In [1]:
```
import pandas as pd
import numpy as np
from mlxtend.frequent_patterns import apriori, association_rules, fpgrowth
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.express as px
```

In [3]:
```
# Load sample data
df = pd.read_excel(r"C:\Users\mukki\OneDrive\Desktop\Online Retail.xlsx", sheet_nam
df.dropna(subset=['InvoiceNo', 'StockCode', 'Description', 'Quantity', 'UnitPrice']
df = df[df['Quantity'] > 0]
df = df[df['UnitPrice'] > 0]
df.head()
```

Out[3]:

	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Cou
0	536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	2010-12-01 08:26:00	2.55	17850.0	Un Kingc
1	536365	71053	WHITE METAL LANTERN	6	2010-12-01 08:26:00	3.39	17850.0	Un Kingc
2	536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	2010-12-01 08:26:00	2.75	17850.0	Un Kingc
3	536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	2010-12-01 08:26:00	3.39	17850.0	Un Kingc
4	536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	6	2010-12-01 08:26:00	3.39	17850.0	Un Kingc

In [5]:
```
# Create basket: UK only, remove credit notes (invoices starting with 'C')
basket = df[~df['InvoiceNo'].astype(str).str.startswith('C')]
basket = basket[basket['Country'] == "United Kingdom"]
basket_pivot = basket.groupby(['InvoiceNo', 'Description'])['Quantity'].sum().unsta
basket_pivot = basket_pivot.applymap(lambda x: 1 if x > 0 else 0)
basket_pivot.head()
```

```
C:\Users\mukki\AppData\Local\Temp\ipykernel_13524\534717784.py:5: FutureWarning: Dat
aFrame.applymap has been deprecated. Use DataFrame.map instead.
  basket_pivot = basket_pivot.applymap(lambda x: 1 if x > 0 else 0)
```

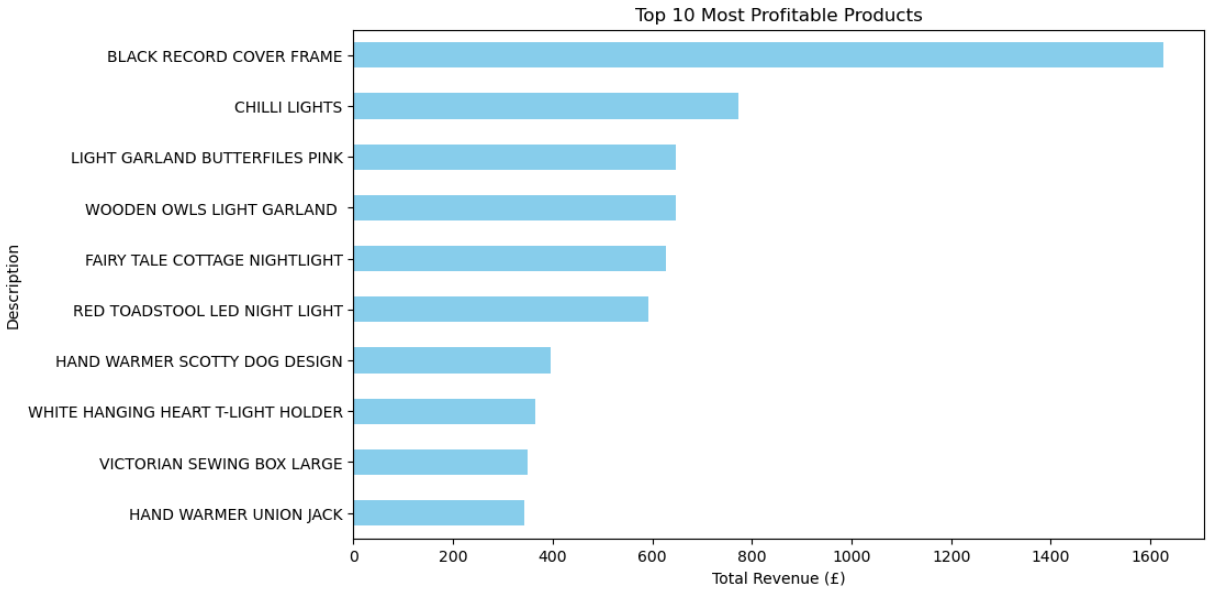
Out[5]:

Description	SET 2 TEA TOWELS I LOVE LONDON	10 COLOUR SPACEBOY PEN	12 DAISY PEGS IN WOOD BOX	12 MESSAGE CARDS WITH ENVELOPES	12 PENCILS TALL TUBE SKULLS	3 PIECE SPACEBOY COOKIE CUTTER SET	3 STRIPEY MICE FELTCRAFT
InvoiceNo							
536365	0	0	0	0	0	0	0
536366	0	0	0	0	0	0	0
536367	0	0	0	0	0	0	0
536368	0	0	0	0	0	0	0
536369	0	0	0	0	0	0	0

5 rows × 571 columns

In [ ]:
```
basket_bool=basket_pivot.astype(bool)
frequent_itemsets_ap = apriori(basket_bool, min_support=0.02, use_colnames=True)
rules_ap = association_rules(frequent_itemsets_ap, metric="lift", min_threshold=1)
rules_ap.sort_values("confidence", ascending=False).head()
```

In [7]:
```
df['TotalPrice'] = df['Quantity'] * df['UnitPrice']
product_profit = df.groupby('Description')['TotalPrice'].sum().sort_values(ascendin
product_profit.plot(kind='barh', title='Top 10 Most Profitable Products', figsize=(
plt.xlabel('Total Revenue (£)')
plt.gca().invert_yaxis()
plt.show()
```



In [ ]: