**Use the Automobiles Dataset.**

1. From given data set print first and last five rows

**CODE:**

import pandas as pd

df = pd.read\_csv("Automobile\_data.csv")

df.head(5)

import pandas as pd

df = pd.read\_csv("Automobile\_data.csv")

df.tail(5)

1. Print All Toyota Cars details (hint: use groupby function)

CODE:

import pandas as pd

df = pd.read\_csv("Automobile\_data.csv")

car\_Manufacturers = df.groupby('company')

toyotaDf = car\_Manufacturers.get\_group('toyota')

toyotaDf

1. Count total cars per company (by using value\_counts function)

CODE:

import pandas as pd

df = pd.read\_csv("Automobile\_data.csv")

df['company'].value\_counts()

1. Find each company’s Higesht price car. (hint: Groupby function and then apply .max function)

CODE:

import pandas as pd

df = pd.read\_csv("Automobile\_data.csv")

car\_Manufacturers = df.groupby('company')

priceDf = car\_Manufacturers['company','price'].max()

priceDf

1. Find the average mileage of each car making company (hint: Groupby function and then apply .mean function)

CODE:

import pandas as pd

df = pd.read\_csv("Automobile\_data.csv")

car\_Manufacturers = df.groupby('company')

mileageDf = car\_Manufacturers['company','average-mileage'].mean()

mileageDf

1. Sort all cars by Price column (hint: using sort\_values function).

CODE:

import pandas as pd

carsDf = pd.read\_csv("Automobile\_data.csv")

carsDf = carsDf.sort\_values(by=['price', 'horsepower'], ascending=False)

carsDf.head(5)

1. Check for the null values for entire dataset.

CODE:

df.isnull().sum()

1. **Use Crime dataset.**
2. find the aggregations like all moments of business decisions for all columns,value counts.

**CODE:**

import pandas as pd

crime\_df = pd.read\_csv('crime\_data.csv', index\_col=0)

agg\_stats = crime\_df.describe()

value\_counts = crime\_df.apply(pd.Series.value\_counts)

print("Aggregations (Statistics):")

print(agg\_stats)

print("\nValue Counts:")

print(value\_counts)

1. do the plottings like plottings like histogram, boxplot, scatterplot, barplot, piechart,dot chart.

**CODE:**

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

# Load the dataset

crime\_df = pd.read\_csv('crime\_data.csv', index\_col=0)

# Histogram for each column

crime\_df.hist(figsize=(12, 10))

plt.suptitle('Histograms of Crime Dataset')

plt.show()

# Boxplot for each column

plt.figure(figsize=(12, 8))

sns.boxplot(data=crime\_df)

plt.title('Boxplot of Crime Dataset')

plt.show()

# Scatterplot between Murder and Assault

plt.figure(figsize=(8, 6))

sns.scatterplot(data=crime\_df, x='Murder', y='Assault')

plt.title('Scatterplot of Murder vs. Assault')

plt.show()

# Barplot of Murder rates for each state

plt.figure(figsize=(12, 8))

sns.barplot(x=crime\_df.index, y='Murder', data=crime\_df)

plt.title('Murder Rates by State')

plt.xticks(rotation=90)

plt.show()

# Pie chart of UrbanPop distribution

plt.figure(figsize=(8, 8))

plt.pie(crime\_df['UrbanPop'], labels=crime\_df.index, autopct='%1.1f%%')

plt.title('UrbanPop Distribution')

plt.show()

# Dot chart of Rape rates for each state

plt.figure(figsize=(12, 8))

plt.plot(crime\_df['Rape'], crime\_df.index, 'bo', markersize=8)

plt.title('Rape Rates by State')

plt.xlabel('Rape Rate')

plt.ylabel('State')

plt.grid(True)

plt.show()

1. **use mtcars dataset.**
2. delete/ drop rows-10 to 15 of all columns

**CODE:**

import pandas as pd

mtcars\_df = pd.read\_csv('mtcars.csv')

mtcars\_df.drop(mtcars\_df.index[10:16], inplace=True)

mtcars\_df

B)drop the VOL column

**CODE:**

# Drop the VOL column

mtcars\_df.drop('vs', axis=1, inplace=True)

mtcars\_df

1. **Use Bank Dataset.**
2. change all the categorical columns into numerical by creating Dummies and using label encoder.  
   **CODE:**

**BY CREATING Dummies :**

import pandas as pd

import numpy as np

import seaborn as sns

df = pd.read\_csv('bank-full.csv')

new\_data =df

data = pd.get\_dummies(new\_dataM1["loan"])

new\_data.drop('loan', axis ='columns')

new\_data["no"] =data["no"].to\_list()

new\_data["yes"] = data["yes"].to\_list()

new\_data

**OR**import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

df = pd.read\_csv('bank-full.csv')

#import warnings

#warnings.filterwarnings('ignore')

#df\_cat['job']=pd.get\_dumies(df\_cat['job'],drop\_first=True)

# convert all categorical to numeric

cols = list(df\_cat.columns)

def cat\_to\_num(cols):

for col in cols:

df\_cat[col]=pd.get\_dumies(df\_cat[col],drop\_first=True)

cat\_to\_num(cols)

df\_cat.dtypes

**By using label encoder:**

from sklearn.preprocessing import LabelEncoder

df = pd.read\_csv('bank-full.csv')

le = LabelEncoder()

label = le.fit\_transform(df["loan"])

#le.classes\_

data = df.drop("loan",axis='columns')

#data.head()

data["loan"] = label

data

1. rename all the column names in DF

**CODE:**

df=df.rename(columns={'age':'Ages', 'job':'Jobs', 'marital':'Marital', 'education':'Edu', 'default':'DEFAULT', 'balance':'Balance', 'housing':'Housing',

'loan':'Loans', 'contact':'Contact', 'day':'Days', 'month':'Months', 'duration':'Durations', 'campaign':'Campaign', 'pdays':'P\_DAYS',

'previous':'Previous', 'poutcome':'Poutcome', 'y':'Y'})

df.columns

**OR**

df.rename(columns=str.upper)

1. Rename only one specific column in DF

**CODE:**

df=df.rename(columns={'age':'Agesssss'})

df.columns

1. After doing all the changes in bank dataset. save the file in your directory in Csv Format.(hint: by using .to\_csv)

**CODE:**

df.to\_csv('bank.csv',index=False)

df