Business Case-Aerofit-Probability and Descriptive Statistics

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

```
df_aerofit=pd.read_csv('/content/aerofit_treadmill.csv')
df_aerofit
```

	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

Next steps: Generate code with df_aerofit | View recommended plots | New interactive sheet

```
df_aerofit.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_aerofit.tail()
```

	Product	Age	Gender	Education	MaritalStatus	Usage	Fitness	Income	Miles
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

Double-click (or enter) to edit

```
df_aerofit.shape
```

```
(180, 9)
```

1.We can see that the number of rows and columns in the above dataset is-180 and 9.

```
df_aerofit.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
```

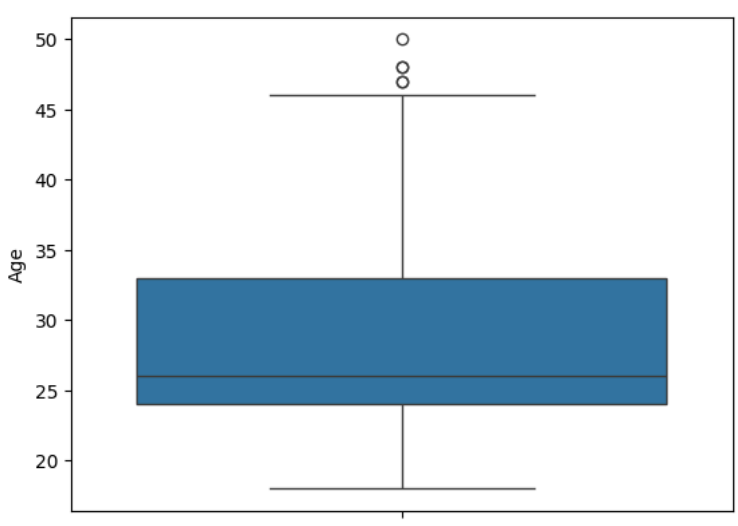
*From the above observation we observed that the columns do not have any null values.

```
df_aerofit.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

- Find the outliers for every continuous variables in the dataset.

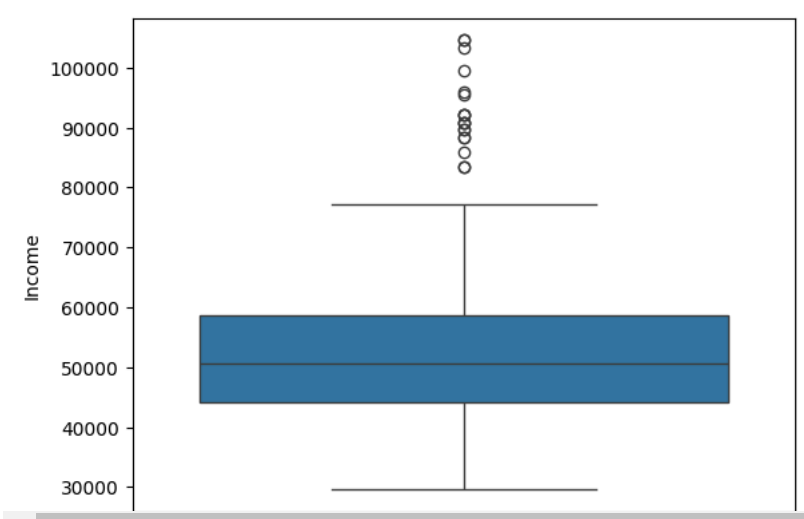
```
a=sns.boxplot(df_aerofit['Age'])
```




```
q1=df_aerofit['Age'].quantile(0.25)
q3=df_aerofit['Age'].quantile(0.75)
IQR=q3-q1
df_aerofit[(df_aerofit['Age']<(q1-1.5*IQR)) | (df_aerofit['Age']>(q3+1.5*IQR))]
```

	Product	Age	Gender	Education	MaritalStatus	Usage	Fitness	Income	Miles
78	KP281	47	Male	16	Partnered	4	3	56850	94
79	KP281	50	Female	16	Partnered	3	3	64809	66
139	KP481	48	Male	16	Partnered	2	3	57987	64
178	KP781	47	Male	18	Partnered	4	5	104581	120
179	KP781	48	Male	18	Partnered	4	5	95508	180

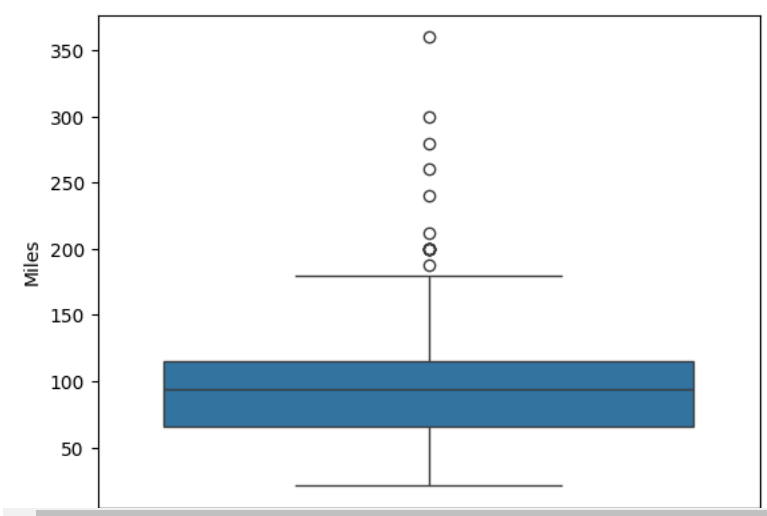
```
b=sns.boxplot(df_aerofit['Income'])
```




```
q1=df_aerofit['Income'].quantile(0.25)
q3=df_aerofit['Income'].quantile(0.75)
IQR=q3-q1
df_aerofit[(df_aerofit['Income']<(q1-1.5*IQR)) | (df_aerofit['Income']>(q3+1.5*IQR))]
```

	Product	Age	Gender	Education	MaritalStatus	Usage	Fitness	Income	Miles
159	KP781	27	Male	16	Partnered	4	5	83416	160
160	KP781	27	Male	18	Single	4	3	88396	100
161	KP781	27	Male	21	Partnered	4	4	90886	100
162	KP781	28	Female	18	Partnered	6	5	92131	180
164	KP781	28	Male	18	Single	6	5	88396	150
166	KP781	29	Male	14	Partnered	7	5	85906	300
167	KP781	30	Female	16	Partnered	6	5	90886	280
168	KP781	30	Male	18	Partnered	5	4	103336	160
169	KP781	30	Male	18	Partnered	5	5	99601	150
170	KP781	31	Male	16	Partnered	6	5	89641	260
171	KP781	33	Female	18	Partnered	4	5	95866	200
172	KP781	34	Male	16	Single	5	5	92131	150
173	KP781	35	Male	16	Partnered	4	5	92131	360
174	KP781	38	Male	18	Partnered	5	5	104581	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
178	KP781	47	Male	18	Partnered	4	5	104581	120
179	KP781	48	Male	18	Partnered	4	5	95508	180

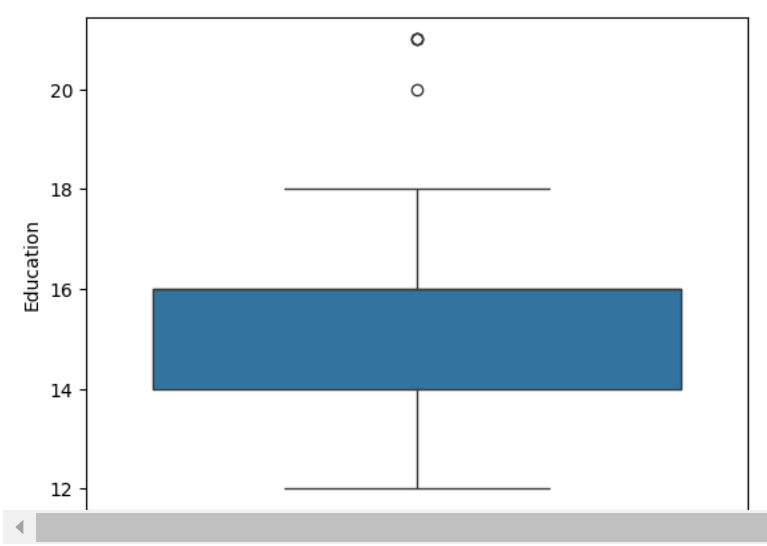
```
c=sns.boxplot(df_aerofit['Miles'])
```




```
q1=df_aerofit['Miles'].quantile(0.25)
q3=df_aerofit['Miles'].quantile(0.75)
IQR=q3-q1
df_aerofit[(df_aerofit['Miles']<(q1-1.5*IQR)) | (df_aerofit['Miles']>(q3+1.5*IQR))]
```

	Product	Age	Gender	Education	MaritalStatus	Usage	Fitness	Income	Miles
23	KP281	24	Female	16	Partnered	5	5	44343	188
84	KP481	21	Female	14	Partnered	5	4	34110	212
142	KP781	22	Male	18	Single	4	5	48556	200
148	KP781	24	Female	16	Single	5	5	52291	200
152	KP781	25	Female	18	Partnered	5	5	61006	200
155	KP781	25	Male	18	Partnered	6	5	75946	240
166	KP781	29	Male	14	Partnered	7	5	85906	300
167	KP781	30	Female	16	Partnered	6	5	90886	280
170	KP781	31	Male	16	Partnered	6	5	89641	260
171	KP781	33	Female	18	Partnered	4	5	95866	200
173	KP781	35	Male	16	Partnered	4	5	92131	360
175	KP781	40	Male	21	Single	6	5	83416	200
176	KP781	42	Male	18	Single	5	4	89641	200

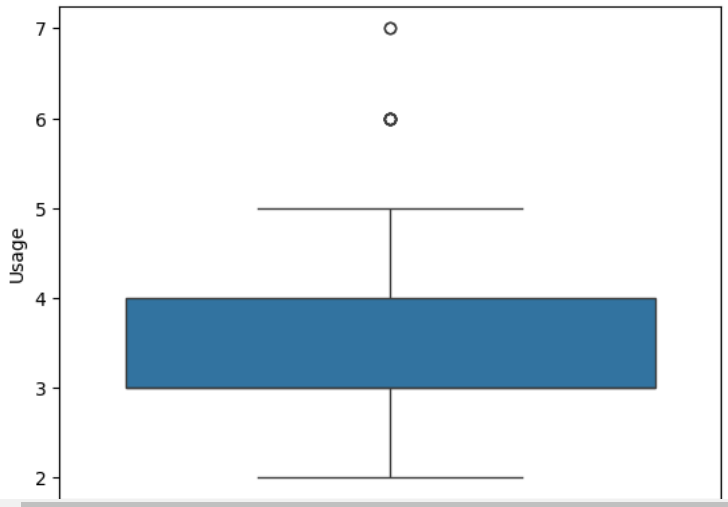
```
d=sns.boxplot(df_aerofit['Education'])
```




```
q1=df_aerofit['Education'].quantile(0.25)
q3=df_aerofit['Education'].quantile(0.75)
IQR=q3-q1
df_aerofit[(df_aerofit['Education']<(q1-1.5*IQR)) | (df_aerofit['Education']>(q3+1.5*IQR))]
```

	Product	Age	Gender	Education	MaritalStatus	Usage	Fitness	Income	Miles
156	KP781	25	Male	20	Partnered	4	5	74701	170
157	KP781	26	Female	21	Single	4	3	69721	100
161	KP781	27	Male	21	Partnered	4	4	90886	100
175	KP781	40	Male	21	Single	6	5	83416	200

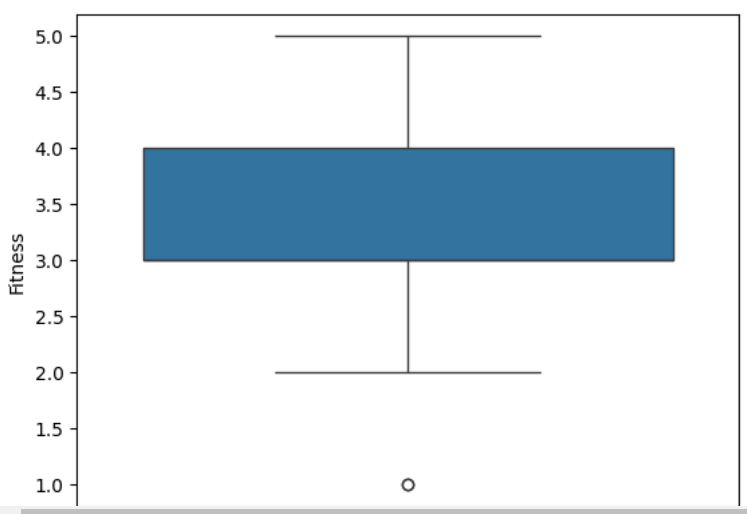
```
e=sns.boxplot(df_aerofit['Usage'])
```




```
q1=df_aerofit['Usage'].quantile(0.25)
q3=df_aerofit['Usage'].quantile(0.75)
IQR=q3-q1
df_aerofit[(df_aerofit['Usage']<(q1-1.5*IQR)) | (df_aerofit['Usage']>(q3+1.5*IQR))]
```

	Product	Age	Gender	Education	MaritalStatus	Usage	Fitness	Income	Miles
154	KP781	25	Male	18	Partnered	6	4	70966	180
155	KP781	25	Male	18	Partnered	6	5	75946	240
162	KP781	28	Female	18	Partnered	6	5	92131	180
163	KP781	28	Male	18	Partnered	7	5	77191	180
164	KP781	28	Male	18	Single	6	5	88396	150
166	KP781	29	Male	14	Partnered	7	5	85906	300
167	KP781	30	Female	16	Partnered	6	5	90886	280
170	KP781	31	Male	16	Partnered	6	5	89641	260
175	KP781	40	Male	21	Single	6	5	83416	200

```
e=sns.boxplot(df_aerofit['Fitness'])
```



```
q1=df_aerofit['Fitness'].quantile(0.25)
q3=df_aerofit['Fitness'].quantile(0.75)
IQR=q3-q1
df_aerofit[(df_aerofit['Fitness']<(q1-1.5*IQR)) | (df_aerofit['Fitness']>(q3+1.5*IQR))]
```

	Product	Age	Gender	Education	MaritalStatus	Usage	Fitness	Income	Miles
14	KP281	23	Male	16	Partnered	3	1	38658	47
117	KP481	31	Female	18	Single	2	1	65220	21

Clipping the data-

```
minn=np.percentile(df_aerofit['Age'],5)
maxx=np.percentile(df_aerofit['Age'],95)
df_aerofit['Age']=np.clip(df_aerofit['Age'],minn,maxx)
```

```
minn1=np.percentile(df_aerofit['Income'],5)
maxx1=np.percentile(df_aerofit['Income'],95)
df_aerofit['Income']=np.clip(df_aerofit['Income'],minn1,maxx1)
```

```
minn2=np.percentile(df_aerofit['Miles'],5)
maxx2=np.percentile(df_aerofit['Miles'],95)
df_aerofit['Miles']=np.clip(df_aerofit['Miles'],minn2,maxx2)
```

```
minn3=np.percentile(df_aerofit['Education'],5)
maxx3=np.percentile(df_aerofit['Education'],95)
df_aerofit['Education']=np.clip(df_aerofit['Education'],minn3,maxx3)
```

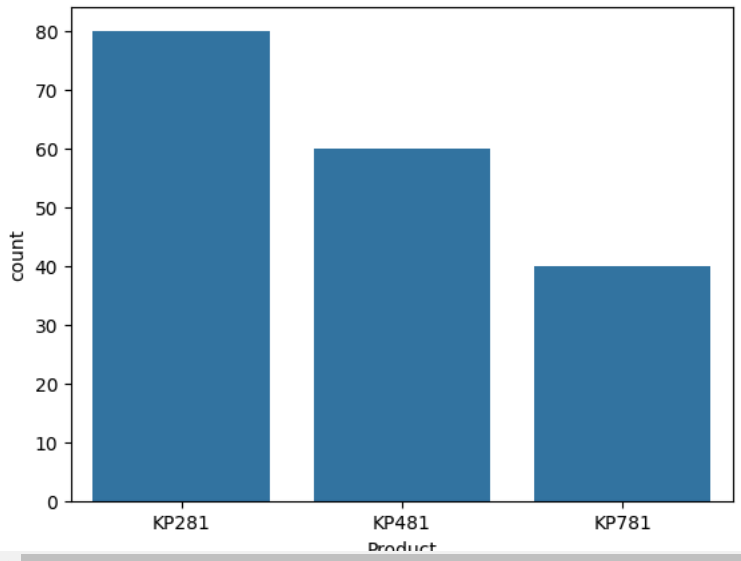
```
minn4=np.percentile(df_aerofit['Usage'],5)
maxx4=np.percentile(df_aerofit['Usage'],95)
df_aerofit['Usage']=np.clip(df_aerofit['Usage'],minn4,maxx4)
```

```
minn5=np.percentile(df_aerofit['Fitness'],5)
maxx5=np.percentile(df_aerofit['Fitness'],95)
df_aerofit['Fitness']=np.clip(df_aerofit['Fitness'],minn5,maxx5)
```

Univariate Analysis

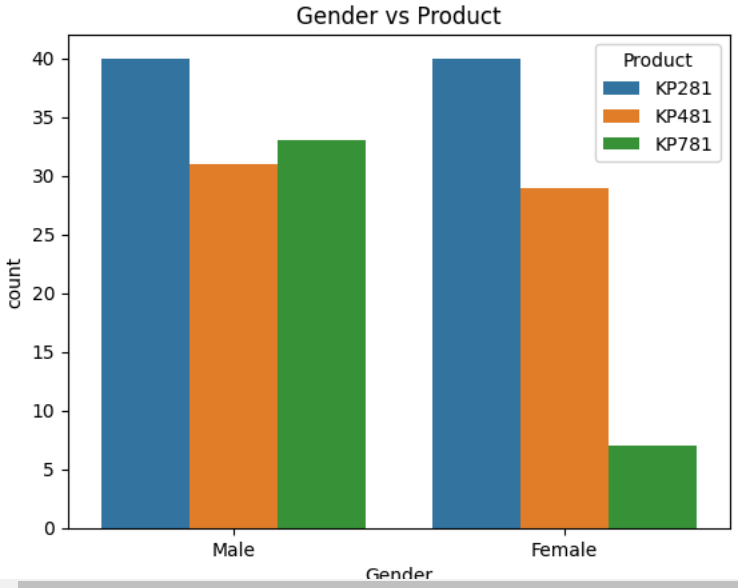
```
sns.countplot(data=df_aerofit,x='Product')
```

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

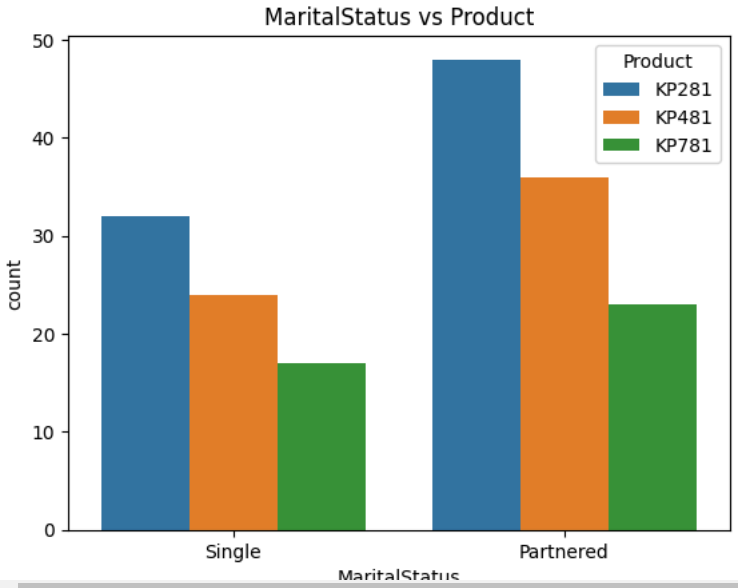


Bivariate Analysis

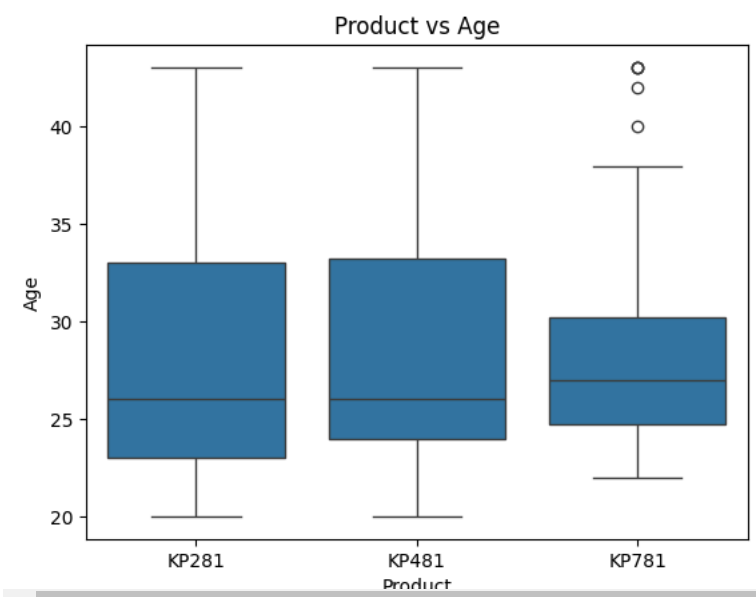
```
sns.countplot(data=df_aerofit,x='Gender',hue='Product')
plt.title('Gender vs Product')
plt.show()
```



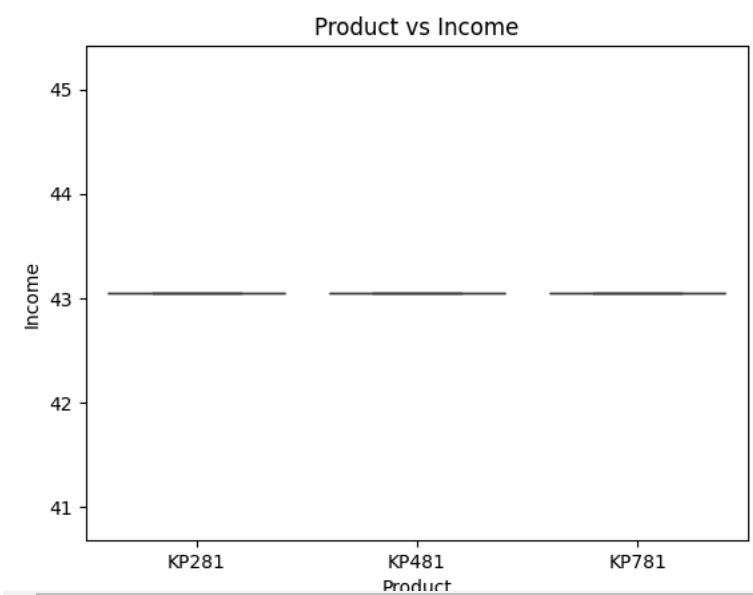

```
sns.countplot(data=df_aerofit,x='MaritalStatus',hue='Product')
plt.title('MaritalStatus vs Product')
plt.show()
```



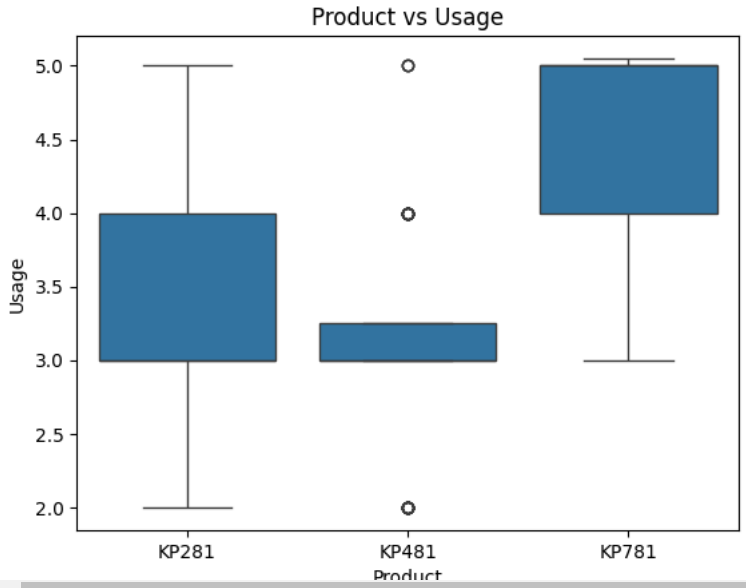

```
sns.boxplot(data=df_aerofit,x='Product', y='Age')
plt.title('Product vs Age')
plt.show()
```




```
sns.boxplot(data=df_aerofit,x='Product', y='Income')
plt.title('Product vs Income')
plt.show()
```




```
sns.boxplot(data=df_aerofit,x='Product', y='Usage')
plt.title('Product vs Usage')
plt.show()
```



Representing the Probabilities

```
df_aerofit.head()
```

	Product	Age	Gender	Education	MaritalStatus	Usage	Fitness	Income	Miles
0	KP281	20.0	Male	14	Single	3.0	4	43.05	112
1	KP281	20.0	Male	15	Single	2.0	3	43.05	75
2	KP281	20.0	Female	14	Partnered	4.0	3	43.05	66
3	KP281	20.0	Male	14	Single	3.0	3	43.05	85
4	KP281	20.0	Male	14	Partnered	4.0	2	43.05	47

```
crosstab1=pd.crosstab(index=df_aerofit['Product'],columns='count',normalize=True)
crosstab1
```

col_0	count
Product	
KP281	0.444444
KP481	0.333333
KP781	0.222222

Products and Gender

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

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

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

Product	KP281	KP481	KP781	All

Conditional Probability

Single	0.177778	0.133333	0.094444	0.405556

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

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

P(using KP281|Female)=0.22 P(using KP481|Female)=0.1611 P(using KP781|Female)=0.03 P(using KP281|Male)=0.22 P(using KP481|Male)=0.17 P(using KP781|Male)=0.18

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

Product	KP281	KP481	KP781	All
MaritalStatus				
Partnered	0.266667	0.200000	0.127778	0.594444
Single	0.177778	0.133333	0.094444	0.405556
All	0.444444	0.333333	0.222222	1.000000

P(KP281|Partnered)=0.26 P(KP481|Partnered)=0.20 P(KP781|Partnered)=0.12 P(KP281|Single)=0.17 P(KP481|Single)=0.13 P(KP781|Single)=0.09

Correlation

```
df_aerofit.head()
```

	Product	Age	Gender	Education	MaritalStatus	Usage	Fitness	Income	Miles
0	KP281	20.0	Male	14	Single	3.0	4	43.05	112
1	KP281	20.0	Male	15	Single	2.0	3	43.05	75
2	KP281	20.0	Female	14	Partnered	4.0	3	43.05	66
3	KP281	20.0	Male	14	Single	3.0	3	43.05	85
4	KP281	20.0	Male	14	Partnered	4.0	2	43.05	47

Customer Profiling

KP281

*Age- 23 to 33 *Income-38000 to 54000 *Usage-3 to 4 times a week *Martial Status-Partenered are most likely to use KP281 *Gender-Both Male and Female are equally using KP281