



---

# CONSULTAS – VISTAS - TRIGGERS

---

Presentado a: Juan Esteban Pineda

Presentado por: Jessica Andrea López Obando



## Consultas y Vistas

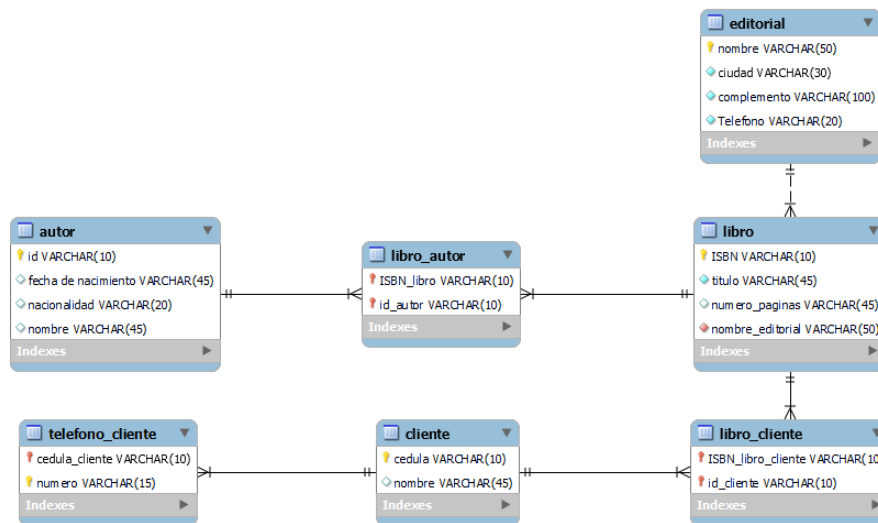
*“Cada día trae una nueva oportunidad”*

**PRIMERA ACTIVIDAD:** Utilizando el ejercicio de la Librería realizado en clase (se adjunta script SQL) realice lo siguiente:

Con el script SQL se ejecuta y se crean las tablas.



Se realiza un reverse engineer solo para tener una visualización más simple del tipo de datos de cada atributo y así realizar el registro en cada una de las tablas.



*Reverse Engineer*

1. Complete la información para las tablas autor, libro, cliente, editorial, libro\_cliente, libro\_autor y teléfono\_cliente con al menos (5,20,7,4,10,10, 12) registros respectivamente usando **únicamente** comandos SQL creados por usted. A continuación adjunto los registros de cada tabla.

### autor (5)

```
INSERT INTO autor (id, `fecha de nacimiento`, nacionalidad, nombre)
VALUES
("A001", "01-01-1900", "Español", "Juan García"),
("A002", "02-02-1920", "Mexicano", "Pedro Martínez"),
("A003", "03-03-1930", "Argentino", "Marta Rodríguez"),
("A004", "04-04-1940", "Colombiano", "Jorge López"),
("A005", "05-05-1950", "Peruano", "Isabel Pérez");
```

## Jessica Andrea López Obando Actividad 1, 2 y 3 (Librería y Hospital)

### libro (20)

```
INSERT INTO libro (ISBN, titulo, numero_paginas, nombre_editorial)
VALUES
('ISBN001', 'Programación para principiantes', '300', 'Editorial 1'),
('ISBN002', 'Autoayuda para tiempos difíciles', '250', 'Editorial 2'),
('ISBN003', 'Literatura clásica: Don Quijote', '500', 'Editorial 3'),
('ISBN004', 'JavaScript: Avanzado', '400', 'Editorial 1'),
('ISBN005', 'El poder de la positividad', '200', 'Editorial 2'),
('ISBN006', 'Cien años de soledad', '450', 'Editorial 3'),
('ISBN007', 'Python: Desde cero', '350', 'Editorial 1'),
('ISBN008', 'Superar miedos y ansiedades', '300', 'Editorial 2'),
('ISBN009', 'El amor en los tiempos del cólera', '400', 'Editorial 3'),
('ISBN010', 'CSS Avanzado', '250', 'Editorial 1'),
('ISBN011', 'El arte de amar', '200', 'Editorial 2'),
('ISBN012', 'One Hundred Years of Solitude', '450', 'Editorial 3'),
('ISBN013', 'SQL Básico', '300', 'Editorial 1'),
('ISBN014', 'La vida después del éxito', '250', 'Editorial 2'),
('ISBN015', 'La mala hora', '400', 'Editorial 3'),
('ISBN016', 'Java: Intermedio', '350', 'Editorial 1'),
('ISBN017', 'Los secretos de la mente millonaria', '300', 'Editorial 2'),
('ISBN018', 'La casa de papel', '250', 'Editorial 3'),
('ISBN019', 'Node.js: Básico', '200', 'Editorial 1'),
('ISBN020', 'El arte de la meditación', '450', 'Editorial 2');
```

### cliente (7)

```
INSERT INTO cliente (cedula, nombre)
VALUES
('123456789', 'Juan Perez'),
('987654321', 'María Rodriguez'),
('121212121', 'Carlos Gonzalez'),
('343434343', 'Sofía Martinez'),
('565656565', 'Luis Lopez'),
('787878788', 'Ana Garcia'),
('909090909', 'Pedro Sánchez');
```

### editorial (4)

```
INSERT INTO editorial(nombre, ciudad, complemento, telefono)
VALUES
('Editorial 1', 'Bogotá', 'Edificio A, Oficina B', '555-555-5555'),
('Editorial 2', 'Buenos Aires', 'Calle C, Número D', '555-555-5556'),
('Editorial 3', 'Lima', 'Esquina E, Edificio F', '555-555-5557'),
('Editorial 4', 'Ciudad de México', 'Calle G, Número H', '555-555-5558');
```

### libro\_cliente (10)

```
INSERT INTO libro_cliente (ISBN_libro_cliente, id_cliente)
VALUES
('ISBN001', '123456789'),
('ISBN005', '987654321'),
('ISBN006', '121212121'),
('ISBN007', '343434343'),
('ISBN009', '565656565'),
('ISBN013', '787878788'),
('ISBN015', '909090909'),
('ISBN002', '123456789'),
('ISBN003', '987654321'),
('ISBN014', '121212121');
```

### libro\_autor (10)

## Jessica Andrea López Obando Actividad 1, 2 y 3 (Librería y Hospital)

```
INSERT INTO libro_autor (ISBN_libro, id_autor)
VALUES
("ISBN001", "A001"),
("ISBN002", "A002"),
("ISBN003", "A003"),
("ISBN004", "A004"),
("ISBN005", "A005"),
("ISBN006", "A001"),
("ISBN007", "A002"),
("ISBN008", "A003"),
("ISBN009", "A004"),
("ISBN010", "A005");
```

### teléfono\_cliente (12)

```
INSERT INTO telefono_cliente (cedula_cliente, numero)
VALUES
("123456789", "555-555-1001"),
("987654321", "555-555-1002"),
("121212121", "555-555-1003"),
("343434343", "555-555-1004"),
("565656565", "555-555-1005"),
("787878788", "555-555-1006"),
("909090909", "555-555-1007"),
("123456789", "555-555-1008"),
("987654321", "555-555-1009"),
("121212121", "555-555-1010"),
("343434343", "555-555-1011"),
("565656565", "555-555-1012"),
("787878788", "555-555-1013");
```

## 2. Realice 5 consultas que me permitan

Conocer el nombre y la fecha de nacimiento de cada escritor.

```
-- -----
-- Conocer el nombre y la fecha de nacimiento de cada escritor
-- -----

SELECT nombre, `fecha de nacimiento`
FROM autor;
```

	nombre	fecha de nacimiento
►	Juan García	01-01-1900
	Pedro Martínez	02-02-1920
	Marta Rodríguez	03-03-1930
	Jorge López	04-04-1940
	Isabel Pérez	05-05-1950

La cantidad de libros diferentes vendidos.

## Jessica Andrea López Obando Actividad 1, 2 y 3 (Librería y Hospital)

```
9  -----
10 -- La cantidad de libros diferentes vendidos
11 -----
12 • SELECT libro.titulo, COUNT(DISTINCT libro_cliente.ISBN_libro_cliente) as Total_de_libros_vendidos
13 FROM libro_cliente
14 JOIN libro
15 ON libro_cliente.ISBN_libro_cliente = libro.ISBN
16 GROUP BY libro.titulo;
17
18 -----
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: [IA](#)

titulo	Total_de_libros_vendidos
Autoayuda para tiempos difíciles	1
Cien años de soledad	1
El amor en los tiempos del cólera	1
El poder de la positividad	1
La mala hora	1
La vida después del éxito	1
Literatura clásica: Don Quijote	1
Programación para principiantes	1
Python: Desde cero	1
SQL Básico	1

El nombre de su cliente acompañado de su número telefónico.

```
18 -----
19 -- El nombre de su cliente acompañado de su numero telefónico
20 -----
21 • SELECT cliente.nombre, telefono_cliente.numero
22 FROM cliente
23 JOIN telefono_cliente
24 ON cliente.cedula = telefono_cliente.cedula_cliente;
25
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: [IA](#)

nombre	numero
Carlos Gonzalez	555-555-1003
Carlos Gonzalez	555-555-1010
Juan Perez	555-555-1001
Juan Perez	555-555-1008
Sofia Martinez	555-555-1004
Sofia Martinez	555-555-1011
Luis Lopez	555-555-1005
Luis Lopez	555-555-1012
Ana Garcia	555-555-1006
Ana Garcia	555-555-1013
Pedro Sánchez	555-555-1007
María Rodriguez	555-555-1002
María Rodriguez	555-555-1009

El nombre del libro acompañado por su autor o sus autores.

```
26 -----
27 -- El nombre del libro acompañado por su autor o sus autores
28 -----
29 • SELECT libro.titulo, autor.nombre
30 FROM libro
31 JOIN libro_autor ON libro.ISBN = libro_autor.ISBN_libro
32 JOIN autor ON libro_autor.id_autor = autor.id;
33
34 -----
```

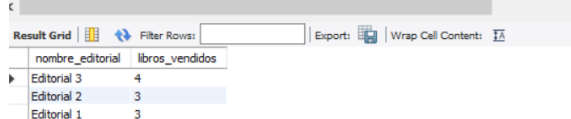
Result Grid | Filter Rows: | Export: | Wrap Cell Content: [IA](#)

titulo	nombre
Programación para principiantes	Juan García
Cien años de soledad	Juan García
Autoayuda para tiempos difíciles	Pedro Martínez
Python: Desde cero	Pedro Martínez
Literatura clásica: Don Quijote	Marta Rodríguez
Superar miedos y ansiedades	Marta Rodríguez
JavaScript: Avanzado	Jorge López
El amor en los tiempos del cólera	Jorge López
El poder de la positividad	Isabel Pérez
CSS Avanzado	Isabel Pérez

El nombre de las editoriales que han logrado vender libros.

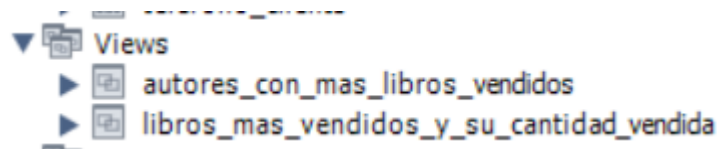
## Jessica Andrea López Obando Actividad 1, 2 y 3 (Librería y Hospital)

```
34 -----
35 -- El nombre de las editoriales que han logrado vender libros.
36 -----
37 • SELECT nombre_editorial, count(*) as libros_vendidos
38 FROM libro
39 JOIN libro_cliente ON libro.ISBN = libro_cliente.ISBN_libro_cliente
40 GROUP BY nombre_editorial;
41
42 -----
```



nombre_editorial	libros_vendidos
Editorial 3	4
Editorial 2	3
Editorial 1	3

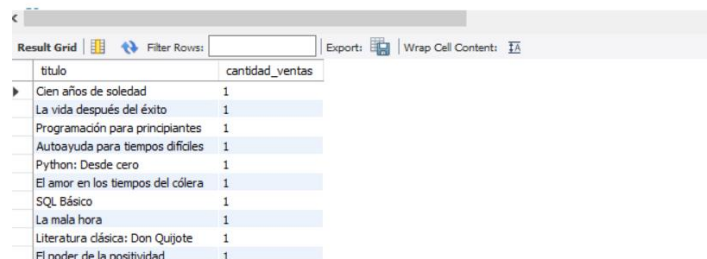
Realice las dos vistas que considere sean las más importantes y explique el motivo de su selección.



### **Consulta para conocer los libros más vendidos, junto con su cantidad de ventas**

Esta consulta es importante porque permite tener una visión general de cuáles son los libros más populares entre los clientes y cuántos ejemplares se han vendido de cada uno. Esto puede ayudar a la librería a tomar decisiones sobre qué libros ordenar en mayores cantidades, qué autores promocionar, y cuáles son las tendencias actuales en cuanto a los temas y géneros más vendidos.

```
42 -----
43 -- Consulta para conocer los libros más vendidos, junto con su cantidad de ventas
44 -----
45 • CREATE VIEW libros_mas_vendidos_y_su_cantidad_vendida AS
46 SELECT libro.titulo, COUNT(libro_cliente.ISBN_libro_cliente) AS cantidad_ventas
47 FROM libro
48 JOIN libro_cliente
49 ON libro.ISBN = libro_cliente.ISBN_libro_cliente
50 GROUP BY libro.titulo
51 ORDER BY cantidad_ventas DESC;
52
```



titulo	cantidad_ventas
Cien años de soledad	1
La vida después del éxito	1
Programación para principiantes	1
Autoayuda para tiempos difíciles	1
Python: Desde cero	1
El amor en los tiempos del cólera	1
SQL Básico	1
La mala hora	1
Literatura clásica: Don Quijote	1
El poder de la positividad	1

### **Consulta para conocer los autores con más libros vendidos**

Esta consulta es importante porque permite conocer a los autores más populares entre los clientes, lo que podría ayudar a la editorial a tomar decisiones sobre futuros lanzamientos de libros, marketing y promoción. También podría permitir identificar a

## Jessica Andrea López Obando Actividad 1, 2 y 3 (Librería y Hospital)

los autores que están teniendo más éxito en términos de ventas, lo que podría ayudar a la editorial a establecer relaciones más estrechas con ellos y a promover su trabajo de manera más efectiva. Por último, también podría ayudar a la editorial a evaluar su estrategia de publicación y a determinar si están publicando el tipo de libros que los clientes desean comprar y leer.

```
53  -----
54  -- Consulta para conocer los autores con más libros vendidos
55  -----
56  • CREATE VIEW autores_con_mas_libros_vendidos AS
57  SELECT autor.nombre AS nombre_autor, COUNT(libro.ISBN) AS cantidad_ventas
58  FROM autor
59  JOIN libro_autor ON autor.id = libro_autor.id_autor
60  JOIN libro ON libro_autor.ISBN_libro = libro.ISBN
61  JOIN libro_cliente ON libro.ISBN = libro_cliente.ISBN_libro_cliente
62  GROUP BY autor.nombre
63  ORDER BY cantidad_ventas DESC;
64
```

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

nombre_autor	cantidad_ventas
Juan García	2
Pedro Martínez	2
Marta Rodríguez	1
Jorge López	1
Isabel Pérez	1

**¿Qué le agregaría al modelo para dar más información y esa información cuál sería?**

Añadiría una tabla de géneros literarios, para categorizar los libros en diferentes géneros (ciencia ficción, romance, suspenso, terror).

En la tabla de libro, añadiría un atributo multivaluado idiomas, que se convertiría en una tabla para especificar los idiomas disponibles para cada libro.

## SEGUNDA ACTIVIDAD:

<https://github.com/Ococho/Hospital-GNECJ.git>

Utilizando el ejercicio del hospital realizado por sus compañeros realice lo siguiente:

- Convierta el MR en una base de datos en MySQL utilizando sentencias SQL o el diagrama EER.

Utilizando el MR en el pdf del compañero, se realiza el diagrama EER en Workbench.

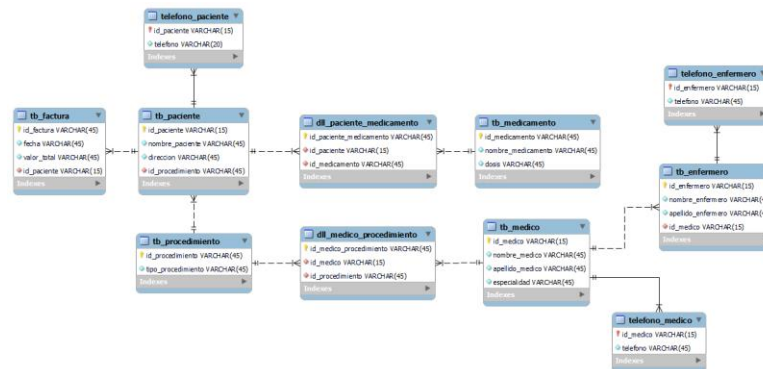


Diagrama EER

Se convierte en base de datos a través del EER realizando un Forward Engineering. Archivo que se adjunta con el nombre de *Forward Engineer Hospital* en la carpeta Segunda Actividad.

```
1 -- MySQL Workbench Forward Engineering
2
3 SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;
4 SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0;
5 SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLES,NO_ZERO_IN_DATE,NO_ZERO_DATE,ERROR_FOR_DIVISION_BY_ZERO,NO_ENGINE_SUBSTITUTION';
6
7
8 -- Schema Hospital
9
10
11
12 -- Schema Hospital
13
14 CREATE SCHEMA IF NOT EXISTS "hospital" DEFAULT CHARACTER SET utf8 ;
15 USE "hospital" ;
16
17
18 -- Table "hospital"."tb_procedimiento"
19
20 CREATE TABLE IF NOT EXISTS "hospital"."tb_procedimiento" (
21   "id_procedimiento" VARCHAR(45) NOT NULL,
22   "tipo_procedimiento" VARCHAR(45) NOT NULL,
23   PRIMARY KEY ("id_procedimiento"))
24 ENGINE = InnoDB;
25
```

- Complete la información para las tablas realizadas con al menos 5 registros por tabla.



## Jessica Andrea López Obando Actividad 1, 2 y 3 (Librería y Hospital)



*Tablas a realizarle los registros*

### dll medico procedimiento

```
86
87 • SELECT * FROM dll_medico_procedimiento;
88
89 • INSERT INTO dll_medico_procedimiento (id_medico_procedimiento, id_medico, id_procedimiento)
90 VALUES
91 ("medpro001", "M1", "pro1"),
92 ("medpro002", "M2", "pro2"),
93 ("medpro003", "M3", "pro3"),
94 ("medpro004", "M4", "pro4"),
95 ("medpro005", "M5", "pro5");
96
```

id_medico_procedimiento	id_medico	id_procedimiento
medpro001	M1	pro1
medpro002	M2	pro2
medpro003	M3	pro3
medpro004	M4	pro4
medpro005	M5	pro5
NULL	NULL	NULL

### dll paciente medicamento

```
97 • SELECT * FROM dll_paciente_medicamento;
98
99 • INSERT INTO dll_paciente_medicamento (id_paciente_medicamento, id_paciente, id_medimento)
100 VALUES
101 ("pamed001", "pac001", "med1"),
102 ("pamed002", "pac002", "med2"),
103 ("pamed003", "pac003", "med3"),
104 ("pamed004", "pac004", "med4"),
105 ("pamed005", "pac005", "med5");
106
```

id_paciente_medicamento	id_paciente	id_medimento
pamed001	pac001	med1
pamed002	pac002	med2
pamed003	pac003	med3
pamed004	pac004	med4
pamed005	pac005	med5
NULL	NULL	NULL

### tb enfermero

## Jessica Andrea López Obando Actividad 1, 2 y 3 (Librería y Hospital)

```
24 • SELECT * FROM tb_enfermero;
25
26 • INSERT INTO tb_enfermero (id_enfermero, nombre_enfermero, apellido_enfermero, id_medico)
27 VALUES
28 ("E1", "Liliana", "Ruiz Lopez", "M1"),
29 ("E2", "María", "Gonzalez Perez", "M2"),
30 ("E3", "Carlos", "Perez Herrera", "M3"),
31 ("E4", "Ana", "Hernandez Santos", "M4"),
32 ("E5", "Juan", "Rodriguez Garcia", "M5");
33
34 • SELECT * FROM telefono_enfermero;
```

Result Grid

id_enfermero	nombre_enfermero	apellido_enfermero	id_medico
E1	Liliana	Ruiz Lopez	M1
E2	María	Gonzalez Perez	M2
E3	Carlos	Perez Herrera	M3
E4	Ana	Hernandez Santos	M4
E5	Juan	Rodriguez Garcia	M5
NULL	NULL	NULL	NULL

### tb factura

```
107 • SELECT * FROM tb_factura;
108
109 • INSERT INTO tb_factura (id_factura, fecha, valor_total, id_paciente)
110 VALUES
111 ("1", "2023-02-13", "250.000", "pac001"),
112 ("2", "2023-02-14", "117.000", "pac002"),
113 ("3", "2023-02-15", "300.000", "pac003"),
114 ("4", "2023-02-16", "1.500.000", "pac004"),
115 ("5", "2023-02-17", "480.000", "pac005");
116
```

Result Grid

id_factura	fecha	valor_total	id_paciente
1	2023-02-13	250.000	pac001
2	2023-02-14	117.000	pac002
3	2023-02-15	300.000	pac003
4	2023-02-16	1.500.000	pac004
5	2023-02-17	480.000	pac005
NULL	NULL	NULL	NULL

### tb medicamento

```
45 • SELECT * FROM tb_medimento;
46
47 • INSERT INTO tb_medimento (id_medimento, nombre_medimento, dosis)
48 VALUES
49 ("med1", "Nitroglicerina", "1 comprimido sublingual cada 30 minutos"),
50 ("med2", "Acetaminofén", "2 comprimidos cada 6 horas"),
51 ("med3", "Betametasona", "1 crema aplicada en la piel cada 12 horas"),
52 ("med4", "Metformina", "1 comprimido cada 8 horas"),
53 ("med5", "Clonazepam", "1 comprimido cada 8 horas");
54
55
```

Result Grid

id_medimento	nombre_medimento	dosis
med1	Nitroglicerina	1 comprimido sublingual cada 30 minutos
med2	Acetaminofén	2 comprimidos cada 6 horas
med3	Betametasona	1 crema aplicada en la piel cada 12 horas
med4	Metformina	1 comprimido cada 8 horas
med5	Clonazepam	1 comprimido cada 8 horas
NULL	NULL	NULL

### tb medico

## Jessica Andrea López Obando Actividad 1, 2 y 3 (Librería y Hospital)

Limit to 1000 rows

```
1 • USE hospital;
2
3 • SELECT * FROM tb_medico;
4
5 • INSERT INTO tb_medico (id_medico, nombre_medico, apellido_medico, especialidad)
6 VALUES
7 ("M1", "Santiago", "García", "Cardiología"),
8 ("M2", "Valentina", "Rodríguez", "Pediatria"),
9 ("M3", "Mariela", "Molina", "Dermatología"),
10 ("M4", "Andrea", "López", "Ginecología"),
11 ("M5", "Carlos", "Martínez", "Neurología");
```

Result Grid

	id_medico	nombre_medico	apellido_medico	especialidad
▶	M1	Santiago	García	Cardiología
	M2	Valentina	Rodríguez	Pediatría
	M3	Mariela	Molina	Dermatología
	M4	Andrea	López	Ginecología
	M5	Carlos	Martínez	Neurología
*	NULL	NULL	NULL	NULL

### tb\_paciente

```
65
66 • SELECT * FROM tb_paciente;
67
68 • INSERT INTO tb_paciente (id_paciente, nombre_paciente, direccion, id_procedimiento)
69 VALUES
70 ("pac001", "Luz Adriana Rodriguez", "Calle 123", "pro1"),
71 ("pac002", "Juan Manuel Sánchez", "Calle 456", "pro2"),
72 ("pac003", "María Cristina García", "Calle 789", "pro3"),
73 ("pac004", "Sofía Elena Díaz", "Calle 246", "pro4"),
74 ("pac005", "Andrés Felipe Ortiz", "Calle 369", "pro5");
75
```

Result Grid

	id_paciente	nombre_paciente	direccion	id_procedimiento
	pac001	Luz Adriana Rodriguez	Calle 123	pro1
	pac002	Juan Manuel Sánchez	Calle 456	pro2
	pac003	María Cristina García	Calle 789	pro3
	pac004	Sofía Elena Díaz	Calle 246	pro4
	pac005	Andrés Felipe Ortiz	Calle 369	pro5
*	NULL	NULL	NULL	NULL

### tb\_procedimiento

```
55
56 • SELECT * FROM tb_procedimiento;
57
58 • INSERT INTO tb_procedimiento (id_procedimiento, tipo_procedimiento)
59 VALUES
60 ("pro1", "Cateterismo Cardíaco"),
61 ("pro2", "Vacunación infantil"),
62 ("pro3", "Biopsia de piel"),
63 ("pro4", "Citología cervical"),
64 ("pro5", "Estudio de EEG");
65
```

Result Grid

	id_procedimiento	tipo_procedimiento
▶	pro1	Cateterismo Cardíaco
	pro2	Vacunación infantil
	pro3	Biopsia de piel
	pro4	Citología cervical
	pro5	Estudio de EEG
*	NULL	NULL

### teléfono enfermero

## Jessica Andrea López Obando Actividad 1, 2 y 3 (Librería y Hospital)

```
33
34 • SELECT * FROM telefono_enfermero;
35
36 • INSERT INTO telefono_enfermero (id_enfermero, telefono)
37 VALUES
38 ("E1", "3171234567"),
39 ("E2", "3181234567"),
40 ("E3", "3191234567"),
41 ("E4", "3161234567"),
42 ("E5", "3151234567");
43
```

Result Grid | Filter Rows: | Edit: | Export/Imp

id_enfermero	telefono
E1	3171234567
E2	3181234567
E3	3191234567
E4	3161234567
E5	3151234567
NULL	NULL

### teléfono medico

```
12
13 • SELECT * FROM telefono_medico;
14
15 • INSERT INTO telefono_medico (id_medico, telefono)
16 VALUES
17 ("M1", "3231214500"),
18 ("M2", "3149876543"),
19 ("M3", "3121098765"),
20 ("M4", "3214120456"),
21 ("M5", "6067236548");
22
```

Result Grid | Filter Rows: | Edit: | Exp

id_medico	telefono
M1	3231214500
M2	3149876543
M3	3121098765
M4	3214120456
M5	6067236548
NULL	NULL

### teléfono paciente

```
75
76 • SELECT * FROM telefono_paciente;
77
78 • INSERT INTO telefono_paciente (id_paciente, telefono)
79 VALUES
80 ("pac001", "3241234567"),
81 ("pac002", "3145670001"),
82 ("pac003", "3110114253"),
83 ("pac004", "3201234567"),
84 ("pac005", "3122434055");
85
86
```

Result Grid | Filter Rows: | Edit: | Export/Impor

id_paciente	telefono
pac001	3241234567
pac002	3145670001
pac003	3110114253
pac004	3201234567
pac005	3122434055
NULL	NULL

- Realice una consulta que me permita conocer que medicamentos ha tomado cada paciente y la dosis suministrada.

```

3  -----
4  -- Consulta que me permite conocer que medicamentos ha tomado cada paciente y la dosis suministrada.
5  -----
6  • SELECT
7  tb_paciente.nombre_paciente,
8  tb_medimento.nombre_medimento,
9  tb_medimento.dosis
10 FROM tb_paciente
11 JOIN dll_paciente_medimento ON tb_paciente.id_paciente = dll_paciente_medimento.id_paciente
12 JOIN tb_medimento ON dll_paciente_medimento.id_medimento = tb_medimento.id_medimento;
13
14 -----

```

nombre_paciente	nombre_medimento	dosis
Luz Adriana Rodriguez	Nitroglicerina	1 comprimido sublingual cada 30 minutos
Juan Manuel Sánchez	Acetaminofén	2 comprimidos cada 6 horas
María Cristina García	Betametasona	1 crema aplicada en la piel cada 12 horas
Sofía Elena Díaz	Metformina	1 comprimido cada 8 horas
Andrés Felipe Ortiz	Clonazepam	1 comprimido cada 8 horas

- Realice una consulta que me permita conocer que enfermeros estuvieron en los procedimientos de los pacientes.

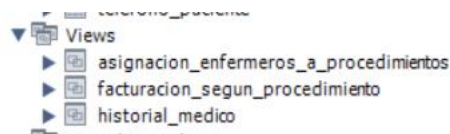
```

14 -----
15 -- Consulta que me permita conocer que enfermeros estuvieron en los procedimientos de los pacientes.
16 -----
17 • CREATE VIEW asignacion_enfermeros_a_procedimientos AS
18 SELECT
19 tb_paciente.nombre_paciente AS 'Paciente',
20 tb_procedimiento.tipo_procedimiento AS 'Procedimiento',
21 tb_enfermero.nombre_enfermero AS 'Enfermero asignado'
22 FROM
23 tb_paciente
24 JOIN tb_procedimiento ON tb_paciente.id_procedimiento = tb_procedimiento.id_procedimiento
25 JOIN dll_medico_procedimiento ON tb_procedimiento.id_procedimiento = dll_medico_procedimiento.id_procedimiento
26 JOIN tb_medico ON dll_medico_procedimiento.id_medico = tb_medico.id_medico
27 JOIN tb_enfermero ON tb_enfermero.id_medico = tb_medico.id_medico;
28
29 -----

```

Paciente	Procedimiento	Enfermero asignado
Luz Adriana Rodriguez	Cateterismo Cardíaco	Liliana
Juan Manuel Sánchez	Vacunación infantil	María
María Cristina García	Biopsia de piel	Carlos
Sofía Elena Díaz	Citología cervical	Ana
Andrés Felipe Ortiz	Estudio de EEG	Juan

- Realice las tres vistas que considere sean las más importantes y explique el motivo de su selección.



La primera vista que considero importante es la de consultar que enfermeros estuvieron en los procedimientos de los pacientes, esto se debe a varias razones:

- a. **Mejora la transparencia:** Al tener un registro de los enfermeros que estuvieron presentes en cada procedimiento, es posible tener una visión más clara y detallada de los procedimientos realizados.
- b. **Facilita la gestión:** Con una vista de este tipo, es más fácil para los responsables de la gestión del personal médico tener un registro de los enfermeros que estuvieron en cada procedimiento y, por lo tanto, poder tomar decisiones informadas en cuanto a la asignación de tareas y responsabilidades.
- c. **Mejora la calidad de atención:** Conocer la presencia de enfermeros en cada procedimiento permite identificar y corregir posibles desviaciones en la calidad de atención, lo que puede ser de gran importancia para la seguridad y satisfacción de los pacientes.

```
-----  
-- Consulta que me permita conocer que enfermeros estuvieron en los procedimientos de los pacientes.  
-----  
CREATE VIEW asignacion_enfermeros_a_procedimientos AS
```

La segunda vista que considero importante es la del Historial médico de los pacientes por varias razones:

- a. **Facilita el acceso a la información:** Con una vista unificada, los profesionales médicos pueden tener una visión completa y organizada de los procedimientos, medicamentos y dosis que se han administrado a un paciente en un momento dado, lo que les ayuda a tomar decisiones informadas.
- b. **Ayuda a la coordinación de cuidados:** La vista del historial médico permite a los profesionales médicos ver la información completa de un paciente en un solo lugar, lo que les permite coordinar mejor los cuidados entre diferentes especialistas.
- c. **Reduce la posibilidad de errores médicos:** Con una vista unificada, los profesionales médicos pueden ver la información completa de un paciente en un solo lugar, lo que reduce la posibilidad de errores médicos causados por la duplicación o la omisión de información.

## Jessica Andrea López Obando Actividad 1, 2 y 3 (Librería y Hospital)

```
29  -----
30  -- Historial medico de los pacientes
31  -----
32  • CREATE VIEW Historial_Medico AS
33      SELECT
34          tb_paciente.nombre_paciente AS 'Paciente',
35          tb_procedimiento.tipo_procedimiento AS 'Procedimiento',
36          tb_medimento.nombre_medimento AS 'Medicamento',
37          tb_medimento.dosis AS 'Dosis'
38      FROM
39          tb_paciente
40      JOIN tb_procedimiento ON tb_paciente.id_procedimiento = tb_procedimiento.id_procedimiento
41      JOIN dll_paciente_medimento ON tb_paciente.id_paciente = dll_paciente_medimento.id_paciente
42      JOIN tb_medimento ON dll_paciente_medimento.id_medimento = tb_medimento.id_medimento;
43
44  -----
```

Paciente	Procedimiento	Medicamento	Dosis
Luz Adriana Rodriguez	Cateterismo Cardiaco	Nitroglicerina	1 comprimido sublingual cada 30 minutos
Juan Manuel Sánchez	Vacunación infantil	Acetaminofén	2 comprimidos cada 6 horas
María Cristina García	Biopsia de piel	Betametasona	1 crema aplicada en la piel cada 12 horas
Sofía Elena Díaz	Citología cervical	Metformina	1 comprimido cada 8 horas
Andrés Felipe Ortiz	Estudio de EEG	Clonazepam	1 comprimido cada 8 horas

La tercera vista que considero importante es el valor de facturación según procedimiento.

Es importante tener una vista de valor de facturación según procedimiento porque proporciona una visión general y clara de los ingresos generados por cada procedimiento, lo que permite a los responsables de la administración tomar decisiones informadas sobre la gestión de costos y el desarrollo de estrategias de marketing. Además, esta vista permite identificar rápidamente cuales son los procedimientos más rentables y por lo tanto, se pueden enfocar más recursos en ellos. Además, también proporciona información valiosa para los pacientes, permitiéndoles conocer el costo de sus procedimientos y tener una mejor comprensión de su historial médico financiero.

```
44  -----
45  -- valor Facturacion segun procedimiento
46  -----
47  • CREATE VIEW Facturacion_segund_procedimiento AS
48      SELECT
49          tb_paciente.nombre_paciente AS 'Paciente',
50          telefono_paciente.telefono AS 'Teléfono',
51          tb_procedimiento.tipo_procedimiento AS 'Procedimiento',
52          tb_factura.valor_total AS 'Factura'
53      FROM
54          tb_paciente
55      JOIN tb_procedimiento ON tb_paciente.id_procedimiento = tb_procedimiento.id_procedimiento
56      JOIN tb_factura ON tb_paciente.id_paciente = tb_factura.id_paciente
57      JOIN telefono_paciente ON tb_paciente.id_paciente = telefono_paciente.id_paciente;
58
59  -----
```

Paciente	Teléfono	Procedimiento	Factura
Luz Adriana Rodriguez	3241234567	Cateterismo Cardiaco	250.000
Juan Manuel Sánchez	3145670001	Vacunación infantil	117.000
María Cristina García	3110114253	Biopsia de piel	300.000
Sofía Elena Díaz	3201234567	Citología cervical	1.500.000
Andrés Felipe Ortiz	3122434055	Estudio de EEG	480.000

**¿Qué le agregaría al modelo para dar más información y esa información cuál sería?**

Añadiría campos de información algunas tablas, como en médicos y enfermero añadiría atributos como los horarios de trabajo de enfermeros y médicos para tener un registro de cuándo están disponibles para atender pacientes.

Añadiría una tabla citas médicas, para llevar un control de los pacientes que están por ser atendidos, tendría los siguientes atributos fecha\_cita, hora\_cita, motivo de la consulta, diagnóstico, estado, que podría ser en espera, completado, o en proceso.

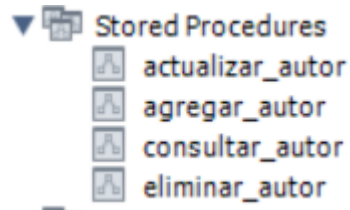
Añadiría una tabla de insumos médicos para registrar los insumos médicos utilizados durante los procedimientos, sería un atributo multivaluado de la tabla tb\_procedimientos, así que tendría el id\_procedimiento y la descripción de los insumos utilizados.

Añadiría una tabla de antecedentes familiares, para registrar los antecedentes familiares de enfermedades de los pacientes y tener en cuenta esta información durante su tratamiento.



### TERCERA ACTIVIDAD (Taller 6)

- Elabore 4 procedimientos almacenados que me permitan agregar, actualizar, consultar y borrar, en una de las tablas de la librería (primera actividad).
- Elegí la tabla autor para realizar los procedimientos.



#### Agregar

```
5  -- -----
6  -- Agregar Autor
7  -- -----
8  DELIMITER //
9  CREATE PROCEDURE agregar_autor(IN id_autor VARCHAR(10), fecha_nacimiento VARCHAR(45), IN nacionalidad_autor VARCHAR(20), IN nombre_autor
10 BEGIN
11 INSERT INTO autor (id, `fecha de nacimiento`, nacionalidad, nombre) VALUES (id_autor, fecha_nacimiento, nacionalidad_autor, nombre_autor);
12 END
13 //
14
15 call libreriaescalibre.agregar_autor('A006', '14-06-1951', 'Italia', 'Walter riso');
16 call libreriaescalibre.agregar_autor('A007', '07-10-1662', 'Alemania', 'Eckhart Tolle');
17
18 -- -----
19 -- Actualizar Autor
```

result Grid			
Filter Rows:			
Edit: Export/Import: Wrap Cell Content: I			
id	fecha de nacimiento	nacionalidad	nombre
A001	02-02-1961	Español	Mario Alonso Puig
A002	02-02-1920	Mexicano	Pedro Martínez
A003	03-03-1930	Argentino	Marta Rodríguez
A004	04-04-1940	Colombiano	Jorge López
A005	05-05-1950	Peruano	Isabel Pérez
A006	14-06-1951	Italia	Walter riso
A007	07-10-1662	Alemania	Eckhart Tolle
A008	21-03-1952	Holanda	Benjamin
NULL	NULL	NULL	NULL

#### Actualizar

```
18  -- -----
19  -- Actualizar Autor
20  -- -----
21  DELIMITER //
22  CREATE PROCEDURE actualizar_autor (IN id_autor VARCHAR(10), fecha_nacimiento VARCHAR(45), IN nacionalidad_autor VARCHAR(20), IN nombre_autor
23 BEGIN
24 UPDATE autor SET nombre = nombre_autor, `fecha de nacimiento` = fecha_nacimiento, nacionalidad = nacionalidad_autor WHERE id = id_autor;
25 END;
26 //
27
28 call libreriaescalibre.actualizar_autor('A001', '02-02-1961', 'Español', 'Mario Alonso Puig');
29
30 -- -----
```

result Grid			
Filter Rows:			
Edit: Export/Import: Wrap Cell Content: I			
id	fecha de nacimiento	nacionalidad	nombre
A001	02-02-1961	Español	Mario Alonso Puig
A002	02-02-1920	Mexicano	Pedro Martínez
A003	03-03-1930	Argentino	Marta Rodríguez
A004	04-04-1940	Colombiano	Jorge López
A005	05-05-1950	Peruano	Isabel Pérez
A006	14-06-1951	Italia	Walter riso
A007	07-10-1662	Alemania	Eckhart Tolle
A008	21-03-1952	Holanda	Benjamin
NULL	NULL	NULL	NULL

#### Consultar

## Jessica Andrea López Obando Actividad 1, 2 y 3 (Librería y Hospital)

```
30  -----
31  -- Consultar Autor
32  -----
33  DELIMITER //
34  ● CREATE PROCEDURE consultar_autor (IN id_autor VARCHAR(10))
35  BEGIN
36  SELECT * FROM autor WHERE id = id_autor;
37  END;
38  //
39
40  call libreriaescalibre.consultar_autor('A002');
```

Result Grid

id	fecha de nacimiento	nacionalidad	nombre
A002	02-02-1920	Mexicano	Pedro Martínez

### Eliminar

```
42  -----
43  -- Eliminar Autor
44  -----
45  DELIMITER //
46  ● CREATE PROCEDURE eliminar_autor(IN id_autor VARCHAR(10))
47  BEGIN
48  DELETE FROM autor WHERE id = id_autor;
49  END;
50  //
51
52  ● call libreriaescalibre.eliminar_autor('A006');
53  call libreriaescalibre.eliminar_autor('A007');
54
55
```

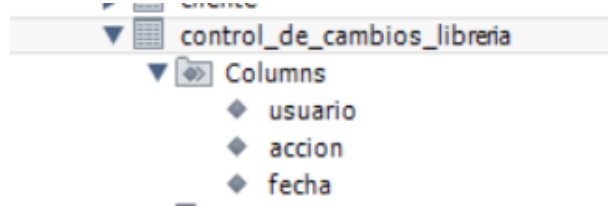
Result Grid

id	fecha de nacimiento	nacionalidad	nombre
A001	02-02-1961	Español	Mario Alonso Puig
A002	02-02-1920	Mexicano	Pedro Martínez
A003	03-03-1930	Argentino	Marta Rodríguez
A004	04-04-1940	Colombiano	Jorge López
A005	05-05-1950	Peruano	Isabel Pérez
A008	21-03-1952	Holanda	Benjamin

- Elabore una nueva tabla llamada "control\_de\_cambios\_librería" la cual debe contener 3 columnas (usuario, accion, fecha) y guarde utilizando 2 Triggers el nombre del usuario que agrego o elimino un registro en la tabla seleccionada en el punto anterior.

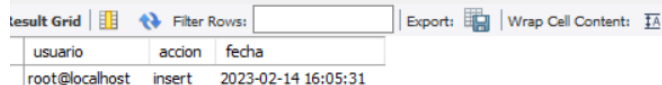
## Jessica Andrea López Obando Actividad 1, 2 y 3 (Librería y Hospital)

```
3  -----
4  -- Creo la tabla control de cambios libreria
5  -----
6  ● CREATE TABLE control_de_cambios_libreria (
7      usuario VARCHAR(45),
8      accion VARCHAR(20),
9      fecha DATETIME
10 );
11
```



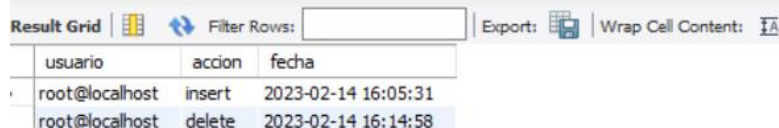
control_de_cambios_libreria		
Columns		
usuario		
accion		
fecha		

```
12 -----
13 -- Triggers Insertar
14 -----
15 DELIMITER //
16 ● CREATE TRIGGER control_libreria_insert
17 AFTER INSERT ON autor
18 FOR EACH ROW
19 BEGIN
20     INSERT INTO control_de_cambios_libreria (usuario, accion, fecha)
21     VALUES (user(), 'insert', NOW());
22 END;
23 //
24 call libreriaescalibre.agregar_autor('A0011', '09-07-1955', 'Africa', 'Angs');
25
```



usuario	accion	fecha
root@localhost	insert	2023-02-14 16:05:31

```
26 -----
27 -- Trigger Eliminar
28 -----
29 DELIMITER //
30 ● CREATE TRIGGER control_libreria_delete
31 AFTER DELETE ON autor
32 FOR EACH ROW
33 BEGIN
34     INSERT INTO control_de_cambios_libreria (usuario, accion, fecha)
35     VALUES (user(), 'delete', NOW());
36 END;
37 //
38
39 ● call libreriaescalibre.eliminar_autor('A0011');
```

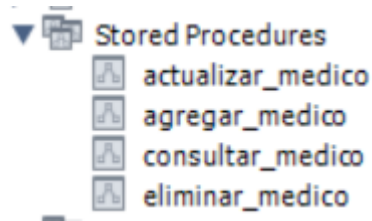


usuario	accion	fecha
root@localhost	insert	2023-02-14 16:05:31
root@localhost	delete	2023-02-14 16:14:58

## Jessica Andrea López Obando Actividad 1, 2 y 3 (Librería y Hospital)

- Elabore 4 procedimientos almacenados que me permitan agregar, actualizar, consultar y borrar, en una de las tablas del Hospital (segunda actividad).

En hospital elegí la tabla tb\_medico para realizar los procedimientos.



### Agregar

```
7  -- Agregar Médico
8  -- -----
9  DELIMITER //
10 CREATE PROCEDURE agregar_medico(IN id VARCHAR(15), IN nombre VARCHAR(45), IN apellido VARCHAR(45), IN especialidad_medico VARCHAR(45))
11 BEGIN
12     INSERT INTO tb_medico (id_medico, nombre_medico, apellido_medico, especialidad) VALUES (id, nombre, apellido, especialidad_medico);
13 END
14 //
15
16 call hospital.agregar_medico('M7', 'Carla', 'Jimenez', 'Ginecología');
17 call hospital.agregar_medico('M8', 'Isabela', 'Torres', 'Cardiología');
18
19 -- -----
```

id_medico	nombre_medico	apellido_medico	especialidad
M1	Santiago	García	Cardiología
M2	Valentina	Rodríguez	Pediatría
M3	Mariela	Molina	Dermatología
M4	Andrea	López	Ginecología
M5	Jimmy	Martínez	Neurología
M6	Mauricio	Arias	Endocrinología
M7	Carla	Jimenez	Ginecología
M8	Isabela	Torres	Cardiología

### Actualizar

```
19  -- Actualizar Médico
20  -- -----
21  DELIMITER //
22 CREATE PROCEDURE actualizar_medico(IN id VARCHAR(15), IN nombre VARCHAR(45), IN apellido VARCHAR(45), IN especialidad_medico VARCHAR(45))
23 BEGIN
24     UPDATE tb_medico SET nombre_medico = nombre, apellido_medico = apellido, especialidad = especialidad_medico WHERE id_medico = id;
25 END;
26 //
27
28 call hospital.actualizar_medico('M5', 'Jimmy', 'Sanchez', 'Neurología');
29
30 -- -----
31
```

id_medico	nombre_medico	apellido_medico	especialidad
M1	Santiago	García	Cardiología
M2	Valentina	Rodríguez	Pediatría
M3	Mariela	Molina	Dermatología
M4	Andrea	Lopez	Ginecología
M5	Jimmy	Sanchez	Neurología
M6	Mauricio	Arias	Endocrinología
M7	Carla	Jimenez	Ginecología
M8	Isabela	Torres	Cardiología

### Consultar

## Jessica Andrea López Obando Actividad 1, 2 y 3 (Librería y Hospital)

```
31  -- -----
32  -- Consultar Medico
33  -- -----
34  DELIMITER //
35  ● CREATE PROCEDURE consultar_medico(IN id VARCHAR(15))
36  BEGIN
37    SELECT * FROM tb_medico WHERE id_medico = id;
38  END;
39  //
40
41  call hospital.consultar_medico('M3');
42
43  -- -----
```

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

id_medico	nombre_medico	apellido_medico	especialidad
M3	Mariela	Molina	Dermatología

### Eliminar

```
43  -- -----
44  -- Eliminar Medico
45  -- -----
46  DELIMITER //
47  ● CREATE PROCEDURE eliminar_medico (IN id VARCHAR(15))
48  BEGIN
49    DELETE FROM tb_medico
50    WHERE id_medico = id;
51  END//
52
53  ● call hospital.eliminar_medico('M7');
```

Result Grid | Filter Rows: | Edit: | Exp

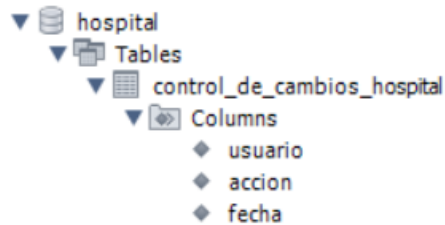
id_medico	nombre_medico	apellido_medico	especialidad
M1	Santiago	García	Cardiología
M2	Valentina	Rodríguez	Pediatría
M3	Mariela	Molina	Dermatología
M4	Andrea	López	Ginecología
M5	Jimmy	Sanchez	Neurología
M6	Mauricio	Arias	Endocrinología
M8	Isabela	Torres	Cardiología
NULL	NULL	NULL	NULL

- Elabore una nueva tabla llamada "control\_de\_cambios\_hospital" la cual debe contener 3 columnas (usuario, accion, fecha) y guarde utilizando 2 Triggers el nombre del usuario que agrego o elimino un registro en la tabla seleccionada en el punto anterior.

### Jessica Andrea López Obando Actividad 1, 2 y 3 (Librería y Hospital)

```
-- Creo la tabla control de cambios Hospital
```

```
CREATE TABLE control_de_cambios_hospital (
    usuario VARCHAR(45),
    accion VARCHAR(20),
    fecha DATETIME
);
```



```

12  -----
13  -- Triggers Insertar
14  -----
15  DELIMITER //
16  • CREATE TRIGGER control_hospital_insert
17  AFTER INSERT ON tb_medico
18  FOR EACH ROW
19  BEGIN
20  INSERT INTO control_de_cambios_hospital (usuario, accion, fecha)
21  VALUES (user(), 'insert', NOW());
22  END;
23  //
24
25  call hospital.agregar_medico('M9', 'Sandra Milena', 'Gonzalez', 'Ginecologia');
26





```

Result Grid				Filter Rows:	Export:	Wrap Cell Content:
	usuario	accion	fecha			
▶	root@localhost	insert	2023-02-14 16:46:19			

```

27 -----
28 -- Trigger Eliminar
29 -----
30 DELIMITER //
31 • CREATE TRIGGER control_hospital_delete
32 AFTER DELETE ON tb_medico
33 FOR EACH ROW
34 BEGIN
35 INSERT INTO control_de_cambios_hospital (usuario, accion, fecha)
36 VALUES (user(), 'delete', NOW());
37 END;
38 //
39
40 • call hospital.eliminar_medico('M9');
41

```

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

	usuario	accion	fecha
▶	root@localhost	insert	2023-02-14 16:46:19
	root@localhost	delete	2023-02-14 16:48:33