

```
--PRAGYNASMITA SAHOO
```

```
USE AdventureWorks2019;
```

```
SELECT * FROM INFORMATION_SCHEMA.TABLES;
```

```
--1. List of all customers
```

```
SELECT * FROM person.person;
```

```
--2. list of all customers where company name ending in N
```

```
SELECT * FROM Person.AddressType where Name LIKE '%n';
```

```
--3. list of all customers who live in Berlin or London
```

```
SELECT * FROM Person.Address where City = 'berlin' or City = 'london'
```

```
--4. list of all customers who live in UK or USA
```

```
SELECT * FROM Person.CountryRegion where CountryRegionCode = 'USA' or  
CountryRegionCode = 'UK';
```

```
--5. list of all products sorted by product name
```

```
SELECT * FROM Sales.Store order by Name
```

```
--6. list of all products where product name starts with an A
```

```
SELECT * FROM Sales.Store where Name like 'A%'
```

```
--7. List of customers who ever placed an order
```

```
SELECT * FROM Sales.Store
```

```
--8. list of Customers who live in London and have bought chai
```

```
SELECT DISTINCT
```

```
c.CustomerID,
```

```
c.PersonID,
```

```
c.StoreID,
```

```
c.TerritoryID,
```

```
c.AccountNumber
```

```
FROM
```

```
Sales.Customer AS c
```

```
JOIN
```

```
Sales.SalesOrderHeader AS soh ON c.CustomerID = soh.CustomerID
```

```
JOIN
```

```
Sales.SalesOrderDetail AS sod ON soh.SalesOrderID = sod.SalesOrderID
```

```
JOIN
```

```
Production.Product AS p ON sod.ProductID = p.ProductID
```

```
JOIN
Person.Address AS a ON c.CustomerID = a.AddressID
WHERE
a.City = 'London'
AND p.Name = 'Chai';
```

--9. List of customers who never place an order

```
SELECT
c.CustomerID,
c.PersonID,
c.StoreID,
c.TerritoryID,
c.AccountNumber
FROM
Sales.Customer AS c
LEFT JOIN
Sales.SalesOrderHeader AS soh ON c.CustomerID = soh.CustomerID
WHERE
soh.SalesOrderID IS NULL;
```

--10. List of customers who ordered Tofu

```
SELECT DISTINCT
c.CustomerID,
c.PersonID,
c.StoreID,
c.TerritoryID,
c.AccountNumber
FROM
Sales.Customer AS c
JOIN
Sales.SalesOrderHeader AS soh ON c.CustomerID = soh.CustomerID
JOIN
Sales.SalesOrderDetail AS sod ON soh.SalesOrderID = sod.SalesOrderID
JOIN
Production.Product AS p ON sod.ProductID = p.ProductID
WHERE
p.Name = 'Tofu';
```

--11. Details of first order of the system

```
SELECT SOH.*
FROM sales.SalesOrderHeader AS SOH
JOIN (
    SELECT MIN(OrderDate) AS MinOrderDate
```

```
FROM sales.SalesOrderHeader
) AS MinDates
ON SOH.OrderDate = MinDates.MinOrderDate;
```

--12. Find the details of most expensive order date

```
SELECT * FROM sales.SalesOrderHeader where SubTotal=(SELECT MAX(SubTotal) FROM
sales.SalesOrderHeader);
```

--13. For each order get the OrderID and Average quantity of items in that order

```
SELECT SalesOrderID , AVG(OrderQty) FROM sales.SalesOrderDetail GROUP BY
SalesOrderID;
```

--14. For each order get the orderID, minimum quantity and maximum quantity for that order

```
SELECT ProductID,MaxOrderQty,MinOrderQty FROM Purchasing.ProductVendor ;
```

--15. Get a list of all managers and total number of employees who report to them.

```
SELECT
    M.BusinessEntityID AS ManagerID,
    M.JobTitle AS ManagerJobTitle,
    COUNT(E.BusinessEntityID) AS NumberOfEmployees
FROM
    HumanResources.Employee AS M
JOIN
    HumanResources.Employee AS E ON M.BusinessEntityID = E.BusinessEntityID
GROUP BY
    M.BusinessEntityID,
    M.JobTitle;
```

--16. Get the OrderID and the total quantity for each order that has a total quantity of greater than 300

```
SELECT ProductID, OnorderQty FROM Purchasing.ProductVendor where OnOrderQty > 300;
```

--17. list of all orders placed on or after 1996/12/31

```
SELECT * FROM Purchasing.PurchaseOrderHeader where OrderDate >= '1996-12-31';
```

--18. list of all orders shipped to Canada

```
SELECT * FROM Person.Address where city = 'Canada';
```

--19. list of all orders with order total > 200

```
SELECT SalesOrderID, OrderQty FROM Sales.SalesOrderDetail where OrderQty > 200;
```

--20. List of countries and sales made in each country

```
SELECT * FROM [Sales].[SalesOrderHeader]
```

--21. List of Customer ContactName and number of orders they placed

```
SELECT * FROM Sales.SalesOrderDetail
```

--22. List of customer contactnames who have placed more than 3 orders

```
SELECT SalesOrderID , OrderQty FROM Sales.SalesOrderDetail where OrderQty>3
```

--23. List of discontinued products which were ordered between 1/1/1997 and 1/1/1998



```
SELECT
    p.ProductID,
    p.Name AS ProductName,
    p.DiscontinuedDate
FROM
    Sales.SalesOrderDetail AS sod
JOIN
    Sales.SalesOrderHeader AS soh ON sod.SalesOrderID = soh.SalesOrderID
JOIN
    Production.Product AS p ON sod.ProductID = p.ProductID
WHERE
    soh.OrderDate BETWEEN '1997-01-01' AND '1998-01-01'
    AND p.DiscontinuedDate IS NOT NULL
    AND p.DiscontinuedDate <= '1998-01-01'
ORDER BY
    p.ProductID;
```

--24. List of employee firstname, lastName, supervisor FirstName, LastName

```
WITH EmployeeHierarchy AS (
    SELECT
        E.BusinessEntityID,
        E.OrganizationNode.GetAncestor(1) AS SupervisorNode
    FROM
        HumanResources.Employee AS E
)
SELECT
    E.BusinessEntityID,
    P.FirstName,
    P.LastName,
    EH.SupervisorNode AS SupervisorNode,
    PM.FirstName AS SupervisorFirstName,
```

```
PM.LastName AS SupervisorLastName
FROM
HumanResources.Employee AS E
JOIN
Person.Person AS P ON E.BusinessEntityID = P.BusinessEntityID
LEFT JOIN
EmployeeHierarchy AS EH ON E.OrganizationNode = EH.SupervisorNode
LEFT JOIN
HumanResources.Employee AS M ON EH.SupervisorNode = M.OrganizationNode
LEFT JOIN
Person.Person AS PM ON M.BusinessEntityID = PM.BusinessEntityID;
```

--25. List of Employees id and total sale conducted by employee

```
SELECT SalesPersonID,SubTotal FROM Sales.SalesOrderHeader;
```

--26. list of employees whose FirstName contains character a

```
SELECT FirstName FROM Person.Person where FirstName like '%a%'
```

--27. List of managers who have more than four people reporting to them.

```
WITH EmployeeHierarchy AS (
    SELECT
        E.BusinessEntityID,
        E.OrganizationLevel
    FROM
        HumanResources.Employee AS E
)
SELECT
    EH.BusinessEntityID AS ManagerID,
    COUNT(*) AS NumberOfEmployees
FROM
    EmployeeHierarchy AS EH
GROUP BY
    EH.BusinessEntityID
HAVING
    COUNT(*) > 4;
```

--28. List of Orders and ProductNames

```
SELECT BusinessEntityID,Name FROM Sales.Store
```

--29. List of orders place by the best customer

```
SELECT * FROM sales.SalesOrderHeader where SubTotal=(SELECT MAX(SubTotal) FROM
sales.SalesOrderHeader)
```



--30. List of orders placed by customers who do not have a Fax number

```
SELECT * FROM Person.PersonPhone;
```

--31. List of Postal codes where the product Tofu was shipped

```
SELECT
    A.PostalCode
FROM
    Sales.SalesOrderDetail AS SOD
JOIN
    Production.Product AS P ON SOD.ProductID = P.ProductID
JOIN
    Sales.SalesOrderHeader AS SOH ON SOD.SalesOrderID = SOH.SalesOrderID
JOIN
    Person.Address AS A ON SOH.ShipToAddressID = A.AddressID
WHERE
    P.Name = 'Tofu'
GROUP BY
    A.PostalCode;
```

--32. List of product Names that were shipped to France

```
SELECT * FROM Person.Address where city = 'France';
```

--33. List of ProductNames and Categories for the supplier 'Specialty Biscuits, Ltd.

```
SELECT name FROM Sales.Store where name = 'specialty biscuit';
```

--34. List of products that were never ordered

```
SELECT * FROM sales.SalesOrderDetail where OrderQty<0
```

--35. List of products where units in stock is less than 10 and units on order are 0.

```
SELECT
    P.Name AS ProductName,
    P.ProductNumber,
    P.SafetyStockLevel,
    P.ReorderPoint
FROM
    Production.Product AS P
WHERE
    P.SafetyStockLevel < 10 AND P.ReorderPoint = 0;
```

--36. List of top 10 countries by sales

```
SELECT TOP 10
    ST.Name AS Country,
    SUM(SOH.TotalDue) AS TotalSales
FROM
    Sales.SalesOrderHeader AS SOH
JOIN
    Sales.SalesTerritory AS ST ON SOH.TerritoryID = ST.TerritoryID
GROUP BY
    ST.Name
ORDER BY
    TotalSales DESC;
```

--37. Number of orders each employee has taken for customers with CustomerIDs between A and AO

```
SELECT e.BusinessEntityID, COUNT(o.SalesOrderID) AS NumberOfOrders
FROM HumanResources.Employee e
JOIN Sales.SalesOrderHeader o ON e.BusinessEntityID = o.SalesPersonID
JOIN Sales.Customer c ON o.CustomerID = c.CustomerID
WHERE TRY_CONVERT(int, c.CustomerID) BETWEEN TRY_CONVERT(int, 'A') AND TRY_CONVERT
(int, 'AO')
GROUP BY e.BusinessEntityID;
```

--38. Orderdate of most expensive order

```
SELECT * FROM sales.SalesOrderHeader where SubTotal=(SELECT MAX(SubTotal) FROM
sales.SalesOrderHeader)
```

--39. Product name and total revenue FROM that product

```
SELECT p.Name, SUM(od.UnitPrice * od.OrderQty) AS TotalRevenue
FROM Production.Product p
JOIN Sales.SalesOrderDetail od ON p.ProductID = od.ProductID
GROUP BY p.Name;
```

--40. Supplierid and number of products offered

```
SELECT SalesOrderID, OrderQty FROM Sales.SalesOrderDetail
```

--41. Top ten customers based on their business

```
SELECT top 10 * FROM sales.SalesOrderDetail ORDER BY OrderQty DESC;
```

--42. What is the total revenue of the company.

```
SELECT SalesOrderID, sum(SubTotal)
FROM Sales.SalesOrderHeader
group by SalesOrderID with rollup
```