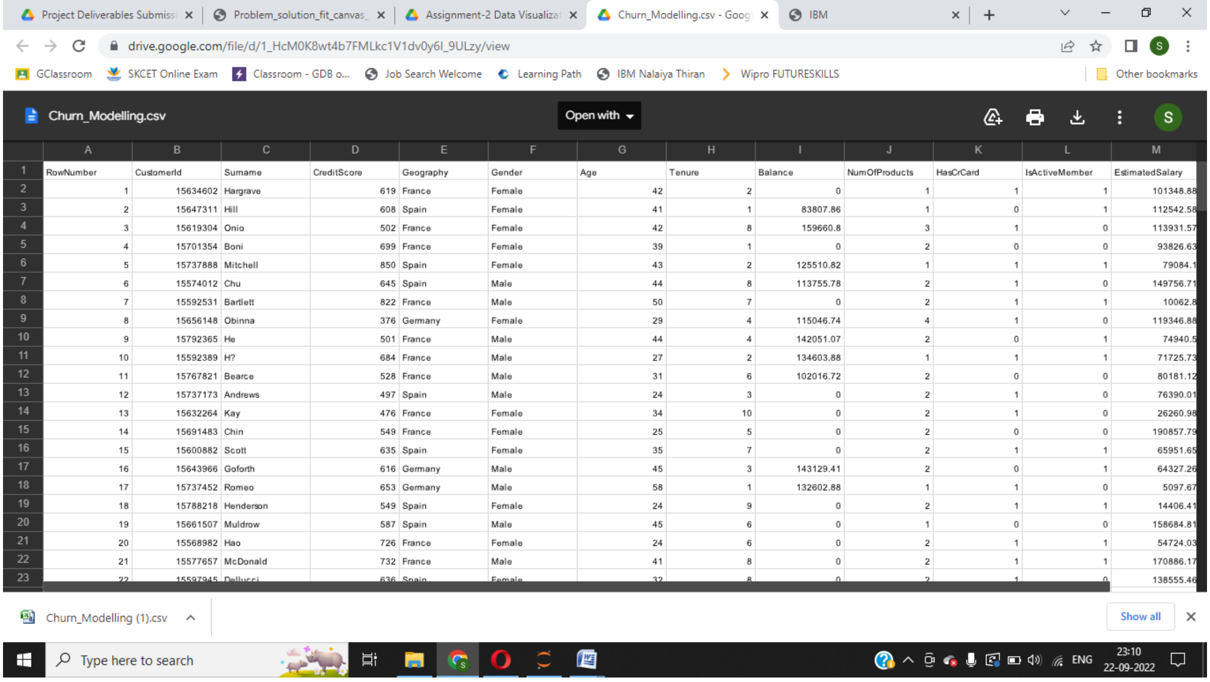
ASSIGNMENT -2
DATA VISUALIZATION AND DATA PRE-PROCESSING

Assignment Date	17 September 2022
Student Name	Mohana Sowdesh R
Student Roll Number	727719EUCS091
Maximum Marks	2 Marks

Question-1:

Download the dataset: Dataset



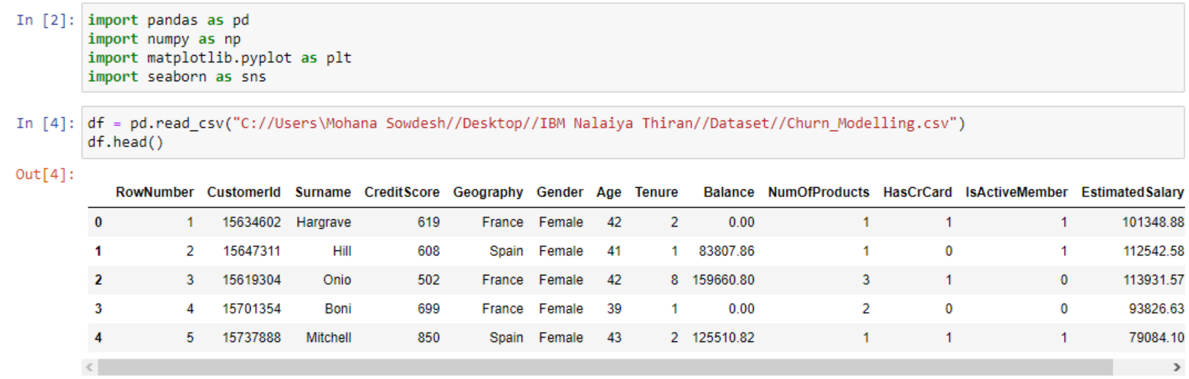
	A	B	C	D	E	F	G	H	I	J	K	L	M
1	RowNumber	CustomerId	Surname	CreditScore	Geography	Gender	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	EstimatedSalary
2	1	15634602	Hargrave	619	France	Female	42	2	0	1	1	1	101348.88
3	2	15647311	Hill	608	Spain	Female	41	1	83807.86	1	0	1	112542.58
4	3	15619304	Onio	502	France	Female	42	8	159660.8	3	1	0	113931.57
5	4	15701354	Boni	699	France	Female	39	1	0	2	0	0	93826.63
6	5	15737888	Mitchell	850	Spain	Female	43	2	125510.82	1	1	1	79084.1
7	6	15574012	Chu	645	Spain	Male	44	8	113755.78	2	1	0	149756.71
8	7	15592531	Bartlett	822	France	Male	50	7	0	2	1	1	10062.8
9	8	15656148	Obinna	376	Germany	Female	29	4	115046.74	4	1	0	119346.88
10	9	15792365	He	501	France	Male	44	4	142051.07	2	0	1	74940.5
11	10	15592389	H?	684	France	Male	27	2	134603.88	1	1	1	71725.73
12	11	15767821	Bearce	528	France	Male	31	6	102016.72	2	0	0	80181.12
13	12	15737173	Andrews	497	Spain	Male	24	3	0	2	1	0	76390.01
14	13	15632264	Kay	476	France	Female	34	10	0	2	1	0	26260.98
15	14	15691483	Chin	549	France	Female	25	5	0	2	0	0	190857.79
16	15	15600882	Scott	635	Spain	Female	35	7	0	2	1	1	65951.65
17	16	15643966	Goforth	616	Germany	Male	45	3	143129.41	2	0	1	64327.26
18	17	15737452	Romeo	653	Germany	Male	58	1	132602.88	1	1	0	5097.67
19	18	15788218	Henderson	549	Spain	Female	24	9	0	2	1	1	14406.41
20	19	15661507	Muldrow	587	Spain	Male	45	6	0	1	0	0	158684.81
21	20	15568982	Hao	726	France	Female	24	6	0	2	1	1	54724.03
22	21	15577657	McDonald	732	France	Male	41	8	0	2	1	1	170886.17
23	22	15597945	Dellucci	636	Spain	Female	32	8	0	2	1	0	138555.46

Question-2:

Load the dataset.

Solution:

```
import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns

df = pd.read_csv("C://Users\Mohana Sowdesh//Desktop//IBM Nalaiya Thiran// Dataset// Churn_ Modelling.csv")
df.head()
```



```
In [2]: import pandas as pd
        import numpy as np
        import matplotlib.pyplot as plt
        import seaborn as sns
```

```
In [4]: df = pd.read_csv("C://Users\Mohana Sowdesh//Desktop//IBM Nalaiya Thiran//Dataset//Churn_Modelling.csv")
        df.head()
```

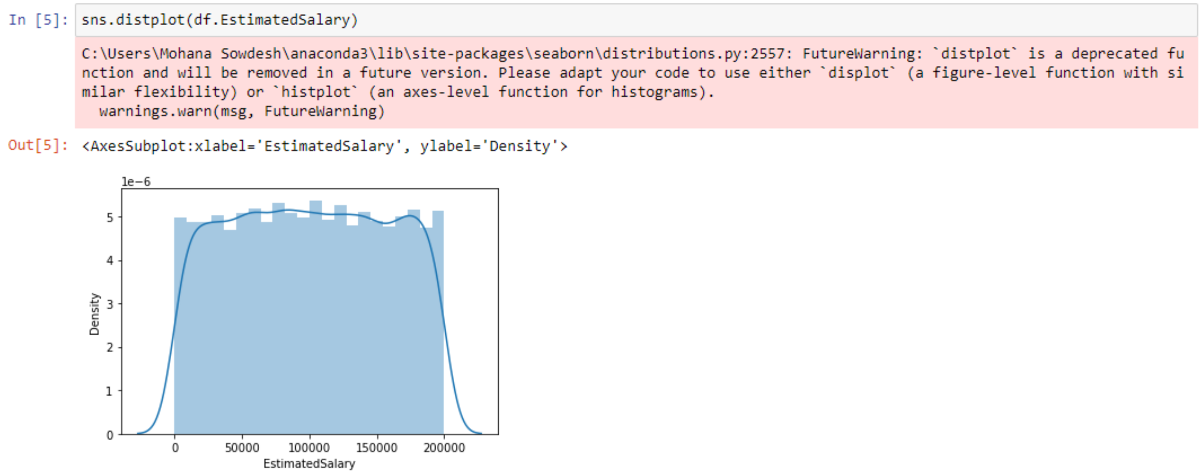
Out[4]:

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

Question-3:

Perform Below Visualizations.

- Univariate Analysis

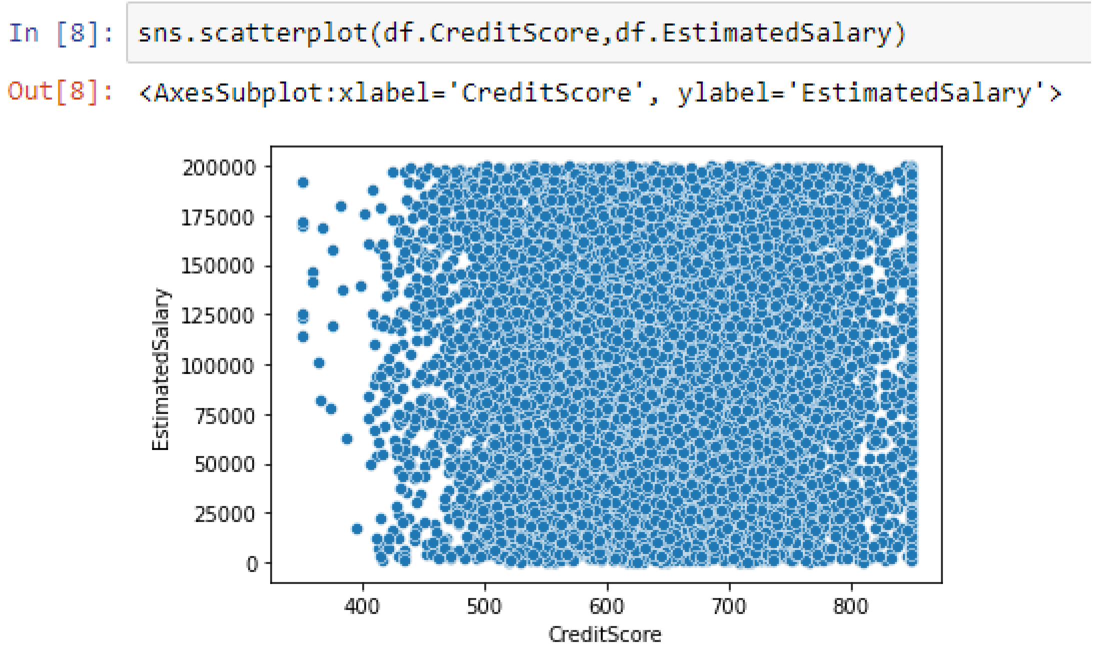


```
In [5]: sns.distplot(df.EstimatedSalary)
```

```
C:\Users\Mohana Sowdesh\anaconda3\lib\site-packages\seaborn\distributions.py:2557: FutureWarning: `distplot` is a deprecated fu
nction and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with si
milar flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

```
Out[5]: <AxesSubplot:xlabel='EstimatedSalary', ylabel='Density'>
```

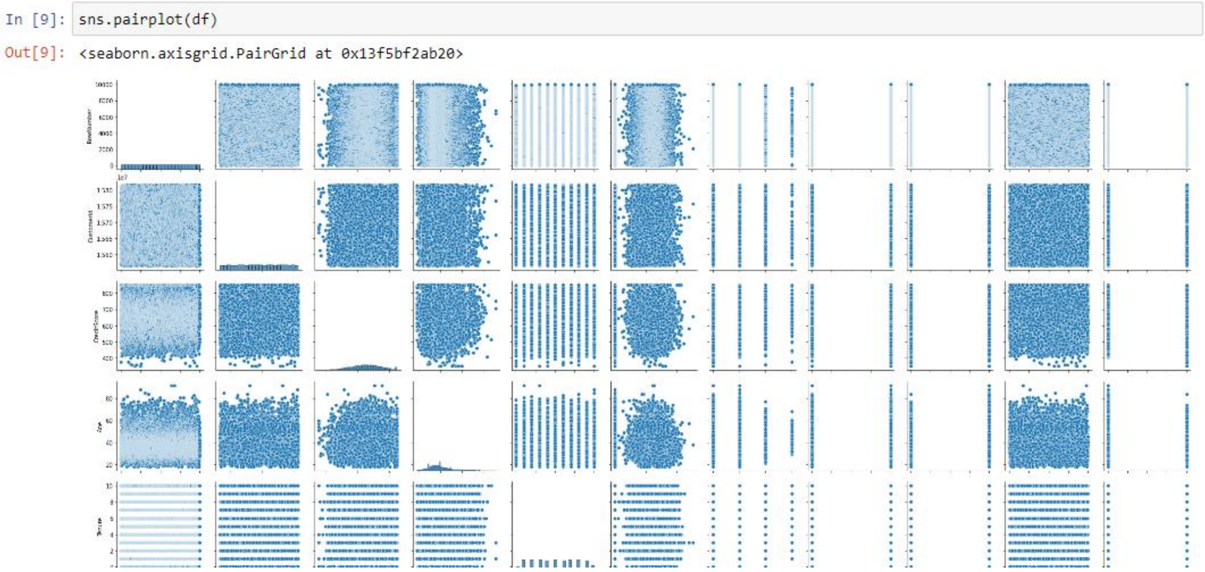
- Bi - Variate Analysis



```
In [8]: sns.scatterplot(df.CreditScore,df.EstimatedSalary)
```

```
Out[8]: <AxesSubplot:xlabel='CreditScore', ylabel='EstimatedSalary'>
```

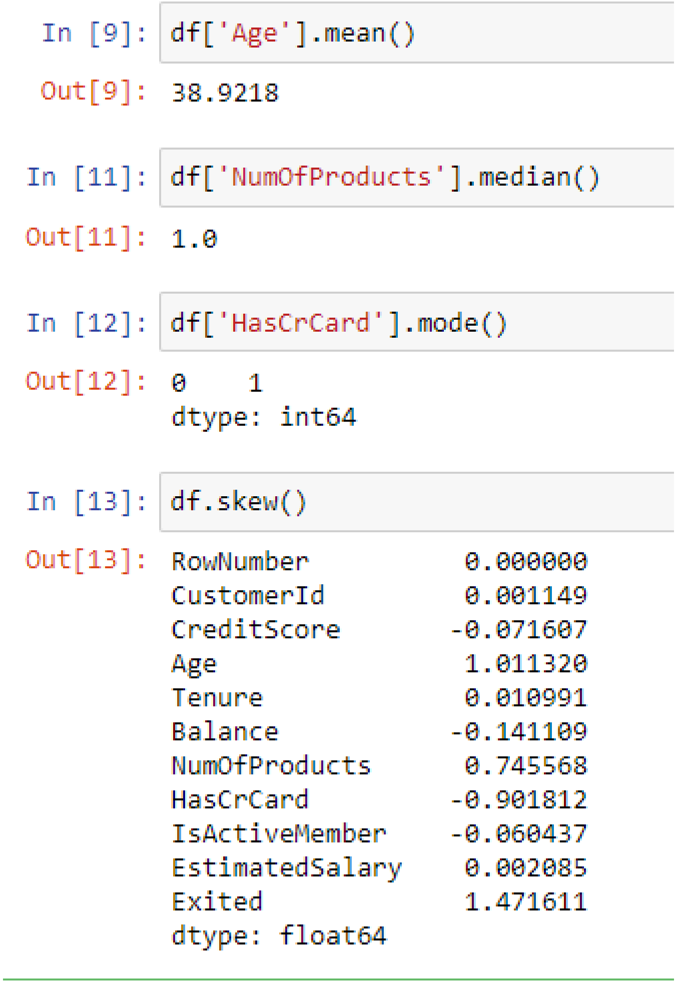
- Multi - Variate Analysis



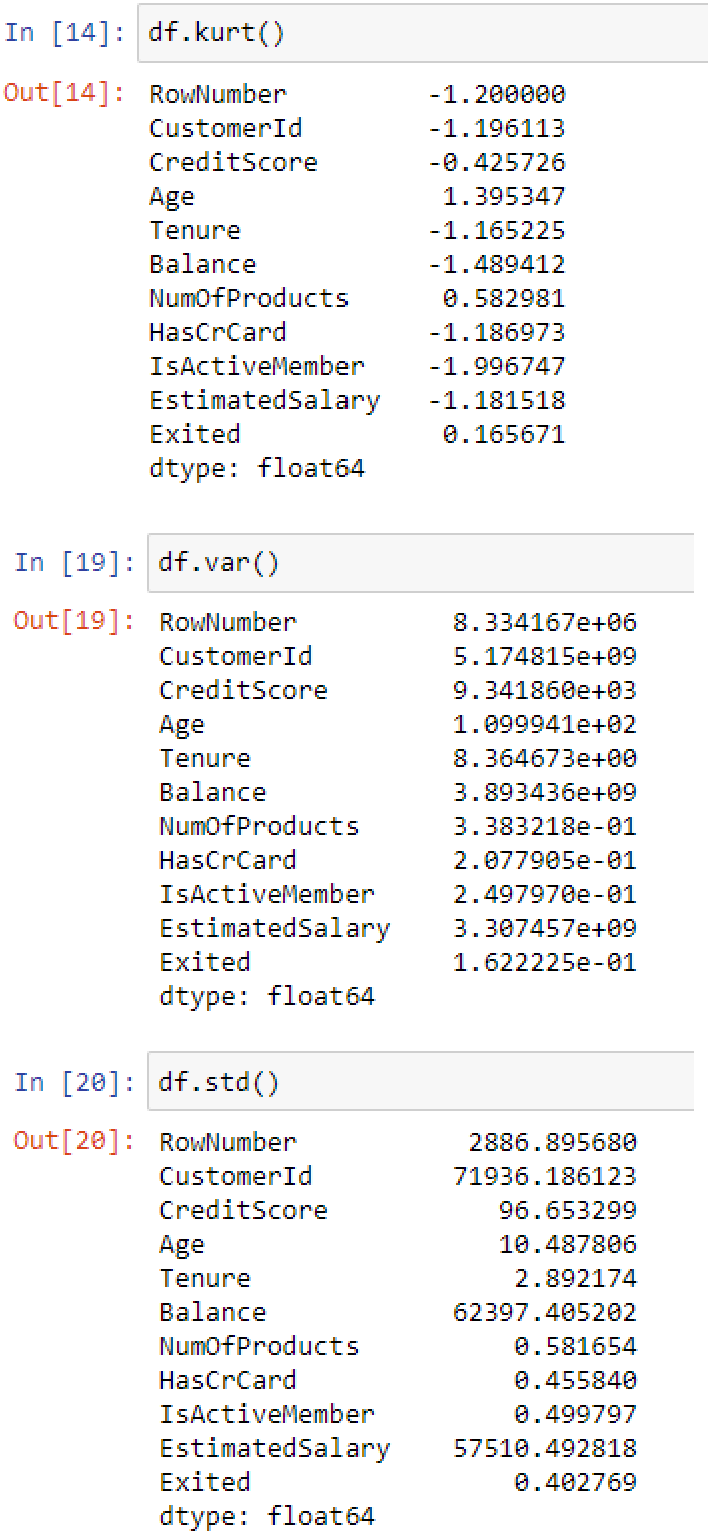
```
In [9]: sns.pairplot(df)

Out[9]: <seaborn.axisgrid.PairGrid at 0x13f5bf2ab20>
```

Question-4:

Perform descriptive statistics on the dataset.



```
In [9]: df['Age'].mean()

Out[9]: 38.9218

In [11]: df['NumOfProducts'].median()

Out[11]: 1.0

In [12]: df['HasCrCard'].mode()

Out[12]: 0    1
         dtype: int64

In [13]: df.skew()

Out[13]: RowNumber          0.000000
         CustomerId         0.001149
         CreditScore       -0.071607
         Age                1.011320
         Tenure             0.010991
         Balance           -0.141109
         NumOfProducts      0.745568
         HasCrCard         -0.901812
         IsActiveMember    -0.060437
         EstimatedSalary    0.002085
         Exited             1.471611
         dtype: float64
```



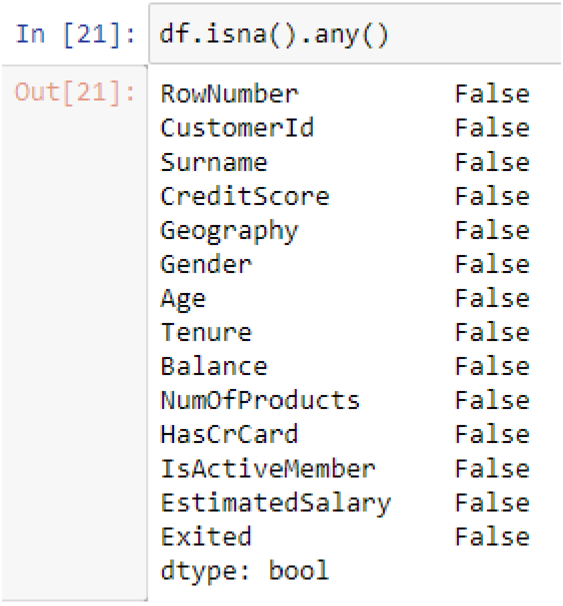
```
In [14]: df.kurt()

Out[14]: RowNumber         -1.200000
         CustomerId        -1.196113
         CreditScore       -0.425726
         Age                1.395347
         Tenure            -1.165225
         Balance           -1.489412
         NumOfProducts      0.582981
         HasCrCard         -1.186973
         IsActiveMember    -1.996747
         EstimatedSalary   -1.181518
         Exited             0.165671
         dtype: float64
```

```
In [19]: df.var()

Out[19]: RowNumber          8.334167e+06
         CustomerId         5.174815e+09
         CreditScore        9.341860e+03
         Age                1.099941e+02
         Tenure             8.364673e+00
         Balance            3.893436e+09
         NumOfProducts      3.383218e-01
         HasCrCard          2.077905e-01
         IsActiveMember     2.497970e-01
         EstimatedSalary    3.307457e+09
         Exited             1.622225e-01
         dtype: float64
```

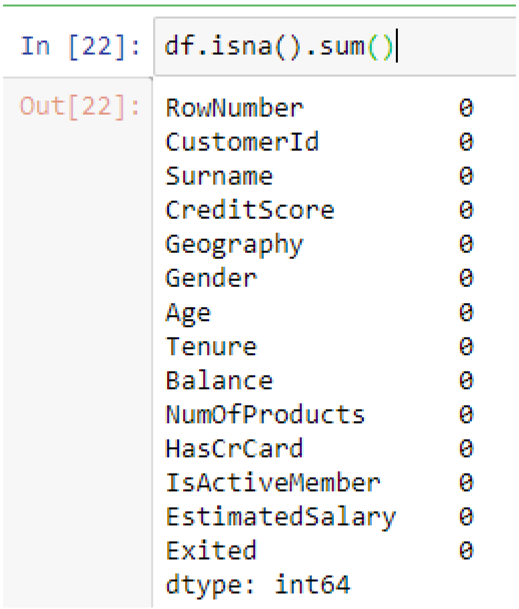
```
In [20]: df.std()

Out[20]: 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
```

Question-5:

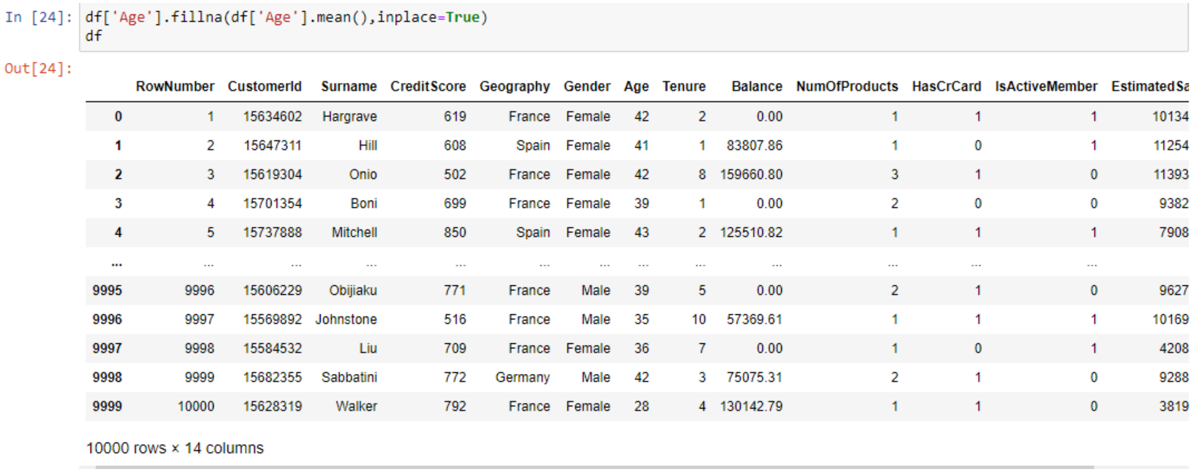
Handle the Missing values.



```
In [21]: df.isna().any()

Out[21]: 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 [22]: df.isna().sum()

Out[22]: 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
```



```
In [24]: df['Age'].fillna(df['Age'].mean(),inplace=True)
         df
```

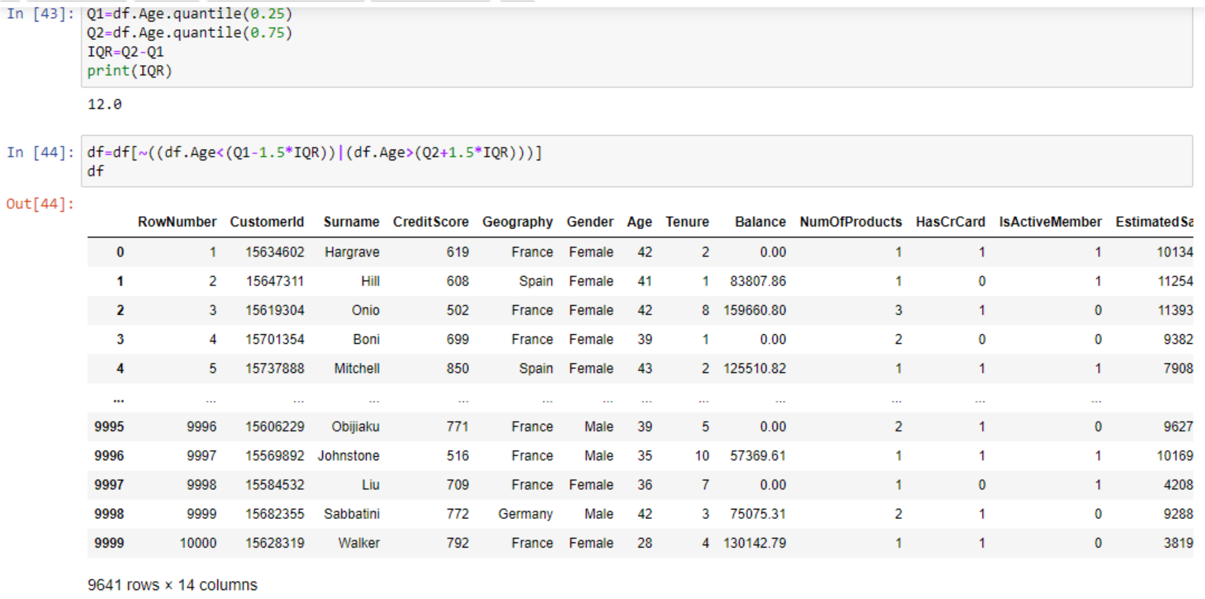
Out[24]:

	RowNumber	CustomerId	Surname	CreditScore	Geography	Gender	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	EstimatedSa
0	1	15634602	Hargrave	619	France	Female	42	2	0.00	1	1	1	10134
1	2	15647311	Hill	608	Spain	Female	41	1	83807.86	1	0	1	11254
2	3	15619304	Onio	502	France	Female	42	8	159660.80	3	1	0	11393
3	4	15701354	Boni	699	France	Female	39	1	0.00	2	0	0	9382
4	5	15737888	Mitchell	850	Spain	Female	43	2	125510.82	1	1	1	7908
...	...	...	...	...	...	...	...	...	...	...	...	...	
9995	9996	15606229	Obijiaku	771	France	Male	39	5	0.00	2	1	0	9627
9996	9997	15569892	Johnstone	516	France	Male	35	10	57369.61	1	1	1	10169
9997	9998	15584532	Liu	709	France	Female	36	7	0.00	1	0	1	4208
9998	9999	15682355	Sabbatini	772	Germany	Male	42	3	75075.31	2	1	0	9288
9999	10000	15628319	Walker	792	France	Female	28	4	130142.79	1	1	0	3819

10000 rows × 14 columns

Question-6:

Find the outliers and replace the outliers



```
In [43]: Q1=df.Age.quantile(0.25)
         Q2=df.Age.quantile(0.75)
         IQR=Q2-Q1
         print(IQR)

         12.0

In [44]: df=df[~((df.Age<(Q1-1.5*IQR))|(df.Age>(Q2+1.5*IQR)))]
         df
```

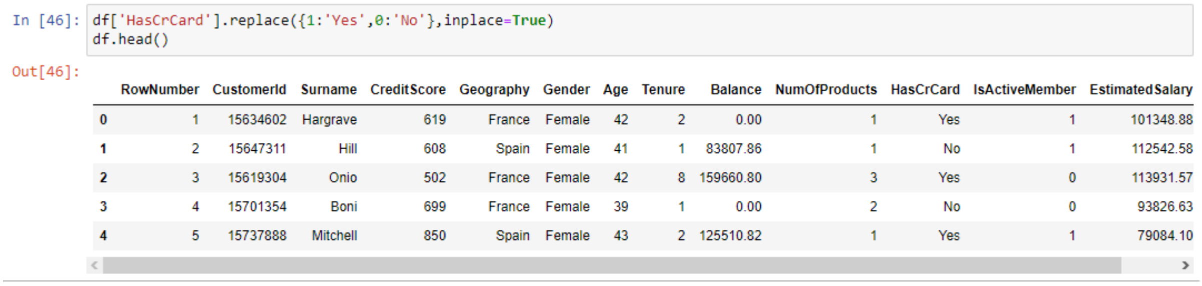
Out[44]:

	RowNumber	CustomerId	Surname	CreditScore	Geography	Gender	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	EstimatedSa
0	1	15634602	Hargrave	619	France	Female	42	2	0.00	1	1	1	10134
1	2	15647311	Hill	608	Spain	Female	41	1	83807.86	1	0	1	11254
2	3	15619304	Onio	502	France	Female	42	8	159660.80	3	1	0	11393
3	4	15701354	Boni	699	France	Female	39	1	0.00	2	0	0	9382
4	5	15737888	Mitchell	850	Spain	Female	43	2	125510.82	1	1	1	7908
...	...	...	...	...	...	...	...	...	...	...	...	...	
9995	9996	15606229	Obijiaku	771	France	Male	39	5	0.00	2	1	0	9627
9996	9997	15569892	Johnstone	516	France	Male	35	10	57369.61	1	1	1	10169
9997	9998	15584532	Liu	709	France	Female	36	7	0.00	1	0	1	4208
9998	9999	15682355	Sabbatini	772	Germany	Male	42	3	75075.31	2	1	0	9288
9999	10000	15628319	Walker	792	France	Female	28	4	130142.79	1	1	0	3819

9641 rows × 14 columns

Question-7:

Check for Categorical columns and perform encoding.



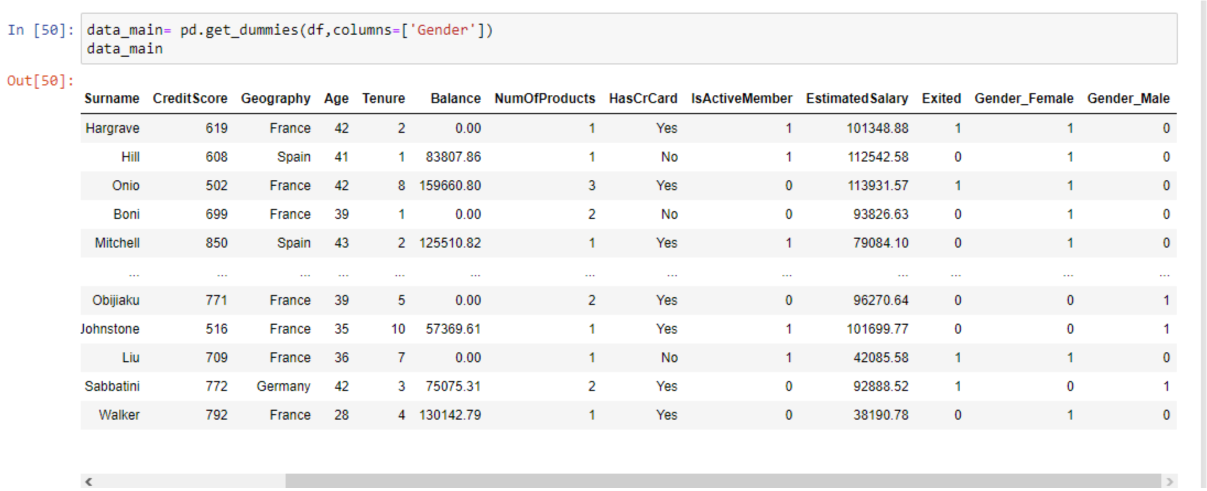
```
In [46]: df['HasCrCard'].replace({1:'Yes',0:'No'},inplace=True)
         df.head()
```

Out[46]:

	RowNumber	CustomerId	Surname	CreditScore	Geography	Gender	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	EstimatedSalary
0	1	15634602	Hargrave	619	France	Female	42	2	0.00	1	Yes	1	101348.88
1	2	15647311	Hill	608	Spain	Female	41	1	83807.86	1	No	1	112542.58
2	3	15619304	Onio	502	France	Female	42	8	159660.80	3	Yes	0	113931.57
3	4	15701354	Boni	699	France	Female	39	1	0.00	2	No	0	93826.63
4	5	15737888	Mitchell	850	Spain	Female	43	2	125510.82	1	Yes	1	79084.10

Question-8:

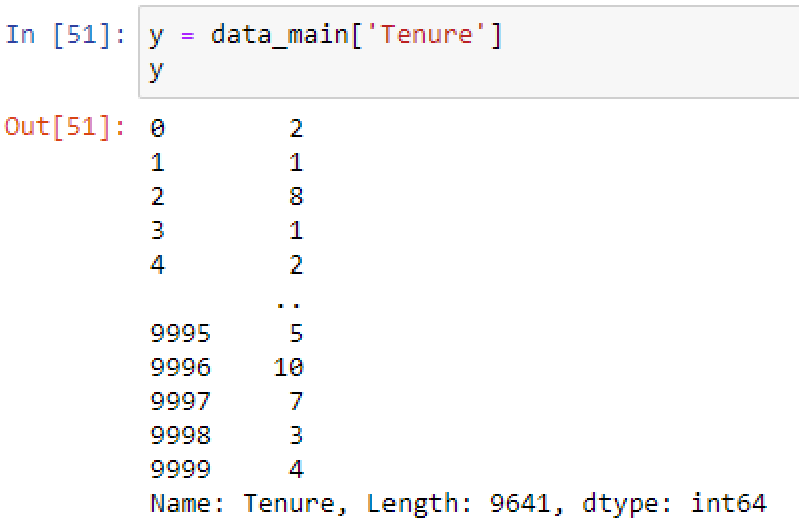
Split the data into dependent and independent variables.



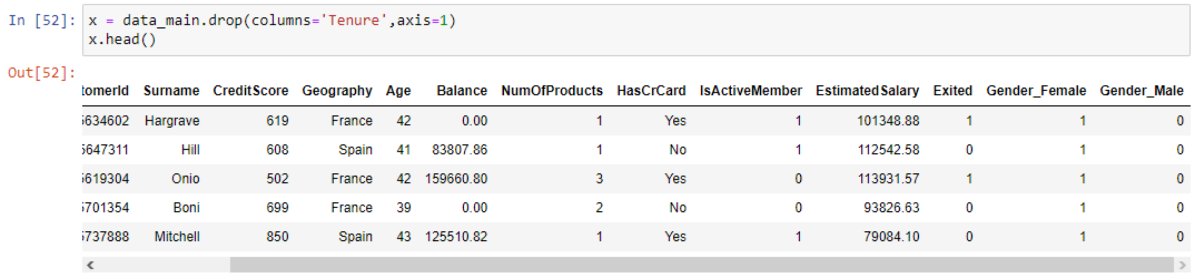
```
In [50]: data_main= pd.get_dummies(df,columns=['Gender'])
         data_main
```

Out[50]:

Surname	CreditScore	Geography	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	EstimatedSalary	Exited	Gender_Female	Gender_Male
Hargrave	619	France	42	2	0.00	1	Yes	1	101348.88	1	1	0
Hill	608	Spain	41	1	83807.86	1	No	1	112542.58	0	1	0
Onio	502	France	42	8	159660.80	3	Yes	0	113931.57	1	1	0
Boni	699	France	39	1	0.00	2	No	0	93826.63	0	1	0
Mitchell	850	Spain	43	2	125510.82	1	Yes	1	79084.10	0	1	0
...	...	...	...	...	...	...	...	...	...	...	...	...
Obijiaku	771	France	39	5	0.00	2	Yes	0	96270.64	0	0	1
Johnstone	516	France	35	10	57369.61	1	Yes	1	101699.77	0	0	1
Liu	709	France	36	7	0.00	1	No	1	42085.58	1	1	0
Sabbatini	772	Germany	42	3	75075.31	2	Yes	0	92888.52	1	0	1
Walker	792	France	28	4	130142.79	1	Yes	0	38190.78	0	1	0



```
In [51]: y = data_main['Tenure']
         y

Out[51]: 0        2
         1        1
         2        8
         3        1
         4        2
                 ..
         9995     5
         9996    10
         9997     7
         9998     3
         9999     4
         Name: Tenure, Length: 9641, dtype: int64
```



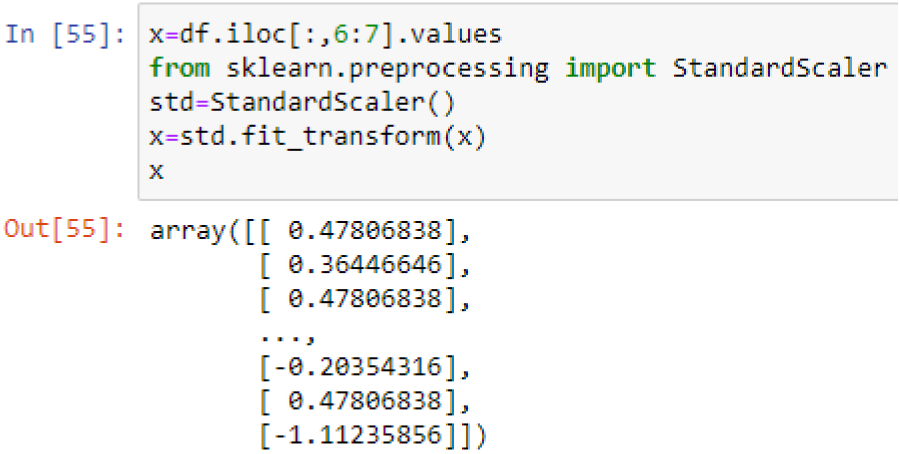
```
In [52]: x = data_main.drop(columns='Tenure',axis=1)
         x.head()
```

Out[52]:

tomerId	Surname	CreditScore	Geography	Age	Balance	NumOfProducts	HasCrCard	IsActiveMember	EstimatedSalary	Exited	Gender_Female	Gender_Male
634602	Hargrave	619	France	42	0.00	1	Yes	1	101348.88	1	1	0
647311	Hill	608	Spain	41	83807.86	1	No	1	112542.58	0	1	0
619304	Onio	502	France	42	159660.80	3	Yes	0	113931.57	1	1	0
701354	Boni	699	France	39	0.00	2	No	0	93826.63	0	1	0
737888	Mitchell	850	Spain	43	125510.82	1	Yes	1	79084.10	0	1	0

Question-9:

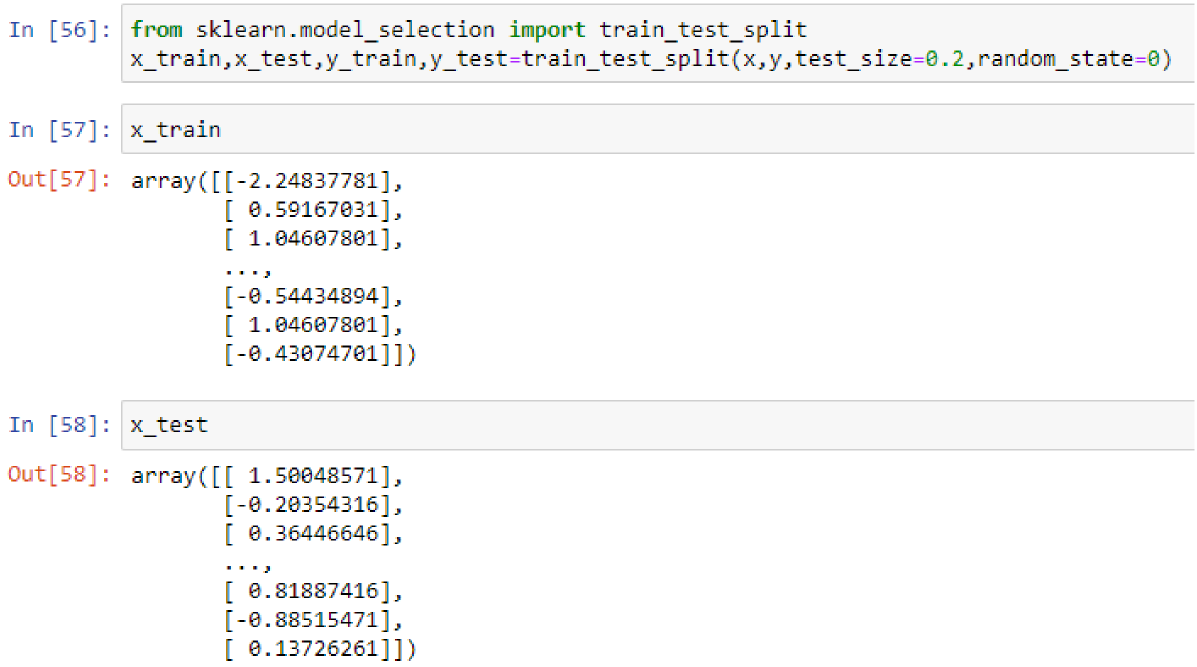
Scale the independent variables



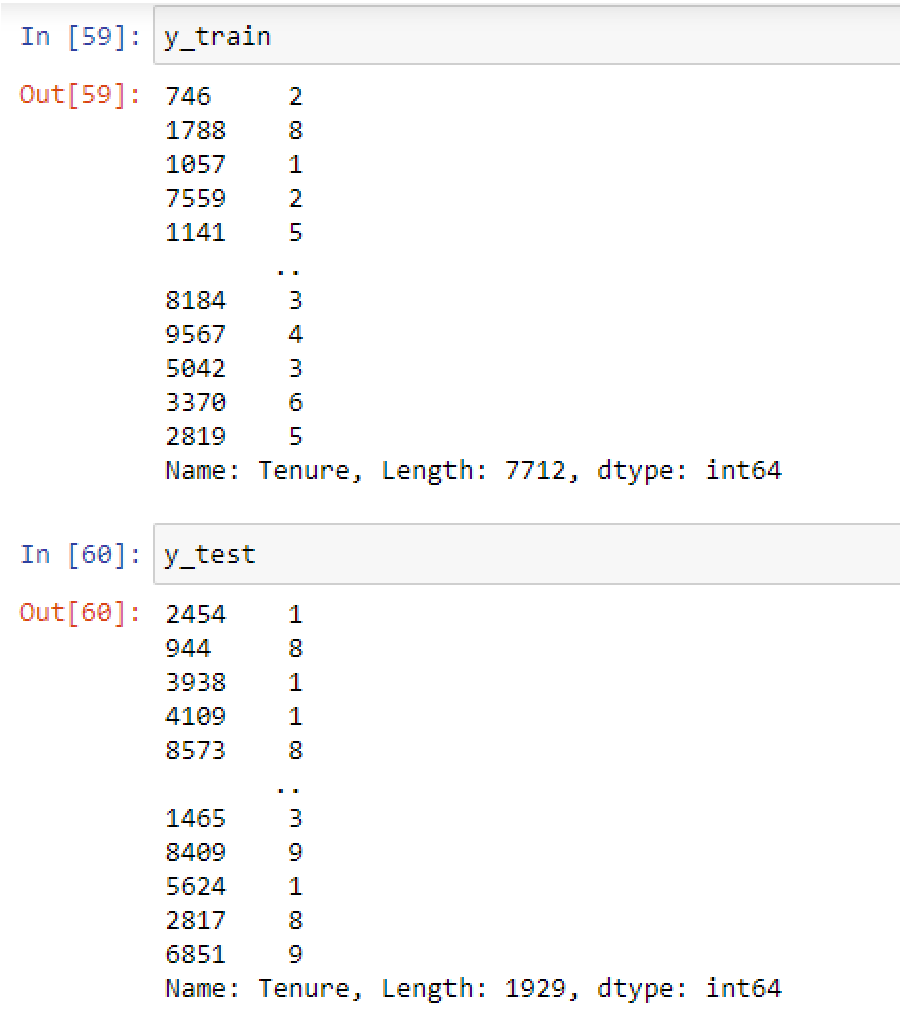
```
In [55]: x=df.iloc[:,6:7].values
         from sklearn.preprocessing import StandardScaler
         std=StandardScaler()
         x=std.fit_transform(x)
         x

Out[55]: array([[ 0.47806838],
                [ 0.36446646],
                [ 0.47806838],
                ...,
                [-0.20354316],
                [ 0.47806838],
                [-1.11235856]])
```

Question-10:

Split the data into training and testing



```
In [56]: 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 [57]: x_train

Out[57]: array([[-2.24837781],
                [ 0.59167031],
                [ 1.04607801],
                ...,
                [-0.54434894],
                [ 1.04607801],
                [-0.43074701]])

In [58]: x_test

Out[58]: array([[ 1.50048571],
                [-0.20354316],
                [ 0.36446646],
                ...,
                [ 0.81887416],
                [-0.88515471],
                [ 0.13726261]])
```



```
In [59]: y_train

Out[59]: 746     2
         1788    8
         1057    1
         7559    2
         1141    5
                ..
         8184    3
         9567    4
         5042    3
         3370    6
         2819    5
         Name: Tenure, Length: 7712, dtype: int64

In [60]: y_test

Out[60]: 2454    1
         944     8
         3938    1
         4109    1
         8573    8
                ..
         1465    3
         8409    9
         5624    1
         2817    8
         6851    9
         Name: Tenure, Length: 1929, dtype: int64
```