Team 10 – SneakerStore

Data Management SQL with Maithily Erande

Hult International Business School

Kaan Korkmaz, Johanna Wameling in Cohort 2

Table of Contents

[Overview 2](#_Toc66655858)

[Assumptions: 2](#_Toc66655859)

[Reports and Insights: 2](#_Toc66655860)

[Bonus: Ready-to-use Processes and procedures stored in the database 3](#_Toc66655861)

[Flowchart: Customer Order Journey and Inventory Management 4](#_Toc66655862)

[Key Entities 5](#_Toc66655863)

[Entity Relational Model 6](#_Toc66655864)

[SQL Queries 7](#_Toc66655865)

[Reports and Queries addressing pain points 7](#_Toc66655866)

[Stored Procedures: Bonus 12](#_Toc66655867)

[Appendix: Create Tables Statements 16](#_Toc66655868)

[Appendix 2: Insert Statements 21](#_Toc66655869)

[Works Cited 28](#_Toc66655870)

# Overview

The following document will outline the database management system for our shoe retailer/wholesaler. The system is designed to address key pain points, produce actionable reports and facilitate transactions. This business sells to retailers (B2B) as well as to consumers (B2C) directly. The system reflects purchase orders, products and inventory on the one hand, as well as sales, on the other. Like any retail business, you need to manage inventory, re-ordering of stock and adding new product releases to the offers. Furthermore, it is important for strategy and marketing to have an insight to the performance of individual products, segments and of the sales representatives selling them. Each process can be accompanied and tied into the database, from ordering to order picking and informing the customer of the estimated arrival date. In a data-driven business, the management is able to steer sales, based on historic data and performance. The database will cater to all of these needs and in the following, we will outline how it will do so, more specifically.

## Assumptions:

* When products are received in the warehouse, the delivered date is updated manually.
  + Based on this date, the inventory is updated the following day (by a stored process)
* Our buying price is fixed based on contracts with our suppliers.
* B2B customers have sales reps assigned to them.

## Reports and Insights:

The database has been designed to address all kinds of bottlenecks and regular processes in the supply chain. As a system user, the following procedures, reports and updates can be used immediately. Many other reports can be created on an as-needed basis, given the well thought out-table structure and model.

1. The inventory is the core of the platform. If the business runs out of stock, it cannot make any money. Therefore, several things need to be monitored.
   1. Monitoring Stock levels is a critical part of the business. This report is a frequently view, used to look at the items that are low in inventory. It forms the basis of several procedures: such as re-ordering items
2. As a business it is important to keep the employees motivated. As such, the sales department needs to track the performance of its sales representatives for the annual “Sales Rep of the Year”-trophy and bonus payment. Who are the top 3 Sales representatives by revenue they brought in?
3. The commercial team is interested in the revenue split between B2B and B2C customer segments. This insight could impact marketing and sales strategy, advertising spend and more.
4. Our customers are our most valuable asset. The marketing team can reward the best customers, especially if they are B2C customers by offering a discount code or other promotions. This report shows how much is each customer is worth sorted by lifetime revenue.
5. The management is interested in the performance of the company and products. This can aid in decision of discontinuing an item or product segment. Furthermore, the marketing strategy can focus on weaker performing products or adapt to demand for others. As such, the attached scripts can generate reports that answer the following business questions:
   1. Which products (models) are selling the most in terms of quantity of units sold?
   2. Which shoe segment (Boat Shoes, Running Shoes etc.) is the strongest segment in terms of revenue?
6. The stock pickers in the warehouse need a list of items to pick from the shelves each day. They will need all the information to prepare an order for shipment. The system generates a report based on the order status, payment status and provides the item and customer information. This way the stock pickers can commission the sales-order and get it ready for shipping.
7. Once the stock pickers have picked an order and it is ready to be shipped, the customer should be updated of the status of their order. To do this, the customer service department needs the finished sales-order picking list and customers’ email address. There is a procedure, just for this purpose, as well.

## Bonus: Ready-to-use Processes and procedures stored in the database

1. When a customer places an order, the system needs to create new entries in the orders, and order details tables, furthermore it needs to check the inventory and reduce inventory by the ordered amount to reflect this sale. A store procedure captures all these standard procedures and allows for fast and comprehensive order tracking.
2. To avoid running out of stock, the system creates a low inventory report which can trigger a new purchase order for low-stock products. This can be done manually or with an automated procedure that checks stock levels and generates purchase orders based on a threshold.
3. A shoe manufacturer (supplier) just launched a new shoe model. Your customers cannot wait to be able to order this item, but first, you need to create the product in your product database and order the shoe model from the supplier.
   1. There are stored procedures available to add new items to the products table and
   2. To order or re-order products.
4. When shipments are received from suppliers, this needs to be reflected in the inventory levels of our products. Once the receiving agent in our warehouse updates the received date, all the products have to be added into the inventory, to start selling them. A stored procedure does this and also updates the status to “In Stock” so customers can immediately start buying it again.

# Flowchart: Customer Order Journey and Inventory Management

The below flow-chart depicts the typical flow of information and data that is facilitated by the proposed database. The customer making an order

Diagram

Description automatically generated

# Key Entities

The database for your shoe shop has been designed with 5 main sections in mind. These (same colors) are also reflected in the ER model and described below. The main entities are the products, the purchase orders, the customers and customer orders. The supporting elements are the shippers, employees and payments records.



# Entity Relational Model

A picture containing diagram

Description automatically generated

# SQL Queries

## Reports and Queries addressing pain points

1. View to display lowest stock items. This view shows all items where there are less than 10 units in stock. Graphical user interface, application

   Description automatically generated
2. View to show the top performing sales rep, ordered by revenue they brought in with their B2B accounts

Graphical user interface, text, application

Description automatically generated

1. Revenue by Customer Segment (B2B or B2C) Graphical user interface, text, application, email

   Description automatically generated
2. Best B2C customers by revenue. Getting their emails to send out discount codes.

Graphical user interface, application

Description automatically generated

1. Best-selling product by quantityGraphical user interface, text, application

   Description automatically generated
2. Strongest product segment by revenue.

Graphical user interface, table

Description automatically generated with medium confidence

1. Updating the customer order details when items have shipped out of the warehouse.   
   Graphical user interface, text, application, email

   Description automatically generated
   1. Calling the procedure: Retrieving the emails for customers whos’ orders have been picked and shipped  
      Graphical user interface, text, application

      Description automatically generated
2. Order picking list for warehouse staff to pick and prepare orders  
   Table

   Description automatically generated

## Stored Procedures: Bonus

1. Creating a purchase order based on manual input of product ID and quantity needed  
   Graphical user interface, text, application, email

   Description automatically generated
   1. Calling the procedure and re-ordering 3 different products.  
      Table

      Description automatically generated
2. Automatically create purchase orders for low-stock itemsGraphical user interface, text, application

   Description automatically generated
   1. Calling the procedure: 3 low-stock Items re-ordered  
      Table

      Description automatically generated
3. Adding a new product to the product catalog table  
   Graphical user interface, text, application

   Description automatically generated
   1. Calling the procedure to add three new shoes to the product catalog

Graphical user interface, table

Description automatically generated

1. Creating a new customer order based on the customer ID, product ID and quantity ordered  
   Graphical user interface, text, application, email

   Description automatically generated  
   Graphical user interface, text, application, email

   Description automatically generated  
   Graphical user interface, text, application, email

   Description automatically generated

# Appendix: Create Tables Statements

-- MySQL Workbench Forward Engineering

SET @OLD\_UNIQUE\_CHECKS=@@UNIQUE\_CHECKS, UNIQUE\_CHECKS=0;

SET @OLD\_FOREIGN\_KEY\_CHECKS=@@FOREIGN\_KEY\_CHECKS, FOREIGN\_KEY\_CHECKS=0;

SET @OLD\_SQL\_MODE=@@SQL\_MODE, SQL\_MODE='ONLY\_FULL\_GROUP\_BY,STRICT\_TRANS\_TABLES,NO\_ZERO\_IN\_DATE,NO\_ZERO\_DATE,ERROR\_FOR\_DIVISION\_BY\_ZERO,NO\_ENGINE\_SUBSTITUTION';

-- -----------------------------------------------------

-- Schema shoe\_shop\_team10

-- -----------------------------------------------------

-- -----------------------------------------------------

-- Schema shoe\_shop\_team10

-- -----------------------------------------------------

CREATE SCHEMA IF NOT EXISTS `shoe\_shop\_team10` DEFAULT CHARACTER SET utf8 ;

USE `shoe\_shop\_team10` ;

-- -----------------------------------------------------

-- Table `shoe\_shop\_team10`.`Employees`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `shoe\_shop\_team10`.`Employees` (

`EmployeeID` INT UNSIGNED NOT NULL AUTO\_INCREMENT,

`FirstName` VARCHAR(45) NOT NULL,

`LastName` VARCHAR(45) NOT NULL,

`Title` VARCHAR(100) NOT NULL,

`hireDate` DATE NOT NULL,

`ReportsTo` INT(10) NULL,

`email` VARCHAR(255) NOT NULL,

PRIMARY KEY (`EmployeeID`),

UNIQUE INDEX `email\_UNIQUE` (`email` ASC) VISIBLE);

-- -----------------------------------------------------

-- Table `shoe\_shop\_team10`.`Customers`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `shoe\_shop\_team10`.`Customers` (

`CustomerID` INT UNSIGNED NOT NULL AUTO\_INCREMENT,

`FirstNameCustomer` VARCHAR(45) NOT NULL,

`LastNameCustomer` VARCHAR(45) NOT NULL,

`Gender` ENUM('M', 'F', 'O') NOT NULL,

`customerSince` DATE NOT NULL,

`Address` VARCHAR(45) NOT NULL,

`postalcode` CHAR(15) NOT NULL,

`city` VARCHAR(50) NOT NULL,

`state` VARCHAR(50) NULL,

`country` VARCHAR(50) NOT NULL,

`customerType` ENUM("B2B", "B2C") NOT NULL,

`SalesRepID` INT UNSIGNED NULL,

`emailCust` VARCHAR(255) NOT NULL,

PRIMARY KEY (`CustomerID`),

INDEX `fk\_customers\_Employees1\_idx` (`SalesRepID` ASC) VISIBLE,

UNIQUE INDEX `emailCust\_UNIQUE` (`emailCust` ASC) VISIBLE,

CONSTRAINT `fk\_customers\_Employees1`

FOREIGN KEY (`SalesRepID`)

REFERENCES `shoe\_shop\_team10`.`Employees` (`EmployeeID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION);

-- -----------------------------------------------------

-- Table `shoe\_shop\_team10`.`Orders`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `shoe\_shop\_team10`.`Orders` (

`OrderID` INT UNSIGNED NOT NULL AUTO\_INCREMENT,

`OrderDate` DATE NOT NULL,

`CustomerID` INT UNSIGNED NOT NULL,

`OrderValue` DECIMAL(10,2) NOT NULL,

`CompletedDate` DATETIME NULL,

`OrderStatus` ENUM("Pending", "Completed", "Cancelled") NULL,

PRIMARY KEY (`OrderID`),

INDEX `fk\_Orders\_customers1\_idx` (`CustomerID` ASC) VISIBLE,

UNIQUE INDEX `OrderID\_UNIQUE` (`OrderID` ASC) VISIBLE,

CONSTRAINT `fk\_Orders\_customers1`

FOREIGN KEY (`CustomerID`)

REFERENCES `shoe\_shop\_team10`.`Customers` (`CustomerID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `shoe\_shop\_team10`.`Payments`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `shoe\_shop\_team10`.`Payments` (

`PaymentID` INT NOT NULL AUTO\_INCREMENT,

`orderID` INT(20) UNSIGNED NOT NULL,

`amount` DECIMAL(10,2) NOT NULL,

`paymentdate` DATE NOT NULL,

`paymentType` ENUM('CC', 'Debit', 'Check', 'Bank Transfer') NOT NULL,

PRIMARY KEY (`PaymentID`),

INDEX `fk\_payments\_Orders1\_idx` (`orderID` ASC) VISIBLE,

CONSTRAINT `fk\_payments\_Orders1`

FOREIGN KEY (`orderID`)

REFERENCES `shoe\_shop\_team10`.`Orders` (`OrderID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `shoe\_shop\_team10`.`Suppliers`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `shoe\_shop\_team10`.`Suppliers` (

`SupplierID` INT UNSIGNED NOT NULL AUTO\_INCREMENT,

`SupplierName` VARCHAR(255) NOT NULL,

`ContactNameSupplier` VARCHAR(50) NOT NULL,

`Address` VARCHAR(45) NOT NULL,

`postalcode` CHAR(15) NOT NULL,

`city` VARCHAR(50) NOT NULL,

`state` VARCHAR(50) NULL,

`country` VARCHAR(50) NOT NULL,

`email` VARCHAR(255) NULL,

PRIMARY KEY (`SupplierID`),

UNIQUE INDEX `email\_UNIQUE` (`email` ASC) VISIBLE);

-- -----------------------------------------------------

-- Table `shoe\_shop\_team10`.`Products`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `shoe\_shop\_team10`.`Products` (

`ProductID` VARCHAR(20) NOT NULL,

`type` ENUM("Running Shoes", "Everyday Sneaker", "Slip-Ons", "Boat Shoes", "High Tops", "Weather Repellent Shoes") NOT NULL,

`ProductName` VARCHAR(100) NOT NULL,

`Brand` VARCHAR(45) NOT NULL,

`Model` VARCHAR(45) NOT NULL,

`Size` VARCHAR(10) NOT NULL,

`Color` VARCHAR(45) NOT NULL,

`Gender` ENUM("M", "F", "U") NOT NULL,

`DateAdded` DATE NOT NULL,

`buyPrice` DECIMAL(10,2) NOT NULL,

`Margin` DECIMAL(10,2) NOT NULL,

`sellPrice` DECIMAL(10,2) NOT NULL,

`StockLevel` INT NULL,

`Status` ENUM("Discontinued", "In Stock", "Sold Out", "On Sale") NULL,

`SupplierPrimary` INT UNSIGNED NULL,

PRIMARY KEY (`ProductID`),

INDEX `fk\_Products\_Suppliers1\_idx` (`SupplierPrimary` ASC) VISIBLE,

CONSTRAINT `fk\_Products\_Suppliers1`

FOREIGN KEY (`SupplierPrimary`)

REFERENCES `shoe\_shop\_team10`.`Suppliers` (`SupplierID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `shoe\_shop\_team10`.`Shippers`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `shoe\_shop\_team10`.`Shippers` (

`ShipperID` INT UNSIGNED NOT NULL AUTO\_INCREMENT,

`ShipperName` VARCHAR(45) NOT NULL,

`ContactName` VARCHAR(100) NULL,

`ContactEmail` VARCHAR(320) NOT NULL,

`Address` VARCHAR(50) NULL,

`City` VARCHAR(50) NULL,

`State` VARCHAR(50) NULL,

`Country` VARCHAR(50) NOT NULL,

`PostalCode` CHAR(15) NOT NULL,

PRIMARY KEY (`ShipperID`))

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `shoe\_shop\_team10`.`OrderDetails`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `shoe\_shop\_team10`.`OrderDetails` (

`OrderID` INT UNSIGNED NOT NULL,

`ProductID` VARCHAR(20) NOT NULL,

`quantity` INT UNSIGNED NULL,

`ETA\_Date` DATE NULL,

`Delivered Date` DATE NULL,

`ShipperID` INT UNSIGNED NULL,

`Status\_OD` ENUM("Pending", "Shipped", "Delivered", "Delayed", "Cancelled") NULL,

PRIMARY KEY (`OrderID`, `ProductID`),

INDEX `fk\_OrderDetails\_Products1\_idx` (`ProductID` ASC) VISIBLE,

INDEX `fk\_OrderDetails\_Shippers1\_idx` (`ShipperID` ASC) VISIBLE,

CONSTRAINT `fk\_OrderDetails\_Products1`

FOREIGN KEY (`ProductID`)

REFERENCES `shoe\_shop\_team10`.`Products` (`ProductID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `fk\_OrderDetails\_Orders1`

FOREIGN KEY (`OrderID`)

REFERENCES `shoe\_shop\_team10`.`Orders` (`OrderID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `fk\_OrderDetails\_Shippers1`

FOREIGN KEY (`ShipperID`)

REFERENCES `shoe\_shop\_team10`.`Shippers` (`ShipperID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `shoe\_shop\_team10`.`PurchaseOrders`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `shoe\_shop\_team10`.`PurchaseOrders` (

`PurchOrderID` INT UNSIGNED NOT NULL AUTO\_INCREMENT,

`OrderDate` DATE NOT NULL,

`OrderValue` DECIMAL(10,2) NOT NULL,

`CompletedDate` DATETIME NULL,

`PurchOrderStatus` ENUM("Pending", "Completed", "Cancelled") NULL,

`SupplierID` INT UNSIGNED NULL,

PRIMARY KEY (`PurchOrderID`),

INDEX `fk\_PurchaseOrders\_Suppliers1\_idx` (`SupplierID` ASC) VISIBLE,

CONSTRAINT `fk\_PurchaseOrders\_Suppliers1`

FOREIGN KEY (`SupplierID`)

REFERENCES `shoe\_shop\_team10`.`Suppliers` (`SupplierID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `shoe\_shop\_team10`.`PurchaseOrderDetails`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `shoe\_shop\_team10`.`PurchaseOrderDetails` (

`PurchOrderID` INT UNSIGNED NOT NULL,

`ProductID` VARCHAR(20) NOT NULL,

`quantity` INT UNSIGNED NULL,

`ETA\_Date` DATE NULL,

`Delivered Date` DATE NULL,

`ShipperID` INT UNSIGNED NULL,

`Status\_purchOD` ENUM("Confirmed", "Shipped", "Delivered", "Cancelled") NULL,

PRIMARY KEY (`PurchOrderID`, `ProductID`),

INDEX `fk\_PurchaseOrderDetails\_Products\_idx` (`ProductID` ASC) VISIBLE,

INDEX `fk\_PurchaseOrderDetails\_Shippers1\_idx` (`ShipperID` ASC) VISIBLE,

CONSTRAINT `fk\_PurchaseOrderDetails\_Products`

FOREIGN KEY (`ProductID`)

REFERENCES `shoe\_shop\_team10`.`Products` (`ProductID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `fk\_PurchaseOrderDetails\_PurchaseOrders1`

FOREIGN KEY (`PurchOrderID`)

REFERENCES `shoe\_shop\_team10`.`PurchaseOrders` (`PurchOrderID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `fk\_PurchaseOrderDetails\_Shippers1`

FOREIGN KEY (`ShipperID`)

REFERENCES `shoe\_shop\_team10`.`Shippers` (`ShipperID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

SET SQL\_MODE=@OLD\_SQL\_MODE;

SET FOREIGN\_KEY\_CHECKS=@OLD\_FOREIGN\_KEY\_CHECKS;

SET UNIQUE\_CHECKS=@OLD\_UNIQUE\_CHECKS;

# Appendix 2: Insert Statements

INSERT INTO `shoe\_shop\_team10`.`Employees`

(`EmployeeID`,

`FirstName`,

`LastName`,

`Title`,

`hireDate`,

`ReportsTo`,

`email`)

VALUES

(1004, "Jordanne ", "Partridge", "Sales Rep ", "2020-05-05", 1012, "Partridge@shoes.com"),

(1005, "Keeva", "Cobb", "Sales Rep ", "2020-05-06", 1012, "Cobb@shoes.com"),

(1006, "Imaad", "Harper", "Sales Rep ", "2020-05-07", 1012, "Harper@shoes.com"),

(1007, "Alessia", "Robbins", "Sales Rep ", "2020-05-08", 1012, "Robbins@shoes.com"),

(1008, "Madison", "Marshall", "Sales Rep ", "2020-05-09", 1012, "Marshall@shoes.com"),

(1009, "Marie", "Partridge", "Sales Rep ", "2020-05-10", 1012, "MPartridge@shoes.com"),

(1010, "Alishia", "Robson", "Sales Rep ", "2020-05-11", 1012, "Robson@shoes.com"),

(1011, "Lucien", "Cleveland", "Head of operations", "2020-05-12",NULL, "Cleveland@shoes.com"),

(1012, "Connah", "Bull", "Head of sales ", "2020-05-13",NULL , "Bull@shoes.com"),

(1013, "Opal", "Ahmed", "Warehouse manager", "2020-05-14", 1011, "Ahmed@shoes.com");

INSERT INTO `shoe\_shop\_team10`.`Suppliers`

(`SupplierID`,

`SupplierName`,

`ContactNameSupplier`,

`Address`,

`postalcode`,

`city`,

`state`,

`country`,

`email`)

VALUES

(55555, "textiles etc", "Jenny", "1 supplies unlimited lane", "33578", "cambridge", "MA", "USA", "jen@textiles.com"),

(88888, "manufacturer 2", "Jonathan", "450 po box road", "13448", "san francisco", "CA", "USA", "jon@manufacturerdeluxe.com"),

(99999, "shoemaker the best", "Jeremy", "555 shoe lane", "74539", "boston", "MA", "USA", "jerry@shoemaker.com"),

(77777, "delivery guru", "James", "760 shoe maker lane", "87838", "new york", "NY", "USA", "james@deliveryguru.com"),

(66666, "best shoes manufacturer", "Jane", "777 road b", "90124", "seattle", "WA", "USA", "jane@bestshoes.com");

INSERT INTO `shoe\_shop\_team10`.`Shippers`

(`ShipperID`,

`ShipperName`,

`ContactName`,

`ContactEmail`,

`Address`,

`City`,

`State`,

`Country`,

`PostalCode`)

VALUES

(12121, "DHL", "Peter", "peter@dhl.com", "1 crircle road", "boston", "MA", "USA", "68710"),

(13131, "UPS", "Bob", "bob@ups.com", "3 ups drive", "new york", "NY", "USA", "62110"),

(14141, "USPS", "Mike", "mike@usps.com", "1 boulevard", "cambridge", "MA", "USA", "38317"),

(15151, "FEDEX", "Petro", "p.morelli@fedex.com", "1 fast street", "seattle", "WA", "USA", "65170"),

(16161, "GLS", "Samantha", "sam.smith@gls.com", "4 sky parcel", "san francisco", "CA", "USA", "88314");

INSERT INTO `shoe\_shop\_team10`.`Products`

(`ProductID`,

`type`,

`ProductName`,

`Brand`,

`Model`,

`Size`,

`Color`,

`Gender`,

`DateAdded`,

`buyPrice`,

`Margin`,

`sellPrice`,

`StockLevel`,

`Status`,

`SupplierPrimary`)

VALUES

("NKE444", "Running Shoes", "speedy", "Nike", "SP234", "8", "Black", "M", "2021-01-13", 29, 0.4, 40.6, 95, "In Stock", 88888),

("ADD4567", "Everyday Sneaker", "gonzalo", "Adidas", "GG97", "8", "Blue", "M", "2021-01-13", 34, 0.4, 47.6, 49, "In Stock", 55555),

("AB5667", "Slip-Ons", "street runner", "Allbirds", "SR60", "8", "Red", "F", "2021-01-13", 32, 0.4, 44.8, 24, "In Stock", 77777),

("D1234567", "Boat Shoes", "boating duck", "Dunlop", "BD45", "8", "Green ", "U", "2021-01-13", 26, 0.4, 36.4, 170, "In Stock", 99999),

("AB2355", "Weather Repellent Shoes", "rain proof gtx", "Allbirds", "GTX", "8", "Black", "F", "2021-01-13", 29, 0.4, 40.6, 68, "In Stock", 55555),

("NKE97453", "High Tops", "skater boyss", "Nike", "SB46", "8", "Orange", "F", "2021-01-13", 28, 0.4, 39.2, 179, "In Stock", 66666),

("NKE445", "Running Shoes", "speedy", "Nike", "SP234", "9", "Black", "M", "2021-01-13", 29, 0.4, 40.6, 164, "In Stock", 88888),

("ADD4568", "Everyday Sneaker", "gonzalo", "Adidas", "GG97", "9", "Blue", "M", "2021-01-13", 34, 0.4, 47.6, 25, "In Stock", 99999),

("AB5668", "Slip-Ons", "street runner", "Allbirds", "SR60", "9", "Red", "F", "2021-01-13", 32, 0.4, 44.8, 158, "In Stock", 77777),

("D1234568", "Boat Shoes", "boating duck", "Dunlop", "BD45", "9", "Green ", "U", "2021-01-13", 26, 0.4, 36.4, 15, "In Stock", 55555),

("AB2356", "Weather Repellent Shoes", "rain proof gtx", "Allbirds", "GTX", "9", "Black", "F", "2021-01-13", 29, 0.4, 40.6, 105, "In Stock", 66666),

("NKE97454", "High Tops", "skater boyss", "Nike", "SB46", "9", "Orange", "F", "2021-01-13", 28, 0.4, 39.2, 130, "In Stock", 88888),

("NKE446", "Running Shoes", "speedy", "Nike", "SP234", "10", "Black", "M", "2021-01-13", 29, 0.4, 40.6, 128, "In Stock", 55555),

("ADD4569", "Everyday Sneaker", "gonzalo", "Adidas", "GG97", "10", "Blue", "M", "2021-01-13", 34, 0.4, 47.6, 2, "In Stock", 77777),

("AB5669", "Slip-Ons", "street runner", "Allbirds", "SR60", "10", "Red", "F", "2021-01-13", 32, 0.4, 44.8, 43, "In Stock", 99999),

("D1234569", "Boat Shoes", "boating duck", "Dunlop", "BD45", "10", "Green ", "U", "2021-01-13", 26, 0.4, 36.4, 176, "In Stock", 55555),

("AB2357", "Weather Repellent Shoes", "rain proof gtx", "Allbirds", "GTX", "10", "Black", "F", "2021-01-13", 29, 0.4, 40.6, 196, "In Stock", 66666),

("NKE97455", "High Tops", "skater boyss", "Nike", "SB46", "10", "Orange", "F", "2021-01-13", 28, 0.4, 39.2, 0, "Sold Out", 88888),

("NKE447", "Running Shoes", "speedy", "Nike", "SP234", "11", "Black", "M", "2021-01-13", 29, 0.4, 40.6, 48, "In Stock", 99999),

("ADD4570", "Everyday Sneaker", "gonzalo", "Adidas", "GG97", "11", "Blue", "M", "2021-01-13", 34, 0.4, 47.6, 132, "In Stock", 77777),

("RR2345", "Running Shoes", "everyday sprint", "Allbirds", "ES345", "9", "Black", "F", "2021-01-13", 27, 0.4, 37.8, 0, "Discontinued", 55555);

INSERT INTO `shoe\_shop\_team10`.`Customers`

(`CustomerID`,

`FirstNameCustomer`,

`LastNameCustomer`,

`Gender`,

`customerSince`,

`Address`,

`postalcode`,

`city`,

`state`,

`country`,

`customerType`,

`SalesRepID`,

`emailCust`)

VALUES

(34051, "Greg", "Cooper", "M", "2018-06-02", "9 New Saddle Street", "1887", "Wilmington", "MA", "USA", "B2B", 1004, "gaerghasg@gmail.com"),

(34052, "Mathew", "Morgan", "M", "2018-06-10", "353 Sugar St.", "70001", "Metairie", "LA", "USA", "B2C", NULL, "afafeaf@gaeg.com"),

(34053, "Sheldon ", "Floyd", "M", "2018-01-05", "874 East St.", "37205", "Nashville", "TN", "Germany", "B2C", NULL, "agaf@sgs.com"),

(34054, "Gerardo", "Curry", "M", "2018-10-09", "4 High Point Drive", "53150", "Muskego", "WI", "Norway", "B2C", NULL, "jfhsfhsjhsrtj@fkhrf.com"),

(34084, "Lola", "Carson", "F", "2019-10-06", "976 N. Durham Drive", "7052", "West Orange", "NJ ", "Brazil", "B2C", NULL, "asgdasgasg@hrdhgsg.com"),

(34056, "Amanda", "Gregory", "F", "2019-04-30", "926 Greystone Street", "22801", "Harrisonburg", "VA", "Turkey", "B2B", 1005, "hilfhkdfkud@gmail.com"),

(34057, "Rodolfo", "Gray", "M", "2018-06-11", "3 Littleton Ave", "15642", "Longwood", "FL", "Indonesia", "B2C", NULL, "ryilkry@dtykdtys.com"),

(34058, "Ronald", "Webster", "M", "2018-04-02", "93 Birchwood St.", "32779", "Irwin", "NJ", "Norway", "B2C", NULL, "dshadthadethae@srhasrh.com"),

(34059, "Jerry", "Riley", "M", "2018-09-29", "796 San Pablo Street", "7652", "Desoto", "OH", "Iraq", "B2C", NULL, "hadehath@fykfhysng.com"),

(34060, "Lorena", "Bowman", "F", "2019-07-12", "8 Fairground St.", "44139", "Solon", "NY ", "Kenya", "B2C", NULL, "arsharhadetjh@srhasrh.com"),

(34061, "Miriam", "Richards", "F", "2019-10-22", "8789 South Jockey Hollow Ave.", "12550", "Paramus", "TX", "Ukraine", "B2C", NULL, "dfhsdgjsf@hsgsg.com"),

(34062, "Shannon", "Alvarez", "F", "2018-08-03", "109 Theatre Dr.", "75115", "Newburgh", "PA", "Poland", "B2C", NULL, "dgtjnadtjha@sdtjat.com"),

(34063, "Janice", "Walters", "F", "2019-11-24", "577 N. Bay Meadows Road", "11757", "Linden", "MI", "Canada", "B2B", 1007, "dajnsfgmnrsa@dtgjndtjasd.com"),

(34064, "Jake", "Gonzalez", "M", "2018-12-17", "9863 Glenwood Ave.", "84010", "Freehold", "UT", "Canada", "B2C", NULL, "arharhah@srwhsarh.com"),

(34065, "Faith", "Copeland", "F", "2018-03-07", "7565 Devonshire St.", "48180", "Lindenhurst", "NJ ", "USA", "B2C", NULL, "eatjsyfmkjsf@jmsfjts.com"),

(34066, "David", "Maxwell", "M", "2018-02-16", "2 Eagle St.", "7728", "Bountiful", "AL", "Germany", "B2C", NULL, "aetjatj@atjadtjh.com"),

(34067, "Janet", "Baldwin", "F", "2019-01-23", "48 Old York Lane", "36532", "Fairhope", "MI", "Turkey", "B2B", 1006, "yjksyjs@sdtkjsfj.com"),

(34068, "Mamie", "Collins", "F", "2020-03-02", "545 Delaware Ave.", "7036", "Lindenhurst", "UT", "Poland", "B2C", NULL, "arhsfkjs@atjsdatja.com"),

(34069, "Nelson", "Harrison", "M", "2018-04-29", "52 Rose Drive", "17325", "Wyandotte", "RI", "Poland", "B2C", NULL, "srtjysrfyksr@atjsdj.com"),

(34070, "Beverly", "Francis", "F", "2019-01-09", "60 Garfield Dr.", "48192", "Gettysburg", "MA", "India ", "B2B", 1006, "yjksyjkwsryj@adztjksdtyyj.com");

INSERT INTO `shoe\_shop\_team10`.`PurchaseOrders`

(`PurchOrderID`,

`OrderDate`,

`OrderValue`,

`CompletedDate`,

`PurchOrderStatus`,

`SupplierID`)

VALUES

(11111, "2021-02-10", 9699.2, "2021-02-14", "Completed", 88888),

(22222, "2021-02-11", 7064.4, "2021-02-15", "Completed", 55555),

(33333, "2021-02-12", 3857, "2021-02-16", "Completed", 77777),

(44444, "2021-02-13", 1055.6, "2021-02-17", "Completed", 99999),

(55555, "2021-02-14", 1827, "2021-02-18", "Completed", 55555),

(66666, "2021-02-15", 1502.2, NULL, "Cancelled", 77777),

(77777, "2021-02-16", 2273.6, "2021-02-20", "Completed", 99999),

(88888, "2021-02-17", 4337.2, "2021-02-21", "Completed", 55555),

(99999, "2021-02-18", 4953.2, "2021-03-01", "Completed", 55555),

(111110, "2021-02-19", 1237.6, "2021-03-01", "Completed", 66666),

(122221, "2021-02-20", 1346.8, NULL, "Pending", 77777),

(133332, "2021-02-21", 1492.4, "2021-03-01", "Completed", 88888),

(144443, "2021-02-22", 2094.4, NULL, "Pending", 55555),

(155554, "2021-02-23", 1299.2, "2021-03-01", "Completed", 99999),

(166665, "2021-02-24", 1055.6, NULL, "Pending", 77777),

(177776, "2021-02-25", 1989.4, NULL, "Pending", 66666),

(188887, "2021-02-26", 2030, NULL, "Pending", 55555),

(199998, "2021-02-27", 1177.4, NULL, "Pending", 66666),

(211109, "2021-02-28", 1951.6, NULL, "Pending", 66666),

(222220, "2021-03-01", 940.8, NULL, "Pending", 88888);

INSERT INTO `shoe\_shop\_team10`.`PurchaseOrderDetails`

(`PurchOrderID`,

`ProductID`,

`quantity`,

`ETA\_Date`,

`Delivered Date`,

`ShipperID`,

`Status\_purchOD`)

VALUES

(11111, "AB2356", 37, "2021-02-13", "2021-02-13", 12121, "Delivered"),

(11111, "AB2357", 43, "2021-02-13", "2021-02-13", 12121, "Delivered"),

(11111, "AB5669", 35, "2021-02-13", "2021-02-13", 12121, "Delivered"),

(11111, "AB2355", 37, "2021-02-14", "2021-02-14", 12121, "Delivered"),

(11111, "AB5668", 43, "2021-02-14", "2021-02-14", 12121, "Delivered"),

(11111, "AB5667", 29, "2021-02-16", "2021-02-14", 12121, "Delivered"),

(22222, "D1234568", 25, "2021-02-13", "2021-02-13", 13131, "Delivered"),

(22222, "NKE97455", 29, "2021-02-13", "2021-02-13", 13131, "Delivered"),

(22222, "NKE97454", 38, "2021-02-13", "2021-02-13", 13131, "Delivered"),

(22222, "NKE97453", 44, "2021-02-13", "2021-02-13", 13131, "Delivered"),

(22222, "NKE445", 46, "2021-02-13", "2021-02-13", 13131, "Delivered"),

(33333, "NKE447", 49, "2021-02-16", "2021-02-16", 14141, "Delivered"),

(33333, "NKE446", 46, "2021-02-15", "2021-02-15", 14141, "Delivered"),

(44444, "NKE446", 26, "2021-02-17", "2021-02-17", 14141, "Delivered"),

(55555, "NKE445", 45, "2021-02-18", "2021-02-18", 14141, "Delivered"),

(66666, "NKE445", 37, "2021-02-18", NULL, NULL, "Cancelled"),

(77777, "NKE447", 34, "2021-02-20", "2021-02-20", 15151, "Delivered"),

(77777, "NKE444", 22, "2021-02-19", "2021-02-19", 15151, "Delivered"),

(88888, "D1234569", 42, "2021-02-22", "2021-02-20", 15151, "Delivered"),

(88888, "ADD4570", 39, "2021-02-22", "2021-02-20", 15151, "Delivered"),

(88888, "ADD4569", 20, "2021-02-22", "2021-02-20", 15151, "Delivered"),

(99999, "D1234568", 21, "2021-03-04", "2021-03-01", 16161, "Delivered"),

(99999, "ADD4568", 47, "2021-03-04", "2021-03-01", 16161, "Delivered"),

(99999, "ADD4567", 41, "2021-03-04", "2021-03-01", 15151, "Delivered"),

(111110, "D1234568", 34, "2021-03-01", "2021-03-01", 16161, "Delivered"),

(122221, "D1234567", 37, "2021-03-04", NULL, 15151, "Shipped"),

(133332, "D1234567", 41, "2021-03-04", NULL, 15151, "Shipped"),

(144443, "ADD4570", 44, "2021-03-05", NULL, 13131, "Shipped"),

(155554, "AB2357", 32, "2021-03-06", NULL, 13131, "Shipped"),

(166665, "AB2357", 26, "2021-03-07", NULL, 13131, "Shipped"),

(177776, "AB2356", 49, "2021-03-08", NULL, 12121, "Shipped"),

(188887, "AB2355", 50, NULL, NULL, NULL, "Confirmed"),

(199998, "AB2355", 29, NULL, NULL, NULL, "Confirmed"),

(211109, "ADD4567", 41, NULL, NULL, NULL, "Confirmed"),

(222220, "AB5667", 21, NULL, NULL, NULL, "Confirmed");

INSERT INTO `shoe\_shop\_team10`.`Orders`

(`OrderID`,

`OrderDate`,

`CustomerID`,

`OrderValue`,

`CompletedDate`,

`OrderStatus`)

VALUES

(101, "2021-02-01", 34051, 800, "2021-12-02", "Completed"),

(102, "2021-02-02", 34052, 59, "2021-02-13", "Completed"),

(103, "2021-02-03", 34053, 180, "2021-02-14", "Completed"),

(104, "2021-02-04", 34054, 73, "2021-02-15", "Completed"),

(121, "2021-02-05", 34084, 93, "2021-02-16", "Completed"),

(106, "2021-02-06", 34056, 893, "2021-02-17", "Completed"),

(107, "2021-02-07", 34057, 196, "2021-02-18", "Completed"),

(108, "2021-02-08", 34058, 180, "2021-02-19", "Completed"),

(109, "2021-02-09", 34059, 97, "2021-02-20", "Completed"),

(110, "2021-02-10", 34060, 122, NULL, "Cancelled"),

(111, "2021-02-11", 34061, 216, NULL, "Pending"),

(112, "2021-02-12", 34062, 62, NULL, NULL),

(113, "2021-02-13", 34063, 1240, NULL, "Pending"),

(114, "2021-02-14", 34064, 271, "2021-02-17", "Completed"),

(115, "2021-02-15", 34065, 154, NULL, "Pending"),

(116, "2021-02-16", 34066, 237, NULL, "Cancelled"),

(117, "2021-02-17", 34067, 1404, NULL, "Pending"),

(118, "2021-02-18", 34068, 73, "2021-02-20", "Completed"),

(119, "2021-02-19", 34069, 204, NULL, "Pending"),

(120, "2021-02-20", 34070, 1476, "2021-02-21", "Completed");

INSERT INTO `shoe\_shop\_team10`.`OrderDetails`

(`OrderID`,

`ProductID`,

`quantity`,

`ETA\_Date`,

`Delivered Date`,

`ShipperID`,

`Status\_OD`)

VALUES

(101, "ADD4567", 20, "2021-02-12", "2021-02-12", 12121, "Delivered"),

(102, "ADD4567", 3, "2021-02-13", "2021-02-13", 13131, "Delivered"),

(103, "AB2356", 1, "2021-02-14", "2021-02-14", 14141, "Delivered"),

(104, "D1234568", 1, "2021-02-15", "2021-02-15", 15151, "Delivered"),

(121, "D1234568", 2, "2021-02-16", "2021-02-16", 16161, "Delivered"),

(106, "ADD4568", 19, "2021-02-17", "2021-02-17", 12121, "Delivered"),

(107, "ADD4568", 1, "2021-02-18", "2021-02-18", 13131, "Delivered"),

(108, "NKE446", 3, "2021-02-19", "2021-02-19", 14141, "Delivered"),

(109, "AB5668", 1, "2021-02-20", "2021-02-20", 15151, "Delivered"),

(110, "ADD4568", 1, "2021-02-21", NULL, 16161, "Cancelled"),

(111, "AB5668", 3, "2021-02-22", NULL, 12121, "Shipped"),

(112, "AB2356", 1, "2021-02-23", NULL, 13131, "Shipped"),

(113, "ADD4569", 31, "2021-02-24", NULL, 14141, "Shipped"),

(114, "ADD4569", 2, "2021-02-20", "2021-02-17", 15151, "Delivered"),

(115, "NKE444", 2, "2021-02-18", NULL, 16161, "Shipped"),

(116, "ADD4568", 1, "2021-02-19", NULL, NULL, "Cancelled"),

(117, "NKE97453", 39, "2021-02-20", NULL, 12121, "Shipped"),

(118, "NKE446", 3, "2021-02-19", "2021-02-20", 13131, "Delivered"),

(119, "AB2356", 1, "2021-02-24", NULL, 14141, "Shipped"),

(120, "RR2345", 41, "2021-02-21", "2021-02-21", 15151, "Delivered"),

(121, "NKE97453", 1, "2021-02-22", NULL, 16161, "Shipped"),

(111, "D1234569", 1, "2021-02-23", NULL, NULL, "Pending"),

(118, "AB2356", 3, "2021-02-24", NULL, NULL, "Pending"),

(102, "D1234568", 2, "2021-02-20", NULL, NULL, "Pending"),

(120, "NKE97453", 1, "2021-02-18", NULL, NULL, "Cancelled"),

(108, "NKE97453", 3, "2021-02-16", NULL, NULL, "Pending"),

(115, "RR2345", 3, "2021-02-17", NULL, NULL, "Pending");

INSERT INTO `shoe\_shop\_team10`.`Payments`

(`PaymentID`,

`orderID`,

`amount`,

`paymentdate`,

`paymentType`)

VALUES

(1001, 101, 422, "2021-02-12", "CC"),

(1002, 102, 59, "2021-02-13", "Debit"),

(1003, 103, 180, "2021-02-14", "Check"),

(1004, 104, 73, "2021-02-15", "Bank Transfer"),

(1005, 121, 86, "2021-02-16", "CC"),

(1006, 106, 410, "2021-02-17", "Debit"),

(1007, 107, 196, "2021-02-18", "Check"),

(1008, 108, 343, "2021-02-19", "Bank Transfer"),

(1009, 109, 97, "2021-02-20", "CC"),

(1010, 110, 122, "2021-02-21", "Debit"),

(1011, 111, 216, "2021-02-22", "Check"),

(1012, 112, 62, "2021-02-23", "Bank Transfer"),

(1013, 113, 306, "2021-02-24", "CC"),

(1014, 114, 271, "2021-02-25", "Debit"),

(1015, 115, 154, "2021-02-26", "Check"),

(1016, 116, 237, "2021-02-27", "Bank Transfer"),

(1017, 117, 428, "2021-02-28", "CC"),

(1018, 118, 73, "2021-02-18", "Debit"),

(1019, 119, 204, "2021-02-19", "Check");

# Works Cited

medium. (2010). *stackoverflow.com*. Retrieved from MYSQL nested loop stored procedure issue: https://stackoverflow.com/questions/2200544/mysql-nested-loop-stored-procedure-issue

Lucid. (2021). *Lucid app*. Retrieved from Lucid: https://www.lucidchart.com/pages/