use AdventureWorks2022

select count(\*) from HumanResources.Employee where Gender='F';

-- 1. Find the employee having salaried flag as 1

select \* from HumanResources.Employee where SalariedFlag='1';

-- 2. Find all the employees havinf vaccation hours more than 70

select \* from HumanResources.Employee where VacationHours>'70';

-- 3. vacation hour more than 70 but less than 90

select \* from HumanResources.Employee where VacationHours>'70' and VacationHours <90;

-- 4. Find all jobs having title as designer

select \* from HumanResources.Employee where JobTitle like('%Designer%');

-- 5. total emp worked as technician

select \* from HumanResources.Employee where JobTitle like('%Technician%')

-- 6. nationl id no,jobtitle,marritial status ,gender for all under marketing job title

select NationalIDNumber,JobTitle,MaritalStatus,Gender from HumanResources.Employee where JobTitle like('%Marketing%')

-- 7. find all unique maritl status

select distinct MaritalStatus from HumanResources.Employee

-- 8. find the max vacaction hours

select max(VacationHours) from HumanResources.Employee

-- 9. find less sick leaves

select MIN(SickLeaveHours) from HumanResources.Employee

select \* from HumanResources.Department

select \* from HumanResources.EmployeeDepartmentHistory

-- 10.all emp from production dpt

select \* from HumanResources.Department where Name='Production'

select \* from HumanResources.Employee

where BusinessEntityID in

(select BusinessEntityID

from HumanResources.EmployeeDepartmentHistory

where DepartmentID=7)

-- 11. all dept under research and dev

select \* from HumanResources.Department

select \* from HumanResources.Department where GroupName='Research and Development'

select \* from HumanResources.Employee

-- 12. all emp under research and dev

select \* from HumanResources.Department where GroupName='Research and Development'

select count(\*) from HumanResources.Employee

where BusinessEntityID in

(select BusinessEntityID

from HumanResources.EmployeeDepartmentHistory

where DepartmentID in(

select DepartmentID

from HumanResources.Department

where GroupName='Research and Development'))

-- 13. find all employees who work in day shift

select count(\*) from HumanResources.Employee where BusinessEntityID

in(select BusinessEntityID from HumanResources.EmployeeDepartmentHistory where ShiftID

in(select ShiftID from HumanResources.Shift where Name='Day'))

-- 14. count of emp having payfreq is 1

select count(\*) from HumanResources.Employee where BusinessEntityID

in(select BusinessEntityID from HumanResources.EmployeePayHistory where PayFrequency=1)

-- 15. all jobID which are not placed

select \* from HumanResources.Employee where BusinessEntityID

in(select JobCandidateID from HumanResources.JobCandidate where BusinessEntityID IS NULL)

-- 16. address of employee

select \* from Person.Address

select \* from Person.BusinessEntityAddress

select \* from HumanResources.Employee

select \* from HumanResources.Department

select \* from Person.Person

select \* from HumanResources.EmployeeDepartmentHistory

select \* from Person.Address where AddressID

in(select AddressID from Person.BusinessEntityAddress where BusinessEntityID

in(select BusinessEntityID from HumanResources.Employee))

-- 17. find name of emp working in R&D

select FirstName from Person.Person where BusinessEntityID

in(select BusinessEntityID from HumanResources.EmployeeDepartmentHistory where DepartmentID

in(select DepartmentID from HumanResources.Department where GroupName='Research and development'))

---------------------------------------------------------------------------------------------------------------------------

---------------------------------------------------------------------------------------------------------------------------

---------------------------------------------------------------------------------------------------------------------------

-- Corelated subquery

------------------

select BusinessEntityID,NationalIDNumber,JobTitle,

(select firstname from Person.Person p where p.BusinessEntityID=e.BusinessEntityID)fname

from HumanResources.Employee e

-- 1. add personal details of emp mname,lastname

select BusinessEntityID,NationalIDNumber,JobTitle,

(select firstname from Person.Person p where p.BusinessEntityID=e.BusinessEntityID)fname,

(select MiddleName from Person.Person p where p.BusinessEntityID=e.BusinessEntityID)mname,

(select LastName from Person.Person p where p.BusinessEntityID=e.BusinessEntityID)lname

from HumanResources.Employee e

-- 2. Concat

select concat(FirstName,' ',MiddleName,' ',LastName)as Full\_Name from Person.Person

select BusinessEntityID,NationalIDNumber,JobTitle,

(select concat(firstname,MiddleName,LastName) from Person.Person p where p.BusinessEntityID=e.BusinessEntityID)Full\_name

from HumanResources.Employee e

-- 3. word separater concat\_ws()

select BusinessEntityID,NationalIDNumber,JobTitle,

(select concat\_ws(' - ',firstname,MiddleName,LastName) from Person.Person p where p.BusinessEntityID=e.BusinessEntityID)Full\_name

from HumanResources.Employee e

-- 4. display national\_id ,department name,department group

select(select concat\_ws(' - ',FirstName,LastName) from Person.Person p where p.BusinessEntityID=ed. BusinessEntityID)Full\_Name,

(select NationalIDNumber from HumanResources.Employee e where e.BusinessEntityID=ed.BusinessEntityID) Nationl\_Id,

(select concat\_ws(' - ',Name,GroupName) from HumanResources.Department d where d.DepartmentID=ed.DepartmentID)Dept

from HumanResources.EmployeeDepartmentHistory ed

select \* from HumanResources.EmployeeDepartmentHistory --be id, d id

select \* from HumanResources.Employee --beid , nid

select \* from HumanResources.Department --department id ,Name,groupname

-- 5. display first\_name,lastname,department,shift time

select \* from Person.Person

select \* from HumanResources.Department

select \* from HumanResources.Shift

select \* from HumanResources.EmployeeDepartmentHistory --be id, d id

select(select concat\_ws(' - ',FirstName,LastName) from Person.Person p where p.BusinessEntityID=ed.BusinessEntityID)Full\_Name,

(select Name from HumanResources.Department d where d.DepartmentID=ed.DepartmentID) DeptName,

(select StartTime from HumanResources.Shift s where s.ShiftID=ed.ShiftID)shift\_time

from HumanResources.EmployeeDepartmentHistory ed

--display product name and product review based on product schema

-- 6. find emp\_name,job title,credit card details,when it expire

select \* from Person.Person -- BEID,Name

select \* from HumanResources.Employee --job titke,BEID

select \* from Sales.CreditCard -- CreditID,Exp

select \* from Sales.PersonCreditCard -- BEID , CreditID

select

(select concat\_ws(' ',FirstName,LastName) from Person.Person p where p.BusinessEntityID=pc.BusinessEntityID)Full\_Name,

(select JobTitle from HumanResources.Employee e where e.BusinessEntityID=pc.BusinessEntityID)JobTitle,(select concat\_ws(' - ',ExpMonth,ExpYear)from Sales.CreditCard cc where cc.CreditCardID=pc. CreditCardID)credit\_details

from Sales.PersonCreditCard pc

-- 7. disp records from currency rate from usd to aud

select \* from Sales.CurrencyRate

select \* from Sales.Currency

select \* from Sales.CurrencyRate where FromCurrencyCode='USD' and ToCurrencyCode='AUD'

-- 8. disp emp name,teritory name,group,sales last yr,sales quota,bonus

select \* from Sales.SalesPerson -- ttID,BEID, Bonus,quota

select \* from Sales.SalesTerritory -- ttID,name,group,sales LY

select \* from Person.Person-- BEID, fname

select

(select concat(FirstName,'',LastName) from Person.Person p where p.BusinessEntityID=sp. BusinessEntityID)FullName,

(select Name from Sales.SalesTerritory st where st. TerritoryID=sp. TerritoryID)TerritoryName,

(select [Group] from Sales.SalesTerritory st where st. TerritoryID=sp. TerritoryID)GroupName,

(select SalesLastYear from Sales.SalesTerritory st where st. TerritoryID=sp. TerritoryID)SalesLY,

SalesQuota,Bonus

from Sales.SalesPerson sp

-- 9. disp emp name,teritory name,group,sales last yr,sales quota,bonus from germany and UK

select

(select concat(FirstName,'',LastName) from Person.Person p where p.BusinessEntityID=sp. BusinessEntityID)FullName,

(select Name from Sales.SalesTerritory st where st. TerritoryID=sp. TerritoryID)TerritoryName,

(select [Group] from Sales.SalesTerritory st where st. TerritoryID=sp. TerritoryID)GroupName,

(select SalesLastYear from Sales.SalesTerritory st where st.TerritoryID=sp.TerritoryID)SalesLY,

SalesQuota,Bonus

from Sales.SalesPerson sp

where sp.TerritoryID in(select TerritoryID from Sales.SalesTerritory st where Name in ('United kingdom','Germany'))

-- 10. find all emp who worked in all North america continent

select

(select concat(FirstName,'',LastName) from Person.Person p where p.BusinessEntityID=sp.BusinessEntityID)FullName,

(select [Group] from Sales.SalesTerritory st where st.TerritoryID=sp.TerritoryID)GroupName

from Sales.SalesPerson sp

where sp.TerritoryID in(select TerritoryID from Sales.SalesTerritory st where [Group]='North America')

---------------------------------------------------------------------------------------------------------------------------

---------------------------------------------------------------------------------------------------------------------------

---------------------------------------------------------------------------------------------------------------------------

-- Questionaries

--1) find the average currency rate conversion from USD to Algerian Dinar and Australian Doller

select \* from Sales.CurrencyRate

select \* from Sales.Currency

select AverageRate from Sales.CurrencyRate where FromCurrencyCode='USD' and ToCurrencyCode='AUD' or ToCurrencyCode='DZD'

-- 2)Find the products having offer on it and display product name , safety Stock Level,

--Listprice, and product model id,type of discount, percentage of discount, offer start date and offer end date

select \* from Sales.SpecialOffer --special offer id,discount type,discount percent,offer S\_dates E\_dates,description.

select \* from Sales.SpecialOfferProduct--special offer id,product id

select \* from Production.Product --prodcut id,product name,safety stock,listprice,product model id

select p.Name,so.Description,so.DiscountPct,so.StartDate,so.EndDate,so.Type,p.SafetyStockLevel,p.ListPrice,p.ProductModelID from Sales.SpecialOfferProduct spo,sales.SpecialOffer so,Production.Product p

where spo.SpecialOfferID=so.SpecialOfferID and spo.ProductID=p.ProductID and so.DiscountPct>0.00

--3)create view to display Product name and Product review

---------------------------------------------------------------------------------------------------------------------------

---------------------------------------------------------------------------------------------------------------------------

---------------------------------------------------------------------------------------------------------------------------

-- Joins

---------------------------------------------------------------------------------------------------------------------------

use AdventureWorks2022

select \* from HumanResources.Department

select \* from HumanResources.EmployeeDepartmentHistory

select BusinessEntityID,ShiftID,GroupName,Name

from HumanResources.EmployeeDepartmentHistory as ed --whereever we want to apply condition put in join

full join HumanResources.Department as d --otherwise use in subquery

on ed.DepartmentID=d.DepartmentID --if we want the particular rec or data then use subquery

--if we want rec from multiple columns then use joins

-- Q1. find all records from production,production control ,executive,

-- who are having bithdate more than 1970,display 1st anme,

-- address details, jobtitle and department of this persons

select \* from HumanResources.Employee --BEID e

select \* from HumanResources.Department --DptID d

select \* from Person.Address --AddID a

select \* from Person.Person --BEID p

select \* from Person.BusinessEntityAddress --BEID,AddID

select \* from HumanResources.EmployeeDepartmentHistory --BEID,DeptID ed

select e.BirthDate,d.Name,e.JobTitle,

(select pa. AddressLine1 from Person.Address pa where pa. AddressID=

(select ba. AddressID from Person.BusinessEntityAddress ba where

ba. BusinessEntityID=e.BusinessEntityID)) Address,

(select FirstName from Person.Person p where p.BusinessEntityID=e.BusinessEntityID) Name

from HumanResources.EmployeeDepartmentHistory ed,

HumanResources.Employee e,

HumanResources.Department d

where ed. BusinessEntityID=e.BusinessEntityID

and ed. DepartmentID=d.DepartmentID

and BirthDate>='01-01-1970'

and d.Name in ('Production','Production Control' ,'Executive')

--Q2. find all product name scrapped more

select \* from Production.Product

select \* from Production.ScrapReason

select \* from Production.WorkOrder

select p.Name from Production.Product p

where p.ProductID=(select wo. ProductID from Production.WorkOrder wo

where wo. ScrappedQty=(select MAX(ScrappedQty) from Production.WorkOrder))

--Q3. find most frequent purchased product name

select \* from Production.Product

select \* from Production.WorkOrder

select \* from Purchasing.PurchaseOrderDetail

select \* from Production.ScrapReason

SELECT p.Name

FROM Production.Product p

WHERE p.ProductID = (

SELECT TOP 1 pd. ProductID

FROM Purchasing.PurchaseOrderDetail pd

GROUP BY pd. ProductID

ORDER BY SUM(pd. OrderQty) DESC

);

--Q3. which product require more inventory

select \* from Production.Product

select \* from Production.ProductInventory

SELECT p.Name

FROM Production.Product p

WHERE p.ProductID = (

SELECT TOP 1 pd. ProductID

FROM Production.ProductInventory pd

GROUP BY pd.ProductID

ORDER BY SUM(pd.Quantity) DESC

);

--Q4. most used ship mode

select \* from Purchasing.ShipMethod

select \* from Purchasing.PurchaseOrderHeader

select sm.Name from Purchasing.ShipMethod sm where ShipMethodID=

(select MAX(ShipMethodID) from Purchasing.PurchaseOrderHeader pd )

--Q5. which currency conv is more avg end of date rate

select \* from Purchasing.ShipMethod

select \* from [Sales].[SalesOrderHeader]

select \* from Purchasing.PurchaseOrderHeader

select \* from Sales.Currency

select \* from sales.CurrencyRate

select FromCurrencyCode,ToCurrencyCode,AverageRate from Sales.CurrencyRate cr where EndofDayRate=

(select max(EndOfDayRate) from Sales.CurrencyRate)

select top 1 cr.FromCurrencyCode,cr.ToCurrencyCode,avg(cr.EndOfDayRate) as average

from Sales.CurrencyRate cr

group by cr.FromCurrencyCode,cr.ToCurrencyCode

order by average desc

--Q6. which currency conversion has max value End of date rate

select top 1 cr.FromCurrencyCode,cr.ToCurrencyCode,max(cr.EndOfDayRate) as max\_conversion

from Sales.CurrencyRate cr

group by cr.FromCurrencyCode,cr.ToCurrencyCode

order by max\_conversion desc

--Q7. which currency conversion has least value End of date rate

select top 1 cr. FromCurrencyCode,cr. ToCurrencyCode,max(cr. EndOfDayRate) as least\_conversion

from Sales.CurrencyRate cr

group by cr. FromCurrencyCode,cr. ToCurrencyCode

order by least\_conversion asc

--Q8. which special offer having more duration

select\*from sales. SpecialOffer

select\*from sales. SpecialOfferProduct

select top 1 sop. ProductID,

(so. EndDate - so. StartDate) as diff

from sales. SpecialOffer so

full join Sales.SpecialOfferProduct sop

on so. SpecialOfferID = sop. SpecialOfferID group by sop. ProductID, so. Description, so. StartDate, so. EndDate

order by diff desc;

--Q9. which are those produ having more spcl offer produ

select\*from sales. SpecialOffer

select\*from sales. SpecialOfferProduct

select\*from Production.Product

select top 1 sop. ProductID,p.Name as Product\_Name,

COUNT(sop. ProductID) AS SpecialOfferCount

from sales. SpecialOfferProduct sop

full join Production.Product p

on sop. ProductID=p.ProductID

group by sop. ProductID,p.Name

order by SpecialOfferCount desc

---------------------------------------------------------------------------------------------------------------------------

---------------------------------------------------------------------------------------------------------------------------

---------------------------------------------------------------------------------------------------------------------------

--Q1.find the average currency rate conversion from USD to Algerian Dinar(DZD) and Australian Doller(AUD)

select \* from Sales.CurrencyRate

select \* from Sales.Currency

--select concat\_ws(' To ',FromCurrencyCode,ToCurrencyCode)as Currency\_Conversion,avg(AverageRate) as Average\_Currency\_Rate

--from sales. CurrencyRate

--where FromCurrencyCode='USD'

--and ToCurrencyCode in ('DZD','AUD')

--group by FromCurrencyCode,ToCurrencyCode

select FromCurrencyCode,ToCurrencyCode,AVG(AverageRate)as average\_rate

from Sales.CurrencyRate where FromCurrencyCode='USD'

and ToCurrencyCode in('DZD','AUD') group by FromCurrencyCode,ToCurrencyCode

--Q2. Find the products having offer on it and display product name , safety Stock Level, Listprice,

-- and product model id, type of discount, percentage of discount, offer start date and offer end date

select \* from Sales.SpecialOffer --special offer id, discount type, discount percent, offer S\_dates E\_dates ,description.

select \* from Sales.SalesOrderDetail --pid, special offer id

select \* from Production.Product --pid, product name, safety stock, listprice,product model id

select

(select p.ProductModelID from Production.Product p where p.ProductID=sop.ProductID)as Product\_ModelID,

(select p.Name from Production.Product p where p.ProductID=sop.ProductID)as Product\_Name,

(select p.SafetyStockLevel from Production.Product p where p.ProductID=sop.ProductID)as Safety\_Stock\_Level,

(select p.ListPrice from Production.Product p where p.ProductID=sop.ProductID)as List\_Price,

(select sp.DiscountPct from sales.SpecialOffer sp where sp.SpecialOfferID=sop.SpecialOfferID)as Percentage\_of\_discount,

(select sp.Type from sales.SpecialOffer sp where sp.SpecialOfferID=sop.SpecialOfferID)as Type\_of\_discount,

(select concat\_ws(' and ',sp.StartDate,sp.EndDate) from sales.SpecialOffer sp where sp.SpecialOfferID=sop.SpecialOfferID)as Start\_and\_end\_date

from sales.SpecialOfferProduct sop

--3. create view to display Product name and Product review

select \* from Production.ProductReview pr

select \* from Production.Product p

--go

--CREATE VIEW Customer AS

--Select pr.Comments,p.Name from Production.ProductReview pr INNER JOIN Production.Product p on pr.ProductID=p.ProductID

--select \* from Customer

--4. find out the vendor for product paint, Adjustable Race and Blade

Select \* from Purchasing.ProductVendor

Select \*from Purchasing.Vendor

Select \* from Production.Product

select

(select Name from Purchasing.Vendor v where v.BusinessEntityID=pv.BusinessEntityID)Vendor\_Name,

(select Name from Production.Product p where p.ProductID=pv.ProductID)Prod\_Name

from Purchasing.ProductVendor pv where ProductID in(select ProductID from Production.Product

where Name='Adjustable race' or Name like('%Paint%') or name='blade')

select pv.BusinessEntityID,

(select v.Name

from Purchasing.Vendor v

where v.BusinessEntityID=pv.BusinessEntityID)

VendorName,

(select p.Name

from Production.Product p

where pv.ProductID=p.ProductID)

ProductName

from Purchasing.ProductVendor pv

where pv.ProductID in

(select p.ProductID

from Production.Product p

where p.Name like '%paint%' or

p.Name like '%Blade%' or

p.Name ='Adjustable Race')

--find product details shipped through ZY - EXPRESS

select \* from Purchasing.ShipMethod

select \* from Production.Product

select \* from Purchasing.PurchaseOrderDetail

select \* from Purchasing.PurchaseOrderHeader

select

(select p.Name from Production.Product p where p.ProductID=pd.ProductID)as ProductName,

(select p.ProductNumber from Production.Product p where p.ProductID=pd.ProductID)as ProductNumber,

(select sm.ShipMethodID from Purchasing.ShipMethod sm where sm.ShipMethodID=ph.ShipMethodID)as ShipID,

(select sm.Name from Purchasing.ShipMethod sm where sm.ShipMethodID=ph.ShipMethodID)as ShipName

FROM Purchasing.PurchaseOrderDetail pd

JOIN Purchasing.PurchaseOrderHeader ph

ON pd.PurchaseOrderID = ph.PurchaseOrderID

WHERE ph.ShipMethodID = (

SELECT s.ShipMethodID

FROM Purchasing.ShipMethod s

WHERE s.Name='ZY - EXPRESS'

)

--Q6.)find the tax amt for products where order date and ship date are on the same day

select \* from Production.Product

select \* from Purchasing.PurchaseOrderHeader

select \* from Purchasing.PurchaseOrderDetail

select

(select p.Name from Production.Product p where p.ProductID=pd.ProductID)as ProductName,

ph.TaxAmt as Tax\_Amount

from Purchasing.PurchaseOrderDetail pd

join Purchasing.PurchaseOrderHeader ph

on pd.PurchaseOrderID = ph.PurchaseOrderID

where day(ph.OrderDate)=day(ph.ShipDate)

--8)find the name of employees working in day shift

select CONCAT\_WS(' ',FirstName,LastName)as Emp\_name from Person.Person

where BusinessEntityID in (select BusinessEntityID from HumanResources.EmployeeDepartmentHistory

where ShiftID in (select ShiftID from HumanResources.Shift where ShiftID=1))

--9.based on product and product cost history find the name ,

--service provider time and average Standardcost

Select \* from Production.Product

select \* from Production.ProductCostHistory

select

p.Name as Product\_Name,

DATEDIFF\_BIG(DAY,MIN(StartDate),MAX(EndDate)) as service\_provider\_time,

AVG(ph.StandardCost)as Average\_Standard\_Cost

from Production.ProductCostHistory ph

join Production.Product p on

ph.ProductID=p.ProductID

group by p.Name

---10.)find products with average cost more than 500

Select \* from Production.Product

select \* from Production.ProductCostHistory

select P.Name,Avg(pc.StandardCost)Avg\_stand\_cost

from Production.ProductCostHistory pc

join Production.Product p on

pc.ProductID=p.ProductID

group by p.Name

having avg(pc.StandardCost)>500

--11.find the employee who worked in multiple territory

select \* from Person.Person

Select \* from HumanResources.Employee

select \* from Sales.SalesTerritory

select \* from Sales.SalesTerritoryHistory

SELECT

p.BusinessEntityID,

CONCAT\_WS(' ', p.FirstName, p.LastName) AS Emp\_name,

COUNT(DISTINCT sth.TerritoryID) AS TerritoryCount

from HumanResources.Employee e

join Person.Person p ON p.BusinessEntityID = e.BusinessEntityID

join Sales.SalesTerritoryHistory sth ON e.BusinessEntityID = sth.BusinessEntityID

group by p.BusinessEntityID, p.FirstName, p.LastName

having COUNT(DISTINCT sth.TerritoryID) > 1

order by TerritoryCount DESC;

--12.)Find out the product model name, product description for culture as Arabic

select \* from Production.ProductModel

select \* from Production.Culture

select \* from Production.ProductDescription

select \* from Production.ProductModelProductDescriptionCulture

select pm.Name as Product\_Model\_Name,

pd.Description as Product\_Description

from Production.ProductModel pm

join Production.ProductModelProductDescriptionCulture pdc

on pm.ProductModelID=pdc.ProductModelID

join Production.ProductDescription pd

on pd.ProductDescriptionID=pd.ProductDescriptionID

join Production.Culture pc

on pc.CultureID=pdc.CultureID

where pc.Name like 'Arabic'

group by pm.Name,pd.Description

--13.display EMP name, territory name, saleslastyear salesquota and bonus

select territoryId,

(Select CONCAT(FirstName,' ',LastName) from Person.Person pp where pp.BusinessEntityID=sp.BusinessEntityID)as EmployeeName,

(Select Name from sales.SalesTerritory st where st.TerritoryID=sp.TerritoryID)as TerritoryName,

(Select [Group] from Sales.SalesTerritory sl where sl.TerritoryID=sp.TerritoryID)as GroupName,

SalesLastYear,

SalesQuota,

Bonus

from sales.SalesPerson sp

--Q14. display employee name, territory name, sales last year, sales quota and bonus from germany and united kingdom

select TerritoryID,

(Select CONCAT(FirstName,' ',LastName) from Person.Person pp where pp.BusinessEntityID=sp.BusinessEntityID)as EmployeeName,

(Select Name from sales.SalesTerritory st where st.TerritoryID=sp.TerritoryID)as TerritoryName,

(Select [Group] from Sales.SalesTerritory sl where sl.TerritoryID=sp.TerritoryID)as GroupName,

SalesLastYear,

SalesQuota,

Bonus

from sales.SalesPerson sp

WHERE sp.TerritoryID IN (

SELECT TerritoryID

FROM Sales.SalesTerritory

WHERE Name IN ('United Kingdom', 'Germany'))

--15.Find all employees who worked in all North America territory

select TerritoryID,

(select concat(' ',FirstName,LastName)from Person.Person p where p.BusinessEntityID=sp.BusinessEntityID)Emp\_Name,

(select name from Sales.SalesTerritory st where st.TerritoryID=sp.TerritoryID) TerritoryName,

(select [Group] from Sales.SalesTerritory st1 where st1.TerritoryID=sp.TerritoryID)Group\_NAme,SalesLastYear,SalesQuota

from Sales.SalesPerson sp WHERE sp.TerritoryID IN (

SELECT TerritoryID

FROM Sales.SalesTerritory

WHERE [Group] IN ('North America'))

--16.find all products in the cart

Select (select Name from Production.Product pp where pp.ProductID=si.ProductID)Prod\_name,

(select ProductNumber from Production.Product pp1 where pp1.ProductID=si.ProductID)Prod\_Number,

Quantity

from Sales.ShoppingCartItem si

--17.find all the products with special offer

select \* from Sales.SpecialOffer

select \* from Sales.SpecialOfferProduct

Select Distinct(Name) from Production.Product pp where pp.ProductID

in(select ProductID from Sales.SpecialOfferProduct)

--18.find all employees name , job title, card details whose credit card expired in the month 11 and year as 2008

select(select CONCAT\_WS(' ',FirstName,LastName)from Person.Person p where p.BusinessEntityID=pc.BusinessEntityID)EmpName,

(select JobTitle from HumanResources.Employee e where e.BusinessEntityID=pc.BusinessEntityID)Job\_Description,

(select CONCAT\_WS(' : ',ExpMonth,ExpYear )from Sales.CreditCard cc where cc.CreditCardID=pc.CreditCardID)Card\_detail

from Sales.PersonCreditCard pc where pc.CreditCardID in

(select CreditCardID from Sales.CreditCard cc where cc.ExpMonth=11 and cc.ExpYear=2008)

--19.Find the employee whose payment might be revised (Hint : Employee payment history)

select \* From Person.Person

select \* from HumanResources.Employee

select \* from HumanResources.EmployeePayHistory

--using subquerry

select

(select CONCAT\_WS(' ',p.FirstName,p.LastName)from Person.Person p where p.BusinessEntityID=e.BusinessEntityID)full\_name,e.BusinessEntityID,

(select count(eph.RateChangeDate)from HumanResources.EmployeePayHistory eph where eph.BusinessEntityID=e.BusinessEntityID)revision

from HumanResources.Employee e

where(select count(eph.RateChangeDate)from HumanResources.EmployeePayHistory eph where eph.BusinessEntityID=e.BusinessEntityID)>1

--usingjoin

SELECT p.FirstName, p.LastName, COUNT(eph.RateChangeDate) AS PayRevisions

FROM HumanResources.EmployeePayHistory eph

JOIN HumanResources.Employee e

ON eph.BusinessEntityID = e.BusinessEntityID

JOIN Person.Person p

ON e.BusinessEntityID = p.BusinessEntityID

GROUP BY p.FirstName, p.LastName

HAVING COUNT(eph.RateChangeDate) > 1;

-------------------------------------------using joins ---------------------------------------------------------------------------------

----------------------------------------------------------------------------------------------------------------------------------------

--20. .Find the personal details with address and address type(hint: Business Entity Address , Address, Address type)

select \* from Person.Person

select \* from Person.Address

select \* from Person.BusinessEntityAddress

select p.FirstName,p.LastName,a.AddressLine1,a.AddressLine2,a.PostalCode,a.City,a.StateProvinceID,

(select name from Person.AddressType at where at.AddressTypeID=be.AddressTypeID)type

from Person.BusinessEntityAddress be,Person.Person p,Person.Address a,Person.AddressType at

where p.BusinessEntityID=be.BusinessEntityID and a.AddressID=be.AddressID and at.AddressTypeID=be.AddressTypeID

--21. . Find the name of employees working in group of North America territory

select \* from Person.Person --beid

select \* from Sales.SalesTerritory --tid

select \* from Sales.SalesPerson --beid,tid

select p.FirstName,p.LastName,st.[Group] from Person.Person p

join Sales.SalesPerson sp on sp.BusinessEntityID=p.BusinessEntityID

join Sales.SalesTerritory st on sp.TerritoryID=st.TerritoryID where [Group]='North America'

--22. Find the employee whose payment is revised for more than once

select \* from HumanResources.Employee

select \* from Person.Person

select \* from HumanResources.EmployeePayHistory

select

(select CONCAT\_WS(' ',p.FirstName,p.LastName)from Person.Person p where p.BusinessEntityID=e.BusinessEntityID)full\_name,

(select count(eph.RateChangeDate)from HumanResources.EmployeePayHistory eph where eph.BusinessEntityID=e.BusinessEntityID)revision

from HumanResources.Employee e

where(select count(eph.RateChangeDate)from HumanResources.EmployeePayHistory eph where eph.BusinessEntityID=e.BusinessEntityID)>1

--23. display the personal details of employee whose payment is revised for more than once.

select \* from HumanResources.EmployeePayHistory

select \* from Person.Person

select\*from person.BusinessEntityAddress

select\* from Person.Address

select eph.BusinessEntityID,CONCAT\_WS('',p.FirstName,p.LastName) as full\_name,a.AddressLine1

from HumanResources.EmployeePayHistory eph

join Person.Person p on eph.BusinessEntityID=p.BusinessEntityID

join Person.BusinessEntityAddress bea on bea.BusinessEntityID=p.BusinessEntityID

join Person.Address a on a.AddressID=bea.AddressID

group by eph.BusinessEntityID,p.FirstName,p.LastName,a.AddressLine1

having count(eph.RateChangeDate)>1;

---24

SELECT

d.BusinessEntityID,

DATEDIFF\_BIG(MONTH, d.Penultimate, d.Ultimate) AS MonthDifference

FROM (

SELECT

t.BusinessEntityID,

(SELECT RateChangeDate

FROM (SELECT BusinessEntityID, RateChangeDate, ROW\_NUMBER() OVER (PARTITION BY BusinessEntityID ORDER BY RateChangeDate DESC) AS rankNumber

FROM HumanResources.EmployeePayHistory) AS sub

WHERE sub.rankNumber = 1 AND sub.BusinessEntityID = t.BusinessEntityID

) AS Ultimate,

(SELECT RateChangeDate

FROM (SELECT BusinessEntityID, RateChangeDate, ROW\_NUMBER() OVER (PARTITION BY BusinessEntityID ORDER BY RateChangeDate DESC) AS rankNumber

FROM HumanResources.EmployeePayHistory) AS sub

WHERE sub.rankNumber = 2 AND sub.BusinessEntityID = t.BusinessEntityID

) AS Penultimate

FROM (SELECT DISTINCT BusinessEntityID FROM HumanResources.EmployeePayHistory) AS t

) AS d

WHERE d.Penultimate IS NOT NULL;

-- 25. check if any employee from jobcandidate table is having any payment revisions

select \* from HumanResources.JobCandidate

select \* from HumanResources.EmployeePayHistory

select \* from HumanResources.Employee

select jc.BusinessEntityID from HumanResources.JobCandidate jc

join HumanResources.EmployeePayHistory eph on eph.BusinessEntityID=jc.BusinessEntityID

join HumanResources.Employee e on e.BusinessEntityID=e.BusinessEntityID

group by jc.BusinessEntityID having count(eph.RateChangeDate)>1

-- 26.check the department having more salary revision

select \* from HumanResources.Department

select \* from HumanResources.EmployeePayHistory

select \* from HumanResources.EmployeeDepartmentHistory

select d.name,count(eph.RateChangeDate) as payrev

from HumanResources.EmployeePayHistory eph

join HumanResources.EmployeeDepartmentHistory edh on edh.BusinessEntityID=eph.BusinessEntityID

join HumanResources.Employee e on e.BusinessEntityID=eph.BusinessEntityID

join HumanResources.Department d on d.DepartmentID=edh.DepartmentID

group by d.Name

order by payrev desc

--24

--27. check the employee whose payment is not yet revised

select \* from HumanResources.Employee

select \* from Person.Person

select \* from HumanResources.EmployeePayHistory

select

(select CONCAT\_WS(' ',p.FirstName,p.LastName)from Person.Person p where p.BusinessEntityID=e.BusinessEntityID)full\_name,

(select count(eph.RateChangeDate)from HumanResources.EmployeePayHistory eph where eph.BusinessEntityID=e.BusinessEntityID)revision

from HumanResources.Employee e

where(select count(eph.RateChangeDate)from HumanResources.EmployeePayHistory eph where eph.BusinessEntityID=e.BusinessEntityID)<2

-- 28. find the job title having more revised payments

select \* from HumanResources.Employee

select \* from Person.Person

select \* from HumanResources.EmployeePayHistory

select e.JobTitle,count(eph.RateChangeDate) as rev from HumanResources.EmployeePayHistory eph

join HumanResources.Employee e on eph.BusinessEntityID=e.BusinessEntityID

group by e.JobTitle

order by rev desc

--30. find the colour wise count of the product (tbl: product)

select color,count(Name) as products from Production.Product where Color is not null group by Color order by products desc

---example

select shiprate,

case

when ShipRate<=0.99 then 'Low'

when ShipRate<=1.99 then 'med'

else 'high'

end as 'category'

from Purchasing.ShipMethod;

--if null, COALESCE

select TerritoryID, COALESCE(CurrencyRateID,0) from Sales.SalesOrderHeader

-- like example

select \* from Person.Person

where FirstName like 'K%'

select \* from Person.Person where FirstName like '[AP]%'

select \* from Person.Person where FirstName like '[A-D]%'

select \* from Person.Person where (FirstName like 'A%' or FirstName like 'P%')

select \* from Person.Person where FirstName like 'pr%'

-- math fun

select ABS (-12.33),

ceiling(12.33),

floor(12.33),

exp(1),

power(2,4),

RADIANS(90),

ROUND(12.33,1),

ascii('A')

select char(65),

CHARINDEX('a','happy'),

DIFFERENCE('ABC','ABC') ,

DATALENGTH('abssaa')

'Bizmetric',

left('Bizmetric',4),

right('Bizmetric',4)

select 'BIZMETRIC' original\_value,

len(' BIZMETRIC ')len12,

LTRIM(' BIZ METRIC ',2),

RTRIM(' BIZMETRIC ',2),

TRIM(' BIZMETRIC '),

lower(' BIZMETRIC '),

upper('xjaha'),str(89),

SUBSTRING(' BIZMETRIC ',6,3)sub\_6\_3,

SUBSTRING(' BIZMETRIC ',6,8)sub\_6\_8

select substring(' BIZMETRIC',

CHARINDEX('M',' BIZMETRIC '),

CHARINDEX('R',' BIZMETRIC ')-CHARINDEX('M',' BIZMETRIC ') )

select TRANSLATE('Monday','Monday','Sunday'),

TRANSLATE('Hi TODAY is MONDAY','MONDAY','SUNDAY'),

TRANSLATE('123','2','3'),

TRANSLATE('3+[34/5]/{0-9}','[]{}','()--')

--date function

select DATEADD(year,1,'2025-01-01')add1,dateadd(YY,1,'2024-01-01')add2,

dateadd(YYYY,1,'2025-01-01')add3,dateadd(month,3,'2024-01-01')add4,

dateadd(week,2,'2024-01-01')add5,dateadd(WEEKDAY,15,'2024-01-01')add6,

dateadd(minute,2,'2024-01-01')add7,dateadd(SECOND,2,'2024-01-01')add8,

dateadd(hour,2,'2024-01-01')add9;

select dateadd(WEEKDAY,15,'2024-01-01')add1,

dateadd(day,15,'2024-01-01')add2,GETDATE()g

select datediff(year,'2024-02-01','2025-02-01')sub1,

datediff(YY,'2024-02-01','2025-02-01')sub2,DATEDIFF(YYYY,'2024-02-01','2025-02-01')sub3,

datediff(Q,'2024-02-01','2025-02-01')sub4,DATEDIFF(QQ,'2024-02-01','2025-02-01')sub5,

DATEDIFF(QUARTER,'2024-02-01','2025-02-01')sub6;

--31.find out the product who are not in position to sell (hint: check the sell start and end date)

select \* from Production.Product

select distinct ProductID, Name, SellStartDate, SellEndDate

from Production.Product where (SellEndDate is not null and SellEndDate < GETDATE())

or SellStartDate is null

--32.find the class wise, style wise average standard cost

select class Class,style Style,avg(StandardCost)Avg\_Cost from Production.Product where

class is not null and Style is not null

group by Class,Style

order by Avg\_Cost

--33.check colour wise standard cost

select \* from Production.Product

select color Color,avg(StandardCost)Color\_AvgCost from Production.Product

where color is not null

group by Color

order by Color\_AvgCost

--34.find the product line wise standard cost

select Productline Product\_line,avg(StandardCost)P\_Std from Production.Product

where ProductLine is not null

group by ProductLine

order by P\_Std

--35.Find the state wise tax rate (hint: Sales.SalesTaxRate, Person.StateProvince)

SELECT sp.Name AS StateName, sp.StateProvinceCode, str.TaxRate

FROM Sales.SalesTaxRate str

JOIN Person.StateProvince sp

ON str.StateProvinceID = sp.StateProvinceID

ORDER BY sp.Name

--Q36. Find the department wise count of employees

select\* from HumanResources.Employee

select\*from HumanResources.Department

select\*from HumanResources.EmployeeDepartmentHistory

select d.Name as DepartmentName,count(e.BusinessEntityID) as EmployeeCount

from HumanResources.Employee e

join HumanResources.EmployeeDepartmentHistory edh

on e.BusinessEntityID=edh.BusinessEntityID

join HumanResources.Department d

on d.DepartmentID=edh.DepartmentID

group by d.Name

--37.Find the department which is having more employees

SELECT d.DepartmentID, d.Name AS DepartmentName, COUNT(e.BusinessEntityID) AS EmployeeCount

FROM HumanResources.Employee e

JOIN HumanResources.Department d ON e.BusinessEntityID = d.DepartmentID

GROUP BY d.DepartmentID, d.Name

ORDER BY EmployeeCount DESC

--38.Find the job title having more employees

Select \* from HumanResources.Employee

select\*from HumanResources.Department

select count(BusinessEntityID)as EmployeeCount,JobTitle from HumanResources.Employee

group by JobTitle

order by EmployeeCount desc

--39.Check if there is mass hiring of employees on single day

Select \* from HumanResources.Employee

select Hiredate, count(BusinessEntityID)Employee\_count From HumanResources.Employee

group by HireDate

Having count(BusinessEntityID)>1

Order by Employee\_count desc

--40.Which product is purchased more? (purchase order details)

select \* from Purchasing.PurchaseOrderDetail

select \* from Production.Product

SELECT p.ProductID, p.Name AS Product\_Name, SUM(pd.OrderQty) AS TotalQuantityPurchased

FROM Purchasing.PurchaseOrderDetail pd

JOIN Production.Product p ON p.ProductID = pd.ProductID

GROUP BY p.ProductID, p.Name

ORDER BY TotalQuantityPurchased DESC

--41.Find the territory wise customers count (hint: customer)

select \* from Sales.Customer

SELECT TerritoryID, COUNT(CustomerID) AS CustomerCount

FROM Sales.Customer

GROUP BY TerritoryID

ORDER BY CustomerCount DESC;

--42.Which territory is having more customers (hint: customer)

SELECT TerritoryID, COUNT(CustomerID) AS CustomerCount

FROM Sales.Customer

GROUP BY TerritoryID

ORDER BY CustomerCount DESC

--43.Which territory is having more stores (hint: customer)

SELECT TerritoryID, COUNT(StoreID) AS Store\_Count

FROM Sales.Customer

GROUP BY TerritoryID

ORDER BY Store\_Count DESC

--44. Is there any person having more than one credit card (hint: PersonCreditCard)

select\*from Person.Person

select \* from Sales.PersonCreditCard

select CONCAT\_WS(' ',p.FirstName,p.LastName)as PersonName,COUNT(pc.CreditCardID)as CreditCardCount

from Person.Person p

join sales.PersonCreditCard pc

on p.BusinessEntityID=pc.BusinessEntityID

group by p.FirstName,p.LastName

having count(pc.CreditCardID)>1

--45.Find the product wise sale price (sales order details)

select \* from Production.Product

select \* from sales.SalesOrderDetail

SELECT p.ProductID, p.Name AS ProductName,

SUM(sod.OrderQty \* sod.UnitPrice) AS TotalSalesPrice

FROM Sales.SalesOrderDetail sod

JOIN Production.Product p ON sod.ProductID = p.ProductID

GROUP BY p.ProductID, p.Name

ORDER BY TotalSalesPrice DESC

--

--46Find the total values for line total product having maximum order

--48.Calculate the age of employees

select\* from HumanResources.Employee

SELECT BusinessEntityID, BirthDate,

DATEDIFF(YEAR, BirthDate, GETDATE()) -

CASE

WHEN (MONTH(BirthDate) > MONTH(GETDATE()))

OR (MONTH(BirthDate) = MONTH(GETDATE()) AND DAY(BirthDate) > DAY(GETDATE()))

THEN 1 ELSE 0

END AS Age

FROM HumanResources.Employee;

-------other way

select concat\_ws(' ',p.FirstName,p.LastName)as EmployeeName,

year(getdate())-year(e.BirthDate)as Age

from HumanResources.Employee e

join Person.Person p

on e.BusinessEntityID=p.BusinessEntityID

--49.Calculate the year of experience of the employee based on hire date

select concat\_ws(' ',p.FirstName,p.LastName)as EmployeeName,

year(getdate())-year(e.HireDate)Experience

from HumanResources.Employee e

join Person.Person p

on e.BusinessEntityID=p.BusinessEntityID

--50.Find the age of employee at the time of joining

select e.BusinessEntityID,concat\_ws(' ',p.FirstName,p.LastName)as EmployeeName,

year(e.HireDate)-year(e.BirthDate)Age\_at\_joining

from HumanResources.Employee e

join Person.Person p

on e.BusinessEntityID=p.BusinessEntityID

SELECT BusinessEntityID,BirthDate, HireDate,

DATEDIFF(YEAR, BirthDate, HireDate) AS AgeAtJoining

FROM HumanResources.Employee

--51.Find the average age of male and female

select \* from HumanResources.Employee

select Gender,Avg(datediff(YEAR,birthdate,GETDATE()))Avg\_Age from HumanResources.Employee

group by Gender

--52.Which product is the oldest product as on the date (refer the product sell start date)

select \* from Production.Product

select top 1 name,

max(year(getdate())-year(SellStartDate))as productage

from Production.Product

group by Name

--53. Find the product wise sale price (sales order details)

SELECT p.ProductID, p.Name AS ProductName,

SUM(sod.OrderQty \* sod.UnitPrice) AS TotalSalesPrice

FROM Sales.SalesOrderDetail sod

JOIN Production.Product p ON sod.ProductID = p.ProductID

GROUP BY p.ProductID, p.Name

ORDER BY TotalSalesPrice DESC

--54. Find the total values for line total product having maximum order

select \* from Purchasing.PurchaseOrderDetail

select Top 1 PurchaseOrderID,

sum(LineTotal)as TotalLines,

max(OrderQty)as Max\_Order

from Purchasing.PurchaseOrderDetail

group by PurchaseOrderID

having max(OrderQty)>1

--55.Calculate the age of employees

select concat\_ws(' ',p.FirstName,p.LastName)as EmployeeName,

year(getdate())-year(e.BirthDate)as Age

from HumanResources.Employee e

join Person.Person p

on e.BusinessEntityID=p.BusinessEntityID

--56.Calculate the year of experience of the employee based on hire date

select concat\_ws(' ',p.FirstName,p.LastName)as EmployeeName,

year(getdate())-year(e.HireDate)Experience

from HumanResources.Employee e

join Person.Person p

on e.BusinessEntityID=p.BusinessEntityID

--57. Find the age of employee at the time of joining

SELECT BusinessEntityID,BirthDate, HireDate,

DATEDIFF(YEAR, BirthDate, HireDate) AS AgeAtJoining

FROM HumanResources.Employee

--58.Find the average age of male and female

select Gender,Avg(datediff(YEAR,birthdate,GETDATE()))Avg\_Age from HumanResources.Employee

group by Gender

--59.Which product is the oldest product as on the date (refer the product sell start date)

select top 1 name,

max(year(getdate())-year(SellStartDate))as productage

from Production.Product

group by Name

--60.Display the product name, standard cost, and time duration for the same cost. (Product cost history)

select \* from Production.ProductCostHistory

select p.Name,

ph.StandardCost,

DATEDIFF(YEAR,ph.EndDate,ph.StartDate)Time\_duration,

avg(ph.Standardcost)over(partition by DATEDIFF(YEAR,ph.EndDate,ph.StartDate))Avg\_StandardCost

from Production.ProductCostHistory ph

join Production.Product p

on p.ProductID=ph.ProductID

where ph.EndDate is not null and

ph.StartDate is not null

--61. Find the purchase id where shipment is done 1 month later of order date

Select \* from Purchasing.ShipMethod

select \* from Purchasing.PurchaseOrderHeader

select PurchaseOrderID

from Purchasing.PurchaseOrderHeader where datediff(MONTH,OrderDate,ShipDate)=1

--62Find the sum of total due where shipment is done 1 month later of order date ( purchase order header)

select sum(TotalDue)Total

from Purchasing.PurchaseOrderHeader where datediff(MONTH,OrderDate,ShipDate)=1

--63.Find the average difference in due date and ship date based on online order flag

SELECT OnlineOrderFlag,

AVG(DATEDIFF(DAY, ShipDate, DueDate)) AS Avg\_Days\_Difference

FROM Sales.SalesOrderHeader

GROUP BY OnlineOrderFlag

--64. Display business entity id, marital status, gender, vacationhr, average vacation based on marital status

select \* from HumanResources.Employee

select \* from HumanResources.Department

select BusinessEntityId,

MaritalStatus,

Gender,

VacationHours,

avg(vacationhours)over(partition by maritalstatus)Vac\_Mari\_Status

from HumanResources.Employee

--65Display business entity id, marital status, gender, vacationhr, average vacation based on gender

select BusinessEntityId,

MaritalStatus,

Gender,

VacationHours,

avg(vacationhours)over(partition by gender)Avg\_Based\_Gender

from HumanResources.Employee

--66.Display business entity id, marital status, gender, vacationhr, average vacation based on organizational level

select BusinessEntityId,

MaritalStatus,

Gender,

VacationHours,

avg(vacationhours)over(partition by Organizationlevel )Vac\_\_Org\_level

from HumanResources.Employee

--67Display entity id, hire date, department name and department wise count of employee and count based on organizational level in each dept

SELECT

e.BusinessEntityID,

e.HireDate,

d.Name AS DepartmentName,

COUNT(e.BusinessEntityID) OVER (PARTITION BY d.Name) AS DepartmentEmployeeCount,

COUNT(e.BusinessEntityID) OVER (PARTITION BY d.Name, ed.OrganizationLevel) AS OrgLevelEmployeeCount,

COALESCE(ed.OrganizationLevel, 0) AS OrganizationLevel -- Handling NULL values

FROM HumanResources.Employee e

JOIN HumanResources.EmployeeDepartmentHistory ed

ON e.BusinessEntityID = ed.BusinessEntityID

JOIN HumanResources.Department d

ON ed.DepartmentID = d.DepartmentID;

--68.Display department name, average sick leave and sick leave per department

select distinct

d.Name DepartmentName,

avg (SickLeaveHours) over(Partition by d.departmentID)Depart\_Wise\_Sickleave,

count(SickLeaveHours) over(Partition by d.departmentid)Org\_lev\_Sickleave

from HumanResources.Employee e join HumanResources.EmployeeDepartmentHistory eh

on e.BusinessEntityID=eh.BusinessEntityID

join HumanResources.Department d on

d.DepartmentID=eh.DepartmentID

--69.Display the employee details first name, last name, with total count

--of various shift done by the person and shifts count per department

Select \* from Person.Person

select \* from HumanResources.Shift

select \* from HumanResources.Employee

select \* from HumanResources.Department

select \* from HumanResources.EmployeeDepartmentHistory

select p.FirstName,

p.LastName,

Count(s.ShiftID)TotalShift,

count(\*)over(partition by d.departmentid)Dept\_Shift\_count

from Person.Person p

join HumanResources.Employee e

on p.BusinessEntityID=e.BusinessEntityID

join HumanResources.EmployeeDepartmentHistory ed

on ed.BusinessEntityID=e.BusinessEntityID

join HumanResources.Department d

on d.DepartmentID=ed.DepartmentID

join HumanResources.Shift s

on s.ShiftID=ed.ShiftID

group by e.BusinessEntityID,p.FirstName,p.LastName,d.DepartmentID,d.Name

--70.Display country region code, group average sales quota based on territory id

select \* from Sales.SalesPerson

select \* from Sales.SalesTerritory

select st.CountryRegionCode,

st.[Group],

avg(sp.SalesQuota) as Avg\_SalesQuota

from Sales.SalesTerritory st

join Sales.SalesPerson sp

on sp.TerritoryID=st.TerritoryID

where SalesQuota is not null

group by st.CountryRegionCode,st.[Group]

order by st.CountryRegionCode,Avg\_SalesQuota Desc

--71. Display special offer description, category and avg(discount pct) per the category

Select \* from Sales.SpecialOfferProduct

Select \* from Sales.SpecialOffer

select distinct description,

Category,

avg(DiscountPct)over(partition by category)Avg\_By\_Dispt\_Cat

from Sales.SpecialOffer so

join Sales.SpecialOfferProduct

sp

on sp.SpecialOfferID=so.SpecialOfferID

--72. Display special offer description, category and avg(discount pct) per the month

SELECT distinct

Description,

Category,

Month(StartDate) AS OfferMonth,

AVG(DiscountPct) OVER (PARTITION BY Month(StartDate)) AS Avg\_Discount\_By\_Year

FROM Sales.SpecialOffer so

JOIN Sales.SpecialOfferProduct sp

ON sp.SpecialOfferID = so.SpecialOfferID;

--73. Display special offer description, category and avg(discount pct) per the year

SELECT distinct

Description,

Category,

YEAR(StartDate) AS OfferYear,

AVG(so.DiscountPct) OVER (PARTITION BY YEAR(so.StartDate),Year(so.Enddate)) AS Avg\_Discount\_By\_Year

FROM Sales.SpecialOffer so

JOIN Sales.SpecialOfferProduct sp

ON sp.SpecialOfferID = so.SpecialOfferID;

--74. Display special offer description, category and avg(discount pct) per the type

select distinct description,

Category,

avg(DiscountPct)over(partition by type)Avg\_By\_Dispt\_Type

from Sales.SpecialOffer so

join Sales.SpecialOfferProduct

sp

on sp.SpecialOfferID=so.SpecialOfferID

--75. Using rank and dense rank find territory wise top sales person

select \* from Sales.SalesTerritory

select \* from HumanResources.Employee

SELECT

sp.BusinessEntityID,

st.TerritoryID,

st.Name AS TerritoryName,

sp.SalesYTD,

RANK() OVER (PARTITION BY st.TerritoryID ORDER BY sp.SalesYTD DESC) AS Rank\_YTD,

DENSE\_RANK() OVER (PARTITION BY st.TerritoryID ORDER BY sp.SalesYTD DESC) AS Dense\_Rank\_YTD

FROM Sales.SalesPerson sp

JOIN HumanResources.Employee e ON sp.BusinessEntityID = e.BusinessEntityID

JOIN Sales.SalesTerritory st ON sp.TerritoryID = st.TerritoryID

WHERE sp.SalesYTD IS NOT NULL

ORDER BY st.TerritoryID, Rank\_YTD;