TEAM ID: PNT2022TMID41512

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

In [35]:
```
df=pd.read_csv('Churn_Modelling.csv')
df.head()
```

Out[35]:

	RowNumber	CustomerId	Surname	CreditScore	Geography	Gender	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	EstimatedSalary	Exited
0	1	15634602	Hargrave	619	France	Female	42	2	0.00	1	1	1	101348.88	1
1	2	15647311	Hill	608	Spain	Female	41	1	83807.86	1	0	1	112542.58	0
2	3	15619304	Onio	502	France	Female	42	8	159660.80	3	1	0	113931.57	1
3	4	15701354	Boni	699	France	Female	39	1	0.00	2	0	0	93826.63	0
4	5	15737888	Mitchell	850	Spain	Female	43	2	125510.82	1	1	1	79084.10	0

UNIVARIATE SCATTER PLOT

In [3]:
```
df.shape
```

Out[3]:
```
(10000, 14)
```

In [4]:
```
df.dtypes
```

Out[4]:
```
RowNumber          int64
CustomerId         int64
Surname           object
```

```
      CreditScore          int64
      Geography           object
      Gender              object
Age                    int64
Tenure                 int64
Balance              float64
NumOfProducts          int64
HasCrCard              int64
IsActiveMember         int64
EstimatedSalary      float64
Exited                 int64
dtype: object
```

In [5]:
```
df.dtypes[df.dtypes == 'float64']
```

Out[5]:
```
Balance            float64
EstimatedSalary    float64
dtype: object
```

```
df.select_dtypes(include=['int64','float64','Int64']).dtypes
```

Out[6]:
```
RowNumber            int64
CustomerId           int64
CreditScore          int64
Age                  int64
Tenure               int64
Balance            float64
NumOfProducts        int64
HasCrCard            int64
IsActiveMember       int64
EstimatedSalary    float64
Exited               int64
dtype: object
```

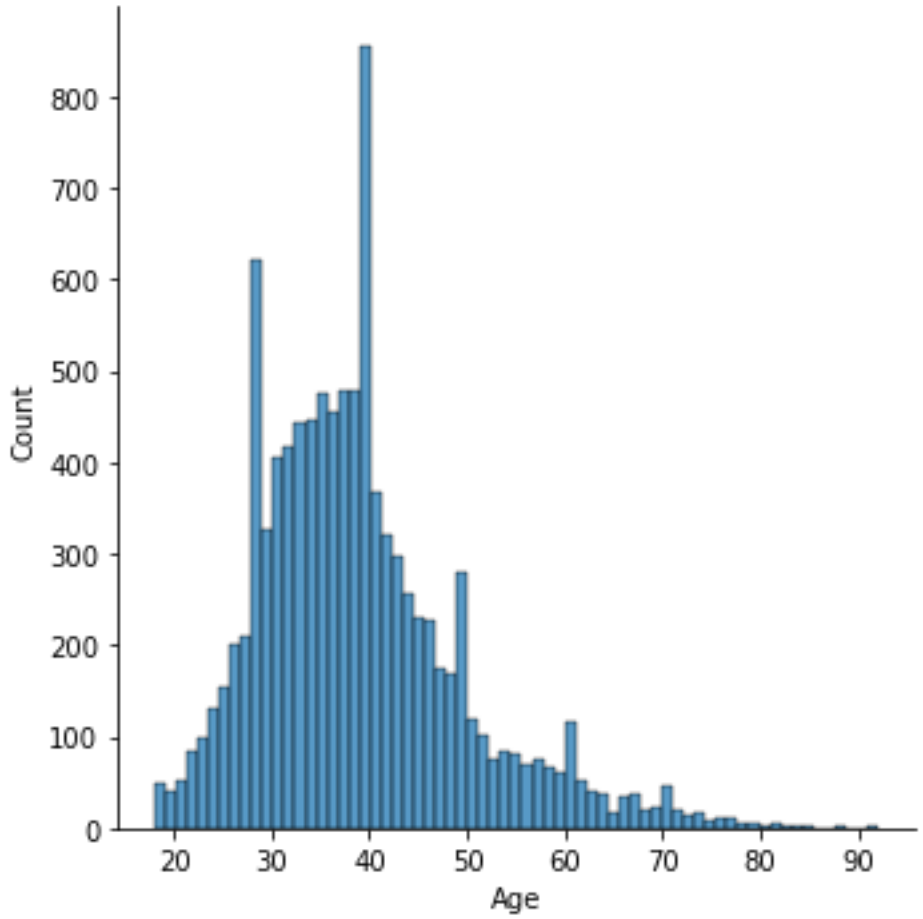
In [7]:
```
df.isnull().any()
```

Out[7]:
```
RowNumber          False
CustomerId         False
Surname            False
CreditScore        False
Geography          False
Gender             False
Age                False
Tenure             False
Balance            False
NumOfProducts      False
HasCrCard          False
IsActiveMember     False
EstimatedSalary    False
Exited             False
```

```
dtype: bool
```

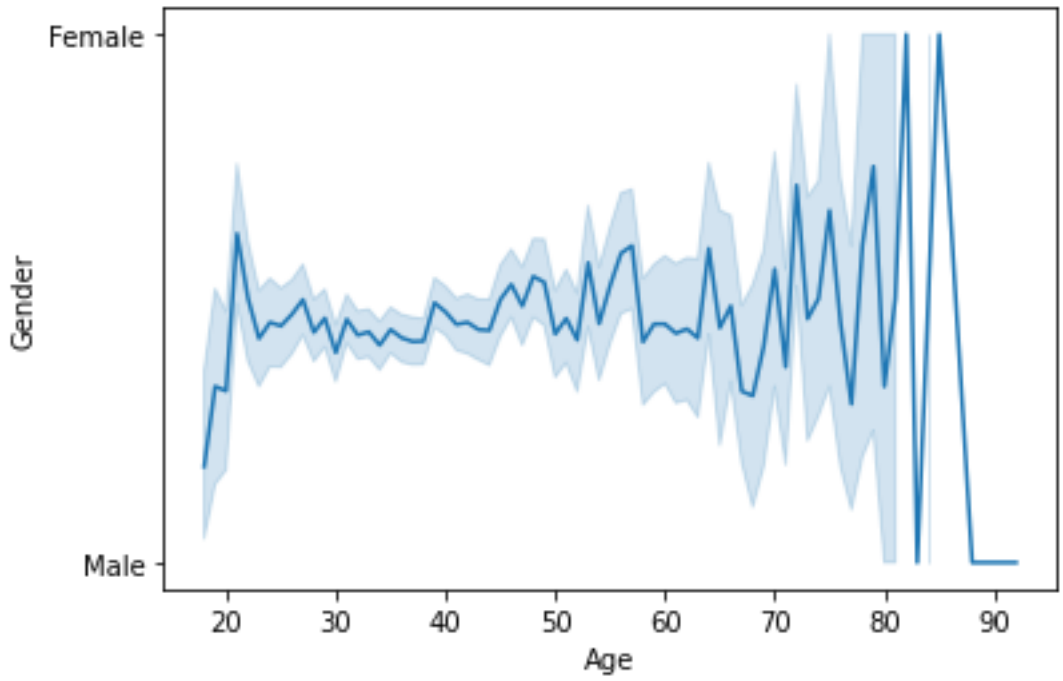
In [8]:
```
sns.displot(df.Age)
```

Out[8]:
```
<seaborn.axisgrid.FacetGrid at 0x1be6afec280>
```



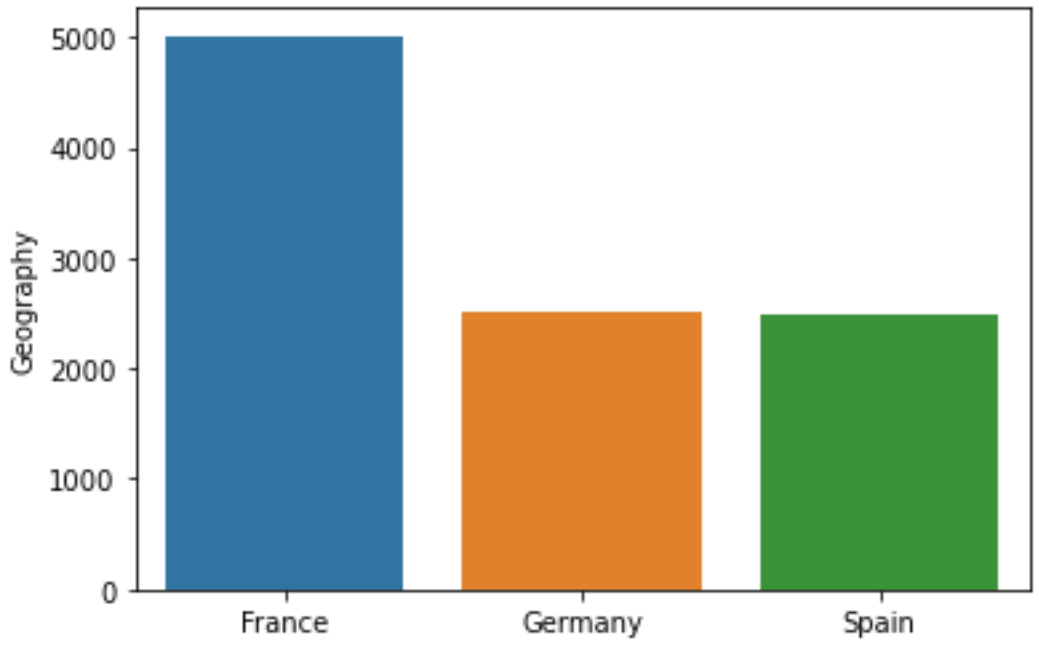

```
In[9]: sns.lineplot(df.Age,df.Gender)
```

Out[9]:
```
C:\Users\sunda\anaconda3\lib\site-
packages\seaborn\_decorators.py:36:             FutureWarning: Pass the
following variables as keyword args: x, y. From version 0.12, the only valid
positional argument will be `data`, and passing other arguments without an
explicit keyword will result in an error or misinterpretation.
  warnings.warn(
<AxesSubplot:xlabel='Age', ylabel='Gender'>
```




```
In[10]:
sns.barplot(df.Geography.value_counts().index,df.Geography.value_counts())
```

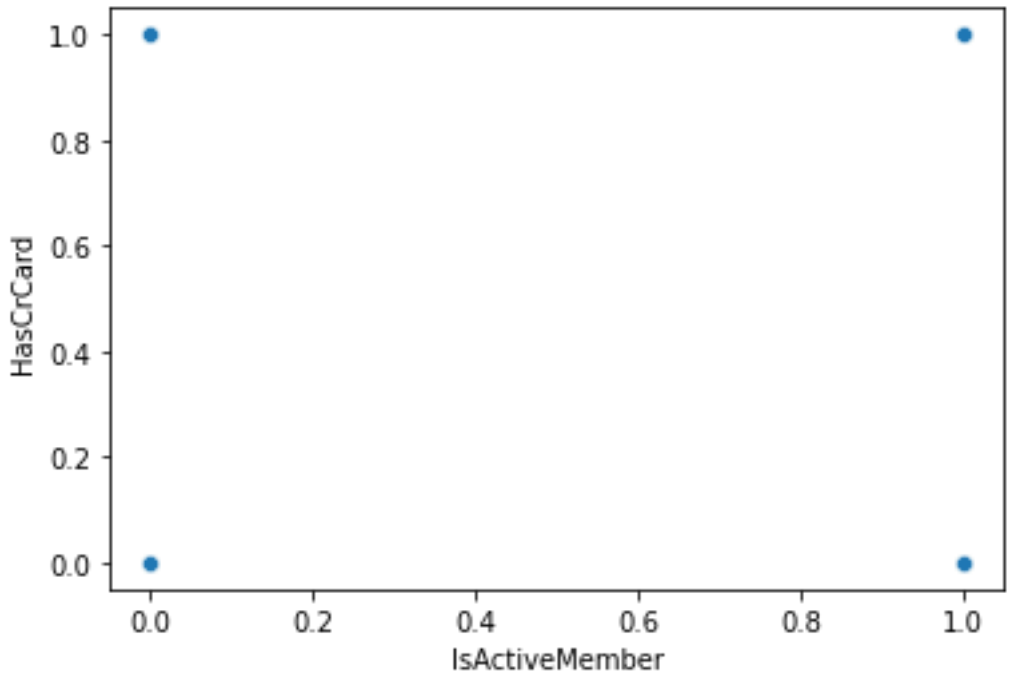
```
Out[10]:
C:\Users\sunda\anaconda3\lib\site-packages\seaborn\_decorators.py:36:
FutureWarning: Pass the following variables as keyword args: x, y. From
version 0.12, the only valid positional argument will be `data`, and passing
other arguments without an explicit keyword will result in an error or
misinterpretation.
  warnings.warn(
<AxesSubplot:ylabel='Geography'>
```




```
In[11]: sns.scatterplot(df.IsActiveMember,df.HasCrCard)
```

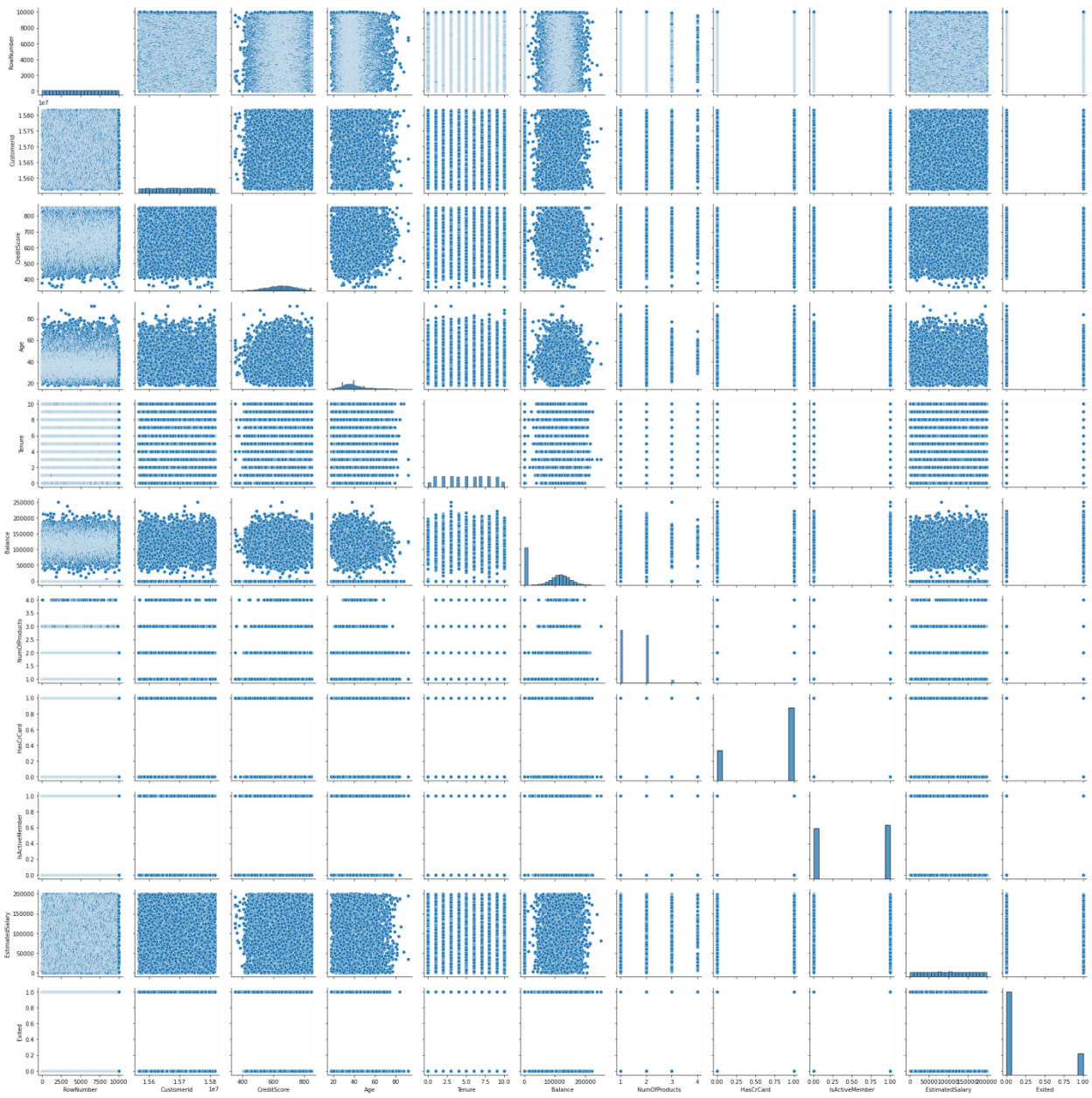
```
C:\Users\sunda\anaconda3\lib\site-packages\seaborn\_decorators.py:36:
FutureWarning: Pass the following variables as keyword args: x, y. From
version 0.12, the only valid positional argument will be `data`, and passing
other arguments without an explicit keyword will result in an error or
misinterpretation.
  warnings.warn(
```

Out[11]: `<AxesSubplot:xlabel='IsActiveMember', ylabel='HasCrCard'>`



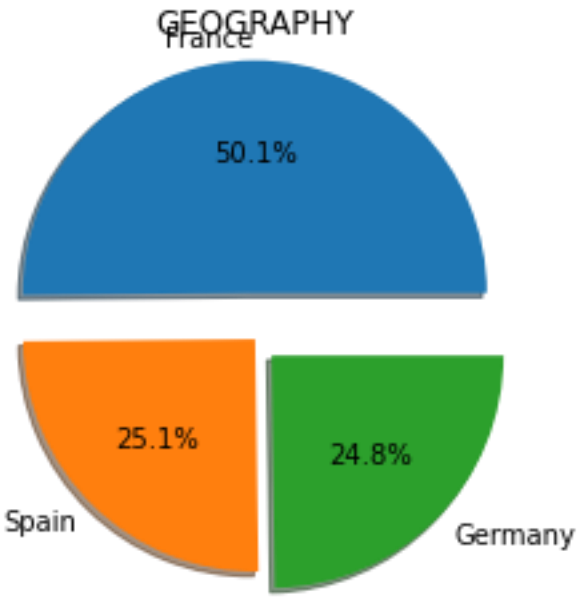

```
In[20]: sns.pairplot(df) #multivariate analysis
```

Out[20]:<seaborn.axisgrid.PairGrid at 0x1be70507670>



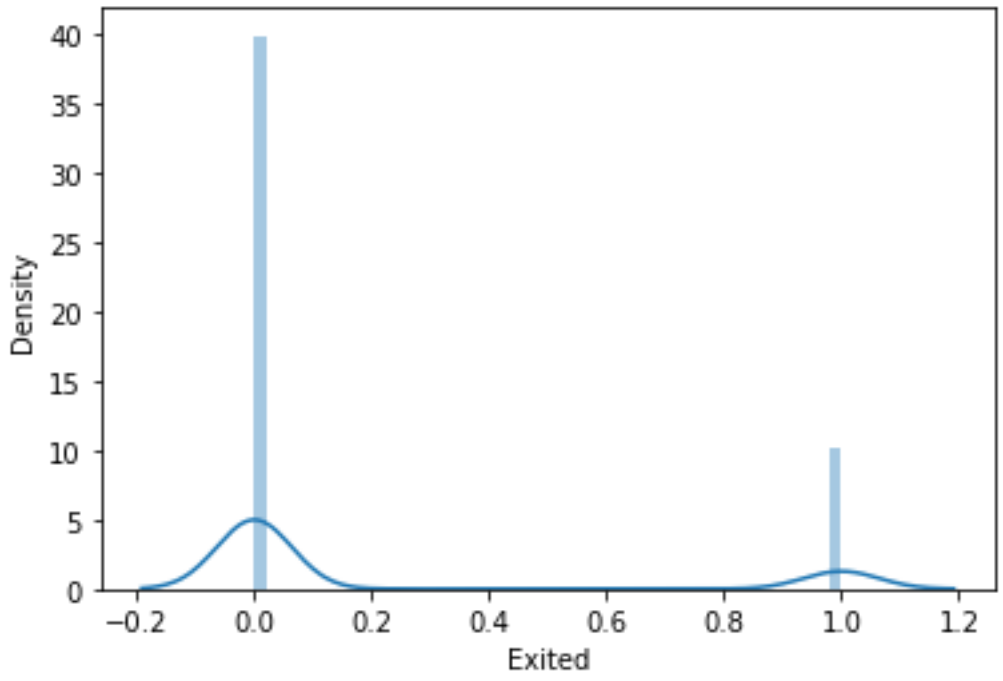
In[23]:

```
plt.pie(df.Geography.value_counts(),[0.2,0,0.1],shadow=True,labels=['France',
'Spain','Germany'],autopct='%1.1f%%')
plt.title('GEOGRAPHY')
plt.show()
```



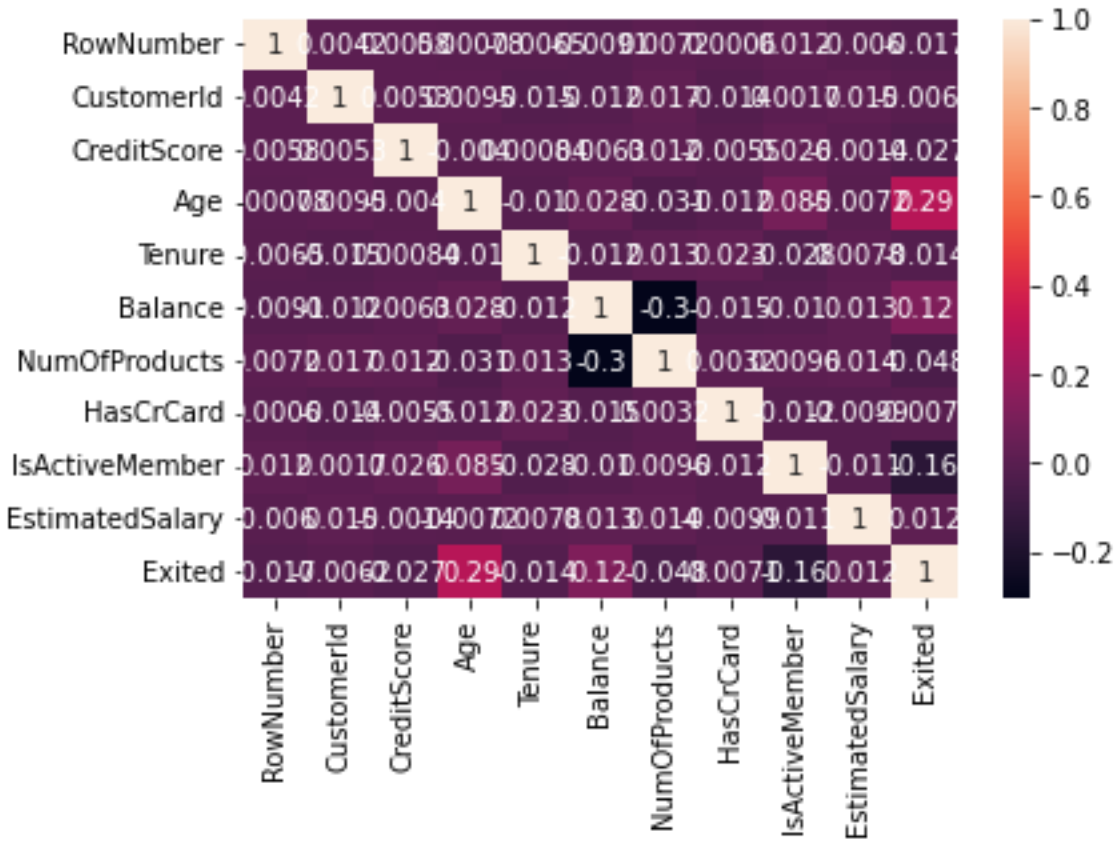

```
In[24]: sns.distplot(df.Exited)

        C:\Users\sunda\anaconda3\lib\site
packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a
deprecated function and will be removed in a future version. Please adapt
your code to use either `displot` (a figure-level function with similar
flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

Out[24]: `<AxesSubplot:xlabel='Exited', ylabel='Density'>`




```
In[25]: sns.heatmap(df.corr(),annot=True)
```

Out[25]: `<AxesSubplot:>`




```
In[26]: df.corr()
```

Out[26]:

	RowNumber	CustomerId	CreditScore	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	EstimatedSalary	Exited
RowNumber	1.000000	0.004202	0.005840	0.000783	-0.006495	-0.009067	0.007246	0.000599	0.012044	-0.005988	-0.016571
CustomerId	0.004202	1.000000	0.005308	0.009497	-0.014883	-0.012419	0.016972	-0.014025	0.001665	0.015271	-0.006248
CreditScore	0.005840	0.005308	1.000000	-0.003965	0.000842	0.006268	0.012238	-0.005458	0.025651	-0.001384	-0.027094

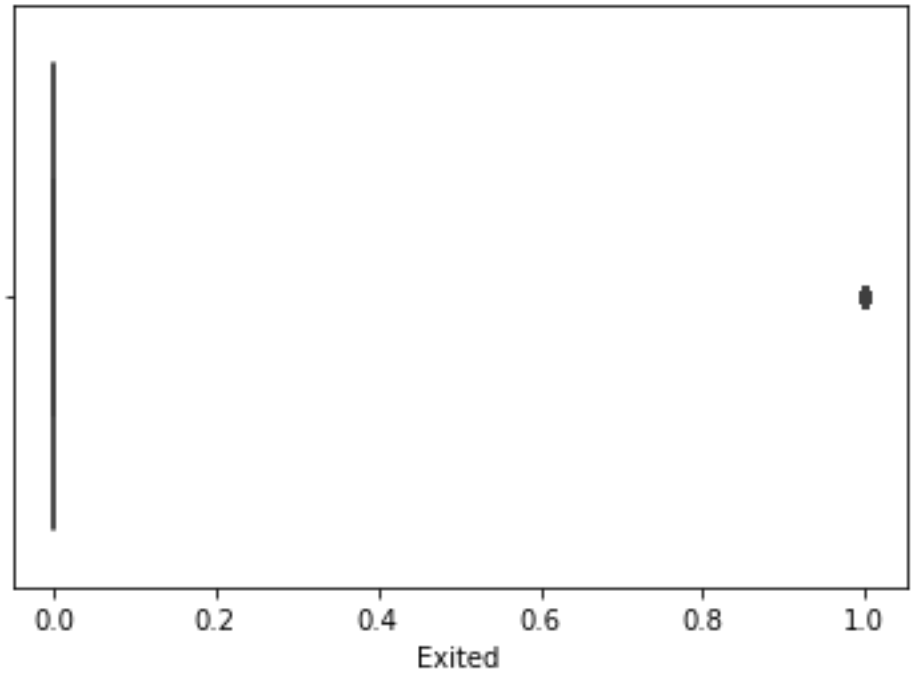
	RowNumber	CustomerId	CreditScore	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	EstimatedSalary	Exited
Age	0.000783	0.009497	-0.003965	1.000000	-0.009997	0.028308	-0.030680	-0.011721	0.085472	-0.007201	0.285323
Tenure	-0.006495	-0.014883	0.000842	-0.009997	1.000000	-0.012254	0.013444	0.022583	-0.028362	0.007784	-0.014001
Balance	-0.009067	-0.012419	0.006268	0.028308	-0.012254	1.000000	-0.304180	-0.014858	-0.010084	0.012797	0.118533
NumOfProducts	0.007246	0.016972	0.012238	-0.030680	0.013444	-0.304180	1.000000	0.003183	0.009612	0.014204	-0.047820
HasCrCard	0.000599	-0.014025	-0.005458	-0.011721	0.022583	-0.014858	0.003183	1.000000	-0.011866	-0.009933	-0.007138
IsActiveMember	0.012044	0.001665	0.025651	0.085472	-0.028362	-0.010084	0.009612	-0.011866	1.000000	-0.011421	-0.156128
EstimatedSalary	-0.005988	0.015271	-0.001384	-0.007201	0.007784	0.012797	0.014204	-0.009933	-0.011421	1.000000	0.012097
Exited	-0.016571	-0.006248	-0.027094	0.285323	-0.014001	0.118533	-0.047820	-0.007138	-0.156128	0.012097	1.000000

In [37]:

```
sns.boxplot(df.Exited)
```

```
C:\Users\sunda\anaconda3\lib\site-packages\seaborn\_decorators.py:36:
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 without an explicit keyword will result in an error or
misinterpretation.
  warnings.warn(
```

Out[37]: `<AxesSubplot:xlabel='Exited'>`




```
In[38]: df.describe()
```

Out[38]:

	RowNumber	CustomerId	Credit Score	Age	Tenure	Balance	NumOf Products	HasCrCard	IsActive Member	EstimatedSalary	Exited
count	10000.00000	1.000000e+04	10000.000000	10000.000000	10000.000000	10000.000000	10000.000000	10000.00000	10000.000000	10000.000000	10000.000000
mean	5000.50000	1.569094e+07	650.528800	38.921800	5.012800	76485.889288	1.530200	0.70550	0.515100	100090.239881	0.203700
std	2886.89568	7.193619e+04	96.653299	10.487806	2.892174	62397.405202	0.581654	0.45584	0.499797	57510.492818	0.402769
min	1.00000	1.556570e+07	350.000000	18.000000	0.000000	0.000000	1.000000	0.00000	0.000000	11.580000	0.000000
25%	2500.75000	1.562853e+07	584.000000	32.000000	3.000000	0.000000	1.000000	0.00000	0.000000	51002.110000	0.000000

	RowNumber	CustomerId	CreditScore	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	EstimatedSalary	Exited
50%	5000.50000	1.569074e+07	652.000000	37.000000	5.000000	97198.540000	1.000000	1.00000	1.000000	100193.915000	0.000000
75%	7500.25000	1.575323e+07	718.000000	44.000000	7.000000	127644.240000	2.000000	1.00000	1.000000	149388.247500	0.000000
max	10000.00000	1.581569e+07	850.000000	92.000000	10.000000	250898.090000	4.000000	1.00000	1.000000	199992.480000	1.000000

In [39]:
```
df.median()
C:\Users\sunda\AppData\Local\Temp\ipykernel_9816\530051474.py:1:
FutureWarning: Dropping of nuisance columns in DataFrame reductions (with
'numeric_only=None') is deprecated; in a future version this will raise
TypeError.  Select only valid columns before calling the reduction.
  df.median()

Out[39]: RowNumber          5.000500e+03
         CustomerId         1.569074e+07
         CreditScore        6.520000e+02
         Age                3.700000e+01
         Tenure             5.000000e+00
         Balance            9.719854e+04
         NumOfProducts      1.000000e+00
         HasCrCard          1.000000e+00
         IsActiveMember     1.000000e+00
         EstimatedSalary    1.001939e+05
         Exited             0.000000e+00
         dtype: float64
```

In [44]:
```
from sklearn.preprocessing import LabelEncoder
```

In [45]:
```
le=LabelEncoder()
```

In [46]:
```
df.Age=le.fit_transform(df.Age)df.HasCrCard=le.fit_transform(df.HasCrCard)
```

In [47]:
```
df.head()
Out[47]:
```

	RowNumber	CustomerId	Surname	CreditScore	Geography	Gender	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	EstimatedSalary	Exited	
0	1	15634602	Hargrave	619	France	Female	24	2	0.00	1	1	1	101348.88	1	
1	2	15647311	Hill	608	Spa	Fema	23	1	83807.8	1	0	1	112542.5	0	

RowNumber	CustomerId	Surname	CreditScore	Geography	Gender	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	EstimatedSalary	Exited	
		11			in	le			6				8	
2	3	15619304	Onio	502	France	Female	24	8	159660.80	3	1	0	113931.57	1
3	4	15701354	Boni	699	France	Female	21	1	0.00	2	0	0	93826.63	0
4	5	15737888	Mitchell	850	Spain	Female	25	2	125510.82	1	1	1	79084.10	0

In [48]:
```
df_main=pd.get_dummies(df,columns=['Gender'])
df_main.head()
```

Out[48]:

	RowNumber	CustomerId	Surname	CreditScore	Geography	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	EstimatedSalary	Exited	Gender_Female	Gender_Male
0	1	15634602	Hargrave	619	France	24	2	0.00	1	1	1	101348.88	1	1	0
1	2	15647311	Hill	608	Spain	23	1	83807.86	1	0	1	112542.58	0	1	0
2	3	15619304	Onio	502	France	24	8	159660.80	3	1	0	113931.57	1	1	0
3	4	15701354	Boni	699	France	21	1	0.00	2	0	0	93826.63	0	1	0
4	5	15737888	Mitchell	850	Spain	25	2	125510.82	1	1	1	79084.10	0	1	0

In [49]:
```
df_main.corr()
```

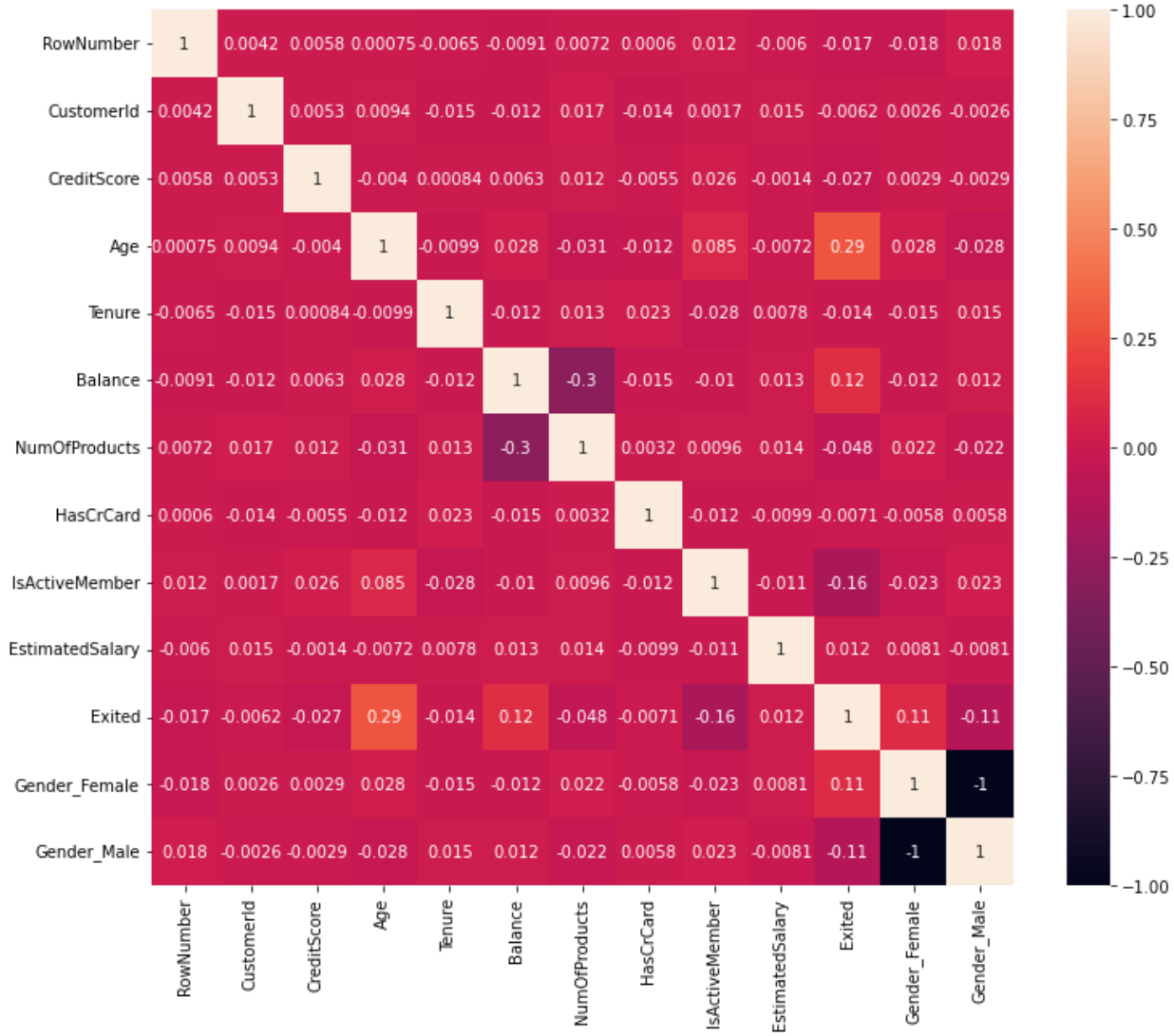
Out[49]:

	RowNumber	CustomerId	CreditScore	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	EstimatedSalary	Exited	Gender_Female	Gender_Male	
RowNumber	1.000000	0.004202	0.005840	0.000747	-0.006495	-0.009067	0.007246	0.000599	0.012044	0.000184	-0.016571	-0.018196	0.018196	
CustomerId	0.004202	1.000000	0.005308	0.009440	-0.014883	-0.012419	0.016972	-0.014025	0.001665	0.015271	-0.006248	0.002641	-0.002641	
CreditScore	0.005840	0.005308	1.000000	-0.004017	0.000842	0.006268	0.012238	-0.005458	0.025651	-0.001384	-0.027094	0.002857	-0.002857	
Age	0.000747	0.009440	-0.004017	1.000000	-0.009936	0.028275	-0.030707	-0.011697	0.085408	-0.007214	0.285537	0.027664	-0.027664	
Tenure	-0.006495	-0.014883	0.000842	-0.009936	1.000000	-0.012254	0.013444	0.022583	-0.028362	0.007784	-0.014001	-0.014733	0.014733	
Balance	-0.009067	-0.012419	0.006268	0.028275	-0.012254	1.000000	-0.304180	-0.014858	-0.010084	0.012797	0.118533	-0.012087	0.012087	
NumOfProducts	0.007246	0.016972	0.012238	-0.030707	0.013444	-0.304180	1.000000	0.003183	0.009612	0.014204	-0.047820	0.021859	-0.021859	
HasCrCard	0.000599	-0.014025	-0.005458	-0.011697	0.022583	-0.014858	0.003183	1.000000	-0.011866	-0.009933	-0.007138	-0.005766	0.005766	
IsActiveMember	0.012044	0.001665	0.025651	0.085408	-0.028362	-0.010084	0.009612	-0.011866	1.000000	-0.011421	-0.156128	-0.022544	0.022544	

	RowNumber	CustomerId	CreditScore	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	EstimatedSalary	Exited	Gender_Female	Gender_Male	
EstimatedSalary	-0.005988	0.015271	-0.001384	-0.007214	0.007784	0.012797	0.014204	-0.009933	-0.011421	1.000000	0.012097	0.008112	-0.008112	
Exited	-0.016571	-0.006248	-0.027094	0.285537	-0.014001	0.118533	-0.047820	-0.007138	-0.156128	0.012097	1.000000	0.106512	-0.106512	
Gender_Female	-0.018196	0.002641	0.002857	0.027664	-0.014733	-0.012087	0.021859	-0.005766	-0.022544	0.008112	0.106512	1.000000	-1.000000	
Gender_Male	0.018196	-0.002641	-0.002857	-0.027664	0.014733	0.012087	-0.021859	0.005766	0.022544	-0.008112	-0.106512	-1.000000	1.000000	

In [51]:
```
plt.figure(figsize=(12,10))
sns.heatmap(df_main.corr(),annot=True)
```

Out[51]: <AxesSubplot:>




```
In[53]: y=df_main['Tenure']
        y
```

Out[53]:

```
0          2
1          1
2          8
3          1
4          2
          ..
9995       5
9996      10
9997       7
9998       3
9999       4
Name: Tenure, Length: 10000, dtype: int64
```

In [59]:
```
X=df_main.drop(columns=['Tenure'],axis=1)
X.head()
```

Out[59]:

	RowNumber	CustomerId	Surname	CreditScore	Geography	Age	Balance	NumOfProducts	HasCrCard	IsActiveMember	EstimatedSalary	Exited	Gender_Female	Gender_Male
0	1	15634602	Hargrave	619	France	24	0.00	1	1	1	101348.88	1	1	0
1	2	15647311	Hill	608	Spain	23	83807.86	1	0	1	112542.58	0	1	0
2	3	15619304	Onio	502	France	24	159660.80	3	1	0	113931.57	1	1	0
3	4	15701354	Boni	699	France	21	0.00	2	0	0	93826.63	0	1	0
4	5	15737888	Mitchell	850	Spain	25	125510.82	1	1	1	79084.10	0	1	0

In [66]:
```
from sklearn.preprocessing import scale
```
In[69]:
```
X= df.drop(columns=['Age'])
X.head()
```

Out[69]:

	RowNumber	CustomerId	Surname	CreditScore	Geography	Gender	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	EstimatedSalary	Exited
0	1	15634602	Hargrave	619	France	Female	2	0.00	1	1	1	101348.88	1
1	2	15647311	Hill	608	Spain	Female	1	83807.86	1	0	1	112542.58	0
2	3	15619304	Onio	502	France	Female	8	159660.80	3	1	0	113931.57	1

	RowNumber	CustomerId	Surname	CreditScore	Geography	Gender	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	EstimatedSalary	Exited
3	4	15701354	Boni	699	France	Female	1	0.00	2	0	0	93826.63	0
4	5	15737888	Mitchell	850	Spain	Female	2	125510.82	1	1	1	79084.10	0

In [71]:
```
y=df.Age
y.head()
```

Out[71]:
```
0    24
1    23
2    24
3    21
4    25
Name: Age, dtype: int64
```

In [72]:
```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(X,y,test_size=0.2,random_state=0)
```

```
In[73]: x_train.shape
```

Out[73]: `(8000, 13)`

In [74]: `x_test.shape`

Out[74]: `(2000, 13)`

In [75]: `y_test`

Out[75]:
```
9394    17
898     22
2398    24
5906    14
2343    20
        ..
1037     6
2899    17
9549    18
2740    16
6690    12
```

```
Name: Age, Length: 2000, dtype: int64
```

In [76]:
```
from sklearn.linear_model import LinearRegression
```

In [77]:
```
MLR= LinearRegression()
```