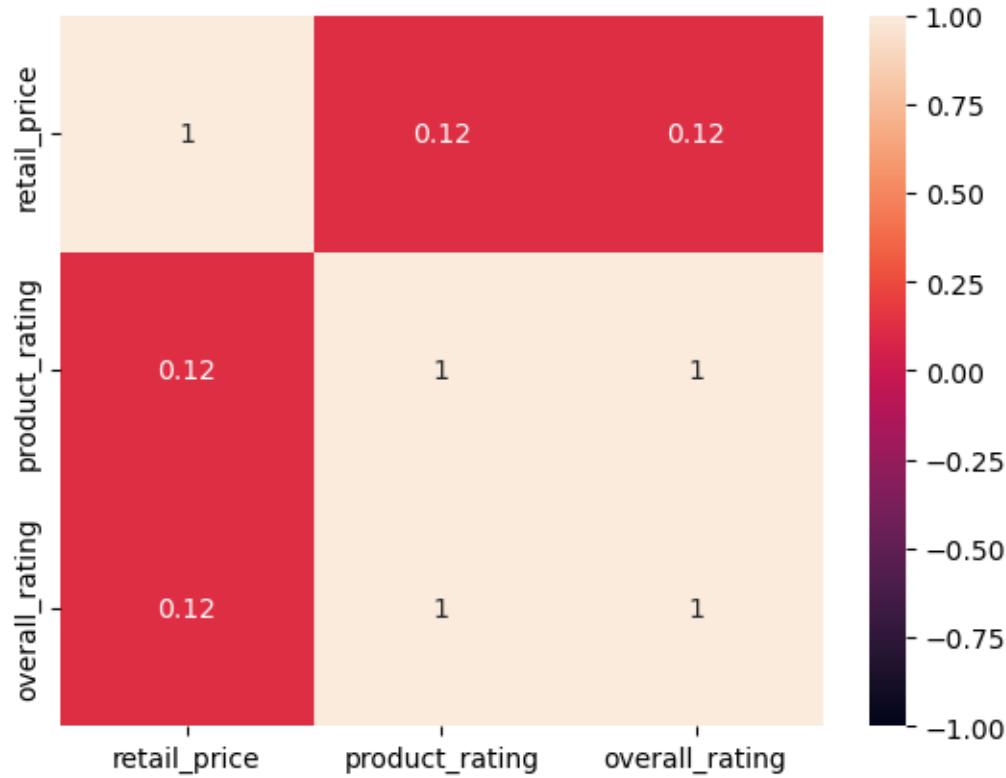


```
[131]: #how retail_price,product_rating,overall_rating related to each other
rating_corr_occasional_products=errorfree_rating_occasional_product[['retail_price','product_r
    ↪corr()
rating_corr_occasional_products
```

	retail_price	product_rating	overall_rating
retail_price	1.000000	0.123905	0.123905
product_rating	0.123905	1.000000	1.000000
overall_rating	0.123905	1.000000	1.000000

```
[132]: sns.heatmap(rating_corr_occasional_products,vmin=-1,vmax=1,annot=True)
```

[132]: <Axes: >



```
[133]: #How have product prices changed over time in occasional product catgeory?
occasional_product_price_over_year=occasional_product.
    ↪groupby(['Year','main_category'])['retail_price'].median()
occasional_product_price_over_year
```

```
[133]: Year  main_category
2015  Automotive          5500.0
      Baby Care            6211.0
      Bags, Wallets & Belts  8200.0
      Cameras & Accessories 6900.0
      Clothing              10796.0
      ...
2016  Sports & Fitness    7690.0
      Tools & Hardware      7794.0
      Toys & School Supplies 8036.0
      Watches                6950.0
      eBooks                 9359.0
Name: retail_price, Length: 62, dtype: float64
```

```
[134]: plt.figure(figsize=(42,6))
plt.xticks(rotation=90)
color=['#FF0000','#008000']
```

```

plt.ylabel("RETAIL PRICE")
plt.title("How Have Product Prices Changed Over Time?", fontsize=20)
sns.lineplot(y=occasional_product_price_over_year.
             ↪values, x=occasional_product_price_over_year.index.
             ↪get_level_values('main_category'),
             hue=occasional_product_price_over_year.index.
             ↪get_level_values('Year'), palette=color, marker='o')

```

[134]: <Axes: title={'center': 'How Have Product Prices Changed Over Time?'}, xlabel='main_category', ylabel='RETAIL PRICE'>



From above visualization we find that price of baby care,bags,wallets& belts,camera&accessories increased in 2016 as compare to 2015 in clothing and computers section price is fall down as well as for home improvement also furniture price is fall in 2016 and rest items price increased and also too many new items are added in 2016.

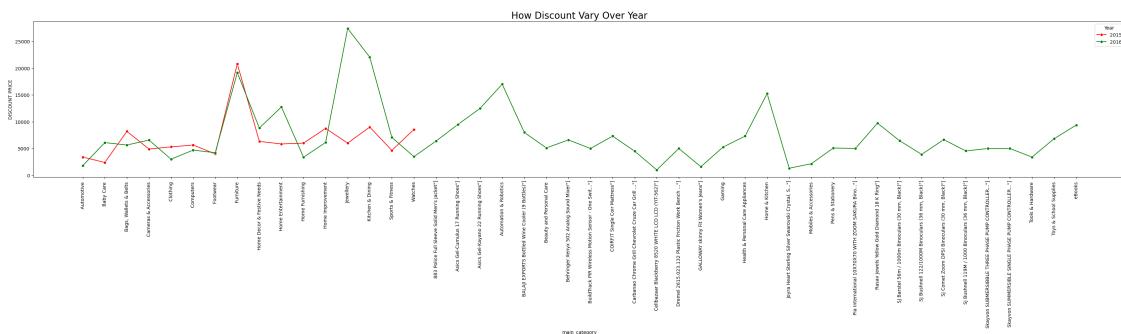
[136]: `# how discount vary over year`
`occasional_product_discount_price_over_year=occasional_product.`
`↪groupby(['Year','main_category'])['discounted_price'].median()`
`occasional_product_discount_price_over_year`

Year	main_category	discounted_price
2015	Automotive	3399.0
	Baby Care	2389.0
	Bags, Wallets & Belts	8200.0
	Cameras & Accessories	4900.0
	Clothing	5299.0
	...	
2016	Sports & Fitness	7075.0
	Tools & Hardware	3399.0
	Toys & School Supplies	6830.0
	Watches	3475.0
	eBooks	9359.0

Name: discounted_price, Length: 62, dtype: float64

```
[137]: plt.figure(figsize=(42,6))
plt.xticks(rotation=90)
color=['#FF0000', '#008000']
plt.ylabel("DISCOUNT PRICE")
plt.title("How Discount Vary Over Year", fontsize=20)
sns.lineplot(y=occasional_product_discount_price_over_year
              .values, x=occasional_product_discount_price_over_year
              .index.get_level_values('main_category'),
              hue=occasional_product_discount_price_over_year
              .index.get_level_values('Year'), palette=color, marker='o')
```

```
[137]: <Axes: title={'center': 'How Discount Vary Over Year'}, xlabel='main_category',  
      ylabel='DISCOUNT PRICE'>
```

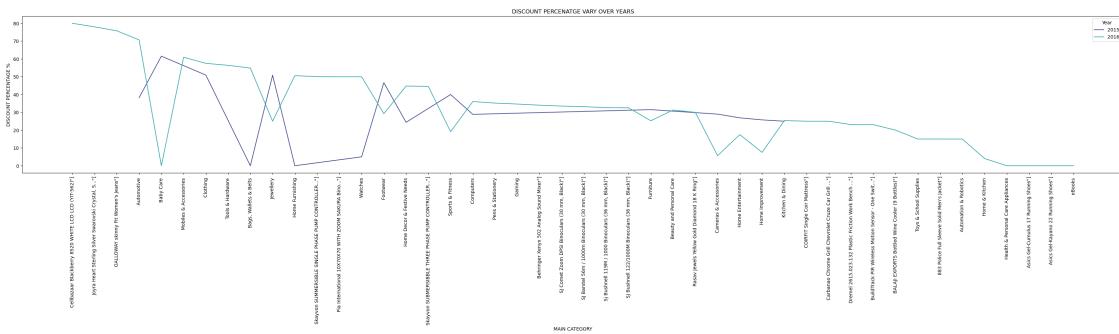


```
[138]: #How discount_percentage of main_category vary from year to year,month to month
#year to year
year_to_year_discount_change_of_occasional =occasional_product.
    ↪groupby(['Year','main_category'])['discount_percentage'].median().
    ↪sort_values(ascending=False)
year_to_year_discount_change_of_occasional
```

```
[138]: Year  main_category
2016  Cellbazaar BlackBerry 8520 WHITE LCD LCD (YIT-562)"] 80.000000
        Joyra Heart Sterling Silver Swarovski Crystal, S..."] 77.998332
        GALLOWAY skinny Fit Women's Jeans"] 75.769056
        Automotive 70.683776
2015  Baby Care 61.535807
        ...
2016  Asics Gel-Cumulus 17 Running Shoes"] 0.000000
        Asics Gel-Kayano 22 Running Shoes"] 0.000000
        Baby Care 0.000000
2015  Bags, Wallets & Belts 0.000000
2016  eBooks 0.000000
Name: discount_percentage, Length: 62, dtype: float64
```

```
[139]: plt.figure(figsize=(42,6))
plt.xticks(rotation=90)
unique_years = year_to_year_discount_change_of_occasional.index.
    ↪get_level_values('Year').nunique()
sns.lineplot(y=year_to_year_discount_change_of_occasional
    .values,x=year_to_year_discount_change_of_occasional.index.
    ↪get_level_values('main_category'),
    hue=year_to_year_discount_change_of_occasional.index.
    ↪get_level_values('Year')
    ,palette=sns.color_palette("mako", unique_years))
plt.xlabel("MAIN CATEGORY")
plt.ylabel("DISCOUNT PERCENTAGE %")
plt.title("DISCOUNT PERCENTAGE VARY OVER YEARS")
```

[139]: Text(0.5, 1.0, 'DISCOUNT PERCENTAGE VARY OVER YEARS')



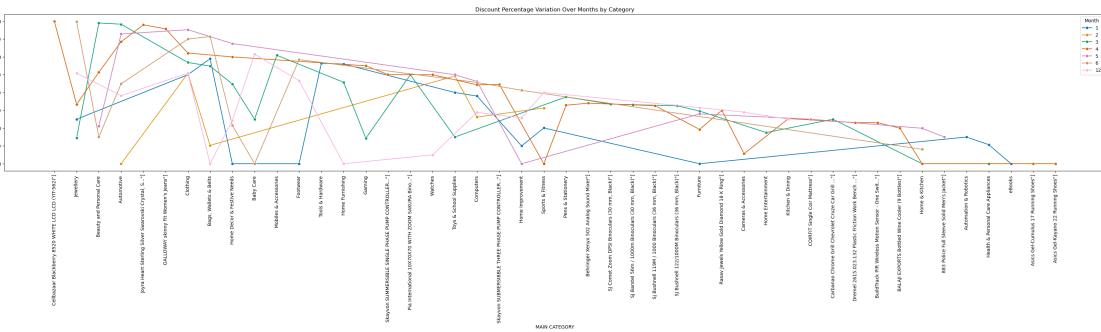
```
[140]: month_to_month_discount_change_occasional=occasional_product.
    ↪groupby(['Month','main_category'])['discount_percentage'].median().
    ↪sort_values(ascending=False)
month_to_month_discount_change_occasional
```

Month	main_category	discount_percentage
4	Cellbazaar Blackberry 8520 WHITE LCD LCD (YIT-562)"	80.000000
6	Jewellery	79.800000
3	Beauty and Personal Care	79.056681
	Automotive	78.363061
4	Joyra Heart Sterling Silver Swarovski Crystal, S..."	77.998332
	...	
	Asics Gel-Cumulus 17 Running Shoes"]	0.000000
	Asics Gel-Kayano 22 Running Shoes"]	0.000000
	Home & Kitchen	0.000000
	Sports & Fitness	0.000000
3	Health & Personal Care Appliances	0.000000

Name: discount_percentage, Length: 107, dtype: float64

```
[141]: plt.figure(figsize=(42,6))
plt.xticks(rotation=90)
unique_years = month_to_month_discount_change_occasional.index
    ↪get_level_values('Month').nunique()
sns.lineplot(y=month_to_month_discount_change_occasional.
    ↪values,x=month_to_month_discount_change_occasional.index.
    ↪get_level_values('main_category'),
        hue=month_to_month_discount_change_occasional.index.
    ↪get_level_values('Month')
            ,palette=sns.color_palette("colorblind", unique_years),marker='o')
plt.xlabel("MAIN CATEGORY")
plt.ylabel("DISCOUNT PERCENTAGE %")
plt.title("Discount Percentage Variation Over Months by Category")
```

[141]: Text(0.5, 1.0, 'Discount Percentage Variation Over Months by Category')



```
[142]: #let's check if there any seasonal sale and which product has heighest sale in
    ↪which season
season_sale_of_occasional=occasional_product.
    ↪pivot_table(values='retail_price',index=('Month','main_category'),aggfunc='sum')
season_sale_of_occasional
```

Month	main_category	retail_price
1	Automation & Robotics	19999.0
	Bags, Wallets & Belts	5000.0
	Clothing	40993.0
	Computers	86530.0
	Footwear	6210.0
...		...
12	Home Improvement	33500.0
	Jewellery	574628.0
	Kitchen & Dining	11999.0
	Sports & Fitness	7750.0

```
Watches 87574.0
```

```
[107 rows x 1 columns]
```

```
[143]: highest_sales_by_month_of_occasional = season_sale_of_occasional.  
       ↪groupby('Month').apply(lambda x: x.loc[x['retail_price'].idxmax()])  
highest_sales_by_month_of_occasional.  
       ↪sort_values(by='retail_price', ascending=False)
```

```
[143]:      retail_price  
Month  
1           23308607.0  
12          1079960.0  
3            683208.0  
4            280659.0  
6            179415.0  
5            148197.0  
2            56878.0
```

In occasional product category most products are ordered in 1,12 and in 3 months.

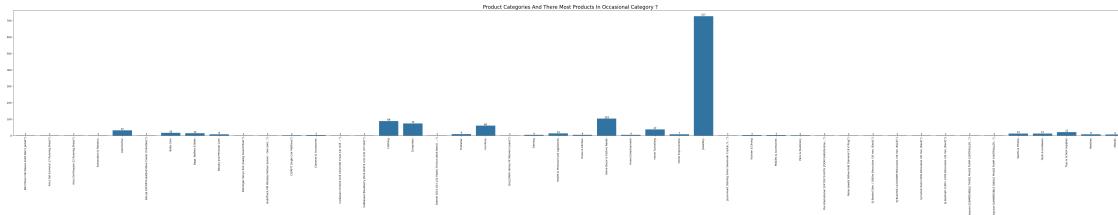
Product Popularity:

```
[146]: #Which product categories have the most products in occasional product category?  
       ↪  
most_products_occasional=occasional_product.  
       ↪groupby('main_category')['main_category'].size()  
most_products_occasional.sort_values(ascending=False)
```

```
[146]: main_category  
Jewellery 727  
Home Decor & Festive Needs 103  
Clothing 88  
Computers 74  
Furniture 60  
Home Furnishing 37  
Automotive 32  
Toys & School Supplies 21  
Baby Care 16  
Bags, Wallets & Belts 14  
Health & Personal Care Appliances 13  
Tools & Hardware 12  
Sports & Fitness 12  
Footwear 9  
Beauty and Personal Care 8  
Watches 8  
Home Improvement 7  
eBooks 6
```

Gaming	4
Home Entertainment	4
Home & Kitchen	4
Cameras & Accessories	3
Kitchen & Dining	3
Mobiles & Accessories	3
COIRFIT Single Coir Mattress"]	2
Pens & Stationery	2
GALLOWAY skinny Fit Women's Jeans"]	1
SJ Bushnell 122/1000M Binoculars (36 mm, Black)"]	1
Asics Gel-Kayano 22 Running Shoes"]	1
Automation & Robotics	1
BALAJI EXPORTS Bottled Wine Cooler (9 Bottles)"]	1
Behringer Xenyx 502 Analog Sound Mixer"]	1
Skayvon SUMMERSIBLE SINGLE PHASE PUMP CONTROLLER..."]	1
Skayvon SUBMERSIBBLE THREE PHASE PUMP CONTROLLER..."]	1
Sj Bushnell 119M / 1000 Binoculars (36 mm, Black)"]	1
SJ Comet Zoom DPSI Binoculars (30 mm, Black)"]	1
Pia International 10X70X70 WITH ZOOM SAKURA Bino..."]	1
SJ Barstel 56m / 1000m Binoculars (30 mm, Black)"]	1
Rasav Jewels Yellow Gold Diamond 18 K Ring"]	1
Dremel 2615.023.132 Plastic Friction Work Bench ..."]	1
Joyra Heart Sterling Silver Swarovski Crystal, S..."]	1
BuildTrack PIR Wireless Motion Sensor - One Swit..."]	1
Carbanao Chrome Grill Chevrolet Cruze Car Grill ..."]	1
Cellbazaar Blackberry 8520 WHITE LCD LCD (YIT-562)"]	1
Asics Gel-Cumulus 17 Running Shoes"]	1
883 Police Full Sleeve Solid Men's Jacket"]	1
Name: main_category, dtype: int64	

```
[147]: plt.figure(figsize=(80,9))
plt.xticks(rotation=90)
ax=sns.barplot(y=most_products_occasional.values,x=most_products_occasional.
                ↪index)
plt.title("Product Categories And There Most Products In Occasional Category ?"
                ↪,fontsize=20)
for container in ax.containers:
    ax.bar_label(container,fmt='%.d')
```



Occasionally Jewellery, home decor, clothing, computers, furniture are mainly buy

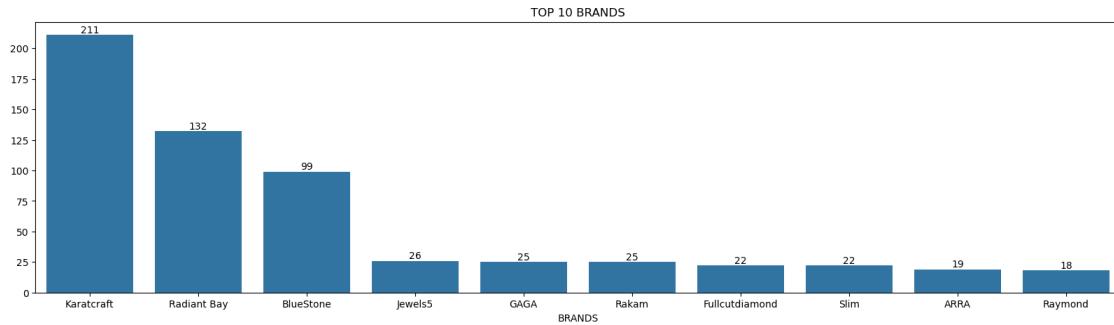
```
[149]: #product popularity over time
product_popularity_occasional=occasional_product.
    ↪groupby(['Year','main_category'])['pid'].count()
product_popularity_occasional
```

```
[149]: Year  main_category
2015  Automotive          14
      Baby Care            11
      Bags, Wallets & Belts  1
      Cameras & Accessories 1
      Clothing              23
      ..
2016  Sports & Fitness      11
      Tools & Hardware       12
      Toys & School Supplies  21
      Watches                1
      eBooks                 6
Name: pid, Length: 62, dtype: int64
```

```
[150]: #What are the top 10 most frequently occurring brands in occasional product
    ↪category?
top_10_brands_occasional=occasional_product.groupby('brand')['brand'].
    ↪value_counts().nlargest(10)
top_10_brands_occasional
```

```
[150]: brand
Karatcraft        211
Radiant Bay       132
BlueStone          99
Jewels5           26
GAGA              25
Rakam              25
Fullcutdiamond    22
Slim               22
ARRA               19
Raymond            18
Name: count, dtype: int64
```

```
[151]: plt.figure(figsize=(20,5))
ax=sns.barplot(y=top_10_brands_occasional.values,x=top_10_brands_occasional.
    ↪index.get_level_values('brand'))
plt.xlabel("BRANDS")
plt.title('TOP 10 BRANDS')
for container in ax.containers:
    ax.bar_label(container,fmt='%d')
```



Category Exploration:

```
[153]: #How many distinct product categories exist?
main_category_occasional_products=occasional_product.
    ↪groupby('main_category')[‘main_category’].nunique()
main_category_occasional_products
```

```
[153]: main_category
883 Police Full Sleeve Solid Men's Jacket"] 1
Asics Gel-Cumulus 17 Running Shoes"] 1
Asics Gel-Kayano 22 Running Shoes"] 1
Automation & Robotics 1
Automotive 1
BALAJI EXPORTS Bottled Wine Cooler (9 Bottles)"] 1
Baby Care 1
Bags, Wallets & Belts 1
Beauty and Personal Care 1
Behringer Xenyx 502 Analog Sound Mixer"] 1
BuildTrack PIR Wireless Motion Sensor - One Swit..."] 1
COIRFIT Single Coir Mattress"] 1
Cameras & Accessories 1
Carbanao Chrome Grill Chevrolet Cruze Car Grill ..."] 1
Cellbazaar Blackberry 8520 WHITE LCD LCD (YIT-562)"] 1
Clothing 1
Computers 1
Dremel 2615.023.132 Plastic Friction Work Bench ..."] 1
Footwear 1
Furniture 1
GALLOWAY skinny Fit Women's Jeans"] 1
Gaming 1
Health & Personal Care Appliances 1
Home & Kitchen 1
Home Decor & Festive Needs 1
Home Entertainment 1
Home Furnishing 1
```

Home Improvement	1
Jewellery	1
Jyora Heart Sterling Silver Swarovski Crystal, S..."]	1
Kitchen & Dining	1
Mobiles & Accessories	1
Pens & Stationery	1
Pia International 10X70X70 WITH ZOOM SAKURA Bino..."]	1
Rasav Jewels Yellow Gold Diamond 18 K Ring"]	1
SJ Barstel 56m / 1000m Binoculars (30 mm, Black)"]	1
SJ Bushnell 122/1000M Binoculars (36 mm, Black)"]	1
SJ Comet Zoom DPSI Binoculars (30 mm, Black)"]	1
Sj Bushnell 119M / 1000 Binoculars (36 mm, Black)"]	1
Skayvon SUBMERSIBLE THREE PHASE PUMP CONTROLLER..."]	1
Skayvon SUMMERSIBLE SINGLE PHASE PUMP CONTROLLER..."]	1
Sports & Fitness	1
Tools & Hardware	1
Toys & School Supplies	1
Watches	1
eBooks	1

Name: main_category, dtype: int64

main_category_occasional_products.value_counts() is use to know the count of different category in occsional product category There are total 46 different categories exist in this category

[155]: *#What are the most common product subcategories within each category of occasional product?*

```
occasional_most_common_product_first_category=occasional_product.
    ↪groupby(['main_category'])['secondary_category'].nunique().
    ↪sort_values(ascending=False)
occasional_most20_first_category=occasional_most_common_product_first_category.
    ↪head(20)
occasional_most20_first_category
```

[155]: main_category

Jewellery	10
Furniture	6
Home Decor & Festive Needs	6
Computers	5
Home Furnishing	4
Home Improvement	4
Beauty and Personal Care	4
Baby Care	4
Sports & Fitness	4
Toys & School Supplies	4
Bags, Wallets & Belts	3
Automotive	3
Clothing	3

```

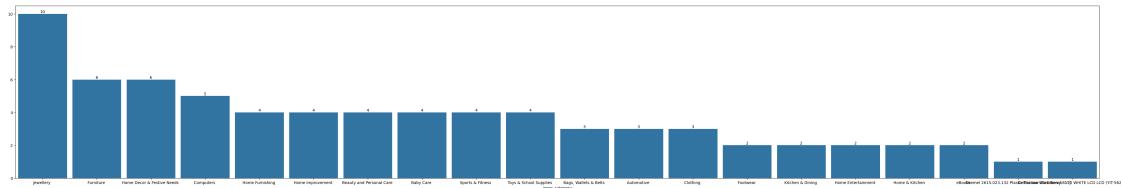
Footwear                                     2
Kitchen & Dining                           2
Home Entertainment                          2
Home & Kitchen                             2
eBooks                                      2
Dremel 2615.023.132 Plastic Friction Work Bench ..."] 1
Cellbazaar Blackberry 8520 WHITE LCD LCD (YIT-562")] 1
Name: secondary_category, dtype: int64

```

```

[156]: plt.figure(figsize=(50,8))
ax=sns.barplot(y=occasional_most20_first_category.
                 ↪values,x=occasional_most20_first_category.index.
                 ↪get_level_values('main_category')
                 ,width=0.90)
for container in ax.containers:
    ax.bar_label(container, fmt='%.d')

```



```

[157]: occasional_most_common_product_second_category=occasional_product.
         ↪groupby(['secondary_category'])['tertiary_category'].nunique().
         ↪sort_values(ascending=False)
occasional_most20_second_category=occasional_most_common_product_second_category.
         ↪head(20)
occasional_most20_second_category

```

```

[157]: secondary_category
Action Figures                      10
Showpieces                         8
Wrist Watches                      7
Accessories & Spare parts        6
Remote Control Toys                5
Men's Clothing                      5
Women's Clothing                   4
Wall Decor & Clocks               4
Bangles, Bracelets & Armlets     4
Rings                               4
Computer Peripherals               3
Men's Footwear                     3
Gaming Accessories                 3

```

```

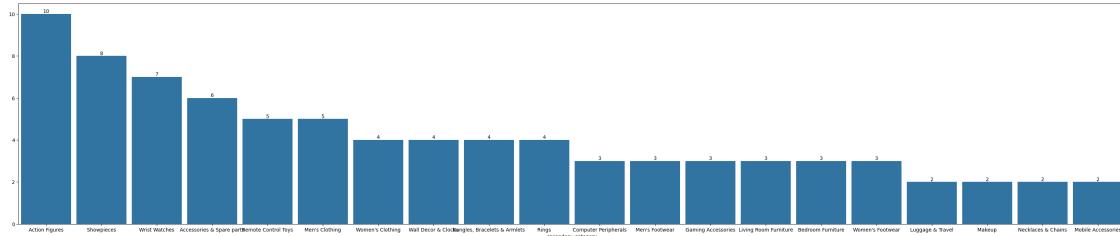
Living Room Furniture      3
Bedroom Furniture          3
Women's Footwear          3
Luggage & Travel          2
Makeup                      2
Necklaces & Chains          2
Mobile Accessories          2
Name: tertiary_category, dtype: int64

```

```

[158]: plt.figure(figsize=(40,8))
ax=sns.barplot(y=occasional_most20_second_category.
                 ↪values,x=occasional_most20_second_category.index.
                 ↪get_level_values('secondary_category')
                 ,width=0.90)
for container in ax.containers:
    ax.bar_label(container, fmt='.%d')

```



```

[159]: occasional_most_common_product_third_category=occasional_product.
         ↪groupby(['tertiary_category'])['quaternary_category'].nunique().
         ↪sort_values(ascending=False)
occasional_most20_third_category=occasional_most_common_product_third_category.
         ↪nlargest(20)
occasional_most20_third_category

```

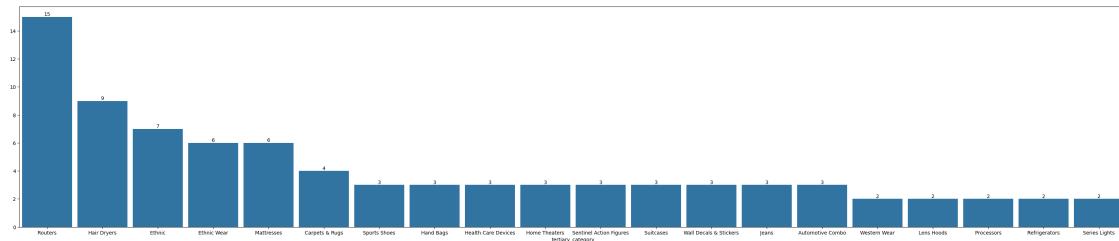
```

[159]: tertiary_category
Routers                      15
Hair Dryers                   9
Ethnic                        7
Ethnic Wear                   6
Mattresses                     6
Carpets & Rugs                4
Sports Shoes                   3
Hand Bags                      3
Health Care Devices            3
Home Theaters                  3
Sentinel Action Figures         3
Suitcases                      3

```

```
Wall Decals & Stickers      3
Jeans                      3
Automotive Combo            3
Western Wear                 2
Lens Hoods                  2
Processors                  2
Refrigerators                2
Series Lights                2
Name: quaternary_category, dtype: int64
```

```
[160]: plt.figure(figsize=(40,8))
ax=sns.barplot(y=occasional_most20_third_category.
                 ↪values,x=occasional_most20_third_category.index.
                 ↪get_level_values('tertiary_category')
                 ,width=0.90)
for container in ax.containers:
    ax.bar_label(container, fmt='%d')
```



```
[161]: # What are the most common product subcategories within each category?
level_1_category_occasional=occasional_product.
    ↪groupby(['main_category','secondary_category']).size().
    ↪sort_values(ascending=False)
level_1_category_occasional
```

main_category	secondary_category	Count
Jewellery	Rings"]	668
Computers	Network Components	62
Clothing	Men's Clothing	50
Home Decor & Festive Needs	Showpieces	45
	Wall Decor & Clocks	39
	...	
Asics Gel-Cumulus 17 Running Shoes"]	Garden & Leisure	1
Home Entertainment	None	1
Home Furnishing	Audio Players	1
Home Decor & Festive Needs	Living	1
Length: 98, dtype: int64	Religion & Devotion	1

```
[162]: level_2_category_occasional=occasional_product.
    ↪groupby(['secondary_category','tertiary_category']).size().
    ↪sort_values(ascending=False)
level_2_category_occasional
```

secondary_category	tertiary_category	
Rings"]	None	668
Network Components	Routers	61
Living Room	Sofas & Sectionals"]	36
Showpieces	Ethnic	31
Bed Linen	Blankets, Quilts & Dohars"]	25
	...	
Computer Components	Motherboards	1
Office Equipments	Home Security Cameras	1
	Biometric Devices	1
Computer Peripherals	CPU	1
Academic Texts	Humanities	1
Length: 156, dtype: int64		


```
[163]: level_3_category_occasional=occasional_product.
    ↪groupby(['tertiary_category','quaternary_category']).size().
    ↪sort_values(ascending=False)
level_3_category_occasional
```

tertiary_category	quaternary_category	
None	None	
700		
Sofas & Sectionals"]	None	
36		
Paintings	GAGA Painting	
25		
Blankets, Quilts & Dohars"]	None	
25		
Car Interior & Exterior	Car Interio	
23		
	...	
Hand Tools	Wrenches & Set	
1		
Hasbro Action Figures	Hasbro Star Wars Clone Wars Exclusive Vehicle	
Ac... 1		
Headphones	Head Kik Headphone	
1		
	Radius Headphone	
1		
Jeans	Gas Jeans	
1		
Length: 219, dtype: int64		

Conclusion: From above analysis we find that there is most sales in “Jewellary” category in which subcategory “rings” is on top. Second most purchased thing is ‘computer’ and in computer ‘Network Component’ are mostly ordered and in this category (Network Compomemt) mostly Routers are placed.

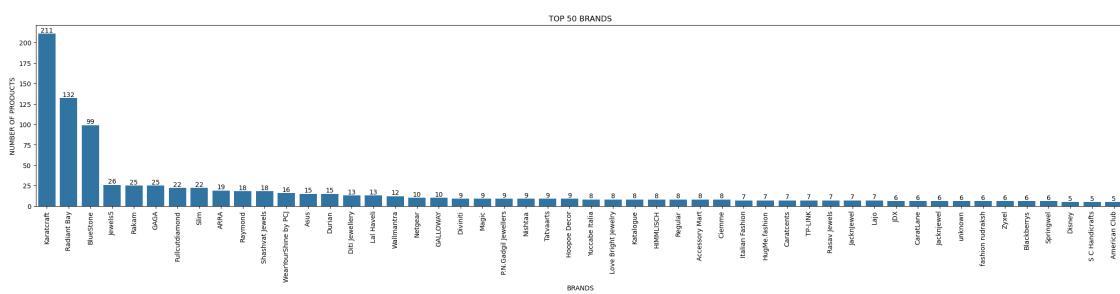
Brand Analysis:

```
[166]: #Which brands have the most products in the occasional product category?  
most_brands_occasional=occasional_product.groupby('brand').size()  
top50_most_brands_occasional=most_brands_occasional.  
    ↪sort_values(ascending=False).head(50)  
top50_most_brands_occasional
```

```
[166]: brand  
Karatcraft 211  
Radiant Bay 132  
BlueStone 99  
Jewels5 26  
Rakam 25  
GAGA 25  
Fullcutdiamond 22  
Slim 22  
ARRA 19  
Raymond 18  
Shashvat Jewels 18  
WearYourShine by PCJ 16  
Asus 15  
Durian 15  
Diti Jewellery 13  
Lal Haveli 13  
Wallmantra 12  
Netgear 10  
GALLOWAY 10  
Diviniti 9  
Magic 9  
P.N.Gadgil Jewellers 9  
Nishtaa 9  
Tatvaarts 9  
Hoopoe Decor 9  
Yuccabe Italia 8  
Love Bright Jewelry 8  
Katalogue 8  
HIMMLISCH 8  
Regular 8  
Accessory Mart 8  
Ciemme 8  
Italian Fashion 7  
HugMe.fashion 7
```

```
Caratcents          7
TP-LINK            7
Rasav Jewels       7
JacknJewel         7
Lajo                7
JDX                 6
CaratLane          6
Jacknjewel         6
unknown             6
fashion rudraksh  6
Zyxel               6
Blackberrys         6
Springwel           6
Disney              5
S C Handicrafts    5
American Club       5
dtype: int64
```

```
[167]: plt.figure(figsize=(30,5))
plt.xticks(rotation=90)
ax=sns.barplot(x=top50_most_brands_occasional.
                 index,y=top50_most_brands_occasional.values)
plt.xlabel('BRANDS')
plt.ylabel('NUMBER OF PRODUCTS')
plt.title("TOP 50 BRANDS")
for x in ax.containers:
    ax.bar_label(x,fmt='%d')
```



```
[168]: most_brands.occasional.nsmallest(50)
```

[168]:	brand	1
	3D Mat	1
	883 Police	1
	@home	1
	ASHAWAY	1
	ASUS	1

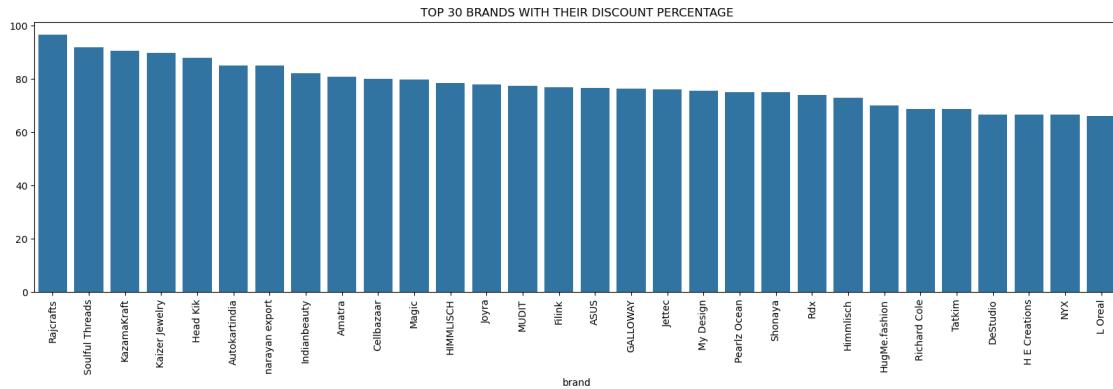
Aakrati	1
Aimedu Toy	1
Ajay	1
Amatra	1
Ambika Lahenga	1
Araanz by Tribhovandas Bhimji Zaveri Delhi	1
Autokartindia	1
Avsar	1
Axcess	1
BACHELORS N BEYOND	1
BALAJI EXPORTS	1
BGS	1
Babyliss	1
Balaji	1
Barbar	1
Behringer	1
Beyblade	1
Big Mike s	1
Black	1
Black & Decker	1
Boss	1
Brahaan BLUE TAG	1
BuildTrack	1
Caratify	1
Carbanao	1
Cellbazaar	1
Collectible India	1
Comfort Couch	1
Conair	1
Craze	1
D-LINK	1
DC Universe	1
DH-Discovery	1
DeStudio	1
Demira Jewels	1
Discovery Toys	1
Doctor Who	1
Dremel	1
EGON	1
ELITE STATUES	1
Filink	1
Flyfilms	1
Furnstyl	1
G-Luck	1
GEMKOLABWELL	1
dtype: int64	

```
[169]: #Are there any brands that consistently offer higher discounts?
#Finding which brands give what percentage of discount to their product in occasional product category
occasional_higher_discount_brands=occasional_product.
    ↪groupby('brand')['discount_percentage'].mean()
top30occasional_higher_discount_brands=occasional_higher_discount_brands.
    ↪nlargest(30)
top30occasional_higher_discount_brands
```

```
[169]: brand
Rajcrafts          96.533333
Soulful Threads   91.952663
KazamaKraft        90.565618
Kaizer Jewelry     89.616918
Head Kik           88.018802
Autokartindia     85.014169
narayan export     85.007084
Indianbeauty       82.020000
Amatra             80.725818
Cellbazaar         80.000000
Magic               79.800000
HIMMLISCH          78.363061
Joyra               77.998332
MUDIT               77.466063
Filink              76.833333
ASUS                76.679447
GALLOWAY            76.223670
Jettec              76.000000
My Design           75.591762
Pearlz Ocean        75.016758
Shonaya             75.000000
Rdx                 73.997400
Himmlisch           72.867553
HugMe.fashion       70.000289
Richard Cole        68.758595
Tatkim               68.758595
DeStudio             66.677780
H E Creations        66.666667
NYX                 66.472515
L Oreal              66.160700
Name: discount_percentage, dtype: float64
```

```
[170]: plt.figure(figsize=(20,5))
plt.xticks(rotation=90)
sns.barplot(x=top30occasional_higher_discount_brands.
    ↪index,y=top30occasional_higher_discount_brands.values)
plt.title('TOP 30 BRANDS WITH THEIR DISCOUNT PERCENTAGE',loc='center')
```

```
[170]: Text(0.5, 1.0, 'TOP 30 BRANDS WITH THEIR DISCOUNT PERCENTAGE')
```



```
[171]: #To check the brands consistency for discount_percentage we can find standard deviation to know more.
```

```
occasional_consistent_higher_discount_brands=occasional_product.
```

```
↳groupby('brand')['discount_percentage'].std()
```

```
top30_occasional_consistent_higher_discount_brands=occasional_consistent_higher_discount_brands
```

```
↳sort_values(ascending=False).head(30)
```

```
top30_occasional_consistent_higher_discount_brands
```

```
[171]: brand
```

JDK NOVELTY	38.157100
Asus	23.114844
Johareez	21.849643
Weldecor	21.216563
Katalogue	18.916028
Zyxel	18.881663
Wallmantra	15.139110
Lajo	14.224484
Belkin	14.113232
TP-LINK	14.010858
Gravolite	13.815858
S C Handicrafts	12.870008
Love Bright Jewelry	12.836812
MCB Exports	12.439157
ARRA	12.054285
4D Mats	11.554344
Slim	11.085936
Linksys	10.717202
Disney	10.539343
JacknJewel	10.247241
NEVI	10.100080
YugshaJewels	9.707910

```

Regular           9.451058
Netgear          8.772180
Speedwav         7.859713
BlueStone        7.541389
Italian Fashion  7.224881
Intel            6.865800
Handiana         6.543629
Clara            6.328591
Name: discount_percentage, dtype: float64

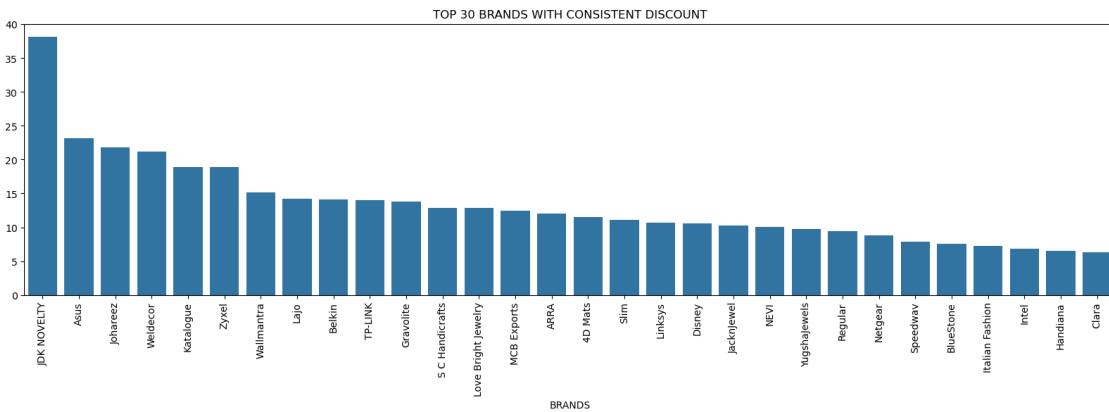
```

```

[172]: plt.figure(figsize=(20,5))
plt.xticks(rotation=90)
sns.barplot(x=top30_occasional_consistent_higher_discount_brands.
             index,y=top30_occasional_consistent_higher_discount_brands.values)
plt.title('TOP 30 BRANDS WITH CONSISTENT DISCOUNT',loc='center')
plt.xlabel('BRANDS')

```

```
[172]: Text(0.5, 0, 'BRANDS')
```



By brand analysis we find that rajcraft brand provide heighest discount with 96.53% on their product but if we try to figure out which brand provide consistent discount we get JDK NOVELTY.

LUXURY PRODUCTS ANALYSIS:

```
[174]: luxury_product.sample(10)
```

```

[174]:          uniq_id \
108    710ed5f2393a4b9e8823aa0029f71f93
11182  c6461a2a84cf121d51294d3878c81686
9782   022a7f08d051e2cfca917ef0a6d3a19a
11312   43e444307157aca6bb055c4bada67928
11646   e5a97beaed03ee8ece9a5749a431e985

```

```

11702 5f8e0c25e2915bc2383e60ad02b1e4aa
11542 4e3e2624f30da4c66c52de92f9b88f18
11631 3a2546675bc399953779e58d84d56650
134   08452abdadb3db1e686b94a9c52fc7b6
11433 e3e08f67f1e0b5724e71c6845a7f0ffa

```

	product_name	pid	\
108	TAG Heuer CAU1116.BA0858 Formula 1 Analog Watc...	WATEA6UGFGKZKDJC	
11182	ARRA Solid Wood 5 Seater Sofa	SOFEDVYZAHXWYTGW	
9782	The Curve Full Sleeve Solid Men's Quilted Jacket	JCKEAEA7REDBGJVV	
11312	ARRA Solid Wood 4 Seater Sofa	SOFEDVYZSG7DTMYG	
11646	Durian Fabric 3 Seater Sofa	SOFEE7ACTFUYMFGG	
11702	Durian Helena Fabric 3 Seater Sofa	SOFE8HPYX66KMC3A	
11542	Durian Clinton/A/3 Leather 3 Seater Sofa	SOFE94HW8TPTGPNX	
11631	Durian Leather 2 Seater Sofa	SOFEE7ACBVAGKYUV	
134	Cartier W6701005 Analog Watch - For Boys, Men	WATEAK8SE2RUK7GM	
11433	ARRA Solid Wood 6 Seater Sofa	SOFEDVYZTDGWBSMN	

	retail_price	discounted_price	is_FK_Advantage_product	product_rating	\
108	107750.0	107750.0	False	0	
11182	51400.0	51400.0	False	0	
9782	69999.0	48999.0	False	0	
11312	57500.0	57500.0	False	0	
11646	73800.0	47970.0	False	0	
11702	108900.0	70785.0	False	0	
11542	73500.0	47775.0	False	5	
11631	250500.0	162825.0	False	0	
134	201000.0	201000.0	False	0	
11433	68400.0	68400.0	False	0	

	overall_rating	brand	Year	Month	Days	main_category	\
108	0	Italian Fashion	2015	12	4	Watches	
11182	0	ARRA	2015	12	31	Furniture	
9782	0	Regular	2016	1	7	Clothing	
11312	0	ARRA	2015	12	31	Furniture	
11646	0	Durian	2015	12	31	Furniture	
11702	0	Durian	2015	12	31	Furniture	
11542	5	Durian	2015	12	31	Furniture	
11631	0	Durian	2015	12	31	Furniture	
134	0	Italian Fashion	2015	12	4	Watches	
11433	0	ARRA	2015	12	31	Furniture	

	secondary_category	tertiary_category	quaternary_category	\
108	Wrist Watches	TAG Heuer Wrist Watches"]	None	
11182	Living Room	Sofas & Sectionals"]	None	
9782	Men's Clothing	Winter & Seasonal Wear	Jacket	
11312	Living Room	Sofas & Sectionals"]	None	

11646	Living Room	Sofas & Sectionals"]	None
11702	Living Room	Sofas & Sectionals"]	None
11542	Living Room	Sofas & Sectionals"]	None
11631	Living Room	Sofas & Sectionals"]	None
134	Wrist Watches	Cartier Wrist Watches"]	None
11433	Living Room	Sofas & Sectionals"]	None
		discount_percentage	product_category
108	0.000000	Luxury Product	
11182	0.000000	Luxury Product	
9782	30.000429	Luxury Product	
11312	0.000000	Luxury Product	
11646	35.000000	Luxury Product	
11702	35.000000	Luxury Product	
11542	35.000000	Luxury Product	
11631	35.000000	Luxury Product	
134	0.000000	Luxury Product	
11433	0.000000	Luxury Product	

Price analysis of luxury products:

[176]: `luxury_product.describe()`

	retail_price	discounted_price	Year	Month	Days	\
count	30.000000	30.000000	30.000000	30.000000	30.000000	
mean	115589.266667	91953.666667	2015.133333	10.733333	26.100000	
std	101329.129824	99072.452163	0.345746	3.300296	9.621635	
min	51400.000000	35295.000000	2015.000000	1.000000	4.000000	
25%	66525.000000	48958.500000	2015.000000	12.000000	31.000000	
50%	79800.000000	61320.000000	2015.000000	12.000000	31.000000	
75%	108900.000000	100600.000000	2015.000000	12.000000	31.000000	
max	571230.000000	571230.000000	2016.000000	12.000000	31.000000	
	discount_percentage					
count	30.000000					
mean	22.166698					
std	17.192309					
min	0.000000					
25%	0.000000					
50%	35.000000					
75%	35.000000					
max	37.500523					

[177]: `#What are the average retail and discounted prices of products?`
`average_retail_price=luxury_product.groupby('main_category')['retail_price'].`
`mean()`
`average_retail_price`

```
[177]: main_category
Clothing           69999.000000
Furniture          98747.826087
Mobiles & Accessories 116292.000000
Pens & Stationery   65103.500000
Watches            293326.666667
Name: retail_price, dtype: float64
```

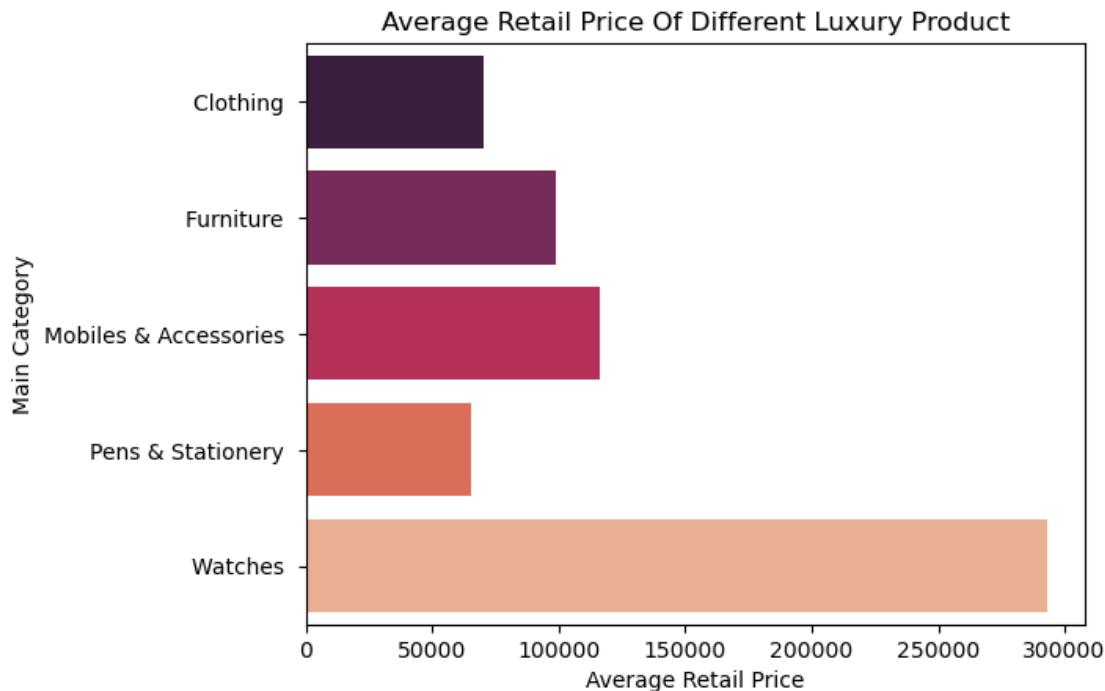
```
[178]: sns.barplot(y=average_retail_price.index,x=average_retail_price.
                   values,palette='rocket')
plt.title("Average Retail Price Of Different Luxury Product")
plt.xlabel("Average Retail Price")
plt.ylabel('Main Category')
```

/var/folders/vc/4h3d9s6d639f63qzbhd805b80000gn/T/ipykernel_1158/1067814051.py:1:
FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `y` variable to `hue` and set `legend=False` for the same effect.

```
sns.barplot(y=average_retail_price.index,x=average_retail_price.values,palette
='rocket')
```

```
[178]: Text(0, 0.5, 'Main Category')
```



```
[179]: # DISCOUNTES PRICE
average_discounted_price=luxury_product.
    ↪groupby('main_category')['discounted_price'].mean()
average_discounted_price
```

```
[179]: main_category
Clothing           48999.000000
Furniture          70954.782609
Mobiles & Accessories 116292.000000
Pens & Stationery    40689.500000
Watches            293326.666667
Name: discounted_price, dtype: float64
```

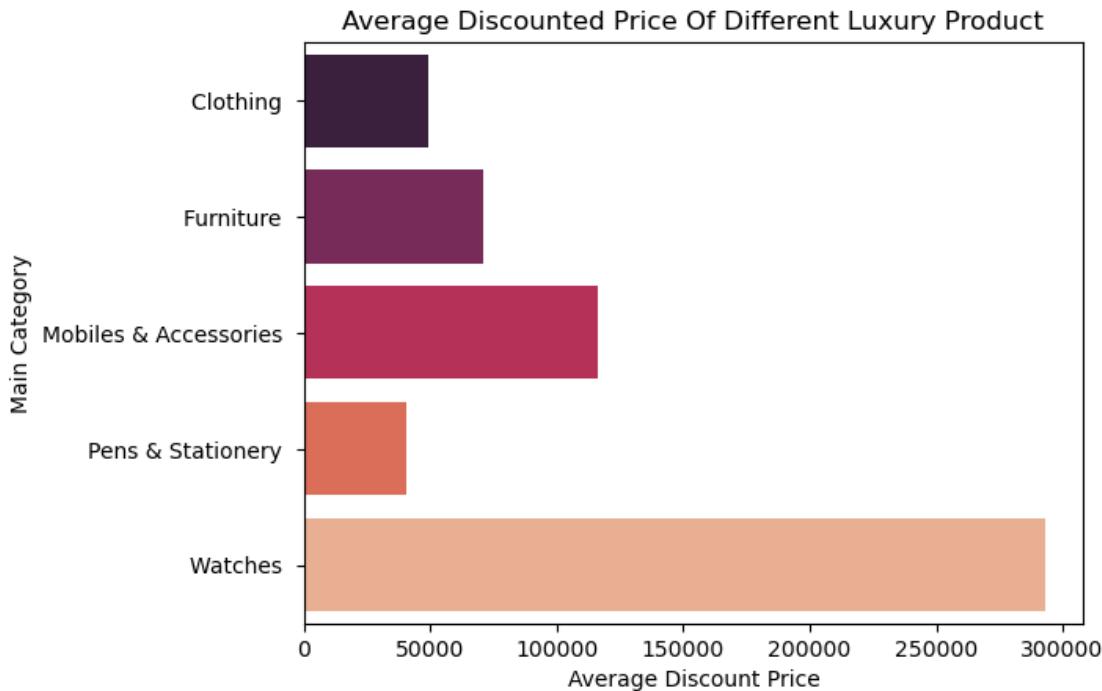
```
[180]: sns.barplot(y=average_discounted_price.index,x=average_discounted_price.
    ↪values,palette='rocket')
plt.title("Average Discounted Price Of Different Luxury Product")
plt.xlabel("Average Discount Price")
plt.ylabel('Main Category')
```

/var/folders/vc/4h3d9s6d639f63qzbhd805b80000gn/T/ipykernel_1158/1726551826.py:1:
FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `y` variable to `hue` and set `legend=False` for the same effect.

```
    sns.barplot(y=average_discounted_price.index,x=average_discounted_price.values
,palette='rocket')
```

```
[180]: Text(0, 0.5, 'Main Category')
```



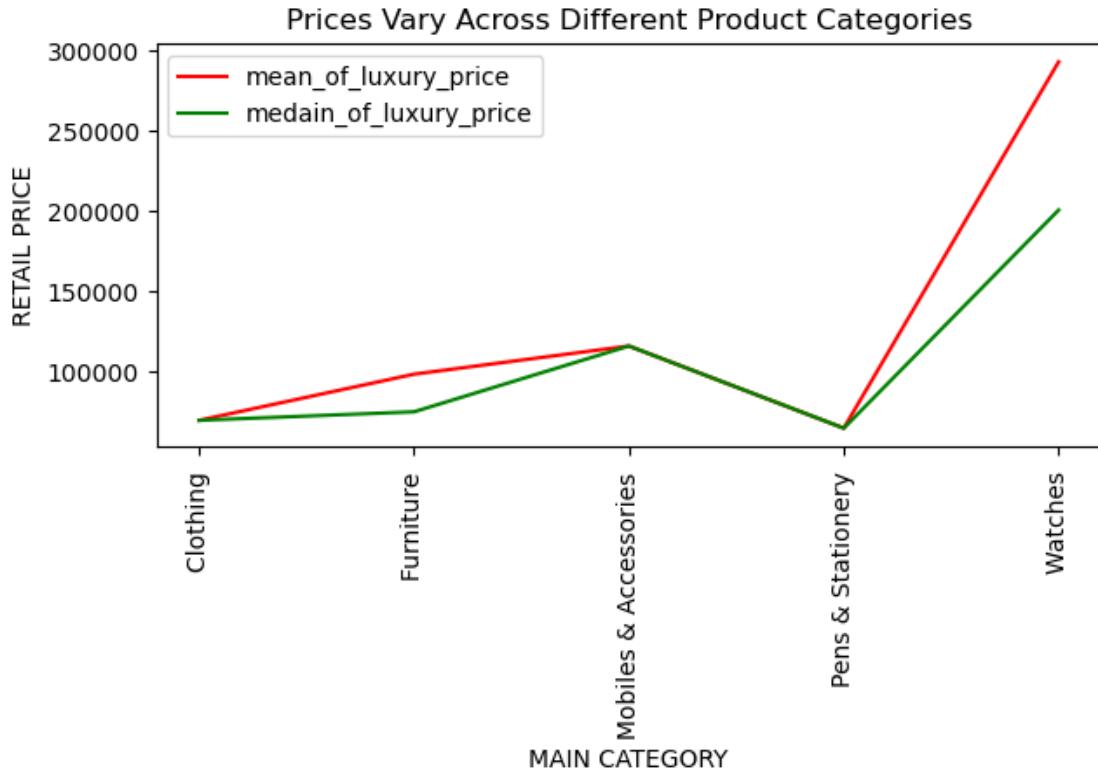
```
[181]: #How do prices vary across different product categories?
mean_category_price_luxury=luxury_product.
    ↪groupby('main_category')['retail_price'].mean()
median_category_price_luxury=luxury_product.
    ↪groupby('main_category')['retail_price'].median()
mean_category_price_luxury
median_category_price_luxury
```

```
[181]: main_category
Clothing           69999.0
Furniture          75300.0
Mobiles & Accessories 116292.0
Pens & Stationery   65103.5
Watches            201000.0
Name: retail_price, dtype: float64
```

```
[182]: plt.figure(figsize=(7,3))
plt.xticks(rotation=90)
sns.lineplot(x=mean_category_price_luxury.
    ↪index,y=mean_category_price_luxury,color='red',label="mean_of_luxury_price")
sns.lineplot(x=median_category_price_luxury.
    ↪index,y=median_category_price_luxury,color='green',label="median_of_luxury_price")
plt.xlabel("MAIN CATEGORY")
plt.ylabel("RETAIL PRICE")
```

```
plt.legend(loc='upper left')
plt.title("Prices Vary Across Different Product Categories")
```

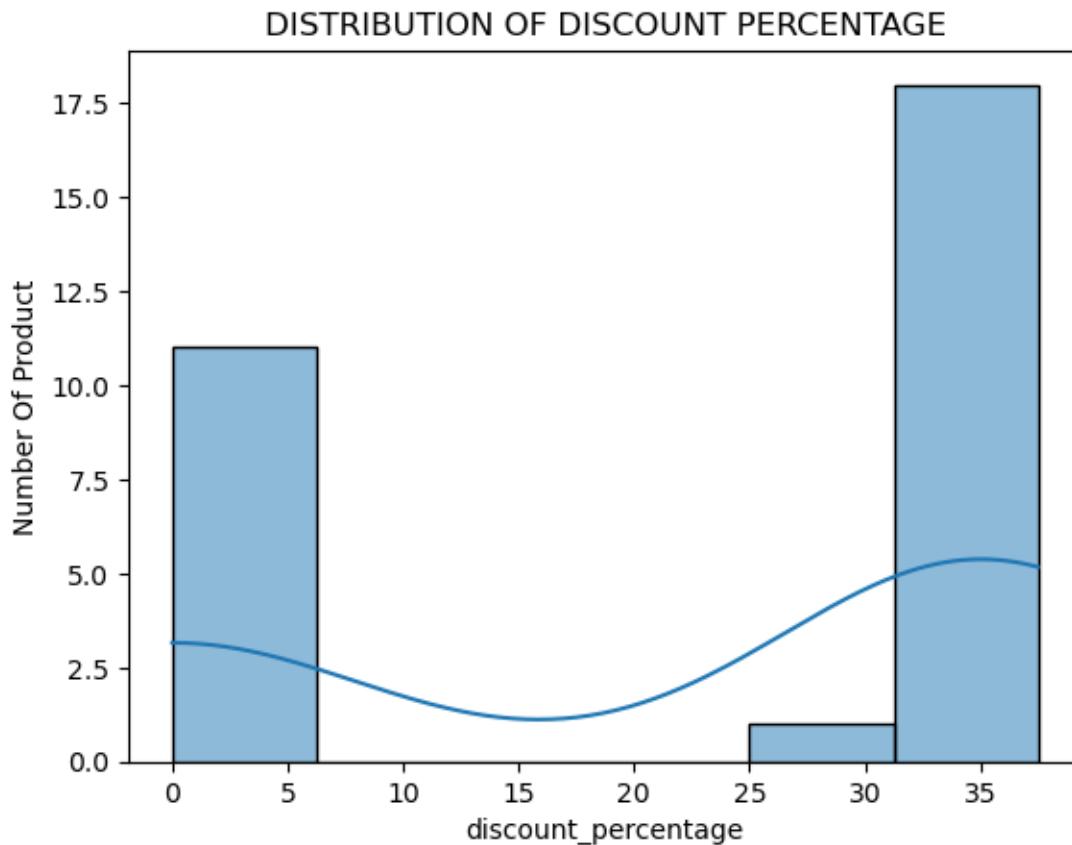
[182]: Text(0.5, 1.0, 'Prices Vary Across Different Product Categories')



[183]: *#What is the distribution of price discounts percentage offered?*

```
sns.histplot(luxury_product['discount_percentage'], kde=True)
plt.ylabel("Number Of Product")
plt.title("DISTRIBUTION OF DISCOUNT PERCENTAGE")
```

[183]: Text(0.5, 1.0, 'DISTRIBUTION OF DISCOUNT PERCENTAGE')



```
[184]: #All products which have more than 30% discounts
discount30=luxury_product[(luxury_product['discount_percentage']>30)]
discount30
```

```
[184]:          uniq_id \
9782  022a7f08d051e2cfca917ef0a6d3a19a
9953  a1fb13ead0d648acd61e600531421a34
9954  6bdbfc8a37c9302fea53891aaad7426
11142  e794bd226967af2298a2c7a1f8e2c9bb
11143  438cde07e5c6f018f3fc2d11717f5c13
11148  44ac0b6808242ce5c71b971d7620c609
11327  dd96000fa1d9e408a4fc47ea5c1123e5
11414  1b1c83326a749b99c9caf740f9136fa
11464  07b0df742cdcac28d09c29a1e246fff2
11507  d9fa5b1d8917b841abaef2a1ce032114
11542  4e3e2624f30da4c66c52de92f9b88f18
11554  e0de6958769d030f7e470e590f374d24
11576  30f324f95b6f5c26284893d0d85becf1
11598  3d17b0bac575544e5274ff836efaaee36
11614  8cfb7f1ab467a2439eeec8f9257abeac
```

11631	3a2546675bc399953779e58d84d56650
11633	307da045bb08837d8fe096b8f929d8a3
11646	e5a97beaed03ee8ece9a5749a431e985
11702	5f8e0c25e2915bc2383e60ad02b1e4aa

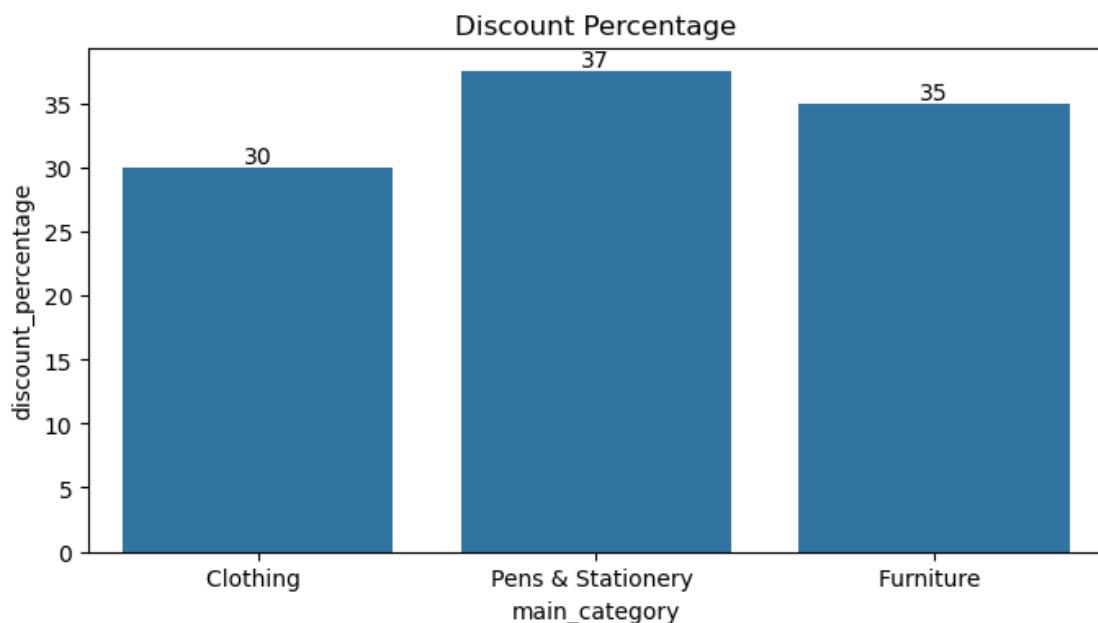
	product_name	pid	\
9782	The Curve Full Sleeve Solid Men's Quilted Jacket	JCKEAEA7REDBGJYV	
9953	NITGEN eNBioAccess T5-HID Time & Attendance, A...	BMDEGVYDBZ7NHRZH	
9954	NITGEN Fingkey Access Plus-HID Time & Attendant...	BMDEGVYEE6HXZJCK	
11142	Durian Dream/2 Leather 2 Seater Sofa	SOFECYFAGGVMMWS7	
11143	Durian Berry Solid Wood 3 Seater Sofa	SOFEE8HPYTFK7ZQKH	
11148	Durian Fabric 3 Seater Sofa	SOFEE7ACDGRPDWGZ	
11327	Durian Club/3 Leather 3 Seater Sofa	SOFECYFAH37ZKUKW	
11414	Durian Harmony/2 Fabric 2 Seater Sofa	SOFECYFACHZTHA2C	
11464	Durian Leather 2 Seater Sofa	SOFEE7AC7XCYXDVC	
11507	Durian Laze/3 Leather 3 Seater Sofa	SOFECYFA8R8T5HNG	
11542	Durian Clinton/A/3 Leather 3 Seater Sofa	SOFEE94HW8TPTGPNX	
11554	Durian Fabric 2 Seater Sofa	SOFEE7ACZNC57QYQ	
11576	Durian Helena Fabric 3 Seater Sofa	SOFEE8HPYPJPUUZ3H	
11598	Durian Tulsa Leatherette 3 Seater Sofa	SOFEE8HPYHFDGPFHV	
11614	Durian Leather 1 Seater Sofa	SOFEE7ADPUPX9MTW	
11631	Durian Leather 2 Seater Sofa	SOFEE7ACBVAGKYUV	
11633	Durian Salina Leatherette 2 Seater Sofa	SOFEE8HPYDEWAGBBH	
11646	Durian Fabric 3 Seater Sofa	SOFEE7ACTFUYMFGG	
11702	Durian Helena Fabric 3 Seater Sofa	SOFEE8HPYX66KMC3A	

	retail_price	discounted_price	is_FK_Advantage_product	product_rating	\
9782	69999.0	48999.0	False	0	
9953	71687.0	44804.0	False	0	
9954	58520.0	36575.0	False	0	
11142	108000.0	70200.0	False	0	
11143	56400.0	36660.0	False	0	
11148	93600.0	60840.0	False	0	
11327	162000.0	105300.0	False	0	
11414	85500.0	55575.0	False	0	
11464	204600.0	132990.0	False	0	
11507	217500.0	141375.0	False	0	
11542	73500.0	47775.0	False	5	
11554	84300.0	54795.0	False	0	
11576	108900.0	70785.0	False	0	
11598	75300.0	48945.0	False	0	
11614	69300.0	45045.0	False	0	
11631	250500.0	162825.0	False	0	
11633	54300.0	35295.0	False	0	
11646	73800.0	47970.0	False	0	
11702	108900.0	70785.0	False	0	

	overall_rating	brand	Year	Month	Days	main_category	\
9782	0	Regular	2016	1	7	Clothing	
9953	0	NITGEN	2016	3	20	Pens & Stationery	
9954	0	NITGEN	2016	3	20	Pens & Stationery	
11142	0	Durian	2015	12	31	Furniture	
11143	0	Durian	2015	12	31	Furniture	
11148	0	Durian	2015	12	31	Furniture	
11327	0	Durian	2015	12	31	Furniture	
11414	0	Durian	2015	12	31	Furniture	
11464	0	Durian	2015	12	31	Furniture	
11507	0	Durian	2015	12	31	Furniture	
11542	5	Durian	2015	12	31	Furniture	
11554	0	Durian	2015	12	31	Furniture	
11576	0	Durian	2015	12	31	Furniture	
11598	0	Durian	2015	12	31	Furniture	
11614	0	Durian	2015	12	31	Furniture	
11631	0	Durian	2015	12	31	Furniture	
11633	0	Durian	2015	12	31	Furniture	
11646	0	Durian	2015	12	31	Furniture	
11702	0	Durian	2015	12	31	Furniture	
							\
		secondary_category		tertiary_category			
9782		Men's Clothing		Winter & Seasonal Wear			
9953		Office Equipments		Biometric Devices			
9954		Office Equipments		Biometric Devices			
11142		Living Room		Sofas & Sectionals"]			
11143		Living Room		Sofas & Sectionals"]			
11148		Living Room		Sofas & Sectionals"]			
11327		Living Room		Sofas & Sectionals"]			
11414		Living Room		Sofas & Sectionals"]			
11464		Living Room		Sofas & Sectionals"]			
11507		Living Room		Sofas & Sectionals"]			
11542		Living Room		Sofas & Sectionals"]			
11554		Living Room		Sofas & Sectionals"]			
11576		Living Room		Sofas & Sectionals"]			
11598		Living Room		Sofas & Sectionals"]			
11614		Living Room		Sofas & Sectionals"]			
11631		Living Room		Sofas & Sectionals"]			
11633		Living Room		Sofas & Sectionals"]			
11646		Living Room		Sofas & Sectionals"]			
11702		Living Room		Sofas & Sectionals"]			
							\
		quaternary_category		discount_percentage		product_category	
9782		Jacket		30.000429		Luxury Product	
9953		NITGEN Biometric Device		37.500523		Luxury Product	
9954		NITGEN Biometric Device		37.500000		Luxury Product	
11142		None		35.000000		Luxury Product	

11143	None	35.000000	Luxury Product
11148	None	35.000000	Luxury Product
11327	None	35.000000	Luxury Product
11414	None	35.000000	Luxury Product
11464	None	35.000000	Luxury Product
11507	None	35.000000	Luxury Product
11542	None	35.000000	Luxury Product
11554	None	35.000000	Luxury Product
11576	None	35.000000	Luxury Product
11598	None	35.000000	Luxury Product
11614	None	35.000000	Luxury Product
11631	None	35.000000	Luxury Product
11633	None	35.000000	Luxury Product
11646	None	35.000000	Luxury Product
11702	None	35.000000	Luxury Product

```
[185]: plt.figure(figsize=(8,4))
ax=sns.barplot(x=discount30.main_category,y=discount30.discount_percentage)
plt.title('Discount Percentage')
for x in ax.containers:
    ax.bar_label(x,fmt='%d')
```



```
[186]: #Are there any significant price differences between FK Advantage products and others?
luxury_product['is_FK_Advantage_product'].value_counts()
```

```
[186]: is_FK_Advantage_product
      False    30
      Name: count, dtype: int64
```

```
[187]: luxury_product.groupby('is_FK_Advantage_product')['retail_price'].describe()
```

```
[187]:
```

	count	mean	std	min	\
is_FK_Advantage_product					
False	30.0	115589.266667	101329.129824	51400.0	
	25%	50%	75%	max	
is_FK_Advantage_product					
False	66525.0	79800.0	108900.0	571230.0	

THERE IS NO PRODUCT WITH FK_ADVANTAGE

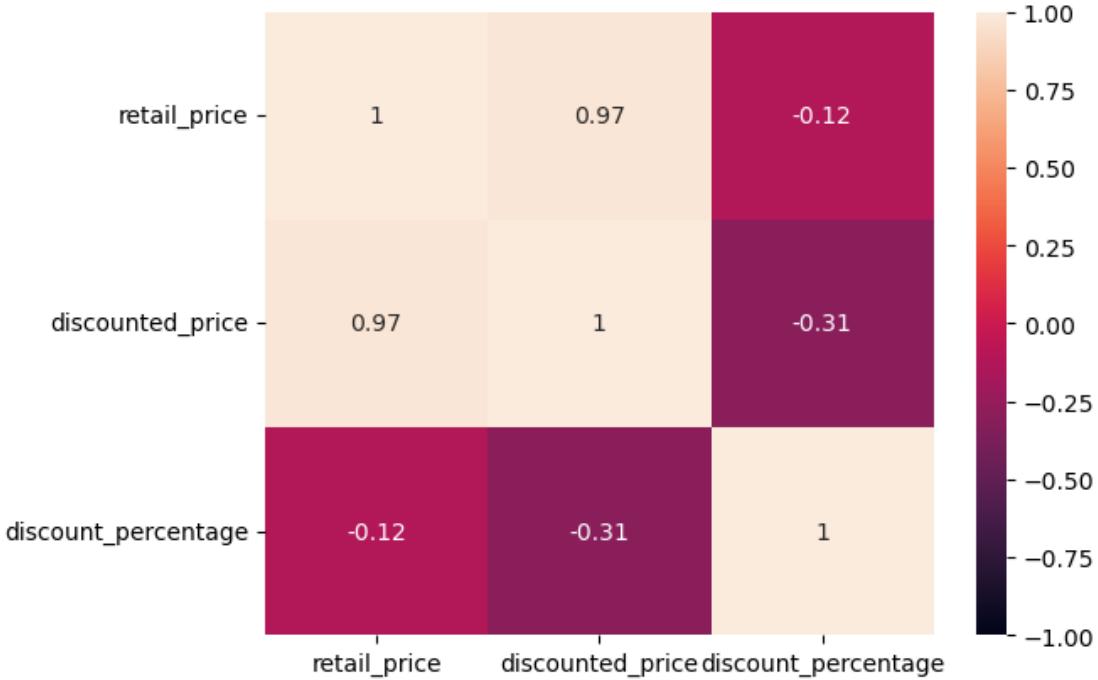
```
[189]: #correlation between retail_price, discounted_price, discount_percent
corr_luxury_products=luxury_product[['retail_price','discounted_price','discount_percentage']]
    ↪corr()
corr_luxury_products
```

```
[189]:
```

	retail_price	discounted_price	discount_percentage
retail_price	1.000000	0.970632	-0.117976
discounted_price	0.970632	1.000000	-0.305231
discount_percentage	-0.117976	-0.305231	1.000000

```
[190]: sns.heatmap(corr_luxury_products,vmin=-1,vmax=1,annot=True)
```

```
[190]: <Axes: >
```



RATING ANALYSIS:

```
[192]: rating_corr_luxury=luxury_product[['product_rating','overall_rating']].corr()
rating_corr_luxury
```

```
[192]: product_rating  overall_rating
product_rating        1.0          1.0
overall_rating        1.0          1.0
```

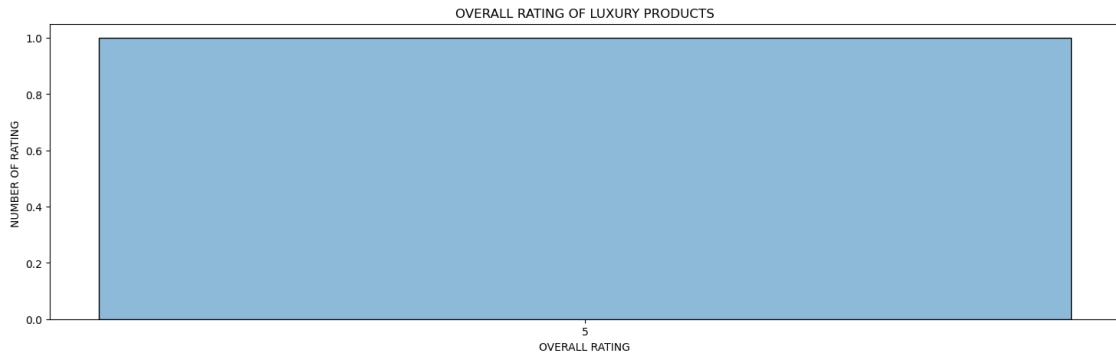
```
[193]: #how rating vary for different products of occasional product category after
       ↵removing zero rating for better results
errorfree_rating_luxury_product=luxury_product[~((luxury_product['product_rating']==0)|
       ↵& (luxury_product['overall_rating']==0))]
errorfree_rating_luxury_product
plt.figure(figsize=(20,5))
sns.histplot(errorfree_rating_luxury_product.product_rating,kde=True)
plt.xlabel("PRODUCT RATING")
plt.ylabel("NUMBER OF RATING ")
plt.title("PRODUCT RATING OF LUXURY PRODUCTS")
```

```
[193]: Text(0.5, 1.0, 'PRODUCT RATING OF LUXURY PRODUCTS')
```



```
[194]: plt.figure(figsize=(18,5))
sns.histplot(errorfree_rating_luxury_product.overall_rating,kde=True)
plt.xlabel("OVERALL RATING")
plt.ylabel("NUMBER OF RATING ")
plt.title("OVERALL RATING OF LUXURY PRODUCTS")
```

```
[194]: Text(0.5, 1.0, 'OVERALL RATING OF LUXURY PRODUCTS')
```



```
[195]: errorfree_rating_luxury_product
```

```
[195]:          uniq_id \
11542  4e3e2624f30da4c66c52de92f9b88f18

                           product_name          pid \
11542  Durian Clinton/A/3 Leather 3 Seater Sofa  SOFE94HW8TPTGPNX

                           retail_price  discounted_price  is_FK_Advantage_product  product_rating \
11542      73500.0              47775.0                  False                      5

                           overall_rating  brand  Year  Month  Days  main_category \
11542          5  Durian  2015     12     31  Furniture
```

```

    secondary_category      tertiary_category quaternary_category \
11542      Living Room      Sofas & Sectionals"]           None

    discount_percentage product_category
11542            35.0      Luxury Product

```

only one product which have rating in luxury product category

```
[197]: #how retail_price,product_rating,overall_rating related to each other
rating_corr_luxury_products=luxury_product[['retail_price','product_rating','overall_rating']]
    .corr()
rating_corr_luxury_products
```

```

[197]:
            retail_price  product_rating  overall_rating
retail_price        1.000000      -0.078451      -0.078451
product_rating      -0.078451       1.000000      1.000000
overall_rating      -0.078451       1.000000      1.000000

```

```
[198]: sns.heatmap(rating_corr_luxury_products,vmin=-1,vmax=1,annot=True)
```

```
[198]: <Axes: >
```

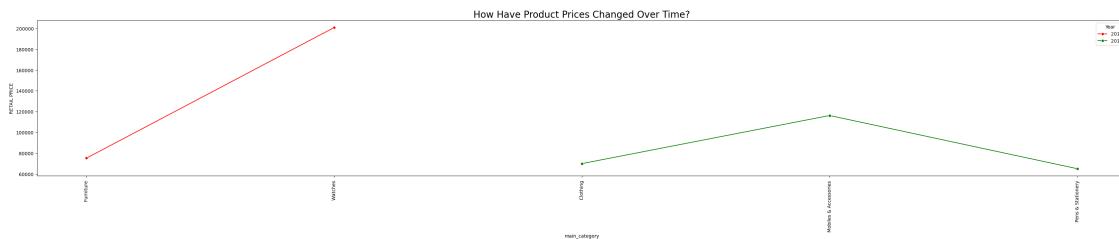


```
[199]: #How have product prices changed over time in luxury product catgeory?
luxury_product_price_over_year=luxury_product.
    ↪groupby(['Year','main_category'])['retail_price'].median()
luxury_product_price_over_year
```

```
[199]: Year  main_category
2015  Furniture           75300.0
      Watches              201000.0
2016  Clothing             69999.0
      Mobiles & Accessories 116292.0
      Pens & Stationery      65103.5
Name: retail_price, dtype: float64
```

```
[200]: plt.figure(figsize=(42,6))
plt.xticks(rotation=90)
color=['#FF0000','#008000']
plt.ylabel("RETAIL PRICE")
plt.title("How Have Product Prices Changed Over Time?",fontsize=20)
sns.lineplot(y=luxury_product_price_over_year.
    ↪values,x=luxury_product_price_over_year.index.
    ↪get_level_values('main_category'),
    hue=luxury_product_price_over_year.index.
    ↪get_level_values('Year'),palette=color,marker='o')
```

```
[200]: <Axes: title={'center': 'How Have Product Prices Changed Over Time?'},
 xlabel='main_category', ylabel='RETAIL PRICE'>
```



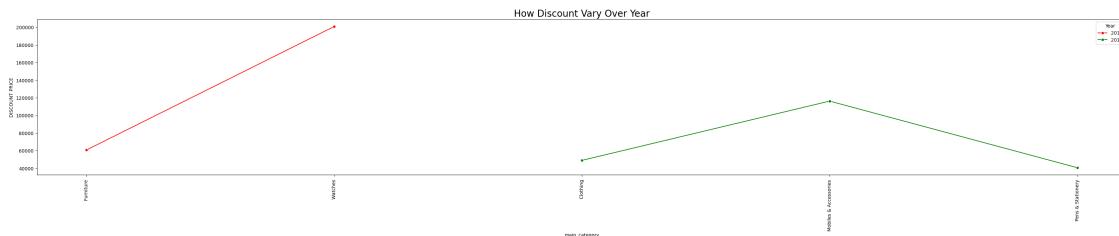
```
[201]: # how discount vary over year
luxury_product_discount_price_over_year=luxury_product.
    ↪groupby(['Year','main_category'])['discounted_price'].median()
luxury_product_discount_price_over_year
```

```
[201]: Year  main_category
2015  Furniture           60840.0
      Watches              201000.0
2016  Clothing             48999.0
      Mobiles & Accessories 116292.0
```

```
Pens & Stationery      40689.5
Name: discounted_price, dtype: float64
```

```
[202]: plt.figure(figsize=(42,6))
plt.xticks(rotation=90)
color=['#FF0000','#008000']
plt.ylabel("DISCOUNT PRICE")
plt.title("How Discount Vary Over Year",fontsize=20)
sns.lineplot(y=luxury_product_discount_price_over_year
              .values,x=luxury_product_discount_price_over_year
              .index.get_level_values('main_category'),
              hue=luxury_product_discount_price_over_year
              .index.get_level_values('Year'),palette=color,marker='o')
```

```
[202]: <Axes: title={'center': 'How Discount Vary Over Year'}, xlabel='main_category',
ylabel='DISCOUNT PRICE'>
```



```
[203]: #How discount_percentag of main_category vary from year to year,month to
      ↪month
      ↪year to year
year_to_year_discount_change_of_luxury =luxury_product.
      ↪groupby(['Year','main_category'])['discount_percentage'].median().
      ↪sort_values(ascending=False)
year_to_year_discount_change_of_luxury
```

```
[203]: Year  main_category
2016  Pens & Stationery      37.500262
2015  Furniture            35.000000
2016  Clothing             30.000429
2015  Watches               0.000000
2016  Mobiles & Accessories 0.000000
Name: discount_percentage, dtype: float64
```

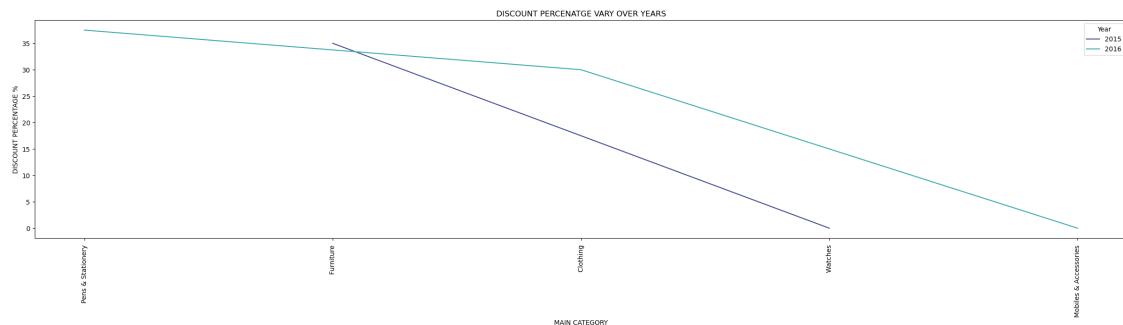
```
[204]: plt.figure(figsize=(30,6))
plt.xticks(rotation=90)
unique_years = year_to_year_discount_change_of_luxury.index.
      ↪get_level_values('Year').nunique()
```

```

sns.lineplot(y=year_to_year_discount_change_of_luxury
             .values, x=year_to_year_discount_change_of_luxury.index,
             ↪get_level_values('main_category'),
             hue=year_to_year_discount_change_of_luxury.index,
             ↪get_level_values('Year')
             , palette=sns.color_palette("mako", unique_years))
plt.xlabel("MAIN CATEGORY")
plt.ylabel("DISCOUNT PERCENTAGE %")
plt.title("DISCOUNT PERCENTAGE VARY OVER YEARS")

```

[204]: Text(0.5, 1.0, 'DISCOUNT PERCENTAGE VARY OVER YEARS')



```

[205]: month_to_month_discount_change_luxury=luxury_product.
         ↪groupby(['Month', 'main_category'])['discount_percentage'].median().
         ↪sort_values(ascending=False)
month_to_month_discount_change_luxury

```

[205]:

Month	main_category	discount_percentage
3	Pens & Stationery	37.500262
12	Furniture	35.000000
1	Clothing	30.000429
3	Mobiles & Accessories	0.000000
12	Watches	0.000000

Name: discount_percentage, dtype: float64

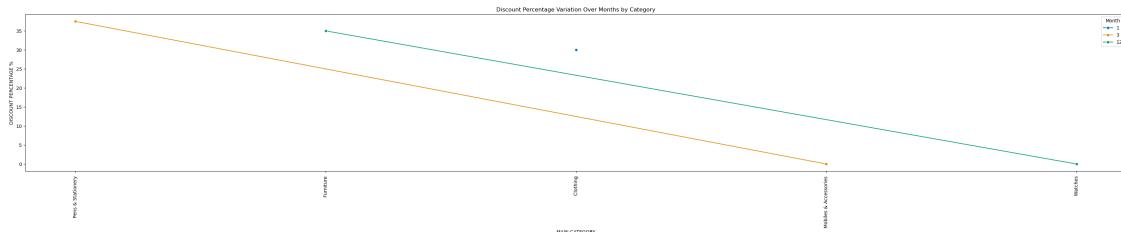
```

[206]: plt.figure(figsize=(42,6))
plt.xticks(rotation=90)
unique_years = month_to_month_discount_change_luxury.index.
               ↪get_level_values('Month').nunique()
sns.lineplot(y=month_to_month_discount_change_luxury.
             ↪values, x=month_to_month_discount_change_luxury.index,
             ↪get_level_values('main_category'),
             hue=month_to_month_discount_change_luxury.index.
             ↪get_level_values('Month')
             , palette=sns.color_palette("colorblind", unique_years), marker='o')

```

```
plt.xlabel("MAIN CATEGORY")
plt.ylabel("DISCOUNT PERCENTAGE %")
plt.title("Discount Percentage Variation Over Months by Category")
```

[206]: Text(0.5, 1.0, 'Discount Percentage Variation Over Months by Category')



[207]: *#let's check if there any seasonal sale and which product has heighest sale in which season*

```
season_sale_of_luxury=luxury_product.
    ↪pivot_table(values='retail_price',index=['Month','main_category'],aggfunc='sum')
season_sale_of_luxury
```

[207]:

Month	main_category	retail_price
1	Clothing	69999.0
3	Mobiles & Accessories	116292.0
	Pens & Stationery	130207.0
12	Furniture	2271200.0
	Watches	879980.0

[208]:

```
highest_sales_by_month_of_luxury = season_sale_of_luxury.groupby('Month').
    ↪apply(lambda x: x.loc[x['retail_price'].idxmax()])
highest_sales_by_month_of_luxury.sort_values(by='retail_price',ascending=False)
```

[208]:

Month	retail_price
12	2271200.0
3	130207.0
1	69999.0

In luxury product category mostly sales are in 12,3,1 months of year. May be it is because of new year, diwali and more festival celebration

Product Popularity:

[211]: *#Which product categories have the most products in luxury product category?*

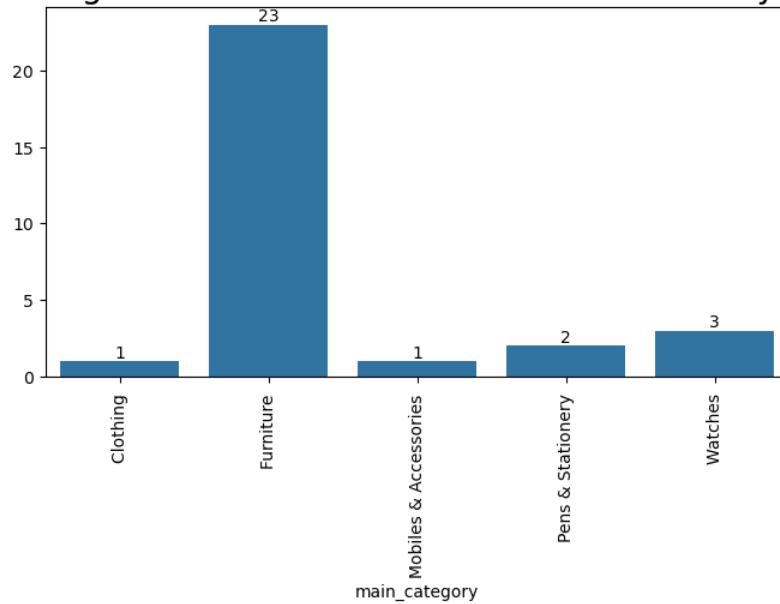
```
most_products_luxury=luxury_product.groupby('main_category')['main_category'].
    ↪size()
```

```
most_products_luxury.sort_values(ascending=False)
```

```
[211]: main_category
Furniture           23
Watches              3
Pens & Stationery      2
Clothing              1
Mobiles & Accessories    1
Name: main_category, dtype: int64
```

```
[212]: plt.figure(figsize=(8,4))
plt.xticks(rotation=90)
ax=sns.barplot(y=most_products_luxury.values,x=most_products_luxury.index)
plt.title("Product Categories And There Most Products In Luxury Category ?\n ↵",fontsize=20)
for container in ax.containers:
    ax.bar_label(container,fmt='%d')
```

Product Categories And There Most Products In Luxury Category ?



```
[213]: #product popularity over time
product_popularity_luxury=luxury_product.
    ↵groupby(['Year','main_category'])['pid'].count()
product_popularity_luxury
```

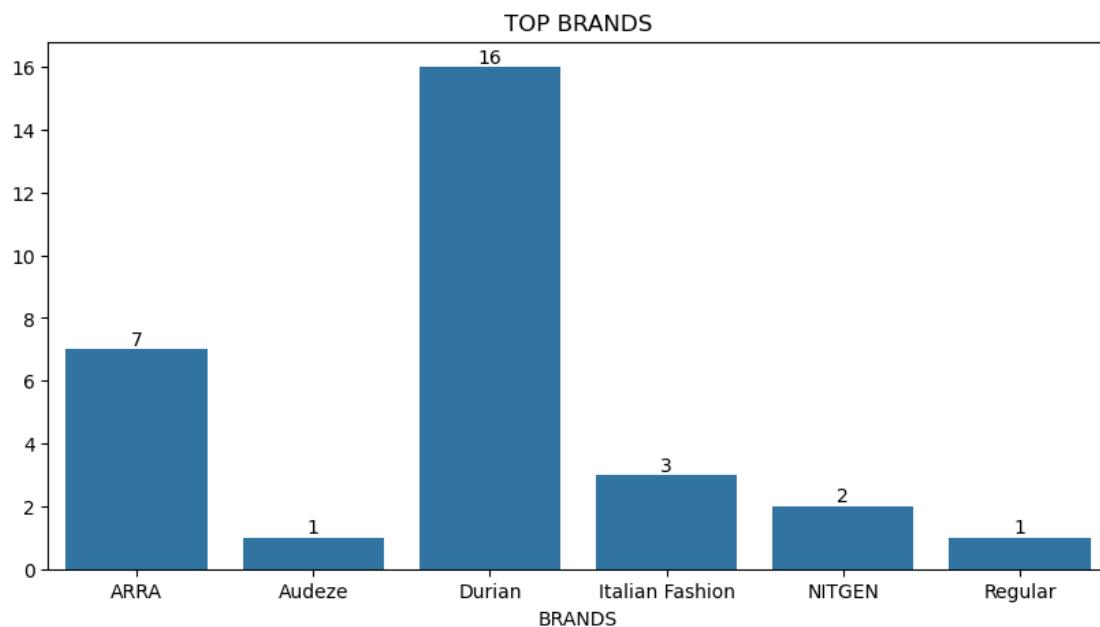
```
[213]: Year  main_category
2015  Furniture           23
          Watches              3
```

```
2016  Clothing          1
      Mobiles & Accessories 1
      Pens & Stationery    2
Name: pid, dtype: int64
```

```
[214]: #What are the top 10 most frequently occurring brands in luxury product
       ↪category?
brands_luxury=luxury_product.groupby('brand')['brand'].value_counts()
brands_luxury
```

```
[214]: brand
ARRA           7
Audeze         1
Durian         16
Italian Fashion 3
NITGEN         2
Regular         1
Name: count, dtype: int64
```

```
[215]: plt.figure(figsize=(10,5))
ax=sns.barplot(y=brands_luxury.values,x=brands_luxury.index.
       ↪get_level_values('brand'))
plt.xlabel("BRANDS")
plt.title('TOP BRANDS')
for container in ax.containers:
    ax.bar_label(container,fmt='%d')
```



```
[216]: #How many distinct product categories exist in luxury segment
main_category_luxury_products=luxury_product.
    ↪groupby('main_category')['main_category'].nunique()
main_category_luxury_products
```

```
[216]: main_category
Clothing           1
Furniture          1
Mobiles & Accessories 1
Pens & Stationery    1
Watches            1
Name: main_category, dtype: int64
```

```
[217]: #What are the most common product subcategories within each category of luxury
    ↪product?
luxury_most_common_product_first_category=luxury_product.
    ↪groupby(['main_category'])['secondary_category'].nunique().
    ↪sort_values(ascending=False)
luxury_most_common_product_first_category
```

```
[217]: main_category
Clothing           1
Furniture          1
Mobiles & Accessories 1
Pens & Stationery    1
Watches            1
Name: secondary_category, dtype: int64
```

```
[218]: luxury_most_common_product_second_category=luxury_product.
    ↪groupby(['secondary_category'])['tertiary_category'].nunique().
    ↪sort_values(ascending=False)
luxury_most_common_product_second_category
```

```
[218]: secondary_category
Wrist Watches      3
Living Room         1
Men's Clothing      1
Mobile Accessories   1
Office Equipments    1
Name: tertiary_category, dtype: int64
```

```
[219]: luxury_most_common_product_third_category=luxury_product.
    ↪groupby(['tertiary_category'])['quaternary_category'].nunique().
    ↪sort_values(ascending=False)
luxury_most_common_product_third_category
```

```
[219]: tertiary_category
Biometric Devices           1
Breitling Wrist Watches"]   1
Cartier Wrist Watches"]     1
Headphones                   1
Sofas & Sectionals"]       1
TAG Heuer Wrist Watches"]   1
Winter & Seasonal Wear     1
Name: quaternary_category, dtype: int64
```

```
[220]: # What are the most common product subcategories within each category?
level_1_category_luxury=luxury_product.
    ↪groupby(['main_category','secondary_category']).size().
    ↪sort_values(ascending=False)
level_1_category_luxury
```

```
[220]: main_category          secondary_category
Furniture                 Living Room           23
Watches                   Wrist Watches        3
Pens & Stationery        Office Equipments   2
Clothing                  Men's Clothing       1
Mobiles & Accessories    Mobile Accessories  1
dtype: int64
```

```
[221]: level_2_category_luxury=luxury_product.
    ↪groupby(['secondary_category','tertiary_category']).size().
    ↪sort_values(ascending=False)
level_2_category_luxury
```

```
[221]: secondary_category  tertiary_category
Living Room                Sofas & Sectionals"]  23
Office Equipments          Biometric Devices      2
Men's Clothing              Winter & Seasonal Wear 1
Mobile Accessories          Headphones            1
Wrist Watches              Breitling Wrist Watches"] 1
                           Cartier Wrist Watches"]   1
                           TAG Heuer Wrist Watches"]  1
dtype: int64
```

```
[222]: level_3_category_luxury=luxury_product.
    ↪groupby(['tertiary_category','quaternary_category']).size().
    ↪sort_values(ascending=False)
level_3_category_luxury
```

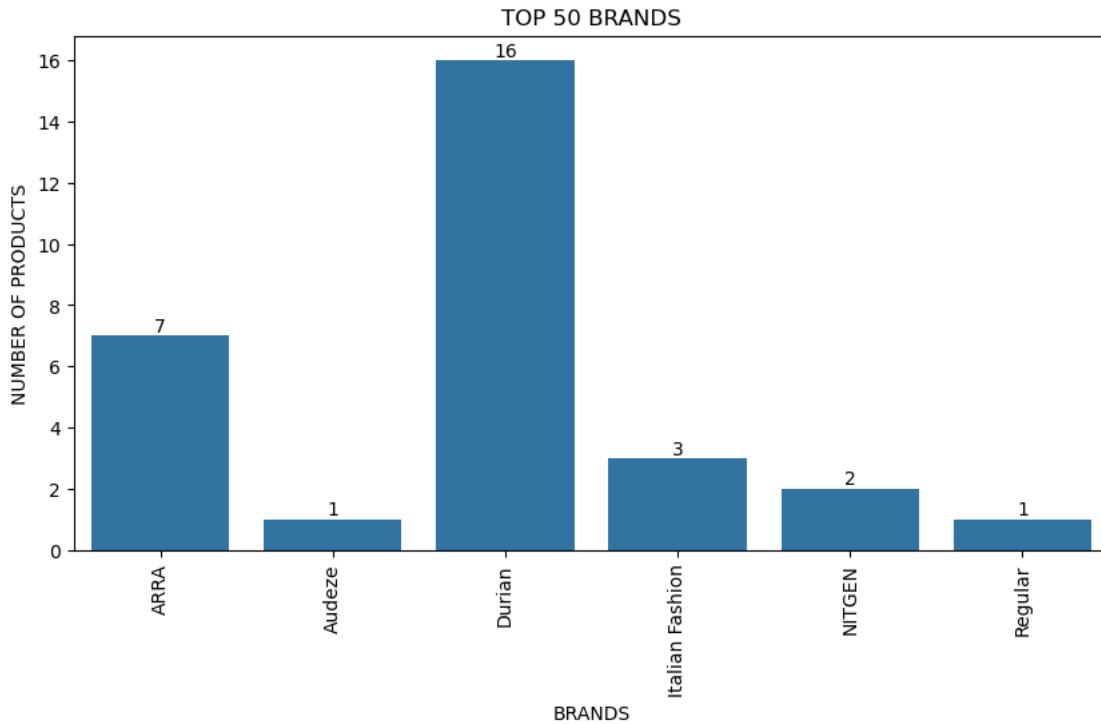
```
[222]: tertiary_category          quaternary_category
Sofas & Sectionals"]        None                  23
Biometric Devices           NITGEN Biometric Device 2
```

```
Breitling Wrist Watches"]    None          1
Cartier Wrist Watches"]    None          1
Headphones                  Audeze Headphone 1
TAG Heuer Wrist Watches"]  None          1
Winter & Seasonal Wear     Jacket        1
dtype: int64
```

```
[223]: #Which brands have the most products in the luxury product category?
most_brands_luxury=luxury_product.groupby('brand').size()
most_brands_luxury.sort_values(ascending=False)
```

```
[223]: brand
Durian          16
ARRA            7
Italian Fashion 3
NITGEN          2
Audeze          1
Regular         1
dtype: int64
```

```
[224]: plt.figure(figsize=(10,5))
plt.xticks(rotation=90)
ax=sns.barplot(x=most_brands_luxury.index,y=most_brands_luxury.values)
plt.xlabel('BRANDS')
plt.ylabel('NUMBER OF PRODUCTS')
plt.title("TOP 50 BRANDS")
for x in ax.containers:
    ax.bar_label(x,fmt='%d')
```



```
[225]: #Are there any brands that consistently offer higher discounts?
#Finding which brands give what percentage of discount to their product in
    ↪luxury product category
luxury_higher_discount_brands=luxury_product.
    ↪groupby('brand')['discount_percentage'].mean()
luxury_higher_discount_brands
```

```
[225]: brand
ARRA          0.000000
Audeze        0.000000
Durian        35.000000
Italian Fashion 0.000000
NITGEN        37.500262
Regular        30.000429
Name: discount_percentage, dtype: float64
```

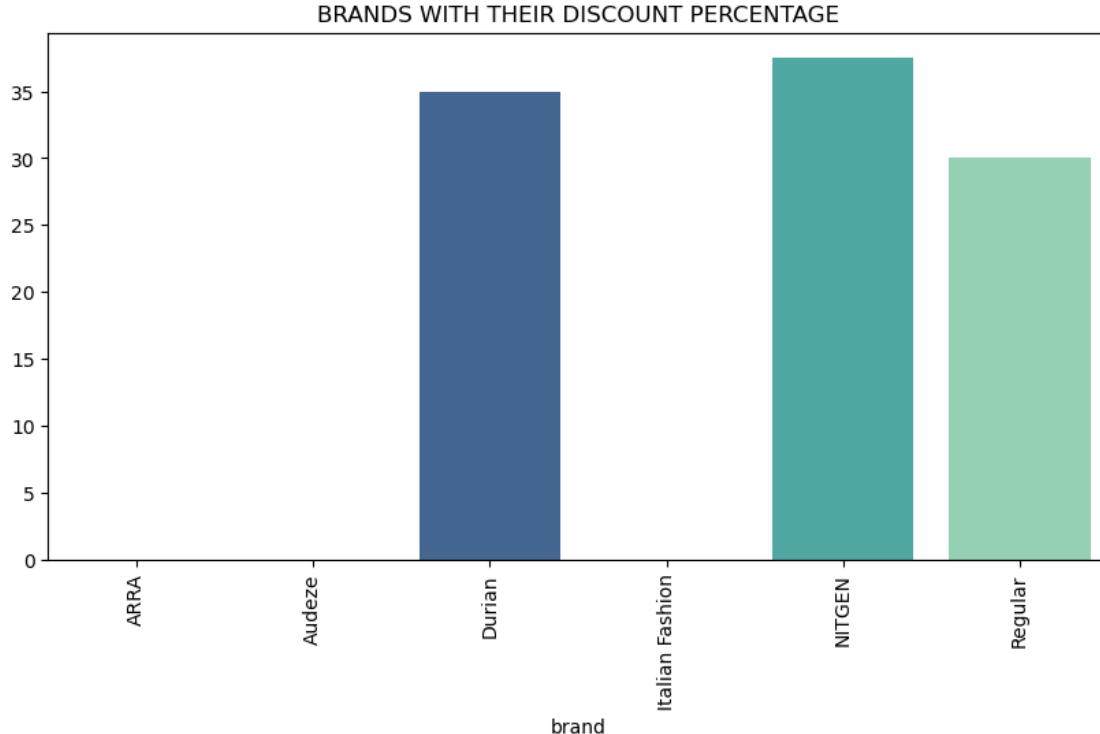
```
[226]: plt.figure(figsize=(10,5))
plt.xticks(rotation=90)
sns.barplot(x=luxury_higher_discount_brands.
    ↪index,y=luxury_higher_discount_brands.values,palette='mako')
plt.title('BRANDS WITH THEIR DISCOUNT PERCENTAGE',loc='center')
```

```
/var/folders/vc/4h3d9s6d639f63qzbhd805b80000gn/T/ipykernel_1158/4167213739.py:3:
FutureWarning:
```

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `x` variable to `hue` and set `legend=False` for the same effect.

```
sns.barplot(x=luxury_higher_discount_brands.index,y=luxury_higher_discount_brands.values,palette='mako')
```

[226]: Text(0.5, 1.0, 'BRANDS WITH THEIR DISCOUNT PERCENTAGE')



[227]: #To check the brands consistency for discount_percentage we can find standard deviation to know more.

```
luxury_consistent_higher_discount_brands=luxury_product.  
groupby('brand')['discount_percentage'].std()  
luxury_consistent_higher_discount_brands.sort_values(ascending=False)
```

[227]: brand

NITGEN	0.00037
ARRA	0.00000
Durian	0.00000
Italian Fashion	0.00000
Audeze	NaN
Regular	NaN

Name: discount_percentage, dtype: float64

[]: