**Name : Dion D Rodrigues Assignment.no: 01**

**Roll.no : 56 Class : FYMCA B Subject : ML Lab Batch : B3**

**------------------------------------------------------------------------------------------------------------------------------------------**

**# a) IMPORTING LIBRARIES**

***import pandas as pd***

***import numpy as np***

***from sklearn.preprocessing import OneHotEncoder***

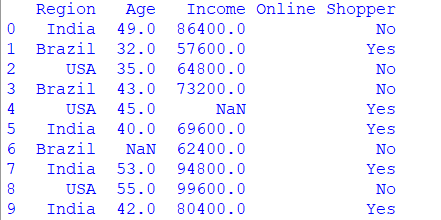
***from sklearn.model\_selection import train\_test\_split***

***from sklearn.impute import SimpleImputer***

#b) **IMPORTING LIBRARIES**

***data= pd.read\_csv('C:/Users/Admin/Desktop/ML/Assignment\_1/Online\_shopper.csv')***

***print(data)***

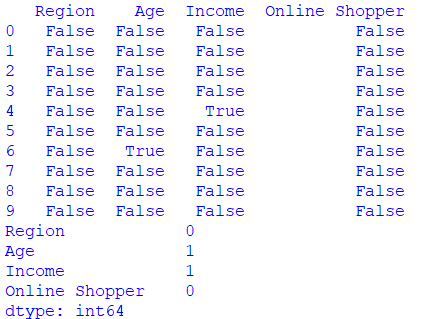


# c) **IDENTIFYING AND HANDLING MISSING DATA**

**#identifying**

***print (data.isnull())***

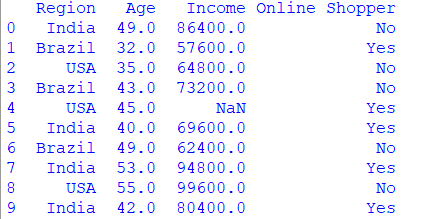
***print(data.isnull().sum())***



**#handling (FILL WITH THE MODE OF THE COLUMN)**

***data['Age']=data['Age'].fillna(data['Age'].mode())***

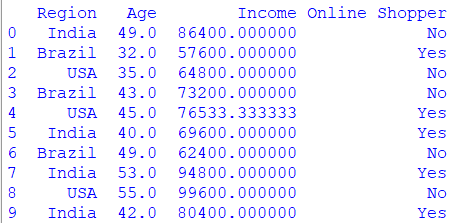
***print(data)***



**#handling (FILL WITH THE MEAN OF THE COLUMN)**

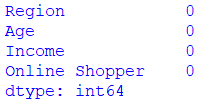
***data['Income']=data['Income'].fillna(data['Income'].mean())***

***print(data)***



**# Verify if missing data is handled**

***print(data.isnull().sum())***



**#d) IDENTIFYING AND HANDLING CATEGORICAL DATAZ**

**#identify categorial columns**

***categorical\_columns = data.select\_dtypes(include=[np.number]).columns***

***print(categorical\_columns)***



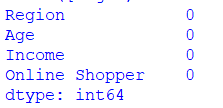
**#handling missing data by replacing with the mode (for categorical columns)**

***imputer = SimpleImputer (strategy='most\_frequent')***

***data[categorical\_columns]=imputer.fit\_transform(data[categorical\_columns])***

**#verify if missing data is handled**

***print(data.isnull().sum())***



**#Apply OneHotEnchoder to Categorical columns**

***encoder = OneHotEncoder(sparse\_output=False,drop='first')***

***encoded\_data = encoder.fit\_transform(data[categorical\_columns])***

**# Getting the names of encoded columns**

***encoded\_columns=encoder.get\_feature\_names\_out(categorical\_columns)***

**#Drop original categorical columns and concatenate encoded columns**

***data =data.drop(categorical\_columns,axis=1)***

***data=pd.concat([data,pd.DataFrame (encoded\_data,columns=encoded\_columns)],axis=1)***

**#Assuming 'Online shopping ' is the target variable, adjust if different**

***x=data.drop('Online Shopper',axis=1)***

***y=data['Online Shopper']***

**#e) SPLIT THE DATA INTO TRAINING AND TESTING SETS (e.g., 80% train,20% test)**

***x\_train, x\_test, y\_train, y\_test = train\_test\_split(x,y,test\_size=0.2,random\_state=42)***

***print("Traning set Shape : ",x\_train.shape,y\_train.shape)***

***print("Testing set Shape : ",x\_test.shape, y\_test.shape)***

