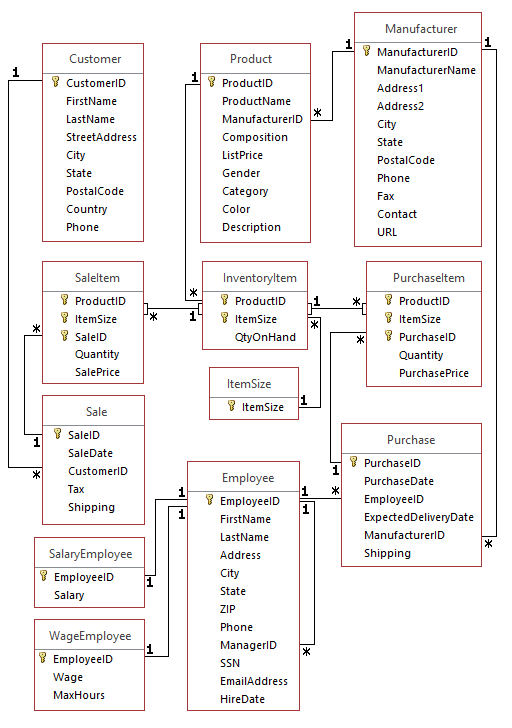
[MyEducator Home Page](https://www.myeducator.com/)

* [BMIS 441 02: Database Manag...](https://app.myeducator.com/course/activity/617eba/2701101310056439809/#)
* [Contents](https://app.myeducator.com/course/activity/617eba/2701101310056439809/#)
* Course
* Nathan Yoon

# **BMIS 441 02: Database Management [Fall 2021]**

## **SQL Assignment #12 - Outer Join Queries**

For each information request below, formulate a single SQL query to produce the required information. In each case, you should **display only the columns requested. Be sure that your queries do not produce duplicate records unless otherwise directed.**



| Points: | 100 |
| --- | --- |

Graded on Nov 16 at 14:12

Your Submission:

| Submission Score: | 100 / 100 (100.00%) |
| --- | --- |
| Grade Time: | Nov 16 at 14:12 |
| Submitted On: | Nov 14 at 09:32 |

1. What are the names of products from manufacturers in Washington state? List DISTINCT names.
2. SELECT DISTINCT P.ProductName
3. FROM Product P
4. JOIN Manufacturer M
5. ON P.ManufacturerID = M.ManufacturerID
6. WHERE M.State = 'WA';
7. Which employees made purchase orders (purchases) from the manufacturer named Timberland? Show DISTINCT first and last names only.
8. SELECT DISTINCT E.FirstName AS 'EmployeeFirstName', E.LastName AS 'EmployeeLastName'
9. FROM Employee E
10. JOIN Purchase P
11. ON P.EmployeeID = E.EmployeeID
12. JOIN PurchaseItem PI
13. ON P.PurchaseID = PI.PurchaseID
14. JOIN Manufacturer M
15. ON P.ManufacturerID = M.ManufacturerID
16. WHERE ManufacturerName = 'Timberland';
17. List Hourly (wage) Employees (first and last names), their hire dates and then the full name (first and last) of their manager. (Rename manager name columns as MgrFirstN, MgrLastN.)
18. SELECT WE.Wage AS 'HourlyWage', E.FirstName AS 'EmployeeFirstName', E.LastName AS 'EmployeeLastName', E.HireDate, M.FirstName AS 'ManagerFirstName', M.LastName AS 'ManagerLastName'
19. FROM Employee E
20. JOIN WageEmployee WE
21. ON E.EmployeeID = WE.EmployeeID
22. JOIN Employee M
23. ON E.ManagerID = M.EmployeeID;
24. List Products (ProductID and ProductName) whose list price is greater than 100 and how many sales (count SaleID in SaleItem) there are for each product (use GROUP BY). Also include those that have not been sold. Sort the output data by the number of sale items; name the column NoOfSales.
25. SELECT P.ProductID, P.ProductName, COUNT(SI.SaleID) AS 'NumberOfSales'
26. FROM SaleItem AS SI
27. LEFT JOIN Product P
28. ON SI.ProductID = P.ProductID
29. WHERE P.ListPrice > 100
30. GROUP BY P.ProductID, P.ProductName
31. ORDER BY 'NumberOfSales';
32. List all manufacturers (ID and name) and the number of purchases made from that manufacturer. Include manufacturers where no purchases have been made. Sort the output data by the number of purchases. Name that column NoOfPurchases.
33. SELECT M.ManufacturerID, M.manufacturerName, COUNT(P.PurchaseID) AS 'NumberOfPurchases'
34. FROM Manufacturer M
35. LEFT JOIN Purchase P
36. ON M.ManufacturerID = P.ManufacturerID
37. GROUP BY M.ManufacturerID, M.ManufacturerName
38. ORDER BY 'NumberOfPurchases';
39. How many times has each product in the "flats" category been purchased? Include those items that have never been purchased. Show ProductName and ProductID. Sort the data by the number of purchases. Name the column of counted purchases NoOfPurchases.
40. SELECT P.ProductID, P.ProductName, COUNT(PurchaseID) AS 'NumberOfPurchases'
41. FROM Product P
42. LEFT JOIN PurchaseItem PI
43. ON P.ProductID = PI.ProductID
44. WHERE Category = 'Flats'
45. GROUP BY P.ProductID, P.ProductName
46. ORDER BY 'NumberOfPurchases';
47. Show Customers (CustomerID, firstname, lastname) from Oregon(OR) and the number of sales for each customer. Include customers with no sales. Name the Count column "NoOfSales". Sort the data ascending by the number of sales.
48. SELECT C.CustomerID, FirstName, LastName, COUNT(SaleID) AS 'NumberOfSales'
49. FROM Customer C
50. LEFT JOIN Sale S
51. ON C.CustomerID = S.CustomerID
52. WHERE State = 'OR'
53. GROUP BY C.CustomerID, FirstName, LastName
54. ORDER BY 'NumberOfSales';
55. Show customers (CustomerID, FirstName, LastName, city) from Oregon and sales (SaleID, SaleDate) to those customers. Include customers with no sales. Sort the data by CustomerID.
56. SELECT C.CustomerID, FirstName, LastName, City, SaleID, SaleDate
57. FROM Customer C
58. LEFT JOIN Sale S
59. ON C.CustomerID = S.CustomerID
60. WHERE State = 'OR'
61. GROUP BY C.CustomerID, FirstName, LastName, City, SaleID, SaleDate
62. ORDER BY C.CustomerID;
63. List Inventory Items (show ProductID and ItemSize) and the number (count the SaleID) of Sale Items for each inventory item. Name the count NoOfSales. Include Inventory Items that have not been sold. Also only include inventory items that have a QtyOnHand greater than 40. Sort the output by NoOfSales. (Hint: To join tables use all the common key fields.)
64. SELECT II.ProductID, II.ItemSize, COUNT(SaleID) AS 'NumberOfSales'
65. FROM InventoryItem II
66. LEFT JOIN SaleItem SI
67. ON II.ProductID = SI.ProductID
68. WHERE QtyOnHand > 40
69. GROUP BY II.ProductID, II.ItemSize
70. ORDER BY 'NumberOfSales';
71. List all employees (EmployeeID, FirstName, LastName) from Employee table and all December Purchases (PurchaseDate, ManufacturerID) from DecPurchase table. EmployeeID is the key field in both tables. Consider full outer join.
72. SELECT E.EmployeeID, E.FirstName, E.LastName, PurchaseDate, ManufacturerID
73. FROM Employee E
74. LEFT JOIN DecPurchase D
75. ON E.EmployeeID = D.EmployeeID
76. UNION
77. SELECT E.EmployeeID, E.FirstName, E.LastName, PurchaseDate, ManufacturerID
78. FROM Employee E
79. RIGHT JOIN DecPurchase D
80. ON E.EmployeeID = D.EmployeeID;

Copyright © 2022 MyEducator. All rights reserved.

[Need Help?](https://myeducator.freshdesk.com/support/home)