untitled2

January 25, 2024

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

[2]:
```
#loading dataset
path="online_retail.csv"
data1=pd.read_csv(path)
data1
```

[2]:
```
       InvoiceNo StockCode                          Description  Quantity  \
0         536365    85123A   WHITE HANGING HEART T-LIGHT HOLDER         6
1         536365     71053                  WHITE METAL LANTERN         6
2         536365    84406B       CREAM CUPID HEARTS COAT HANGER         8
3         536365    84029G  KNITTED UNION FLAG HOT WATER BOTTLE         6
4         536365    84029E       RED WOOLLY HOTTIE WHITE HEART.         6
…          …       …                                  …       …
541904    581587     22613          PACK OF 20 SPACEBOY NAPKINS        12
541905    581587     22899         CHILDREN'S APRON DOLLY GIRL          6
541906    581587     23254        CHILDRENS CUTLERY DOLLY GIRL          4
541907    581587     23255      CHILDRENS CUTLERY CIRCUS PARADE         4
541908    581587     22138        BAKING SET 9 PIECE RETROSPOT          3

                InvoiceDate  UnitPrice  CustomerID         Country
0       2010-12-01 08:26:00       2.55     17850.0  United Kingdom
1       2010-12-01 08:26:00       3.39     17850.0  United Kingdom
2       2010-12-01 08:26:00       2.75     17850.0  United Kingdom
3       2010-12-01 08:26:00       3.39     17850.0  United Kingdom
4       2010-12-01 08:26:00       3.39     17850.0  United Kingdom
…                     …        …         …             …
541904  2011-12-09 12:50:00       0.85     12680.0          France
541905  2011-12-09 12:50:00       2.10     12680.0          France
541906  2011-12-09 12:50:00       4.15     12680.0          France
541907  2011-12-09 12:50:00       4.15     12680.0          France
541908  2011-12-09 12:50:00       4.95     12680.0          France

[541909 rows x 8 columns]
```

```
[3]: #Getting shape of data
     data1.shape
```

```
[3]: (541909, 8)
```

```
[4]: #getting first 5 datasets
     data1.head()
```

```
[4]:   InvoiceNo StockCode                          Description  Quantity  \
     0    536365    85123A   WHITE HANGING HEART T-LIGHT HOLDER         6
     1    536365     71053                  WHITE METAL LANTERN         6
     2    536365    84406B       CREAM CUPID HEARTS COAT HANGER         8
     3    536365    84029G  KNITTED UNION FLAG HOT WATER BOTTLE         6
     4    536365    84029E       RED WOOLLY HOTTIE WHITE HEART.         6

               InvoiceDate  UnitPrice  CustomerID         Country
     0  2010-12-01 08:26:00       2.55     17850.0  United Kingdom
     1  2010-12-01 08:26:00       3.39     17850.0  United Kingdom
     2  2010-12-01 08:26:00       2.75     17850.0  United Kingdom
     3  2010-12-01 08:26:00       3.39     17850.0  United Kingdom
     4  2010-12-01 08:26:00       3.39     17850.0  United Kingdom
```

```
[5]: #describing dataset
     data1.describe()
```

```
[5]:             Quantity      UnitPrice     CustomerID
     count  541909.000000  541909.000000  406829.000000
     mean        9.552250       4.611114   15287.690570
     std       218.081158      96.759853    1713.600303
     min    -80995.000000  -11062.060000   12346.000000
     25%         1.000000       1.250000   13953.000000
     50%         3.000000       2.080000   15152.000000
     75%        10.000000       4.130000   16791.000000
     max     80995.000000   38970.000000   18287.000000
```

```
[6]: #Checking if there are any null values
     data1.isnull().sum()*100/data1.shape[0]
```

```
[6]: InvoiceNo       0.000000
     StockCode       0.000000
     Description     0.268311
     Quantity        0.000000
     InvoiceDate     0.000000
     UnitPrice       0.000000
     CustomerID     24.926694
     Country         0.000000
     dtype: float64
```

[7]:

```
#Dropping null values
data = data1.dropna()
print(data.shape)
```

```
(406829, 8)
```

[8]:

```
#New data
data
```

[8]:

```
       InvoiceNo StockCode                          Description  Quantity  \
0         536365    85123A   WHITE HANGING HEART T-LIGHT HOLDER         6
1         536365     71053                  WHITE METAL LANTERN         6
2         536365    84406B       CREAM CUPID HEARTS COAT HANGER         8
3         536365    84029G  KNITTED UNION FLAG HOT WATER BOTTLE         6
4         536365    84029E       RED WOOLLY HOTTIE WHITE HEART.         6
…          …       …                                  …       …
541904    581587     22613          PACK OF 20 SPACEBOY NAPKINS        12
541905    581587     22899         CHILDREN'S APRON DOLLY GIRL          6
541906    581587     23254        CHILDRENS CUTLERY DOLLY GIRL          4
541907    581587     23255      CHILDRENS CUTLERY CIRCUS PARADE         4
541908    581587     22138        BAKING SET 9 PIECE RETROSPOT          3

                InvoiceDate  UnitPrice  CustomerID         Country
0       2010-12-01 08:26:00       2.55     17850.0  United Kingdom
1       2010-12-01 08:26:00       3.39     17850.0  United Kingdom
2       2010-12-01 08:26:00       2.75     17850.0  United Kingdom
3       2010-12-01 08:26:00       3.39     17850.0  United Kingdom
4       2010-12-01 08:26:00       3.39     17850.0  United Kingdom
…                     …        …          …             …
541904  2011-12-09 12:50:00       0.85     12680.0          France
541905  2011-12-09 12:50:00       2.10     12680.0          France
541906  2011-12-09 12:50:00       4.15     12680.0          France
541907  2011-12-09 12:50:00       4.15     12680.0          France
541908  2011-12-09 12:50:00       4.95     12680.0          France

[406829 rows x 8 columns]
```

[9]:

```
#Checking if there are no null values
data.isnull().sum()*100/data.shape[0]
```

[9]:

```
InvoiceNo      0.0
StockCode      0.0
Description    0.0
Quantity       0.0
InvoiceDate    0.0
UnitPrice      0.0
CustomerID     0.0
```

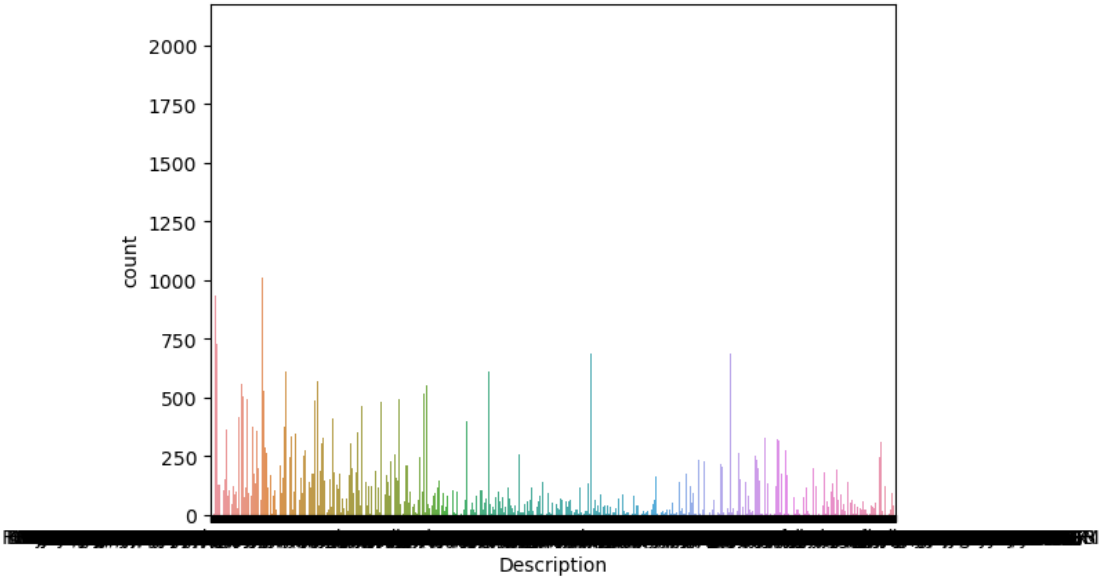
```
Country        0.0
dtype: float64
```

[10]:
```
#Checking duplicates in dataset
data.duplicated()
```

[10]:
```
0         False
1         False
2         False
3         False
4         False
          …
541904    False
541905    False
541906    False
541907    False
541908    False
Length: 406829, dtype: bool
```

[11]:
```
a=data1['Description'].value_counts()
```

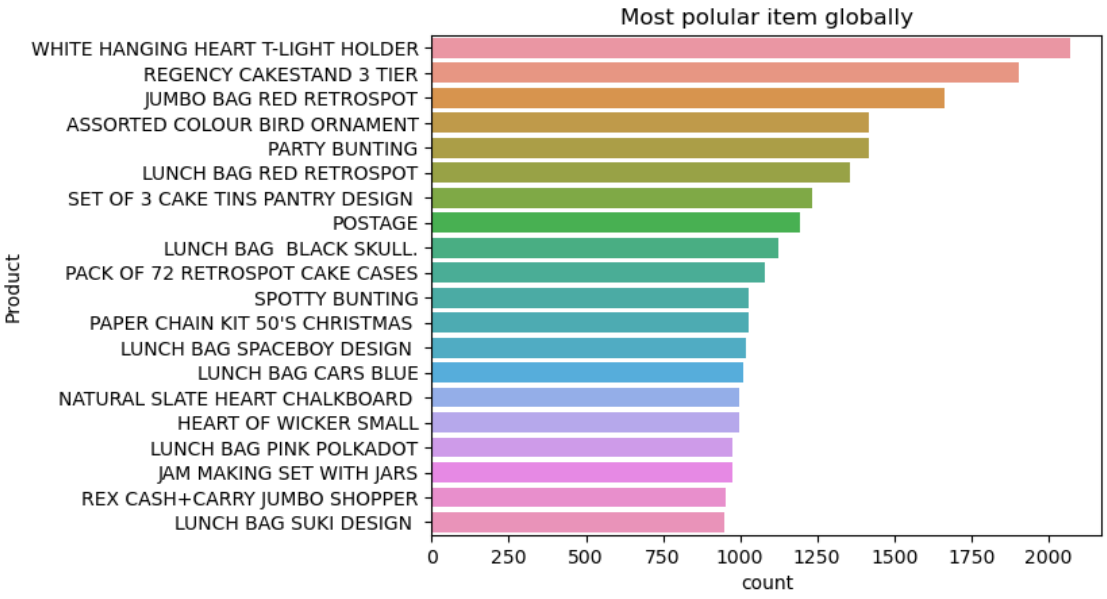
[12]:
```
#Plotting dataset
import seaborn as sns
import matplotlib.pyplot as plt
sns.countplot(x='Description',data=data)
plt.show()
```



```
[13]: #Plotting most popular item globally
      df1=data['Description'].value_counts().rename_axis('Product').
       ↪reset_index(name='count')
      df1
```

```
[13]:                               Product  count
      0     WHITE HANGING HEART T-LIGHT HOLDER   2070
      1               REGENCY CAKESTAND 3 TIER   1905
      2                JUMBO BAG RED RETROSPOT   1662
      3          ASSORTED COLOUR BIRD ORNAMENT   1418
      4                          PARTY BUNTING   1416
      …                                    …     …
      3891    ANTIQUE RASPBERRY FLOWER EARRINGS      1
      3892              WALL ART,ONLY ONE PERSON      1
      3893      GOLD/AMBER DROP EARRINGS W LEAF      1
      3894                 INCENSE BAZAAR PEACH      1
      3895     PINK BAROQUE FLOCK CANDLE HOLDER      1

      [3896 rows x 2 columns]
```

```
[14]: sns.barplot(y=df1['Product'].head(20),x=df1['count'].head(20),data=df1)
      plt.title('Most polular item globally')
      plt.show()
```



```
[15]: data['month']=data["InvoiceDate"].str[5:7]
      data
```

```
C:\Users\Lenovo\AppData\Local\Temp\ipykernel_10372\1925164116.py:1:
SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  data['month']=data["InvoiceDate"].str[5:7]
```

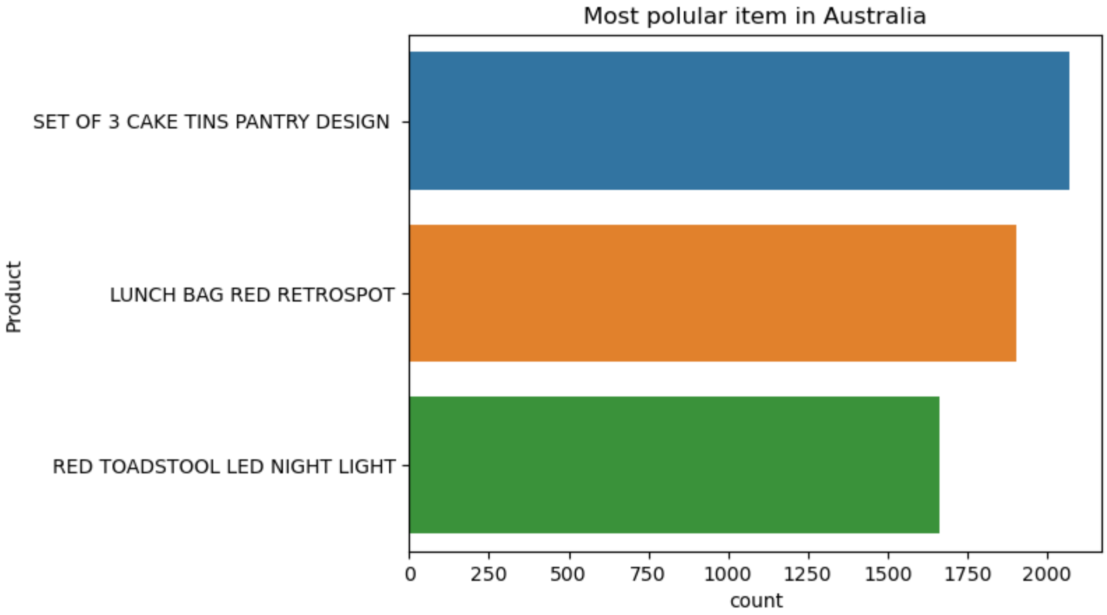
[15]:

```
       InvoiceNo StockCode                          Description  Quantity  \
0         536365    85123A   WHITE HANGING HEART T-LIGHT HOLDER         6
1         536365     71053                  WHITE METAL LANTERN         6
2         536365    84406B       CREAM CUPID HEARTS COAT HANGER         8
3         536365    84029G  KNITTED UNION FLAG HOT WATER BOTTLE         6
4         536365    84029E       RED WOOLLY HOTTIE WHITE HEART.         6
…            …         …                                  …       …
541904    581587     22613          PACK OF 20 SPACEBOY NAPKINS        12
541905    581587     22899         CHILDREN'S APRON DOLLY GIRL          6
541906    581587     23254        CHILDRENS CUTLERY DOLLY GIRL          4
541907    581587     23255      CHILDRENS CUTLERY CIRCUS PARADE         4
541908    581587     22138        BAKING SET 9 PIECE RETROSPOT          3

                InvoiceDate  UnitPrice  CustomerID         Country month
0       2010-12-01 08:26:00       2.55     17850.0  United Kingdom    12
1       2010-12-01 08:26:00       3.39     17850.0  United Kingdom    12
2       2010-12-01 08:26:00       2.75     17850.0  United Kingdom    12
3       2010-12-01 08:26:00       3.39     17850.0  United Kingdom    12
4       2010-12-01 08:26:00       3.39     17850.0  United Kingdom    12
…                       …        …          …             …   …
541904  2011-12-09 12:50:00       0.85     12680.0          France    12
541905  2011-12-09 12:50:00       2.10     12680.0          France    12
541906  2011-12-09 12:50:00       4.15     12680.0          France    12
541907  2011-12-09 12:50:00       4.15     12680.0          France    12
541908  2011-12-09 12:50:00       4.95     12680.0          France    12

[406829 rows x 9 columns]
```

[16]:

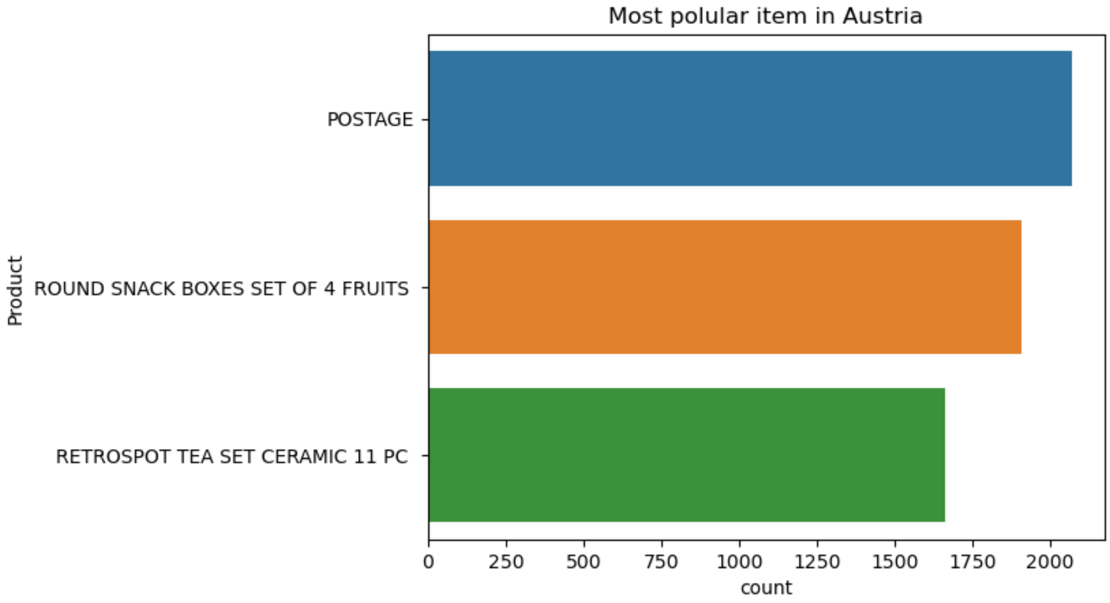
```python
#Grouping dataset by country and plotting them
df2=data.groupby('Country')
i=0
j=1
for name,cont in df2:
    df3=df2.get_group(name)
    df3=df3['Description'].value_counts().rename_axis('Product').
  ↪reset_index(name='count')
    plt.title(f'Most polular item in {name}')
    sns.barplot(y=df3['Product'].head(3),x=df1['count'].head(3),data=df3)
    print("The most popular item is "+df3['Product'].head(1))
    plt.figure(figsize=(3,3))
    i+=1
    plt.show()
```

```
0    The most popular item is SET OF 3 CAKE TINS PA…
Name: Product, dtype: object
```



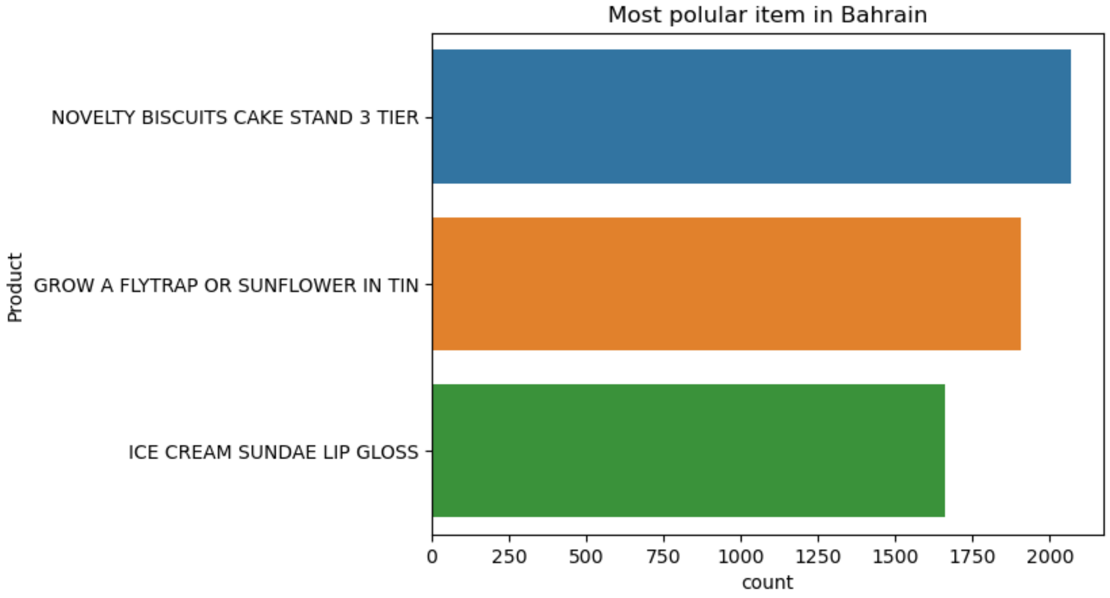
```
<Figure size 300x300 with 0 Axes>
```

```
0    The most popular item is POSTAGE
Name: Product, dtype: object
```



```
<Figure size 300x300 with 0 Axes>
```

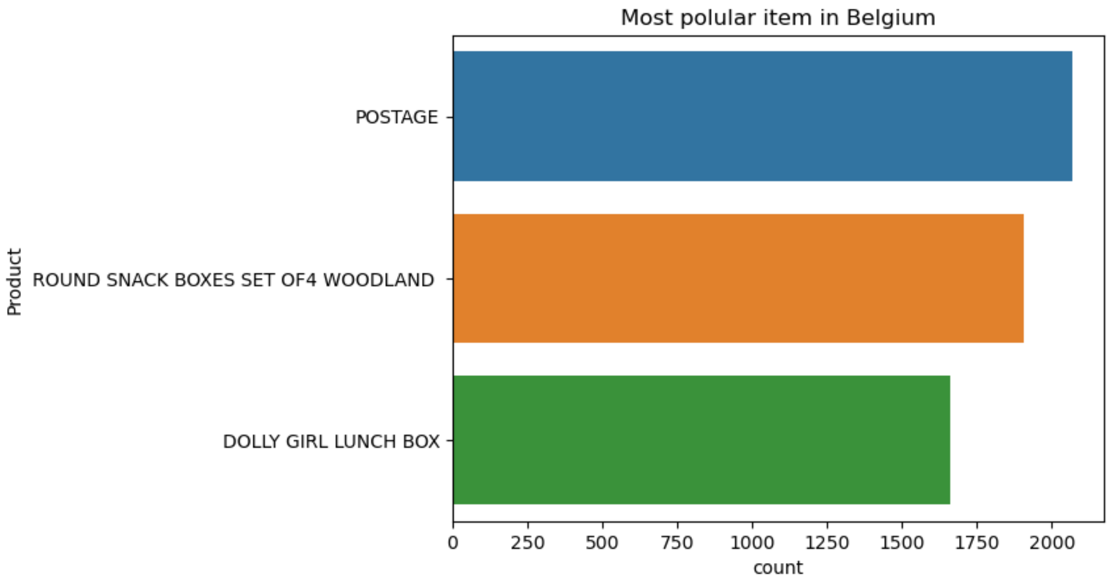
```
0    The most popular item is NOVELTY BISCUITS CAKE…
Name: Product, dtype: object
```



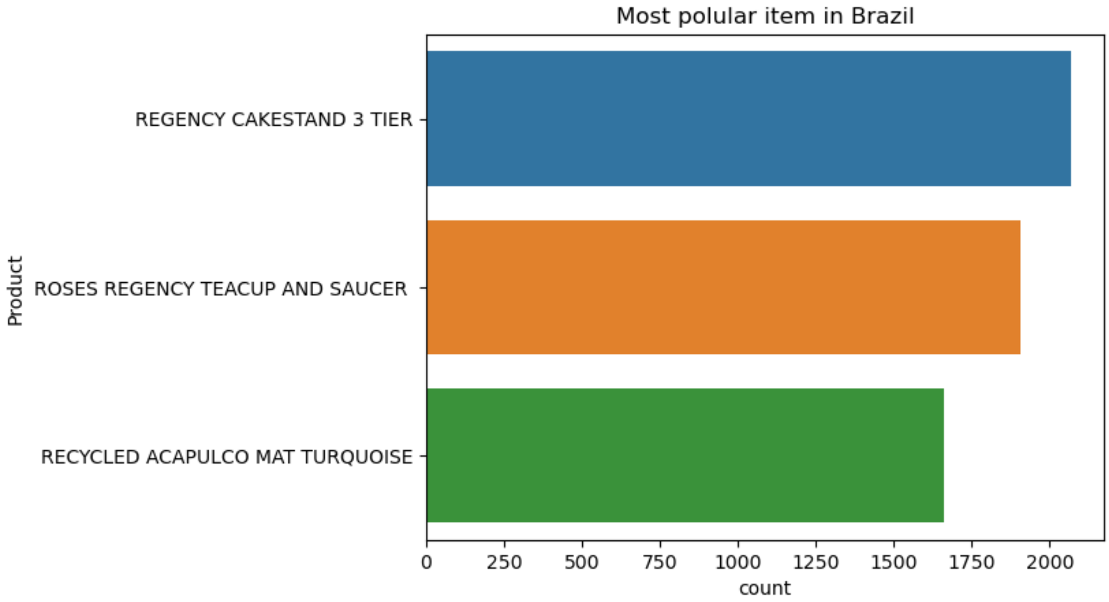
```
<Figure size 300x300 with 0 Axes>
```

```
0    The most popular item is POSTAGE
```

```
Name: Product, dtype: object
```

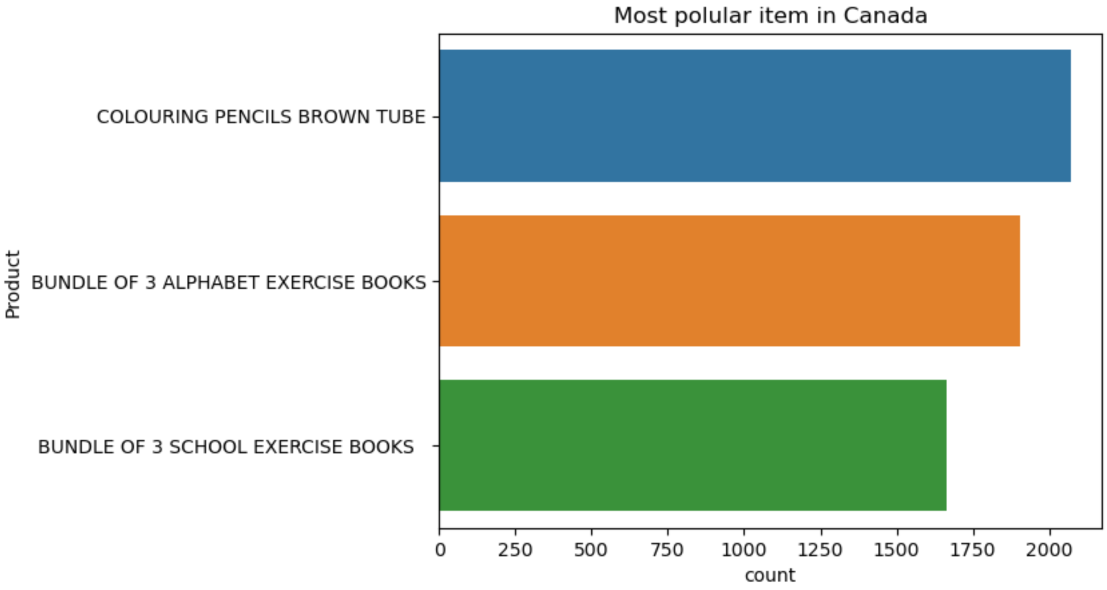


```
<Figure size 300x300 with 0 Axes>

0    The most popular item is REGENCY CAKESTAND 3 TIER
Name: Product, dtype: object
```



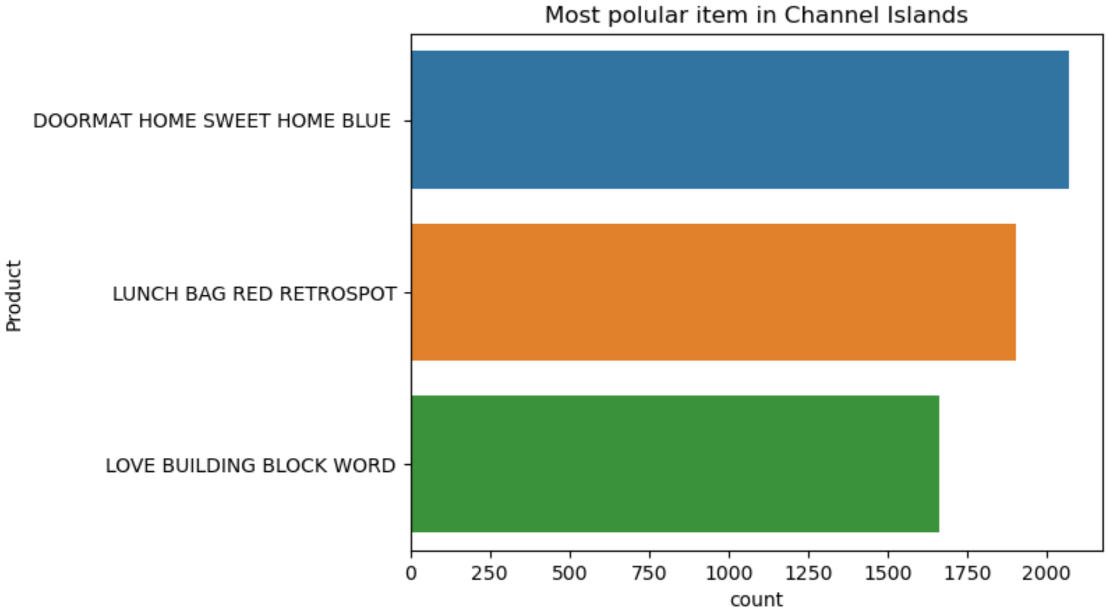
```
<Figure size 300x300 with 0 Axes>
```

```
0    The most popular item is COLOURING PENCILS BRO…
Name: Product, dtype: object
```



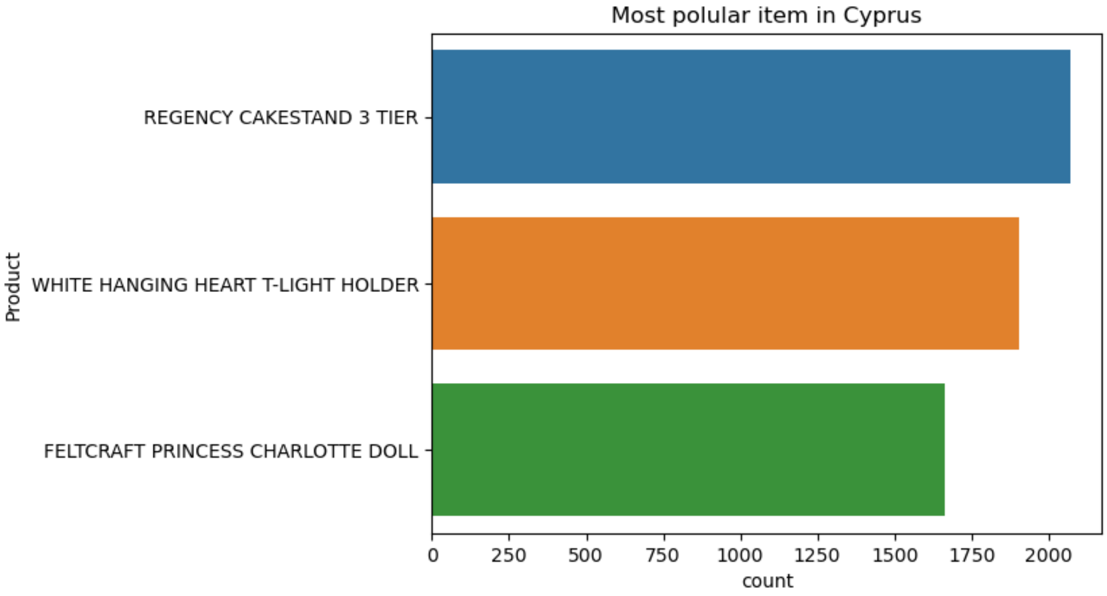
```
<Figure size 300x300 with 0 Axes>
```

```
0    The most popular item is DOORMAT HOME SWEET HO…
Name: Product, dtype: object
```



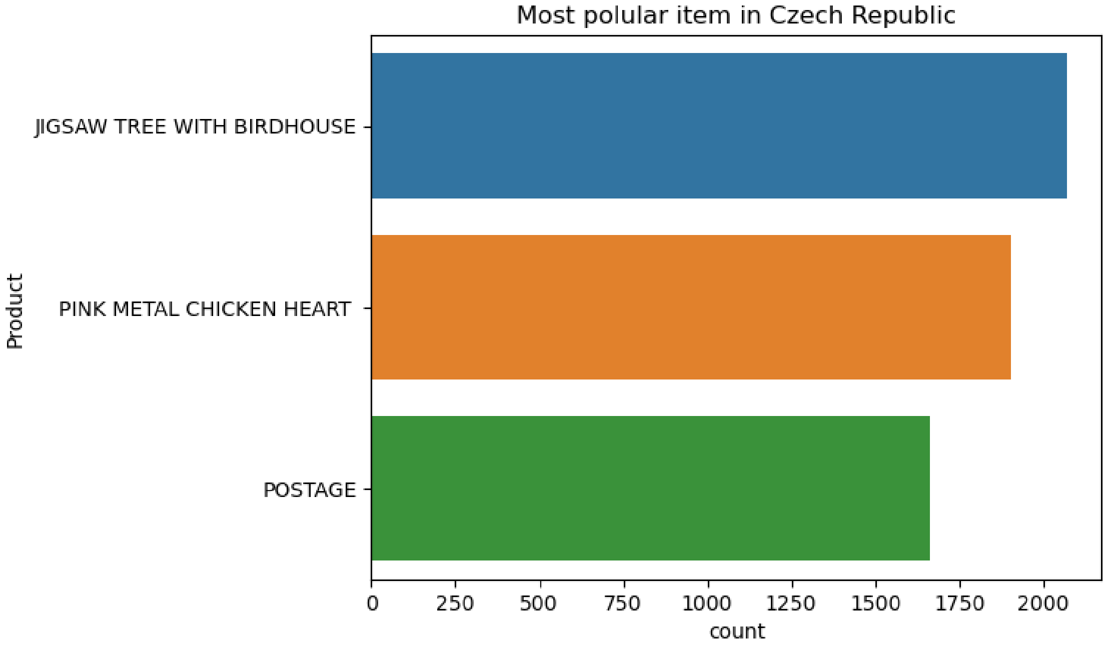
```
<Figure size 300x300 with 0 Axes>
```

```
0    The most popular item is REGENCY CAKESTAND 3 TIER
Name: Product, dtype: object
```



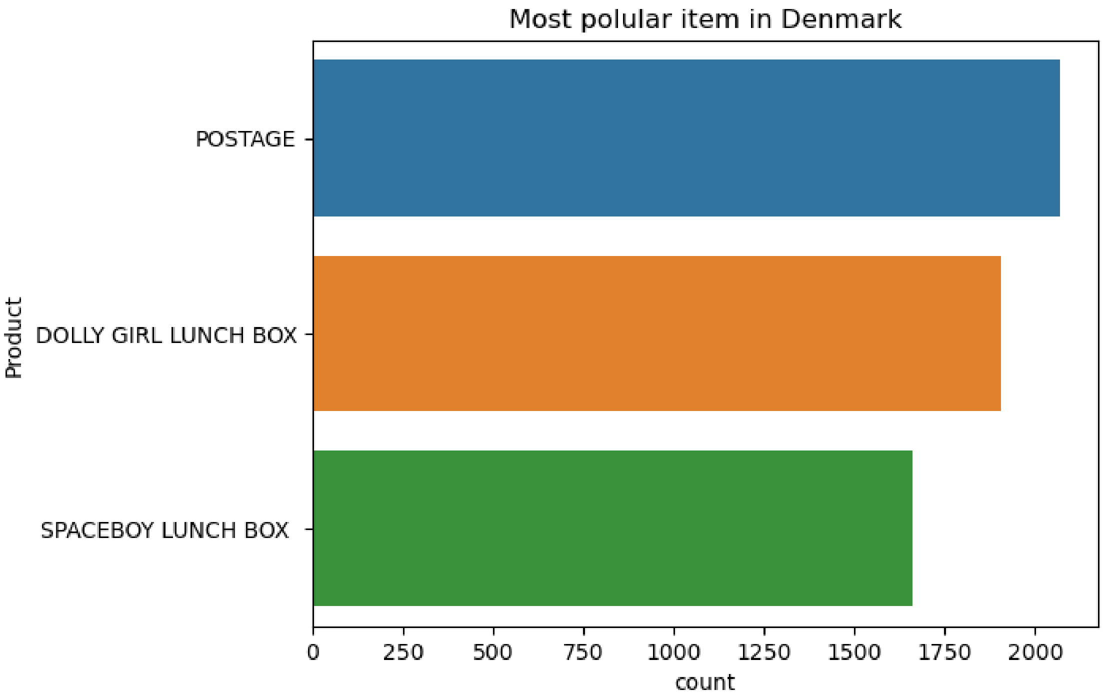
```
<Figure size 300x300 with 0 Axes>
```

```
0    The most popular item is JIGSAW TREE WITH BIRD…
Name: Product, dtype: object
```



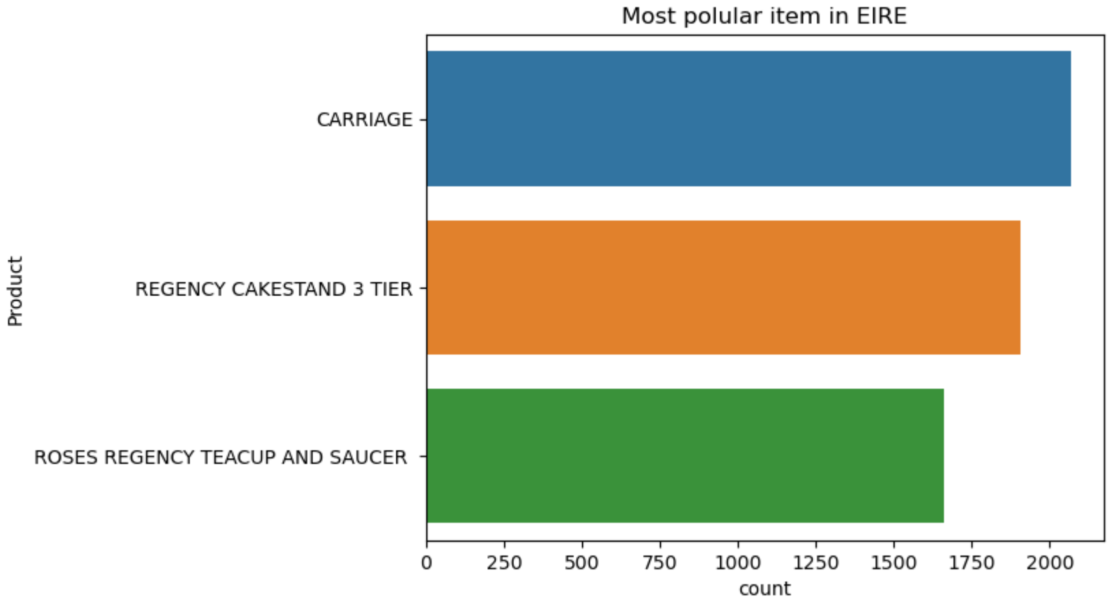
```
<Figure size 300x300 with 0 Axes>
```

```
0    The most popular item is POSTAGE
Name: Product, dtype: object
```



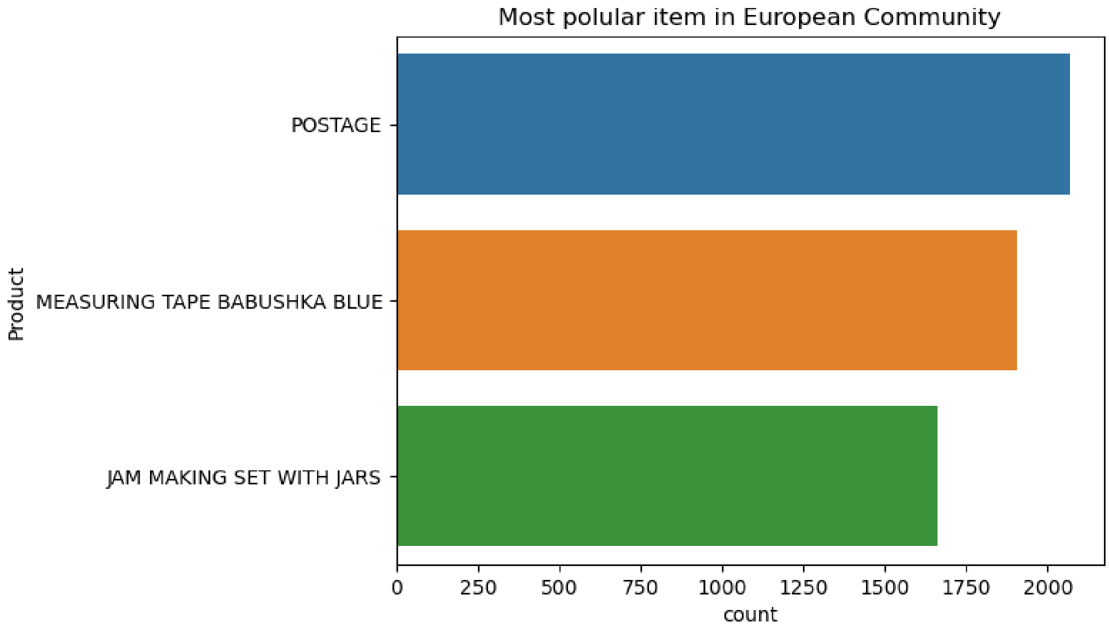
```
<Figure size 300x300 with 0 Axes>
```

```
0    The most popular item is CARRIAGE
Name: Product, dtype: object
```



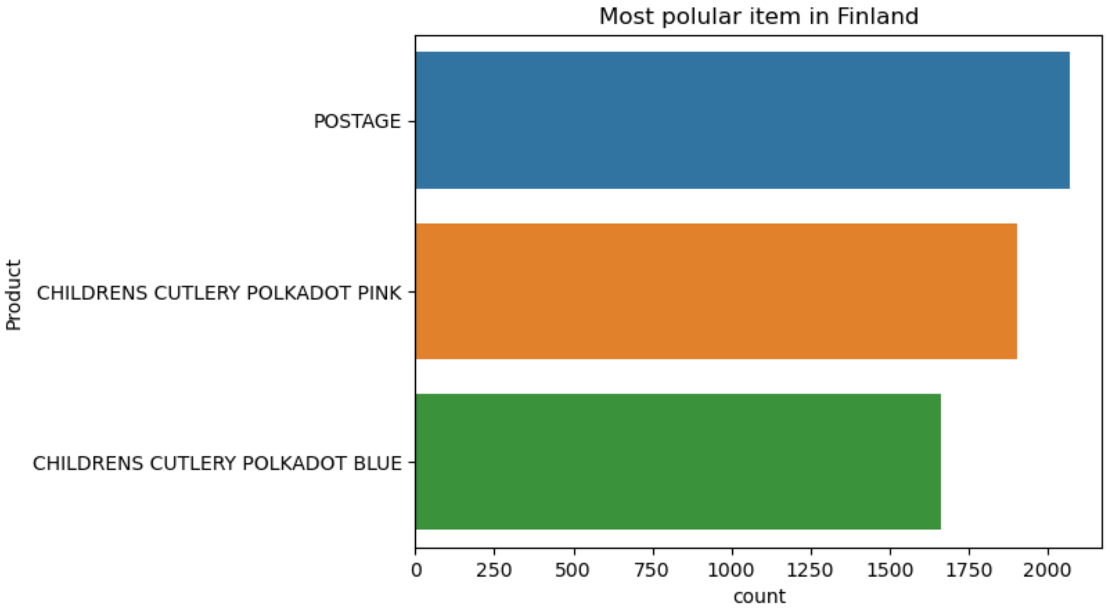
```
<Figure size 300x300 with 0 Axes>
```

```
0    The most popular item is POSTAGE
Name: Product, dtype: object
```



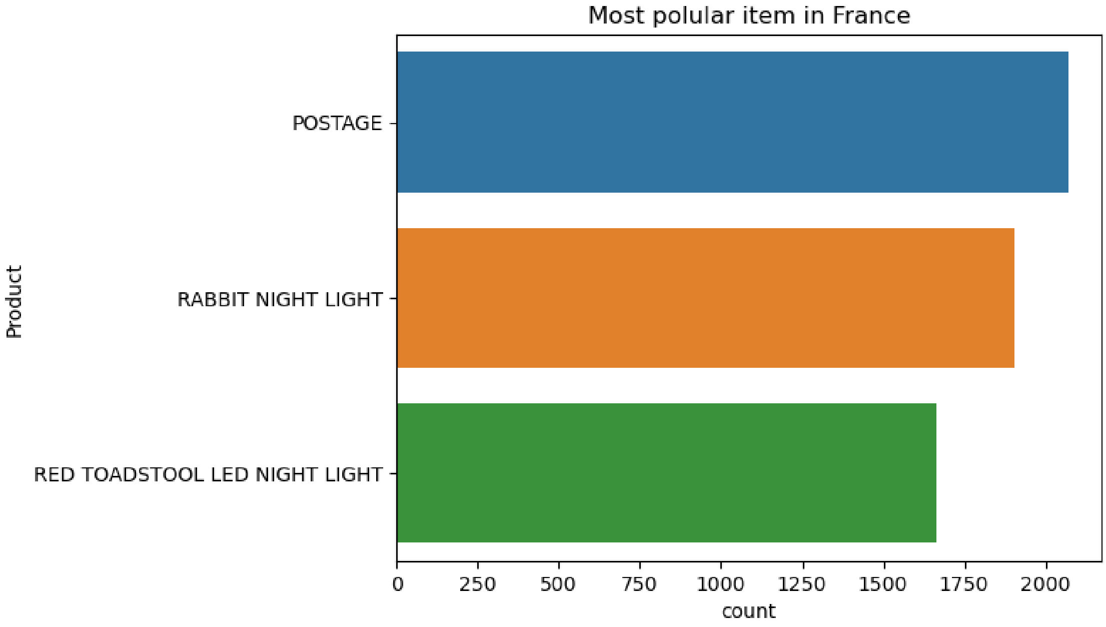
```
<Figure size 300x300 with 0 Axes>
```

```
0    The most popular item is POSTAGE
Name: Product, dtype: object
```



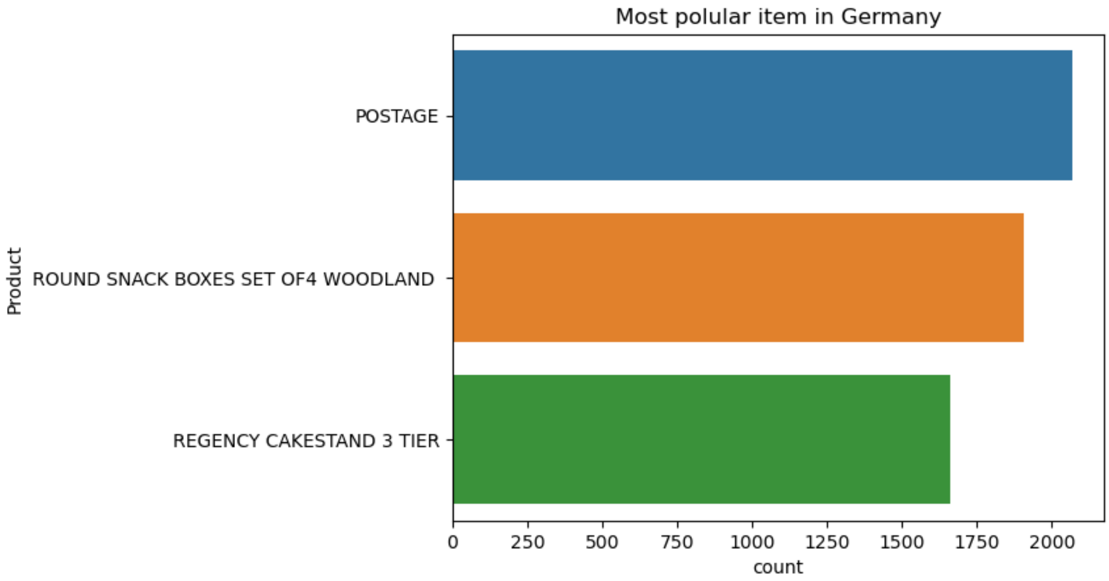
```
<Figure size 300x300 with 0 Axes>
```

```
0    The most popular item is POSTAGE
Name: Product, dtype: object
```



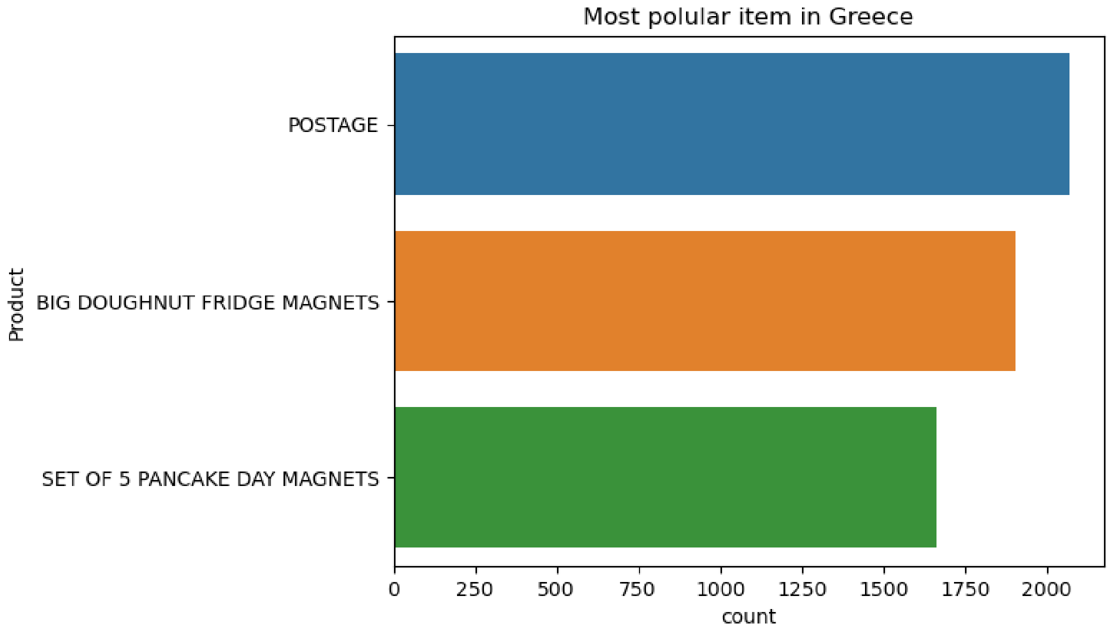
```
<Figure size 300x300 with 0 Axes>
```

```
0    The most popular item is POSTAGE
Name: Product, dtype: object
```



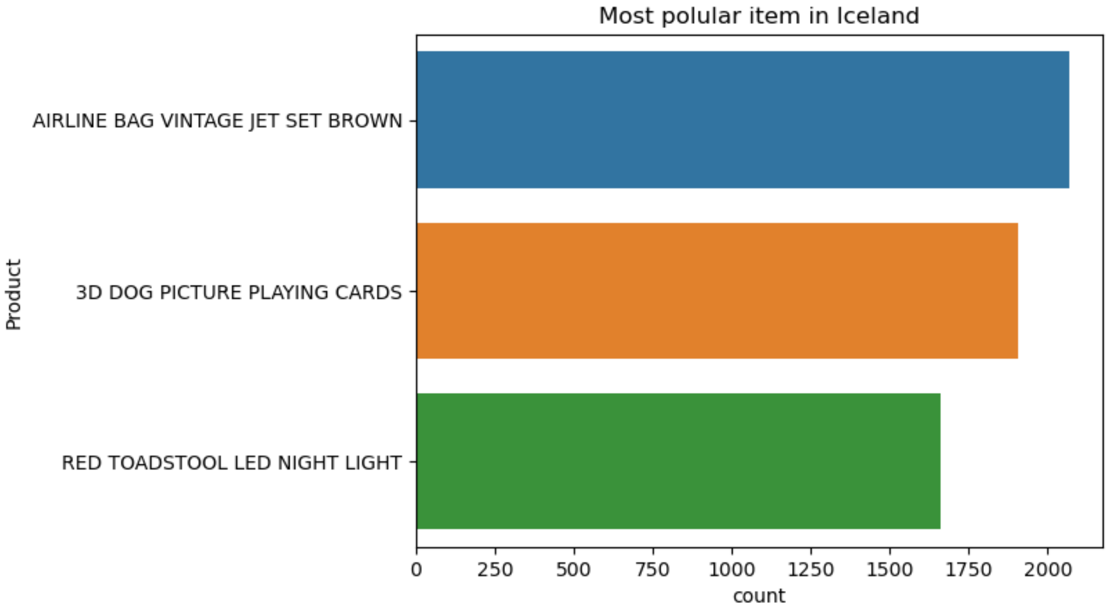
```
<Figure size 300x300 with 0 Axes>
```

```
0    The most popular item is POSTAGE
Name: Product, dtype: object
```



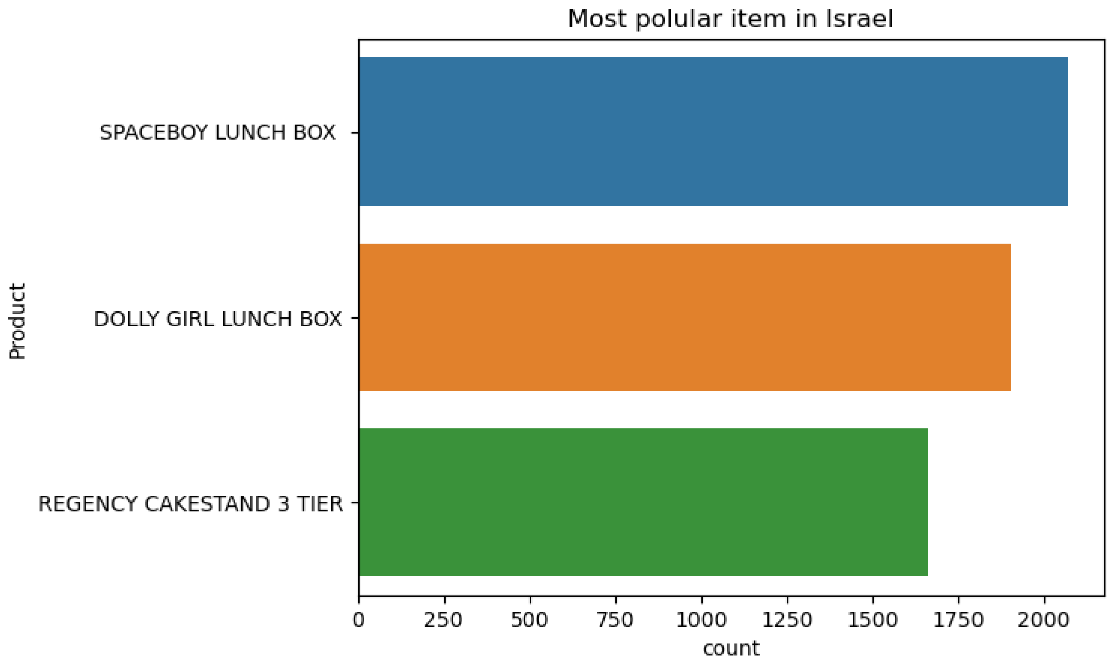
```
<Figure size 300x300 with 0 Axes>
```

```
0    The most popular item is AIRLINE BAG VINTAGE J…
Name: Product, dtype: object
```



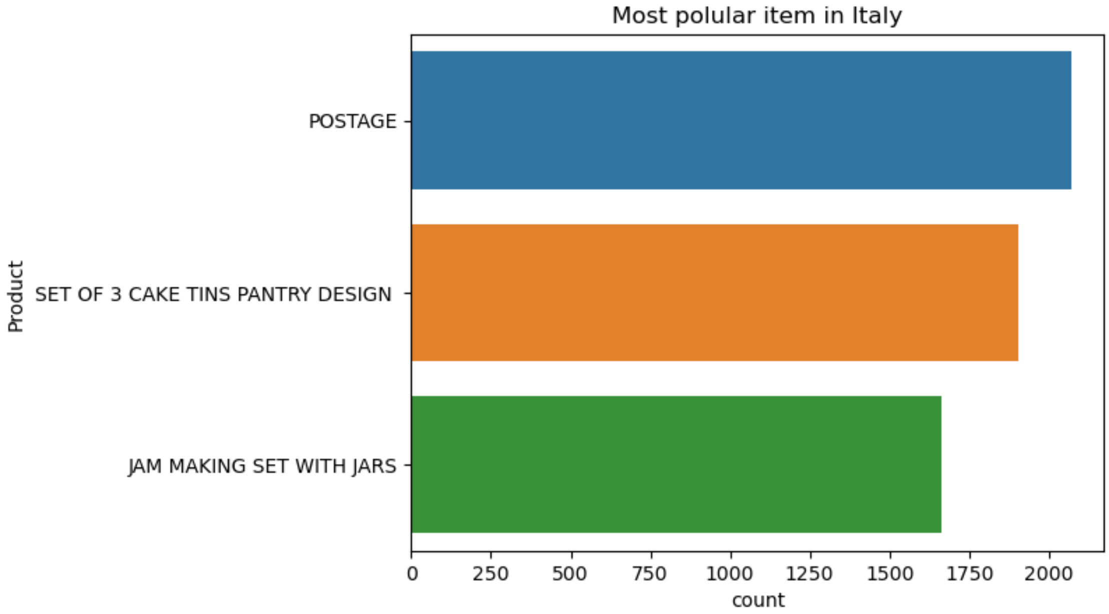
```
<Figure size 300x300 with 0 Axes>
```

```
0    The most popular item is SPACEBOY LUNCH BOX
Name: Product, dtype: object
```



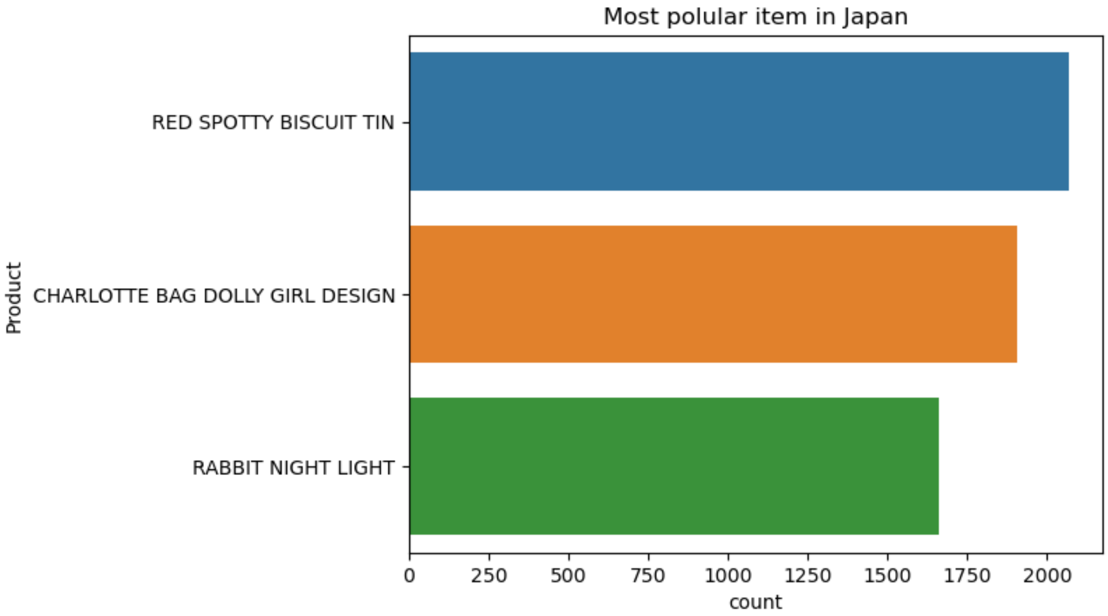
```
<Figure size 300x300 with 0 Axes>
```

```
0    The most popular item is POSTAGE
Name: Product, dtype: object
```



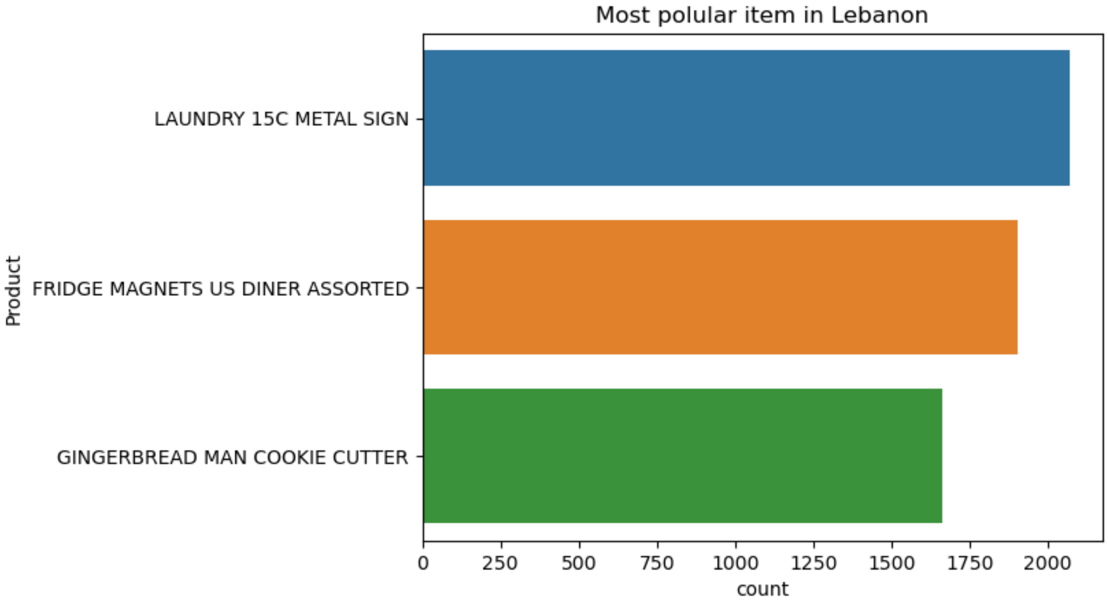
```
<Figure size 300x300 with 0 Axes>
```

```
0    The most popular item is RED SPOTTY BISCUIT TIN
Name: Product, dtype: object
```



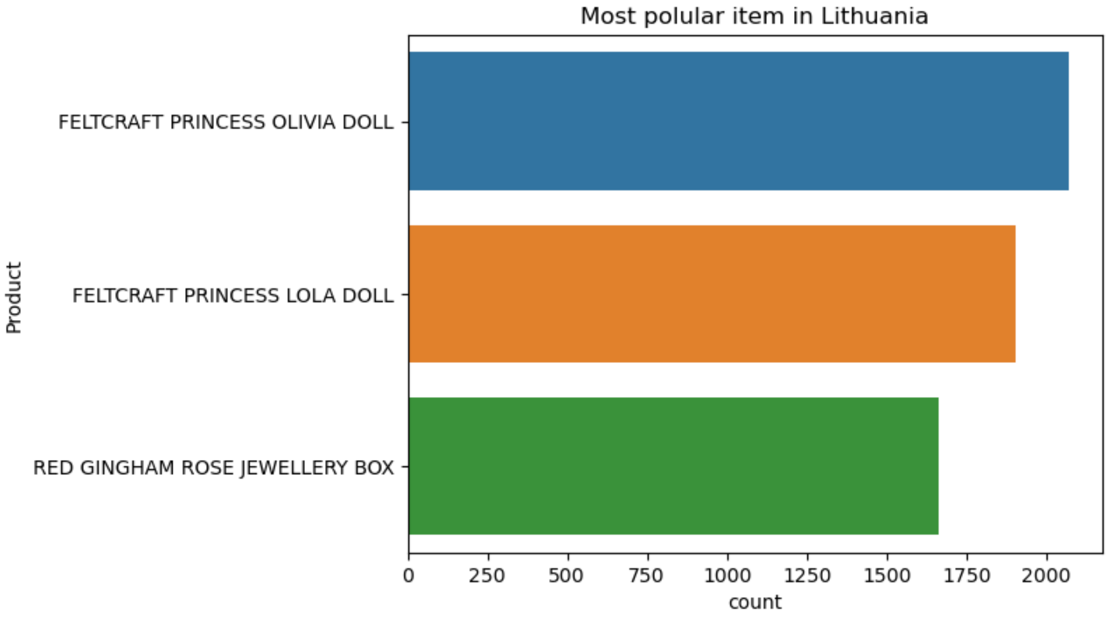
```
<Figure size 300x300 with 0 Axes>
```

```
0    The most popular item is LAUNDRY 15C METAL SIGN
Name: Product, dtype: object
```



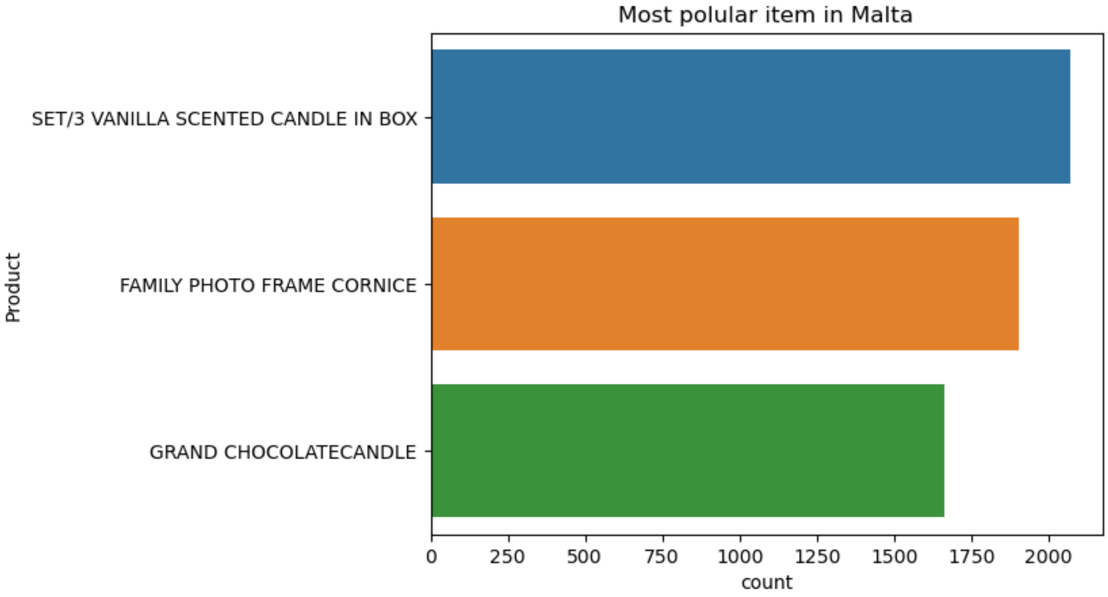
```
<Figure size 300x300 with 0 Axes>
```

```
0    The most popular item is FELTCRAFT PRINCESS OL…
Name: Product, dtype: object
```



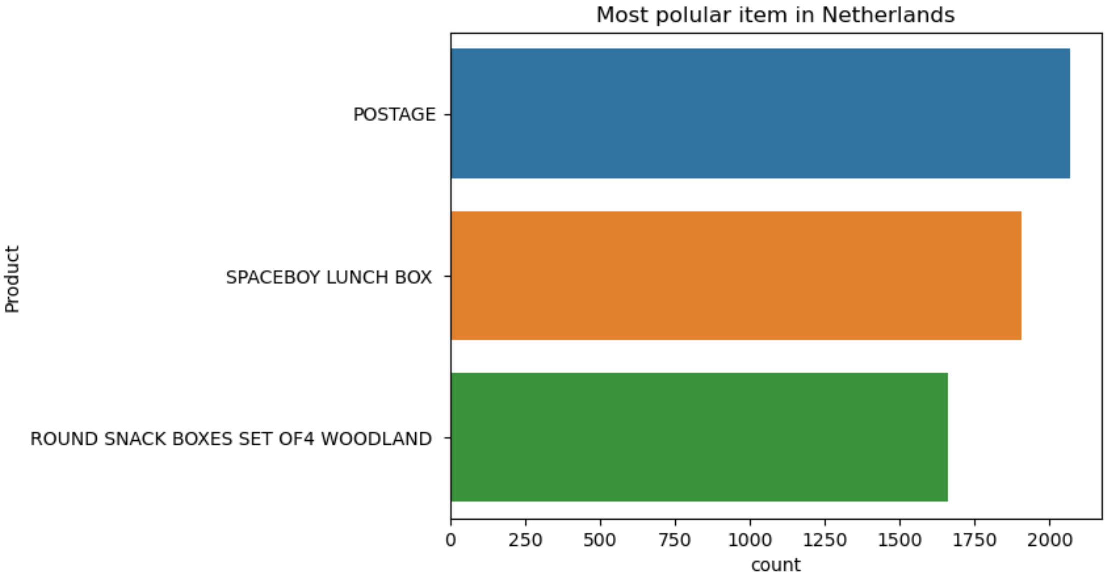
```
<Figure size 300x300 with 0 Axes>
```

```
0    The most popular item is SET/3 VANILLA SCENTED…
Name: Product, dtype: object
```



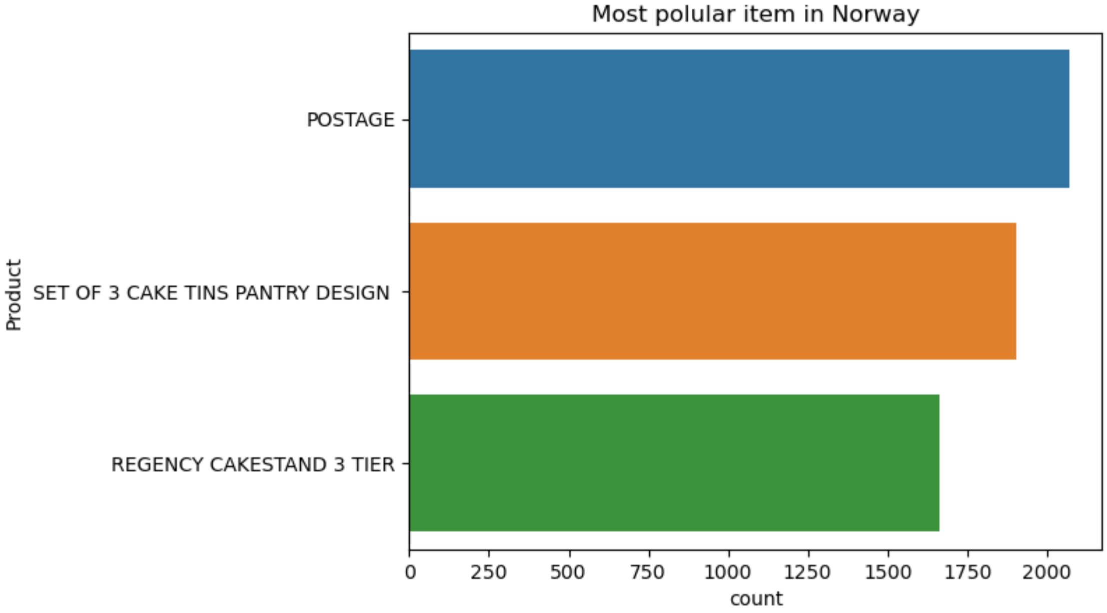
```
<Figure size 300x300 with 0 Axes>
```

```
0    The most popular item is POSTAGE
Name: Product, dtype: object
```



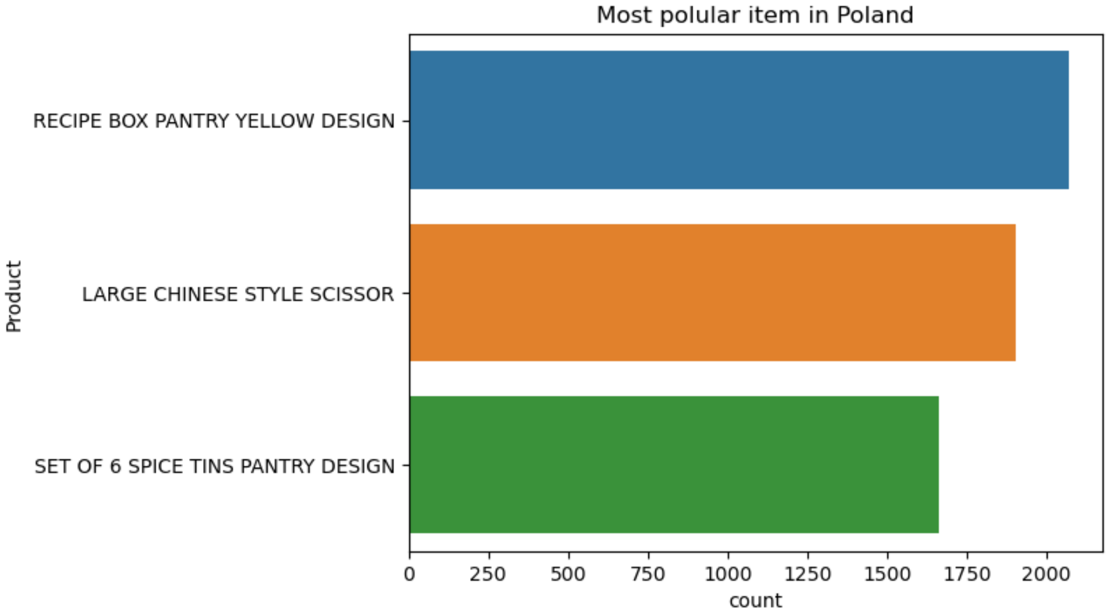
```
<Figure size 300x300 with 0 Axes>
```

```
0    The most popular item is POSTAGE
Name: Product, dtype: object
```



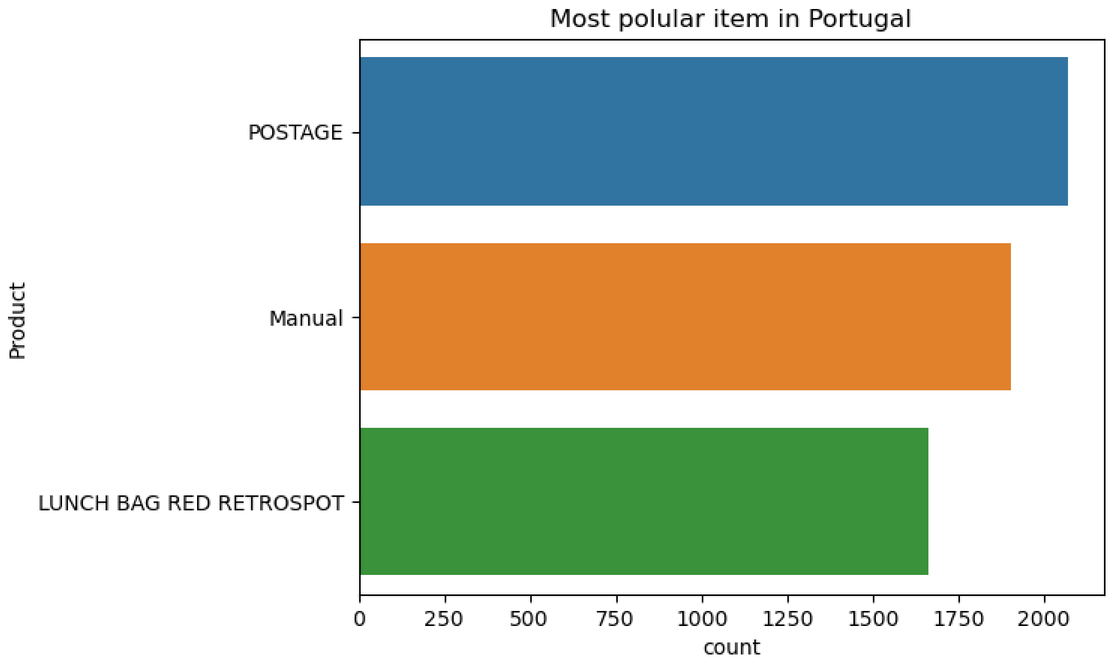
```
<Figure size 300x300 with 0 Axes>
```

```
0    The most popular item is RECIPE BOX PANTRY YEL…
Name: Product, dtype: object
```



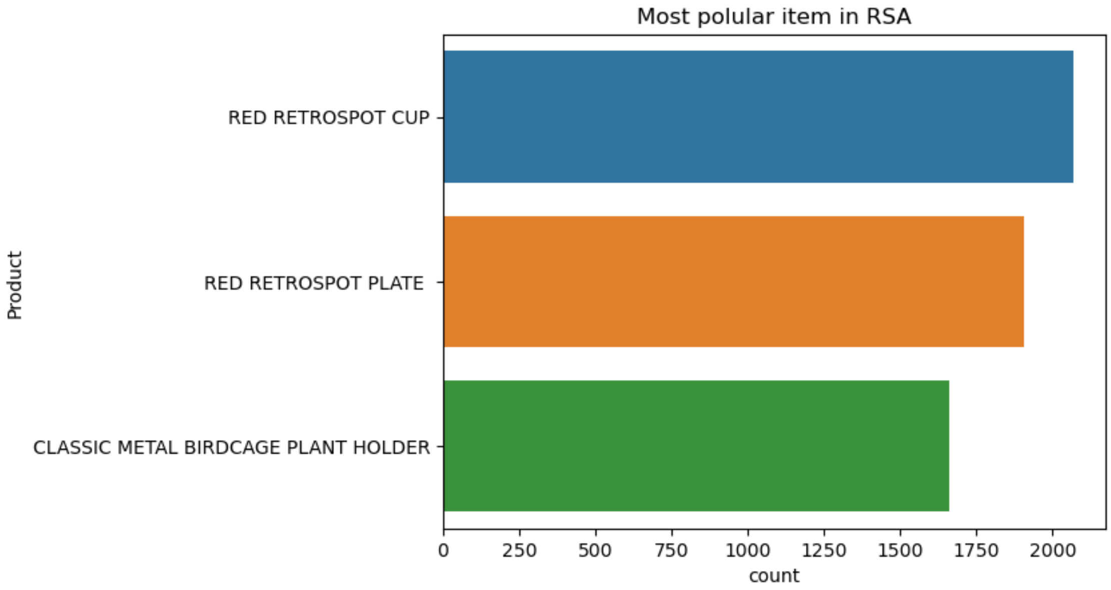
```
<Figure size 300x300 with 0 Axes>
```

```
0    The most popular item is POSTAGE
Name: Product, dtype: object
```



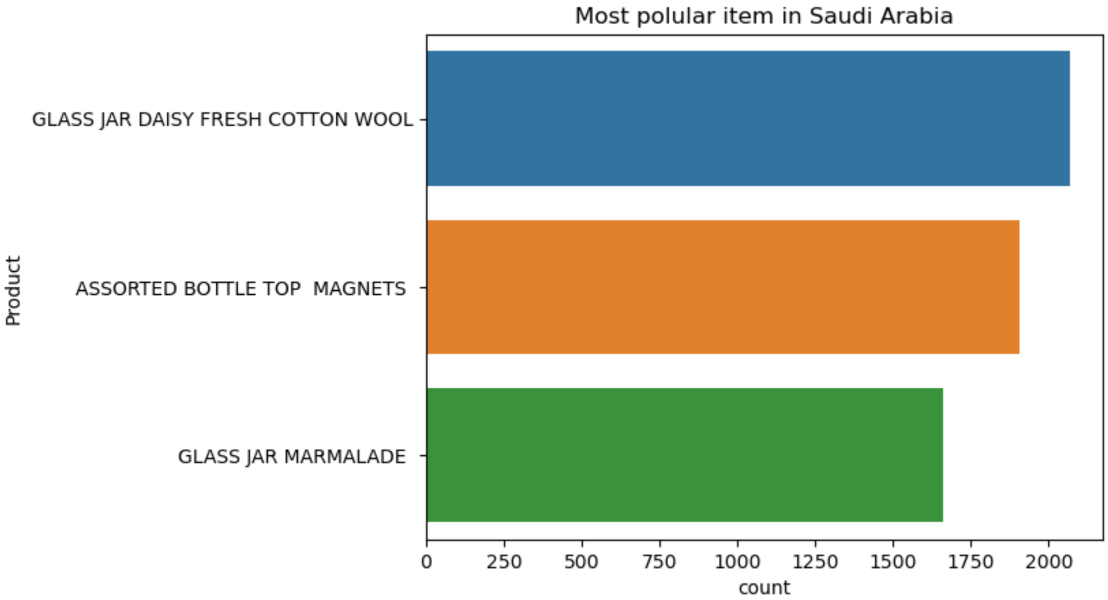
```
<Figure size 300x300 with 0 Axes>
```

```
0    The most popular item is RED RETROSPOT CUP
Name: Product, dtype: object
```



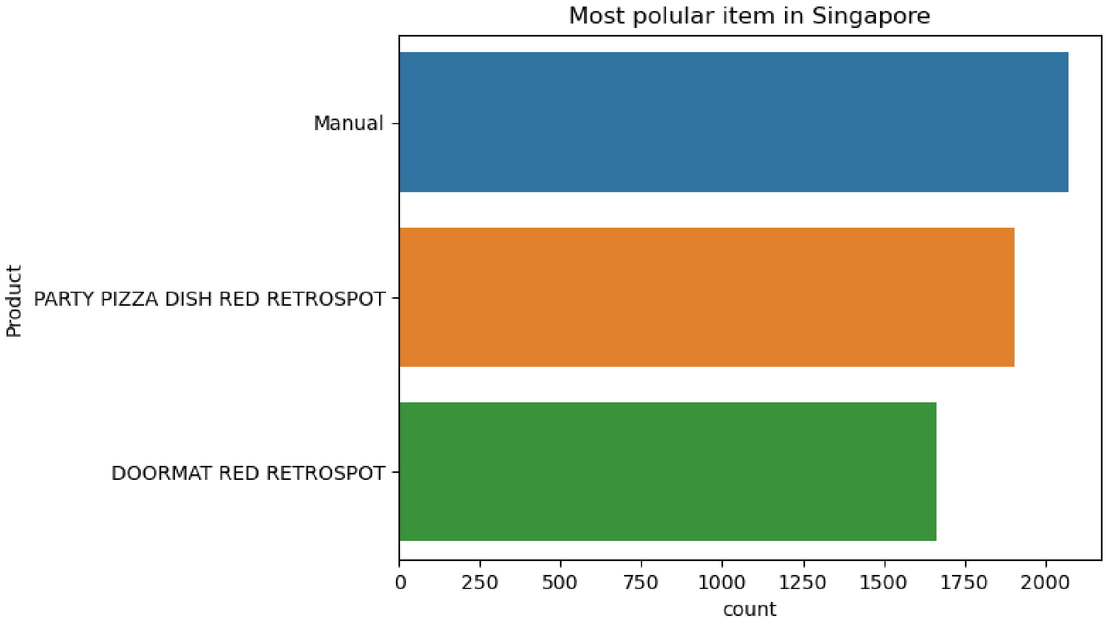
```
<Figure size 300x300 with 0 Axes>
```

```
0    The most popular item is GLASS JAR DAISY FRESH…
Name: Product, dtype: object
```



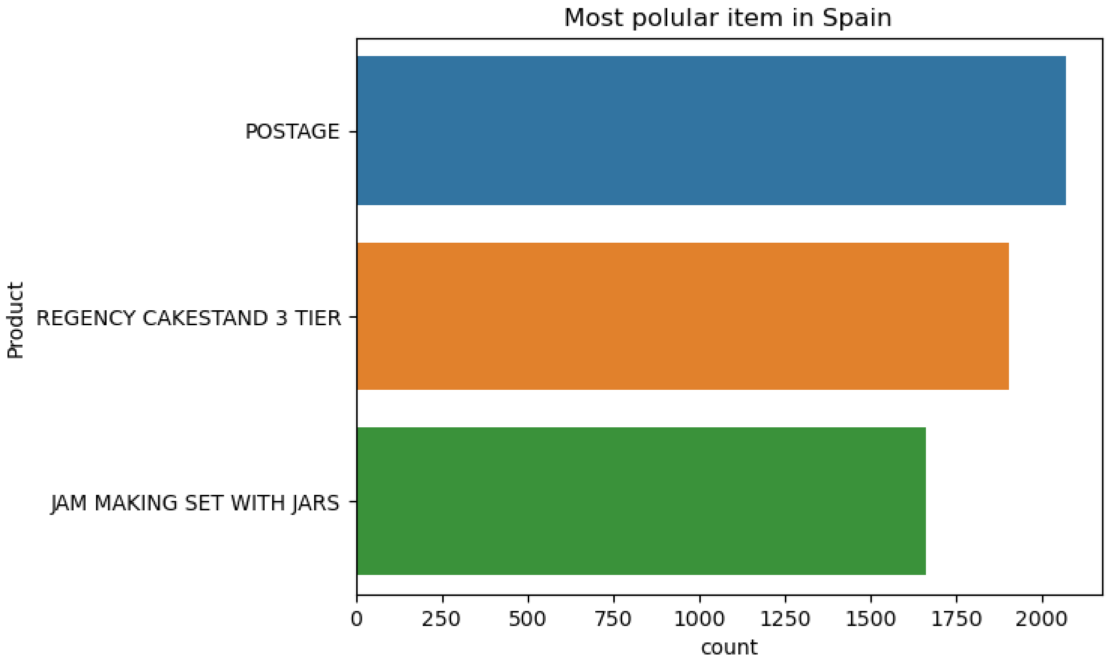
```
<Figure size 300x300 with 0 Axes>
```

```
0    The most popular item is Manual
Name: Product, dtype: object
```



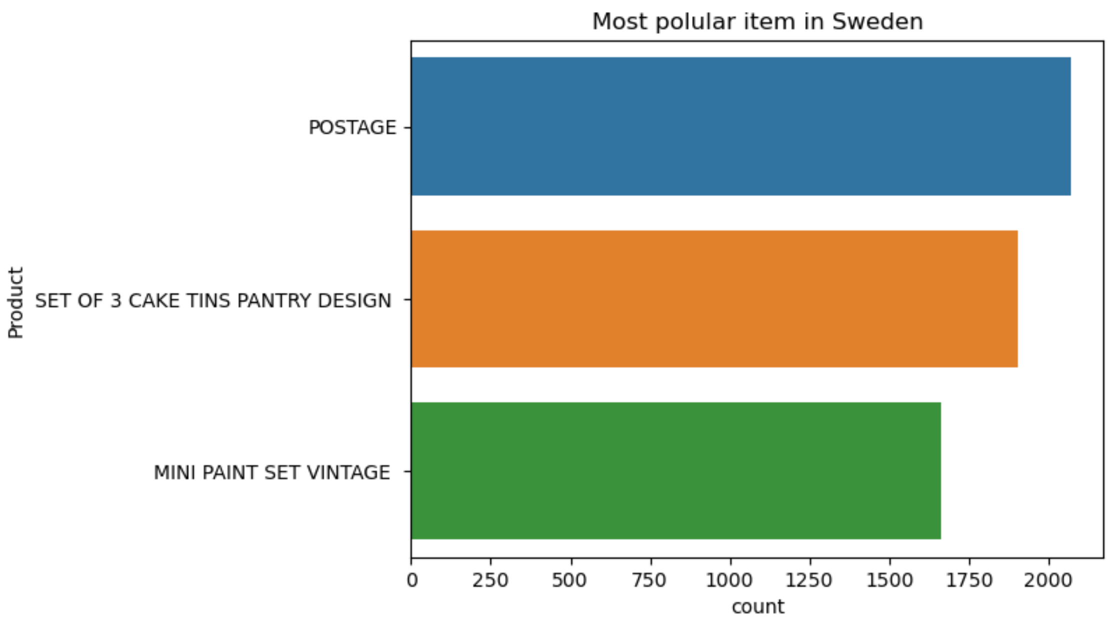
```
<Figure size 300x300 with 0 Axes>
```

```
0    The most popular item is POSTAGE
Name: Product, dtype: object
```



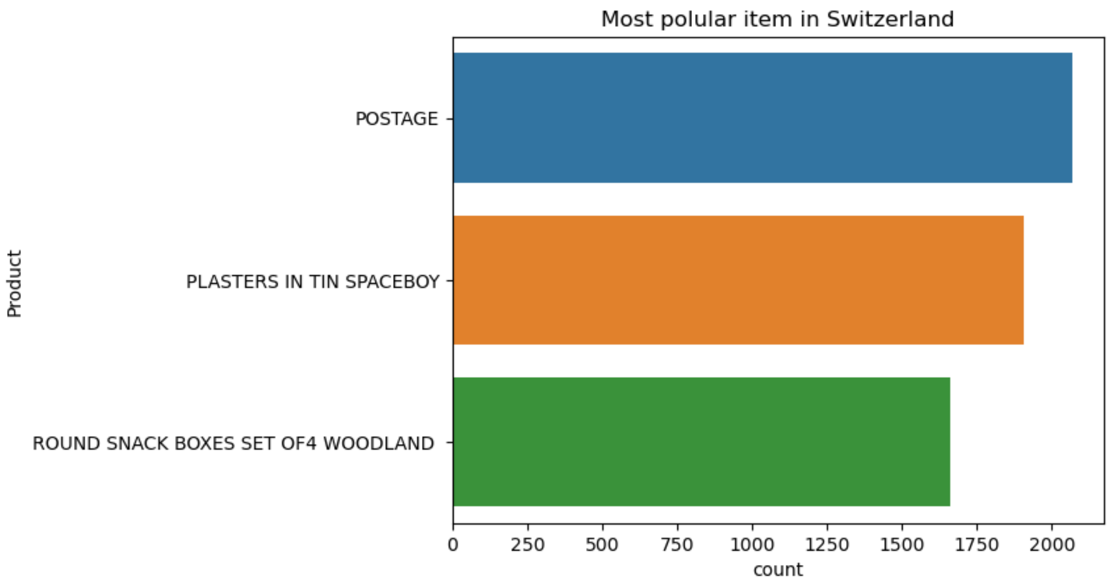
```
<Figure size 300x300 with 0 Axes>
```

```
0    The most popular item is POSTAGE
Name: Product, dtype: object
```



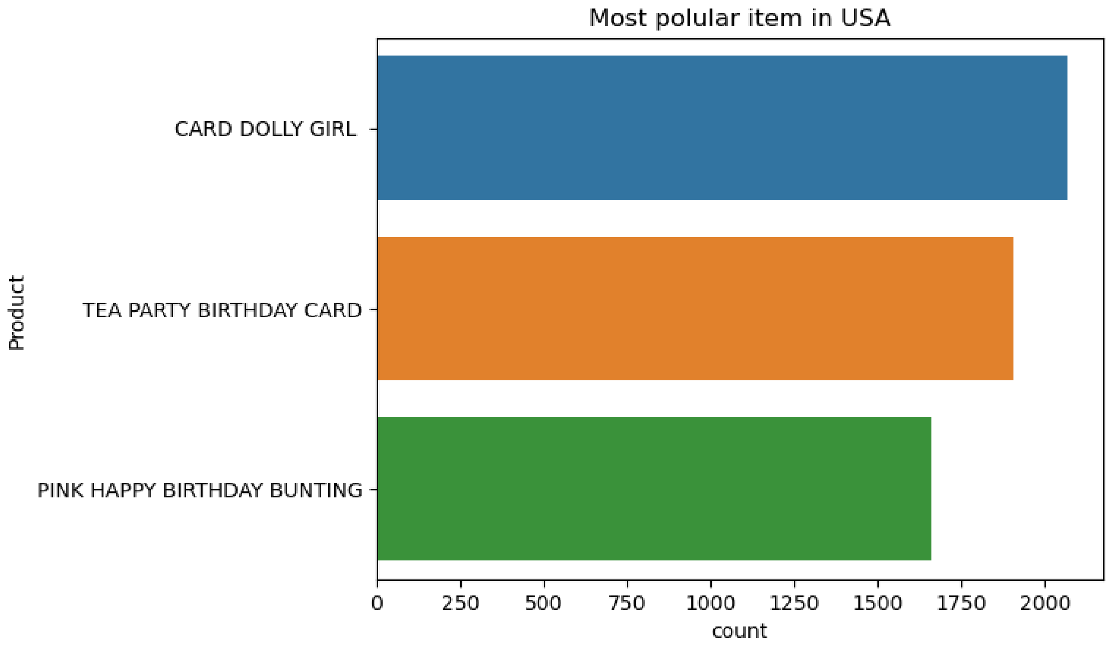
```
<Figure size 300x300 with 0 Axes>
```

```
0    The most popular item is POSTAGE
Name: Product, dtype: object
```



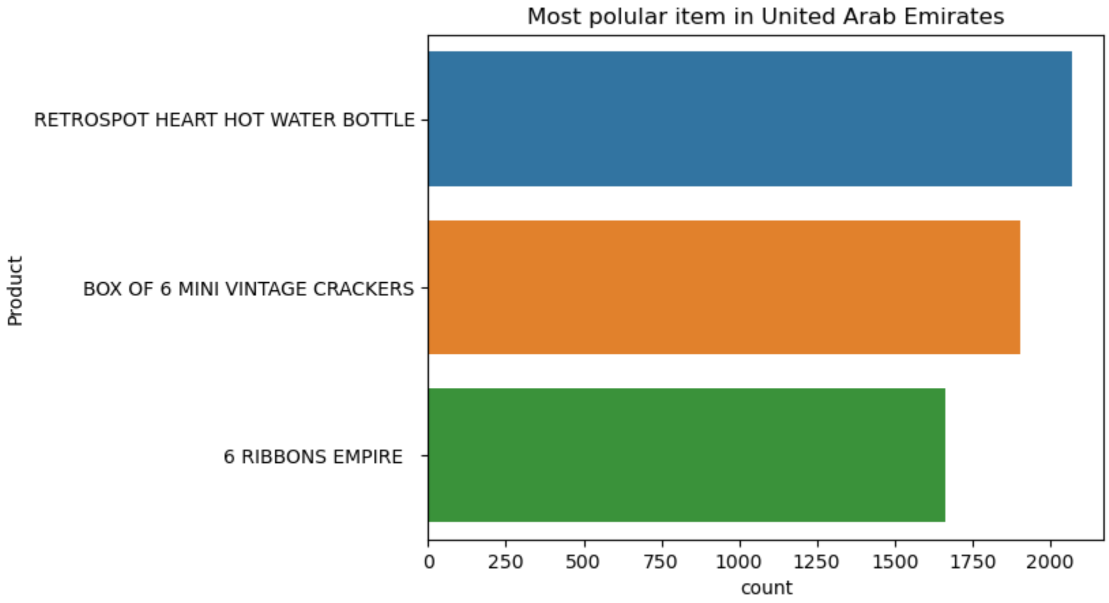
```
<Figure size 300x300 with 0 Axes>
```

```
0    The most popular item is CARD DOLLY GIRL
Name: Product, dtype: object
```



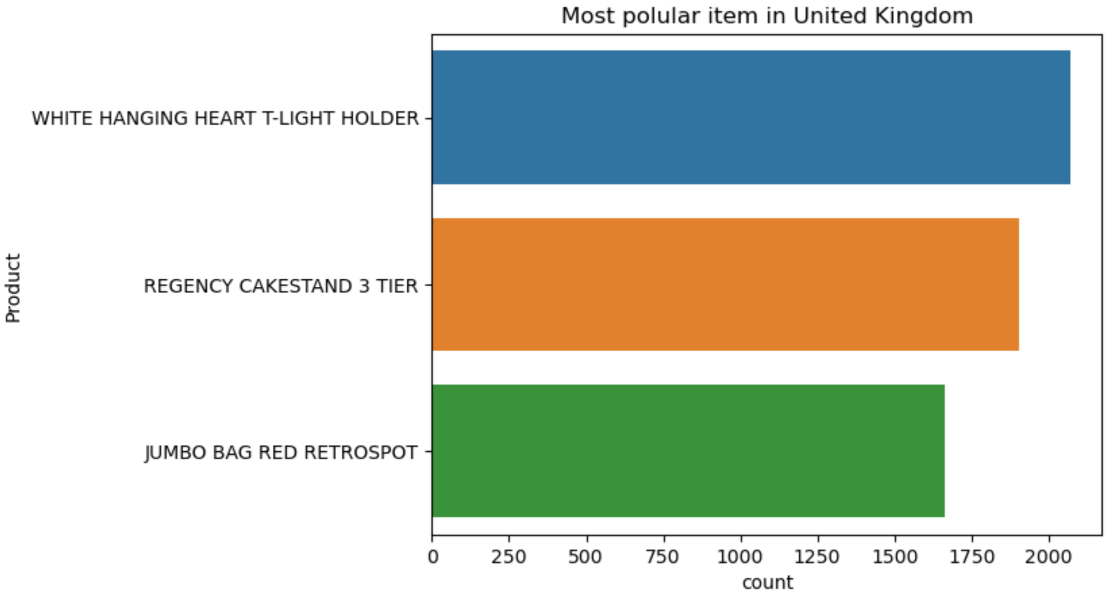
```
<Figure size 300x300 with 0 Axes>
```

```
0    The most popular item is RETROSPOT HEART HOT W…
Name: Product, dtype: object
```



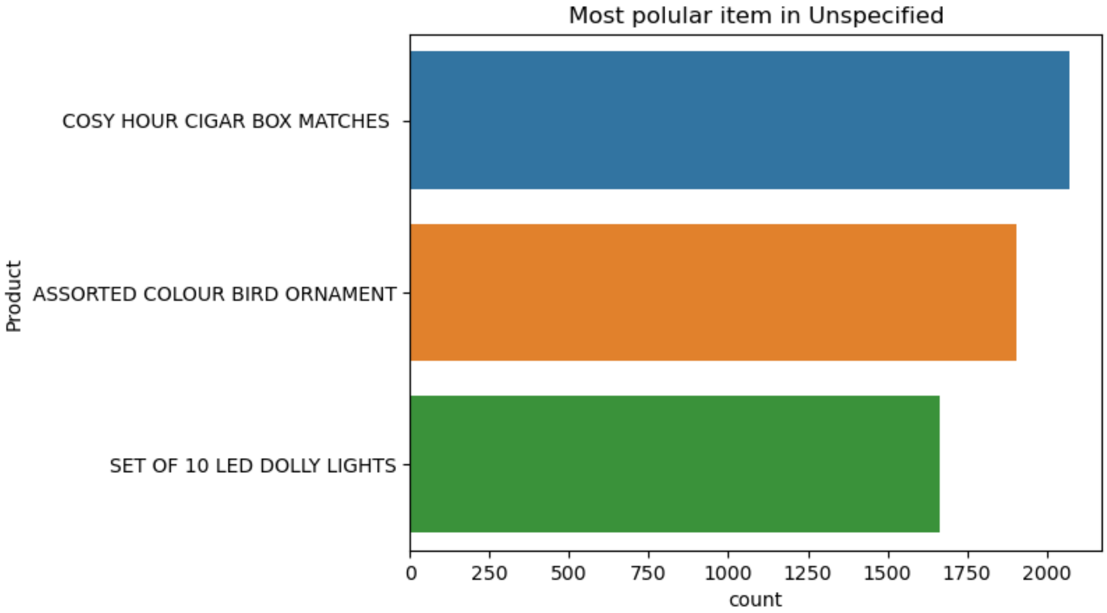
```
<Figure size 300x300 with 0 Axes>
```

```
0    The most popular item is WHITE HANGING HEART T…
Name: Product, dtype: object
```



```
<Figure size 300x300 with 0 Axes>
```

```
0    The most popular item is COSY HOUR CIGAR BOX M…
Name: Product, dtype: object
```

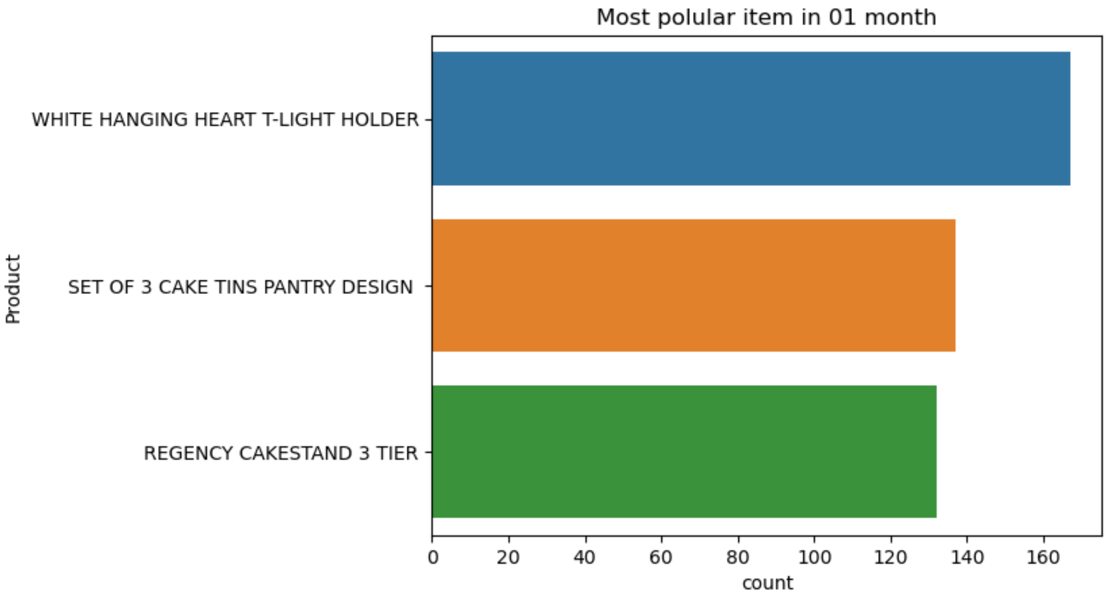


```
<Figure size 300x300 with 0 Axes>
```

[17]:

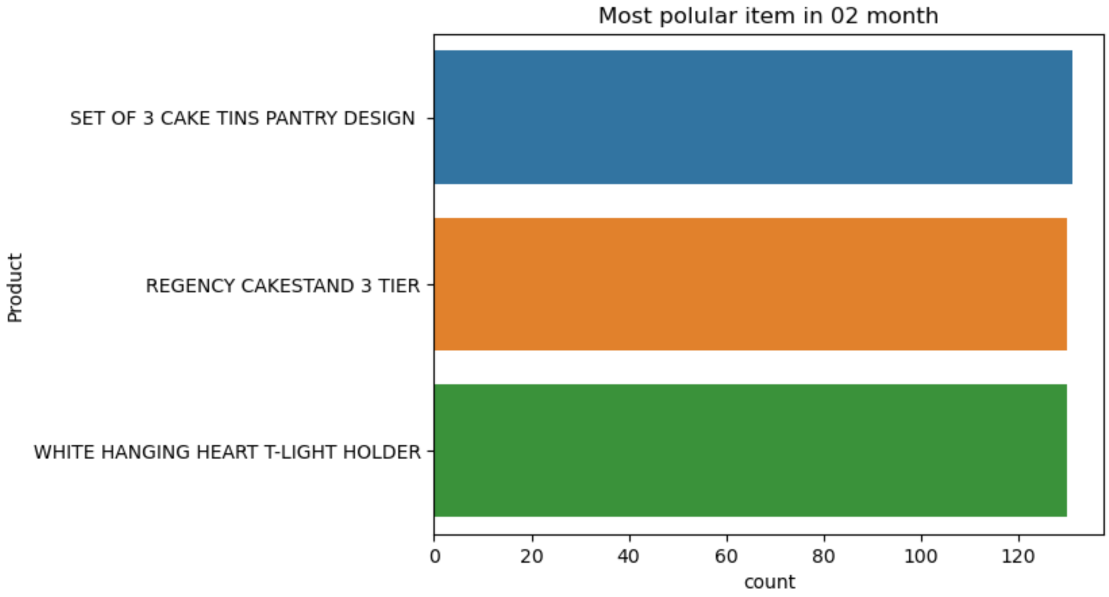
```
#Getting most polpular item monthwise and plotting them
df3=data.groupby('month')
i=0
j=1
for name,cont in df3:
    df4=df3.get_group(name)
    df4=df4['Description'].value_counts().rename_axis('Product').
  ↪reset_index(name='count')
    plt.title(f'Most polular item in {name} month')
    sns.barplot(y=df4['Product'].head(3),x=df4['count'].head(3),data=df4)
    print("The most popular item is "+df4['Product'].head(1))
    plt.figure(figsize=(3,3))
    i+=1
    plt.show()
```

```
0    The most popular item is WHITE HANGING HEART T…
Name: Product, dtype: object
```



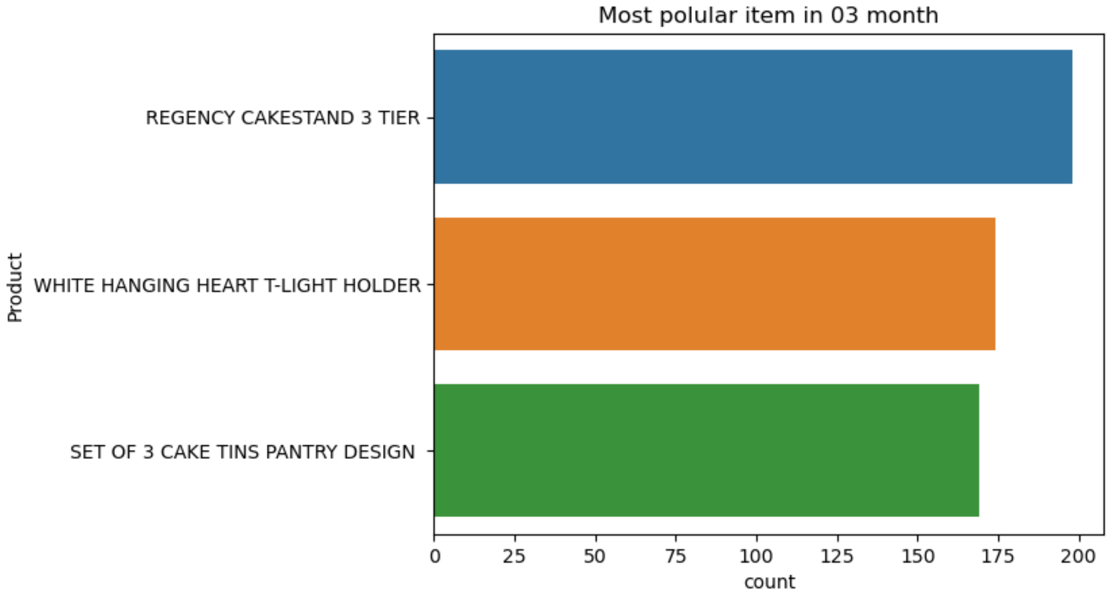
```
<Figure size 300x300 with 0 Axes>
```

```
0    The most popular item is SET OF 3 CAKE TINS PA…
Name: Product, dtype: object
```



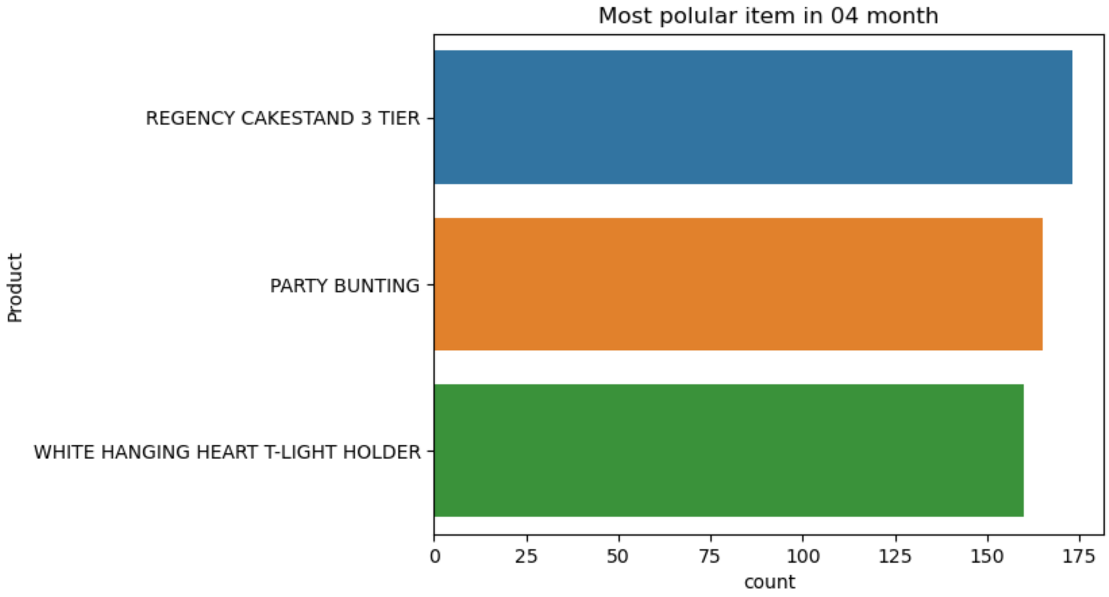
```
<Figure size 300x300 with 0 Axes>
```

```
0    The most popular item is REGENCY CAKESTAND 3 TIER
Name: Product, dtype: object
```



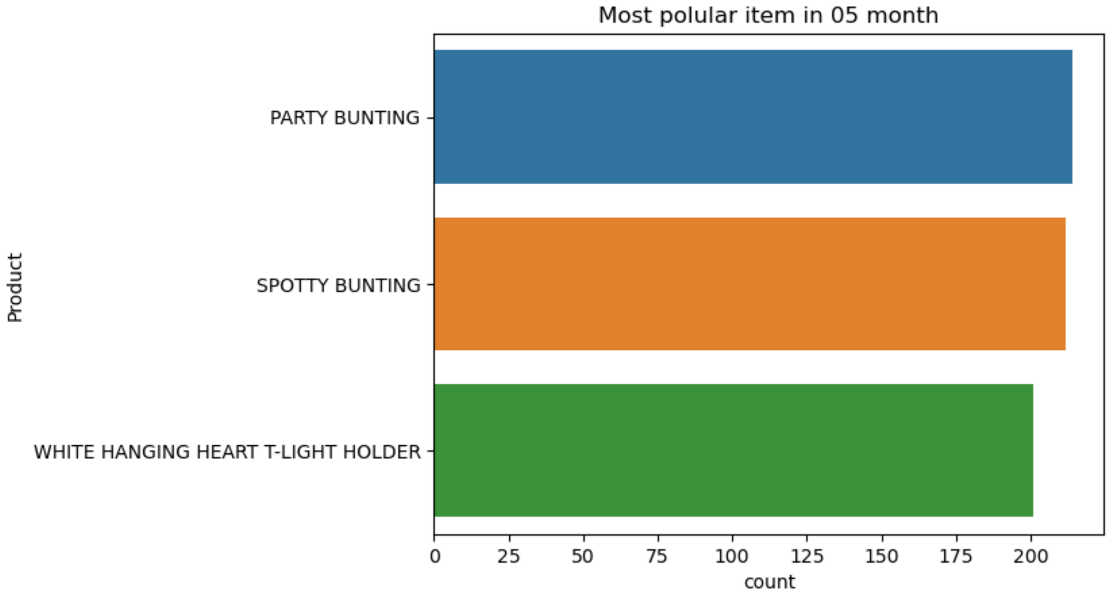
```
<Figure size 300x300 with 0 Axes>
```

```
0    The most popular item is REGENCY CAKESTAND 3 TIER
Name: Product, dtype: object
```



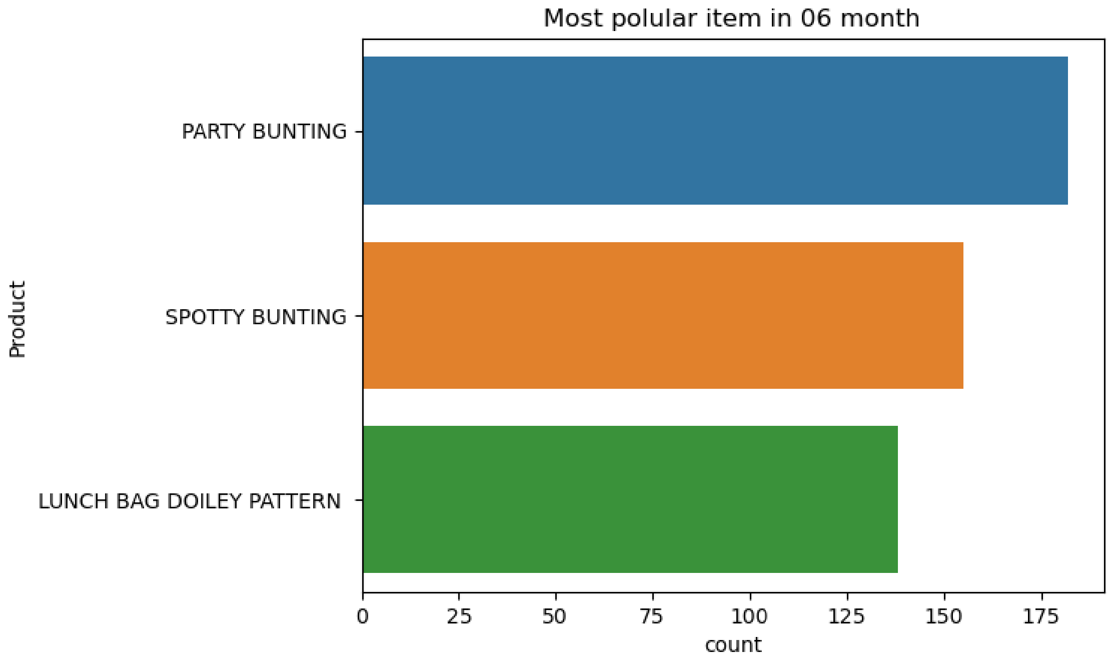
```
<Figure size 300x300 with 0 Axes>
```

```
0    The most popular item is PARTY BUNTING
Name: Product, dtype: object
```



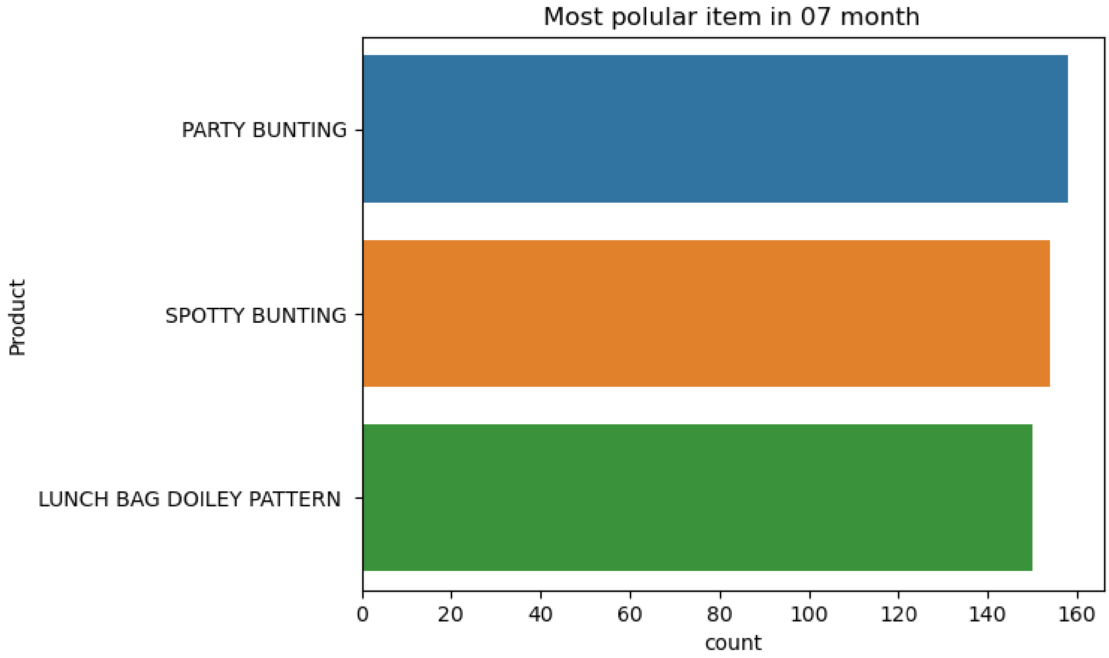
```
<Figure size 300x300 with 0 Axes>
```

```
0    The most popular item is PARTY BUNTING
Name: Product, dtype: object
```



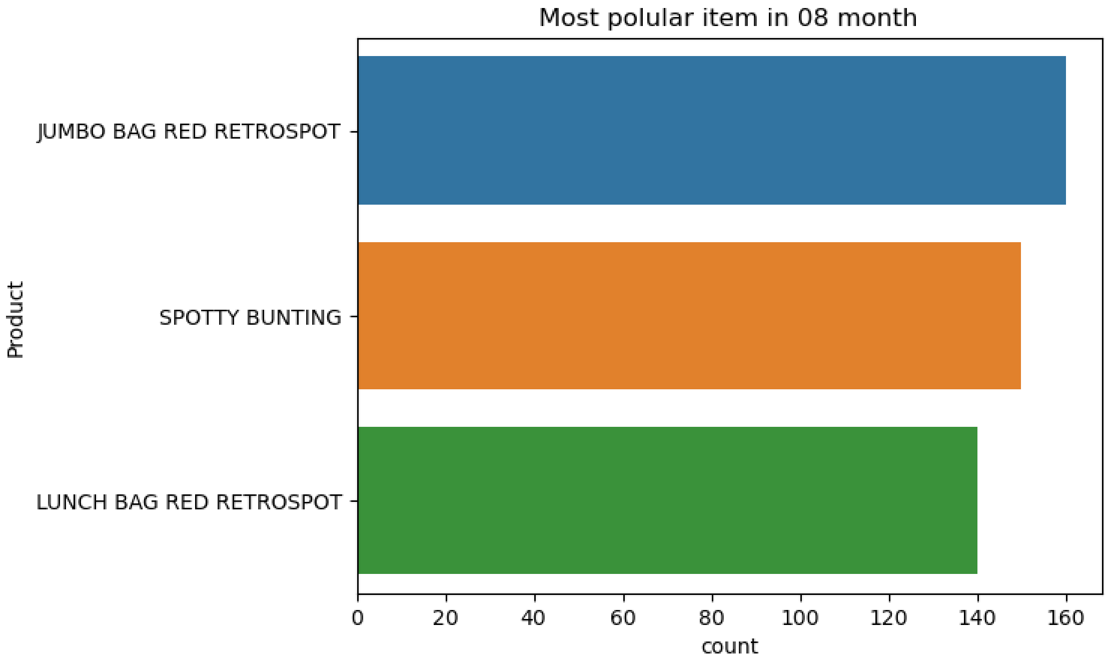
```
<Figure size 300x300 with 0 Axes>
```

```
0    The most popular item is PARTY BUNTING
Name: Product, dtype: object
```



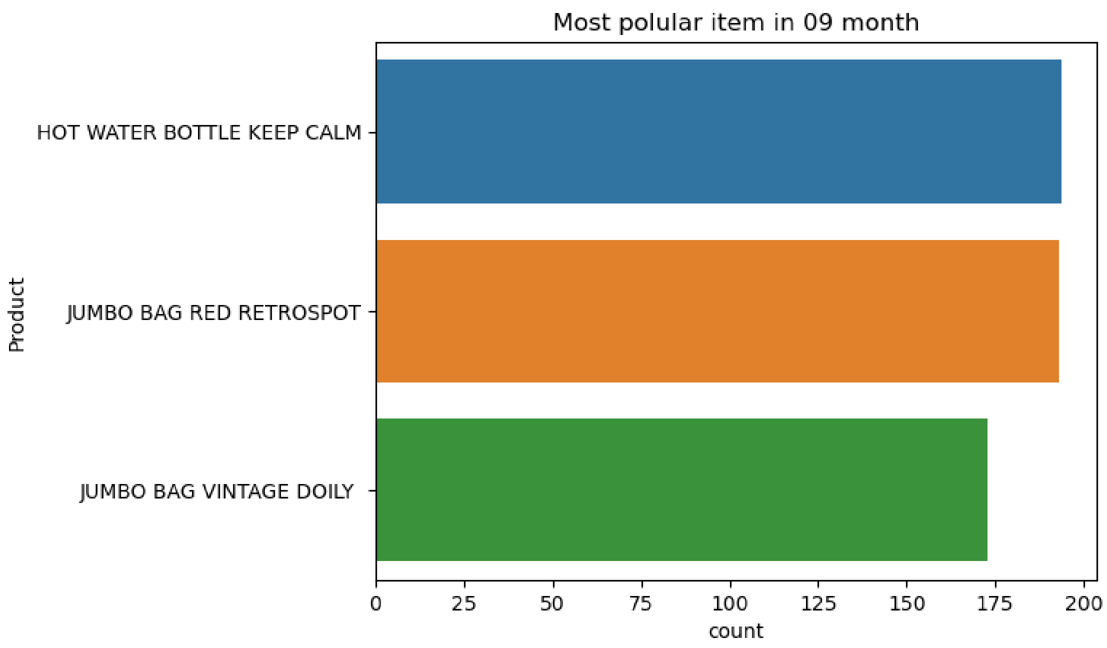
```
<Figure size 300x300 with 0 Axes>
```

```
0    The most popular item is JUMBO BAG RED RETROSPOT
Name: Product, dtype: object
```



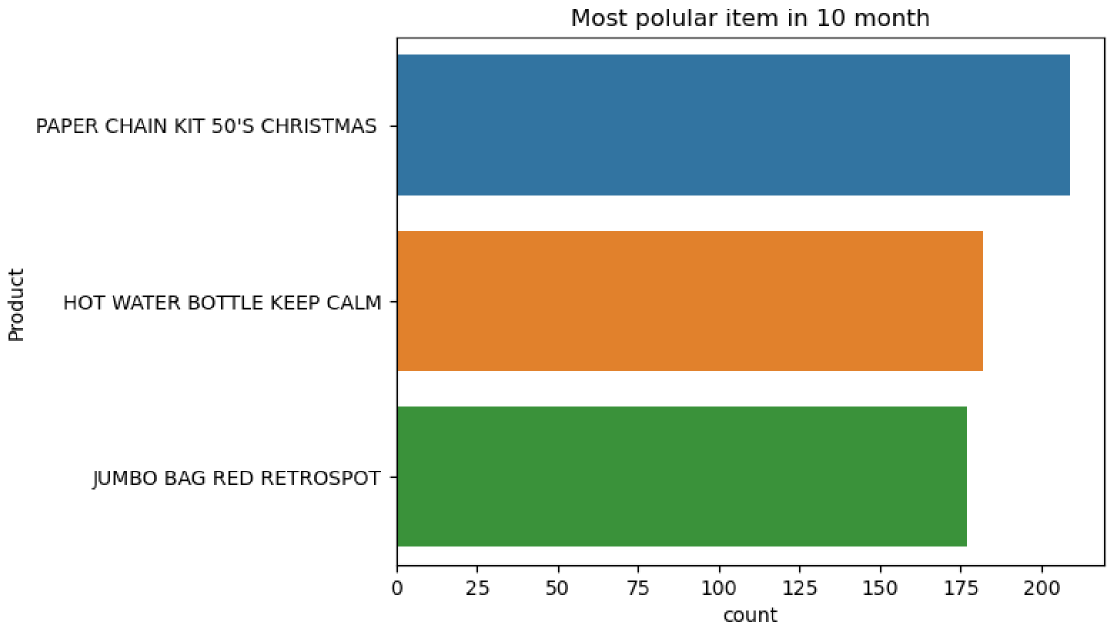
```
<Figure size 300x300 with 0 Axes>
```

```
0    The most popular item is HOT WATER BOTTLE KEEP…
Name: Product, dtype: object
```



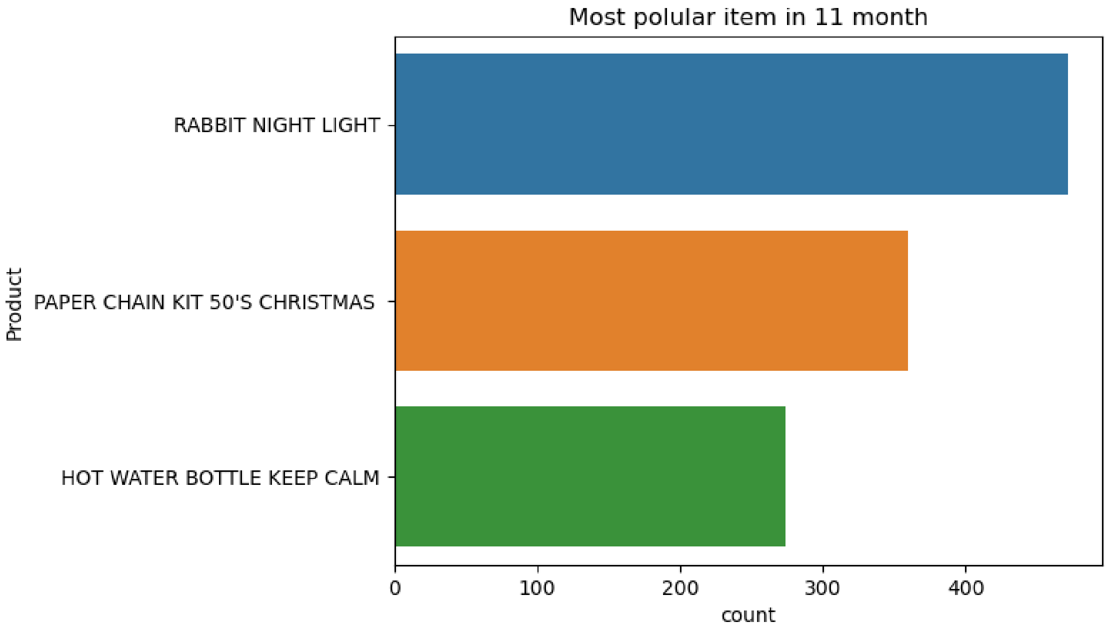
```
<Figure size 300x300 with 0 Axes>
```

```
0    The most popular item is PAPER CHAIN KIT 50'S …
Name: Product, dtype: object
```



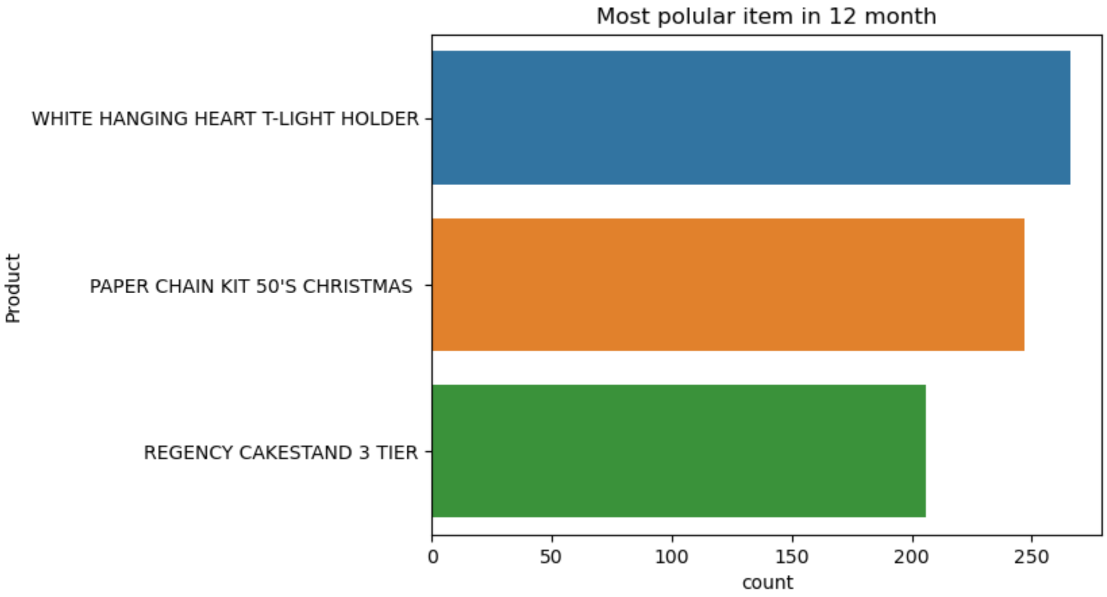
```
<Figure size 300x300 with 0 Axes>
```

```
0    The most popular item is RABBIT NIGHT LIGHT
Name: Product, dtype: object
```



```
<Figure size 300x300 with 0 Axes>
```

```
0    The most popular item is WHITE HANGING HEART T…
Name: Product, dtype: object
```



```
<Figure size 300x300 with 0 Axes>
```

[18]:
```
#Dataframe without duplicates in product description
d=data.pivot_table(index=['Description'],aggfunc='size').
 ↪rename_axis('Description').reset_index(name='count')
d
```

[18]:
```
                           Description  count
0        4 PURPLE FLOCK DINNER CANDLES     39
1        50'S CHRISTMAS GIFT BAG LARGE    110
2                    DOLLY GIRL BEAKER    140
3          I LOVE LONDON MINI BACKPACK     70
4          I LOVE LONDON MINI RUCKSACK      1
…                                  …    …
3891   ZINC T-LIGHT HOLDER STARS SMALL    241
3892    ZINC TOP  2 DOOR WOODEN SHELF      11
3893  ZINC WILLIE WINKIE  CANDLE STICK    193
3894       ZINC WIRE KITCHEN ORGANISER     12
3895  ZINC WIRE SWEETHEART LETTER TRAY     20

[3896 rows x 2 columns]
```

[19]:
```
#Cleaning the dataset
def clean_text(name):
  res = str(name).lower()
  return(res)

d["Description"] = d["Description"].apply(clean_text)
d = d.assign(index=range(len(d)))
d
```

[19]:
```
                           Description  count  index
0        4 purple flock dinner candles     39      0
1        50's christmas gift bag large    110      1
2                    dolly girl beaker    140      2
3          i love london mini backpack     70      3
4          i love london mini rucksack      1      4
…                                  …    …    …
3891   zinc t-light holder stars small    241   3891
3892    zinc top  2 door wooden shelf      11   3892
3893  zinc willie winkie  candle stick    193   3893
3894       zinc wire kitchen organiser     12   3894
3895  zinc wire sweetheart letter tray     20   3895

[3896 rows x 3 columns]
```

[20]:

```
#Vectorizing the dataset
from sklearn.feature_extraction.text import TfidfVectorizer
vectorizer = TfidfVectorizer()
vectorized = vectorizer.fit_transform(d['Description'])
print(vectorized)
```

```
  (0, 314)      0.47384875131636056
  (0, 565)      0.530925823261383
  (0, 717)      0.5127056737234867
  (0, 1481)     0.4803311599322848
  (1, 1031)     0.41475442632031
  (1, 114)      0.3735821621525685
  (1, 813)      0.47287999311389567
  (1, 400)      0.3832940529344711
  (1, 35)       0.5638131916395636
  (2, 157)      0.6650539016186924
  (2, 818)      0.5257854051348654
  (2, 587)      0.5303329291010697
  (3, 111)      0.6179607049312806
  (3, 1158)     0.4290250915573877
  (3, 1082)     0.49999950878894367
  (3, 1087)     0.4290250915573877
  (4, 1571)     0.6271467796346737
  (4, 1158)     0.42503726172097234
  (4, 1082)     0.49535196486067706
  (4, 1087)     0.42503726172097234
  (5, 1866)     0.46956151674438495
  (5, 1267)     0.4933956815647955
  (5, 613)      0.4088236372793164
  (5, 1245)     0.6074009524256885
  (6, 558)      0.4961063209352801
  :     :
  (3890, 1031)  0.39796907210704274
  (3891, 1759)  0.5737035435727462
  (3891, 2038)  0.46013939022694283
  (3891, 915)   0.3899568275959636
  (3891, 1693)  0.390699714119591
  (3891, 1059)  0.3929674180537274
  (3892, 592)   0.44545196437245044
  (3892, 1896)  0.4927095293705211
  (3892, 2038)  0.3861973195713076
  (3892, 1646)  0.5059530545609033
  (3892, 2020)  0.39201148457747226
  (3893, 2009)  0.5150325122320809
  (3893, 2004)  0.5150325122320809
  (3893, 1765)  0.49240489724610237
  (3893, 2038)  0.3575470665217882
  (3893, 311)   0.3149294128492006
  (3894, 1287)  0.574389409371327
  (3894, 1004)  0.48514394618336826
  (3894, 2010)  0.49816933668950625
  (3894, 2038)  0.4319021531157542
  (3895, 1051)  0.4515105947042097
  (3895, 1919)  0.4467929816448368
  (3895, 2010)  0.47720894049261386
  (3895, 2038)  0.41372993820634996
  (3895, 1823)  0.444537264769169
```

[21]:
```
#Building similarity matrix
from sklearn.metrics.pairwise import cosine_similarity
similarities = cosine_similarity(vectorized)
print(similarities)
```

```
[[1.         0.         0.         … 0.         0.         0.        ]
 [0.         1.         0.         … 0.         0.         0.        ]
 [0.         0.         1.         … 0.         0.         0.        ]
 …
 [0.         0.         0.         … 1.         0.15442535 0.14792793]
 [0.         0.         0.         … 0.15442535 1.         0.41642171]
 [0.         0.         0.         … 0.14792793 0.41642171 1.        ]]
```

[22]:
```
#Building recommendation system
import difflib
def recommend(product):
  prod_list = d['Description'].tolist()
  match_close = difflib.get_close_matches(product,prod_list)
  prod_ind = d[d.Description == match_close[0]]['index'].values[0]
  similarity_score = list(enumerate(similarities[prod_ind]))
  sorted_similar_movies = sorted(similarity_score, key = lambda x:x[1], reverse␣
 ↪= True)
  print('products suggested for you : \n')
  i = 1
  for prod in sorted_similar_movies:
    index = prod[0]
    title_from_index = d[d.index==index]['Description'].values[0]
    if (i<11):
      print(i, '.',title_from_index,' ')
      i+=1

recommend("dolly girl beaker")
```

```
products suggested for you :

1 .  dolly girl beaker
2 . card dolly girl
```

```
3 . wrap dolly girl
4 . dolly girl wall art
5 . wall art dolly girl
6 . dolly girl lunch box
7 . childrens dolly girl mug
8 . dolly girl childrens bowl
9 . spaceboy beaker
10 . dolly girl childrens cup
```