Code for SQL Querying and Pipeling

In [ ]:
```
from odps import ODPS
import pandas as pd
from odps import options
from odps.df import DataFrame
import csv

o=ODPS('accessid','accesskey','project name',endpoint='',options.tunnel.endpoin
t='')
```

In [ ]:
```
#this is an SQL query to get Sales data through ODPS server from Trade Table

query="""SELECT  daraz_sku,product_name,short_code,seller_name,industry
        ,venture_category1_name_en
        ,venture_category2_name_en
        ,venture_category3_name_en
        ,business_area
        ,business_type
        ,ROUND(SUM((actual_gmv)*(exchange_rate))) AS GMV
        ,COUNT(DISTINCT sales_order_item_id) AS GIS
        ,COUNT(DISTINCT order_number) AS GOS
FROM    daraz_cdm.dwd_drz_trd_core_df_bd
WHERE   ds = TO_CHAR(DATEADD(GETDATE(), - 1, 'dd'), 'yyyymmdd')
AND     venture = 'BD'
AND     TO_CHAR(fulfillment_create_date, 'YYYYMMDD') BETWEEN 20230101
AND     20230107
AND     item_status_esm NOT IN ('invalid')
AND     is_fulfilled = 1
AND     actual_gmv > 0
AND     venture_category1_name_en IN ('Cameras','Home Appliances','Computers &
Laptops','TV, Audio / Video, Gaming & Wearables','Mobiles & Tablets')
GROUP BY daraz_sku,product_name,short_code,seller_name,industry
        ,venture_category1_name_en
        ,venture_category2_name_en
        ,venture_category3_name_en
        ,business_area
        ,business_type
ORDER BY GMV
;
"""
```

In [ ]:
```
#this code runs the query
df=o.execute_sql(query).open_reader().to_result_frame().to_pandas()

#this code exports sales data into an excel file
df.to_excel('EL Huntinglist Sale.csv',index=false)
```

In [ ]:
```
#this is an SQL query to get stock and price data through ODPS server from Prod
uct Table

query="""SELECT  daraz_sku,product_name,short_code,seller_name,industry
        ,venture_category1_name_en
        ,venture_category2_name_en
        ,venture_category3_name_en
        ,business_area
        ,business_type
        ,current_price
        ,stock_available
FROM    daraz_cdm.dwd_drz_prd_core_df_bd
WHERE   ds = TO_CHAR(DATEADD(GETDATE(), - 1, 'dd'), 'yyyymmdd')
        AND     venture = 'BD'
        AND is_visible = 1
AND     venture_category1_name_en IN ('Cameras','Home Appliances','Computers &
Laptops','TV, Audio / Video, Gaming & Wearables','Mobiles & Tablets')
GROUP BY daraz_sku,product_name,short_code,seller_name,industry
        ,venture_category1_name_en
        ,venture_category2_name_en
        ,venture_category3_name_en
        ,business_area
        ,business_type
ORDER BY GMV
;
"""
```

In [ ]:
```
#this code runs the query
df=o.execute_sql(query).open_reader().to_result_frame().to_pandas()

#this code exports sales data into an excel file
df.to_excel('EL Huntinglist Stock & Price.csv',index=false)
```

Code for Performance & Stock Analysis Using Python

In [7]:
```
import os

print(os.getcwd())  # Get current working directory
print(os.listdir())
```

```
C:\Users\user
['.anaconda', '.android', '.conda', '.condarc', '.InstallAnywhere', '.IntelliJIdea2019.3', '.ipynb_checkpoints', '.ip
ython', '.jupyter', '.m2', '.matplotlib', '.packettracer', '.spyder-py3', '3D Objects', 'Anaconda3', 'AnacondaProject
s', 'AppData', 'Application Data', 'Cisco Packet Tracer 7.3.0', 'Contacts', 'Cookies', 'Daraz SKU performance & Stock
Analysis.ipynb', 'Desktop', 'Documents', 'Downloads', 'edb_npgsql.exe', 'edb_pgjdbc.exe', 'edb_psqlodbc.exe', 'edb_ps
qlodbc.exe-20230929101224', 'EL Huntinglist Sale.csv', 'EL Huntinglist Stock & Price.csv', 'Favorites', 'help', 'hpee
sof', 'IdeaProjects', 'Links', 'Local Settings', 'MicrosoftEdgeBackups', 'Music', 'My Documents', 'NetHood', 'NTUSER.
DAT', 'ntuser.dat.LOG1', 'ntuser.dat.LOG2', 'NTUSER.DAT{1c2b59c6-c5f5-11eb-bacb-000d3a96488e}.TM.blf', 'NTUSER.DAT{1c
2b59c6-c5f5-11eb-bacb-000d3a96488e}.TMContainer00000000000000000001.regtrans-ms', 'NTUSER.DAT{1c2b59c6-c5f5-11eb-bacb
-000d3a96488e}.TMContainer00000000000000000002.regtrans-ms', 'ntuser.ini', 'OneDrive', 'OneDrive - au.edu', 'Picture
s', 'Postman', 'PrintHood', 'PycharmProjects', 'Recent', 'Saved Games', 'Searches', 'SendTo', 'SQL Querying to get Da
raz SKU with Sale data and Stock & Price data.ipynb', 'Start Menu', 'Templates', 'Videos']
```

In [8]:
```
import matplotlib.pyplot as plt
import pandas as pd
```

In [9]:
```
#Importing SKU wise performamce for last week

df1 = pd.read_csv("EL Huntinglist Sale.csv")
df1.head()

#GMV:Gross Merchandise Value
#GIS:Gross Item Sold
#GOS:Gross Order Sold
```

Out[9]:

	Daraz Sku	Product Name	Short Code	Seller Name	Industry	Category1 Name	Category2 Name	Category3 Name	Business Area	Business Type	GMV
0	208667145_BD-1158896479	realme narzo 50i-(4gb ram/64gb rom)	BDAW2HJ2VV	realme_Bangladesh	Electronics	Mobiles & Tablets	Tablets	NaN	Marketplace	DarazMall	354,459
1	208667145_BD-1158896480	realme narzo 50i-(4gb ram/64gb rom)	BDAW2HJ2VV	realme_Bangladesh	Electronics	Mobiles & Tablets	Tablets	NaN	Marketplace	DarazMall	347,432
2	215645559_BD-1164135194	realme C21Y - 3GB RAM / 32GB ROM	BDAW2HJ2VV	realme_Bangladesh	Electronics	Mobiles & Tablets	Tablets	NaN	Marketplace	DarazMall	107,685
3	213469548_BD-1162761018	Realme c11 4gb/64gb	BDAW2HJ2VV	realme_Bangladesh	Electronics	Mobiles & Tablets	Tablets	NaN	Marketplace	DarazMall	105,090
4	207938207_BD-1174459785	realme C11 - 2GB RAM/32GB ROM	BDAW2HJ2VV	realme_Bangladesh	Electronics	Mobiles & Tablets	Tablets	NaN	Marketplace	DarazMall	89,024

In [10]:
```
#Importing Stock & Price of Top performing electronic SKUs for that Week

df2 = pd.read_csv("EL Huntinglist Stock & Price.csv")
df2.head()
```

Out[10]:

	Daraz Sku	Product Name	Short Code	Seller Name	Industry	Category1 Name	Category2 Name	Category3 Name	Business Area	Business Type	Current Price
0	208667145_BD-1158896479	realme narzo 50i-(4gb ram/64gb rom)	BDAW2HJ2VV	realme_Bangladesh	Electronics	Mobiles & Tablets	Tablets	NaN	Marketplace	DarazMall	10,919
1	208667145_BD-1158896480	realme narzo 50i-(4gb ram/64gb rom)	BDAW2HJ2VV	realme_Bangladesh	Electronics	Mobiles & Tablets	Tablets	NaN	Marketplace	DarazMall	10,919
2	215645559_BD-1164135194	realme C21Y - 3GB RAM / 32GB ROM	BDAW2HJ2VV	realme_Bangladesh	Electronics	Mobiles & Tablets	Tablets	NaN	Marketplace	DarazMall	11,114
3	213469548_BD-1162761018	Realme c11 4gb/64gb	BDAW2HJ2VV	realme_Bangladesh	Electronics	Mobiles & Tablets	Tablets	NaN	Marketplace	DarazMall	10,868
4	207938207_BD-1174459785	realme C11 - 2GB RAM/32GB ROM	BDAW2HJ2VV	realme_Bangladesh	Electronics	Mobiles & Tablets	Tablets	NaN	Marketplace	DarazMall	8,960

In [11]:
```
#Merging two dataframes to create a single data frame
combined=df1.merge(df2)
combined.head()
```

Out[11]:

	Daraz Sku	Product Name	Short Code	Seller Name	Industry	Category1 Name	Category2 Name	Category3 Name	Business Area	Business Type	GMV
0	208667145_BD-1158896479	realme narzo 50i-(4gb ram/64gb rom)	BDAW2HJ2VV	realme_Bangladesh	Electronics	Mobiles & Tablets	Tablets	NaN	Marketplace	DarazMall	354,459
1	208667145_BD-1158896480	realme narzo 50i-(4gb ram/64gb rom)	BDAW2HJ2VV	realme_Bangladesh	Electronics	Mobiles & Tablets	Tablets	NaN	Marketplace	DarazMall	347,432
2	215645559_BD-1164135194	realme C21Y - 3GB RAM / 32GB ROM	BDAW2HJ2VV	realme_Bangladesh	Electronics	Mobiles & Tablets	Tablets	NaN	Marketplace	DarazMall	107,685
3	213469548_BD-1162761018	Realme c11 4gb/64gb	BDAW2HJ2VV	realme_Bangladesh	Electronics	Mobiles & Tablets	Tablets	NaN	Marketplace	DarazMall	105,090
4	207938207_BD-1174459785	realme C11 - 2GB RAM/32GB ROM	BDAW2HJ2VV	realme_Bangladesh	Electronics	Mobiles & Tablets	Tablets	NaN	Marketplace	DarazMall	89,024

In [12]:
```
combined.dtypes
```

Out[12]:
```
Daraz Sku         object
Product Name      object
Short Code        object
Seller Name       object
Industry          object
Category1 Name    object
Category2 Name    object
Category3 Name    object
Business Area     object
Business Type     object
GMV               object
GIS               object
GOS               object
Current Price     object
Stock Available   object
dtype: object
```

In [13]:
```
#Cleaning Data from columns GMV,GIS & GOS
combined['GMV']=combined['GMV'].str.replace(',','')
combined['GOS']=combined['GOS'].str.replace(',','')
combined['GIS']=combined['GIS'].str.replace(',','')
combined['Current Price']=combined['Current Price'].str.replace(',','')
combined['Stock Available']=combined['Stock Available'].str.replace(',','')
combined['Stock Available']=combined['Stock Available'].str.replace('########','0')


#Converting data type of GMV,GIS,GOS,Current Price & Stock from Object to Int in order perform agg functions


combined['GMV'] = pd.to_numeric(combined['GMV'])
combined['GOS'] = pd.to_numeric(combined['GOS'])
combined['GIS'] = pd.to_numeric(combined['GIS'])
combined['Current Price'] = pd.to_numeric(combined['Current Price'])
combined['Stock Available'] = pd.to_numeric(combined['Stock Available'])
```

In [14]:
```
combined.head()
```

Out[14]:

	Daraz Sku	Product Name	Short Code	Seller Name	Industry	Category1 Name	Category2 Name	Category3 Name	Business Area	Business Type	GMV
0	208667145_BD-1158896479	realme narzo 50i-(4gb ram/64gb rom)	BDAW2HJ2VV	realme_Bangladesh	Electronics	Mobiles & Tablets	Tablets	NaN	Marketplace	DarazMall	354459
1	208667145_BD-1158896480	realme narzo 50i-(4gb ram/64gb rom)	BDAW2HJ2VV	realme_Bangladesh	Electronics	Mobiles & Tablets	Tablets	NaN	Marketplace	DarazMall	347432
2	215645559_BD-1164135194	realme C21Y - 3GB RAM / 32GB ROM	BDAW2HJ2VV	realme_Bangladesh	Electronics	Mobiles & Tablets	Tablets	NaN	Marketplace	DarazMall	107685
3	213469548_BD-1162761018	Realme c11 4gb/64gb	BDAW2HJ2VV	realme_Bangladesh	Electronics	Mobiles & Tablets	Tablets	NaN	Marketplace	DarazMall	105090
4	207938207_BD-1174459785	realme C11 - 2GB RAM/32GB ROM	BDAW2HJ2VV	realme_Bangladesh	Electronics	Mobiles & Tablets	Tablets	NaN	Marketplace	DarazMall	89024

In [15]:
```
combined.dtypes
```

Out[15]:
```
Daraz Sku         object
Product Name      object
Short Code        object
Seller Name       object
Industry          object
Category1 Name    object
Category2 Name    object
Category3 Name    object
Business Area     object
Business Type     object
GMV                int64
GIS                int64
GOS                int64
Current Price      int64
Stock Available    int64
dtype: object
```

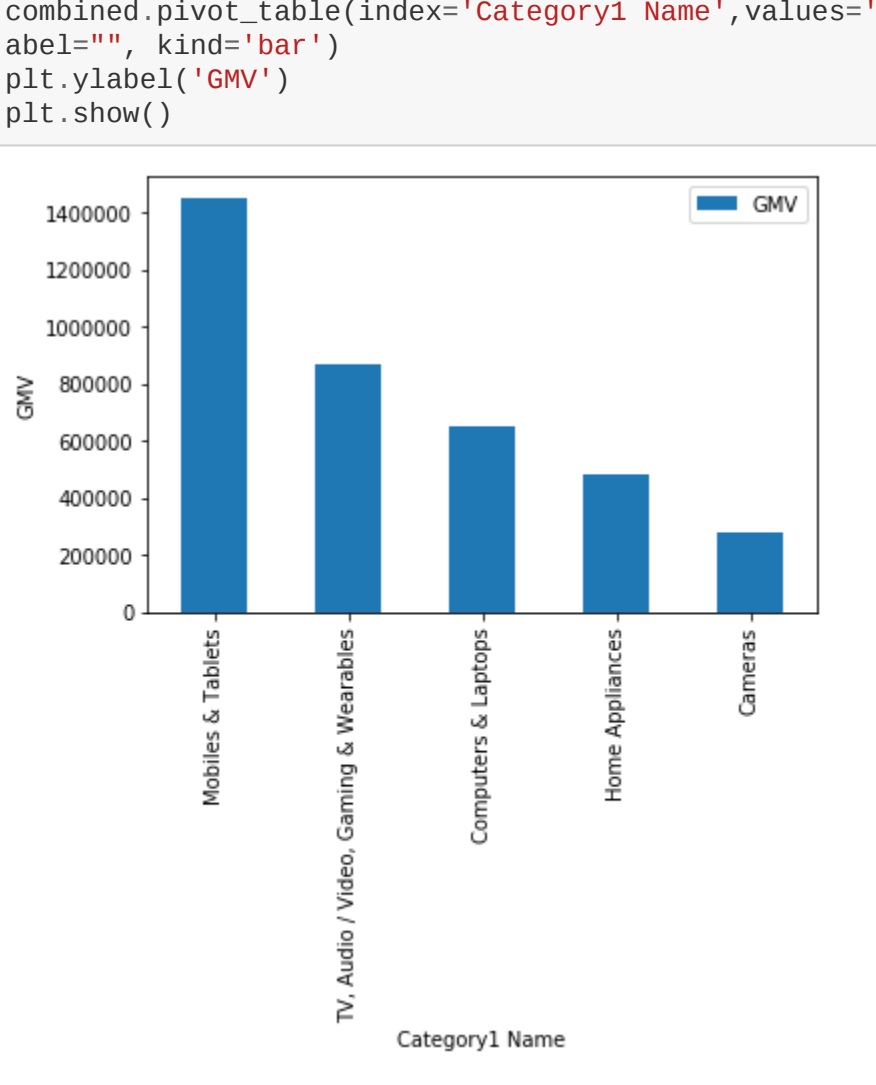
In [16]:
```
#Finding top performing category in terms of GMV & GOS

Category_Summary=combined.pivot_table(index='Category1 Name',values=['GMV','GOS'],aggfunc='sum').sort_values(by='GM
V',ascending=False)
Category_Summary=Category_Summary.reset_index()
Category_Summary
```

Out[16]:

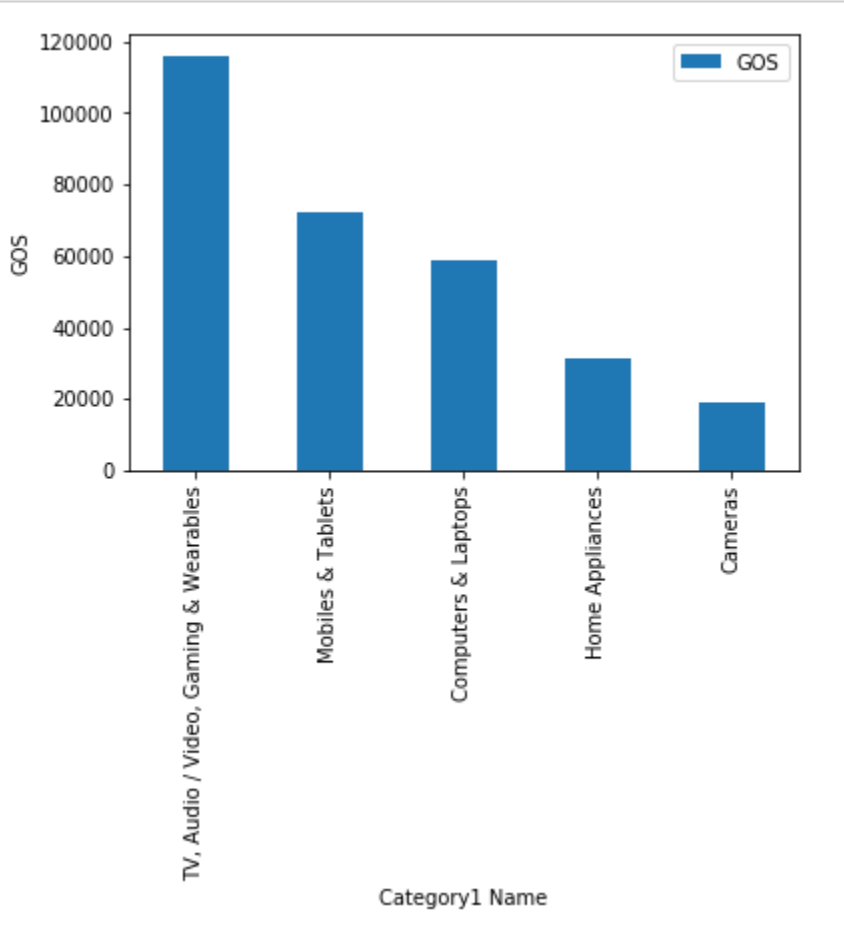
	Category1 Name	GMV	GOS
0	Mobiles & Tablets	1453121	72330
1	TV, Audio / Video, Gaming & Wearables	869937	116059
2	Computers & Laptops	655217	58807
3	Home Appliances	480443	31494
4	Cameras	280386	19076

In [17]:
```
#A visualization of top performing category in terms of GMV
combined.pivot_table(index='Category1 Name',values='GMV',aggfunc='sum').sort_values(by='GMV',ascending=False).plot(l
abel="", kind='bar')
plt.ylabel('GMV')
plt.show()
```



In [18]:
```
#A visualization of top performing category in terms of GOS

combined.pivot_table(index='Category1 Name',values='GOS',aggfunc='sum').sort_values(by='GOS',ascending=False).plot(l
abel="", kind='bar')
plt.ylabel('GOS')
plt.show()
```



In [19]:
```
# Here we can see the subcategory with highest GMV at what average price
combined.pivot_table(index=['Category1 Name','Category2 Name','Category3 Name'],values=['Current Price','GMV'],aggfu
nc={'Current Price':'mean','GMV':'sum'}).sort_values(by='GMV',ascending=False)
```

Out[19]:

			Current Price	GMV
Category1 Name	**Category2 Name**	**Category3 Name**		
TV, Audio / Video, Gaming & Wearables	**Audio**	**Headphones & Headsets**	840.268078	418298
	Wearable Technology	**Smartwatches**	3937.011765	105841
Cameras	**Security Cameras & Systems**	**IP Security Cameras**	2312.012987	94541
Computers & Laptops	**Computer Accessories**	**Laptop stands**	919.672131	88193
Home Appliances	**Cooling & Heating**	**Fan**	3459.219858	83619
TV, Audio / Video, Gaming & Wearables	**Televisions**	**Smart Televisions**	27256.550000	82116
Home Appliances	**Cooling & Heating**	**Air Conditioning**	34732.022222	75205
TV, Audio / Video, Gaming & Wearables	**Audio**	**Portable Speakers**	1567.655556	67741
Computers & Laptops	**Network Components**	**Routers**	3022.521739	49748
Mobiles & Tablets	**Mobile Accessories**	**Phone Cases**	339.595556	44230
		Docks & Stands	230.307692	43641
Computers & Laptops	**Computer Accessories**	**Drawing Tools**	1980.136364	42515
Home Appliances	**Small Kitchen Appliances**	**Food Preparation**	3355.325843	37215
Mobiles & Tablets	**Mobile Accessories**	**Power Banks**	1663.414634	35853
Computers & Laptops	**Network Components**	**Switches**	2872.114286	33952
Mobiles & Tablets	**Mobile Accessories**	**Cables & Converters**	419.826087	33303
Computers & Laptops	**Laptops**	**Gaming**	13161.700000	31391
	Computer Accessories	**Mice**	1006.338028	31164
	Computer Components	**Processors**	6590.238095	30937
Cameras	**Camera Accessories**	**Tripods & Monopods**	992.361111	28197
Home Appliances	**Appliances Parts & Accessories**	**Water Purifiers & Filters Accessories**	1161.950000	27361
Computers & Laptops	**Computer Components**	**RAM**	9510.825243	26348
	Computer Accessories	**Keyboards**	1946.377358	25486
TV, Audio / Video, Gaming & Wearables	**Audio**	**Live Sound & Stage Equipment**	2103.584615	24896
Cameras	**Camera Accessories**	**Lighting & Studio Equipment**	2339.255319	24138
Home Appliances	**Vacuums & Floor Care**	**Vacuum Cleaners**	6277.400000	23962
Computers & Laptops	**Storage**	**Flash Drives**	889.416667	23662
TV, Audio / Video, Gaming & Wearables	**TV Accessories**	**TV Receivers**	3317.473684	22882
Home Appliances	**Kitchen Appliances**	**Water Dispensers & Purifiers**	8315.482759	22532
TV, Audio / Video, Gaming & Wearables	**Televisions**	**LED Televisions**	15815.176471	20763
...	**...**	**...**	...	...
Cameras	**Camera Accessories**	**Gimbals & Stabilizers Accessories**	2455.500000	367
Computers & Laptops	**Desktops Computers**	**DIY**	9532.500000	365
Cameras	**Lenses**	**Smartphone Lenses**	974.500000	360
TV, Audio / Video, Gaming & Wearables	**Video**	**Projector Accessories**	880.000000	324
Cameras	**Camera Accessories**	**Camera Screen Protector**	297.000000	316
TV, Audio / Video, Gaming & Wearables	**Gadgets**	**Universal Chargers**	1688.000000	313
Home Appliances	**Appliances Parts & Accessories**	**Fan Parts & Accessories**	789.500000	302
		Water Dispenser Accessories	509.000000	301
		Ventilation Parts & Accessories	987.000000	277
Computers & Laptops	**PC Gaming**	**PC Games**	10707.000000	266
	Network Components	**Access Points**	1430.000000	264
	Computer Accessories	**Surge Protector**	1447.000000	260
TV, Audio / Video, Gaming & Wearables	**TV Accessories**	**TV Adapters**	20969.000000	256
Cameras	**Camera Accessories**	**Flashes**	7272.500000	247
TV, Audio / Video, Gaming & Wearables	**Video**	**Blu-Ray/DVD Players**	5303.500000	239
Home Appliances	**Small Kitchen Appliances**	**Electric Knives & Sharpeners**	528.000000	233
	Large Appliances	**Refrigerators**	23000.000000	225
Computers & Laptops	**Storage**	**Solid State Drives**	1412.500000	222
Home Appliances	**Appliances Parts & Accessories**	**Air Purifier Accessories**	8915.000000	208
Computers & Laptops	**Computer Accessories**	**External DVD Writers**	700.000000	188
		Monitor Stands	850.000000	187
Home Appliances	**Appliances Parts & Accessories**	**Small Kitchen Appliance Parts & Accessories**	1176.000000	142
Cameras	**Optics**	**Monoculars**	760.000000	134
Computers & Laptops	**Storage**	**OTG Drives**	870.000000	131
	Software	**Operating System**	2116.000000	123
Cameras	**Gadgets & Other Cameras**	**Lomography**	1000.000000	117
	Instant Camera	**Instant Camera Accessories**	149.000000	114
Home Appliances	**Small Kitchen Appliances**	**Electric Slow Cookers**	2374.000000	114
TV, Audio / Video, Gaming & Wearables	**Gadgets**	**Walkie-Talkies**	4800.000000	112
Home Appliances	**Small Kitchen Appliances**	**Electric Sandwich Makers & Presses**	956.000000	111

198 rows × 2 columns

In [20]:
```
#Top 5 order generating Seller
Top5=combined.pivot_table(index='Seller Name',values='GOS',aggfunc='sum').sort_values(by='GOS',ascending=False)
Top5.head()
```

Out[20]:

	GOS
Seller Name	
realme_Bangladesh	8299
Origin Technology	7462
PC DOKAN	6596
Time Square	6328
Jahan Telecom and photocopy center	6201

In [21]:
```
#AOV:Average order Value

#Calculating Average Order Value of individual Category
Category_Summary['AOV']=combined['GMV']/combined['GOS']
Category_Summary
```

Out[21]:

	Category1 Name	GMV	GOS	AOV
0	Mobiles & Tablets	1453121	72330	127.320043
1	TV, Audio / Video, Gaming & Wearables	869937	116059	127.779331
2	Computers & Laptops	655217	58807	135.794451
3	Home Appliances	480443	31494	132.522068
4	Cameras	280386	19076	105.980952

In [22]:
```
#Finding top 5 performing sub-category 2
top5_SubCat=combined.pivot_table(index=['Category1 Name','Category2 Name'],values=['GMV','GOS'],aggfunc='sum').sort_
values(by='GMV',ascending=False)
top5_SubCat.head()
```

Out[22]:

		GMV	GOS
Category1 Name	**Category2 Name**		
Mobiles & Tablets	**Tablets**	1181653	9069
TV, Audio / Video, Gaming & Wearables	**Audio**	529393	93411
Computers & Laptops	**Computer Accessories**	286271	38613
Mobiles & Tablets	**Mobile Accessories**	210300	62131
Home Appliances	**Cooling & Heating**	186679	8895