

NETFLIX USERBASE DATASET

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
```

```
df = pd.read_csv(r"C:\Users\shagu\Downloads\archive.zip")
df
```

	User ID	Subscription Type	Monthly Revenue	Join Date	Last Payment Date
0	1	Basic	10	15-01-22	10-06-23
1	2	Premium	15	05-09-21	22-06-23
2	3	Standard	12	28-02-23	27-06-23
3	4	Standard	12	10-07-22	26-06-23
4	5	Basic	10	01-05-23	28-06-23
...
2495	2496	Premium	14	25-07-22	12-07-23
2496	2497	Basic	15	04-08-22	14-07-23
2497	2498	Standard	12	09-08-22	15-07-23
2498	2499	Standard	13	12-08-22	12-07-23
2499	2500	Basic	15	13-08-22	12-07-23

	Country	Age	Gender	Device	Plan	Duration
0	United States	28	Male	Smartphone		1 Month
1	Canada	35	Female	Tablet		1 Month
2	United Kingdom	42	Male	Smart TV		1 Month
3	Australia	51	Female	Laptop		1 Month
4	Germany	33	Male	Smartphone		1 Month
...
2495	Spain	28	Female	Smart TV		1 Month
2496	Spain	33	Female	Smart TV		1 Month
2497	United States	38	Male	Laptop		1 Month
2498	Canada	48	Female	Tablet		1 Month
2499	United States	35	Female	Smart TV		1 Month

```
[2500 rows x 10 columns]
```

```
df.head()
```

	User ID	Subscription Type	Monthly Revenue	Join Date	Last Payment Date \
0	1	Basic	10	15-01-22	10-06-23
1	2	Premium	15	05-09-21	22-06-23
2	3	Standard	12	28-02-23	27-06-23
3	4	Standard	12	10-07-22	26-06-23
4	5	Basic	10	01-05-23	28-06-23

	Country	Age	Gender	Device	Plan	Duration
0	United States	28	Male	Smartphone		1 Month
1	Canada	35	Female	Tablet		1 Month
2	United Kingdom	42	Male	Smart TV		1 Month
3	Australia	51	Female	Laptop		1 Month
4	Germany	33	Male	Smartphone		1 Month

```
df.tail()
```

	User ID	Subscription Type	Monthly Revenue	Join Date	Last Payment Date \
2495	2496	Premium	14	25-07-22	12-07-23
2496	2497	Basic	15	04-08-22	14-07-23
2497	2498	Standard	12	09-08-22	15-07-23
2498	2499	Standard	13	12-08-22	12-07-23
2499	2500	Basic	15	13-08-22	12-07-23

	Country	Age	Gender	Device	Plan	Duration
2495	Spain	28	Female	Smart TV		1 Month
2496	Spain	33	Female	Smart TV		1 Month
2497	United States	38	Male	Laptop		1 Month
2498	Canada	48	Female	Tablet		1 Month
2499	United States	35	Female	Smart TV		1 Month

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>  
RangeIndex: 2500 entries, 0 to 2499
```

```
Data columns (total 10 columns):
#   Column              Non-Null Count  Dtype
---  -
0   User ID              2500 non-null   int64
1   Subscription Type     2500 non-null   object
2   Monthly Revenue      2500 non-null   int64
3   Join Date            2500 non-null   object
4   Last Payment Date    2500 non-null   object
5   Country              2500 non-null   object
6   Age                  2500 non-null   int64
7   Gender               2500 non-null   object
8   Device               2500 non-null   object
9   Plan Duration        2500 non-null   object
dtypes: int64(3), object(7)
memory usage: 195.4+ KB
```

check stats information of data set

```
df.describe()

      User ID  Monthly Revenue      Age
count  2500.000000      2500.000000  2500.000000
mean   1250.500000      12.508400    38.795600
std     721.83216      1.686851     7.171778
min      1.000000     10.000000    26.000000
25%     625.750000     11.000000    32.000000
50%    1250.500000     12.000000    39.000000
75%    1875.250000     14.000000    45.000000
max    2500.000000     15.000000    51.000000

df.shape
(2500, 10)
```

MISSING VALUE

```
df.isnull().sum()

User ID              0
Subscription Type    0
Monthly Revenue      0
Join Date            0
Last Payment Date    0
Country              0
Age                  0
Gender               0
Device               0
```

```
Plan Duration      0
dtype: int64
```

```
df.rename(columns={"Subscription Type":"Subscription"}, inplace=True)
df
```

Date \	User ID	Subscription	Monthly Revenue	Join Date	Last Payment
0	1	Basic	10	15-01-22	10-06-23
1	2	Premium	15	05-09-21	22-06-23
2	3	Standard	12	28-02-23	27-06-23
3	4	Standard	12	10-07-22	26-06-23
4	5	Basic	10	01-05-23	28-06-23
...
2495	2496	Premium	14	25-07-22	12-07-23
2496	2497	Basic	15	04-08-22	14-07-23
2497	2498	Standard	12	09-08-22	15-07-23
2498	2499	Standard	13	12-08-22	12-07-23
2499	2500	Basic	15	13-08-22	12-07-23

	Country	Age	Gender	Device	Plan	Duration
0	United States	28	Male	Smartphone		1 Month
1	Canada	35	Female	Tablet		1 Month
2	United Kingdom	42	Male	Smart TV		1 Month
3	Australia	51	Female	Laptop		1 Month
4	Germany	33	Male	Smartphone		1 Month
...
2495	Spain	28	Female	Smart TV		1 Month
2496	Spain	33	Female	Smart TV		1 Month
2497	United States	38	Male	Laptop		1 Month
2498	Canada	48	Female	Tablet		1 Month
2499	United States	35	Female	Smart TV		1 Month

```
[2500 rows x 10 columns]
```

top 5 countries which have more Monthly Revenue

```
df1= df.nlargest(5,"Monthly Revenue")
df1
```

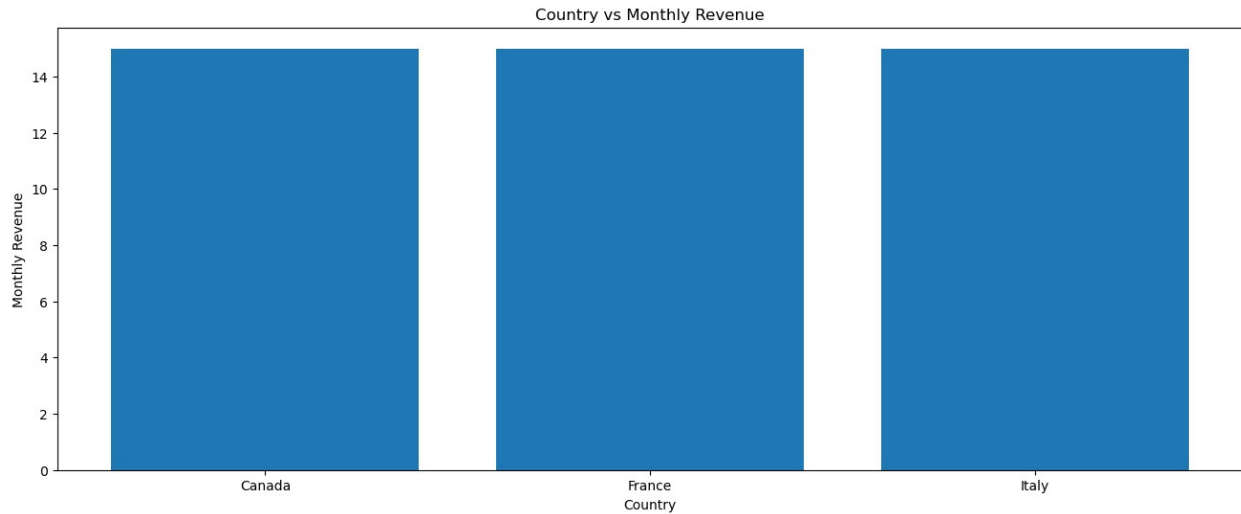
	User ID	Subscription	Monthly Revenue	Join Date	Last Payment Date
Country \					
1	2	Premium	15	05-09-21	22-06-23
Canada					
5	6	Premium	15	18-03-22	27-06-23
France					
9	10	Premium	15	07-01-23	22-06-23
Italy					
11	12	Premium	15	23-03-23	28-06-23
Canada					
15	16	Premium	15	07-04-22	27-06-23
France					

	Age	Gender	Device	Plan	Duration
1	35	Female	Tablet		1 Month
5	29	Female	Smart TV		1 Month
9	44	Female	Smart TV		1 Month
11	45	Male	Tablet		1 Month
15	36	Male	Tablet		1 Month

Plot a Graph

```
plt.figure(figsize=(16,6))
plt.bar(x=df1["Country"], height=df1["Monthly Revenue"])

plt.xlabel("Country")
plt.ylabel("Monthly Revenue")
plt.title("Country vs Monthly Revenue")
plt.show()
```



```
df2 = df.drop_duplicates(subset="Country")
```

```
print(df2)
```

	User ID	Subscription	Monthly Revenue	Join Date	Last Payment Date
0	1	Basic	10	15-01-22	10-06-23
1	2	Premium	15	05-09-21	22-06-23
2	3	Standard	12	28-02-23	27-06-23
3	4	Standard	12	10-07-22	26-06-23
4	5	Basic	10	01-05-23	28-06-23
5	6	Premium	15	18-03-22	27-06-23
6	7	Standard	12	09-12-21	25-06-23
7	8	Basic	10	02-04-23	24-06-23
8	9	Standard	12	20-10-22	23-06-23
9	10	Premium	15	07-01-23	22-06-23

	Country	Age	Gender	Device	Plan	Duration
0	United States	28	Male	Smartphone		1 Month
1	Canada	35	Female	Tablet		1 Month
2	United Kingdom	42	Male	Smart TV		1 Month
3	Australia	51	Female	Laptop		1 Month
4	Germany	33	Male	Smartphone		1 Month
5	France	29	Female	Smart TV		1 Month
6	Brazil	46	Male	Tablet		1 Month

7	Mexico	39	Female	Laptop	1 Month
8	Spain	37	Male	Smartphone	1 Month
9	Italy	44	Female	Smart TV	1 Month

df2

	User ID	Subscription	Monthly Revenue	Join Date	Last Payment
Date \					
0	1	Basic	10	15-01-22	10-06-23
1	2	Premium	15	05-09-21	22-06-23
2	3	Standard	12	28-02-23	27-06-23
3	4	Standard	12	10-07-22	26-06-23
4	5	Basic	10	01-05-23	28-06-23
5	6	Premium	15	18-03-22	27-06-23
6	7	Standard	12	09-12-21	25-06-23
7	8	Basic	10	02-04-23	24-06-23
8	9	Standard	12	20-10-22	23-06-23
9	10	Premium	15	07-01-23	22-06-23

	Country	Age	Gender	Device	Plan	Duration
0	United States	28	Male	Smartphone		1 Month
1	Canada	35	Female	Tablet		1 Month
2	United Kingdom	42	Male	Smart TV		1 Month
3	Australia	51	Female	Laptop		1 Month
4	Germany	33	Male	Smartphone		1 Month
5	France	29	Female	Smart TV		1 Month
6	Brazil	46	Male	Tablet		1 Month
7	Mexico	39	Female	Laptop		1 Month
8	Spain	37	Male	Smartphone		1 Month
9	Italy	44	Female	Smart TV		1 Month

df2=df.nsmallest(5,"Monthly Revenue")
df2

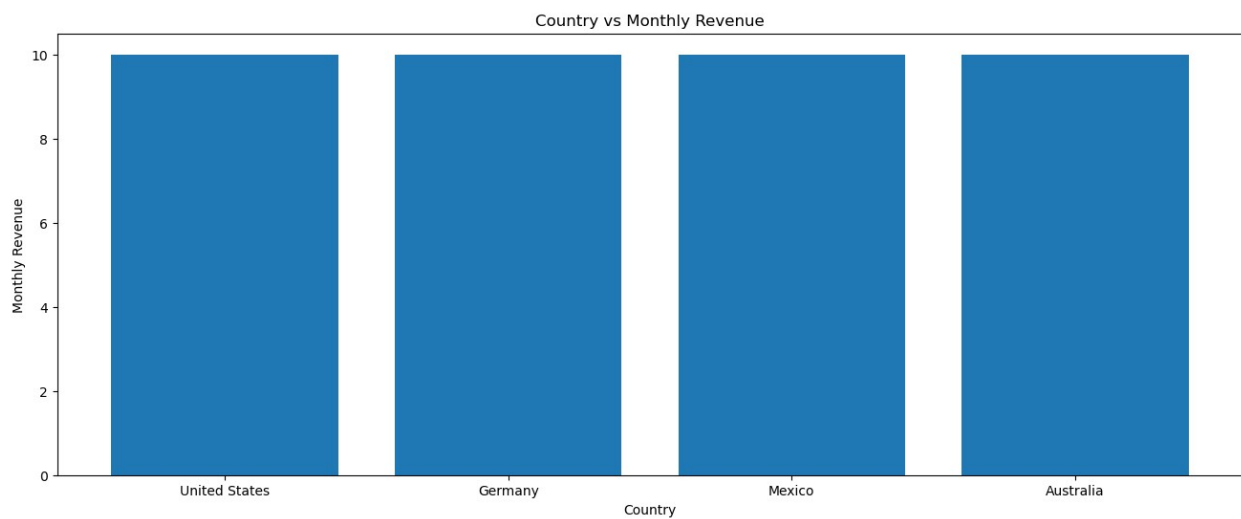
	User ID	Subscription	Monthly Revenue	Join Date	Last Payment	Date
\						
0	1	Basic	10	15-01-22		10-06-23
4	5	Basic	10	01-05-23		28-06-23
7	8	Basic	10	02-04-23		24-06-23

10	11	Basic	10	16-05-22	22-06-23
13	14	Basic	10	01-08-22	26-06-23

	Country	Age	Gender	Device	Plan	Duration
0	United States	28	Male	Smartphone		1 Month
4	Germany	33	Male	Smartphone		1 Month
7	Mexico	39	Female	Laptop		1 Month
10	United States	31	Female	Smartphone		1 Month
13	Australia	27	Male	Smartphone		1 Month

```
plt.figure(figsize=(16,6))
plt.bar(x=df2["Country"], height=df2["Monthly Revenue"])

plt.xlabel("Country")
plt.ylabel("Monthly Revenue")
plt.title("Country vs Monthly Revenue")
plt.show()
```



```
res = df.groupby("Country")["Monthly Revenue"].sum().reset_index
print(res)
```

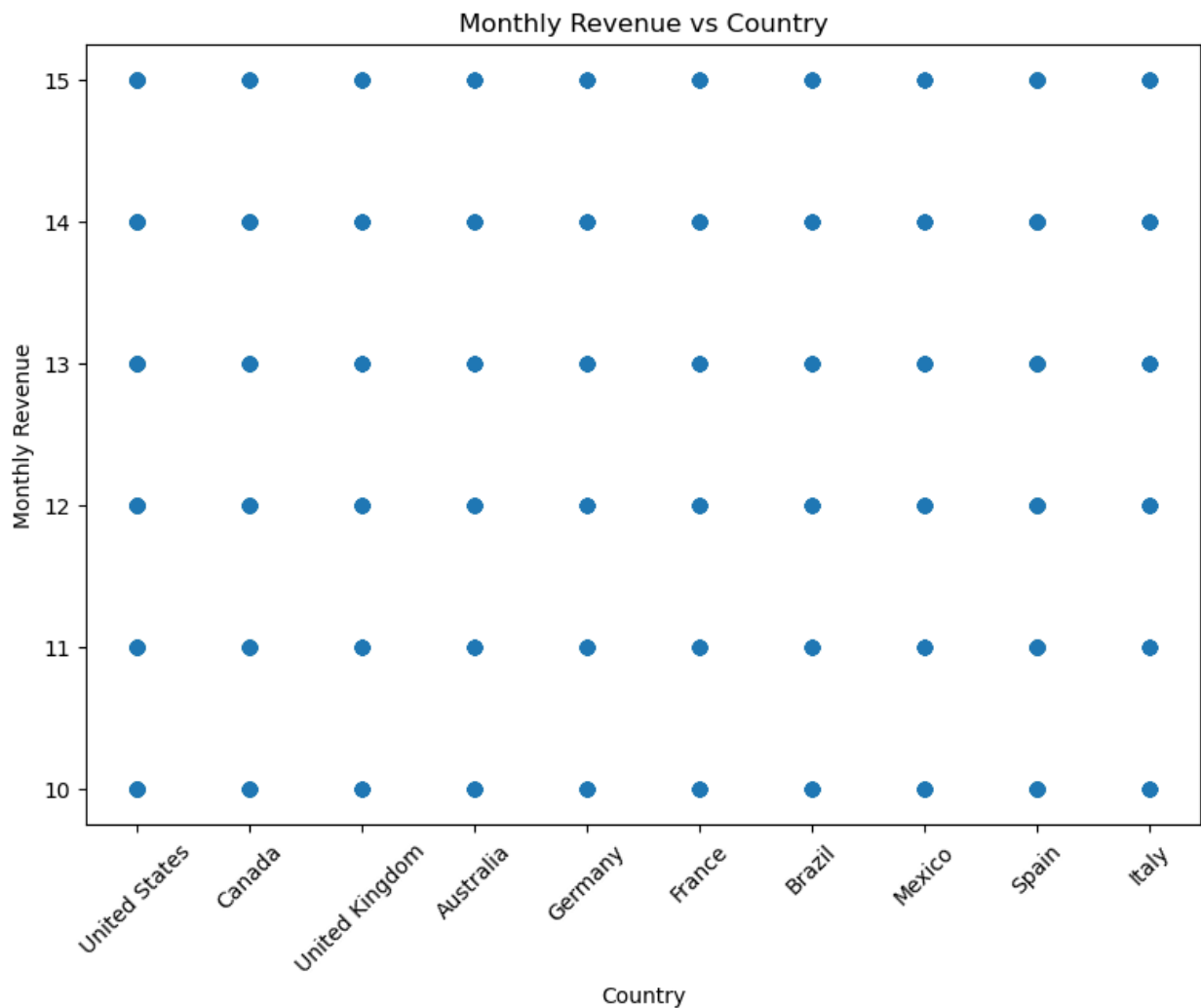
```
<bound method Series.reset_index of Country>
Australia      2271
Brazil         2285
Canada         3950
France         2307
Germany        2260
Italy          2317
Mexico         2237
Spain          5662
United Kingdom 2318
```



```
United States      5664  
Name: Monthly Revenue, dtype: int64>
```

Scatter Graph

```
plt.figure(figsize=(8, 6))  
plt.scatter(df['Country'], df['Monthly Revenue'], alpha =0.5)  
plt.xlabel('Country')  
plt.ylabel('Monthly Revenue')  
plt.title('Monthly Revenue vs Country')  
plt.tight_layout()  
plt.xticks(rotation=45)  
plt.show()
```

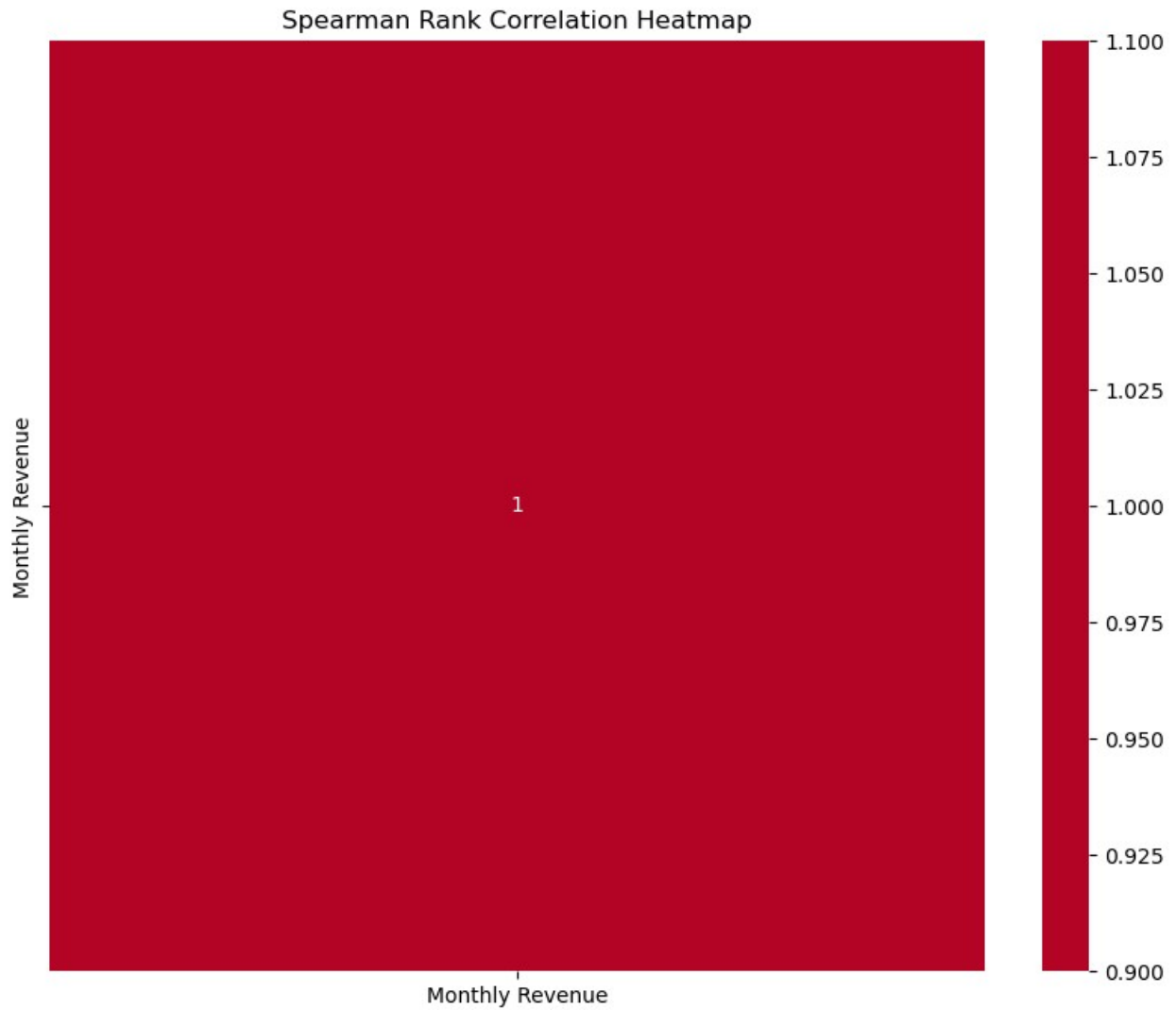


```
correlation_matrix = df[['Monthly Revenue',  
                          'Country']].corr(method='spearman')
```

C:\Users\shagu\AppData\Local\Temp\ipykernel_11348\2775018755.py:1:
FutureWarning: The default value of numeric_only in DataFrame.corr is
deprecated. In a future version, it will default to False. Select only
valid columns or specify the value of numeric_only to silence this
warning.

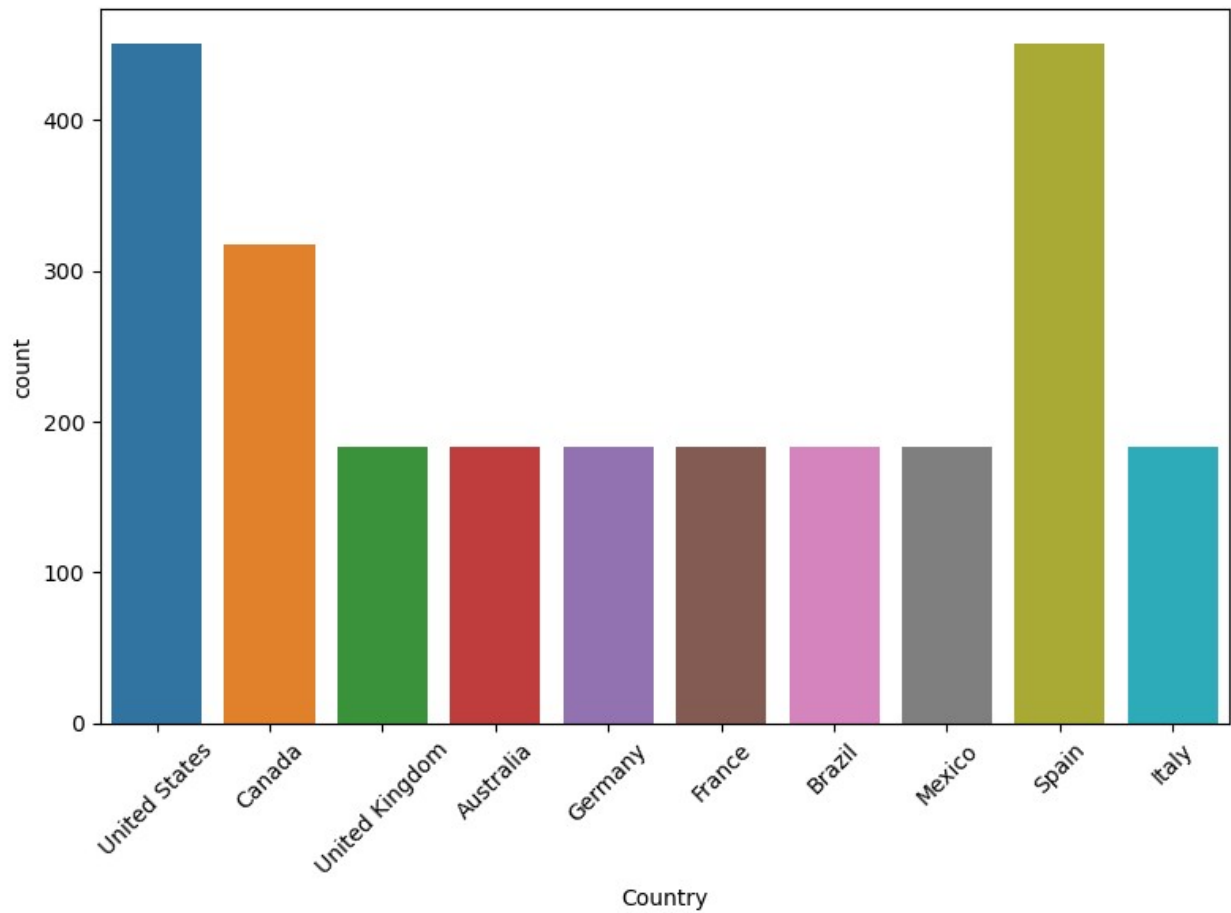
```
correlation_matrix = df[['Monthly Revenue',  
                          'Country']].corr(method='spearman')
```

```
plt.figure(figsize=(10, 8))  
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', center=0)  
plt.title('Spearman Rank Correlation Heatmap')  
plt.show()
```

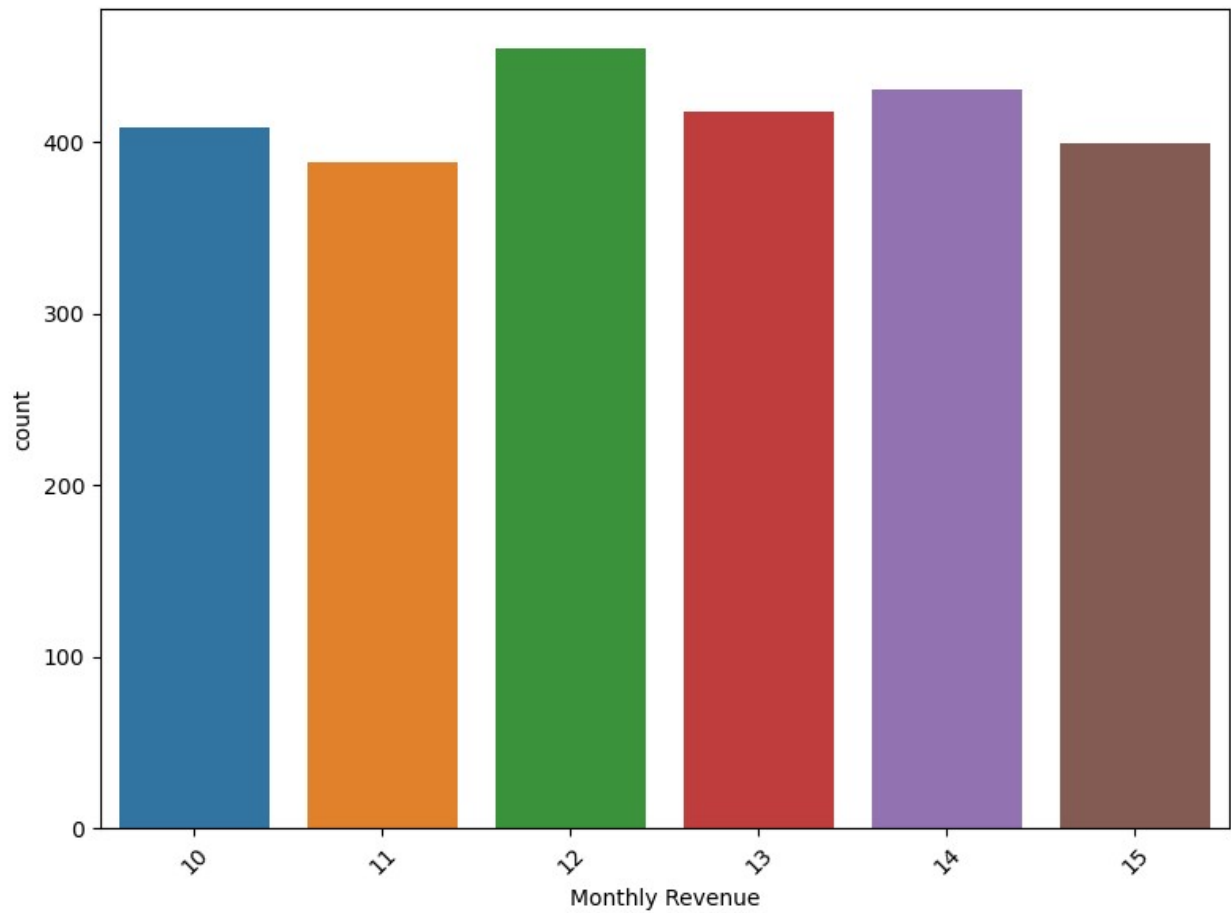


COUNT PLOT

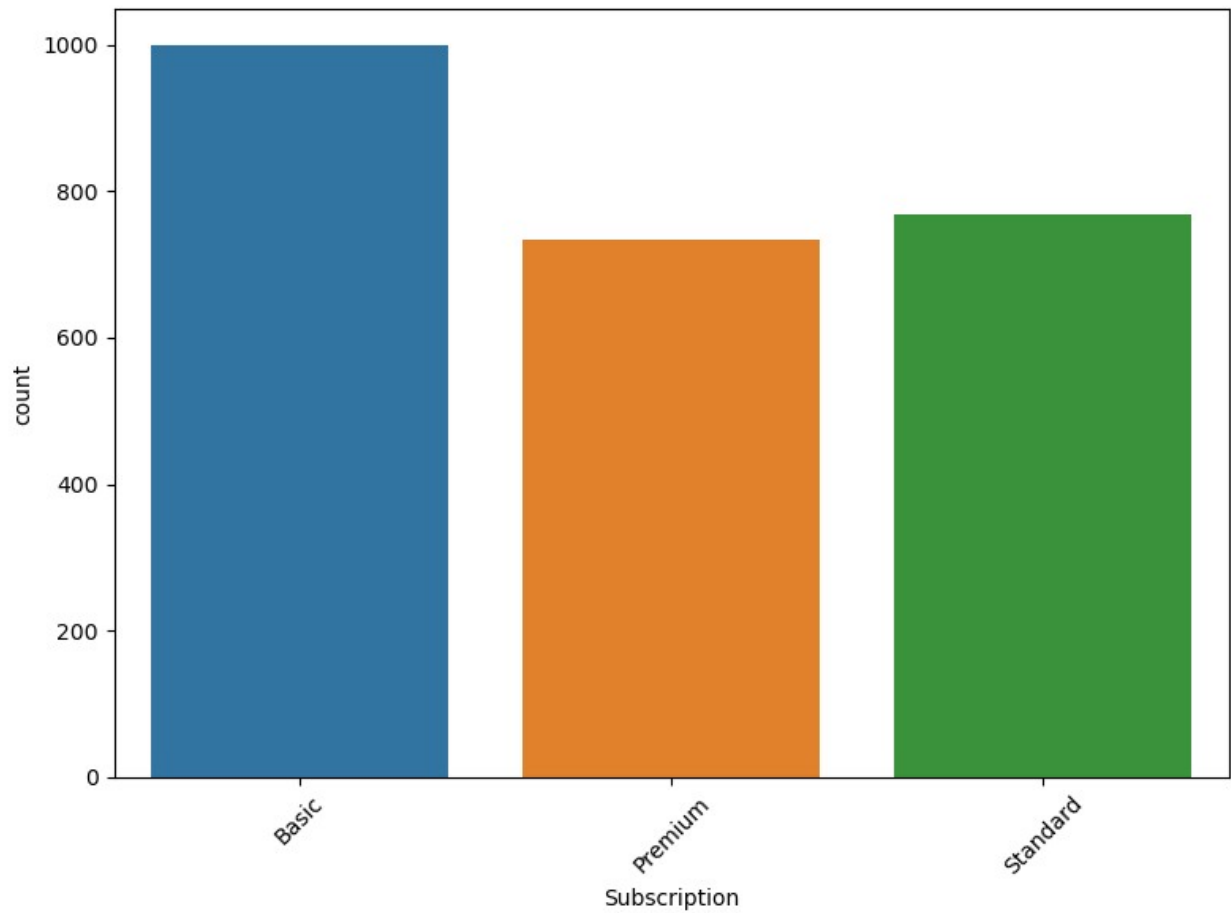
```
plt.figure(figsize=(8, 6))
sns.countplot(x=df["Country"])
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```



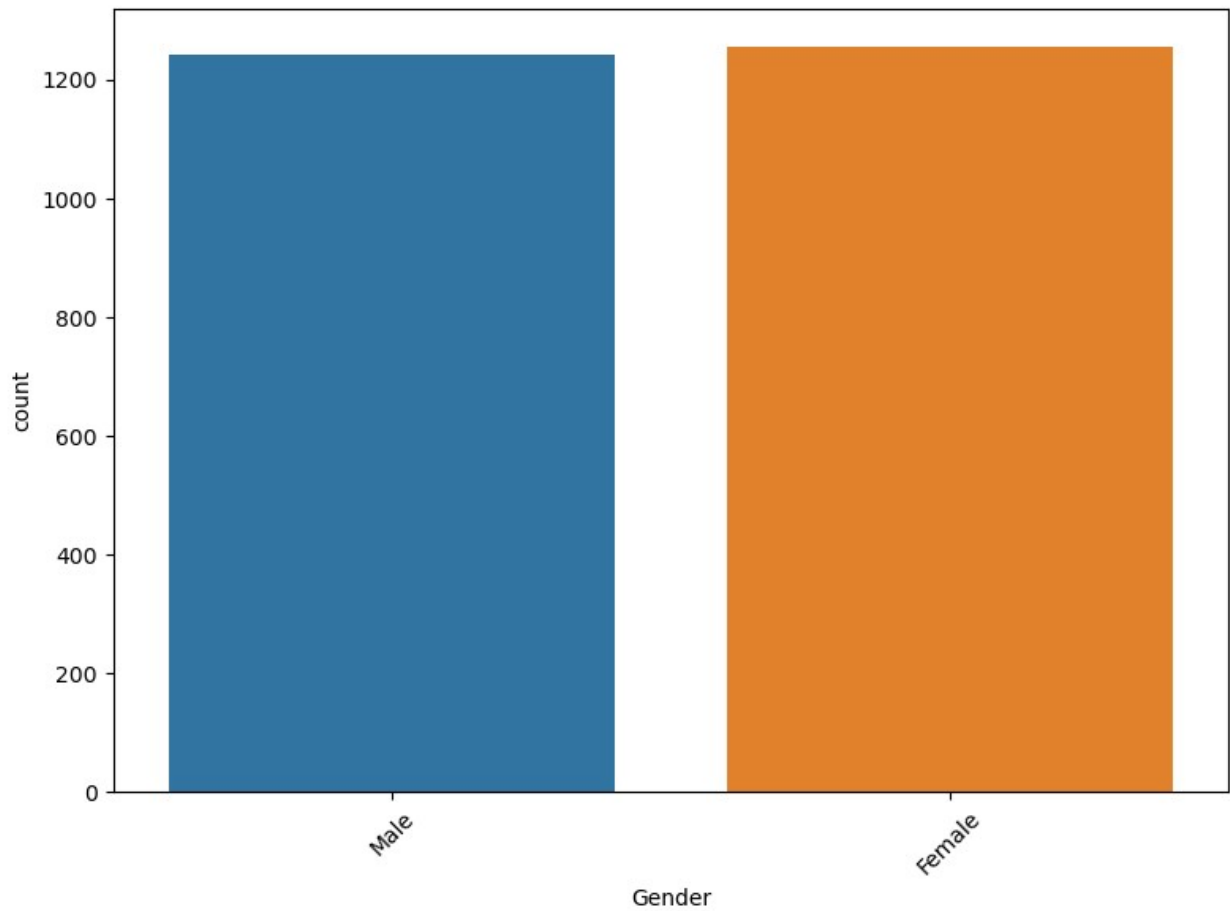
```
plt.figure(figsize=(8, 6))
sns.countplot(x=df["Monthly Revenue"])
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```



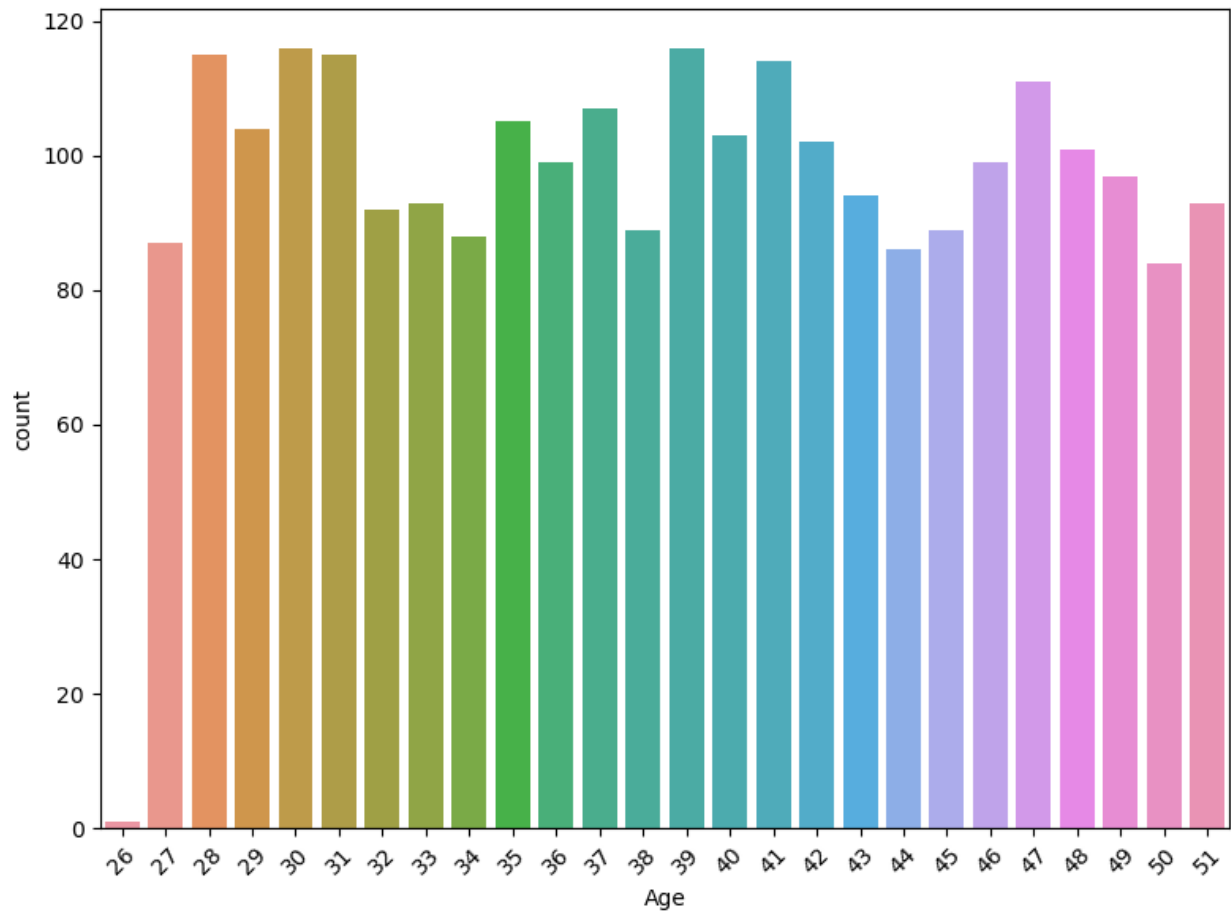
```
plt.figure(figsize=(8, 6))
sns.countplot(x=df["Subscription"])
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```



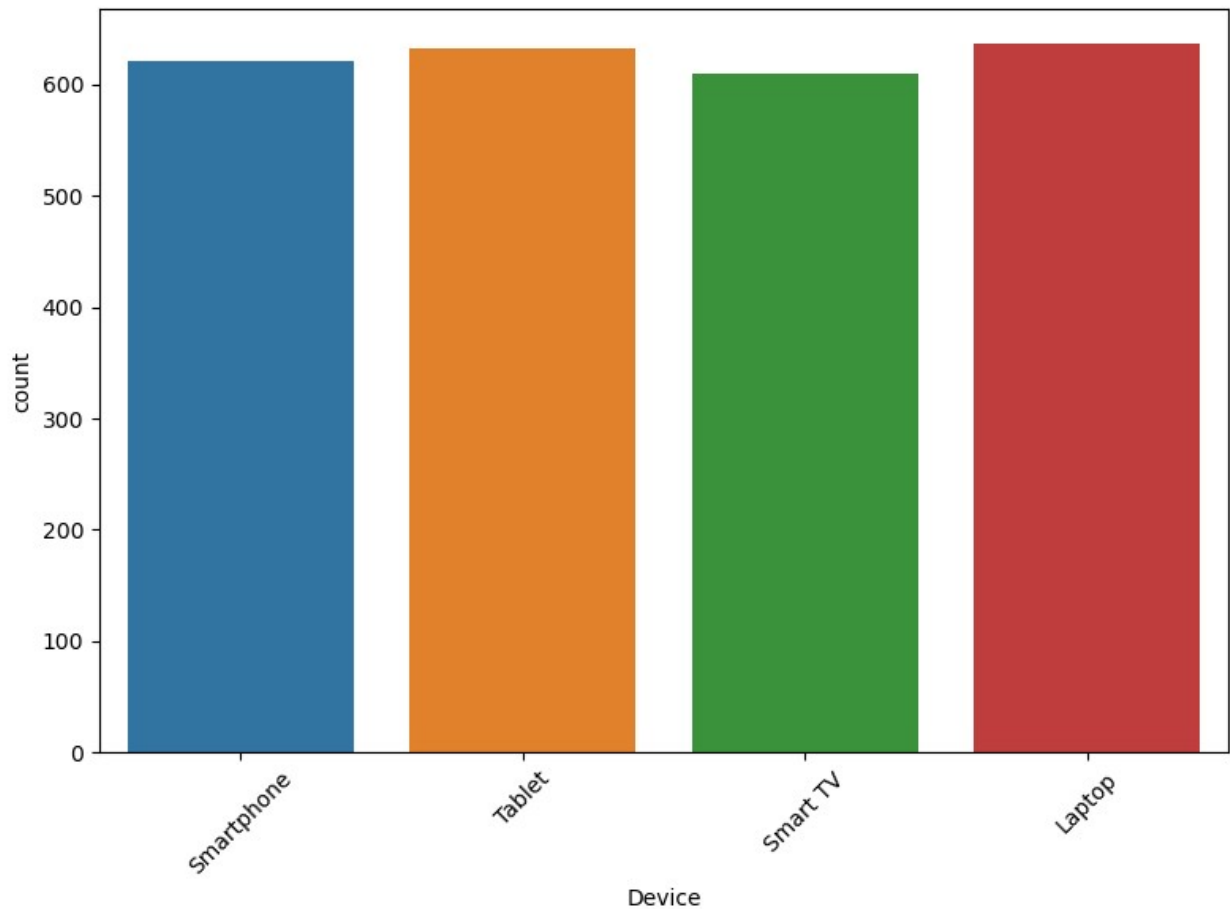
```
plt.figure(figsize=(8, 6))
sns.countplot(x=df["Gender"])
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```



```
plt.figure(figsize=(8, 6))  
sns.countplot(x=df["Age"])  
plt.xticks(rotation=45)  
plt.tight_layout()  
plt.show()
```



```
plt.figure(figsize=(8, 6))
sns.countplot(x=df["Device"])
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```

```
# COUNTPLOT :- It shows the count of occurrences of each category
within a dataset.
# In a countplot:

# The x-axis represents the categories or levels of the categorical
variable.
# The y-axis represents the count or frequency of occurrences for each
category.
```