**Data Analytics**

**Project On**

***Sales Performance Dashboard***

**By, Abhishek Tahasildar**

**Requirements: Power BI Libraries: Pandas**

**Jupyter Notebook NumPy**

**SQL Matplotlib**

**Python Seaborn**

**MS-Excel**

**Objective:**

The goal of this project is to perform data analysis on sales data and create a **Sales Performance Dashboard**. The key objectives are:

* Analyze sales trends over time.
* Identify the best-performing product categories and regions.
* Calculate key performance indicators (KPIs) like total revenue, average sales volume, and sales by region.
* Visualize the data using **Python**, **SQL**, and **Power BI** for decision-making insights.

**Required Tools:**

1. **Python**:
   * **Libraries**: Pandas, NumPy, Matplotlib, Seaborn
   * Purpose: Data cleaning, exploration, manipulation, and visualization.
2. **SQL**:
   * Purpose: Querying the dataset using SQL (SQLite, MySQL, or PostgreSQL recommended).
3. **Power BI**:
   * Purpose: Visualizing the data in an interactive dashboard format, highlighting key insights.

**Dataset:**

The dataset contains sales data for various products. Here are the columns in the dataset:

* **Date**: Date of the sale.
* **Product Category**: The category of the product sold (e.g., Electronics, Furniture, etc.).
* **Sales Volume**: The number of units sold.
* **Revenue**: The total revenue from the sale.
* **Region**: Sales region (e.g., North, South, etc.).
* **Salesperson**: The name of the salesperson who made the sale.

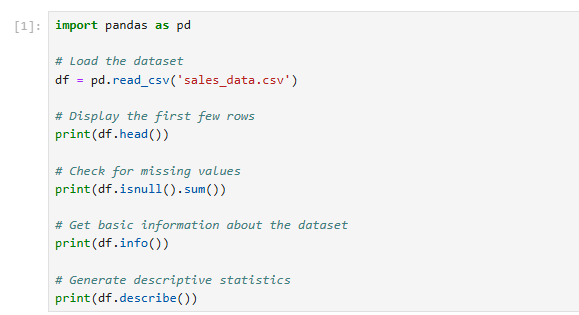
You can download the sample dataset here: [Sales\_Data.CSV](https://github.com/182001Abhi/Sales_Performance_Dashboard.git)

**Step 1: Load the Dataset in Python**

In this step, we'll use Python to load the dataset and perform an initial exploration.

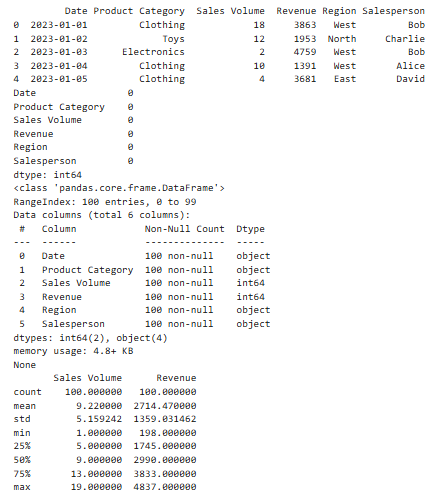
1. **Load the dataset** using pandas.
2. **Check the first few rows** to understand its structure.
3. **Check for any missing values or inconsistencies.**

Here’s the Python code for loading and exploring the dataset:



Output:

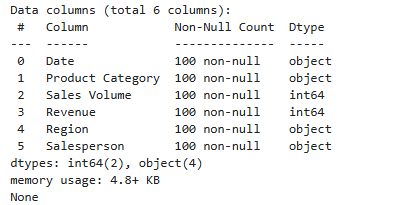
1.Display the first few rows:



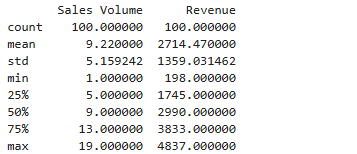
2.Check for missing values:



3. Get basic information about the dataset:



4. Generate descriptive statistics:



**Step 2: Data Cleaning**

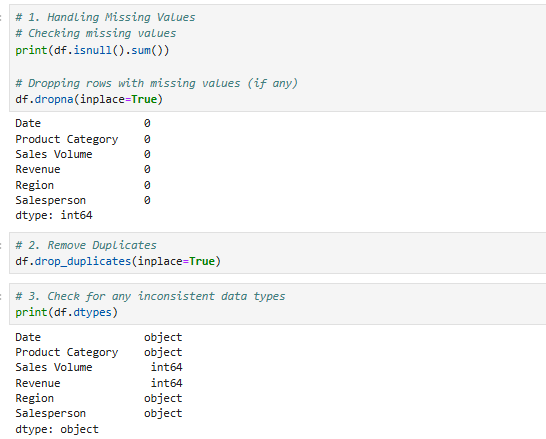
In this step, we will:

1. Handle any missing values.
2. Remove duplicates (if any).
3. Correct any inconsistencies in the data, such as incorrect data types or outliers.

**Actions to Perform:**

1. **Handling Missing Values:**
   * Check if any columns have missing values and decide how to handle them.
     + You can either drop rows with missing values or fill them with appropriate values (like the mean, median, or a specific value).
2. **Remove Duplicates:**
   * Ensure there are no duplicate rows in the dataset.
3. **Check for Inconsistent Data Types:**
   * Ensure that each column has the correct data type (e.g., dates should be in datetime format, numeric columns should be of type int or float).

**Python Code for Data Cleaning and Output:**



**Step 3: Exploratory Data Analysis (EDA)**

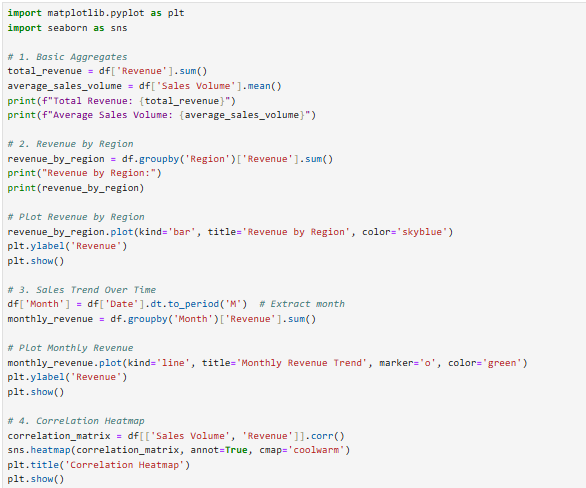
In this step, we will analyze the data to uncover insights. This involves:

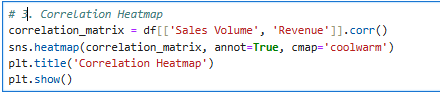
1. Understanding overall trends.
2. Visualizing key metrics.
3. Exploring relationships between variables.

**Actions to Perform:**

1. **Basic Aggregates:**
   * Calculate the total revenue, average sales volume, and other key metrics.
2. **Trend Analysis:**
   * Analyze sales trends over time (e.g., monthly revenue).
3. **Visualize the Data:**
   * Use plots to visualize trends and relationships.
   * Example: Bar chart for revenue by region, line chart for sales over time.

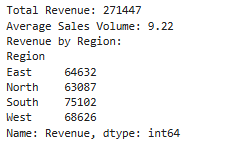
**Python Code for EDA:**

****

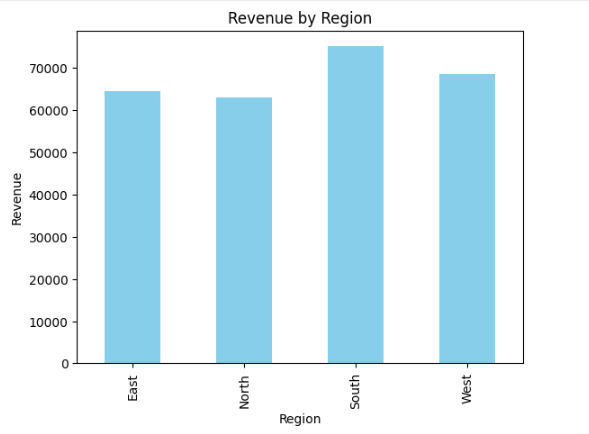
****

**Output:**

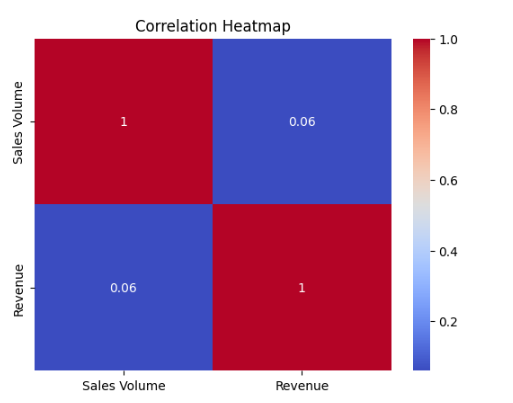
**1.Basic Aggregates:**

****

**2.Revenue By Region:**

****

**3.Correlation Heatmap:**

****

**Step 4: Integrating SQL for Data Analysis**

**In this step, we will:**

1. **Save the cleaned dataset to an SQL database.**
2. **Write SQL queries to analyze the data further.**

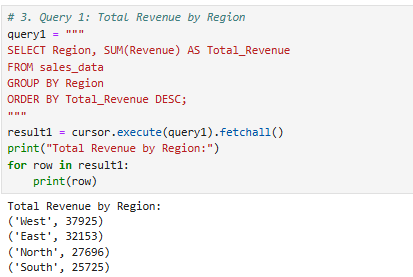
**Actions to Perform:**

1. **Set Up SQLite:**
   * **We will use SQLite, a lightweight SQL database.**
   * **Save the dataset into a database table.**
2. **Perform SQL Queries:**
   * **Query the database for insights, such as:**
     + **Total revenue by region.**
     + **Monthly sales trends.**
     + **Top-performing product categories.**

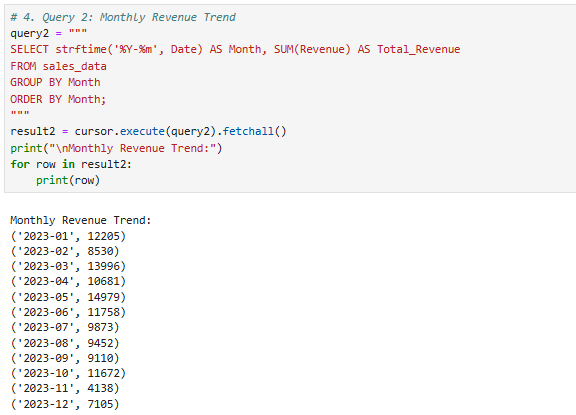
**1.Creating Connection to SQL DB and Save the Data frame to SQL Table:**



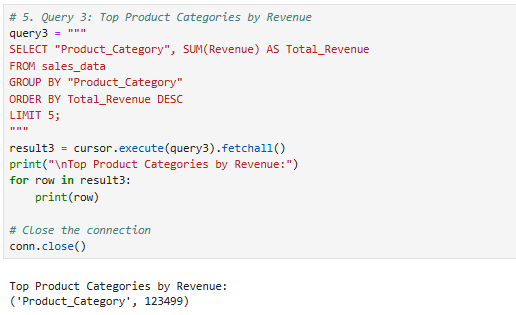
**2. SQL Query for Total Revenue By Region:**



**3. SQL Query For Monthly Revenue Trend**



**4. SQL Query for Top Product Categories by Revenue:**



**Step 5: Creating a Power BI Dashboard**

Now that we have performed data cleaning, exploratory analysis, and SQL integration, we can move on to creating a **Power BI Dashboard** to visualize and interact with the data.

**Actions to Perform:**

1. **Import Data into Power BI**:
   * Import the dataset or connect Power BI directly to the SQLite database.
2. **Build Visuals**:
   * Create key visualizations such as:
     + Bar charts for revenue by region.
     + Line charts for monthly revenue trends.
     + Tables for the top product categories.
3. **Design the Dashboard**:
   * Combine these visualizations into a cohesive dashboard layout.

**Steps to Create the Power BI Dashboard:**

**Step 1: Import Data into Power BI**

1. Open **Power BI Desktop**.
2. **Import the Dataset**:
   * If you're using the CSV dataset:
     + Go to Home > Get Data > Text/CSV and select the sales\_data.csv file.
   * If you're connecting to the SQLite database:
     + Go to Home > Get Data > More...
     + Select SQLite Database > Connect.
     + Enter the database file path (e.g., sales\_data.db).

**Step 2: Create Visualizations**

1. **Revenue by Region (Bar Chart)**:
   * Drag the Region field to the Axis area.
   * Drag the Revenue field to the Values area.
   * Set the chart type to Bar Chart.
   * Add any formatting (color, title, etc.).
2. **Monthly Revenue Trend (Line Chart)**:
   * Drag the Month field to the Axis area.
   * Drag the Revenue field to the Values area.
   * Set the chart type to Line Chart.
   * Format the axis labels for clarity.
3. **Top Product Categories by Revenue (Table)**:
   * Drag the Product\_Category field to the Values area.
   * Drag the Revenue field to the Values area.
   * Apply sorting by Revenue in descending order.
   * Add a filter to show only the top 5 categories.

**Step 3: Design the Dashboard**

* Once the visuals are created, arrange them in a dashboard layout.
  + Resize and align the charts.
  + Add any relevant filters or slicers (e.g., to filter by region or product category).
  + Customize the color scheme and labels for better presentation.

**Step 4: Publish and Share (Optional)**

* If you want to share the dashboard:
  + Go to Home > Publish and select your Power BI workspace.

[Click Here](https://github.com/182001Abhi/Sales_Performance_Dashboard.git) to Access Sales Performance Dashboard