ASSIGNMENT - 2

Data Visualization and Pre-processing

Load the dataset.

```
from google.colab import files
uploaded = files.upload()
```

```
<IPython.core.display.HTML object>
```

```
Saving Churn_Modelling.csv to Churn_Modelling.csv
```

Importing Libraries

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

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

```
df.head()
```

```
   RowNumber  CustomerId    Surname  CreditScore Geography  Gender
Age  \
0       1839    15758813   Campbell          350   Germany    Male
39
1       9625    15668309     Maslow          350    France  Female
40
2       8724    15803202  Onyekachi          350    France    Male
51
3       1632    15685372   Azubuike          350     Spain    Male
54
4       8763    15765173        Lin          350    France  Female
60

   Tenure    Balance  NumOfProducts  HasCrCard  IsActiveMember  \
0       0  109733.20              2          0               0
1       0  111098.85              1          1               1
2      10       0.00              1          1               1
3       1  152677.48              1          1               1
4       3       0.00              1          0               0

   EstimatedSalary  Exited
0        123602.11       1
1        172321.21       1
2        125823.79       1
3        191973.49       1
4        113796.15       1
```

```
df.info()
```

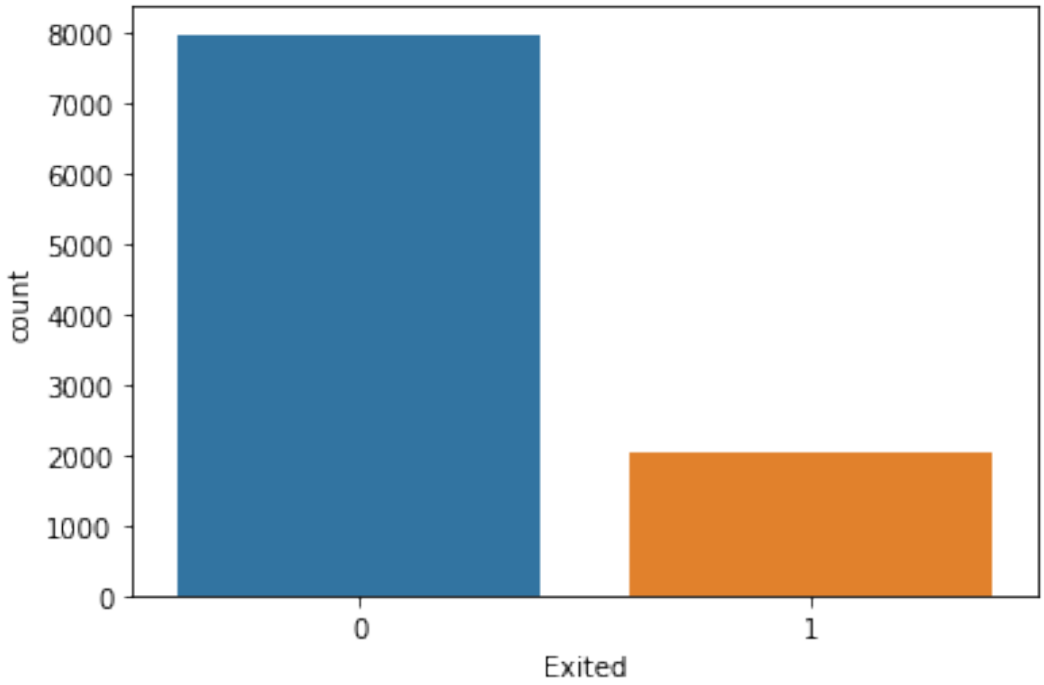
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10000 entries, 0 to 9999
Data columns (total 14 columns):
 #   Column           Non-Null Count  Dtype  
---  ------           --------------  -----  
 0   RowNumber        10000 non-null  int64  
 1   CustomerId       10000 non-null  int64  
 2   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
```

Perform Below Visualizations.

Univariate Analysis

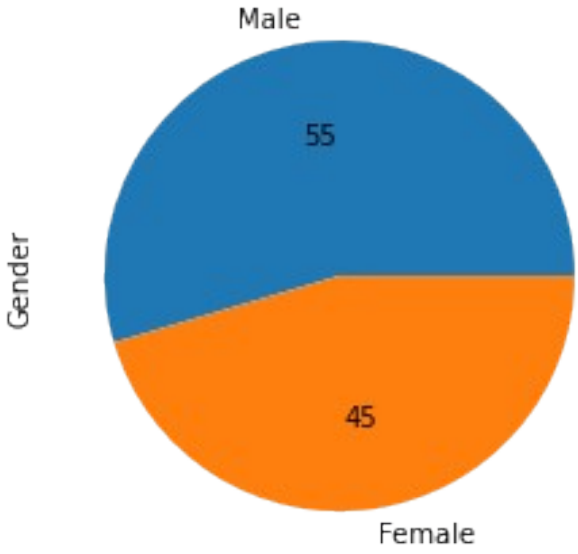
```
sns.countplot(x=df['Exited'])
df['Exited'].value_counts()
```

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




```
df['Gender'].value_counts().plot(kind='pie',autopct='%.0f')
```

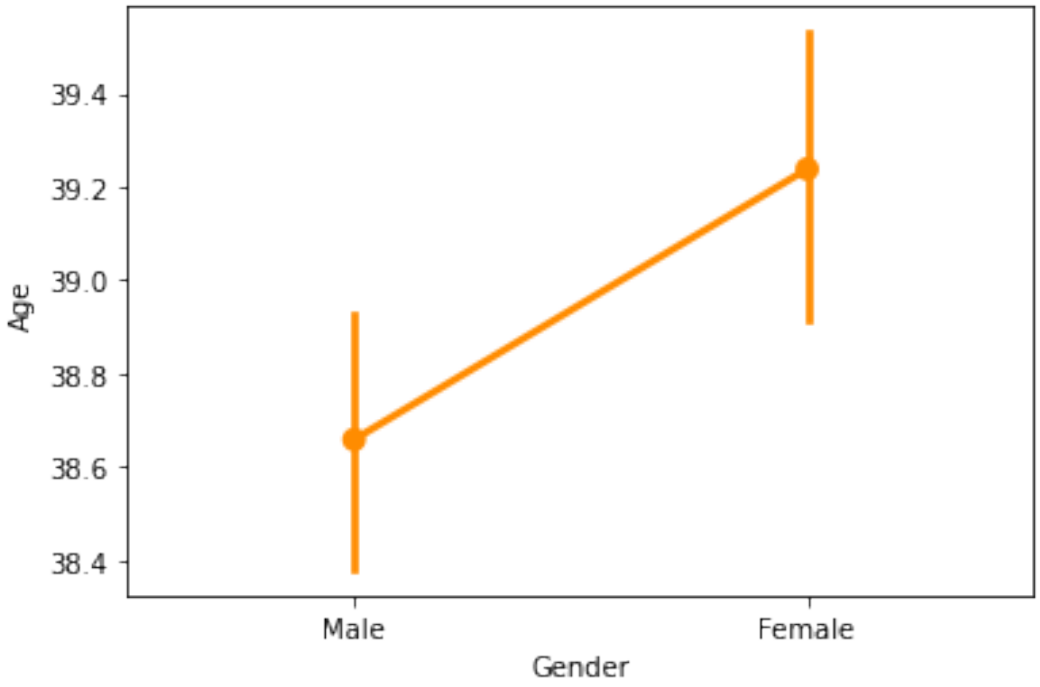
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f79285267d0>
```



Bi - Variate Analysis

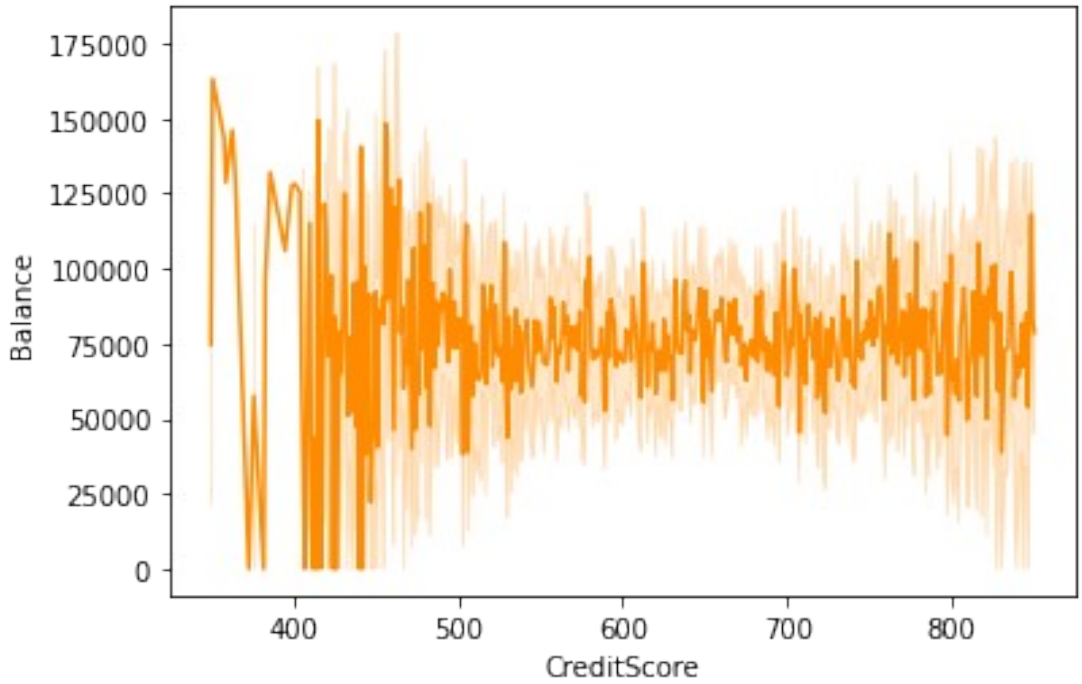
```
sns.pointplot(x='Gender',y='Age',data=df,color='darkorange')
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f7928485950>
```



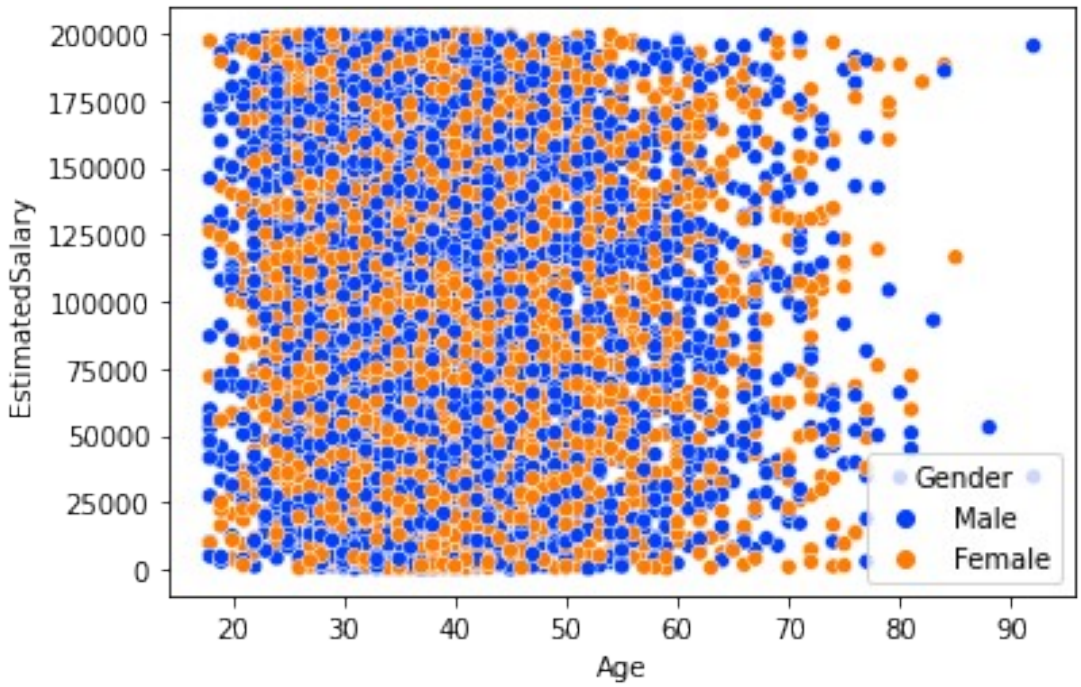

```
sns.lineplot(x=df['CreditScore'],y=df['Balance'],color='darkorange')
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f79283feed0>
```

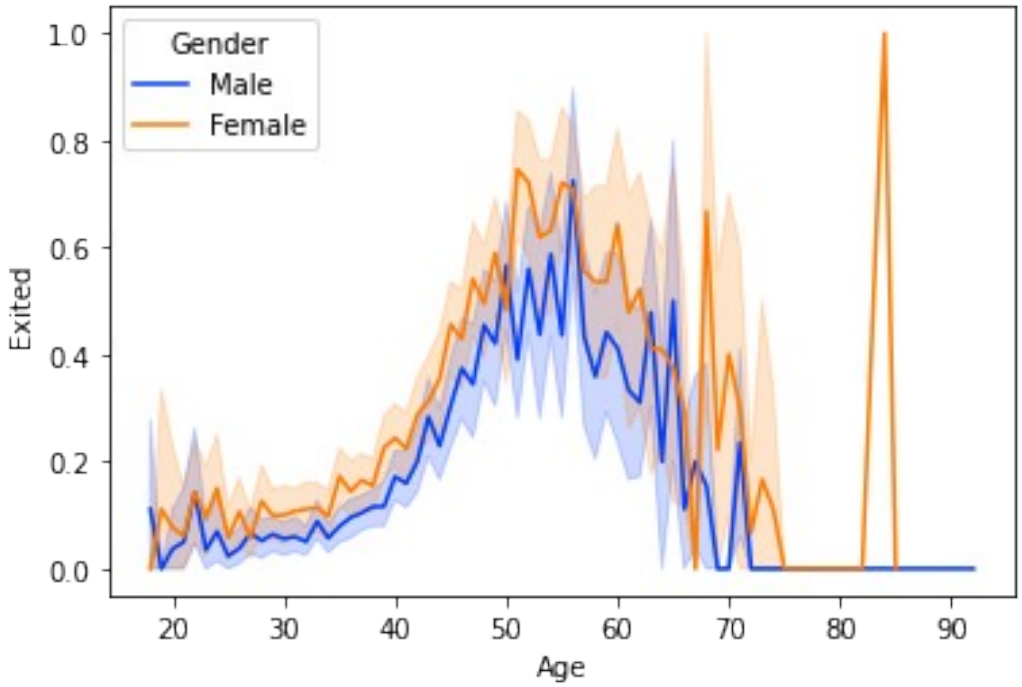


Multi - Variate Analysis

```
sns.scatterplot(
    x='Age',
    y='EstimatedSalary',
    data=df,
    palette='bright',
    hue='Gender');
```




```
sns.lineplot(
    x="Age",
    y="Exited",
    data=df,
    palette='bright',
    hue='Gender');
```




```
df.describe()
```

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

             Balance  NumOfProducts    HasCrCard  IsActiveMember  \
count   10000.000000   10000.000000  10000.00000    10000.000000
mean    76485.889288       1.530200      0.70550        0.515100
std     62397.405202       0.581654      0.45584        0.499797
min         0.000000       1.000000      0.00000        0.000000
25%         0.000000       1.000000      0.00000        0.000000
50%     97198.540000       1.000000      1.00000        1.000000
75%    127644.240000       2.000000      1.00000        1.000000
```

```
max    250898.090000       4.000000      1.00000        1.000000

       EstimatedSalary        Exited
count     10000.000000  10000.000000
mean     100090.239881      0.203700
std       57510.492818      0.402769
min          11.580000      0.000000
25%       51002.110000      0.000000
50%      100193.915000      0.000000
75%      149388.247500      0.000000
max      199992.480000      1.000000
```

```
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
```

Perform descriptive statistics on the dataset

```
df.sum()
```

```
RowNumber                                                     50005000
CustomerId                                                156909405694
Surname            CampbellMaslowOnyekachiAzubuikeLinChouAikenhea...
CreditScore                                                 6505288
Geography          GermanyFranceFranceSpainFranceGermanySpainFran...
Gender             MaleFemaleMaleMaleFemaleFemaleFemaleFemaleFema...
Age                                                          389218
Tenure                                                        50128
Balance                                                764858892.88
NumOfProducts                                                 15302
HasCrCard                                                      7055
IsActiveMember                                                 5151
EstimatedSalary                                         1000902398.81
Exited                                                         2037
dtype: object
```

```
df.mean(numeric_only=True)
```

```
RowNumber          5.000500e+03
CustomerId         1.569094e+07
CreditScore        6.505288e+02
Age                3.892180e+01
Tenure             5.012800e+00
Balance            7.648589e+04
NumOfProducts      1.530200e+00
HasCrCard          7.055000e-01
IsActiveMember     5.151000e-01
EstimatedSalary    1.000902e+05
Exited             2.037000e-01
dtype: float64
```

```
df.median(numeric_only=True)
```

```
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
```

```
df.mode(numeric_only=True)
```

```
      RowNumber  CustomerId  CreditScore   Age  Tenure  Balance  \
0             1    15565701        850.0  37.0     2.0      0.0
1             2    15565706          NaN   NaN     NaN      NaN
2             3    15565714          NaN   NaN     NaN      NaN
3             4    15565779          NaN   NaN     NaN      NaN
4             5    15565796          NaN   NaN     NaN      NaN
...         ...         ...          ...   ...     ...      ...
9995       9996    15815628          NaN   NaN     NaN      NaN
9996       9997    15815645          NaN   NaN     NaN      NaN
9997       9998    15815656          NaN   NaN     NaN      NaN
9998       9999    15815660          NaN   NaN     NaN      NaN
9999      10000    15815690          NaN   NaN     NaN      NaN

      NumOfProducts  HasCrCard  IsActiveMember  EstimatedSalary
Exited
0               1.0        1.0             1.0         24924.92
0.0
1               NaN        NaN             NaN              NaN
NaN
2               NaN        NaN             NaN              NaN
NaN
```

```
3                NaN        NaN              NaN              NaN
NaN
4                NaN        NaN              NaN              NaN
NaN
...              ...        ...              ...              ...     ..
.
9995             NaN        NaN              NaN              NaN
NaN
9996             NaN        NaN              NaN              NaN
NaN
9997             NaN        NaN              NaN              NaN
NaN
9998             NaN        NaN              NaN              NaN
NaN
9999             NaN        NaN              NaN              NaN
NaN

[10000 rows x 11 columns]
```

```
df.count()
```

```
RowNumber          10000
CustomerId         10000
Surname            10000
CreditScore        10000
Geography          10000
Gender             10000
Age                10000
Tenure             10000
Balance            10000
NumOfProducts      10000
HasCrCard          10000
IsActiveMember     10000
EstimatedSalary    10000
Exited             10000
dtype: int64
```

```
df.std(numeric_only=True)
```

```
RowNumber           2886.895680
CustomerId         71936.186123
CreditScore           96.653299
Age                   10.487806
Tenure                 2.892174
Balance            62397.405202
NumOfProducts          0.581654
HasCrCard              0.455840
IsActiveMember         0.499797
EstimatedSalary    57510.492818
Exited                 0.402769
dtype: float64
```

```
df.min()
```

```
RowNumber                1
CustomerId          15565701
Surname                Abazu
CreditScore              350
Geography             France
Gender                Female
Age                       18
Tenure                     0
Balance                  0.0
NumOfProducts              1
HasCrCard                  0
IsActiveMember             0
EstimatedSalary        11.58
Exited                     0
dtype: object
```

```
df.max()
```

```
RowNumber               10000
CustomerId           15815690
Surname                Zuyeva
CreditScore               850
Geography               Spain
Gender                   Male
Age                        92
Tenure                     10
Balance             250898.09
NumOfProducts               4
HasCrCard                   1
IsActiveMember              1
EstimatedSalary    199992.48
Exited                      1
dtype: object
```

Handle the Missing values

```
df.notnull()
```

```
      RowNumber  CustomerId  Surname  CreditScore  Geography  Gender
Age  \
0          True        True     True         True       True    True
True
1          True        True     True         True       True    True
True
2          True        True     True         True       True    True
True
3          True        True     True         True       True    True
True
4          True        True     True         True       True    True
```

```
True
...         ...         ...      ...          ...        ...     ...
...
9995       True        True     True         True       True    True
True
9996       True        True     True         True       True    True
True
9997       True        True     True         True       True    True
True
9998       True        True     True         True       True    True
True
9999       True        True     True         True       True    True
True

      Tenure  Balance  NumOfProducts  HasCrCard  IsActiveMember  \
0       True     True           True       True            True
1       True     True           True       True            True
2       True     True           True       True            True
3       True     True           True       True            True
4       True     True           True       True            True
...      ...      ...            ...        ...             ...
9995    True     True           True       True            True
9996    True     True           True       True            True
9997    True     True           True       True            True
9998    True     True           True       True            True
9999    True     True           True       True            True

      EstimatedSalary  Exited
0                True    True
1                True    True
2                True    True
3                True    True
4                True    True
...               ...     ...
9995             True    True
9996             True    True
9997             True    True
9998             True    True
9999             True    True

[10000 rows x 14 columns]

df.fillna(0)

      RowNumber  CustomerId    Surname  CreditScore Geography  Gender
Age  \
0          1839    15758813   Campbell          350   Germany    Male
39
1          9625    15668309     Maslow          350    France  Female
40
```

```
2          8724    15803202  Onyekachi          350    France    Male
51
3          1632    15685372   Azubuike          350     Spain    Male
54
4          8763    15765173        Lin          350    France  Female
60
...         ...         ...        ...          ...       ...     ...
...
9995       4464    15778975     Nnonso          850   Germany  Female
70
9996       8459    15728542  Vorobyova          850    France  Female
71
9997       9647    15603111       Muir          850     Spain    Male
71
9998       7527    15800554      Perry          850    France  Female
81
9999       7957    15731569     Hudson          850    France    Male
81

      Tenure    Balance  NumOfProducts  HasCrCard  IsActiveMember  \
0          0  109733.20              2          0               0
1          0  111098.85              1          1               1
2         10       0.00              1          1               1
3          1  152677.48              1          1               1
4          3       0.00              1          0               0
...      ...        ...            ...        ...             ...
9995       1   96947.58              3          1               0
9996       4       0.00              2          1               1
9997      10   69608.14              1          1               0
9998       1       0.00              2          1               1
9999       5       0.00              2          1               1

      EstimatedSalary  Exited
0           123602.11       1
1           172321.21       1
2           125823.79       1
3           191973.49       1
4           113796.15       1
...               ...     ...
9995         62282.99       1
9996        107236.87       0
9997         97893.40       1
9998         59568.24       0
9999         44827.47       0

[10000 rows x 14 columns]
```

FILLING NULL VALUES WITH PREVIOUS VALUES

```
df.fillna(method ='pad')
```

```
      RowNumber  CustomerId    Surname  CreditScore Geography  Gender
Age  \
0          1839    15758813   Campbell          350   Germany    Male
39
1          9625    15668309     Maslow          350    France  Female
40
2          8724    15803202  Onyekachi          350    France    Male
51
3          1632    15685372   Azubuike          350     Spain    Male
54
4          8763    15765173        Lin          350    France  Female
60
...         ...         ...        ...          ...       ...     ...
...
9995       4464    15778975     Nnonso          850   Germany  Female
70
9996       8459    15728542  Vorobyova          850    France  Female
71
9997       9647    15603111       Muir          850     Spain    Male
71
9998       7527    15800554      Perry          850    France  Female
81
9999       7957    15731569     Hudson          850    France    Male
81

      Tenure    Balance  NumOfProducts  HasCrCard  IsActiveMember  \
0          0  109733.20              2          0               0
1          0  111098.85              1          1               1
2         10       0.00              1          1               1
3          1  152677.48              1          1               1
4          3       0.00              1          0               0
...      ...        ...            ...        ...             ...
9995       1   96947.58              3          1               0
9996       4       0.00              2          1               1
9997      10   69608.14              1          1               0
9998       1       0.00              2          1               1
9999       5       0.00              2          1               1

      EstimatedSalary  Exited
0           123602.11       1
1           172321.21       1
2           125823.79       1
3           191973.49       1
4           113796.15       1
...               ...     ...
9995         62282.99       1
9996        107236.87       0
9997         97893.40       1
9998         59568.24       0
9999         44827.47       0
```

```
[10000 rows x 14 columns]
```

FILLING NULL VALUES WITH THE NEXT ONES:

```
df.fillna(method ='bfill')
```

```
      RowNumber  CustomerId    Surname  CreditScore Geography  Gender
Age  \
0          1839    15758813   Campbell          350   Germany    Male
39
1          9625    15668309     Maslow          350    France  Female
40
2          8724    15803202  Onyekachi          350    France    Male
51
3          1632    15685372   Azubuike          350     Spain    Male
54
4          8763    15765173        Lin          350    France  Female
60
...         ...         ...        ...          ...       ...     ...
...
9995       4464    15778975     Nnonso          850   Germany  Female
70
9996       8459    15728542  Vorobyova          850    France  Female
71
9997       9647    15603111       Muir          850     Spain    Male
71
9998       7527    15800554      Perry          850    France  Female
81
9999       7957    15731569     Hudson          850    France    Male
81

      Tenure    Balance  NumOfProducts  HasCrCard  IsActiveMember  \
0          0  109733.20              2          0               0
1          0  111098.85              1          1               1
2         10       0.00              1          1               1
3          1  152677.48              1          1               1
4          3       0.00              1          0               0
...      ...        ...            ...        ...             ...
9995       1   96947.58              3          1               0
9996       4       0.00              2          1               1
9997      10   69608.14              1          1               0
9998       1       0.00              2          1               1
9999       5       0.00              2          1               1

      EstimatedSalary  Exited
0           123602.11       1
1           172321.21       1
2           125823.79       1
3           191973.49       1
```

```
4           113796.15       1
...               ...     ...
9995         62282.99       1
9996        107236.87       0
9997         97893.40       1
9998         59568.24       0
9999         44827.47       0

[10000 rows x 14 columns]
```

Find the outliers and replace the outliers

```
qnt  =  df.quantile(q = (0.25,0.75))
iqr = qnt.loc[0.75] - qnt.loc[0.25]

iqr
```

```
RowNumber            4999.5000
CustomerId         124705.5000
CreditScore           134.0000
Age                    12.0000
Tenure                  4.0000
Balance            127644.2400
NumOfProducts           1.0000
HasCrCard               1.0000
IsActiveMember          1.0000
EstimatedSalary     98386.1375
Exited                  0.0000
dtype: float64
```

```
lower = qnt.loc[0.25] - 1.5*iqr
lower
```

```
RowNumber         -4.998500e+03
CustomerId         1.544147e+07
CreditScore        3.830000e+02
Age                1.400000e+01
Tenure            -3.000000e+00
Balance           -1.914664e+05
NumOfProducts     -5.000000e-01
HasCrCard         -1.500000e+00
IsActiveMember    -1.500000e+00
EstimatedSalary   -9.657710e+04
Exited             0.000000e+00
dtype: float64
```

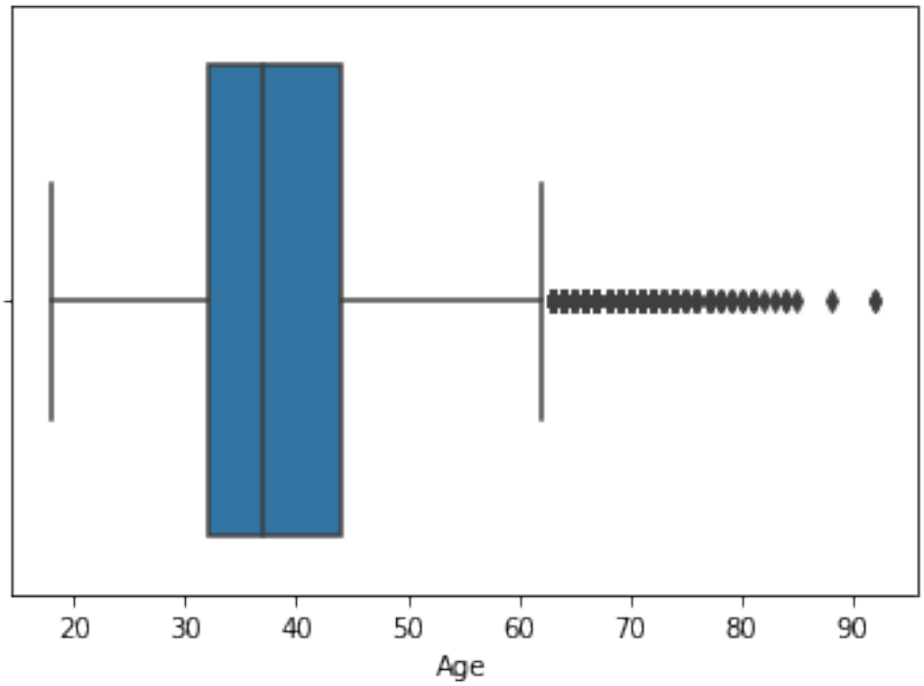
```
upper = qnt.loc[0.75] + 1.5 * iqr
upper
```

```
RowNumber          1.499950e+04
CustomerId         1.594029e+07
```

```
CreditScore        9.190000e+02
Age                6.200000e+01
Tenure             1.300000e+01
Balance            3.191106e+05
NumOfProducts      3.500000e+00
HasCrCard          2.500000e+00
IsActiveMember     2.500000e+00
EstimatedSalary    2.969675e+05
Exited             0.000000e+00
dtype: float64
```

```
sns.boxplot(x=df["Age"])
```

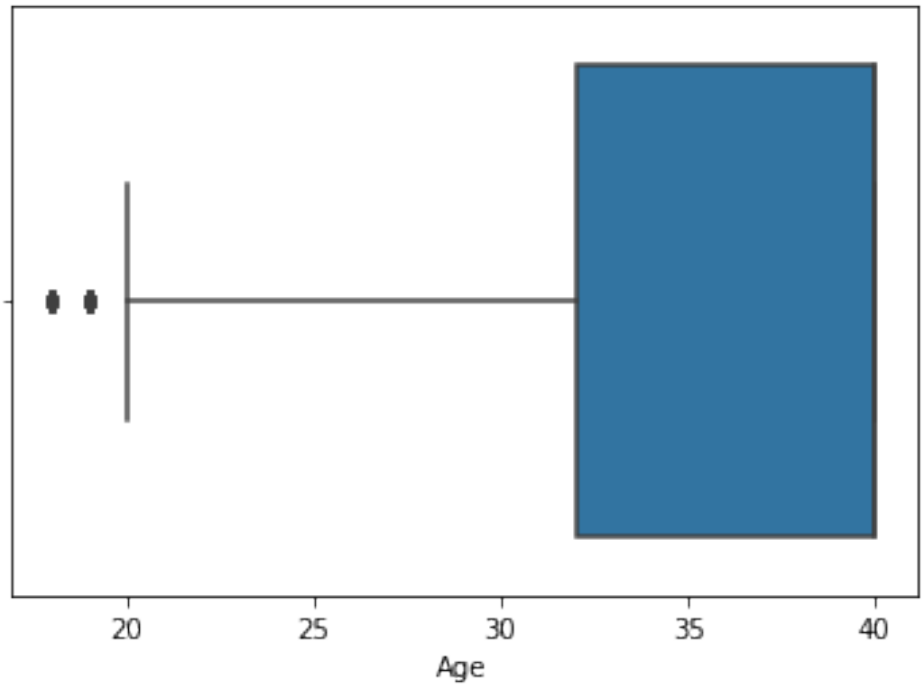
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f7925db2290>
```




```
df["Age"] = np.where(df["Age"]>35,40,df["Age"])
```

```
sns.boxplot(x=df["Age"])
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f7928aef050>
```



Check for Categorical columns and perform encoding

```
df.dtypes
```

```
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
```

```
df["Gender"].replace({"Female":0,"Male":1},inplace = True)
```

```
df.head(6)
```

```
   RowNumber  CustomerId    Surname  CreditScore Geography  Gender
Age  \
0       1839    15758813   Campbell          350   Germany       1
40
1       9625    15668309     Maslow          350    France       0
```

```
40
2       8724    15803202  Onyekachi          350    France       1
40
3       1632    15685372   Azubuike          350     Spain       1
40
4       8763    15765173        Lin          350    France       0
40
5       2474    15679249       Chou          351   Germany       0
40
```

```
   Tenure    Balance  NumOfProducts  HasCrCard  IsActiveMember  \
0       0  109733.20              2          0               0
1       0  111098.85              1          1               1
2      10       0.00              1          1               1
3       1  152677.48              1          1               1
4       3       0.00              1          0               0
5       4  163146.46              1          1               0

   EstimatedSalary  Exited
0        123602.11       1
1        172321.21       1
2        125823.79       1
3        191973.49       1
4        113796.15       1
5        169621.69       1
```

Split the data into dependent and independent variables

```
x= df.iloc[:,:-1].values
y= df.iloc[:,3].values
```

```
x
```

```
array([[1839, 15758813, 'Campbell', ..., 0, 0, 123602.11],
       [9625, 15668309, 'Maslow', ..., 1, 1, 172321.21],
       [8724, 15803202, 'Onyekachi', ..., 1, 1, 125823.79],
       ...,
       [9647, 15603111, 'Muir', ..., 1, 0, 97893.4],
       [7527, 15800554, 'Perry', ..., 1, 1, 59568.24],
       [7957, 15731569, 'Hudson', ..., 1, 1, 44827.47]], dtype=object)
```

```
y
```

```
array([350, 350, 350, ..., 850, 850, 850])
```

Scale the independent variables

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

```
credit_score = df[["CreditScore","EstimatedSalary"]]
```

```
scaler = StandardScaler()
scaler.fit(credit_score)
```

```
StandardScaler()
```

Split the data into training and testing

```
from sklearn.datasets import make_blobs
from sklearn.model_selection import train_test_split
g, k = make_blobs(n_samples=1000)

g_train, g_test, k_train, k_test = train_test_split(g, k,
test_size=0.33)
print(g_train.shape, g_test.shape, k_train.shape, k_test.shape)
```

```
(670, 2) (330, 2) (670,) (330,)
```