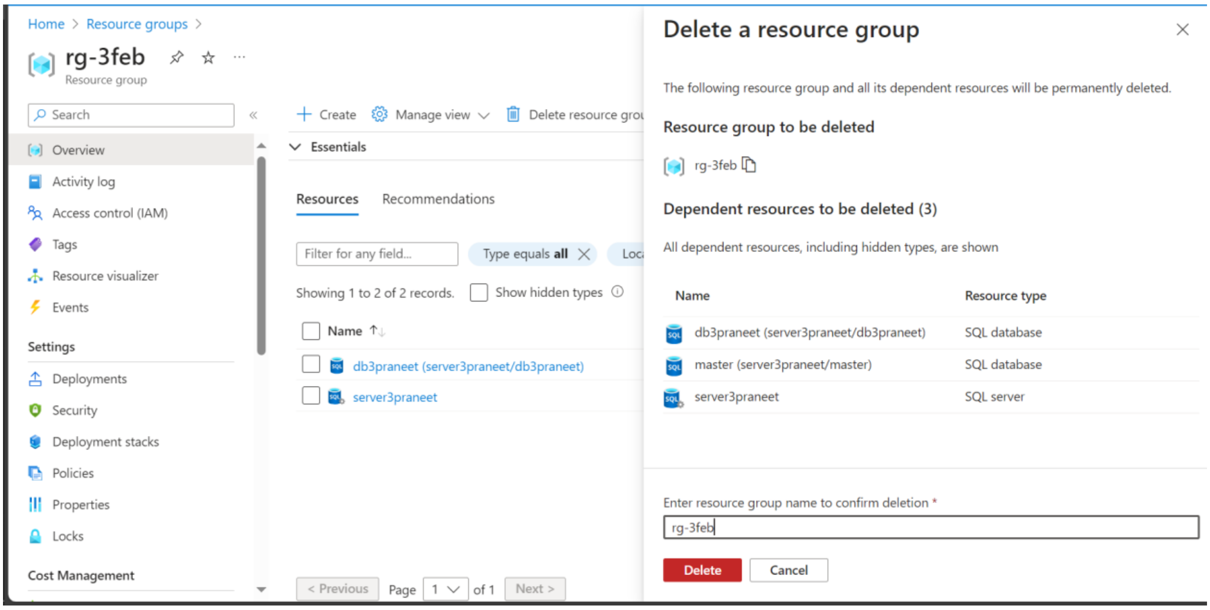


## Delete a resource group

The following resource group and all its dependent resources will be permanently deleted.

**Resource group to be deleted**

rg-3feb

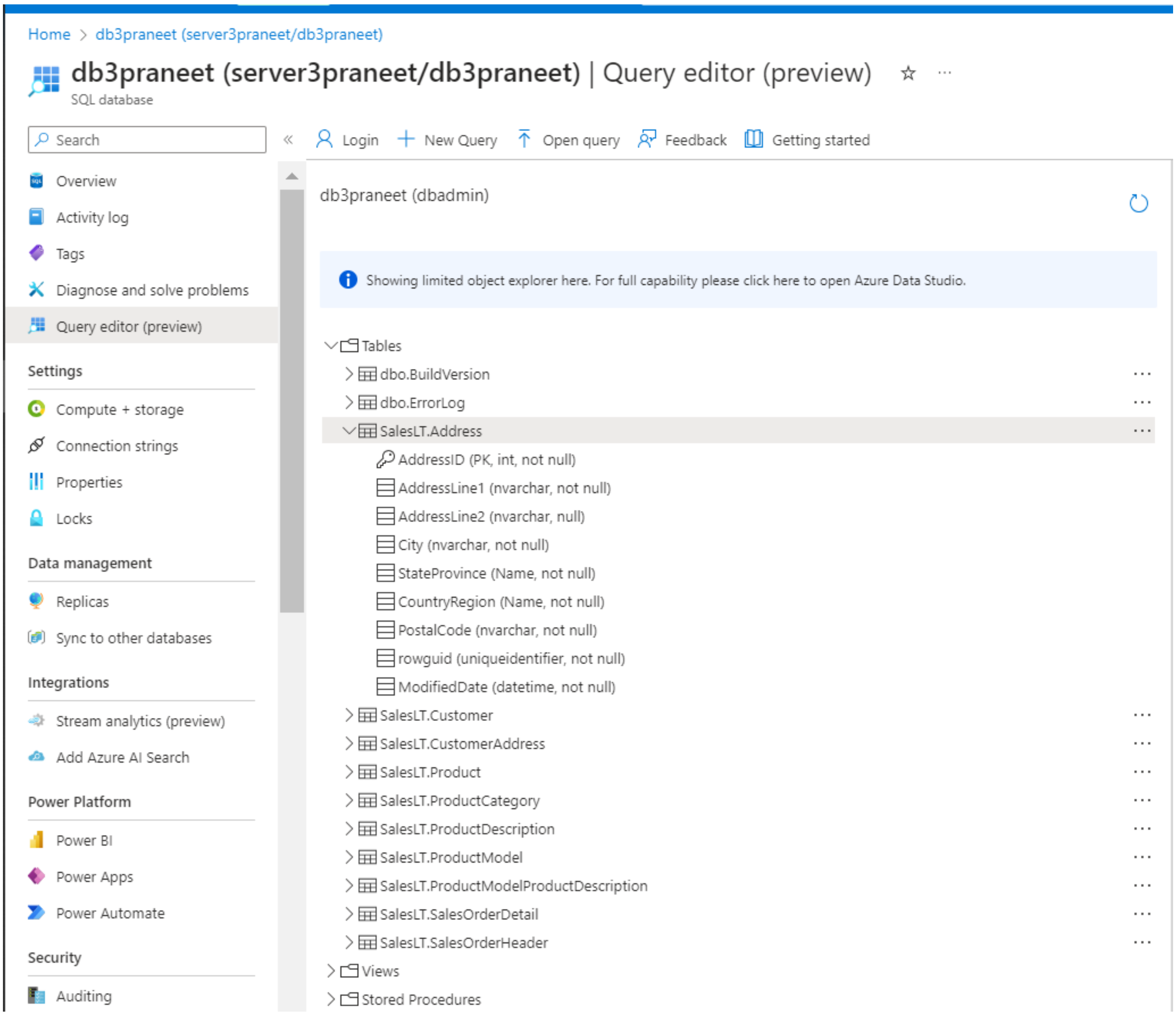
**Dependent resources to be deleted (3)**

All dependent resources, including hidden types, are shown

| Name | Resource type |
|---|---|
| db3praneet (server3praneet/db3praneet) | SQL database |
| master (server3praneet/master) | SQL database |
| server3praneet | SQL server |

Enter resource group name to confirm deletion *

rg-3feb



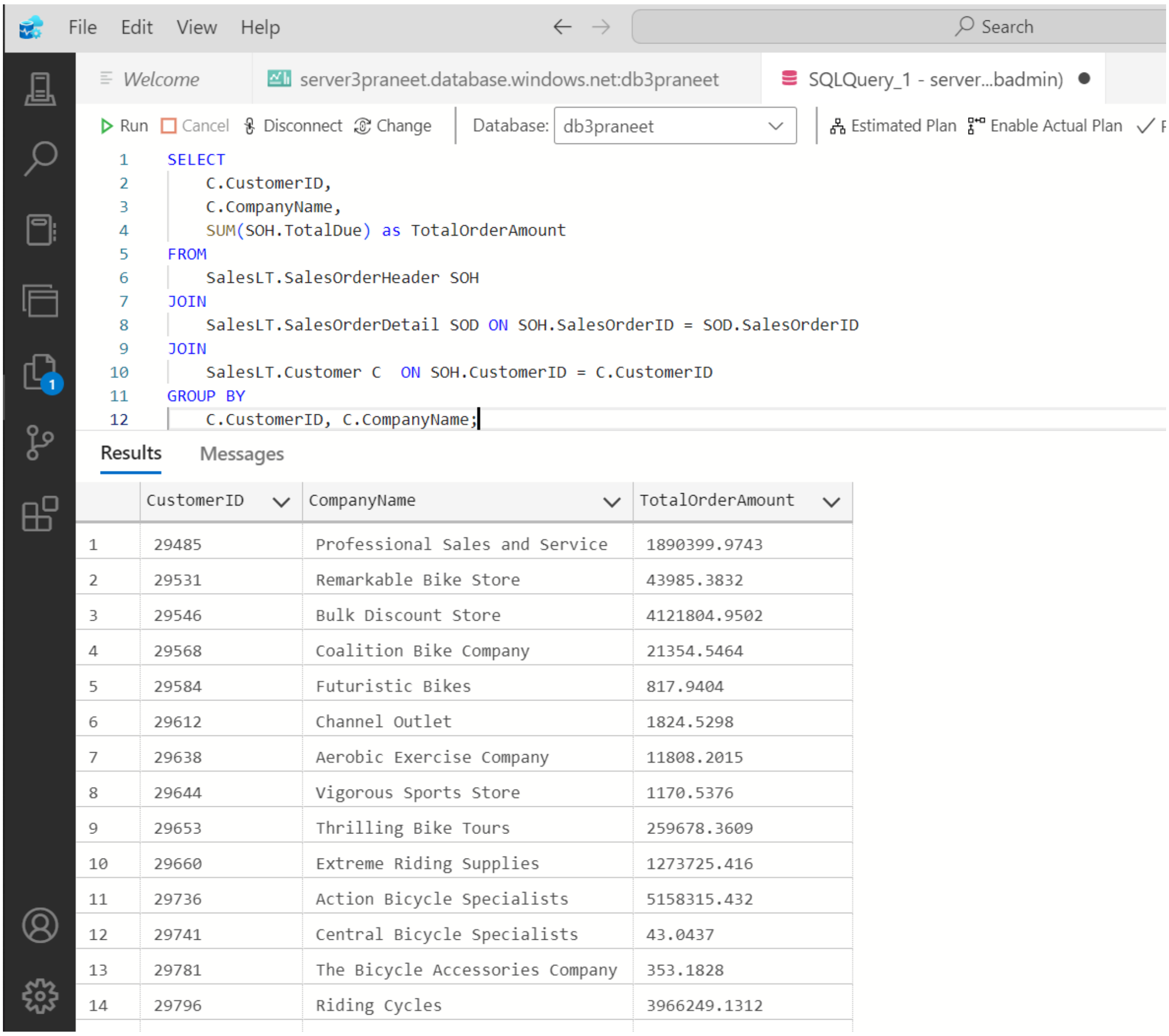
db3praneet (dbadmin)

Showing limited object explorer here. For full capability please click here to open Azure Data Studio.

```
Tables
  dbo.BuildVersion
  dbo.ErrorLog
  SalesLT.Address
    AddressID (PK, int, not null)
    AddressLine1 (nvarchar, not null)
    AddressLine2 (nvarchar, null)
    City (nvarchar, not null)
    StateProvince (Name, not null)
    CountryRegion (Name, not null)
    PostalCode (nvarchar, not null)
    rowguid (uniqueidentifier, not null)
    ModifiedDate (datetime, not null)
  SalesLT.Customer
  SalesLT.CustomerAddress
  SalesLT.Product
  SalesLT.ProductCategory
  SalesLT.ProductDescription
  SalesLT.ProductModel
  SalesLT.ProductModelProductDescription
  SalesLT.SalesOrderDetail
  SalesLT.SalesOrderHeader
Views
Stored Procedures
```

1. **Retrieve a list of customers along with their total order amounts.**

```
SELECT
    C.CustomerID,
    C.CompanyName,
    SUM(SOH.TotalDue) as TotalOrderAmount
FROM
    SalesLT.SalesOrderHeader SOH
JOIN
    SalesLT.SalesOrderDetail SOD ON SOH.SalesOrderID = SOD.SalesOrderID
JOIN
    SalesLT.Customer C  ON SOH.CustomerID = C.CustomerID
GROUP BY
    C.CustomerID, C.CompanyName;
```

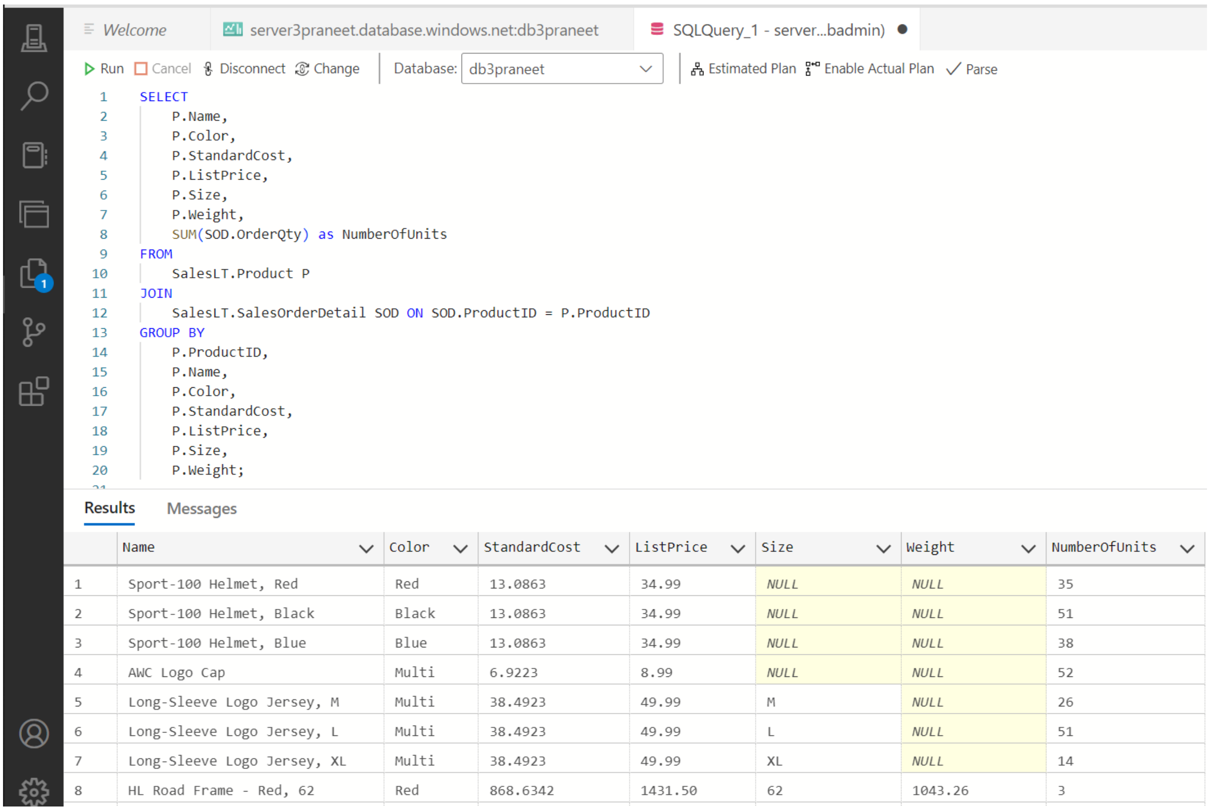


Database: db3praneet

```
SELECT
    C.CustomerID,
    C.CompanyName,
    SUM(SOH.TotalDue) as TotalOrderAmount
FROM
    SalesLT.SalesOrderHeader SOH
JOIN
    SalesLT.SalesOrderDetail SOD ON SOH.SalesOrderID = SOD.SalesOrderID
JOIN
    SalesLT.Customer C  ON SOH.CustomerID = C.CustomerID
GROUP BY
    C.CustomerID, C.CompanyName;
```

| | CustomerID | CompanyName | TotalOrderAmount |
|---|---|---|---|
| 1 | 29485 | Professional Sales and Service | 1890399.9743 |
| 2 | 29531 | Remarkable Bike Store | 43985.3832 |
| 3 | 29546 | Bulk Discount Store | 4121804.9502 |
| 4 | 29568 | Coalition Bike Company | 21354.5464 |
| 5 | 29584 | Futuristic Bikes | 817.9404 |
| 6 | 29612 | Channel Outlet | 1824.5298 |
| 7 | 29638 | Aerobic Exercise Company | 11808.2015 |
| 8 | 29644 | Vigorous Sports Store | 1170.5376 |
| 9 | 29653 | Thrilling Bike Tours | 259678.3609 |
| 10 | 29660 | Extreme Riding Supplies | 1273725.416 |
| 11 | 29736 | Action Bicycle Specialists | 5158315.432 |
| 12 | 29741 | Central Bicycle Specialists | 43.0437 |
| 13 | 29781 | The Bicycle Accessories Company | 353.1828 |
| 14 | 29796 | Riding Cycles | 3966249.1312 |

2. **Display product information along with the number of units sold for each product.**

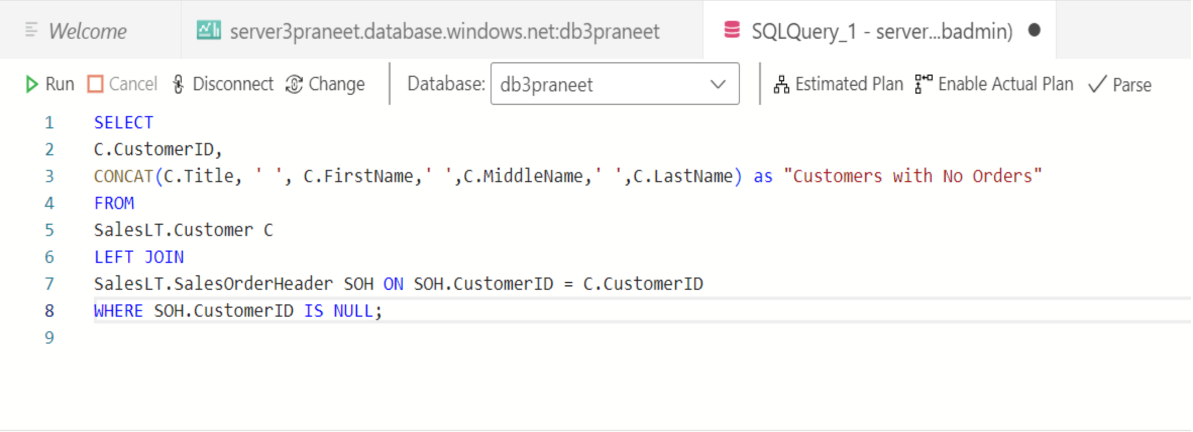


```sql
SELECT
    P.Name,
    P.Color,
    P.StandardCost,
    P.ListPrice,
    P.Size,
    P.Weight,
    SUM(SOD.OrderQty) as NumberOfUnits
FROM
    SalesLT.Product P
JOIN
    SalesLT.SalesOrderDetail SOD ON SOD.ProductID = P.ProductID
GROUP BY
    P.ProductID,
    P.Name,
    P.Color,
    P.StandardCost,
    P.ListPrice,
    P.Size,
    P.Weight;
```

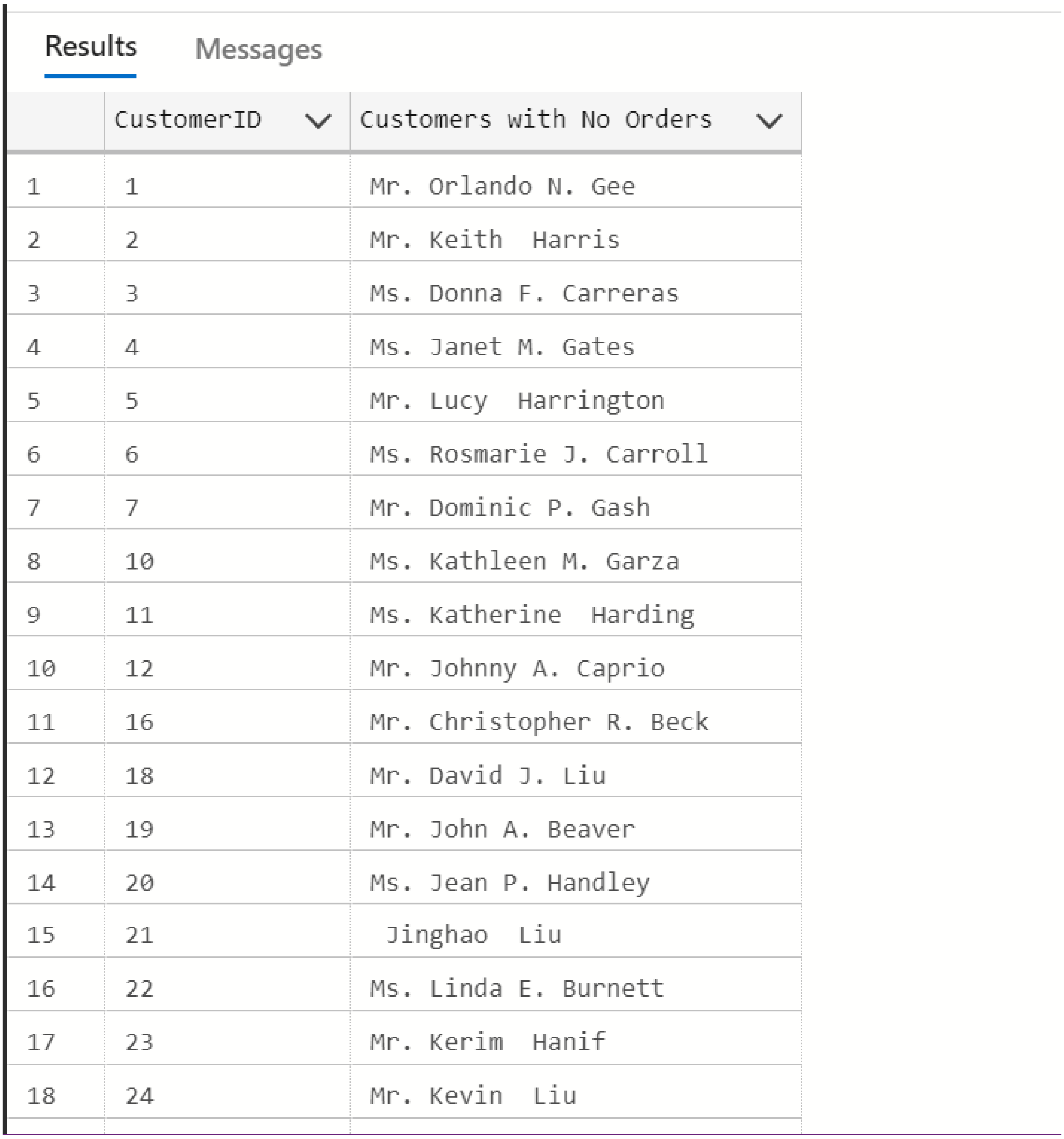
|  | Name | Color | StandardCost | ListPrice | Size | Weight | NumberOfUnits |
|---|---|---|---|---|---|---|---|
| 1 | Sport-100 Helmet, Red | Red | 13.0863 | 34.99 | NULL | NULL | 35 |
| 2 | Sport-100 Helmet, Black | Black | 13.0863 | 34.99 | NULL | NULL | 51 |
| 3 | Sport-100 Helmet, Blue | Blue | 13.0863 | 34.99 | NULL | NULL | 38 |
| 4 | AWC Logo Cap | Multi | 6.9223 | 8.99 | NULL | NULL | 52 |
| 5 | Long-Sleeve Logo Jersey, M | Multi | 38.4923 | 49.99 | M | NULL | 26 |
| 6 | Long-Sleeve Logo Jersey, L | Multi | 38.4923 | 49.99 | L | NULL | 51 |
| 7 | Long-Sleeve Logo Jersey, XL | Multi | 38.4923 | 49.99 | XL | NULL | 14 |
| 8 | HL Road Frame - Red, 62 | Red | 868.6342 | 1431.50 | 62 | 1043.26 | 3 |

3. **Find employees who have the same manager. (DATA INSUFFICIENT)**

4. **List all customers who have never placed an order.**

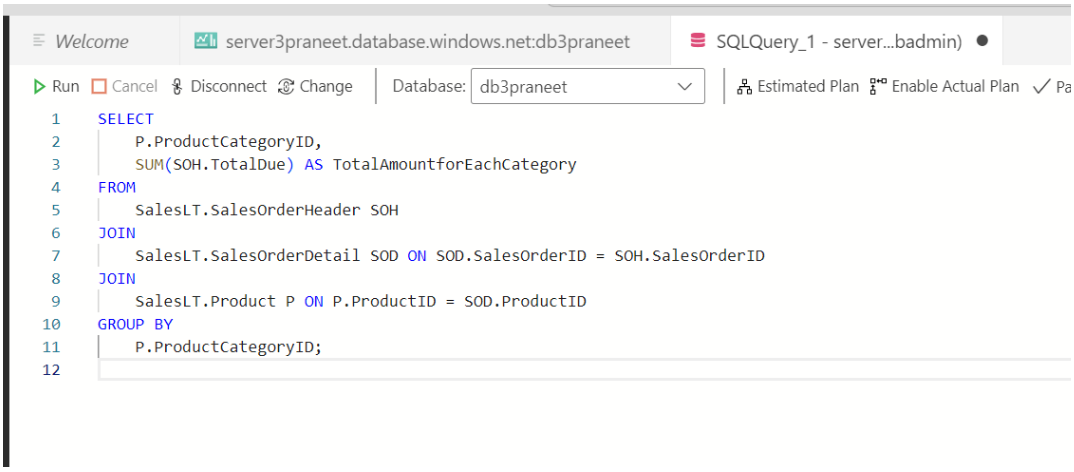


```sql
SELECT
C.CustomerID,
CONCAT(C.Title, ' ', C.FirstName,' ',C.MiddleName,' ',C.LastName) as "Customers with No Orders"
FROM
SalesLT.Customer C
LEFT JOIN
SalesLT.SalesOrderHeader SOH ON SOH.CustomerID = C.CustomerID
WHERE SOH.CustomerID IS NULL;
```

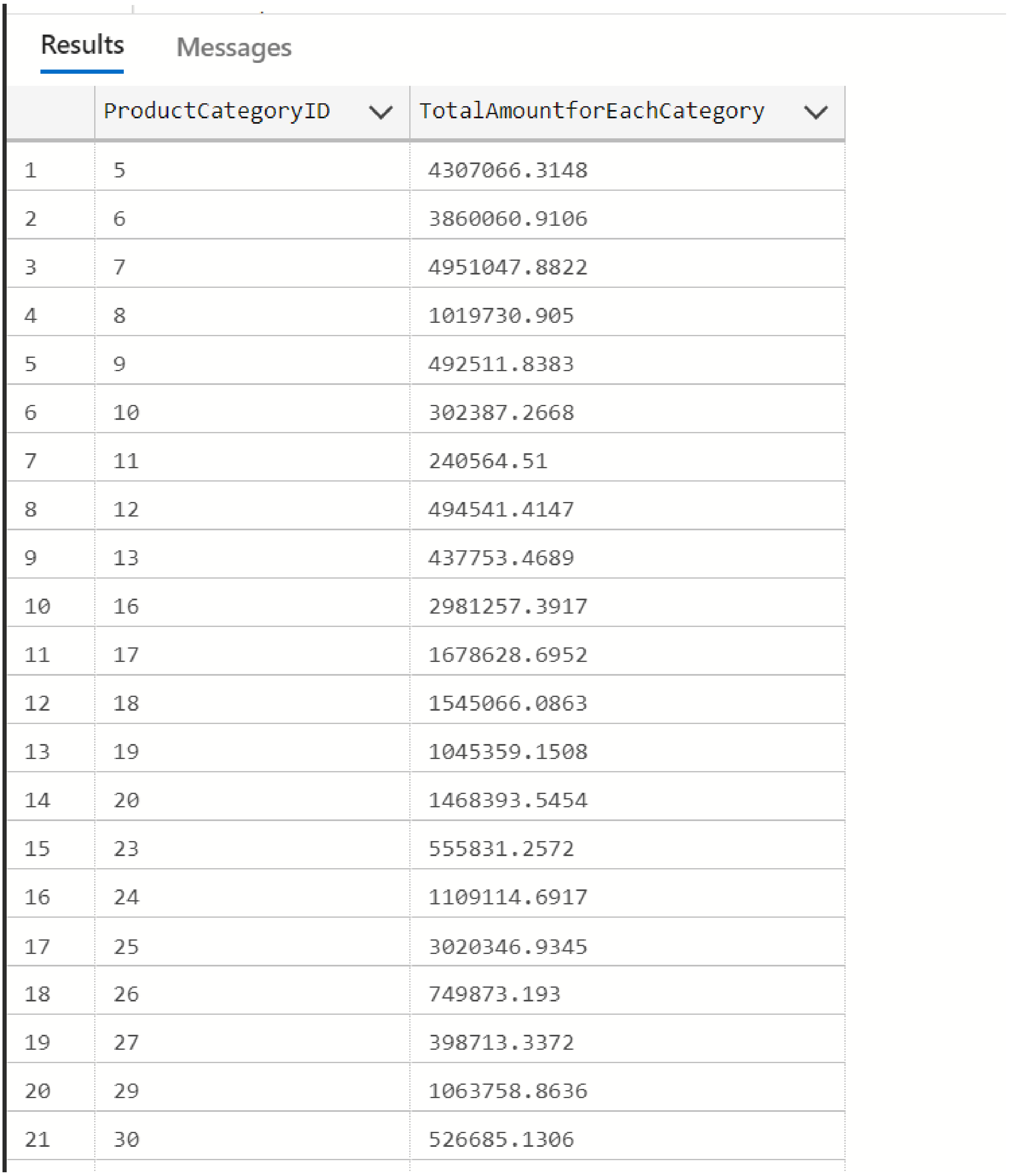


|  | CustomerID | Customers with No Orders |
|---|---|---|
| 1 | 1 | Mr. Orlando N. Gee |
| 2 | 2 | Mr. Keith  Harris |
| 3 | 3 | Ms. Donna F. Carreras |
| 4 | 4 | Ms. Janet M. Gates |
| 5 | 5 | Mr. Lucy  Harrington |
| 6 | 6 | Ms. Rosmarie J. Carroll |
| 7 | 7 | Mr. Dominic P. Gash |
| 8 | 10 | Ms. Kathleen M. Garza |
| 9 | 11 | Ms. Katherine  Harding |
| 10 | 12 | Mr. Johnny A. Caprio |
| 11 | 16 | Mr. Christopher R. Beck |
| 12 | 18 | Mr. David J. Liu |
| 13 | 19 | Mr. John A. Beaver |
| 14 | 20 | Ms. Jean P. Handley |
| 15 | 21 | Jinghao  Liu |
| 16 | 22 | Ms. Linda E. Burnett |
| 17 | 23 | Mr. Kerim  Hanif |
| 18 | 24 | Mr. Kevin  Liu |

5. **Retrieve the total sales amount for each product category.**



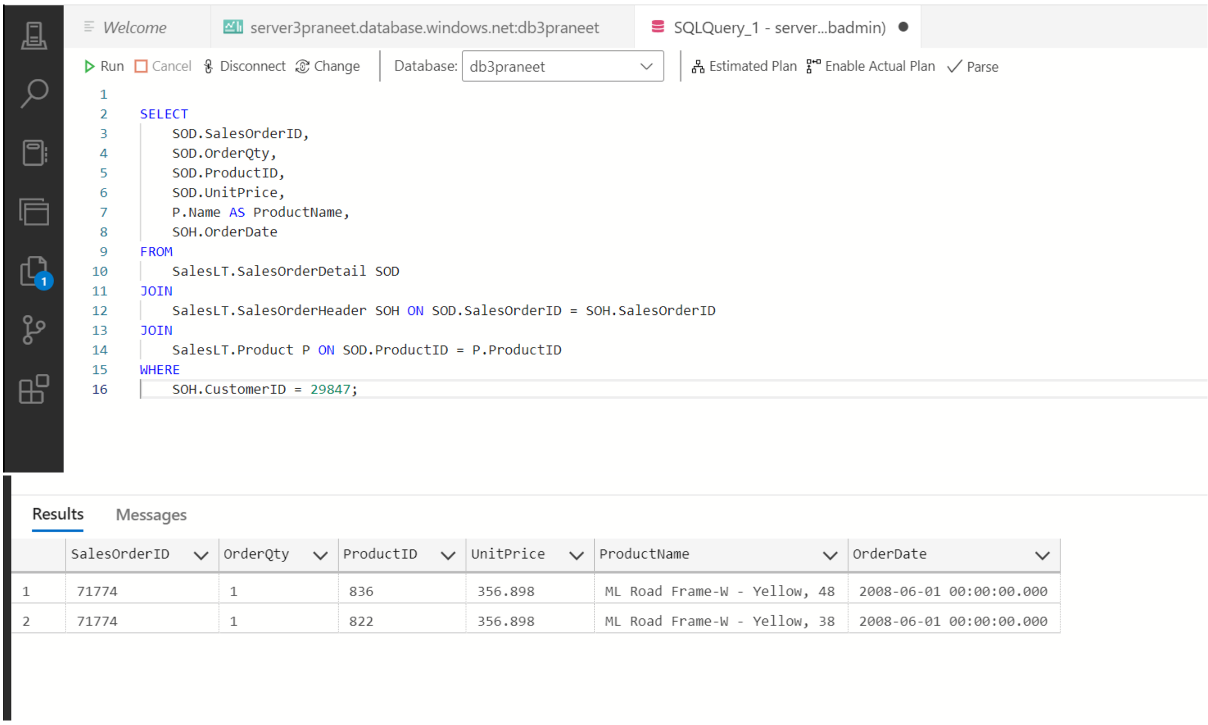
```sql
SELECT
    P.ProductCategoryID,
    SUM(SOH.TotalDue) AS TotalAmountforEachCategory
FROM
    SalesLT.SalesOrderHeader SOH
JOIN
    SalesLT.SalesOrderDetail SOD ON SOD.SalesOrderID = SOH.SalesOrderID
JOIN
    SalesLT.Product P ON P.ProductID = SOD.ProductID
GROUP BY
    P.ProductCategoryID;
```



| | ProductCategoryID | TotalAmountforEachCategory |
|---|---|---|
| 1 | 5 | 4307066.3148 |
| 2 | 6 | 3860060.9106 |
| 3 | 7 | 4951047.8822 |
| 4 | 8 | 1019730.905 |
| 5 | 9 | 492511.8383 |
| 6 | 10 | 302387.2668 |
| 7 | 11 | 240564.51 |
| 8 | 12 | 494541.4147 |
| 9 | 13 | 437753.4689 |
| 10 | 16 | 2981257.3917 |
| 11 | 17 | 1678628.6952 |
| 12 | 18 | 1545066.0863 |
| 13 | 19 | 1045359.1508 |
| 14 | 20 | 1468393.5454 |
| 15 | 23 | 555831.2572 |
| 16 | 24 | 1109114.6917 |
| 17 | 25 | 3020346.9345 |
| 18 | 26 | 749873.193 |
| 19 | 27 | 398713.3372 |
| 20 | 29 | 1063758.8636 |
| 21 | 30 | 526685.1306 |

6. **Display the names of employees and their direct managers. (DATA INSUFFICIENT).**

7. **Show the order details with product names for a specific customer.**



Database: db3praneet

```sql

SELECT
    SOD.SalesOrderID,
    SOD.OrderQty,
    SOD.ProductID,
    SOD.UnitPrice,
    P.Name AS ProductName,
    SOH.OrderDate
FROM
    SalesLT.SalesOrderDetail SOD
JOIN
    SalesLT.SalesOrderHeader SOH ON SOD.SalesOrderID = SOH.SalesOrderID
JOIN
    SalesLT.Product P ON SOD.ProductID = P.ProductID
WHERE
    SOH.CustomerID = 29847;
```

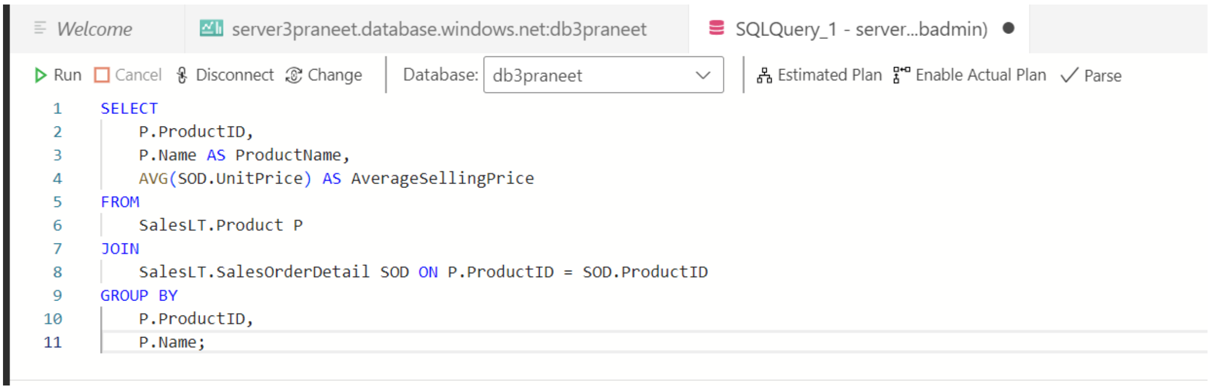
| | SalesOrderID | OrderQty | ProductID | UnitPrice | ProductName | OrderDate |
|---|---|---|---|---|---|---|
| 1 | 71774 | 1 | 836 | 356.898 | ML Road Frame-W - Yellow, 48 | 2008-06-01 00:00:00.000 |
| 2 | 71774 | 1 | 822 | 356.898 | ML Road Frame-W - Yellow, 38 | 2008-06-01 00:00:00.000 |

8. **List customers who have made purchases in the last 30 days.**

```sql
SELECT DISTINCT
    C.CustomerID,
    C.FirstName,
    C.LastName
FROM
    SalesLT.Customer C
JOIN
    SalesLT.SalesOrderHeader SOH ON C.CustomerID = SOH.CustomerID
WHERE
    SOH.OrderDate >= DATEADD(day, -30, GETDATE());
```

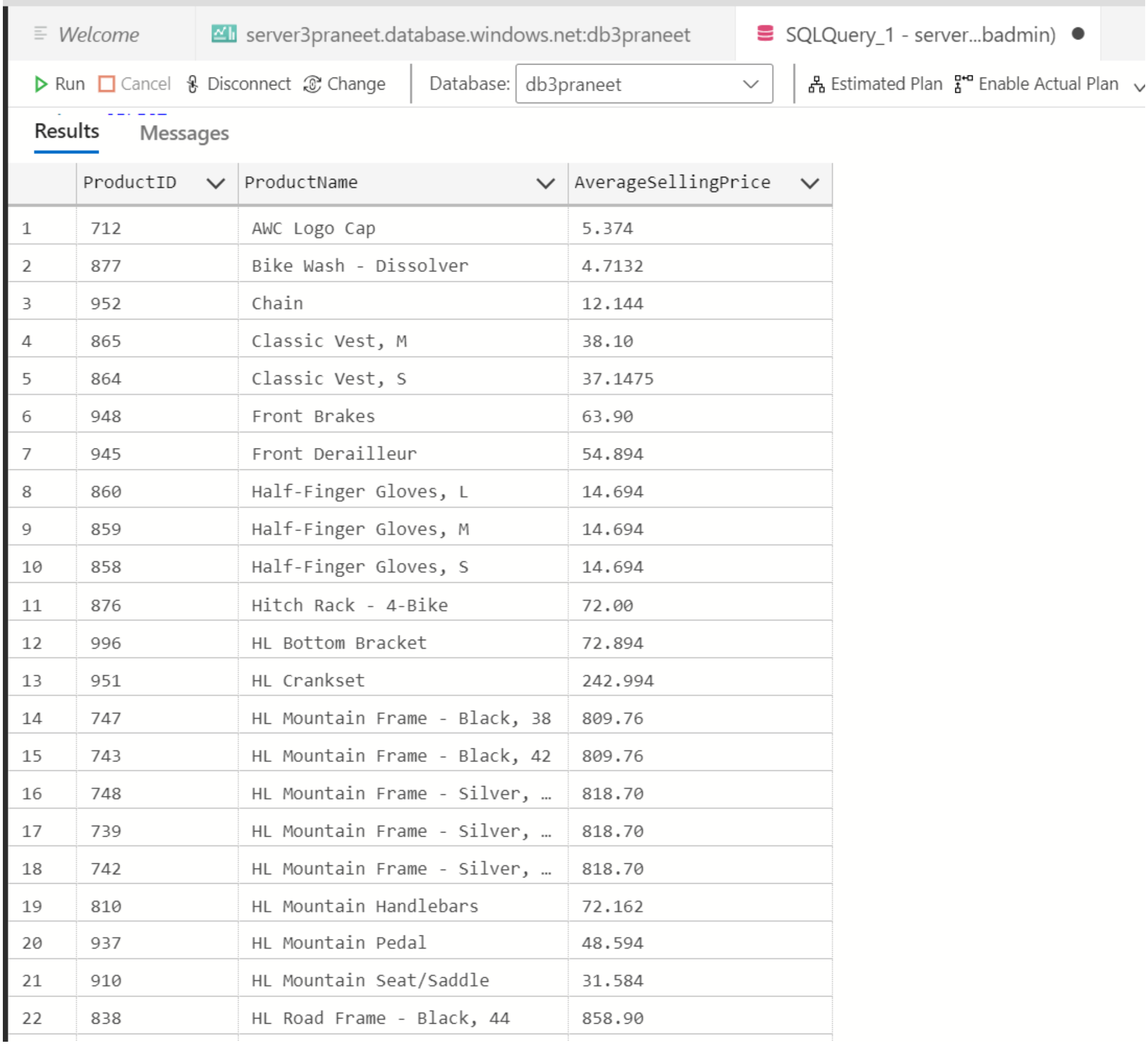
9. **Find employees who do not have any direct reports. (DATA INSUFFICIENT)**

10. **Retrieve all products along with their average selling prices.**


Welcome
server3praneet.database.windows.net:db3praneet
SQLQuery_1 - server...badmin) ●
Run
Cancel
Disconnect
Change
Estimated Plan
Enable Actual Plan
Parse

Database: db3praneet

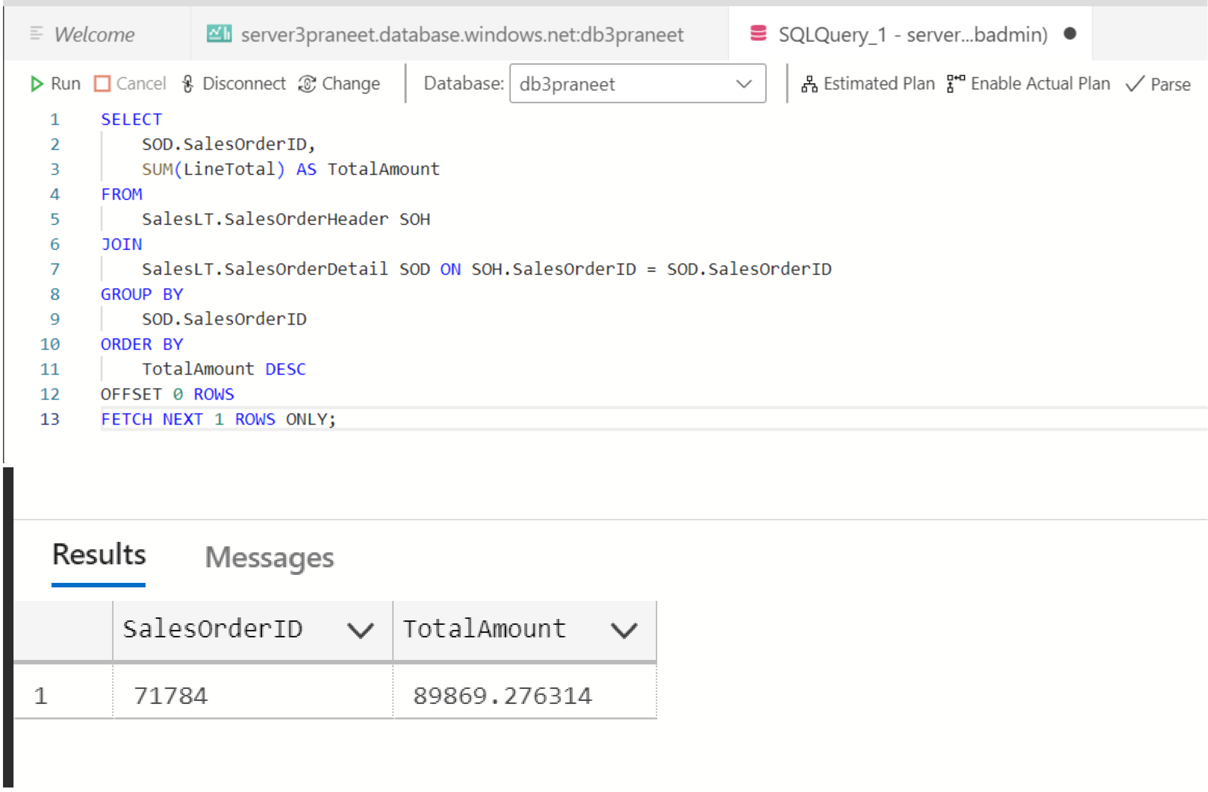
```sql
SELECT
    P.ProductID,
    P.Name AS ProductName,
    AVG(SOD.UnitPrice) AS AverageSellingPrice
FROM
    SalesLT.Product P
JOIN
    SalesLT.SalesOrderDetail SOD ON P.ProductID = SOD.ProductID
GROUP BY
    P.ProductID,
    P.Name;
```



| | ProductID | ProductName | AverageSellingPrice |
|---|---|---|---|
| 1 | 712 | AWC Logo Cap | 5.374 |
| 2 | 877 | Bike Wash - Dissolver | 4.7132 |
| 3 | 952 | Chain | 12.144 |
| 4 | 865 | Classic Vest, M | 38.10 |
| 5 | 864 | Classic Vest, S | 37.1475 |
| 6 | 948 | Front Brakes | 63.90 |
| 7 | 945 | Front Derailleur | 54.894 |
| 8 | 860 | Half-Finger Gloves, L | 14.694 |
| 9 | 859 | Half-Finger Gloves, M | 14.694 |
| 10 | 858 | Half-Finger Gloves, S | 14.694 |
| 11 | 876 | Hitch Rack - 4-Bike | 72.00 |
| 12 | 996 | HL Bottom Bracket | 72.894 |
| 13 | 951 | HL Crankset | 242.994 |
| 14 | 747 | HL Mountain Frame - Black, 38 | 809.76 |
| 15 | 743 | HL Mountain Frame - Black, 42 | 809.76 |
| 16 | 748 | HL Mountain Frame - Silver, … | 818.70 |
| 17 | 739 | HL Mountain Frame - Silver, … | 818.70 |
| 18 | 742 | HL Mountain Frame - Silver, … | 818.70 |
| 19 | 810 | HL Mountain Handlebars | 72.162 |
| 20 | 937 | HL Mountain Pedal | 48.594 |
| 21 | 910 | HL Mountain Seat/Saddle | 31.584 |
| 22 | 838 | HL Road Frame - Black, 44 | 858.90 |

# SUBQUERIES:

11. **Find the order with the highest total amount.**

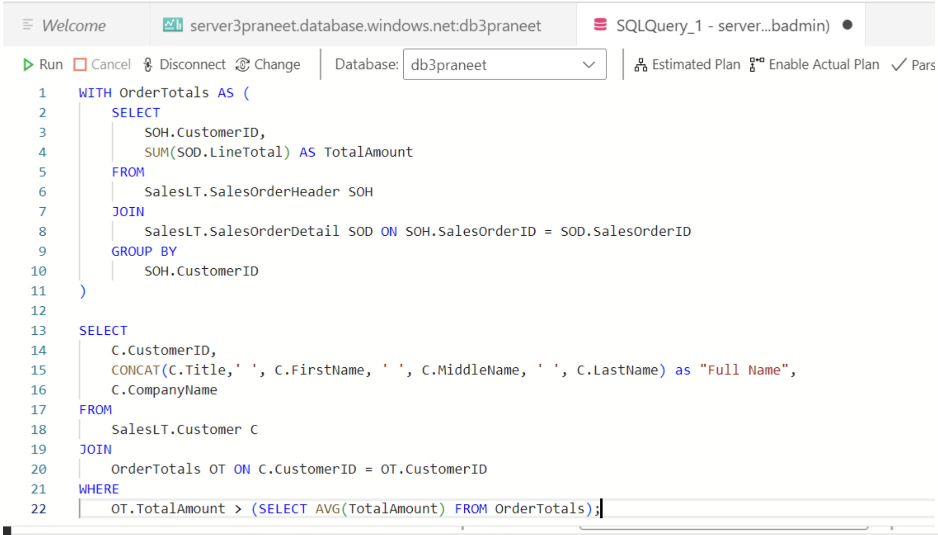


```sql
SELECT
    SOD.SalesOrderID,
    SUM(LineTotal) AS TotalAmount
FROM
    SalesLT.SalesOrderHeader SOH
JOIN
    SalesLT.SalesOrderDetail SOD ON SOH.SalesOrderID = SOD.SalesOrderID
GROUP BY
    SOD.SalesOrderID
ORDER BY
    TotalAmount DESC
OFFSET 0 ROWS
FETCH NEXT 1 ROWS ONLY;
```

| | SalesOrderID | TotalAmount |
|---|---|---|
| 1 | 71784 | 89869.276314 |

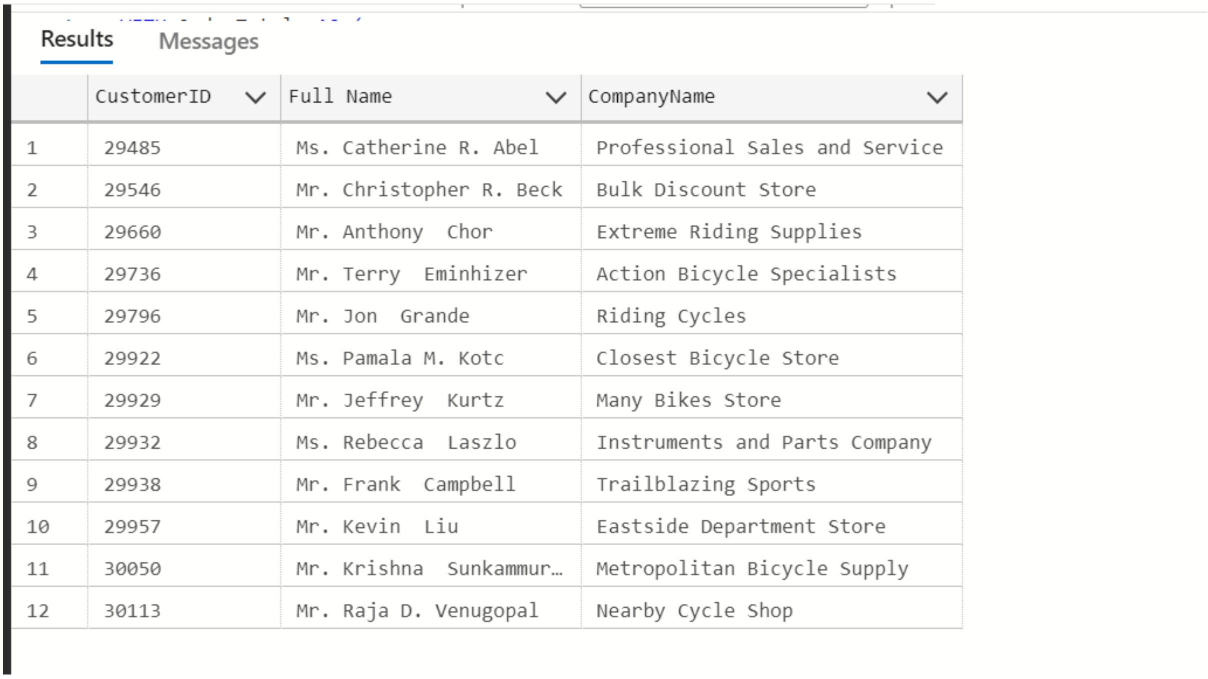
12. **Display customers who have placed orders with a total amount greater than the average.**

```
WITH OrderTotals AS (
    SELECT
        SOH.CustomerID,
        SUM(SOD.LineTotal) AS TotalAmount
    FROM
        SalesLT.SalesOrderHeader SOH
    JOIN
        SalesLT.SalesOrderDetail SOD ON SOH.SalesOrderID = SOD.SalesOrderID
    GROUP BY
        SOH.CustomerID
)

SELECT
    C.CustomerID,
    CONCAT(C.Title,' ', C.FirstName, ' ', C.MiddleName, ' ', C.LastName) as "Full Name",
    C.CompanyName
FROM
    SalesLT.Customer C
JOIN
    OrderTotals OT ON C.CustomerID = OT.CustomerID
WHERE
    OT.TotalAmount > (SELECT AVG(TotalAmount) FROM OrderTotals);
```


Welcome
server3praneet.database.windows.net:db3praneet
SQLQuery_1 - server...badmin)
Run
Cancel
Disconnect
Change
Estimated Plan
Enable Actual Plan
Pars

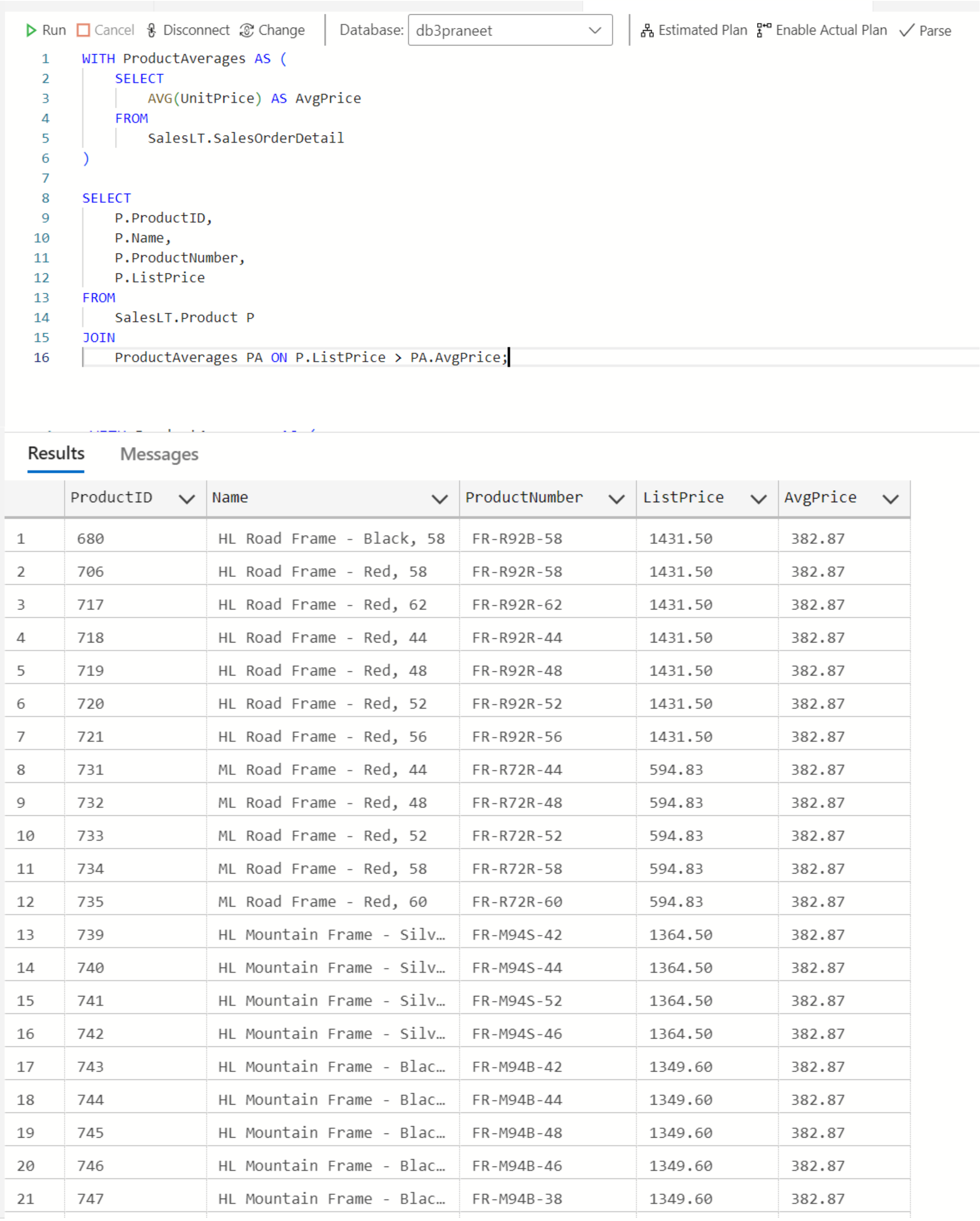
Database: db3praneet

```
WITH OrderTotals AS (
    SELECT
        SOH.CustomerID,
        SUM(SOD.LineTotal) AS TotalAmount
    FROM
        SalesLT.SalesOrderHeader SOH
    JOIN
        SalesLT.SalesOrderDetail SOD ON SOH.SalesOrderID = SOD.SalesOrderID
    GROUP BY
        SOH.CustomerID
)

SELECT
    C.CustomerID,
    CONCAT(C.Title,' ', C.FirstName, ' ', C.MiddleName, ' ', C.LastName) as "Full Name",
    C.CompanyName
FROM
    SalesLT.Customer C
JOIN
    OrderTotals OT ON C.CustomerID = OT.CustomerID
WHERE
    OT.TotalAmount > (SELECT AVG(TotalAmount) FROM OrderTotals);
```



| | CustomerID | Full Name | CompanyName |
|---|---|---|---|
| 1 | 29485 | Ms. Catherine R. Abel | Professional Sales and Service |
| 2 | 29546 | Mr. Christopher R. Beck | Bulk Discount Store |
| 3 | 29660 | Mr. Anthony  Chor | Extreme Riding Supplies |
| 4 | 29736 | Mr. Terry  Eminhizer | Action Bicycle Specialists |
| 5 | 29796 | Mr. Jon  Grande | Riding Cycles |
| 6 | 29922 | Ms. Pamala M. Kotc | Closest Bicycle Store |
| 7 | 29929 | Mr. Jeffrey  Kurtz | Many Bikes Store |
| 8 | 29932 | Ms. Rebecca  Laszlo | Instruments and Parts Company |
| 9 | 29938 | Mr. Frank  Campbell | Trailblazing Sports |
| 10 | 29957 | Mr. Kevin  Liu | Eastside Department Store |
| 11 | 30050 | Mr. Krishna  Sunkammur… | Metropolitan Bicycle Supply |
| 12 | 30113 | Mr. Raja D. Venugopal | Nearby Cycle Shop |

13. **List products with prices higher than the average product price.**

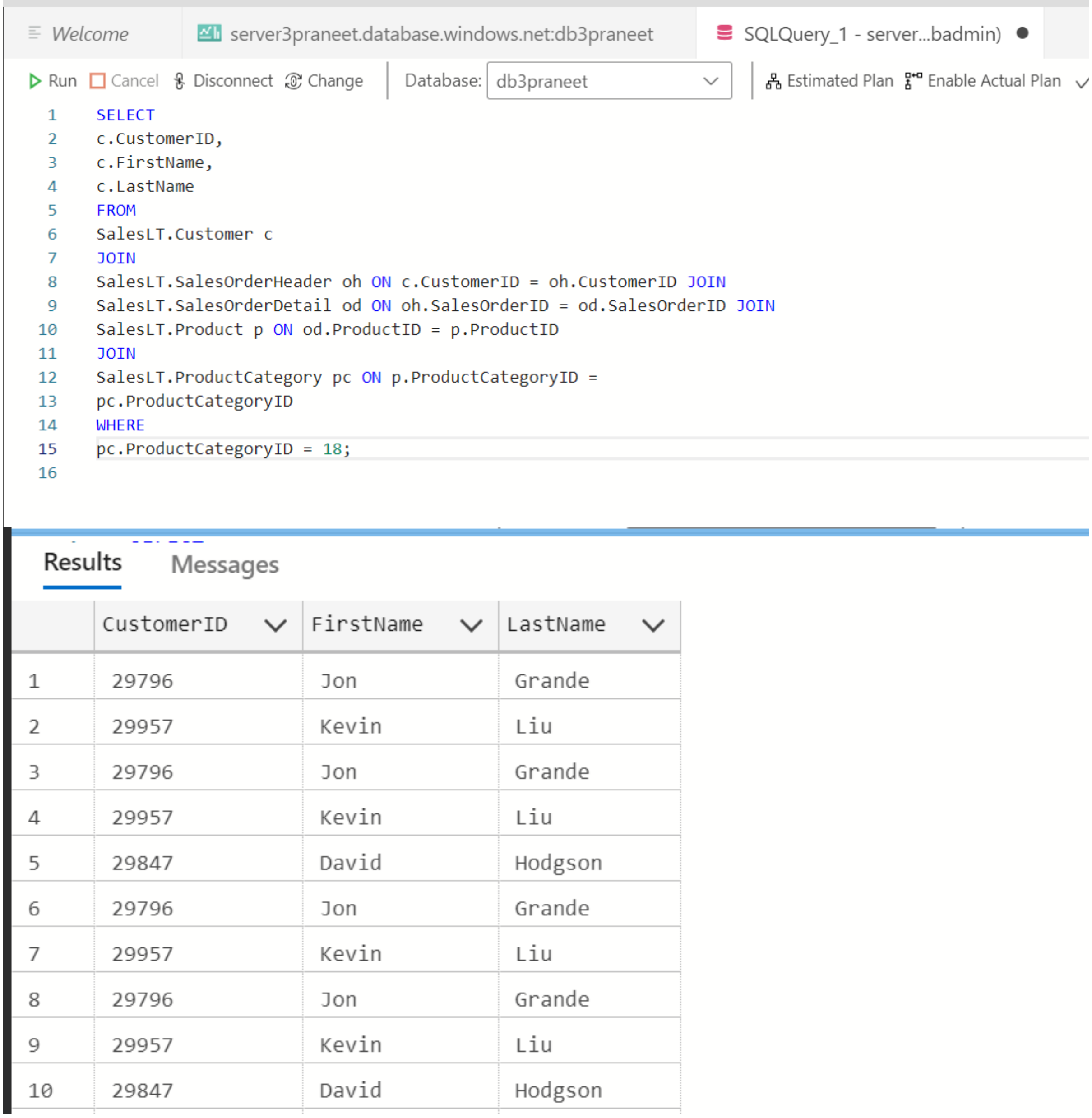


```sql
WITH ProductAverages AS (
    SELECT
        AVG(UnitPrice) AS AvgPrice
    FROM
        SalesLT.SalesOrderDetail
)

SELECT
    P.ProductID,
    P.Name,
    P.ProductNumber,
    P.ListPrice
FROM
    SalesLT.Product P
JOIN
    ProductAverages PA ON P.ListPrice > PA.AvgPrice;
```

| | ProductID | Name | ProductNumber | ListPrice | AvgPrice |
|---|---|---|---|---|---|
| 1 | 680 | HL Road Frame - Black, 58 | FR-R92B-58 | 1431.50 | 382.87 |
| 2 | 706 | HL Road Frame - Red, 58 | FR-R92R-58 | 1431.50 | 382.87 |
| 3 | 717 | HL Road Frame - Red, 62 | FR-R92R-62 | 1431.50 | 382.87 |
| 4 | 718 | HL Road Frame - Red, 44 | FR-R92R-44 | 1431.50 | 382.87 |
| 5 | 719 | HL Road Frame - Red, 48 | FR-R92R-48 | 1431.50 | 382.87 |
| 6 | 720 | HL Road Frame - Red, 52 | FR-R92R-52 | 1431.50 | 382.87 |
| 7 | 721 | HL Road Frame - Red, 56 | FR-R92R-56 | 1431.50 | 382.87 |
| 8 | 731 | ML Road Frame - Red, 44 | FR-R72R-44 | 594.83 | 382.87 |
| 9 | 732 | ML Road Frame - Red, 48 | FR-R72R-48 | 594.83 | 382.87 |
| 10 | 733 | ML Road Frame - Red, 52 | FR-R72R-52 | 594.83 | 382.87 |
| 11 | 734 | ML Road Frame - Red, 58 | FR-R72R-58 | 594.83 | 382.87 |
| 12 | 735 | ML Road Frame - Red, 60 | FR-R72R-60 | 594.83 | 382.87 |
| 13 | 739 | HL Mountain Frame - Silv… | FR-M94S-42 | 1364.50 | 382.87 |
| 14 | 740 | HL Mountain Frame - Silv… | FR-M94S-44 | 1364.50 | 382.87 |
| 15 | 741 | HL Mountain Frame - Silv… | FR-M94S-52 | 1364.50 | 382.87 |
| 16 | 742 | HL Mountain Frame - Silv… | FR-M94S-46 | 1364.50 | 382.87 |
| 17 | 743 | HL Mountain Frame - Blac… | FR-M94B-42 | 1349.60 | 382.87 |
| 18 | 744 | HL Mountain Frame - Blac… | FR-M94B-44 | 1349.60 | 382.87 |
| 19 | 745 | HL Mountain Frame - Blac… | FR-M94B-48 | 1349.60 | 382.87 |
| 20 | 746 | HL Mountain Frame - Blac… | FR-M94B-46 | 1349.60 | 382.87 |
| 21 | 747 | HL Mountain Frame - Blac… | FR-M94B-38 | 1349.60 | 382.87 |

14. **Retrieve orders placed by employees who have a specific job title. (DATA INSUFFICIENT)**

15. **Display customers who have placed orders for a specific product category.**



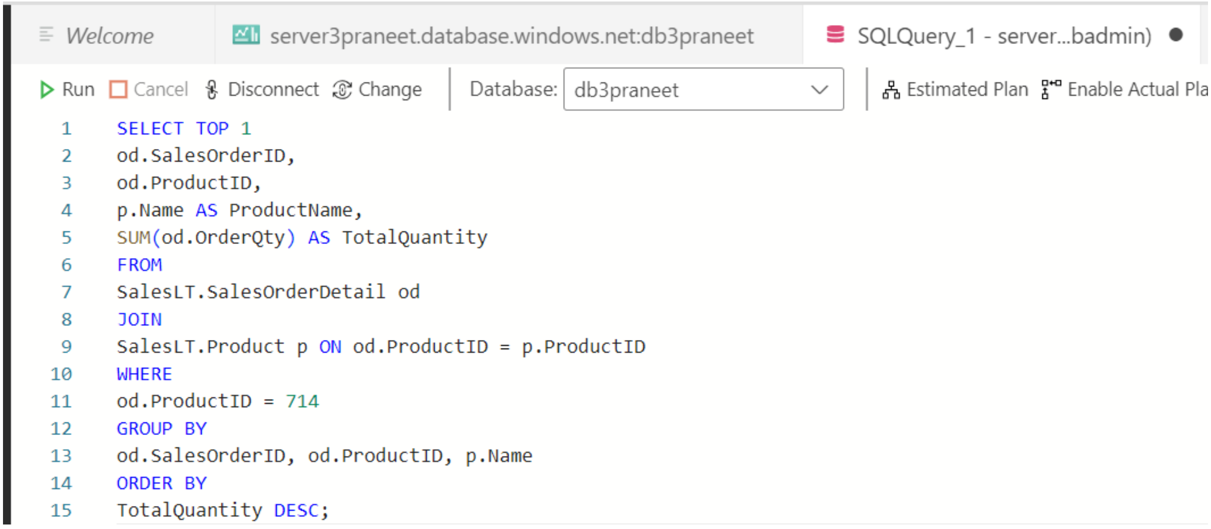
```sql
SELECT
c.CustomerID,
c.FirstName,
c.LastName
FROM
SalesLT.Customer c
JOIN
SalesLT.SalesOrderHeader oh ON c.CustomerID = oh.CustomerID JOIN
SalesLT.SalesOrderDetail od ON oh.SalesOrderID = od.SalesOrderID JOIN
SalesLT.Product p ON od.ProductID = p.ProductID
JOIN
SalesLT.ProductCategory pc ON p.ProductCategoryID =
pc.ProductCategoryID
WHERE
pc.ProductCategoryID = 18;
```

| | CustomerID | FirstName | LastName |
|---|---|---|---|
| 1 | 29796 | Jon | Grande |
| 2 | 29957 | Kevin | Liu |
| 3 | 29796 | Jon | Grande |
| 4 | 29957 | Kevin | Liu |
| 5 | 29847 | David | Hodgson |
| 6 | 29796 | Jon | Grande |
| 7 | 29957 | Kevin | Liu |
| 8 | 29796 | Jon | Grande |
| 9 | 29957 | Kevin | Liu |
| 10 | 29847 | David | Hodgson |

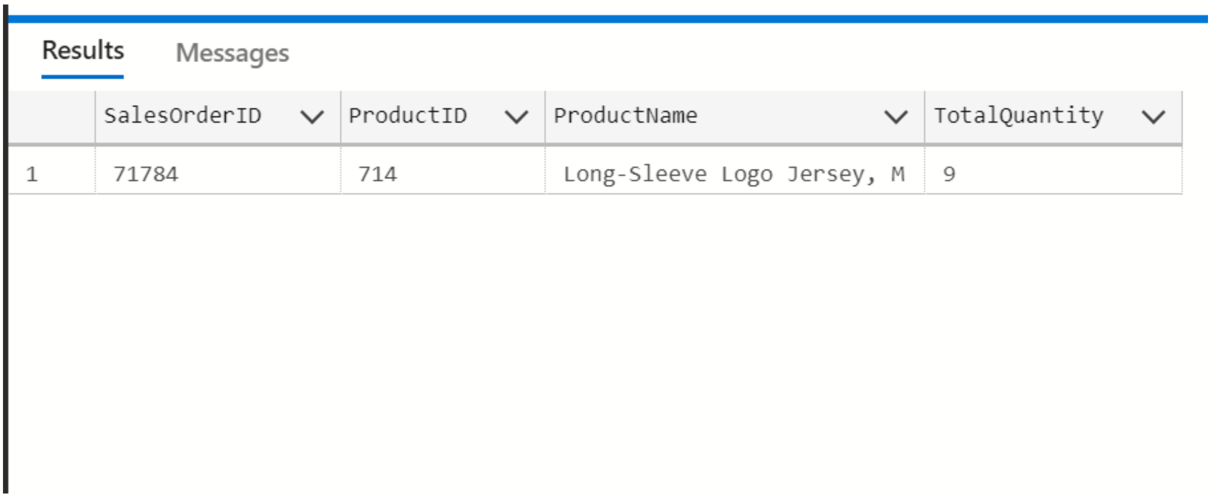
16. **Find employees with salaries greater than the average salary in their department. (DATA INSUFFICIENT)**

17. **List customers who have placed orders before a specific date. (DATA INSUFFICIENT)**

18. **Retrieve the order with the highest quantity of a specific product.**

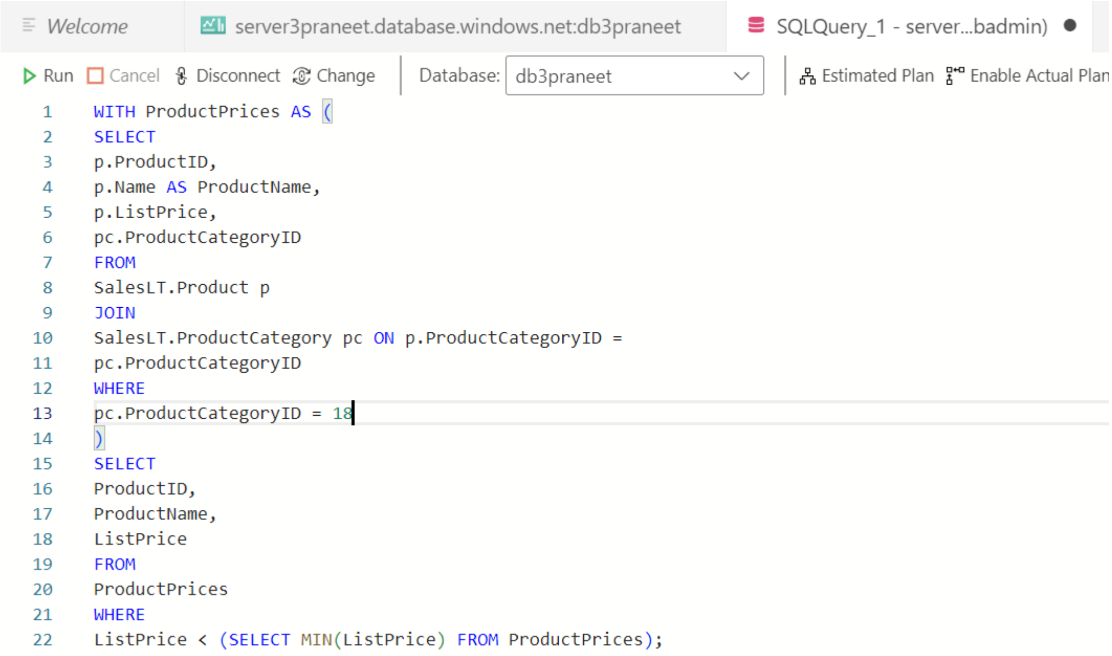


```sql
SELECT TOP 1
od.SalesOrderID,
od.ProductID,
p.Name AS ProductName,
SUM(od.OrderQty) AS TotalQuantity
FROM
SalesLT.SalesOrderDetail od
JOIN
SalesLT.Product p ON od.ProductID = p.ProductID
WHERE
od.ProductID = 714
GROUP BY
od.SalesOrderID, od.ProductID, p.Name
ORDER BY
TotalQuantity DESC;
```



| | SalesOrderID | ProductID | ProductName | TotalQuantity |
|---|---|---|---|---|
| 1 | 71784 | 714 | Long-Sleeve Logo Jersey, M | 9 |

19. **Display products with prices lower than the lowest product price in a specific category.**


Welcome
server3praneet.database.windows.net:db3praneet
SQLQuery_1 - server...badmin) ●
Run
Cancel
Disconnect
Change
Estimated Plan
Enable Actual Plan

Database: db3praneet

```sql
WITH ProductPrices AS (
SELECT
p.ProductID,
p.Name AS ProductName,
p.ListPrice,
pc.ProductCategoryID
FROM
SalesLT.Product p
JOIN
SalesLT.ProductCategory pc ON p.ProductCategoryID =
pc.ProductCategoryID
WHERE
pc.ProductCategoryID = 18
)
SELECT
ProductID,
ProductName,
ListPrice
FROM
ProductPrices
WHERE
ListPrice < (SELECT MIN(ListPrice) FROM ProductPrices);
```

20. **Find employees who have the same job title as their manager. (DATA INSUFFICIENT).**

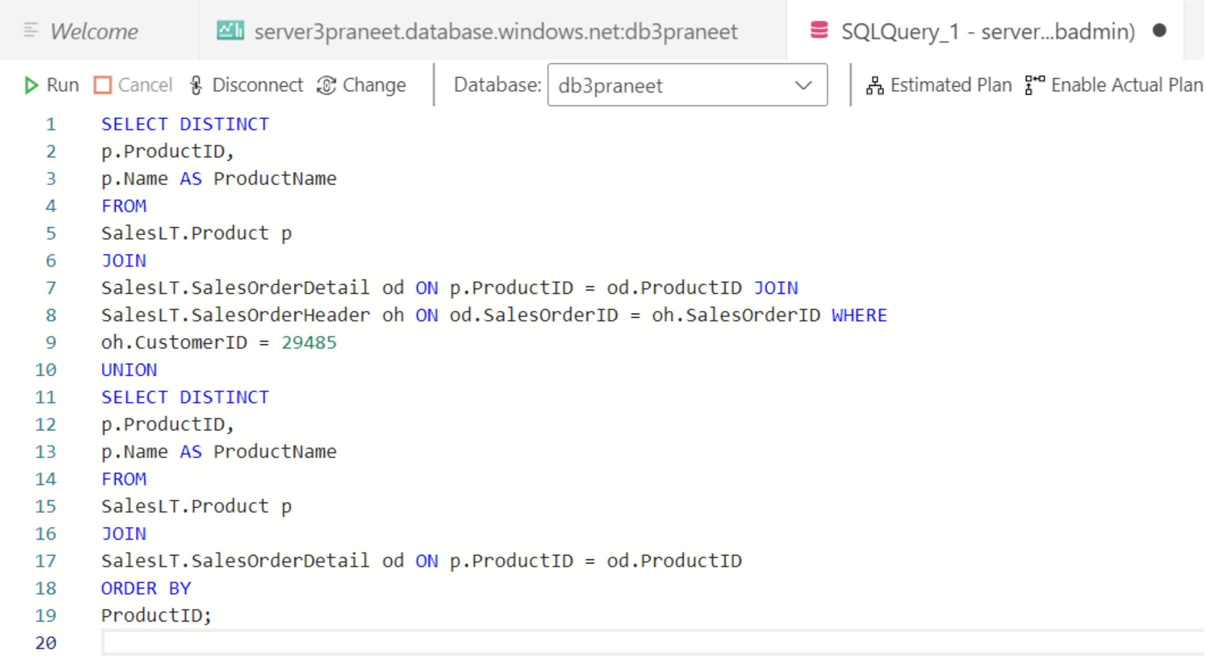
21. **Combine results from two queries to get a list of unique customer and employee names. (DATA INSUFFICIENT).**

22. **Retrieve product names that are common in two different product categories. (DATA INSUFFICIENT).**

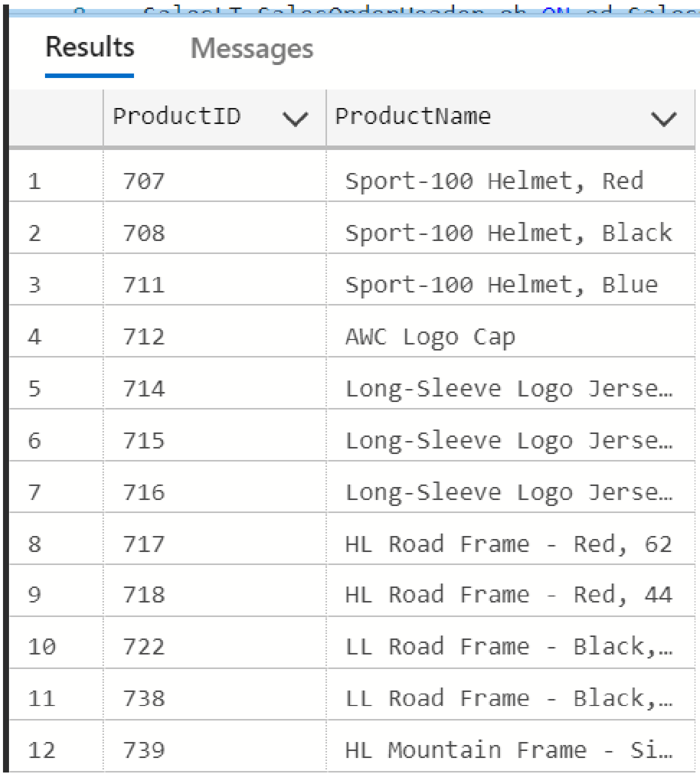
23. **Display the names of employees and customers in a single result set. (DATA INSUFFICIENT).**

24. **List products that are in stock or have been discontinued. (DATA INSUFFICIENT).**

25. **Combine the results of two queries to find unique products ordered by a specific customer. (DATA INSUFFICIENT)**



```
SELECT DISTINCT
p.ProductID,
p.Name AS ProductName
FROM
SalesLT.Product p
JOIN
SalesLT.SalesOrderDetail od ON p.ProductID = od.ProductID JOIN
SalesLT.SalesOrderHeader oh ON od.SalesOrderID = oh.SalesOrderID WHERE
oh.CustomerID = 29485
UNION
SELECT DISTINCT
p.ProductID,
p.Name AS ProductName
FROM
SalesLT.Product p
JOIN
SalesLT.SalesOrderDetail od ON p.ProductID = od.ProductID
ORDER BY
ProductID;
```



| | ProductID | ProductName |
|---|---|---|
| 1 | 707 | Sport-100 Helmet, Red |
| 2 | 708 | Sport-100 Helmet, Black |
| 3 | 711 | Sport-100 Helmet, Blue |
| 4 | 712 | AWC Logo Cap |
| 5 | 714 | Long-Sleeve Logo Jerse… |
| 6 | 715 | Long-Sleeve Logo Jerse… |
| 7 | 716 | Long-Sleeve Logo Jerse… |
| 8 | 717 | HL Road Frame - Red, 62 |
| 9 | 718 | HL Road Frame - Red, 44 |
| 10 | 722 | LL Road Frame - Black,… |
| 11 | 738 | LL Road Frame - Black,… |
| 12 | 739 | HL Mountain Frame - Si… |

26. **Retrieve orders placed by customers and employees in a single result set. (DATA INSUFFICIENT).**

27. **Display products that are either in a specific category or have a specific safety stock level. (DATA INSUFFICIENT).**

28. **List customers who have placed orders and employees who have direct reports in a single result set. (DATA INSUFFICIENT).**

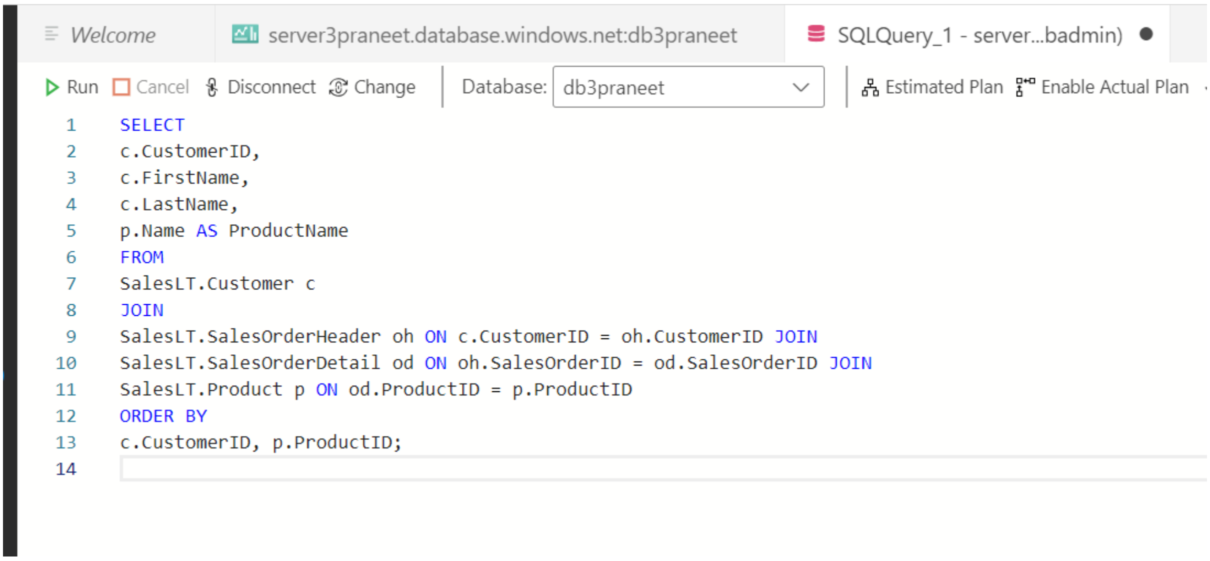
29. **Retrieve products that are in stock in one location and out of stock in another. (DATA INSUFFICIENT).**

30. **Combine information about employees who are managers and employees who have managers. (DATA INSUFFICIENT).**

# INTERMEDIATE

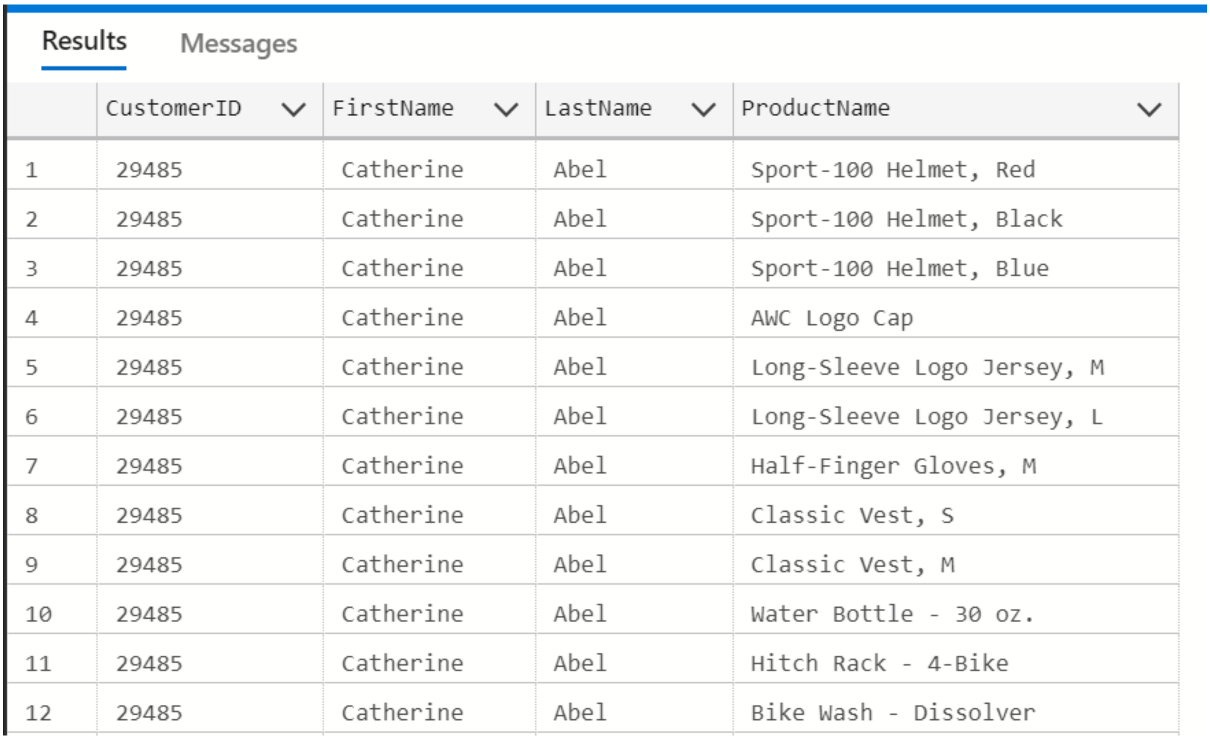
31. **Retrieve a list of customers along with the names of the products they have purchased.**

```sql
SELECT
    c.CustomerID,
    c.FirstName,
    c.LastName,
    p.Name AS ProductName
FROM
    SalesLT.Customer c
JOIN
    SalesLT.SalesOrderHeader oh ON c.CustomerID = oh.CustomerID JOIN
    SalesLT.SalesOrderDetail od ON oh.SalesOrderID = od.SalesOrderID JOIN
    SalesLT.Product p ON od.ProductID = p.ProductID
ORDER BY
    c.CustomerID, p.ProductID;
```


Welcome
server3praneet.database.windows.net:db3praneet
SQLQuery_1 - server...badmin)
Run
Cancel
Disconnect
Change
Estimated Plan
Enable Actual Plan

Database: db3praneet

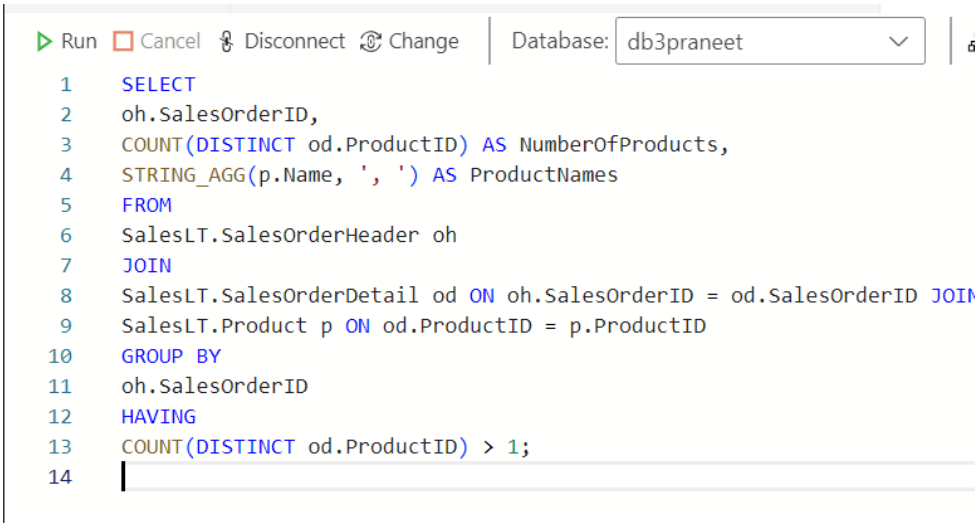
```sql
SELECT
c.CustomerID,
c.FirstName,
c.LastName,
p.Name AS ProductName
FROM
SalesLT.Customer c
JOIN
SalesLT.SalesOrderHeader oh ON c.CustomerID = oh.CustomerID JOIN
SalesLT.SalesOrderDetail od ON oh.SalesOrderID = od.SalesOrderID JOIN
SalesLT.Product p ON od.ProductID = p.ProductID
ORDER BY
c.CustomerID, p.ProductID;
```



| | CustomerID | FirstName | LastName | ProductName |
|---|---|---|---|---|
| 1 | 29485 | Catherine | Abel | Sport-100 Helmet, Red |
| 2 | 29485 | Catherine | Abel | Sport-100 Helmet, Black |
| 3 | 29485 | Catherine | Abel | Sport-100 Helmet, Blue |
| 4 | 29485 | Catherine | Abel | AWC Logo Cap |
| 5 | 29485 | Catherine | Abel | Long-Sleeve Logo Jersey, M |
| 6 | 29485 | Catherine | Abel | Long-Sleeve Logo Jersey, L |
| 7 | 29485 | Catherine | Abel | Half-Finger Gloves, M |
| 8 | 29485 | Catherine | Abel | Classic Vest, S |
| 9 | 29485 | Catherine | Abel | Classic Vest, M |
| 10 | 29485 | Catherine | Abel | Water Bottle - 30 oz. |
| 11 | 29485 | Catherine | Abel | Hitch Rack - 4-Bike |
| 12 | 29485 | Catherine | Abel | Bike Wash - Dissolver |

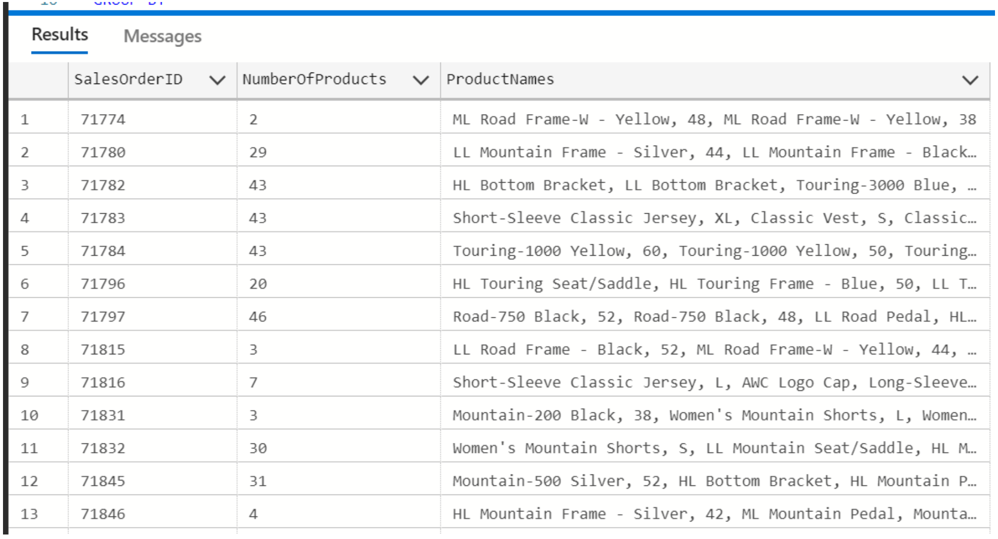
32. **Display employees who have the same manager, including indirect reports. (DATA INSUFFICIENT).**

33. **Find orders with multiple products and display the product names.**



Database: db3praneet

```
SELECT
oh.SalesOrderID,
COUNT(DISTINCT od.ProductID) AS NumberOfProducts,
STRING_AGG(p.Name, ', ') AS ProductNames
FROM
SalesLT.SalesOrderHeader oh
JOIN
SalesLT.SalesOrderDetail od ON oh.SalesOrderID = od.SalesOrderID JOIN
SalesLT.Product p ON od.ProductID = p.ProductID
GROUP BY
oh.SalesOrderID
HAVING
COUNT(DISTINCT od.ProductID) > 1;
```

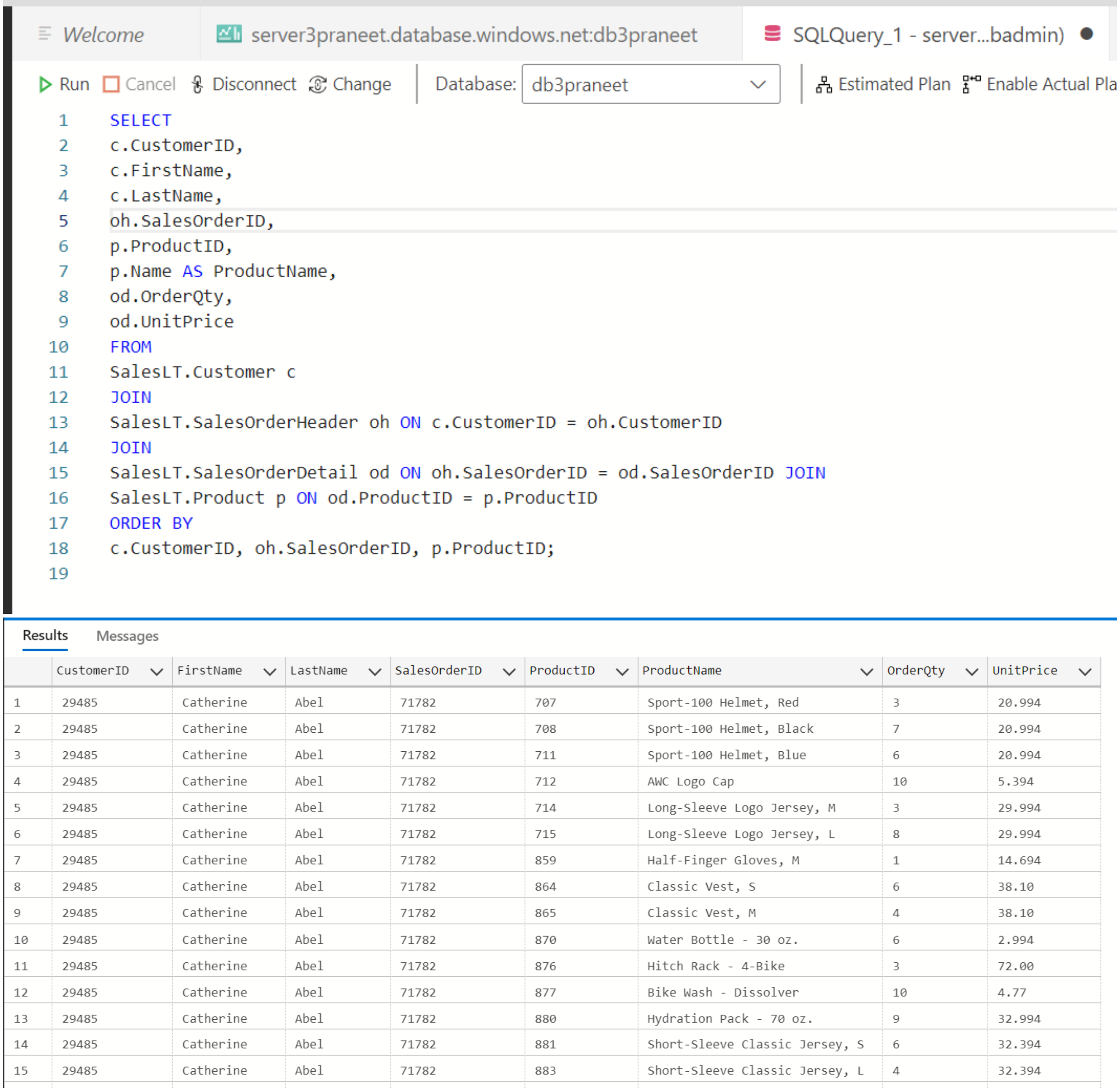


| | SalesOrderID | NumberOfProducts | ProductNames |
|---|---|---|---|
| 1 | 71774 | 2 | ML Road Frame-W - Yellow, 48, ML Road Frame-W - Yellow, 38 |
| 2 | 71780 | 29 | LL Mountain Frame - Silver, 44, LL Mountain Frame - Black… |
| 3 | 71782 | 43 | HL Bottom Bracket, LL Bottom Bracket, Touring-3000 Blue, … |
| 4 | 71783 | 43 | Short-Sleeve Classic Jersey, XL, Classic Vest, S, Classic… |
| 5 | 71784 | 43 | Touring-1000 Yellow, 60, Touring-1000 Yellow, 50, Touring… |
| 6 | 71796 | 20 | HL Touring Seat/Saddle, HL Touring Frame - Blue, 50, LL T… |
| 7 | 71797 | 46 | Road-750 Black, 52, Road-750 Black, 48, LL Road Pedal, HL… |
| 8 | 71815 | 3 | LL Road Frame - Black, 52, ML Road Frame-W - Yellow, 44, … |
| 9 | 71816 | 7 | Short-Sleeve Classic Jersey, L, AWC Logo Cap, Long-Sleeve… |
| 10 | 71831 | 3 | Mountain-200 Black, 38, Women's Mountain Shorts, L, Women… |
| 11 | 71832 | 30 | Women's Mountain Shorts, S, LL Mountain Seat/Saddle, HL M… |
| 12 | 71845 | 31 | Mountain-500 Silver, 52, HL Bottom Bracket, HL Mountain P… |
| 13 | 71846 | 4 | HL Mountain Frame - Silver, 42, ML Mountain Pedal, Mounta… |

34. **List customers along with the names of the salespeople who handled their orders. (DATA INSUFFICIENT).**

35. **Retrieve a list of products along with the names of suppliers. (DATA INSUFFICIENT).**

36. **Display customers who have placed orders and the products they have purchased, including product details.**



```sql
SELECT
c.CustomerID,
c.FirstName,
c.LastName,
oh.SalesOrderID,
p.ProductID,
p.Name AS ProductName,
od.OrderQty,
od.UnitPrice
FROM
SalesLT.Customer c
JOIN
SalesLT.SalesOrderHeader oh ON c.CustomerID = oh.CustomerID
JOIN
SalesLT.SalesOrderDetail od ON oh.SalesOrderID = od.SalesOrderID JOIN
SalesLT.Product p ON od.ProductID = p.ProductID
ORDER BY
c.CustomerID, oh.SalesOrderID, p.ProductID;
```

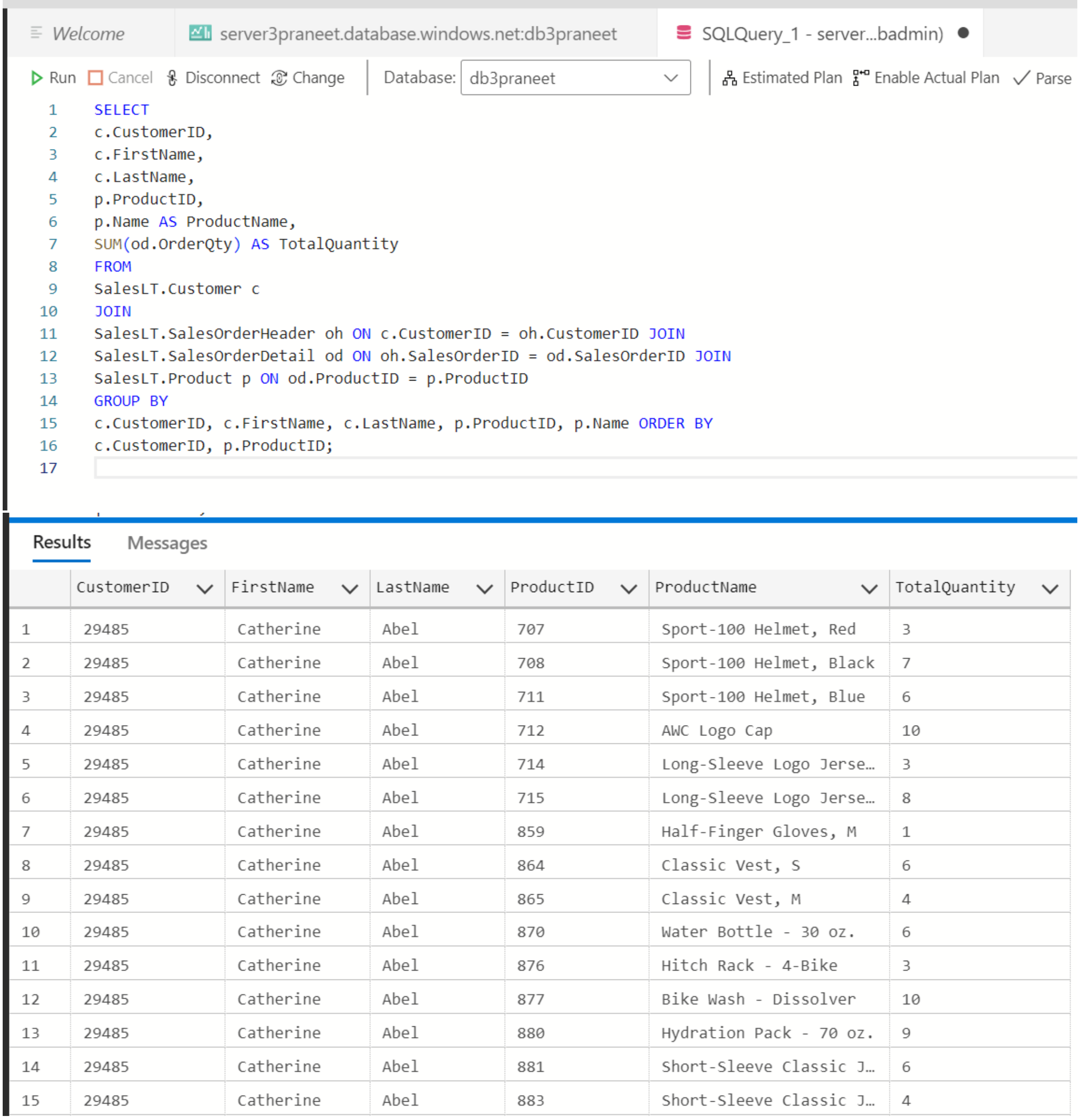
| | CustomerID | FirstName | LastName | SalesOrderID | ProductID | ProductName | OrderQty | UnitPrice |
|---|---|---|---|---|---|---|---|---|
| 1 | 29485 | Catherine | Abel | 71782 | 707 | Sport-100 Helmet, Red | 3 | 20.994 |
| 2 | 29485 | Catherine | Abel | 71782 | 708 | Sport-100 Helmet, Black | 7 | 20.994 |
| 3 | 29485 | Catherine | Abel | 71782 | 711 | Sport-100 Helmet, Blue | 6 | 20.994 |
| 4 | 29485 | Catherine | Abel | 71782 | 712 | AWC Logo Cap | 10 | 5.394 |
| 5 | 29485 | Catherine | Abel | 71782 | 714 | Long-Sleeve Logo Jersey, M | 3 | 29.994 |
| 6 | 29485 | Catherine | Abel | 71782 | 715 | Long-Sleeve Logo Jersey, L | 8 | 29.994 |
| 7 | 29485 | Catherine | Abel | 71782 | 859 | Half-Finger Gloves, M | 1 | 14.694 |
| 8 | 29485 | Catherine | Abel | 71782 | 864 | Classic Vest, S | 6 | 38.10 |
| 9 | 29485 | Catherine | Abel | 71782 | 865 | Classic Vest, M | 4 | 38.10 |
| 10 | 29485 | Catherine | Abel | 71782 | 870 | Water Bottle - 30 oz. | 6 | 2.994 |
| 11 | 29485 | Catherine | Abel | 71782 | 876 | Hitch Rack - 4-Bike | 3 | 72.00 |
| 12 | 29485 | Catherine | Abel | 71782 | 877 | Bike Wash - Dissolver | 10 | 4.77 |
| 13 | 29485 | Catherine | Abel | 71782 | 880 | Hydration Pack - 70 oz. | 9 | 32.994 |
| 14 | 29485 | Catherine | Abel | 71782 | 881 | Short-Sleeve Classic Jersey, S | 6 | 32.394 |
| 15 | 29485 | Catherine | Abel | 71782 | 883 | Short-Sleeve Classic Jersey, L | 4 | 32.394 |

37. **Find orders where multiple employees were involved, showing the employee names. (DATA INSUFFICIENT).**

38. **List products that have similar names but belong to different categories. (DATA INSUFFICIENT).**

39. **Retrieve a list of employees along with their training courses and training dates. (DATA INSUFFICIENT).**

40. **Display customers who have placed orders and the total quantity of each product ordered.**

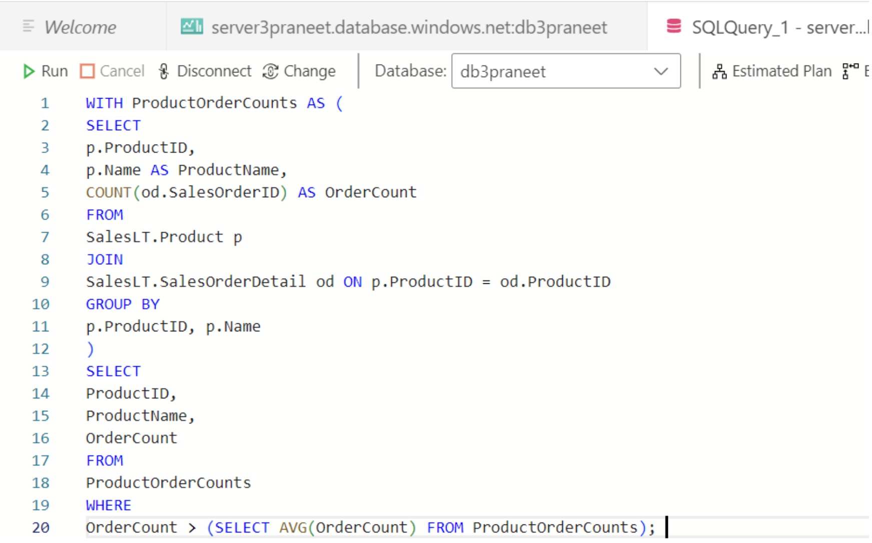


```sql
SELECT
c.CustomerID,
c.FirstName,
c.LastName,
p.ProductID,
p.Name AS ProductName,
SUM(od.OrderQty) AS TotalQuantity
FROM
SalesLT.Customer c
JOIN
SalesLT.SalesOrderHeader oh ON c.CustomerID = oh.CustomerID JOIN
SalesLT.SalesOrderDetail od ON oh.SalesOrderID = od.SalesOrderID JOIN
SalesLT.Product p ON od.ProductID = p.ProductID
GROUP BY
c.CustomerID, c.FirstName, c.LastName, p.ProductID, p.Name ORDER BY
c.CustomerID, p.ProductID;
```

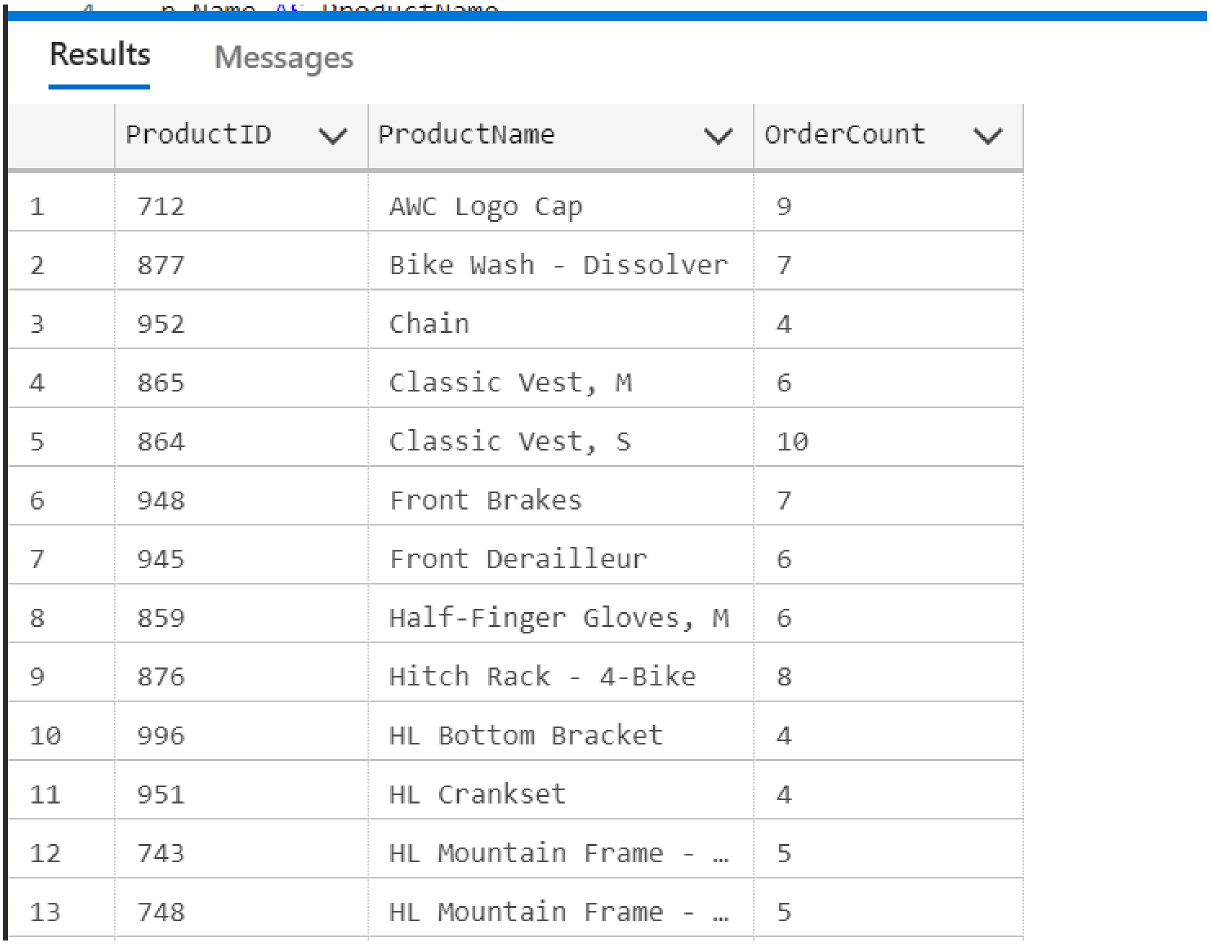
| | CustomerID | FirstName | LastName | ProductID | ProductName | TotalQuantity |
|---|---|---|---|---|---|---|
| 1 | 29485 | Catherine | Abel | 707 | Sport-100 Helmet, Red | 3 |
| 2 | 29485 | Catherine | Abel | 708 | Sport-100 Helmet, Black | 7 |
| 3 | 29485 | Catherine | Abel | 711 | Sport-100 Helmet, Blue | 6 |
| 4 | 29485 | Catherine | Abel | 712 | AWC Logo Cap | 10 |
| 5 | 29485 | Catherine | Abel | 714 | Long-Sleeve Logo Jerse… | 3 |
| 6 | 29485 | Catherine | Abel | 715 | Long-Sleeve Logo Jerse… | 8 |
| 7 | 29485 | Catherine | Abel | 859 | Half-Finger Gloves, M | 1 |
| 8 | 29485 | Catherine | Abel | 864 | Classic Vest, S | 6 |
| 9 | 29485 | Catherine | Abel | 865 | Classic Vest, M | 4 |
| 10 | 29485 | Catherine | Abel | 870 | Water Bottle - 30 oz. | 6 |
| 11 | 29485 | Catherine | Abel | 876 | Hitch Rack - 4-Bike | 3 |
| 12 | 29485 | Catherine | Abel | 877 | Bike Wash - Dissolver | 10 |
| 13 | 29485 | Catherine | Abel | 880 | Hydration Pack - 70 oz. | 9 |
| 14 | 29485 | Catherine | Abel | 881 | Short-Sleeve Classic J… | 6 |
| 15 | 29485 | Catherine | Abel | 883 | Short-Sleeve Classic J… | 4 |

41. **Find customers who have made more purchases than the average number of purchases. (DATA INSUFFICIENT).**

42. **Display products that have been ordered more than the average number of times.**



```sql
WITH ProductOrderCounts AS (
SELECT
p.ProductID,
p.Name AS ProductName,
COUNT(od.SalesOrderID) AS OrderCount
FROM
SalesLT.Product p
JOIN
SalesLT.SalesOrderDetail od ON p.ProductID = od.ProductID
GROUP BY
p.ProductID, p.Name
)
SELECT
ProductID,
ProductName,
OrderCount
FROM
ProductOrderCounts
WHERE
OrderCount > (SELECT AVG(OrderCount) FROM ProductOrderCounts);
```



| | ProductID | ProductName | OrderCount |
|---|---|---|---|
| 1 | 712 | AWC Logo Cap | 9 |
| 2 | 877 | Bike Wash - Dissolver | 7 |
| 3 | 952 | Chain | 4 |
| 4 | 865 | Classic Vest, M | 6 |
| 5 | 864 | Classic Vest, S | 10 |
| 6 | 948 | Front Brakes | 7 |
| 7 | 945 | Front Derailleur | 6 |
| 8 | 859 | Half-Finger Gloves, M | 6 |
| 9 | 876 | Hitch Rack - 4-Bike | 8 |
| 10 | 996 | HL Bottom Bracket | 4 |
| 11 | 951 | HL Crankset | 4 |
| 12 | 743 | HL Mountain Frame - … | 5 |
| 13 | 748 | HL Mountain Frame - … | 5 |

43. **Retrieve orders placed by employees who have completed a specific training course. (DATA INSUFFICIENT).**

44. **List employees who have a higher salary than at least one employee in another department. (DATA INSUFFICIENT).**

45. **Display products that have not been ordered in the last 60 days. (DATA INSUFFICIENT).**

46. **Find employees who have the same job title as the employee with the highest salary. (DATA INSUFFICIENT).**

47. **List customers who have placed orders with a total amount greater than the total amount of a specific order. (DATA INSUFFICIENT).**

48. **Retrieve products that have been ordered by customers with the same shipping address. (DATA INSUFFICIENT).**

**49. Display orders with quantities higher than the average quantity for a specific product.**

```
WITH AvgQuantityPerProduct AS (
    SELECT
        od.ProductID,
        AVG(od.OrderQty) AS AvgQuantity
    FROM
        SalesLT.SalesOrderDetail od
    GROUP BY
        od.ProductID
)

SELECT
    od.SalesOrderID,
    od.ProductID,
    p.Name AS ProductName,
    od.OrderQty
FROM
    SalesLT.SalesOrderDetail od
JOIN
    SalesLT.Product p ON od.ProductID = p.ProductID
JOIN
    AvgQuantityPerProduct avgQ ON od.ProductID = avgQ.ProductID
WHERE
    od.OrderQty > avgQ.AvgQuantity
ORDER BY
    od.SalesOrderID, od.ProductID;
```

### 50. Find customers who have placed orders for products that have not been ordered by any other customer.

```
WITH ProductsOrderedByCustomers AS (
    SELECT DISTINCT
        od.ProductID
    FROM
        SalesLT.SalesOrderDetail od
)

SELECT
    c.CustomerID,
    c.FirstName,
    c.LastName,
    oh.SalesOrderID,
    p.ProductID,
    p.Name AS ProductName
FROM
    SalesLT.Customer c
JOIN
    SalesLT.SalesOrderHeader oh ON c.CustomerID = oh.CustomerID
JOIN
    SalesLT.SalesOrderDetail od ON oh.SalesOrderID = od.SalesOrderID
JOIN
    SalesLT.Product p ON od.ProductID = p.ProductID
WHERE
    p.ProductID NOT IN (SELECT ProductID FROM ProductsOrderedByCustomers)
ORDER BY
    c.CustomerID, oh.SalesOrderID, p.ProductID;
```