LAB: AI MID

Name: Sabih-Ul-Hassan  
Reg.No: 2112126

Q3)

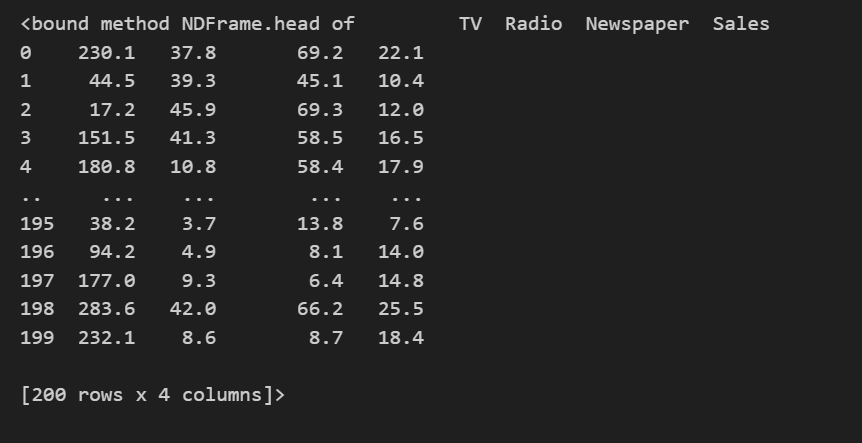
import pandas as pd

from matplotlib import pyplot as plt

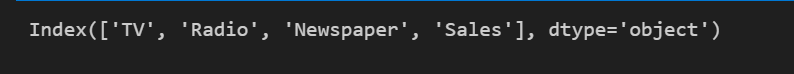
from sklearn import linear\_model

df = pd.read\_csv('advertising.csv')

df.head



df.columns



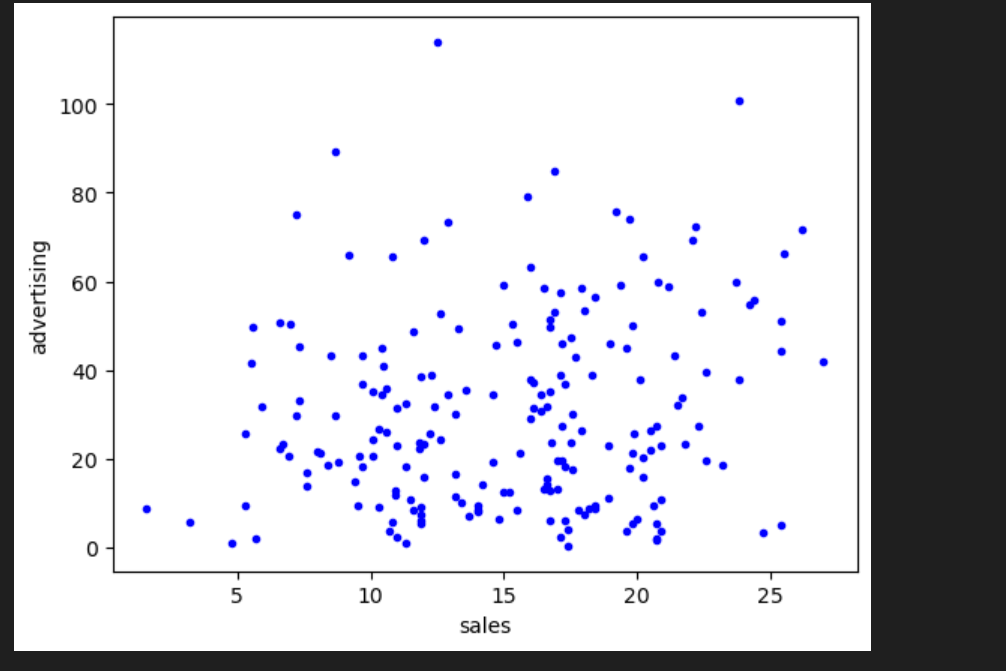
from matplotlib import pyplot as plt

plt.xlabel('sales')

plt.ylabel('advertising')

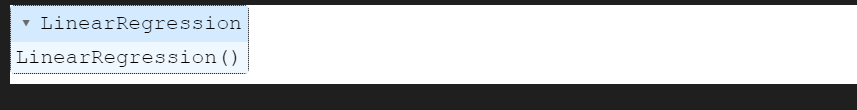
plt.scatter(df[['Sales']],df.Newspaper,color='blue',marker='.')

plt.show()



linear\_Reg\_Obj = linear\_model.LinearRegression()

linear\_Reg\_Obj.fit(df[['TV']],df.Sales)



linear\_Reg\_Obj.intercept\_

180616.43835616

Q1)

#REG. NO: 2112126

from collections import defaultdict

def generateAjacencyList(edges):

    adjacencyList = defaultdict(list)

    for u, v in adjacencyList:

        adjacencyList[u].append(v)

        adjacencyList[v].append(u)

    return adjacencyList

    edges[

        ['6','12'],['6','18'],['12','24'],['12','30'],['18','36'],['18','42'],['24','48'],['24','54'],['30','60'],['30','66'],['36','72']

    ]

    adjacencyList = generateAjacencyList(edges)

    adjacencyList

def dfs(adjacencyList,vertex, visited= None,path=None):

    if visited is None:

        visited = set()

        if path is None:

            path = []

        visited.add(vertex)

        path.append(vertex)

    if vertex is adjacencyList:

        for neighbor in adjacencyList[vertex]:

            if neighbor not in visited:

                dfs(adjacencyList,neighbor,visited,path)

            return path

        dfs(adjacencyList,'6')