

Actividad #2 – hospital

Creación base de datos de hospital

```
CREATE DATABASE if not exists db_hospital;
```

✓ 54 22:44:32 CREATE DATABASE db_hospital

```
-- Indica cuál será la BD a la cual se le aplicarán las siguientes consultas:  
USE db_hospital;
```

Creación de una tabla

```
CREATE TABLE tb_procedimiento(  
  id_procedimiento VARCHAR(10) NOT NULL,  
  tipo_procedimiento VARCHAR(30) NOT NULL,  
  PRIMARY KEY(id_procedimiento)  
);
```

✓ 6 22:50:02 CREATE TABLE tb_procedimiento(id_procedimiento VARCHAR(10) NOT NULL, tipo_procedimiento VARCHAR(30) NOT NULL, PRIMARY KEY(id_proce...

Creación de una tabla con clave foránea o ajena (1:N)

```
CREATE TABLE tb_paciente(  
  id_paciente VARCHAR(10) NOT NULL,  
  nombre_paciente VARCHAR(40) NOT NULL,  
  apellido_paciente VARCHAR(40) NOT NULL,  
  direccion VARCHAR(30) NOT NULL,  
  id_procedimiento VARCHAR(10) NOT NULL,  
  PRIMARY KEY(id_paciente),  
  FOREIGN KEY(id_procedimiento) REFERENCES tb_procedimiento(id_procedimiento)  
);
```

✓ 9 22:50:02 CREATE TABLE tb_paciente(id_paciente VARCHAR(10) NOT NULL, nombre_paciente VARCHAR(40) NOT NULL, apellido_paciente VARCHAR(40) NO...

Creación de una tabla con una clave compuesta (1:N – en este caso es la representación de un atributo multivaluado)

```
CREATE TABLE tb_telefono_paciente (  
  id_paciente VARCHAR(10) NOT NULL,  
  telefono VARCHAR(15) NOT NULL,  
  PRIMARY KEY(id_paciente, telefono),  
  FOREIGN KEY(telefono) REFERENCES tb_paciente(id_paciente)  
);
```

✓ 10 22:50:02 CREATE TABLE tb_telefono_paciente (id_paciente VARCHAR(10) NOT NULL, telefono VARCHAR(15) NOT NULL, PRIMARY KEY(id_paciente, telefono...

Creación de una tabla detalle (M:N)

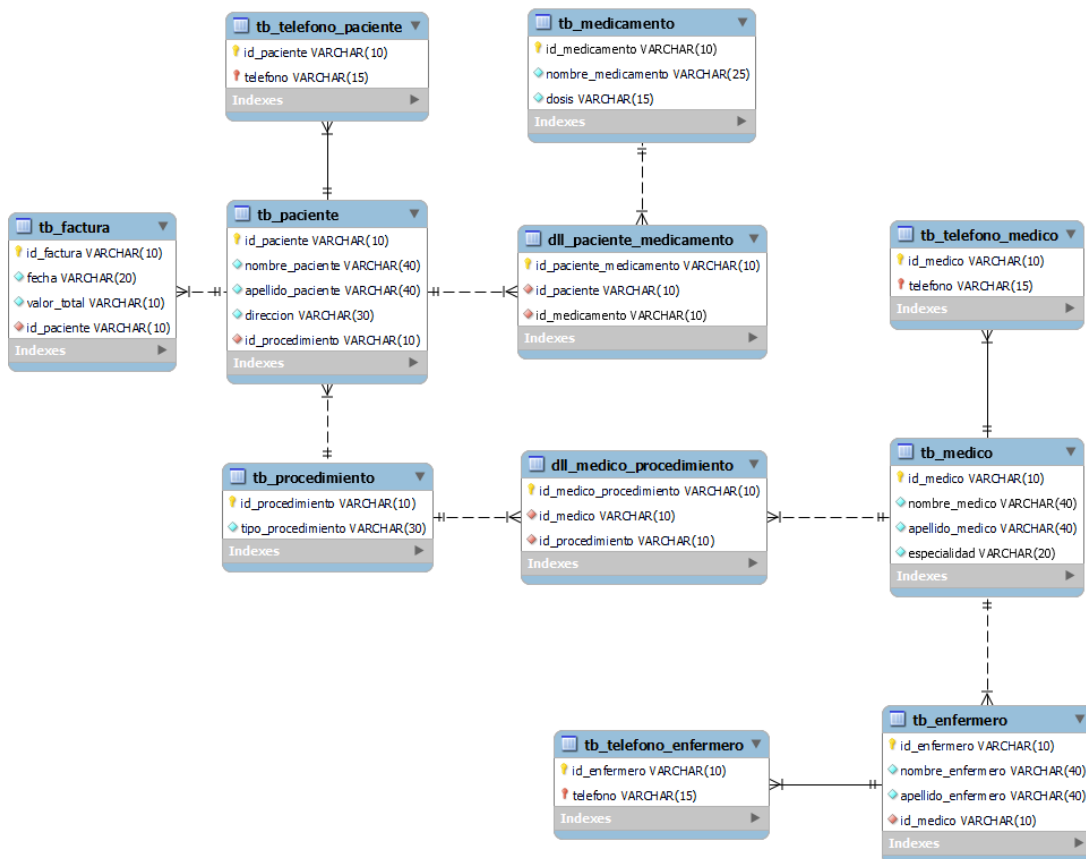
```
CREATE TABLE dll_paciente_medicamento (  
  id_paciente_medicamento VARCHAR(10) NOT NULL,  
  id_paciente VARCHAR(10) NOT NULL,  
  id_medicamento VARCHAR(10) NOT NULL,  
  PRIMARY KEY(id_paciente_medicamento),  
  FOREIGN KEY(id_paciente) REFERENCES tb_paciente(id_paciente),  
  FOREIGN KEY(id_medicamento) REFERENCES tb_medicamento(id_medicamento)  
);
```

✓ 14 22:50:02 CREATE TABLE dll_paciente_medicamento (id_paciente_medicamento VARCHAR(10) NOT NULL, id_paciente VARCHAR(10) NOT NULL, id_medicam...

Después de ejecutar el script anterior, se genera el diagrama EER.

Diagrama del modelo relación (EER) generado por Workbench

Un diagrama relacional representa las tablas que forman parte de la base de datos, y muestra cómo están relacionadas entre sí.



Identificar esto ayuda a comprender cómo se estructura la información en la base de datos y a entender cómo se relacionan las diferentes tablas para construir las consultas que se mostrarán a continuación:

A continuación, se visualizará la inserción de los datos en cada una de las tablas:

```
-- Se insertan datos en la tabla procedimiento
INSERT INTO tb_procedimiento(id_procedimiento, tipo_procedimiento)
VALUES ('1', 'Cirugía cardíaca'),
('2', 'Cirugía curativa'),
('3', 'Cirugía paliativa'),
('4', 'Cirugía exploratoria'),
('5', 'Cirugía diagnóstica');

-- Se insertan datos en la tabla médico
INSERT INTO tb_medico(id_medico, nombre_medico, apellido_medico, especialidad)
VALUES('13547899', 'Antonio', 'Pérez López', 'Anestesiología'),
('458796584', 'Ana', 'Monsalve', 'Cardiología'),
('657489658', 'Maria', 'Taborda Rendón', 'Endocrinología'),
('245786354', 'Jairo', 'Mosquera Lopera', 'Oftalmología'),
('425875148', 'Alejandro', 'Serna Agudelo', 'Otorrinolaringología');

-- Se insertan datos en la tabla medicamento
INSERT INTO tb_medicamento(id_medicamento, nombre_medicamento, dosis)
VALUES ('145', 'naproxeno', '200mg'),
('465', 'celecoxib', '150mg'),
('324', 'ibuprofeno', '300mg'),
('874', 'ketorolaco', '50mg'),
('957', 'acetaminofén', '150mg');

-- Se insertan datos a la tabla paciente
INSERT INTO tb_paciente (id_paciente, nombre_paciente, apellido_paciente, direccion, id_procedimiento)
VALUES ('49875124', 'Tatiana', 'Monsalve Herrera', 'c11 24 #89-12', '1'),
('49251875', 'Catalina', 'Cardona Bustamante', 'c11 87 #87-58', '2'),
('57412698', 'Valeria', 'Aguirre Montoya', 'cra 56 #58-78', '2'),
('32148556', 'Daniela', 'Castro Torres', 'c11 76 #74-52', '3'),
('28446468', 'Maritza', 'Taborda Giraldo', 'cra 23 #45-76', '5');

-- Insertar datos en la tabla enfermero
INSERT INTO tb_enfermero(id_enfermero, nombre_enfermero, apellido_enfermero, id_medico)
VALUES ('13541258', 'Ángel', 'Aguirre Bustamante', '13547899'),
('25489647', 'Antonio', 'Herrera Patiño', '458796584'),
('25698741', 'Angélica', 'Gómez Pérez', '657489658'),
('42156465', 'Maria', 'Mosquera Giraldo', '245786354'),
('64454454', 'Andrea', 'Taborda Pérez', '425875148');

-- Insertar datos en la tabla detalle paciente medicamento
INSERT INTO dll_paciente_medicamento(id_paciente_medicamento, id_paciente, id_medicamento)
VALUES('1', '49875124', '145'),
('2', '49251875', '465'),
('3', '57412698', '324'),
('4', '32148556', '324'),
('5', '28446468', '957');

-- Insertar datos en la tabla detalle médico procedimiento
INSERT INTO dll_medico_procedimiento(id_medico_procedimiento, id_medico, id_procedimiento)
VALUES ('1', '13547899', '1'),
('2', '458796584', '2'),
('3', '657489658', '3'),
('4', '245786354', '4'),
('5', '425875148', '5');

-- Insertar datos en la tabla factura
INSERT INTO tb_factura(id_factura, fecha, valor_total, id_paciente)
VALUES ('F647', '12/01/2023', '15000', '49875124'),
('F135', '08/01/2022', '8000', '49251875'),
('F021', '15/02/2022', '20000', '57412698'),
('F754', '02/10/2021', '13000', '32148556'),
('F325', '25/11/2022', '14000', '28446468');
```

```
-- Se insertan datos a la tabla teléfono paciente
INSERT INTO tb_telefono_paciente (id_paciente, telefono)
VALUES ('49875124', '3121028745'),
('49251875', '3452147021'),
('57412698', '3100546544'),
('32148556', '3004464648'),
('28446468', '3044545451');

INSERT INTO tb_telefono_enfermero (id_enfermero, telefono)
VALUES ('13541258', '3121547525'),
('25489647', '3201456987'),
('25698741', '3215465431'),
('42156465', '3001456465'),
('64454454', '3115454466');

-- Se insertan datos a la tabla teléfono médico
INSERT INTO tb_telefono_medico (id_medico, telefono)
VALUES('13547899', '3201557854'),
('458796584', '3202541200'),
('657489658', '3054546484'),
('245786354', '3154165468'),
('425875148', '3154564854');
```

Resultado satisfactorio después de insertar cada uno de los registros

```
✓ 19 10:12:00 INSERT INTO tb_procedimiento(id_procedimiento, tipo_procedimiento) VALUES ('1', 'Cirugía cardíaca'), ('2', 'Cirugía curativa'), ('3', 'Cirugía paliativa'), ('4', '...' 5 row(s) affected Records: 5 Duplicates: 0 Warnings: 0
✓ 20 10:12:00 INSERT INTO tb_medico(id_medico, nombre_medico, apellido_medico, especialidad) VALUES('13547899', 'Antonio', 'Pérez López', 'Anestesiología'), ('458...' 5 row(s) affected Records: 5 Duplicates: 0 Warnings: 0
✓ 21 10:12:00 INSERT INTO tb_medamento(id_medamento, nombre_medamento, dosis) VALUES ('145', 'haproxeno', '200mg'), ('465', 'celecoxib', '150mg'), ('324', 'ib...' 5 row(s) affected Records: 5 Duplicates: 0 Warnings: 0
✓ 22 10:12:00 INSERT INTO tb_paciente (id_paciente, nombre_paciente, apellido_paciente, direccion, id_procedimiento) VALUES ('49875124', 'Tatiana', 'Monsalve Herr...' 5 row(s) affected Records: 5 Duplicates: 0 Warnings: 0
✓ 23 10:12:00 INSERT INTO tb_telefono_paciente (id_paciente, telefono) VALUES ('49875124', '3121028745'), ('49251875', '3452147021'), ('57412698', '3100546544'), ... 5 row(s) affected Records: 5 Duplicates: 0 Warnings: 0
✓ 24 10:12:00 INSERT INTO tb_enfermero(id_enfermero, nombre_enfermero, apellido_enfermero, id_medico) VALUES ('13541258', 'Ángel', 'Aguirre Bustamante', '135478...' 5 row(s) affected Records: 5 Duplicates: 0 Warnings: 0
✓ 25 10:12:00 INSERT INTO dll_paciente_medamento(id_paciente_medamento, id_paciente, id_medamento) VALUES('1', '49875124', '145'), ('2', '49251875', '465'), ... 5 row(s) affected Records: 5 Duplicates: 0 Warnings: 0
✓ 26 10:12:00 INSERT INTO dll_medico_procedimiento(id_medico_procedimiento, id_medico, id_procedimiento) VALUES ('1', '13547899', '1'), ('2', '458796584', '2'), ('3', '...' 5 row(s) affected Records: 5 Duplicates: 0 Warnings: 0
✓ 27 10:12:00 INSERT INTO tb_factura(id_factura, fecha, valor_total, id_paciente) VALUES ('F647', '12/01/2023', '15000', '49875124'), ('F135', '08/01/2022', '8000', '49...' 5 row(s) affected Records: 5 Duplicates: 0 Warnings: 0
✓ 28 10:12:00 INSERT INTO tb_telefono_enfermero (id_enfermero, telefono) VALUES ('13541258', '3121547525'), ('25489647', '3201456987'), ('25698741', '3215465431')... 5 row(s) affected Records: 5 Duplicates: 0 Warnings: 0
✓ 29 10:12:00 INSERT INTO tb_telefono_medico (id_medico, telefono) VALUES('13547899', '3201557854'), ('458796584', '3202541200'), ('657489658', '3054546484'), (... 5 row(s) affected Records: 5 Duplicates: 0 Warnings: 0
```

Se consultan todos los datos de una tabla, utilizando la siguiente sentencia:

Tabla teléfono médico

```
195 • SELECT *
196 FROM tb_telefono_medico;
197
```

id_medico	telefono
13547899	3201557854
245786354	3154165468
425875148	3154564854
458796584	3202541200
657489658	3054546484

Tabla médico

```
198 • SELECT *
199 FROM tb_medico;
200
```

id_medico	nombre_medico	apellido_medico	especialidad
13547899	Antonio	Pérez López	Anestesiología
245786354	Jairo	Mosquera Lopera	Oftalmología
425875148	Alejandro	Serna Agudelo	Otorrinolaringología
458796584	Ana	Monsalve	Cardiología
657489658	Maria	Taborda Rendón	Endocrinología

Tabla procedimiento

```
201 • SELECT *
202 FROM tb_procedimiento;
203
```

Result Grid		Filter Rows:
	id_procedimiento	tipo_procedimiento
▶	1	Cirugía cardíaca
	2	Cirugía curativa
	3	Cirugía paliativa
	4	Cirugía exploratoria
	5	Cirugía diagnóstica
*	NULL	NULL

Tabla medicamento

```
204 • SELECT *
205 FROM tb_medimento;
206
```

Result Grid		Filter Rows:	Edit:
	id_medimento	nombre_medimento	dosis
▶	145	naproxeno	200mg
	324	ibuprofeno	300mg
	465	celecoxib	150mg
	874	ketorolaco	50mg
	957	acetaminofén	150mg
*	NULL	NULL	NULL

Tabla paciente

```
207 • SELECT *
208 FROM tb_paciente;
209
```

Result Grid

Filter Rows:

Edit

Export/Import:

	id_paciente	nombre_paciente	apellido_paciente	direccion	id_procedimiento
▶	28446468	Maritza	Taborda Giraldo	cra 23 #45-76	5
	32148556	Daniela	Castro Torres	cl 76 #74-52	3
	49251875	Catalina	Cardona Bustamante	cl 87 #87-58	2
	49875124	Tabiana	Monsalve Herrera	cl 24 #89-12	1
	57412698	Valeria	Aguirre Montoya	cra 56 #58-78	2
*	NULL	NULL	NULL	NULL	NULL

Tabla teléfono paciente

```
210 • SELECT *
211 FROM tb_telefono_paciente;
212
```

Result Grid		Filter Rows:
	id_paciente	telefono
▶	28446468	3044545451
	32148556	3004464648
	49251875	3452147021
	49875124	3121028745
	57412698	3100546544
*	NULL	NULL

Tabla enfermero

```
213 • SELECT *
214 FROM tb_enfermero;
215
```

	id_enfermero	nombre_enfermero	apellido_enfermero	id_medico
▶	13541258	Ángel	Aguirre Bustamante	13547899
	25489647	Antonio	Herrera Patiño	458796584
	25698741	Angélica	Gómez Pérez	657489658
	42156465	Maria	Mosquera Giraldo	245786354
	64454454	Andrea	Taborda Pérez	425875148
*	NULL	NULL	NULL	NULL

Tabla teléfono enfermero

```
216 • SELECT *
217 FROM tb_telefono_enfermero;
218
```

	id_enfermero	telefono
▶	13541258	3121547525
	25489647	3201456987
	25698741	3215465431
	42156465	3001456465
	64454454	3115454466
*	NULL	NULL

Tabla detalle paciente medicamento (M:N)

```
219 • SELECT *
220 FROM dll_paciente_medimento;
221
```

	id_paciente_medimento	id_paciente	id_medimento
▶	1	49875124	145
	2	49251875	465
	3	57412698	324
	4	32148556	324
	5	28446468	957
*	NULL	NULL	NULL

Tabla detalle médico procedimiento (M:N)

```
222 • SELECT *
223 FROM dll_medico_procedimiento;
224
```

	id_medico_procedimiento	id_medico	id_procedimiento
▶	1	13547899	1
	2	458796584	2
	3	657489658	3
	4	245786354	4
	5	425875148	5
*	NULL	NULL	NULL

Tabla factura

```
225 • SELECT *
226 FROM tb_factura;
227
```

	id_factura	fecha	valor_total	id_paciente
▶	F021	15/02/2022	20000	57412698
	F135	08/01/2022	8000	49251875
	F325	25/11/2022	14000	28446468
	F647	12/01/2023	15000	49875124
	F754	02/10/2021	13000	32148556
*	NONE	NONE	NONE	NONE

Estructura de las consultas que se realizaron

Consulta #1. Conocer que medicamentos a tomado cada paciente y la dosis suministrada.

Se utiliza la instrucción **SELECT**, seguido de esta se indican los campos que se desean visualizar (documento, nombre y apellido del paciente, nombre del medicamento y dosis), la palabra clave **AS** nos permite asignarle un *alias* a la columna que le precede (es decir, cambiar el texto del encabezado que aparece en la columna de la respuesta), la palabra clave **FROM** seguido del nombre de la tabla de donde se desea traer los datos, **INNER JOIN** es una palabra clave que nos permite unir la información de dos tablas en una sola consulta, el nombre de la tabla especificada por el **FROM** (también conocida como tabla de la izquierda) y el nombre de la tabla siguiente a la palabra clave **JOIN** (también conocida como la tabla de la derecha), seguido de la palabra clave **ON** y una condición bajo la cual se unirá la información de dos tablas (en este caso la información se unirá en el primer JOIN solo en las filas donde el id del paciente sea igual en ambas tablas y en el segundo JOIN solo en las filas donde el id medicamento sea igual en ambas tablas).

```
230 • SELECT tb_paciente.id_paciente AS "Documento paciente", tb_paciente.nombre_paciente AS "Nombre", tb_paciente.apellido_paciente AS "Apellido", tb_medimento.nombre_medimento AS "Medicamento", tb_medimento.dosis
231 FROM tb_paciente
232 INNER JOIN dll_paciente_medimento ON tb_paciente.id_paciente = dll_paciente_medimento.id_paciente
233 INNER JOIN tb_medimento ON dll_paciente_medimento.id_medimento = tb_medimento.id_medimento;
234
```

	Documento paciente	Nombre	Apellido	Medicamento	dosis
▶	28446468	Maritza	Taborda Giraldo	acetaminofén	150mg
	32148556	Daniela	Castro Torres	ibuprofeno	300mg
	49251875	Catalina	Cardona Bustamante	celecoxib	150mg
	49875124	Tatiana	Monsalve Herrera	naproxeno	200mg
	57412698	Valeria	Aguirre Montoya	ibuprofeno	300mg

Consulta #2. Conocer que enfermeros estuvieron en los procedimientos de los pacientes.

Se utiliza la instrucción **SELECT**, seguido de esta se indican los campos que se desean visualizar (nombre y apellido del enfermero, nombre y apellido del paciente y el tipo de procedimiento que se le realizó al paciente), la palabra clave **AS** nos permite asignarle un *alias* a la columna que le precede (es decir, cambiar el texto del encabezado que aparece en la columna de la respuesta), función **CONCAT** seguido de los campos que quiero concatenar o unir en un solo campo cuando se visualiza la información (en este caso nombre y apellido), la palabra clave **FROM** seguido del nombre de la tabla de donde se desea traer los datos, **INNER JOIN** es una palabra clave que nos permite unir la información de dos tablas en una sola consulta, el nombre de la tabla especificada por el **FROM** (también conocida como tabla de la izquierda) y el nombre de la tabla siguiente a la palabra clave **JOIN** (también conocida como la tabla de la derecha), seguido de la palabra clave **ON** y una condición bajo la cual se unirá la información de dos tablas, en este caso donde el id sea igual en ambas tablas:

Primer JOIN: id_procedimiento (igual ambas tablas)

Segundo JOIN: id_procedimiento (igual ambas tablas)

Tercer JOIN: id_medico (igual ambas tablas)

Cuarto JOIN: id_medico (igual ambas tablas)

```
236 • SELECT CONCAT(tb_enfermero.nombre_enfermero, " ", tb_enfermero.apellido_enfermero) AS "Nombre enfermero",
237        CONCAT(tb_paciente.nombre_paciente, " ", tb_paciente.apellido_paciente) AS "Nombre del paciente",
238        tb_procedimiento.tipo_procedimiento AS "Tipo de procedimiento"
239 FROM tb_paciente
240 INNER JOIN tb_procedimiento ON tb_procedimiento.id_procedimiento = tb_paciente.id_procedimiento
241 INNER JOIN dll_medico_procedimiento ON dll_medico_procedimiento.id_procedimiento = tb_procedimiento.id_procedimiento
242 INNER JOIN tb_medico ON tb_medico.id_medico = dll_medico_procedimiento.id_medico
243 INNER JOIN tb_enfermero ON tb_enfermero.id_medico = tb_medico.id_medico;
244
```

Nombre enfermero	Nombre del paciente	Tipo de procedimiento
Andrea Tabora Pérez	Maritza Tabora Giraldo	Cirugía diagnóstica
Angélica Gómez Pérez	Daniela Castro Torres	Cirugía paliativa
Antonio Herrera Patiño	Catalina Cardona Bustamante	Cirugía curativa
Ángel Aguirre Bustamante	Tatiana Monsalve Herrera	Cirugía cardíaca
Antonio Herrera Patiño	Valeria Aguirre Montoya	Cirugía curativa

¿Qué es una vista?

Una vista es una tabla virtual que se guarda en la base de datos, pero no como estructura sino como consultas con un nombre que la identifica y se utilizan para guardar consultas que se utilizan o ejecutan de manera frecuente.

Sintaxis para crear una vista

```
CREATE VIEW view_name AS
SELECT column1, column2, ...
FROM table_name
WHERE condition;
```

Vista #1: Se crea una vista la cual contiene la consulta que muestra toda la información de los **enfermeros** que acompañan los **médicos**, esto nos permitirá reconocer las habilidades de cada enfermero y saber en qué procedimientos médicos pueden realizar un acompañamiento.

Esta vista **presencia_enfermero**, crea una tabla virtual que combina información de tres tablas existentes: **tb_medico**, **tb_enfermero** y **tb_telefono_enfermero**.

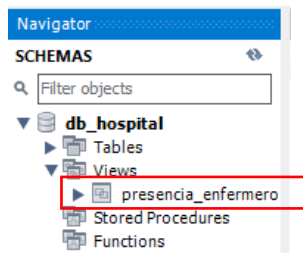
La vista nos muestra dos columnas: el nombre del enfermero y el nombre del médico, esta información se obtiene a través de dos INNER JOIN entre la tabla **tb_medico** y **tb_enfermero** – **tb_enfermero** y **tb_telefono_enfermero**.

Se utiliza la siguiente sentencia SQL para crear una vista:

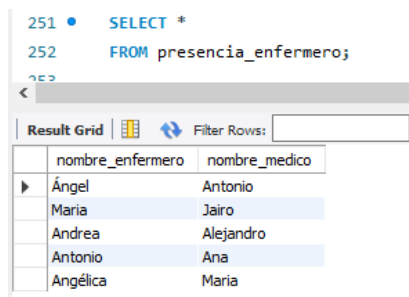
```
CREATE VIEW presencia_enfermero AS
SELECT tb_enfermero.nombre_enfermero, tb_medico.nombre_medico
FROM tb_medico
INNER JOIN tb_enfermero ON tb_enfermero.id_medico = tb_medico.id_medico
INNER JOIN tb_telefono_enfermero ON tb_telefono_enfermero.id_enfermero = tb_enfermero.id_enfermero;
```

✓ 24 13:10:38 CREATE VIEW presencia_enfermero AS SELECT tb_enfermero.nombre_enfermero, tb_medico.nombre_medico FROM tb_medico INNER JOIN tb_enfer...

Una vez creada la vista, podremos visualizarla en la parte superior izquierda, en el panel de *Navigator* de Workbench, posteriormente presionar la opción *Views* para desplegar las vistas disponibles.



La información de las vistas puede ser accedida de la misma manera que se accede a la información de una tabla, utilizando la sentencia *SELECT* de la siguiente manera:



Vista #2: Se crea una vista la cual contiene la consulta de los **enfermeros** que estuvieron en los **procedimientos** de los **pacientes**, lo cual nos permitiría llevar un seguimiento de este acompañamiento.

Esta vista **enfermero_procedimiento**, crea una tabla virtual que combina información de tres tablas existentes: *tb_paciente*, *tb_procedimiento*, *dll_medico_procedimiento* *tb_medico* y *tb_enfermero*.

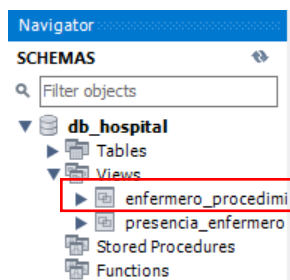
La vista nos muestra tres columnas: el nombre del enfermero, el nombre, nombre del paciente y nombre del procedimiento que se le realizó al paciente, esta información se obtiene a través de cuatro INNER JOIN entre la tabla *tb_paciente* y *tb_procedimiento* – *tb_procedimiento* y *dll_medico_procedimiento* – *dll_medico_procedimiento* y *tb_medico* – *tb_medico* y *tb_enfermero*.

Se utiliza la siguiente sentencia SQL para crear una vista:

```
CREATE VIEW enfermero_procedimiento AS
SELECT tb_enfermero.nombre_enfermero, tb_paciente.nombre_paciente, tb_procedimiento.tipo_procedimiento
FROM tb_paciente
INNER JOIN tb_procedimiento ON tb_procedimiento.id_procedimiento = tb_paciente.id_procedimiento
INNER JOIN dll_medico_procedimiento ON dll_medico_procedimiento.id_procedimiento = tb_procedimiento.id_procedimiento
INNER JOIN tb_medico ON tb_medico.id_medico = dll_medico_procedimiento.id_medico
INNER JOIN tb_enfermero ON tb_enfermero.id_medico = tb_medico.id_medico;
```

✓ 26 13:15:04 CREATE VIEW enfermero_procedimiento AS SELECT tb_enfermero.nombre_enfermero, tb_paciente.nombre_paciente, tb_procedimiento.tipo_procedimi...

Una vez creada la vista, podremos visualizarla en la parte superior izquierda, en el panel de *Navigator* de Workbench, posteriormente presionar la opción *Views* para desplegar las vistas disponibles.



La información de las vistas puede ser accedida de la misma manera que se accede a la información de una tabla, utilizando la sentencia *SELECT* de la siguiente manera:

```
263 • SELECT *
264 FROM enfermero_procedimiento;
265
```

nombre_enfermero	nombre_paciente	tipo_procedimiento
Andrea	Maritza	Cirugía diagnóstica
Angélica	Daniela	Cirugía paliativa
Antonio	Catalina	Cirugía curativa
Ángel	Tatiana	Cirugía cardíaca
Antonio	Valeria	Cirugía curativa

Vista #3: Se crea una vista la cual contiene la consulta de los **medicamentos** que ha tomado cada **paciente** y la dosis suministrada, lo anterior es importante para saber qué medicamento se le recetó a cada paciente y no llegar a alterar su dosis.

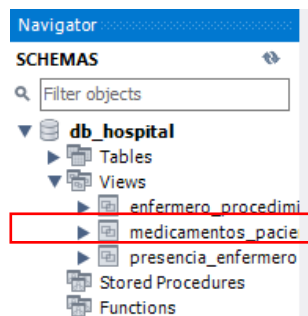
Esta vista **medicamento_paciente**, crea una tabla virtual que combina información de tres tablas existentes: *tb_paciente*, *dll_paciente_medicamento* y *tb_medicamento*.

La vista nos muestra tres columnas: nombre del paciente, nombre del medicamento que se le recetó al paciente y su dosis, esta información se obtiene a través de dos INNER JOIN entre la tabla *tb_paciente* y *dll_paciente_medicamento* – *dll_paciente_medicamento* y *tb_medicamento*.

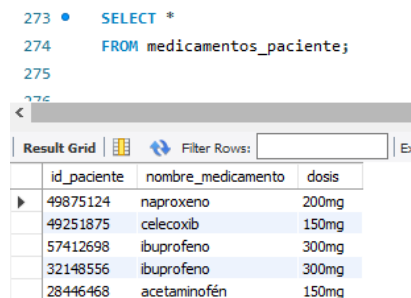
```
CREATE VIEW medicamentos_paciente AS
SELECT tb_paciente.id_paciente, tb_medicamento.nombre_medicamento, tb_medicamento.dosis
FROM tb_paciente
INNER JOIN dll_paciente_medicamento ON tb_paciente.id_paciente = dll_paciente_medicamento.id_paciente
INNER JOIN tb_medicamento ON dll_paciente_medicamento.id_medicamento = tb_medicamento.id_medicamento;
```

✓ 28 13:19:46 CREATE VIEW medicamentos_paciente AS SELECT tb_paciente.id_paciente, tb_medicamento.nombre_medicamento, tb_medicamento.dosis FROM tb...

Una vez creada la vista, podremos visualizarla en la parte superior izquierda, en el panel de *Navigator* de Workbench, posteriormente presionar la opción *Views* para desplegar las vistas disponibles.



La información de las vistas puede ser accedida de la misma manera que se accede a la información de una tabla, utilizando la sentencia *SELECT* de la siguiente manera:



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

Nueva tabla de historias clínicas: la creación de esta tabla permitiría almacenar y agrupar de una manera más organizada la información médica de un paciente, lo cual facilita el acceso y modificación de dicha información por parte de los profesionales de la salud en el hospital, además nos permite realizar un seguimiento del paciente a largo plazo, esto puede ser útil para concertar los procedimientos y mejorar los resultados en la atención médica en general.