```sql
USE AdventureWorks2019;

-- 1
SELECT COUNT(ProductID) AS product_count
FROM Production.Product;

-- 2
SELECT COUNT(ProductID) AS CountedProducts
FROM Production.Product
WHERE ProductSubcategoryID IS NOT NULL

-- 3
SELECT ProductSubcategoryID AS
ProductSubcategoryID,COUNT(ProductID) AS CountedProducts
FROM Production.Product
WHERE ProductSubcategoryID IS NOT NULL
GROUP BY ProductSubcategoryID;

-- 4
SELECT ProductSubcategoryID AS
ProductSubcategoryID,COUNT(ProductID) AS CountedProducts
FROM Production.Product
WHERE ProductSubcategoryID IS NULL
GROUP BY ProductSubcategoryID;

-- 5
SELECT SUM(Quantity) as SumQuantity
FROM Production.ProductInventory

-- 6
SELECT ProductID, SUM(Quantity) as TheSum
FROM Production.ProductInventory
WHERE LocationID = 40
GROUP BY ProductID
HAVING SUM(Quantity) < 100;

-- 7
SELECT Shelf, ProductID, SUM(Quantity) AS TheSum
FROM Production.ProductInventory
WHERE LocationID = 40
GROUP BY Shelf, ProductID
HAVING SUM(Quantity) < 100;

-- 8
SELECT AVG(Quantity) AS AverageQuantity
FROM Production.ProductInventory
WHERE LocationID = 10
```

```sql
-- 9
SELECT ProductID, Shelf, AVG(Quantity) AS TheAvg
FROM Production.ProductInventory
GROUP BY Shelf, ProductID

-- 10
SELECT ProductID, Shelf, AVG(Quantity) AS TheAvg
FROM Production.ProductInventory
WHERE NOT Shelf = 'N/A'
GROUP BY Shelf, ProductID

-- 11
SELECT Color, Class, COUNT(*) AS TheCount, AVG(ListPrice) AS
AvgPrice
FROM Production.Product
WHERE Color IS NOT NULL AND Class IS NOT NULL
GROUP BY Color, Class

-- 12
SELECT c.Name as Country, s.Name as Province
FROM Person.CountryRegion AS c INNER JOIN Person.StateProvince
as s
ON c.CountryRegionCode = s.CountryRegionCode

-- 13
WITH CountryCTE
AS(
    SELECT Name, CountryRegionCode
    FROM Person.CountryRegion
    WHERE Name = 'Germany' or Name = 'Canada'
)
SELECT c.Name as Country, s.Name AS Province
FROM CountryCTE AS c INNER JOIN Person.StateProvince as s
ON c.CountryRegionCode = s.CountryRegionCode

USE Northwind
-- 14
SELECT DISTINCT ProductName
FROM Products
JOIN [Order Details] ON Products.ProductID = [Order
Details].ProductID
JOIN Orders ON [Order Details].OrderID = Orders.OrderID
WHERE Orders.OrderDate >= DATEADD(year, -25, GETDATE());

-- 15
SELECT TOP 5 ShipPostalCode AS ZipCode, SUM(Quantity) AS
```

```sql
TotalQuantity
FROM [Order Details] as od
JOIN Orders ON od.OrderID = Orders.OrderID
GROUP BY ShipPostalCode
ORDER BY TotalQuantity DESC;

-- 16
SELECT TOP 5 ShipPostalCode as ZipCode, SUM(Quantity) AS
TotalQuantity
FROM [Order Details] as od INNER JOIN Orders
ON od.OrderID = Orders.OrderID
WHERE Orders.OrderDate >= DATEADD(YEAR, -25, GETDATE())
GROUP by ShipPostalCode
ORDER BY TotalQuantity DESC

-- 17
SELECT City, COUNT(CustomerID) as CountedCustomer
FROM Customers
GROUP BY City

-- 18
SELECT City, COUNT(CustomerID) as CountedCustomer
FROM Customers
GROUP BY City
HAVING COUNT(CustomerID) > 2

-- 19
SELECT c.CompanyName as CustomerName, o.OrderDate
FROM Orders o
INNER JOIN Customers c
ON o.CustomerID = c.CustomerID
WHERE o.OrderDate > '1998-01-01';

-- 20
SELECT c.CompanyName as CustomerName, MAX(o.OrderDate) AS
MostRecentOrderDate
FROM Orders o
INNER JOIN Customers c
ON o.CustomerID = c.CustomerID
GROUP BY c.CompanyName
ORDER BY MostRecentOrderDate DESC

-- 21
SELECT c.CompanyName, COUNT(DISTINCT od.ProductID) AS
NumProductsBought
FROM Customers as c
JOIN Orders as o ON c.CustomerID = o.CustomerID
```

```sql
JOIN [Order Details] as od ON o.OrderID = od.OrderID
GROUP BY c.CompanyName;

-- 22
SELECT c.CustomerID, COUNT(od.ProductID) AS NumProductsBought
FROM Customers as c
INNER JOIN Orders o ON c.CustomerID = o.CustomerID
INNER JOIN [Order Details] as od ON o.OrderID = od.OrderID
GROUP BY c.CustomerID
HAVING COUNT(od.ProductID) > 100;

-- 23
SELECT su.CompanyName AS "Supplier Company Name", sh.CompanyName
AS "Shipping Company Name"
FROM Suppliers su
CROSS JOIN Shippers sh;

-- 24
SELECT o.OrderDate, p.ProductName
FROM Orders o
JOIN [Order Details] od ON o.OrderID = od.OrderID
JOIN Products p ON od.ProductID = p.ProductID
ORDER BY o.OrderDate, p.ProductName;

-- 25
SELECT e1.EmployeeID, e1.Title AS JobTitle, e2.EmployeeID
FROM Employees e1
JOIN Employees e2 ON e1.Title = e2.Title AND NOT e1.EmployeeID =
e2.EmployeeID
ORDER BY e1.Title, e1.EmployeeID, e2.EmployeeID;

-- 26
SELECT m.EmployeeID, m.FirstName, m.LastName, COUNT(*) AS
EmployeeCount
FROM Employees m
INNER JOIN Employees e ON e.ReportsTo = m.EmployeeID
WHERE m.Title LIKE '%Manager%'
GROUP BY m.EmployeeID, m.FirstName, m.LastName
HAVING COUNT(*) > 2
ORDER BY m.LastName, m.FirstName;

-- 27
SELECT City, CompanyName AS Name, ContactName, 'Customer' AS
Type
FROM Customers
UNION
SELECT City, CompanyName AS Name, ContactName, 'Supplier' AS
```

```
Type
FROM Suppliers
ORDER BY City, Type, CompanyName;
```