Name : Devkumar Biswas

Class: BE(AI&DS)

Div: B

Subject : DMV(CL-I Lab)

Roll no. : BEAD21267

**Assignment No. – 5**

**Problem Statement** : Data Aggregation

Problem Statement: Analyzing Sales Performance by Region in a Retail Company Dataset: "Retail\_Sales\_Data.csv" Description: The dataset contains information about sales transactions in a retail company. It includes attributes such as transaction date, product category, quantity sold, and sales amount. The goal is to perform data aggregation to analyze the sales performance by region and identify the top-performing regions. Tasks to Perform: 1. Import the "Retail\_Sales\_Data.csv" dataset. 2. Explore the dataset to understand its structure and content. 3. Identify the relevant variables for aggregating sales data, such as region, sales amount, and product category. 4. Group the sales data by region and calculate the total sales amount for each region. 5. Create bar plots or pie charts to visualize the sales distribution by region. 6. Identify the top-performing regions based on the highest sales amount. 7. Group the sales data by region and product category to calculate the total sales amount for each combination. 8. Create stacked bar plots or grouped bar plots to compare the sales amounts across different regions and product categories.

CODE:-

import pandas as pd

import tensorflow as tf

import numpy as np

#Creating dataframe

df = pd.read\_csv('sales\_data\_sample 2.csv', encoding = "latin")

dfcsv = pd.DataFrame(df)

#Printing first 5 rows of dataset

dfcsv.head(5)

dfcsv.shape

dfcsv.isna().sum()

dfcsv.describe

dfcsv = dfcsv.drop(['ADDRESSLINE1','ADDRESSLINE2','CITY','STATE','TERRITORY'],axis = 1)

dfcsv.isna().sum()

dfcsv = dfcsv['POSTALCODE'].fillna(dfcsv.POSTALCODE.mode(), inplace=True)

columns\_to\_drop = ['ADDRESSLINE1', 'ADDRESSLINE2', 'CITY', 'STATE', 'TERRITORY']

import plotly.express as px

fig = px.bar(df, x='YEAR\_ID', y='SALES', title='Total Sales by Year')

fig.show()

fig = px.bar(df, x='QTR\_ID', y='SALES', title='Total Sales by Quarter')

fig.show()

fig = px.line(df, x='ORDERDATE', y='SALES', title='Sales Over Time')

fig.show()

OUTPUT:-

      