AEROFIT Case-Study

In [3]:

```
# Importing libraries

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from scipy.stats import norm
```

In [4]:

```
# Importing dataset

!gdown 119irQdTR8exno8jMdpg9BzPiOFPV0WgN
```

```
Downloading...
From: https://drive.google.com/uc?id=119irQdTR8exno8jMdpg9BzPiOFPV0WgN (ht
tps://drive.google.com/uc?id=119irQdTR8exno8jMdpg9BzPiOFPV0WgN)
To: C:\Users\Admin\Prob and Stat - Aerofit dataset.csv

  0%|          | 0.00/7.28k [00:00<?, ?B/s]
100%|##########| 7.28k/7.28k [00:00<?, ?B/s]
```

In [5]:

```
df = pd.read_csv("C:/Users/Admin/Prob and Stat - Aerofit dataset.csv")
df
```

Out[5]:

	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

In [22]:

```
# Let's understand our data

df.info()
df.describe(include = "all")
```

```
<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
```

Out[22]:

	Product	Age	Gender	Education	MaritalStatus	Usage	Fitness	
count	180	180.000000	180	180.000000	180	180.000000	180.000000	
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	53
std	NaN	6.943498	NaN	1.617055	NaN	1.084797	0.958869	16
min	NaN	18.000000	NaN	12.000000	NaN	2.000000	1.000000	29
25%	NaN	24.000000	NaN	14.000000	NaN	3.000000	3.000000	44
50%	NaN	26.000000	NaN	16.000000	NaN	3.000000	3.000000	50
75%	NaN	33.000000	NaN	16.000000	NaN	4.000000	4.000000	58
max	NaN	50.000000	NaN	21.000000	NaN	7.000000	5.000000	104

In [34]:

```
# Unique values in product column

df["Product"].value_counts()
```

Out[34]:

```
KP281    80
KP481    60
KP781    40
Name: Product, dtype: int64
```

In [53]:

```
# Minimum age in Age column
min_age = df["Age"].min()

# Maximum age in Age column
max_age = df["Age"].max()

min_age, max_age
```

Out[53]:

```
(18, 50)
```

In [52]:

```
# Minimum age of Male
male_min_age = df[df["Gender"]=="Male"]["Age"].min()

# Maximum age of Male
male_max_age = df[df["Gender"]=="Male"]["Age"].max()

male_min_age, male_max_age
```

Out[52]:

```
(18, 48)
```

In [50]:

```
# Minimum age of Female
female_min_age = df[df["Gender"]=="Female"]["Age"].min()

# Maximum age of Female
female_max_age = df[df["Gender"]=="Female"]["Age"].max()

female_min_age, female_max_age
```

Out[50]:

```
(19, 50)
```

In [64]:

```
pd.crosstab(index = df["Age"], columns = [df["Gender"], df["Product"]], values = df["Pro
```

Out[64]:

Gender	Female			Male			All
Product	**KP281**	**KP481**	**KP781**	**KP281**	**KP481**	**KP781**	
Age							
18	NaN	NaN	NaN	1.0	NaN	NaN	1
19	1.0	NaN	NaN	2.0	1.0	NaN	4
20	1.0	1.0	NaN	1.0	2.0	NaN	5
21	2.0	1.0	NaN	2.0	2.0	NaN	7
22	3.0	NaN	NaN	1.0	NaN	3.0	7
23	3.0	3.0	1.0	5.0	4.0	2.0	18
24	3.0	2.0	1.0	2.0	1.0	3.0	12
25	4.0	5.0	1.0	3.0	6.0	6.0	25
26	3.0	2.0	1.0	4.0	1.0	1.0	12
27	2.0	NaN	NaN	1.0	1.0	3.0	7
28	4.0	NaN	1.0	2.0	NaN	2.0	9
29	2.0	1.0	NaN	1.0	NaN	2.0	6
30	NaN	2.0	1.0	2.0	NaN	2.0	7
31	1.0	2.0	NaN	1.0	1.0	1.0	6
32	1.0	NaN	NaN	1.0	2.0	NaN	4
33	2.0	3.0	1.0	NaN	2.0	NaN	8
34	1.0	1.0	NaN	1.0	2.0	1.0	6
35	2.0	2.0	NaN	1.0	2.0	1.0	8
36	NaN	NaN	NaN	1.0	NaN	NaN	1
37	1.0	1.0	NaN	NaN	NaN	NaN	2
38	1.0	1.0	NaN	3.0	1.0	1.0	7
39	NaN	NaN	NaN	1.0	NaN	NaN	1
40	NaN	2.0	NaN	1.0	1.0	1.0	5
41	NaN	NaN	NaN	1.0	NaN	NaN	1
42	NaN	NaN	NaN	NaN	NaN	1.0	1
43	NaN	NaN	NaN	1.0	NaN	NaN	1
44	1.0	NaN	NaN	NaN	NaN	NaN	1
45	NaN	NaN	NaN	NaN	1.0	1.0	2
46	1.0	NaN	NaN	NaN	NaN	NaN	1
47	NaN	NaN	NaN	1.0	NaN	1.0	2
48	NaN	NaN	NaN	NaN	1.0	1.0	2
50	1.0	NaN	NaN	NaN	NaN	NaN	1
All	40.0	29.0	7.0	40.0	31.0	33.0	180

```
From above contingency table we can observe following things:
```

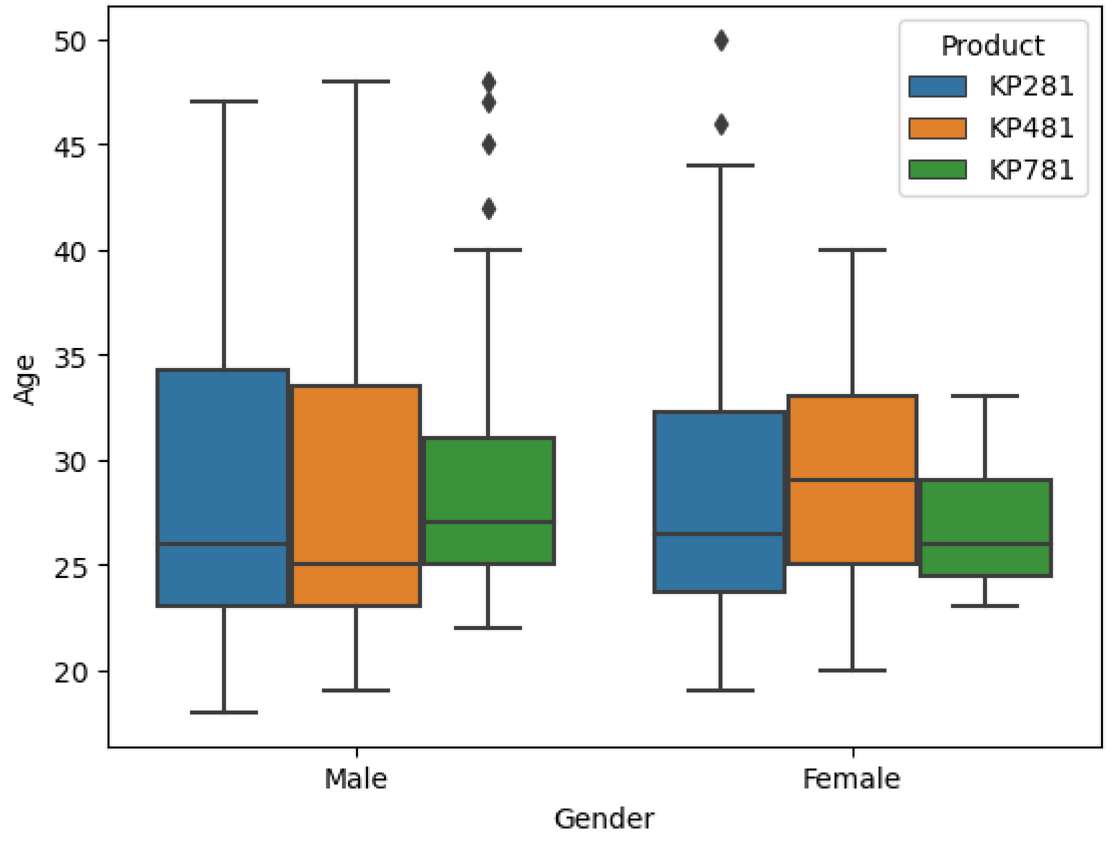
```
1) In females and males most of them have bought KP281 (40 units across all age
group).
2) This might be due to price affordability.
3) In females, having age between 22 and 28 have bought more KP281.
4) In males, having age between 23 and 26 have bought more KP281.
5) The age group of people who are more fitness freak is observed between 23 and 26.

So, we can state that youngsters between 23 and 26 who are fitness addicted are most
likely to buy KP281.
```

In [66]:

```python
sns.boxplot(data=df, x="Gender", y="Age", hue="Product")
```

Out[66]:

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



In [68]:

```python
df.groupby(["Gender","Product"])["Age"].median()
```

Out[68]:

```
Gender  Product
Female  KP281      26.5
        KP481      29.0
        KP781      26.0
Male    KP281      26.0
        KP481      25.0
        KP781      27.0
Name: Age, dtype: float64
```

In [71]:

```
df.loc[(df["Gender"]=="Male")]["Age"].median()
```

Out[71]:

```
26.0
```

In [72]:

```
df.loc[(df["Gender"]=="Female")]["Age"].median()
```

Out[72]:

```
26.5
```

In [73]:

```
min_age
```

Out[73]:

```
18
```

In [75]:

```
perc_25 = np.percentile(df["Age"],25)
perc_25
```

Out[75]:

```
24.0
```

In [76]:

```
perc_50 = np.percentile(df["Age"],50)
perc_50
```

Out[76]:

```
26.0
```

In [77]:

```
perc_75 = np.percentile(df["Age"],75)
perc_75
```

Out[77]:

```
33.0
```

In [78]:

```
max_age
```

Out[78]:

```
50
```

In [79]:

```
IQR = perc_75 - perc_25
IQR
```

Out[79]:

```
9.0
```

In [80]:

```
lower_whisker= max(perc_25-(1.5*IQR),df["Age"].min())
lower_whisker
```

Out[80]:

```
18
```

In [82]:

```
upper_whisker= min(perc_75+(1.5*IQR),df["Age"].max())
upper_whisker
```

Out[82]:

```
46.5
```

In [97]:

```
outliers = df.loc[(df["Age"]>upper_whisker) | (df["Age"]<lower_whisker)]
outliers
```

Out[97]:

	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

In [98]:

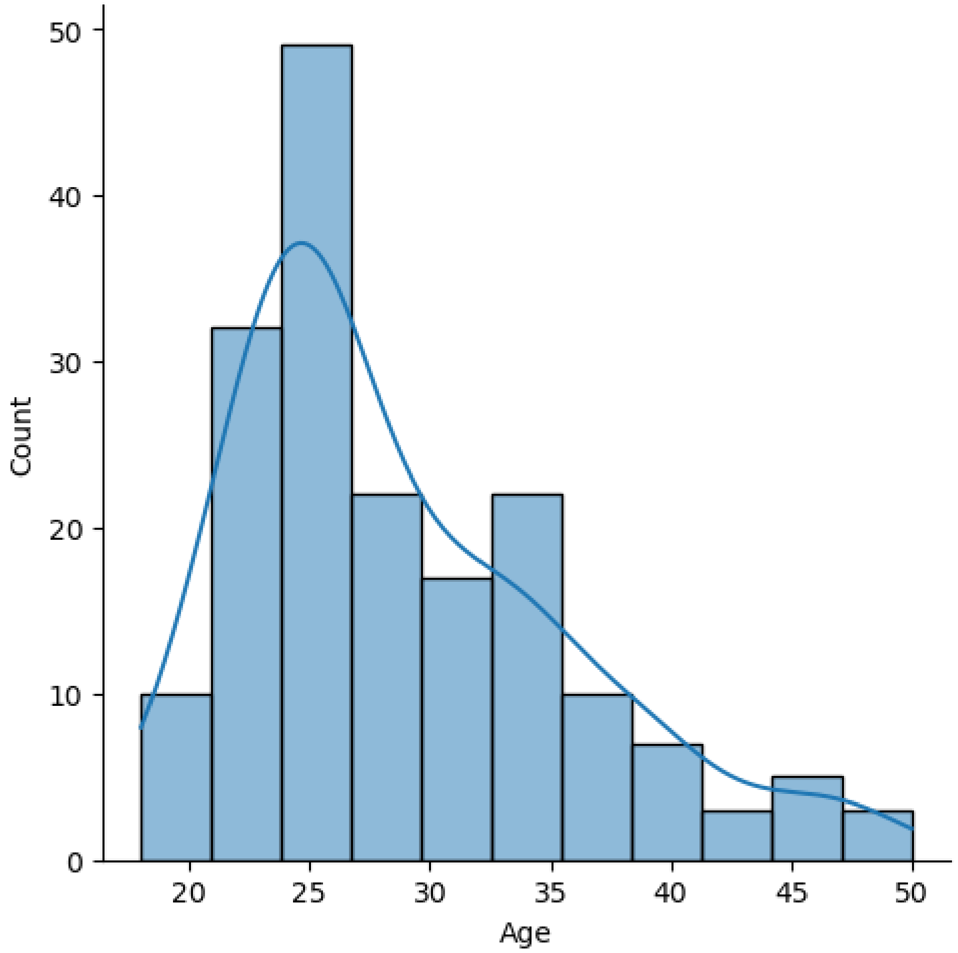
```
# Avereage no. of outliers in percentage is

(outliers.shape[0]/df.shape[0])*100
```

Out[98]:

```
2.7777777777777777
```

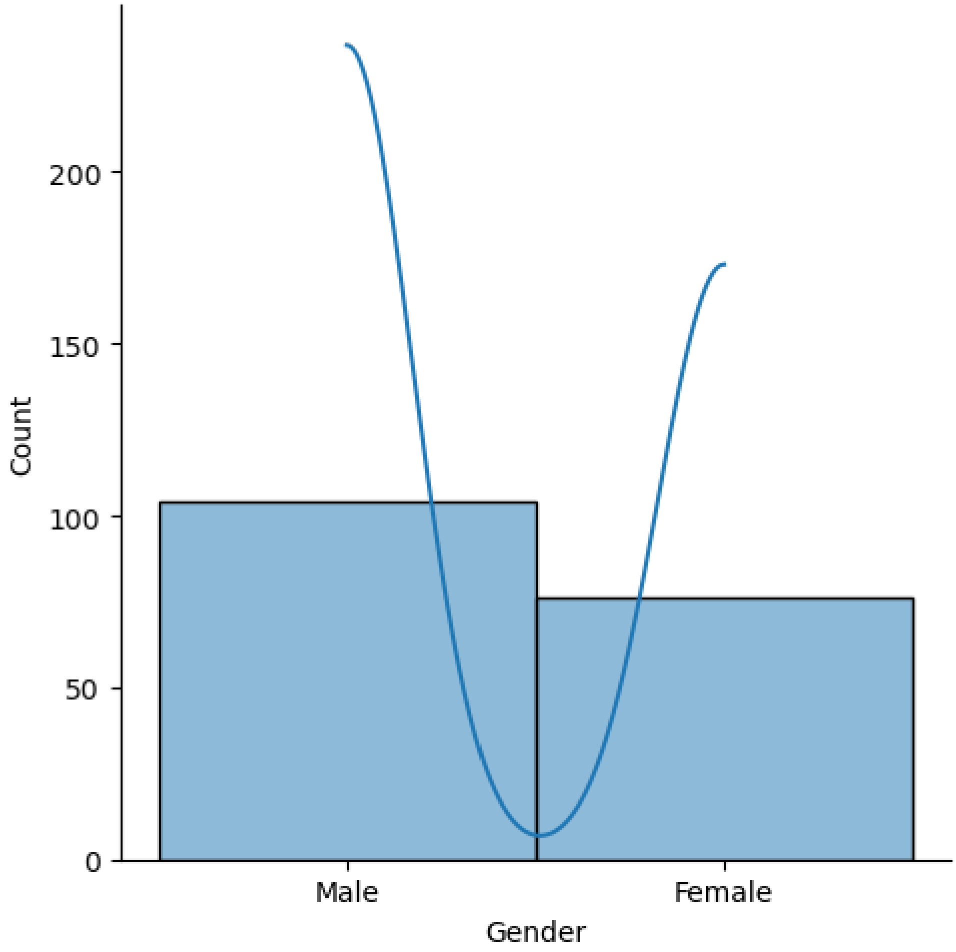
In [109]:

```
sns.displot(df["Age"], kde=True)
plt.show()
```



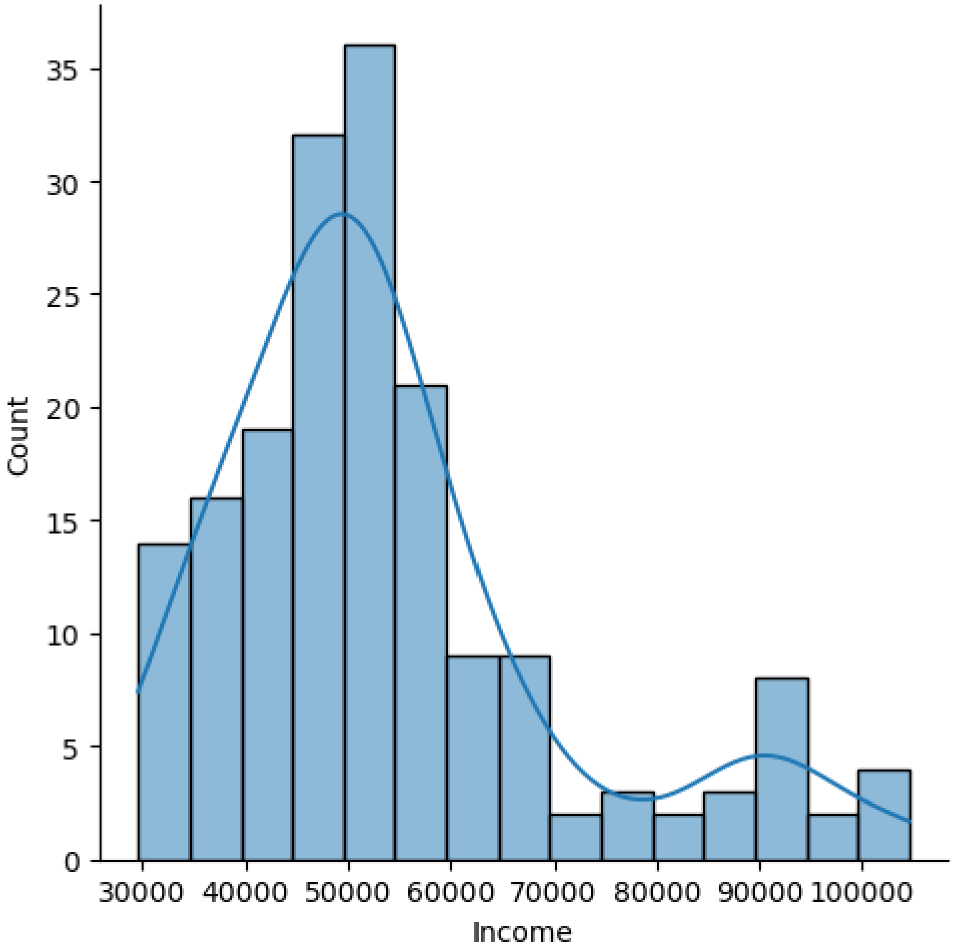
In [110]:

```
sns.displot(df["Gender"], kde=True)
plt.show()
```



In [113]:

```
sns.displot(df["Income"], kde=True)
plt.show()
```



In [114]:

```
df["Income"].min()
```

Out[114]:

```
29562
```

In [115]:

```
df["Income"].max()
```

Out[115]:

```
104581
```

In [116]:

```
df["Income"].median()
```

Out[116]:

```
50596.5
```

In [127]:

```
norm.interval(0.90, loc=165, scale=(8/np.sqrt(100)))
```

Out[127]:

```
(163.68411709843883, 166.31588290156117)
```

In [6]:

```
pd.crosstab(index=df["Gender"], columns=df["Product"], values=df["Product"], aggfunc="co
```

Out[6]:

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

In [ ]:

```
# In above table we can observe that, KP281 is bought the most among the 3 types of tread
# If we compare male and female, the males are the most buyers who have the bought the m
```

In [19]:

```
buyer_perc = pd.crosstab(index=df["Gender"], columns=df["Product"], normalize="all", mar
buyer_perc
```

Out[19]:

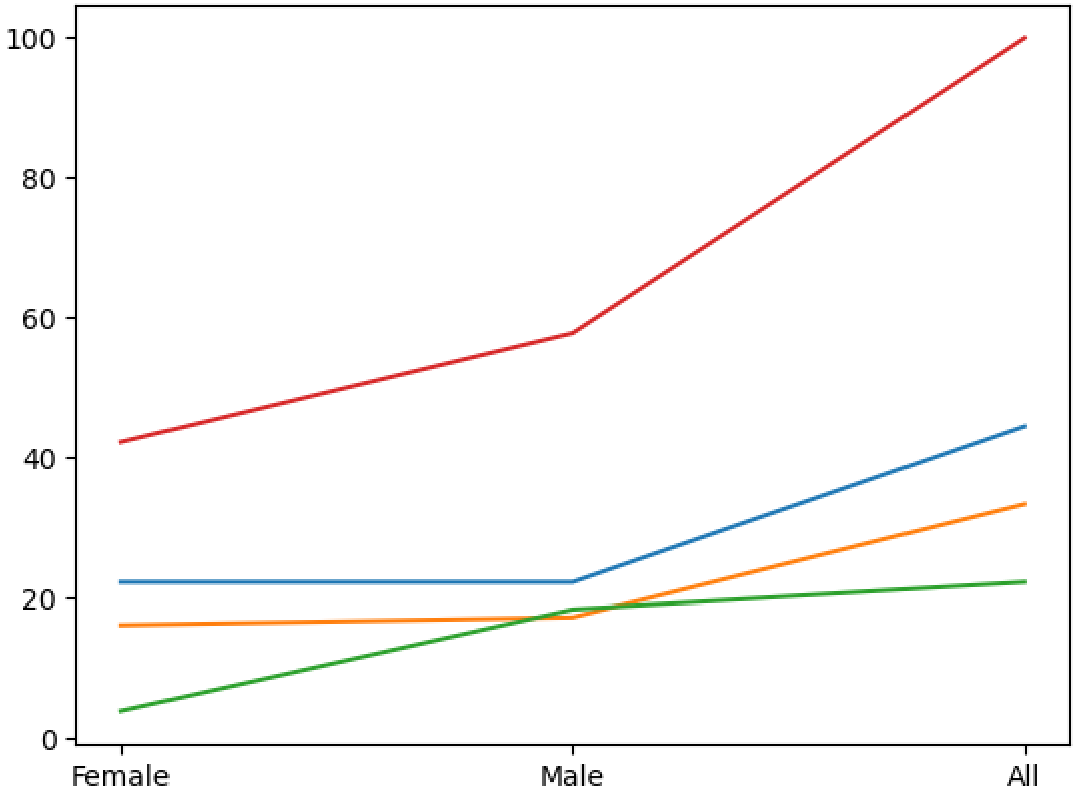
Product	KP281	KP481	KP781	All
Gender				
Female	22.222222	16.111111	3.888889	42.222222
Male	22.222222	17.222222	18.333333	57.777778
All	44.444444	33.333333	22.222222	100.000000

In [30]:

```
plt.plot(buyer_perc)
```

Out[30]:

```
[<matplotlib.lines.Line2D at 0x1d81cc1db10>,
 <matplotlib.lines.Line2D at 0x1d81cc1dab0>,
 <matplotlib.lines.Line2D at 0x1d81cc1dae0>,
 <matplotlib.lines.Line2D at 0x1d81cc1dc60>]
```



In [15]:

```
# Here we can see that 22% of females and males have bought KP281
```

In [16]:

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

Out[16]:

Product	KP281	KP481	KP781
Gender			
Female	52.631579	38.157895	9.210526
Male	38.461538	29.807692	31.730769
All	44.444444	33.333333	22.222222

In [17]:

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

Out[17]:

Product	KP281	KP481	KP781	All
Gender				
Female	50.0	48.333333	17.5	42.222222
Male	50.0	51.666667	82.5	57.777778

In [ ]:

```

```