# SALES VISION

About: Sales Vision aims to offer a clear and detailed understanding of sales patterns, customer behavior, and product performance based on past data.

Description:

* Loading given past data from CSV file into the project environment.
* Processing and cleaning the data.
* Storing of data in structured format in MySql Repository.
* Identifying patterns and trends and generating insights.

Installation:

* IDE used Jupyter Notebook.
* Download MySql from <https://www.mysql.com/> website and install.
* Install Pandas into environment: !pip install pandas.
* Install Numpy into environment: !pip install numpy.

Importing and cleaning data:

* Import pandas to create data frames out of CSV file data.
* Import numpy for data typecasting if necessary.
* Import mysql connector to establish connection with MySql to interact with.
* Using pandas import data into dataframe from CSV file.
* The tables include data belonging to customers, stores, sales, products, exchange rates.
* Cleaning data:

1. Check for null values in the data. If any, fill the values with front values.
2. Drop the rows if it does not change characteristics of the data.
3. Remove outliers by standardization method on the columns with numeric values like price of the product, no. of orders, quantity of product etc..,.

* Processing data:

1. If necessary, type casting the data using numpy like converting from int32, int64 to int to ensure compatibility with MySql.
2. Casting the data using numpy like converting from float32, float64 to float to ensure compatibility with MySql.
3. Changing date time format of date columns in data frame to date time format to ensure compatibility with MySql.
4. Removing special characters if any.

Saving data:

* Establish a connection with MySql using mysql connector.
* Create a database name “mde92”.
* Create a table name “customers” with columns: customer\_id, customer\_key, gender, name, city, state\_code, state, zip\_code, country, continent, birthday.
* Create a table name “exchange\_rates” with columns: Erid, ExchangeDate, currency, ExchangeRate.
* Create a table name “products” with columns: ProductKey, ProductName, Brand, Color, UnitCostUSD, UnitPriceUSD, Subcategorykey, Subcategory, CategoryKey, Category.
* Create a table name “sales” with columns: id, order\_number, line\_item, order\_date, delivery\_date, customer\_id, store\_id, product\_id, quantity, currency\_code.
* Create a table name “stores” with columns: store\_key, country, state, square\_meters, open\_date.
* Cleaned and Processed data is stored in the respective tables.

Identifying patterns and trends:

* Application used: Power BI
* Connect to MySql Database by giving server name and database name, username, password.
* Import the tables needed to create patterns.
* The tables are customers, products, stores, sales, exchange rates.
* From the visualization pane, select any type of chart or map or table which reflects the appropriate trend or pattern in the data.
* Ribbon chart is used to showcase the pattern of sales corresponding to product and category.
* Line chart is used to illustrate the trend of sales volume by country over time.
* Map is used to display the geographical distribution of stores across the globe.
* Add legends to easily identify trend about specific column.
* Ribbon Chart: A graph showing a number of data

  Description automatically generated with medium confidence
* Line chart: A graph of sales and sales

  Description automatically generated with medium confidence
* Map: A map of the world with blue dots

  Description automatically generated

Identifying metrics using sql queries:

* Write 10 sql queries for uncovering trends and providing actionable insights based on historical data.
* Import CSV file 10 times, each corresponding to one of the 10 queries, and showcase the extracted data across 10 different pages.
* In the Transform Data tab, update Advanced Editor with appropriate DAX queries to load data for each of the 10 sql queries.
* DAX query: A screenshot of a computer

  Description automatically generated
* After loading data, select table visualization pane to display the data.
* Add effects to table to make it attractive and easy to understand.
* Add slicers to filter the data as needed.
* Slicer: A screenshot of a product pricing information

  Description automatically generated
* Add textbox for the title each page appropriately.
* Finally, output is visually appealing and easy to understand.

Conclusion:

The "Sales Vision" project successfully integrates multiple data sources and analytical tools to provide a view of sales performance and trends. By using Power BI for data visualization and SQL for trend identification, this project enables deep insights into customer behavior, product performance, and geographical sales distribution.