

UNIVERSIDAD TECNOLÓGICA NACIONAL

FACULTAD REGIONAL RESISTENCIA

INGENIERÍA EN SISTEMAS DE INFORMACIÓN

**BASES DE DATOS**

Trabajo Práctico Integrador

Segunda Entrega

**Grupo 4**

Integrantes:

* **Andres**, Aldo Omar.
* **Bravo Pérez**, Agustín Nicolás.
* **Brites,** Agustín.
* **Sáez Franci**, Juliana Carla Desiree.
* **Salomón**, Hilel Mauricio.
* **Machuain**, Ezequiel.

*1 de Julio de 2023*

# Índice

# 

[Índice](#_vwyv4mu5qrpo) 1

[Escenario Trabajado](#_han8eqh0gcn7) 2

[Diagrama Entidad Relación](#_v5kik6zfawjy) 4

[Esquema Relacional](#_sm0anmwef22o) 5

[Diccionario de datos (del DER)](#_vgptxxtg5j23) 8

[Modelo Conceptual en SQL](#_3s2ey774mr4h) 15

[Generación de datos](#_hc4ir3ytofxp) 21

[Consultas SQL](#_j08bwf5yul97) 21

# 

# Introducción

Este informe contiene la documentación solicitada para la entrega del Trabajo Práctico Integrador del grupo 4, dentro de la cursada 2023 de Bases de Datos.

A la par de este informe, dentro del mismo archivo comprimido, se encuentran otros archivos, más relevantes:

* **Grupo4-DER.png** con el Diagrama Entidad Relación en mejor calidad.
* **schema.sql** contiene un script que crea una base de datos `farmaciasdb` vacía.
* **dump.sql** crea la base de datos `farmaciasdb` y la carga con entre 10 mil y 100 mil registros (generados aleatoriamente) por cada tabla.
* **table\_row\_counts.sql** contiene una consulta que trae la cantidad de registros en cada tabla de la base de datos. Solo valida la cantidad de datos generados.
* **delete\_queries.sql** contiene las 6 consultas de eliminación de registros.
* **update\_queries.sql** contiene las 6 consultas de modificación de registros.
* **insert\_queries.sql** contiene las 6 consultas de creación de registros.
* **select\_queries.sql** contiene las 8 consultas de lectura.

# Escenario Trabajado

Se desea mantener una base de datos para la gestión de una cadena de farmacias distribuida en diferentes ciudades. De la ciudad se sabe su nombre y su código postal. De cada farmacia, su ID, su dirección (calle y número) y los días que le corresponde guardia.

Una farmacia está ubicada en una sola ciudad, pero en una ciudad hay más de una farmacia. A su vez, sabemos que por cada ciudad existe un único farmacéutico; es decir, en las ciudades en las que hubiere más de una farmacia, el mismo farmacéutico estará afectado a todas las farmacias de esa ciudad. En cada farmacia trabajan varios empleados.

De cada empleado queremos saber su CUIT, su nombre, la fecha de ingreso laboral. Tenga en cuenta que cada empleado trabaja en una sola farmacia.

Esta cadena de farmacias vende medicamentos solo a sus afiliados. Los datos que se deben guardar son: Id de afiliado, apellido y nombre, tipo y número de documento, dirección, localidad, fecha de ingreso, fecha de nacimiento.

Existen dos tipos de afiliados: eventuales, que reciben un 20% de descuento sobre las compras realizadas, y crónicos, cuyo descuento es del 70%. Para el caso de los crónicos se debe guardar información sobre código de diagnóstico y fecha de diagnóstico. Un afiliado crónico puede tener varios diagnósticos.

De cada venta se debe generar un comprobante que contenga: número, fecha, id de afiliado. A su vez cada comprobante contiene un conjunto de medicamentos que se deben cargar teniendo en cuenta los siguientes datos: código de medicamento, precio, descuento (directamente relacionado al tipo de afiliado) y total.

Para poder efectivizar una venta, cada farmacia, posee su stock de cada medicamento. Cada medicamento se identifica por código, descripción, presentación (ej: ampollas de 5 unidades, jarabe de 100ml, inyecciones por 10 unidades) y precio, que es el mismo para todas las farmacias. También se conoce la o las monodrogas que componen cada medicamento, el laboratorio que lo comercializa y las acciones terapéuticas que tiene.

De cada monodroga sabemos el nombre científico y el nombre comercial. De cada laboratorio sabemos CUIT, razón social, domicilio. Un laboratorio provee varios medicamentos a esta cadena de farmacias.

De las acciones terapéuticas conocemos el nombre y el tiempo que tarda en hacer efecto. Tenga en cuenta que una acción terapéutica puede repetirse para distintos medicamentos. Por ejemplo, el medicamento Dorixina Forte es un medicamento que cuesta $1360 y su presentación es en caja de 20 comprimidos. Tiene como monodrogas Clonixinato de lisina (nombre científico) en 125,00 mg y Dextropropoxifeno napsilato 2 (nombre científico) en 98,00 mg. Sus acciones terapéuticas son analgésicas y antiinflamatorias y tardan 4 horas en hacer efecto en la persona que toma el medicamento.

El sistema deberá permitir consultar la base de datos de diferentes alternativas para medicamentos compuestos por una monodroga, medicamentos de un laboratorio, medicamentos con el mismo nombre y distinta presentación, entre otras.

La cadena de farmacias cuenta con un depósito central, que realiza todas las compras y recibe todos los medicamentos solicitados por cada farmacia a los proveedores (laboratorios). Este deposito posee su propio stock de medicamentos que luego redistribuye a cada farmacia que lo solicite.

La forma de ingresar medicamentos al stock del depósito central es a través de los INGRESOS. De cada ingreso se registra: código de Ingreso, fecha de ingreso, código de transporte (es la denominación de la empresa que efectuó el traslado de los medicamentos), CUIT del proveedor.

Cada ingreso contiene un conjunto de medicamentos que se deben cargar teniendo en cuenta los siguientes datos: código de medicamento, cantidad.

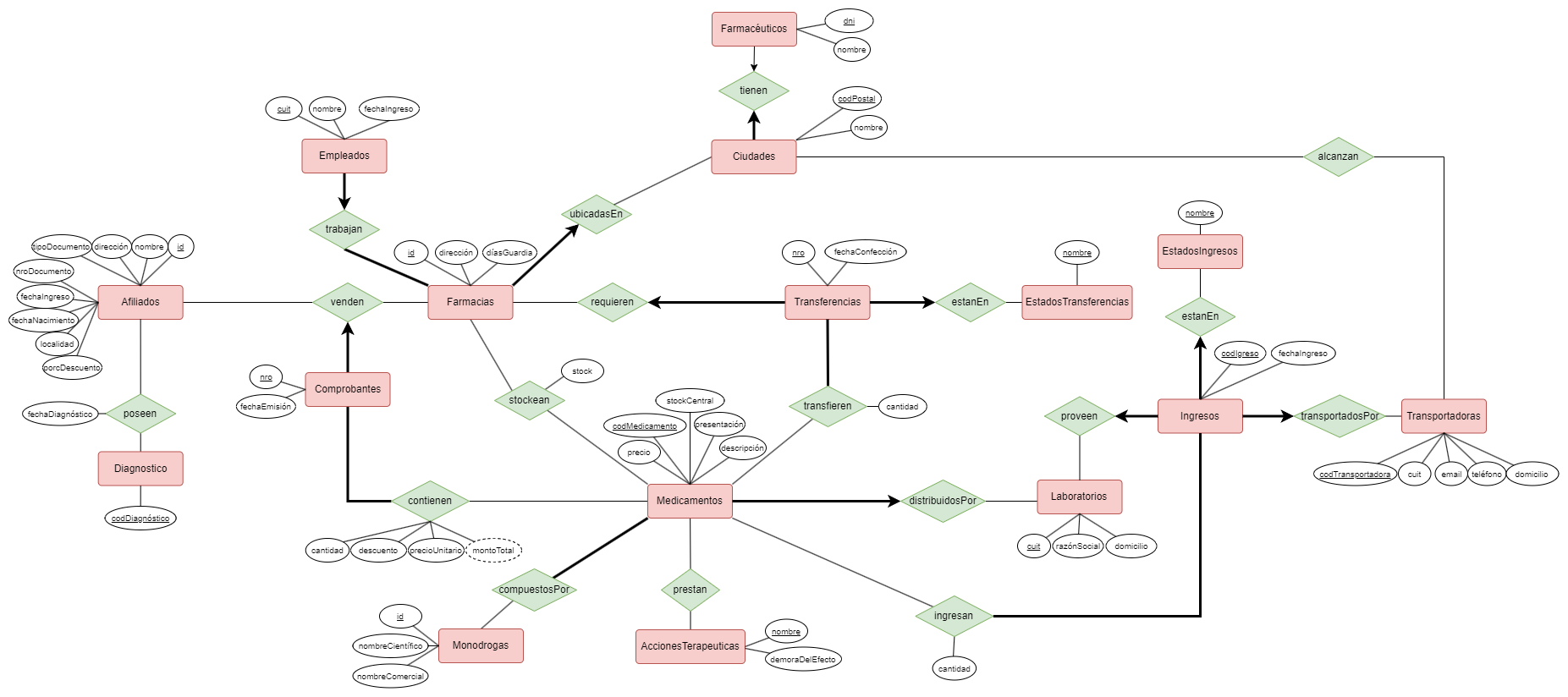
Se debe tener en cuenta que es necesario poder determinar en qué estado se encuentra cada Ingreso (en confección, terminado, procesado), ya que sólo se puede impactar en el stock del depósito un ingreso terminado.

Existe una lista de empresas de trasportes a quienes se le puede designar la tarea de trasladar medicamentos de un depósito a otro. Los datos con que se cuentan son: código de transporte, razón social, CUIT/CUIL, e-mail, teléfono y domicilio. Además, se debe conocer de cada transportista a qué localidades (sucursales) alcanza su servicio.

Cada farmacia que necesite medicamentos, lo debe requerir al depósito central por medio de TRANFERENCIAS, las cuales deben contener: número de solicitud, Id de farmacia, fecha de confección (es la fecha actual y no puede modificarse).

Cada transferencia contiene un conjunto de medicamentos que se deben cargar teniendo en cuenta los siguientes datos: código de medicamento y cantidad. Se debe tener en cuenta que es necesario poder determinar en qué estado se encuentra cada solicitud (En confección, terminado, procesado), ya que sólo se puede impactar en el stock de la farmacia una transferencia terminada.

# Diagrama Entidad Relación



**Consideraciones**

1) Los afiliados eventuales se guardan en la entidad "Afiliados" con porcDescuento = 20. Los afiliados crónicos se guardan en la misma entidad con porcDescuento = 70, y se sabe que son crónicos dado que participarían en la relación 'poseen' con Diagnóstico.

2) Se agregó atributo clave 'id' a la entidad Monodrogas para identificarla de manera clara.

3) El stock de cada medicamentos del Depósito Central se guarda en la entidad Medicamentos en el atributo 'stockCentral'.

4) Los atributos dni y nombre de un farmacéutico fueron propuestos por nosotros.

Modificaciones para la segunda entrega:

1. Se modificó la relación entre Farmacias y Transferencias (requieren) ya que en el modelo anterior se permitía la posibilidad de que una misma transferencia pertenezca o se asocie a varias farmacias. Ahora toda transferencia corresponde a una sola farmacia.
2. Se quitó la herencia entre "Afiliados" y "AfiliadosCrónicos". Ahora un afiliado es crónico si participa en la relación "poseen" con Diagnóstico. Esta relación antes tenía restricción de participación total, ahora ya no.

# Esquema Relacional

**ciudades**(codPostal:*entero(10)*, nombre:*caracter(200), dniF: entero(8)*)

CP: codPostal

CF: dniF → farmaceuticos(dni)

CNN:

**farmacias**(id:*entero(10)*, direccion:*caracter(50)*, diasGuardia: *caracter(100)*, codPostal:entero(10))

CP: id

CF: codPostal → ciudades(codPostal)

CNN: direccion, diasGuardia, codPostal

**empleados**(cuit: *entero(11)*, nombre:*caracter(50)*, fechaIngreso: *DATE*, idFarmacia: entero(10))

CP: cuit

CP: idFarmacia → farmacias(id)

CNN:nombre, fechaIngreso, idFarmacia

**afiliados**(id: *entero(10)*, nombre:*caracter(50)*, direccion: *caracter(50)*, tipoDocumento:*caracter(5)*, NroDocumento:*entero(10)*, fechaIngreso:*DATE*, fechaNacimento:*DATE*, localidad:*caracter(50)*, porcDescuento: *entero(3)*)

CP: id

CNN: nombre, tipoDocumento, nroDocumento

**diagnostico**(codDiagnostico: entero(10))

CP: codDiagnostico

CF:

**farmaceuticos**(DNI: entero(10),nombre:nombre:*caracter(50)*)

CP: DNI

**comprobantes**(nro: entero(10), fechaEmision: *DATE),* idA: entero(10), idF: entero(10))

CP:nro

CF: idA → afiliados(id)

idF → farmacias(id)

CNN: fechaEmision, idA, idF

**laboratorios**(cuit:*entero(11)*, razonSocial: *caracter(100)*, domicilio:*caracter(100)*)

CP: cuit

CNN:

**medicamentos**(codMedicamento:*entero(10)* , descripcion:*caracter(250)*, precio:*decimal(10,3)*, stockCentral: entero(10), presentacion:*caracter(50)*, cuitLab: entero(11))

CP: codMedicamento

CF:cuitLab → Laboratios(cuit)

CNN:cuitLab, stockCentral, descripcion

**contienen**(codMed: *entero(10)*, nroComprobante: *entero(10*), cantidad: entero(10), descuento: *decimal(10,2)*, precioUnitario: *decimal(10,3)*, montoTotal: *decimal(10,3)*)

CP: nroComprobante, codMed

CF:nroComprobante → comprobantes(nro),

codMed → medicamentos(codMedicamento)

CNN: cantidad, precioUnitario

**stockean**(codMed:*entero(10)*, idF:*entero(10), cantidad:entero(10), stock:entero(10))*

CP:idF, codMed

CF:idF → farmacias(id),

codMed → medicamentos(codMedicamento)

**monodrogas**(id: entero(10), *nombreCientifico: caracter(70)*, nombreComercial: *caracter(70)*)

CP: id

CNN: nombreComercial

**poseen**(codDiagnostico: entero(10),idDiag*: caracter(10)*, fechaDiagnostico: *DATE*)

CP: codDiagnostico,Id

CF:idDiag → afiliados(id),

codDiagnostico → Diagnostico()

CNN: fechaDiagnostico

**compuestosPor**(codMed:*entero(10)*,idMono:*entero(10*)

CP:codMed, idMono

CF:idMono → monodrogas(id),

codMed → medicamentos(codMedicamento)

CNN:

**accionesTerapeuticas**(nombre:*caracter(80)*, demoraDelEfecto:*entero(10)*)

CP: nombre

CNN: demoraDelEfecto

**prestan**(codMed:*entero(10)*, nombreAccion:*caracter(80)*)

CP:codMed, nombreAccion

CF:nombreAccion → accionesTerapeuticas(nombre),

codMed → Medicamentos(codMedicament9o)

CNN:

**estadosTransferencias**(nombre:*caracter(30)*)

CP: nombre

**transferencias**(nro: *entero(10)*, fechaConfeccion: *DATE*, estado:*caracter(30), idF: entero(10)*)

CP: nro

CF:estado → estadosTransferencias(nombre)

idF → farmacias(id)

CNN:fechaConfeccion, estado, idF

**transfieren**(nroTransf: *entero(10),* codMed:*entero(10), cantidad:entero(10)*)

CP:nroTransf, codMed

CF:nroTransf → transferencias(nro),

codMed → Medicamentos(codMedicamento)

CNN: cantidad

**transportadoras**(*codTransportadora*:entero(10), cuit:entero(11), email:caracter(20), telefono:entero(10),domicilio:caracter(200))

CP:codTransportadoras

CNN: cuit, email

**estadosIngresos**(nombre:*caracter(200)*)

CP:nombre

**ingresos**(codIngreso: *entero(10)*, fechaIngreso: *DATE*, estado: caracter(200), cuitLab: entero(11), codTrasp: entero(10))

CP: codIngreso

CF:cuitLab → laboratorios(cuit),

codTransp → transportadoras(codTransportadora),

estado → estadosIngresos(nombre)

CNN: cuitLab, codTransp, nombre, fechaIngreso

**ingresan**(codIngreso:*entero(10),* codMed:*entero(10), cantidad:entero(10)*)

CP:codIngreso, codMed

CF:codIngreso → ingresos(codIngreso),

codMed → medicamentos(codMedicamento)

CNN: cantidad

**alcanzan**(codTransp:*entero(10)*, codPostal:*entero(10)*)

CP:codPostal, codTransp

CF:codPostal → ciudades(codPostal),

codTransp → transportadoras(codTransp)

CNN:

# Diccionario de datos

| *nombre campo* | *descripción* | *entidad/es a la que pertenece* | *Funcion (PK, FK, etc)* | *dominio/*  *longitud* | *¿Acepta nulos?* |
| --- | --- | --- | --- | --- | --- |

| *nombre* | *describe el nombre del estado de transferencia* | *EstadoTransferencia* | *PK,FK(Tranferencias)* | *caracter(200)* | *NO* |
| --- | --- | --- | --- | --- | --- |

| *nombre* | *describe el nombre del estado* | *EstadosIngreso* | *PK,FK(Ingresos)* | *caracter(200)* | *NO* |
| --- | --- | --- | --- | --- | --- |

| *cantidad* | *indica la cantidad de medicamentos que ingresan* | *Ingresa* | *-* | *entero(10)* | *SI* |
| --- | --- | --- | --- | --- | --- |

| *codDiagnostico* | *indica el codigo del diagnostico* | *Diagnostico* | *PK* | *entero(10)* | *NO* |
| --- | --- | --- | --- | --- | --- |

| *fechaDiagnostico* | *indica la fecha en la sucedió un diagnostico a un afiliado* | *poseen* | *-* | *DATE* | *NO* |
| --- | --- | --- | --- | --- | --- |

| *fechaConfeccion* | *indica la fecha de transferencias bancarias* | *Transferencias* | *-* | *DATE* | *SI* |
| --- | --- | --- | --- | --- | --- |
| *nro* | *indica un numero de operacion unico* | *Transferencias* | *PK* | *entero(10)* | *NO* |

| *nombre* | *indica la fecha de transferencias bancarias* | *Farmaceutico* | *-* | *caracter(200)* | *SI* |
| --- | --- | --- | --- | --- | --- |
| *dni* | *indica un numero de operacion unico* | *Farmaceutico* | *PK* | *entero(8)* | *NO* |

| *codIngreso* | *es el codigo del ingreso de medicamentos* | *Ingresos* | *PK* | *entero(10)* | *NO* |
| --- | --- | --- | --- | --- | --- |
| *fechaIngreso* | *indica la fecha del ingreso* | *Ingresos* | *-* | *DATE* | *SI* |

| *codPostal* | *Indica el codigo postal de una ciudad* | *Ciudades* | *PK,FK(Farmaceuticos)* | *entero(10)* | *NO* |
| --- | --- | --- | --- | --- | --- |
| *nombre* | *indica el nombre de la ciudad* | *Ciudades* | *-* | *caracter(200)* | *Si* |

| *fechaEmision* | *indica la fecha de la emision de un comprobantes* | *Comprobantes* | *-* | *DATE* | *SI* |
| --- | --- | --- | --- | --- | --- |
| *nro* | *indica el numero de comprobante que es, este valor se auto incrementa* | *Comprobantes* | *PK* | *entero(10)* | *NO* |

| *nombre* | *indica el nombre de la accion terapeutica* | *AccionesTerapeuticas* | *PK* | *caracter(200)* | *NO* |
| --- | --- | --- | --- | --- | --- |
| *demoraDelEfecto* | *indica el tiempo de demora del efecto, pude ser semana, horas, dias, etc* | *AccionesTerapeuticas* | *-* | *caracter(200)* | *SI* |

| *nombreCientifico* | *nombre cientifico de la monodroga* | *Monodroga* | *PK* | *caracter(2000)* | *NO* |
| --- | --- | --- | --- | --- | --- |
| *nombreComercial* | *nombre comercial de la monodroga* | *Monodroga* | *-* | *caracter(2000)* | *SI* |

| *CUIT* | *es el CUIT del laboratorio* | *Laboratorios* | *PK,FK(Ingresos),FK(Medicamentos)* | *entero(11)* | *NO* |
| --- | --- | --- | --- | --- | --- |
| *razonSocial* | *Indica la razón social del laboratorio* | *Laboratorios* | *-* | *caracter(200)* | *SI* |
| *domicilio* | *indica el domicilio del laboratorio* | *Laboratorios* | *-* | *caracter(200)* | *SI* |

| *CUIT* | *indica el CUIT de un empleado* | *Empleados* | *PK* | *entero(11)* | *NO* |
| --- | --- | --- | --- | --- | --- |
| *nombre* | *indica el nombre de un empleado* | *Empleados* | *-* | *caracter(200)* | *SI* |
| *fechaIngreso* | *indica la fecha de ingreso de un empleado* | *Empleados* | *-* | *DATE* | *SI* |

| *precioUnitario* | *indica el precio unitario de un medicamento en un tiempo especifico* | *contienen* | *-* | *decimal(10,3)* | *SI* |
| --- | --- | --- | --- | --- | --- |
| *descuento* | *indica el descuento que el medicamento tuvo* | *contienen* | *-* | *decimal(10,2)* | *SI* |
| *cantidad* | *indica la cantidad de medicamentos vendidos* | *contienen* | *-* | *entero(10)* | *SI* |

| *ID* | *indica el ID de la farmacia* | *Farmacias* | *PK,FK(Empleados),FK(Comprobantes)* | *entero(10)* | *NO* |
| --- | --- | --- | --- | --- | --- |
| *dirección* | *indica la dirección de la farmacia* | *Farmacias* | *-* | *caracter(200)* | *SI* |
| *diasGuardia* | *indica los dias que tiene guardia una farmacia* | *Farmacias* | *-* | *DATE* | *SI* |

| *codMedicamento* | *Indica el codico de un medicamento* | *Medicamentos* | *PK* | *entero(10)* | *NO* |
| --- | --- | --- | --- | --- | --- |
| *descrpicion* | *es una descripción de un medicamento* | *Medicamentos* | *-* | *caracter(3000)* | *SI* |
| *precio* | *indica el precio del medicamento* | *Medicamentos* | *-* | *decimal(10,3)* | *SI* |
| *presentación* | *indica la presentación de un medicamento* | *Medicamentos* | *-* | *caracter(1000)* | *SI* |
| *stockCentral* | *representa la cantidad de medicamentos total de todas las farmacias* | *Medicamentos* | *-* | *entero(10)* | *NO* |

| *codTransportadpra* | *indica el codigo de empresa transportadora* | *Transportadoras* | *PK,FK(Ingresos)* | *entero(10)* | *NO* |
| --- | --- | --- | --- | --- | --- |
| *CUIT* | *Indica el CUIT de la empresa Transportadora* | *Transportadoras* | *-* | *entero(11)* | *SI* |
| *email* | *indica el email de la misma empresa* | *Transportadoras* | *-* | *caracter(200)* | *SI* |
| *teléfono* | *indica el número telefónico de la empresa* | *Transportadoras* | *-* | *entero(10)* | *SI* |
| *domicilio* | *indica el domicilio de la empresa* | *Transportadoras* | *-* | *caracter(200)* | *SI* |

| *ID* | *Indica el ID de un afiliado* | *Afiliados* | *PK,FK(comprobantes)* | *entero(10)* | *NO* |
| --- | --- | --- | --- | --- | --- |
| *nombre* | *indica el nombre de un afiliado* | *Afiliados* | *-* | *caracter(200)* | *SI* |
| *dirección* | *indica la dirección de un afiliado* | *Afiliados* | *-* | *caracter(200)* | *SI* |
| *tipoDocumento* | *indica el tipo de documento que posee un afiliado* | *Afiliados* | *-* | *caracter(5)* | *SI* |
| *nroDocumento* | *indica el número de afiliado* | *Afiliados* | *-* | *entero(10)* | *SI* |
| *fechaIngreso* | *indica la fecha de ingreso de un afiliado* | *Afiliados* | *-* | *DATE* | *SI* |
| *fechaNacimiento* | *indica la fecha en que nacio un afiliado* | *Afiliados* | *-* | *DATE* | *SI* |
| *localidad* | *indica la localidad del afiliado* | *Afiliados* | *-* | *caracter(100)* | *SI* |
| *porcDescuento* | *indica el porcentaje de descuento que le corresponde a un afilaido* | *Afiliados* | *-* | *decimal(3,3* | *SI* |

# Modelo Conceptual en SQL

**DROP** DATABASE **IF** **EXISTS** `farmaciasdb`;

**CREATE** DATABASE `farmaciasdb`;

USE `farmaciasdb`;

**DROP** **TABLE** **IF** **EXISTS** `laboratorios`;

**CREATE** **TABLE** `laboratorios` (

`cuit` **varchar**(11) **NOT** **NULL**,

`razonSocial` **varchar**(100) **DEFAULT** **NULL**,

`domicilio` **varchar**(100) **DEFAULT** **NULL**,

**PRIMARY** **KEY** (`cuit`)

);

**DROP** **TABLE** **IF** **EXISTS** `medicamentos`;

**CREATE** **TABLE** `medicamentos` (

`codMedicamento` **int** **NOT** **NULL**,

`precio` **float** **NOT** **NULL**,

`presentacion` **varchar**(50) **DEFAULT** **NULL**,

`descripcion` **varchar**(250) **NOT** **NULL**,

`stockCentral` **int** **NOT** **NULL**,

`cuitLab` **varchar**(11) **NOT** **NULL**,

**PRIMARY** **KEY** (`codMedicamento`),

**KEY** `cuitLab` (`cuitLab`),

**CONSTRAINT** `medicamentos\_ibfk\_1` **FOREIGN** **KEY** (`cuitLab`) **REFERENCES** `laboratorios` (`cuit`)

);

**DROP** **TABLE** **IF** **EXISTS** `accionesterapeuticas`;

**CREATE** **TABLE** `accionesterapeuticas` (

`nombre` **varchar**(80) **NOT** **NULL**,

`demoraDelEfecto` **int** **NOT** **NULL**,

**PRIMARY** **KEY** (`nombre`)

);

**DROP** **TABLE** **IF** **EXISTS** `afiliados`;

**CREATE** **TABLE** `afiliados` (

`id` **int** **NOT** **NULL**,

`nombre` **varchar**(50) **NOT** **NULL**,

`direccion` **varchar**(50) **DEFAULT** **NULL**,

`tipoDocumento` **varchar**(5) **NOT** **NULL**,

`nroDocumento` **int** **NOT** **NULL**,

`fechaIngreso` datetime **DEFAULT** **NULL**,

`fechaNacimiento` datetime **DEFAULT** **NULL**,

`localidad` **varchar**(50) **DEFAULT** **NULL**,

`porcDescuento` **int** **DEFAULT** **NULL**,

**PRIMARY** **KEY** (`id`)

);

**DROP** **TABLE** **IF** **EXISTS** `farmaceuticos`;

**CREATE** **TABLE** `farmaceuticos` (

`dni` **int** **NOT** **NULL**,

`nombre` **varchar**(50) **NOT** **NULL**,

**PRIMARY** **KEY** (`dni`)

);

**DROP** **TABLE** **IF** **EXISTS** `ciudades`;

**CREATE** **TABLE** `ciudades` (

`codPostal` **int** **NOT** **NULL**,

`dniFarmaceutico` **int** **NOT** **NULL**,

`nombre` **varchar**(50) **NOT** **NULL**,

**PRIMARY** **KEY** (`codPostal`),

**UNIQUE** **KEY** `dniFarmaceutico` (`dniFarmaceutico`),

**CONSTRAINT** `ciudades\_ibfk\_1` **FOREIGN** **KEY** (`dniFarmaceutico`) **REFERENCES** `farmaceuticos` (`dni`)

);

**DROP** **TABLE** **IF** **EXISTS** `farmacias`;

**CREATE** **TABLE** `farmacias` (

`id` **int** **NOT** **NULL**,

`direccion` **varchar**(50) **NOT** **NULL**,

`diasGuardia` **varchar**(100) **NOT** **NULL**,

`codPostal` **int** **NOT** **NULL**,

**PRIMARY** **KEY** (`id`),

**KEY** `codPostal` (`codPostal`),

**CONSTRAINT** `farmacias\_ibfk\_1` **FOREIGN** **KEY** (`codPostal`) **REFERENCES** `ciudades` (`codPostal`)

);

**DROP** **TABLE** **IF** **EXISTS** `comprobantes`;

**CREATE** **TABLE** `comprobantes` (

`nro` **int** **NOT** **NULL**,

`idA` **int** **NOT** **NULL**,

`idF` **int** **NOT** **NULL**,

`fechaEmision` datetime **NOT** **NULL**,

**PRIMARY** **KEY** (`nro`),

**KEY** `idA` (`idA`),

**KEY** `idF` (`idF`),

**CONSTRAINT** `comprobantes\_ibfk\_1` **FOREIGN** **KEY** (`idA`) **REFERENCES** `afiliados` (`id`),

**CONSTRAINT** `comprobantes\_ibfk\_2` **FOREIGN** **KEY** (`idF`) **REFERENCES** `farmacias` (`id`)

);

**DROP** **TABLE** **IF** **EXISTS** `diagnosticos`;

**CREATE** **TABLE** `diagnosticos` (

`codDiagnostico` **varchar**(10) **NOT** **NULL**,

**PRIMARY** **KEY** (`codDiagnostico`)

);

**DROP** **TABLE** **IF** **EXISTS** `empleados`;

**CREATE** **TABLE** `empleados` (

`cuit` **varchar**(11) **NOT** **NULL**,

`nombre` **varchar**(50) **NOT** **NULL**,

`fechaIngreso` datetime **NOT** **NULL**,

`idFarmacia` **int** **NOT** **NULL**,

**PRIMARY** **KEY** (`cuit`),

**KEY** `idFarmacia` (`idFarmacia`),

**CONSTRAINT** `empleados\_ibfk\_1` **FOREIGN** **KEY** (`idFarmacia`) **REFERENCES** `farmacias` (`id`) **ON** **DELETE** **CASCADE** **ON** **UPDATE** **CASCADE**

);

**DROP** **TABLE** **IF** **EXISTS** `estadosingresos`;

**CREATE** **TABLE** `estadosingresos` (

`nombre` **varchar**(80) **NOT** **NULL**,

**PRIMARY** **KEY** (`nombre`)

);

**DROP** **TABLE** **IF** **EXISTS** `estadostransferencias`;

**CREATE** **TABLE** `estadostransferencias` (

`nombre` **varchar**(30) **NOT** **NULL**,

**PRIMARY** **KEY** (`nombre`)

);

**DROP** **TABLE** **IF** **EXISTS** `transportadoras`;

**CREATE** **TABLE** `transportadoras` (

`codTransportadora` **int** **NOT** **NULL**,

`cuit` **varchar**(11) **NOT** **NULL**,

`email` **varchar**(40) **NOT** **NULL**,

`telefono` **varchar**(14) **DEFAULT** **NULL**,

`domicilio` **varchar**(50) **DEFAULT** **NULL**,

**PRIMARY** **KEY** (`codTransportadora`)

);

**DROP** **TABLE** **IF** **EXISTS** `ingresos`;

**CREATE** **TABLE** `ingresos` (

`codIngreso` **int** **NOT** **NULL**,

`fechaIngreso` datetime **NOT** **NULL**,

`estado` **varchar**(80) **NOT** **NULL**,

`cuitLab` **varchar**(11) **NOT** **NULL**,

`codTransp` **int** **NOT** **NULL**,

**PRIMARY** **KEY** (`codIngreso`),

**KEY** `estado` (`estado`),

**KEY** `cuitLab` (`cuitLab`),

**KEY** `codTransp` (`codTransp`),

**CONSTRAINT** `ingresos\_ibfk\_1` **FOREIGN** **KEY** (`estado`) **REFERENCES** `estadosingresos` (`nombre`),

**CONSTRAINT** `ingresos\_ibfk\_2` **FOREIGN** **KEY** (`cuitLab`) **REFERENCES** `laboratorios` (`cuit`),

**CONSTRAINT** `ingresos\_ibfk\_3` **FOREIGN** **KEY** (`codTransp`) **REFERENCES** `transportadoras` (`codTransportadora`)

);

**DROP** **TABLE** **IF** **EXISTS** `ingresan`;

**CREATE** **TABLE** `ingresan` (

`codIngreso` **int** **NOT** **NULL**,

`codMed` **int** **NOT** **NULL**,

`cantidad` **int** **NOT** **NULL**,

**PRIMARY** **KEY** (`codIngreso`,`codMed`),

**KEY** `codMed` (`codMed`),

**CONSTRAINT** `ingresan\_ibfk\_1` **FOREIGN** **KEY** (`codMed`) **REFERENCES** `medicamentos` (`codMedicamento`) **ON** **DELETE** **CASCADE** **ON** **UPDATE** **CASCADE**,

**CONSTRAINT** `ingresan\_ibfk\_2` **FOREIGN** **KEY** (`codIngreso`) **REFERENCES** `ingresos` (`codIngreso`) **ON** **DELETE** **CASCADE** **ON** **UPDATE** **CASCADE**

);

**DROP** **TABLE** **IF** **EXISTS** `monodrogas`;

**CREATE** **TABLE** `monodrogas` (

`id` **int** **NOT** **NULL**,

`nombreCientifico` **varchar**(70) **DEFAULT** **NULL**,

`nombreComercial` **varchar**(70) **NOT** **NULL**,

**PRIMARY** **KEY** (`id`)

);

**DROP** **TABLE** **IF** **EXISTS** `compuestospor`;

**CREATE** **TABLE** `compuestospor` (

`idMono` **int** **NOT** **NULL**,

`codMed` **int** **NOT** **NULL**,

**PRIMARY** **KEY** (`codMed`,`idMono`),

**KEY** `idMono` (`idMono`),

**CONSTRAINT** `compuestospor\_ibfk\_1` **FOREIGN** **KEY** (`idMono`) **REFERENCES** `monodrogas` (`id`) **ON** **DELETE** **CASCADE** **ON** **UPDATE** **CASCADE**,

**CONSTRAINT** `compuestospor\_ibfk\_2` **FOREIGN** **KEY** (`codMed`) **REFERENCES** `medicamentos` (`codMedicamento`) **ON** **DELETE** **CASCADE** **ON** **UPDATE** **CASCADE**

);

**DROP** **TABLE** **IF** **EXISTS** `poseen`;

**CREATE** **TABLE** `poseen` (

`codDiagnostico` **varchar**(10) **NOT** **NULL**,

`idAfiliado` **int** **NOT** **NULL**,

`fechaDiagnostico` datetime **NOT** **NULL**,

**PRIMARY** **KEY** (`idAfiliado`,`codDiagnostico`),

**KEY** `codDiagnostico` (`codDiagnostico`),

**CONSTRAINT** `poseen\_ibfk\_1` **FOREIGN** **KEY** (`idAfiliado`) **REFERENCES** `afiliados` (`id`) **ON** **DELETE** **CASCADE** **ON** **UPDATE** **CASCADE**,

**CONSTRAINT** `poseen\_ibfk\_2` **FOREIGN** **KEY** (`codDiagnostico`) **REFERENCES** `diagnosticos` (`codDiagnostico`) **ON** **DELETE** **CASCADE** **ON** **UPDATE** **CASCADE**

);

**DROP** **TABLE** **IF** **EXISTS** `prestan`;

**CREATE** **TABLE** `prestan` (

`codMed` **int** **NOT** **NULL**,

`nombreAccion` **varchar**(80) **NOT** **NULL**,

**PRIMARY** **KEY** (`codMed`,`nombreAccion`),

**KEY** `nombreAccion` (`nombreAccion`),

**CONSTRAINT** `prestan\_ibfk\_1` **FOREIGN** **KEY** (`codMed`) **REFERENCES** `medicamentos` (`codMedicamento`) **ON** **DELETE** **CASCADE** **ON** **UPDATE** **CASCADE**,

**CONSTRAINT** `prestan\_ibfk\_2` **FOREIGN** **KEY** (`nombreAccion`) **REFERENCES** `accionesterapeuticas` (`nombre`) **ON** **DELETE** **CASCADE** **ON** **UPDATE** **CASCADE**

);

**DROP** **TABLE** **IF** **EXISTS** `stockean`;

**CREATE** **TABLE** `stockean` (

`idF` **int** **NOT** **NULL**,

`codMed` **int** **NOT** **NULL**,

`stock` **int** **NOT** **NULL**,

**PRIMARY** **KEY** (`idF`,`codMed`),

**KEY** `codMed` (`codMed`),

**CONSTRAINT** `stockean\_ibfk\_1` **FOREIGN** **KEY** (`idF`) **REFERENCES** `farmacias` (`id`) **ON** **DELETE** **CASCADE** **ON** **UPDATE** **CASCADE**,

**CONSTRAINT** `stockean\_ibfk\_2` **FOREIGN** **KEY** (`codMed`) **REFERENCES** `medicamentos` (`codMedicamento`) **ON** **DELETE** **CASCADE** **ON** **UPDATE** **CASCADE**

);

**DROP** **TABLE** **IF** **EXISTS** `transferencias`;

**CREATE** **TABLE** `transferencias` (

`nro` **int** **NOT** **NULL**,

`fechaConfeccion` datetime **NOT** **NULL**,

`estado` **varchar**(30) **NOT** **NULL**,

`idF` **int** **DEFAULT** **NULL**,

**PRIMARY** **KEY** (`nro`),

**KEY** `estado` (`estado`),

**KEY** `idF` (`idF`),

**CONSTRAINT** `transferencias\_ibfk\_1` **FOREIGN** **KEY** (`estado`) **REFERENCES** `estadostransferencias` (`nombre`),

**CONSTRAINT** `transferencias\_ibfk\_2` **FOREIGN** **KEY** (`idF`) **REFERENCES** `farmacias` (`id`) **ON** **DELETE** **CASCADE** **ON** **UPDATE** **CASCADE**

);

**DROP** **TABLE** **IF** **EXISTS** `transfieren`;

**CREATE** **TABLE** `transfieren` (

`nroTransf` **int** **NOT** **NULL**,

`codMed` **int** **NOT** **NULL**,

`cantidad` **int** **NOT** **NULL**,

**PRIMARY** **KEY** (`nroTransf`,`codMed`),

**KEY** `codMed` (`codMed`),

**CONSTRAINT** `transfieren\_ibfk\_1` **FOREIGN** **KEY** (`nroTransf`) **REFERENCES** `transferencias` (`nro`) **ON** **DELETE** **CASCADE** **ON** **UPDATE** **CASCADE**,

**CONSTRAINT** `transfieren\_ibfk\_2` **FOREIGN** **KEY** (`codMed`) **REFERENCES** `medicamentos` (`codMedicamento`) **ON** **DELETE** **CASCADE** **ON** **UPDATE** **CASCADE**

);

**DROP** **TABLE** **IF** **EXISTS** `alcanzan`;

**CREATE** **TABLE** `alcanzan` (

`codTransp` **int** **NOT** **NULL**,

`codPostal` **int** **NOT** **NULL**,

**PRIMARY** **KEY** (`codTransp`,`codPostal`),

**KEY** `codPostal` (`codPostal`),

**CONSTRAINT** `alcanzan\_ibfk\_1` **FOREIGN** **KEY** (`codTransp`) **REFERENCES** `transportadoras` (`codTransportadora`) **ON** **DELETE** **CASCADE** **ON** **UPDATE** **CASCADE**,

**CONSTRAINT** `alcanzan\_ibfk\_2` **FOREIGN** **KEY** (`codPostal`) **REFERENCES** `ciudades` (`codPostal`) **ON** **DELETE** **CASCADE** **ON** **UPDATE** **CASCADE**

);

**DROP** **TABLE** **IF** **EXISTS** `contienen`;

**CREATE** **TABLE** `contienen` (

`nroComprobante` **int** **NOT** **NULL**,

`codMed` **int** **NOT** **NULL**,

`cantidad` **int** **NOT** **NULL**,

`descuento` **float** **DEFAULT** **NULL**,

`precioUnitario` **float** **NOT** **NULL**,

**PRIMARY** **KEY** (`nroComprobante`,`codMed`),

**KEY** `codMed` (`codMed`),

**CONSTRAINT** `contienen\_ibfk\_1` **FOREIGN** **KEY** (`nroComprobante`) **REFERENCES** `comprobantes` (`nro`) **ON** **DELETE** **CASCADE** **ON** **UPDATE** **CASCADE**,

**CONSTRAINT** `contienen\_ibfk\_2` **FOREIGN** **KEY** (`codMed`) **REFERENCES** `medicamentos` (`codMedicamento`) **ON** **DELETE** **CASCADE** **ON** **UPDATE** **CASCADE**

);

# Generación de datos

Para la generación de datos, utilizamos una herramienta llamada Advanced Data Generator, para MySQL. Se puede descargar gratuitamente en el siguiente link: <https://www.upscene.com/downloads/#adg>

Observamos que la herramienta, si bien fue la única que encontramos que nos permita generar entre 10 mil y 100 mil registros por tabla de manera gratuita y fácil, no ofrece mucha flexibilidad en los valores que genera, por lo que los datos insertados no son tan específicos ni regionalizados como se quisiera. Debido a esto, al dump.sql generado le debimos alterar "a mano" algunos datos, por ejemplo para tener una ciudad de nombre Resistencia.

# Consultas SQL

Realizamos las consultas en el motor de base de datos MySQL versión 8.0, con la ayuda de la GUI MySQL Workbench. Se seleccionó este motor porque su extensa documentación y porque la mayoría de los integrantes del grupo ya estábamos familiarizados con él.

Algunas consultas DELETE, UPDATE, e INSERT propuestas por nosotros facilitan datos que luego son leídos por las consultas SELECT, propuestas por la cátedra. Particularmente las que afectan al medicamento 'Amoxidal Dúo' o a la ciudad 'Resistencia'. Si se desea ejecutarlas, sugerimos ejecutarlas en el orden que las presentamos en este informe: DELETE -> UPDATE -> INSERT -> SELECT.

**Consultas DELETE**

USE farmaciasdb;

/\*

\* 1. Se eliminan los farmacéuticos que no están asignados a ninguna farmacia.

\*/

**DELETE** **FROM** farmaceuticos

**WHERE** dni **NOT** **IN** (**SELECT** dniFarmaceutico **FROM** ciudades);

/\*

\* 2. Se elimina toda transferencia pendiente que se solicitó hace más de 3 años.

\*/

**DELETE** **FROM** transferencias

**WHERE** fechaConfeccion >= DATE\_SUB(NOW(), **INTERVAL** 3 **YEAR**)

**AND** estado = "Pendiente";

/\*

\* 3. Se eliminan las transportadoras que nunca transportaron un ingreso y que solo alcanzan una ciudad.

\*/

**DELETE** **FROM** transportadoras

**WHERE** codTransportadora **NOT** **IN** (**SELECT** codTransp **FROM** ingresos)

**AND** codTransportadora = (

**SELECT** a.codTransp **FROM** alcanzan a

**WHERE** a.codTransp = codTransportadora

**GROUP** **BY** a.codTransp

**HAVING** **COUNT**(a.codPostal) = 1);

/\*

\* 4. Se eliminan los afiliados eventuales que se registraron hace más de 5 años y nunca realizaron una compra.

\*/

**DELETE** **FROM** afiliados

**WHERE** id **NOT** **IN** (**SELECT** idAfiliado **FROM** poseen)

**AND** id **NOT** **IN** (**SELECT** idA **FROM** comprobantes)

**AND** fechaIngreso >= DATE\_SUB(NOW(), **INTERVAL** 5 **YEAR**);

/\*

\* 5. Se elimina, de cada farmacia, el stock de los medicamentos que no vendió ningún ejemplar en los últimos 8 años.

\*/

**DELETE** **FROM** stockean

**WHERE** idF **NOT** **IN** (

**SELECT** **DISTINCT** f.id **FROM** farmacias f

**INNER** **JOIN** comprobantes c **ON** c.idF = f.id

**WHERE** c.fechaEmision >= DATE\_SUB(NOW(), **INTERVAL** 8 **YEAR**)

);

/\*

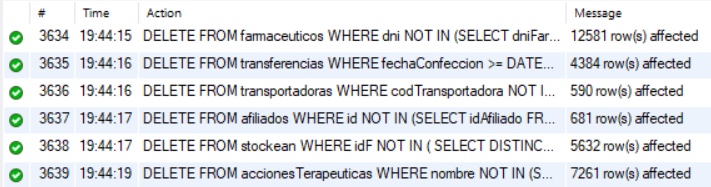
\* 6. Se eliminan las acciones terapéuticas que no tengan ningún medicamento asociado.

\*/

**DELETE** **FROM** accionesTerapeuticas

**WHERE** nombre **NOT** **IN** (**SELECT** nombreAccion **FROM** prestan);

Registros afectados:



**Consultas UPDATE**

**USE** farmaciasdb;

/\*

\* 1. Se establecen los porcentajes de descuento de los afiliados, dependiendo de si son crónicos o no.

\* A los crónicos les corresponde 70%, a los no crónicos les corresponde 20%.

\* NOTA: En nuestro esquema, un afiliado es crónico si su id está en la tabla 'poseen' un diagnóstico.

\*/

**UPDATE** afiliados a

**SET** porcDescuento = (

**CASE**

**WHEN** (**SELECT** **DISTINCT** p.idAfiliado

**FROM** poseen p

**WHERE** a.id = p.idAfiliado) **IS** **NULL** **THEN** 20

**ELSE** 70

**END**);

/\*

\* 2. Actualización del porcentaje de descuento a 80% para los afiliados crónicos que tienen más de 5 años de antigüedad.

\*/

**UPDATE** afiliados a

**SET** a.porcDescuento = 80

**WHERE** a.fechaIngreso < **DATE\_SUB**(**CURDATE**(), **INTERVAL** 5 **YEAR**)

**AND** a.id **IN** (

**SELECT** p.idAfiliado

**FROM** poseen p

**WHERE** p.idAfiliado = a.id

);

/\*

\* 3. Se incrementa un 10% el precio de todos los medicamentos con stock en el depósito central.

\*/

**WITH** conStock **AS** (**SELECT** codMedicamento **FROM** medicamentos **WHERE** stockCentral > 0)

**UPDATE** medicamentos

**SET** precio = precio \* 1.10

**WHERE** codMedicamento **IN** (**SELECT** codMedicamento **FROM** conStock);

/\*

\* 4. Se incrementa en 10 unidades el stock de un medicamento en cada farmacia con stock inferior al promedio general.

\* NOTA: Parece que en MySQL no se puede hacer un FROM de la tabla que se está UPDATEando. Para evitar ese problema usamos WITH.

\* https://dev.mysql.com/doc/refman/8.0/en/with.html

\*/

**WITH** s2 **AS** (**SELECT** \* **FROM** stockean)

**UPDATE** stockean s1

**SET** s1.stock = s1.stock + 10

**WHERE** s1.stock < (

**SELECT** **AVG**(s2.stock) **FROM** s2

**WHERE** s1.codMed = s2.codMed

**GROUP** **BY** codMed);

/\*

\* 5. Cambio del estado de las transferencias a "EnProceso" si su estado actual es "Pendiente" y

\* participa en la tabla 'transfieren', lo que indica que una transportadora ya la llevó o está llevando.

\*/

**UPDATE** transferencias

**SET** estado = 'EnProceso'

**WHERE** estado = 'Pendiente'

**AND** nro **IN** (**SELECT** nroTransf **FROM** transfieren);

/\*

\* 6. Reducción del stock central en base a las transferencias requeridas en la última semana.

\*/

**UPDATE** medicamentos m

**SET** m.stockCentral = m.stockCentral - **IFNULL**((

**SELECT** **SUM**(tn.cantidad) **FROM** transferencias ts

**INNER** **JOIN** transfieren tn **ON** tn.nroTransf = ts.nro

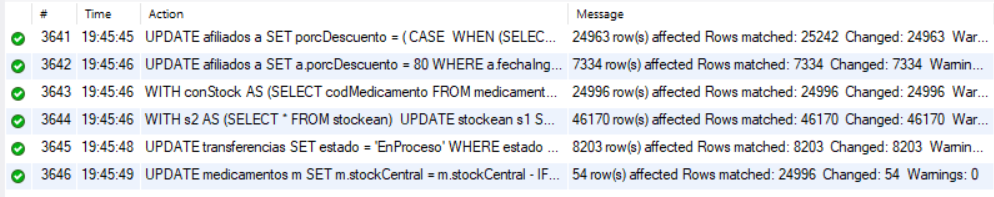
**WHERE** m.codMedicamento = tn.codMed

**AND** ts.fechaConfeccion > **DATE\_SUB**(**CURDATE**(), **INTERVAL** 1 **WEEK**)

**GROUP** **BY** tn.codMed

), 0);

Registros afectados:



**Consultas INSERT**

USE farmaciasdb;

/\*

\* 1. Al comprobante más reciente de cada farmacia, que no contenga una venta del medicamento "Amoxidal Dúo",

\* se le agrega una unidad vendida de ese medicamento.

\* NOTA: si una farmacia ya tiene todos sus comprobantes con Amoxidal Dúo, entonces no se le agrega nada.

\*/

**INSERT** **INTO** contienen(nroComprobante, cantidad, descuento, codMed, precioUnitario)

**WITH** amoxidal **AS** (**SELECT** codMedicamento, precio **FROM** medicamentos **WHERE** descripcion = "Amoxidal Dúo")

**SELECT** compr.nro, 1,

(**SELECT** porcDescuento **FROM** afiliados **WHERE** id = compr.idA),

(**SELECT** codMedicamento **FROM** amoxidal),

(**SELECT** precio **FROM** amoxidal)

**FROM** farmacias f

**INNER** **JOIN** comprobantes compr **ON** f.id = compr.idF

**LEFT** **JOIN** contienen cont **ON** compr.nro = cont.nroComprobante

**WHERE** compr.nro **NOT** **IN** ( -- El comprobante no debe contener el medicamento

**SELECT** nroComprobante **FROM** contienen **WHERE** codMed = (**SELECT** codMedicamento **FROM** amoxidal)

)

**GROUP** **BY** f.id

**ORDER** **BY** compr.fechaEmision **DESC**;

/\*

2. Al medicamento con presentación (nombre) 'Obenauf' se le agrega una monodroga con nombre científico 'CT').

\*/

**INSERT** **INTO** compuestospor(idMono, codMed)

**VALUES** (

(**SELECT** id **FROM** monodrogas **WHERE** nombreCientifico = 'CT' **LIMIT** 1),

(**SELECT** codMedicamento **FROM** medicamentos **WHERE** presentacion = "Obenauf")

) ;

/\*

3. Se inserta una transferencia con fecha actual y estado 'Pendiente' hacia una farmacia de Resistencia.

\*/

**INSERT** **INTO** transferencias(nro, fechaConfeccion, estado, idF)

**VALUES**(1, now(), "Pendiente", (**SELECT** id **FROM** farmacias **WHERE** codPostal = 3500 **LIMIT** 1));

/\*

4. Se inserta que la última transferencia creada requiere 5 unidades de Amoxidal Dúo.

\*/

**INSERT** **INTO** transfieren(nroTransf, codMed, cantidad) **VALUES**(

(**SELECT** nro **FROM** transferencias **WHERE** fechaConfeccion = (

**SELECT** **MAX**(fechaConfeccion) **FROM** transferencias)

),

(**SELECT** codMedicamento **FROM** medicamentos **WHERE** descripcion = "Amoxidal Dúo"),

5);

/\*

5. Se inserta una compra en una farmacia de la ciudad Resistencia, del afiliado con id 68.

\*/

**INSERT** **INTO** comprobantes(nro, idA, idF, fechaEmision) **VALUES** (

2,

68,

(**SELECT** id **FROM** farmacias **WHERE** codPostal = 3500 **LIMIT** 1),

now());

-- A ese comprobante insertado se le agregan los medicamentos que se compraron.

**INSERT** **INTO** contienen(nroComprobante, codMed, cantidad, descuento, precioUnitario)

**VALUES** (

2,

831,

2,

(**SELECT** porcDescuento **FROM** afiliados **WHERE** id = 68),

(**SELECT** precio **FROM** medicamentos **WHERE** codMedicamento = 831)

);

/\*

6. A una transportadora que tenga menos de 10 destinos se le inserta un nuevo alcance a Resistencia.

\*/

**INSERT** **INTO** alcanzan(codTransp, codPostal) **VALUES**(

(**SELECT** a.codTransp **FROM** alcanzan a

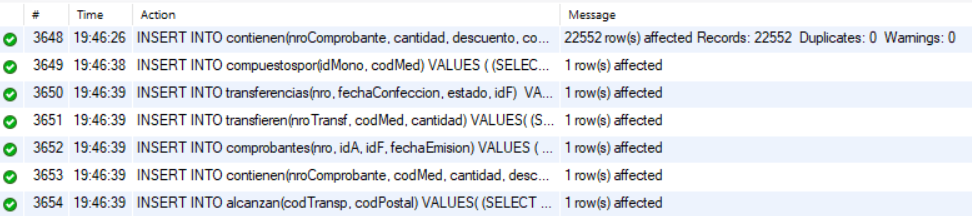
**GROUP** **BY** a.codTransp

**HAVING** **count**(a.codPostal) < 10

**LIMIT** 1),

3500);

Registros afectados:



**Consultas SELECT**

USE farmaciasdb;

/\*

\* 1. Mostrar el ranking de los diez medicamentos con mayor cantidad de ventas en todas las farmacias de la cadena.

\*/

-- NOTA: Entendemos que el ranking es de la cantidad total de ejemplares vendidos.

**SELECT** m.descripcion, **SUM**(c.cantidad) **AS** unidades\_vendidas **FROM** medicamentos m

**INNER** **JOIN** contienen c **ON** c.codMed = m.codMedicamento

**GROUP** **BY** c.codMed

**ORDER** **BY** **SUM**(c.cantidad) **DESC**

**LIMIT** 10;



/\*

\* 2. Listar los códigos y nombres de los medicamentos que fueron vendidos en todas las farmacias.

\*/

-- Nota: hacemos la división con GROUP BY y HAVING porque la técnica que usa dos subconsultas nos daba un timeout.

**SELECT** m.codMedicamento, m.descripcion **FROM** medicamentos m

**INNER** **JOIN** contienen cont **ON** m.codMedicamento = cont.codMed

**INNER** **JOIN** comprobantes compr **ON** cont.nroComprobante = compr.nro

**GROUP** **BY** m.codMedicamento

**HAVING** **count**(**DISTINCT** compr.idF) = (**SELECT** **count**(\*) **FROM** farmacias);



/\*

\* 3. Mostrar cantidad de afiliados crónicos y eventuales que compraron medicamentos en la farmacia de Resistencia en la última semana.

\*/

-- compradores es una tabla virtual con el 'id' del afiliado y 'idAfiliado' que es la FK de la tabla poseen.

-- Si 'idAfiliado' es NULL, entonces no tiene ningún diagnóstico (y es un afiliado eventual).

**WITH** compradores **AS** (

**SELECT** **DISTINCT** a.id, p.idAfiliado **FROM** afiliados a

**LEFT** **JOIN** poseen p **ON** p.idAfiliado = a.id

**INNER** **JOIN** comprobantes c **ON** c.idA = a.id

**WHERE** c.idF **IN** (**SELECT** id **FROM** farmacias **WHERE** codPostal = 3500)

**AND** 7 > DATEDIFF(now(), c.fechaEmision)

)

**SELECT**

(**SELECT** **count**(c.id) **FROM** compradores c **WHERE** c.idAfiliado **IS** **NOT** **NULL**) **AS** crónicos,

(**SELECT** **count**(c.id) **FROM** compradores c **WHERE** c.idAfiliado **IS** **NULL**) **AS** eventuales;



/\*

\* 4. Informar el top de las 5 farmacias que solicitaron mayores cantidades de amoxidal duo en los últimos 15 dias.

\*/

**SELECT** f.id, f.direccion, **SUM**(transfieren.cantidad) **AS** total\_solicitado **FROM** farmacias f

**INNER** **JOIN** transferencias t **ON** f.id = t.idF

**INNER** **JOIN** transfieren **ON** transfieren.nroTransf = t.nro

**INNER** **JOIN** medicamentos m **ON** transfieren.codMed = m.codMedicamento

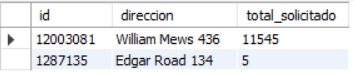
**WHERE** m.descripcion = "Amoxidal Dúo"

**AND** t.fechaConfeccion >= DATE\_SUB(CURDATE(), **INTERVAL** 15 **DAY**)

**GROUP** **BY** f.id

**ORDER** **BY** **SUM**(transfieren.cantidad) **DESC**

**LIMIT** 5;



/\*

\* 5. Listado de farmacias con la menor cantidad de transferencias pendientes.

\*/

**SELECT** f.id, f.direccion, **COUNT**(t.nro) **AS** transferencias\_pendientes **FROM** farmacias f

**INNER** **JOIN** transferencias t **ON** f.id = t.idF

**WHERE** t.estado = "Pendiente"

**GROUP** **BY** f.id

**HAVING** **COUNT**(t.nro) = (

**SELECT** **MIN**(cant) **FROM** (

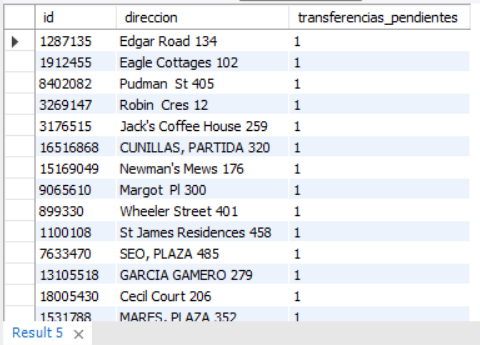
**SELECT** **COUNT**(t.nro) cant **FROM** farmacias f

**INNER** **JOIN** transferencias t **ON** f.id = t.idF

**GROUP** **BY** f.id

) **AS** temp

);

 (1889 filas)

/\*

\* 6. Identificar para una farmacia determinada cuales son los medicamentos sin stock en la misma pero con stock en deposito central.

\*/

-- Nota: el literal 153 es el 'id' de la farmacia determinada, podría ser cualquiera.

-- Si para algún codMedicamento no existe una fila 'stockean' con la farmacia, entonces también está sin stock.

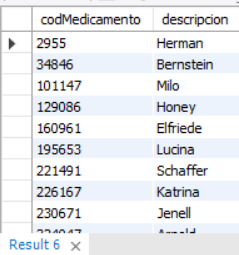
**SELECT** m.codMedicamento, m.descripcion **FROM** medicamentos m

**LEFT** **JOIN** stockean s **ON** m.codMedicamento = s.codMed

**WHERE** (s.idF = 153 **OR** s.idF **IS** **NULL**)

**AND** (s.stock = 0 **OR** s.idF **IS** **NULL**)

**AND** m.stockCentral > 0;

 (696 filas)

/\*

\* 7. ¿Cuál es la empresa de transporte con mayor actividad en el último mes?

\*/

-- Consideramos que la de mayor actividad es la que más ingresos realizó.

-- Como pueden ser varias, traemos todas las que tienen la mayor cantidad de ingresos realizados.

**SELECT** t.cuit, **COUNT**(i.codIngreso) **AS** cantidad\_ingresos **FROM** transportadoras t

**INNER** **JOIN** ingresos i **ON** t.codTransportadora = i.codTransp

**WHERE** i.fechaIngreso >= DATE\_SUB(CURDATE(), **INTERVAL** 1 **MONTH**)

**GROUP** **BY** i.codTransp

**HAVING** **COUNT**(i.codIngreso) = (

**SELECT** **MAX**(cant) **FROM** (

**SELECT** **COUNT**(codIngreso) cant **FROM** ingresos

**WHERE** fechaIngreso >= DATE\_SUB(CURDATE(), **INTERVAL** 1 **MONTH**)

**GROUP** **BY** codTransp

) **AS** temp

);



/\*

\* 8. Informar el monto total de ventas por farmacia en el último trimestre ordenado en forma descendente.

\*/

**SELECT** f.id, **SUM**(c.precioUnitario \* c.cantidad \* (100 - c.descuento) / 100) **AS** total\_ventas **FROM** farmacias f

**INNER** **JOIN** comprobantes co **ON** f.id = co.idF

**INNER** **JOIN** contienen c **ON** co.nro = c.nroComprobante

**WHERE** co.fechaEmision >= DATE\_SUB(CURDATE(), **INTERVAL** 3 **MONTH**)

**GROUP** **BY** f.id

**ORDER** **BY** total\_ventas **DESC**;

**WITH** ventas\_trimestre **AS** (

**SELECT** f.id, **SUM**(c.precioUnitario \* c.cantidad \* (100 - c.descuento) / 100) **AS** total\_ventas

**FROM** farmacias f

**INNER** **JOIN** comprobantes co **ON** f.id = co.idF

**INNER** **JOIN** contienen c **ON** co.nro = c.nroComprobante

**WHERE** co.fechaEmision >= DATE\_SUB(CURDATE(), **INTERVAL** 3 **MONTH**)

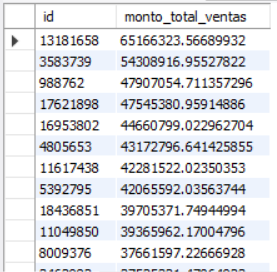
**GROUP** **BY** f.id)

**SELECT** f.id,

IFNULL((**SELECT** total\_ventas **FROM** ventas\_trimestre **WHERE** id = f.id), 0) **AS** monto\_total\_ventas

**FROM** farmacias f

**ORDER** **BY** monto\_total\_ventas **DESC**;

(22553 filas, una por farmacia)