

**Tecnicatura Universitaria en Programación - Universidad Tecnológica
Nacional**

TRABAJO PRÁCTICO INTEGRADOR

Base de Datos I

Alumno:

Gomez Saucedo Augusto Nahuel - gomeznahuel@hotmail.com

Docente Titular:

Docente Tutor:

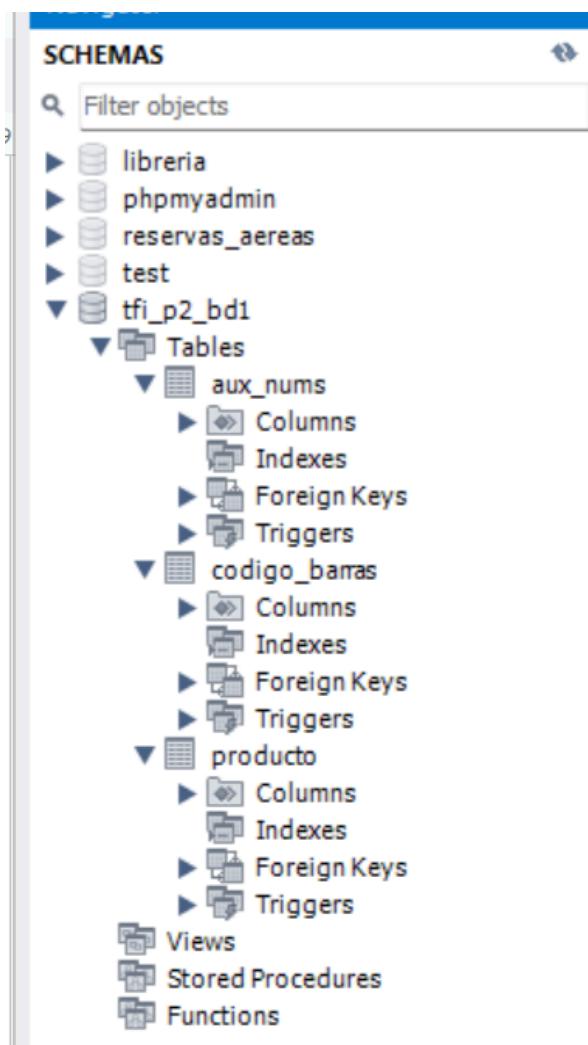
Tema: Producto (A) – Código de barras (B) relación 1 a 1

1. Introducción

Este trabajo integra Programación 2 (CRUD Java JDBC) y Bases de Datos I. Se implementa un modelo con dos tablas relacionadas 1 a 1: **producto** y **codigo_barras**, con integridad referencial, validaciones de dominio y pruebas de rendimiento mediante carga masiva, consultas, índices, seguridad y transacciones.

2. Etapa 1 – Modelo y estructura

2.1 DER y modelo relacional



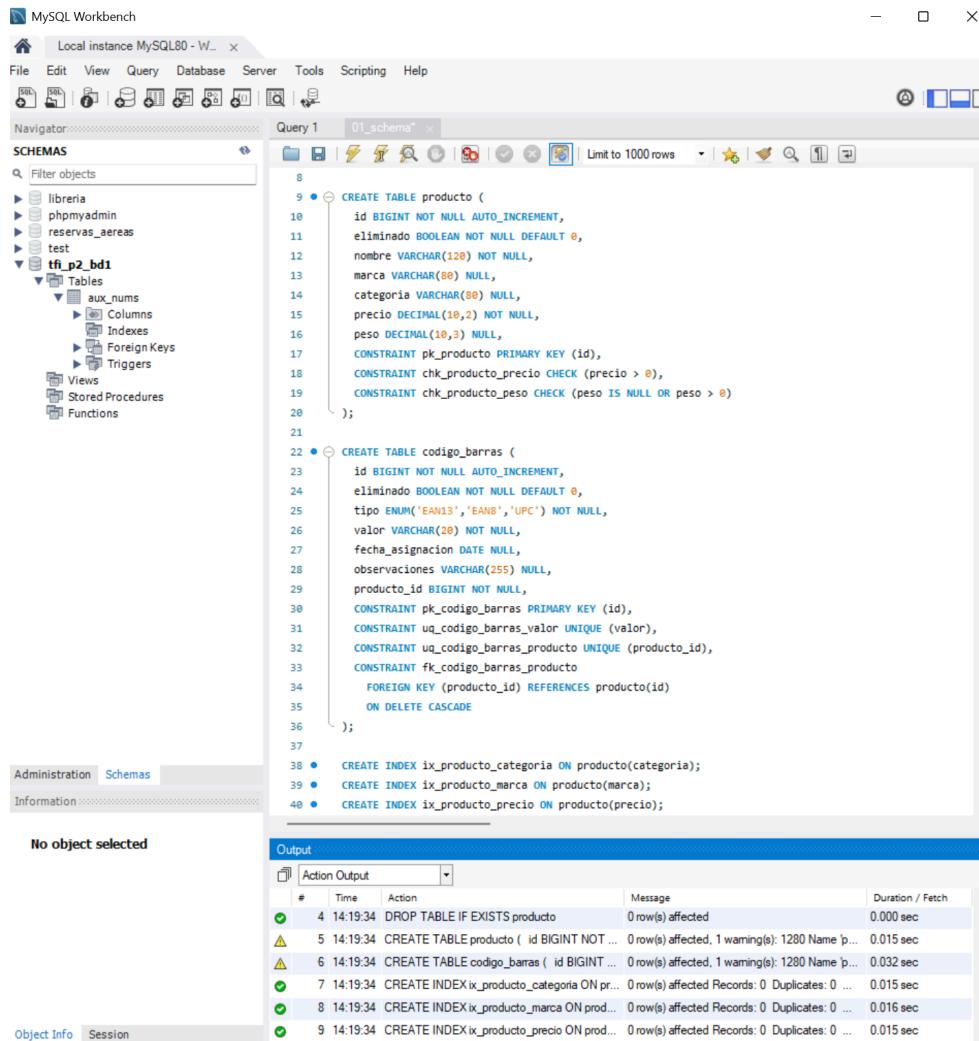
DER:

- **Modelo relacional:**

- producto(id PK, eliminado, nombre, marca, categoria, precio, peso)
- codigo_barras(id PK, eliminado, tipo, valor UNIQUE, producto_id UNIQUE FK → producto.id)

2.2 Script de creación

- Archivo: `sql/01_schema.sql`



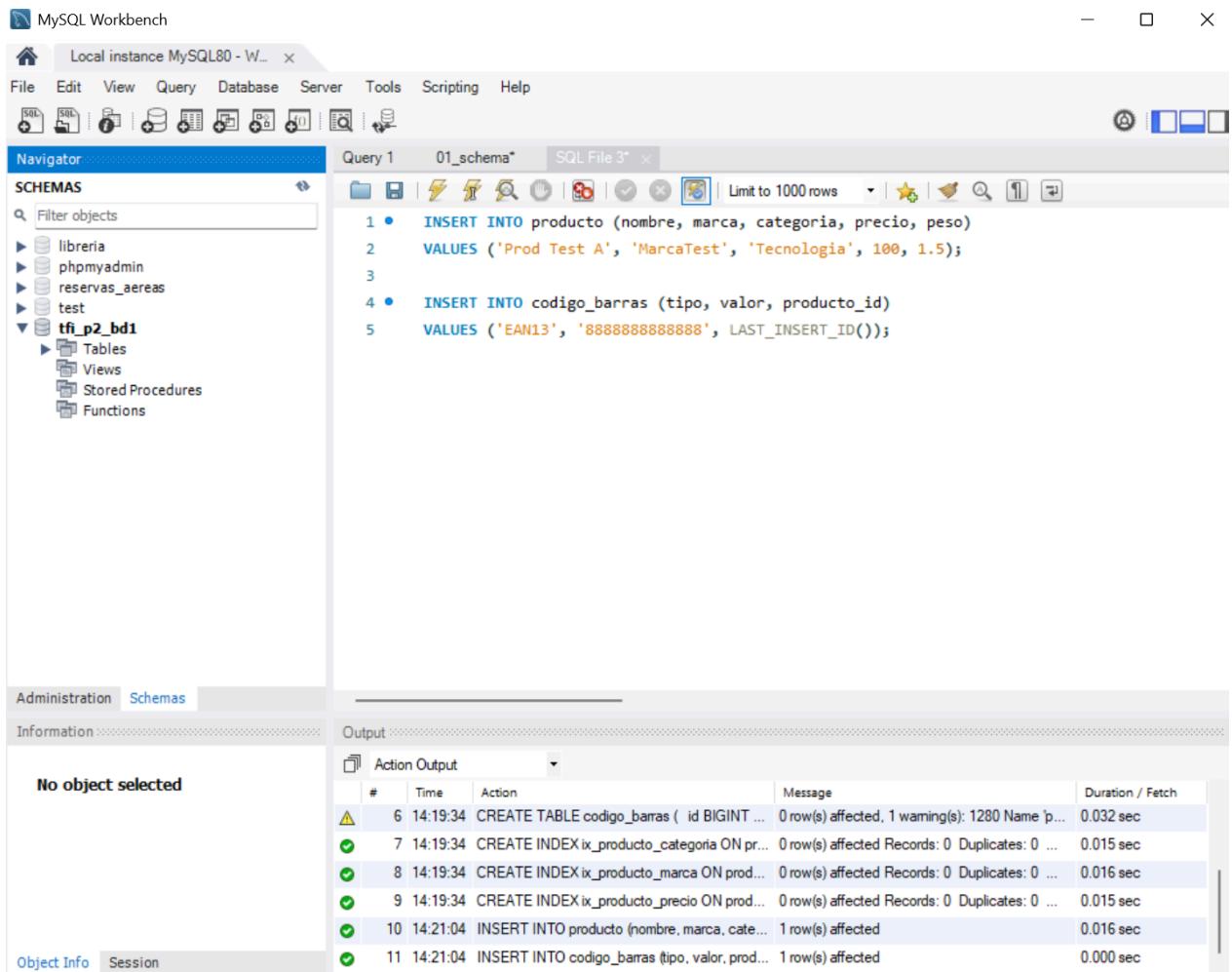
The screenshot shows the MySQL Workbench interface with the following details:

- Navigator:** Shows the database structure under the schema `tf_p2_bd1`. It includes tables like `aux_nums`, and constraints like `pk_producto`, `chk_producto_precio`, and `chk_producto_peso`.
- Query Editor:** The query `01_schema.sql` contains the SQL code for creating the `producto` and `codigo_barras` tables, and creating indexes `ix_producto_categoria`, `ix_producto_marca`, and `ix_producto_precio`.
- Output:** The results of the execution show the following log entries:

#	Time	Action	Message	Duration / Fetch
4	14:19:34	DROP TABLE IF EXISTS producto	0 row(s) affected	0.000 sec
5	14:19:34	CREATE TABLE producto (id BIGINT NOT ...	0 row(s) affected, 1 warning(s): 1280 Name 'p...'	0.015 sec
6	14:19:34	CREATE TABLE codigo_barras (id BIGINT ...	0 row(s) affected, 1 warning(s): 1280 Name 'p...'	0.032 sec
7	14:19:34	CREATE INDEX ix_producto_categoria ON pr...	0 row(s) affected Records: 0 Duplicates: 0 ...	0.015 sec
8	14:19:34	CREATE INDEX ix_producto_marca ON prod...	0 row(s) affected Records: 0 Duplicates: 0 ...	0.016 sec
9	14:19:34	CREATE INDEX ix_producto_precio ON prod...	0 row(s) affected Records: 0 Duplicates: 0 ...	0.015 sec

2.3 Pruebas de constraints

2 inserts correctos



The screenshot shows the MySQL Workbench interface with the following details:

- Navigator:** Shows the schema tree with the current database set to `01_schema`.
- Query Editor:** Displays two successful SQL statements:

```
1 • INSERT INTO producto (nombre, marca, categoria, precio, peso)
2 VALUES ('Prod Test A', 'MarcaTest', 'Tecnologia', 100, 1.5);
3
4 • INSERT INTO codigo_barra (tipo, valor, producto_id)
5 VALUES ('EAN13', '888888888888', LAST_INSERT_ID());
```
- Output Window:** Shows the execution log with the following entries:

#	Time	Action	Message	Duration / Fetch
6	14:19:34	CREATE TABLE codigo_barra (id BIGINT ...	0 row(s) affected, 1 warning(s): 1280 Name 'p...' 0 row(s) affected Records: 0 Duplicates: 0 ...	0.032 sec
7	14:19:34	CREATE INDEX ix_producto_categoria ON pr...	0 row(s) affected Records: 0 Duplicates: 0 ...	0.015 sec
8	14:19:34	CREATE INDEX ix_producto_marca ON prod...	0 row(s) affected Records: 0 Duplicates: 0 ...	0.016 sec
9	14:19:34	CREATE INDEX ix_producto_precio ON prod...	0 row(s) affected Records: 0 Duplicates: 0 ...	0.015 sec
10	14:21:04	INSERT INTO producto (nombre, marca, cate...	1 row(s) affected	0.016 sec
11	14:21:04	INSERT INTO codigo_barra (tipo, valor, prod...	1 row(s) affected	0.000 sec

2 inserts con error

MySQL Workbench

Local instance MySQL80 - W...

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- libreria
- phpmyadmin
- reservas_aereas
- test
- tf_p2_bd1**
 - Tables
 - Views
 - Stored Procedures
 - Functions

Query 1 01_schema* SQL File 3* x

1 • INSERT INTO codigo_barra (tipo, valor, producto_id)
2 VALUES ('EAN13', '888888888888', 1);

Limit to 1000 rows

Administration Schemas

Information

No object selected

Action Output

#	Time	Action	Message	Duration / Fetch
7	14:19:34	CREATE INDEX ix_producto_categoria ON pr...	0 row(s) affected Records: 0 Duplicates: 0 ...	0.015 sec
8	14:19:34	CREATE INDEX ix_producto_marca ON prod...	0 row(s) affected Records: 0 Duplicates: 0 ...	0.016 sec
9	14:19:34	CREATE INDEX ix_producto_precio ON prod...	0 row(s) affected Records: 0 Duplicates: 0 ...	0.015 sec
10	14:21:04	INSERT INTO producto (nombre, marca, cate...	1 row(s) affected	0.016 sec
11	14:21:04	INSERT INTO codigo_barra (tipo, valor, prod...	1 row(s) affected	0.000 sec
12	14:21:57	INSERT INTO codigo_barra (tipo, valor, prod...	Error Code: 1062. Duplicate entry '888888888...'	0.016 sec

Object Info Session

3. Etapa 2 – Carga masiva y verificación

3.1 Descripción conceptual

Se realizó una carga masiva utilizando `INSERT ... SELECT` y una tabla auxiliar numérica para generar registros sintéticos consistentes. Se verificó integridad referencial (sin huérfanos), cardinalidad 1 a 1 (sin productos sin código), unicidad de código de barras y cumplimiento del dominio (precio > 0 y peso NULL o > 0). Se adjuntan capturas de conteos y verificaciones.

3.2 Cantidad de registros cargados

The screenshot shows the MySQL Workbench interface with a query editor and an output pane.

Query Editor:

```
1 • USE tfi_p2_bd1;
2
3 -- 1) Tabla auxiliar de numeros (0..9999)
4 • DROP TABLE IF EXISTS aux_nums;
5 • CREATE TABLE aux_nums (n INT NOT NULL PRIMARY KEY);
6
7 • INSERT INTO aux_nums (n)
8   SELECT a.d + b.d*10 + c.d*100 + d.d*1000
9   FROM (SELECT 0 d UNION ALL SELECT 1 UNION ALL SELECT 2 UNION ALL SELECT 3 UNION ALL SEL
10      UNION ALL SELECT 5 UNION ALL SELECT 6 UNION ALL SELECT 7 UNION ALL SELECT 8 UNION
11      CROSS JOIN (SELECT 0 d UNION ALL SELECT 1 UNION ALL SELECT 2 UNION ALL SELECT 3 UNION A
12         UNION ALL SELECT 5 UNION ALL SELECT 6 UNION ALL SELECT 7 UNION ALL SELECT 8 UNION
13      CROSS JOIN (SELECT 0 d UNION ALL SELECT 1 UNION ALL SELECT 2 UNION ALL SELECT 3 UNION A
14         UNION ALL SELECT 5 UNION ALL SELECT 6 UNION ALL SELECT 7 UNION ALL SELECT 8 UNION
15      CROSS JOIN (SELECT 0 d UNION ALL SELECT 1 UNION ALL SELECT 2 UNION ALL SELECT 3 UNION A
16         UNION ALL SELECT 5 UNION ALL SELECT 6 UNION ALL SELECT 7 UNION ALL SELECT 8 UNION
17
18 -- Si ya tenes datos y queres empezar "limpio" (opcional):
```

Output:

Action Output	#	Time	Action	Message	Duration / Fetch
1 14:25:17 USE tfi_p2_bd1	1	14:25:17	USE tfi_p2_bd1	0 row(s) affected	0.000 sec
2 14:25:17 DROP TABLE IF EXISTS aux_nums	2	14:25:17	DROP TABLE IF EXISTS aux_nums	0 row(s) affected	0.016 sec
3 14:25:17 CREATE TABLE aux_nums (n INT NOT NULL ...	3	14:25:17	CREATE TABLE aux_nums (n INT NOT NULL ...	0 row(s) affected	0.016 sec
4 14:25:17 INSERT INTO aux_nums (n) SELECT a.d + b.d...	4	14:25:17	INSERT INTO aux_nums (n) SELECT a.d + b.d...	10000 row(s) affected Records: 10000 Duplica...	0.062 sec
5 14:25:17 INSERT INTO producto (eliminado, nombre, ma...	5	14:25:17	INSERT INTO producto (eliminado, nombre, ma...	10000 row(s) affected Records: 10000 Duplica...	0.344 sec
6 14:25:18 INSERT INTO codigo_barras (eliminado, tipo, v...	6	14:25:18	INSERT INTO codigo_barras (eliminado, tipo, v...	10000 row(s) affected Records: 10000 Duplica...	0.344 sec

MySQL Workbench

Local instance MySQL80 - W_

File Edit View Query Database Server Tools Scripting Help

Navigator: 01_schema* SQL File 3* 03_mass_insert

SCHEMAS

libreria
phpmyadmin
reservas_aereas
test
tfi_p2_bd1
Tables
Views
Stored Procedures
Functions

Query 1 01_schema* SQL File 3* 03_mass_insert

1 • USE tfi_p2_bd1;
2
3 • SELECT COUNT(*) AS total_productos FROM producto;
4 • SELECT COUNT(*) AS total_codigos FROM codigo_barra;

Result Grid | Filter Rows: Export: Wrap Cell Content: Result Grid

total_codigos
10001

Administration Schemas

No object selected

Action Output

#	Time	Action	Message	Duration / Fetch
4	14:25:17	INSERT INTO aux_nums (n) SELECT a.d + ...	10000 row(s) affected Records: 10000 Duplic... 0.062 sec	
5	14:25:17	INSERT INTO producto (eliminado, nombre, ...)	10000 row(s) affected Records: 10000 Duplic... 0.344 sec	
6	14:25:18	INSERT INTO codigo_barra (eliminado, tipo, ...)	10000 row(s) affected Records: 10000 Duplic... 0.344 sec	
7	14:26:14	USE tfi_p2_bd1	0 row(s) affected 0.000 sec	
8	14:26:14	SELECT COUNT(*) AS total_productos FRO...	1 row(s) returned 0.000 sec / 0.000 sec	
9	14:26:14	SELECT COUNT(*) AS total_codigos FROM ...	1 row(s) returned 0.000 sec / 0.000 sec	

Object Info Session

MySQL Workbench

File Edit View Query Database Server Tools Scripting Help

Navigator: 01_schema* SQL File 3* 03_mass_insert

SCHEMAS

- libreria
- phpmyadmin
- reservas_aereas
- test
- tfi_p2_bd1**

Tables

Views

Stored Procedures

Functions

Query 1: 01_schema*

```
1 • SELECT COUNT(*) AS productos_sin_codigo
2 FROM producto p
3 LEFT JOIN codigo_barra c ON c.producto_id = p.id
4 WHERE c.producto_id IS NULL;
```

Result Grid | Filter Rows: Export: Wrap Cell Content: Result Grid

productos_sin_codigo
0

Result 3: x

No object selected

Action Output

#	Time	Action	Message	Duration / Fetch
5	14:25:17	INSERT INTO producto (eliminado, nombre, ...)	10000 row(s) affected Records: 10000 Duplic... 0.344 sec	
6	14:25:18	INSERT INTO codigo_barra (eliminado, tipo...)	10000 row(s) affected Records: 10000 Duplic... 0.344 sec	
7	14:26:14	USE tfi_p2_bd1	0 row(s) affected 0.000 sec	
8	14:26:14	SELECT COUNT(*) AS total_productos FRO...	1 row(s) returned 0.000 sec / 0.000 sec	
9	14:26:14	SELECT COUNT(*) AS total_codigos FROM ...	1 row(s) returned 0.000 sec / 0.000 sec	
10	14:26:49	SELECT COUNT(*) AS productos_sin_codig...	1 row(s) returned 0.015 sec / 0.000 sec	

MySQL Workbench

Local instance MySQL80 - W...

File Edit View Query Database Server Tools Scripting Help

Navigator: 01_schema* SQL File 3* 03_mass_insert

SCHEMAS

libreria
phpmyadmin
reservas_aereas
test
tfi_p2_bd1
Tables
Views
Stored Procedures
Functions

1 • SELECT COUNT(*) AS codigos_huerfanos
2 FROM codigo_barra c
3 LEFT JOIN producto p ON p.id = c.producto_id
4 WHERE p.id IS NULL;

Result Grid | Filter Rows: Export: Wrap Cell Content: Result Grid

	codigos_huerfanos
0	0

Administration Schemas

No object selected

Action Output

#	Time	Action	Message	Duration / Fetch
6	14:25:18	INSERT INTO codigo_barra (eliminado, tipo...)	10000 row(s) affected Records: 10000 Duplic...	0.344 sec
7	14:26:14	USE tfi_p2_bd1	0 row(s) affected	0.000 sec
8	14:26:14	SELECT COUNT(*) AS total_productos FRO...	1 row(s) returned	0.000 sec / 0.000 sec
9	14:26:14	SELECT COUNT(*) AS total_codigos FROM ...	1 row(s) returned	0.000 sec / 0.000 sec
10	14:26:49	SELECT COUNT(*) AS productos_sin_codig...	1 row(s) returned	0.015 sec / 0.000 sec
11	14:27:24	SELECT COUNT(*) AS codigos_huerfanos F...	1 row(s) returned	0.032 sec / 0.000 sec

Object Info Session

MySQL Workbench

Local instance MySQL80 - W...

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- libreria
- phpmyadmin
- reservas_aereas
- test
- tfi_p2_bd1**
- Tables
- Views
- Stored Procedures
- Functions

Query 1 01_schema* SQL File 3* 03_mass_insert

```
1 • SELECT COUNT(*) AS codigos_duplicados
2 ⚡ FROM (
3     SELECT valor
4         FROM codigo_barra
5             GROUP BY valor
6             HAVING COUNT(*) > 1
7 ) t;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Result Grid

codigos_duplicados
0

Administration Schemas

Information

No object selected

Action Output

#	Time	Action	Message	Duration / Fetch
7	14:26:14	USE tfi_p2_bd1	0 row(s) affected	0.000 sec
8	14:26:14	SELECT COUNT() AS total_productos FRO...	1 row(s) returned	0.000 sec / 0.000 sec
9	14:26:14	SELECT COUNT() AS total_codigos FROM ...	1 row(s) returned	0.000 sec / 0.000 sec
10	14:26:49	SELECT COUNT() AS productos_sin_codig...	1 row(s) returned	0.015 sec / 0.000 sec
11	14:27:24	SELECT COUNT() AS codigos_huerfanos F...	1 row(s) returned	0.032 sec / 0.000 sec
12	14:27:51	SELECT COUNT() AS codigos_duplicados ...	1 row(s) returned	0.016 sec / 0.000 sec

Object Info Session

MySQL Workbench

Local instance MySQL80 - W...

File Edit View Query Database Server Tools Scripting Help

Navigator: 01_schema* SQL File 3* 03_mass_insert

SCHEMAS: libreria, phpmyadmin, reservas_aereas, test, tfi_p2_bd1 (Tables, Views, Stored Procedures, Functions)

```

1 •  SELECT COUNT(*) AS productos_fuera_deDominio
2   FROM producto
3   WHERE precio <= 0
4   OR (peso IS NOT NULL AND peso <= 0);

```

Result Grid | Filter Rows: Export: Wrap Cell Content: Result Grid

productos_fuera_deDominio
0

Administration Schemas Information

No object selected

Action Output

#	Time	Action	Message	Duration / Fetch
8	14:26:14	SELECT COUNT(*) AS total_productos FRO...	1 row(s) returned	0.000 sec / 0.000 sec
9	14:26:14	SELECT COUNT(*) AS total_codigos FROM ...	1 row(s) returned	0.000 sec / 0.000 sec
10	14:26:49	SELECT COUNT(*) AS productos_sin_codig...	1 row(s) returned	0.015 sec / 0.000 sec
11	14:27:24	SELECT COUNT(*) AS codigos_huerfanos F...	1 row(s) returned	0.032 sec / 0.000 sec
12	14:27:51	SELECT COUNT(*) AS codigos_duplicados ...	1 row(s) returned	0.016 sec / 0.000 sec
13	14:28:18	SELECT COUNT(*) AS productos_fuera_de...	1 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

- productos: **10001**
- códigos: **10001**
- validación final: **productos_sin_codigo = 0**

3.3 Verificaciones

- huérfanos = 0
- duplicados = 0
- fuera de dominio = 0
- productos sin código = 0

4. Etapa 3 – Consultas, vistas, índices y performance

4.1 Consultas requeridas

Se implementaron 4 consultas:

- Q1 JOIN (producto + código)

The screenshot shows a SQL development interface with the following components:

- Query Editor:** Displays the SQL code for Q1 JOIN (producto + código). The code includes joins between 'producto' and 'codigo_barras' tables, filtering by elimination status, ordering by product ID, and limiting the results to 50 rows. It also includes an EXPLAIN command.
- Result Grid:** Shows the execution results with columns: id, nombre, marca, categoria, precio, tipo, valor, and fecha_asignacion. The data lists various products like 'Zen', 'Fenix', 'Nova', etc., with their respective details.
- Action Output:** Displays the history of actions taken during the session, including the execution of the EXPLAIN command and the SELECT statement itself, along with their execution times and message details.

01_schema 02_seed 03_mass_insert 04_checks 05_consultas 06_indices SQL File 13*

```

19 LIMIT 50;
20
21 -- EXPLAIN Q1 (para captura)
22 • EXPLAIN
23 SELECT
24     p.id, p.nombre, p.marca, p.categoría, p.precio,
25     c.tipo, c.valor, c.fecha_asignacion
26 FROM producto p
27 JOIN codigo_barras c ON c.producto_id = p.id
28 WHERE p.eliminado = 0 AND c.eliminado = 0
29 ORDER BY p.id DESC;
30
31

```

Result Grid | Filter Rows: Export: Wrap Cell Content:

	id	select_type	table	type	possible_keys	key	key_len	ref	rows	Extra
▶	1	SIMPLE	c	ALL	usi_codigo_barras_producto	HASH	HASH	tfl_p2_bdi.c.producto_id	9825	Using where
	1	SIMPLE	p	eq_ref	PRIMARY	PRIMARY	8	tfl_p2_bdi.c.producto_id	1	Using where

SQLAdditions: Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

Result 7 x

Action Output

#	Time	Action	Message	Duration / Fetch
23	14:52:13	EXPLAIN SELECT p.id, p.nombre, p.marca, p.categoría, p.precio, c.tipo, c.valor, c.fecha_asignacion FROM pro...	2 row(s) returned	0.016 sec / 0.000 sec
24	14:52:50	SELECT p.id, p.nombre, p.marca, p.categoría, p.precio, c.tipo, c.valor, c.fecha_asignacion FROM pro...	50 row(s) returned	0.000 sec / 0.000 sec
25	14:53:31	SELECT p.id, p.nombre, p.marca, p.categoría, p.precio, c.tipo, c.valor, c.fecha_asignacion FROM pro...	50 row(s) returned	0.000 sec / 0.000 sec
26	14:53:39	SELECT p.id, p.nombre, p.marca, p.categoría, p.precio, c.tipo, c.valor, c.fecha_asignacion FROM pro...	50 row(s) returned	0.000 sec / 0.000 sec
27	14:58:04	SELECT p.id, p.nombre, p.marca, p.categoría, p.precio, c.tipo, c.valor, c.fecha_asignacion FROM pro...	1000 row(s) returned	0.016 sec / 0.000 sec
28	14:59:25	EXPLAIN SELECT p.id, p.nombre, p.marca, p.categoría, p.precio, c.tipo, c.valor, c.fecha_asignacion FROM pro...	2 row(s) returned	0.000 sec / 0.000 sec

Read Only Context Help Snippets

● Q2 JOIN (filtros por categoría + rango de fechas)

01_schema 02_seed 03_mass_insert 04_checks 05_consultas 06_indices SQL File 13*

```

37 COUNT(*) AS cantidad,
38 ROUND(AVG(p.precio), 2) AS precio_promedio
39
FROM producto p
40 JOIN codigo_barras c ON c.producto_id = p.id
41 WHERE p.eliminado = 0
42 AND c.eliminado = 0
43 AND p.categoría IN ('Tecnología', 'Alimentos')
44 AND c.fecha_asignacion BETWEEN '2025-03-01' AND '2025-06-30'
45 GROUP BY p.categoría
46 ORDER BY cantidad DESC;
47
48 -- EXPLAIN Q2
49 • EXPLAIN

```

Result Grid | Filter Rows: Export: Wrap Cell Content:

categoria	cantidad	precio_promedio
Tecnología	665	290.86
Alimentos	665	289.71

SQLAdditions: Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

Result 8 x

Action Output

#	Time	Action	Message	Duration / Fetch
24	14:52:50	SELECT p.id, p.nombre, p.marca, p.categoría, p.precio, c.tipo, c.valor, c.fecha_asignacion FROM pro...	50 row(s) returned	0.000 sec / 0.000 sec
25	14:53:31	SELECT p.id, p.nombre, p.marca, p.categoría, p.precio, c.tipo, c.valor, c.fecha_asignacion FROM pro...	50 row(s) returned	0.000 sec / 0.000 sec
26	14:53:39	SELECT p.id, p.nombre, p.marca, p.categoría, p.precio, c.tipo, c.valor, c.fecha_asignacion FROM pro...	50 row(s) returned	0.000 sec / 0.000 sec
27	14:58:04	SELECT p.id, p.nombre, p.marca, p.categoría, p.precio, c.tipo, c.valor, c.fecha_asignacion FROM pro...	1000 row(s) returned	0.016 sec / 0.000 sec
28	14:59:25	EXPLAIN SELECT p.id, p.nombre, p.marca, p.categoría, p.precio, c.tipo, c.valor, c.fecha_asignacion FROM product...	2 row(s) returned	0.000 sec / 0.000 sec
29	15:00:38	SELECT p.categoría, COUNT(*) AS cantidad, ROUND(AVG(p.precio), 2) AS precio_promedio FROM product...	2 row(s) returned	0.015 sec / 0.000 sec

Read Only Context Help Snippets

SQL Additions SQL File 13*

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

```

52 COUNT(*) AS cantidad,
53     ROUND(AVG(p.precio), 2) AS precio_promedio
54 FROM producto p
55 JOIN codigo_barras c ON c.producto_id = p.id
56 WHERE p.eliminado = 0
57 AND c.eliminado = 0
58 AND p.categoría IN ('Tecnología', 'Alimentos')
59 AND c.fecha_asignacion BETWEEN '2025-03-01' AND '2025-06-30'
60 GROUP BY p.categoría;
61
62
63 -- -----
64 -- Q3 (GROUP BY + HAVING): marcas con muchos productos y precio promedio alto

```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

	id	select_type	table	type	possible_keys	key	key_len	ref	rows	Extra
▶	1	SIMPLE	p	ALL	PRIMARY,ix_producto_categoria	HULL	HULL	HULL	9838	Using where; Using temporary; Using filesort
	1	SIMPLE	c	eq_ref	uq_codigo_barras_producto	uq_codigo_barras_producto	8	tfi_p2_bd1.p.id	1	Using where

Result Grid | Form Editor | Field Types | Query Stats

Action Output

#	Time	Action	Message	Duration / Fetch
25	14:53:31	SELECT p.id, p.nombre, p.marca, p.categoría, p.precio, c.tipo, c.valor, c.fecha_asignacion FROM producto p JOIN codigo_barras c ON c.producto_id = p.id WHERE p.eliminado = 0 AND c.eliminado = 0 AND p.categoría IN ('Tecnología', 'Alimentos') AND c.fecha_asignacion BETWEEN '2025-03-01' AND '2025-06-30' GROUP BY p.categoría;	50 row(s) returned	0.000 sec / 0.000 sec
26	14:53:39	SELECT p.id, p.nombre, p.marca, p.categoría, p.precio, c.tipo, c.valor, c.fecha_asignacion FROM producto p JOIN codigo_barras c ON c.producto_id = p.id WHERE p.eliminado = 0 AND c.eliminado = 0 AND p.categoría IN ('Tecnología', 'Alimentos') AND c.fecha_asignacion BETWEEN '2025-03-01' AND '2025-06-30' GROUP BY p.categoría;	50 row(s) returned	0.000 sec / 0.000 sec
27	14:58:04	SELECT p.id, p.nombre, p.marca, p.categoría, p.precio, c.tipo, c.valor, c.fecha_asignacion FROM producto p JOIN codigo_barras c ON c.producto_id = p.id WHERE p.eliminado = 0 AND c.eliminado = 0 AND p.categoría IN ('Tecnología', 'Alimentos') AND c.fecha_asignacion BETWEEN '2025-03-01' AND '2025-06-30' GROUP BY p.categoría;	1000 row(s) returned	0.016 sec / 0.000 sec
28	14:59:25	EXPLAIN SELECT p.id, p.nombre, p.marca, p.categoría, p.precio, c.tipo, c.valor, c.fecha_asignacion FROM producto p JOIN codigo_barras c ON c.producto_id = p.id WHERE p.eliminado = 0 AND c.eliminado = 0 AND p.categoría IN ('Tecnología', 'Alimentos') AND c.fecha_asignacion BETWEEN '2025-03-01' AND '2025-06-30' GROUP BY p.categoría;	2 row(s) returned	0.000 sec / 0.000 sec
29	15:00:38	SELECT p.categoría, COUNT() AS cantidad, ROUND(AVG(p.precio), 2) AS precio_promedio FROM producto p JOIN codigo_barras c ON c.producto_id = p.id WHERE p.eliminado = 0 AND c.eliminado = 0 AND p.categoría IN ('Tecnología', 'Alimentos') AND c.fecha_asignacion BETWEEN '2025-03-01' AND '2025-06-30' GROUP BY p.categoría;	2 row(s) returned	0.015 sec / 0.000 sec
30	15:01:42	EXPLAIN SELECT p.categoría, COUNT() AS cantidad, ROUND(AVG(p.precio), 2) AS precio_promedio FROM producto p JOIN codigo_barras c ON c.producto_id = p.id WHERE p.eliminado = 0 AND c.eliminado = 0 AND p.categoría IN ('Tecnología', 'Alimentos') AND c.fecha_asignacion BETWEEN '2025-03-01' AND '2025-06-30' GROUP BY p.categoría;	2 row(s) returned	0.000 sec / 0.000 sec

Result Grid | Form Editor | Field Types | Query Stats

Output: Action Output | Read Only | Context Help | Snippets

● Q3 GROUP BY + HAVING

SQL Editor:

```

01_schema 02_seed 03_mass_insert 04_checks 05_consultas* 06_indices SQL File 13*
67     p.marca,
68     COUNT(*) AS cantidad_productos,
69     ROUND(AVG(p.precio), 2) AS precio_promedio
70   FROM producto p
71   WHERE p.eliminado = 0
72   GROUP BY p.marca
73   HAVING COUNT(*) >= 50
74 ORDER BY cantidad_productos DESC;
75
76 -- EXPLAIN Q3
77 * EXPLAIN
78 SELECT
79     p.marca,

```

Result Grid:

marca	cantidad_productos	precio_promedio
Zen	1667	292.44
Nova	1667	292.56
Fenix	1667	291.29
Acme	1667	291.41
Delta	1666	292.50
Orion	1666	291.35

Output:

#	Time	Action	Message	Duration / Fetch
27	14:58:04	SELECT p.id, p.nombre, p.marca, p.categoría, p.precio, c.tipo, c.valor, c.fecha_asignacion FROM producto p JOIN categoria c ON p.categoría = c.id WHERE p.eliminado = 0 GROUP BY p.marca, p.categoría, p.precio, c.tipo, c.valor, c.fecha_asignacion ORDER BY cantidad_productos DESC;	1000 row(s) returned	0.016 sec / 0.000 sec
28	14:59:25	EXPLAIN SELECT p.id, p.nombre, p.marca, p.categoría, p.precio, c.tipo, c.valor, c.fecha_asignacion FROM producto p JOIN categoria c ON p.categoría = c.id WHERE p.eliminado = 0 GROUP BY p.marca, p.categoría, p.precio, c.tipo, c.valor, c.fecha_asignacion ORDER BY cantidad_productos DESC;	2 row(s) returned	0.000 sec / 0.000 sec
29	15:00:38	SELECT p.categoría, COUNT(*) AS cantidad, ROUND(AVG(p.precio), 2) AS precio_promedio FROM producto p GROUP BY p.categoría HAVING COUNT(*) >= 50;	2 row(s) returned	0.015 sec / 0.000 sec
30	15:01:42	EXPLAIN SELECT p.categoría, COUNT(*) AS cantidad, ROUND(AVG(p.precio), 2) AS precio_promedio FROM producto p GROUP BY p.categoría HAVING COUNT(*) >= 50;	2 row(s) returned	0.000 sec / 0.000 sec
31	15:02:06	SELECT p.marca, COUNT(*) AS cantidad_productos, ROUND(AVG(p.precio), 2) AS precio_promedio FROM producto p WHERE p.eliminado = 0 GROUP BY p.marca HAVING COUNT(*) >= 50;	0 row(s) returned	0.031 sec / 0.000 sec
32	15:03:15	SELECT p.marca, COUNT(*) AS cantidad_productos, ROUND(AVG(p.precio), 2) AS precio_promedio FROM producto p WHERE p.eliminado = 0 GROUP BY p.marca ORDER BY cantidad_productos DESC;	6 row(s) returned	0.016 sec / 0.000 sec

01_schema 02_seed 03_mass_insert 04_checks 05_consultas* x 06_indices SQL File 13*

-- EXPLAIN Q3

```

77 • EXPLAIN
78 SELECT
79     p.marca,
80     COUNT(*) AS cantidad_productos,
81     ROUND(AVG(p.precio), 2) AS precio_promedio
82 FROM producto p
83 WHERE p.eliminado = 0
84 GROUP BY p.marca
85 HAVING COUNT(*) >= 3000 AND AVG(p.precio) >= 150;
86
87
88 -- -----

```

Result Grid | Filter Rows: Export: Wrap Cell Content: SQL

id	select_type	table	type	possible_keys	key	key_len	ref	rows	Extra
1	SIMPLE	p	index	ix_producto_marca	ix_producto_marca	323	const	9838	Using where

Result Grid | Form Editor | Field Types | Query Stats

Action Output

#	Time	Action	Message	Duration / Fetch
28	14:59:25	EXPLAIN SELECT p.id, p.nombre, p.marca, p.categoría, p.precio, c.tipo, c.valor, c.fecha_asignacion FROM producto p	2 row(s) returned	0.000 sec / 0.000 sec
29	15:00:38	SELECT p.categoría, COUNT() AS cantidad, ROUND(AVG(p.precio), 2) AS precio_promedio FROM producto p	2 row(s) returned	0.015 sec / 0.000 sec
30	15:01:42	EXPLAIN SELECT p.categoría, COUNT() AS cantidad, ROUND(AVG(p.precio), 2) AS precio_promedio FROM producto p	2 row(s) returned	0.000 sec / 0.000 sec
31	15:02:06	SELECT p.marca, COUNT() AS cantidad_productos, ROUND(AVG(p.precio), 2) AS precio_promedio FROM producto p	0 row(s) returned	0.031 sec / 0.000 sec
32	15:03:15	SELECT p.marca, COUNT() AS cantidad_productos, ROUND(AVG(p.precio), 2) AS precio_promedio FROM producto p	6 row(s) returned	0.016 sec / 0.000 sec
33	15:04:25	EXPLAIN SELECT p.marca, COUNT() AS cantidad_productos, ROUND(AVG(p.precio), 2) AS precio_promedio FROM producto p	1 row(s) returned	0.000 sec / 0.000 sec

Output | Action Output | Result Grid | Form Editor | Field Types | Query Stats | Read Only | Context Help | Snippets

● Q4 Subconsulta

SQL Editor:

```

 01_schema 02_seed 03_mass_insert 04_checks 05_consulta* 06_indices SQL File 13*
 94     p.categoría,
 95     p.precio
 96 FROM producto p
 97 WHERE p.eliminado = 0
 98 AND p.precio >
 99     (SELECT AVG(p2.precio)
 100    FROM producto p2
 101   WHERE p2.eliminado = 0
 102     AND p2.categoría = p.categoría)
 103 ORDER BY p.precio DESC
 104 LIMIT 50;
 105
 106 -- EXPLAIN Q4
  
```

Result Grid:

ID	Nombre	Categoría	Precio
1997	Producto 1995	Alimentos	574.25
9997	Producto 9995	Alimentos	574.25
8997	Producto 8995	Alimentos	574.25
9497	Producto 9495	Alimentos	574.25
3997	Producto 3995	Alimentos	574.25
4497	Producto 4495	Alimentos	574.25
4997	Producto 4995	Alimentos	574.25
3996	Producto 3994	Bebidas	573.10
4996	Producto 4994	Bebidas	573.10
4496	Producto 4494	Bebidas	573.10
7496	Producto 7494	Bebidas	573.10
6996	Producto 6994	Bebidas	573.10
6496	Producto 6494	Bebidas	573.10
2996	Producto 2994	Bebidas	573.10
3496	Producto 3494	Bebidas	573.10
5496	Producto 5494	Bebidas	573.10
5996	Producto 5994	Bebidas	573.10

Action Output:

#	Time	Action	Message	Duration / Fetch
29	15:00:38	SELECT p.categoría, COUNT(*) AS cantidad, ROUND(AVG(p.precio), 2) AS precio_promedio FROM producto p WHERE p.eliminado = 0 AND p.precio > (SELECT AVG(p2.precio) FROM producto p2 WHERE p2.eliminado = 0 AND p2.categoría = p.categoría) ORDER BY p.precio DESC LIMIT 50;	2 row(s) returned	0.015 sec / 0.000 sec
30	15:01:42	EXPLAIN SELECT p.categoría, COUNT(*) AS cantidad, ROUND(AVG(p.precio), 2) AS precio_promedio FROM producto p WHERE p.eliminado = 0 AND p.precio > (SELECT AVG(p2.precio) FROM producto p2 WHERE p2.eliminado = 0 AND p2.categoría = p.categoría) ORDER BY p.precio DESC LIMIT 50;	0 row(s) returned	0.000 sec / 0.000 sec
31	15:02:06	SELECT p.marca, COUNT(*) AS cantidad_productos, ROUND(AVG(p.precio), 2) AS precio_promedio FROM producto p WHERE p.eliminado = 0 AND p.precio > (SELECT AVG(p2.precio) FROM producto p2 WHERE p2.eliminado = 0 AND p2.categoría = p.categoría) ORDER BY p.precio DESC LIMIT 1000;	0 row(s) returned	0.031 sec / 0.000 sec
32	15:03:15	SELECT p.marca, COUNT(*) AS cantidad_productos, ROUND(AVG(p.precio), 2) AS precio_promedio FROM producto p WHERE p.eliminado = 0 AND p.precio > (SELECT AVG(p2.precio) FROM producto p2 WHERE p2.eliminado = 0 AND p2.categoría = p.categoría) ORDER BY p.precio DESC LIMIT 1000;	6 row(s) returned	0.016 sec / 0.000 sec
33	15:04:25	EXPLAIN SELECT p.marca, COUNT(*) AS cantidad_productos, ROUND(AVG(p.precio), 2) AS precio_promedio FROM producto p WHERE p.eliminado = 0 AND p.precio > (SELECT AVG(p2.precio) FROM producto p2 WHERE p2.eliminado = 0 AND p2.categoría = p.categoría) ORDER BY p.precio DESC LIMIT 1000;	1 row(s) returned	0.000 sec / 0.000 sec
34	15:05:42	SELECT p.id, p.nombre, p.categoría, p.precio FROM producto p WHERE p.eliminado = 0 AND p.precio > (SELECT AVG(p2.precio) FROM producto p2 WHERE p2.eliminado = 0 AND p2.categoría = p.categoría) ORDER BY p.precio DESC LIMIT 50;	1000 row(s) returned	0.047 sec / 0.000 sec

01_schema 02_seed 03_mass_insert 04_checks 05_consultas* 06_indices SQL File 13*

109 p.id, p.nombre, p.categoría, p.precio
110 FROM producto p
111 WHERE p.eliminado = 0
112 AND p.precio >
113 (SELECT AVG(p2.precio)
114 FROM producto p2
115 WHERE p2.eliminado = 0
116 AND p2.categoría = p.categoría)
117 ORDER BY p.precio DESC
118 LIMIT 50;

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

Result Grid Filter Rows: Export: Wrap Cell Content:

ID	Select Type	Table	Type	Possible Keys	Key	Key Len	Ref	Rows	Extra
1	PRIMARY	p	index	NULL	ix_producto_precio	5	NULL	50	Using where
2	DEPENDENT SUBQUERY	p2	ref	ix_producto_categoria	ix_producto_categoria	323	tf_p2_bd1.p.categoría	983	Using where

Result Grid

Form Editor

Field Types

Query Stats

Output Action Output

#	Time	Action	Message	Duration / Fetch
30	15:01:42	EXPLAIN SELECT p.categoría, COUNT(*) AS cantidad, ROUND(AVG(p.precio), 2) AS precio_promedio FROM producto p WHERE p.eliminado = 0 AND p.precio > (SELECT AVG(p2.precio) FROM producto p2 WHERE p2.eliminado = 0 AND p2.categoría = p.categoría) ORDER BY p.precio DESC LIMIT 50;	2 row(s) returned	0.000 sec / 0.000 sec
31	15:02:06	SELECT p.marca, COUNT(*) AS cantidad_productos, ROUND(AVG(p.precio), 2) AS precio_promedio FROM producto p WHERE p.eliminado = 0 AND p.precio > (SELECT AVG(p2.precio) FROM producto p2 WHERE p2.eliminado = 0 AND p2.categoría = p.categoría) ORDER BY p.precio DESC LIMIT 50;	0 row(s) returned	0.031 sec / 0.000 sec
32	15:03:15	SELECT p.marca, COUNT(*) AS cantidad_productos, ROUND(AVG(p.precio), 2) AS precio_promedio FROM producto p WHERE p.eliminado = 0 AND p.precio > (SELECT AVG(p2.precio) FROM producto p2 WHERE p2.eliminado = 0 AND p2.categoría = p.categoría) ORDER BY p.precio DESC LIMIT 50;	6 row(s) returned	0.016 sec / 0.000 sec
33	15:04:25	EXPLAIN SELECT p.marca, COUNT(*) AS cantidad_productos, ROUND(AVG(p.precio), 2) AS precio_promedio FROM producto p WHERE p.eliminado = 0 AND p.precio > (SELECT AVG(p2.precio) FROM producto p2 WHERE p2.eliminado = 0 AND p2.categoría = p.categoría) ORDER BY p.precio DESC LIMIT 50;	1 row(s) returned	0.000 sec / 0.000 sec
34	15:05:42	SELECT p.id, p.nombre, p.categoría, p.precio FROM producto p WHERE p.eliminado = 0 AND p.precio > (SELECT AVG(p2.precio) FROM producto p2 WHERE p2.eliminado = 0 AND p2.categoría = p.categoría) ORDER BY p.precio DESC LIMIT 50;	1000 row(s) returned	0.047 sec / 0.000 sec
35	15:06:27	EXPLAIN SELECT p.id, p.nombre, p.categoría, p.precio FROM producto p WHERE p.eliminado = 0 AND p.precio > (SELECT AVG(p2.precio) FROM producto p2 WHERE p2.eliminado = 0 AND p2.categoría = p.categoría) ORDER BY p.precio DESC LIMIT 50;	2 row(s) returned	0.000 sec / 0.000 sec

Read Only Context Help Snippets

4.2 Vista

Se creó una vista para facilitar consultas frecuentes.

The screenshot shows the MySQL Workbench interface with the following details:

- File Bar:** Local instance MySQL80 - W..., File, Edit, View, Query, Database, Server, Tools, Scripting, Help.
- Navigator:** Schemas (libreria, phpmyadmin, reservas_areas, test, lib_p2.bd1), Tables (aux_nums, codigo_barcas, products, vw_productos_detalle).
- SQL Editor:** SQL File 13*, 1 • SELECT * FROM vw_productos_detalle LIMIT 20;
- Result Grid:** Shows 16 rows of product data from the vw_productos_detalle view.
- Information:** Table: codigo_baras, Columns: id, eliminado, tipo, valor, fecha_asignacion, observaciones, producto_id.
- Output:** Action Output showing the execution history of the query.

#	Time	Action	Message	Duration / Fetch
16	14:41:36	SELECT COUNT() AS total_codigos FROM codigo_baras LIMIT 0, 1000	1 row(s) returned	0.016 sec / 0.000 sec
17	14:41:36	SELECT COUNT() AS productos_sin_codigo FROM producto p LEFT JOIN codigo_baras c ON c.producto_id = p.id WHERE c.codigo_baras IS NULL	1 row(s) returned	0.015 sec / 0.000 sec
18	14:43:06	SELECT COUNT() FROM producto LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec
19	14:43:06	SELECT COUNT() FROM codigo_baras LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec
20	14:47:27	CREATE OR REPLACE VIEW vw_productos_detalle AS SELECT p.id, p.nombre, p.marca, p.categor... 0 rows affected	0.015 sec	
21	14:48:21	SELECT * FROM vw_productos_detalle LIMIT 20	20 row(s) returned	0.000 sec / 0.000 sec

4.3 Índices y comparación

Screenshot of MySQL Workbench showing a query editor and output window.

Query Editor:

```

55 JOIN codigo_barras c ON c.producto_id = p.id
56 WHERE p.eliminado = 0
57 AND c.eliminado = 0
58 AND p.categoría IN ('Tecnología', 'Alimentos')
59 AND c.fecha_asignacion BETWEEN '2025-03-01' AND '2025-06-30'
60 GROUP BY p.categoría;
61
62
63
64 -- Q3 (GROUP BY + HAVING): marcas con muchos productos y precio promedio alto
65
66 • SELECT
67     p.marca,

```

Execution Plan:

id	select_type	table	type	possible_keys	key	key_len	ref	rows	Extra
1	SIMPLE	c	ALL	uq_codigo_barras_producto_idx_codigo_fecha	NULL	NULL	NULL	9825	Using where; Using temporary; Using filesort
1	SIMPLE	p	eq_ref	PRIMARY,ix_producto_categoria,ix_producto...	PRIMARY	8	tf_p2_bd1.c.producto_id	1	Using where

Output Window:

Action Output

#	Time	Action	Message	Duration / Fetch
46	15:10:16	SELECT p.categoría, COUNT() AS cantidad, ROUND(AVG(p.precio), 2) AS precio_promedio FROM producto p	2 row(s) returned	0.031 sec / 0.000 sec
47	15:10:59	EXPLAIN SELECT p.categoría, COUNT() AS cantidad, ROUND(AVG(p.precio), 2) AS precio_promedio FRO...	2 row(s) returned	0.000 sec / 0.000 sec
48	15:11:47	USE tf_p2_bd1	0 row(s) affected	0.000 sec
49	15:11:47	CREATE INDEX idx_producto_categoria ON producto(categoría)	0 row(s) affected, 1 warning(s): 1831 Duplicate index 'idx_producto_categoria'. This is deprecated and will be dis...	0.047 sec
50	15:11:47	CREATE INDEX idx_codigo_fecha ON codigo_barras(fecha_asignacion)	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.031 sec
51	15:12:12	EXPLAIN SELECT p.categoría, COUNT() AS cantidad, ROUND(AVG(p.precio), 2) AS precio_promedio FRO...	2 row(s) returned	0.015 sec / 0.000 sec

01_schema 02_seed 03_mass_insert 04_checks 05_consultas* 06_indices SQL File 13*

```

52 COUNT(*) AS cantidad,
53 ROUND(AVG(p.precio), 2) AS precio_promedio
54 FROM producto p
55 JOIN codigo_barra c ON c.producto_id = p.id
56 WHERE p.eliminado = 0
57 AND c.eliminado = 0
58 AND p.categoría IN ('Tecnología', 'Alimentos')
59 AND c.fecha_asignacion BETWEEN '2025-03-01' AND '2025-06-30'
60 GROUP BY p.categoría;
61
62
63
64 -- Q3 (GROUP BY + HAVING): marcas con muchos productos y precio promedio alto

```

Result Grid | Filter Rows: Export: Wrap Cell Content: □

id	select_type	table	type	possible_keys	key	key_len	ref	rows	Extra
1	SIMPLE	p	ALL	PRIMARY,ix_producto_categoria	PRIMARY	4	tfl_p2_bd1.p.id	1	Using where; Using temporary; Using filesort
1	SIMPLE	c	eq_ref	uq_codigo_barra_producto	uq_codigo_barra_producto	8			Using where

SQLAdditions: <| > | ⌂ ⌂ ⌂ ⌂ Jump to

Automatic context help is disabled. Use the toolbar manually get help for the current caret position or toggle automatic help.

Result 16 x

Output

Action Output

#	Time	Action	Message	Duration / Fetch
42	15:08:42	CREATE INDEX idx_producto_categoria ON producto(categoría)	Error Code: 1061. Duplicate key name 'idx_producto_categoria'	0.000 sec
43	15:09:35	USE tfl_p2_bd1	0 row(s) affected	0.000 sec
44	15:09:35	DROP INDEX idx_producto_categoria ON producto	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.015 sec
45	15:09:35	DROP INDEX idx_codigo_fecha ON codigo_barra	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.016 sec
46	15:10:16	SELECT p.categoría, COUNT() AS cantidad, ROUND(AVG(p.precio), 2) AS precio_promedio FROM producto p JOIN codigo_barra c ON c.producto_id = p.id WHERE p.eliminado = 0 AND c.eliminado = 0 AND p.categoría IN ('Tecnología', 'Alimentos') AND c.fecha_asignacion BETWEEN '2025-03-01' AND '2025-06-30' GROUP BY p.categoría ORDER BY cantidad DESC;	2 row(s) returned	0.031 sec / 0.000 sec
47	15:10:59	EXPLAIN SELECT p.categoría, COUNT() AS cantidad, ROUND(AVG(p.precio), 2) AS precio_promedio FROM producto p JOIN codigo_barra c ON c.producto_id = p.id WHERE p.eliminado = 0 AND c.eliminado = 0 AND p.categoría IN ('Tecnología', 'Alimentos') AND c.fecha_asignacion BETWEEN '2025-03-01' AND '2025-06-30' GROUP BY p.categoría ORDER BY cantidad DESC;	2 row(s) returned	0.000 sec / 0.000 sec

SQLAdditions: <| > | ⌂ ⌂ ⌂ ⌂ Jump to

Automatic context help is disabled. Use the toolbar manually get help for the current caret position or toggle automatic help.

01_schema 02_seed 03_mass_insert 04_checks 05_consultas* 06_indices SQL File 13*

```

37 COUNT(*) AS cantidad,
38 ROUND(AVG(p.precio), 2) AS precio_promedio
39 FROM producto p
40 JOIN codigo_barra c ON c.producto_id = p.id
41 WHERE p.eliminado = 0
42 AND c.eliminado = 0
43 AND p.categoría IN ('Tecnología', 'Alimentos')
44 AND c.fecha_asignacion BETWEEN '2025-03-01' AND '2025-06-30'
45 GROUP BY p.categoría
46 ORDER BY cantidad DESC;
47
48 -- EXPLAIN Q2
49 • EXPLAIN

```

Result Grid | Filter Rows: Export: Wrap Cell Content: □

categoria	cantidad	precio_promedio
Alimentos	665	289.71
Tecnología	665	290.86

SQLAdditions: <| > | ⌂ ⌂ ⌂ ⌂ Jump to

Automatic context help is disabled. Use the toolbar manually get help for the current caret position or toggle automatic help.

Result 15 x

Output

Action Output

#	Time	Action	Message	Duration / Fetch
41	15:08:42	USE tfl_p2_bd1	0 row(s) affected	0.000 sec
42	15:08:42	CREATE INDEX idx_producto_categoria ON producto(categoría)	Error Code: 1061. Duplicate key name 'idx_producto_categoria'	0.000 sec
43	15:09:35	USE tfl_p2_bd1	0 row(s) affected	0.000 sec
44	15:09:35	DROP INDEX idx_producto_categoria ON producto	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.015 sec
45	15:09:35	DROP INDEX idx_codigo_fecha ON codigo_barra	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.016 sec
46	15:10:16	SELECT p.categoría, COUNT() AS cantidad, ROUND(AVG(p.precio), 2) AS precio_promedio FROM producto p JOIN codigo_barra c ON c.producto_id = p.id WHERE p.eliminado = 0 AND c.eliminado = 0 AND p.categoría IN ('Tecnología', 'Alimentos') AND c.fecha_asignacion BETWEEN '2025-03-01' AND '2025-06-30' GROUP BY p.categoría ORDER BY cantidad DESC;	2 row(s) returned	0.031 sec / 0.000 sec

SQLAdditions: <| > | ⌂ ⌂ ⌂ ⌂ Jump to

Automatic context help is disabled. Use the toolbar manually get help for the current caret position or toggle automatic help.

Se realizaron pruebas comparativas con y sin índices sobre las columnas categoría y fecha_asignacion.

En ausencia de índices, la tabla producto fue recorrida completamente (type = ALL).

Con índices aplicados, el optimizador modificó el orden del join y utilizó accesos por clave primaria en producto.

Sin embargo, debido a la baja selectividad del rango de fechas, el motor optó por mantener escaneo completo sobre `codigo_barras`.

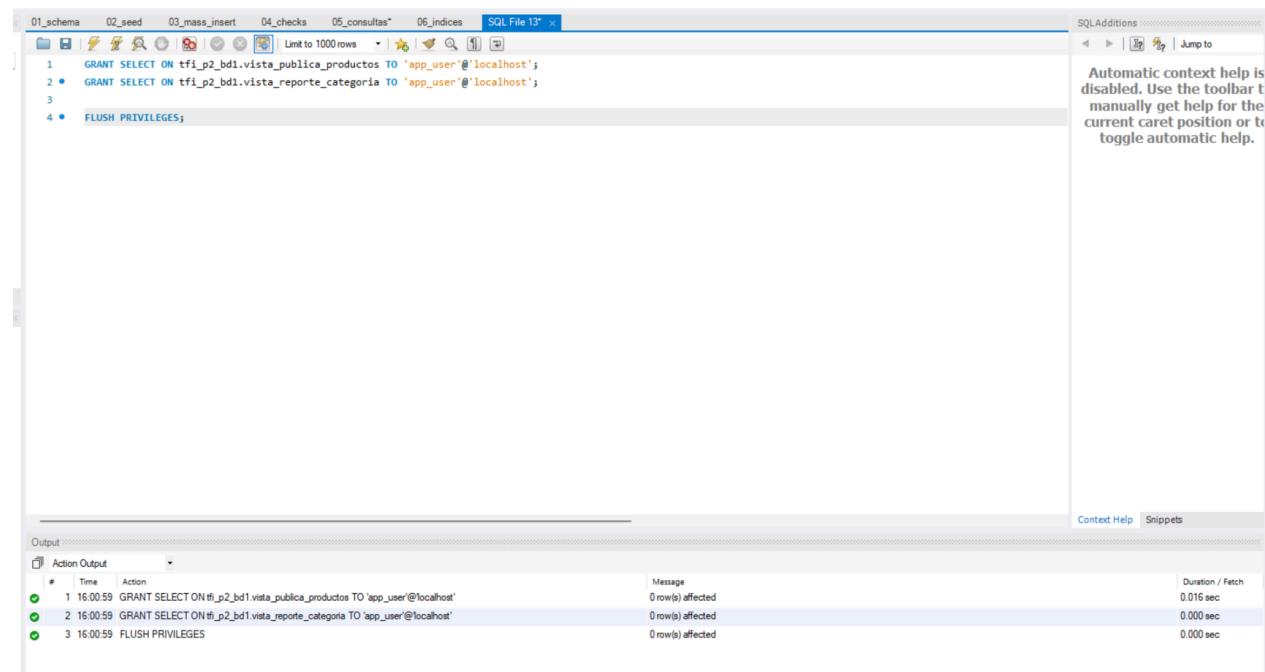
Escenario	type	rows	Extra
Sin índice	ALL	~9800	Using temporary
Con índice	eq_ref / PRIMARY	1	Using where

5. Etapa 4 – Seguridad

4.1 Creación del usuario de aplicación

Se creó un usuario de aplicación llamado `app_user` con acceso restringido.

Este usuario no posee permisos sobre tablas, únicamente puede consultar vistas específicas del sistema.



The screenshot shows the MySQL Workbench interface with the following details:

SQL Editor: Contains the following SQL code:

```
1 GRANT SELECT ON tfi_p2_bd1.vista_publica_productos TO 'app_user'@'localhost';
2 • GRANT SELECT ON tfi_p2_bd1.vista_reporte_categoria TO 'app_user'@'localhost';
3
4 • FLUSH PRIVILEGES;
```

Output Window: Shows the execution results:

#	Action	Time	Message	Duration / Fetch
1	GRANT SELECT ON tfi_p2_bd1.vista_publica_productos TO 'app_user'@'localhost'	16:00:59	0 row(s) affected	0.016 sec
2	GRANT SELECT ON tfi_p2_bd1.vista_reporte_categoria TO 'app_user'@'localhost'	16:00:59	0 row(s) affected	0.000 sec
3	FLUSH PRIVILEGES	16:00:59	0 row(s) affected	0.000 sec

4.2 Verificación de permisos

The screenshot shows the MySQL Workbench interface with a query editor titled "Query 1". The query "SHOW GRANTS" has been run, and the results are displayed in a result grid. The results show grants for the user "app_user" on the database "localhost". The grants include USAGE, SELECT, and SELECT privileges on specific tables within the database.

GRANT	ON	FOR
GRANT USAGE ON .* TO 'app_user'@'localhost';		
GRANT SELECT ON `tf1_p2_bd1`.`vista_reporte...`;		
GRANT SELECT ON `tf1_p2_bd1`.`vista_publica...`;		

The output pane shows the execution details: 3 row(s) returned, duration 0.000 sec / 0.000 sec.

4.3 Prueba funcional del usuario

The screenshot shows the MySQL Workbench interface with a query editor titled "Query 1". Three queries have been run:

- USE tf1_p2_bd1;
-
- SELECT * FROM vista_publica_productos LIMIT 5;
- SELECT * FROM vista_reporte_categoria LIMIT 5;

The results of the second query are displayed in a result grid:

categoria	cantidad_productos	precio_promedio
Alimentos	2002	289.34
Bebidas	2000	294.23
Hogar	2000	293.08
Limpieza	2000	291.93
Tecnología	2001	290.68

The output pane shows the execution details for all four queries: 0 row(s) affected, 5 row(s) returned, 5 row(s) returned, and 0.016 sec / 0.000 sec.

4.4 Prueba de restricción

The screenshot shows the MySQL Workbench interface. In the top-left, the 'Query 1' tab is active, displaying the following SQL code:

```
1 • USE tfi_p2_bd1;
2
3 • SELECT * FROM producto LIMIT 5;
```

In the top-right corner, there is a message from the 'SQLAdditions' panel: "Automatic context help is disabled. Use the toolbar or manually get help for the current caret position or toggle automatic help."

Below the tabs, the 'Output' tab is selected, showing the execution log:

Action	Time	Message	Duration / Fetch
1 16:02:53	USE tfi_p2_bd1	0 row(s) affected	0.000 sec
2 16:02:53	SELECT * FROM vista_publica_productos LIMIT 5	5 row(s) returned	0.000 sec / 0.000 sec
3 16:02:53	SELECT * FROM vista_reporte_categoria LIMIT 5	5 row(s) returned	0.016 sec / 0.000 sec
4 16:03:45	USE tfi_p2_bd1	0 row(s) affected	0.000 sec
5 16:03:45	SELECT * FROM producto LIMIT 5	Error Code: 1142. SELECT command denied to user 'app_user'@'localhost' for table 'tfi_p2_bd1`.`producto'	0.015 sec

Se intentó acceder directamente a la tabla producto, obteniendo error de permisos.

Esto confirma que el usuario solo puede consultar vistas y no modificar ni acceder a las tablas base.

5. Etapa 5 – Conurrencia y Transacciones

5.1 Simulación de deadlock

Se realizaron pruebas utilizando dos sesiones simultáneas en MySQL Workbench con el usuario administrador.

En la primera sesión se inició una transacción y se actualizó el producto con `id = 1`, manteniendo el bloqueo sin realizar COMMIT.

En la segunda sesión se inició otra transacción y se actualizó el producto con `id = 2`.

Posteriormente, cada sesión intentó actualizar el registro bloqueado por la otra sesión (orden inverso de acceso), generando un ciclo de espera circular.

El motor InnoDB detectó automáticamente la situación y abortó una de las transacciones, devolviendo el siguiente error:

MySQL Workbench

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

- libreria
- phpmyadmin
- reservas_aereas
- test
- tfi_p2_bd1**
- Tables
- Views
- Stored Procedures
- Functions

05_consultas* 06_indices SQL File 13* SQL File 9* SQL File 10*

1 • `SELECT precio FROM producto WHERE id = 1;`

Result Grid | Filter Rows: | Export: | Wrap Cell Content | Result Grid | Form Editor | Field Types

precio
120.00

Administration Schemas

No object selected

product 3 x Read Only Context Help Snippets

Action Output

#	Time	Action	Message	Duration / Fetch
7	18:05:45	CALL sp_actualizar_marca_con_retry(1, 'MARC...')	1 row(s) affected	0.016 sec
8	18:06:45	SET SESSION TRANSACTION ISOLATION LE...	0 row(s) affected	0.000 sec
9	18:06:45	START TRANSACTION	0 row(s) affected	0.000 sec
10	18:06:45	SELECT precio FROM producto WHERE id = 1...	1 row(s) returned	0.000 sec / 0.000 sec
11	18:07:28	SELECT precio FROM producto WHERE id = 1...	1 row(s) returned	0.000 sec / 0.000 sec
12	18:07:48	ROLLBACK	0 row(s) affected	0.000 sec
13	18:07:48	SET SESSION TRANSACTION ISOLATION LE...	0 row(s) affected	0.000 sec
14	18:07:48	START TRANSACTION	0 row(s) affected	0.000 sec
15	18:07:48	SELECT precio FROM producto WHERE id = 1...	1 row(s) returned	0.000 sec / 0.000 sec
16	18:08:18	SELECT precio FROM producto WHERE id = 1...	1 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

Query Completed

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

MySQL Workbench

File Edit View Query Database Server Tools Scripting Help

Navigator Schemas

Filter objects

LIBERIA
phpmyadmin
reservas_aereas
test
tfi_p2_bd1
Tables
Views
Stored Procedures
Functions

Query 1 SQL File 1* x

START TRANSACTION;
UPDATE producto SET precio = precio + 10 WHERE id = 1;
COMMIT;

SQLAdditions

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

No object selected

Information

Action Output

#	Time	Action	Message	Duration / Fetch
3	18:03:03	START TRANSACTION	0 row(s) affected	0.000 sec
4	18:03:03	UPDATE producto SET marca = 'B_LOCK' WHERE id = 1	1 row(s) affected Rows matched: 1 Changed: 1 ...	0.000 sec
5	18:03:56	UPDATE producto SET marca = 'B_TRY' WHERE id = 1	1 row(s) affected Rows matched: 1 Changed: 1 ...	0.000 sec
6	18:04:20	ROLLBACK	0 row(s) affected	0.015 sec
7	18:07:01	START TRANSACTION	0 row(s) affected	0.016 sec
8	18:07:01	UPDATE producto SET precio = precio + 10 WHERE id = 1	1 row(s) affected Rows matched: 1 Changed: 1 ...	0.000 sec
9	18:07:01	COMMIT	0 row(s) affected	0.015 sec
10	18:08:01	START TRANSACTION	0 row(s) affected	0.000 sec
11	18:08:01	UPDATE producto SET precio = precio + 10 WHERE id = 1	1 row(s) affected Rows matched: 1 Changed: 1 ...	0.016 sec
12	18:08:01	COMMIT	0 row(s) affected	0.000 sec

Object Info Session

Query Completed

MySQL Workbench

File Edit View Query Database Server Tools Scripting Help

Navigator : 05_consultas* 06_indices SQL File 13* SQL File 9* SQL File 10*

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

SCHEMAS : libreria, phpmyadmin, reservas_aereas, test, tfi_p2_bd1

Result Grid : precio | 110.00

No object selected

Output : Action Output

#	Time	Action	Message	Duration / Fetch
6	18:05:32	CREATE PROCEDURE sp_actualizar_marca_c...	0 row(s) affected	0,016 sec
7	18:05:45	CALL sp_actualizar_marca_con_retry(1, 'MARC...	1 row(s) affected	0,016 sec
8	18:06:45	SET SESSION TRANSACTION ISOLATION LE...	0 row(s) affected	0,000 sec
9	18:06:45	START TRANSACTION	0 row(s) affected	0,000 sec
10	18:06:45	SELECT precio FROM producto WHERE id = 1...	1 row(s) returned	0,000 sec / 0,000 sec
11	18:07:28	SELECT precio FROM producto WHERE id = 1...	1 row(s) returned	0,000 sec / 0,000 sec
12	18:07:48	ROLLBACK	0 row(s) affected	0,000 sec
13	18:07:48	SET SESSION TRANSACTION ISOLATION LE...	0 row(s) affected	0,000 sec
14	18:07:48	START TRANSACTION	0 row(s) affected	0,000 sec
15	18:07:48	SELECT precio FROM producto WHERE id = 1...	1 row(s) returned	0,000 sec / 0,000 sec

MySQL Workbench

File Edit View Query Database Server Tools Scripting Help

Navigator 05_consultas* 06_indices SQL File 13* SQL File 9* SQL File 10* SQLAdditions

SCHEMAS Filter objects libreria phpmyadmin reservas_aereas test tfi_p2_bd1 Tables Views Stored Procedures Functions

1 • SELECT precio FROM producto WHERE id = 1;

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

Result Grid | Filter Rows: Export: Wrap Cell Content | Result Grid | Form Editor | Field Types

precio
100.00

No object selected

Output

Action Output

#	Time	Action	Message	Duration / Fetch
2	18:00:32	SET autocommit = 0	0 row(s) affected	0,000 sec
3	18:00:32	START TRANSACTION	0 row(s) affected	0,000 sec
4	18:00:32	UPDATE producto SET marca = 'A_LOCK' WHERE id = 1	0 row(s) affected Rows matched: 1 Changed: 0,000 sec	0,015 sec
5	18:03:37	UPDATE producto SET marca = 'A_TRY' WHERE id = 1	Error Code: 1213. Deadlock found when trying to lock row in table 'producto'. Lost connection while waiting for lock. Aborting transaction.	19,438 sec
6	18:05:32	CREATE PROCEDURE sp_actualizar_marca_c...	0 row(s) affected	0,016 sec
7	18:05:45	CALL sp_actualizar_marca_con_retry(1, 'MARC...', 1)	1 row(s) affected	0,016 sec
8	18:06:45	SET SESSION TRANSACTION ISOLATION LEVEL REPEATABLE READ	0 row(s) affected	0,000 sec
9	18:06:45	START TRANSACTION	0 row(s) affected	0,000 sec
10	18:06:45	SELECT precio FROM producto WHERE id = 1	1 row(s) returned	0,000 sec / 0,000 sec
11	18:07:28	SELECT precio FROM producto WHERE id = 1	1 row(s) returned	0,000 sec / 0,000 sec

Object Info Session

MySQL Workbench

File Edit View Query Database Server Tools Scripting Help

Navigator :: mass_insert 04_checks 05_consultas* 06_indices SQL File 13* Limit to 1000 rows SQLAdditions Jump to

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

```

13
14      IF v_intentos > 2 THEN
15          SIGNAL SQLSTATE '45000'
16              SET MESSAGE_TEXT = 'Deadlock: reintentos agotados';
17      END IF;
18
19      DO SLEEP(0.2 * v_intentos);
20  END;
21
22  START TRANSACTION;
23  UPDATE producto SET marca = p_marca WHERE id = p_id;
24  COMMIT;
25
26  LEAVE retry_loop;
27  END;
28 END LOOP;
29
30 END$$
31
32 DELIMITER ;
33
34 -----
35
36 CALL sp_actualizar_marca_con_retry(1, 'MARCA_OK');

```

No object selected

Context Help Snippets

Output :: Action Output

#	Time	Action	Message	Duration / Fetch
1	18:00:32	USE tfi_p2_bd1	0 row(s) affected	0.000 sec
2	18:00:32	SET autocommit = 0	0 row(s) affected	0.000 sec
3	18:00:32	START TRANSACTION	0 row(s) affected	0.000 sec
4	18:00:32	UPDATE producto SET marca = 'A_LOCK' WHERE...	0 row(s) affected Rows matched: 1 Changed: 0 ...	0.015 sec
5	18:03:37	UPDATE producto SET marca = 'A_TRY' WHERE...	Error Code: 1213. Deadlock found when trying to g...	19.438 sec
6	18:05:32	CREATE PROCEDURE sp_actualizar_marca_con...	0 row(s) affected	0.016 sec
7	18:05:45	CALL sp_actualizar_marca_con_retry(1, 'MARCA_...	1 row(s) affected	0.016 sec

Object Info Session

MySQL Workbench

Local instance MySQL80 (tfi_p2_b.x) Local instance MySQL80 (tfi_p2_b.x)

File Edit View Query Database Server Tools Scripting Help

Navigator

Query 1

SCHEMAS

libreria
phpmyadmin
reservas_aereas
test
tfi_p2_bd1
Tables
Views
Stored Procedures
Functions

Administration Schemas

No object selected

Information

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

1 USE tfi_p2_bd1;
2
3 SET autocommit = 0;
4 START TRANSACTION;
5
6 -- Bloquea producto 2
7 UPDATE producto
SET marca = 'B_LOCK'
WHERE id = 2;
8
9
10 ----
11 ----
12
13 UPDATE producto
SET marca = 'B_TRY'
WHERE id = 1;
14
15 ----
16
17 ----
18
19 ROLLBACK;

Output

Action Output

#	Time	Action	Message	Duration / Fetch
1	18:03:03	USE tfi_p2_bd1	0 row(s) affected	0.000 sec
2	18:03:03	SET autocommit = 0	0 row(s) affected	0.000 sec
3	18:03:03	START TRANSACTION	0 row(s) affected	0.000 sec
4	18:03:03	UPDATE producto SET marca = 'B_LOCK' WHERE id = 2;	1 row(s) affected Rows matched: 1 Changed: 1 ...	0.000 sec
5	18:03:56	UPDATE producto SET marca = 'B_TRY' WHERE id = 1;	1 row(s) affected Rows matched: 1 Changed: 1 ...	0.000 sec
6	18:04:20	ROLLBACK	0 row(s) affected	0.015 sec

Object Info Session

Ready

MySQL Workbench

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- libreria
- phpmyadmin
- reservas_aereas
- test
- tfi_p2_bd1
 - Tables
 - Views
 - Stored Procedures
 - Functions

SQL File 13*

Limit to 1000 rows

SQLAdditions

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

```
1 USE tfi_p2_bd1;
2
3 SET autocommit = 0;
4 START TRANSACTION;
5
6 -- Bloquea producto 1
7 UPDATE producto
8 SET marca = 'A_LOCK'
9 WHERE id = 1;
10
11 ----
12
13 UPDATE producto
14 SET marca = 'A_TRY'
15 WHERE id = 2;
```

Administration Schemas

No object selected

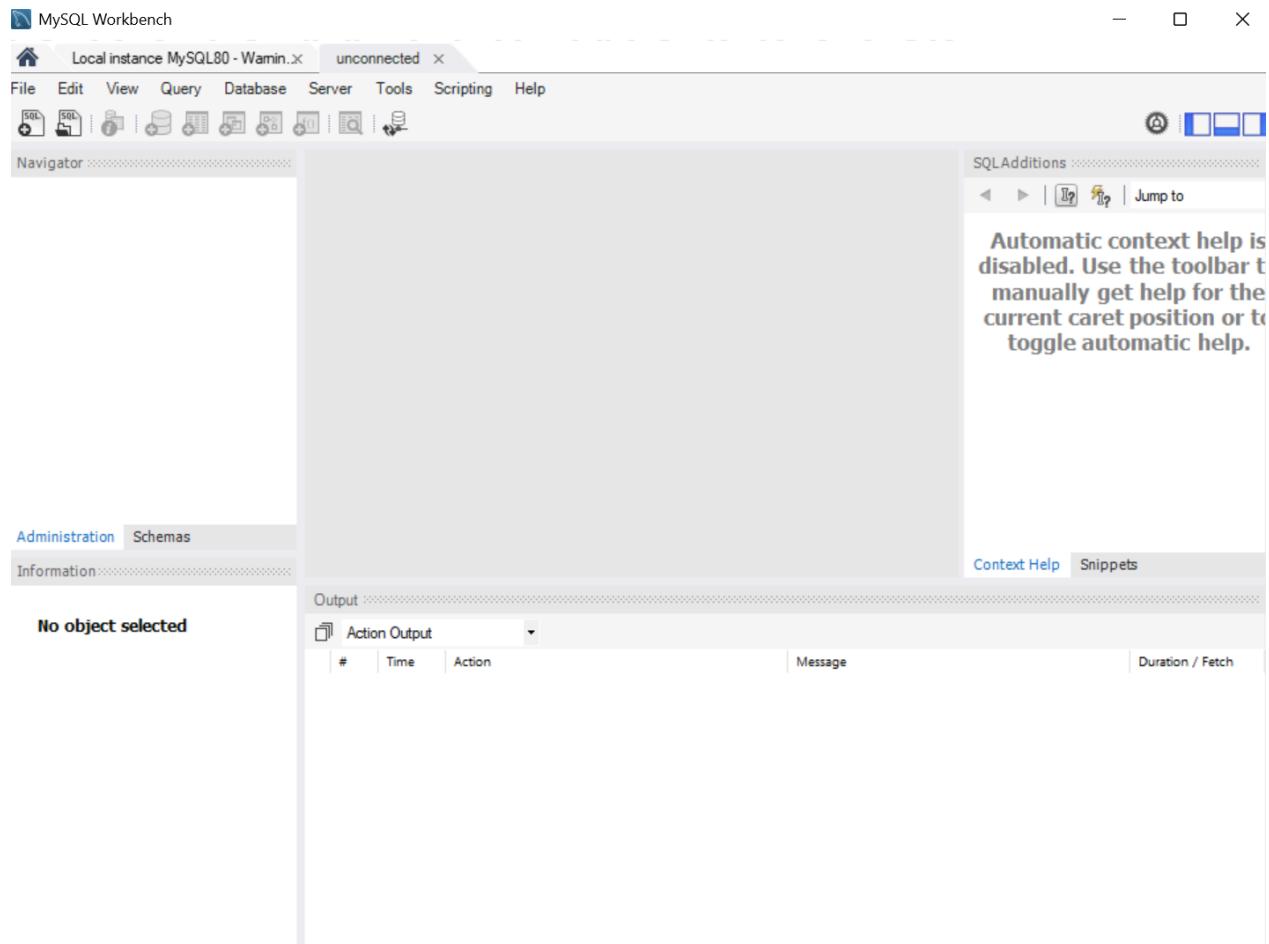
Information

Action Output

#	Time	Action	Message	Duration / Fetch
1	18:00:32	USE tfi_p2_bd1	0 row(s) affected	0.000 sec
2	18:00:32	SET autocommit = 0	0 row(s) affected	0.000 sec
3	18:00:32	START TRANSACTION	0 row(s) affected	0.000 sec
4	18:00:32	UPDATE producto SET marca = 'A_LOCK' WHERE...	0 row(s) affected Rows matched: 1 Changed: 0 ...	0.015 sec
5	18:03:37	UPDATE producto SET marca = 'A_TRY' WHERE...	Error Code: 1213. Deadlock found when trying to g...	19.438 sec

Object Info Session

Ready



MySQL Workbench

Local instance MySQL80 - Wamin.x app_user - Warning - not sup...

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

test

tfi_p2_bd1

- Tables
- Views
- Stored Procedures
- Functions

Query 1

```
1 • USE tfi_p2_bd1;
2
3 SET autocommit = 0;
4 • START TRANSACTION;
5
6 -- Bloquea producto 2
7 • UPDATE producto
8 SET marca = 'B_LOCK'
9 WHERE id = 2;
10
11 -- No hagas COMMIT todavía
```

SQLAdditions

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or toggle automatic help.

Administration Schemas

No object selected

Information

Output

Action Output

#	Time	Action	Message	Duration / Fetch
1	17:58:34	USE tfi_p2_bd1	0 row(s) affected	0.000 sec
2	17:58:34	SET autocommit = 0	0 row(s) affected	0.000 sec
3	17:58:34	START TRANSACTION	0 row(s) affected	0.000 sec
4	17:58:34	UPDATE producto SET marca = 'B_LOCK' WHERE id = 2;	Error Code: 1142. UPDATE command denied to user 'wamin'@'localhost' for table 'producto'	0.000 sec

Context Help Snippets