Лаб 1

1. SELECT Name, Color, Size

FROM Production.Product

2. SELECT Name, Color, Size

FROM Production.Product

WHERE ListPrice > 100

3. SELECT Name, Color, Size

FROM Production.Product

WHERE ListPrice < 100 AND Color = 'BLack'

4. SELECT Name, Color, Size

FROM Production.Product

WHERE ListPrice < 100 AND Color = 'Black'

ORDER BY ListPrice

5. SELECT TOP 3 Name, Size

FROM Production.Product

WHERE Color = 'Black'

ORDER BY ListPrice DESC

6. SELECT Name, Color

FROM Production.Product

WHERE Color IS NOT NULL AND Size IS NOT NULL

7. SELECT DISTINCT Color

FROM Production.Product

WHERE ListPrice BETWEEN 10 AND 50 AND Color IS NOT NULL

8. SELECT Color

FROM Production.Product

WHERE Name LIKE 'L\_N%'

9. SELECT Name

FROM Production.Product

WHERE Name LIKE '[DM]\_\_\_%'

10. SELECT Name

FROM Production.Product

WHERE SellStartDate <= '2012-31-12'

11. SELECT Name

FROM Production.ProductSubCategory

12. SELECT Name

FROM Production.ProductCategory

13. SELECT FirstName

FROM Person.Person

WHERE Title = 'Mr.'

14. SELECT FirstName

FROM Person.Person

WHERE Title IS NULL

Лаб 2

1. SELECT Color, Count(\*)

FROM Production.Product

WHERE Color IS NOT NULL AND ListPrice >= 30

GROUP BY Color

2. SELECT Color

FROM Production.Product

GROUP BY Color

HAVING MIN(ListPrice) > 100

3. SELECT ProductSubcategoryID, Count(\*)

FROM Production.Product

WHERE ProductSubcategoryID IS NOT NULL

GROUP BY ProductSubcategoryID

4. SELECT ProductID, Count(\*)

FROM Sales.SalesOrderDetail

GROUP BY ProductID

5. SELECT ProductID

FROM Sales.SalesOrderDetail

GROUP BY ProductID

HAVING Count(\*) > 5

6. SELECT CustomerID

FROM Sales.SalesOrderHeader

GROUP BY CustomerID, OrderDate

HAVING Count(\*) > 1

7. SELECT SalesOrderID

FROM Sales.SalesOrderDetail

GROUP BY SalesOrderID

HAVING Count(\*) > 3

8. SELECT ProductID

FROM Sales.SalesOrderDetail

GROUP BY ProductID

HAVING Count(\*) > 3

9. SELECT ProductID

FROM Sales.SalesOrderDetail

GROUP BY ProductID

HAVING Count(\*) = 3 OR Count(\*) = 5

10. SELECT ProductSubcategoryID

FROM Production.Product

WHERE ProductSubcategoryID IS NOT NULL

GROUP BY ProductSubcategoryID

HAVING Count(\*) > 10

11. SELECT ProductID

FROM Sales.SalesOrderDetail

WHERE OrderQty = 1

GROUP BY ProductID

12. SELECT TOP 1 SalesOrderID

FROM Sales.SalesOrderDetail

GROUP BY SalesOrderID

ORDER BY Count(\*) DESC

13. SELECT TOP 1 SalesOrderID

FROM Sales.SalesOrderDetail

GROUP BY SalesOrderID

ORDER BY SUM(UnitPrice\*OrderQty) DESC

14. SELECT ProductSubcategoryID, Count(\*) AS Amount

FROM Production.Product

WHERE ProductSubcategoryID IS NOT NULL AND Color IS NOT NULL

GROUP BY ProductSubcategoryID

15. SELECT Color

FROM Production.Product

WHERE Color IS NOT NULL

GROUP BY Color

ORDER BY Count(\*) DESC

16. SELECT ProductID

FROM Sales.SalesOrderDetail

GROUP BY ProductID

HAVING Count(\*) > 2 AND MIN(OrderQty)>1

Лаб 3

1. SELECT P.Name, PC.Name

FROM Production.Product AS P INNER JOIN

Production.ProductSubcategory AS PSC

ON P.ProductSubcategoryID = PSC.ProductSubcategoryID

INNER JOIN Production.ProductCategory AS PC

ON PSC.ProductCategoryID = PC.ProductCategoryID

WHERE P.Color = 'Red' AND P.ListPrice >= 100

1. SELECT DISTINCT p1.Name

FROM Production.ProductSubcategory AS p1,

Production.ProductSubcategory AS p2

WHERE p1.ProductSubcategoryID != p2.ProductSubcategoryID AND

p1.Name = p2.Name

1. SELECT pc.Name, COUNT(\*)

FROM Production.Product AS p INNER JOIN

Production.ProductSubcategory AS psc

ON p.ProductSubcategoryID = psc.ProductSubcategoryID

INNER JOIN Production.ProductCategory AS pc

ON psc.ProductCategoryID = pc.ProductCategoryID

GROUP BY pc.Name

1. SELECT ps.Name, COUNT(\*)

FROM Production.Product AS p

INNER JOIN Production.ProductSubcategory AS ps

ON p.ProductSubcategoryID = ps.ProductSubcategoryID

GROUP BY ps.ProductSubcategoryID, ps.Name

1. SELECT TOP 3 ps.Name

FROM Production.Product AS p

INNER JOIN Production.ProductSubcategory AS ps

ON p.ProductSubcategoryID = ps.ProductSubcategoryID

GROUP BY ps.Name

ORDER BY COUNT(\*) DESC

1. SELECT ps.Name, MAX(p.ListPrice)

FROM Production.Product AS p

INNER JOIN Production.ProductSubcategory AS ps

ON p.ProductSubcategoryID = ps.ProductSubcategoryID

WHERE p.Color = 'Red'

GROUP BY ps.Name

1. SELECT v.Name, COUNT(\*) AS ProductAmount

FROM Purchasing.ProductVendor AS pv

INNER JOIN Purchasing.Vendor AS v

ON pv.BusinessEntityID = v.BusinessEntityID

GROUP BY v.Name

1. SELECT p.Name

FROM Production.Product AS p

INNER JOIN Purchasing.ProductVendor AS pv

ON p.ProductID = pv.ProductID

GROUP BY p.Name

HAVING Count(\*)>1

1. SELECT TOP 1 p.Name

FROM Production.Product AS p

INNER JOIN Sales.SalesOrderDetail AS sod

ON p.ProductID = sod.ProductID

GROUP BY p.Name

ORDER BY SUM(sod.OrderQty) DESC

1. SELECT TOP 1 pc.Name

FROM Production.Product AS p

INNER JOIN Production.ProductSubcategory AS ps

ON p.ProductSubcategoryID = ps.ProductSubcategoryID

INNER JOIN Production.ProductCategory AS pc

ON ps.ProductCategoryID = pc.ProductCategoryID

INNER JOIN Sales.SalesOrderDetail AS sod

ON p.ProductID = sod.ProductID

GROUP BY pc.Name

ORDER BY SUM(sod.OrderQty) DESC

1. SELECT pc.Name, COUNT(DISTINCT ps.ProductSubcategoryID) AS SubcategoryAmount,

COUNT(p.ProductID) AS ProductAmount

FROM Production.Product AS p

INNER JOIN Production.ProductSubcategory AS ps

ON p.ProductSubcategoryID = ps.ProductSubcategoryID

INNER JOIN Production.ProductCategory AS pc

ON ps.ProductCategoryID = pc.ProductCategoryID

GROUP BY pc.Name

1. SELECT v.CreditRating, COUNT(DISTINCT pv.ProductID) AS ProductAmount

FROM Purchasing.ProductVendor AS pv

INNER JOIN Purchasing.Vendor AS v

ON pv.BusinessEntityID = v.BusinessEntityID

GROUP BY v.CreditRating

Лаб 4

1. SELECT p.Name

FROM Production.Product AS p

WHERE p.ProductID =

(SELECT TOP 1 sod.ProductID

FROM Sales.SalesOrderDetail AS sod

GROUP BY sod.ProductID

ORDER BY SUM(sod.OrderQty) DESC)

1. SELECT soh.CustomerID

FROM Sales.SalesOrderHeader AS soh

WHERE soh.SalesOrderID =

(SELECT TOP 1 sod.SalesOrderID

FROM Sales.SalesOrderDetail AS sod

GROUP BY sod.SalesOrderID

ORDER BY SUM(sod.UnitPrice \* sod.OrderQty) DESC)

1. SELECT sod.ProductID

FROM Sales.SalesOrderDetail AS sod

JOIN Sales.SalesOrderHeader AS soh

ON sod.SalesOrderID = soh.SalesOrderID

GROUP BY sod.ProductID

HAVING Count(DISTINCT soh.CustomerID) = 1

1. SELECT p1.ProductID

FROM Production.Product AS p1

WHERE p1.ListPrice > (SELECT AVG(p2.ListPrice)

FROM Production.Product AS p2

WHERE p1.ProductSubcategoryID = p2.ProductSubcategoryID)

1. SELECT sod.ProductID

FROM Sales.SalesOrderDetail AS sod

JOIN Sales.SalesOrderHeader AS soh

ON sod.SalesOrderID = soh.SalesOrderID

WHERE soh.CustomerID IN

(SELECT soh2.CustomerID

FROM Sales.SalesOrderHeader AS soh2

JOIN Sales.SalesOrderDetail AS sod2

ON sod2.SalesOrderID = soh2.SalesOrderID

JOIN Production.Product AS p

ON p.ProductID = sod2.ProductID

WHERE sod2.ProductID NOT IN (

SELECT sod3.ProductID

FROM Sales.SalesOrderDetail AS sod3

JOIN Sales.SalesOrderHeader AS soh3

ON sod3.SalesOrderID = soh3.SalesOrderID

WHERE soh3.CustomerID IN (

SELECT soh4.CustomerID

FROM Sales.SalesOrderHeader AS soh4

JOIN Sales.SalesOrderDetail AS sod4

ON sod4.SalesOrderID = soh4.SalesOrderID

JOIN Production.Product AS p2

ON p2.ProductID = sod4.ProductID

GROUP BY soh4.CustomerID

HAVING Count(DISTINCT p2.Color) = 2

)

)

GROUP BY soh2.CustomerID

HAVING Count(DISTINCT p.Color) = 1

)

GROUP BY sod.ProductID

HAVING Count(DISTINCT soh.CustomerID) > 1

1. (?) SELECT DISTINCT sod.ProductID

FROM Sales.SalesOrderDetail AS sod

JOIN Sales.SalesOrderHeader AS soh

ON sod.SalesOrderID = soh.SalesOrderID

WHERE soh.CustomerID IN

(SELECT soh2.CustomerID

FROM Sales.SalesOrderDetail AS sod2

JOIN Sales.SalesOrderHeader AS soh2

ON sod2.SalesOrderID = soh2.SalesOrderID

GROUP BY soh2.CustomerID

HAVING sod.ProductID IN ALL (soh2.SalesOrderID)

)

Лаб 5

1. Найти среднее количество покупок на чек для каждого покупателя

SELECT CustomerID, avg(countPurchases) AS averageCountPurchases

FROM Sales.SalesOrderHeader AS soh,

(SELECT SalesOrderID, count(DISTINCT ProductID) AS countPurchases

FROM Sales.SalesOrderDetail AS sod

GROUP BY SalesOrderID) AS t1

WHERE t1.SalesOrderID = soh.SalesOrderID

GROUP BY CustomerID

1. Найти для каждого продукта и каждого покупателя соотношение количества

фактов покупки данного товара данным покупателем к общему количеству

фактов покупки товаров данным покупателем

SELECT t1.ProductID, t1.CustomerID, countProduct\*1.0/countAll

FROM (SELECT ProductID, CustomerID, Count(\*) AS countProduct

FROM Sales.SalesOrderHeader AS soh

JOIN Sales.SalesOrderDetail AS sod

ON soh.SalesOrderID = sod.SalesOrderID

GROUP BY ProductID, CustomerID) AS t1,

(SELECT CustomerID, Count(\*) AS countAll

FROM Sales.SalesOrderHeader AS soh

JOIN Sales.SalesOrderDetail AS sod

ON soh.SalesOrderID = sod.SalesOrderID

GROUP BY CustomerID) AS t2

WHERE t1.CustomerID = t2.CustomerID

3. Вывести на экран следующую информацию: Название продукта, Общее

количество фактов покупки этого продукта, Общее количество покупателей

этого продукта

SELECT t1.ProductID, CountPurchasesProduct, CountCustomersProduct

FROM (SELECT ProductID, Count(\*) as CountPurchasesProduct

FROM Sales.SalesOrderDetail AS sod

GROUP BY ProductID) AS t1,

(SELECT ProductID, Count(DISTINCT soh.CustomerID) as CountCustomersProduct

FROM Sales.SalesOrderDetail AS sod

JOIN Sales.SalesOrderHeader AS soh

ON sod.SalesOrderID = soh.SalesOrderID

GROUP BY ProductID) AS t2

WHERE t1.ProductID = t2.ProductID

4. Вывести для каждого покупателя информацию о максимальной и минимальной

стоимости одной покупки, чеке, в виде таблицы: номер покупателя,

максимальная сумма, минимальная сумма.

SELECT t1.CustomerID, MaxPurchase, MinPurchase

FROM (SELECT CustomerID, max(OrderQty\*UnitPrice) AS MaxPurchase

FROM Sales.SalesOrderHeader AS soh

JOIN Sales.SalesOrderDetail AS sod

ON soh.SalesOrderID = sod.SalesOrderID

GROUP BY CustomerID) AS t1,

(SELECT CustomerID, min(OrderQty\*UnitPrice) AS MinPurchase

FROM Sales.SalesOrderHeader AS soh

JOIN Sales.SalesOrderDetail AS sod

ON soh.SalesOrderID = sod.SalesOrderID

GROUP BY CustomerID) AS t2

WHERE t1.CustomerID = t2.CustomerID

5. Найти номера покупателей, у которых не было ни одной пары чеков с

одинаковым количеством наименований товаров.

SELECT CustomerID

FROM (SELECT CustomerID, sod.SalesOrderID, CHECKSUM\_AGG(sod.ProductID) AS CheckSumOrder

FROM Sales.SalesOrderHeader AS soh

JOIN Sales.SalesOrderDetail AS sod

ON soh.SalesOrderID = sod.SalesOrderID

GROUP BY CustomerID, sod.SalesOrderID) AS t1

GROUP BY CustomerID

HAVING COUNT(CheckSumOrder) = COUNT(DISTINCT CheckSumOrder)

6. Найти номера покупателей, у которых все купленные ими товары были

куплены как минимум дважды, т.е. на два разных чека.

SELECT CustomerID

FROM (SELECT CustomerID, ProductID, count(\*) AS countProducts

FROM Sales.SalesOrderHeader AS soh

JOIN Sales.SalesOrderDetail AS sod

ON soh.SalesOrderID = sod.SalesOrderID

GROUP BY CustomerID, ProductID) AS t1

GROUP BY CustomerID

HAVING min(countProducts)>=2

Лаб 6

1. Найти долю продаж каждого продукта (цена продукта \* количество продукта), на каждый чек, в денежном выражении.

SELECT ProductID, SalesOrderID, OrderQty\*ProductID\*1.0/SUM(OrderQty\*ProductID)

OVER (PARTITION BY SalesOrderID) as amount

FROM Sales.SalesOrderDetail AS sod

2. Вывести на экран список продуктов, их стоимость, а также разницу между стоимостью этого продукта и стоимостью самого дешевого продукта в той же подкатегории, к которой относится продукт.

SELECT p.ProductID, UnitPrice, UnitPrice-MIN(UnitPrice)

OVER (PARTITION BY p.ProductSubcategoryID) AS diff

FROM Sales.SalesOrderDetail AS sod

JOIN Production.Product AS p

ON sod.ProductID = p.ProductID

JOIN Production.ProductSubcategory AS ps

ON p.ProductSubcategoryID = ps.ProductSubcategoryID

3. Вывести три колонки: номер покупателя, номер чека покупателя (отсортированный по возрастанию даты чека) и искусственно введенный порядковый номер текущего чека, начиная с 1, для каждого покупателя.

SELECT CustomerID, SalesOrderID, ROW\_NUMBER()

OVER (PARTITION BY CustomerID ORDER BY OrderDate) AS RowNumber

FROM Sales.SalesOrderHeader AS soh

4. Вывести номера продуктов, таких что их цена выше средней цены продукта в подкатегории, к которой относится продукт. Запрос реализовать двумя способами. В одном из решений допускается использование обобщенного табличного выражения.

SELECT ProductID

FROM (SELECT ProductID, ListPrice, AVG(ListPrice)

OVER (PARTITION BY ProductSubcategoryID) AS averageInSubcategory

FROM Production.Product AS p) AS tmp

WHERE ListPrice > averageInSubcategory

5. Вывести на экран номер продукта, название продукта, а также информацию о

среднем количестве этого продукта, приходящихся на три последних по дате

чека, в которых был этот продукт.

WITH LastOrders AS (

SELECT ProductID, OrderQty, ROW\_NUMBER()

OVER (PARTITION BY ProductID ORDER BY OrderDate DESC) AS OrderNumber

FROM Sales.SalesOrderDetail AS sod

JOIN Sales.SalesOrderHeader AS soh

ON sod.SalesOrderID = soh.SalesOrderID

)

SELECT p.ProductID, Name, AVG(OrderQty\*1.0) AS AverageAmount

FROM Production.Product AS p

JOIN LastOrders AS lo

ON p.ProductID = lo.ProductID AND lo.OrderNumber <= 3

GROUP BY p.ProductID, Name