Assignment Day2 –SQL: Comprehensive practice

**Answer following questions**

* What is a result set?

It's the output of a query

* What is the difference between Union and Union All?

Union All will contain duplicate records but Union will only contain distinct records. Union perfromance is slower than Union All.Union result will be sorted but Union all not.

* What are the other Set Operators SQL Server has?

INTERSECT,EXCEPT

* What is the difference between Union and Join?

Result set can have different types of data types but Data type should be same as the result set of each select statement.JOIN doesn't remove duplicate data but UNION removes duplicate rows between the various select statements.

* What is the difference between INNER JOIN and FULL JOIN?

INNER JOIN only returns matched rows from both tables but FULL JOIN return all rows from both the tables no matter with matches.

* What is difference between left join and outer join

LEFT JOIN return all rows from the left table, and the matched rows from the right table.

Outer JOIN returns all rows from both the tables

* What is cross join?

CROSS JOIN returns the Cartesian product of rows from the rowsets in the join. In other words, it will combine each row from the first rowset with each row from the second rowset.

* What is the difference between WHERE clause and HAVING clause?

Where clause excute before the group by clause, so it can filter data from each row, but having clause excute after the group by clause,so it only can filter the group data. Having clause can use aggreate function but where clause can not.

* Can there be multiple group by columns?

Yes

**Write queries for following scenarios**

* How many products can you find in the Production.Product table?

select count(DISTINCT productid) from Production.Product

* Write a query that retrieves the number of products in the Production.Product table that are included in a subcategory. The rows that have NULL in column ProductSubcategoryID are considered to not be a part of any subcategory.

select count(distinct productid) from Production.Product where productid in ProductSubcategoryID and ProductSubcategoryID is not null

* How many Products reside in each SubCategory? Write a query to display the results with the following titles.

ProductSubcategoryID CountedProducts

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

select ProductSubcategoryID,count(distinct productid) as CountedProducts

from Production.Product

group by ProductSubcategoryID

* How many products that do not have a product subcategory.

select count(distinct productid) from Production.Product where ProductSubcategoryID=null

* Write a query to list the sum of products quantity in the Production.ProductInventory table.

select productid, sum(Quantity)

from Production.ProductInventory

group by productid

* Write a query to list the sum of products in the Production.ProductInventory table and LocationID set to 40 and limit the result to include just summarized quantities less than 100.

ProductID TheSum

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

select productid, sum(Quantity) as TheSum

from Production.ProductInventory

where LocationID=40

group by productid

having sum(Quantity) <100

* Write a query to list the sum of products with the shelf information in the Production.ProductInventory table and LocationID set to 40 and limit the result to include just summarized quantities less than 100

Shelf ProductID TheSum

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

SELECT Shelf, ProductID, SUM(quantity) AS TheSum

FROM Production.ProductInventory

WHERE LocationID = 40

GROUP BY ProductID, shelf

HAVING SUM(quantity) <100

* Write the query to list the average quantity for products where column LocationID has the value of 10 from the table Production.ProductInventory table.

SELECT AVG(Quantity) AS TheAvg

FROM Production.ProductInventory

Where LocationID = 10

* Write query to see the average quantity of products by shelf from the table Production.ProductInventory

ProductID Shelf TheAvg

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

SELECT ProductID, Shelf, AVG(Quantity) AS TheAvg

FROM Production.ProductInventory

GROUP BY Shelf, ProductID

* Write query to see the average quantity of products by shelf excluding rows that has the value of N/A in the column Shelf from the table Production.ProductInventory

ProductID Shelf TheAvg

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

SELECT ProductID, Shelf, AVG(Quantity) AS TheAvg

FROM Production.ProductInventory

WHERE LocationID = 10 AND Shelf != 'N/A'

GROUP BY Shelf, ProductID

ORDER BY Shelf

* List the members (rows) and average list price in the Production.Product table. This should be grouped independently over the Color and the Class column. Exclude the rows where Color or Class are null.

Color Class TheCount AvgPrice

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

select color,class,count(\*) as TheCount,avg(ListPrice) as AvgPrice from

where color is not null and class is not null

group by color,class

**Joins:**

* Write a query that lists the country and province names from person. CountryRegion and person. StateProvince tables. Join them and produce a result set similar to the following.

Country Province

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

select distinct c.Name as Country, s.Name as Province

from person.CountryRegion c join person. StateProvince s on s.CountryRegionCode = c.CountryRegionCode

* Write a query that lists the country and province names from person. CountryRegion and person. StateProvince tables and list the countries filter them by Germany and Canada. Join them and produce a result set similar to the following.

Country Province

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

select distinct c.Name as Country, s.Name as Province

from person.CountryRegion c join person. StateProvince s on s.CountryRegionCode = c.CountryRegionCode

where c.Name='Germany' or c.Name='Canada'

**Using Northwnd Database: (Use aliases for all the Joins)**

* List all Products that has been sold at least once in last 25 years.

select distinct p.ProductName

from Products p inner join [Order Details] o

on p.ProductID = o.ProductID join Orders r

on r.OrderID = o.OrderID

where datediff(year,r.OrderDate,getdate())<25 and datediff(year,r.OrderDate,getdate())>=0

* List top 5 locations (Zip Code) where the products sold most.

select top 5 ShipPostalCode from Orders

group by ShipPostalCode

order by count(ShipPostalCode) desc

* List top 5 locations (Zip Code) where the products sold most in last 25 years.

select top 5 ShipPostalCode from Orders

where datediff(year,r.OrderDate,getdate())<25 and datediff(year,r.OrderDate,getdate())>=0

group by ShipPostalCode

order by count(ShipPostalCode) desc

* List all city names and number of customers in that city.

select City, count(ContactName) as 'Number customer for city'

from Customers

group by City

* List city names which have more than 2 customers, and number of customers in that city

select City, count(ContactName) as 'Number customer for city'

from Customers

group by City

having count(ContactName) > 10

* List the names of customers who placed orders after 1/1/98 with order date.

select distinct c.ContactName from Orders o inner join Customers c

on o.CustomerID = c.CustomerID

where OrderDate between '1998-01-01' and '2021-12-31'

* List the names of all customers with most recent order dates

select CustomerID, OrderDate from

(select distinct CustomerID, OrderDate ,dense\_rank() over (partition by CustomerID order by orderDate desc) rk from Orders) d

where d.rk = 1

* Display the names of all customers along with the count of products they bought

select c.ContactName, count(c.ContactName)

from Orders o inner join Customers c

on o.CustomerID = c.CustomerID

group by c.ContactName

order by count(c.ContactName) desc

* Display the customer ids who bought more than 100 Products with count of products.

select c.ContactName, sum(r.Quantity)

from Orders o inner join Customers c

on o.CustomerID = c.CustomerID

inner join [Order Details] r

on r.OrderID = o.OrderID

group by c.ContactName

having sum(r.Quantity) > 100

order by sum(r.Quantity) desc

* List all of the possible ways that suppliers can ship their products. Display the results as below

Supplier Company Name Shipping Company Name

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

select u.CompanyName, s.CompanyName from Shippers s

cross join Suppliers u

* Display the products order each day. Show Order date and Product Name.

select distinct r.OrderDate, p.ProductName

from Products p inner join [Order Details] o

on p.ProductID = o.ProductID

inner join Orders r

on r.OrderID = o.OrderID

* Displays pairs of employees who have the same job title.

select \* from Employees e inner join Employees m

on e.Title = m.Title

* Display all the Managers who have more than 2 employees reporting to them.

select e.EmployeeID, e.LastName, e.FirstName, e.Title from Employees e inner join

Employees m

on e.EmployeeID = m.ReportsTo

where e.Title like '%manager%'

group by e.EmployeeID, e.LastName, e.FirstName, e.Title

having count(m.ReportsTo) > 2

* Display the customers and suppliers by city. The results should have the following columns

City

Name

Contact Name,

Type (Customer or Supplier)

select city, ContactName, 'Customer' as Type from Customers

union

select city, ContactName, 'Supplier' as Type from Suppliers

28. Have two tables T1 and T2

|  |  |
| --- | --- |
| F1.T1 | F2.T2 |
| 1 | 2 |
| 2 | 3 |
| 3 | 4 |

Please write a query to inner join these two tables and write down the result of this query.

Select \* from F1 inner join F2 on F1.T1 = F2.T2

F1.T1 F2.T2

2 2

3 3

29. Based on above two table, Please write a query to left outer join these two tables and write down the result of this query.

Select \* from F1 left join F2 on F1.T1 = F2.T2

F1.T1 F2.T2

1 null

2 2

3 3

GOOD LUCK.