

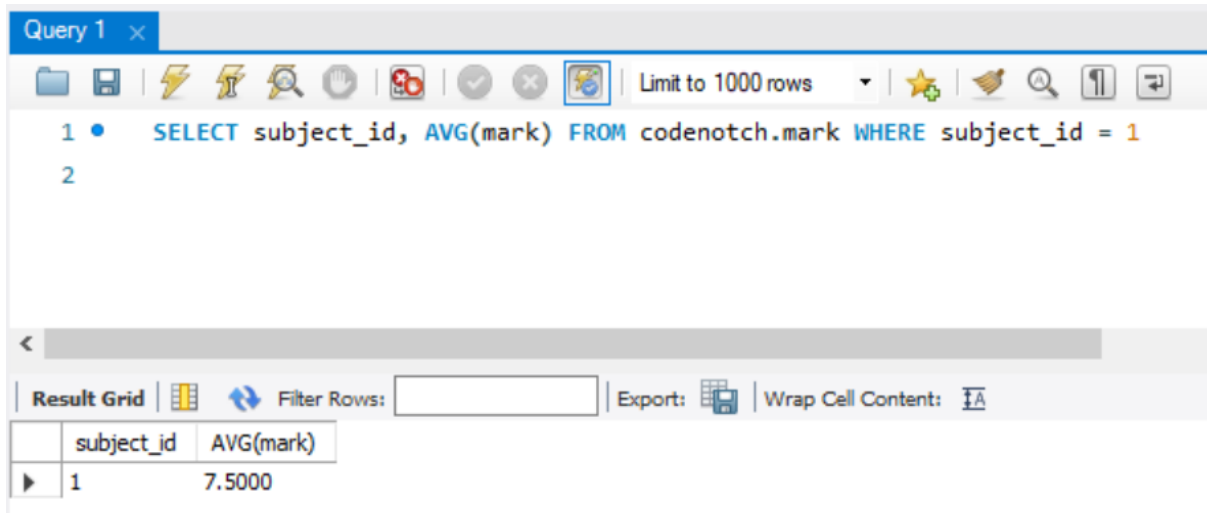
## Módulo 4 - Día 2

### Reto 1

1. Usando el ejemplo de base de datos que tenemos ya implementado desde unidades anteriores, calcular la nota media de los alumnos de la asignatura 1.
2. Calcular el número total de alumnos que hay en el bootcamp.
3. Listar todos los campos de la tabla "groups".
4. Elimina todas las notas de la base de datos que estén por encima de 5 y que sean del año pasado (no utilices BETWEEN).
5. Obtén los datos de todos los estudiantes que estén en el bootcamp este año. Para ello la tabla de estudiantes debe tener un campo que sea el año de ingreso.
6. Calcular el número de profesores que hay por cada asignatura.

1. Usando el ejemplo de base de datos que tenemos ya implementado desde unidades anteriores, calcular la nota media de los alumnos de la asignatura 1.

### Workbench



The screenshot shows the MySQL Workbench interface. At the top, there's a tab labeled 'Query 1'. Below it is a toolbar with various icons. The SQL editor contains the following query:

```
1 • SELECT subject_id, AVG(mark) FROM codenotch.mark WHERE subject_id = 1
2
```

Below the editor is a 'Result Grid' section. It has a 'Filter Rows' input field and an 'Export' button. The result grid shows a single row of data:

	subject_id	AVG(mark)
▶	1	7.5000

### nodejs



The screenshot shows a Node.js application running a SQL query. The code is as follows:

```
13
14 // calcular la nota media de los alumnos de la asignatura 1
15 let sql = "SELECT subject_id, AVG(mark) FROM codenotch.mark WHERE subject_id = 1;"
16
17 codenotchDB.query(sql, (error, result) => {
18   if (!error) {
19     console.log('Operación correcta');
20     console.log(result);
21   } else {
22     console.log(error)
23   }
24 })
```

The terminal output shows the following:

```
PS C:\Users\ajura\Documents\codenotch\Bootcamp\Modulo04\mysql\dia2\reto1> node .\index.js
Conectado a la bdd codenotch
Operación correcta
[ { subject_id: 1, 'AVG(mark)': '7.5000' } ]
```

2. Calcular el número total de alumnos que hay en el bootcamp.

### Workbench

The screenshot shows the MySQL Workbench interface. At the top, there's a tab labeled 'Query 1'. Below it is a toolbar with various icons. The SQL editor contains the following query:

```
1 • SELECT COUNT(*) AS total_alumnos FROM codenotch.student
2
```

Below the editor is a 'Result Grid' section. It has a 'Filter Rows' input field and an 'Export' button. The grid itself shows one column named 'total\_alumnos' with a value of '10'.

total_alumnos
10

### nodejs

The screenshot shows a code editor with a JavaScript file. The code is as follows:

```
25
26 // Calcular el número total de alumnos que hay en el bootcamp
27 let sql = "SELECT COUNT(*) AS total_alumnos FROM codenotch.student;"
28
29 codenotchDB.query(sql, (error, result) => {
30   if (!error) {
31     console.log('Operación correcta');
32     console.log(result);
33   } else {
34     console.log(error)
35   }
36 })
37
38 // Conectar a la base de datos codenotch
```

Below the code editor is a terminal window. It shows the command to run the application and its output:

```
PS C:\Users\ajura\Documents\codenotch\Bootcamp\Modulo04\mysql\dia2\reto1> node .\index.js
Conectado a la bdd codenotch
Operación correcta
[ { total_alumnos: 10 } ]
```

3. Listar todos los campos de la tabla "groups".

## Workbench

The screenshot shows the MySQL Workbench interface. At the top, a tab labeled 'Query 1' is active. Below the toolbar, the SQL editor contains the following query:

```
1 • SELECT * FROM `codenotch`.`group`
2
```

Below the SQL editor, the 'Result Grid' tab is selected, displaying the results of the query in a table format. The table has two columns: 'group\_id' and 'name'. The results are as follows:

group_id	name
1	Grupo A
2	Grupo B
3	Grupo C
4	Grupo D
5	Grupo E
6	Grupo F
7	Grupo G
8	Grupo H
9	Grupo I
10	Grupo J
NULL	NULL

## nodejs

The screenshot shows a Node.js application running a MySQL query. The code is as follows:

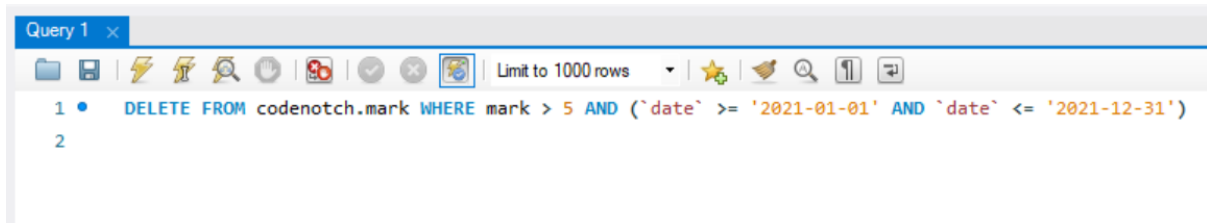
```
37
38 // Calcular el número total de alumnos que hay en el bootcamp
39 let sql = "SELECT * FROM `codenotch`.`group`;";
40
41 codenotchDB.query(sql, (error, result) => {
42   if (!error) {
43     console.log('Operación correcta');
44     console.log(result);
45   } else {
46     console.log(error)
47   }
48 })
```

The application is running in a terminal window. The output is as follows:

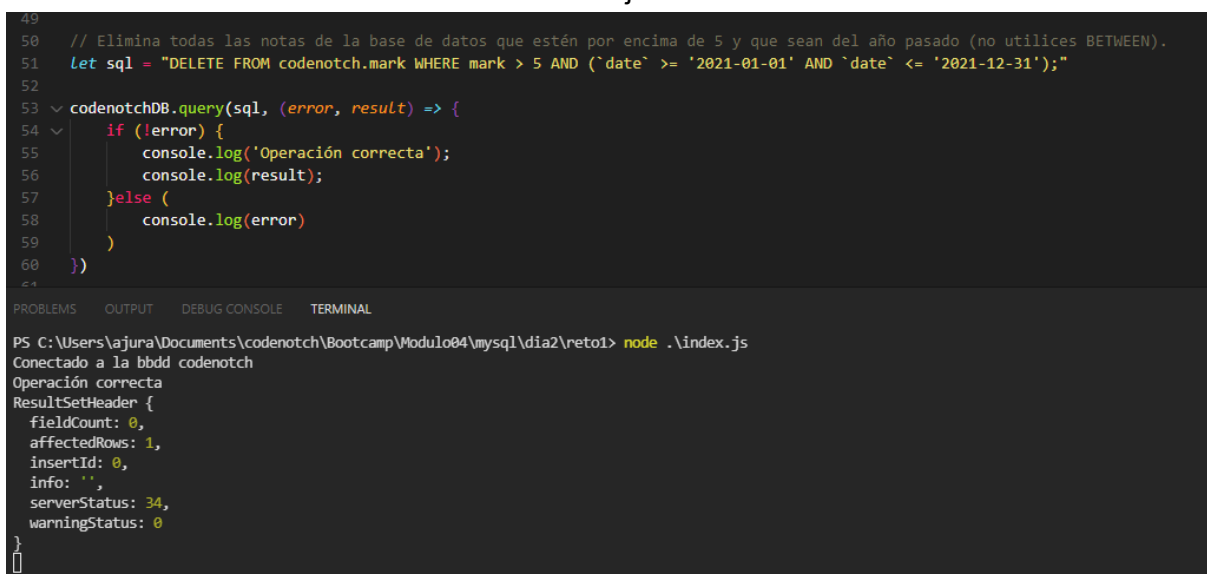
```
PS C:\Users\ajura\Documents\codenotch\Bootcamp\Modulo04\mysql\dia2\reto1> node .\index.js
Conectado a la bbdd codenotch
Operación correcta
[
  { group_id: 1, name: 'Grupo A' },
  { group_id: 2, name: 'Grupo B' },
  { group_id: 3, name: 'Grupo C' },
  { group_id: 4, name: 'Grupo D' },
  { group_id: 5, name: 'Grupo E' },
  { group_id: 6, name: 'Grupo F' },
  { group_id: 7, name: 'Grupo G' },
  { group_id: 8, name: 'Grupo H' },
  { group_id: 9, name: 'Grupo I' },
  { group_id: 10, name: 'Grupo J' }
]
```

4. Elimina todas las notas de la base de datos que estén por encima de 5 y que sean del año pasado (no utilices BETWEEN).

### Workbench



### nodejs



5. Obtén los datos de todos los estudiantes que estén en el bootcamp este año. Para ello la tabla de estudiantes debe tener un campo que sea el año de ingreso.

### Workbench

Query 1

```
1 • SELECT * FROM codenotch.student WHERE entry_year = 2022
```

2

Result Grid

	student_id	group_id	first_name	last_name	entry_year
▶	1	3	Ander	Jurado	2022
	4	6	Pedro	Abenza	2022
	7	6	Laura	Henríquez	2022
*	NULL	NULL	NULL	NULL	NULL

### nodejs

```
61
62 // Obtén los datos de todos los estudiantes que estén en el bootcamp este año. Para ello la tabla de estud
63 let sql = "SELECT * FROM codenotch.student WHERE entry_year = 2022;"
64
65 codenotchDB.query(sql, (error, result) => {
66   if (!error) {
67     console.log('Operación correcta');
68     console.log(result);
69   } else {
70     console.log(error)
71   }
72 })
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

Operación correcta

```
[
  {
    student_id: 1,
    group_id: 3,
    first_name: 'Ander',
    last_name: 'Jurado',
    entry_year: 2022
  },
  {
    student_id: 4,
    group_id: 6,
    first_name: 'Pedro',
    last_name: 'Abenza',
    entry_year: 2022
  },
  {
    student_id: 7,
    group_id: 6,
    first_name: 'Laura',
    last_name: 'Henríquez',
    entry_year: 2022
  }
]
```

6. Calcular el número de profesores que hay por cada asignatura.

### Workbench

Query 1

```
1 • SELECT subject_id, COUNT(teacher_id) AS num_teachers FROM subject_teacher GROUP BY subject_id
```

2

Result Grid

	subject_id	num_teachers
1	1	2
2	2	2
3	3	1
4	4	2
5	5	1
8	8	1
9	9	1

### nodejs

```
73
74 // Calcular el número de profesores que hay por cada asignatura.
75 let sql = "SELECT subject_id, COUNT(teacher_id) AS num_teachers FROM subject_teacher GROUP BY subject_id;"
76
77 codenotchDB.query(sql, (error, result) => {
78   if (!error) {
79     console.log('Operación correcta');
80     console.log(result);
81   } else {
82     console.log(error)
83   }
84 })
85
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

PS C:\Users\ajura\Documents\codenotch\Bootcamp\Modulo04\mysql\dia2\reto1> node .\index.js

Conectado a la bbdd codenotch

Operación correcta

```
[
  { subject_id: 1, num_teachers: 2 },
  { subject_id: 2, num_teachers: 2 },
  { subject_id: 3, num_teachers: 1 },
  { subject_id: 4, num_teachers: 2 },
  { subject_id: 5, num_teachers: 1 },
  { subject_id: 8, num_teachers: 1 },
  { subject_id: 9, num_teachers: 1 }
]
```