# 1. Defining Problem Statement and Analysing basic metrics

**Problem Statement:**

The market research team at AeroFit wants to identify the characteristics of the target audience for each type of treadmill offered by the company, to provide a better recommendation of the treadmills to the new customers. The team decides to investigate whether there are differences across the product with respect to customer characteristics.

1. Perform descriptive analytics to create a customer profile for each AeroFit treadmill product by developing appropriate tables and charts.
2. For each AeroFit treadmill product, construct two-way contingency tables and compute all conditional and marginal probabilities along with their insights/impact on the business.

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

```
df=pd.read_csv("https://d2beiqkhq929f0.cloudfront.net/public_assets/assets/000/001/125/original/aerofit_treadmill.csv?1639992749")
df.head()
```

| | Product | Age | Gender | Education | MaritalStatus | Usage | Fitness | Income | Miles |
|---|---|---|---|---|---|---|---|---|---|
| **0** | KP281 | 18 | Male | 14 | Single | 3 | 4 | 29562 | 112 |
| **1** | KP281 | 19 | Male | 15 | Single | 2 | 3 | 31836 | 75 |
| **2** | KP281 | 19 | Female | 14 | Partnered | 4 | 3 | 30699 | 66 |
| **3** | KP281 | 19 | Male | 12 | Single | 3 | 3 | 32973 | 85 |
| **4** | KP281 | 20 | Male | 13 | Partnered | 4 | 2 | 35247 | 47 |

```
df.keys()
```

```
Index(['Product', 'Age', 'Gender', 'Education', 'MaritalStatus', 'Usage',
       'Fitness', 'Income', 'Miles'],
      dtype='object')
```

```
df.shape
```

```
(180, 9)
```

```
df.size
```

```
1620
```

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 180 entries, 0 to 179
Data columns (total 9 columns):
 #   Column         Non-Null Count  Dtype 
---  ------         --------------  ----- 
 0   Product        180 non-null    object
 1   Age            180 non-null    int64 
 2   Gender         180 non-null    object
 3   Education      180 non-null    int64 
 4   MaritalStatus  180 non-null    object
 5   Usage          180 non-null    int64 
 6   Fitness        180 non-null    int64 
 7   Income         180 non-null    int64 
 8   Miles          180 non-null    int64 
dtypes: int64(6), object(3)
memory usage: 12.8+ KB
```

```
df.describe()
```

| | Age | Education | Usage | Fitness | Income | Miles |
|---|---|---|---|---|---|---|
| **count** | 180.000000 | 180.000000 | 180.000000 | 180.000000 | 180.000000 | 180.000000 |
| **mean** | 28.788889 | 15.572222 | 3.455556 | 3.311111 | 53719.577778 | 103.194444 |
| **std** | 6.943498 | 1.617055 | 1.084797 | 0.958869 | 16506.684226 | 51.863605 |
| **min** | 18.000000 | 12.000000 | 2.000000 | 1.000000 | 29562.000000 | 21.000000 |
| **25%** | 24.000000 | 14.000000 | 3.000000 | 3.000000 | 44058.750000 | 66.000000 |
| **50%** | 26.000000 | 16.000000 | 3.000000 | 3.000000 | 50596.500000 | 94.000000 |
| **75%** | 33.000000 | 16.000000 | 4.000000 | 4.000000 | 58668.000000 | 114.750000 |

```
df.isna().sum()
```

```
Product          0
Age              0
Gender           0
Education        0
MaritalStatus    0
Usage            0
Fitness          0
Income           0
Miles            0
dtype: int64
```

## 2. Non-Graphical Analysis: Value counts and Unique attributes

```
df['Product'].unique()
```

```
array(['KP281', 'KP481', 'KP781'], dtype=object)
```

```
df['Product'].value_counts()
```

```
KP281    80
KP481    60
KP781    40
Name: Product, dtype: int64
```

```
df['Gender'].value_counts()
```

```
Male      104
Female     76
Name: Gender, dtype: int64
```

```
df['Education'].value_counts()
```

```
16    85
14    55
18    23
15     5
13     5
12     3
21     3
20     1
Name: Education, dtype: int64
```

```
df['MaritalStatus'].value_counts()
```

```
Partnered    107
Single        73
Name: MaritalStatus, dtype: int64
```

```
df['Fitness'].value_counts()
```

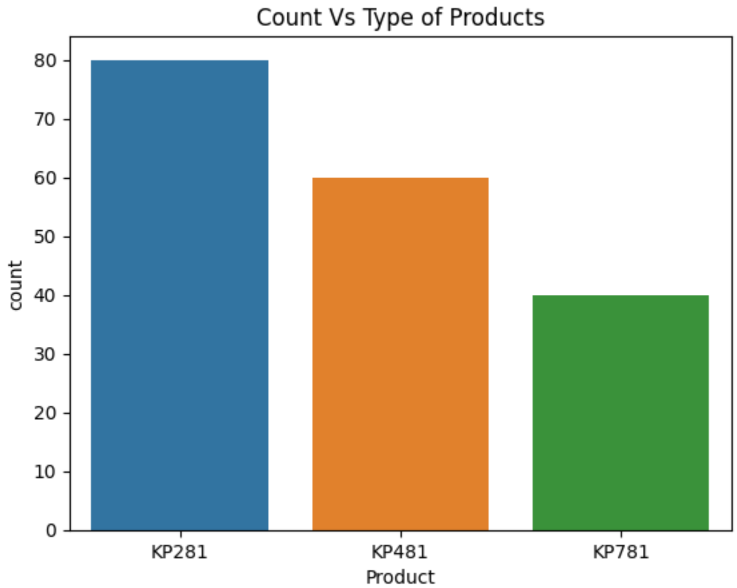
```
3    97
5    31
2    26
4    24
1     2
Name: Fitness, dtype: int64
```

## 3.Visual Analysis - Univariate, Bivariate after pre-processing of the data

#### 1.Analysis / Continuous Variables

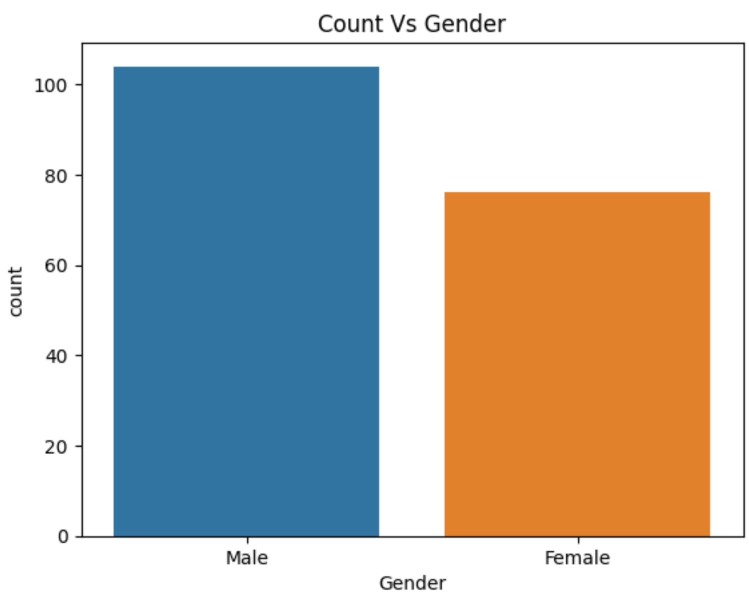
```
sns.countplot(x='Product',data=df)
plt.title('Count Vs Type of Products')
```

```
Text(0.5, 1.0, 'Count Vs Type of Products')
```



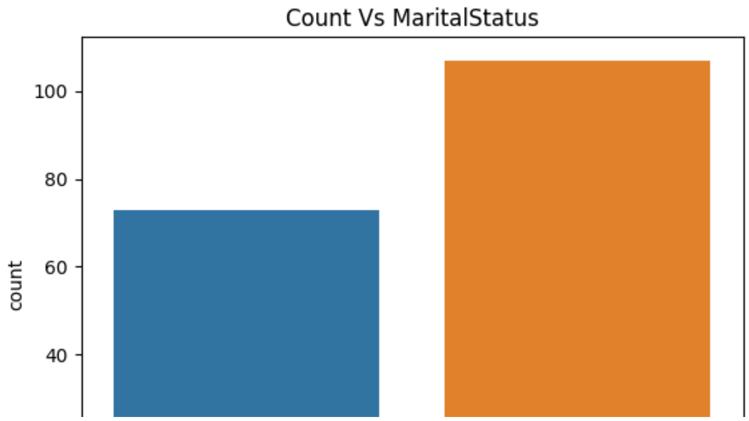

```
sns.countplot(x='Gender',data=df)
plt.title('Count Vs Gender')
```

```
Text(0.5, 1.0, 'Count Vs Gender')
```



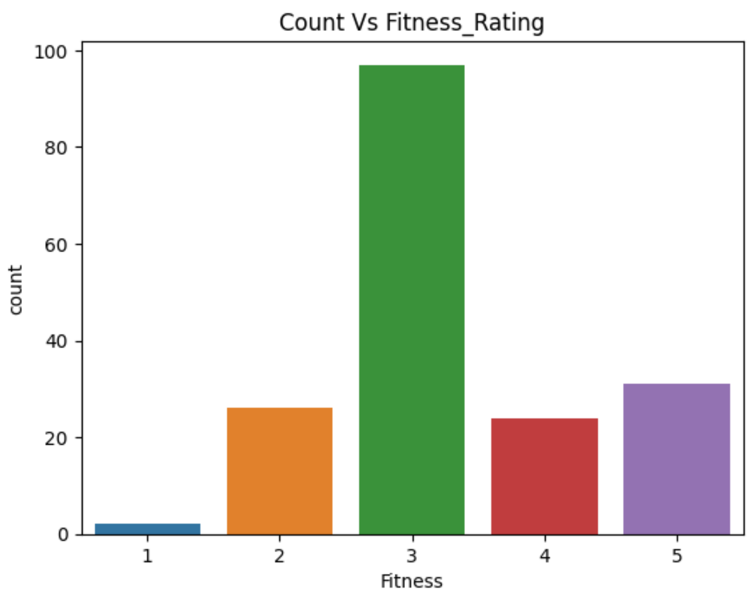

```
sns.countplot(x='MaritalStatus',data=df)
plt.title('Count Vs MaritalStatus')
```

```
Text(0.5, 1.0, 'Count Vs MaritalStatus')
```



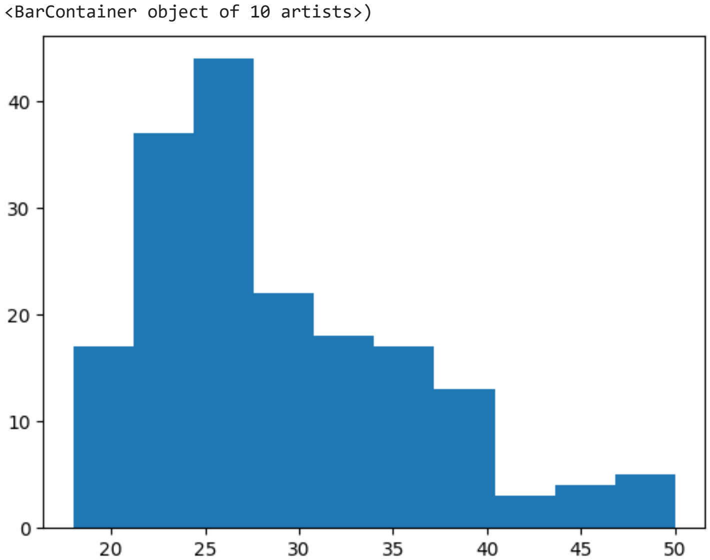

```
sns.countplot(x='Fitness',data=df)
plt.title('Count Vs Fitness_Rating')
```

```
Text(0.5, 1.0, 'Count Vs Fitness_Rating')
```



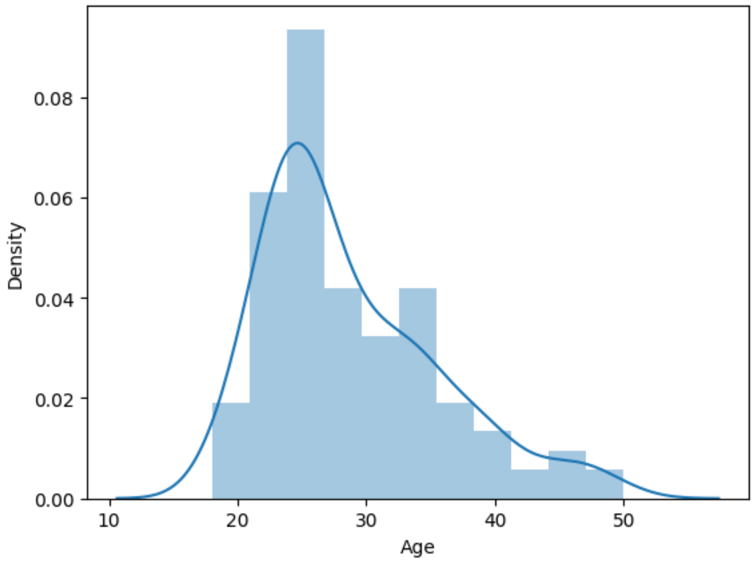

```
plt.hist(df['Age'])
```

```
(array([17., 37., 44., 22., 18., 17., 13.,  3.,  4.,  5.]),
 array([18. , 21.2, 24.4, 27.6, 30.8, 34. , 37.2, 40.4, 43.6, 46.8, 50. ]),
 <BarContainer object of 10 artists>)
```



```
sns.distplot(df['Age'])
```

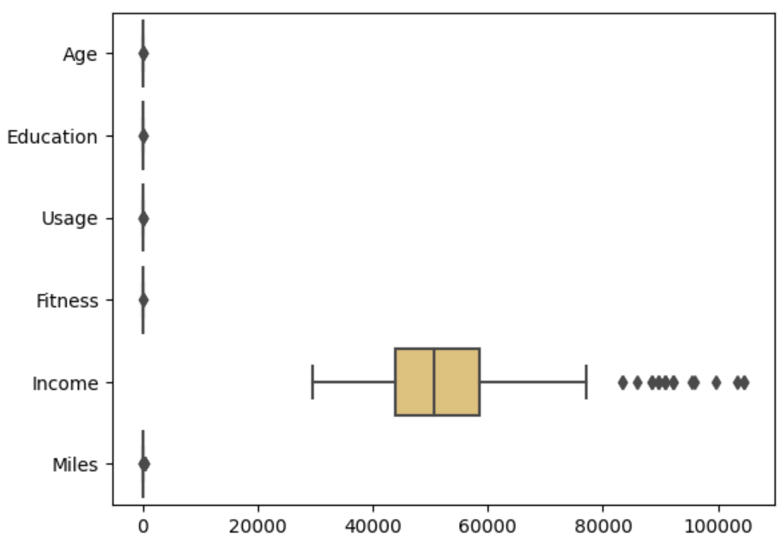
```
<ipython-input-19-0fafe04ea3f6>:1: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  sns.distplot(df['Age'])
<Axes: xlabel='Age', ylabel='Density'>
```



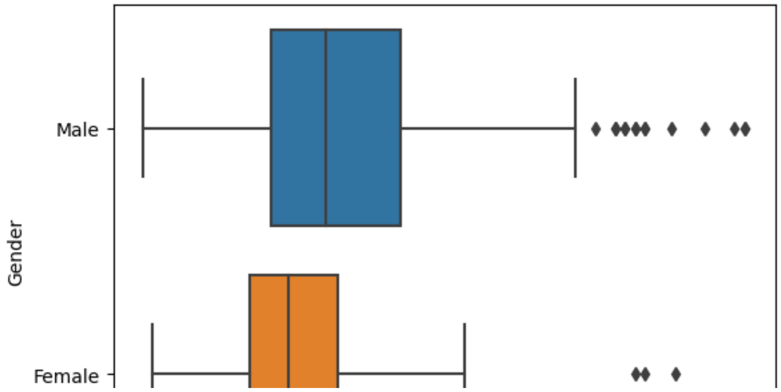

**4.2 Categorical Variable**

```
sns.boxplot(data=df, palette='rainbow',orient='h')
```

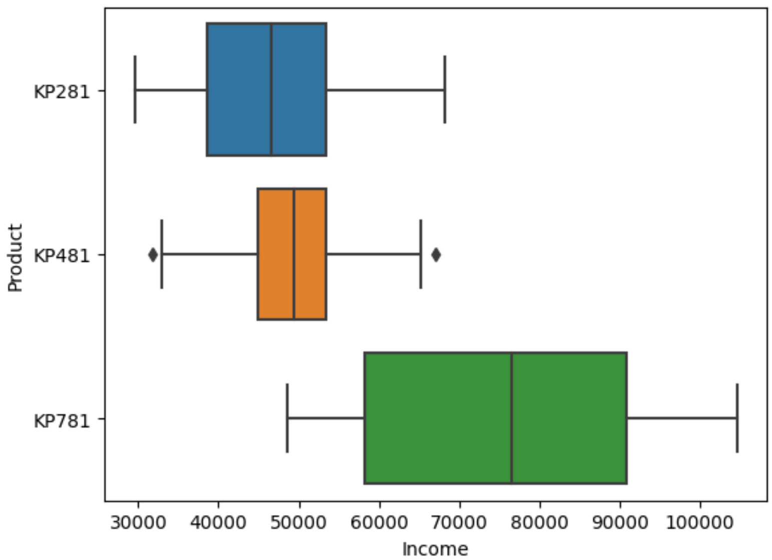
```
<Axes: >
```



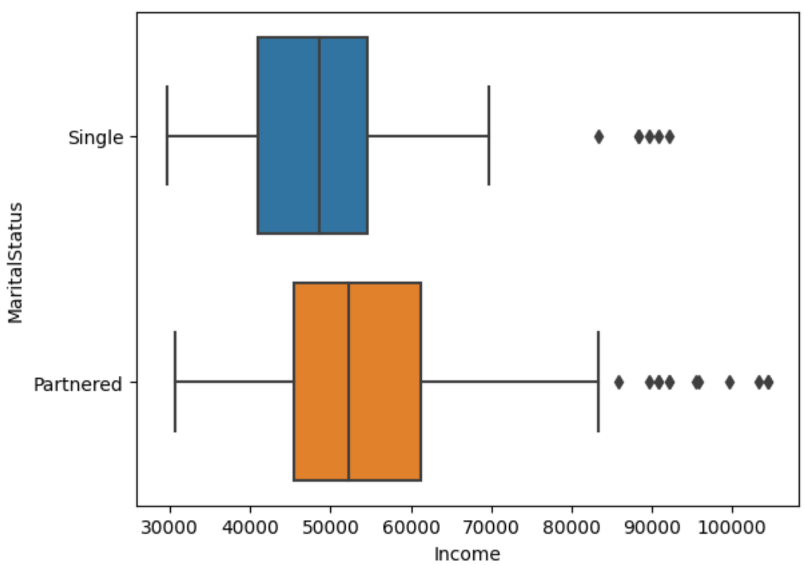

```
sns.boxplot(x='Income',y='Gender',data=df,orient='h')
plt.show()
```



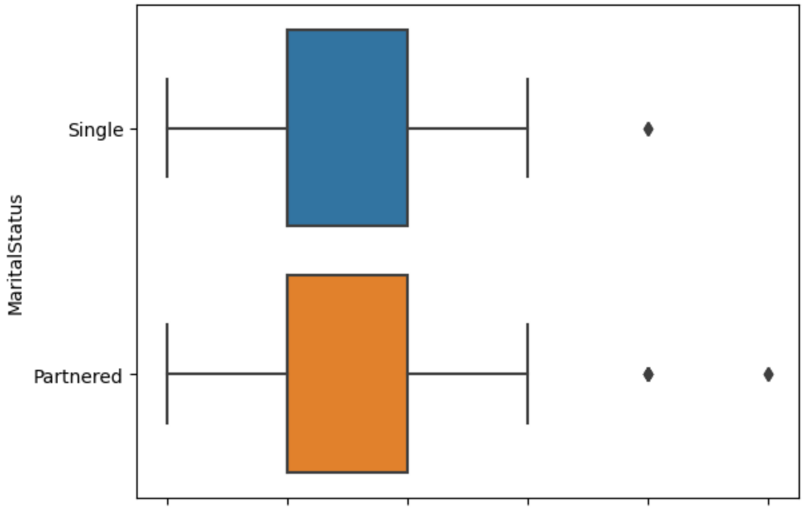

```
sns.boxplot(x='Income',y='Product',data=df,orient='h')
plt.show()
```



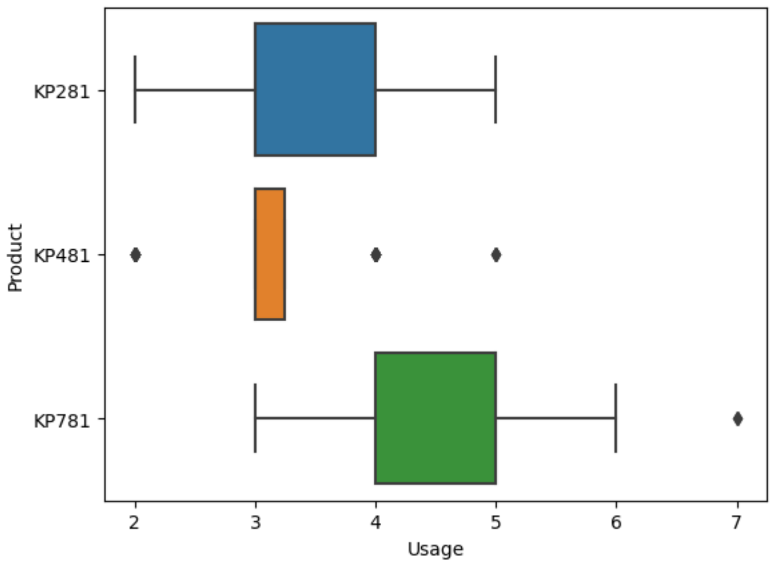

```
sns.boxplot(x='Income',y='MaritalStatus',data=df,orient='h')
plt.show()
```




```
sns.boxplot(x='Usage',y='MaritalStatus',data=df,orient='h')
plt.show()
```




```
sns.boxplot(x='Usage',y='Product',data=df,orient='h')
plt.show()
```



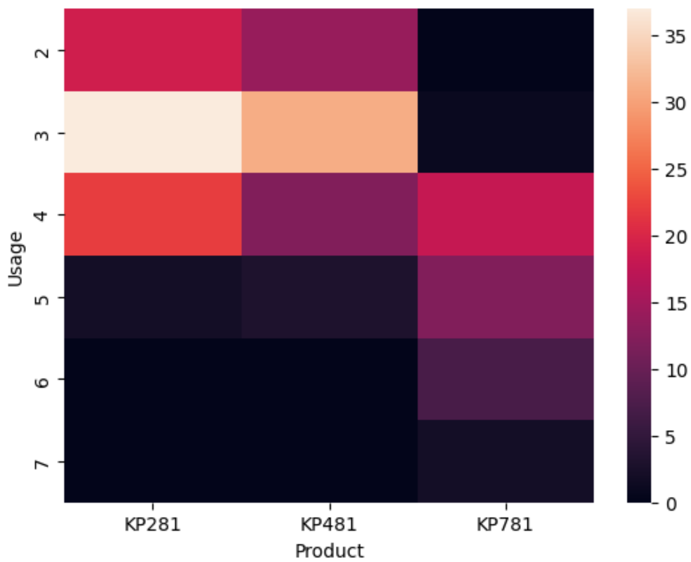

#### 4.3 For Correlation

```
sns.heatmap(pd.crosstab(df['Income'],df['Product']))
```

```
<Axes: xlabel='Product', ylabel='Income'>
```

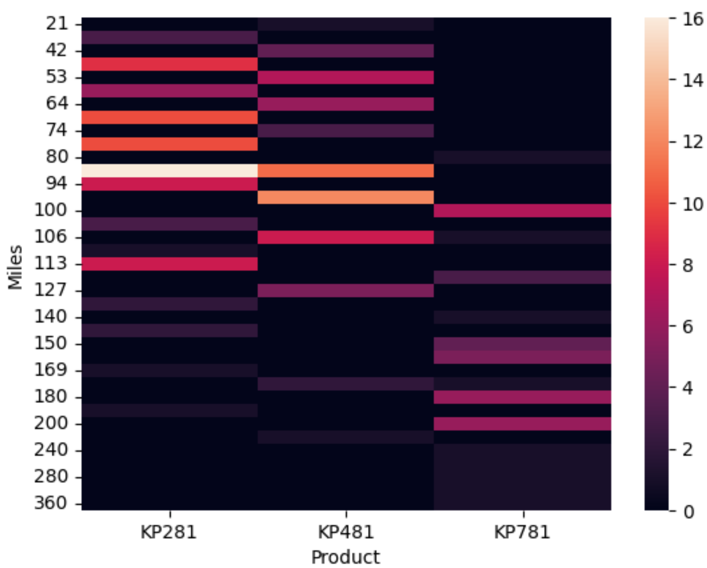
```
sns.heatmap(pd.crosstab(df['Usage'],df['Product']))
```

```
<Axes: xlabel='Product', ylabel='Usage'>
```




```
sns.heatmap(pd.crosstab(df['Miles'],df['Product']))
```

```
<Axes: xlabel='Product', ylabel='Miles'>
```



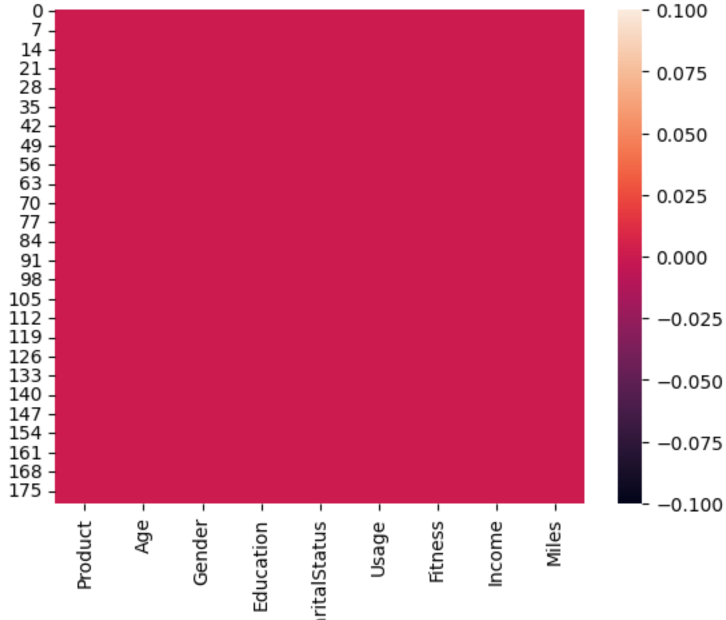

## 4.Missing Value & Outlier Detection

```
df.isna().sum()
```

```
Product          0
Age              0
Gender           0
Education        0
MaritalStatus    0
Usage            0
Fitness          0
Income           0
Miles            0
dtype: int64
```

```
sns.heatmap(df.isnull())
```

<Axes: >




## 5.Insights based on Non-Graphical and Visual Analysis

1. From the above analysis we observe that KP281 has be puurchased by maximum customer followed by KP481.
2. Purchase of tread mill is more among the Males than the Females.
3. Most of the Married or partnered people have purchased it than the singles.
4. Interms of Fitness rating, rating 3 is given to most of the customers.
5. Age group for the treadmill lie mostly between 25 to 30.
6. Interms of income, males having higher income go for purchasing while female average income lower than male purchase it.
7. KP781 as it is a higher segment product,mostly afforded by people having higher income.

```
a=df['Miles'].max()
a
```

```
360
```

```
b=df['Miles'].min()
b
```

```
21
```

```
range(a,b)
```

```
range(360, 21)
```

```
df['Age'].max()
```

```
50
```

```
df['Age'].min()
```

```
18
```

## 6.Recommendations

1. As income segment differs the product selling will be intensive in the case of KP281 as most people fall under the segment.
2. KP781 to be pitched to high end customers having higher incomes.
3. Most of the product selling coming from married customers which should be the focussed area for selling.

4. Most of the young people having age ranging from 25-30 being health cautious, are more inclined towards fitness and can be prospect customers.
5. Males purchase is more compared to female o the other hand female having lower average income purchase than the average income og male. So focus towards female segment can bring in lots of profits.

0s completed at 9:00 PM