

Capítulo 6: Consultas básicas con lenguaje SQL

Capítulo 7: Consultas avanzadas con lenguaje SQL

Capítulo 8: Gestionando los datos con lenguaje SQL



# 7 Consultas avanzadas con lenguaje SQL

SQL y modelamiento de base de datos - SQL Server 2016

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



## Objetivos

- Realizar consultas complejas aplicando agrupamiento, combinación y sub-consultas.
- Construir vistas para simplificar el acceso a la información.

7 - 2

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



## Agenda

- Funciones de agrupamiento
- Consultas multitabla
- Consultas anidadas
- Creando vistas

7 - 3

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



## Funciones de agrupamiento GROUP BY

- Agrupa un conjunto de filas de acuerdo con los valores de una o más columnas o expresiones, devolviendo una fila por cada grupo.

```
/*
This query calculates the average freight in orders
table for each customer. GROUP BY clause is used
to group all orders for each individual customer.
*/
select CustomerID, avg(Freight) as average_freight
from orders
group by CustomerID;
```

```
/*
This query shows that, for each group of product
categories, only one row of values is produced.
*/
select c.CategoryID,
       c.CategoryName,
       sum(UnitsInStock) as "Total units in stock"
from products as p
inner join categories as c on c.CategoryID=p.CategoryID
group by p.CategoryID;
```

CategoryID	CategoryName	Total units in stock
1	Beverages	559
2	Condiments	507
3	Confections	386
4	Dairy Products	393
5	Grains/Cereals	308
6	Meat/Poultry	165
7	Produce	100
8	Seafood	701

7 - 4

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



## Funciones de agrupamiento SUM

- Devuelve la suma de todos los valores de un grupo de datos que fue agrupado por la sentencia GROUP BY.

```
SELECT Color, SUM(ListPrice), SUM(StandardCost)
FROM Production.Product
WHERE Color IS NOT NULL
AND ListPrice != 0.00
AND Name LIKE 'Mountain%'
GROUP BY Color
ORDER BY Color;
GO
```

Country	Gender	TotalSalary
India	Female	4000
UK	Female	5000
USA	Female	12500
India	Male	8000
UK	Male	12000
USA	Male	10000
India	NULL	12000
UK	NULL	17000
USA	NULL	22500

Sum of Salary by Country

7 - 5

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



## Funciones de agrupamiento AVG

- Devuelve el promedio de todos los valores de un grupo de datos que fue agrupado por la sentencia GROUP BY.

```
SELECT TerritoryID, AVG(Bonus), SUM(SalesYTD)
FROM Sales.SalesPerson
GROUP BY TerritoryID;
GO
```

1	4133.3333	4502152.2674
2	4100.00	3763178.1787
3	2500.00	3189418.3662
4	2775.00	6709904.1666
5	6700.00	2315185.611
6	2750.00	4058260.1825
7	985.00	3121616.3202
8	75.00	1827066.7118
9	5650.00	1421810.9242
10	5150.00	4116871.2277

(10 row(s) affected)

7 - 6

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



## Funciones de agrupamiento MAX y MIN

- Devuelve el promedio máximo o mínimo de todos los valores de un grupo de datos que fue agrupado por la sentencia GROUP BY.

```
SELECT Name
      , MIN(Rate) AS MinSalary
      , MAX(Rate) AS MaxSalary
      , AVG(Rate) AS AvgSalary
FROM HumanResources.EmployeePayHistory AS eph
JOIN HumanResources.EmployeeDepartmentHistory AS edh
  ON eph.BusinessEntityID = edh.BusinessEntityID
JOIN HumanResources.Department AS d
  ON d.DepartmentID = edh.DepartmentID
WHERE edh.EndDate IS NULL
ORDER BY Name;
```

7 - 7

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



## Funciones de agrupamiento COUNT

- Devuelve el número de elementos de un grupo.

```
SELECT COUNT(DISTINCT Title)
FROM HumanResources.Employee;
GO
```

```
SELECT COUNT(*)
FROM HumanResources.Employee;
GO
```

7 - 8

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



## Funciones de agrupamiento Operador CUBE

- Genera un conjunto de resultados que es un cubo multidimensional.
- Genera filas de agregado mediante la clausula GROUP BY simple.

```
SELECT YEAR(OrderDate) as n_year,
       CustomerID,
       COUNT(OrderID) As n_count
FROM   orders
WHERE  CustomerID LIKE 'D%'
GROUP BY CUBE(YEAR(OrderDate), CustomerID)
```

	n_year	CustomerID	n_count
1	1996	DRACD	2
2	1997	DRACD	1
3	1998	DRACD	3
4	NULL	DRACD	6
5	1996	DUMON	1
6	1997	DUMON	2
7	1998	DUMON	1
8	NULL	DUMON	4
9	NULL	NULL	10
10	1996	NULL	3
11	1997	NULL	3
12	1998	NULL	4

7 - 9

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



## Funciones de agrupamiento Operador ROLLUP

- Es útil para generar reportes que contienen subtotales y totales.
- Para las filas agrupadas, se usa un valor de NULL para representar los valores de la columna.

```
SELECT E.LastName + ' ' + E.FirstName as Empleado,
       YEAR(O.OrderDate) As Año,
       COUNT(O.OrderID) As 'Cantidad de Órdenes'
FROM   Orders As O
INNER JOIN Employees As E ON O.EmployeeID = E.EmployeeID
WHERE  E.LastName LIKE '[ABC]%'
GROUP BY E.LastName + ' ' + E.FirstName, YEAR(O.OrderDate)
WITH ROLLUP
```

	Empleado	Año	Cantidad de Órdenes
1	Buchanan Steven	1996	11
2	Buchanan Steven	1997	18
3	Buchanan Steven	1998	13
4	Buchanan Steven	NULL	42
5	Callahan Laura	1996	19
6	Callahan Laura	1997	54
7	Callahan Laura	1998	31
8	Callahan Laura	NULL	104
9	NULL	NULL	146

7 - 10

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



## Funciones de agrupamiento HAVING

- Especifica una condición de búsqueda para un grupo.
- HAVING solo se puede utilizar con la instrucción SELECT.
- Se utiliza en una cláusula GROUP BY.

```
SELECT SalesOrderID, SUM(LineTotal) AS SubTotal
FROM Sales.SalesOrderDetail
GROUP BY SalesOrderID
HAVING SUM(LineTotal) > 100000.00
ORDER BY SalesOrderID ;
```

```
SELECT SalesOrderID, SUM(LineTotal) AS SubTotal
FROM Sales.SalesOrderDetail
GROUP BY SalesOrderID
HAVING SUM(LineTotal) > 100000.00
ORDER BY SalesOrderID
```

SalesOrderID	SubTotal
43075	122765.939600
43084	115696.331324
44518	126190.236168
44528	108783.587500
44530	104950.806836
44795	104111.515642
46066	100378.907800

7 - 11

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



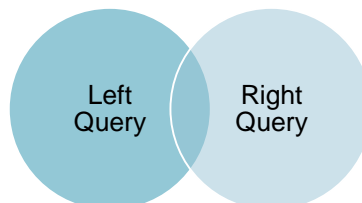
## Funciones de agrupamiento UNIÓN

- Combina los resultados de dos o más consultas en un solo conjunto de resultados.

```
/*
This query is the same as the one in Practice #4
except that it uses UNION rather than UNION ALL.

UNION eliminates duplicate rows in the result set.

Total 93 rows returned.
*/
select City, Country from customers
union
select City, Country from suppliers
order by City;
```



7 - 12

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



## Consultas multitabla INNER JOIN

- Devuelven todos los pares de las filas coincidentes.
- Rechaza las filas no coincidentes de las dos tablas.

```
SELECT *  
FROM Products P  
INNER JOIN Suppliers S ON P.SupplierID = S.SupplierID  
WHERE P.CategoryID = 1
```

7 - 13

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



## Consultas multitabla LEFT OUTER JOIN

- Especifica que todas las filas de la tabla de la izquierda que no cumplan con la condición de combinación se incluyan en el conjunto de resultados.

```
SELECT Customers.CompanyName, COUNT(Orders.OrderID) as n_count  
FROM Customers  
LEFT JOIN Orders ON Customers.CustomerID = Orders.CustomerID  
GROUP BY Customers.CompanyName  
ORDER BY Customers.CompanyName
```

7 - 14

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



## Consultas multitabla FULL OUTER JOIN

- Especifica que todas las filas de la tabla de la izquierda y la derecha que no cumplan con la condición de combinación se incluyan en el conjunto de resultados.

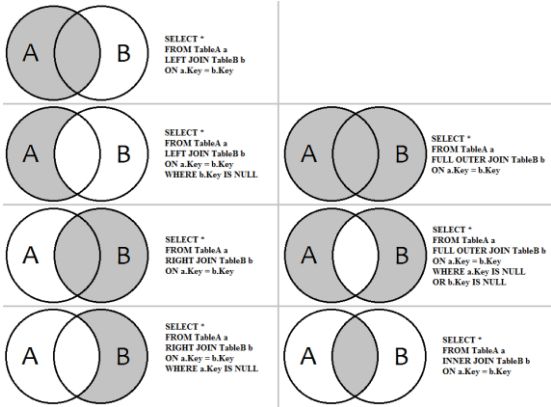
```
SELECT C.CompanyName as CompanyNameCustomers,  
       C.Country as CountryCustomers,  
       S.CompanyName as CompanyNameSuppliers,  
       S.Country as CountrySuppliers  
FROM   Customers C  
FULL   OUTER JOIN Suppliers S ON C.Country = S.Country  
ORDER  BY C.Country, S.Country
```

7 - 15

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



## Consultas multitabla LEFT & RIGHT JOINS



7 - 16

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.





## Consulta anidada

- Devuelve un valor único y está anidada en una instrucción:
  - SELECT
  - INSERT
  - UPDATE
  - DELETE
- La sub-consulta incluye siempre paréntesis.
- No puede contener un COMPUTE.
- Puede incluir una cláusula ORDER BY.



7 - 17

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



## Consulta anidada PIVOT

- Permite rotar datos de filas (visualizadas en vertical) a columnas (visualizadas en horizontal).

Sales Table		PIVOT Query												
Salestid	SalesDate	<pre>SELECT * FROM (SELECT YEAR(SalesDate) [Year], DATENAME(MONTH, SalesDate) [Month], COUNT(1) [Sales Count] FROM #Sales GROUP BY YEAR(SalesDate), DATENAME(MONTH, SalesDate)) AS MontlySalesData PIVOT( SUM([Sales Count]) FOR Month IN ([January],[February],[March],[April],[May], [June],[July],[August],[September],[October],[November],[December])) AS #NamePivot</pre>												
1	2012-01-01 00:49:01.403													
2	2012-01-03 00:49:01.277													
3	2012-01-05 00:49:01.547													
4	2012-01-05 00:49:01.563													
5	2012-01-06 00:49:01.550													
6	2012-01-06 00:49:01.563													
7	2012-01-07 00:49:01.187													
8	2012-01-07 00:49:01.523													
9	2012-01-07 00:49:01.570													
...	...	 <b>PIVOT Result with Months as columns</b>												
...	...													
...	...													
		Year	January	February	March	April	May	June	July	August	September	October	November	December
2012		32	32	37	38	47	38	24	43	39	38	33	32	
2013		45	39	42	47	35	37	57	34	35	35	49	32	
2014		39	28	13	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	

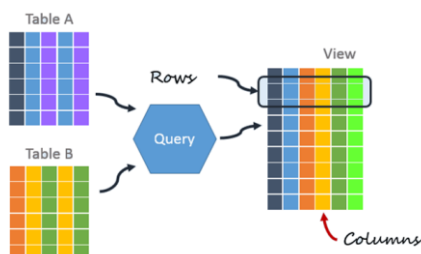
7 - 18

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



## Creando vistas ¿Qué es una view?

- Es una tabla virtual almacenada en el gestor de datos.
- Contienen un SELECT y el resultado forma la tabla virtual.
- Restringe el acceso a filas y columnas de una tabla real.
- Se utiliza mediante instrucción SELECT.
- Puede contener múltiples columnas la consulta.



7 - 19

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.

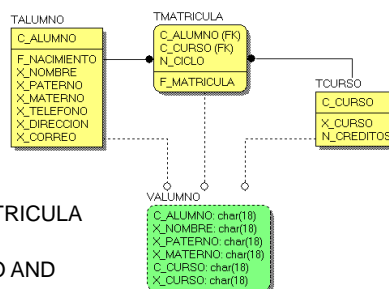


## Vista de datos

### CREATE OR ALTER VIEW VALUMNO

AS

```
SELECT A.C_ALUMNO,
       A.X_NOMBRE,
       A.X_PATERNO,
       A.X_MATERNO,
       M.C_CURSO,
       M.X_CURSO
FROM TALUMNO A, TCURSO C, TMATRICULA
M
WHERE A.C_ALUMNO = M.C_ALUMNO AND
      M.C_CURSO = C.C_CURSO
```

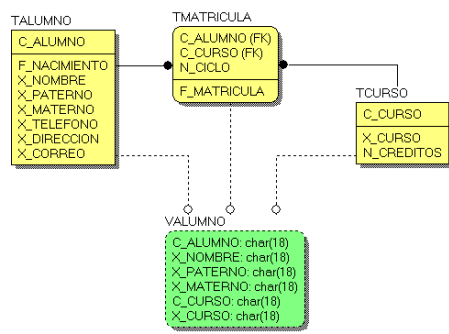


7 - 20

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



### Vista de datos



SELECT \* FROM VALUMNO



### Ejercicio Nº 7.1: Comprender las cláusulas avanzadas del lenguaje SQL

Al finalizar el laboratorio, el alumno logrará:

- Crear una consulta utilizando funciones de agrupamiento.



### **Ejercicio Nº 7.2: Comprender las cláusulas avanzadas del lenguaje SQL**

Al finalizar el laboratorio, el alumno logrará:

- Crear una consulta utilizando: GROUP BY / HAVING / CUBE / ROLLUP.

7 - 23

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



### **Ejercicio Nº 7.3: Comprender las cláusulas avanzadas del lenguaje SQL**

Al finalizar el laboratorio, el alumno logrará:

- Crear una consulta utilizando: INNER JOIN, LEFT OUTER JOIN, RIGHT OUTER JOIN y FULL OUTER JOIN.

7 - 24

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



### **Ejercicio Nº 7.4: Comprender las cláusulas avanzadas del lenguaje SQL**

Al finalizar el laboratorio, el alumno logrará:

- Crear una consulta utilizando: IN y EXISTS.

7 - 25

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



### **Ejercicio Nº 7.5: Comprender las cláusulas avanzadas del lenguaje SQL**

Al finalizar el laboratorio, el alumno logrará:

- Crear una tabla virtual.

7 - 26

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



## Lecturas adicionales

Para obtener información adicional, puede consultar el siguiente enlace :

- [http://www.famaf.proed.unc.edu.ar/pluginfile.php/35290/mod\\_resource/content/2/SQL%20Parte%202.pdf](http://www.famaf.proed.unc.edu.ar/pluginfile.php/35290/mod_resource/content/2/SQL%20Parte%202.pdf)
- <https://docplayer.es/295864-Sub-consultas-avanzadas.html>

7 - 27

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



## Resumen

En este capítulo, usted aprendió que:

- SQL permite escribir sentencias que agrupan, combinan y fusionan la información según las necesidades del cliente.
- Las vistas simplifican el trabajo al transformar varias tablas en una sola entidad virtual.

7 - 28

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



## **Tarea N° 7: Comprender las características avanzadas del SELECT**

Aplicar sentencias avanzadas del SQL.

Al finalizar el laboratorio, el alumno logrará contestar:

- ¿Cuál es el objetivo del predicado PIVOT?
- ¿Cuál es el objetivo de la función MERGE?
- ¿Qué consideraciones deben tenerse al elaborar SELECT?
- ¿Qué es el Explain Plan?

