PROJECT-AEROFIT

About Aerofit

Aerofit is a leading brand in the field of fitness equipment. Aerofit provides a product range including machines such as treadmills, exercise bikes, gym equipment, and fitness accessories to cater to the needs of all categories of people.

Business Problem

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.

About Data

- Product Purchased: KP281, KP481, or KP781
- Age: In years
- Gender: Male/Female
- Education: In years
- MaritalStatus: Single or partnered
- Usage: The average number of times the customer plans to use the treadmill each week.
- Income: Annual income (in $)
- Fitness: Self-rated fitness on a 1-to-5 scale, where 1 is the poor shape and 5 is the excellent shape.
- Miles: The average number of miles the customer expects to walk/run each week

In [27]:
```
import pandas as pd
import seaborn as sns
import numpy as np
import matplotlib.pyplot as plt
```

In [28]:
```
dt = pd.read_csv('aerofit_treadmill.csv')
dt.head()
```

Out[28]:

	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

Shape of Data

In [5]:
```
print('The Dataset has {0} Rows and {1} Columns'.format(dt.shape[0], dt.shape[1]))
```

```
The Dataset has 180 Rows and 9 Columns
```

Data Type

In [163...
```
cit.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
```

- Product, Gender and Martial Status Columns are Object types, rest of the columns are Integer data types

Statistical Summary

In [164...

```
# Summary of Integer Columns
cit.describe()
```

Out[164...

	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 [165...

```
# Summary of Object type Columns
dt.describe(include = 'object')
```

Ou[t165...

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

Insights -

- The Cutomers who bought had minimum age as 18 and maximum age as 50, which means people start investing on fitness after entering teenage.
- Most of the people Come from 20 - 30 age group. It can be clearly seen that the interset in fitness looses as age progresses
- The Customer with minimum Income is 29,562.0 Dollars and Customer with Maximum Income is 104,581.0 Dollars.
- As per Internet, Middle income is made up of people who make between 43,350 Dollars and 130,000 Dollars. If we consider this

2.Non-Graphical Analysis: Value counts and unique attributes

In [21]:
```
dt['Gender'].value_counts()
```

Out[21]:
```
Male      104
Female     76
Name: Gender, dtype: int64
```

In [32]:
```
pd.crosstab(dt['Gender'], dt['MaritalStatus'])
```

Out[32]:

MaritalStatus	Partnered	Single
Gender		
Female	46	30
Male	61	43

Insights-

- Most of the Customers are Males over Females
- Most of the Custoers are Partnered over being Single
- Most of the Male and Female customers are Partnered over being Single

In [29]:
```
dt['MaritalStatus'].value_counts()
```

Out[29]:
```
Partnered   107
Single       73
Name: MaritalStatus, dtype: int64
```

In [18]:
```
dt['Product'].value_counts()
```

Out[18]:
```
KP281    80
KP481    60
KP781    40
Name: Product, dtype: int64
```

In [33]:
```
pd.crosstab(dt['Gender'], dt['Product'])
```

Out[33]:

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

In [23]:
```
pd.crosstab(dt['Usage'], dt['Product'])
```

Out[23]:

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

In [24]:
```
pd.crosstab(dt['Fitness'], dt['Gender'])
```

Out[24]:

Gender	Female	Male
Fitness		
1		
2	16	10
3	45	52
4	8	16
5	6	25

In [61]:
```
pd.crosstab(dt['Fitness'], dt['Product'])
```

Out[61]:

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

In [36]:
```
pd.crosstab(dt['MaritalStatus'], dt['Product'])
```

Out[36]:

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

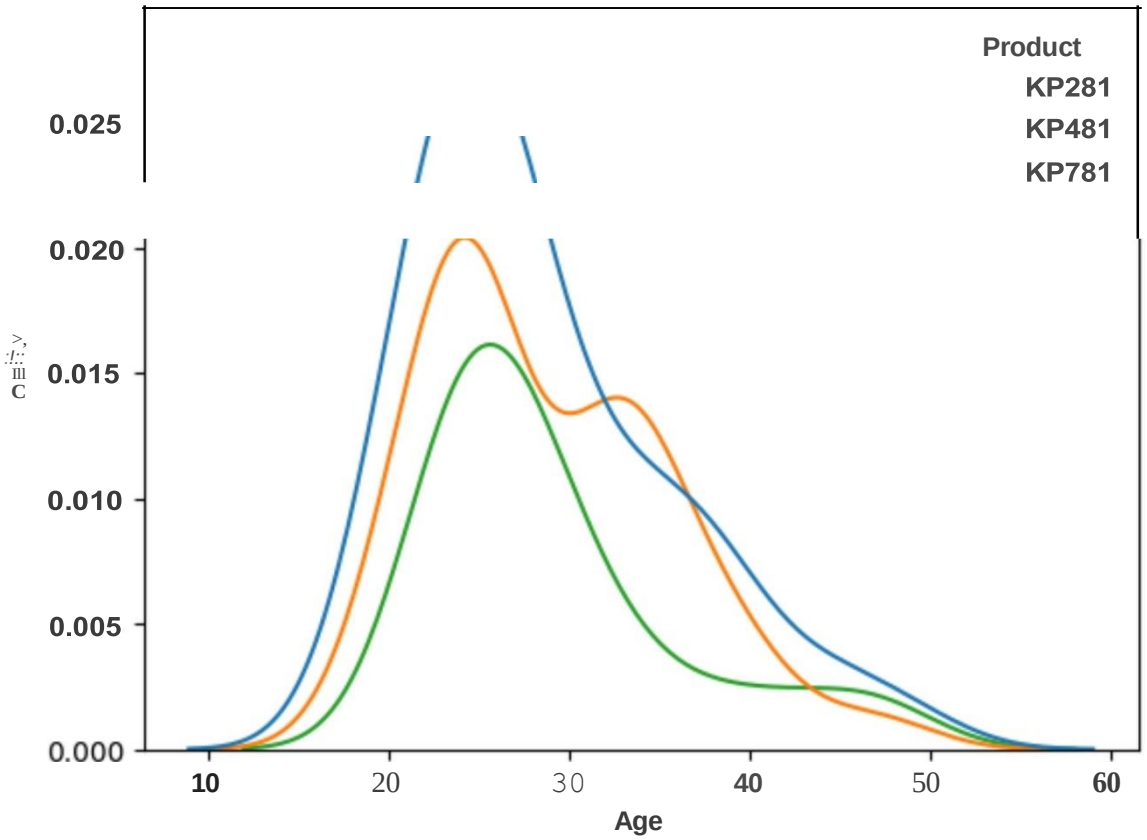
Insight-

- Most of the customers bought KP218, followed by KP481, followed by KP781
- Very less Females (7) bought KP781
- People with High level of fitness prefer KP781

3.Distribution Plot of Age

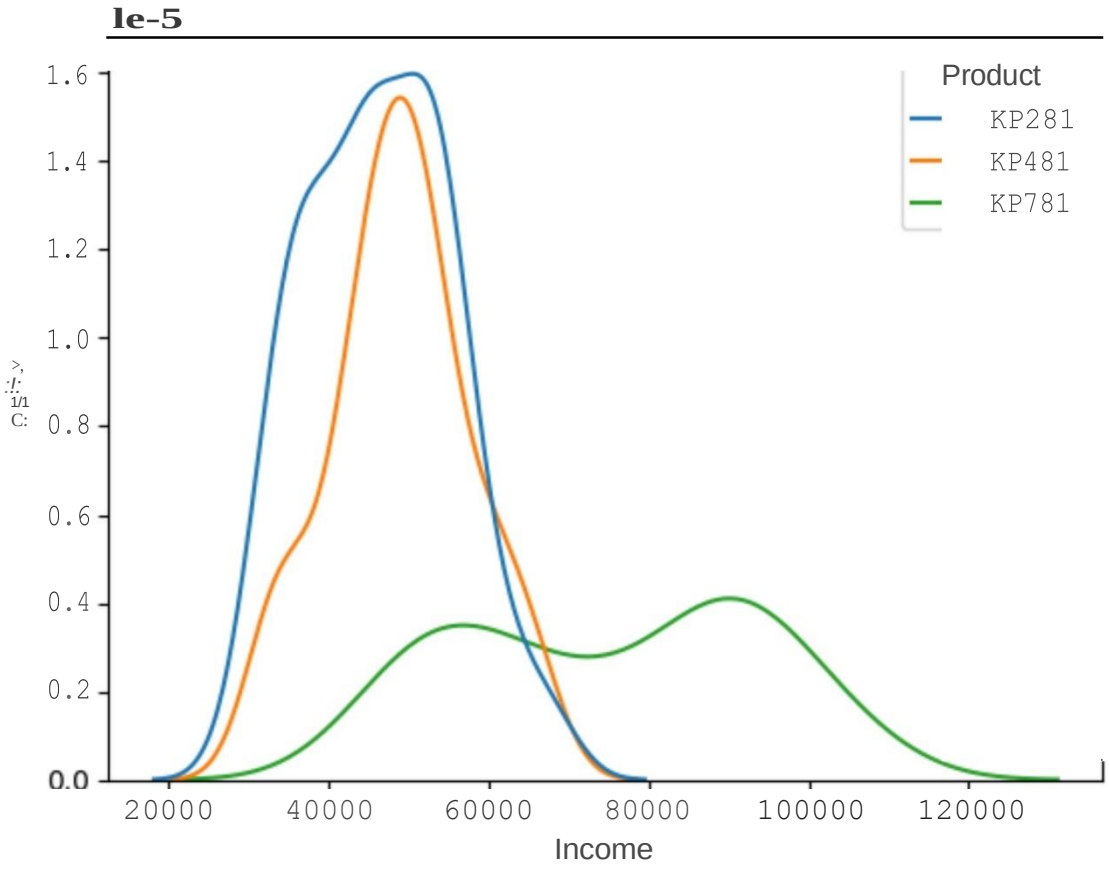
In [46]:

```
sns.kdeplot(data = dt, x = 'Age', hue= 'Product')
plt.show()
```



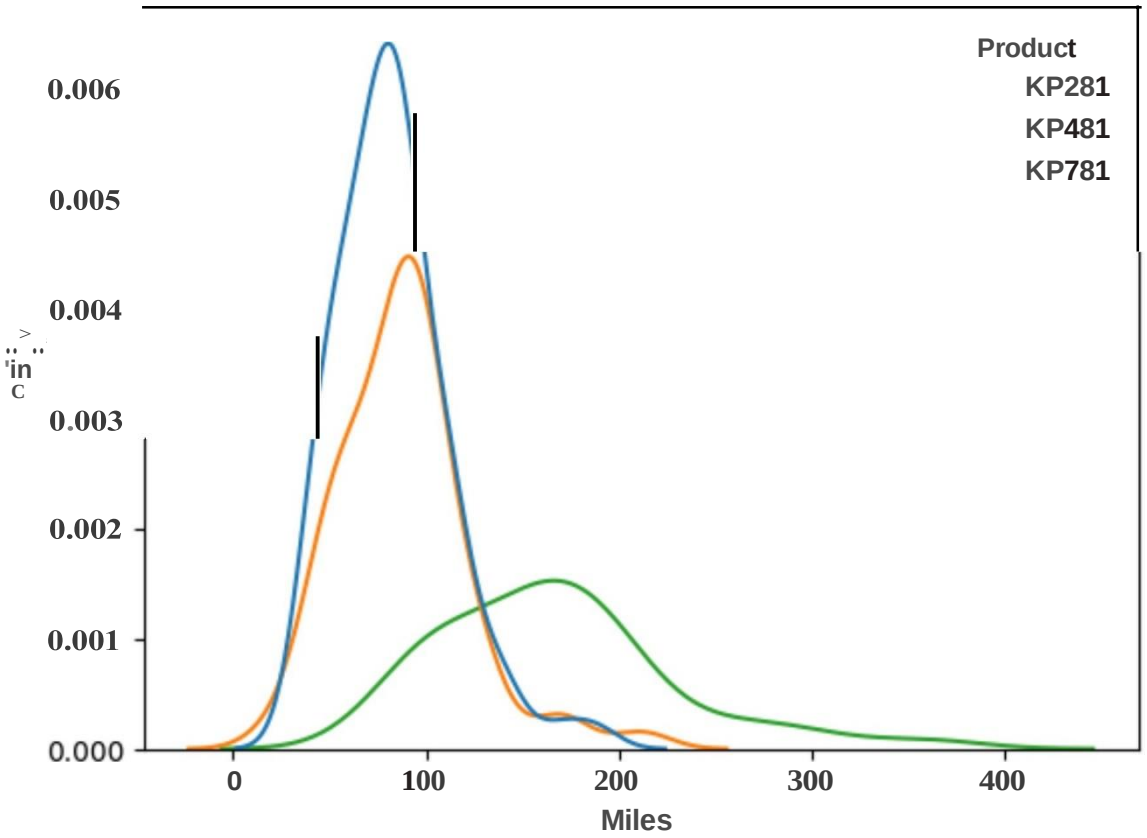
Distribution Plot of Income

In [47]:
```
sns.kdeplot(data = dt, x = 'Income', hue= 'Product')
plt.show()
```



Distribution Plot of Miles

In [49]:
```
sns.kdeplot(data = dt, x = 'Miles', hue= 'Product')
plt.show()
```



Insights -

- The Age distribution of all Product users is almost the same.
- Product KP781 Users have larger spread of Income. People with larger income tend to buy KP781
- Users of KP781 tend to run/walk more number of Average miles each week.

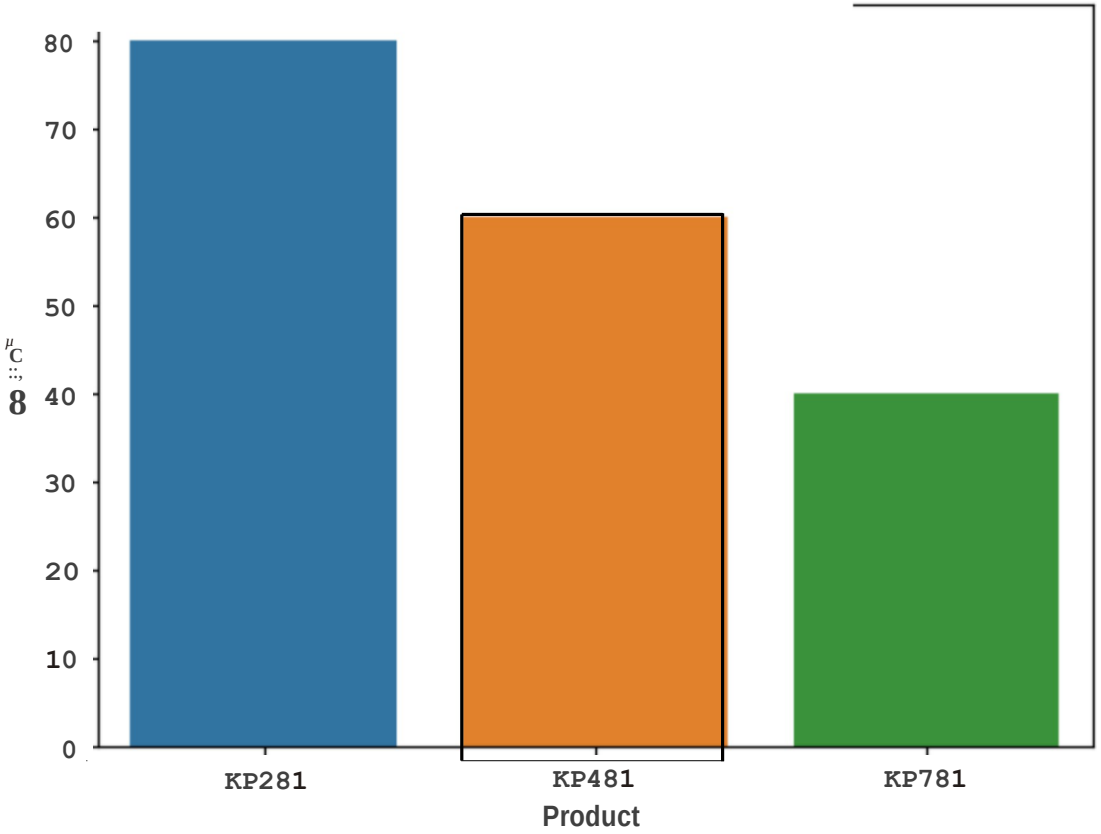
Count plot of Product

In [55]:

```
sns.countplot(dt['Product'])
plt.show()
```

```
C:\Users\manish\Anaconda3\lib\site-packages\seaborn\_decorators.py:43: FutureWarning: Pass the following variable as a
keyword arg: x. From version 0.12, the only valid positional argument will be 'data', and passing other arguments with
```

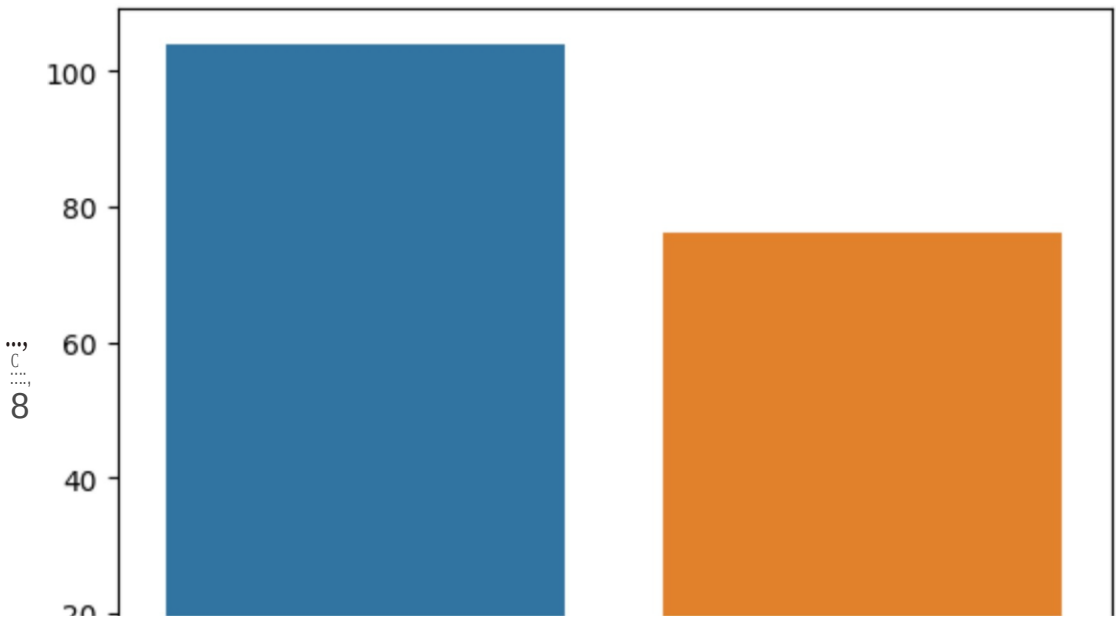
```
out an explicit keyword will result in an error or misinterpretation.
```



In [56]:

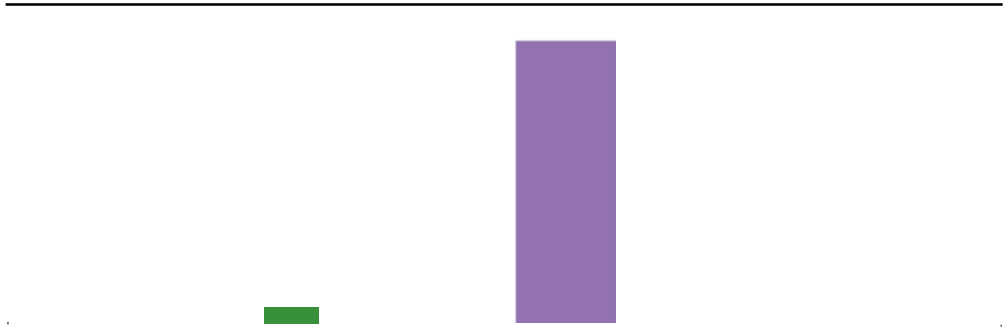
```
sns.countplot(dt['Gender'])
plt.show()
```

```
C:\Users\manish\Anaconda3\lib\site-packages\seaborn\_decorators.py:43: FutureWarning: Pass the following variable as a
keyword arg: x. From version 0.12, the only valid positional argument will be 'data', and passing other arguments with
out an explicit keyword will result in an error or misinterpretation.
  FutureWarning
```



In [66]:
```
sns.countplot(dt['Education'])
plt.show()
```

```
C:\Users\manish\Anaconda3\lib\site-packages\seaborn\_decorators.py:43: FutureWarning: Pass the following variable as a
keyword arg: x. From version 0.12, the only valid positional argument will be 'data', and passing other arguments with
out an explicit keyword will result in an error or misinterpretation.
  FutureWarning
```

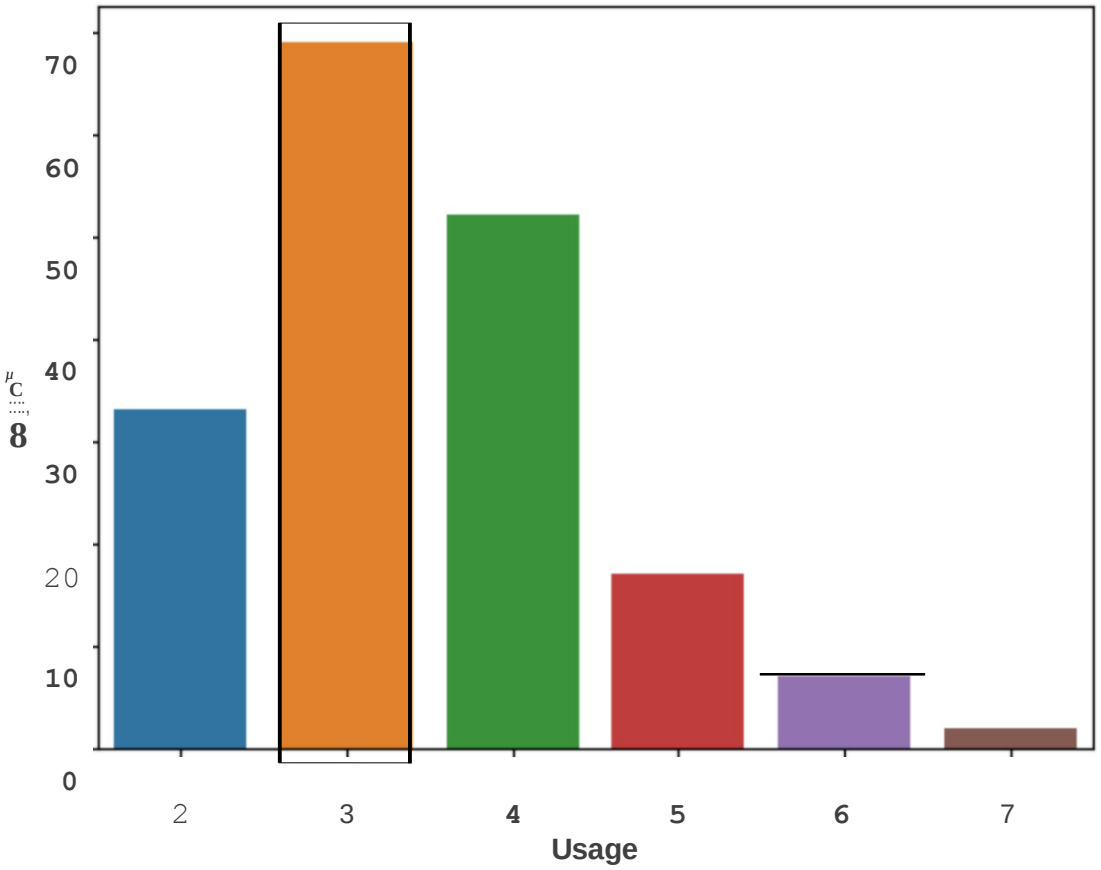


In [58]:
```
sns.countplot(dt['MaritalStatus'])
plt.show()
```

```
C:\Users\manish\Anaconda3\lib\site-packages\seaborn\_decorators.py:43: FutureWarning: Pass the following variable as a
keyword arg: x. From version 0.12, the only valid positional argument will be 'data', and passing other arguments with
out an explicit keyword will result in an error or misinterpretation.
  FutureWarning
```

In [67]:
```
sns.countplot(dt['Usage'])
plt.show()
```

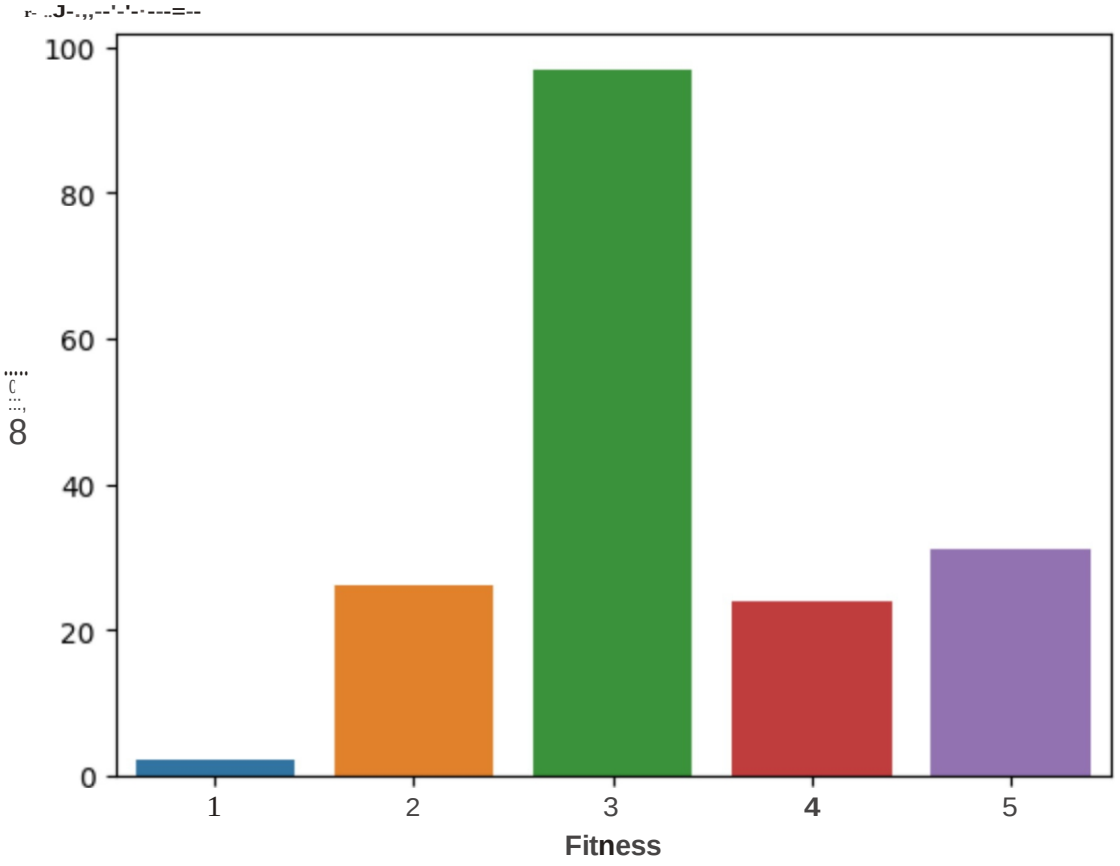
```
C:\Users\manish\Anaconda3\lib\site-packages\seaborn\_decorators.py:43: FutureWarning: Pass the following variable as a
keyword arg: x. From version 0.12, the only valid positional argument will be 'data', and passing other arguments with
out an explicit keyword will result in an error or misinterpretation.
  FutureWarning
```



In [17]:
```
sns.countplot(dt['Fitness'])
plt.show()
```

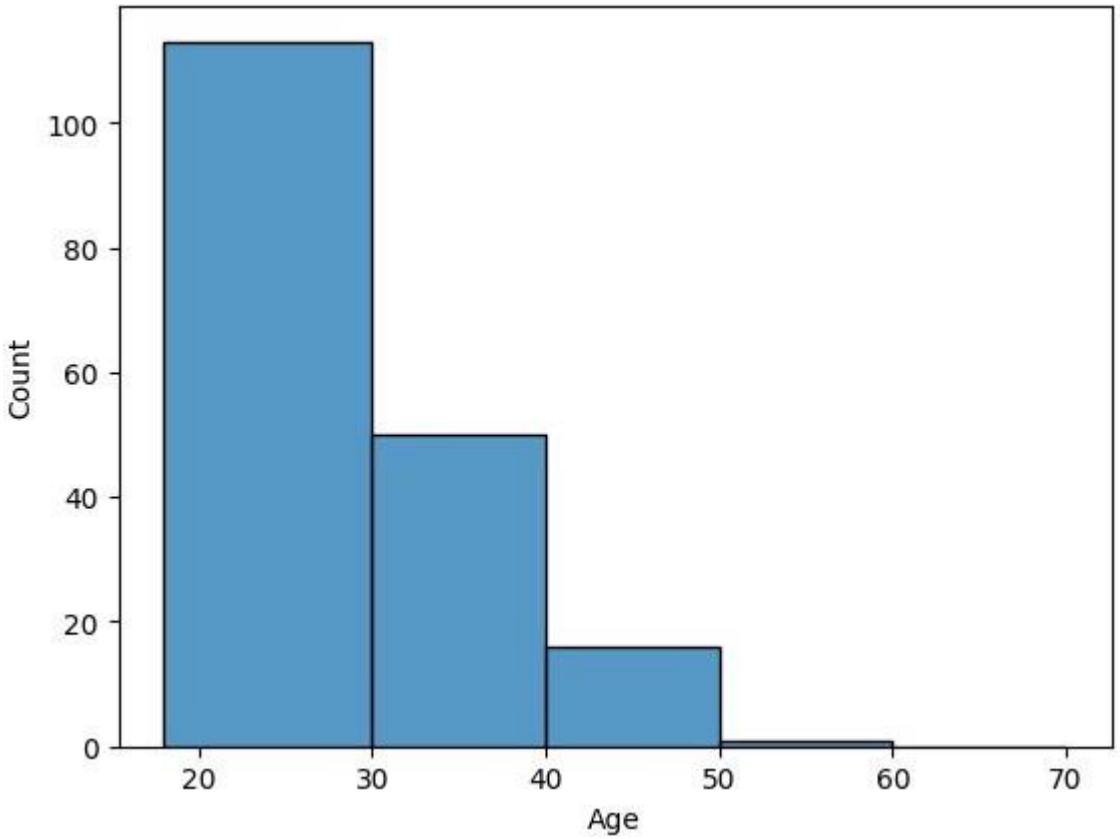
```
C:\Users\manish\Anaconda3\lib\site-packages\seaborn\_decorators.py:43: FutureWarning: Pass the following variable as a
keyword arg: x. From version 0.12, the only valid positional argument will be 'data', and passing other arguments with
```

out an explicit keyword will result in an error or misinterpretation.



In [22]:

```
sns.histplot(dt['Age'], bins= [18, 30, 40, 50, 60, 70])
plt.show()
```



Insights -

- KP281 is the most popular Product followed by KP481 and KP781
- Males bought more when compared to Female.
- People with Education level - 16, 14, 18 tend to buy it more.
- Partnered folks bought more when compared to Single
- Most of the people use it 3 times a week followed by 4, 2 and 5.
- People have Fitness level ranging from 1(Poor Shape) to 5(Perfect Shape), and most of them have fitness level 3, followed by 5 and 2
- Very least people have a Poor body shape.

In [182...

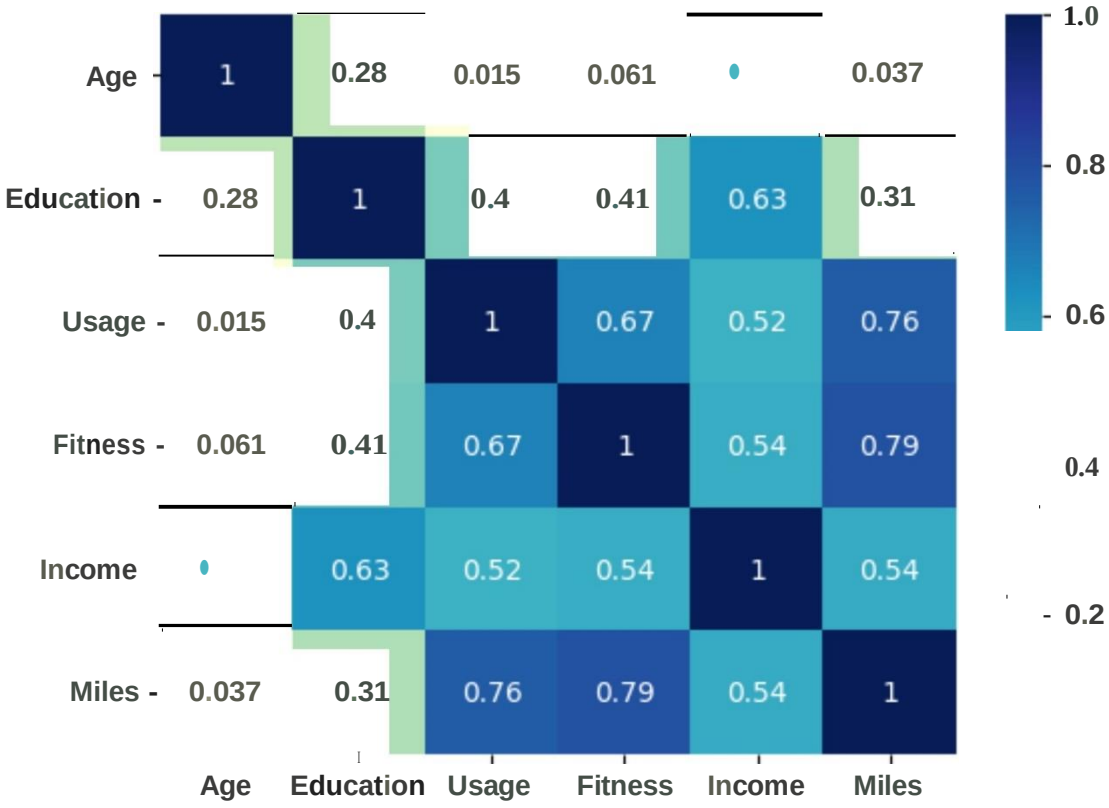
```
cit.columns
```

Out [182...

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

In [181...

```
sns.heatmap(dt.corr(),cmap="YlGnBu", annot=True)
plt.show()
```



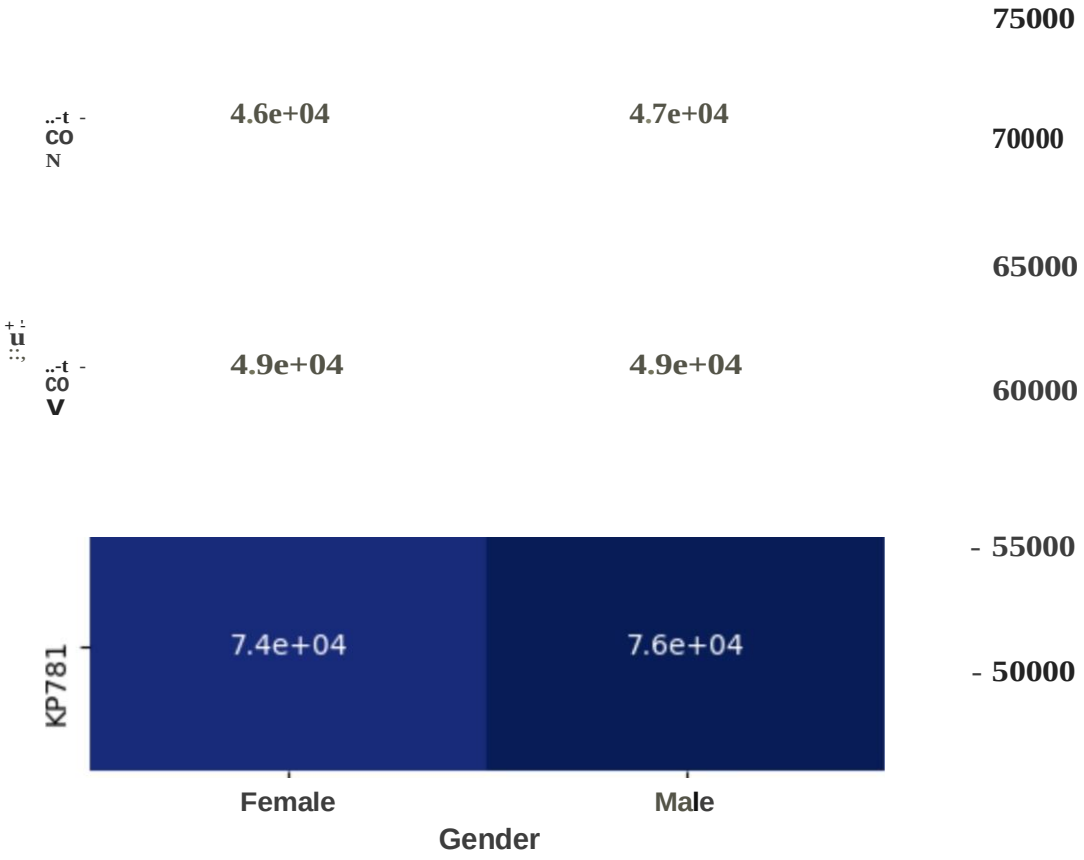
The Features which has correlation -

- Miles - Fitness
- Miles - Usage

- Income - Education

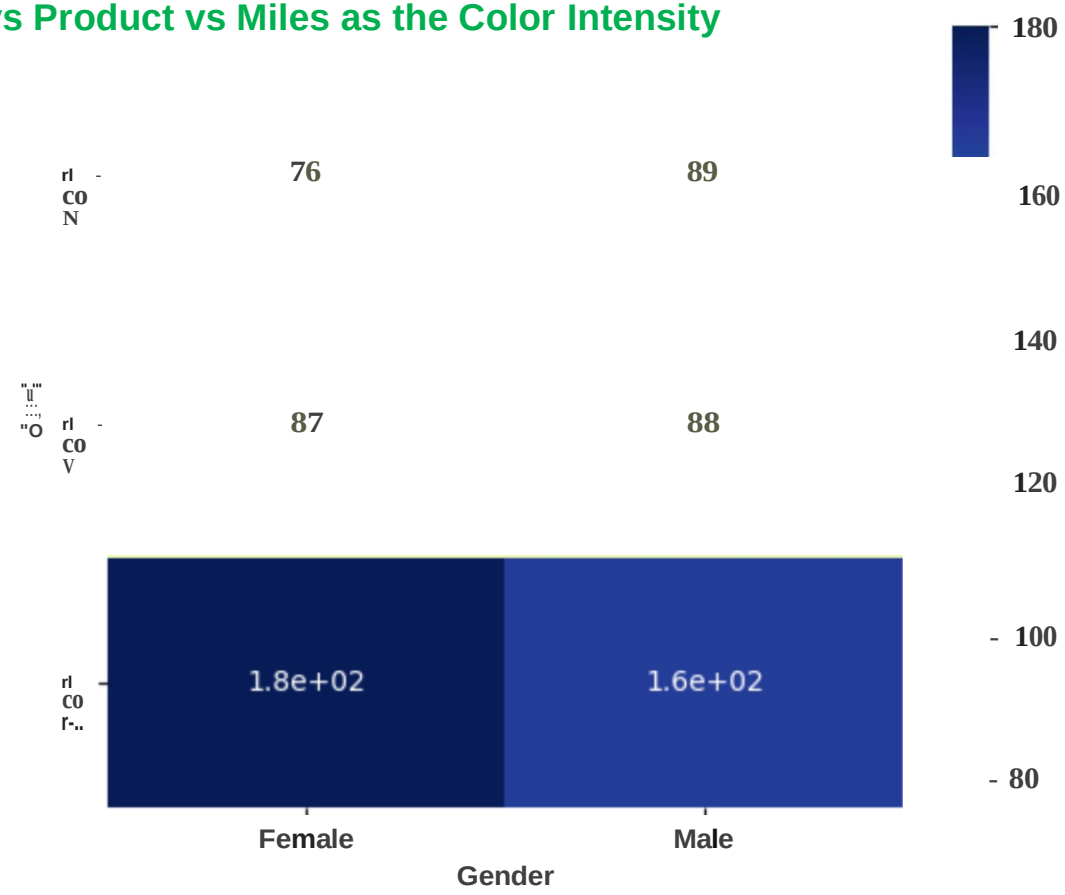
In [11]:
```
sns.heatmap(pd.crosstab(dt['Product'], dt['Gender'], values= dt['Income'], aggfunc = np.mean), cmap = ''VlGnBu", annot
plt.title('Gender vs Product vs Income as the Color Intensity ')
plt.show()
```

Gender vs Product vs Income as the Color Intensity



In [13]:
```
sns.heatmap(pd.crosstab(dt['Product'], dt['Gender'], values= dt['Miles'], aggfunc  = np.mean), cmap  = "VlGnBu", annot=
plt.title('Gender vs Product vs Miles as the Color Intensity ')
plt.show()
```

Gender vs Product vs Miles as the Color Intensity



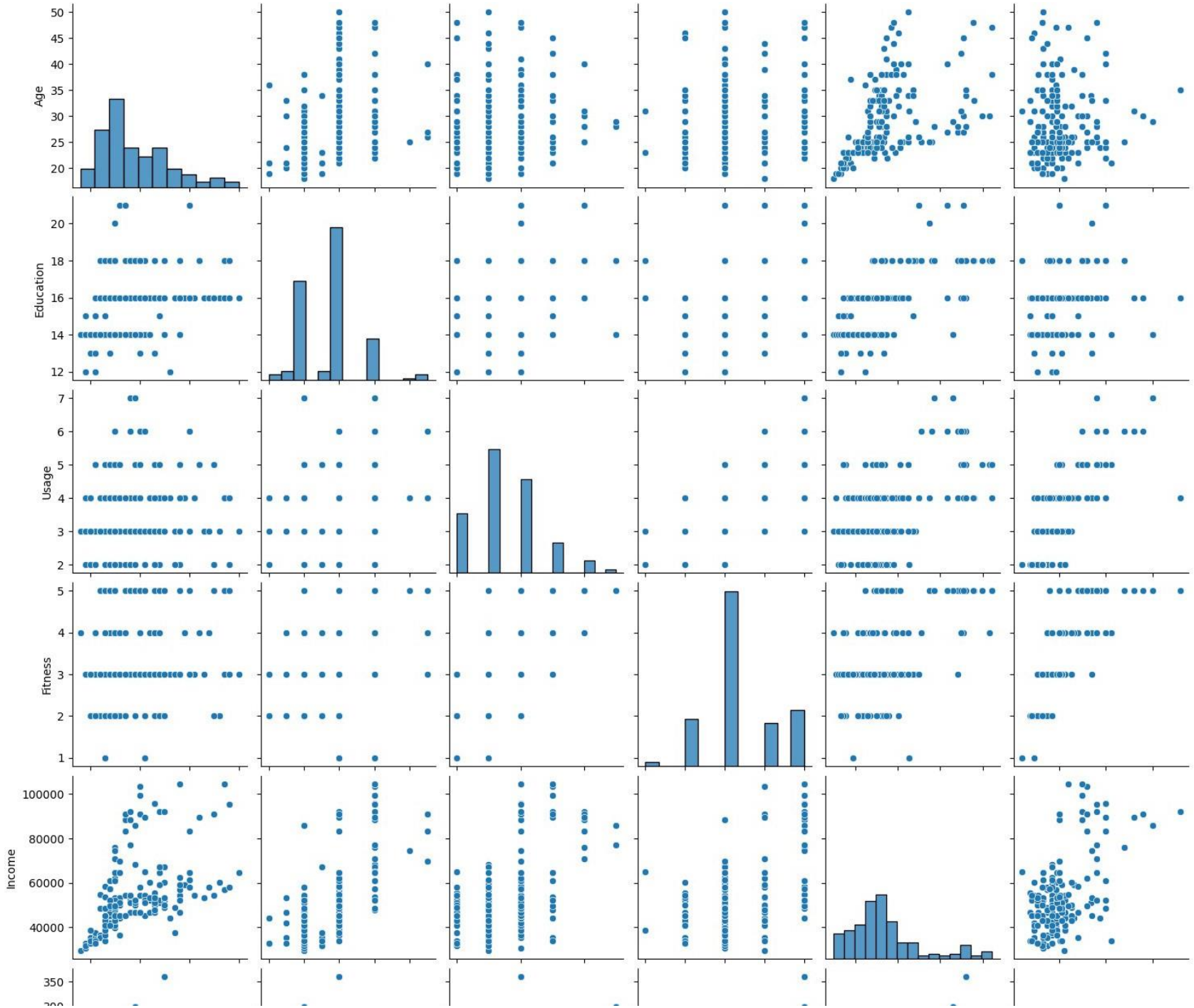
Insights -

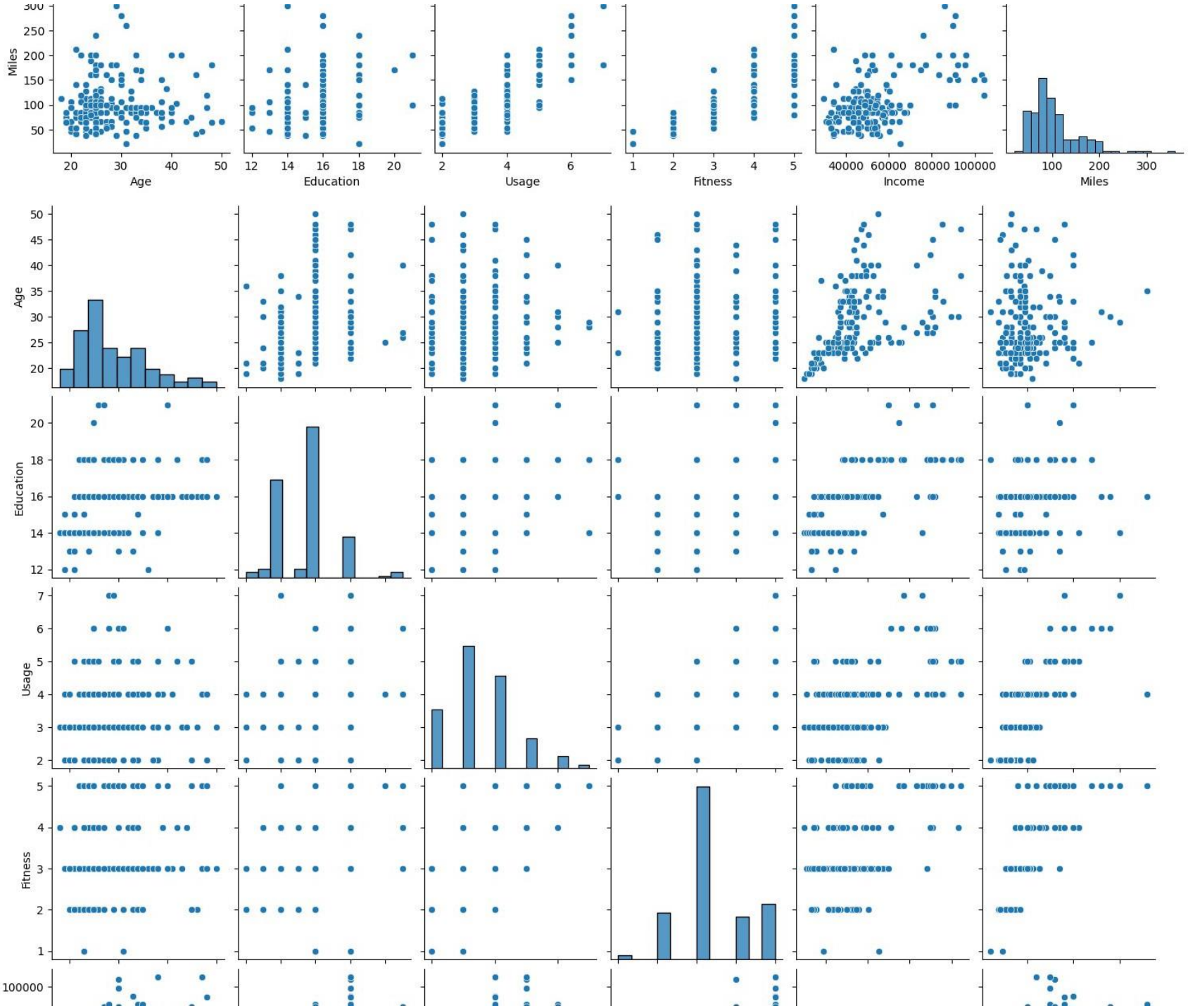
- People who has Higher Income bought KP781 and people who has Lower Income bought KP281
- People who run more Miles bought KP781 over other.

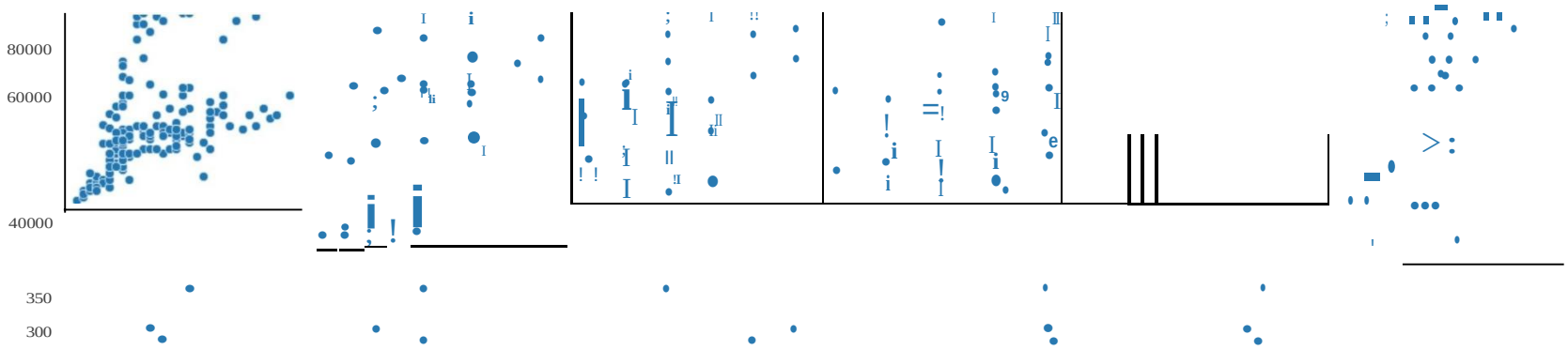
Pair Plot

In [154...

```
sns.pairplot(dt)
plt.show()
```

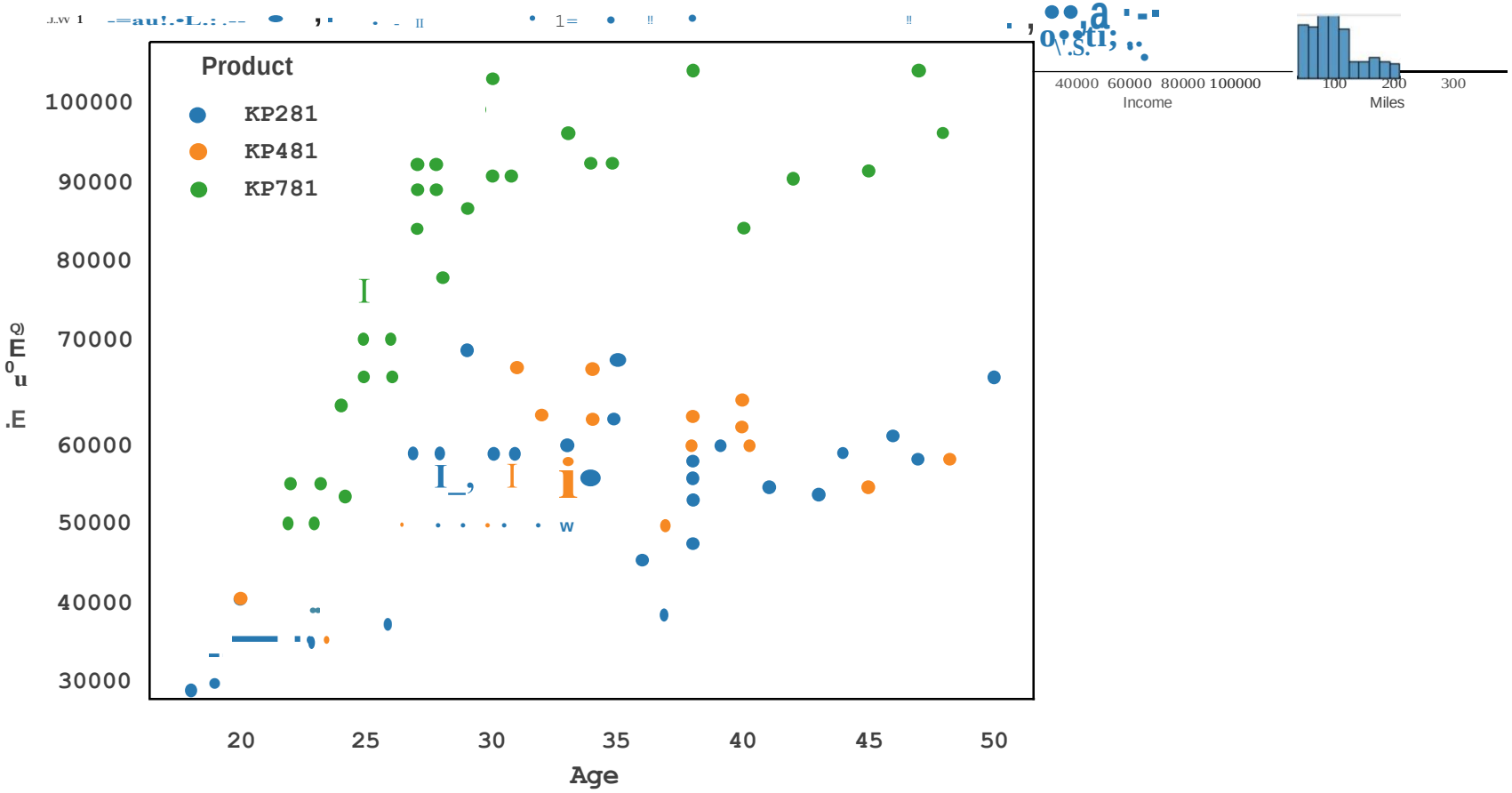

Age
Education
Usage
Fitness
Income
50
45
40
35
30
25
20
18
16
14
12
7
6
5
4
3
2
1
100000
80000
60000
40000
350


Miles
Age
Education
Usage
Fitness
Income
Miles
Age
Education
Usage
Fitness
100000



In [7]:

```
sns.scatterplot(data = dt, x = 'Age', y = 'Income', hue= 'Product')
plt.show()
```



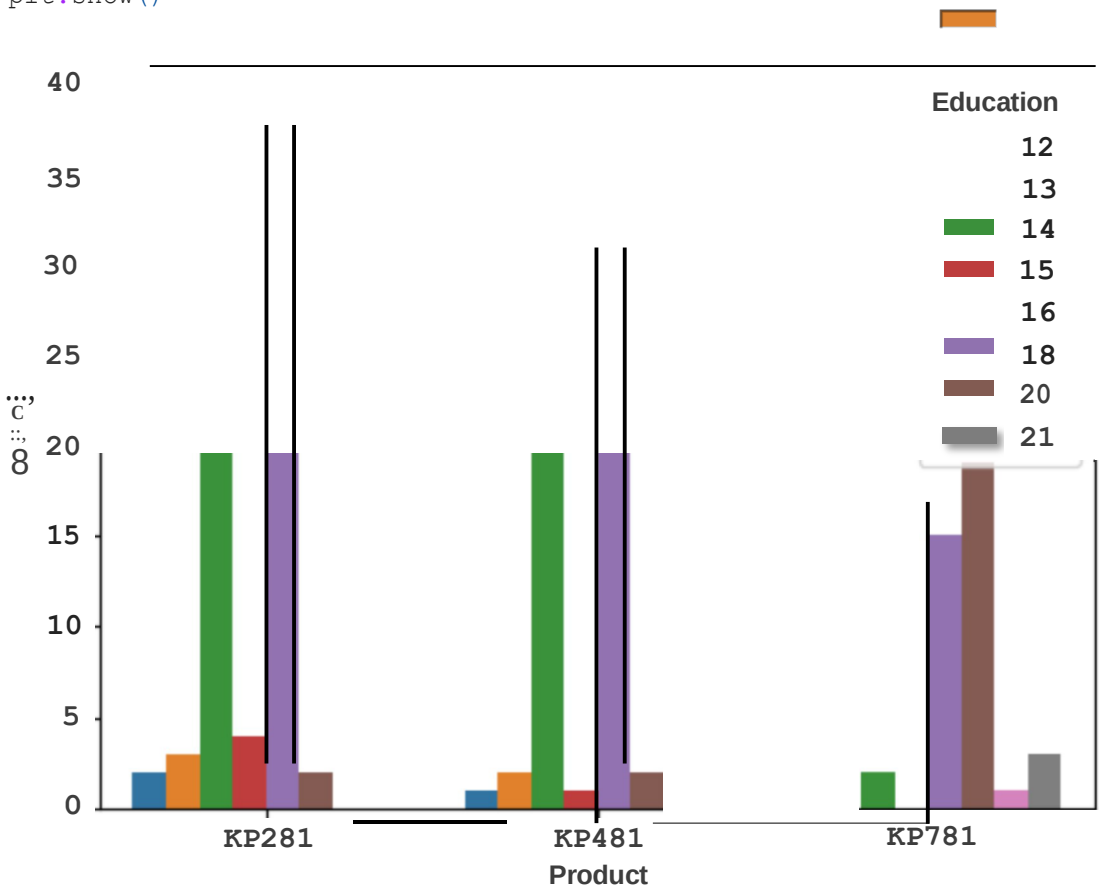
In [34]:
```
dt[dt['Product']== 'KP781']['Income'].min()
```

Out[34]: 48556

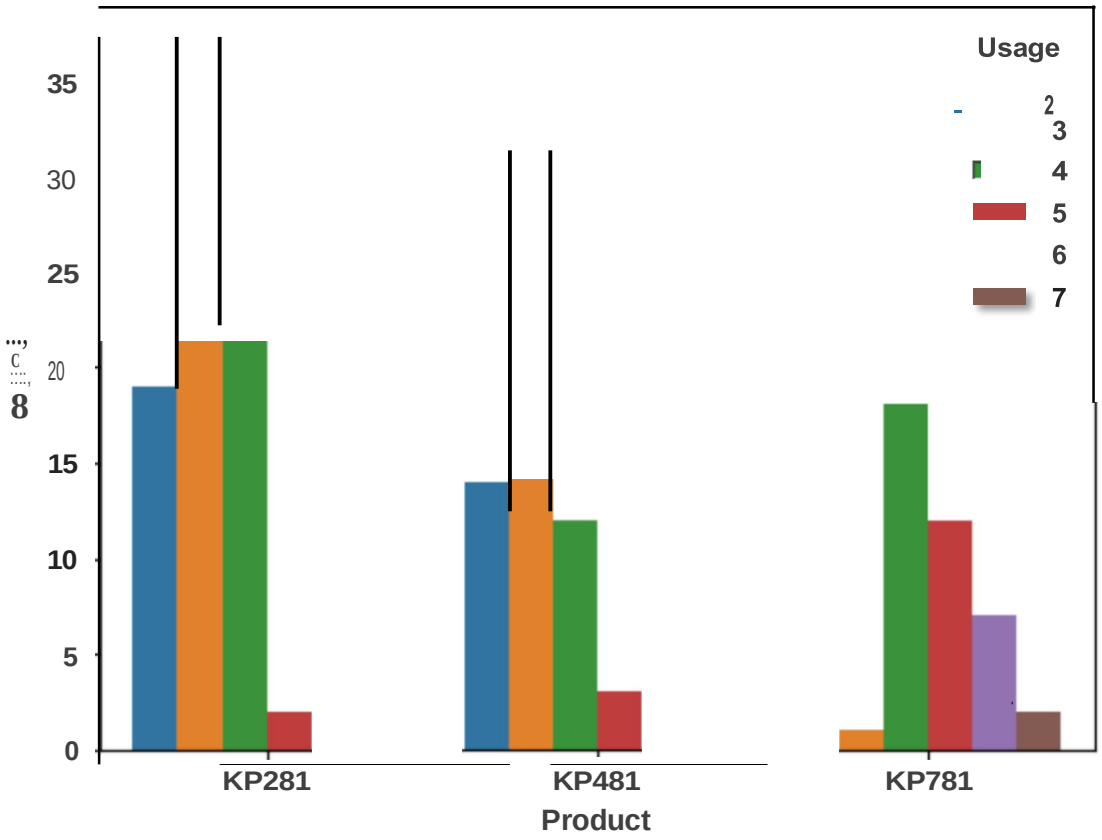
In [39]:
```
cit.columns
```

Out[39]:
```
Index(['Product', 'Age', 'Gender', 'Education', 'MaritalStatus', 'Usage',
       'Fitness', 'Income', 'Miles'],
      dtype='object')
```

In [45]:
```
sns.countplot(data = dt, x = 'Product', hue= 'Education')
plt.show()
```

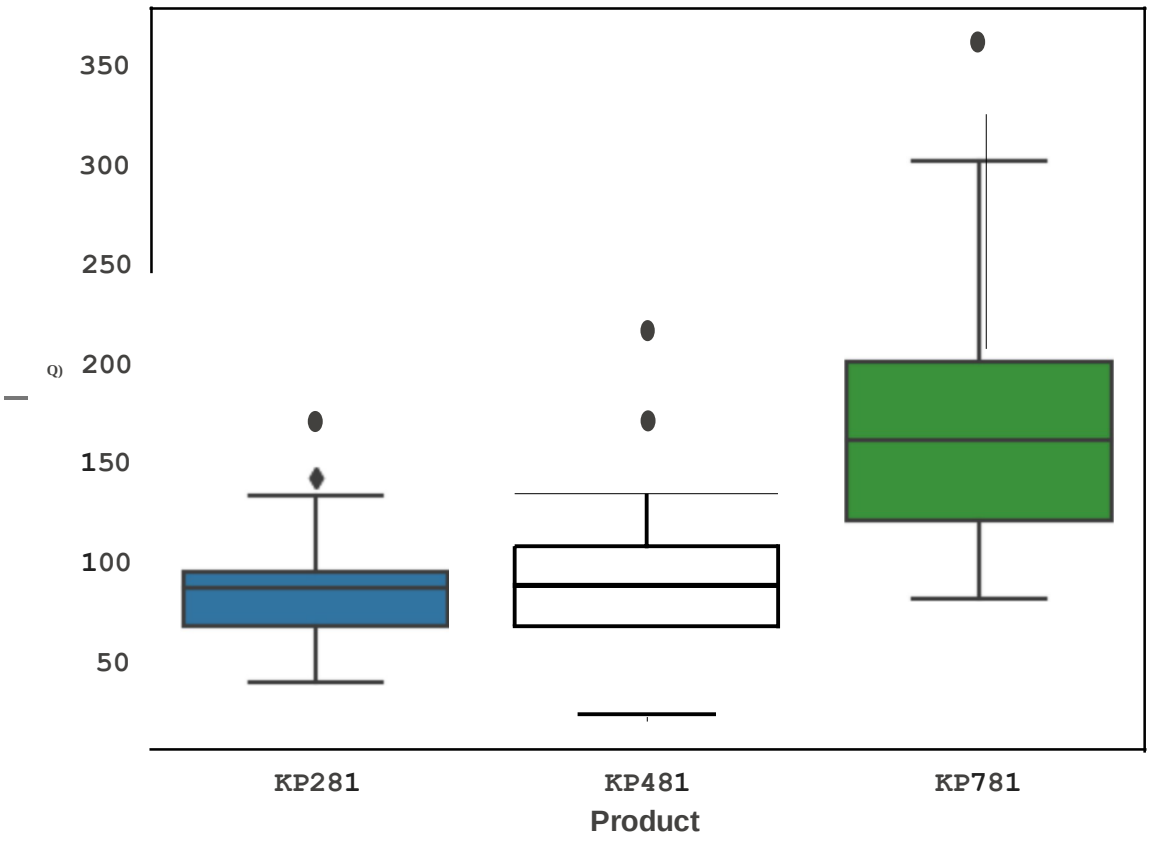


In [46]:
```
sns.countplot(data =  dt, x = 'Product', hue= 'Usage')
plt.show()
```



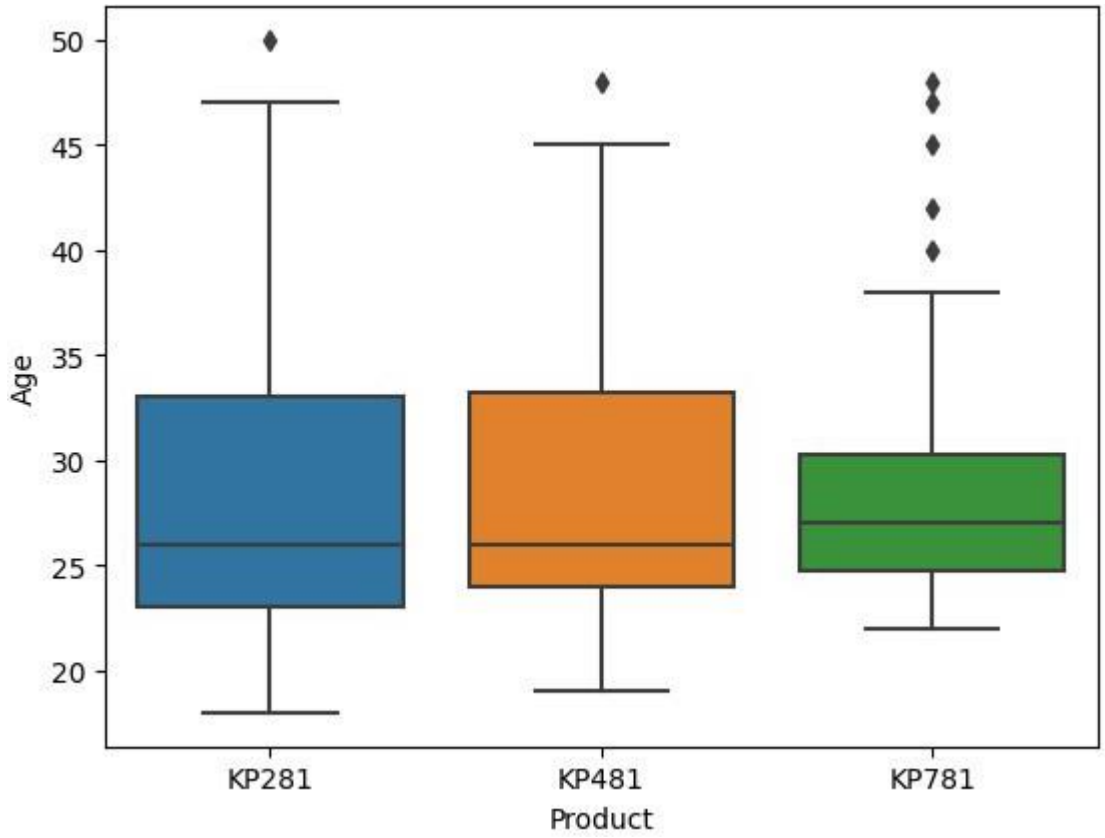
In [53]:

```
sns.boxplot(y = dt['Miles'], x = dt['Product'])
plt.show()
```



In [59]:

```
sns.boxplot(y = dt['Age'], x = dt['Product'])
plt.show()
```



Insights

- No one aged less than 22 bought KP781.
- No one with Salary less than 48556 bought KP781
- All People with Salary > 70000 bought KP781
- Most of the KP281 and KP481 users have 16 years of Education, followed by 14 years, where as most of the KP781 users have 18 years of Education.
- Most of the users of KP281 and KP481 use it for 3 times, where as the Most of the KP781 users use it 4 times and More
- KP781 users tend to run more number of miles compared to other users.

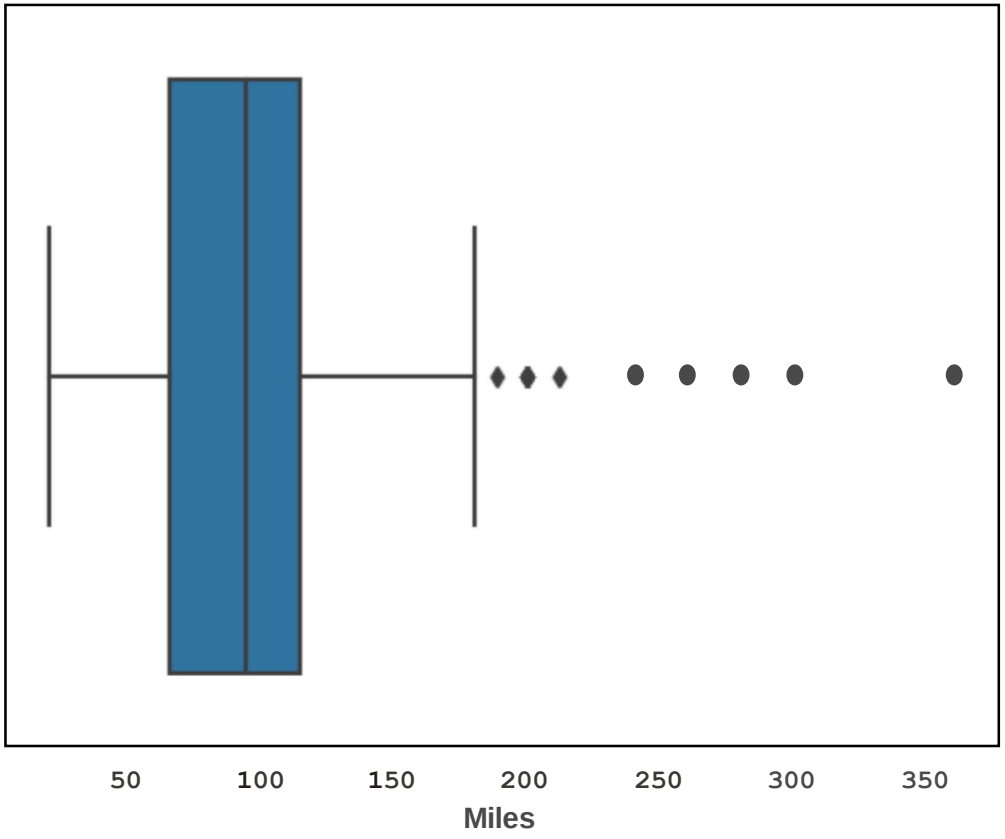
4.Boxplots and Detection Outliers

Miles

In [185...

```
sns.boxplot(dt['Miles'])
plt.show()
```

```
C:\Users\manish\Anaconda3\lib\site-packages\seaborn\_decorators.py:43: FutureWarning: Pass the following variable as a
keyword arg: x. From version 0.12, the only valid positional argument will be 'data', and passing other arguments with
out an explicit keyword will result in an error or misinterpretation.
  FutureWarning
```



In [202..

```
iqr = np.percentile(dt['Miles'], 75) - np.percentile(dt['Miles'], 25)
upper_limit = np.percentile(dt['Miles'], 75) + 1.5*iqr
lower_limit = np.percentile(dt['Miles'], 25) - 1.5*iqr
print('The Upper Limit to Detect the Outliers is=  ', upper_limit)
print('The outliers are', dt[dt['Miles']>upper_limit]['Miles'].values)
```

```
The Upper Limit to Detect the Outliers is     187.875
The outliers are [188 212 200 200 200 240 300 280 260 200 360 200 200]
```

Age

In [83]:

```
sns.boxplot(dt['Age'])
plt.show()
```

```
C:\Users\manish\Anaconda3\lib\site-packages\seaborn\_decorators.py:43: FutureWarning: Pass the following variable as a
keyword arg: x. From version 0.12, the only valid positional argument will be 'data', and passing other arguments with
out an explicit keyword will result in an error or misinterpretation.
  FutureWarning
```

In [203...

```
iqr = np.percentile(dt['Age'], 75) - np.percentile(dt['Age'], 25)
upper_limit = np.percentile(dt['Age'], 75) + 1.5*iqr
lower_limit = np.percentile(dt['Age'], 25) - 1.5*iqr
print('The Upper Limit to Detect the Outliers is=  ', upper_limit)
print('The outliers are', dt[dt['Age']>upper_limit]['Age'].values)
```

```
The Upper Limit to Detect the Outliers is=  46.5
The outliers are [47 50 48 47 48]
```

Income

In [84]:

```
sns.boxplot(dt['Income'])
plt.show()
```

```
C:\Users\manish\Anaconda3\lib\site-packages\seaborn\_decorators.py:43: FutureWarning: Pass the following variable as a
keyword arg: x. From version 0.12, the only valid positional argument will be 'data', and passing other arguments with
out an explicit keyword will result in an error or misinterpretation.
  FutureWarning
```

In [204...

```
iqr = np.percentile(dt['Income'], 75) - np.percentile(dt['Income'], 25)
upper_limit = np.percentile(dt['Income'], 75) + 1.5*iqr
lower_limit = np.percentile(dt['Income'], 25) - 1.5*iqr
print('The Upper Limit to Detect the Outliers is=  ', upper_limit)
print('The outliers  are', dt[dt['Income']>upper_limit]['Income'].values)
```

```
The Upper Limit to Detect the Outliers is=   80581.875
The outliers are [ 83416  88396  90886  92131  88396  85906  90886 103336  99601  89641
  95866  92131  92131 104581  83416  89641  90886 104581  95508]
```

5.

I have written Insights Immediatley after each plot

6.Business Recommendations -

- Most of the people buying the equipments are from 18 to 30 years age group. Since most of the people in this age group are Students, Business can come up with offers and Discounts specific to Students.
- People who use it for more than 4 times a week, most of them have good fitness level too. Since they are serious about Fitness, its good to recommend all advanced fitness equipment to these group of people, they most probably would purchase it.
- Only the People who have Income > 48556 bought KP781. We can increases its number of users by offering discounts to people who have good level of fitness, they might most probably end up buying it.
- Since most the people are from level 3 of fitness, its recommned to come up with products which suits them .
- Similarly People with education level 14, 16 and 18 mostly buy equipments, so its good to target these Group of people