#Create Database

create database DataSolution;

#Use Database

use DataSolution;

#Create Table

CREATE TABLE sales (

Row\_ID INT,

Order\_ID VARCHAR(100),

Order\_Date DATE,

Ship\_Date DATE,

Ship\_Mode VARCHAR(100),

Customer\_Segment VARCHAR(150),

City VARCHAR(100),

State VARCHAR(100),

Country VARCHAR(100),

Continent VARCHAR(150),

Product\_ID VARCHAR(150),

Category VARCHAR(100),

Sub\_Category VARCHAR(100),

Product\_Name VARCHAR(255),

Quantity INT,

Unit\_Price DECIMAL(10,2),

Discount\_Perc DECIMAL(5,2),

Unit\_Manufacturing\_Cost DECIMAL(10,2),

Unit\_Shipping\_Cost DECIMAL(10,2),

Order\_Priority VARCHAR(50)

);

#Show table

Select \* from sales;

#Load CSV file using INFilE method

LOAD DATA INFILE 'C:/ProgramData/MySQL/MySQL Server 8.0/Uploads/sales.csv'

INTO TABLE sales

FIELDS TERMINATED BY ','

ENCLOSED BY '"'

LINES TERMINATED BY '\n'

IGNORE 1 LINES

(@Row\_ID, @Order\_ID, @Order\_Date, @Ship\_Date, @Ship\_Mode, @Customer\_Segment, @City, @State, @Country, @Continent,

@Product\_ID, @Category, @Sub\_Category, @Product\_Name, @Quantity, @Unit\_Price, @Discount\_Perc, @Unit\_Manufacturing\_Cost,

@Unit\_Shipping\_Cost, @Order\_Priority)

SET

Row\_ID = @Row\_ID,

Order\_ID = @Order\_ID,

Order\_Date = STR\_TO\_DATE(@Order\_Date, '%m/%d/%Y'),

Ship\_Date = STR\_TO\_DATE(@Ship\_Date, '%m/%d/%Y'),

Ship\_Mode = @Ship\_Mode,

Customer\_Segment = @Customer\_Segment,

City = @City,

State = @State,

Country = @Country,

Continent = @Continent,

Product\_ID = @Product\_ID,

Category = @Category,

Sub\_Category = @Sub\_Category,

Product\_Name = @Product\_Name,

Quantity = @Quantity,

Unit\_Price = @Unit\_Price,

Discount\_Perc = @Discount\_Perc,

Unit\_Manufacturing\_Cost = @Unit\_Manufacturing\_Cost,

Unit\_Shipping\_Cost = @Unit\_Shipping\_Cost,

Order\_Priority = @Order\_Priority;

#Show sales table

Select \* from sales;

-- Dim Location Table

CREATE TABLE dim\_location (

location\_id INT AUTO\_INCREMENT PRIMARY KEY,

city VARCHAR(100),

state VARCHAR(100),

country VARCHAR(100),

continent VARCHAR(150),

UNIQUE(city, state, country, continent)

);

-- Dim Category Table

CREATE TABLE dim\_category (

category\_id INT AUTO\_INCREMENT PRIMARY KEY,

category VARCHAR(100),

UNIQUE(category)

);

-- Dim Sub-Category Table

CREATE TABLE dim\_sub\_category (

sub\_category\_id INT AUTO\_INCREMENT PRIMARY KEY,

sub\_category VARCHAR(100),

category\_id INT,

FOREIGN KEY (category\_id) REFERENCES dim\_category(category\_id),

UNIQUE(sub\_category, category\_id)

);

-- Dim Product Table

CREATE TABLE dim\_product (

product\_id VARCHAR(150) PRIMARY KEY,

product\_name VARCHAR(255),

sub\_category\_id INT,

unit\_price DECIMAL(10,2),

unit\_manufacturing\_cost DECIMAL(10,2),

FOREIGN KEY (sub\_category\_id) REFERENCES dim\_sub\_category(sub\_category\_id)

);

CREATE TABLE fact\_transaction (

transaction\_id INT AUTO\_INCREMENT PRIMARY KEY,

order\_id VARCHAR(100),

order\_date DATE,

ship\_date DATE,

ship\_mode VARCHAR(100),

order\_priority VARCHAR(50),

customer\_segment VARCHAR(150),

location\_id INT,

product\_id VARCHAR(150),

quantity INT,

discount\_perc DECIMAL(5,2),

unit\_shipping\_cost DECIMAL(10,2),

FOREIGN KEY (location\_id) REFERENCES dim\_location(location\_id),

FOREIGN KEY (product\_id) REFERENCES dim\_product(product\_id)

);

# Normalize and Populate in mySQL

DELIMITER //

CREATE PROCEDURE normalize\_and\_populate()

BEGIN

-- Insert locations

INSERT IGNORE INTO dim\_location (city, state, country, continent)

SELECT DISTINCT City, State, Country, Continent FROM sales;

-- Insert categories

INSERT IGNORE INTO dim\_category (category)

SELECT DISTINCT Category FROM sales;

-- Insert sub-categories

INSERT IGNORE INTO dim\_sub\_category (sub\_category, category\_id)

SELECT DISTINCT s.Sub\_Category, dc.category\_id

FROM sales s

JOIN dim\_category dc ON s.Category = dc.category;

-- Insert products

INSERT IGNORE INTO dim\_product (product\_id, product\_name, sub\_category\_id, unit\_price, unit\_manufacturing\_cost)

SELECT DISTINCT s.Product\_ID, s.Product\_Name, dsc.sub\_category\_id, s.Unit\_Price, s.Unit\_Manufacturing\_Cost

FROM sales s

JOIN dim\_category dc ON s.Category = dc.category

JOIN dim\_sub\_category dsc ON s.Sub\_Category = dsc.sub\_category AND dsc.category\_id = dc.category\_id;

-- Insert fact transactions

INSERT INTO fact\_transaction (

order\_id, order\_date, ship\_date, ship\_mode, order\_priority,

customer\_segment, location\_id, product\_id, quantity,

discount\_perc, unit\_shipping\_cost

)

SELECT

s.Order\_ID, s.Order\_Date, s.Ship\_Date, s.Ship\_Mode, s.Order\_Priority,

s.Customer\_Segment, dl.location\_id, s.Product\_ID, s.Quantity,

s.Discount\_Perc, s.Unit\_Shipping\_Cost

FROM sales s

JOIN dim\_location dl ON s.City = dl.city AND s.State = dl.state AND s.Country = dl.country AND s.Continent = dl.continent;

END //

DELIMITER ;

CALL normalize\_and\_populate();

select \* from dim\_sub\_category;

select \* from dim\_product;

select \* from dim\_location;

select \* from dim\_category;

select \* from fact\_transaction;

#Count the records

SELECT COUNT(\*) FROM sales;

#Checking null records

SELECT COUNT(\*) FROM sales WHERE Product\_ID IS NULL;

CREATE TABLE return\_table (

Order\_ID VARCHAR(100),

Returned VARCHAR(10)

);

#Create table for returned.csv

CREATE TABLE return\_table (

returned VARCHAR(10),

order\_id VARCHAR(100)

);

select \* from return\_table;