

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

```
df=pd.read_csv('/content/Churn_Modelling.csv')
```

```
df.head()
```

	RowNumber	CustomerId	Surname	CreditScore	Geography	Gender	Age	Tenure	Balance
0	1	15634602	Hargrave	619	France	Female	42	2	0.0
1	2	15647311	Hill	608	Spain	Female	41	1	83807.1
2	3	15619304	Onio	502	France	Female	42	8	159660.1
3	4	15701354	Boni	699	France	Female	39	1	0.0
4	5	15737888	Mitchell	850	Spain	Female	43	2	125510.1

```
df.describe()
```

	RowNumber	CustomerId	CreditScore	Age	Tenure	Ba
count	10000.00000	1.000000e+04	10000.000000	10000.000000	10000.000000	10000.0
mean	5000.50000	1.569094e+07	650.528800	38.921800	5.012800	76485.8
std	2886.89568	7.193619e+04	96.653299	10.487806	2.892174	62397.4
min	1.00000	1.556570e+07	350.000000	18.000000	0.000000	0.0
25%	2500.75000	1.562853e+07	584.000000	32.000000	3.000000	0.0
50%	5000.50000	1.569074e+07	652.000000	37.000000	5.000000	97198.5
75%	7500.25000	1.575323e+07	718.000000	44.000000	7.000000	127644.2
max	10000.00000	1.581569e+07	850.000000	92.000000	10.000000	250898.0

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10000 entries, 0 to 9999
Data columns (total 14 columns):
```

```

#   Column      Non-Null Count  Dtype
---  -
0   RowNumber    10000 non-null    int64
1   CustomerId    10000 non-null    int64
2   Surname       10000 non-null    object
3   CreditScore    10000 non-null    int64
4   Geography     10000 non-null    object
5   Gender        10000 non-null    object
6   Age           10000 non-null    int64
7   Tenure        10000 non-null    int64
8   Balance       10000 non-null    float64
9   NumOfProducts 10000 non-null    int64
10  HasCrCard     10000 non-null    int64
11  IsActiveMember 10000 non-null    int64
12  EstimatedSalary 10000 non-null    float64
13  Exited        10000 non-null    int64
dtypes: float64(2), int64(9), object(3)
memory usage: 1.1+ MB

```

```
df.head(2)
```

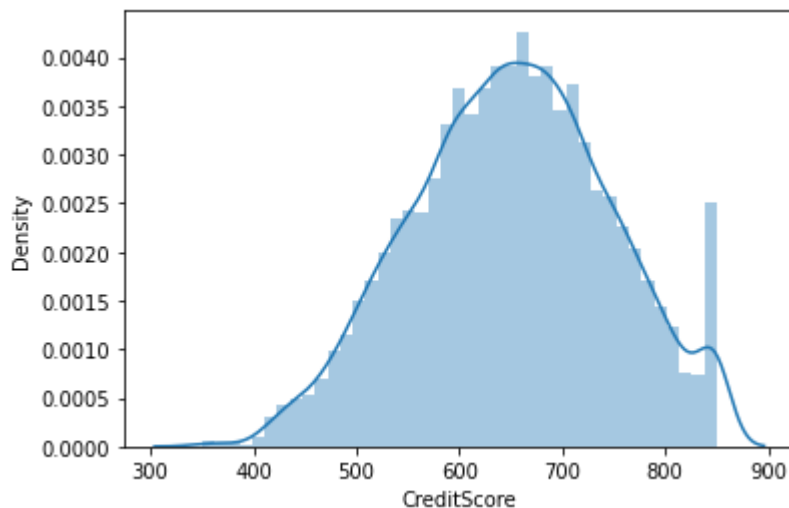
	RowNumber	CustomerId	Surname	CreditScore	Geography	Gender	Age	Tenure	Balance
0	1	15634602	Hargrave	619	France	Female	42	2	0.00
1	2	15647311	Hill	608	Spain	Female	41	1	835.00

```
sns.distplot(df.CreditScore)
```

```

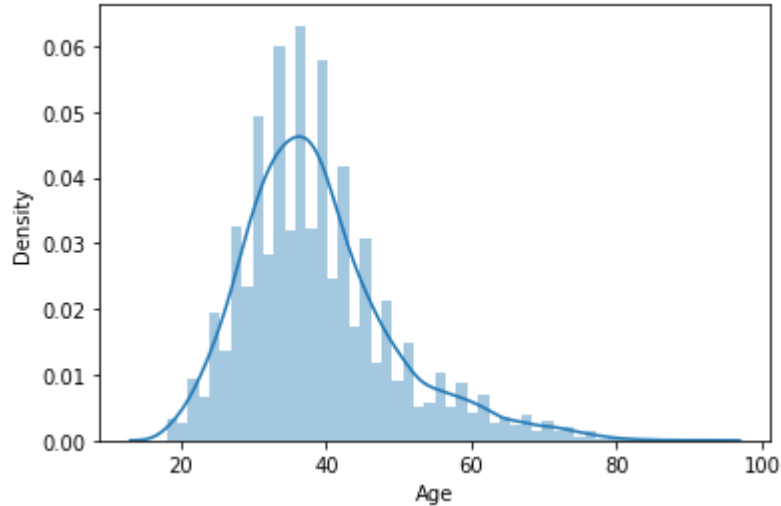
/usr/local/lib/python3.7/dist-packages/seaborn/distributions.py:2619: FutureWarning: `distplot` is a deprecated alias for `displot`. Please adjust your code to use `displot` instead.
warnings.warn(msg, FutureWarning)
<matplotlib.axes._subplots.AxesSubplot at 0x7f4bb4b6b1d0>

```



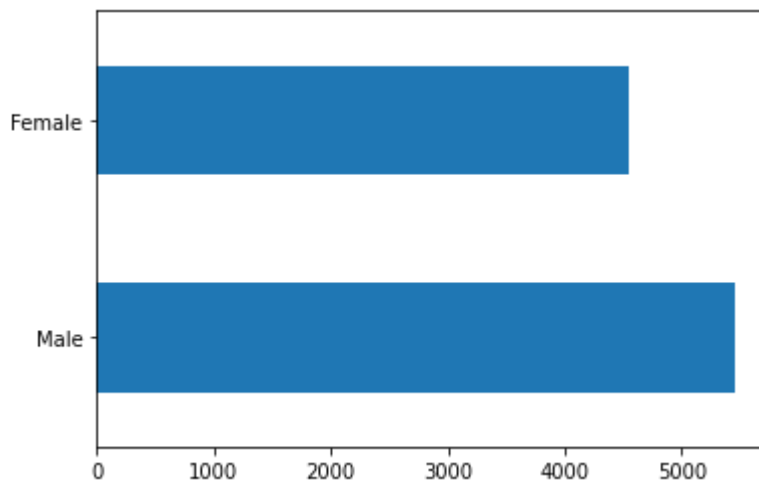
```
sns.distplot(df.Age)
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/distributions.py:2619: FutureWarning: `distplot` is deprecated and will be removed in a future version. Use `displot` instead.  
warnings.warn(msg, FutureWarning)  
<matplotlib.axes._subplots.AxesSubplot at 0x7f4bb49df710>
```



```
df.Gender.value_counts().plot(kind='barh')
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f4bb4461490>
```



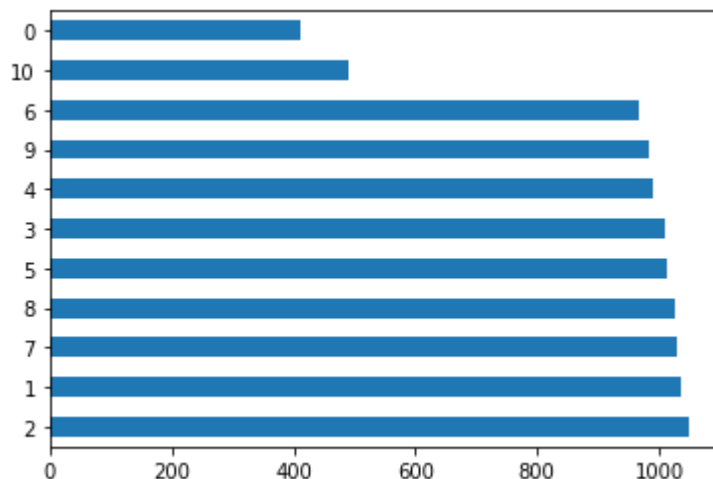
```
df.Geography.value_counts().plot(kind='barh')
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f4bb44d2e90>
```



```
df.Tenure.value_counts().plot(kind='barh')
```

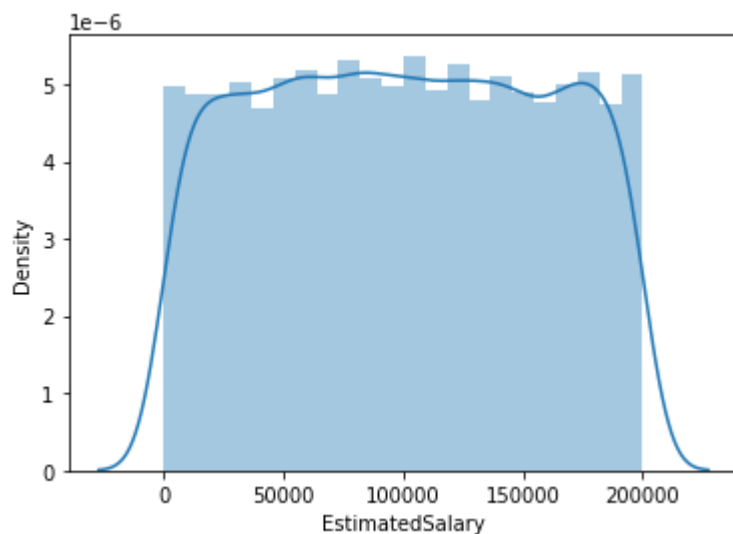
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f4bb4332050>
```



```
sns.distplot(df.EstimatedSalary)
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/distributions.py:2619: FutureWarning: `di
warnings.warn(msg, FutureWarning)
```

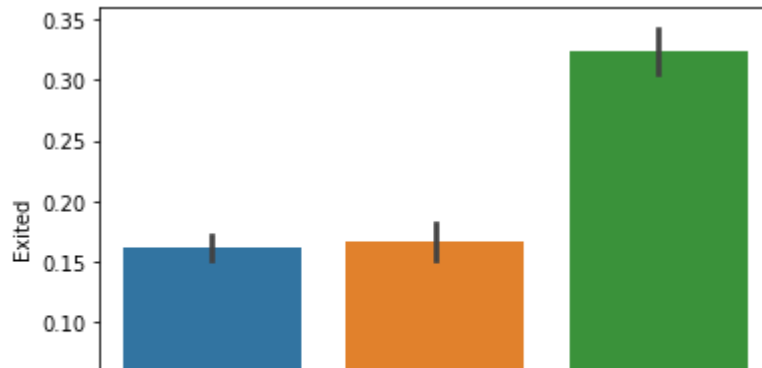
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f4bb42dd050>
```



```
sns.barplot(df.Geography, df.Exited)
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/_decorators.py:43: FutureWarning: Pass the following variables as keyword arguments: 'x', 'y' and 'color'.
FutureWarning
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f4bb41faf50>
```

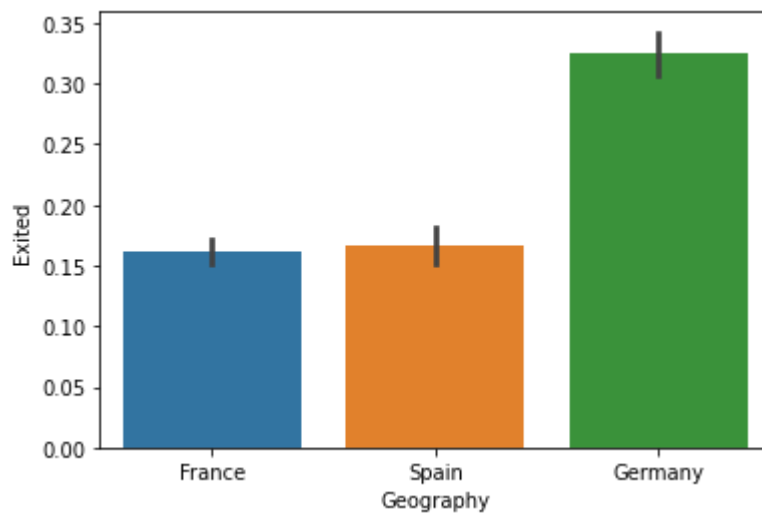


```
df.head(2)
```

	RowNumber	CustomerId	Surname	CreditScore	Geography	Gender	Age	Tenure	Bal
0	1	15634602	Hargrave	619	France	Female	42	2	83803.61
1	2	15647311	Hill	608	Spain	Female	41	1	84396.12

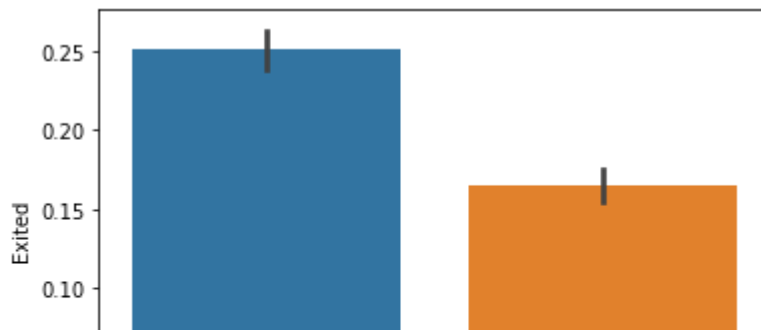
```
sns.barplot(x='Geography',y='Exited',data=df)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f4bb41709d0>
```



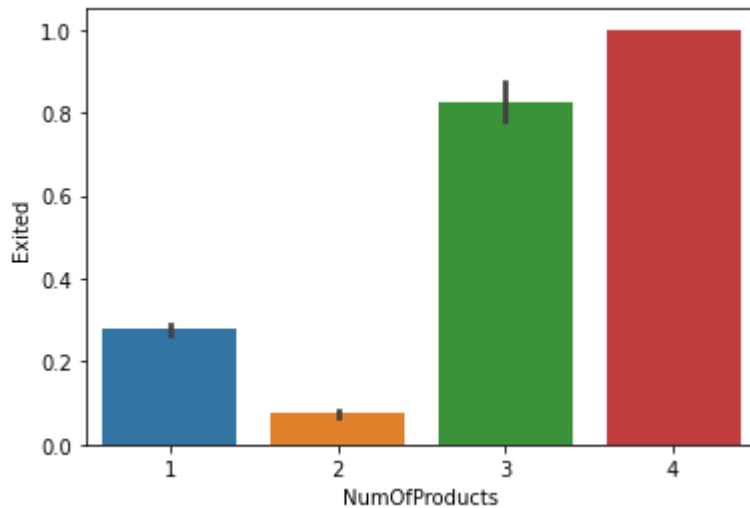
```
sns.barplot(x='Gender',y='Exited',data=df)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f4bb40f1c90>
```



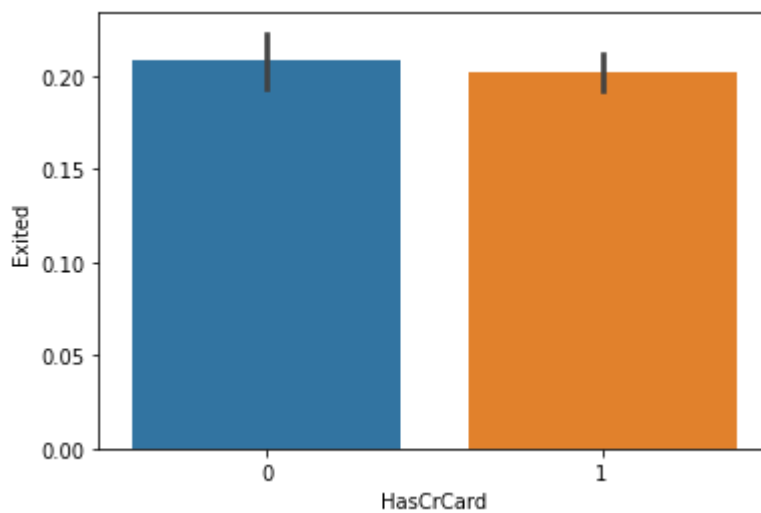
```
sns.barplot(x='NumOfProducts',y='Exited',data=df)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f4bb40d60d0>
```



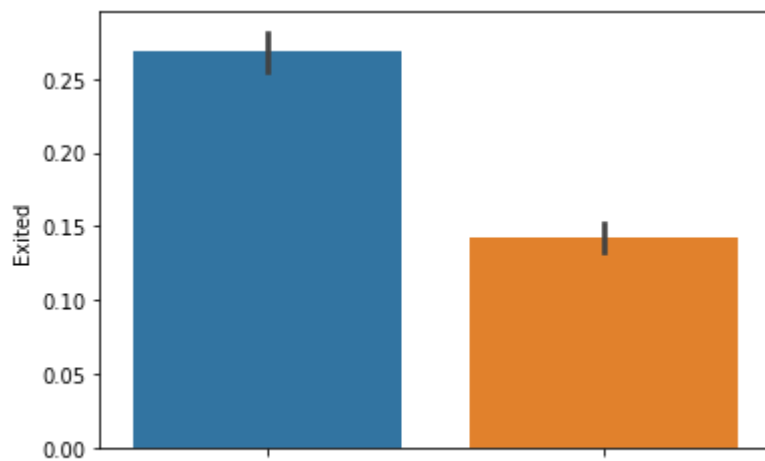
```
sns.barplot(x='HasCrCard',y='Exited',data=df)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f4bb403db10>
```



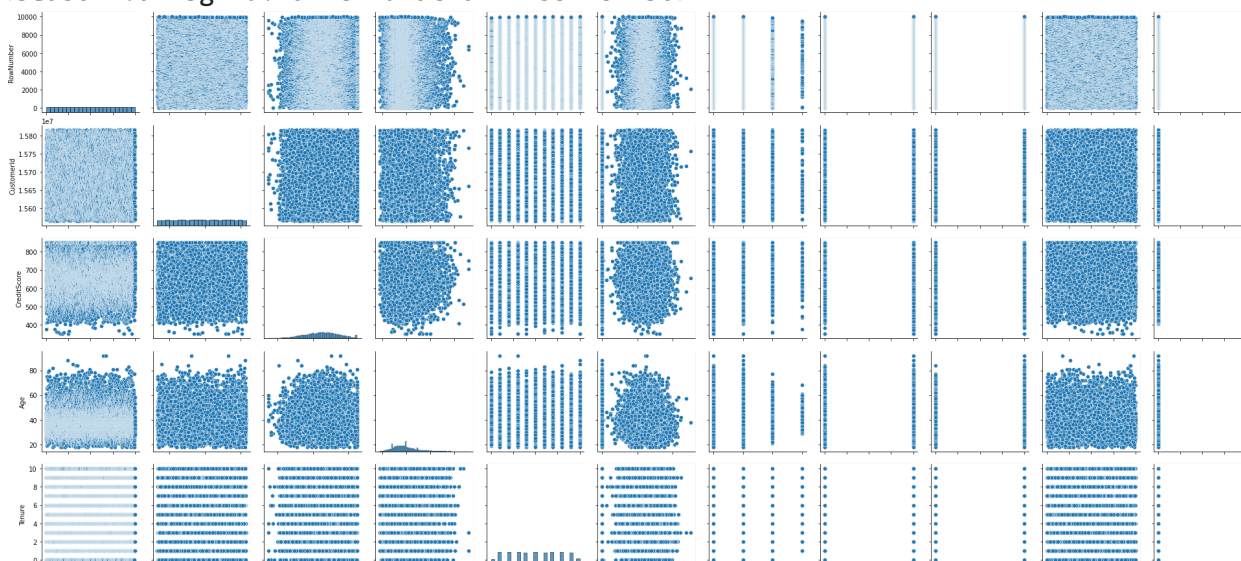
```
sns.barplot(x='IsActiveMember',y='Exited',data=df)
```

<matplotlib.axes._subplots.AxesSubplot at 0x7f4bb4011e10>



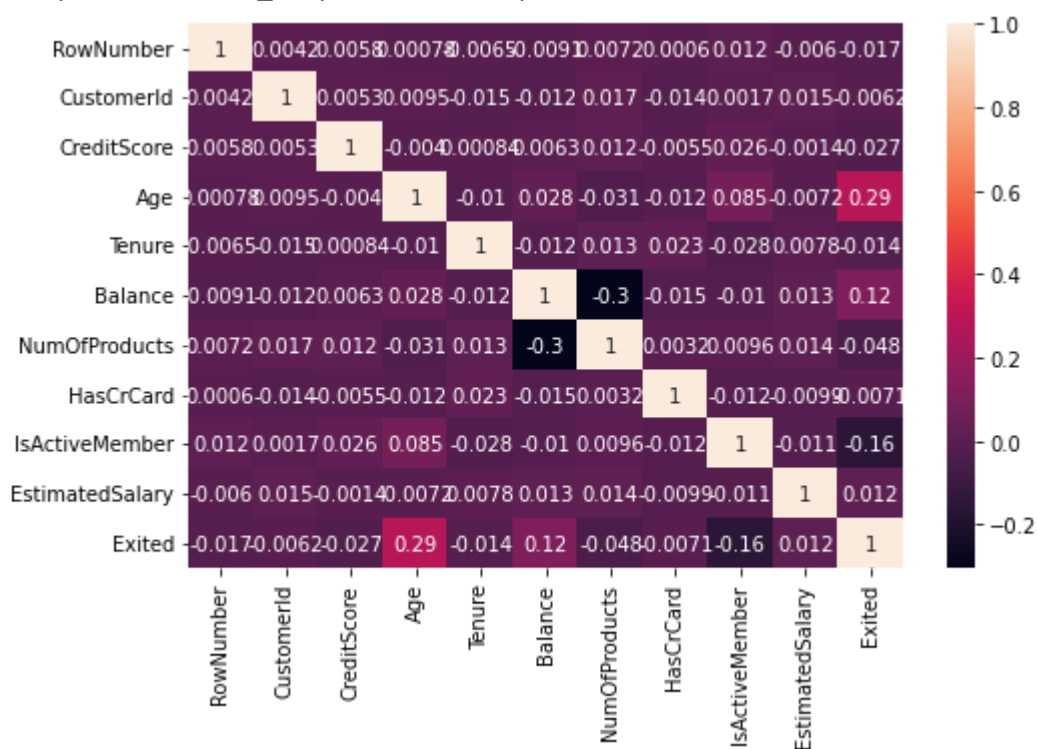
sns.pairplot(df)

<seaborn.axisgrid.PairGrid at 0x7f4bb428f750>



```
plt.figure(figsize=(8,5))
sns.heatmap(df.corr(),annot=True)
```

<matplotlib.axes._subplots.AxesSubplot at 0x7f4bc6252a50>



```
df.Exited.value_counts()
```

```
0    7963
1     2037
Name: Exited, dtype: int64
```

```
df.isnull().sum()
```

```
RowNumber    0
```



```

CustomerId      0
Surname         0
CreditScore     0
Geography       0
Gender          0
Age            0
Tenure         0
Balance        0
NumOfProducts  0
HasCrCard       0
IsActiveMember  0
EstimatedSalary 0
Exited         0
dtype: int64

```

```
#No missing values
```

```
df.head(2)
```

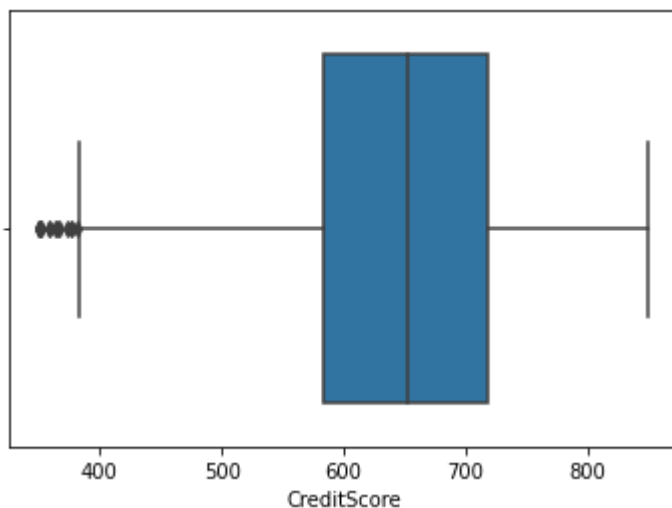
	RowNumber	CustomerId	Surname	CreditScore	Geography	Gender	Age	Tenure	EstimatedSalary
0	1	15634602	Hargrave	619	France	Female	42	2	59054
1	2	15647311	Hill	608	Spain	Female	41	1	81591

```
sns.boxplot(df.CreditScore)
```

```

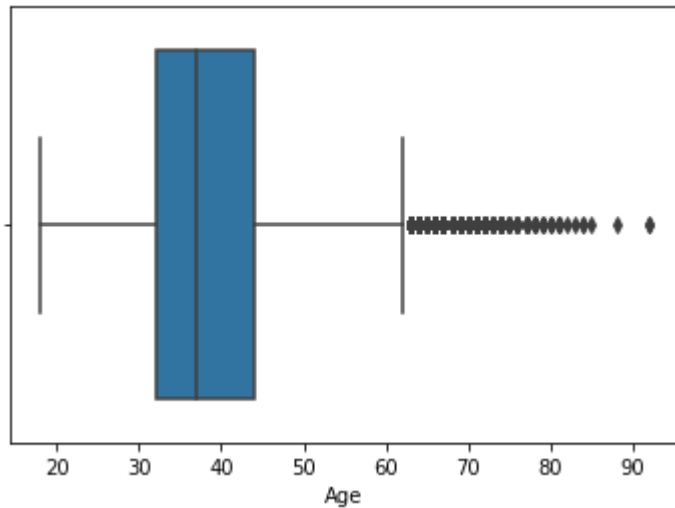
/usr/local/lib/python3.7/dist-packages/seaborn/_decorators.py:43: FutureWarning: P
FutureWarning
<matplotlib.axes._subplots.AxesSubplot at 0x7f4bacb745d0>

```



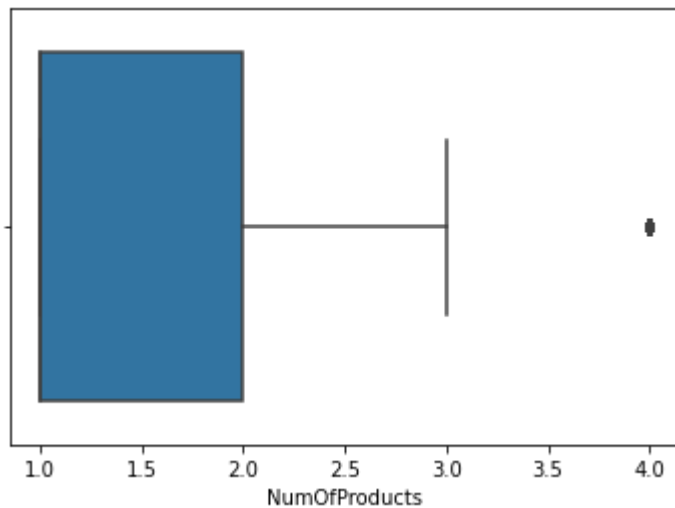
```
sns.boxplot(df.Age)
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/_decorators.py:43: FutureWarning: P
FutureWarning
<matplotlib.axes._subplots.AxesSubplot at 0x7f4bacb5a590>
```



```
sns.boxplot(df.NumOfProducts)
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/_decorators.py:43: FutureWarning: P
FutureWarning
<matplotlib.axes._subplots.AxesSubplot at 0x7f4bacac4750>
```



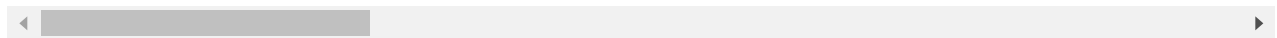
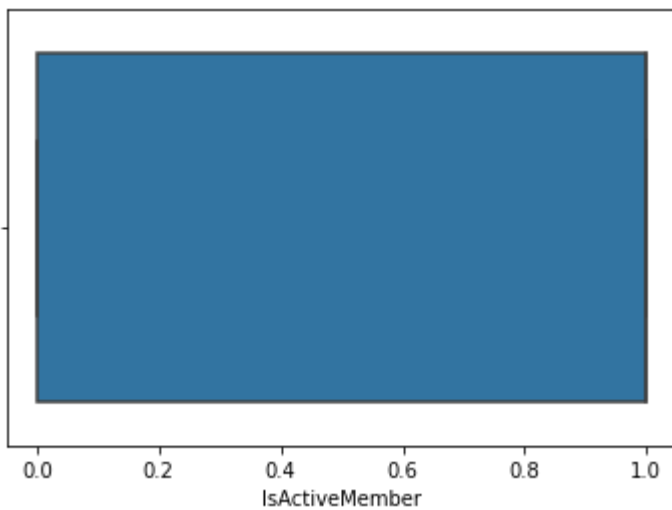
```
sns.boxplot(df.HasCrCard)
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/_decorators.py:43: FutureWarning: P
FutureWarning
<matplotlib.axes._subplots.AxesSubplot at 0x7f4baca430d0>
```



```
sns.boxplot(df.IsActiveMember)
```

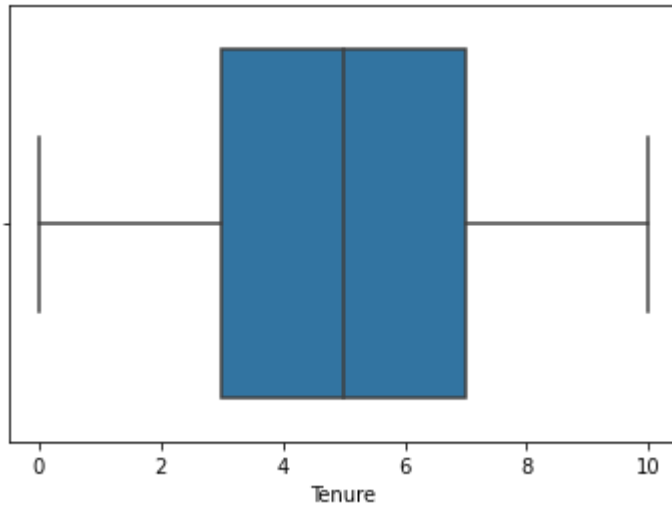
```
/usr/local/lib/python3.7/dist-packages/seaborn/_decorators.py:43: FutureWarning: P
FutureWarning
<matplotlib.axes._subplots.AxesSubplot at 0x7f4bae48e690>
```



```
sns.boxplot(df.EstimatedSalary)
```

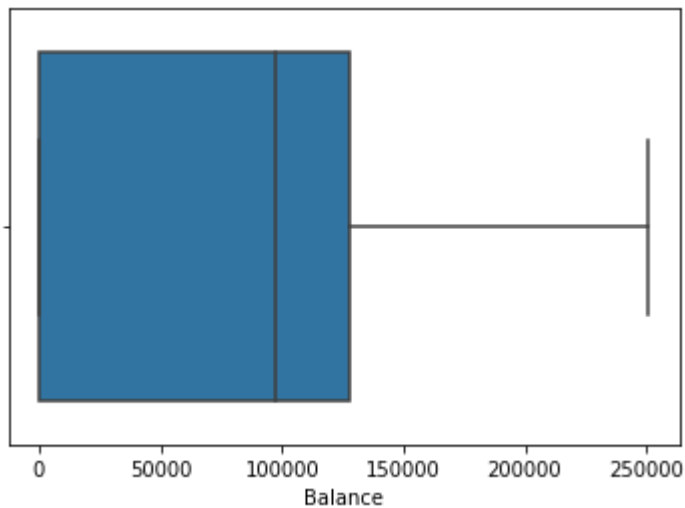
```
/usr/local/lib/python3.7/dist-packages/seaborn/_decorators.py:43: FutureWarning: P
sns.boxplot(df.Tenure)
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/_decorators.py:43: FutureWarning: P
FutureWarning
<matplotlib.axes._subplots.AxesSubplot at 0x7f4bac907c90>
```



```
sns.boxplot(df.Balance)
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/_decorators.py:43: FutureWarning: P
FutureWarning
<matplotlib.axes._subplots.AxesSubplot at 0x7f4bac868910>
```



```
#Outlier Removal
```

```
def outlier_credit_score(df):
    IQR = df['CreditScore'].quantile(0.75) - df['CreditScore'].quantile(0.25)

    lower_range = df['CreditScore'].quantile(0.25) - (1.5 * IQR)
```

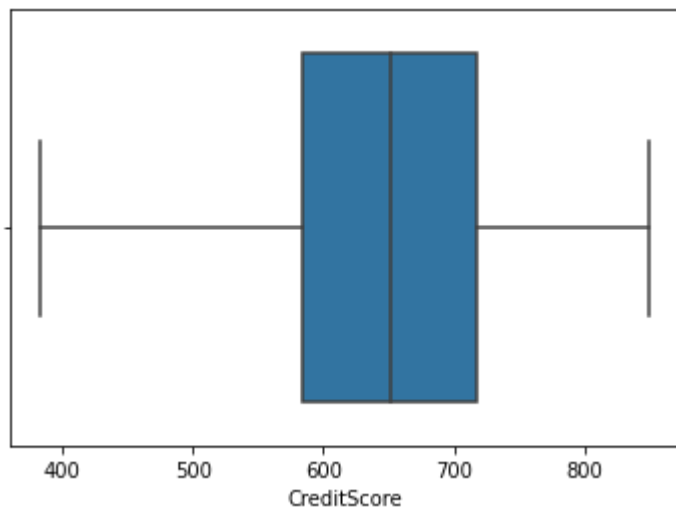
```
upper_range = df['CreditScore'].quantile(0.75) + (1.5 * IQR)
```

```
df.loc[df['CreditScore'] <= lower_range, 'CreditScore'] = lower_range
df.loc[df['CreditScore'] >= upper_range, 'CreditScore'] = upper_range
```

```
outlier_credit_score(df)
```

```
sns.boxplot(df.CreditScore)
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/_decorators.py:43: FutureWarning: P
FutureWarning
<matplotlib.axes._subplots.AxesSubplot at 0x7f4bac85f510>
```



```
def outlier_NOP(df):
```

```
    IQR = df['NumOfProducts'].quantile(0.75) - df['NumOfProducts'].quantile(0.25)
```

```
    lower_range = df['NumOfProducts'].quantile(0.25) - (1.5 * IQR)
```

```
    upper_range = df['NumOfProducts'].quantile(0.75) + (1.5 * IQR)
```

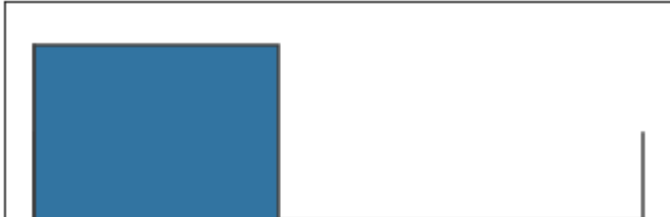
```
    df.loc[df['NumOfProducts'] <= lower_range, 'NumOfProducts'] = lower_range
```

```
    df.loc[df['NumOfProducts'] >= upper_range, 'NumOfProducts'] = upper_range
```

```
outlier_NOP(df)
```

```
sns.boxplot(df.NumOfProducts)
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/_decorators.py:43: FutureWarning: P
FutureWarning
<matplotlib.axes._subplots.AxesSubplot at 0x7f4bac7aa350>
```



```
def outlier_age(df):
    IQR = df['Age'].quantile(0.75) - df['Age'].quantile(0.25)

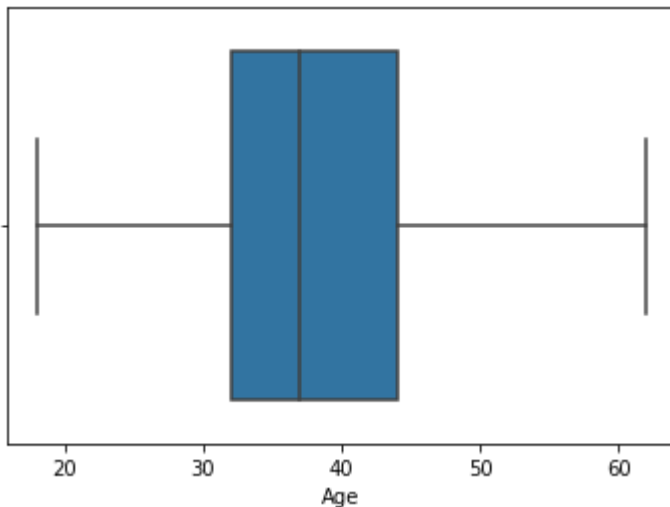
    lower_range = df['Age'].quantile(0.25) - (1.5 * IQR)
    upper_range = df['Age'].quantile(0.75) + (1.5 * IQR)

    df.loc[df['Age'] <= lower_range, 'Age'] = lower_range
    df.loc[df['Age'] >= upper_range, 'Age'] = upper_range
```

```
outlier_age(df)
```

```
sns.boxplot(df.Age)
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/_decorators.py:43: FutureWarning: P
FutureWarning
<matplotlib.axes._subplots.AxesSubplot at 0x7f4bac724410>
```



```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10000 entries, 0 to 9999
Data columns (total 14 columns):
#   Column          Non-Null Count  Dtype
---  -
0   RowNumber       10000 non-null  int64
1   CustomerId      10000 non-null  int64
```

```

2  Surname          10000 non-null object
3  CreditScore      10000 non-null int64
4  Geography        10000 non-null object
5  Gender           10000 non-null object
6  Age              10000 non-null int64
7  Tenure           10000 non-null int64
8  Balance          10000 non-null float64
9  NumOfProducts    10000 non-null float64
10 HasCrCard        10000 non-null int64
11 IsActiveMember   10000 non-null int64
12 EstimatedSalary  10000 non-null float64
13 Exited           10000 non-null int64
dtypes: float64(3), int64(8), object(3)
memory usage: 1.1+ MB

```

```
df.head(2)
```

	RowNumber	CustomerId	Surname	CreditScore	Geography	Gender	Age	Tenure	EstimatedSalary
0	1	15634602	Hargrave	619	France	Female	42	2	101334.36
1	2	15647311	Hill	608	Spain	Female	41	1	113564.92

```
df.drop(['CustomerId', 'RowNumber', 'Surname'],axis=1,inplace=True)
```

```
df.head(2)
```

	CreditScore	Geography	Gender	Age	Tenure	Balance	NumOfProducts	HasCrCard
0	619	France	Female	42	2	0.00	1.0	1
1	608	Spain	Female	41	1	83807.86	1.0	0

```

from sklearn.preprocessing import LabelEncoder
le_geo = LabelEncoder()
le_gen = LabelEncoder()
df['Sex']=le_gen.fit_transform(df.Gender)
df['Country']=le_geo.fit_transform(df.Geography)
df.drop(['Geography', 'Gender'],axis=1,inplace=True)

```

```
df.head(2)
```

	CreditScore	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	E
0	619	42	2	0.00	1.0	1		1
1	608	41	1	83807.86	1.0	0		1

```
X=df.drop('Exited',axis=1)
```

```
y=df.Exited
```

```
X
```

	CreditScore	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMembe
0	619	42	2	0.00	1.0	1	
1	608	41	1	83807.86	1.0	0	
2	502	42	8	159660.80	3.0	1	
3	699	39	1	0.00	2.0	0	
4	850	43	2	125510.82	1.0	1	
...	
9995	771	39	5	0.00	2.0	1	
9996	516	35	10	57369.61	1.0	1	
9997	709	36	7	0.00	1.0	0	
9998	772	42	3	75075.31	2.0	1	
9999	792	28	4	130142.79	1.0	1	

10000 rows × 10 columns

```
from sklearn.preprocessing import StandardScaler
```

```
sc=StandardScaler()
```

```
X = sc.fit_transform(X)
```


[Colab paid products](#) - [Cancel contracts here](#)

