## HW-2 SQL Queries - Instructions

**Database**: AdventureWorksLT2016.mdf

Show the SQL statement and the data results for each question below. Submit a Word document (no PDFs or other file types) with each SQL query (copied and pasted as text) followed by the image of the returned data only. The Windows snipping tool will allow you to capture the output from each query.

Example of how your work should be shown:

Q1. Select \* From BuildVersion



1. Show the company name with which James D. Kramer works.

SELECT CompanyName

from SalesLT.Customer

WHERE FirstName='James' and MiddleName='D.' and LastName='Kramer'

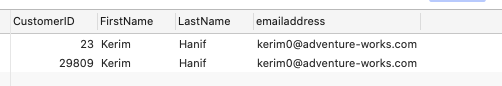


1. Show the customer ID, first name, last name and the email address of the customer whose company name is Bike World.

SELECT CustomerID,FirstName, LastName, emailaddress

from SalesLT.Customer

WHERE CompanyName='Bike World'



1. Show the company names for all customers with an address in the city of Dallas. Eliminate duplicates.

SELECT distinct c.CompanyName

from SalesLT.Customer as c

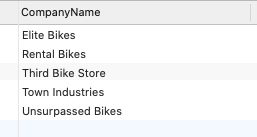
join SalesLT.CustomerAddress as ca

on c.CustomerID = ca.CustomerID

join SalesLT.Address as a

on a.AddressID = ca.AddressID

where a.City='Dallas'



1. Show all the addresses listed for the company named Modular Cycle Systems.

SELECT a.AddressID, a.AddressLine1, a.AddressLine2, a.City, a.StateProvince, a.CountryRegion, a.PostalCode

from SalesLT.Customer as c

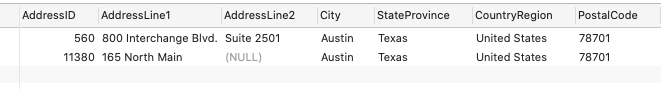
join SalesLT.CustomerAddress as ca

on c.CustomerID = ca.CustomerID

join SalesLT.Address as a

on a.AddressID = ca.AddressID

where c.CompanyName='Modular Cycle Systems'



1. Show the order quantity, the product Name and the list price of the order made by the customer whose ID is 29877.

SELECT od.OrderQty,p.name,od.UnitPrice,od.UnitPriceDiscount,(od.UnitPrice - od.UnitPriceDiscount )\* od.OrderQty as total\_price, od.LineTotal

from SalesLT.SalesOrderDetail as od

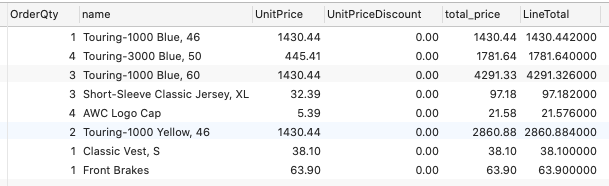
join SalesLT.product as p

on od.ProductID=p.ProductID

join SalesLT.SalesOrderHeader as so

on so.SalesOrderID = od.SalesOrderID

where so.CustomerID='29877'



1. Show the company name and total amount due of those customers with orders over $100,000, along with the order subtotal, the tax and the freight.

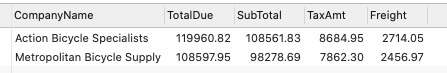
select c.CompanyName,soh.TotalDue,soh.SubTotal,soh.TaxAmt,soh.Freight

from SalesLT.SalesOrderHeader as soh

join SalesLT.Customer as c

on soh.CustomerID = c.CustomerID

where soh.TotalDue > 100000



1. List all product names that belong to the product model called ‘Racing Socks’.

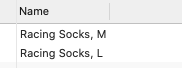
select p.Name

from SalesLT. Product as p

join SalesLT.ProductModel as pm

on p.ProductModelID=pm.ProductModelID

where pm.Name='Racing Socks'



1. How many different items that have a list price of more than $1,000 have been sold? *(Note: this is the count of different products, not the total quantity of products that have been ordered, that meet the price criteria).*

select count(t.ProductID) as over\_1000\_sold from

(select distinct sod.ProductID, p.name, p.ListPrice

from SalesLT.SalesOrderHeader as soh

join SalesLT.SalesOrderDetail as sod

on soh.SalesOrderID = sod.SalesOrderID

join SalesLT.Product as p

on sod.ProductID = p.ProductID

where p.ListPrice > 1000 ) as t



1. Find the quantity of large racing socks ('Racing Socks, L') ordered by the company named Riding Cycles.

select sod.OrderQty

from SalesLT.SalesOrderHeader as soh

join SalesLT.SalesOrderDetail as sod

on soh.SalesOrderID = sod.SalesOrderID

join SalesLT.Product as p

on sod.ProductID = p.ProductID

join SalesLT.Customer as c

on c.CustomerID = soh.CustomerID

where p.name='Racing Socks, L' and c.CompanyName='Riding Cycles'



1. List the product name and the company name for all customers who ordered any products with a product model of 'Racing Socks'.

select p.name as product\_name,c.CompanyName

from SalesLT.SalesOrderHeader as soh

join SalesLT.SalesOrderDetail as sod

on soh.SalesOrderID = sod.SalesOrderID

join SalesLT.Product as p

on sod.ProductID = p.ProductID

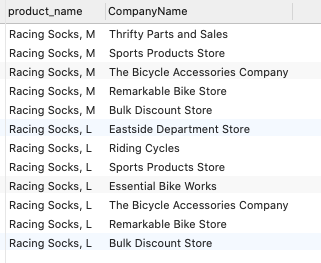
join SalesLT.Customer as c

on c.CustomerID = soh.CustomerID

join SalesLT.ProductModel as pm

on p.ProductModelID = pm.ProductModelID

where pm.Name='Racing Socks'



1. Use the SubTotal value in SaleOrderHeader to list orders from the largest to the smallest. For each order show the order ID, company name, the subtotal, and the total weight of the order.

select sod.SalesOrderID,c.CompanyName,soh.SubTotal,sum(sod.OrderQty\*Weight) as total\_weight

from SalesLT.SalesOrderHeader as soh

join SalesLT.SalesOrderDetail as sod

on soh.SalesOrderID = sod.SalesOrderID

join SalesLT.Customer as c

on c.CustomerID = soh.CustomerID

join SalesLT.Product as p

on sod.ProductID = p.ProductID

group by sod.SalesOrderID,c.CompanyName,soh.SubTotal

order by soh.SubTotal desc



1. A "Single Item Order" is a customer order in which only one item is ordered. Show the SalesOrderID for every Single Item Order.

select t1.SalesOrderID from (

select count(t.SalesOrderID) as order\_num,t.SalesOrderID

from (

select soh.SalesOrderID,soh.CustomerID,sod.ProductID

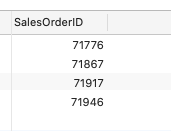
from SalesLT.SalesOrderHeader as soh

left join SalesLT.SalesOrderDetail as sod

on soh.SalesOrderID = sod.SalesOrderID) as t

group by t.CustomerID,t.SalesOrderID

having count(t.SalesOrderID) =1 ) as t1



1. A "Single Item Order" is a customer order in which only one item is ordered. Show the SalesOrderID, Product ID, and the list price for every Single Item Order.

select details.SalesOrderID,details.productId,details.unitPrice

from SalesLT.SalesOrderDetail as details

where details.SalesOrderID in (

select t1.SalesOrderID from (

select count(t.SalesOrderID) as order\_num,t.SalesOrderID

from (

select soh.SalesOrderID,soh.CustomerID,sod.ProductID

from SalesLT.SalesOrderHeader as soh

left join SalesLT.SalesOrderDetail as sod

on soh.SalesOrderID = sod.SalesOrderID) as t

group by t.CustomerID,t.SalesOrderID

having count(t.SalesOrderID) =1 ) as t1)

