



Cátedra Gestión de Datos

Unidad Temática N° 4: Lenguaje SQL

Revisión

SELECT	{lista de columnas}
FROM	tablas
WHERE	condiciones-filas
GROUP BY	{lista de columnas}
HAVING	condiciones-grupo
ORDER BY	{lista de columnas}

↩ **Condiciones de Búsqueda**

- *Test de comparación*
- *Test de rango*
- *Test de pertenencia a conjunto*
- *Test de correspondencia con patrón*
- *Test de valor nulo*



Cátedra Gestión de Datos

Unidad Temática N° 4: Lenguaje SQL

↳ Consulta Multitabla

- *Emparejamiento en Where.*

```
SELECT  E.nroemp,E.apelemp,E.nomemp,C.catego,C.descrip
FROM    Empleados E, Categorías C
WHERE   E.catego = C.id
```



- *Emparejamiento en Join.*

```
SELECT  E.nroemp,E.apelemp,E.nomemp,C.catego,C.descrip
FROM    Empleados E INNER JOIN Categorías C ON E.catego = C.id
```



Cátedra Gestión de Datos

Unidad Temática N° 4: Lenguaje SQL

Composición externa (OUTER JOIN)

➤ Permite extraer filas que no cumplen la condición de emparejamiento.

NroEmp	ApelEmp	NomEmp	Catego	DNI
10235	Sánchez	Miguel	B	12.459.078
09674	Fresno	Edith	NULL	10.648.315
12058	Abraham	Yamila	A	11.568.021

Tabla Empleados

Id	Descrip
C	Operación
B	Administración
A	Gerente

Tabla Categorías

➤ Obtener datos de empleados aún cuando no tienen categoría asignada

```
SELECT  E.aplemp,E.nomemp,C.descrip
FROM    Empleados E
LEFT JOIN Categorías C ON E.catego = C.id
```

```
...RIGHT JOIN Categorías C ON ...
```

```
...CROSS JOIN Categorías C ON ...
```



Relación resultado

ApelEmp	NomEmp	Descrip
Sánchez	Miguel	Administración
Fresno	Edith	null
Abraham	Yamila	Gerente



Cátedra Gestión de Datos

Unidad Temática N° 4: Lenguaje SQL

↪ Autocomposición: combinar una tabla consigo misma

NroEmp	ApelEmp	NomEmp	Catego	NroJefe
10235	Sánchez	Miguel	B	12058
09674	Fresno	Edith	NULL	12058
12058	Abraham	Yamila	A	NULL

Tabla Empleados

FK apunta a
PK en la misma tabla

```
SELECT  E.apelemp,E.nomemp,JEFE.apelemp AS apeJefe
FROM    empleados E
LEFT JOIN empleados JEFE ON E.nrojefe = JEFE.nroemp
```

Simula 2 tablas
con alias

Relación resultado

ApelEmp	NomEmp	ApeJefe
Sánchez	Miguel	Abraham
Fresno	Edith	Abraham
Abraham	Yamila	null



Cátedra Gestión de Datos

Unidad Temática N° 4: Lenguaje SQL

Consultas SUMARIAS

- *Permiten resumir datos de una BD, mediante funciones que aceptan una columna como argumento y producen un dato resumen.*
- **Funciones de COLUMNA disponibles:**
 - *SUM: calcula total de la columna*
 - *AVG : calcula promedio*
 - *COUNT: cuenta cantidad de valores*
 - *MAX: obtiene el máximo valor*
 - *MIN: obtiene el mínimo valor*

Funciones de SQL Server: <http://msdn.microsoft.com/es-es/library/ms190317>



Cátedra Gestión de Datos

Unidad Temática N° 4: Lenguaje SQL

Consultas SUMARIAS

- **Mostrar el monto total facturado**

```
SELECT SUM ( Monto )  
FROM Facturas
```
- **Mostrar la cantidad total de facturas del año 2009**

```
SELECT COUNT ( * )  
FROM Facturas  
WHERE YEAR(fecha)=2009
```
- **Mostrar el Monto Promedio Facturado en la sucursal 3**

```
SELECT AVG(Monto)  
FROM Facturas  
WHERE nrosuc=3
```



Cátedra Gestión de Datos

Unidad Temática N° 4: Lenguaje SQL

Consultas SUMARIAS

- *Mostrar el monto máximo facturado en febrero de 2010.*

```
SELECT MAX(Monto)
FROM Facturas
WHERE YEAR(fecha)=2010 AND MONTH(fecha)=2
```
- *Mostrar la fecha de la 1ra. factura emitida y de la última*

```
SELECT MIN(fecha),MAX(fecha)
FROM Facturas
```



Cátedra Gestión de Datos

Unidad Temática N° 4: Lenguaje SQL

Consultas SUMARIAS

➤ Consideraciones

- **AVG(n):** Promedio y argumento valor numérico y el resultado de otro tipo por la división. Ignora valores nulos.
- **COUNT(n):** Cantidad de valores en columna. Argumento de cualquier tipo. Resultado un entero. Ignora nulos.
- **COUNT(*):** Cantidad de filas, incluyendo duplicadas y nulas
- **SUM(n):** Suma y como argumento valor numérico y resultado del mismo tipo de dato. Ignora valores nulos.
- **MAX(n):** Valor máximo de n. Argumento de cualquier tipo de dato y resultado del mismo tipo. Ignora nulos.
- **MIN(n) :** Valor mínimo de n. Argumento cualquier tipo de dato y resultado del mismo tipo. Ignora nulos.
- **COUNT(DISTINCT n) :** Cantidad de filas con valores no repetidos.
- **TODA LA COLUMNA CON NULL:** SUM, AVG, MIN y MAX devuelven NULL. COUNT devuelve 0.
- **TODAS LAS CONSULTAS MOSTRADAS TIENEN POR RESULTADO UNA FILA DE RESULTADO**



Cátedra Gestión de Datos

Unidad Temática N° 4: Lenguaje SQL

Consultas AGRUPADAS

Ejemplo: Mostrar el monto total comprado por cliente

```
SELECT F.idcliente , SUM (F.total)
FROM Facturas F
GROUP BY F.idcliente
```

Tabla Facturas

idcliente	factura	fecha	total
1	1	10/4/09	100
2	2	13/4/09	250
5	3	13/5/09	630
6	4	14/5/09	470
2	5	15/5/09	200
6	6	17/5/09	300
5	7	17/5/09	400
5	8	1/6/09	800

Tabla Agrupada

idcliente	factura	fecha	total
1	1	10/04/09	100
2	2	13/04/09	250
2	5	15/5/09	200
5	3	13/5/09	630
5	7	17/5/09	400
5	8	1/6/09	800
6	4	14/5/09	470
6	6	17/5/09	300

Resultado Obtenido

idcliente	SUM (total)
1	100
2	450
5	1830
6	770



Cátedra Gestión de Datos

Unidad Temática N° 4: Lenguaje SQL

Consultas con GROUP/HAVING

Ejemplo: Obtener Idcliente y total comprado por cliente, durante mayo'2009 y cuando el total supera los \$500

```
SELECT F.idcliente , SUM (F.total)
FROM Facturas F
WHERE MONTH(F.fecha)=5 AND YEAR(F.fecha)=2009
GROUP BY F.idcliente
HAVING SUM(F.total) > 500
```

Tabla Facturas

idcliente	factura	fecha	total
1	1	10/4/09	100
2	2	13/4/09	250
5	3	13/5/09	630
6	4	14/5/09	470
2	5	15/5/09	200
6	6	17/5/09	300
5	7	17/5/09	400
5	8	1/6/09	800

Tabla Agrupada

idcliente	factura	fecha	total
2	5	15/5/09	200
5	3	13/5/09	630
5	7	17/5/09	400
6	4	14/5/09	470
6	6	17/5/09	300

Resultado Obtenido

idcliente	SUM (total)
5	1030
6	770



Cátedra Gestión de Datos

Unidad Temática N° 4: Lenguaje SQL

Consultas con GROUP/HAVING

↩ Consideraciones

- *El motor devuelve una fila por cada grupo conformado.*
- *WHERE incluye condiciones para las filas.*
- *HAVING condiciona al resultado a nivel de grupo, lo que puede restar filas en el resultado.*
- **NUNCA** va una función sumaria (SUM, COUNT, MIN, MAX, AVG) en WHERE.
- Las funciones sumarias **SIEMPRE** van en HAVING.
- *HAVING comúnmente va con GROUP BY, en casos extraordinarios puede estar solo y en ese caso el grupo es uno.*
- *El resultado de una consulta con agrupamiento sale ordenado por los atributos del GROUP BY.*
- *Los atributos de la lista de selección del SELECT **DEBEN** estar todos en la cláusula GROUP BY.*