In [1]:
```
import numpy as np
```

In [5]:
```
import pandas as pd
```

In [6]:
```
import matplotlib.pyplot as plt
```

In [7]:
```
import seaborn as sns
```

In [9]:
```
df=pd.read_csv('aerofit_treadmill.csv')
```

In [10]:
```
df
```

Out[10]:

	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

```
### dataset has 180 records and 9 features

```

In [12]:
```
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
```

3 features have categorical data; one feature (Fitness) which is a rating on a scale of 1 to 5 is also a
categorical variable with numerical ratings; rest of the features have integer data types

In [31]:
```
df.isna().sum()
```

Out[31]:
```
Product          0
Age              0
Gender           0
Education        0
MaritalStatus    0
Usage            0
Fitness          0
Income           0
Miles            0
dtype: int64
```

No "NA"s / "NaN" in the data set

In [14]:

```
df.describe()
```

Out[14]:

	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
max	50.000000	21.000000	7.000000	5.000000	104581.000000	360.000000

In [15]:

```
df.describe(include='object')
```

Out[15]:

	Product	Gender	MaritalStatus
count	180	180	180
unique	3	2	2
top	KP281	Male	Partnered
freq	80	104	107

```
1 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
max	50.000000	21.000000	7.000000	5.000000	104581.000000	360.000000

mean and median do not show major significant difference for any of the numerical features (Age, Education, Usage, Fitness, Income, Miles)

In [22]:
```
products=df['Product'].value_counts()
```

In [23]:
```
products.index
```

Out[23]:
```
Index(['KP281', 'KP481', 'KP781'], dtype='object', name='Product')
```

In [24]:
```
products.values
```

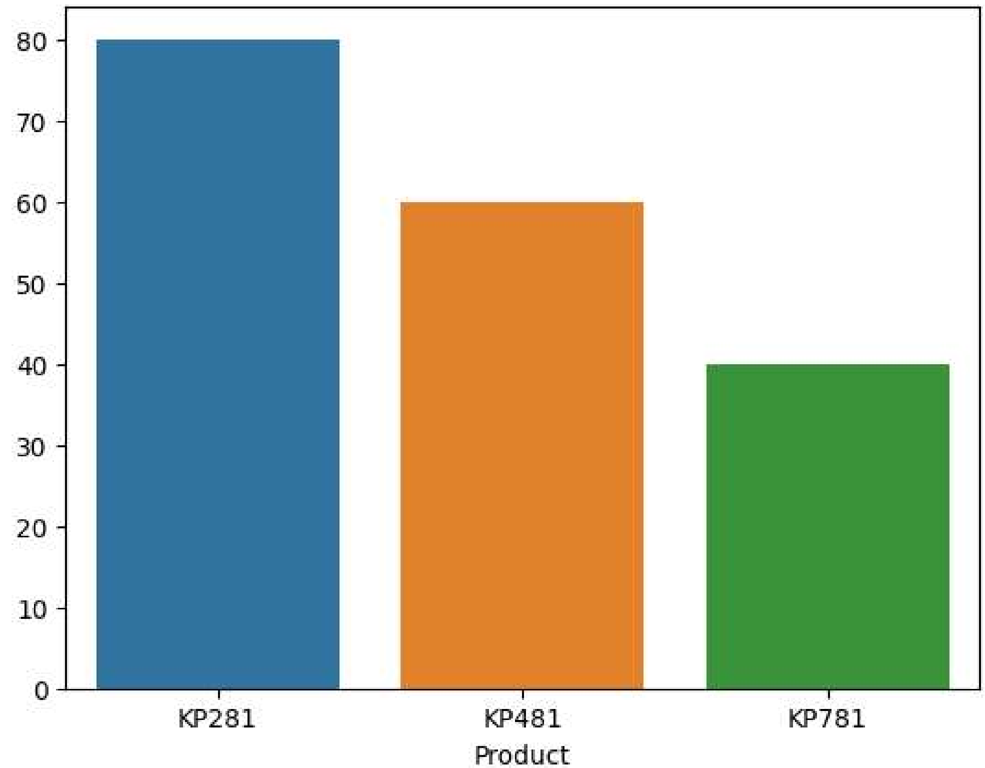
Out[24]:
```
array([80, 60, 40], dtype=int64)
```

In [29]:
```
products
```

Out[29]:
```
Product
KP281    80
KP481    60
KP781    40
Name: count, dtype: int64
```

In [28]:
```
sns.barplot(x=products.index,y=products.values)
```

Out[28]: <Axes: xlabel='Product'>



KP281 is the highest selling product followed by KP481 and KP781

In [32]:
```
genders=df['Gender'].value_counts()
```

In [33]:
```
genders
```

Out[33]:
```
Gender
Male      104
Female     76
Name: count, dtype: int64
```

In [34]:
```
genders.index
```

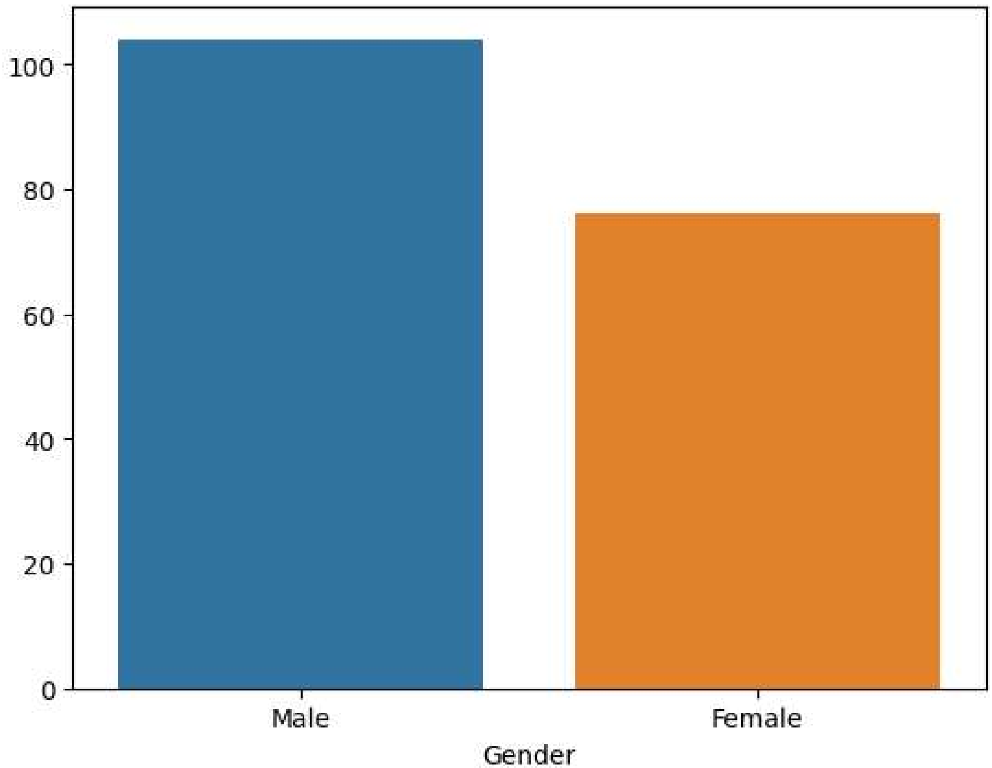
Out[34]: `Index(['Male', 'Female'], dtype='object', name='Gender')`

In [35]:
```
genders.values
```

Out[35]: `array([104,  76], dtype=int64)`

In [36]:
```
sns.barplot(x=genders.index,y=genders.values)
```

Out[36]: `<Axes: xlabel='Gender'>`



104 Males and 76 Females in the dataset

In [38]:
```
education=df['Education'].value_counts()
```

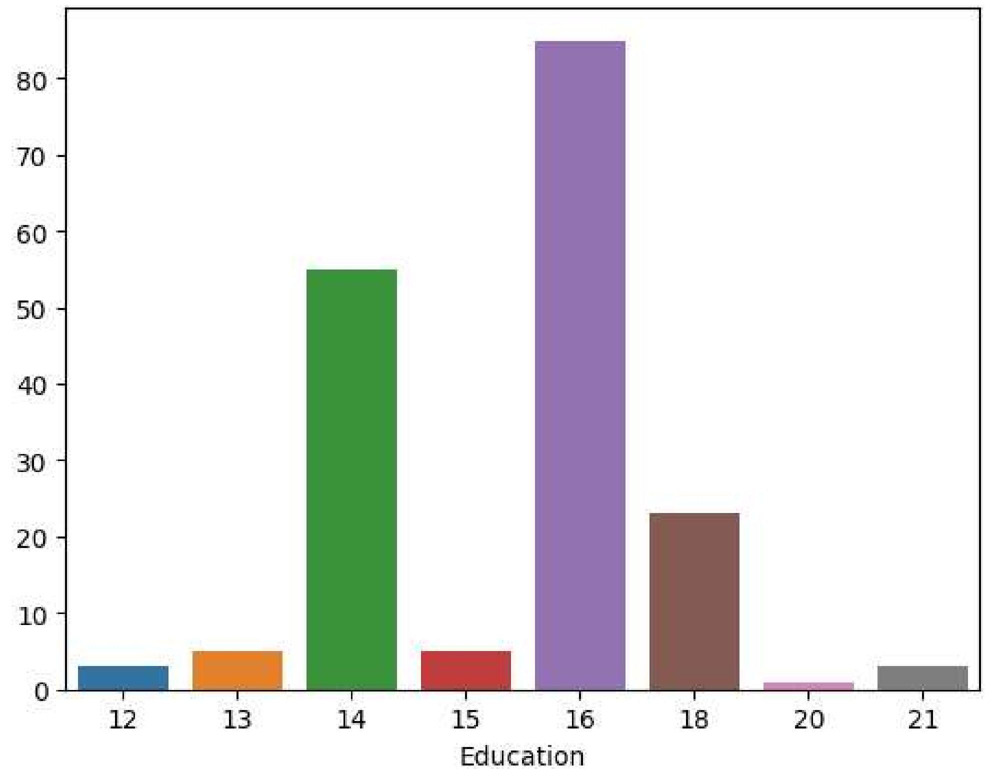
In [39]:
```
education
```

Out[39]:
```
Education
16    85
14    55
18    23
15     5
13     5
12     3
21     3
20     1
Name: count, dtype: int64
```

85 people have 16 years of education; 55 have 14 years of education; 23 have 18 years of education

In [40]:
```
sns.barplot(x=education.index,y=education.values)
```

Out[40]:
```
<Axes: xlabel='Education'>
```

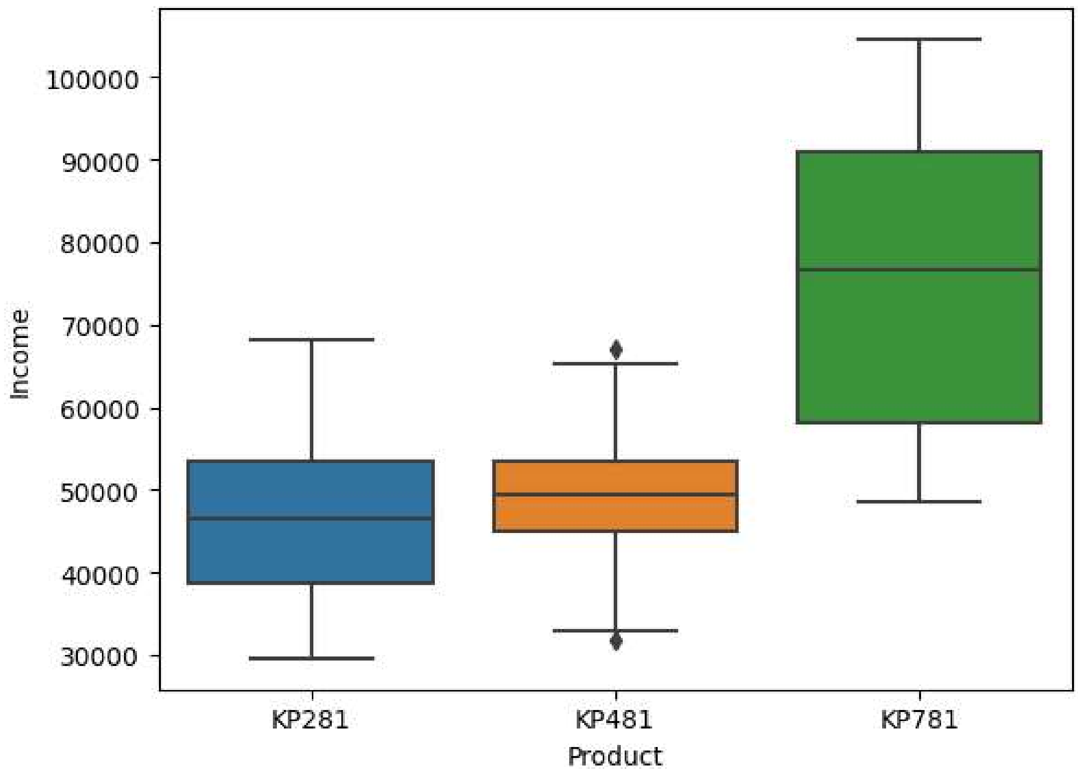


people with 16 years education are highest (85 people), followed by people with 14 years education (55
people) and people with 18 years education (23 people); people with less than 14 years education and
more than 18 years education are relatively very less.

In [49]:

```python
sns.boxplot(data=df,x='Product',y='Income')
```

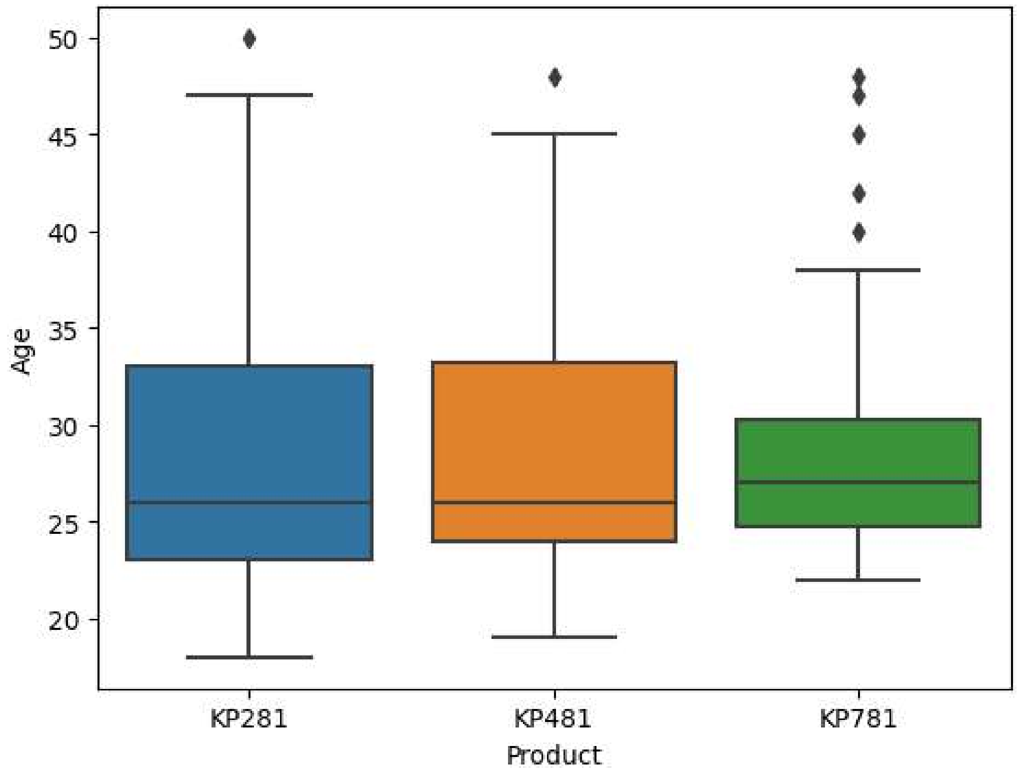
Out[49]: <Axes: xlabel='Product', ylabel='Income'>



The above box plot shows that higher income people go for KP781., medium income people go for KP481,
Medium to lower income people go for KP281

In [50]:
```python
sns.boxplot(data=df,x='Product',y='Age')
```

Out[50]: <Axes: xlabel='Product', ylabel='Age'>

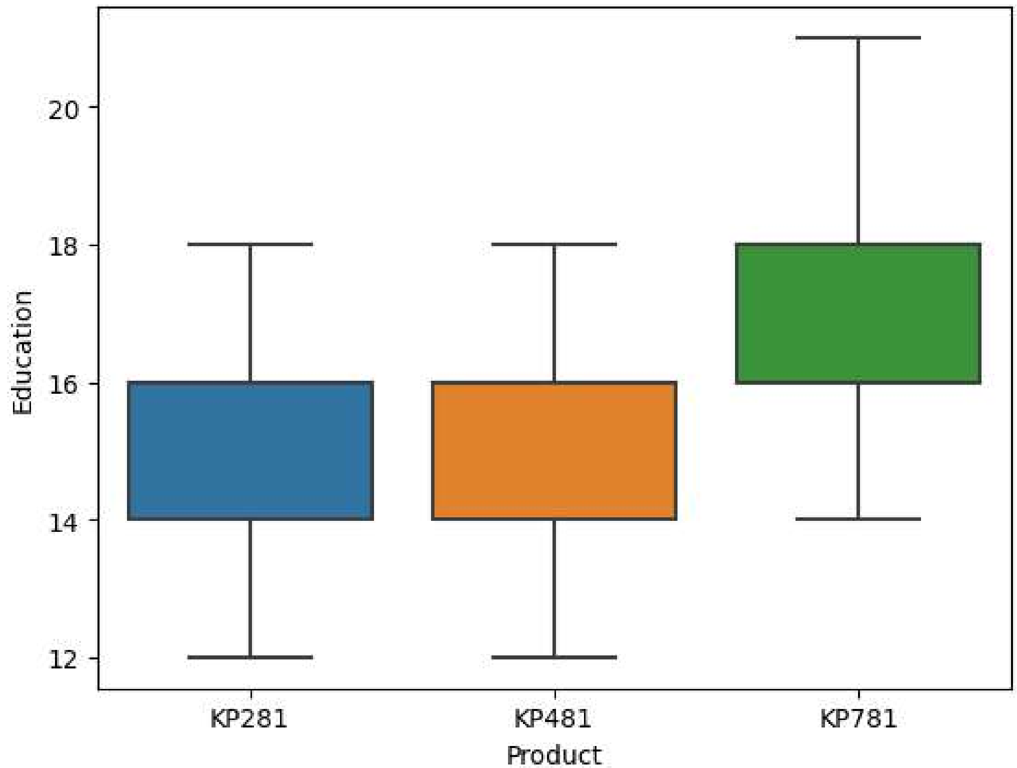


Younger people (age ~24 to ~30) have affinity towards KP781.
Age group is more wider for people who prefer KP481 and KP281

In [52]:

```
sns.boxplot(data=df,x='Product',y='Education')
```

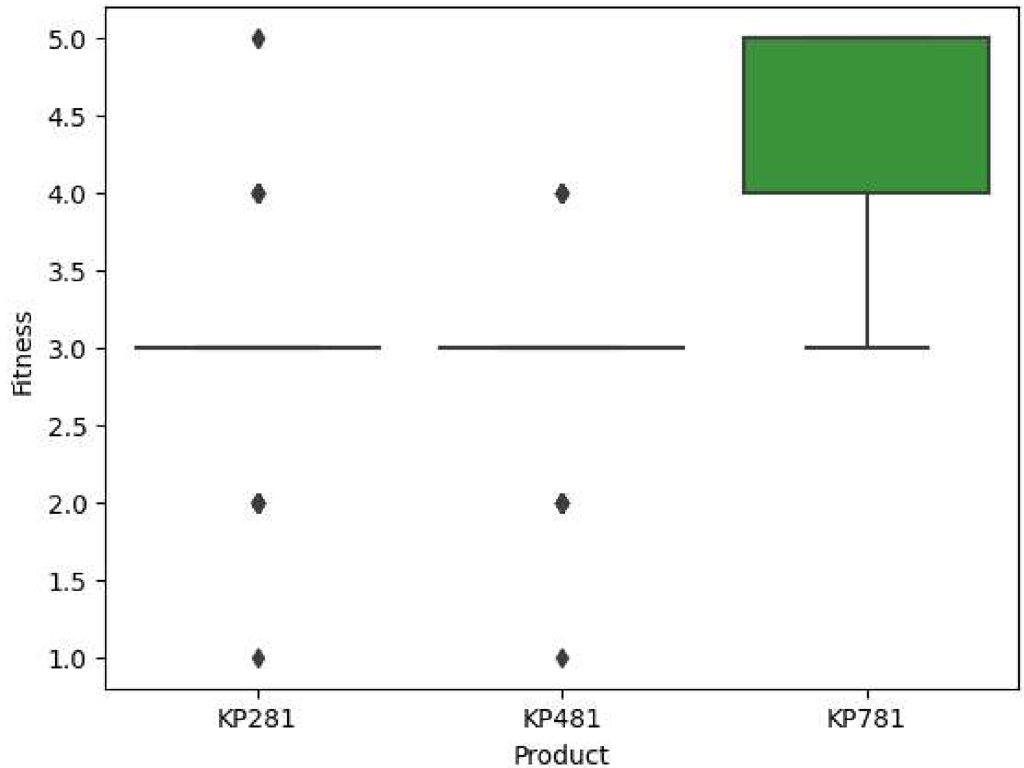
Out[52]: <Axes: xlabel='Product', ylabel='Education'>




```
### more educated (16 years and more) people are preferring KP781
```

In [53]:
```
sns.boxplot(data=df,x='Product',y='Fitness')
```

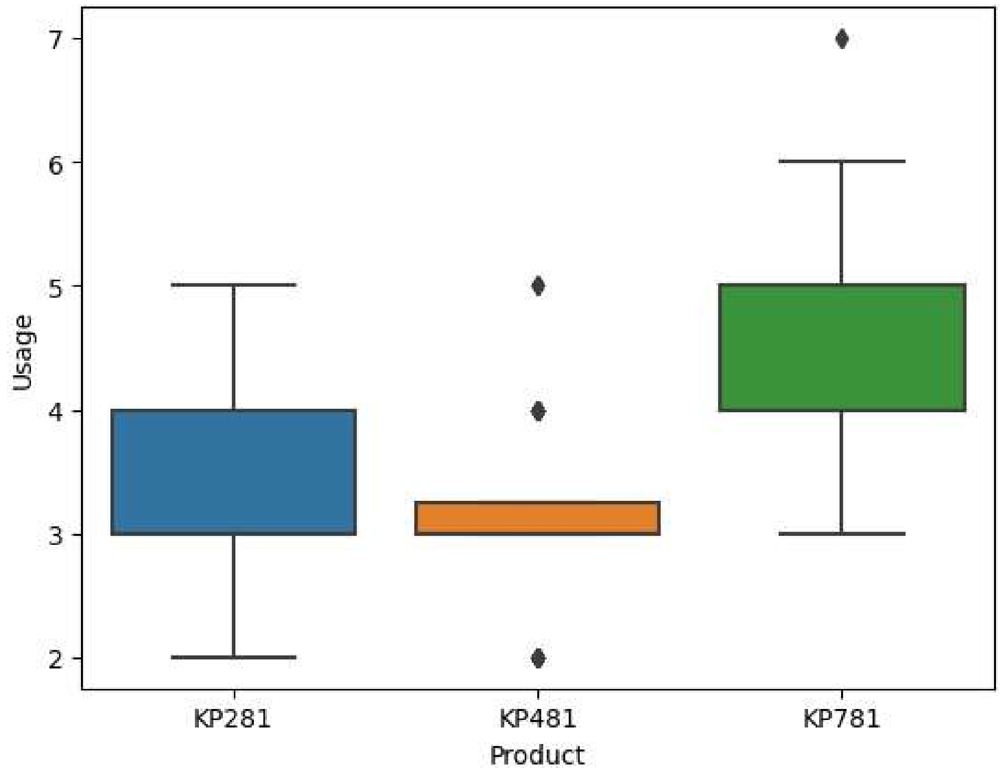
Out[53]: <Axes: xlabel='Product', ylabel='Fitness'>



more fitness conscious people are freferring KP781

In [54]:
```
sns.boxplot(data=df,x='Product',y='Usage')
```

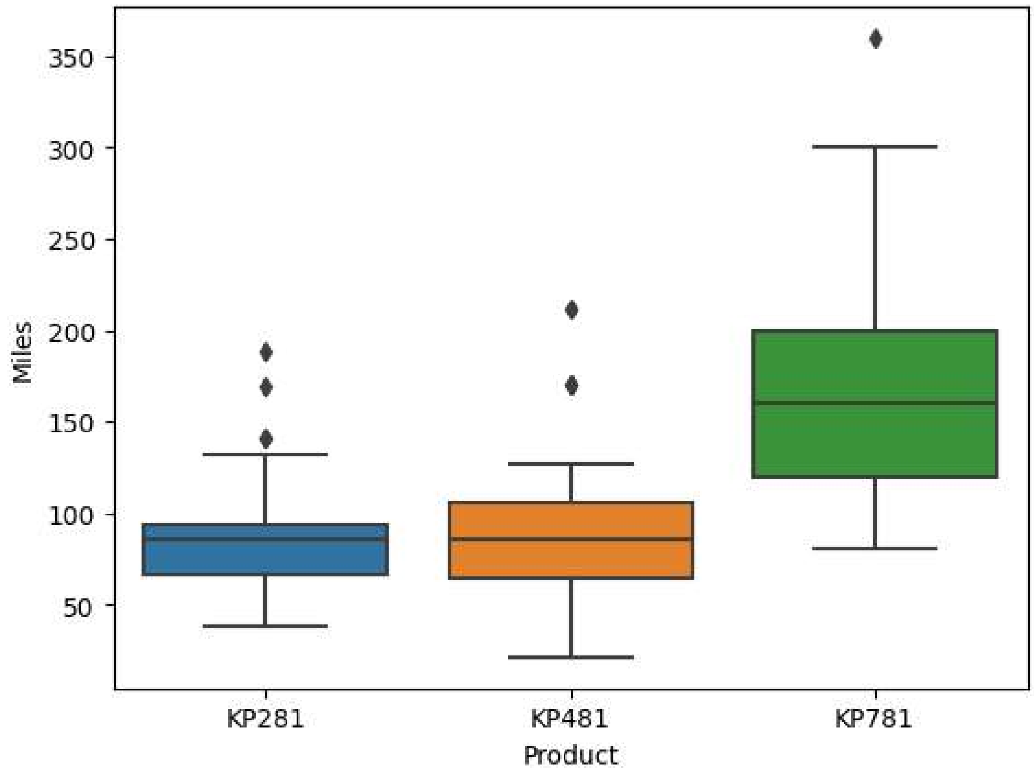
Out[54]: <Axes: xlabel='Product', ylabel='Usage'>




```
### people who use the product more (4 times a week or more) have preferred KP781
```

In [55]:
```
sns.boxplot(data=df,x='Product',y='Miles')
```

Out[55]: <Axes: xlabel='Product', ylabel='Miles'>



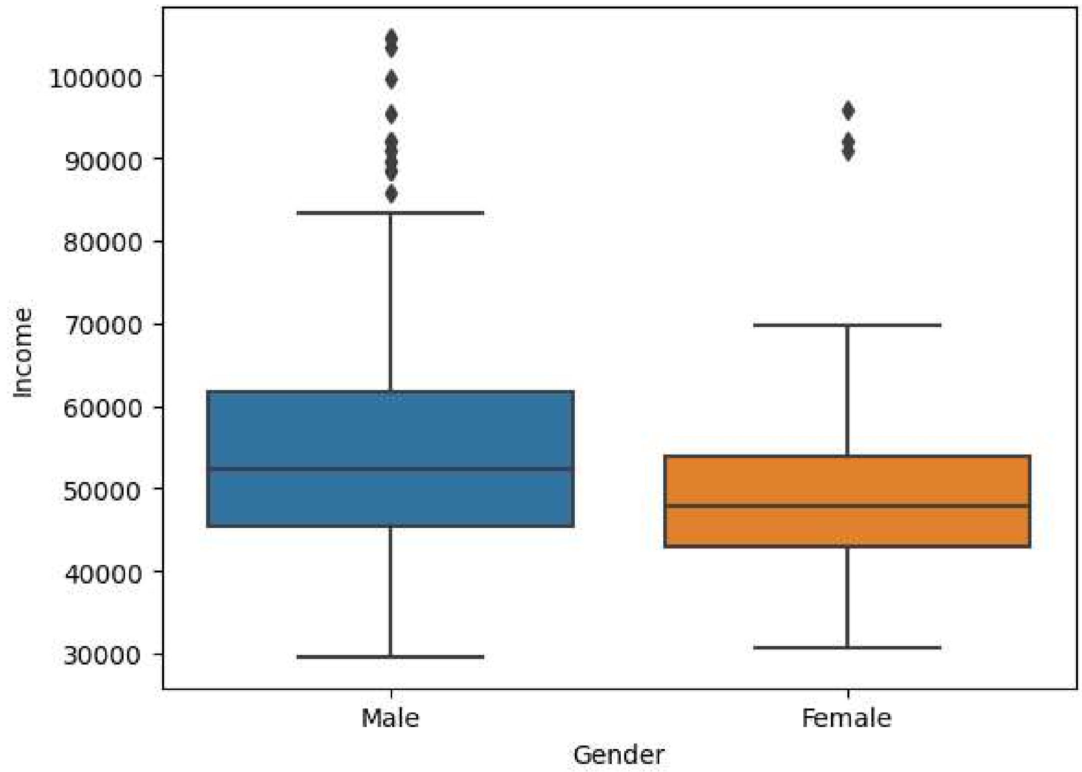

```
### people who walk (run) more ~150 miles or more have preferred KP781
```

In [58]:

```
sns.boxplot(data=df,x='Gender',y='Income')
```

Out[58]:

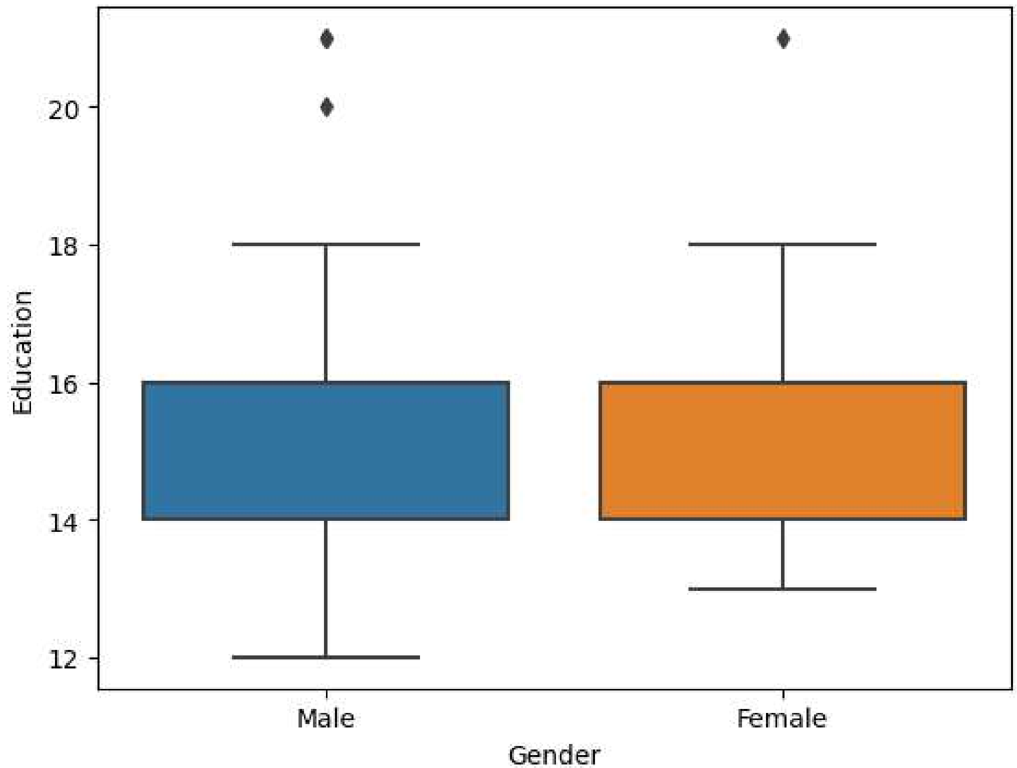
```
<Axes: xlabel='Gender', ylabel='Income'>
```



In [59]:

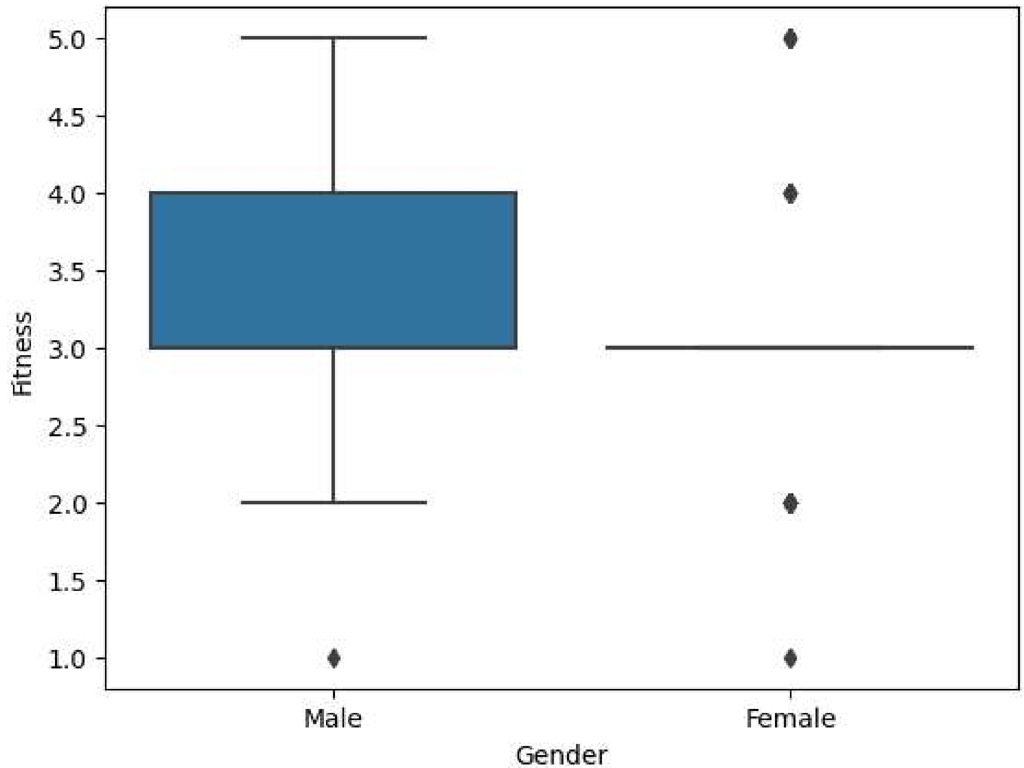
```
sns.boxplot(data=df,x='Gender',y='Education')
```

Out[59]: <Axes: xlabel='Gender', ylabel='Education'>



In [60]:
```
sns.boxplot(data=df,x='Gender',y='Fitness')
```

Out[60]: <Axes: xlabel='Gender', ylabel='Fitness'>

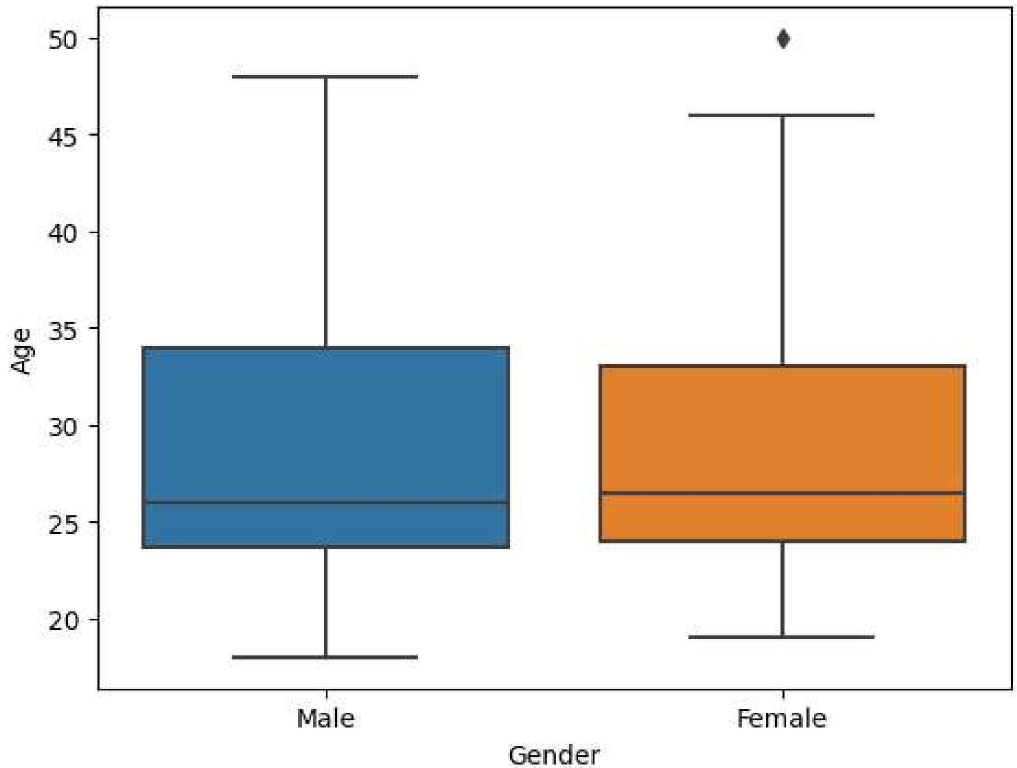


Women are less conscious about fitness compared to men

In [61]:

```
sns.boxplot(data=df,x='Gender',y='Age')
```

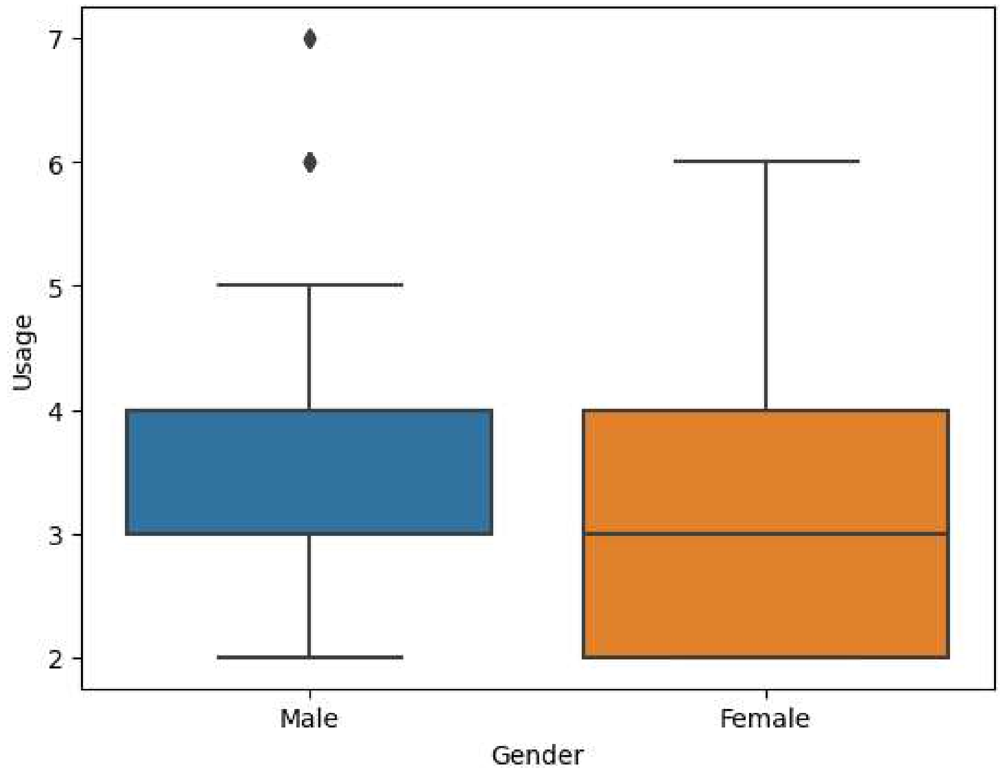
Out[61]: <Axes: xlabel='Gender', ylabel='Age'>



In [62]:

```
sns.boxplot(data=df,x='Gender',y='Usage')
```

Out[62]: <Axes: xlabel='Gender', ylabel='Usage'>

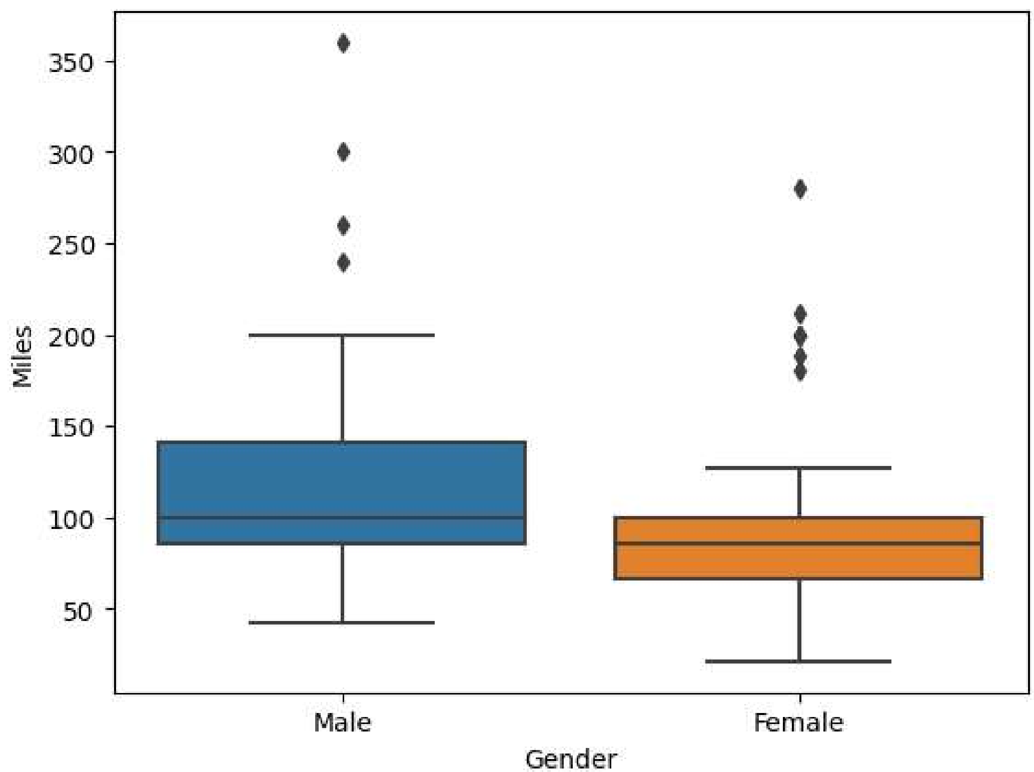



```
### women have less usage compared to men
```

In [63]:

```
sns.boxplot(data=df,x='Gender',y='Miles')
```

Out[63]: <Axes: xlabel='Gender', ylabel='Miles'>




```
### data shows men walk (run) more miles compared to women
```

In [72]:

```
pd.crosstab(df['Product'], df['Gender'])
```

Out[72]:

Gender	Female	Male
Product		
KP281	40	40
KP481	29	31
KP781	7	33

In [79]:

```
pd.crosstab(df['Age'], df['Product'])
```

Out[79]:

Product	KP281	KP481	KP781
Age			
18	1	0	0
19	3	1	0
20	2	3	0
21	4	3	0
22	4	0	3
23	8	7	3
24	5	3	4
25	7	11	7
26	7	3	2
27	3	1	3
28	6	0	3
29	3	1	2
30	2	2	3
31	2	3	1
32	2	2	0
33	2	5	1
34	2	3	1
35	3	4	1
36	1	0	0
37	1	1	0
38	4	2	1
39	1	0	0
40	1	3	1
41	1	0	0
42	0	0	1
43	1	0	0
44	1	0	0
45	0	1	1

Product	KP281	KP481	KP781
Age			
46	1	0	0
47	1	0	1
48	0	1	1
50	1	0	0

In [74]:
```
pd.crosstab(df['Product'], df['Education'])
```

Out[74]:

Education	12	13	14	15	16	18	20	21
Product								
KP281	2	3	30	4	39	2	0	0
KP481	1	2	23	1	31	2	0	0
KP781	0	0	2	0	15	19	1	3

In [75]:
```
pd.crosstab(df['Product'], df['MaritalStatus'])
```

Out[75]:

MaritalStatus	Partnered	Single
Product		
KP281	48	32
KP481	36	24
KP781	23	17

In [77]:
```
pd.crosstab(df['Product'], df['Usage'])
```

Out[77]:

Usage	2	3	4	5	6	7
Product						
KP281	19	37	22	2	0	0
KP481	14	31	12	3	0	0
KP781	0	1	18	12	7	2

In [78]:
```
pd.crosstab(df['Product'], df['Fitness'])
```

Out[78]:

Fitness Product	1	2	3	4	5
KP281	1	14	54	9	2
KP481	1	12	39	8	0
KP781	0	0	4	7	29

In [ ]:
```

```

I felt excel is more suitable for summarising contingency tables; so used excel for the same.

From excel: marginal probabilities (Gender based)

Two way Contingency Table						
Count of Product	Column Labels			proportion of ownership of given product and gender (marginal probalitity)		
Row Labels	Female	Male	Row Total	Female Owners	Male Owners	proportion of owners (overall)
KP281	40	40	80	0.22	0.22	0.44
KP481	29	31	60	0.16	0.17	0.33
KP781	7	33	40	0.04	0.18	0.22
Column Total	76	104	180	0.42	0.58	1.00
marginal probability of Column Total	0.42	0.58	1.00			

From Excel: Row relative and Column relative frequencies (probabilities)

Two way Contingency Table

Count of Product	Column Labels		
Row Labels	**Female**	**Male**	**Row Total**
KP281	40	40	**80**
KP481	29	31	**60**
KP781	7	33	**40**
Column Total	**76**	**104**	**180**
proportion of Gender who own KP281	0.53	0.38	0.44
proportion of Gender who own KP481	0.38	0.30	0.33
proportion of Gender who own KP781	0.09	0.32	0.22

proportion of ownership of given product & gender

Female Owners	Male Owners	proportion of owners (overall)
0.50	0.50	0.44
0.48	0.52	0.33
0.18	0.83	0.22
0.42	0.58	1.00

```
1
```

From Excel: marginal probabilities (Education based)

Two-way contingency Table

Count of Product	Column Labels (Education in years)								
Row Labels	12	13	14	15	16	18	20	21	Row Total
KP281	2	3	30	4	39	2			80
KP481	1	2	23	1	31	2			60
KP781			2		15	19	1	3	40
Column Total	3	5	55	5	85	23	1	3	180
Marginal probalility of Column Totals	0.02	0.03	0.31	0.03	0.47	0.13	0.01	0.02	1.00

proportion ownership of given product & Education (marginal probability)									
	12	13	14	15	16	18	20	21	Row Total
KP281	0.01	0.02	0.17	0.02	0.22	0.01	0.00	0.00	0.44
KP481	0.01	0.01	0.13	0.01	0.17	0.01	0.00	0.00	0.33
KP781	0.00	0.00	0.01	0.00	0.08	0.11	0.01	0.02	0.22
Column Total	0.02	0.03	0.31	0.03	0.47	0.13	0.01	0.02	1.00

From Excel: Row relative and Column relative frequencies (probabilities)

Two-way contingency Table:

Count of Product	Column Labels								
Row Labels	12	13	14	15	16	18	20	21	Row Total
KP281	2	3	30	4	39	2			80
KP481	1	2	23	1	31	2			60
KP781			2		15	19	1	3	40
Column Total	3	5	55	5	85	23	1	3	180
proportion of ownership of given product under each education level	0.67	0.60	0.55	0.80	0.46	0.09	0.00	0.00	0.44
proportion of ownership of given product under each education level	0.33	0.40	0.42	0.20	0.36	0.09	0.00	0.00	0.33
proportion of ownership of given product under each education level	0.00	0.00	0.04	0.00	0.18	0.83	1.00	1.00	0.22

proportion ownership of given product & Education									
	12	13	14	15	16	18	20	21	Row Total
KP281	0.03	0.04	0.38	0.05	0.49	0.03	0.00	0.00	0.44
KP481	0.02	0.03	0.38	0.02	0.52	0.03	0.00	0.00	0.33
KP781	0.00	0.00	0.05	0.00	0.38	0.48	0.03	0.08	0.22
Column Total	0.02	0.03	0.31	0.03	0.47	0.13	0.01	0.02	1.00

From Excel: marginal probabilities (Marital status based)

Two-way Contingency Table:

Count of Product	Column Labels		
Row Labels	**Partnered**	**Single**	**Row Total**
KP281	48	32	**80**
KP481	36	24	**60**
KP781	23	17	**40**
Column Total	**107**	**73**	**180**
Marginal probability (column Total)	0.59	0.41	1.00

proportion ownership of given product & Marital Status

Partnered	Single	**Row Total**
0.27	0.18	0.44
0.20	0.13	0.33
0.13	0.09	0.22
0.59	0.41	1.00

From Excel: Row relative and Column Relative frequencies (probabilities)

Two-way Contingency Table:

Count of Product	Column Labels				proportion ownership of given product & Marital Status		
Row Labels	**Partnered**	**Single**	**Row Total**		Partnered	Single	**Row Total**
KP281	48	32	**80**		0.60	0.40	0.44
KP481	36	24	**60**		0.60	0.40	0.33
KP781	23	17	**40**		0.58	0.43	0.22
Column Total	**107**	**73**	**180**		0.59	0.41	1.00
proportion od ownership of KP281 among all products	0.45	0.44	0.44				
proportion od ownership of KP481 among all products	0.34	0.33	0.33				
proportion od ownership of KP781 among all products	0.21	0.23	0.22				

From Excel: marginal probabilities based on Fitness ratings

Two-way Contingency Table:

Count of Product	**Column Labels: Fitness self ratings on a scale of 1 to 5**					
Row Labels	**1**	**2**	**3**	**4**	**5**	**Row Total**
KP281	1	14	54	9	2	**80**
KP481	1	12	39	8		**60**
KP781			4	7	29	**40**
Column Total	**2**	**26**	**97**	**24**	**31**	**180**
marginal probability of Column Total	0.01	0.14	0.54	0.13	0.17	**1.00**

proportion ownership of given product & Fitness rating

1	**2**	**3**	**4**	**5**	**Row Total**
0.01	0.08	0.30	0.05	0.01	0.44
0.01	0.07	0.22	0.04	0.00	0.33
0.00	0.00	0.02	0.04	0.16	0.22
0.01	0.14	0.54	0.13	0.17	**1.00**

From Excel: Row relative and Column relative frequencies (probabilities)

Two-way Contingency Table:

Count of Product	**Column Labels: Fitness self ratings on a scale of 1 to 5**					
Row Labels	**1**	**2**	**3**	**4**	**5**	**Row Total**
KP281	1	14	54	9	2	**80**
KP481	1	12	39	8		**60**
KP781			4	7	29	**40**
Column Total	**2**	**26**	**97**	**24**	**31**	**180**
proportion of KP281 ownership among all products	0.50	0.54	0.56	0.38	0.06	0.44
proportion of KP 481 ownership among all products	0.50	0.46	0.40	0.33	0.00	0.33
proportion of KP781 ownership among all products	0.00	0.00	0.04	0.29	0.94	0.22

proportion ownership of given product vs Fitness

1	**2**	**3**	**4**	**5**	**Row Total**
0.01	0.18	0.68	0.11	0.03	0.44
0.02	0.20	0.65	0.13	0.00	0.33
0.00	0.00	0.10	0.18	0.73	0.22
0.01	0.14	0.54	0.13	0.17	**1.00**

In [ ]:

```
1
```

```
1
```

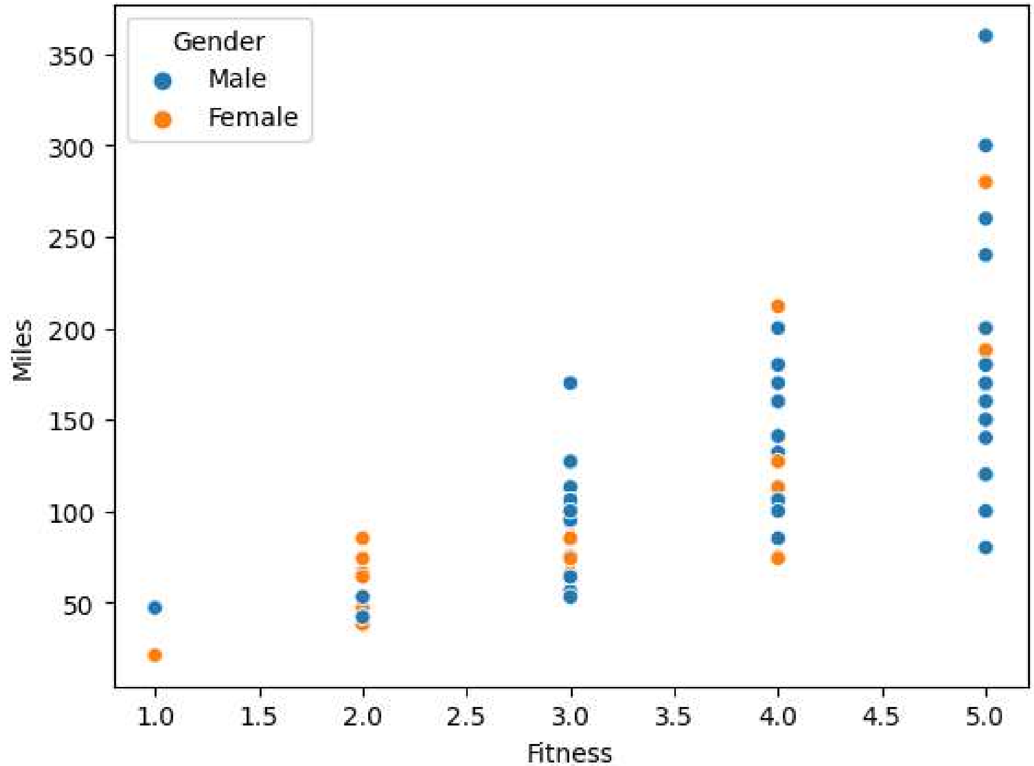
In [ ]:

```
1
```

In [83]:

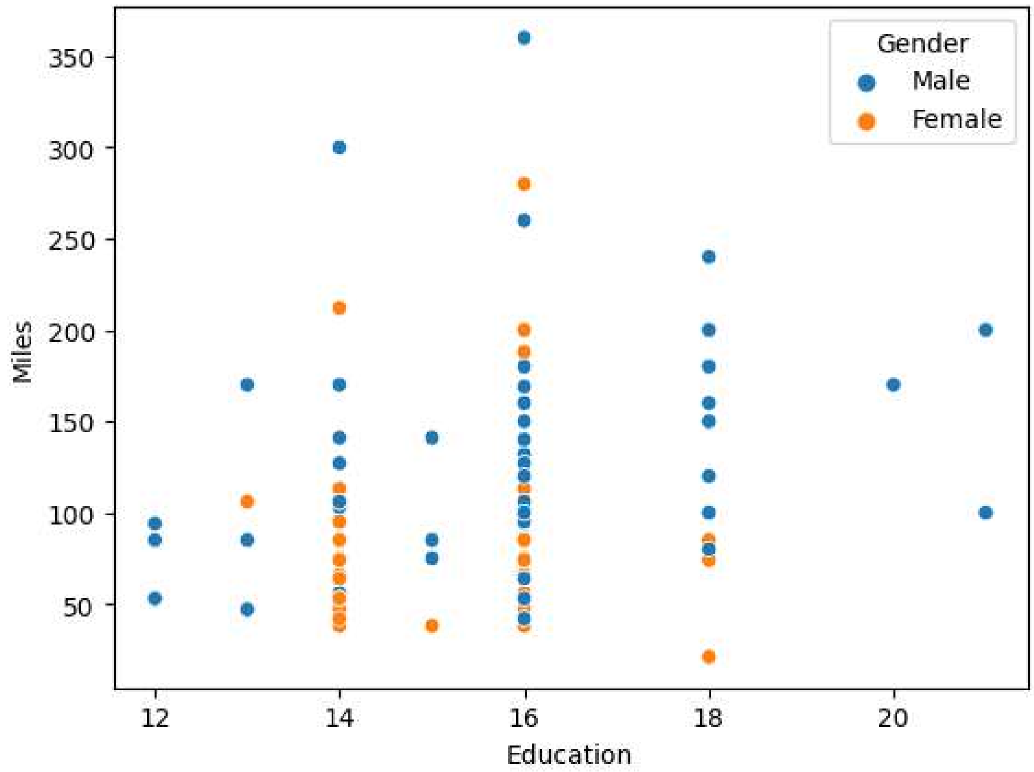
```
1 sns.scatterplot(data=df, x='Fitness',y='Miles',hue="Gender")
```

Out[83]: <Axes: xlabel='Fitness', ylabel='Miles'>



In [82]:
```
sns.scatterplot(data=df, x='Education',y='Miles', hue='Gender')
```

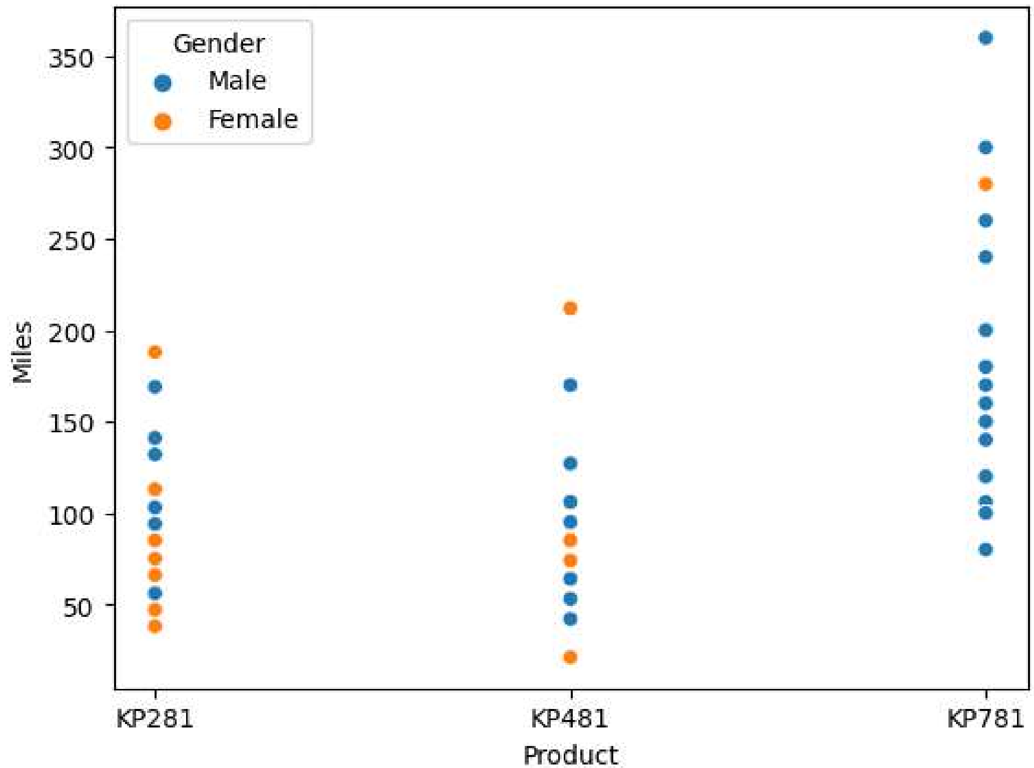
Out[82]: <Axes: xlabel='Education', ylabel='Miles'>



In [85]:

```
sns.scatterplot(data=df, x='Product',y='Miles',hue='Gender')
```

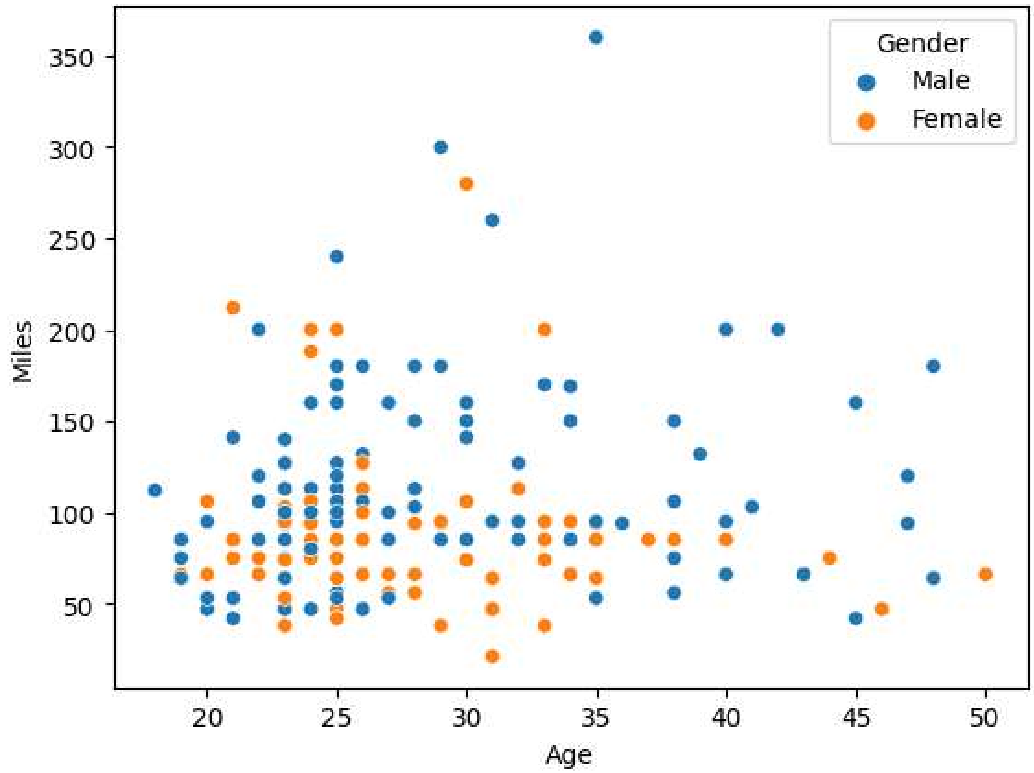
Out[85]: <Axes: xlabel='Product', ylabel='Miles'>



women have bought KP281 preferrably where as men have preferred better versions of the products

In [87]:
```
sns.scatterplot(data=df, x='Age',y='Miles',hue='Gender')
```

Out[87]: <Axes: xlabel='Age', ylabel='Miles'>

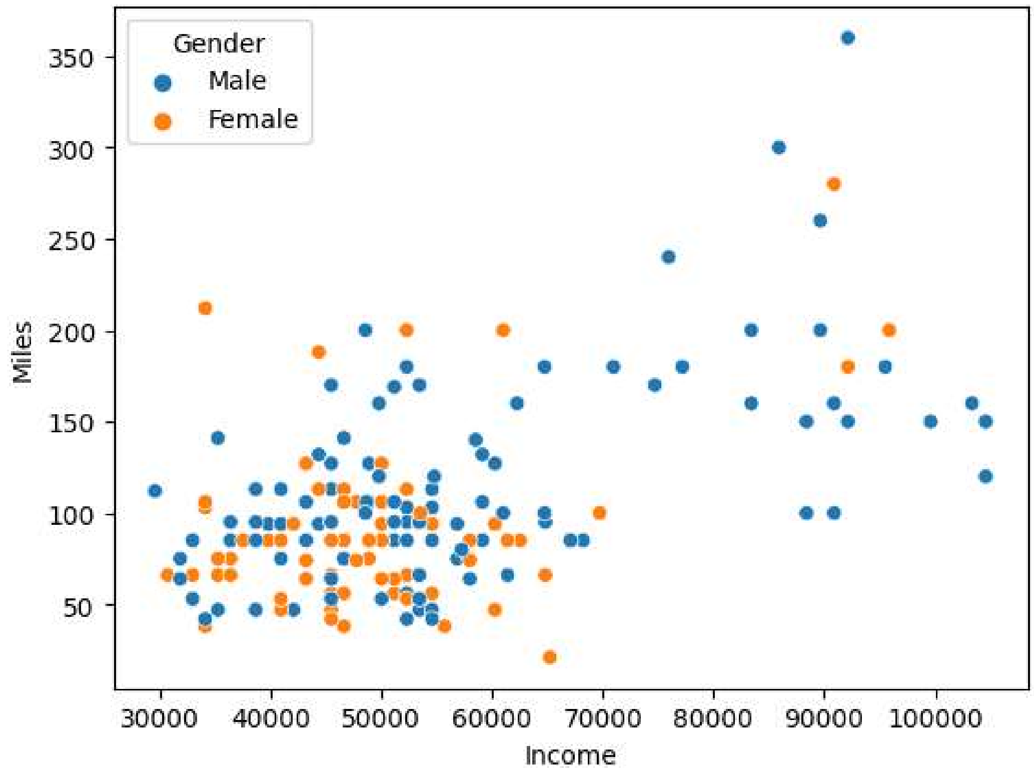


Younger people have more miles covered compared to older people; predominantly women have lesser
miles covered.

In [88]:

```
sns.scatterplot(data=df, x='Income',y='Miles',hue='Gender')
```

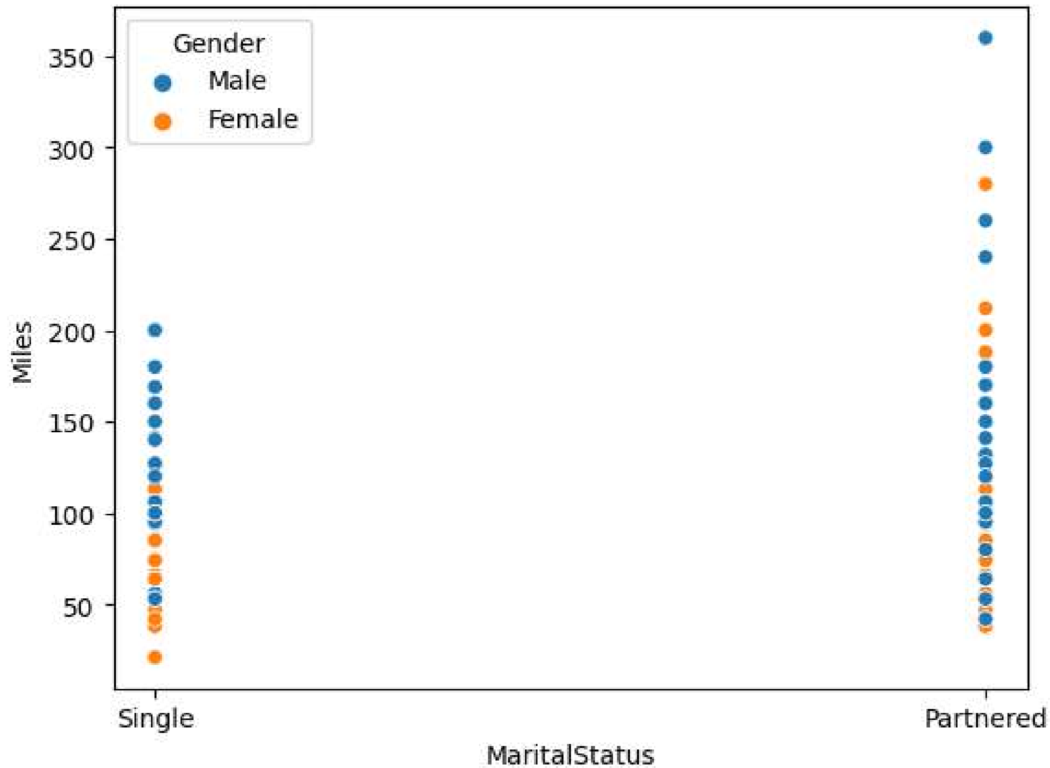
Out[88]: <Axes: xlabel='Income', ylabel='Miles'>




```
### interesting to see a dense cluster of lower income and lower miles people and
### a rarely distributed higher income and higher miles people cluster
```

In [89]:
```
sns.scatterplot(data=df, x='MaritalStatus',y='Miles',hue='Gender')
```

Out[89]: <Axes: xlabel='MaritalStatus', ylabel='Miles'>



Partnered people have more miles covered compared to singles... (little counter intuitive at least for me)

overall: people who have 14 to 18 years of education are more inclined towards fitness and prefer
products with more features.

In [ ]:

In [ ]:
```
1
```

In [ ]:
```
1
```

In [ ]:
```
1
```

In [ ]:
```
1
```

In [ ]:
```
1
```

In [ ]:
```
1
```

In [ ]:
```
1
```