

TRABAJO PRÁCTICO – Modulo 5 | Parte 4

Subconsultas

Alumno: Ignacio Figueroa – 45.406.120

Tecnicatura Universitaria en Programación – UTN

Materia: Base de Datos I

Comisión: 8

Introducción:

En este trabajo se practican las **subconsultas** o **subqueries**, que permiten anidar una consulta dentro de otra.

Las subconsultas son útiles para obtener datos intermedios, como promedios, conteos o valores de referencia, y luego utilizarlos en la consulta principal.

Ejercicio 1: Alumnos con edad menor al promedio

Mostrar el nombre y apellido de los alumnos que tienen una edad menor al promedio general.

The screenshot shows a SQL IDE window titled "SQL File 2*" with a query editor containing the following SQL code:

```
-- 1
select nombre, apellido
from alumnos
where edad < (select avg(edad) from alumnos);
```

Below the query editor, the "Result Grid" displays the results of the query:

nombre	apellido
Ana	Gómez

The "Output" panel at the bottom shows the execution details:

#	Time	Action	Message
1	13:38:39	select nombre, apellido from alumnos where edad < (select avg(edad) from alumnos) LIMIT 0, 1000	1 row(s) returned

La subconsulta calcula el promedio de edades, y la consulta principal muestra solo los alumnos cuya edad es menor que ese promedio.

Ejercicio 2: Carreras sin asignaturas

Mostrar el id y nombre de las carreras que no tienen asignaturas asociadas.

The screenshot shows a SQL IDE window titled "SQL File 2*" with a query editor containing the following SQL code:

```
-- 2
select id, nombre_carrera
from carreras
where id not in (select id_carrera from asignaturas);
```

Below the query editor, the "Result Grid" displays the results of the query:

id	nombre_carrera
NULL	NULL

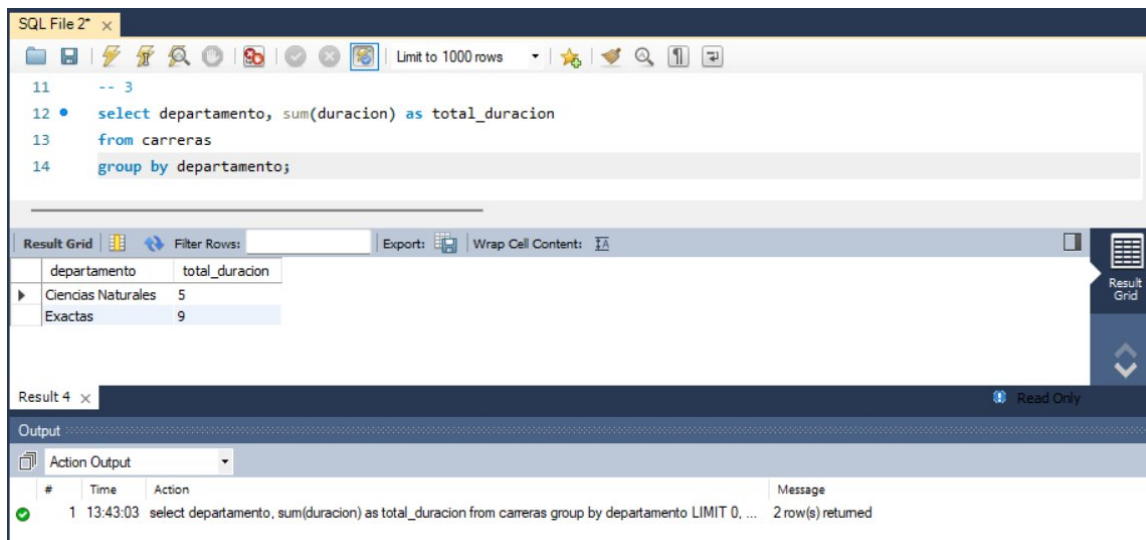
The "Output" panel at the bottom shows the execution details:

#	Time	Action	Message
1	13:40:59	select id, nombre_carrera from carreras where id not in (select id_carrera from asignaturas) LIMIT 0, 1...	0 row(s) returned

La subconsulta obtiene todos los id_carrera que existen en asignaturas. La consulta principal muestra las carreras cuyos id **no** aparecen allí (es decir, sin materias).

Ejercicio 3: Suma de duración total por departamento

Mostrar el nombre del departamento y la suma total de la duración de sus carreras.



The screenshot shows a SQL IDE window titled "SQL File 2". The query is as follows:

```
-- 3
select departamento, sum(duracion) as total_duracion
from carreras
group by departamento;
```

The "Result Grid" displays the following data:

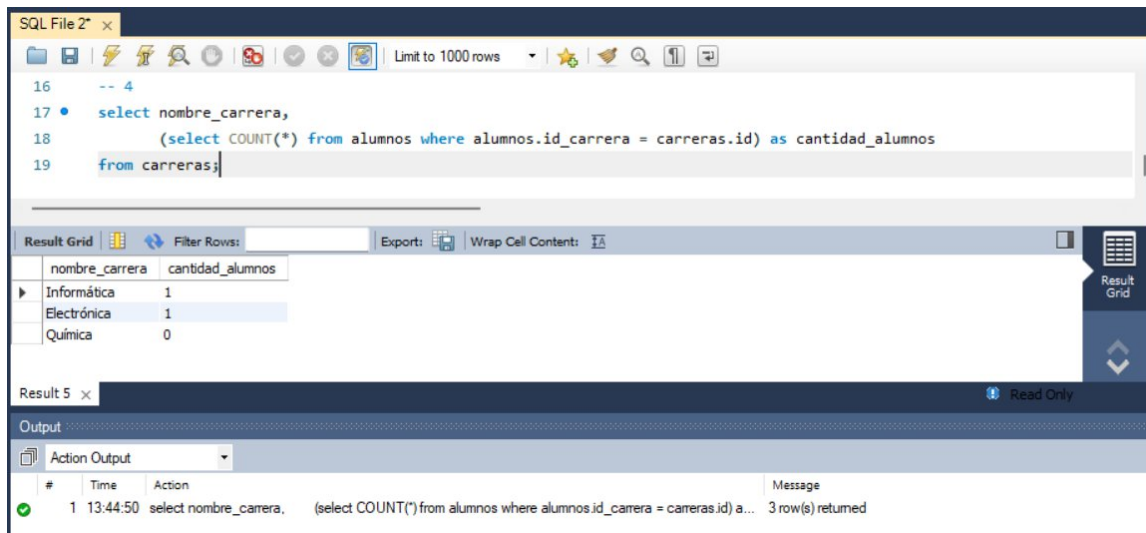
departamento	total_duracion
Ciencias Naturales	5
Exactas	9

The "Output" pane shows the execution message: "1 13:43:03 select departamento, sum(duracion) as total_duracion from carreras group by departamento LIMIT 0, ... 2 row(s) returned".

Aunque no requiere una subconsulta estricta, se agrupan las carreras por departamento para calcular la suma total de años.

Ejercicio 4: Cantidad de alumnos por carrera

Mostrar el nombre de la carrera y la cantidad de alumnos que la cursan.



The screenshot shows a SQL IDE window titled "SQL File 2". The query is as follows:

```
-- 4
select nombre_carrera,
(select COUNT(*) from alumnos where alumnos.id_carrera = carreras.id) as cantidad_alumnos
from carreras;
```

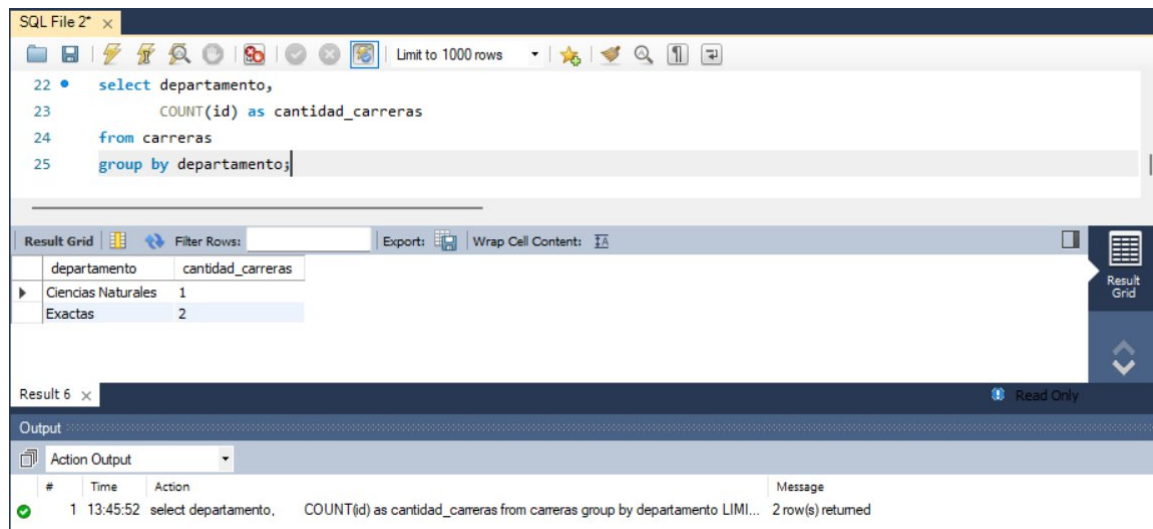
The "Result Grid" displays the following data:

nombre_carrera	cantidad_alumnos
Informática	1
Electrónica	1
Química	0

The "Output" pane shows the execution message: "1 13:44:50 select nombre_carrera, (select COUNT(*) from alumnos where alumnos.id_carrera = carreras.id) a... 3 row(s) returned".

La subconsulta cuenta cuántos alumnos pertenecen a cada carrera, y se muestra junto al nombre de la carrera.

Ejercicio 5: Cantidad de carreras por departamento



SQL File 2

```

22 • select departamento,
23       COUNT(id) as cantidad_carreras
24 from carreras
25 group by departamento;

```

Result Grid

departamento	cantidad_carreras
Ciencias Naturales	1
Exactas	2

Result 6

Output

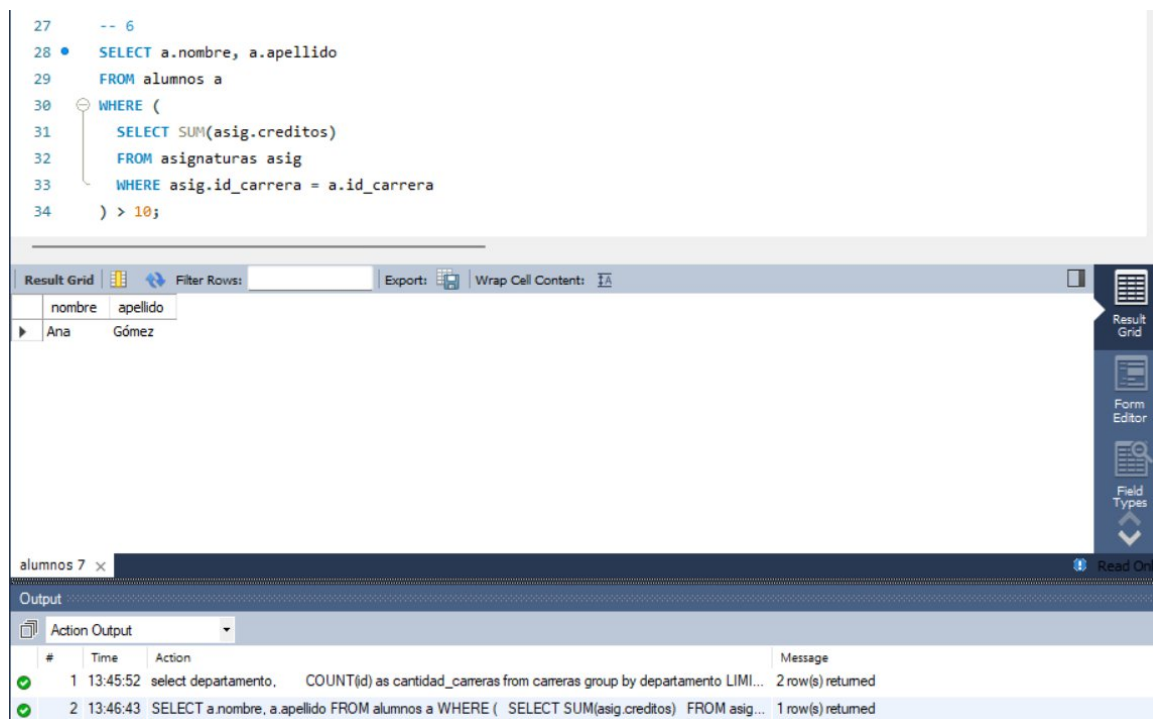
Action Output

#	Time	Action	Message
1	13:45:52	select departamento, COUNT(id) as cantidad_carreras from carreras group by departamento LIM...	2 row(s) returned

Cuenta cuántas carreras hay por cada departamento.
(Es una agrupación, no una subconsulta, pero corresponde al ejercicio).

Ejercicio 6: Alumnos con créditos mayores a 10

Mostrar el nombre y apellido de los alumnos que tienen créditos totales mayores a 10.



SQL File 7

```

27 -- 6
28 • SELECT a.nombre, a.apellido
29 FROM alumnos a
30 WHERE (
31     SELECT SUM(asig.creditos)
32     FROM asignaturas asig
33     WHERE asig.id_carrera = a.id_carrera
34 ) > 10;

```

Result Grid

nombre	apellido
Ana	Gómez

alumnos 7

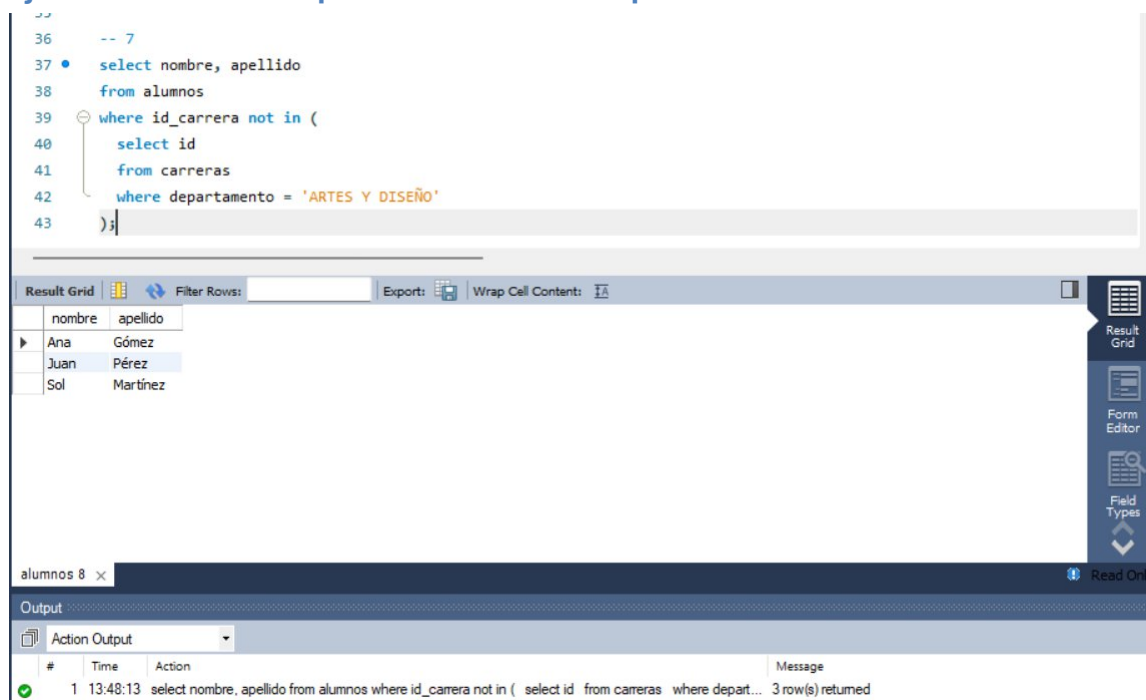
Output

Action Output

#	Time	Action	Message
1	13:45:52	select departamento, COUNT(id) as cantidad_carreras from carreras group by departamento LIM...	2 row(s) returned
2	13:46:43	SELECT a.nombre, a.apellido FROM alumnos a WHERE (SELECT SUM(asig.creditos) FROM asig...	1 row(s) returned

La subconsulta suma los créditos de todas las asignaturas de la carrera de cada alumno, y se muestran solo los alumnos cuyo total supera 10.

Ejercicio 7: Alumnos que no cursan en el departamento de “ARTES Y DISEÑO”



The screenshot shows a database query interface. The SQL query is as follows:

```

36  -- 7
37  • select nombre, apellido
38  from alumnos
39  where id_carrera not in (
40  select id
41  from carreras
42  where departamento = 'ARTES Y DISEÑO'
43  );

```

The results are displayed in a table with columns 'nombre' and 'apellido':

nombre	apellido
Ana	Gómez
Juan	Pérez
Sol	Martínez

The interface also shows an 'Output' section with the following message:

```

1 13:48:13 select nombre, apellido from alumnos where id_carrera not in ( select id from carreras where depart... 3 row(s) returned

```

La subconsulta selecciona las carreras de “ARTES Y DISEÑO”, y la principal muestra solo los alumnos cuya carrera **no** pertenece a ese departamento.

Ejercicio 8: Asignaturas del promedio de los alumnos más jóvenes

Mostrar el nombre de las asignaturas donde se encuentre el promedio de los alumnos más jóvenes.

The screenshot displays a database management interface. At the top, a SQL query is written in a text editor:

```

44
45 -- 8
46 • select nombre
47 from asignaturas
48 where id_carrera in (
49     select id_carrera
50     from alumnos
51     where edad = (select min(edad) from alumnos)
52 )

```

Below the query editor, the 'Result Grid' is visible, showing a table with the following data:

nombre
Programación I
Bases de Datos

The bottom section of the interface shows the 'Output' window, which contains a table with the following data:

#	Time	Action	Message
1	13:49:57	select nombre from asignaturas where id_carrera in (select id_carrera from alumnos where edad =...	2 row(s) returned

Conclusión:

En este trabajo se aplicaron **subconsultas anidadas, correlacionadas y simples**, logrando obtener resultados dependientes de otras consultas. Se reforzó el uso de IN, NOT IN, AVG(), MIN(), SUM() y la integración de subconsultas dentro de condiciones WHERE.