**ASSIGNMENT-4**

**TEAM ID-PNT2022TMID52462**

1. Download the dataset: Dataset

2. Load the dataset.

In [3]:import numpy as np import pandas as pd df = pd.read\_csv(

"/Churn\_Modelling.csv")

3. Perform Below Visualizations

● Univariate Analysis

In [4]:import seaborn as sns sns.histplot(df.EstimatedSalary,kde=True) Out[4]:

● Bi - Variate Analysis

In [ ]:import seaborn as sns import matplotlib.pyplot as plt sns.scatterplot(df

.Balance,df.EstimatedSalary) plt.ylim(0,15000)

C:\Users\ELCOT\anaconda3\lib\site-packages\seaborn\\_decorators.py:36:

3.Read dataset and do preprocessing

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[ ]:(0.0, 15000.0)

● Multi - Variate Analysis

4. Perform descriptive statistics on the dataset.

In [7]:df=pd.read\_csv("/Churn\_Modelling.csv") df.describe(include='all')

count 10000.00000 1.000000e+04 10000 10000.000000 10000 10000 10000.000000

10000.000000 10000.000000 10000.000000 10000.00000 10000.000000 10000.000000

10000.000000unique NaN NaN 2932 NaN 3 2 NaN NaN NaN NaN NaN NaN NaN NaNtop NaN

NaN Smith NaN France Male NaN NaN NaN NaN NaN NaN NaN NaNfreq NaN NaN 32 NaN

5014 5457 NaN NaN NaN NaN NaN NaN NaN NaNmean 5000.50000 1.569094e+07 NaN

650.528800 NaN NaN 38.921800 5.012800 76485.889288 1.530200 0.70550 0.515100

100090.239881 0.203700std 2886.89568 7.193619e+04 NaN 96.653299 NaN NaN

10.487806 2.892174 62397.405202 0.581654 0.45584 0.499797 57510.492818 0.402769

min 1.00000 1.556570e+07 NaN 350.000000 NaN NaN 18.000000 0.000000 0.000000

1.000000 0.00000 0.000000 11.580000 0.00000025% 2500.75000 1.562853e+07 NaN

584.000000 NaN NaN 32.000000 3.000000 0.000000 1.000000 0.00000 0.000000

51002.110000 0.00000050% 5000.50000 1.569074e+07 NaN 652.000000 NaN NaN

37.000000 5.000000 97198.540000 1.000000 1.00000 1.000000 100193.915000 0.000000

75% 7500.25000 1.575323e+07 NaN 718.000000 NaN NaN 44.000000 7.000000

127644.240000 2.000000 1.00000 1.000000 149388.247500 0.000000max 10000.00000

1.581569e+07 NaN 850.000000 NaN NaN 92.000000 10.000000 250898.090000 4.000000

1.00000 1.000000 199992.480000 1.00000In [8]:import seaborn as snsdf=pd.read\_csv

("/Churn\_Modelling.csv") sns.pairplot(df)from ast import increment\_linenoimport

pandas as pdimport numpy as npimport seaborn as snsimport matplotlib.pyplot as

plt%matplotlib inlinesns.set(color\_codes=True)

df=pd.read\_csv("/Churn\_Modelling.csv")df.head()RowNumber CustomerId Surname

CreditScore Geography Gender Age Tenure Balance NumOfProducts HasCrCard

IsActiveMember EstimatedSalary Exited0 1 15634602 Hargrave 619 France Female 42

2 0.00 1 1 1 101348.88 11 2 15647311 Hill 608 Spain Female 41 1 83807.86 1 0 1

112542.58 02 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 04 5 15737888

Mitchell 850 Spain Female 43 2 125510.82 1 1 1 79084.10 0

7. Check for Categorical columns and perform encoding.

In [11]:df=pd.read\_csv("/Churn\_Modelling.csv") df.columns import pandas as pd

import numpy as np headers=['RowNumber','CustomerID','Surname','CreditScore',

'Geography', 'Gender','Age','Tenure','Balance','NumofProducts','HasCard'

'IsActiveMember','EstimatedSalary','Exited'] import seaborn as sns df.head()

Out[11]:

RowNumberCustomerIdSurnameCreditScoreGeographyGenderAgeTenureBalance

NumOfProductsHasCrCardIsActiveMemberEstimatedSalaryExited

0115634602Hargrave619FranceFemale4220.00111101348.881

1215647311Hill608SpainFemale41183807.86101112542.580

2315619304Onio502FranceFemale428159660.80310113931.571

3415701354Boni699FranceFemale3910.0020093826.630

4515737888Mitchell850SpainFemale432125510.8211179084.1008. Split the data into

dependent and independent variables.x=df.iloc[:,:-1].valuesprint(x)

y=df.iloc[:,-1].\_valuesprint(y)[[1 15634602 'Hargrave' ... 1 1 101348.88] [2

15647311 'Hill' ... 0 1 112542.58] [3 15619304 'Onio' ... 1 0 113931.57] ...

[9998 15584532 'Liu' ... 0 1 42085.58] [9999 15682355 'Sabbatini' ... 1 0

92888.52] [10000 15628319 'Walker' ... 1 0 38190.78]][1 0 1 ... 1 1 0]

9. Scale the independent variables

In [13]:import seaborn as sns df=pd.read\_csv("/Churn\_Modelling.csv") dff=df[[

'Balance','Age']] sns.heatmap(dff.corr(), annot=True) sns.set(rc={

'figure.figsize':(40,40)}) 10. Split the data into training and testingfrom

scipy.sparse.construct import randomx=df.iloc[:, 1:2].values

y=df.iloc[:,2].valuesfrom 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)print('Row count of

x\_train table'+'-'+str(f"{len(x\_train):,}"))print('Row count of y\_train

table'+'-'+str(f"{len(y\_train):,}"))print('Row count of x\_test

table'+'-'+str(f"{len(x\_test):,}"))print('Row count of y\_test

table'+'-'+str(f"{len(y\_test):,}"))Row count of x\_train table-8,000Row count of

y\_train table-8,000Row count of x\_test table-2,000Row count of y\_test

table-2,000