Name: Joshua Steward Lab 6

Database Systems October 4, 2017

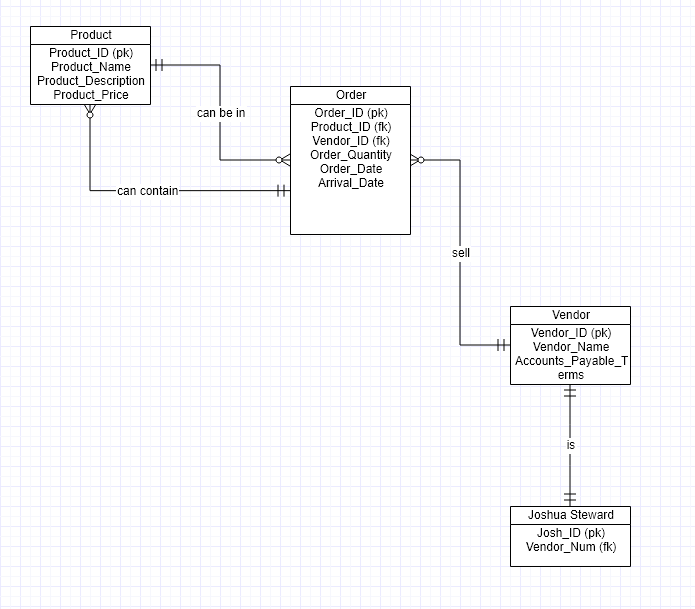
**Part I. Get Ready (5pts)**

Refer to the data file AirSupply-2017.xlsx

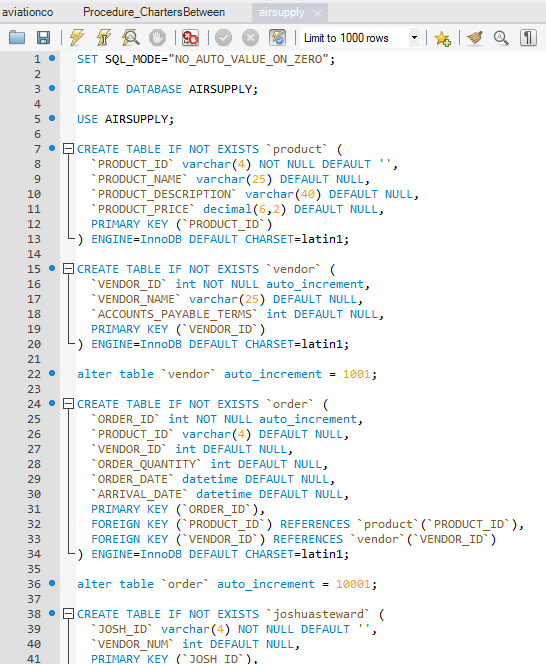
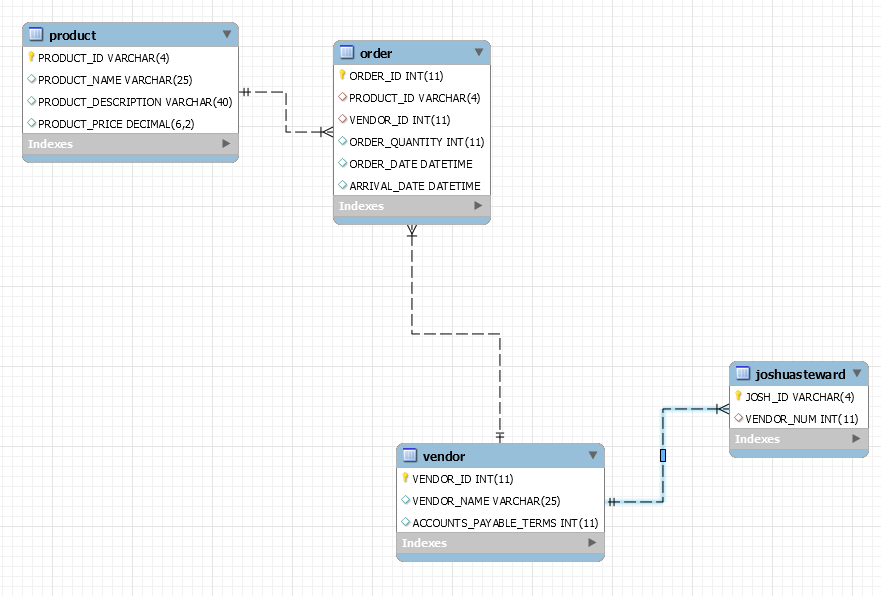
Background: The Excel sheet is based on supply orders placed against multiple suppliers. Many items can be sold by many different suppliers (i.e. M:N). This means that same items may only share a common description, not a common item number. A/P refers to the accounts payable, or the number of days before payment is due. The default arrival date should be set to NULL for new orders. Cost per order can be calculated by item cost and quantity.

1. Use Gliffy to deconstruct the data within AirSupply-2017.xlsx. Create the ERD by breaking the worksheet into 3rd Normal Form (3NF). Your ERD should use Crows Foot Notation and include:

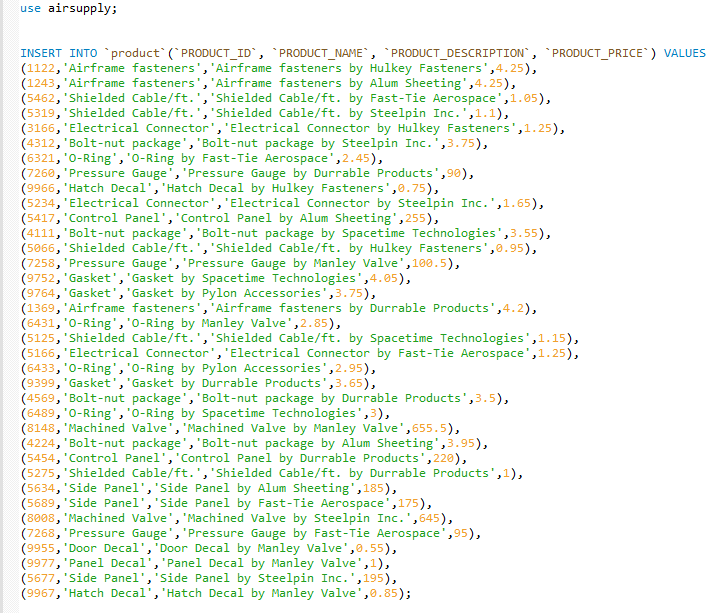
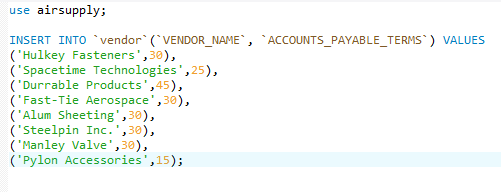
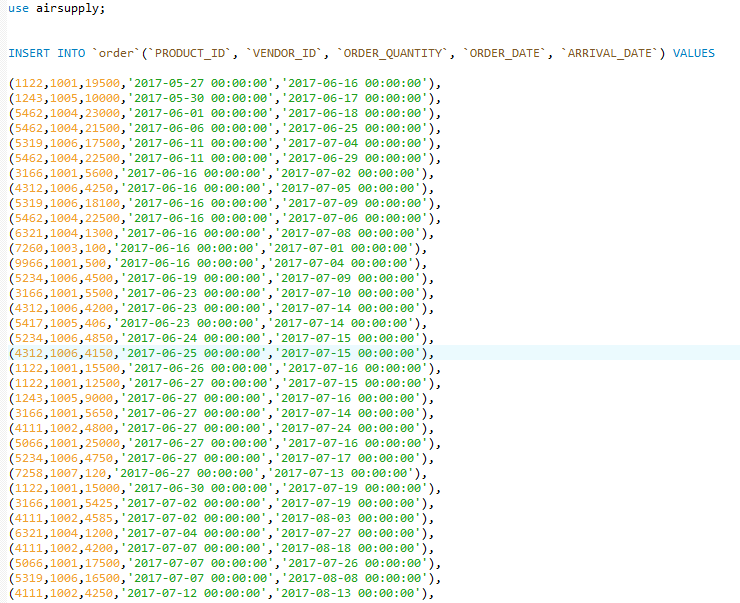
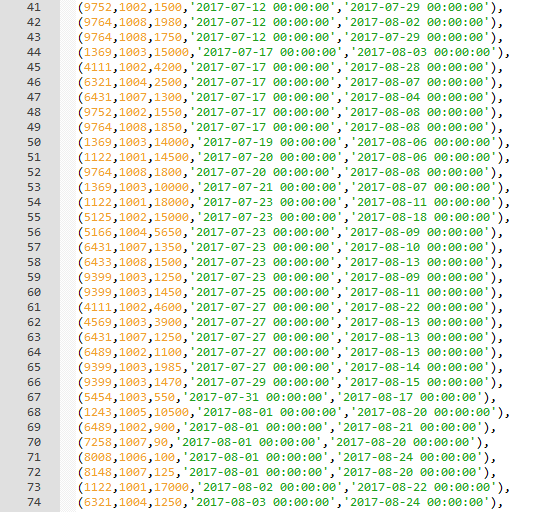
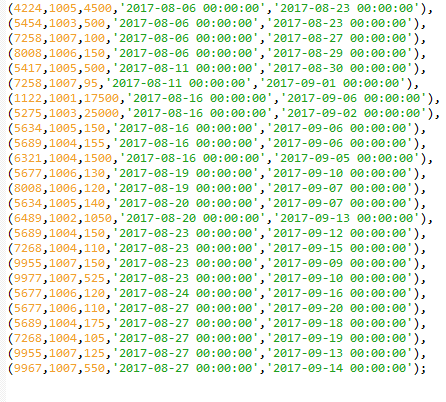
* 3rd Normal Form – each cell has single value
  + Each record is unique
  + Single column primary key
  + No transitive functional dependencies
  1. Relationship Types (i.e. Vendors Sell Products)
  2. Connectivity (i.e. Crow’s Foot Notation)
  3. Primary Keys
  4. Foreign Keys
  5. Proper Naming Conventions for entity name.
  6. Add one additional entity *yourname* with a single attribute to store vendor number. Link this entity to Vendor.

Note: Keep in mind that a product’s price can be both transactional and static data.   
  


**Part II. Get Set (15pts)**

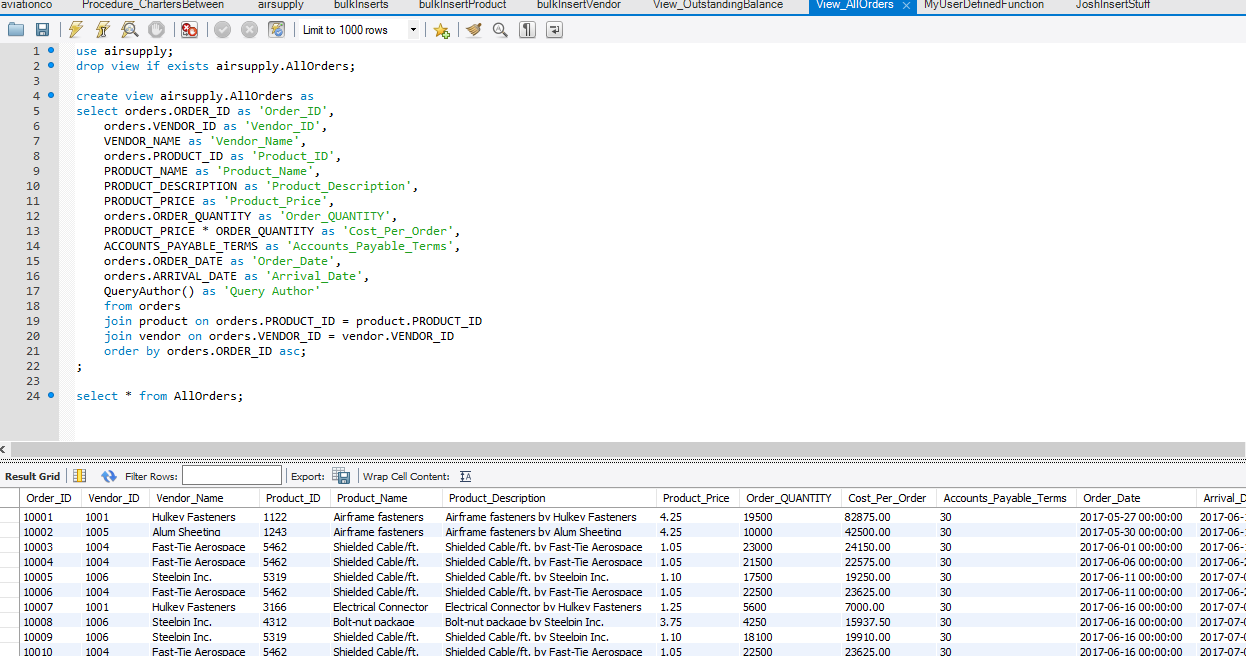
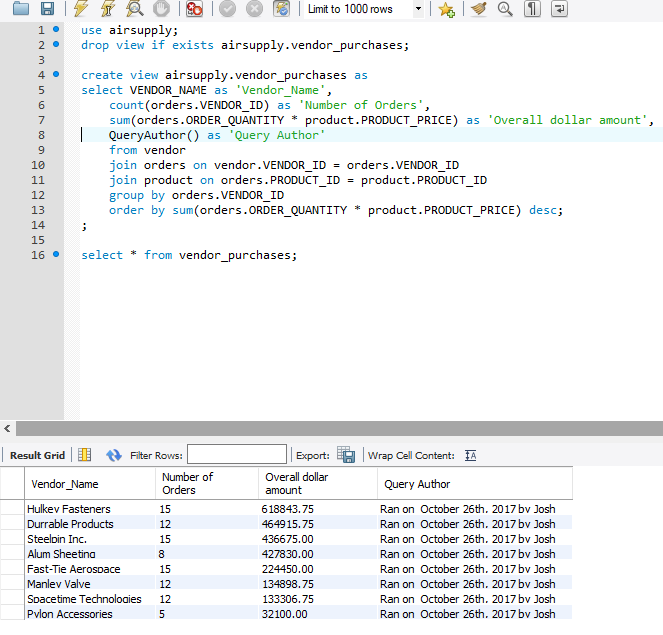
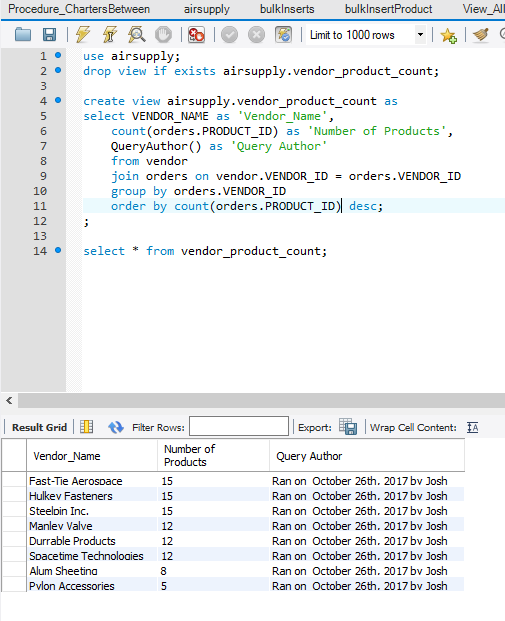
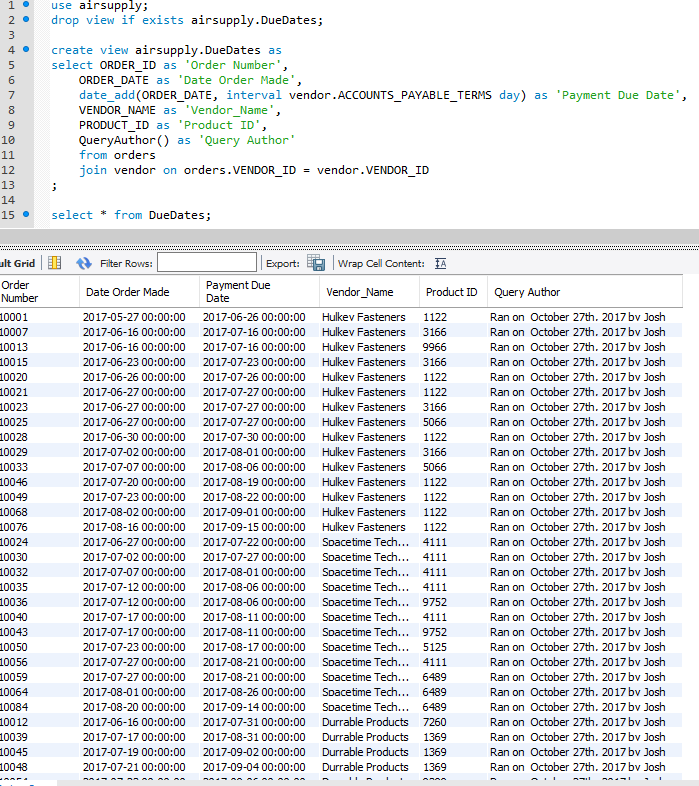
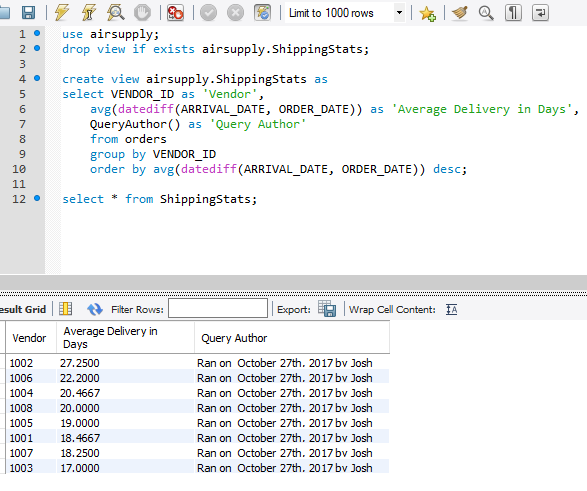
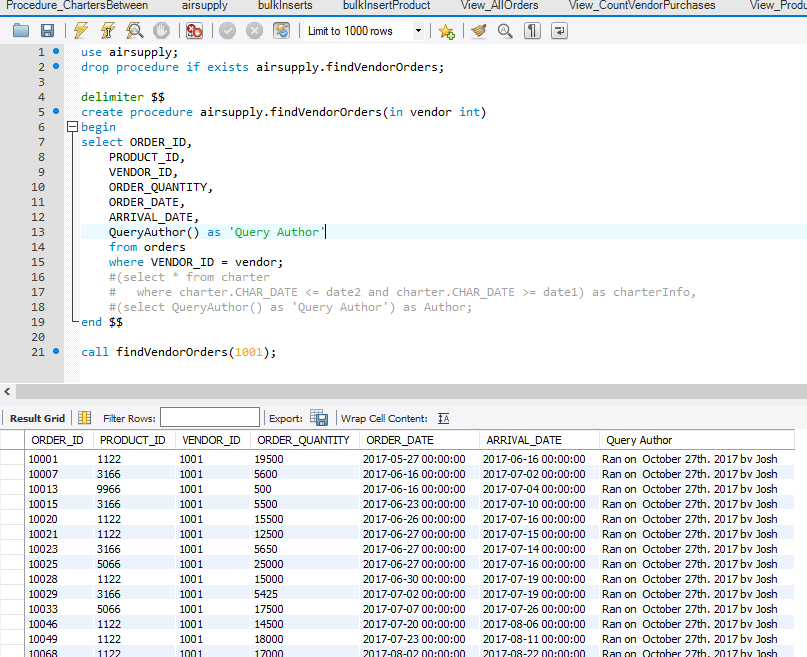
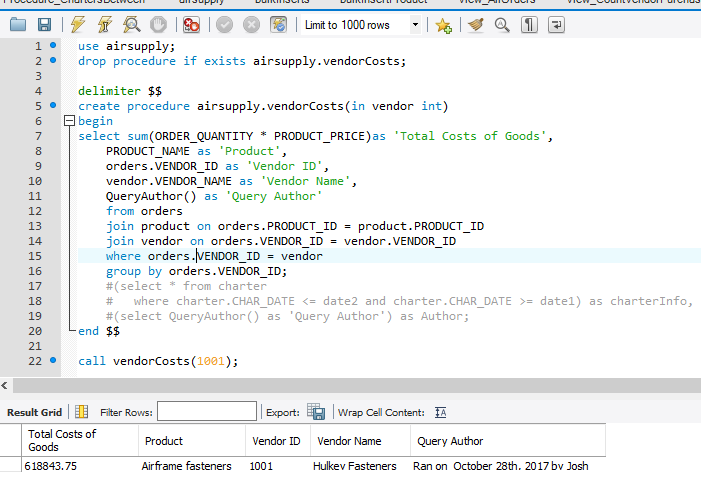
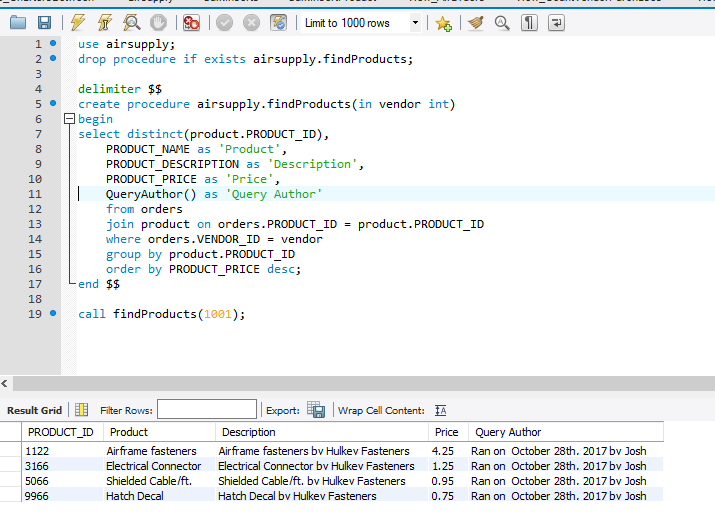
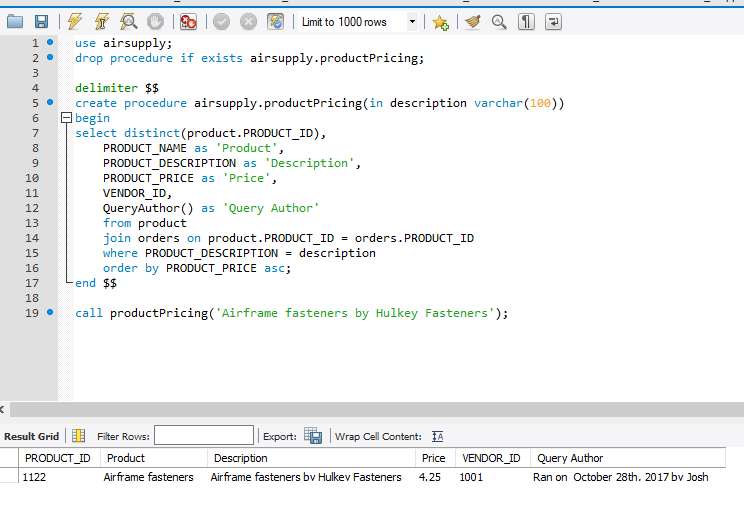
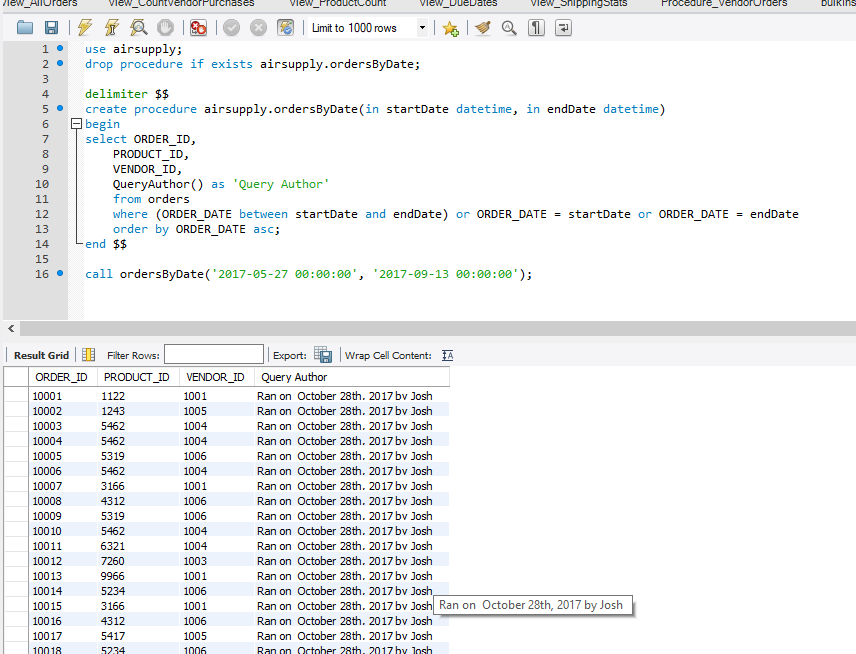
1. Create the database schema for airsupply.
   1. Create the DDL based on Part I (Note: Do not include derived attributes in your DDL). Do not include derived attributes in your DDL.
   2. Insert vendor information into Vendor with your information:
      * Vendor Name: Your Name
      * Vendor No: Auto-increment
      * A/P Terms: 30 days
   3. Also add avendor number for *yourname*
   4. Reverse engineer your database in MySQL Workbench.  
        
        
        
        
        
        
        
        
      

**Part III. Go (5pts)**

1. Construct *n* bulk insert statements (1 per entity) to insert the data found in AirSupply-2017.xlsx.
   1. Note: Data cleansing **may** be required. Do not include data for derived attributes.  
        
        
        
        
        
        
        
        
        
        
        
        
        
        
        
        
      

**Part IV. Go (20pts)**

For all views and stored procedures, include a column calling your function from Lab 5, which displays your full name and the current date using now(). Also be sure to capture all relevant information in your queries.

1. Create the view `all\_orders` that recreates the output found in AirSupply-2017.xlsx (minus the heading Orders and Supplies. Also include a column with *yourname*.  
     
     
   
2. Create the view `vendor\_purchases` that returns a listing of vendors and the total amount of purchases made with that vendor. Sort your results from largest to smallest overall dollar amount.  
     
   
3. Create the view `vendor\_product\_count` that returns a listing of vendors and the number of products provided by each vendor. Sort your results high to low.   
     
   
4. Create the view `payments` that outputs the future dates for when payments are due. Use the accounts payable terms to determine the last date for payment. Capture all relevant information from order, item and vendor tables.   
     
   
5. Create the view `vendor\_shipping\_stats` that produces a listing of the average ship times for each vendor. Sort the results by longest to shortest duration.  
     
     
   
6. Create the stored procedure `find\_vendor\_orders` that accepts vendor data as input and produces a listing of orders placed with that vendor.  
     
     
   
7. Create the stored procedure `show\_vendor\_purchases` that accepts a vendor number as input and returns the total cost of goods purchased through that vendor. Sort by the total cost highest to lowest and capture all relevant information from order, item and vendor tables.  
     
   
8. Create the stored procedure `find\_products` that accepts a vendor number as input and outputs all relevant product information for products manufactured by that vendor, including item number, description and cost. Sort by vendor lowest to highest.   
     
     
   
9. Create the stored procedure `show\_product\_pricing` that accepts an item description as input and returns a listing of vendors and their prices for that specific product. Sort results by item cost low to high.   
     
     
   
10. Create the stored procedure `show\_orders\_by\_date` that accepts a start date and end date as parameters and returns all order, product and vendor information for orders placed on or between those dates sorted by order date. Capture all relevant information from order, item and vendor tables.  
      
      
    
11. (Extra Credit) Assuming the current 3-month interest rate for a U.S. treasury 0.53% (as of Feb. 28, 2017), construct a view that calculates the total dollar amount the company saves by not paying until the last possible date according to the A/P terms of the vendor versus paying on the invoice date. Capture all relevant information from order, item and vendor tables.