Name: Melissa Hunfalvay

**Part 2 – Construct your Warehouse SQL statements**

1. [SQL Week 2 Question 200-501] For every shipment in the shipment table, you want to know its shipment details, the customer information, and the employee information.

Include exactly and only all the records for which a shipment ID exists.

* 1. Sort it ascending by Scantag.
  2. Your output should look a little like this. Note the “shipment information” will be in the form of several distinct columns. (Hint: look up the “ Select \* ” command to select several columns at once.)

-- Selects all columns from the required tables but does not concatenate them

SELECT shipment.\*, customer.\*, employee.\*

From shipment

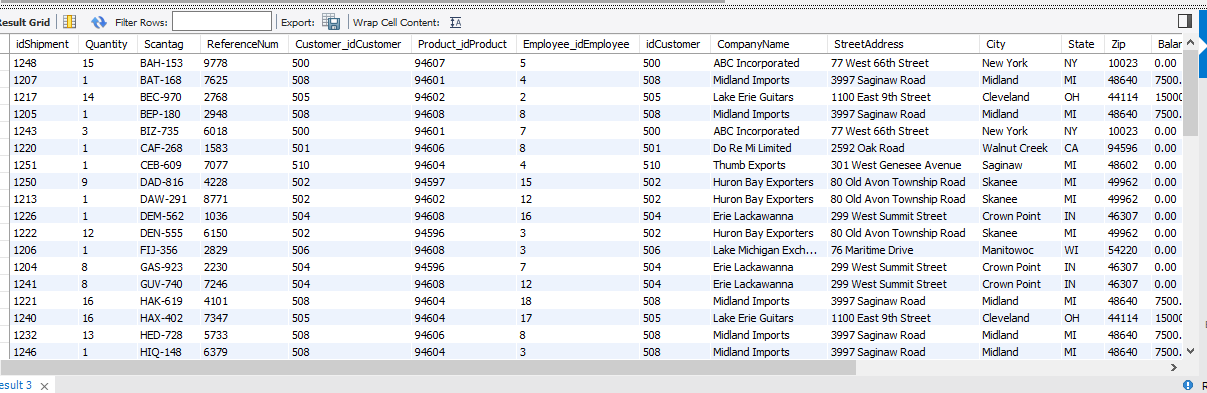
-- Joins

LEFT Join Customer ON Customer\_idCustomer= idCustomer

LEFT Join Employee ON shipment.Employee\_idEmployee= idEmployee

-- ORDER by

Order By warehouse.shipment.Scantag ASC;



1. [SQL Week 2 Question 200-502] You want to make a report which will give you information about shipments sent by your lower-commission employees. Create a query which will
   1. Include only all those shipments which were sent by an employee whose commission is 10% or less (include commissions of exactly 10%)
   2. Print the state, company name, total quantity in the shipments for that customer, and the maximum employee commission in that order left to right.
   3. Sort it alphabetically by state (ascending). If one state has several companies in it, they should be sorted within the state, with the highest total order quantity on top.

SELECT c.state, c.CompanyName, SUM(s.quantity), MAX(e.Commission)

FROM employee AS e JOIN shipment AS s

ON e.IDemployee = s.Employee\_IDEmployee

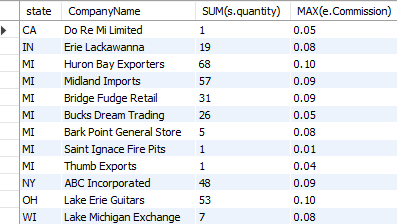
JOIN customer AS c

ON s.Customer\_idCustomer = c.idCustomer

WHERE e.Commission <= 0.1

GROUP BY c.state, c.CompanyName

ORDER BY c.state, SUM(s.quantity) DESC;



1. [SQL Week 2 Question 200-503] You suspect certain employees are not servicing very many shipments, and some may not have any shipments at all.

Create SQL to run a report which will

* 1. List all your employees and the customers they serviced along with the number of shipments for each customer. Include all employees, including those who may not have serviced any customers.
  2. Note you want the number of shipments, not the sum of shipment quantity. For example, if you have one shipment of 10 units, and another shipment of 20 units, our number of shipments here is 2.
  3. Sort the list ascending on number of shipments (so the smallest numbers of shipments are at the top), and then on employee id. If an employee has not serviced any customers, display that employee at the very top.

SELECT e.idEmployee, e.EmpFirstName, c.idCustomer, COUNT(idShipment) AS NumberOfShipments

FROM employee AS e LEFT OUTER JOIN shipment AS s

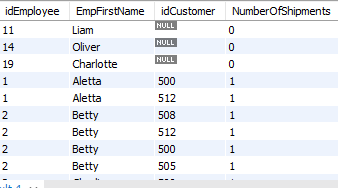
ON e.IDemployee = s.Employee\_IDEmployee

LEFT OUTER JOIN customer AS c

ON s.Customer\_idCustomer = c.idCustomer

GROUP BY e.idEmployee, e.EmpFirstName, c.idCustomer

ORDER BY COUNT(idShipment) ASC, e.idEmployee ASC;



1. [SQL Week 2 Question 200-504] You want to compute the order total for your shipments. For all of your shipments, put together a report which contains the following columns.
   1. Sort it ascending on idShipment
   2. Create a new field in the output result called OrderTotal, which is the price per unit multiplied by the order quantity. You will need to calculate this field in SQL.
   3. Your output will look a little bit like the following. Your numbers may vary from this example.

SELECT s.idShipment, e.idEmployee, c.CompanyName, p.Manufacturer, s.Scantag, e.EmpFirstName, e.EmpLastName, c.CompanyName, c.idCustomer, p.ProductName, s.quantity, p.PricePerUnit, COUNT(idShipment) AS NumberOfShipments, s.quantity \* p.PricePerUnit AS OrderTotal

FROM employee AS e LEFT OUTER JOIN shipment AS s

ON e.IDemployee = s.Employee\_IDEmployee

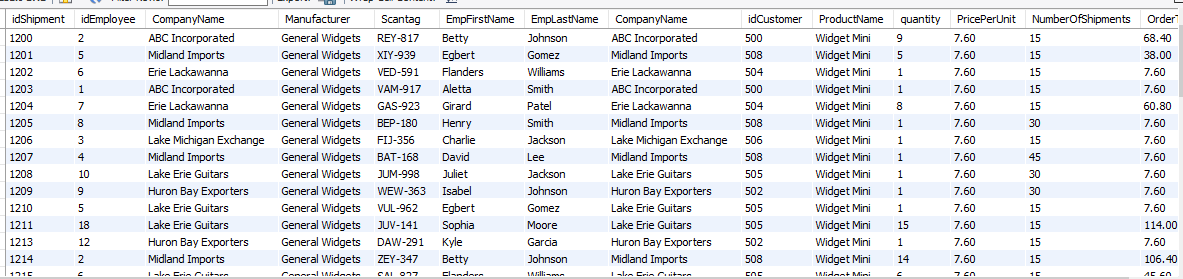
INNER JOIN customer AS c

INNER JOIN product AS p

ON s.Customer\_idCustomer = c.idCustomer

GROUP BY e.idEmployee, c.idCustomer

ORDER BY idShipment ASC, e.idEmployee;



1. [SQL Week 2 Question 200-505] You want to report on a certain group of your products. Select
   1. Only customers who have a balance of 0
   2. Only shipments which were sent by employees whose employee location is California, Maryland, or New York
   3. Only products which have had a shipment
   4. Only employees who have made a shipment
   5. Only customers who have received a shipment
   6. Compute the total quantity of shipments which meet these criteria. Here we want the number of items shipped. If you have one shipment of 10 items, and another shipment of 20 items, this number should be 30.
   7. Aggregate it by manufacturer and by product name. This means each manufacturer/product name combination appears on only one row in the table. If a manufacturer makes more than one product, it’s OK if it shows up once per product.
   8. Sort it descending by the sum of the shipment quantity, so the largest shipment quantity shows up on top. Within shipment quantities, sort it alphabetically on manufacturer and then product name.

SELECT p.Manufacturer, p.ProductName, SUM(s.Quantity)

FROM customer AS c JOIN shipment AS s

ON s.Customer\_idCustomer = c.idCustomer

JOIN employee AS e

ON e.IDemployee = s.Employee\_IDEmployee

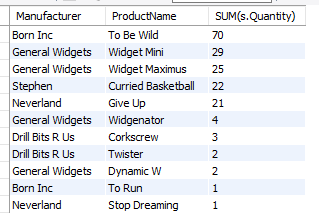
JOIN product AS p

ON s.Product\_idProduct = p.idProduct

WHERE c.Balance=0 AND e.EmpLocation IN ('CA','MD','NY')

GROUP BY p.Manufacturer, p.ProductName

ORDER BY SUM(s.Quantity) DESC, p.Manufacturer ASC, p.ProductName ASC;



1. [SQL Week 2 Question 200-506] Last one! You want to know how many times products were shipped to customers. Note you want the total number of shipments, not the number of items shipped – one shipment of 10 items and one shipment of 20 items would be two shipments, not 30 items.
   1. Include all products, even if they had no shipments. In this case, their count would be 0.
   2. Include the manufacturer and the product name
   3. Include the customer company name
   4. Sort it so the smallest count of shipments is on top. After that, sort ascending by Manufacturer name and then product name.

SELECT product.Manufacturer, product.ProductName, customer.CompanyName, COUNT(shipment.quantity)

FROM (

(product LEFT JOIN shipment

ON product.idProduct = shipment.Product\_idProduct)

LEFT JOIN customer ON shipment.Customer\_idCustomer = customer.idCustomer)

GROUP BY product.Manufacturer, product.ProductName, customer.CompanyName

ORDER BY COUNT(shipment.quantity) asc, product.Manufacturer ASC, product.ProductName asc

;

Top of the list:



Bottom of the list:

