PREDICTING HOUSE PRICES WITH MACHINE LEARNING

import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

import re

from sklearn.metrics import r2\_score, mean\_absolute\_error, mean\_squared\_error

from sklearn.datasets import load\_boston

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

from sklearn.linear\_model import LinearRegression

boston = load\_boston()

df = pd.DataFrame(boston.data)

df.head()

df.columns = boston.feature\_names

df.head()

df['PRICE']= boston.target

df.head()

df.tail()

df.shape

df.columns

df.dtypes

df.nunique()

df.isnull()

df.isnull()

df.describe()

df.corr

plt.figure(figsize=(10,10))

sns.heatmap(data=df.corr(), annot=True, cmap='Greens')

sns.pairplot(df, size=5)

plt.figure(figsize=(50,50))

df.boxplot()

df.PRICE.min()

df.PRICE.std()

df.to\_csv('boston\_datset.csv',)

df.head()

X = np.array(df.drop('PRICE' , axis=1))

y = np.array(df.PRICE)

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

len(X\_train)

len(y\_train)

len(X\_test)

len(y\_test)

model = LinearRegression()

model.fit(X\_train,y\_train)

model.intercept\_

model.coef\_

y\_test

y\_pred = model.predict(X\_test)

y\_pred

model.score(X\_test,y\_test)

r2\_score(y\_test,y\_pred)

mean\_squared\_error(y\_test,y\_pred)