Data Model

Customer_Info
- CustomerID INT
- Name VARCHAR(45) VARCHAR(45)
- Phone1 INT
- Phone2 INT
- CustomerZIP INT
- CustomerCity VARCHAR(45)
- CustomerState VARCHAR(32)
- Address1 VARCHAR(45)
- Address2 VARCHAR(45)
- ReferredBy VARCHAR(45)
- Mowing TINYINT
- Landscaping TINYINT
- Other TINYINT
- CustomerNotes VARCHAR(128)
- Indexes

CustomerInvoice
- InvoiceID INT
- CustomerID INT
- InvoiceDate DATE
- TotalAmountDueNow DECIMAL(9,2)
- AmountRemitted DECIMAL(9,2)
- TotalCustomerCharges DECIMAL(9,2)
- TotalCustomerCredits DECIMAL(9,2)
- PriorBalance DECIMAL(9,2)
- TotalNewCharges DECIMAL(9,2)
- Taxes DECIMAL(9,2)
- TotalCredits DECIMAL(9,2)
- TotalAmountDue DECIMAL(9,2)
- Indexes

Services
- ServiceID INT
- ServicesRendered VARCHAR(512)
- Indexes

Land Scaping Inventory needs to update/be updated

LS_Materials
- LSMaterialsID INT
- Name VARCHAR(45)
- Description VARCHAR(512)
- Quantity INT
- Price DECIMAL(9,2)
- AnnualDepreciation DECIMAL(9,2)
- Indexes

InvoiceDetails
- InvoiceDetailsID INT
- InvoiceID INT
- ServicesID INT
- LSMaterialsID INT
- LiveStockID INT
- Charges DECIMAL(9,2)
- Credits DECIMAL(9,2)
- PurchaseDate DATE
- Indexes

LiveStock
- LiveStockID INT
- Name VARCHAR(45)
- Description VARCHAR(45)
- Quantity INT
- Price DECIMAL(9,2)
- PlantAge INT
- Indexes

Association Entity

LS_MaterialsOrders
- ConfNum INT
- LSMaterialsID INT
- Indexes

OrderLog
- ConfNum INT
- Date DATE
- Supplier VARCHAR(45)
- Item VARCHAR(45)
- Quantity INT
- ExPrice DECIMAL(9,2)
- Taxes DECIMAL(9,2)
- Shipping DECIMAL(9,2)
- Total DECIMAL(9,2)
- Category VARCHAR(45)
- ShipperNumber INT
- Indexes

LiveStock updates OrderLog, vice versa

Association Entity

LiveStockOrders
- LiveStockID INT
- ConfNum INT
- Indexes

Association Entity

OrderSupplier
- ConfNum INT
- SupplierInfoID INT
- Indexes

Supplier and OrderLog need mutual access

BOL
- ShipperNumber INT
- SupplierInfoID INT
- InvoiceNum INT
- OrderDate DATE
- Control VARCHAR(45)
- ItemID INT
- QuantityOrdered INT
- QuantityShipped INT
- QuantityBackOrd INT
- ETA DATE
- Description VARCHAR(128)
- UOM VARCHAR(45)
- UnitPrice DECIMAL(9,2)
- ExtendedAmount DECIMAL(9,2)
- Phone INT
- SpecialInstructions VARCHAR(1024)
- NumberOfItems INT
- NumberOfPieces INT
- OrderLog_ConfNum INT
- Indexes

Supplier
- SupplierInfoID INT
- Preferred TINYINT
- Category VARCHAR(45)
- Name VARCHAR(45)
- Address1 VARCHAR(45)
- Address2 VARCHAR(45)
- City VARCHAR(45)
- State VARCHAR(32)
- ZIP INT
- OrderPhone INT
- Contact VARCHAR(45)
- ContactPhone INT
- ShippingBillingTerms VARCHAR(512)
- Notes VARCHAR(512)
- Indexes

Single Supplier may have multiple Bills of Lading
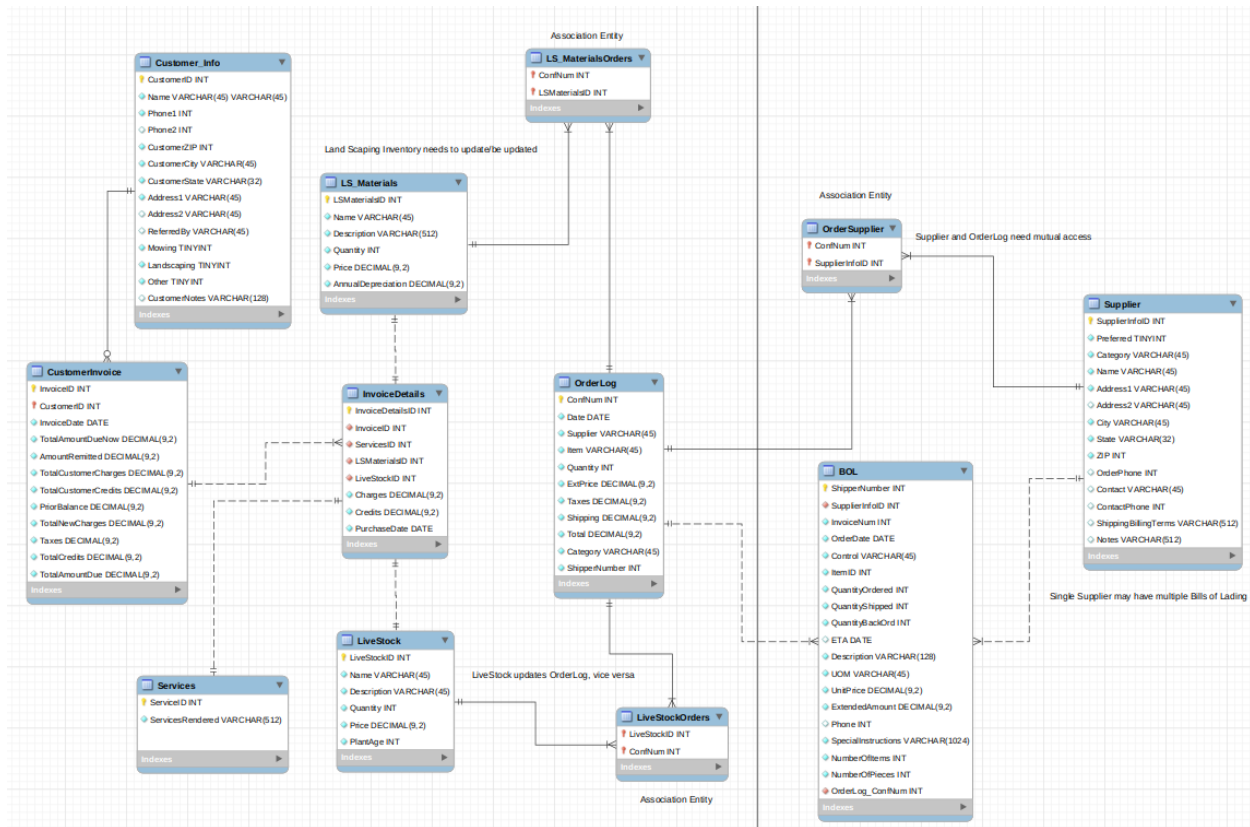
DDL

-- MySQL Script generated by MySQL Workbench

-- Wed 30 Oct 2019 09:55:06 PM MDT

-- Model: New Model   Version: 1.0

-- 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 mydb

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


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

-- Schema mydb

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

CREATE SCHEMA IF NOT EXISTS `mydb` DEFAULT CHARACTER SET utf8 ;

USE `mydb` ;


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

-- Table `mydb`.`table1`

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

CREATE TABLE IF NOT EXISTS `mydb`.`table1` (

)

ENGINE = InnoDB;



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

-- Table `mydb`.`Customer_Info`

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

CREATE TABLE IF NOT EXISTS `mydb`.`Customer_Info` (

  `CustomerID` INT NOT NULL,

  `Name VARCHAR(45)` VARCHAR(45) NOT NULL,

  `Phone1` INT NOT NULL,

  `Phone2` INT NULL,

  `CustomerZIP` INT NOT NULL,

  `CustomerCity` VARCHAR(45) NOT NULL,

  `CustomerState` VARCHAR(32) NOT NULL,

  `Address1` VARCHAR(45) NOT NULL,
```

```
  `Address2` VARCHAR(45) NULL,

  `ReferredBy` VARCHAR(45) NULL,

  `Mowing` TINYINT NOT NULL,

  `Landscaping` TINYINT NOT NULL,

  `Other` TINYINT NOT NULL,

  `CustomerNotes` VARCHAR(128) NULL,

  PRIMARY KEY (`CustomerID`))

ENGINE = InnoDB;




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

-- Table `mydb`.`CustomerInvoice`

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

CREATE TABLE IF NOT EXISTS `mydb`.`CustomerInvoice` (

  `InvoiceID` INT NOT NULL,

  `CustomerID` INT NOT NULL,

  `InvoiceDate` DATE NOT NULL,

  `TotalAmountDueNow` DECIMAL(9,2) NOT NULL,

  `AmountRemitted` DECIMAL(9,2) NOT NULL,

  `TotalCustomerCharges` DECIMAL(9,2) NOT NULL,

  `TotalCustomerCredits` DECIMAL(9,2) NOT NULL,

  `PriorBalance` DECIMAL(9,2) NOT NULL,

  `TotalNewCharges` DECIMAL(9,2) NOT NULL,

  `Taxes` DECIMAL(9,2) NOT NULL,

  `TotalCredits` DECIMAL(9,2) NOT NULL,

  `TotalAmountDue` DECIMAL(9,2) NOT NULL,

  PRIMARY KEY (`InvoiceID`, `CustomerID`),

  INDEX `fk_CustomerInvoice_Customers_idx` (`CustomerID` ASC) VISIBLE,

  CONSTRAINT `fk_CustomerInvoice_Customers`

    FOREIGN KEY (`CustomerID`)

    REFERENCES `mydb`.`Customer_Info` (`CustomerID`)
```

```
    ON DELETE NO ACTION

    ON UPDATE NO ACTION)

ENGINE = InnoDB;




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

-- Table `mydb`.`Services`

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

CREATE TABLE IF NOT EXISTS `mydb`.`Services` (

  `ServiceID` INT NOT NULL,

  `ServicesRendered` VARCHAR(512) NOT NULL,

  PRIMARY KEY (`ServiceID`))

ENGINE = InnoDB;




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

-- Table `mydb`.`LS_Materials`

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

CREATE TABLE IF NOT EXISTS `mydb`.`LS_Materials` (

  `LSMaterialsID` INT NOT NULL,

  `Name` VARCHAR(45) NOT NULL,

  `Description` VARCHAR(512) NOT NULL,

  `Quantity` INT NOT NULL,

  `Price` DECIMAL(9,2) NOT NULL,

  `AnnualDepreciation` DECIMAL(9,2) NOT NULL,

  PRIMARY KEY (`LSMaterialsID`))

ENGINE = InnoDB;




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

-- Table `mydb`.`LiveStock`
```

```sql
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `mydb`.`LiveStock` (
  `LiveStockID` INT NOT NULL,
  `Name` VARCHAR(45) NOT NULL,
  `Description` VARCHAR(45) NOT NULL,
  `Quantity` INT NOT NULL,
  `Price` DECIMAL(9,2) NOT NULL,
  `PlantAge` INT NOT NULL,
  PRIMARY KEY (`LiveStockID`))
ENGINE = InnoDB;



-- -----------------------------------------------------
-- Table `mydb`.`InvoiceDetails`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `mydb`.`InvoiceDetails` (
  `InvoiceDetailsID` INT NOT NULL,
  `InvoiceID` INT NOT NULL,
  `ServicesID` INT NOT NULL,
  `LSMaterialsID` INT NOT NULL,
  `LiveStockID` INT NOT NULL,
  `Charges` DECIMAL(9,2) NOT NULL,
  `Credits` DECIMAL(9,2) NOT NULL,
  `PurchaseDate` DATE NOT NULL,
  PRIMARY KEY (`InvoiceDetailsID`),
  INDEX `fk_InvoiceDetails_CustomerInvoice1_idx` (`InvoiceID` ASC) VISIBLE,
  INDEX `fk_InvoiceDetails_Services1_idx` (`ServicesID` ASC) VISIBLE,
  INDEX `fk_InvoiceDetails_LS_Materials1_idx` (`LSMaterialsID` ASC) VISIBLE,
  INDEX `fk_InvoiceDetails_LiveStock1_idx` (`LiveStockID` ASC) VISIBLE,
  CONSTRAINT `fk_InvoiceDetails_CustomerInvoice1`
    FOREIGN KEY (`InvoiceID`)
```

```
    REFERENCES `mydb`.`CustomerInvoice` (`InvoiceID`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `fk_InvoiceDetails_Services1`
    FOREIGN KEY (`ServicesID`)
    REFERENCES `mydb`.`Services` (`ServiceID`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `fk_InvoiceDetails_LS_Materials1`
    FOREIGN KEY (`LSMaterialsID`)
    REFERENCES `mydb`.`LS_Materials` (`LSMaterialsID`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `fk_InvoiceDetails_LiveStock1`
    FOREIGN KEY (`LiveStockID`)
    REFERENCES `mydb`.`LiveStock` (`LiveStockID`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;


-- -----------------------------------------------------
-- Table `mydb`.`OrderLog`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `mydb`.`OrderLog` (
  `ConfNum` INT NOT NULL,
  `Date` DATE NOT NULL,
  `Supplier` VARCHAR(45) NOT NULL,
  `Item` VARCHAR(45) NOT NULL,
  `Quantity` INT NOT NULL,
  `ExtPrice` DECIMAL(9,2) NOT NULL,
```

```
  `Taxes` DECIMAL(9,2) NOT NULL,

  `Shipping` DECIMAL(9,2) NOT NULL,

  `Total` DECIMAL(9,2) NOT NULL,

  `Category` VARCHAR(45) NOT NULL,

  `ShipperNumber` INT NOT NULL,

  PRIMARY KEY (`ConfNum`))

ENGINE = InnoDB;




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

-- Table `mydb`.`LS_MaterialsOrders`

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

CREATE TABLE IF NOT EXISTS `mydb`.`LS_MaterialsOrders` (

  `ConfNum` INT NOT NULL,

  `LSMaterialsID` INT NOT NULL,

  INDEX `fk_LS_MaterialsOrders_OrderLog1_idx` (`ConfNum` ASC) VISIBLE,

  PRIMARY KEY (`LSMaterialsID`, `ConfNum`),

  CONSTRAINT `fk_LS_MaterialsOrders_OrderLog1`

    FOREIGN KEY (`ConfNum`)

    REFERENCES `mydb`.`OrderLog` (`ConfNum`)

    ON DELETE NO ACTION

    ON UPDATE NO ACTION,

  CONSTRAINT `fk_LS_MaterialsOrders_LS_Materials1`

    FOREIGN KEY (`LSMaterialsID`)

    REFERENCES `mydb`.`LS_Materials` (`LSMaterialsID`)

    ON DELETE NO ACTION

    ON UPDATE NO ACTION)

ENGINE = InnoDB;




-- ---------------------------------------------------
```

-- Table `mydb`.`LiveStockOrders`

-- ---

CREATE TABLE IF NOT EXISTS `mydb`.`LiveStockOrders` (

 `LiveStockID` INT NOT NULL,

 `ConfNum` INT NOT NULL,

 INDEX `fk_LiveStockOrders_LiveStock1_idx` (`LiveStockID` ASC) VISIBLE,

 PRIMARY KEY (`ConfNum`, `LiveStockID`),

 CONSTRAINT `fk_LiveStockOrders_LiveStock1`

  FOREIGN KEY (`LiveStockID`)

  REFERENCES `mydb`.`LiveStock` (`LiveStockID`)

  ON DELETE NO ACTION

  ON UPDATE NO ACTION,

 CONSTRAINT `fk_LiveStockOrders_OrderLog1`

  FOREIGN KEY (`ConfNum`)

  REFERENCES `mydb`.`OrderLog` (`ConfNum`)

  ON DELETE NO ACTION

  ON UPDATE NO ACTION)

ENGINE = InnoDB;

-- ---

-- Table `mydb`.`Supplier`

-- ---

CREATE TABLE IF NOT EXISTS `mydb`.`Supplier` (

 `SupplierInfoID` INT NOT NULL,

 `Preferred` TINYINT NOT NULL,

 `Category` VARCHAR(45) NOT NULL,

 `Name` VARCHAR(45) NOT NULL,

 `Address1` VARCHAR(45) NOT NULL,

 `Address2` VARCHAR(45) NULL,

 `City` VARCHAR(45) NOT NULL,

```
  `State` VARCHAR(32) NOT NULL,

  `ZIP` INT NOT NULL,

  `OrderPhone` INT NULL,

  `Contact` VARCHAR(45) NULL,

  `ContactPhone` INT NULL,

  `ShippingBillingTerms` VARCHAR(512) NULL,

  `Notes` VARCHAR(512) NULL,

  PRIMARY KEY (`SupplierInfoID`))

ENGINE = InnoDB;




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

-- Table `mydb`.`BOL`

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

CREATE TABLE IF NOT EXISTS `mydb`.`BOL` (

  `ShipperNumber` INT NOT NULL,

  `SupplierInfoID` INT NOT NULL,

  `InvoiceNum` INT NOT NULL,

  `OrderDate` DATE NOT NULL,

  `Control` VARCHAR(45) NOT NULL,

  `ItemID` INT NOT NULL,

  `QuantityOrdered` INT NOT NULL,

  `QuantityShipped` INT NOT NULL,

  `QuantityBackOrd` INT NOT NULL,

  `ETA` DATE NULL,

  `Description` VARCHAR(128) NOT NULL,

  `UOM` VARCHAR(45) NOT NULL,

  `UnitPrice` DECIMAL(9,2) NOT NULL,

  `ExtendedAmount` DECIMAL(9,2) NOT NULL,

  `Phone` INT NULL,

  `SpecialInstructions` VARCHAR(1024) NOT NULL,
```

```sql
  `NumberOfItems` INT NOT NULL,

  `NumberOfPieces` INT NOT NULL,

  `OrderLog_ConfNum` INT NOT NULL,

  PRIMARY KEY (`ShipperNumber`),

  INDEX `fk_Shipment_SupplierInfo1_idx` (`SupplierInfoID` ASC) VISIBLE,

  INDEX `fk_BOL_OrderLog1_idx` (`OrderLog_ConfNum` ASC) VISIBLE,

  CONSTRAINT `fk_Shipment_SupplierInfo1`

    FOREIGN KEY (`SupplierInfoID`)

    REFERENCES `mydb`.`Supplier` (`SupplierInfoID`)

    ON DELETE NO ACTION

    ON UPDATE NO ACTION,

  CONSTRAINT `fk_BOL_OrderLog1`

    FOREIGN KEY (`OrderLog_ConfNum`)

    REFERENCES `mydb`.`OrderLog` (`ConfNum`)

    ON DELETE NO ACTION

    ON UPDATE NO ACTION)

ENGINE = InnoDB;




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

-- Table `mydb`.`OrderSupplier`

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

CREATE TABLE IF NOT EXISTS `mydb`.`OrderSupplier` (

  `ConfNum` INT NOT NULL,

  `SupplierInfoID` INT NOT NULL,

  INDEX `fk_OrderSupplier_SupplierInfo1_idx` (`SupplierInfoID` ASC) VISIBLE,

  PRIMARY KEY (`ConfNum`, `SupplierInfoID`),

  CONSTRAINT `fk_OrderSupplier_OrderLog1`

    FOREIGN KEY (`ConfNum`)

    REFERENCES `mydb`.`OrderLog` (`ConfNum`)

    ON DELETE NO ACTION
```

```
    ON UPDATE NO ACTION,

  CONSTRAINT `fk_OrderSupplier_SupplierInfo1`

   FOREIGN KEY (`SupplierInfoID`)

   REFERENCES `mydb`.`Supplier` (`SupplierInfoID`)

   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;
```

Assumptions

- A customer doesn't necessarily need to have any orders
- Bill of Lading is provided from each Supplier upon delivery
- when an order is placed (filed on the OrderLog), the Supplier will provide some shipment (doesn't necessarily need to be the correct shipment, incorrect shipping can be seen when comparing Bill of Lading with Order Log)

Surrogate Keys:

- All surrogate keys are created when no good Candidate Keys exist in the respective table
- CustomerID (from Customer_Info)
- CustomerInvoiceID (from CustomerInvoice)
- ServiceID (from Services)
- InvoiceDetailsID (from InvoiceDetails)
- SupplierInfoID (from Supplier)
- Non-surrogates:
    - ShipperNumber (from BOL)
    - ConfNum (from OrderLog)