ELEVATE LABS TASK-1

In [3]:
```
import pandas as pd
df = pd.read_csv("C:/Users/lekhs/OneDrive/Documents/TASK 1.zip")
print(df.head(5))
```

```
   product_id product_name        category   price  review_score  \
0           1    Product_1        Clothing  190.40           1.7
1           2    Product_2  Home & Kitchen  475.60           3.2
2           3    Product_3            Toys  367.34           4.5
3           4    Product_4            Toys  301.34           3.9
4           5    Product_5           Books   82.23           4.2

   review_count  sales_month_1  sales_month_2  sales_month_3  sales_month_4  \
0           220            479            449             92            784
1           903             21            989            861            863
2           163            348            558            567            143
3           951            725            678             59             15
4           220            682            451            649            301

   sales_month_5  sales_month_6  sales_month_7  sales_month_8  sales_month_9  \
0            604            904            446            603            807
1            524            128            610            436            176
2            771            409            290            828            340
3            937            421            670            933             56
4            620            293            411            258            854

   sales_month_10  sales_month_11  sales_month_12
0             252             695             306
1             294             772             353
2             667             267             392
3             157             168             203
4             548             770             257
```

MISSING VALUES

In [4]:
```
df.isnull()
```

Out[4]:

	product_id	product_name	category	price	review_score	review_count	sales_month_
0	False	False	False	False	False	False	Fals
1	False	False	False	False	False	False	Fals
2	False	False	False	False	False	False	Fals
3	False	False	False	False	False	False	Fals
4	False	False	False	False	False	False	Fals
...	...	...	...	...	...	...	.
995	False	False	False	False	False	False	Fals
996	False	False	False	False	False	False	Fals
997	False	False	False	False	False	False	Fals
998	False	False	False	False	False	False	Fals
999	False	False	False	False	False	False	Fals

1000 rows × 18 columns

REMOVE DUPLICATES

In [11]:
```
df.drop_duplicates()
```

Out[11]:

	product_id	product_name	category	price	review_score	review_count	sales_month
0	1	Product_1	Clothing	190.40	1.7	220	4
1	2	Product_2	Home & Kitchen	475.60	3.2	903	
2	3	Product_3	Toys	367.34	4.5	163	3
3	4	Product_4	Toys	301.34	3.9	951	7
4	5	Product_5	Books	82.23	4.2	220	6
...	...	...	...	...	...	...	
995	996	Product_996	Home & Kitchen	50.33	3.6	494	4
996	997	Product_997	Home & Kitchen	459.07	4.8	701	
997	998	Product_998	Sports	72.73	1.3	287	7
998	999	Product_999	Sports	475.37	1.2	720	1
999	1000	Product_1000	Toys	225.77	2.1	114	8

1000 rows × 18 columns

STANDARDIZE TEXT VALUES

In [26]:
```
df['category']=df['category'].str.upper()
print(df.tail(100))
```

```
     product_id product_name        category   price  review_score  \
900         901          nan          SPORTS  107.90           3.2
901         902          nan        CLOTHING   18.13           1.0
902         903          nan          SPORTS   94.81           4.0
903         904          nan  HOME & KITCHEN  293.61           1.1
904         905          nan          SPORTS  213.61           4.0
..          ...          ...             ...     ...           ...
995         996          nan  HOME & KITCHEN   50.33           3.6
996         997          nan  HOME & KITCHEN  459.07           4.8
997         998          nan          SPORTS   72.73           1.3
998         999          nan          SPORTS  475.37           1.2
999        1000          nan            TOYS  225.77           2.1

     review_count  sales_month_1  sales_month_2  sales_month_3  sales_month_4  \
900           787            127            803            194            901
901           165            451            437            910            940
902           999            913            537            227             74
903           853            368            586            430            653
904           520            850            967            941            833
..            ...            ...            ...            ...            ...
995           494            488            359            137            787
996           701             18            906            129             78
997           287            725            109            193            657
998           720            196            191            315            622
999           114            890            903            983            769

     sales_month_5  sales_month_6  sales_month_7  sales_month_8  \
900              8            625            207             91
901            270            238            168            138
902            386            132            329            480
903             92            998            279            448
904            965            785            977            332
..             ...            ...            ...            ...
995            678            970            282            155
996             19            110            403            683
997            215            337            664            476
998            854            122             65            938
999            134            704            648            400

     sales_month_9  sales_month_10  sales_month_11  sales_month_12
900            256             699             777              14
901            345             341             782             686
902            697             481              41             258
903            344              18              92             722
904            399             627             696             411
..             ...             ...             ...             ...
995             57             575             634             393
996            104             858             729             474
997            265             344             888             654
998            521             268              60             394
999            495             839             611             110

[100 rows x 18 columns]
```

In [13]:
```
print(df.columns)
```

```
Index(['product_id', 'product_name', 'category', 'price', 'review_score',
       'review_count', 'sales_month_1', 'sales_month_2', 'sales_month_3',
       'sales_month_4', 'sales_month_5', 'sales_month_6', 'sales_month_7',
       'sales_month_8', 'sales_month_9', 'sales_month_10', 'sales_month_11',
       'sales_month_12'],
      dtype='object')
```

RENAME COLUMN HEADERS

In [36]:
```
df.rename(columns={"product_id":"Product ID","product_name":"Product name","categor
                   "sales_month_1":"Sales_month1","sales_month_2":"Sales_month2","s
                   "sales_month_5":"Sales_month5","sales_month_6":"Sales_month6","s
                   "sales_month_9":"Sales_month9","sales_month_10":"Sales_month10",
                    inplace= True)
print(df.head(23))
```

	Product ID	Product name	Category	Price	Review score	\
0	1	nan	CLOTHING	190.40	1.7	
1	2	nan	HOME & KITCHEN	475.60	3.2	
2	3	nan	TOYS	367.34	4.5	
3	4	nan	TOYS	301.34	3.9	
4	5	nan	BOOKS	82.23	4.2	
5	6	nan	TOYS	82.22	3.6	
6	7	nan	ELECTRONICS	33.75	3.8	
7	8	nan	ELECTRONICS	433.76	4.4	
8	9	nan	ELECTRONICS	302.55	2.0	
9	10	nan	CLOTHING	355.50	3.0	
10	11	nan	TOYS	15.19	1.9	
11	12	nan	HEALTH	485.11	5.0	
12	13	nan	ELECTRONICS	417.06	4.8	
13	14	nan	BOOKS	110.11	1.2	
14	15	nan	CLOTHING	95.00	3.8	
15	16	nan	BOOKS	95.79	4.7	
16	17	nan	TOYS	155.60	1.7	
17	18	nan	HOME & KITCHEN	264.75	3.3	
18	19	nan	BOOKS	218.81	4.7	
19	20	nan	HOME & KITCHEN	149.16	1.1	
20	21	nan	CLOTHING	307.87	3.8	
21	22	nan	BOOKS	74.05	2.2	
22	23	nan	BOOKS	149.61	4.7	

	review_count	Sales_month1	Sales_month2	Sales_month3	Sales_month4	\
0	220	479	449	92	784	
1	903	21	989	861	863	
2	163	348	558	567	143	
3	951	725	678	59	15	
4	220	682	451	649	301	
5	270	834	288	38	422	
6	632	565	140	539	244	
7	906	905	83	48	926	
8	848	809	693	419	106	
9	400	319	390	636	87	
10	182	125	590	12	128	
11	991	276	214	991	906	
12	618	975	525	35	886	
13	729	203	214	100	340	
14	205	193	402	793	3	
15	582	680	127	85	326	
16	556	663	821	515	409	
17	841	546	476	222	878	
18	813	695	759	337	305	
19	228	285	403	959	907	
20	696	56	28	165	606	
21	619	733	791	382	720	
22	596	302	337	855	224	

	Sales_month5	Sales_month6	Sales_month7	Sales_month8	Sales_month9	\
0	604	904	446	603	807	
1	524	128	610	436	176	
2	771	409	290	828	340	
3	937	421	670	933	56	
4	620	293	411	258	854	

5	329	245	614	979	851
6	188	700	896	965	447
7	94	555	280	102	202
8	616	85	42	992	696
9	7	555	485	566	632
10	66	841	595	768	737
11	495	374	962	100	201
12	190	445	702	197	329
13	365	377	50	812	362
14	767	389	897	225	891
15	520	656	209	942	453
16	989	50	125	462	355
17	116	241	491	236	215
18	266	293	508	25	971
19	610	919	959	120	554
20	484	562	969	909	886
21	508	265	580	505	22
22	297	119	776	181	154

	Sales_month10	Sales_month11	Sales_month12
0	252	695	306
1	294	772	353
2	667	267	392
3	157	168	203
4	548	770	257
5	630	738	485
6	66	32	516
7	753	376	445
8	298	213	708
9	462	561	179
10	724	439	341
11	819	491	132
12	100	392	442
13	55	488	835
14	331	721	252
15	835	945	264
16	750	28	92
17	392	38	661
18	742	858	410
19	873	611	137
20	380	172	207
21	91	16	636
22	790	55	745

CHECKING DATATYPE

In [34]:
```
print(df.dtypes)
```

```
Product ID         int64
Product name      object
Category          object
price            float64
Review score     float64
review_count       int64
Sales_month1       int64
Sales_month2       int64
Sales_month3       int64
Sales_month4       int64
Sales_month5       int64
Sales_month6       int64
Sales_month7       int64
Sales_month8       int64
Sales_month9       int64
Sales_month10      int64
Sales_month11      int64
Sales_month12      int64
dtype: object
```

In [46]:
```
import pandas as pd
df=pd.read_csv("C:/Users/lekhs/OneDrive/Documents/netflix task 1.zip")
print(df.head(5))
```

```
  show_id     type                  title         director  \
0      s1    Movie   Dick Johnson Is Dead  Kirsten Johnson
1      s2  TV Show          Blood & Water              NaN
2      s3  TV Show              Ganglands  Julien Leclercq
3      s4  TV Show  Jailbirds New Orleans              NaN
4      s5  TV Show           Kota Factory              NaN

                                                cast        country  \
0                                                NaN  United States
1  Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...   South Africa
2  Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...            NaN
3                                                NaN            NaN
4  Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...          India

         date_added  release_year rating   duration  \
0  September 25, 2021          2020  PG-13     90 min
1  September 24, 2021          2021  TV-MA  2 Seasons
2  September 24, 2021          2021  TV-MA   1 Season
3  September 24, 2021          2021  TV-MA   1 Season
4  September 24, 2021          2021  TV-MA  2 Seasons

                                     listed_in  \
0                                Documentaries
1  International TV Shows, TV Dramas, TV Mysteries
2  Crime TV Shows, International TV Shows, TV Act...
3                          Docuseries, Reality TV
4  International TV Shows, Romantic TV Shows, TV ...

                                     description
0  As her father nears the end of his life, filmm...
1  After crossing paths at a party, a Cape Town t...
2  To protect his family from a powerful drug lor...
3  Feuds, flirtations and toilet talk go down amo...
4  In a city of coaching centers known to train I...
```

In [67]:
```
df['date_added'] = pd.to_datetime(df['date_added'], errors='coerce', dayfirst= True
df["date_added"]=df["date_added"].dt.strftime('%d-%m-%Y')
print(df[['date_added']].head(25))
```

```
    date_added
0   25-09-2021
1   24-09-2021
2   24-09-2021
3   24-09-2021
4   24-09-2021
5   24-09-2021
6   24-09-2021
7   24-09-2021
8   24-09-2021
9   24-09-2021
10  24-09-2021
11  23-09-2021
12  23-09-2021
13  22-09-2021
14  22-09-2021
15  22-09-2021
16  22-09-2021
17  22-09-2021
18  22-09-2021
19  22-09-2021
20  22-09-2021
21  22-09-2021
22  21-09-2021
23  21-09-2021
24  21-09-2021
```

In [51]:
```
print(df.columns)
```

```
Index(['show_id', 'type', 'title', 'director', 'cast', 'country', 'date_added',
       'release_year', 'rating', 'duration', 'listed_in', 'description'],
      dtype='object')
```

MISSING VALUES

In [59]:
```
df.isnull()
```

Out[59]:

	show_id	type	title	director	cast	country	date_added	release_year	rating	dur
0	False	False	False	False	True	False	False	False	False	
1	False	False	False	True	False	False	False	False	False	
2	False	False	False	False	False	True	False	False	False	
3	False	False	False	True	True	True	False	False	False	
4	False	False	False	True	False	False	False	False	False	
...	...	...	...	...	...	...	...	...	...	
8802	False	False	False	False	False	False	False	False	False	
8803	False	False	False	True	True	True	False	False	False	
8804	False	False	False	False	False	False	False	False	False	
8805	False	False	False	False	False	False	False	False	False	
8806	False	False	False	False	False	False	False	False	False	

8807 rows × 12 columns

DATATYPES

In [61]:
```
print(df.dtypes)
```

```
show_id         object
type            object
title           object
director        object
cast            object
country         object
date_added      object
release_year     int64
rating          object
duration        object
listed_in       object
description     object
dtype: object
```