



BASE DE DATOS AVANZADO I

Unidad 2: Lenguaje de manipulación de datos DML

Tema 5: Recuperación de datos - Parte II





Tema 05: Recuperación de datos

Tema 5: Recuperación de datos

2.2.4. Funciones agregadas: SUM, MIN, MAX, AVG, COUNT

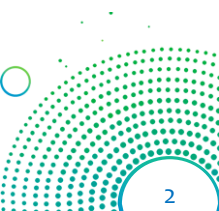
2.2.5. Uso de las cláusulas GROUP BY y HAVING

- Opciones Cube y Rollup

2.2.6. Combinación de tablas:

- Internas: INNER JOIN.
- Externas: LEFT JOIN, RIGHT JOIN.
- Cruzadas: CROSS JOIN, FULL JOIN.

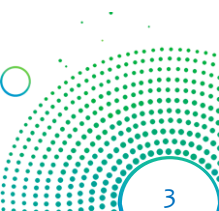
2.2.7. Agregar conjunto de resultados: UNION





Capacidades

1. Identifica los comandos de manipulación de datos y sus diferentes opciones.
2. Implementa sentencias optimizadas para consultar, ingresar y eliminar registros así también actualiza datos



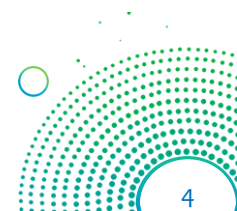


Introducción

Las diferentes áreas de negocio de una empresa, requieren conocer cierta información de forma oportuna acerca de los indicadores de ventas, producción, gastos y finanzas.

Para ello solicitan al área de sistemas consultas sobre:

- Cual es el monto facturado.
- En que distrito consumen mas productos lácteos y en que distritos no se consume.





Introducción

El equipo de desarrolladores de base de datos, acordaron las siguientes tareas:

- Implementar consultas combinadas con agrupamiento de registros.
- Realizar consultas con combinación interna y externa .





1. Combinación de tablas

Recuperación avanzada de datos

- Técnicas de consultas más avanzadas.
- Una de estas técnicas es la de combinar contenidos de una o más tablas para producir un conjunto de resultados que incorpore filas y columnas de cada tabla.
- Otra técnica es la de agrupar los datos para obtener, desde un conjunto de filas, datos agrupados



1. Combinación de tablas

- *JOIN*
- JOIN en el lenguaje de consulta, permite combinar registros de dos o más tablas en una base de datos relacional.
- La sentencia JOIN se pueden especificar en las cláusulas FROM o WHERE, aunque se recomienda que se especifiquen en la cláusula FROM.





1. Combinación de tablas

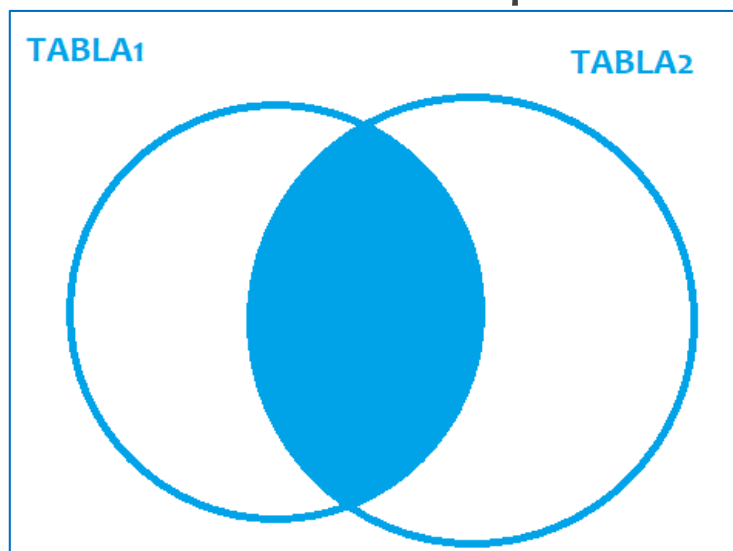
JOIN

```
--Combinando 03 tablas
Select  PD.IdProducto,
        PD.NomProducto ,
        PV.NomProveedor ,
        CA.NombreCategoria ,
        PD.PrecioUnidad
from Compras.productos PO Join Compras.proveedores PV
    On PD.IdProveedor=PV.IdProveedor Join Compras.categorias CA
    On PD.IdCategoria=CA.IdCategoria
go
```



1. Combinación de tablas

- *Combinación Interna INNER JOIN*
- El operador INNER JOIN permite la combinación de los registros de las tablas, comparando los valores de la columna específica en ambas tablas.
- Cuando no existe esta correspondencia, el registro no se muestra.



1. Combinación de tablas

Combinación Interna INNER JOIN

```
--Interna (Muestra solo los que coinciden en la comparación)
Select  PR.IdProducto,
        PR.NomProducto,
        PV.NomProveedor,
        PR.PrecioUnidad
From Compras.proveedores PV INNER JOIN Compras.productos PR
    On PV.IdProveedor = PR.IdProveedor
go
```

1. Combinación de tablas

Combinación Interna INNER JOIN

```
/*Combinar 03 Tablas */  
Select  PR.IdProducto,  
        PR.NomProducto,  
        PV.NomProveedor,  
        CA.NombreCategoria,  
        PR.PrecioUnidad  
From Compras.proveedores PV INNER JOIN Compras.productos PR  
    On PV.IdProveedor = PR.IdProveedor INNER JOIN Compras.categorias CA  
    On PR.IdCategoria = Ca.IdCategoria  
go
```



1. Combinación de tablas

Combinación Externa

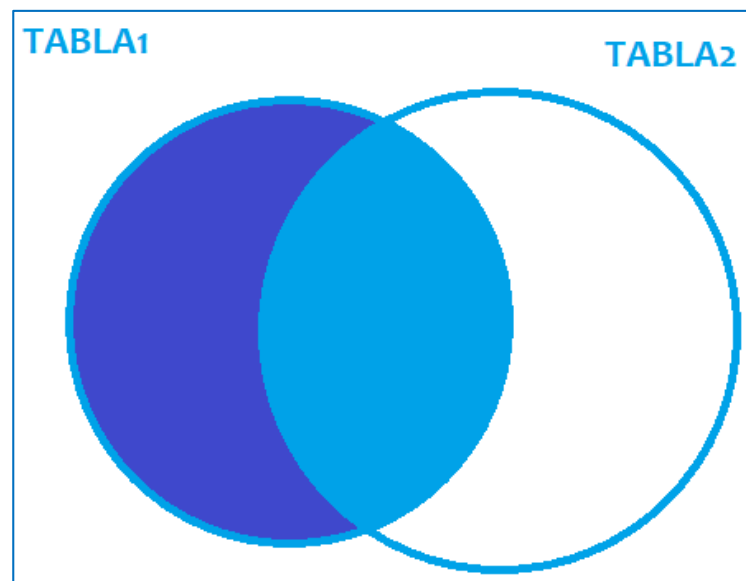
- Mediante esta operación no se requiere que cada registro en las tablas a tratar tenga un registro equivalente en la otra tabla. El registro es mantenido en la tabla combinada si no existe otro registro que le corresponda.
- Las combinaciones externas especificadas en una cláusula FROM:
 - ❑ LEFT OUTER JOIN o LEFT JOIN
 - ❑ RIGHT OUTER JOIN o RIGHT JOIN
 - ❑ FULL OUTER JOIN o FULL JOIN



1. Combinación de tablas

Combinación Externa LEFT JOIN

- El operador de combinación LEFT JOIN retorna la pareja de todos los valores de la izquierda con los valores de la tabla de la derecha correspondientes, o retorna un valor nulo NULL en caso de no correspondencia.



1. Combinación de tablas

Combinación Externa LEFT JOIN

```
--(Muestra los que coinciden y los que no coinciden en la comparación  
--con la tabla del lado Izquierdo)  
Select      CL.*,  
            PC.*  
From Ventas.clientes CL LEFT JOIN Ventas.pedidoscabe PC  
    On CL.IdCliente = PC.IdCliente  
go
```

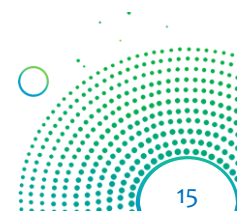
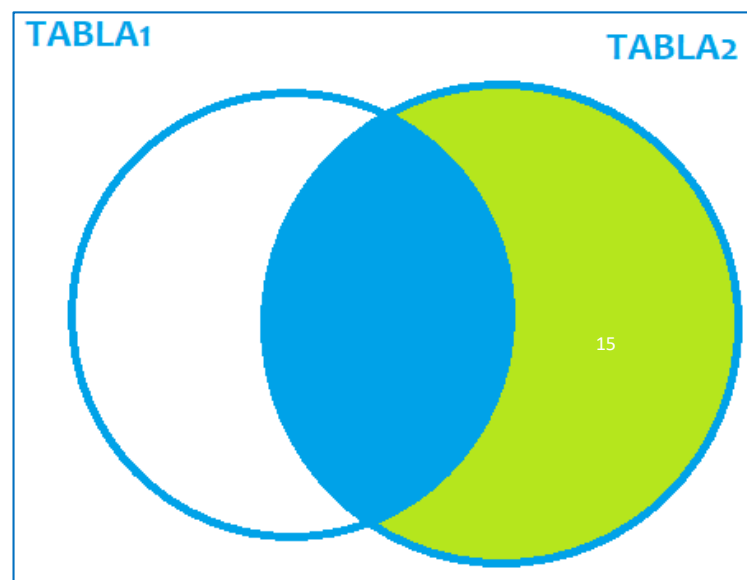
Results		Messages									
	IdCliente	NomCliente	DirCliente	idpais	fonoCliente	IdPedido	IdCliente	IdEmpleado	FechaPedido	FechaEntrega	FechaEnvio
826	PERIC	Pericles Comidas cla...	Calle Dr. Jorge Cas...	003	(5) 545-3745	11073	PERIC	2	2010-11-15 00:00:00.000	2007-12-02 00:00:00.000	1899-12-30 00:00:00.000
827	SIMOB	Simons bistro	Vinbaeltet 34	006	31 13 35 57	11074	SIMOB	7	2011-01-16 00:00:00.000	2007-12-02 00:00:00.000	1899-12-30 00:00:00.000
828	RICSU	Richter Supermarkt	Grenzacherweg 237	003		11075	RICSU	8	2011-03-16 00:00:00.000	2007-12-03 00:00:00.000	1899-12-30 00:00:00.000
829	BONAP	Bon app	12, rue des Bouch	001	91.24.45.41	11076	BONAP	4	2011-04-02 00:00:00.000	2007-12-08 00:00:00.000	1899-12-30 00:00:00.000
830	RATTC	Rattlesnake Canyon ...	2817 Milton Dr.	001	(505) 555-3620	11077	RATTC	1	2011-04-19 00:00:00.000	2007-12-10 00:00:00.000	1899-12-30 00:00:00.000
831	BONAP	Bon app	12, rue des Bouch	001	91.24.45.41	11078	BONAP	1	2010-04-26 00:00:00.000	2007-12-10 00:00:00.000	1899-12-30 00:00:00.000
832	PARIS	Paris specialites	265, boulevard Ch...	008	(1) 42.34.22.77	NULL	NULL	NULL	NULL	NULL	NULL
833	FISSA	FISSA Fabrica Inter. ...	C/ Moralarzal, 86	003	(91) 555 55 93	NULL	NULL	NULL	NULL	NULL	NULL



1. Combinación de tablas

Combinación Externa RIGHT JOIN

- El operador de combinación RIGHT JOIN retorna la pareja de todos los valores de la derecha con los valores de la tabla de la izquierda correspondientes, o retorna un valor nulo NULL en caso de no correspondencia.





1. Combinación de tablas

Combinación Externa RIGHT JOIN

```
--(Muestra los que coinciden y los que no coinciden en la comparación  
--con la tabla del lado Derecho)  
Select CL.*,  
       PC.*  
From Ventas.clientes CL RIGHT JOIN Ventas.pedidoscabe PC  
On CL.IdCliente = PC.IdCliente  
go
```

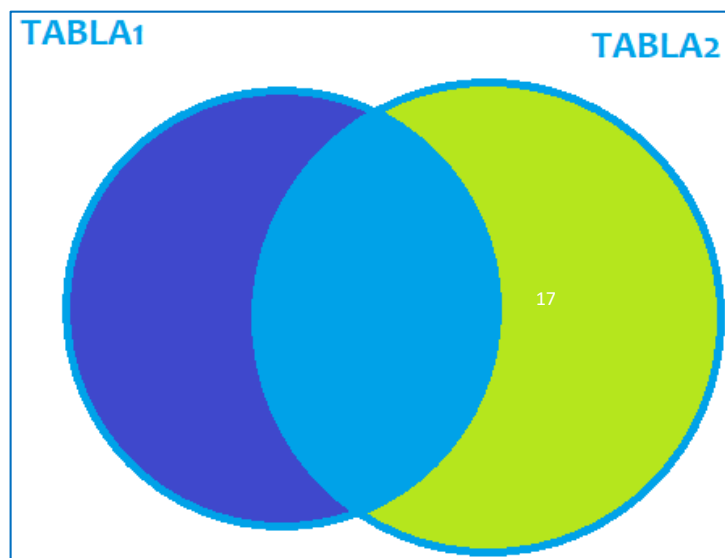
Results		Messages								
	IdCliente	NomCliente	DirCliente	idpais	fonoCliente	IdPedido	IdCliente	IdEmpleado	FechaPedido	FechaEntrega
1	NULL	NULL	NULL	NULL	NULL	8888	NULL	NULL	2018-05-28 10:51:59.943	2018-05-29 10:51:59.943
2	NULL	NULL	NULL	NULL	NULL	9999	NULL	NULL	2018-05-28 10:51:59.943	2018-05-29 10:51:59.943
3	WILMK	Wilman Kala	Keskuskatu 45	007	90-224 8858	10248	WILMK	5	1996-07-04 00:00:00.000	1996-08-01 00:00:00.000
4	TOMSP	Toms Spezialitäten	Luisenstr. 48	004	0251-035695	10249	TOMSP	6	1996-07-05 00:00:00.000	1996-08-16 00:00:00.000
5	HANAR	Hanari Cames	Rua do Paço, 67	001	(21) 555-8765	10250	HANAR	4	1996-07-08 00:00:00.000	1996-08-05 00:00:00.000
6	VICTE	Victuailles en stock	2, rue du Commerce	007	78.32.54.87	10251	VICTE	3	1996-07-08 00:00:00.000	1996-08-05 00:00:00.000



1. Combinación de tablas

Combinación Externa FULL JOIN

- Retorna todas las filas de las tablas de la izquierda y la derecha. Cada vez que una fila no tenga coincidencia en la otra tabla, las columnas de la lista de selección de la otra tabla contendrán valores NULL. Cuando haya una coincidencia entre las tablas, la fila contendrá los valores de datos de las tablas base.





1. Combinación de tablas

Combinación Externa FULL JOIN

```
--(Muestra los que coinciden y los que no coinciden  
--en la comparación con ambas tablas)  
- Select CL.*,  
      PC.*  
From Ventas.clientes CL FULL JOIN Ventas.pedidoscabe PC  
  On CL.IdCliente = PC.IdCliente  
go
```

Results		Messages								
	IdCliente	NomCliente	DirCliente	idpais	fonoCliente	IdPedido	IdCliente	IdEmpleado	FechaPedido	FechaEntrega
1	NULL	NULL	NULL	NULL	NULL	8888	NULL	NULL	2018-05-28 10:51:59.943	2018-05-29 10:51:59.943
2	NULL	NULL	NULL	NULL	NULL	9999	NULL	NULL	2018-05-28 10:51:59.943	2018-05-29 10:51:59.943
3	WILMK	Wilman Kala	Keskuskatu 45	007	90-224 8858	10248	WILMK	5	1996-07-04 00:00:00.000	1996-08-01 00:00:00.000
4	TOMSP	Toms Spezialitäten	Luisenstr. 48	004	0251-035695	10249	TOMSP	6	1996-07-05 00:00:00.000	1996-08-16 00:00:00.000
831	BONAP	Bon app	12, rue des Bouch	001	91.24.45.41	11076	BONAP	4	2011-04-02 00:00:00.000	2007-12-08 00:00:00.000
832	RATTC	Rattlesnake Can...	2817 Milton Dr.	001	(505) 555-3620	11077	RATTC	1	2011-04-19 00:00:00.000	2007-12-10 00:00:00.000
833	BONAP	Bon app	12, rue des Bouch	001	91.24.45.41	11078	BONAP	1	2010-04-26 00:00:00.000	2007-12-10 00:00:00.000
834	PARIS	Paris specialites	265, boulevard Ch...	008	(1) 42.34.22.77	NULL	NULL	NULL	NULL	NULL
835	FISSA	FISSA Fabrica Int...	C/ Moralezarai, 86	003	(91) 555 55 93	NULL	NULL	NULL	NULL	NULL



2. Combinación cruzada

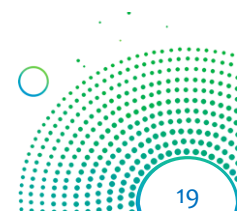
Combinación Cruzada CROSS JOIN

- Retorna el producto cartesiano entre 02 tablas.
- Combina todos los registros de una tabla con cada uno de los registros de la otra Tabla.

EmpName
Rafferty
Jones
Heisenberg
Robinson
Smith

DepName
Sales
Engineering
Clerical
Marketing

EmpName	DepName
Rafferty	Engineering
Jones	Engineering
Heisenberg	Engineering
Robinson	Engineering
Smith	Engineering



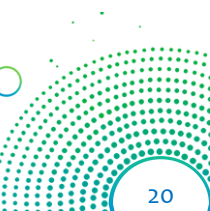


2. Combinación cruzada

Combinación Cruzada CROSS JOIN

```
/*Combinación Cruzada*/  
Select  PR.NomProducto,  
        PV.NomProveedor  
From  Compras.productos PR Cross Join Compras.proveedores PV  
Order by PR.NomProducto  
GO
```

Devuelve todas las combinaciones de todos los productos y todos los proveedores.

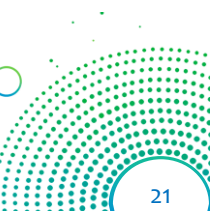




3. Unión

Agregar Conjunto de resultado. UNION

- Combina los resultados de dos o más consultas en un solo conjunto de resultados que incluye todas las filas que pertenecen a las consultas de la unión.
- Para utilizar la operación UNION debemos aplicar algunas reglas básicas para combinar los conjuntos de resultados:
 - El número y el orden de las columnas deben ser idénticos en todas las consultas.
 - Los tipos de datos deben ser compatibles.





3. Unión

Agregar Conjunto de resultado. UNION

- Muestre un listado con los nombres de los clientes y proveedores que serán invitados a la fiesta de aniversario de la Empresa.

```
/*UNION*/
Select  NomCliente      As [Invitado],
        'Cliente'       As [Tipo]
From Ventas.clientes
      UNION
Select  NomProveedor    As [Invitado],
        'Proveedor'     As [Tipo]
from Compras.proveedores
go
```





3. Unión

Agregar conjunto de resultado. *UNION*

- Resultado:

	Invitado	Tipo
1	Alfreds Futterkiste	Ciente
2	Ana Trujillo Emparedados y helados	Ciente
3	Antonio Moreno Taqueria	Ciente
4	Around the Horn	Ciente
5	Aux joyeux ecclesiastiques	Proveedor
6	B's Beverages	Ciente
7	Berglunds snabbköp	Ciente
8	Bigfoot Breweries	Proveedor
9	Blauer See Delikatessen	Ciente



4. Consulta agrupada

¿Qué es consulta agrupada?

- Los resultados de consultas se pueden resumir, agrupar y ordenar utilizando funciones agregadas y las cláusulas GROUP BY, HAVING y ORDER BY con la instrucción SELECT.
- También se puede usar la cláusula compute (una extensión Transact-SQL) con funciones agregadas para generar un informe con filas detalladas y resumidas.





4. Consulta agrupada

- *Funciones de Agregación*
- Permiten calcular valores sumarios a partir de datos de una columna concreta.
- Se pueden aplicar a todas las filas de una tabla, a un subconjunto de la tabla especificada por una cláusula WHERE o a uno o más grupos de filas de la tabla.





4. Consulta agrupada

Funciones de Agregación

Función Agregada	Resultado
SUM(expresión)	Retorna la suma total de los valores (distintos) de la expresión o columna
AVG (expresión)	Retorna el promedio de los valores (distintos) de la expresión o columna
COUNT (expresión)	Retorna el número de valores (distintos) no nulos de la expresión
COUNT(*)	Numero de filas seleccionadas
MAX(expresión)	Retorna el máximo valor de la expresión o columna
MIN(expresión)	Retorna el mínimo valor de la expresión o columna

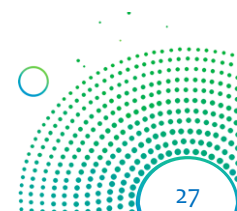




4. Consulta agrupada

Cláusula GROUP BY

- La cláusula GROUP BY se utiliza en las instrucciones SELECT para dividir la salida de una tabla en grupos. El número máximo de columnas o expresiones es 16.
- La cláusula GROUP BY aparece en instrucciones donde incluyen funciones agregadas. A estos valores se les llama agregados vectoriales. Un agregado escalar es un solo valor generado por una función agregada sin una cláusula GROUP BY.



4. Consulta agrupada

Cláusula GROUP BY

```
/*Agrupar los pedidos por cliente mostrando  
la cantidad de pedidos de cada uno*/  
Select  C.NomCliente,  
        Count(P.IdPedido) As [CantidadPedidos]  
From Ventas.clientes C Join Ventas.pedidoscabe P  
    On C.IdCliente = P.IdCliente  
Group By C.NomCliente  
go
```

Results Messages		
	NomCliente	CantidadPedidos
1	Alfreds Futterkiste	6
2	Ana Trujillo Emparedados y helados	4
3	Antonio Moreno Taqueria	7
4	Around the Horn	12
5	B's Beverages	10



4. Consulta agrupada

Cláusula HAVING

- Es posible que necesitemos calcular un agregado, pero que no necesitemos obtener todos los datos, solo los que cumplan una condición del agregado.
- Por ejemplo, podemos calcular el valor de las ventas por producto, pero que solo queramos ver los datos de los productos que hayan vendido más o menos de una determinada cantidad. En estos casos debemos utilizar la cláusula HAVING.



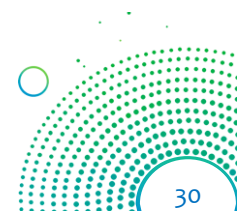


4. Consulta agrupada

Cláusula HAVING

- Muestra el año, nombre del empleado, nombre del cliente y la cantidad de pedidos. Considere solo si la cantidad es mayor a 2.

```
[-] Select  Year(P.FechaPedido) As [Año],  
           E.NomEmpleado,  
           C.NomCliente,  
           Count(P.IdPedido)  As [CantidadPedidos]  
From Ventas.clientes C Join Ventas.pedidoscabe P  
  On C.IdCliente = P.IdCliente Join RRHH.empleados E  
  On E.IdEmpleado = P.IdEmpleado  
Group By Year(P.FechaPedido),E.NomEmpleado,C.NomCliente  
Having Count(P.IdPedido) > 2  
Order By 1,2,3  
go
```





4. Consulta agrupada

Cláusula HAVING

- El resultado sería.

	Results	Messages			
	Año	NomEmpleado	NomCliente	CantidadPedidos	
1	1996	Janet	LILA-Supernmercado	3	
2	1996	Margaret	Frankenversand	3	
3	1997	Andrew	QUICK-Stop	3	
4	1997	Andrew	Wartian Herkku	3	
5	1997	Anne	Hungry Owl All-Night Grocers	3	
6	1997	Janet	Berglunds snabbköp	4	
7	1997	Janet	Bottom-Dollar Markets	3	
8	1997	Janet	Lehmanns Marktstand	3	
9	1997	Janet	Mère Paillarde	3	



4. Consulta agrupada

Resumen de datos. CUBE

- El operador CUBE es una opción adicional de la cláusula GROUP BY en una sentencia SELECT.
- Se utiliza para producir juegos de resultados que, normalmente, se utilizan para informes de tabulación cruzada.
- CUBE produce subtotales para todas las posibles combinaciones de agrupamientos especificados en la cláusula GROUP BY y una suma total.





4. Consulta agrupada

Resumen de datos. CUBE

- Muestra el año, nombre del empleado, nombre del cliente y la cantidad de pedidos. Genere totales y subtotales.

```

Select  Year(P.FechaPedido) As [Año],
        E.NomEmpleado,
        C.NomCliente,
        Count(P.IdPedido) As [CantidadPedidos]
From Ventas.clientes C Join Ventas.pedidoscabe P
    On C.IdCliente = P.IdCliente Join RRHH.empleados E
    On E.IdEmpleado = P.IdEmpleado
Group By Cube (Year(P.FechaPedido),E.NomEmpleado,C.NomCliente)
Order By 1,2,3
go
```





4. Consulta agrupada

Resumen de datos. ROLLUP

- El operador ROLLUP es útil para generar reportes que contienen subtotales y totales, genera un conjunto de resultados que es similar al conjunto de resultados del CUBE.
- Las diferencias entre CUBE y ROLLUP son:
 - CUBE genera un conjunto de resultados mostrando agregaciones para todas las combinaciones de valores en las columnas seleccionadas.
 - ROLLUP genera un conjunto de resultados mostrando agregaciones para jerarquías en las columnas seleccionadas.





4. Consulta agrupada

Resumen de datos. ROLLUP

- Muestra el año, nombre del empleado, nombre del cliente y la cantidad de pedidos. Genere totales y subtotales. Utilice ROLLUP.

```
[-] Select Year(P.FechaPedido) As Año,  
          E.NomEmpleado,  
          C.NomCliente,  
          Count(P.IdPedido) As [CantidadPedidos]  
From Ventas.clientes C Join Ventas.pedidoscabe P  
  On C.IdCliente = P.IdCliente Join RRHH.empleados E  
  On E.IdEmpleado = P.IdEmpleado  
Group By ROLLUP (Year(P.FechaPedido),E.NomEmpleado,C.NomCliente)  
Order By 1,2,3  
go
```





Ejercicio 1

- Listar la cantidad de pedidos registrados por cada empleado y año, totalizando la cantidad de pedidos por cada año.
- Solución:

```
SELECT ISNULL(e.nomEmpleado+', '+e.apeEmpleado, 'SUBTOTAL') as 'Empleado',  
       Year(p.FechaPedido) as 'Año',  
       COUNT(*) as 'Cantidad'  
FROM Ventas.pedidoscabe p JOIN RRHH.empleados e  
ON p.IdEmpleado = e.IdEmpleado  
GROUP BY e. nomEmpleado +', '+e. apeEmpleado, Year(p.FechaPedido)  
WITH CUBE  
Go
```



Ejercicio 1

- Resultado:

Results Messages			
	Empleado	Año	Cantidad
1	Andrew Fuller	1996	16
2	Anne Dodsworth	1996	5
3	Janet Leverling	1996	18
4	Laura Callahan	1996	19
5	Margaret Peacock	1996	31
6	Michael Suyama	1996	15
7	Nancy Davolio	1996	26
8	Robert King	1996	11
9	Steven Buchanan	1996	11
10	SUBTOTAL ==>	1996	152
11	Andrew Fuller	1997	41
12	Anne Dodsworth	1997	19
13	Janet Leverling	1997	71
14	Laura Callahan	1997	54
15	Margaret Peacock	1997	81
16	Michael Suyama	1997	33
17	Nancy Davolio	1997	55
18	Robert King	1997	36
19	Steven Buchanan	1997	18
20	SUBTOTAL ==>	1997	408

Ejercicio 2

- Muestre los clientes que no realizaron pedido alguno.

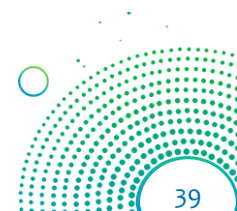
```
-- Select C.IdCliente, C.NomCliente, C.DirCliente  
From ventas.clientes C Left Join Ventas.pedidoscabe P  
    On C.IdCliente = P.IdCliente  
Where P.IdPedido is Null  
go
```

Results		Messages	
	IdCliente	NomCliente	DirCliente
1	PARIS	Paris specialites	265, boulevard Charonne
2	FISSA	FISSA Fabrica Inter. Salchichas S.A.	C/ Morazarzal, 86



Conclusiones

- Una combinación interna combinará las columnas seleccionadas de los dos conjuntos de filas unidos para cada combinación de filas que satisfaga el predicado de comparación de combinación.
- Una combinación externa combinará las columnas seleccionadas de los dos conjuntos de filas unidos para cada combinación de filas que satisfaga el predicado de unión y agregará las filas que no tienen una coincidencia para el lado de unión especificado.
- La cláusula GROUP BY de la instrucción SELECT que divide el resultado de la consulta en grupos de filas, generalmente con el propósito de realizar una o más agregaciones en cada grupo. La instrucción SELECT devuelve una fila por grupo.



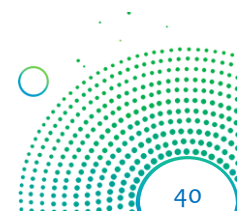


Bibliografía

Microsoft (2017) SELECT - GROUP BY- Transact-SQL. Recuperado de: <https://docs.microsoft.com/en-us/sql/t-sql/queries/select-group-by-transact-sql?view=sql-server-2017>

Microsoft (2017) INNER JOIN (U-SQL) (Transact-SQL) Recuperado de: <https://msdn.microsoft.com/en-us/azure/data-lake-analytics/u-sql/inner-join-u-sql>

Microsoft (2017) OUTER JOIN (U-SQL) (Transact-SQL) Recuperado de: <https://msdn.microsoft.com/en-us/azure/data-lake-analytics/u-sql/outer-join-u-sql>



GRACIAS



SEDE LIMA CENTRO
Av. Uruguay 514
Cercado – Lima
Teléfono: 419-2900

SEDE INDEPENDENCIA
Av. Carlos Izaguirre 233
Independencia – Lima
Teléfono: 633-5555

SEDE BREÑA
Av. Brasil 714 – 792
(CC La Rambla – Piso 3)
Breña – Lima
Teléfono: 633-5555

SEDE TRUJILLO
Calle Borgoño 361
Trujillo
Teléfono: (044) 60-2000

SEDE SAN JUAN DE LURIGANCHO
Av. Próceres de la Independencia 3023-3043
San Juan de Lurigancho – Lima
Teléfono: 633-5555

SEDE BELLAVISTA
Av. Mariscal Oscar R. Benavides 3866 – 4070
(CC Mall Aventura Plaza)
Bellavista – Callao
Teléfono: 633-5555

SEDE AREQUIPA
Av. Porongoche 500
(CC Mall Aventura Plaza)
Paucarpata - Arequipa
Teléfono: (054) 60-3535