# ASSIGNMENT -2

**Data Visualization and Pre-processing**

|  |  |
| --- | --- |
| **Assignment Date** | 24 September 2022 |
| **Team ID** | PNT2022TMID45335 |
| **Project Name** | AI BASED DISCOURSE FOR BANKING INDUSTRY |
| **Student Name** | B.M.DON DAVIES |
| **Student Roll Number** | E1195016 |
| **Maximum Marks** | 2 Marks |

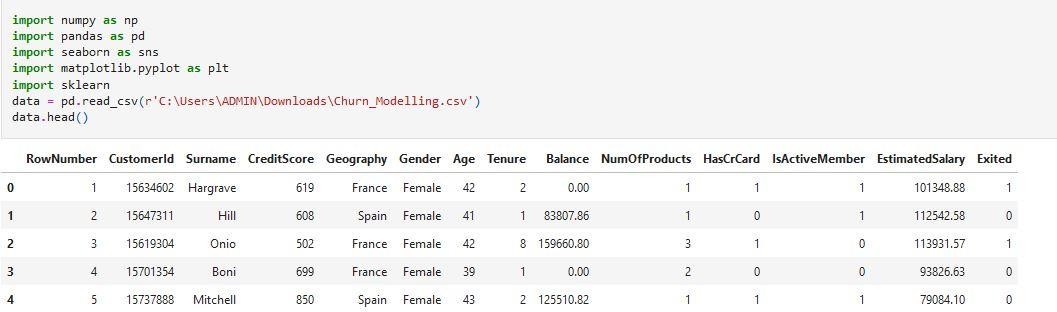
# Question 1

1.Download the Data set 2.Load The Dataset **Solution:**

# import numpy as np import pandas as pd import seaborn as sns

**import matplotlib.pyplot as plt import sklearn**

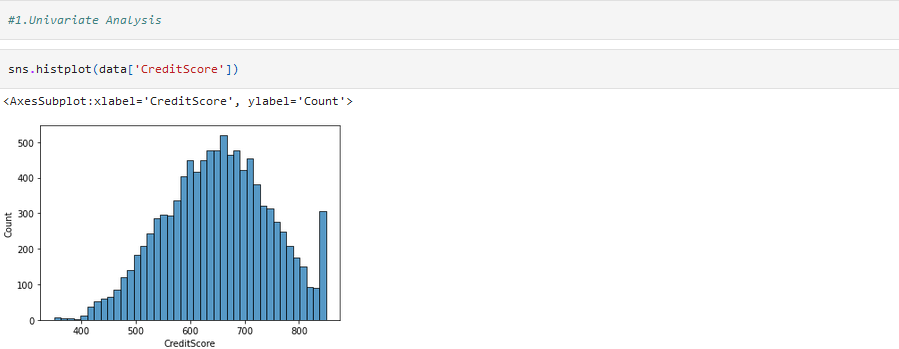
# data = pd.read\_csv(r'C:\Users\ADMIN\Downloads\Churn\_Modelling.csv') data.head()

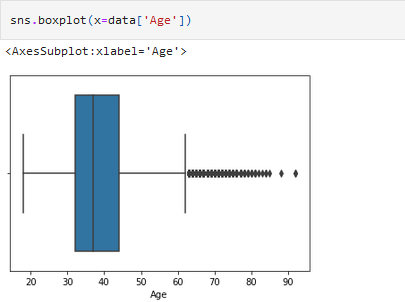


**Question-3**

3.Perform Below Visualizations

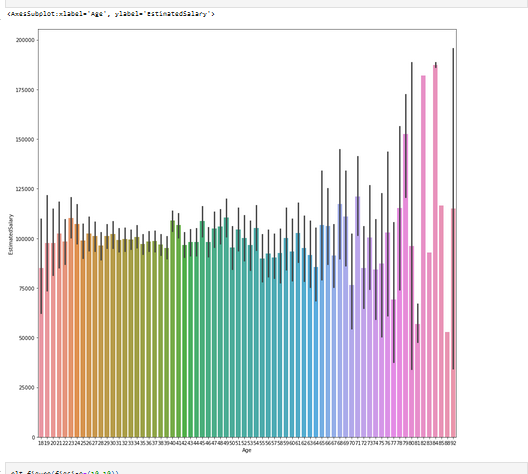
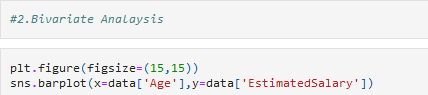
**Solution:**

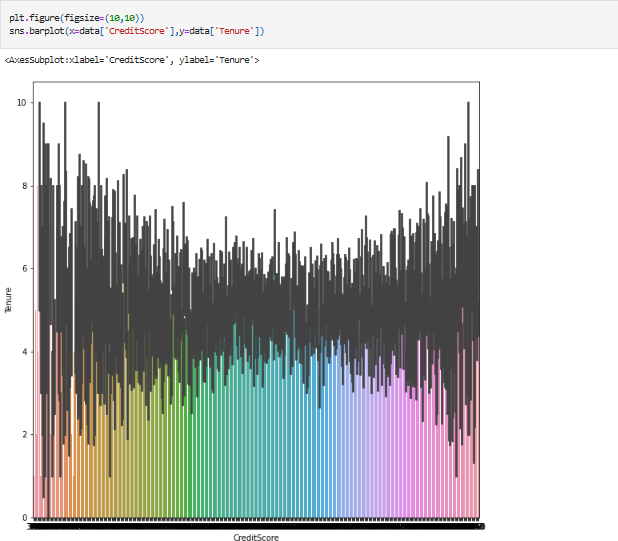
#1.Univariate Analysis sns.histplot(data['CreditScore'])

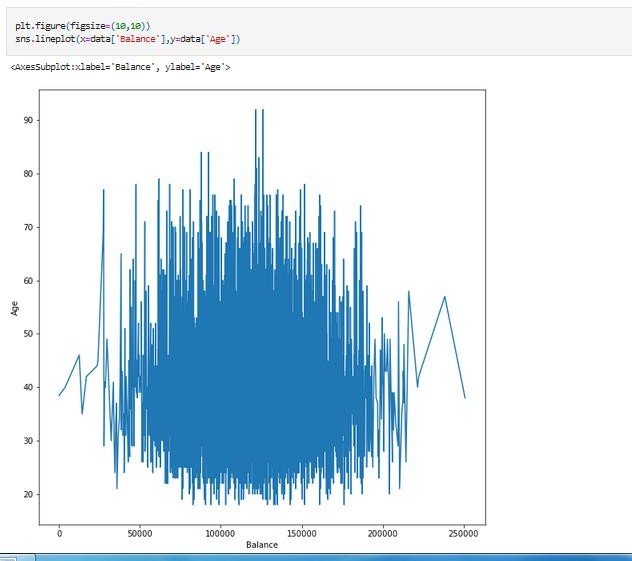


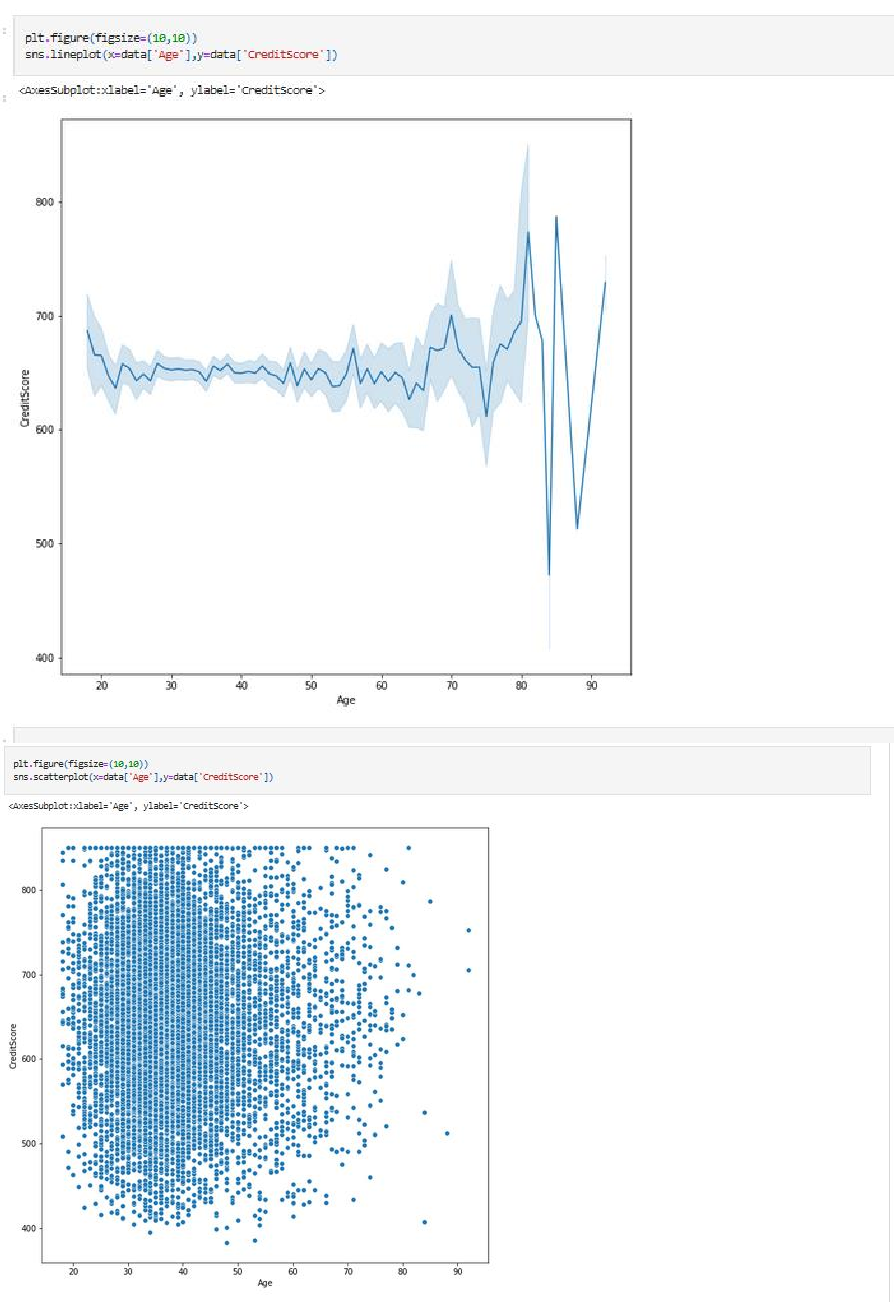
#2.Bivariate Analaysis

plt.figure(figsize=(15,15)) sns.barplot(x=data['Age'],y=data['EstimatedSalary'])









#3.Multi variate analysis

sns.pairplot(data= data [['Age','EstimatedSalary','Tenure']],hue='Age')



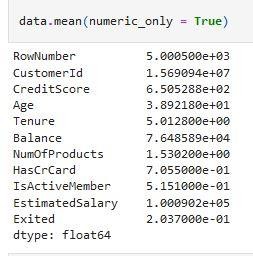


# Question.4:

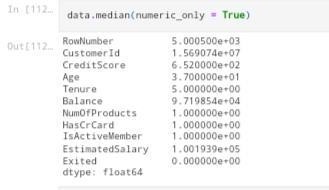
Perform descriptive statistics on the dataset

**Solution:**

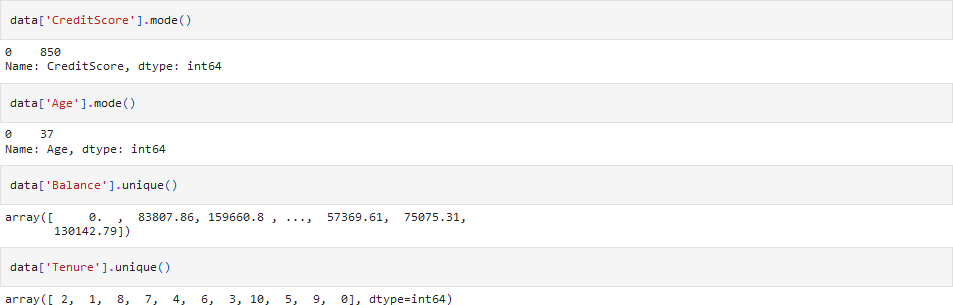
data.mean(numeric\_only = True)



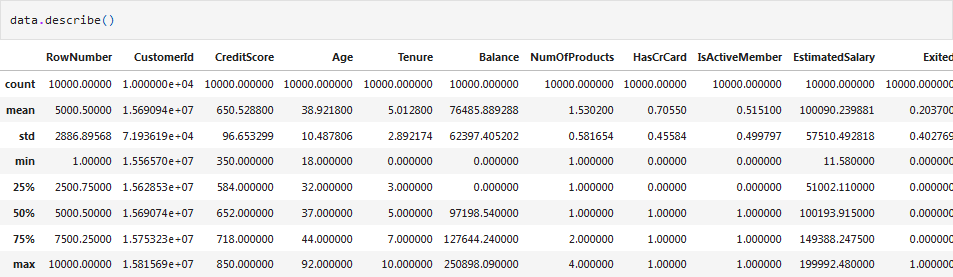
data.median(numeric\_only = True)



data['CreditScore'].mode() data['Age'].mode() data['Balance'].unique() data['Tenure'].unique() data.std(numeric\_only=True)



data.describe()



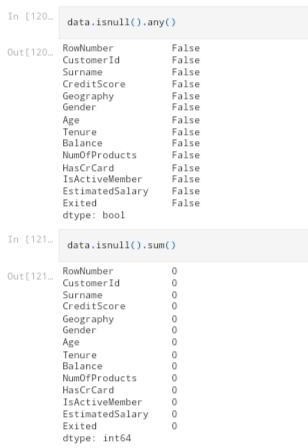
data['NumOfProducts'].value\_counts()



# Question.5

Handle the Missing values **Solution:** data.isnull().any()

data.isnull().sum()

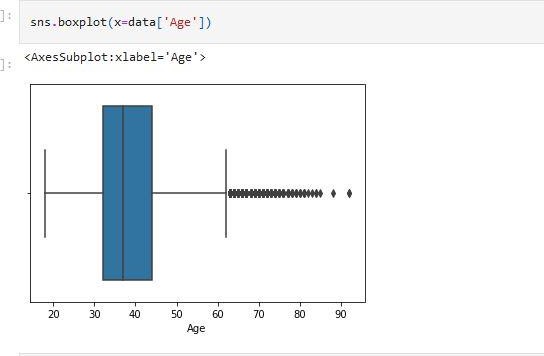


# Question.6

Find the outliers and replace the outliers

**Solution:**

sns.boxplot(x=data['Age'])



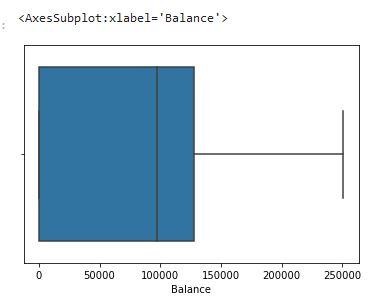
fig, ax = plt.subplots(figsize = (5,3)) #Outlier detection - Scatter plot ax.scatter(data['Balance'], data['Exited'])

# x-axis label ax.set\_xlabel('Balance')

# y-axis label ax.set\_ylabel('Exited') plt.show()

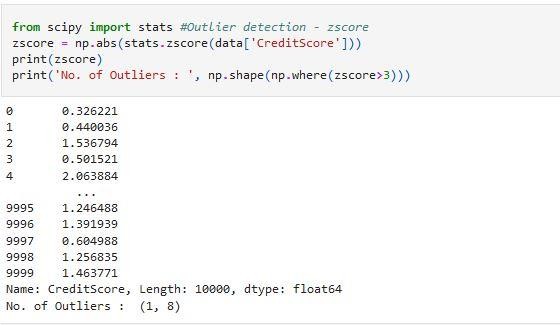
sns.boxplot(x=data['Balance'])



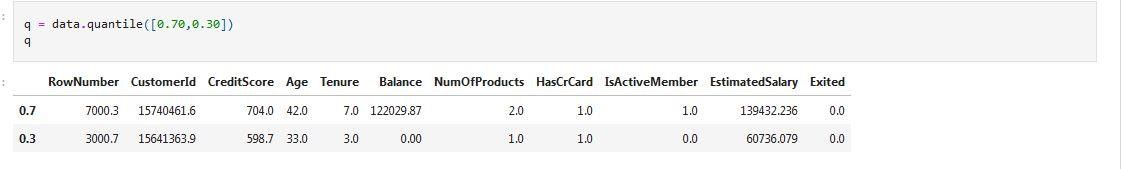


from scipy import stats #Outlier detection - zscore zscore = np.abs(stats.zscore(data['CreditScore'])) print(zscore)

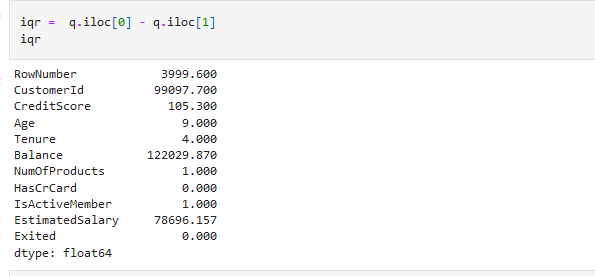
print('No. of Outliers : ', np.shape(np.where(zscore>3)))



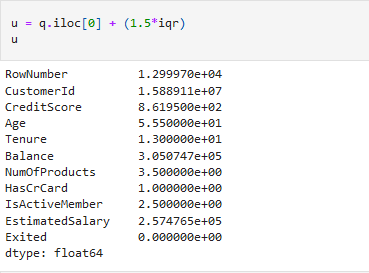
q = data.quantile([0.70,0.30]) q



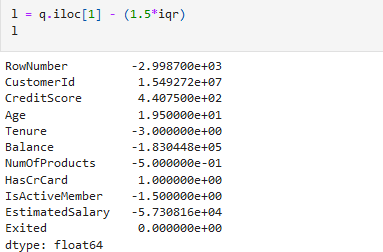
iqr = q.iloc[0] - q.iloc[1] iqr



u = q.iloc[0] + (1.5\*iqr) u



l = q.iloc[1] - (1.5\*iqr) l



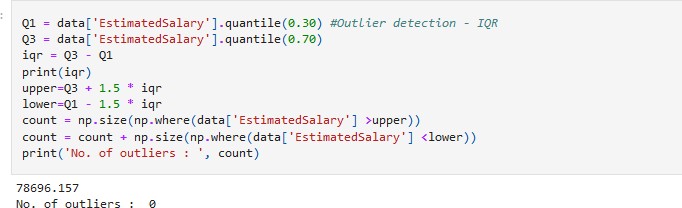
Q1 = data['EstimatedSalary'].quantile(0.30) #Outlier detection - IQR Q3 = data['EstimatedSalary'].quantile(0.70)

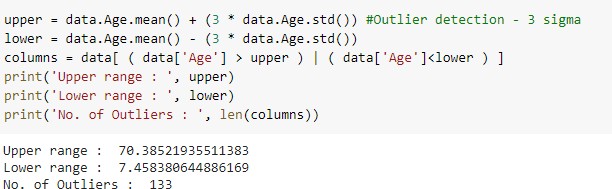
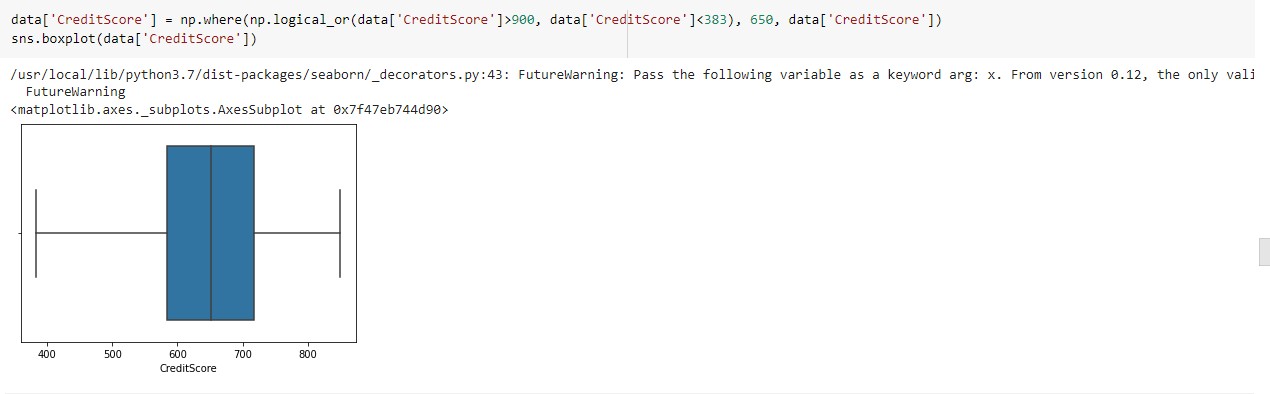
iqr = Q3 - Q1 print(iqr)

upper=Q3 + 1.5 \* iqr lower=Q1 - 1.5 \* iqr

count = np.size(np.where(data['EstimatedSalary'] >upper))

count = count + np.size(np.where(data['EstimatedSalary'] <lower)) print('No. of outliers : ', count)





columns = ['EstimatedSalary', 'Balance', 'Tenure'] #After outlier removal

for i in columns:

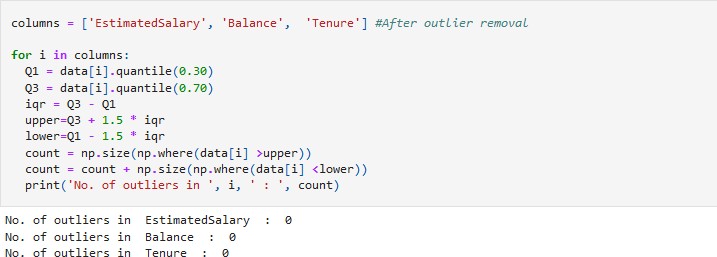
Q1 = data[i].quantile(0.30) Q3 = data[i].quantile(0.70) iqr = Q3 - Q1

upper=Q3 + 1.5 \* iqr lower=Q1 - 1.5 \* iqr

count = np.size(np.where(data[i] >upper))

count = count + np.size(np.where(data[i] <lower))

print('No. of outliers in ', i, ' : ', count)



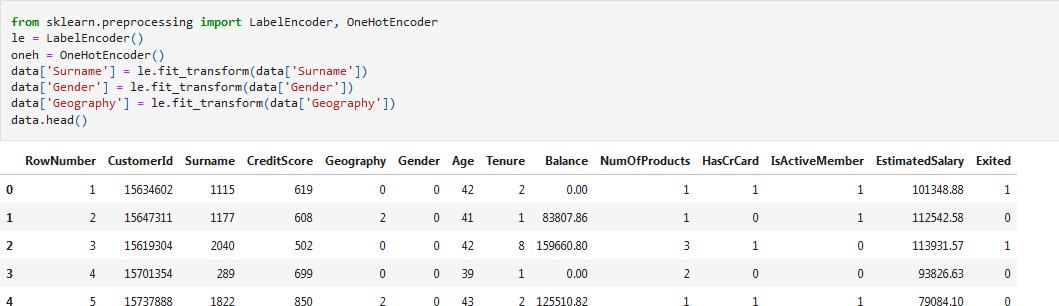
# Question:7

**Check for Categorical columns and perform encoding Solution:**

from sklearn.preprocessing import LabelEncoder, OneHotEncoder le = LabelEncoder()

oneh = OneHotEncoder()

data['Surname'] = le.fit\_transform(data['Surname']) data['Gender'] = le.fit\_transform(data['Gender']) data['Geography'] = le.fit\_transform(data['Geography']) data.head()

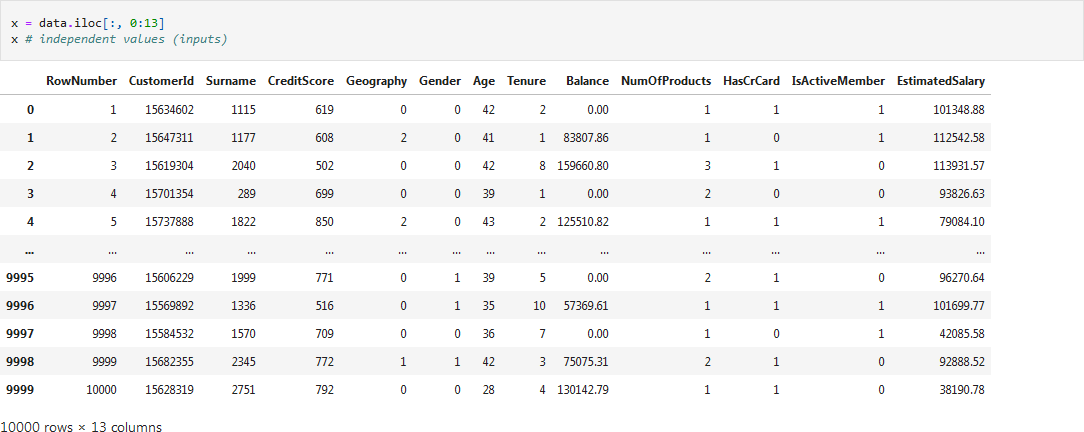


# Question.8

**Split the data into dependent and independent variables split the data in X and Y Solution:**

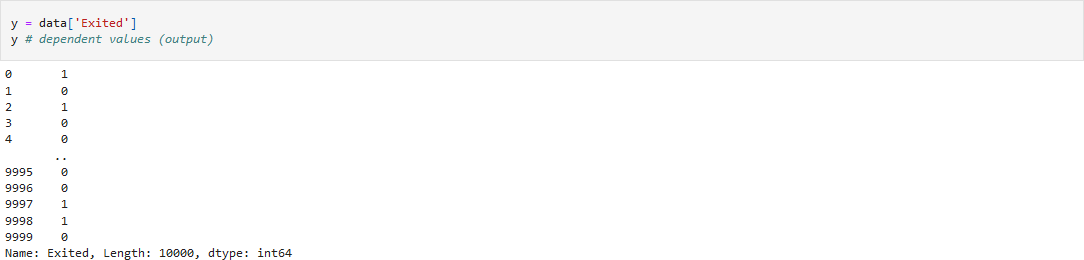
x = data.iloc[:, 0:13]

x # independent values (inputs)



y = data['Exited']

y # dependent values (output)

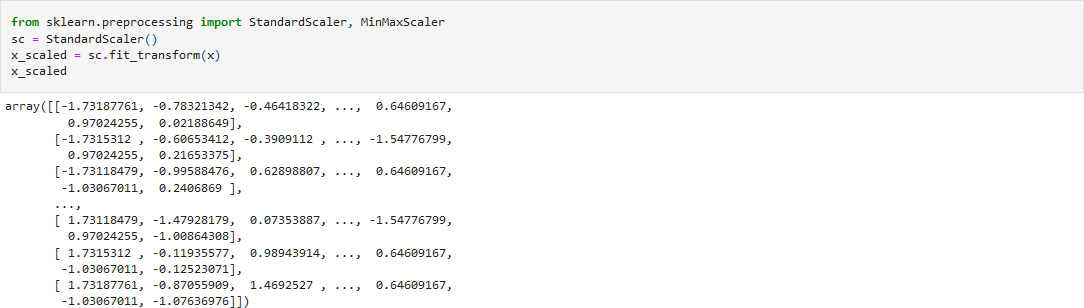


# Question:9

**Scale the independent variables Solution:**

from sklearn.preprocessing import StandardScaler, MinMaxScaler sc = StandardScaler()

x\_scaled = sc.fit\_transform(x) x\_scaled

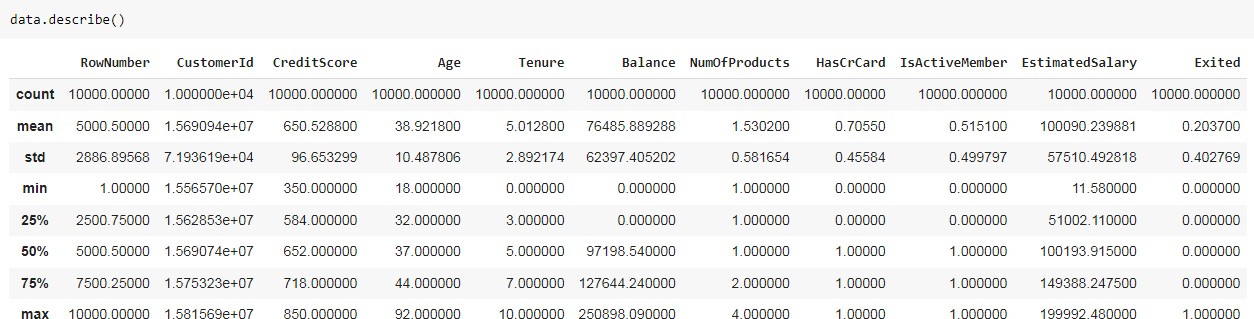


# Question:10

Split x and y into Training and Testing

**Solution:**

import pandas as pd data=pd.read\_csv("/content/Churn\_Modelling.csv")



import numpy as np x=np.array(data["CustomerId"]).reshape(-1,1) x.shape

