In [1]:

```
import pandas as pd
```

In [2]:

```
df=pd.read_csv('/Users/suraaj/Downloads/aerofit_1.csv')
```

In [3]:

```
df.head()
```

Out[3]:

	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

In [4]:

```
df.shape
```

Out[4]:

```
(180, 9)
```

In [5]:

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

Out[5]:

	Product	Age	Gender	Education	MaritalStatus	Usage	Fitness	
count	180	180.000000	180	180.000000	180	180.000000	180.000000	180
unique	3	NaN	2	NaN	2	NaN	NaN	
top	KP281	NaN	Male	NaN	Partnered	NaN	NaN	
freq	80	NaN	104	NaN	107	NaN	NaN	
mean	NaN	28.788889	NaN	15.572222	NaN	3.455556	3.311111	53719
std	NaN	6.943498	NaN	1.617055	NaN	1.084797	0.958869	16506
min	NaN	18.000000	NaN	12.000000	NaN	2.000000	1.000000	29562
25%	NaN	24.000000	NaN	14.000000	NaN	3.000000	3.000000	44058
50%	NaN	26.000000	NaN	16.000000	NaN	3.000000	3.000000	50596
75%	NaN	33.000000	NaN	16.000000	NaN	4.000000	4.000000	58668
max	NaN	50.000000	NaN	21.000000	NaN	7.000000	5.000000	104581

In [6]:

```
df.isnull().sum()/len(df)*100
```

Out[6]:

```
Product          0.0
Age              0.0
Gender           0.0
Education        0.0
MaritalStatus    0.0
Usage            0.0
Fitness          0.0
Income           0.0
Miles            0.0
dtype: float64
```

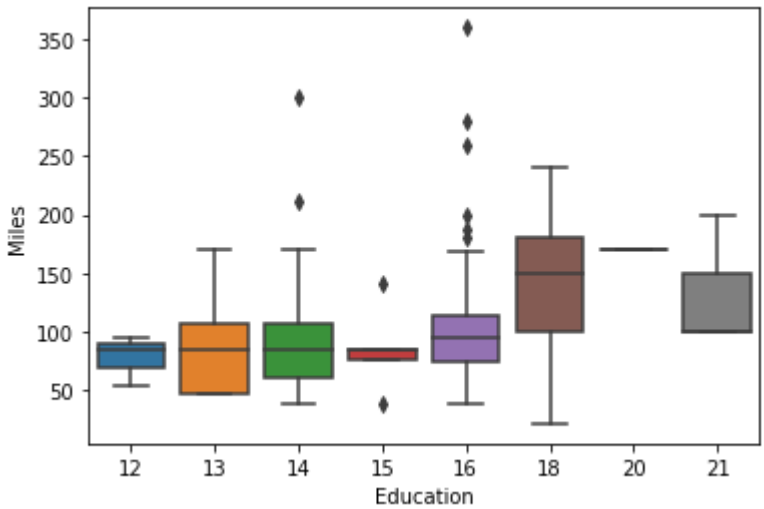
In [7]:

```
#bivariate analysis
import seaborn as sbn

sbn.boxplot(x='Education', y='Miles', data=df)
```

Out[7]:

```
<AxesSubplot:xlabel='Education', ylabel='Miles'>
```

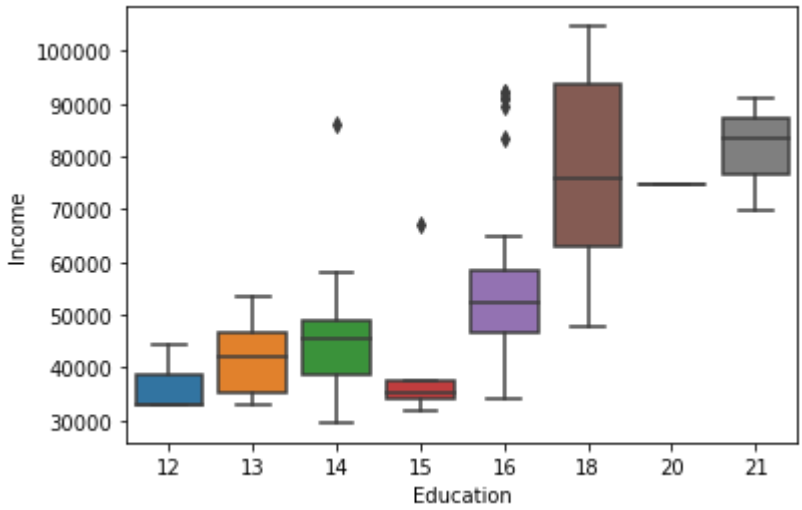


In [8]:

```
sbn.boxplot(x='Education', y='Income', data=df)
```

Out[8]:

```
<AxesSubplot:xlabel='Education', ylabel='Income'>
```

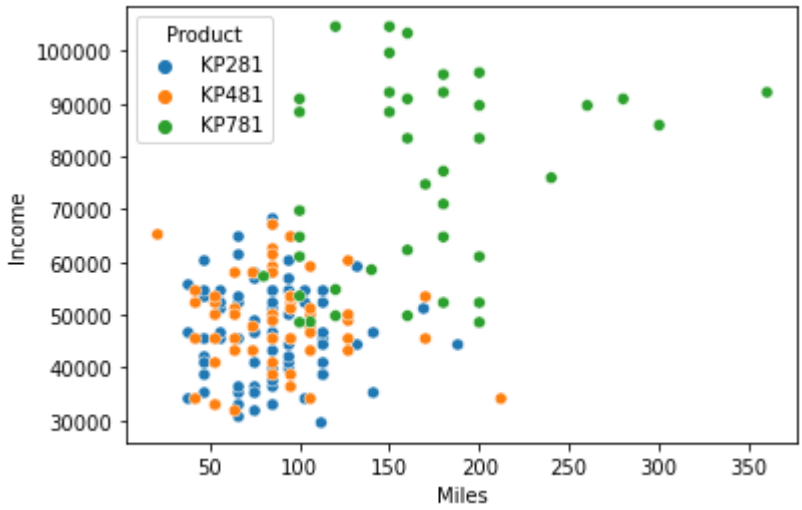


In [9]:

```
sbn.scatterplot(x='Miles', y='Income', data=df, hue='Product')
```

Out[9]:

```
<AxesSubplot:xlabel='Miles', ylabel='Income'>
```

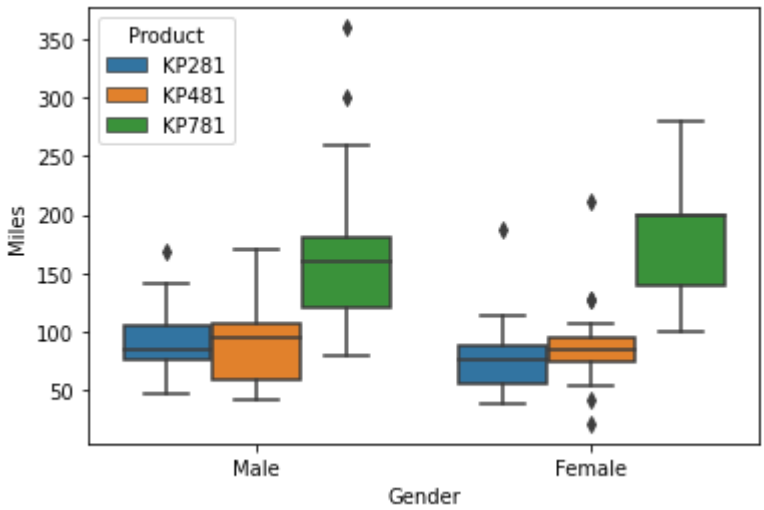


In [10]:

```
sbn.boxplot(x='Gender', y='Miles', data=df, hue='Product')
```

Out[10]:

```
<AxesSubplot:xlabel='Gender', ylabel='Miles'>
```



In [11]:

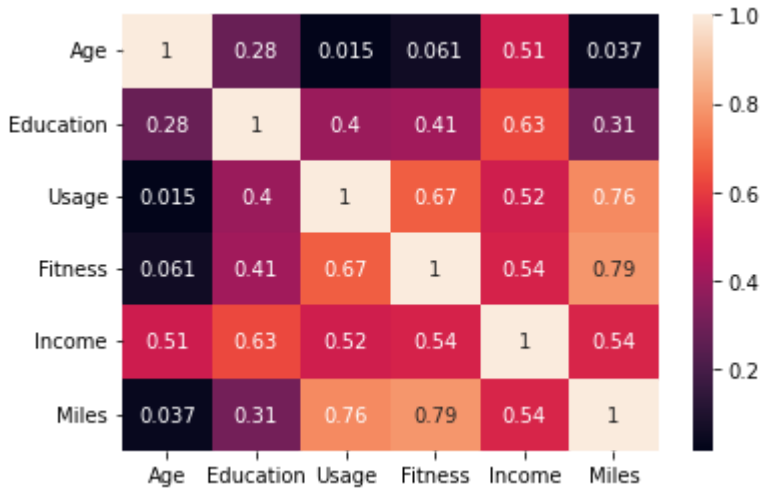
```
#IQR (optional)
```

In [12]:

```
sbn.heatmap(df.corr(), annot=True)
```

Out[12]:

<AxesSubplot:>



for i in ['Income','Miles']: q1=df[i].quantile(0.25) q3=df[i].quantile(0.75) iqr=q3-q1

```
    df=df[(df[i]>=q1-1.5*iqr) & (df[i]<=q3+1.5*iqr)]
```

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

sbn.scatterplot(x='Miles', y='Income', data=df)

sbn.boxplot(x='Education', y='Income', data=df)

sbn.boxplot(x='Education', y='Miles', data=df)

In [13]:

```
# pie chart  % contribution of each product
```

In [14]:

```
#marginal and the conditional probability test

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

Out[14]:

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

In [15]:

```
pd.crosstab(index=df['Gender'], columns=df['Product'], margins=True)
```

Out[15]:

Product	KP281	KP481	KP781	All
Gender				
Female	40	29	7	76
Male	40	31	33	104
All	80	60	40	180

In [16]:

```
pd.crosstab(index=df['Gender'], columns=df['Product'], margins=True, normalize=True)
```

Out[16]:

Product	KP281	KP481	KP781	All
Gender				
Female	0.222222	0.161111	0.038889	0.422222
Male	0.222222	0.172222	0.183333	0.577778
All	0.444444	0.333333	0.222222	1.000000

In [17]:

```
#conditional
pd.crosstab(index=df['Gender'], columns=df['Product'], margins=True, normalize='inde
```

Out[17]:

Product	KP281	KP481	KP781
Gender			
Female	0.526316	0.381579	0.092105
Male	0.384615	0.298077	0.317308
All	0.444444	0.333333	0.222222

In [18]:

```
#P(KP281|FEMALE)=0.5263
```

In [19]:

```
#conditional
pd.crosstab(index=df['Gender'], columns=df['Product'], margins=True, normalize='colu
```

Out[19]:

Product	KP281	KP481	KP781	All
Gender				
Female	0.5	0.483333	0.175	0.422222
Male	0.5	0.516667	0.825	0.577778

In [20]:

```
#P(FEMALE|KP781)=0.0.175
```