
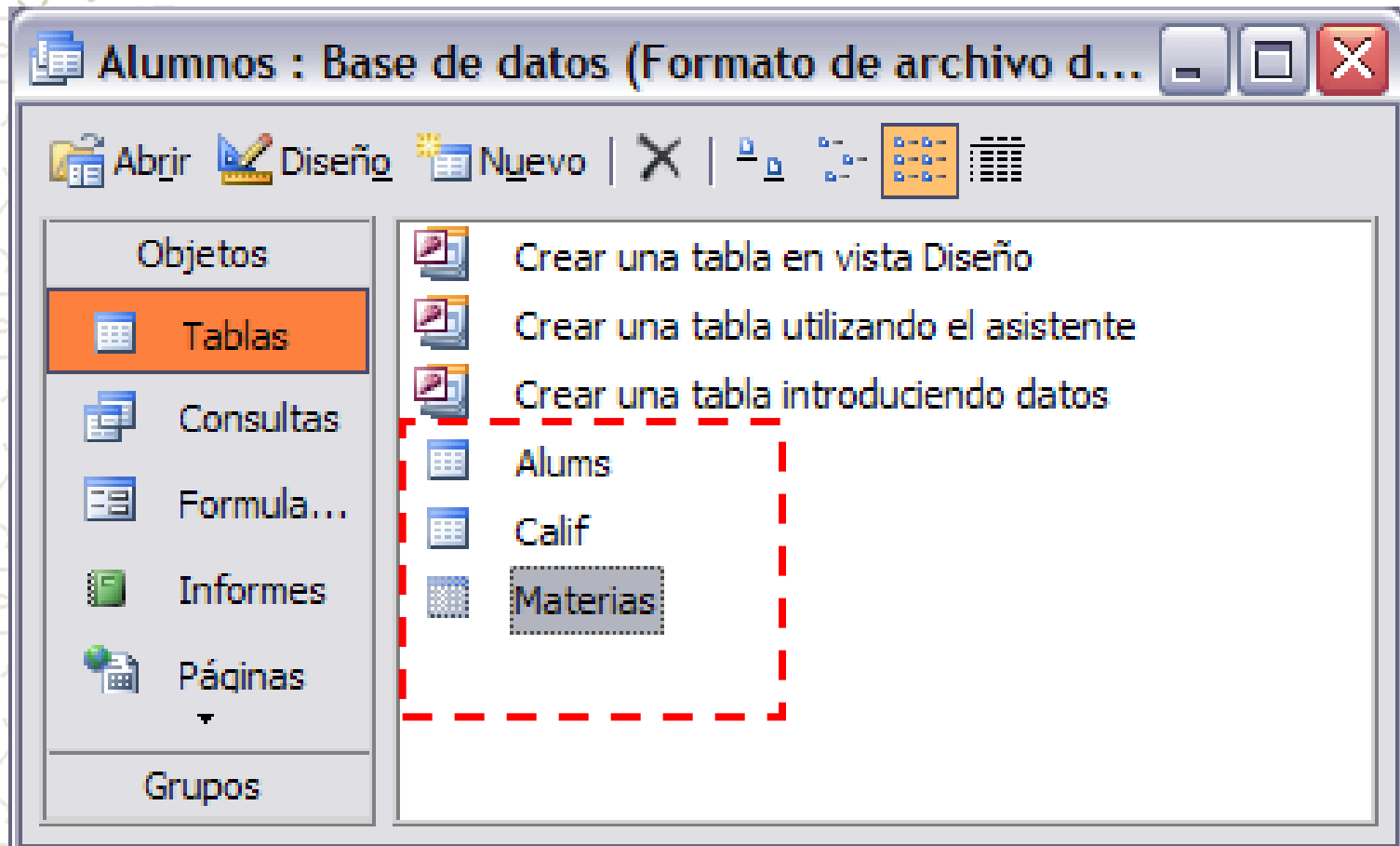


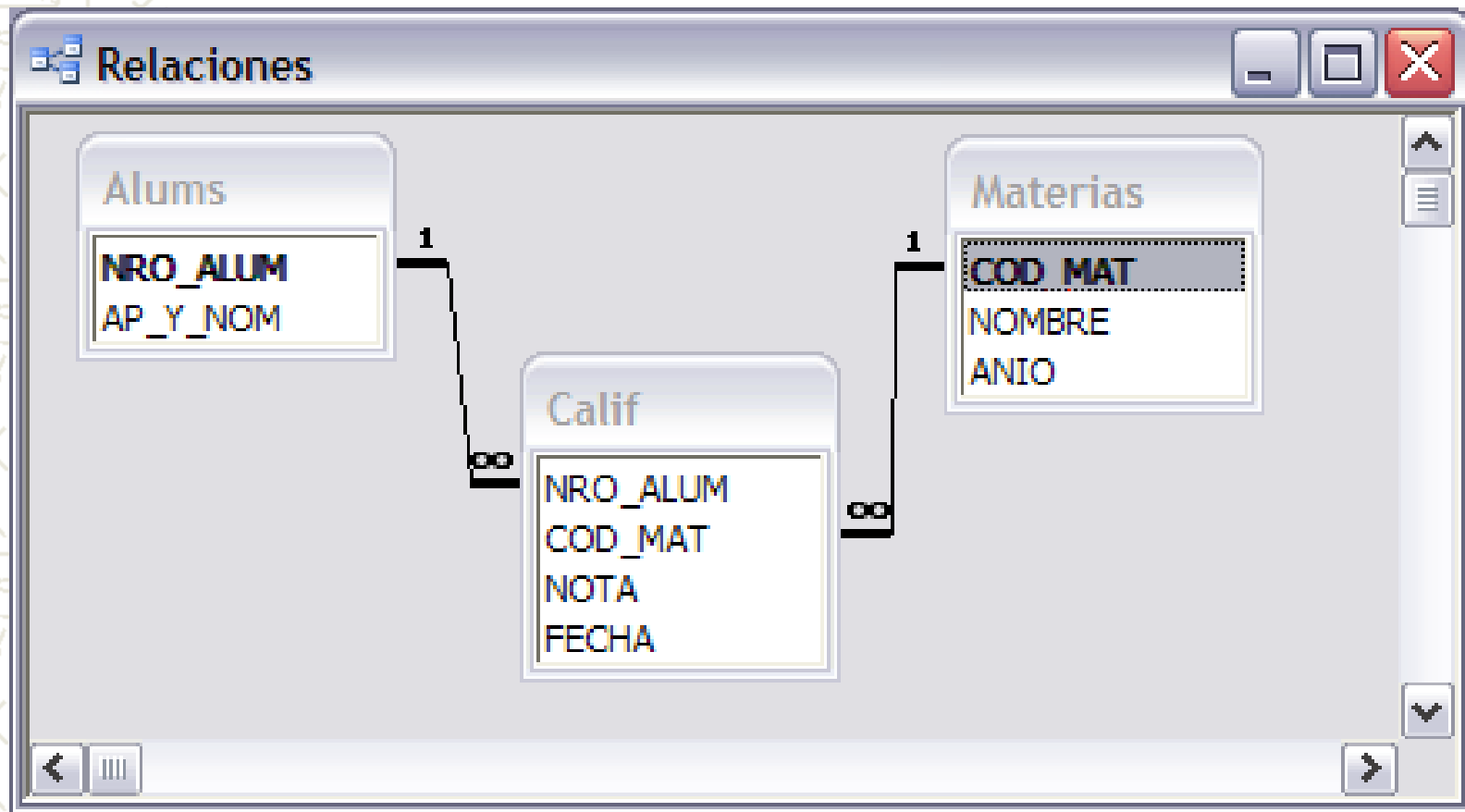
La instrucción SELECT

 **SELECT** [**distinct**] lista_de_expresiones
from lista_de_tablas
[**where** condición_de_selección]
[**group by** lista_de_columnas]
[**having** condición_de_selección_de_grupos]
[**order by** lista_de_columnas]

BBDD Alumnos.mdb



BBDD Alumnos.mdb



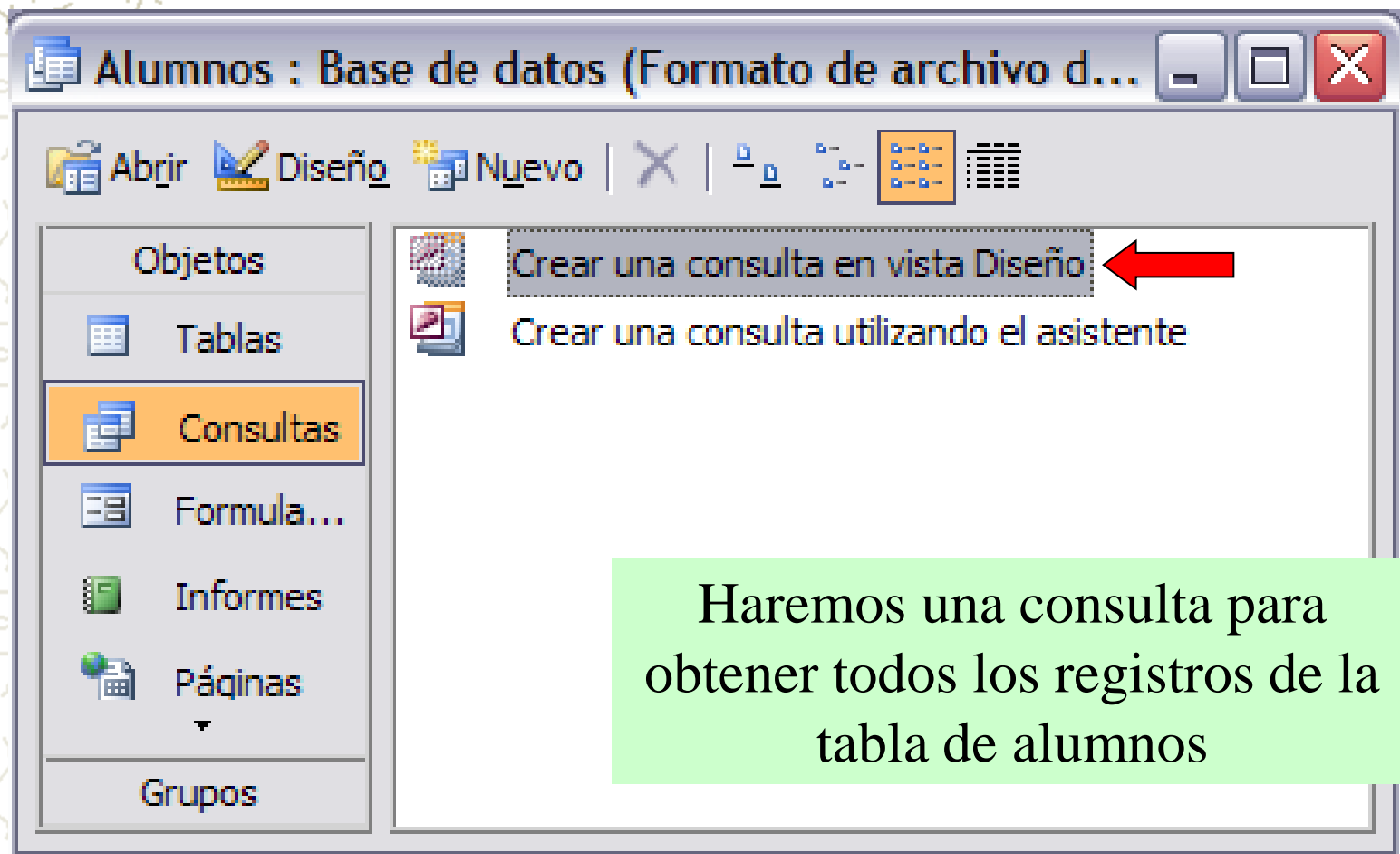
Select - from

✚ **SELECT** Lista_de_campos
FROM nombre_de_la_tabla

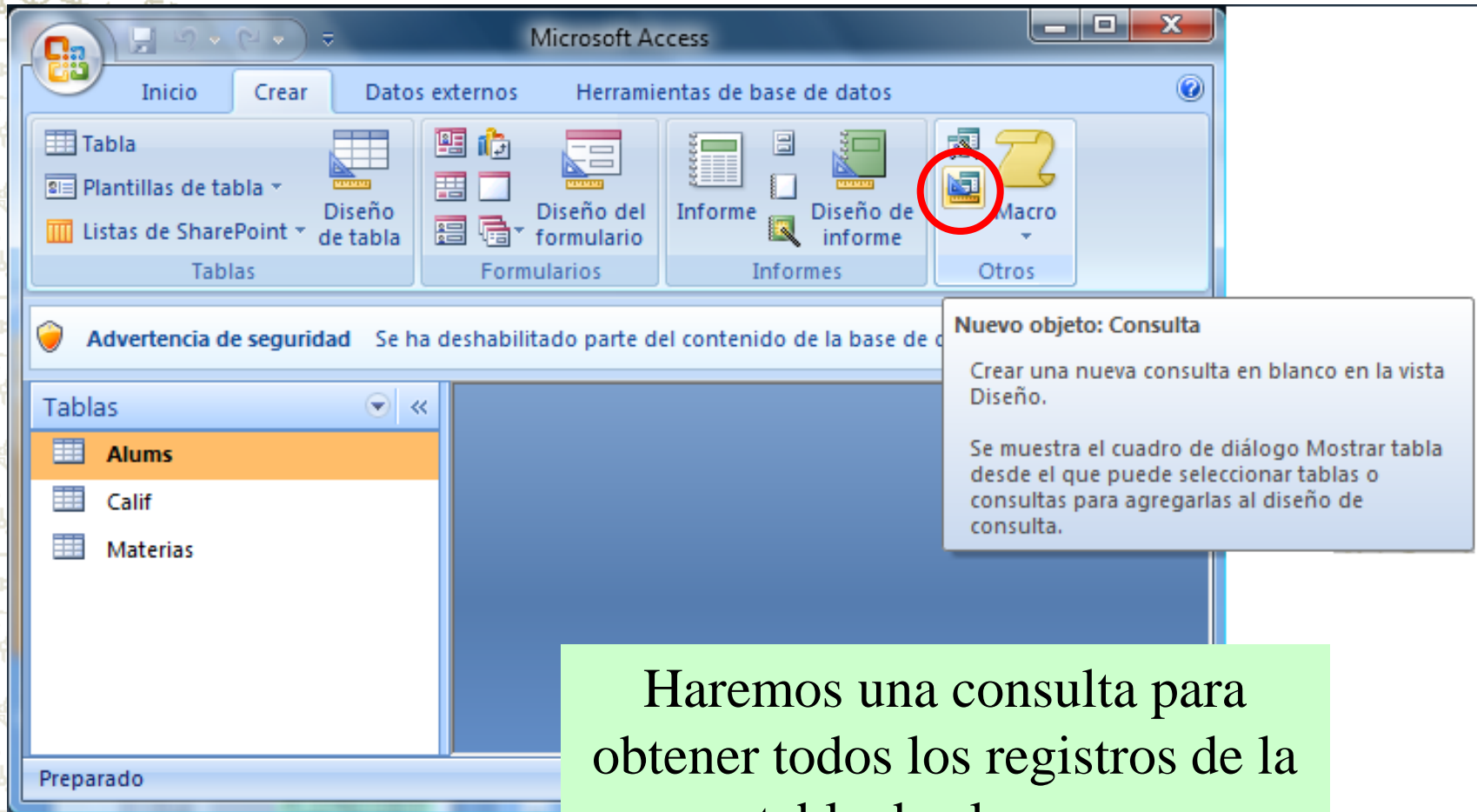
Permite obtener los campos (columnas) indicados para TODOS los registros de la tabla.

✚ **SELECT * FROM** nombre_de_la_Tabla
Retorna la tabla completa. El carácter * representa todas las columnas.

Consultas desde Access 2003

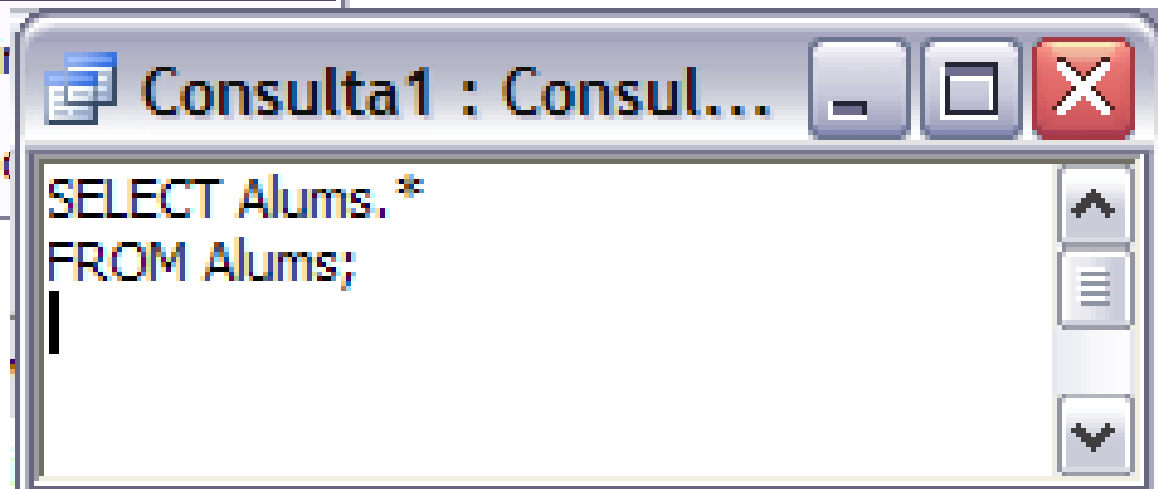
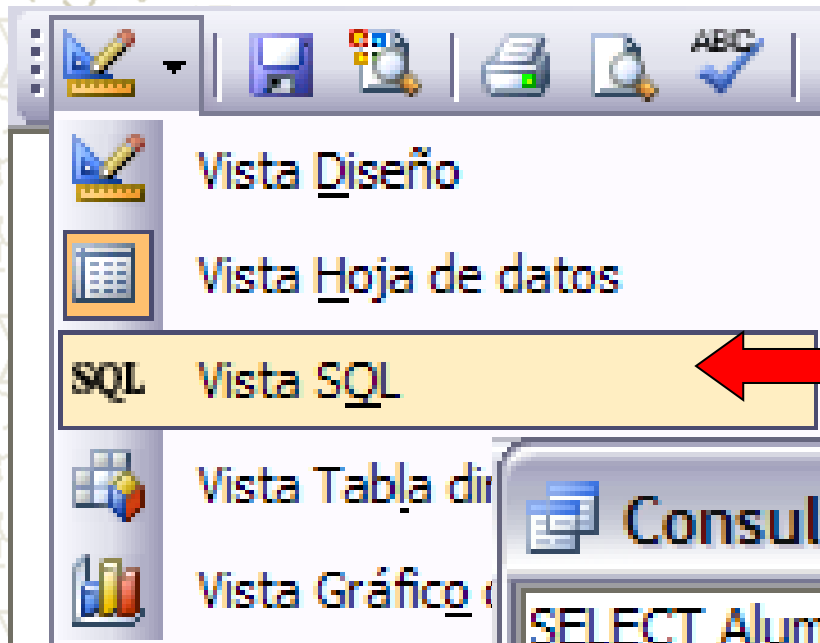


Consultas desde Access 2007

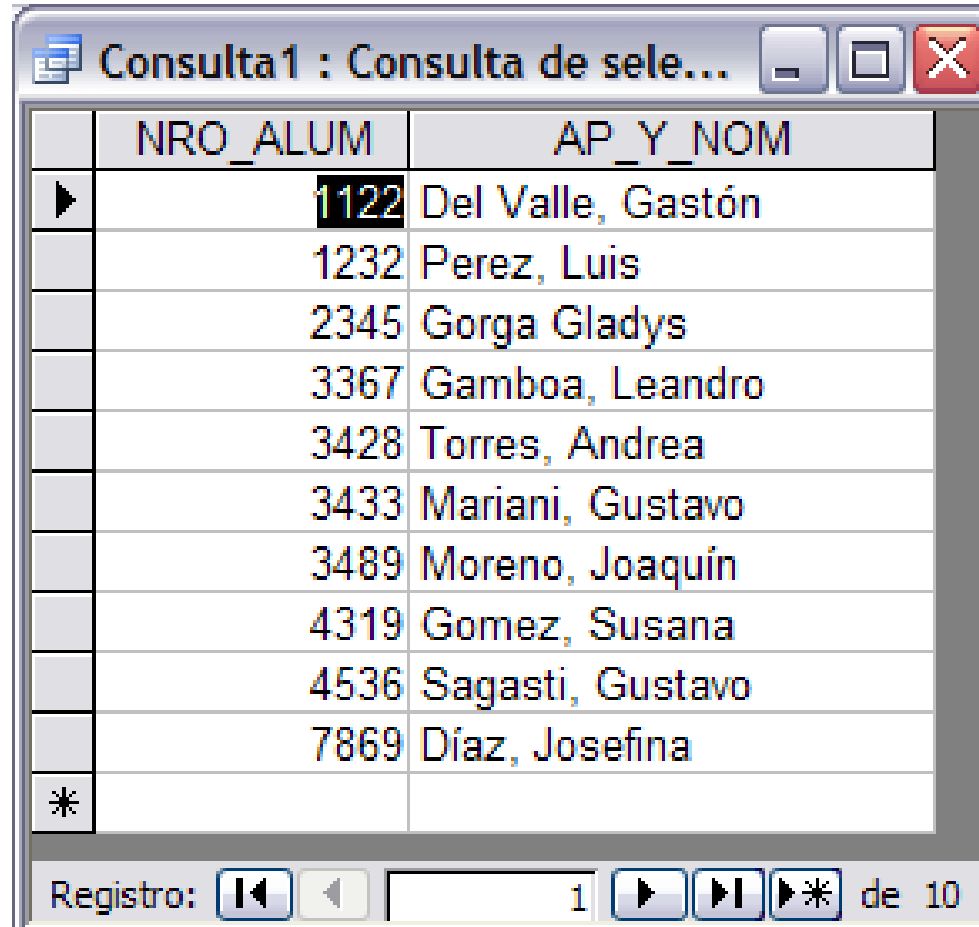


Haremos una consulta para
obtener todos los registros de la
tabla de alumnos

Visualización de la sintaxis SQL



Resultado de la consulta



	NRO_ALUM	AP_Y_NOM
▶	1122	Del Valle, Gastón
	1232	Perez, Luis
	2345	Gorga Gladys
	3367	Gamboa, Leandro
	3428	Torres, Andrea
	3433	Mariani, Gustavo
	3489	Moreno, Joaquín
	4319	Gomez, Susana
	4536	Sagasti, Gustavo
	7869	Díaz, Josefina
*		

Registro: 1 de 10

Ejemplo

SELECT * from Materias

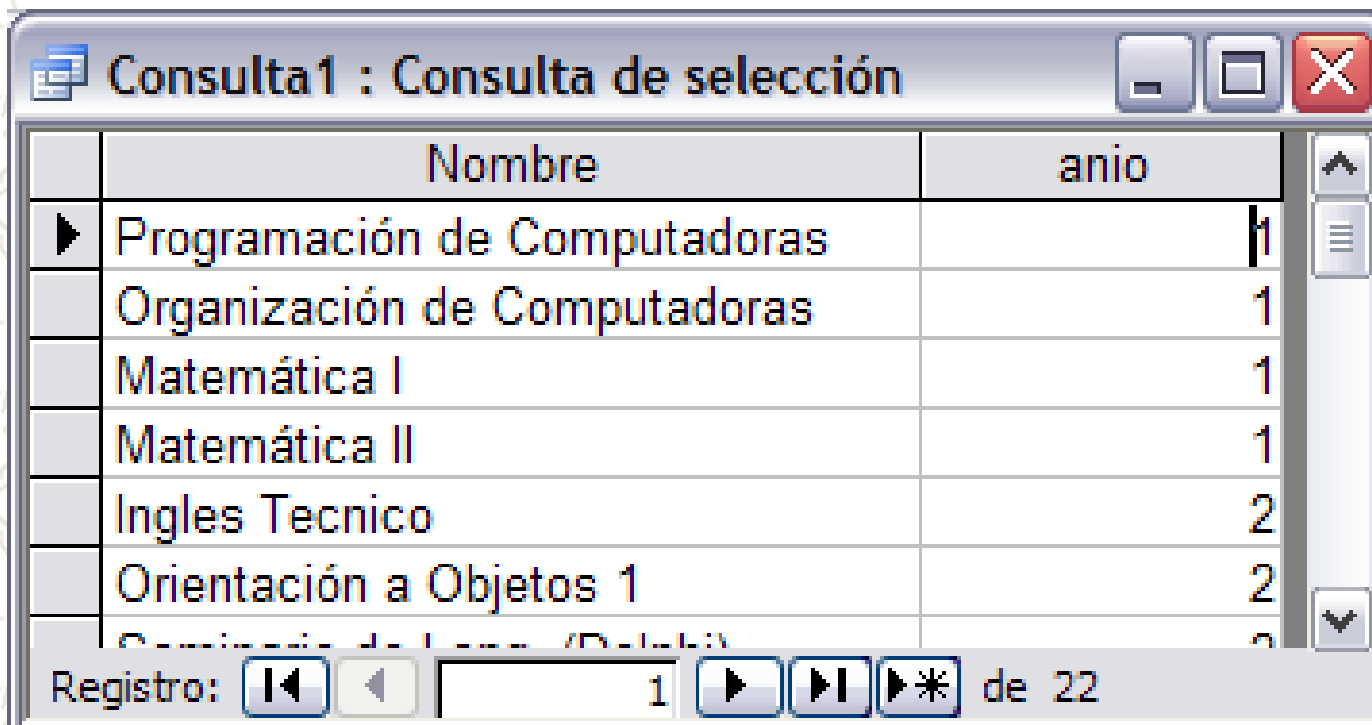
Consulta1 : Consulta de selección

	COD_MAT	NOMBRE	ANIO
▶	1	Programación de Computadoras	1
	2	Organización de Computadoras	1
	3	Matemática I	1
	4	Matemática II	1
	5	Ingles Tecnico	2
	6	Orientación a Objetos 1	2

Registro: 1 de 22

Ejemplo

SELECT nombre, anio from Materias

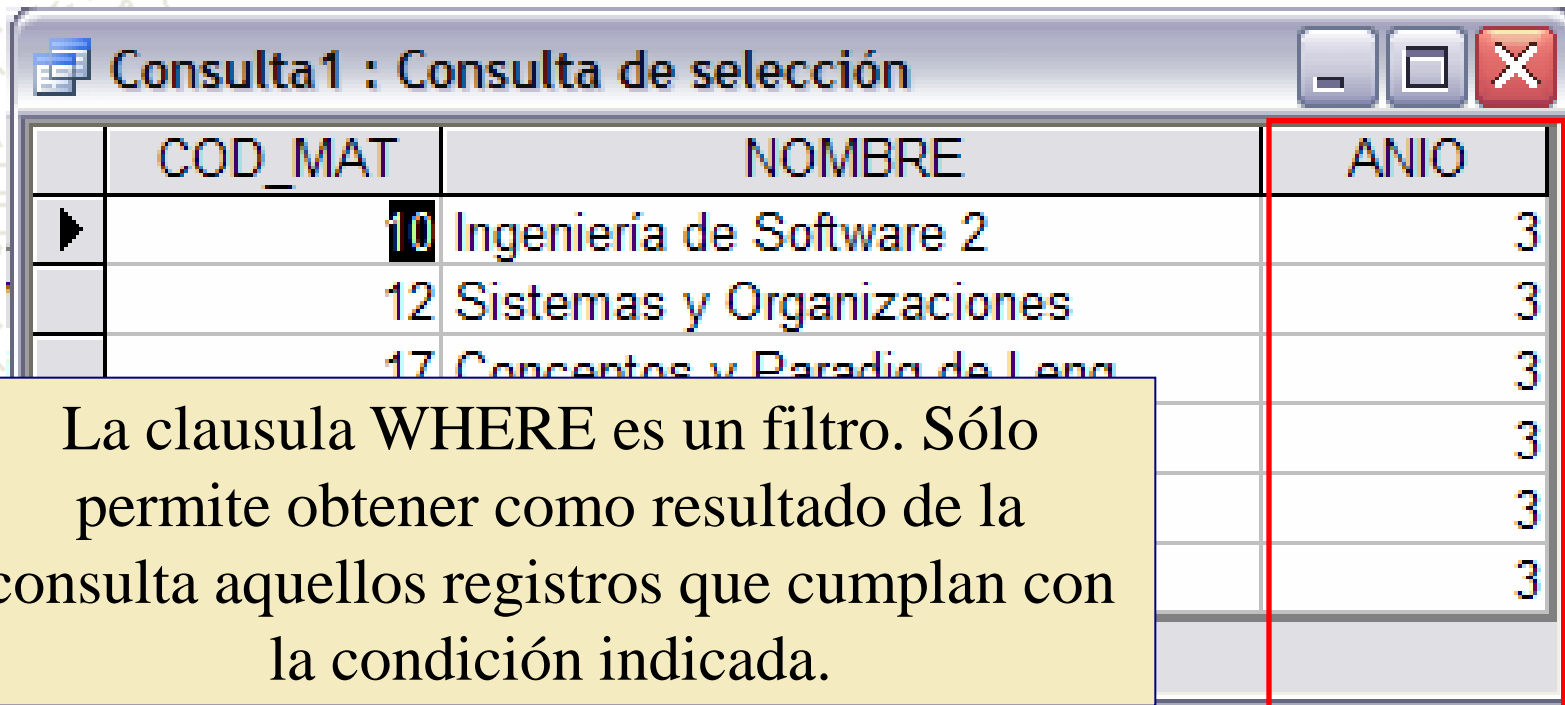


	Nombre	anio
▶	Programación de Computadoras	1
	Organización de Computadoras	1
	Matemática I	1
	Matemática II	1
	Ingles Tecnico	2
	Orientación a Objetos 1	2
	Comunicación de Datos (Delat)	2

Registro: 1 de 22

La condición de selección

🔗 **SELECT * from Materias where Anio=3**



Consulta1 : Consulta de selección

	COD_MAT	NOMBRE	ANIO
▶	10	Ingeniería de Software 2	3
	12	Sistemas y Organizaciones	3
	17	Conceptos y Paradigmas de Leng	3
			3
			3
			3
			3

La clausula WHERE es un filtro. Sólo permite obtener como resultado de la consulta aquellos registros que cumplan con la condición indicada.

La condición de selección

SELECT * from Calif
where (nota>=4) and (fecha <= #04/01/2007#)

NRO_ALUM	COD_MAT	NOTA	FECHA
1232	3	10	02/02/2007
3433	3	10	02/02/2007
4319	3	10	02/02/2007
7869	9	7	02/02/2007
2345	3	7	02/03/2007
3428	7	7	02/03/2007
1232	1	8	01/03/2007

Operadores de String

✳ Operador like

- Permite averiguar si una cadena satisface o no cierto patrón de caracteres
- Su segundo operando es el patrón, un string dentro del cual pueden incluirse los siguientes caracteres:

Carácter

Significado

*

cero o más caracteres arbitrarios

?

Un carácter cualquiera

Operadores de Strings

✚ Ejemplos

Expresión	String aceptado	String no aceptado
Nombre like '* Perez'	'Cecilia Perez'	'Perez Luciano'
Fruta like 'Manzana?'	'Manzanas'	'Manzana'

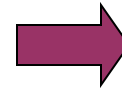
Operadores de String

✚ **SELECT** ap_y_nom from Alums
where (ap_y_nom like "M*") or (ap_y_nom like "*b*")
or (ap_y_nom like "Torres, Andre?")

ap_y_nom
Moreno, Joaquín
Mariani, Gustavo
Gamboa, Leandro
Torres, Andrea

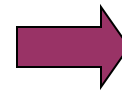
El valor nulo

✶ **SELECT * from Alums**
where ap_y_nom is null



Quedan sólo los
alumnos que no
tienen nombre.

✶ **SELECT * from Calif**
where Fecha is not null



Quedan sólo los
registros que
tienen una fecha
cargada.

Eliminando duplicados

```
SELECT nro_alum  
from Calif  
where nota >= 4
```



```
SELECT distinct nro_alum  
from Calif  
where nota >= 4
```

nro_alum
1232
1232
2345
3433
4319
7869
2345
3428
1232
3428
3428

Eliminando duplicados

```
SELECT nro_alum  
from Calif  
where nota >= 4
```

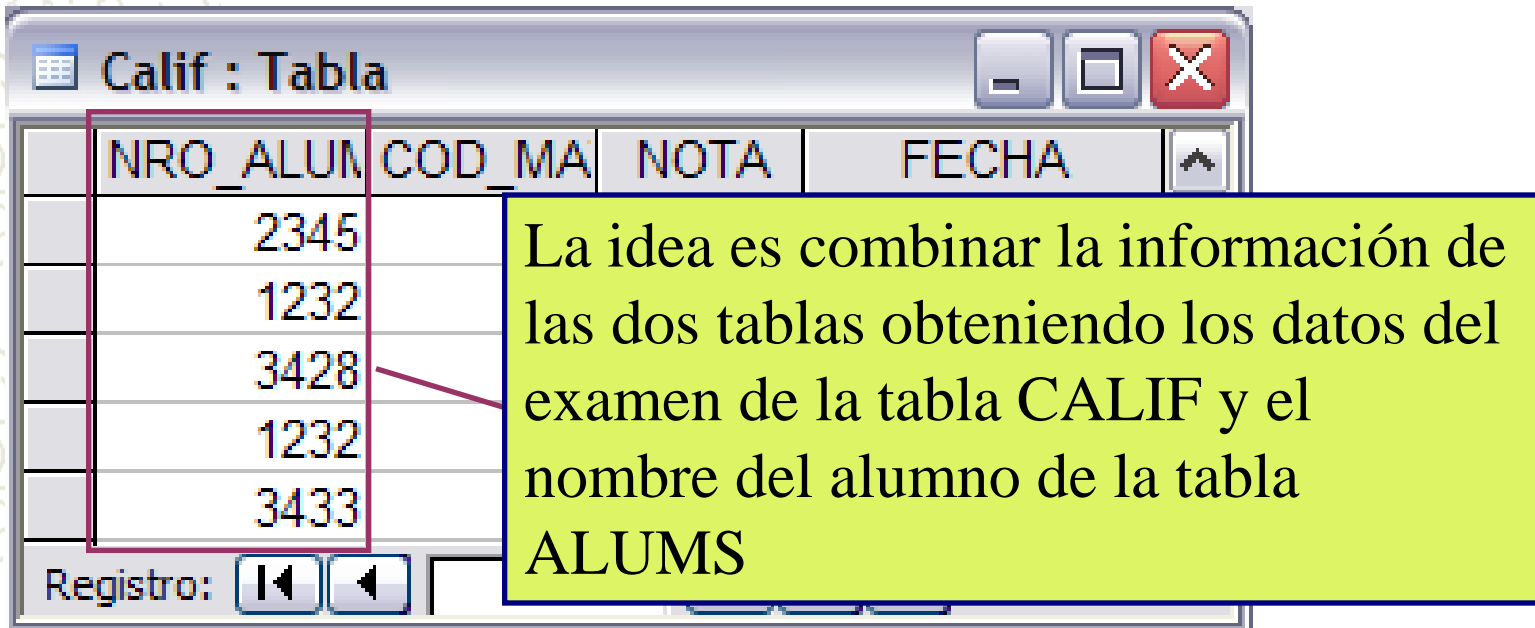
```
SELECT distinct nro_alum  
from Calif  
where nota >= 4
```



nro_alum
1232
2345
3428
3433
4319
7869

Consultas con varias tablas

- Supongamos que para cada calificación de la tabla CALIF se desea conocer el nombre del alumno



La idea es combinar la información de las dos tablas obteniendo los datos del examen de la tabla CALIF y el nombre del alumno de la tabla ALUMS

NRO_ALUM	COD_MA	NOTA	FECHA
2345			
1232			
3428			
1232			
3433			

Registro: |< <

Consultas con varias tablas

✳ Ud. podría llegar a pensar que esta consulta resolverá su problema:

```
select A.ap_y_nom, C.Cod_mat, C.Nota, C.fecha  
from Alums A, Calif C
```

Esta letra a continuación del nombre de la tabla funciona como un alias que permite, dentro de la consulta, reconocer de qué tabla se trata.

Producto Cartesiano

select A.ap_y_nom, C.Cod_mat, C.Nota, C.fecha
from Alums A, Calif C

	NRO_ALUM	AP_Y_NOM
+	1122	Del Valle, Gastón
+	1232	Perez, Luis
+	2345	Gorga Gladys M
+	3367	Gamboa, Leandro
+	3428	Torres, Andrea
+	3433	Mariani, Gustavo
+	3489	Moreno, Joaquín
+	4319	Gomez, Susana
+	4536	Sagasti, Gustavo Mar
+	7869	Díaz, Josefina

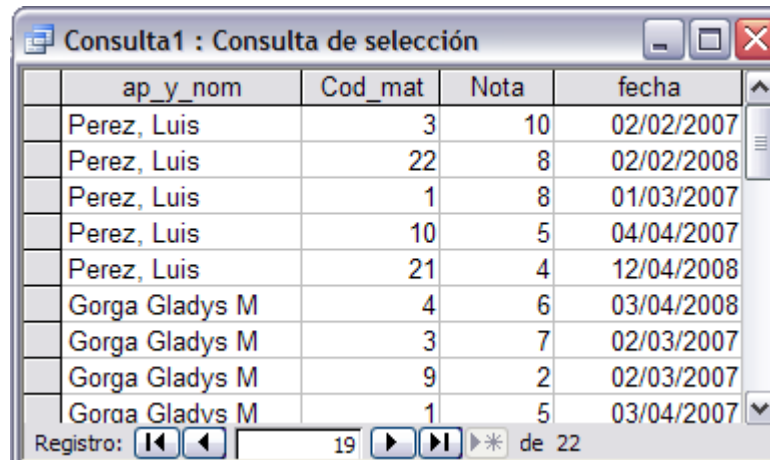
	ap_y_nom	Cod_mat	Nota	fecha
	Gorga Gladys M	3	10	02/02/2007
	Perez, Luis	3	10	02/02/2007
	Sagasti, Gustavo Marti	3	10	02/02/2007
	Gomez, Susana	3	10	02/02/2007
	Moreno, Joaquín	3	10	02/02/2007
	Mariani, Gustavo	3	10	02/02/2007
	Del Valle, Gastón	3	10	02/02/2007
	Gamboa, Leandro	3	10	02/02/2007
	Díaz, Josefina	3	10	02/02/2007
	Torres, Andrea	3	10	02/02/2007
▶	Gorga Gladys M	22	8	02/02/2008
	Perez, Luis	22	8	02/02/2008

Cada calificación se repite tantas veces como alumnos haya

Producto cartesiano

```
select A.ap_y_nom, C.Cod_mat, C.N  
from Alums A, Calif C  
where A.nro_alum=C.nro_alum
```

Filtra los registros incorrectamente asociados por el producto cartesiano de las tablas.



	ap_y_nom	Cod_mat	Nota	fecha	
	Perez, Luis	3	10	02/02/2007	
	Perez, Luis	22	8	02/02/2008	
	Perez, Luis	1	8	01/03/2007	
	Perez, Luis	10	5	04/04/2007	
	Perez, Luis	21	4	12/04/2008	
	Gorga Gladys M	4	6	03/04/2008	
	Gorga Gladys M	3	7	02/03/2007	
	Gorga Gladys M	9	2	02/03/2007	
	Gorga Gladys M	1	5	03/04/2007	

INNER JOIN

```
select A.ap_y_nom, C.Cod_mat,  
        C.Nota, C.fecha  
from Calif C inner join Alums A  
on A.nro_alum=C.nro_alum
```

Condición que establece la
relación entre las tablas

Ordenando los resultados

```
SELECT A.ap_y_nom, C.Cod_mat, C.Nota, C.fecha
from Calif C, Alums A
where (A.nro_alum=C.nro_alum) and
      (ap_y_nom like "*J*")
order by ap_y_nom
```

Rehacer esta
consulta
utilizando
INNER JOIN

ap_y_nom	Cod_mat	Nota	fecha
Díaz, Josefina	9	7	02/02/2007
Moreno, Joaquín	9	2	04/04/2008

Ordenando los resultados


```
SELECT A.ap_y_nom, C.Cod_mat, C.Nota, C.fecha  
from Calif C inner join Alums A  
      on A.nro_alum=C.nro_alum  
where (ap_y_nom like "*J*")  
order by ap_y_nom
```

ap_y_nom	Cod_mat	Nota	fecha
Díaz, Josefina	9	7	02/02/2007
Moreno, Joaquín	9	2	04/04/2008

Ordenando los resultados

```
SELECT ap_y_nom, cod_mat, nota, fecha
from Calif C, Alums A
where (C.nro_alum=A.nro_alum)
order by fecha, ap_y_nom desc
```

Rehacer esta
consulta
utilizando
INNER JOIN

	ap_y_nom	Cod_mat	Nota	fecha
	Torres, Andrea	9	2	02/02/2007
	Perez, Luis	3	10	02/02/2007
	Mariani, Gustavo	3	10	02/02/2007
	Gomez, Susana	3	10	02/02/2007
	Díaz, Josefina	9	7	02/02/2007
	Perez, Luis	1	8	01/03/2007
	Torres, Andrea	7	7	02/03/2007
Registro:   16    de 22				

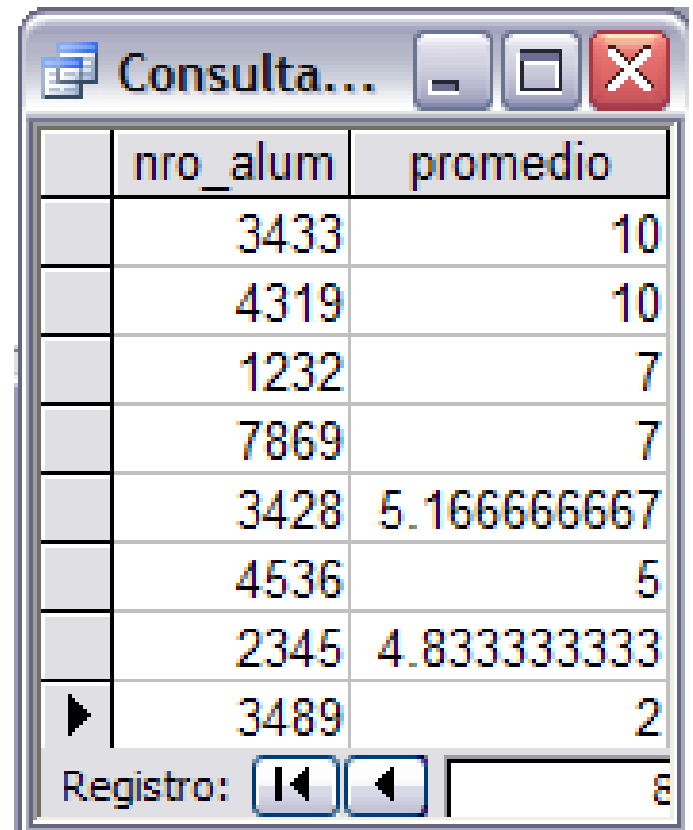
Ordenando los resultados

```
SELECT A.ap_y_nom, C.Cod_mat, C.Nota, C.fecha  
from Calif C inner join Alums A  
  on A.nro_alum=C.nro_alum  
order by fecha, ap_y_nom desc
```


	ap_y_nom	Cod_mat	Nota	fecha
	Torres, Andrea	9	2	02/02/2007
	Perez, Luis	3	10	02/02/2007
	Mariani, Gustavo	3	10	02/02/2007
	Gomez, Susana	3	10	02/02/2007
	Díaz, Josefina	9	7	02/02/2007
	Perez, Luis	1	8	01/03/2007
	Torres, Andrea	7	7	02/03/2007
Registro:   16    de 22				

El uso de grupos

```
select C.nro_alum, avg(nota) as promedio  
from Calif C inner join Alums A  
  on A.nro_alum=C.nro_alum  
group by C.nro_alum  
order by 2 desc, C.nro_alum
```



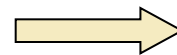
	nro_alum	promedio
	3433	10
	4319	10
	1232	7
	7869	7
	3428	5.166666667
	4536	5
	2345	4.833333333
▶	3489	2

Registro:  8

El uso de grupos

- ✦ Cada fila de la consulta es el resultado de operar sobre un GRUPO de registros

nro_alum	nota
1232	8
1232	10
1232	5
1232	4
1232	8
2345	2
2345	6

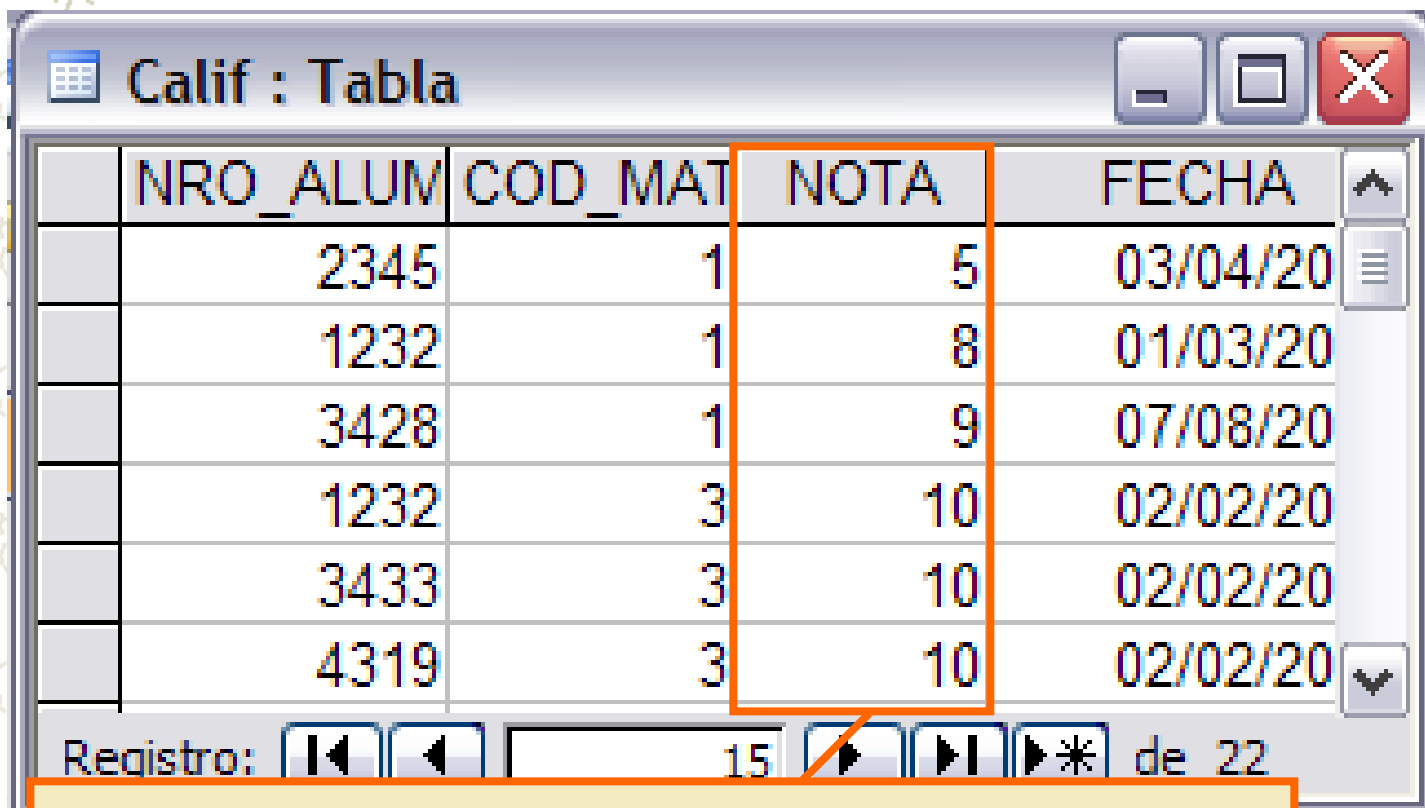


nro_alum	promedio
3433	10
4319	10
1232	7
7869	7
3428	5.166666667
4536	5
2345	4.833333333
3489	2

Funciones

Función	Significado
count	Cantidad de valores no nulo en el grupo
min	El valor mínimo de la columna dentro del grupo
max	El valor máximo de la columna dentro del grupo
sum	La suma de los valores de la columna dentro del grupo
avg	El promedio de la columna dentro del grupo

CALIF



	NRO_ALUM	COD_MAT	NOTA	FECHA
	2345	1	5	03/04/20
	1232	1	8	01/03/20
	3428	1	9	07/08/20
	1232	3	10	02/02/20
	3433	3	10	02/02/20
	4319	3	10	02/02/20

Registro: 15 de 22

Calcular para el campo NOTA las siguientes operaciones: sum, avg, count, min y max

Funciones

```
SELECT sum(nota) as suma, avg(nota) as promedio ,  
count(*) as cant, min(nota) as minimo,  
max(nota) as maximo  
from Calif
```

suma	promedio	cant	minimo	maximo
129	5.8636363636	22	2	10

La cláusula having

```
SELECT C.nro_alum, avg(Nota) as Promedio  
from Alums A INNER JOIN Calif C ON A.NRO_ALUM = C.NRO_ALUM  
group by C.nro_alum  
having avg(nota) >= 6  
order by 2
```

NRO_ALUM	Promedio
7869	7
1232	7
4319	10
3433	10

Es un filtro para las columnas calculadas a través de una expresión.

Consulta con parámetro

```
SELECT C.Cod_mat, C.Nota, C.fecha  
from Calif C inner join Alums A  
    on A.nro_alum=C.nro_alum  
where C.Nro_Alum= :dato
```

**El parámetro se indica con :
delante del identificador**