AEROFIT CASE STUDY

Import the dataset and do usual data analysis steps like checking the structure & characteristics of the dataset

In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

In [2]:
```
df=pd.read_csv(r'\Users\Home\Downloads\aerofit_treadmill.csv')
df
```

Out[2]:

	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
...	...	...	...	...	...	...	...	...	...
175	KP781	40	Male	21	Single	6	5	83416	200
176	KP781	42	Male	18	Single	5	4	89641	200
177	KP781	45	Male	16	Single	5	5	90886	160
178	KP781	47	Male	18	Partnered	4	5	104581	120
179	KP781	48	Male	18	Partnered	4	5	95508	180

180 rows × 9 columns

Defining Problem Statement and Analysing basic metrics

Aerofit is a leading brand in the field of fitness equipment that provides a product range including machines such as treadmills, exercise bikes and other fitness accessories to cater to the needs of all categories of people. The present case study aims 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 case study also aims at finding out whether there are differences across the product with respect to customer characteristics.

Observations on shape of data, data types of all the attributes, conversion of categorical attributes to 'category' (If required), statistical summary

In [3]:
```
df.shape  #shape of data
```

Out[3]:
```
(180, 9)
```

In [4]:
```
df.info()  #data type of all attributes
```

```
<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
```

In [5]:
```
df.describe(include='all')  #statistical summary
```

Out[5]:

	Product	Age	Gender	Education	MaritalStatus	Usage	Fitness	Incom
count	180	180.000000	180	180.000000	180	180.000000	180.000000	180.00000
unique	3	NaN	2	NaN	2	NaN	NaN	Na
top	KP281	NaN	Male	NaN	Partnered	NaN	NaN	Na
freq	80	NaN	104	NaN	107	NaN	NaN	Na
mean	NaN	28.788889	NaN	15.572222	NaN	3.455556	3.311111	53719.57777
std	NaN	6.943498	NaN	1.617055	NaN	1.084797	0.958869	16506.68422
min	NaN	18.000000	NaN	12.000000	NaN	2.000000	1.000000	29562.00000
25%	NaN	24.000000	NaN	14.000000	NaN	3.000000	3.000000	44058.75000
50%	NaN	26.000000	NaN	16.000000	NaN	3.000000	3.000000	50596.50000
75%	NaN	33.000000	NaN	16.000000	NaN	4.000000	4.000000	58668.00000
max	NaN	50.000000	NaN	21.000000	NaN	7.000000	5.000000	104581.00000

In [6]:
```
#conversion of categorical attributes to 'category'
KP281 = df[df['Product']=='KP281']
KP281
```

Out[6]:

	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
...	...	...	...	...	...	...	...	...	...
75	KP281	43	Male	16	Partnered	3	3	53439	66
76	KP281	44	Female	16	Single	3	4	57987	75
77	KP281	46	Female	16	Partnered	3	2	60261	47
78	KP281	47	Male	16	Partnered	4	3	56850	94
79	KP281	50	Female	16	Partnered	3	3	64809	66

80 rows × 9 columns

In [7]:

```
KP481 = df[df['Product']=='KP481']
KP481.head()
```

Out[7]:

	Product	Age	Gender	Education	MaritalStatus	Usage	Fitness	Income	Miles
80	KP481	19	Male	14	Single	3	3	31836	64
81	KP481	20	Male	14	Single	2	3	32973	53
82	KP481	20	Female	14	Partnered	3	3	34110	106
83	KP481	20	Male	14	Single	3	3	38658	95
84	KP481	21	Female	14	Partnered	5	4	34110	212

In [8]:

```
KP781 = df[df['Product']=='KP781']
KP781.head()
```

Out[8]:

	Product	Age	Gender	Education	MaritalStatus	Usage	Fitness	Income	Miles
140	KP781	22	Male	14	Single	4	3	48658	106
141	KP781	22	Male	16	Single	3	5	54781	120
142	KP781	22	Male	18	Single	4	5	48556	200
143	KP781	23	Male	16	Single	4	5	58516	140
144	KP781	23	Female	18	Single	5	4	53536	100

In [9]:

```
Male = df[df['Gender']=='Male']
Male
```

Out[9]:

	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
3	KP281	19	Male	12	Single	3	3	32973	85
4	KP281	20	Male	13	Partnered	4	2	35247	47
7	KP281	21	Male	13	Single	3	3	32973	85
...	...	...	...	...	...	...	...	...	...
175	KP781	40	Male	21	Single	6	5	83416	200
176	KP781	42	Male	18	Single	5	4	89641	200
177	KP781	45	Male	16	Single	5	5	90886	160
178	KP781	47	Male	18	Partnered	4	5	104581	120
179	KP781	48	Male	18	Partnered	4	5	95508	180

104 rows × 9 columns

In [10]:

```
Female = df[df['Gender']=='Female']
Female
```

Out[10]:

	Product	Age	Gender	Education	MaritalStatus	Usage	Fitness	Income	Miles
2	KP281	19	Female	14	Partnered	4	3	30699	66
5	KP281	20	Female	14	Partnered	3	3	32973	66
6	KP281	21	Female	14	Partnered	3	3	35247	75
9	KP281	21	Female	15	Partnered	2	3	37521	85
11	KP281	22	Female	14	Partnered	3	2	35247	66
...	...	...	...	...	...	...	...	...	...
152	KP781	25	Female	18	Partnered	5	5	61006	200
157	KP781	26	Female	21	Single	4	3	69721	100
162	KP781	28	Female	18	Partnered	6	5	92131	180
167	KP781	30	Female	16	Partnered	6	5	90886	280
171	KP781	33	Female	18	Partnered	4	5	95866	200

76 rows × 9 columns

In [11]:

```
Single = df[df['MaritalStatus']=='Single']
Single
```

Out[11]:

	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
3	KP281	19	Male	12	Single	3	3	32973	85
7	KP281	21	Male	13	Single	3	3	32973	85
8	KP281	21	Male	15	Single	5	4	35247	141
...	...	...	...	...	...	...	...	...	...
165	KP781	29	Male	18	Single	5	5	52290	180
172	KP781	34	Male	16	Single	5	5	92131	150
175	KP781	40	Male	21	Single	6	5	83416	200
176	KP781	42	Male	18	Single	5	4	89641	200
177	KP781	45	Male	16	Single	5	5	90886	160

73 rows × 9 columns

In [12]:

```
Partnered = df[df['MaritalStatus']=='Partnered']
Partnered
```

Out[12]:

	Product	Age	Gender	Education	MaritalStatus	Usage	Fitness	Income	Miles
2	KP281	19	Female	14	Partnered	4	3	30699	66
4	KP281	20	Male	13	Partnered	4	2	35247	47
5	KP281	20	Female	14	Partnered	3	3	32973	66
6	KP281	21	Female	14	Partnered	3	3	35247	75
9	KP281	21	Female	15	Partnered	2	3	37521	85
...	...	...	...	...	...	...	...	...	...
171	KP781	33	Female	18	Partnered	4	5	95866	200
173	KP781	35	Male	16	Partnered	4	5	92131	360
174	KP781	38	Male	18	Partnered	5	5	104581	150
178	KP781	47	Male	18	Partnered	4	5	104581	120
179	KP781	48	Male	18	Partnered	4	5	95508	180

107 rows × 9 columns

Non-Graphical Analysis: Value counts and unique attributes

Questionnaire:1. What is the total count of each product present in the dataset?

In [13]:

```
df.Product.value_counts()
```

Out[13]:
```
KP281    80
KP481    60
KP781    40
Name: Product, dtype: int64
```

In [14]:
```
df.Age.value_counts()
```

Out[14]:
```
25    25
23    18
24    12
26    12
28     9
35     8
33     8
30     7
38     7
21     7
22     7
27     7
31     6
34     6
29     6
20     5
40     5
32     4
19     4
48     2
37     2
45     2
47     2
46     1
50     1
18     1
44     1
43     1
41     1
39     1
36     1
42     1
Name: Age, dtype: int64
```

In [15]:
```
df.Gender.value_counts()
```

Out[15]:
```
Male      104
Female     76
Name: Gender, dtype: int64
```

In [16]:
```
df.Education.value_counts()
```

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

In [17]:
```
df.MaritalStatus.value_counts()
```

Out[17]:
```
Partnered    107
Single        73
Name: MaritalStatus, dtype: int64
```

In [18]:
```
df.Usage.value_counts()
```

Out[18]:
```
3    69
4    52
2    33
5    17
6     7
7     2
Name: Usage, dtype: int64
```

In [19]:
```
df.Fitness.value_counts()
```

Out[19]:
```
3    97
5    31
2    26
4    24
1     2
Name: Fitness, dtype: int64
```

In [20]:
```
df.Income.value_counts()
```

Out[20]:
```
45480    14
52302     9
46617     8
54576     8
53439     8
         ..
65220     1
55713     1
68220     1
30699     1
95508     1
Name: Income, Length: 62, dtype: int64
```

In [21]:
```
df.Miles.value_counts()
```

Out[21]:
```
85     27
95     12
66     10
75     10
47      9
106     9
94      8
113     8
53      7
100     7
180     6
200     6
56      6
64      6
127     5
160     5
42      4
150     4
38      3
74      3
170     3
120     3
103     3
132     2
141     2
280     1
260     1
300     1
240     1
112     1
212     1
80      1
140     1
21      1
169     1
188     1
360     1
Name: Miles, dtype: int64
```

Missing Value & Outlier Detection

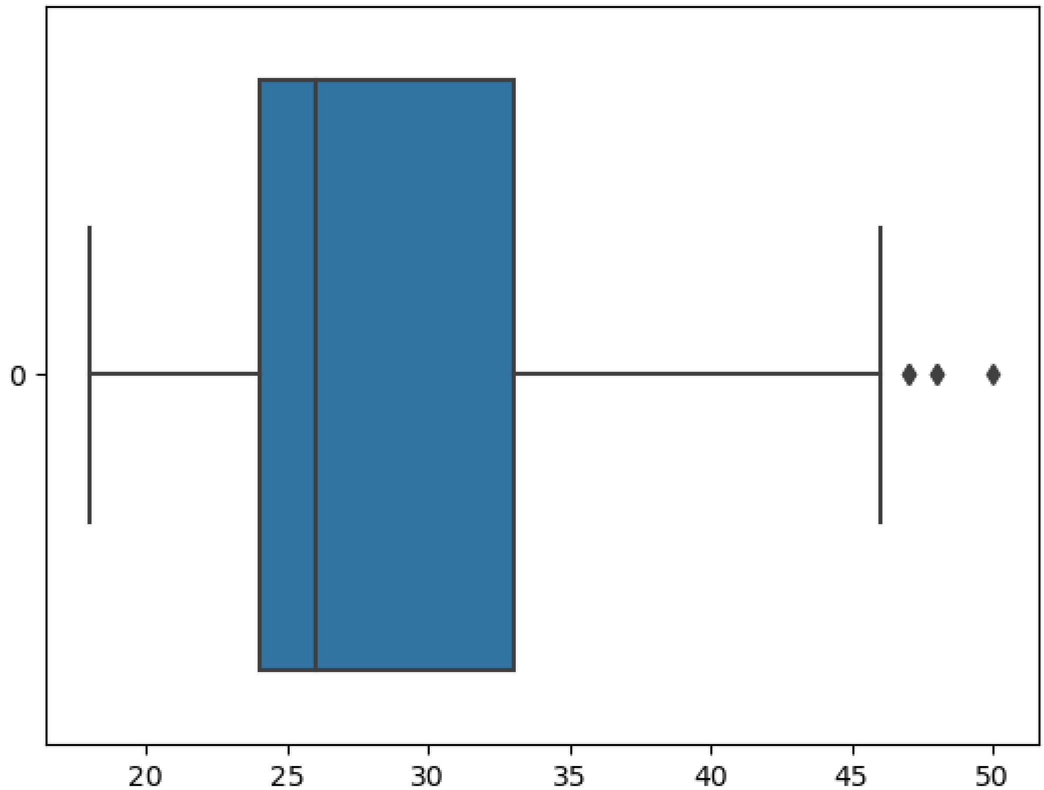
In [22]:
```
df.isna().sum()  #checking for null values
```

Out[22]:
```
Product          0
Age              0
Gender           0
Education        0
MaritalStatus    0
Usage            0
Fitness          0
Income           0
Miles            0
dtype: int64
```

Outlier Detection

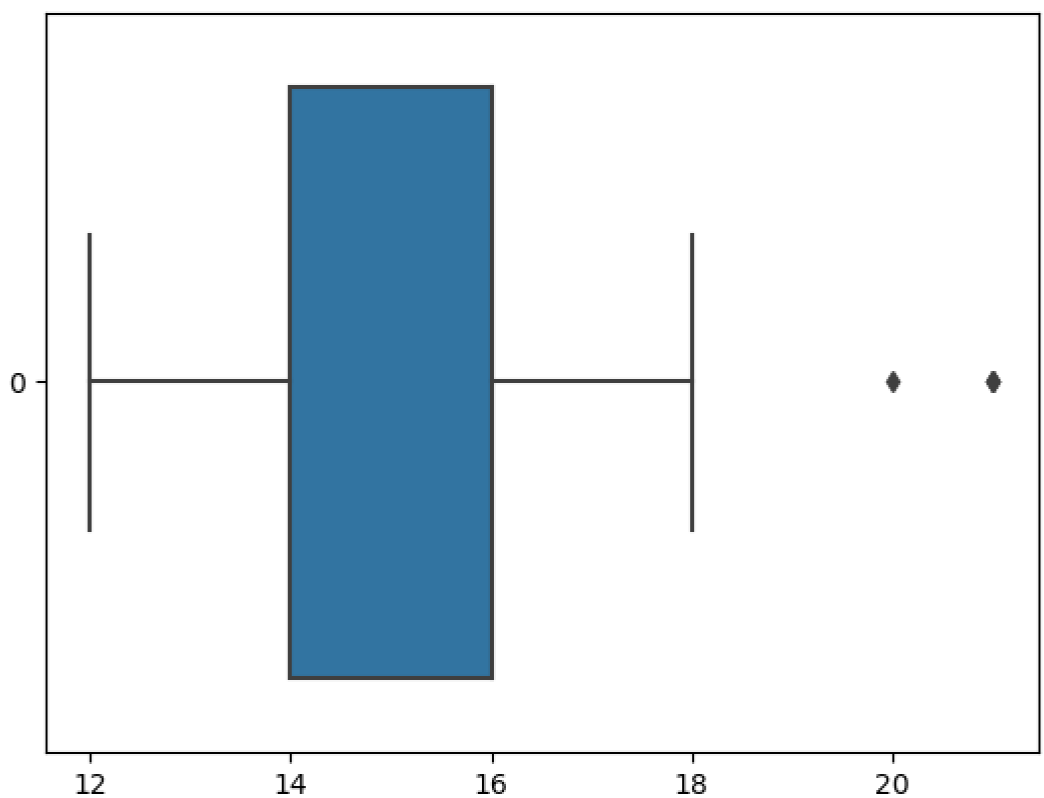
In [23]:
```
sns.boxplot(data=df["Age"], orient="h")
```

Out[23]:
```
<AxesSubplot:>
```



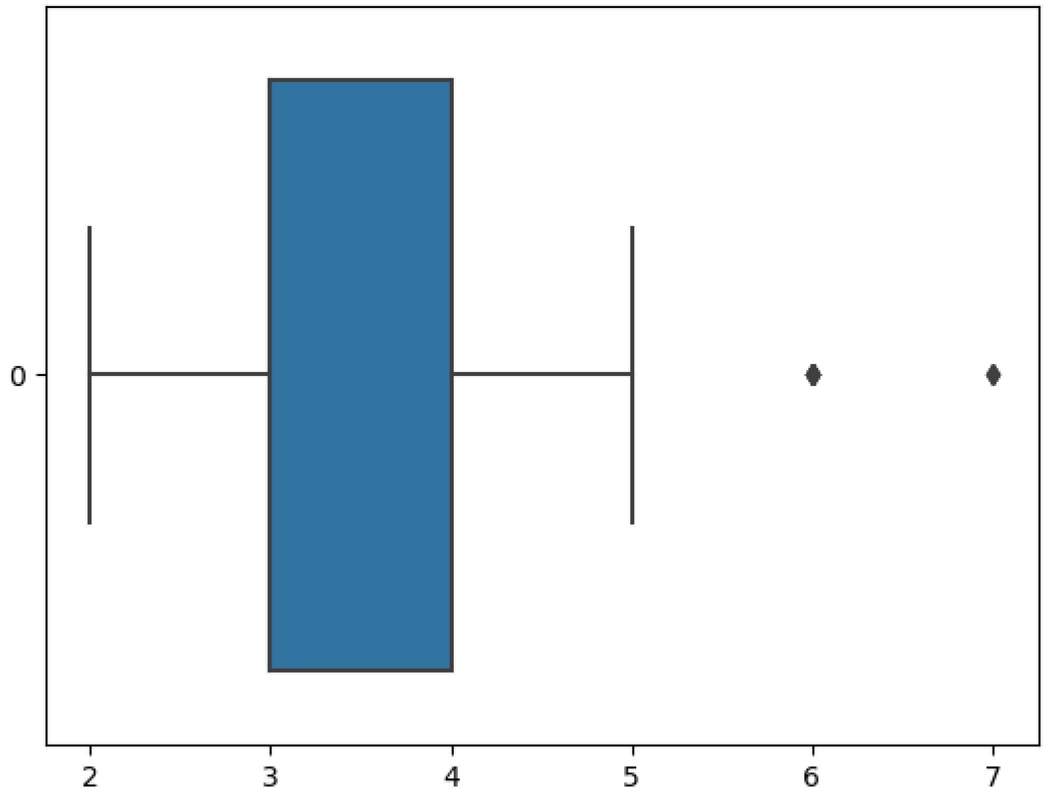
In [24]:
```
sns.boxplot(data=df["Education"], orient="h")
```

Out[24]: <AxesSubplot:>



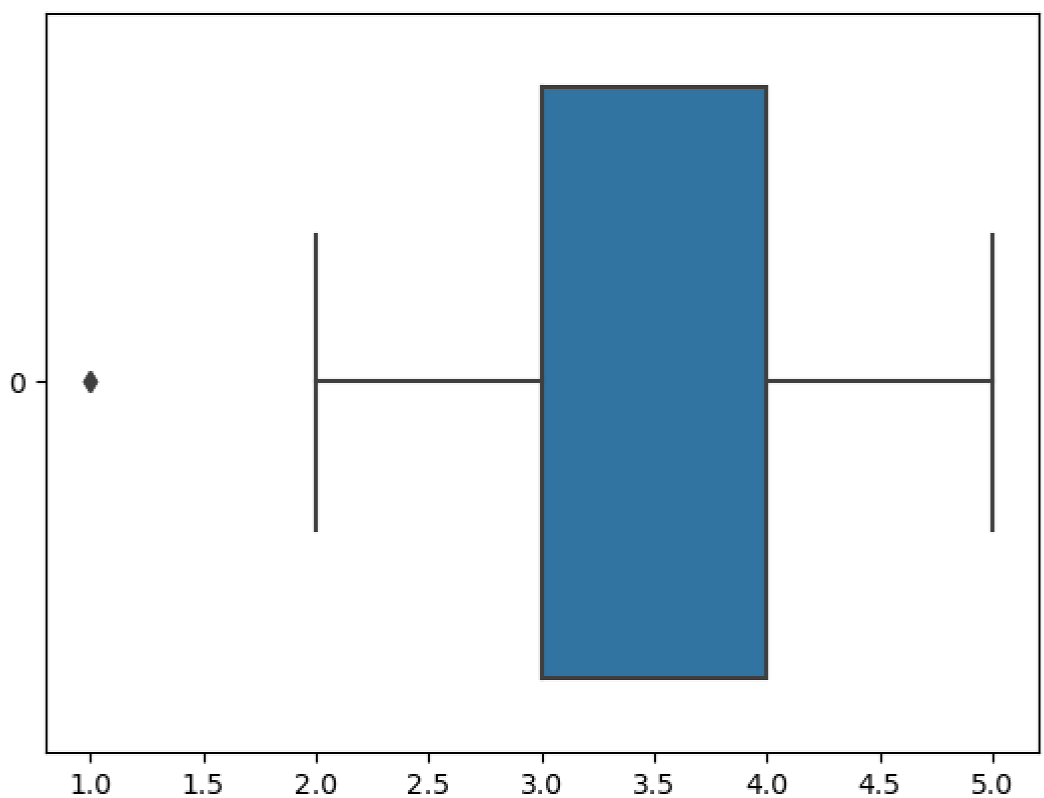
In [25]:
```
sns.boxplot(data=df["Usage"], orient="h")
```

Out[25]: <AxesSubplot:>



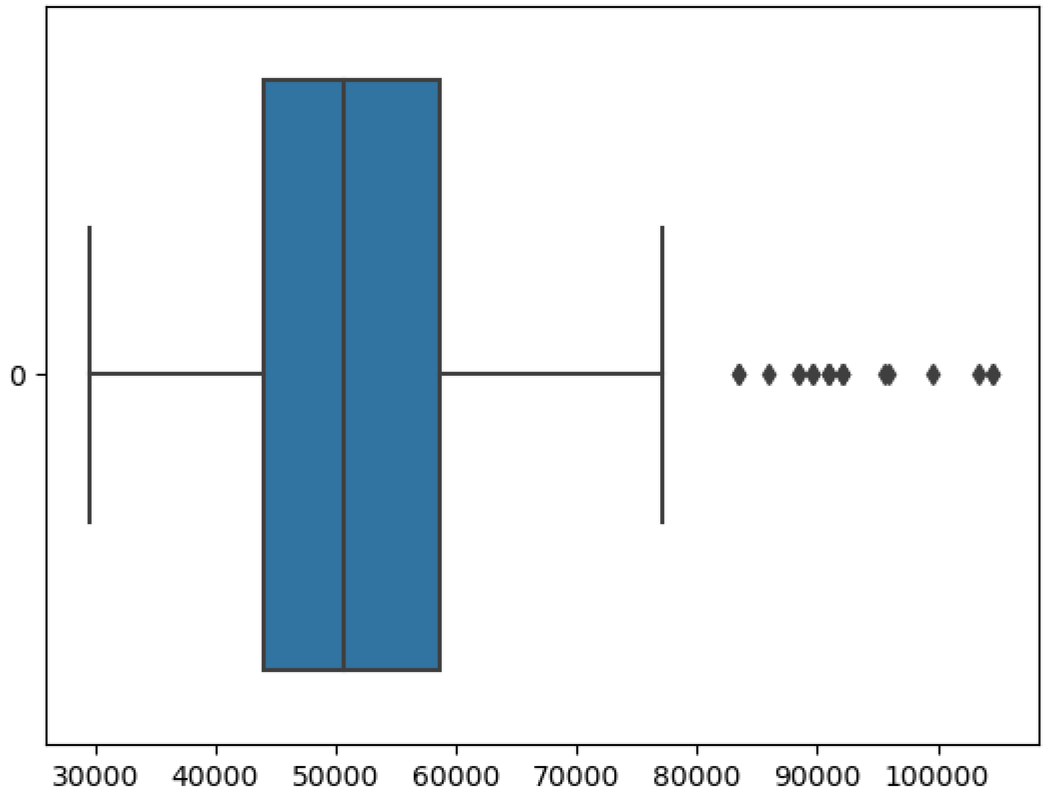
In [26]:
```
sns.boxplot(data=df["Fitness"], orient="h")
```

Out[26]: <AxesSubplot:>



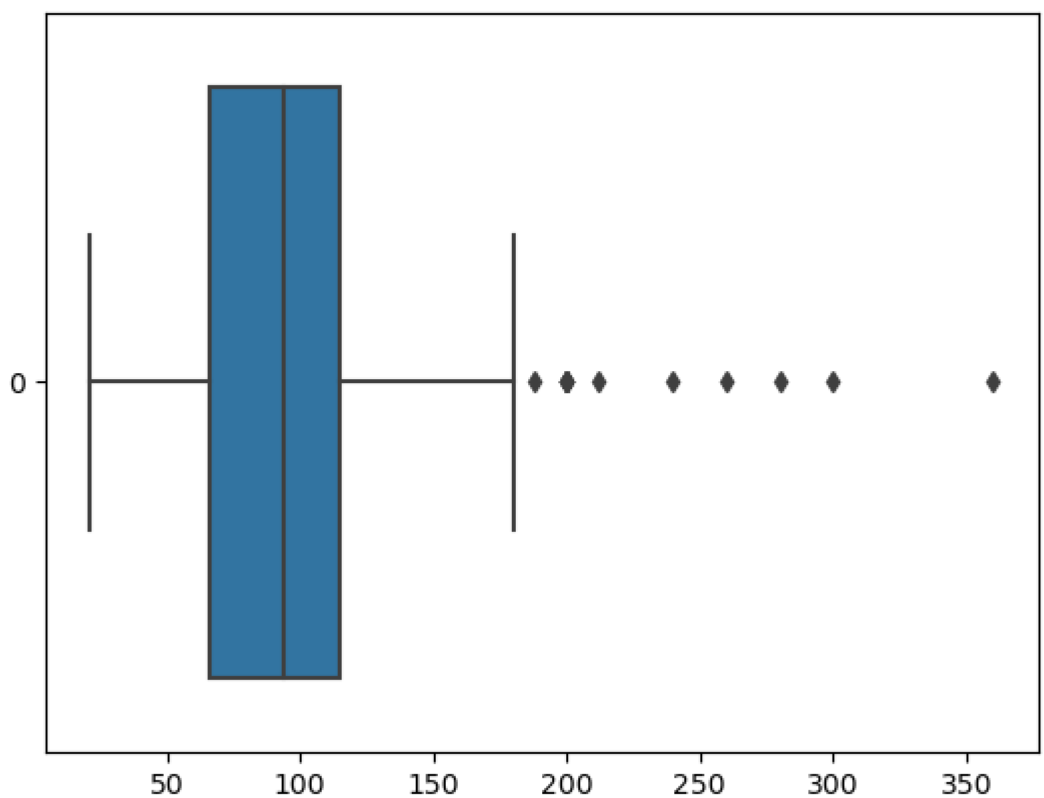
In [27]:
```
sns.boxplot(data=df["Income"], orient="h")
```

Out[27]: <AxesSubplot:>



In [28]:
```
sns.boxplot(data=df["Miles"], orient="h")
```

Out[28]: <AxesSubplot:>



Questionnaire: 4. Were there any outliers present in the data? If yes, suggest a suitable method for their treatment.

There are some outliers present in the dataframe as depicted in the above boxplots. There are several methods for treatment of these outliers. If the number of outliers is not high, then the same can be dropped as it would not affect the calculations. Otherwise, rescaling of the data can also be done.

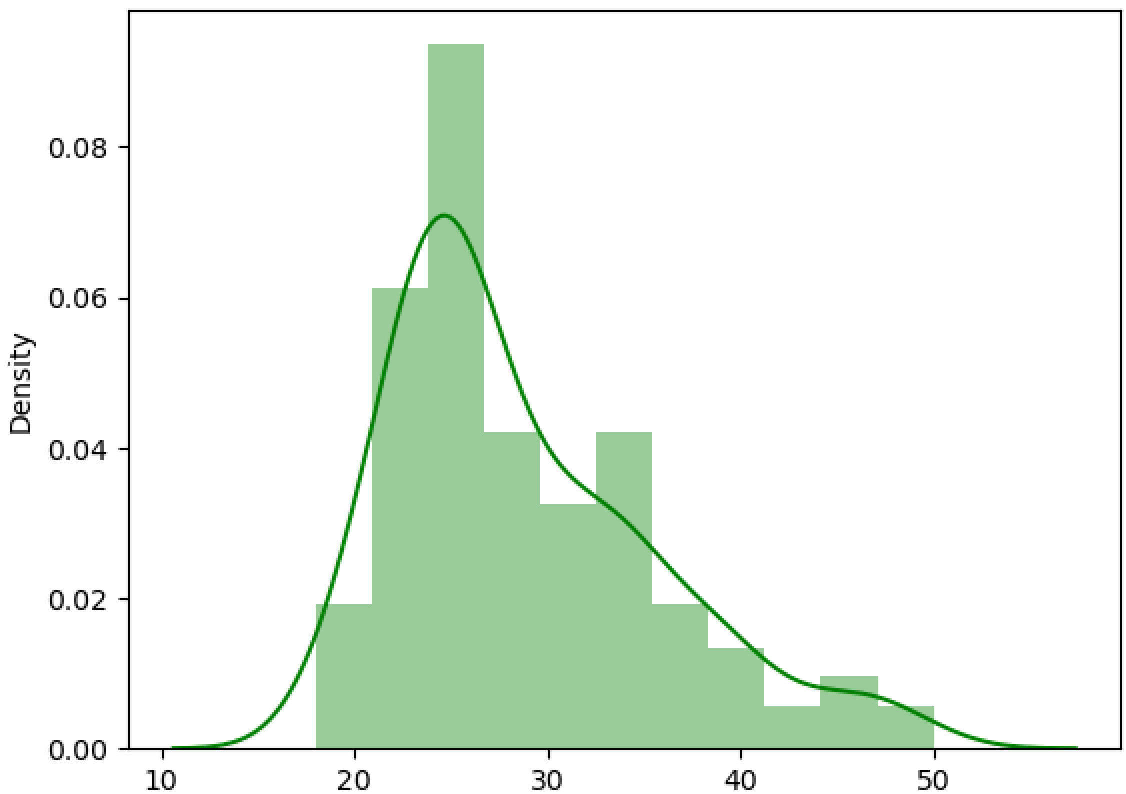
Visual Analysis - Univariate & Bivariate

For continuous variable(s)

In [29]:
```
sns.distplot(x=df['Age'],color="Green")
```

```
C:\Users\Home\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarnin
g: `distplot` is a deprecated function and will be removed in a future version. Pleas
e adapt your code to use either `displot` (a figure-level function with similar flexi
bility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

Out[29]: <AxesSubplot:ylabel='Density'>

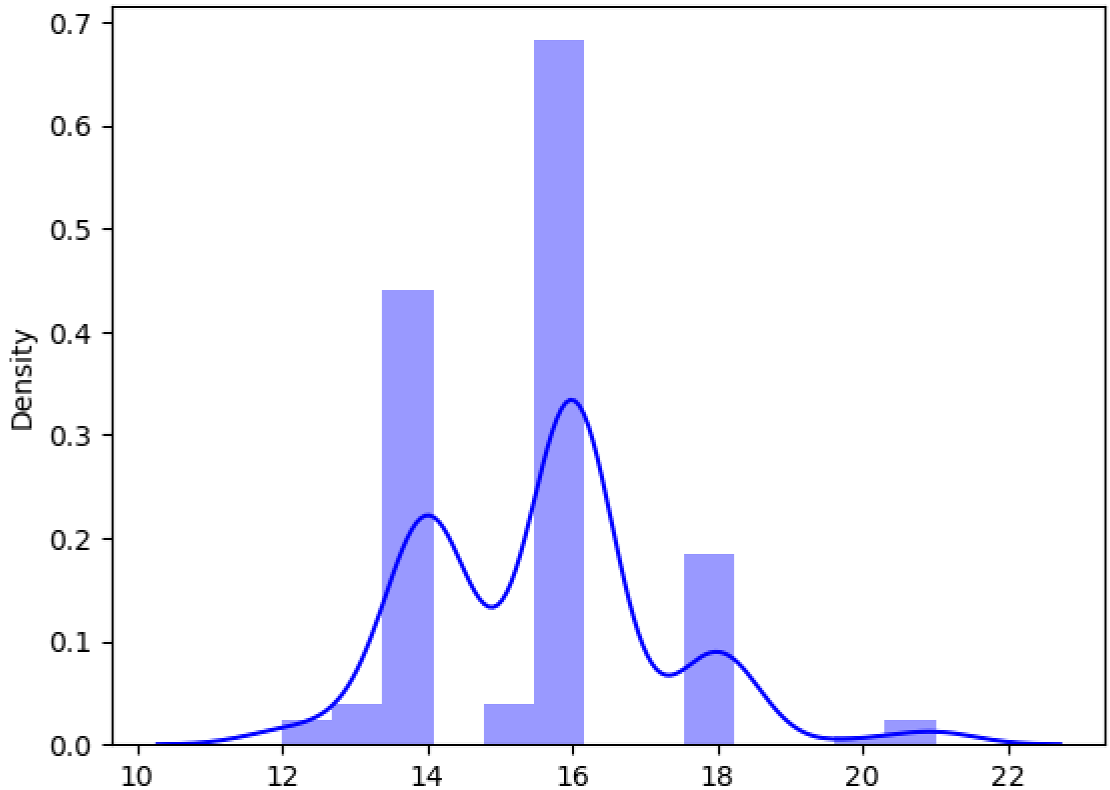


From the above, we can observe that the maximum number of users are in their mid-20s.

In [30]:
```
sns.distplot(x=df['Education'], color='blue')
```

```
C:\Users\Home\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarnin
g: `distplot` is a deprecated function and will be removed in a future version. Pleas
e adapt your code to use either `displot` (a figure-level function with similar flexi
bility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

Out[30]: <AxesSubplot:ylabel='Density'>



From the above, we can observe that users with 16 years of education are highest in number.

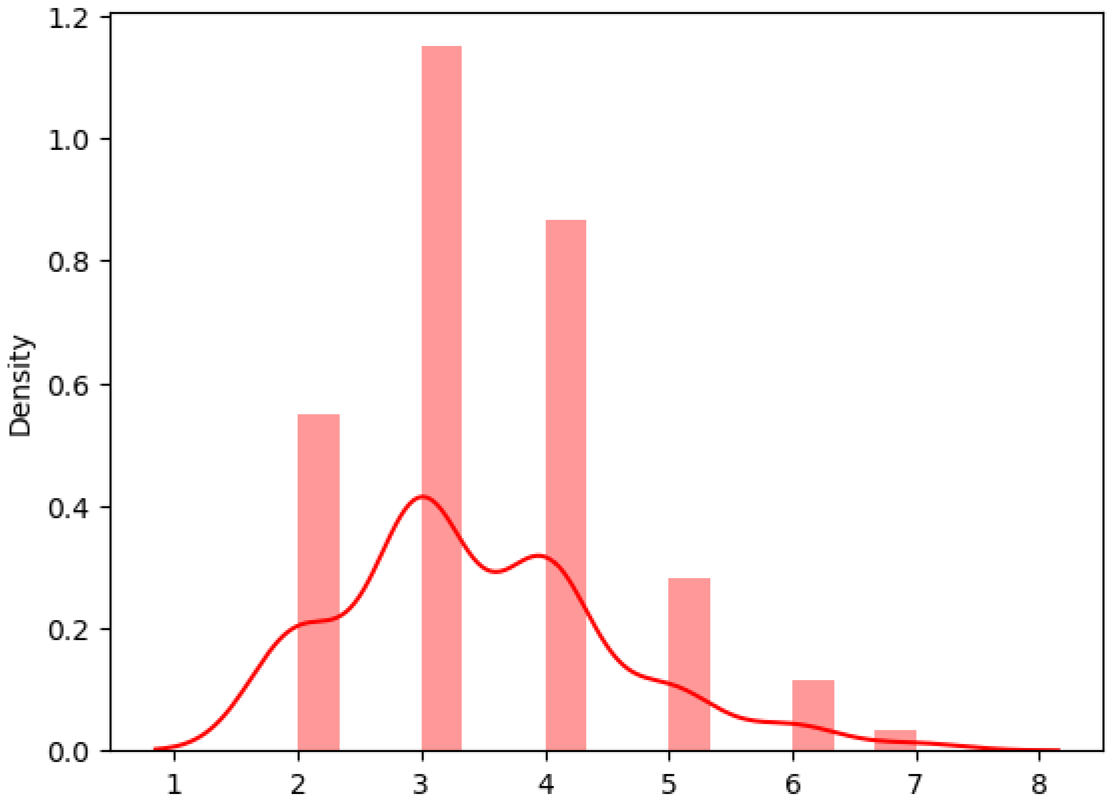
In [31]:

```
sns.distplot(x=df['Usage'], color='red')
```

```
C:\Users\Home\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarnin
g: `distplot` is a deprecated function and will be removed in a future version. Pleas
e adapt your code to use either `displot` (a figure-level function with similar flexi
bility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

Out[31]:

```
<AxesSubplot:ylabel='Density'>
```



From the above, we can observe that maximum number of users plan to use the treadmill thrice a week, while only very few plan to use it 6-7 times a week.

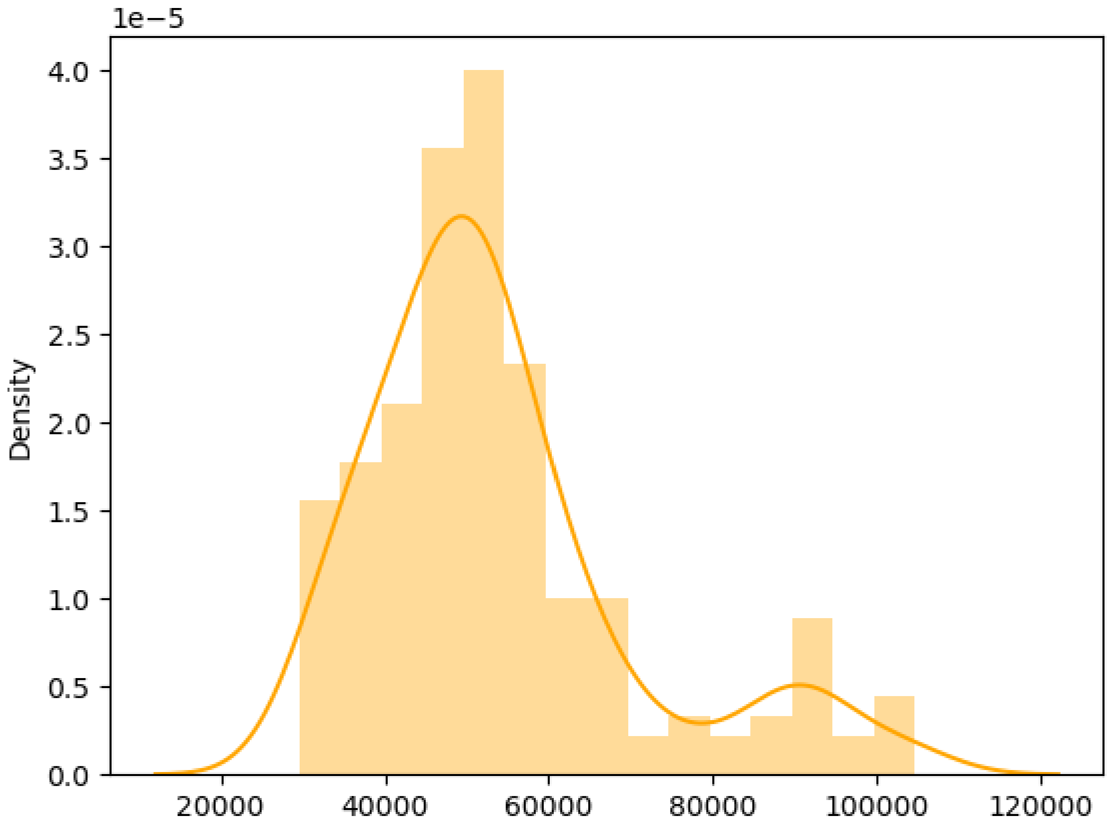
Questionnaire: 6. The variance of income in lower ages is smaller as compared to the variance in higher ages, In statistics, this is known as.. a) Heteroscedasticity b) Linearity c)Homoscedasticity d)Normality

Ans. Heteroscedasticity - The same is visible from the below distplot.

In [32]:
```
sns.distplot(x=df['Income'], color='orange')
```

```
C:\Users\Home\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarnin
g: `distplot` is a deprecated function and will be removed in a future version. Pleas
e adapt your code to use either `displot` (a figure-level function with similar flexi
bility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

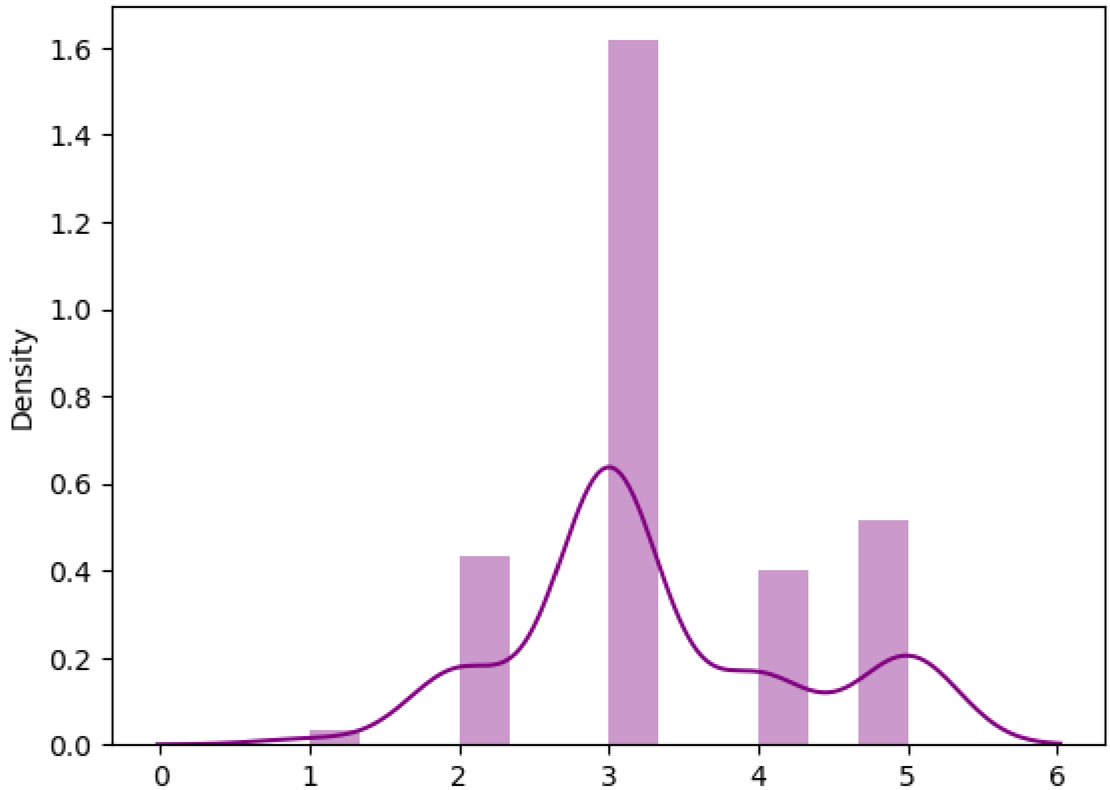
Out[32]:
```
<AxesSubplot:ylabel='Density'>
```



In [33]:
```
sns.distplot(x=df['Fitness'], color='purple')
```

```
C:\Users\Home\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarnin
g: `distplot` is a deprecated function and will be removed in a future version. Pleas
e adapt your code to use either `displot` (a figure-level function with similar flexi
bility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

Out[33]:
```
<AxesSubplot:ylabel='Density'>
```

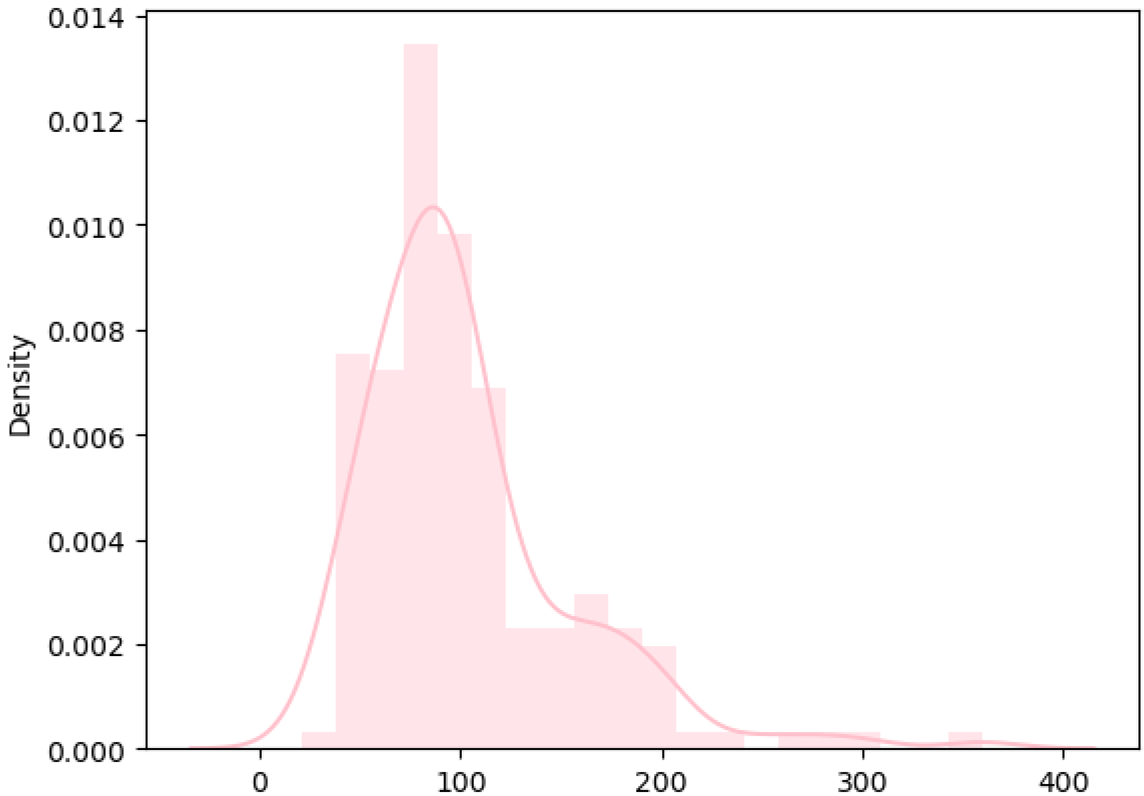


From the above, we observe that maximum number of people feel that they are at point 3 on the fitness scale.

In [34]:
```
sns.distplot(x=df['Miles'], color='pink')
```

```
C:\Users\Home\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarnin
g: `distplot` is a deprecated function and will be removed in a future version. Pleas
e adapt your code to use either `displot` (a figure-level function with similar flexi
bility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

Out[34]:
```
<AxesSubplot:ylabel='Density'>
```



We observe that most number of people expect to run/walk atmost 100 miles each week.

For categorical variable(s)

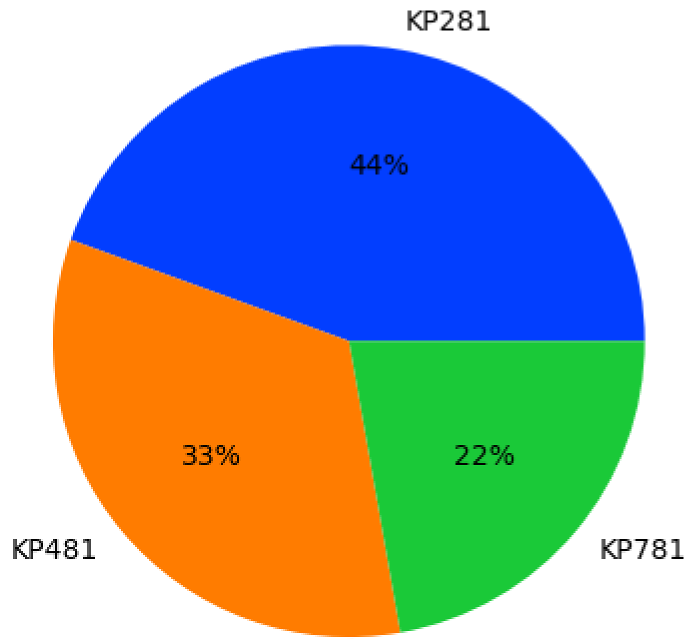
Questionnaire: 9. The overall Probability of Purchase for KP281, KP481 & KP781 treadmills is 0.44, 0.33 & 0.22.

In [35]:
```
Product = pd.DataFrame({"Product":['KP281', 'KP481', 'KP781'],
                        "Probability":['0.44', '0.33', '0.22']}) #from pie-chart belo
Product
```

Out[35]:

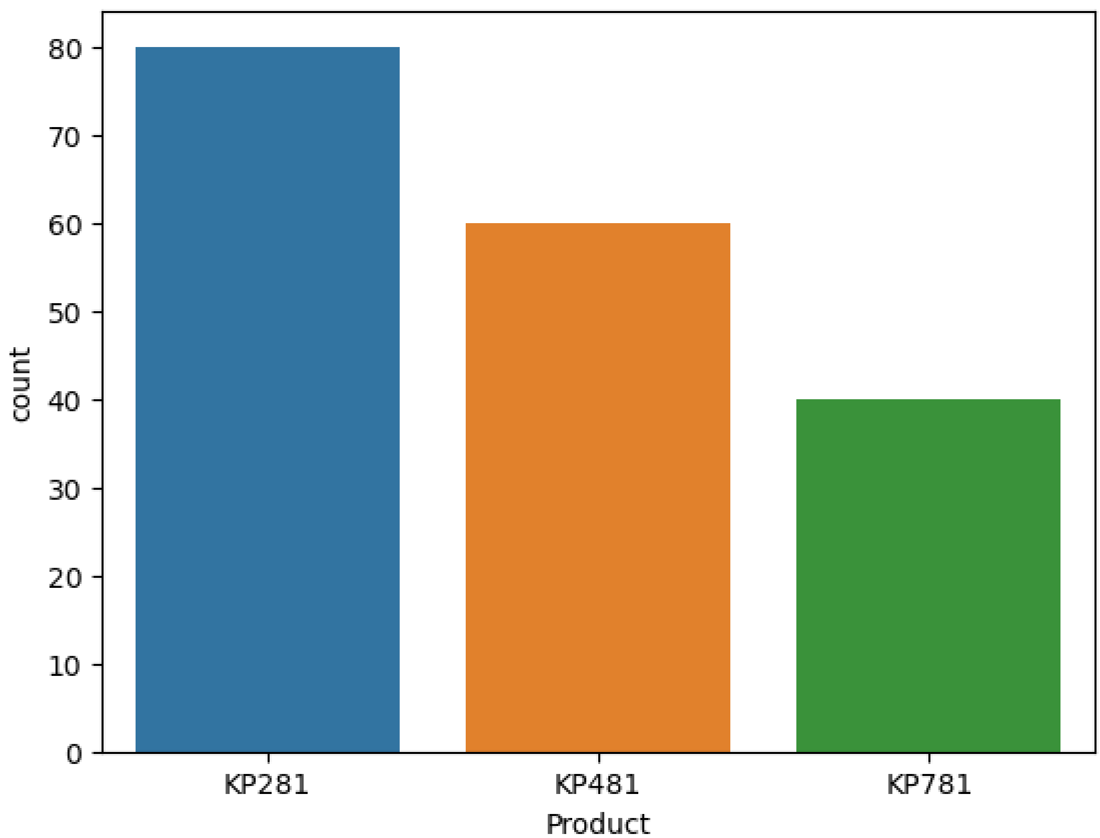
	Product	Probability
0	KP281	0.44
1	KP481	0.33
2	KP781	0.22

In [36]:
```
palette_color = sns.color_palette('bright')
keys=['KP281','KP481', 'KP781']
plt.pie(df.Product.value_counts(), labels=keys, colors=palette_color, autopct='%.0f%%'
plt.show()
```



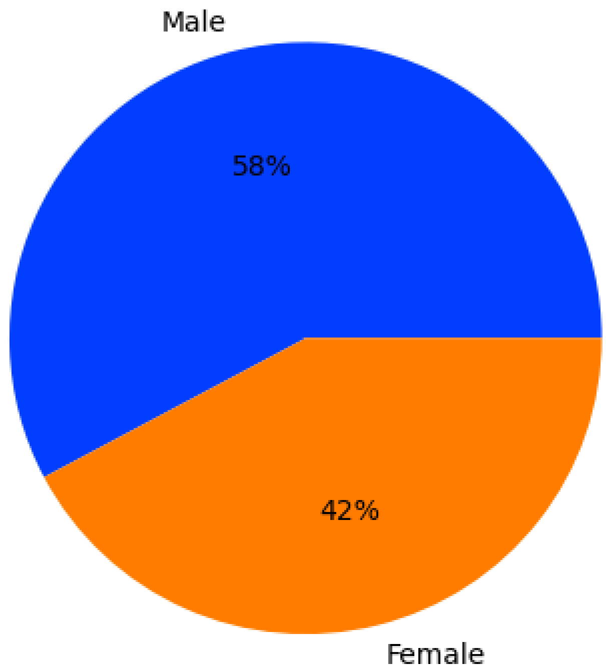
In [37]:
```
sns.countplot(x=df['Product'])
```

Out[37]: <AxesSubplot:xlabel='Product', ylabel='count'>



From the above, we can observe that fewer people use KP781 treadmill as compared to KP281.

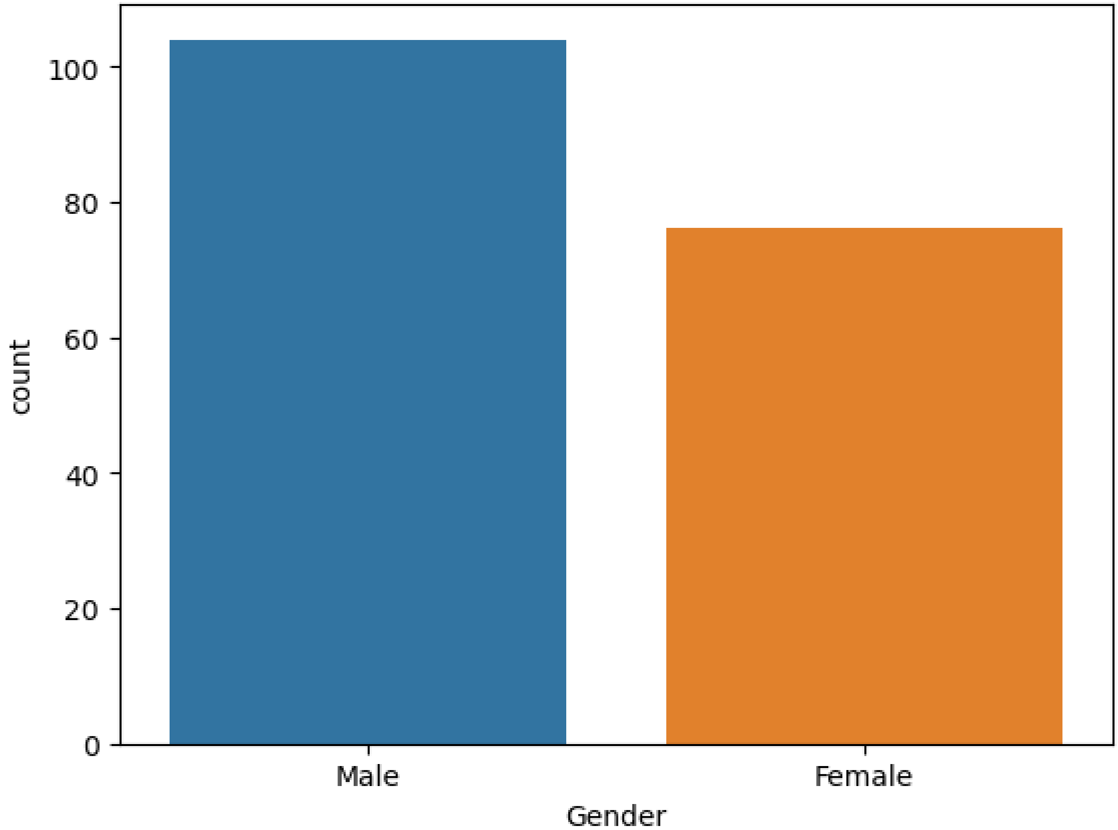
In [38]:
```
palette_color = sns.color_palette('bright')
keys=['Male','Female']
plt.pie(df.Gender.value_counts(), labels=keys, colors=palette_color, autopct='%.0f%%')
plt.show()
```



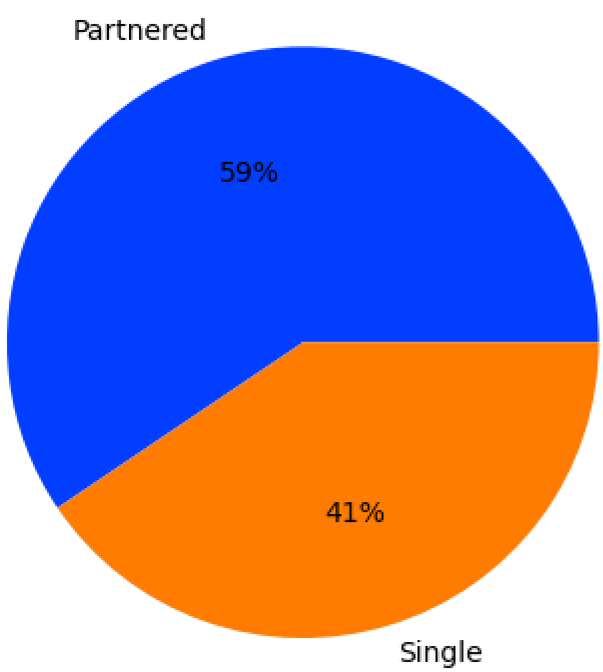
The data covers 58% males and 42% females.

In [39]:
```
sns.countplot(x=df['Gender'])
```

Out[39]:
```
<AxesSubplot:xlabel='Gender', ylabel='count'>
```

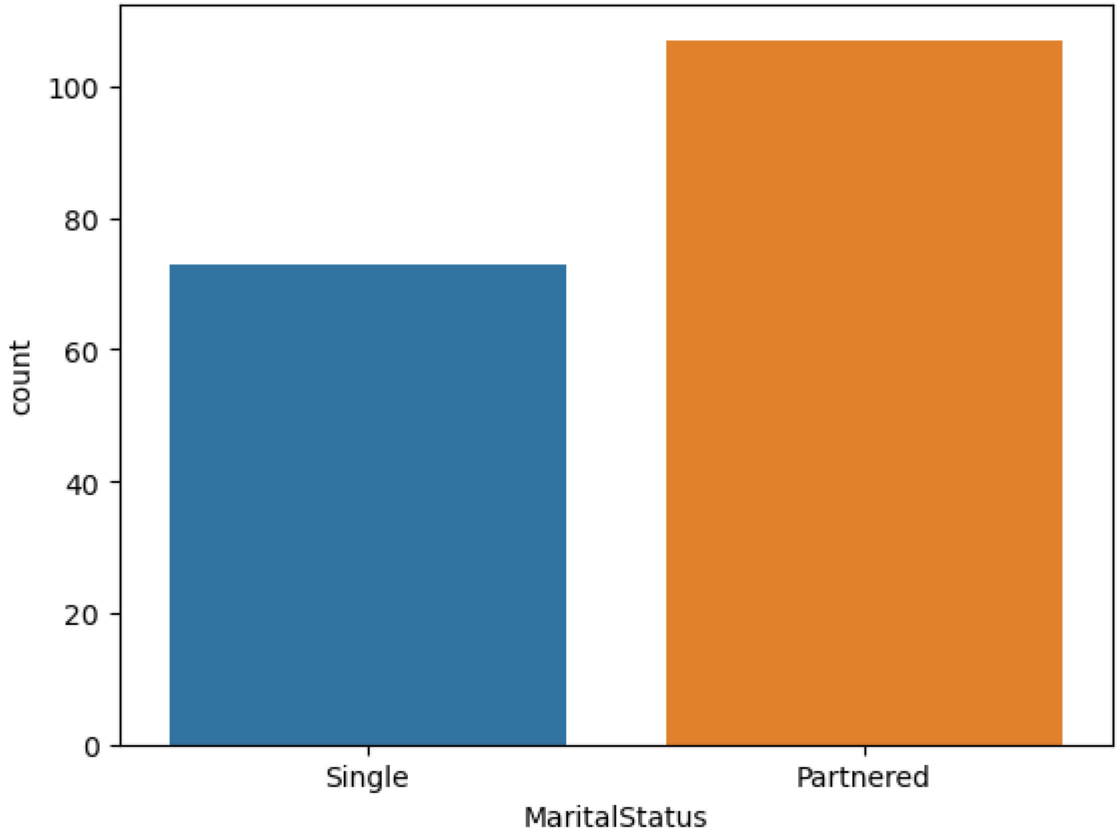


In [40]:
```
palette_color = sns.color_palette('bright')
keys=['Partnered','Single']
plt.pie(df.MaritalStatus.value_counts(), labels=keys, colors=palette_color, autopct='%
plt.show()
```



In [41]:
```
sns.countplot(x=df['MaritalStatus'])
```

Out[41]: <AxesSubplot:xlabel='MaritalStatus', ylabel='count'>

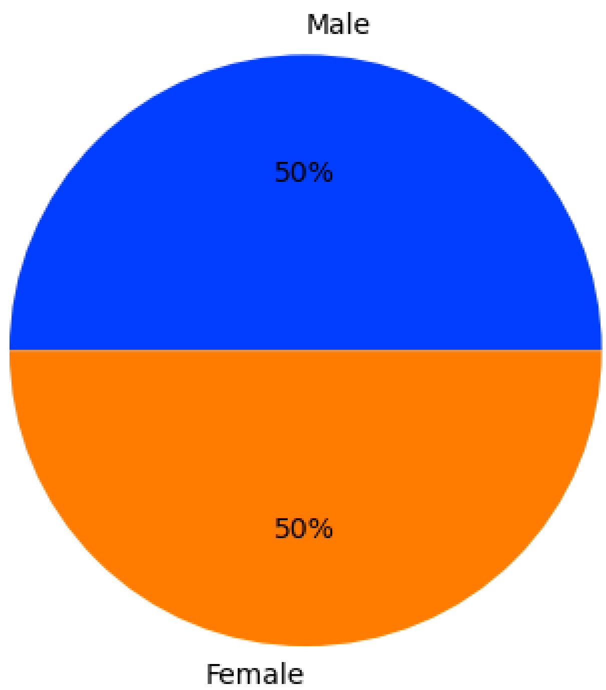


Bivariate Analysis

In [42]:
```
KP281.Gender.value_counts()
```

Out[42]:
```
Male      40
Female    40
Name: Gender, dtype: int64
```

In [43]:
```
palette_color = sns.color_palette('bright')
keys=['Male','Female']
plt.pie(KP281.Gender.value_counts(), labels=keys, colors=palette_color, autopct='%.0f%
plt.show()
```

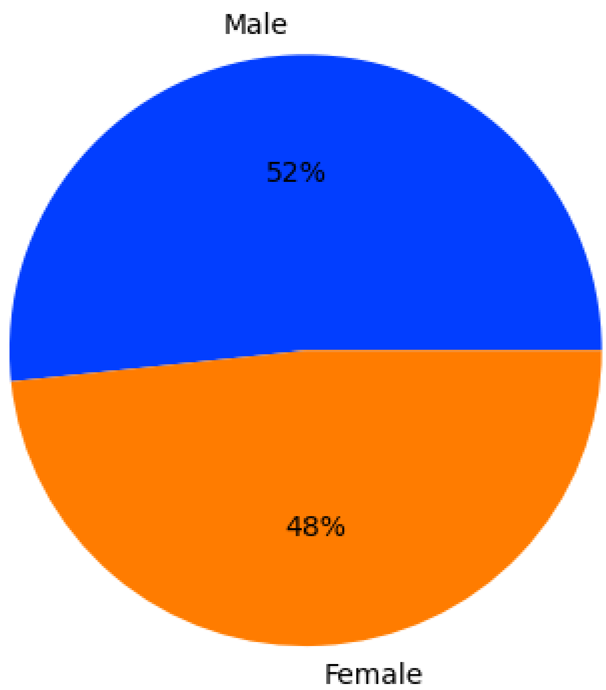


From the above, we observe that equal number of males and females use KP281.

In [44]:
```
KP481.Gender.value_counts()
```

Out[44]:
```
Male      31
Female    29
Name: Gender, dtype: int64
```

In [45]:
```
palette_color = sns.color_palette('bright')
keys=['Male','Female']
plt.pie(KP481.Gender.value_counts(), labels=keys, colors=palette_color, autopct='%.0f%
plt.show()
```



From the above, we observe that the number of males who use KP481 is slightly higher than the number of females.

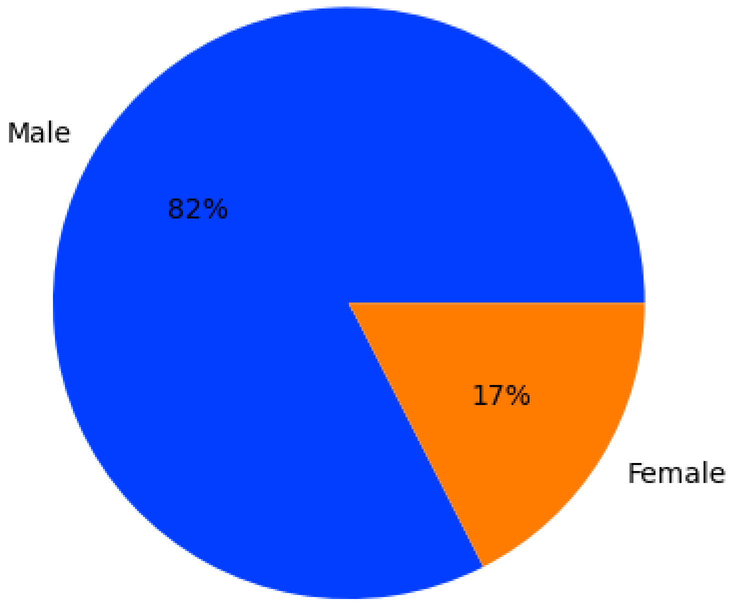
In [46]:
```
KP781.Gender.value_counts()
```

Out[46]:
```
Male      33
Female     7
Name: Gender, dtype: int64
```

Questionnaire: 7. What proportion of women have bought the KP781 treadmill? Give the reason behind your answer.

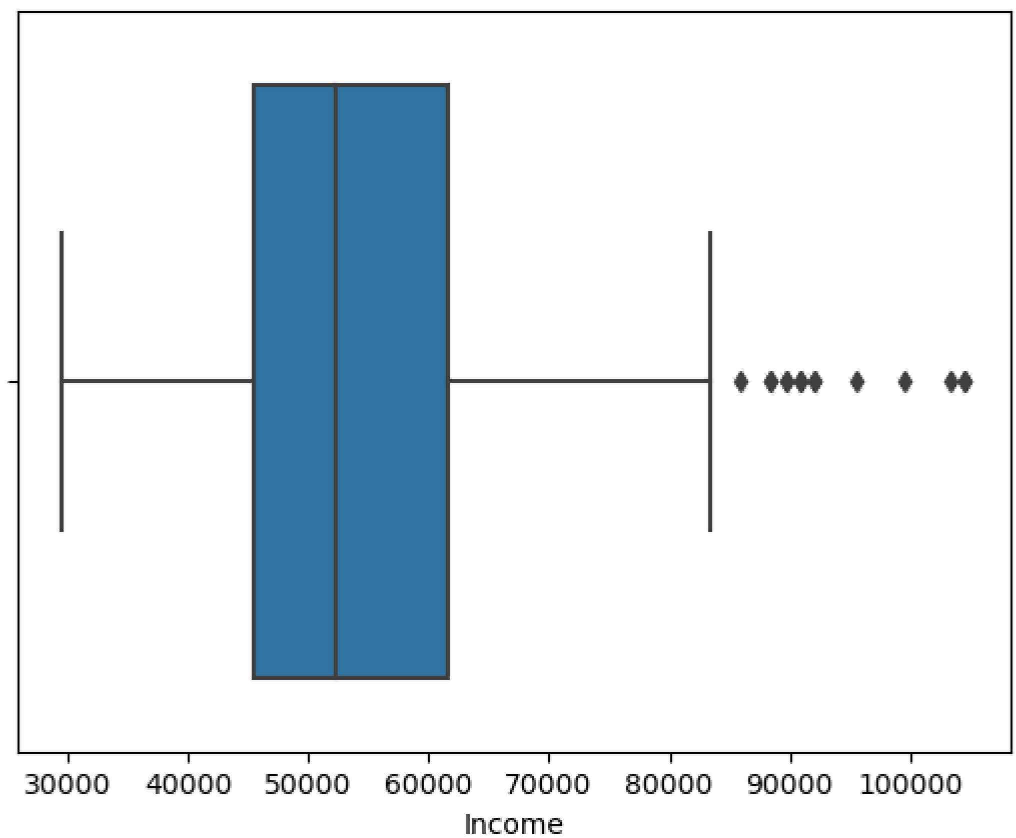
Only 17% of women have bought the KP781 treadmill. The same is visible from the pie-chart below. One of the reasons behind this could be lack of awareness among women regarding the benefits of the KP781 treadmill. Income could also be one of the reasons for influencing the decision of women not to purchase KP781 treadmill as it is costly compared to the other treadmills. We can see from the boxplots below that the average income of females is less than males.

In [47]:
```
palette_color = sns.color_palette('bright')
keys=['Male','Female']
plt.pie(KP781.Gender.value_counts(), labels=keys, colors=palette_color, autopct='%.0f%
plt.show()
```



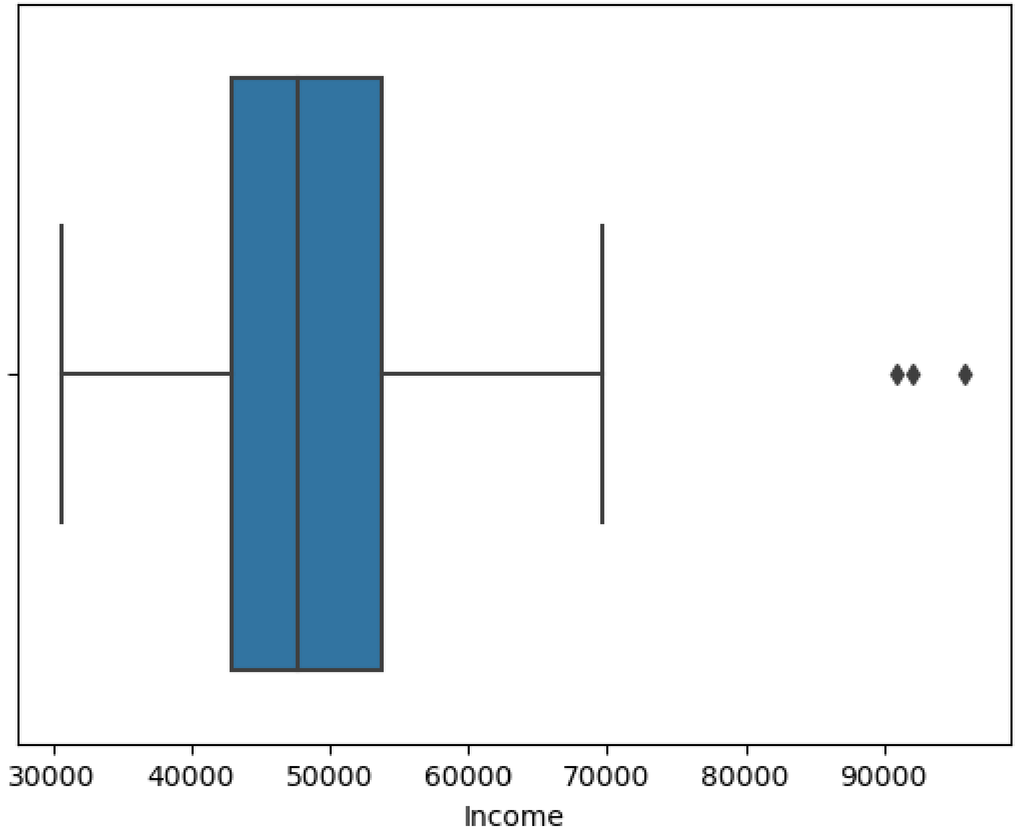
In [48]:
```
sns.boxplot(data= Male, x="Income", orient="h")
```

Out[48]: <AxesSubplot:xlabel='Income'>



In [49]:
```
sns.boxplot(data= Female, x="Income", orient="h")
```

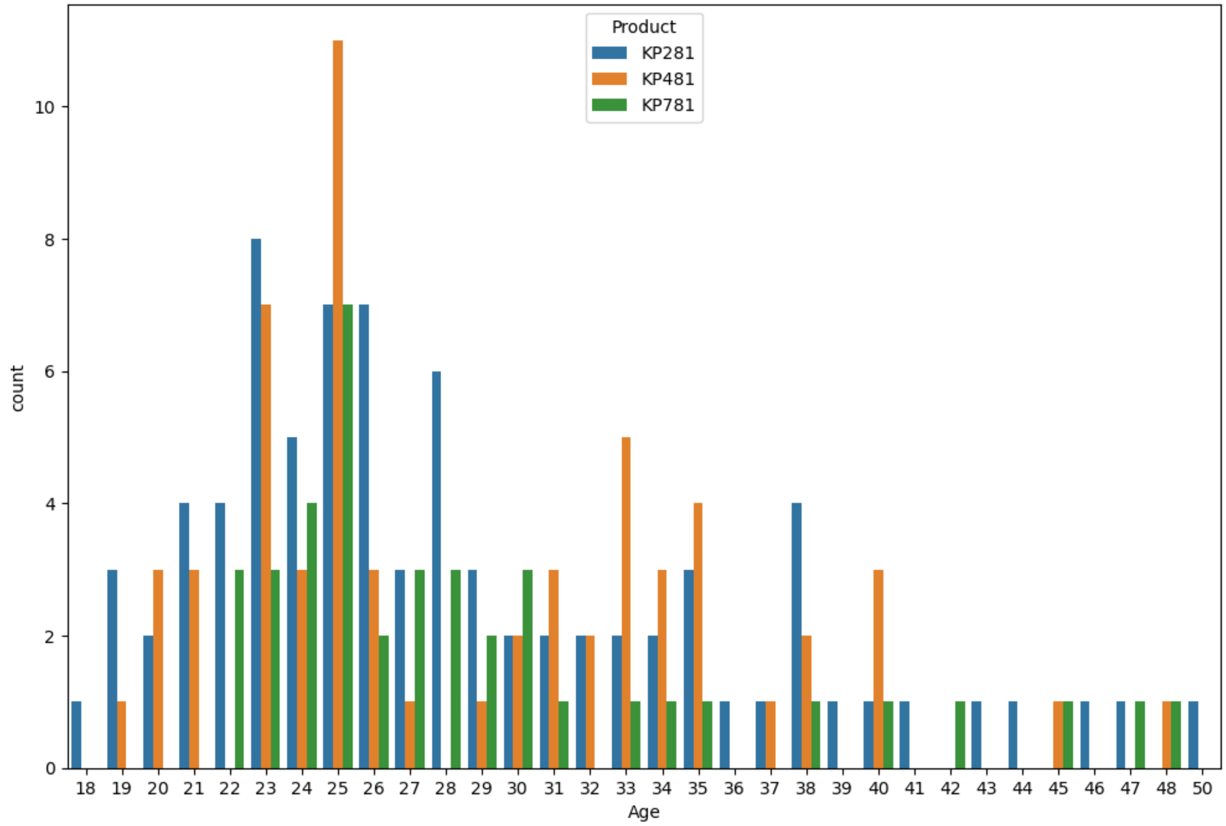
Out[49]: <AxesSubplot:xlabel='Income'>



From the above, we can see that very few females use KP781 as compared to males.

In [50]:
```
plt.figure(figsize=(12,8))
sns.countplot(x='Age',hue='Product', data=df)
```

Out[50]:
```
<AxesSubplot:xlabel='Age', ylabel='count'>
```



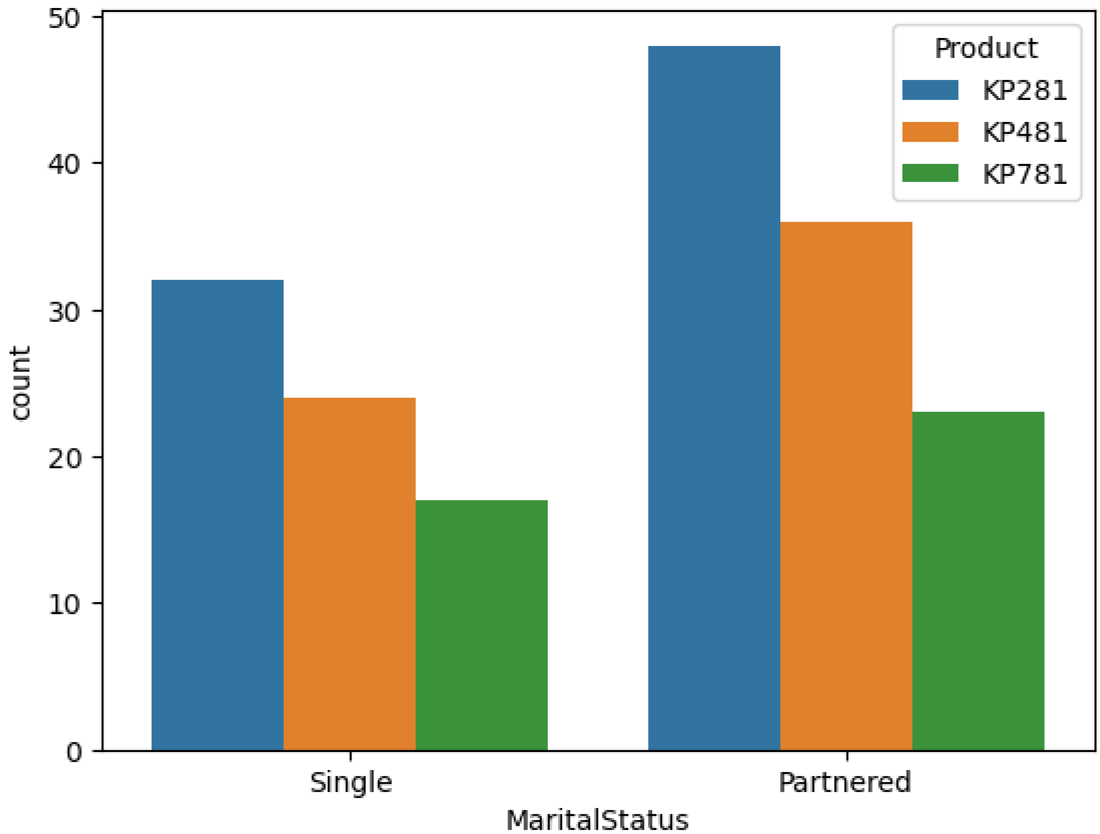
From the above, we observe that for KP281 treadmill, maximum number of users are of age 23 years. For KP481 and KP781 treadmills, maximum number of people are of age 25 years.

Questionnaire: 5. Marital Status implies no significant information on the usages of different treadmills. (T/F)

Comparing the single and partnered users, there is similar difference in use of different treadmills. Even though from the countplot, it appears that the number of partnered people use treadmills more, it would be wrong to ignore that the dataset has more number of partneres users as compared to single people. Hence, we can say that Marital Status implies no significant information on the usages of different treadmills.

In [51]:
```
sns.countplot(x='MaritalStatus',hue='Product', data=df)
```

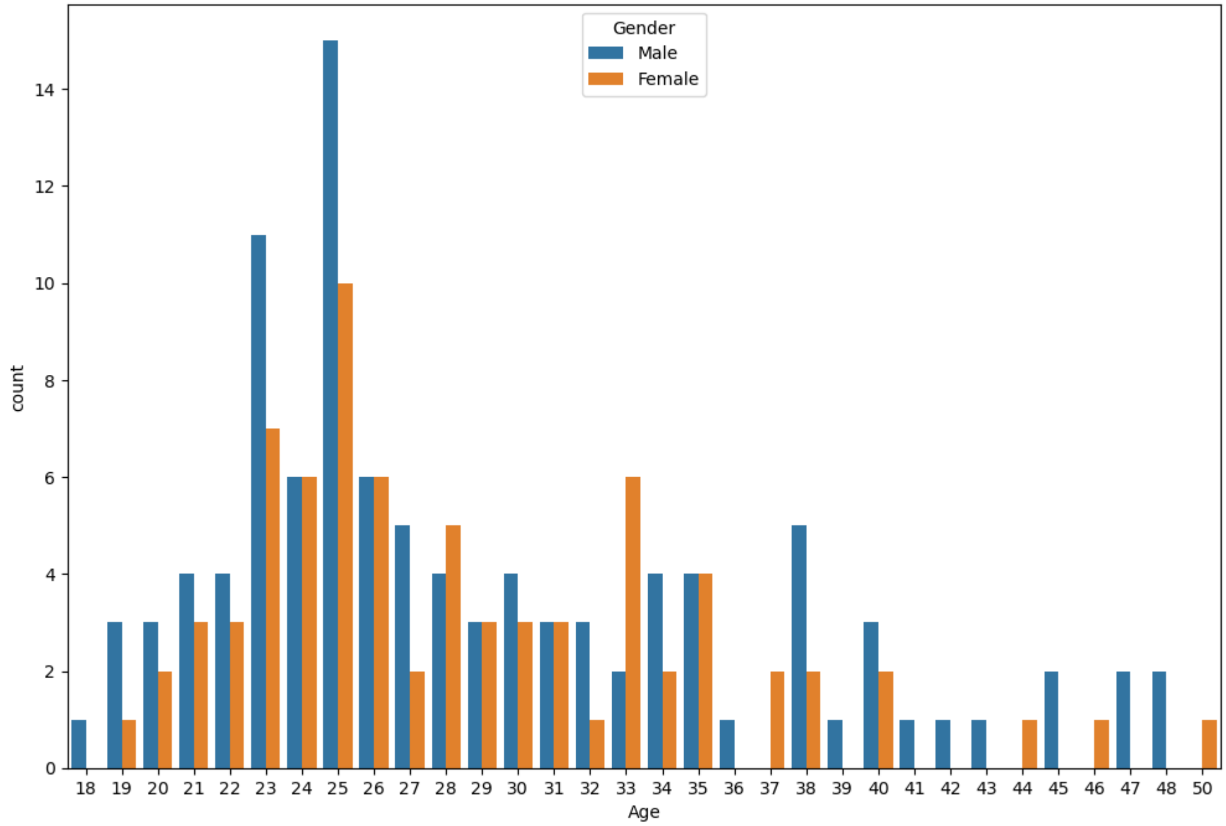
Out[51]:
```
<AxesSubplot:xlabel='MaritalStatus', ylabel='count'>
```



Questionnaire: 2. Describe the Age & Gender distribution of all the customers.

In [52]:
```
plt.figure(figsize=(12,8))
sns.countplot(x='Age',hue='Gender', data=df)
```

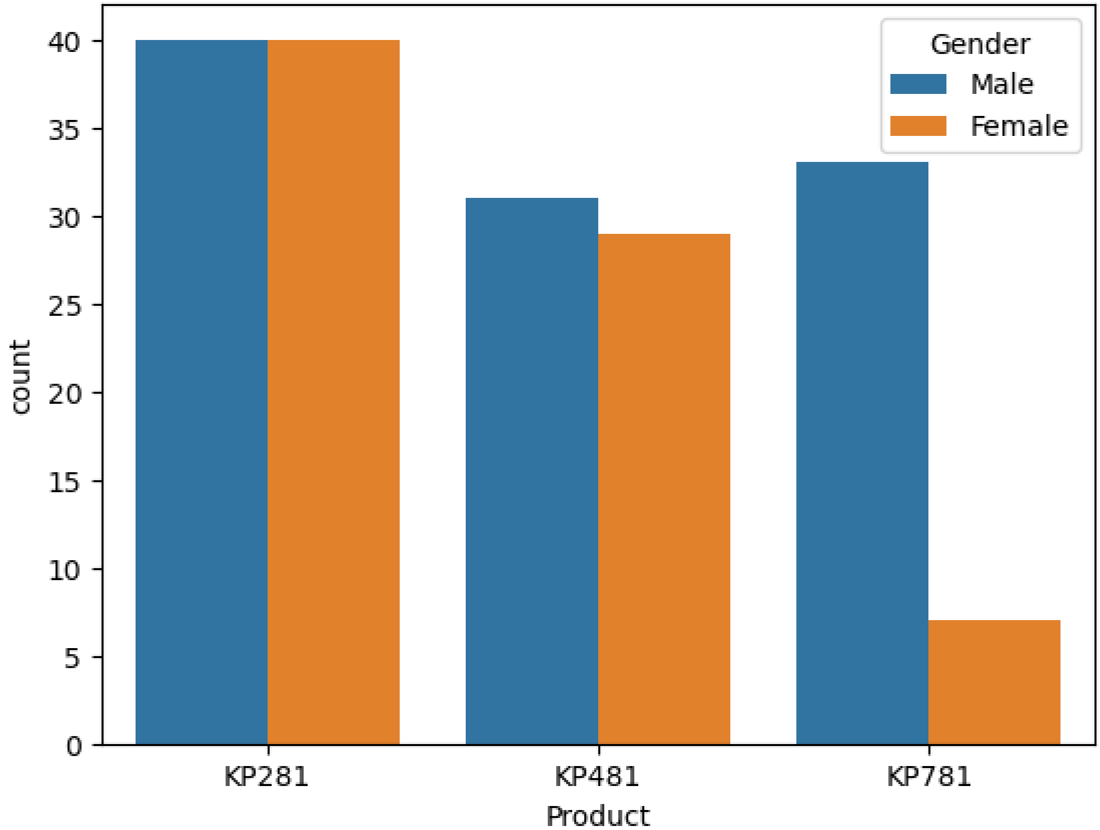
Out[52]:
```
<AxesSubplot:xlabel='Age', ylabel='count'>
```



From the above, we observe that males and females are not equally spread out over different ages. Most of the females are in the age group 20-35 years.

In [53]:
```
sns.countplot(x='Product',hue='Gender', data=df)
```

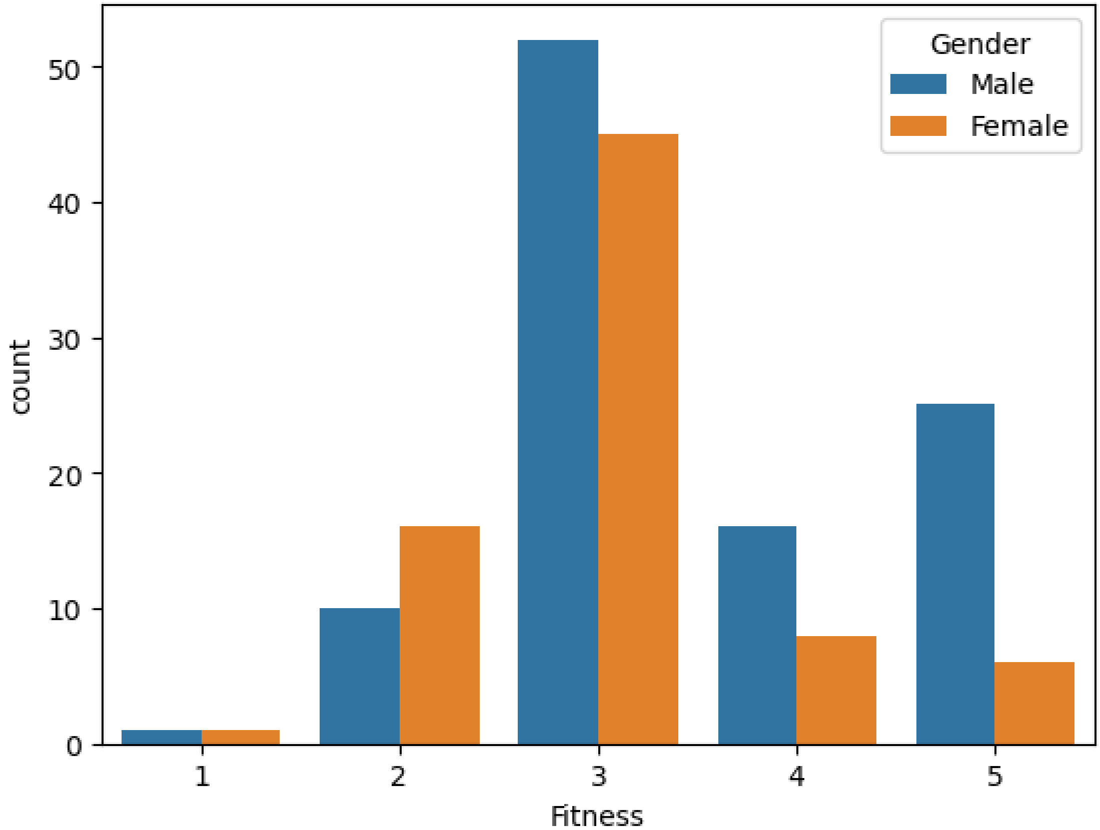
Out[53]: <AxesSubplot:xlabel='Product', ylabel='count'>



From the above, we observe that there is not much difference in number of males and females using KP281 and KP481 treadmills. However, very few females use KP781 treadmill as compared to males.

In [54]:
```
sns.countplot(x='Fitness',hue='Gender', data=df)
```

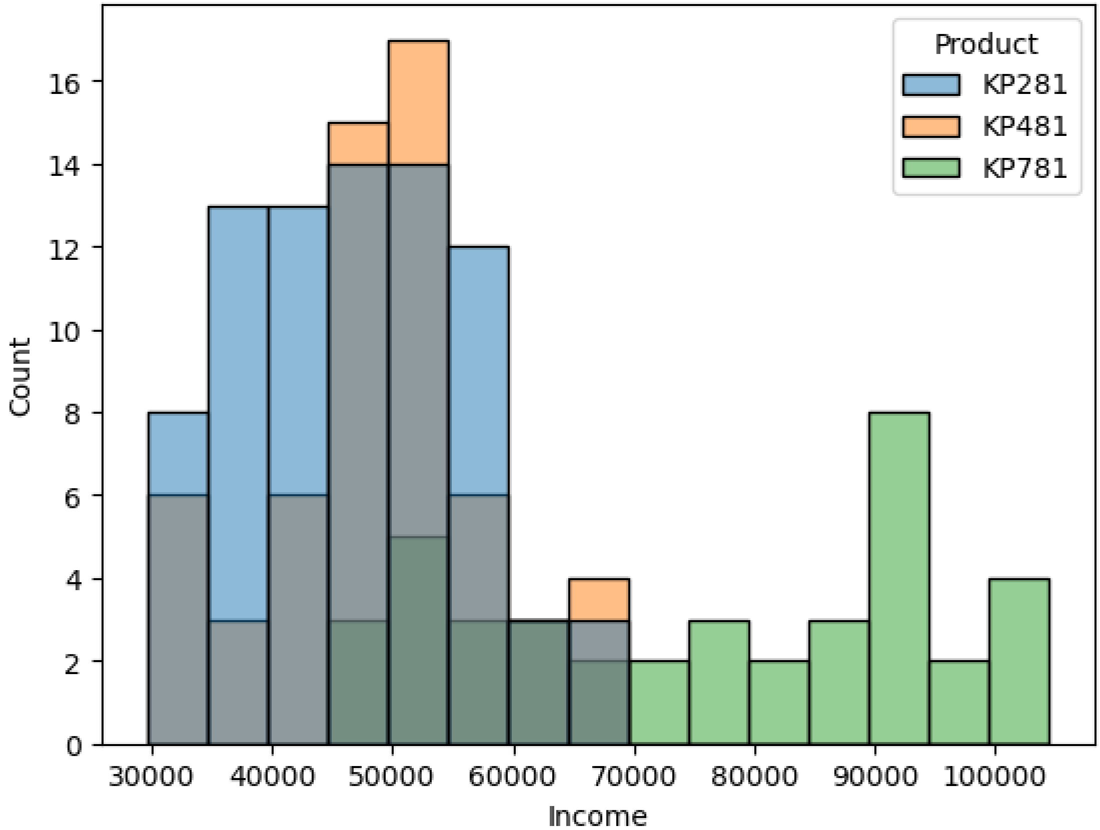
Out[54]: <AxesSubplot:xlabel='Fitness', ylabel='count'>



From the above, we observe that maximum number of males and females consider themselves at point 3 on the fitness scale.

In [55]:
```
sns.histplot(x='Income',hue='Product',data=df)
```

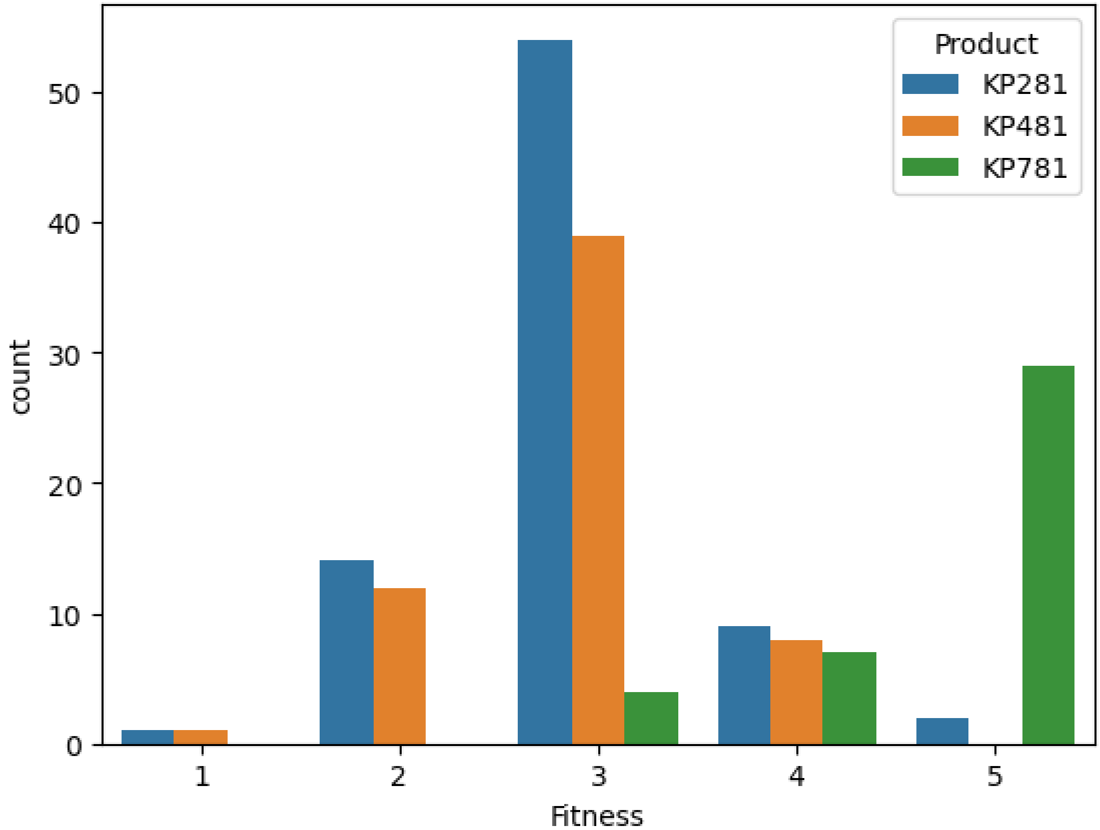
Out[55]:
```
<AxesSubplot:xlabel='Income', ylabel='Count'>
```



From the above, we observe that users below income range of Rs.60,000 prefer to use KP281 and KP481 treadmills and people with income range of Rs.70,000 and above only use KP781 treadmill.

In [56]:
```
sns.countplot(x='Fitness',hue='Product', data=df)
```

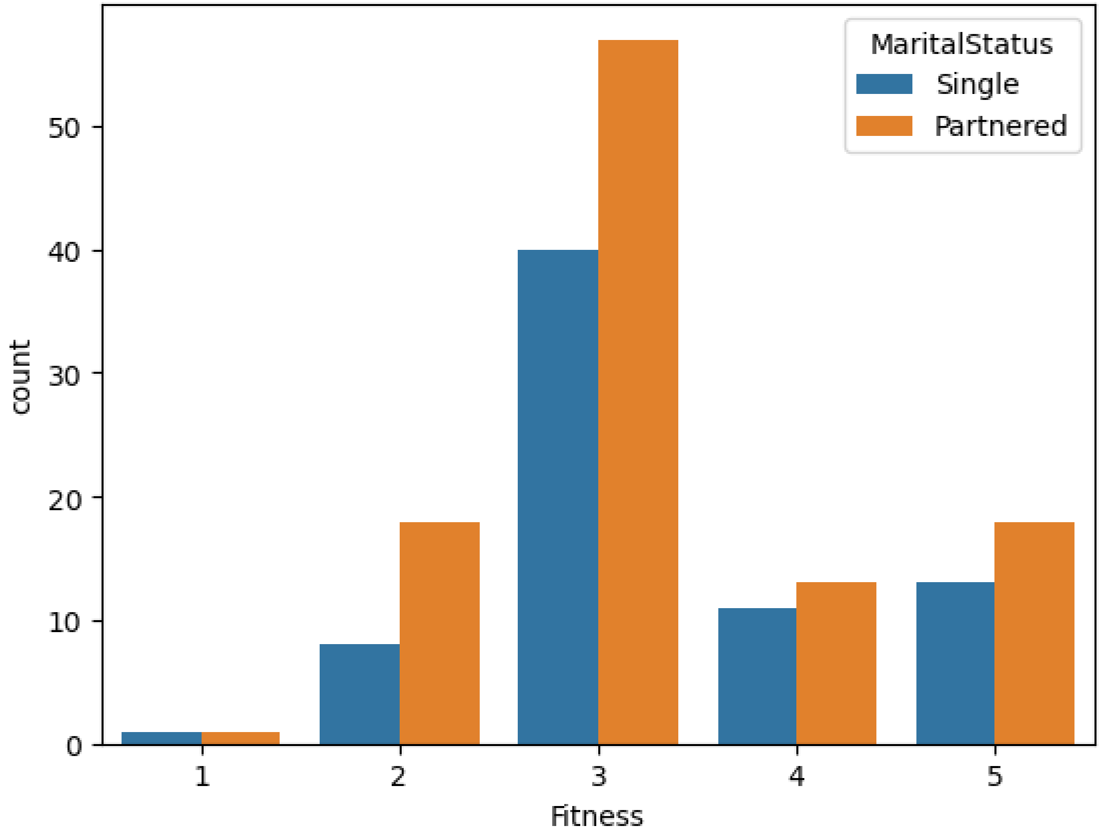
Out[56]:
```
<AxesSubplot:xlabel='Fitness', ylabel='count'>
```



From the above, we observe that more number of people who use KP781 treadmill feel that they are at point 5 on the fitness scale as compared to people who use other treadmills.

In [57]:
```
sns.countplot(x='Fitness',hue='MaritalStatus', data=df)
```

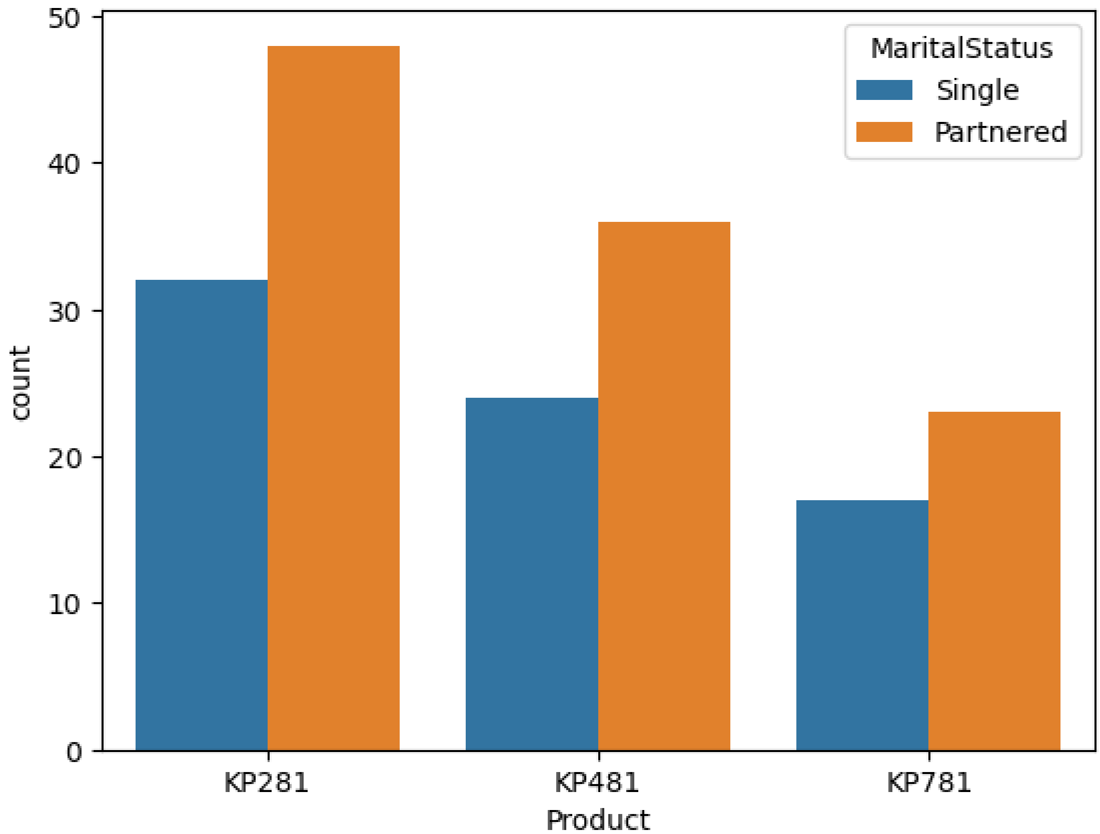
Out[57]:
```
<AxesSubplot:xlabel='Fitness', ylabel='count'>
```



From the above, we observe that more users who have a Partner consider themselves fit as compared to the ones who are Single. However, since there are more number of married people in the data group as compared to single people, we can say that marital status does not have much effect on fitness.

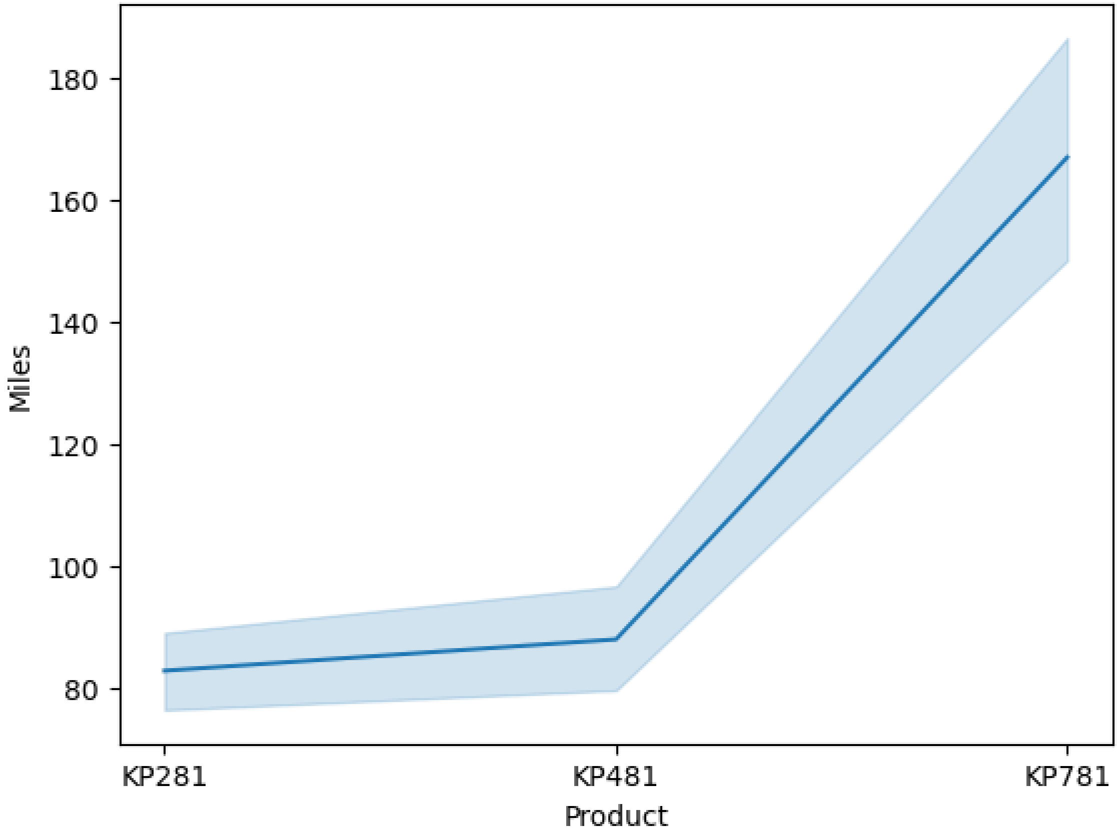
In [58]:
```
sns.countplot(x='Product',hue='MaritalStatus', data=df)
```

Out[58]:
```
<AxesSubplot:xlabel='Product', ylabel='count'>
```



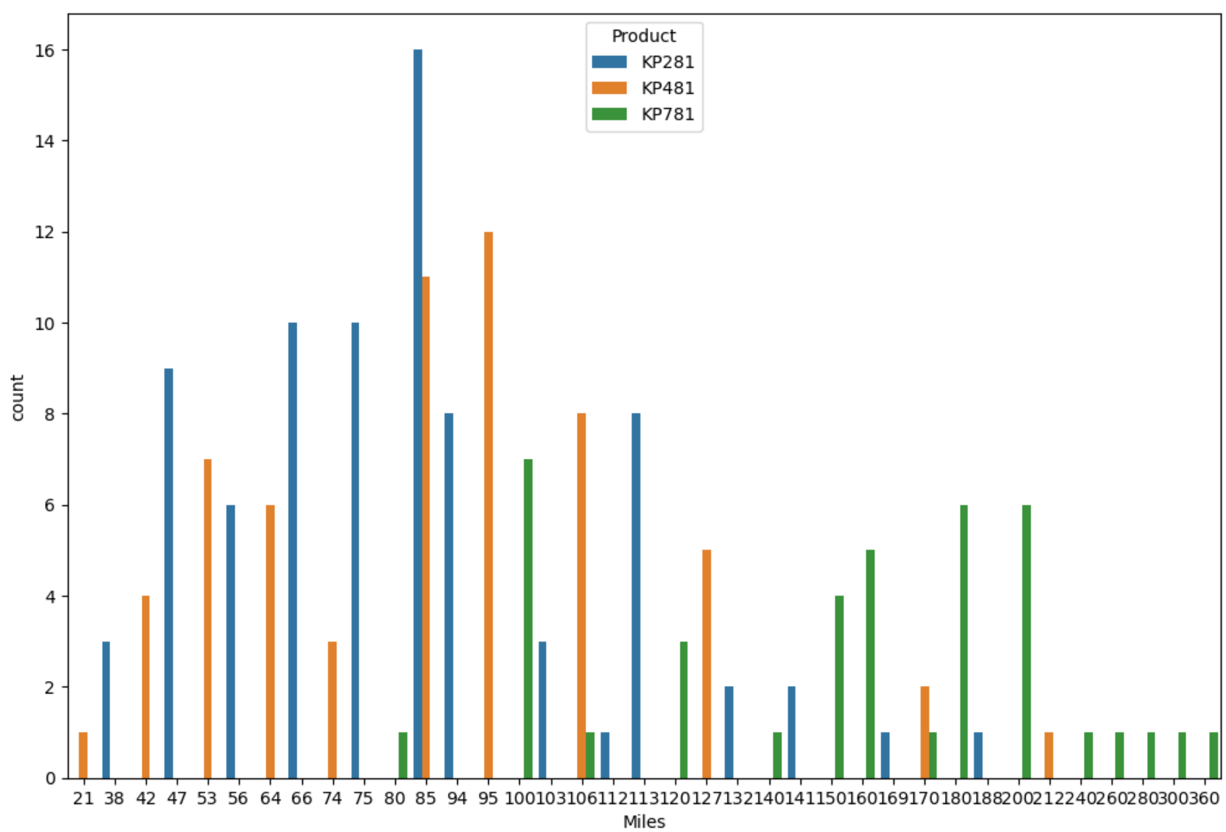
In [59]:
```
sns.lineplot(data=df,
             x="Product",
             y="Miles")
```

Out[59]: <AxesSubplot:xlabel='Product', ylabel='Miles'>



In [60]:
```
plt.figure(figsize=(12,8))
sns.countplot(x='Miles',hue='Product', data=df)
```

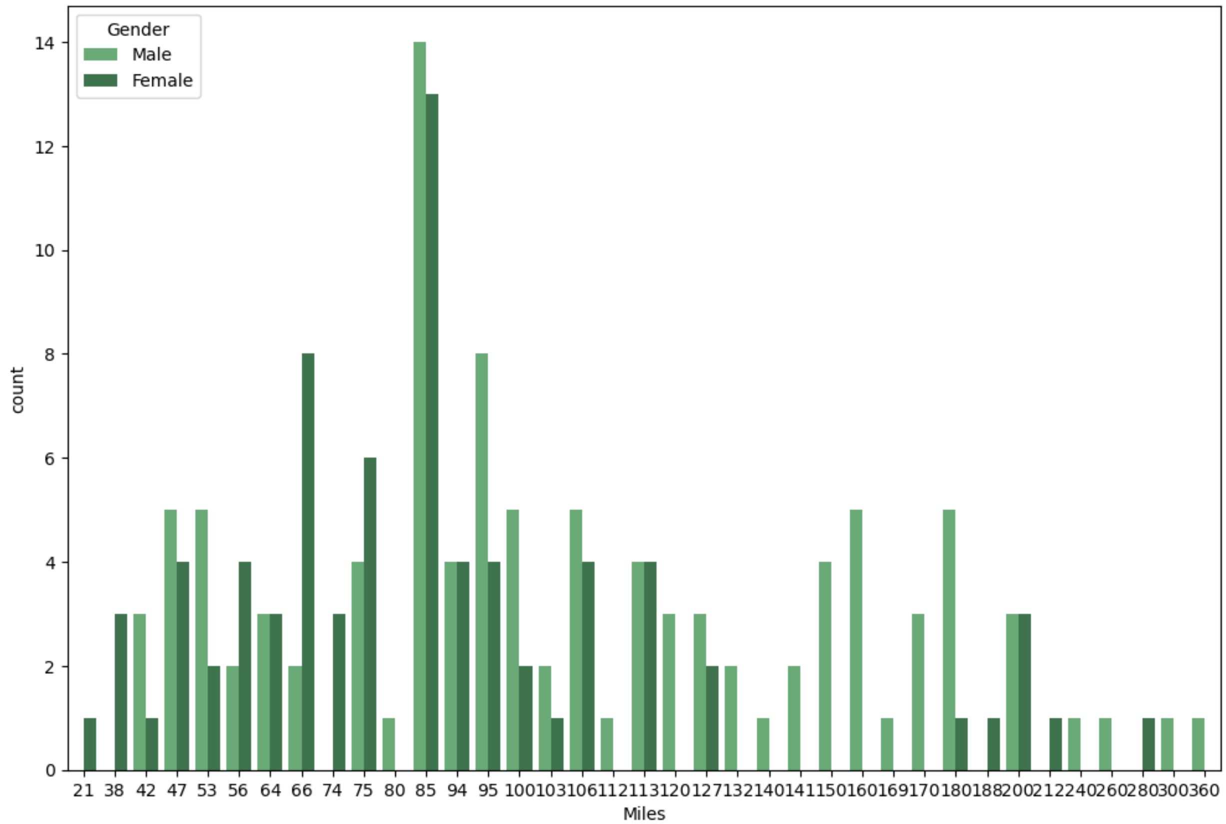
Out[60]: <AxesSubplot:xlabel='Miles', ylabel='count'>



From the above, we observe that people using KP781 treadmill run more number of miles as compared to those using other treadmills.

In [61]:
```
plt.figure(figsize=(12,8))
sns.countplot(x='Miles', hue= 'Gender', data=df, palette="Greens_d")
```

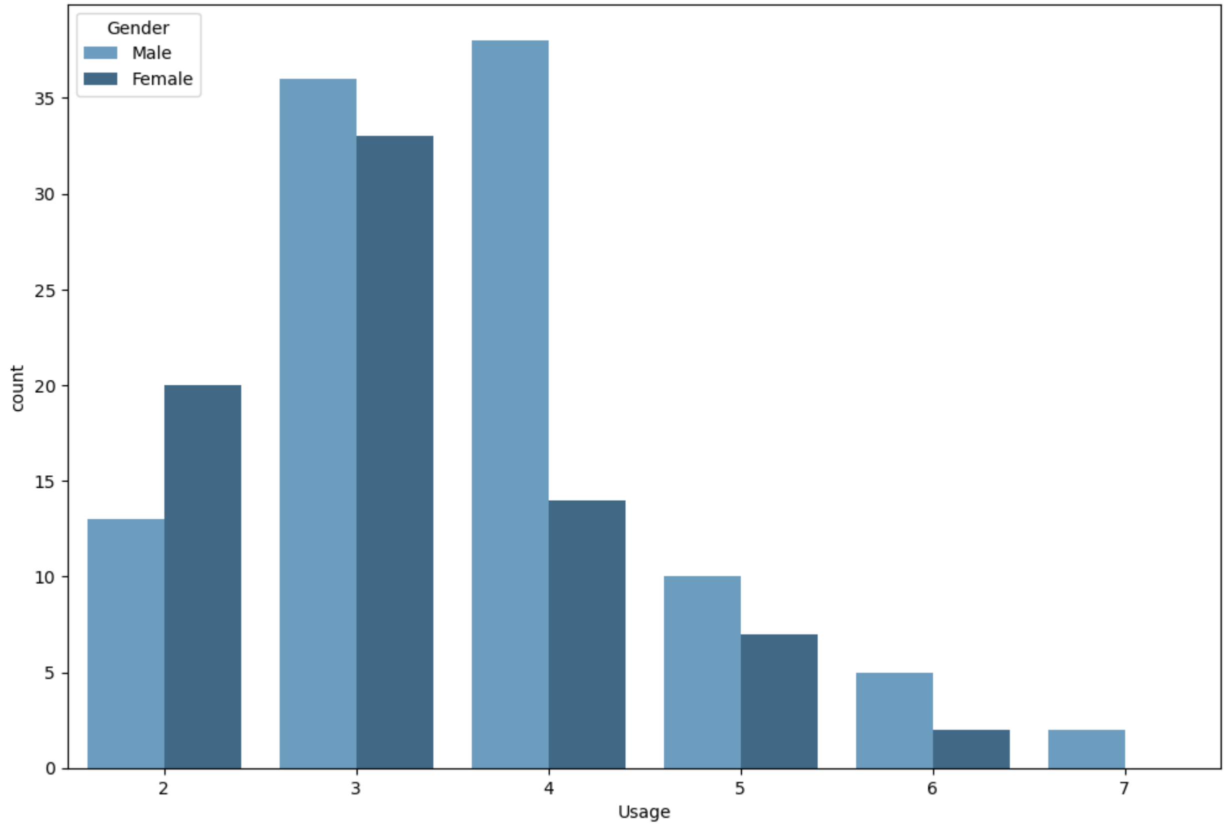
Out[61]: <AxesSubplot:xlabel='Miles', ylabel='count'>



From the above, we observe that maximum number of males and females run 85 miles each week.

In [62]:
```
plt.figure(figsize=(12,8))
sns.countplot(x='Usage', hue= 'Gender', data=df, palette="Blues_d")
```

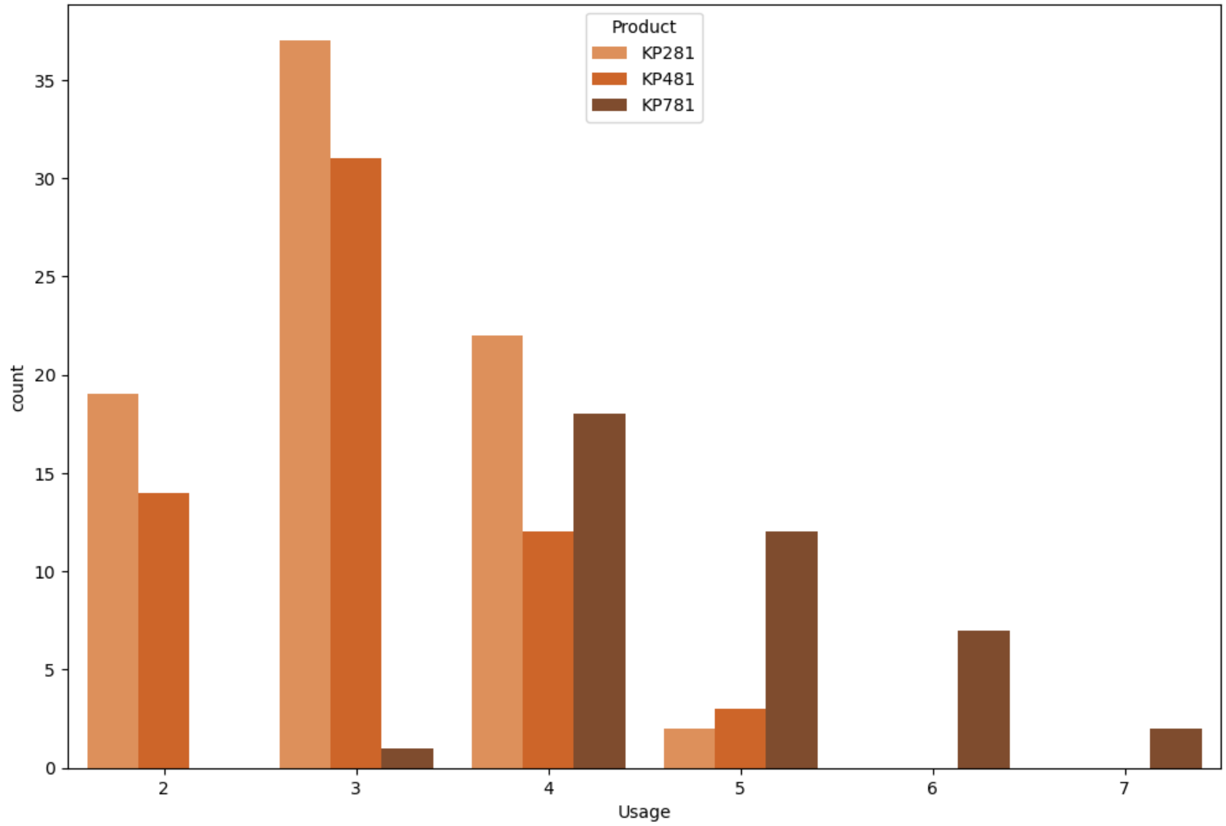
Out[62]: <AxesSubplot:xlabel='Usage', ylabel='count'>



From the above, we observe that males tend to use the treadmill for 4 hours as compared to females.

In [63]:
```
plt.figure(figsize=(12,8))
sns.countplot(x='Usage', hue= 'Product', data=df, palette="Oranges_d")
```

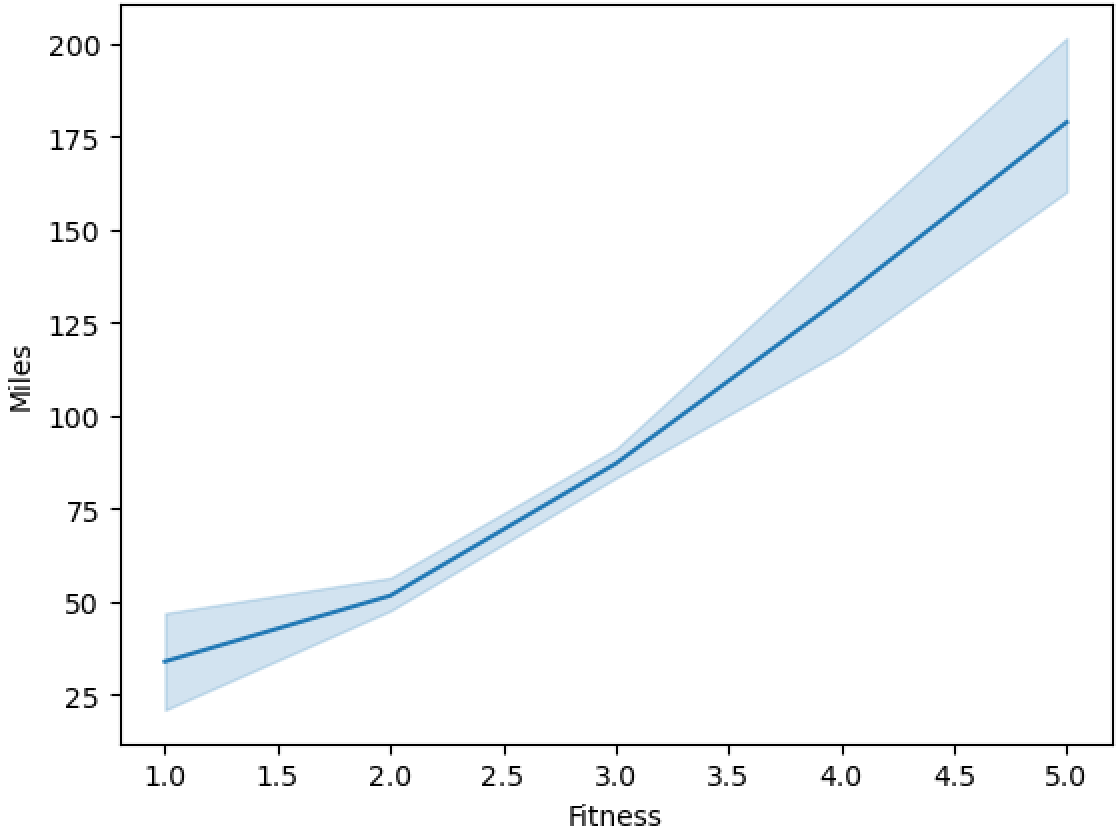
Out[63]: <AxesSubplot:xlabel='Usage', ylabel='count'>



From the above, we observe that people with KP781 treadmill tend to put in more hours as compared to other treadmills.

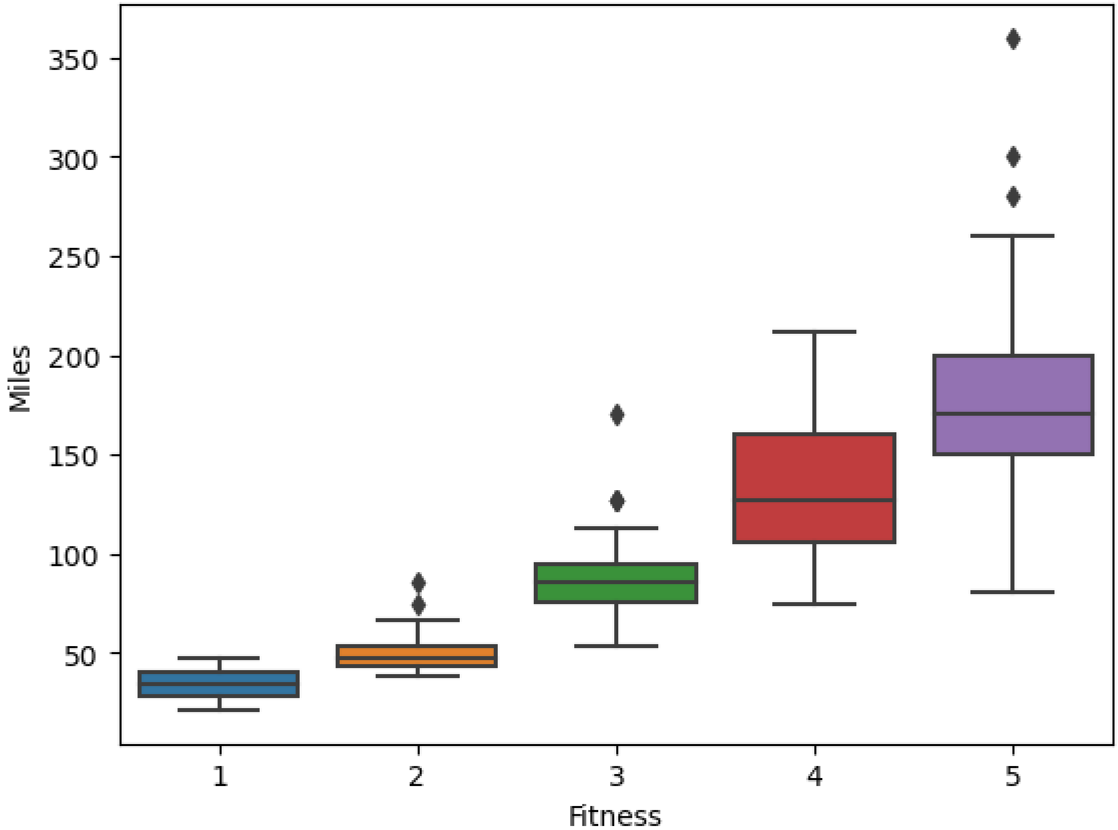
In [64]:
```
sns.lineplot(data=df,
             x="Fitness",
             y="Miles")
```

Out[64]: <AxesSubplot:xlabel='Fitness', ylabel='Miles'>



In [65]:
```
sns.boxplot(data=df,
            x="Fitness",
            y="Miles")
```

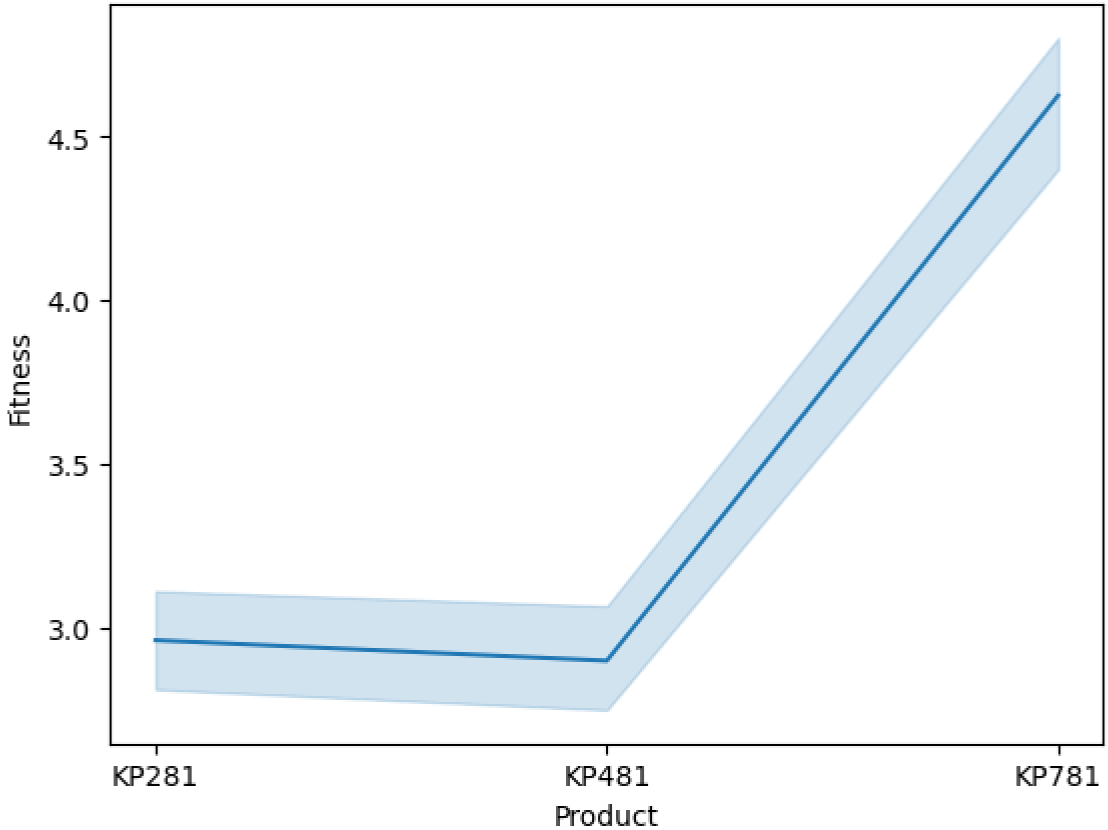
Out[65]:
```
<AxesSubplot:xlabel='Fitness', ylabel='Miles'>
```



From the above, we observe that people who run more number of miles consider themselves more fit.

In [66]:
```
sns.lineplot(data=df,
             x="Product",
             y="Fitness")
```

Out[66]:
```
<AxesSubplot:xlabel='Product', ylabel='Fitness'>
```

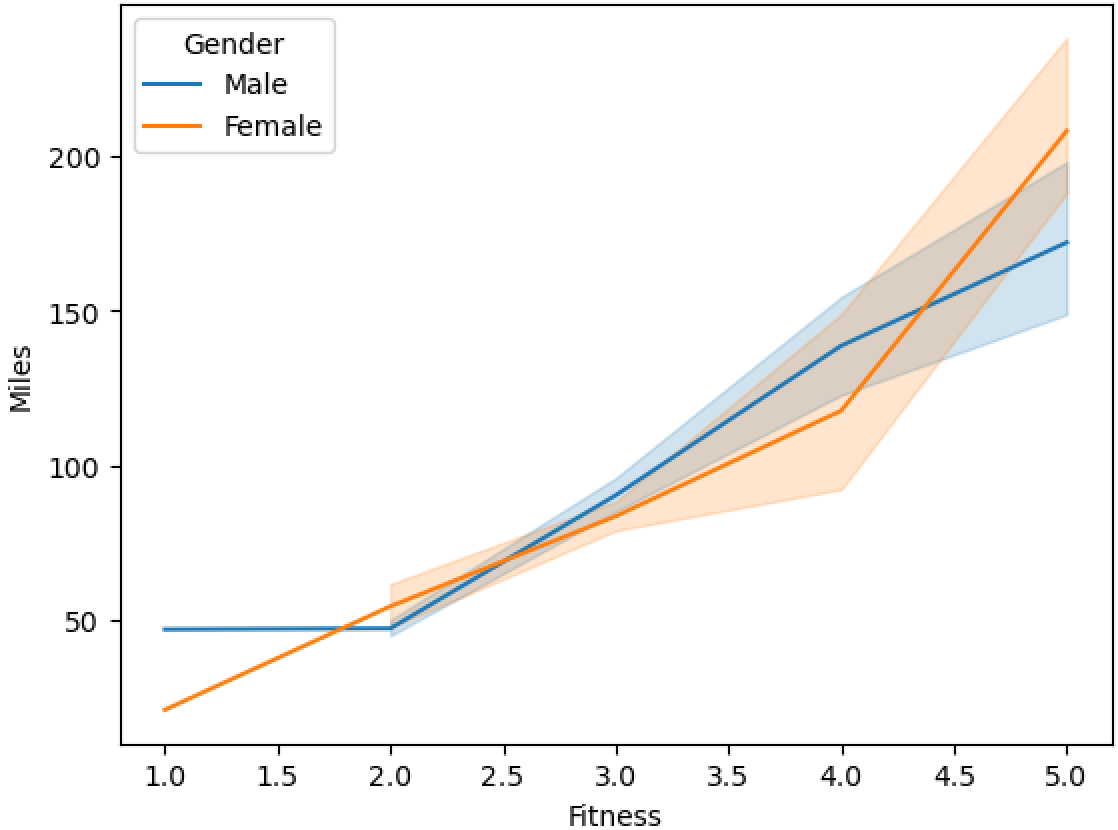


From the above, we observe that people who use KP781 consider themselves more fit as compared to others.

Multivariate

In [67]:
```
sns.lineplot(data=df,
             x="Fitness",
             y="Miles",
             hue="Gender")
```

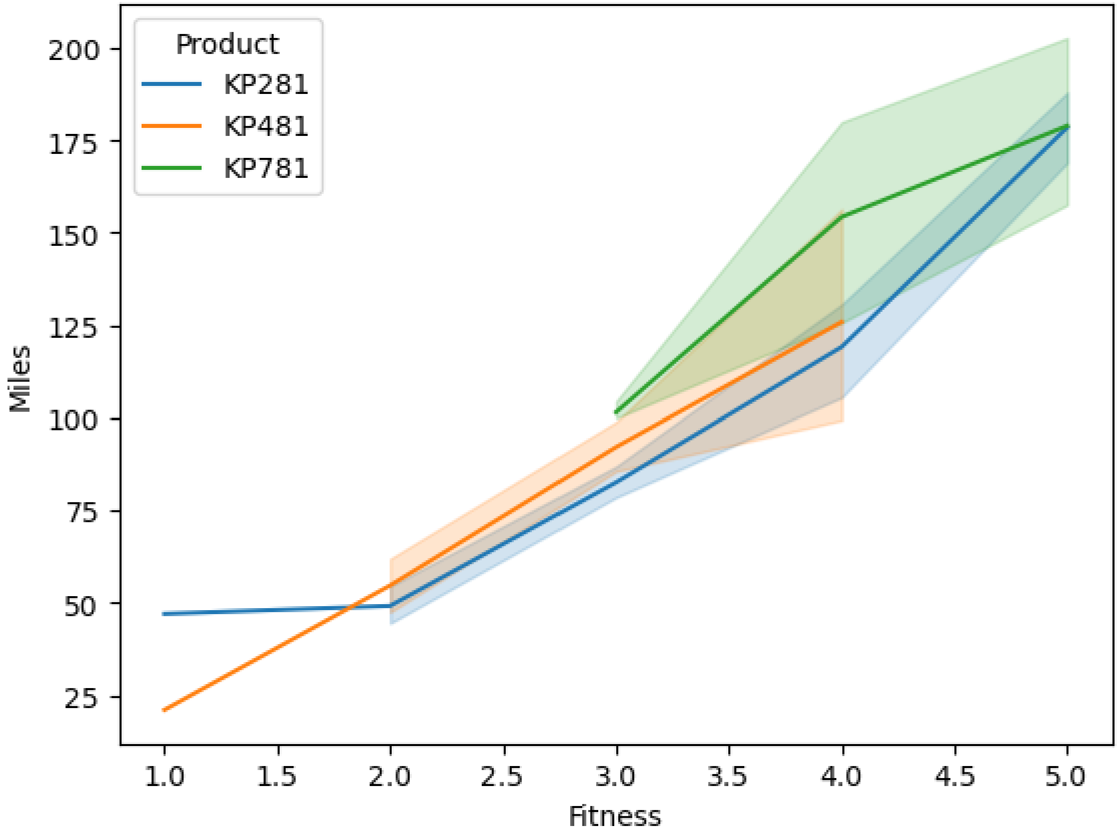
Out[67]: <AxesSubplot:xlabel='Fitness', ylabel='Miles'>



From the above, we observe that even though males run more miles, they feel less fit as compared to females.

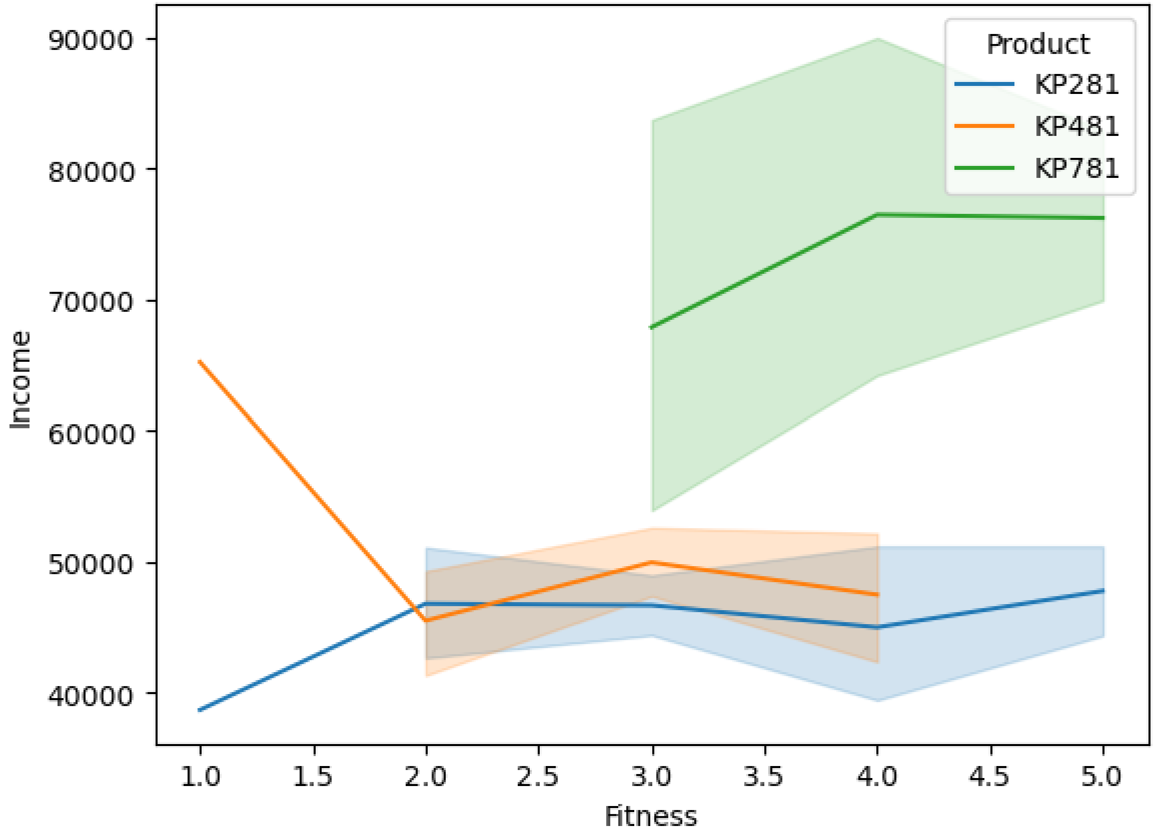
In [68]:
```
sns.lineplot(data=df,
             x="Fitness",
             y="Miles",
             hue="Product")
```

Out[68]:
```
<AxesSubplot:xlabel='Fitness', ylabel='Miles'>
```



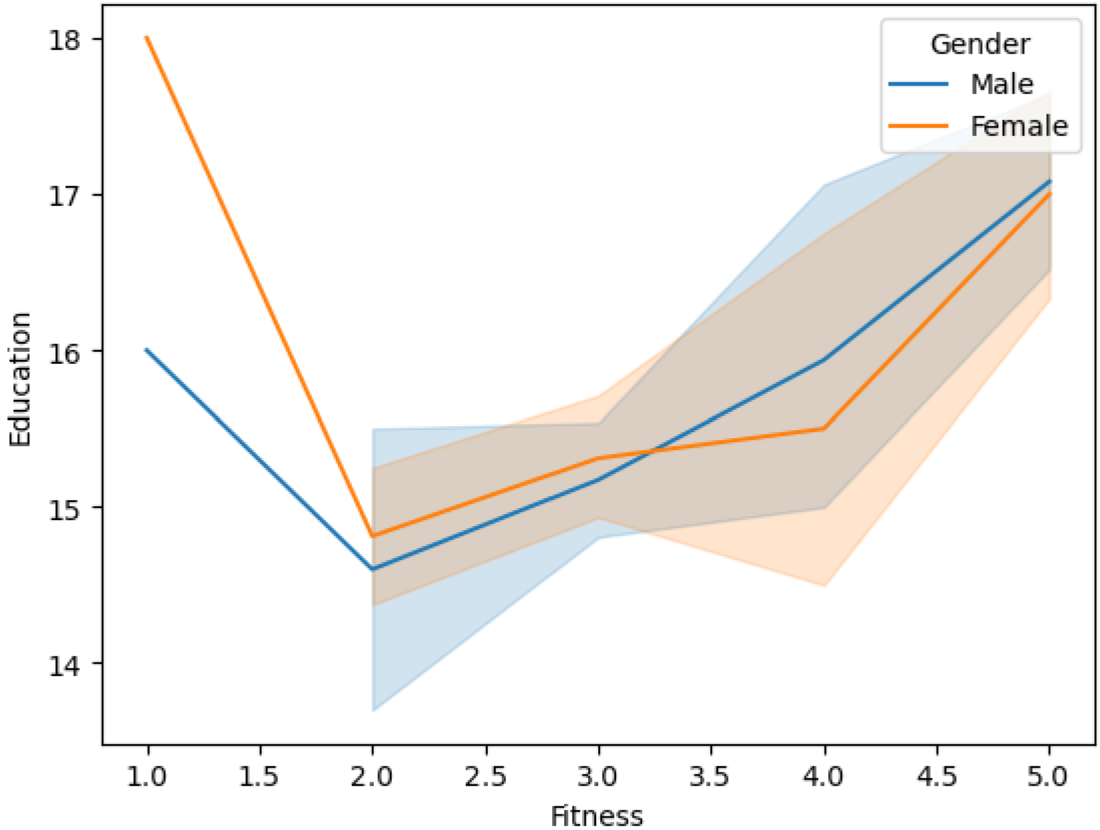
In [69]:
```
sns.lineplot(data=df,
             x="Fitness",
             y="Income",
             hue="Product")
```

Out[69]: <AxesSubplot:xlabel='Fitness', ylabel='Income'>



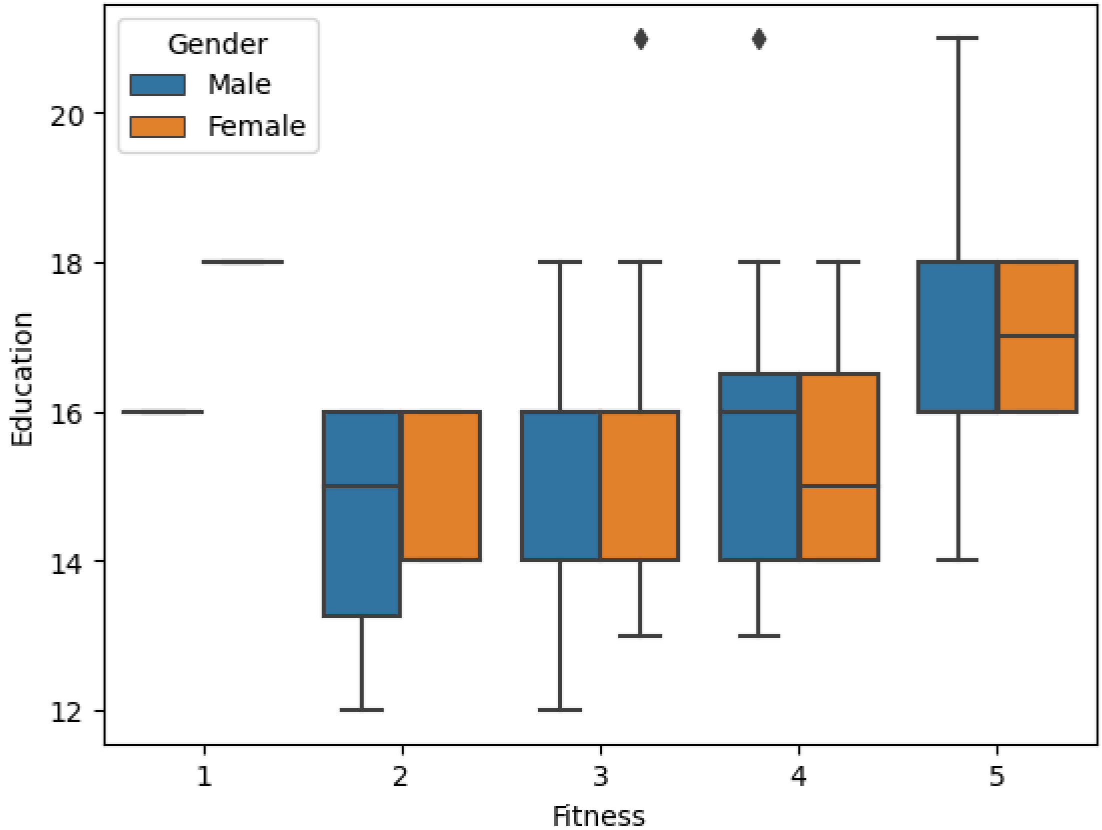
In [70]:
```
sns.lineplot(data=df,
             x="Fitness",
             y="Education",
             hue="Gender")
```

Out[70]:
```
<AxesSubplot:xlabel='Fitness', ylabel='Education'>
```



In [71]:
```
sns.boxplot(data=df,
            x="Fitness",
            y="Education",
            hue="Gender")
```

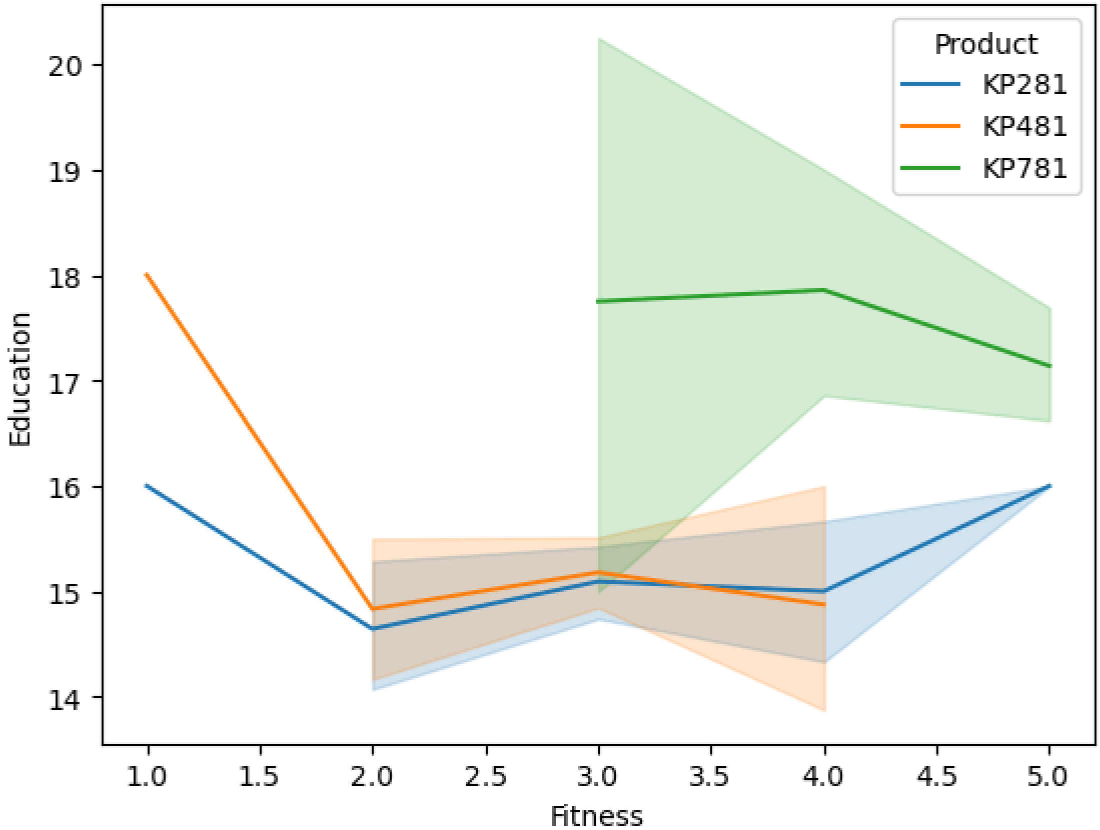
Out[71]:
```
<AxesSubplot:xlabel='Fitness', ylabel='Education'>
```



From the above, we observe that people with higher years of education tend to be more fit and gender does not have much effect in this case.

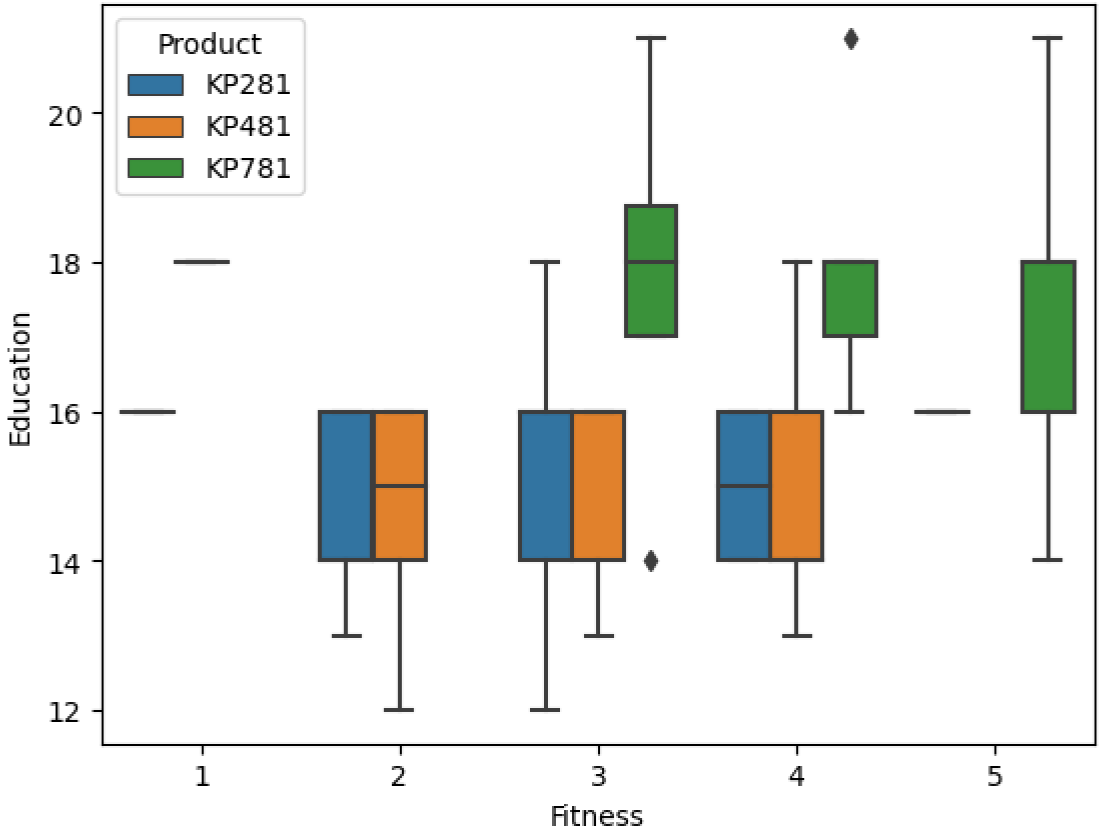
In [72]:
```
sns.lineplot(data=df,
             x="Fitness",
             y="Education",
             hue="Product")
```

Out[72]: <AxesSubplot:xlabel='Fitness', ylabel='Education'>



In [73]:
```
sns.boxplot(data=df,
            x="Fitness",
            y="Education",
            hue='Product')
```

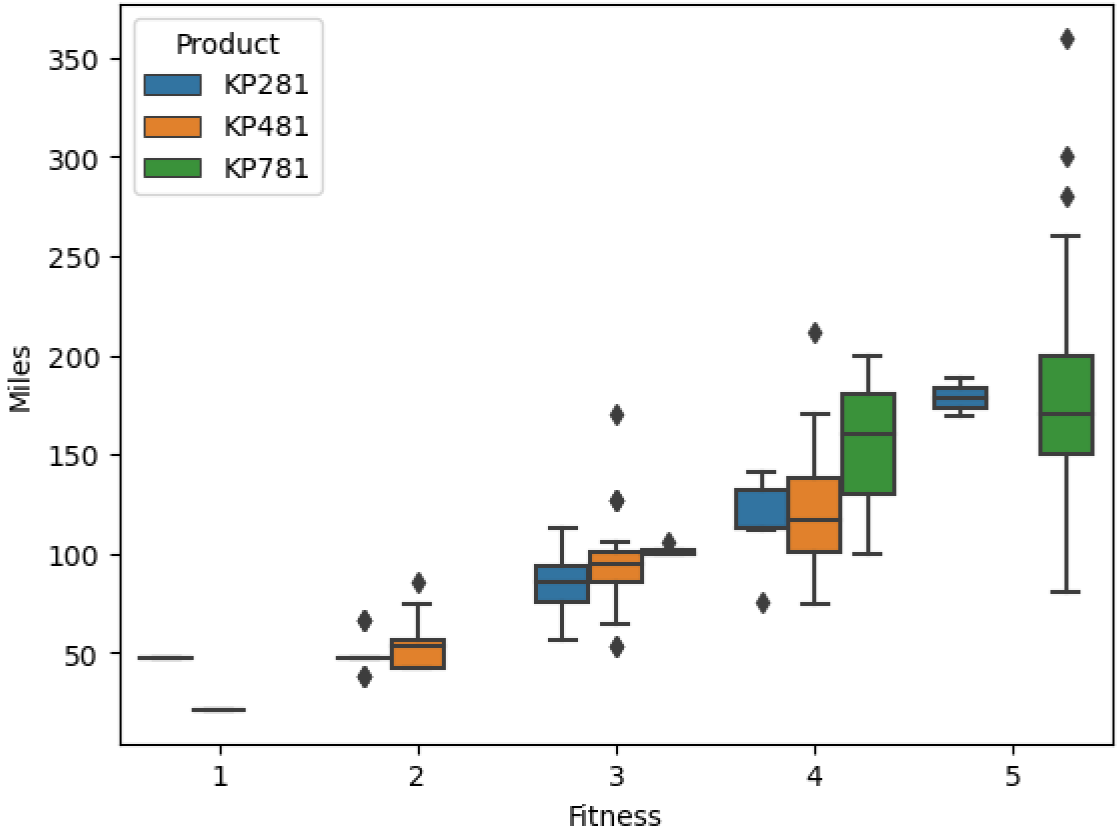
Out[73]:
```
<AxesSubplot:xlabel='Fitness', ylabel='Education'>
```



From the above, we observe that people with 16 years of education and more prefer to buy KP781 treadmill and consider themselves more fit.

In [74]:
```
sns.boxplot(data=df,
            x="Fitness",
            y="Miles",
            hue='Product')
```

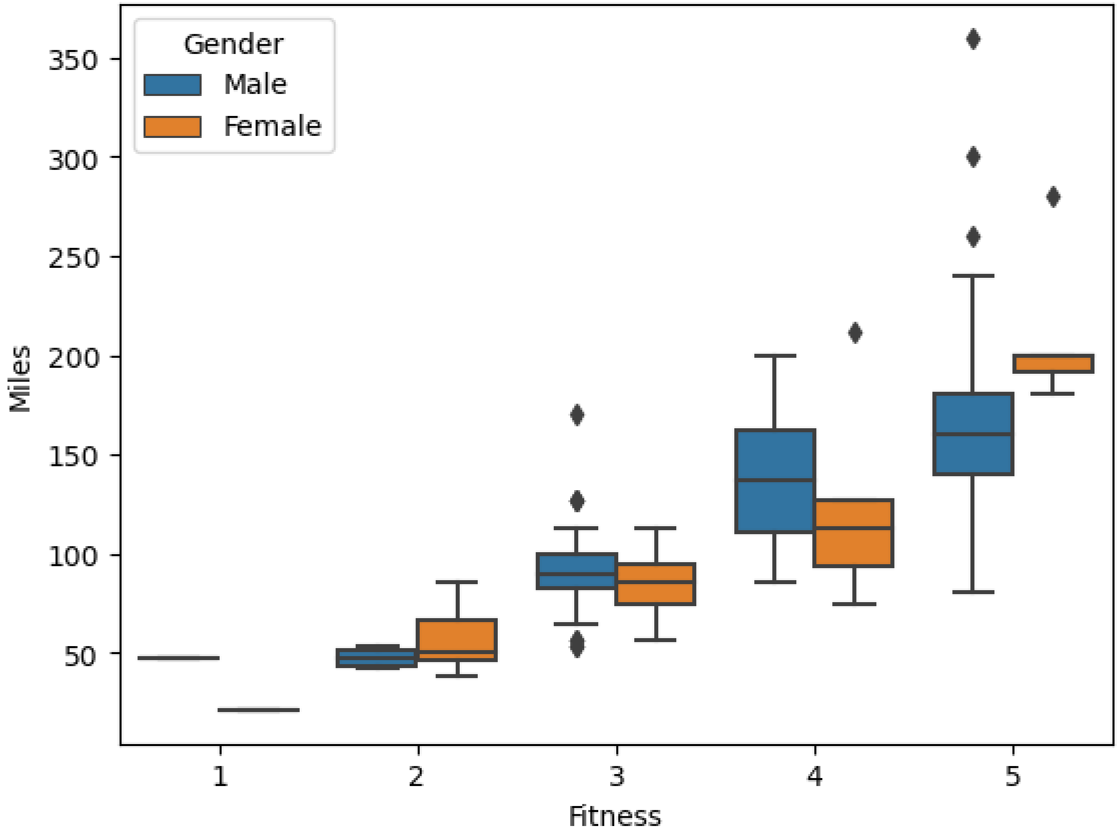
Out[74]: <AxesSubplot:xlabel='Fitness', ylabel='Miles'>



From the above, we observe that people who use KP781 treadmill run more miles and are more fit as compared to the others.

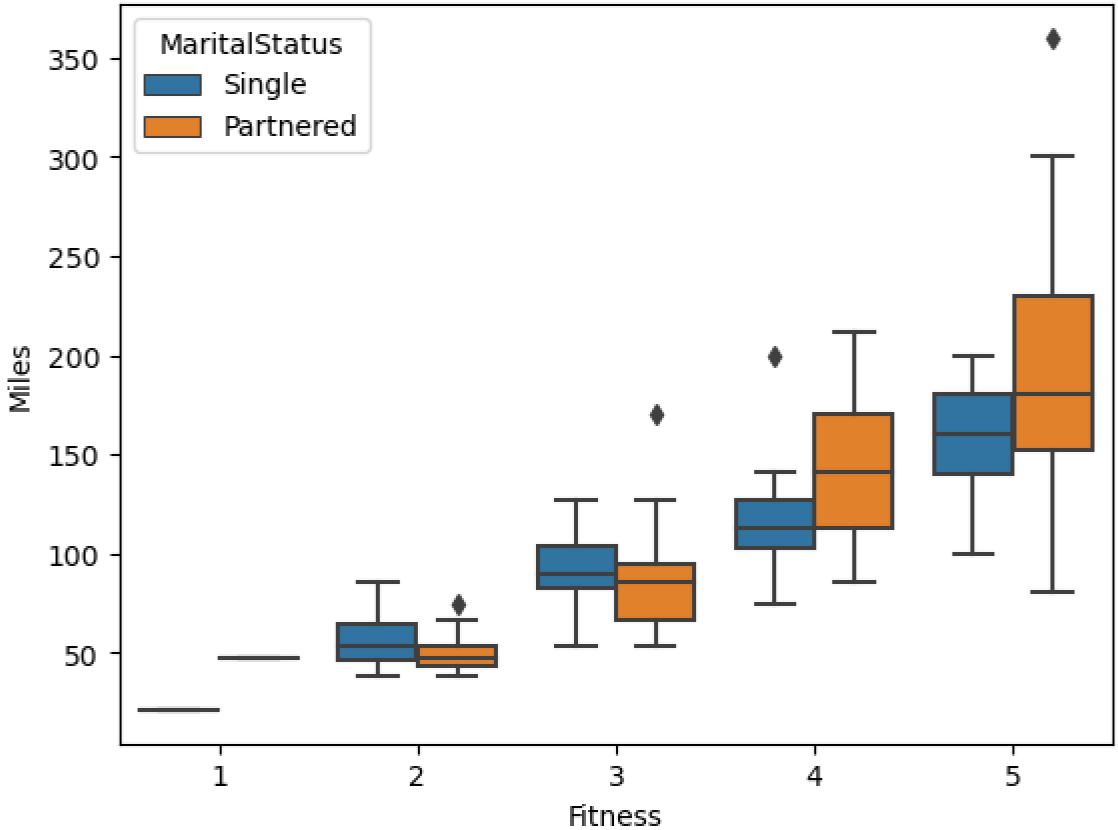
In [75]:
```
sns.boxplot(data=df,
            x="Fitness",
            y="Miles",
            hue='Gender')
```

Out[75]:
```
<AxesSubplot:xlabel='Fitness', ylabel='Miles'>
```



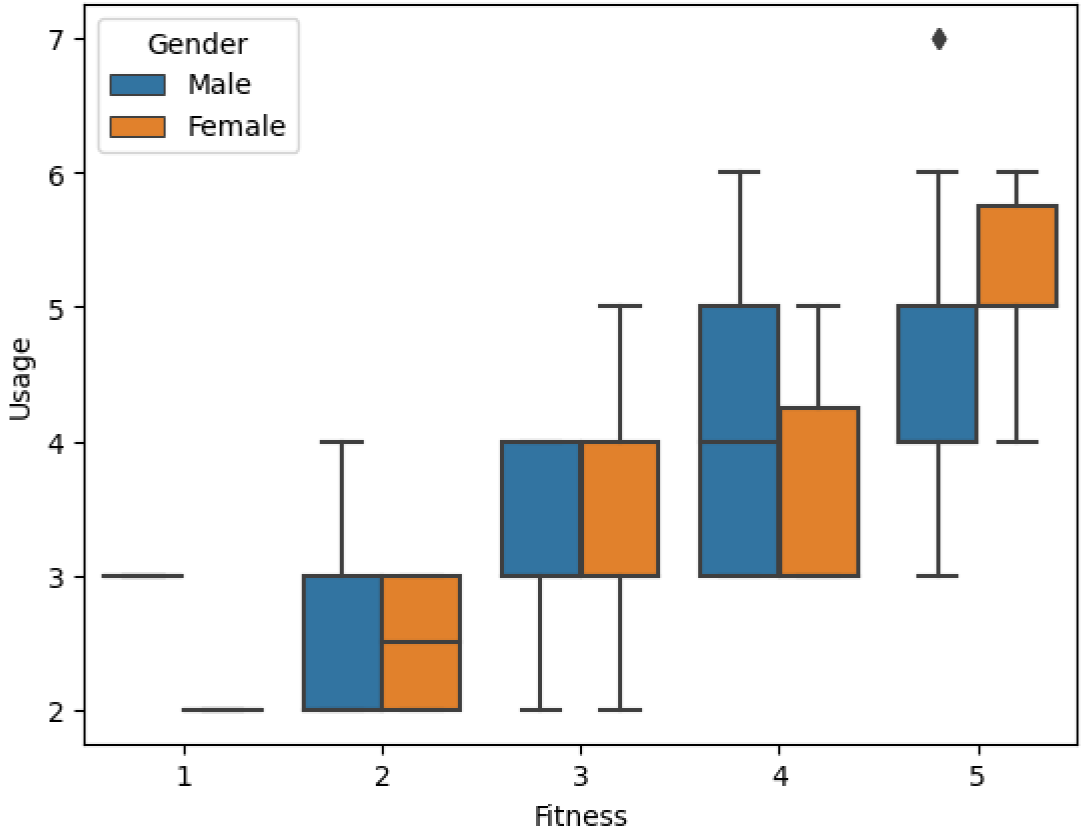
In [76]:
```
sns.boxplot(data=df,
            x="Fitness",
            y="Miles",
            hue="MaritalStatus")
```

Out[76]:
```
<AxesSubplot:xlabel='Fitness', ylabel='Miles'>
```



In [77]:
```
sns.boxplot(data=df,
            x="Fitness",
            y="Usage",
            hue='Gender')
```

Out[77]:
```
<AxesSubplot:xlabel='Fitness', ylabel='Usage'>
```



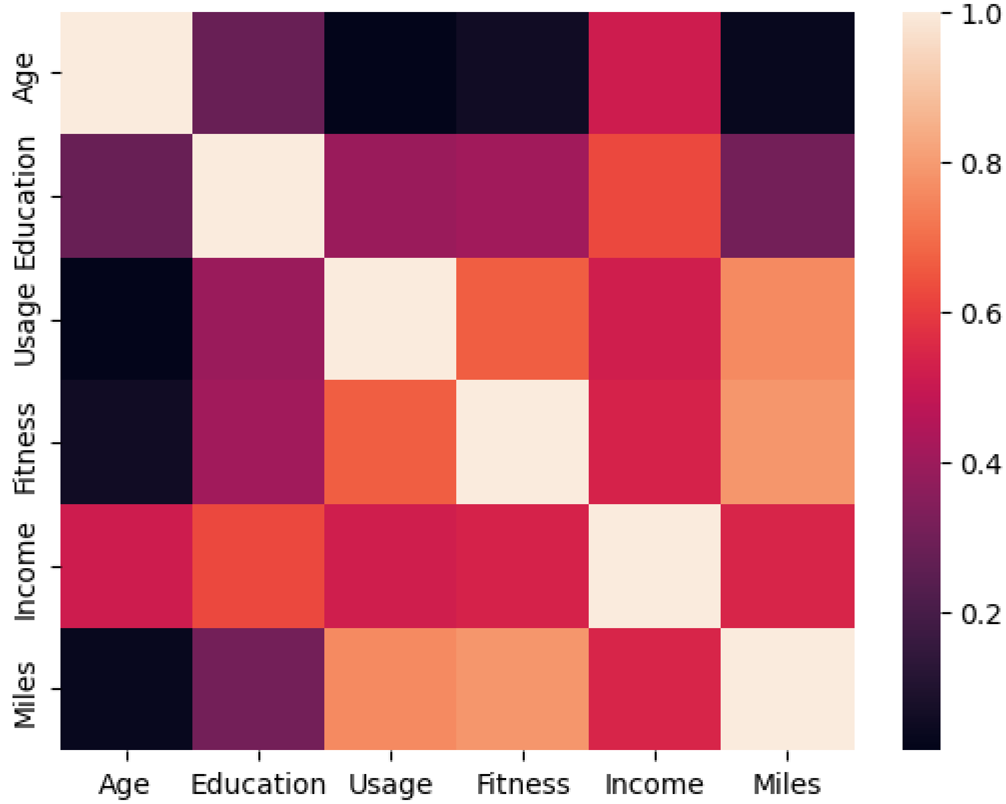
For correlation: Heatmaps, Pairplots

In [78]:
```
sns.pairplot(df)
```

Out[78]: <seaborn.axisgrid.PairGrid at 0x23dc5847af0>

In [79]:
```
sns.heatmap(df.corr())
```

Out[79]: <AxesSubplot:>



Questionnaire: 3. Name the top 3 features having the highest correlation with the 'Product' column. Also, provide possible reasons behind those correlations

1. Gender and Product : Only 17% of women have bought the KP781 treadmill. One of the reasons behind this could be lack of awareness among women regarding the benefits of the KP781 treadmill.
2. Income and Product: Income could also be one of the reasons for influencing the decision of women not to purchase KP781 treadmill as it is costly compared to the other treadmills. We can see from the observations made in this case study that the average income of females is less than males.
3. Fitness and Product : People who use KP781 treadmill run more miles and are more fit as compared to the others.

Questionnaire: 8. Distinguish between Customer Profiles for KP281 & KP481 treadmills.

1. The KP281 is an entry-level treadmill that sells for 1,500, while the KP481 is for mid-level runners that sell for 1,750. More percentage of users (44%) use KP281 as compared to KP481 (33%).
2. We observe that equal number of males and females use the KP281 treadmill, but the number of males who use KP481 treadmill is slightly higher than the number of females.
3. We also observe that for KP281 treadmill, maximum number of users are of age 23 years, while for KP481 treadmill, maximum number of people are of age 25 years.

Questionnaire: 10. Give conditions when you will and when you'll not recommend KP781 treadmill to a customer.

The KP781 treadmill is having advanced features that sell for 2,500. Users with higher income and more experience can use this treadmill as they will be able to use it to its full potential.

Business Insights based on Non-Graphical and Visual Analysis

A. Comments on the range of attributes

The dataset analyses 180 users of 3 different types of treadmills costing from 1500 - 2500 USD. The users consist of both married and single males and females in the age group 18-50 years.It also includes data regarding self-rated fitness on a 1-to-5 scale, where 1 is the poor shape and 5 is the excellent shape. The usage of treadmills ranges from 3-7 times each week.The users run/walk for 21-360 miles each week. The users have an education of 12-21 years with an income of 29,562 - 104,581 USD.

B. Comments on the distribution of the variables and relationship between them (Univariate, Bivariate and Multivariate plots)

Univariate

1. We observe that the maximum number of users are in their mid-20s.
2. We observe that users with 16 years of education are highest in number.
3. We observe that maximum number of users plan to use the treadmill thrice a week, while only very few plan to use it 6-7 times a week.
4. The variance of income in lower ages is smaller as compared to the variance in higher ages. In statistics, this is known as Heteroscedasticity.
5. We observe that maximum number of people feel that they are at point 3 on the fitness scale.
6. We observe that most number of people expect to run/walk atmost 100 miles each week.
7. The overall Probability of Purchase for KP281, KP481 & KP781 treadmills is 0.44, 0.33 and 0.22. We can observe that fewer people use KP781 treadmill as compared to KP281.
8. The data covers 58% males and 42% females.

Bivariate

1. We observe that equal number of males and females use KP281, while the number of males who use KP481 is slightly higher than the number of females. Only 17% of women have bought the KP781 treadmill.
2. We can observe that the average income of females is less than males.
3. We observe that for KP281 treadmill, maximum number of users are of age 23 years. For KP481 and KP781 treadmills, maximum number of people are of age 25 years.
4. We observe that males and females are not equally spread out over different ages. Most of the females are in the age group 20-35 years.
5. We observe that maximum number of males and females consider themselves at point 3 on the fitness scale.
6. We observe that users below income range of Rs.60,000 prefer to use KP281 and KP481 treadmills and people with income range of Rs.70,000 and above only use KP781 treadmill.

7. We observe that more number of people who use KP781 treadmill feel that they are at point 5 on the fitness scale as compared to people who use other treadmills.
8. We observe that more users who have a Partner consider themselves fit as compared to the ones who are Single. However, since there are more number of married people in the data group as compared to single people, we can say that marital status does not have much effect on fitness.
9. We observe that people using KP781 treadmill run more number of miles as compared to those using other treadmills.
10. We observe that maximum number of males and females run 85 miles each week.
11. We observe that males tend to use the treadmill for 4 hours as compared to females.
12. We observe that people with KP781 treadmill tend to put in more hours as compared to other treadmills.
13. We observe that people who run more number of miles consider themselves more fit.

Multivariate

1. We observe that even though males run more miles, they feel less fit as compared to females.
2. We observe that people with higher years of education tend to be more fit and gender does not have much effect in this case.
3. We observe that people with 16 years of education and more prefer to buy KP781 treadmill and consider themselves more fit.
4. We observe that people who use KP781 treadmill run more miles and are more fit as compared to the others.

Recommendations

After analysing the data, the following recommendations are made:

1. The target audience is in the age-group 20-30 years. With special focus on these users, Aerofit should also focus on widening its customer base by spreading awareness about the benefits of their treadmills and their ease of use so that customers from other age-groups also buy their products.
2. Along with selling treadmills, Aerofit can also start fitness training sessions/manuals for exercises which would encourage its users to exercise more and boost their fitness levels. This will in turn help promote their brand and the users might recommend it to their friends and family as well.
3. Since KP781 treadmill has a niche market, Aerofit can focus on that particular target audience only and strategize to boost their sales.