**Project: Sales Data Analysis**

**Dataset Overview:**

The dataset contains information about sales orders, including order details, customer information, and product information. There's also a "Returns" dataset indicating which orders were returned.

**Project Questions:**

1. **Exploratory Data Analysis (EDA):**
   * What is the total number of orders in the dataset?
   * How many unique customers are there?
   * What are the top 5 product categories by sales?
   * Which segment has the highest average sales?
2. **Time Analysis:**
   * How has the number of orders evolved over the years?
   * What is the busiest month in terms of orders placed?
3. **Geographic Analysis:**
   * Which state has the highest total sales?
   * What are the top 5 cities by total sales?
4. **Returns Analysis:**
   * How many orders were returned?
   * What is the return rate (percentage of orders returned)?
   * Are there any specific product categories more prone to returns?

**Solutions:**

1. **Exploratory Data Analysis (EDA):**

* To find the total number of orders, you can run a SQL query like:
  + SELECT COUNT(\*) as TotalOrders FROM Orders;
* For the number of unique customers:
  + SELECT COUNT(DISTINCT CustomerID) as UniqueCustomers FROM Orders;
* To identify the top 5 product categories by sales:
* sqlCopy code
  + SELECT Category, SUM(Sales) as TotalSales FROM Orders GROUP BY Category ORDER BY TotalSales DESC LIMIT 5;
* To determine the segment with the highest average sales:
  + SELECT Segment, AVG(Sales) as AvgSales FROM Orders GROUP BY Segment ORDER BY AvgSales DESC LIMIT 1;

1. **Time Analysis:**
   * To analyze how the number of orders evolved over the years, you can create a time series chart or use SQL to group orders by year and count them.

* To find the busiest month:
  + SELECT strftime('%m', OrderDate) as Month, COUNT(\*) as OrderCount FROM Orders GROUP BY Month ORDER BY OrderCount DESC LIMIT 1;

1. **Geographic Analysis:**

* To find the state with the highest total sales:
  + SELECT State, SUM(Sales) as TotalSales FROM Orders GROUP BY State ORDER BY TotalSales DESC LIMIT 1;
  + For the top 5 cities by total sales, modify the query accordingly.

1. **Returns Analysis:**

* To count the number of returned orders:
  + SELECT COUNT(\*) as ReturnedOrders FROM Returns;
* To calculate the return rate:
  + SELECT (COUNT(\*) \* 1.0 / (SELECT COUNT(\*) FROM Orders)) \* 100 as ReturnRate FROM Returns;
  + To identify product categories prone to returns, you can compare return rates for different categories.

**complex**

**Project: Advanced Sales Data Analysis**

**Dataset Overview:**

The dataset contains detailed information about sales orders, customer information, and product details. Additionally, there's a dataset indicating which orders were returned.

**Step 1: Data Preparation**

1. **Data Import**: Load the provided dataset into your SQL database.
2. sqlCopy code
3. -- Assuming you have the dataset in a CSV file COPY Orders FROM 'path\_to\_csv\_file' DELIMITER ',' CSV HEADER; COPY Returns FROM 'path\_to\_csv\_file' DELIMITER ',' CSV HEADER;
4. Replace 'path\_to\_csv\_file' with the actual path to your CSV files.
5. **Data Cleansing**: Perform data cleansing tasks such as handling missing values, converting data types, and removing duplicates.
6. sqlCopy code
7. -- Handle missing values (for example, replacing NULL values with appropriate defaults) UPDATE Orders SET ShipMode = 'Standard Class' WHERE ShipMode IS NULL; -- Convert data types (e.g., date conversion) ALTER TABLE Orders ALTER COLUMN OrderDate TYPE DATE USING to\_date(OrderDate, 'MM/DD/YYYY');
8. **Add Constraints**: Define appropriate constraints for data integrity.
9. sqlCopy code
10. -- Add primary keys ALTER TABLE Orders ADD PRIMARY KEY (OrderID); ALTER TABLE Returns ADD PRIMARY KEY (OrderID); -- Add foreign keys ALTER TABLE Returns ADD FOREIGN KEY (OrderID) REFERENCES Orders (OrderID);

**Step 2: Exploratory Data Analysis (EDA)**

1. **Top Products by Sales**: Identify the top 10 products by total sales.
2. sqlCopy code
3. SELECT "Product Name", SUM(Sales) as TotalSales FROM Orders GROUP BY "Product Name" ORDER BY TotalSales DESC LIMIT 10;
4. **Customer Analysis**: Find the top 5 customers with the highest order count.
5. sqlCopy code
6. SELECT "Customer Name", COUNT(\*) as OrderCount FROM Orders GROUP BY "Customer Name" ORDER BY OrderCount DESC LIMIT 5;

**Step 3: Time Analysis**

1. **Monthly Sales Trend**: Visualize the monthly sales trend over the years.
2. sqlCopy code
3. SELECT TO\_CHAR(OrderDate, 'YYYY-MM') as Month, SUM(Sales) as TotalSales FROM Orders GROUP BY Month ORDER BY Month;

**Step 4: Geographic Analysis**

1. **Sales by Region**: Calculate the total sales for each region.
2. sqlCopy code
3. SELECT Region, SUM(Sales) as TotalSales FROM Orders GROUP BY Region ORDER BY TotalSales DESC;
4. **Top Cities by Sales**: Identify the top 3 cities with the highest sales in each region.
5. sqlCopy code
6. SELECT Region, "City", SUM(Sales) as TotalSales FROM Orders GROUP BY Region, "City" ORDER BY Region, TotalSales DESC LIMIT 3;

**Step 5: Returns Analysis**

1. **Return Rate by Product Category**: Calculate the return rate for each product category.
2. sqlCopy code
3. SELECT o."Category", (COUNT(r.OrderID) \* 100.0 / COUNT(o.OrderID)) as ReturnRate FROM Orders o LEFT JOIN Returns r ON o.OrderID = r.OrderID GROUP BY o."Category" ORDER BY ReturnRate DESC;
4. **Orders with Highest Returns**: Identify the orders with the highest returns.
5. sqlCopy code
6. SELECT o.OrderID, o."Customer Name", SUM(r.Sales) as TotalReturns FROM Orders o JOIN Returns r ON o.OrderID = r.OrderID GROUP BY o.OrderID, o."Customer Name" ORDER BY TotalReturns DESC;

**Tableau:**

**Step 1: Connect to Data**

1. Open Tableau and connect to your dataset (CSV file) by clicking on "Connect to Data."
2. Choose the CSV file with your sales data and follow the import wizard to load the data into Tableau.

**Step 2: Data Preparation**

1. Cleanse your data if necessary, handling missing values, and ensuring proper data types.

**Step 3: Create Visualizations**

1. **Total Sales by Product Category:**
   * Drag "Category" to Rows and "Sales" to Columns.
   * Change the mark type to Bar.
   * You can add filters or use parameters for interactivity.
2. **Top 5 Customers by Sales:**
   * Drag "Customer Name" to Rows and "Sales" to Columns.
   * Sort the data by Sales in descending order to see the top customers.
3. **Monthly Sales Trend:**
   * Drag "Order Date" to Columns, right-click it, and choose "Continuous Month."
   * Drag "Sales" to Rows.
   * Change the mark type to Line.
   * Use the "Show Me" feature to create a trend line.
4. **Sales by Region:**
   * Drag "Region" to Rows and "Sales" to Columns.
   * Use a map visualization for geographic data.

**Step 4: Create Dashboards**

1. Combine your visualizations into a dashboard by clicking "New Dashboard."
2. Add filters, quick actions, and interactivity to make the dashboard dynamic.

**Step 5: Share and Publish**

1. Save your Tableau workbook.
2. Publish the workbook to Tableau Server or Tableau Online to share it with others.

**Power BI:**

**Step 1: Connect to Data**

1. Open Power BI Desktop and click on "Get Data."
2. Choose "Text/CSV" and select your CSV file with sales data.
3. Follow the import wizard to load the data into Power BI.

**Step 2: Data Preparation**

1. Cleanse and transform your data using Power Query if needed.

**Step 3: Create Visualizations**

1. **Total Sales by Product Category:**
   * Drag "Category" to Axis and "Sales" to Values in a table or matrix visual.
   * You can also use a bar chart.
2. **Top 5 Customers by Sales:**
   * Drag "Customer Name" to Axis and "Sales" to Values.
   * Sort by Sales in descending order.
3. **Monthly Sales Trend:**
   * Create a new measure for Total Sales: Total Sales = SUM('Sales'[Sales])
   * Use a line chart with "Order Date" on the Axis and "Total Sales" as Values.
4. **Sales by Region:**
   * Create a map visual by adding a map visualization and selecting "Region" as Location and "Sales" as Size.

**Step 4: Create a Report**

1. Build a report by adding visuals to the canvas.
2. Use slicers, filters, and drill-through options for interactivity.

**Step 5: Publish and Share**

1. Save your Power BI report.
2. Publish it to the Power BI Service to share with others.